

УНИВЕРЗИТЕТ ПРИВРЕДНА АКАДЕМИЈА У НОВОМ
САДУ

ФАКУЛТЕТ ЗА ЕКОНОМИЈУ И ИНЖЕЊЕРСКИ
МЕНАѢМЕНТ У НОВОМ САДУ

Перспектива наслеђених информационих
система - Студија случаја на моделу ЈКП
„Београдски водовод и канализација“

- докторска дисертација -

Ментор:
Проф. др Марко Царић

Кандидат:
Славимир Љ. Весић

Нови Сад, 2020

UNIVERSITY BUSINESS ACADEMY IN NOVI SAD
FACULTY OF ECONOMICS AND ENGINEERING
MANAGEMENT IN NOVI SAD

A Perspective of Legacy Information Systems -
A Case Study of the PUC „Belgrade
Waterworks and Sewerage“ Model

- doctoral dissertation -

MENTOR:
Prof. dr Marko Carić

CANDIDATE:
Slavimir Lj. Vesić

Novi Sad, 2020

| | |
|--|---|
| Врста рада: | Докторска дисертација |
| Име и презиме аутора: | Славимир Љ. Весић |
| Ментор (титула, име, презиме, звање, институција) | Проф. др Марко Царић, редовни професор Правног факултета за привреду и правосуђе Универзитета Привредна академија у Новом Саду, ужа научна област: Мултидисциплинарна економска. |
| Наслов рада: | Перспектива наслеђених информационих система - Студија случаја на моделу ЈКП „Београдски водовод и канализација“ |
| Језик публикације (писмо): | Српски језик (ћирилица) |
| Физички опис рада: | Унети број: Страница 402 Поглавља 9 Референци 304 Табела 46 Слика 64 |
| Научна област: | Информациони системи |
| Предметна одредница, кључне речи: | информациони системи, наслеђени системи, перспективе наслеђених система, модернизација система, одржавање софтвера, еволуција софтвера, старење софтвера |
| Извод (апстракт или резиме) на језику завршног рада: | Након стављања информационог система у употребу започиње процес старења софтвера, који временом доводи до нарушавања његове структуре па се спроводи одржавање, које има за циљ да уклони одређене грешке и побољша његове перформансе. Како информациони систем има потребу да еволуира заједно са окружењем у којем функционише, тако процеси одржавања такође имају за циљ и да прилагоде систем да буде у складу са променама. У једном тренутку потребно је да се спроведу промене јачег интензитета него што је одржавање, познатије као модернизација да би се повратила еволутивна способност информационог система. Наслеђени системи се одликују смањеном |

еволутивном способношћу, дуго су употреби и тешки за промену, најчешће су настали у старим технологијама, при чему је процес њиховог одржавања тежак за све оне који га одржавају и развијају, а наведени процес је истовремено скуп за предузеће. Са једне стране они ограничавају пословање, јер не дозвољавају једноставне измене, док са друге садрже вредна вишедесетничка знања и пословна правила која су формализована и на које се предузеће ослања приликом обављања дневних операција. У теорији постоји велики број дефиниција наслеђених система, улажу се огромна средства у њихово одржавање и постоји велики број неуспелих пројекта модернизације и поред значајних инвестиција. Наведене чињенице указују да су наслеђени системи комплексан феномен, недовољно схваћен, а да је велики број неуспелих пројеката модернизације у директној вези са њиховим неразумевањем, односно њиховим изолованим посматрањем. Дисертација има циљ да сагледа наслеђене информационе системе свеобухватно из различитих перспектива: развојне, оперативне, организационе и стратешке. Такав начин посматрања наслеђених система води ка њиховом бољем схватању, као и могућности да се процес модернизације усмери у одређеном правцу. Дат је теоријски оквир који инкорпорира све напред наведене перспективе, а уједно је и презентован кроз студију случаја ЈКП „Београдски водовод и канализација“, а након тога евалуиран на водоводним предузећима у Србији и региону. Резултати истраживања су потврдили да оквир може да се користи као алат за евалуацију наслеђених система. На основи свих резултата, дат је предлог модернизационе стратегије која треба да реши

| | |
|---|---|
| | проблеме наслеђених система идентификованих у свим перспективама на случају ЈКП „Београдски водовод и канализација“. |
| Датум одбране (Попуњава накнадно одговарајућа служба) | |
| Чланови комисије: (титула, име, презиме, звање, институција) | <p>Председник: Проф. др Драган Солеша, редовни професор Факултета за економију и инжењерски менаџмент Универзитета Привредна академија у Новом Саду, ужа научна област: Квантитативне методе и информациони системи.</p> <p>Ментор: Проф. др Марко Царић, редовни професор Правног факултета за привреду и правосуђе Универзитета Привредна академија у Новом Саду, ужа научна област: Мултидисциплинарна економска.</p> <p>Члан: Проф. др Петар Спалевић, редовни професор на Факултету техничких наука, Универзитета у Приштини са привременим седиштем у Косовској Митровици и на Универзитету Сингидунум у Београду, ужа научна област: Рачунарство и информатика и Телекомуникације и информациони системи</p> |
| Напомена: | <p>Аутор докторске дисертације потписао је следеће Изјаве:</p> <ol style="list-style-type: none"> 1. Изјава о ауторству 2. Изјава о истоветности штампане и електронске верзије докторског рада и 3. Изјава о коришћењу. <p>Ове изјаве се чувају на факултету у штампаном и електронском облику.</p> |
| УДК: | |

| | |
|--|--|
| Document type: | Doctoral dissertation |
| Author: | Slavimir Lj. Vesić |
| Mentor (title, first name, last name, position, institution) | Marko Carić, Ph.D., Full Professor at the Faculty of Law for Commerce and Judiciary in Novi Sad, University Business Academy in Novi Sad, scientific area: Multidisciplinary Economics. |
| Title: | A Perspective of Legacy Information Systems - A Case Study of the PUC „Belgrade Waterworks and Sewerage“ Model |
| Language of text (script): | Serbian language (cyrilic) |
| Physical description: | Number of: Pages 402 Chapters 9 References 304 Tables 46 Illustrations 64 |
| Scientific field: | Information systems |
| Subjectm Key words: | information systems, legacy systems, legacy systems perspectives, system modernization, software maintenance, software evolution, software aging |
| Abstract (or resume) in the language if the text: | After putting the information system into use, the aging process of the software begins, which over time leads to the disruption of its structure, so maintenance is carried out, which aims to eliminate certain errors and improve its performance. As the information system needs to evolve along with the environment in which it operates, so maintenance processes also aim to adapt the system to change. At some point, changes of greater intensity than maintenance, better known as modernization, are needed to restore the evolutionary capability of the information system. Legacy systems are characterized by reduced evolutionary ability, are long in use and difficult to change, most often originated in old technologies, and the process of their maintenance is difficult for all those who maintain and develop it, and |

| | |
|--|--|
| | <p>this process is expensive for the company. On the one hand, they limit business, because they do not allow simple changes, while on the other hand, they contain valuable decades of knowledge and business rules that are formalized and that the company relies on when performing daily operations. In theory, there are a large number of definitions of legacy systems, huge funds are invested in their maintenance and there are a large number of failed modernization projects despite significant investments. These facts indicate that legacy systems are a complex phenomenon, insufficiently understood and that a large number of failed modernization projects are directly related to their misunderstanding, ie their isolated observation. The dissertation aims to look at legacy information systems comprehensively from different perspectives: developmental, operational, organizational and strategic. Such a way of observing legacy systems leads to a better understanding of them, as well as the possibility of directing the process of modernization. A theoretical framework is given that incorporates all the above perspectives, and is also presented through a case study of PUC "Belgrade Waterworks and Sewerage", and then evaluated at water companies in Serbia and the region. The results of the research confirmed that the framework can be used as a tool for the evaluation of inherited systems. Based on all the results, a proposal for a modernization strategy is given, which should solve the problems of legacy systems identified in all perspectives in the case of PUC "Belgrade Waterworks and Sewerage".</p> |
| <p>Defended: (The faculty service fills later.)</p> | |
| <p>Thesis Defend Board: (title, first name, last name,</p> | <p>President: Dragan Soleša, Ph.D., Full Professor at the Faculty of Economics and Engineering Management in</p> |

| | |
|------------------------|--|
| position, institution) | <p>Novi Sad, University Business Academy in Novi Sad, scientific area: Quantitative Methods and Information Systems.</p> <p>Menthor: Marko Carić, Ph.D., Full Professor at the Faculty of Law for Commerce and Judiciary in Novi Sad, University Business Academy in Novi Sad, scientific area: Multidisciplinary Economics.</p> <p>Member of the Commission: Prof. dr Petar Spalević, Ph.D., Full Professor at the Faculty of Technical Sciences, University of Prishtina with temporary headquarters in Kosovska Mitrovica and Singidunum University in Belgrade, scientific area: Computer Science and Informatics and Telecommunications and Information Systems</p> |
| Note: | <p>The autor of doctoral dissertation has signed the following Statements:</p> <ol style="list-style-type: none"> 1. Statement on the authority. 2. Statement that the printed and e-version od doctoral dissertation are identical and 3. Statement on copyright licenses. <p>The paper and e-versions of Statements are held at the faculty.</p> |
| UDC: | |

Изјаве захвалности

Посвећено успомени на мог покојног оца Љубишу Љубу Весића.

Желим да се захвалим свом стрицу Проф. др Добрици Весићу, који је након преране смрти мога оца преузео бригу о мени, као родитељ, усмерио ме ка путу науке и знања, препознао искру радозналости која је потребна и распирио је.

Желео бих да се захвалим породици за сву пружену љубав и подршку.

Садржај

| | |
|---|-----------|
| 1. УВОД..... | 1 |
| 1.1. ПРОБЛЕМ И ПРЕДМЕТ ИСТРАЖИВАЊА | 4 |
| 1.2. ЦИЉЕВИ ИСТРАЖИВАЊА | 7 |
| 1.3. ОЧЕКИВАНИ РЕЗУЛТАТИ (ХИПОТЕЗЕ) | 8 |
| 1.4. ПЛАН РАДА | 9 |
| 1.5. МЕТОД И УЗОРАК ИСТРАЖИВАЊА | 10 |
| 2. ИНФОРМАЦИОНЕ ТЕХНОЛОГИЈЕ И ИНФОРМАЦИОНИ СИСТЕМИ..... | 12 |
| 2.1. ИНФОРМАЦИОНЕ ТЕХНОЛОГИЈЕ И ИНФОРМАЦИОНИ СИСТЕМИ | 12 |
| | 14 |
| | 14 |
| 2.1.1. Дефинисање информационих система | 14 |
| 2.1.2. Компоненте и активности информационих система | 18 |
| 2.1.3. Моделовање информационих система и њихов положај у реалном времену ... | 21 |
| 2.1.3.1. Водопад модел..... | 23 |
| 2.1.3.2. V модел..... | 25 |
| 2.1.3.3. Инкрементални модел..... | 26 |
| 2.1.3.4. Прототипски модел | 26 |
| 2.1.3.5. Рапидни модел..... | 27 |
| 2.1.3.6. Агилни модел | 27 |
| 2.1.3.6.1. Scrum | 29 |
| 2.1.3.6.2. Kanban | 29 |
| 2.1.3.6.3. Екстремно програмирање..... | 30 |
| 2.1.4. Ресурси информационих система..... | 32 |
| 2.1.5. Врсте информационих система | 33 |
| 2.1.5.1. Информациони системи за подршку операцијама | 33 |
| 2.1.5.2. Информациони системи за подршку менаџменту..... | 34 |
| 2.2. ИНТЕРАКЦИЈА ИНФОРМАЦИОНИХ СИСТЕМА И ОРГАНИЗАЦИЈЕ | 35 |
| 2.2.1. Информациони системи и нивои управљања у организацијама | 36 |
| 2.2.2. Нове улоге информационих система у организацијама | 37 |
| 2.2.3. Значај информационих технологија за земље у развоју | 39 |
| 2.2.4. Утицај информационих технологија на организационе перформансе | 41 |
| 3. КАРАКТЕРИСТИКЕ САВРЕМЕНОГ ПРОЦЕСА ПРОИЗВОДЊЕ, НАБАВКЕ И | |
| УПОТРЕБЕ СОФТВЕРА..... | 43 |
| 3.1. НАБАВКА СОФТВЕРСКОГ ПРОИЗВОДА | 43 |
| 3.1.1. Уговарање и набавка софтвера..... | 43 |
| 3.1.2. Употреба стандарда при уговарању софтвера..... | 46 |

| | |
|---|-----------|
| 3.1.3. Стандарди у информационим технологијама..... | 46 |
| 3.2. ИНТЕРАКЦИЈА ЧОВЕК-РАЧУНАР (НСИ)..... | 48 |
| 3.2.1. НСИ - појам, дефиниција и циљ..... | 48 |
| 3.2.2. Проблем интеракције човек-рачунар..... | 50 |
| 3.3.3. Инжењеринг употребљивости..... | 50 |
| 3.3. УПОТРЕБЉИВОСТ СОФТВЕРА..... | 52 |
| 3.3.1. Појам употребљивости софтвера..... | 52 |
| 3.3.2. Дефиниције употребљивости..... | 53 |
| 3.3.3. Декомпозиција употребљивости..... | 55 |
| 3.3.4. Модели употребљивости..... | 56 |
| 3.3.5. Стандарди употребљивости..... | 57 |
| 3.3.6. Модели квалитета софтвера..... | 58 |
| 3.4. МЕТОДЕ ЗА ЕВАЛУАЦИЈУ УПОТРЕБЉИВОСТИ СОФТВЕРА..... | 60 |
| 3.4.1. Основни концепти евалуације употребљивости..... | 60 |
| 3.4.1.1. Циљевни евалуације..... | 60 |
| 3.4.1.2. Стилони евалуације..... | 61 |
| 3.4.1.3. Процес евалуације квалитета..... | 62 |
| 3.4.2. Методе и технике за процену употребљивости софтвера..... | 65 |
| 3.4.2.1. Методе прегледања..... | 66 |
| 3.4.2.1.1. Метода когнитивног пролаза..... | 66 |
| 3.4.2.1.2. Хеуристичка евалуација..... | 66 |
| 3.4.2.1.3. Инспекција конзистентности..... | 67 |
| 3.4.2.2. Методе тестирања..... | 68 |
| 3.4.2.2.1. Размишљање наглас..... | 68 |
| 3.4.2.2.2. Метода праћења ока..... | 69 |
| 3.4.2.3. Методе испитивања..... | 69 |
| 3.4.2.3.1. Интервју..... | 69 |
| 3.4.2.3.2. Упитник..... | 70 |
| 3.4.3. Методе за евалуацију квалитета Web апликација..... | 70 |
| 3.4.3.1. WebQEM..... | 71 |
| 3.4.3.2. WAMMI..... | 73 |
| 4. НАСЛЕЂЕНИ СИСТЕМИ..... | 74 |
| 4.1. СТАРЕЊЕ СОФТВЕРА..... | 74 |
| 4.2. ОДРЖАВАЊЕ СОФТВЕРА..... | 77 |
| 4.3. ЕВОЛУЦИЈА СОФТВЕРА..... | 83 |
| 4.4. НАСЛЕЂЕНИ ИНФОРМАЦИОНИ СИСТЕМИ..... | 89 |
| 4.4.1. Појам и дефиниције наслеђених система..... | 89 |
| 4.4.2. Перспективе наслеђених система..... | 98 |
| 4.4.2.1. Развојна перспектива..... | 99 |
| 4.4.2.2. Оперативна перспектива..... | 101 |

| | |
|--|-----|
| 4.4.2.3. Организациона перспектива..... | 103 |
| 4.4.2.4. Стратешка перспектива..... | 103 |
| 4.4.3. Трошкови одржавања постојећих наслеђених система..... | 104 |
| 4.4.4. Ризици и трошкови замене наслеђених система..... | 105 |
| 4.4.5. Актуелност проблема..... | 107 |
| 4.5. НАЧИНИ РЕШАВАЊА ПРОБЛЕМА НАСЛЕЂЕНИХ СИСТЕМА..... | 109 |
| 4.5.1. Софтверски реинжењеринг..... | 110 |
| 4.5.1.1. Софтверски инжењеринг..... | 110 |
| 4.5.1.2. Сврха и циљеви софтверског реинжењеринга..... | 111 |
| 4.5.1.3. Основни појмови софтверског реинжењеринга..... | 115 |
| 4.5.1.3.1. Инжењеринг у напред..... | 117 |
| 4.5.1.3.2. Реверзни инжењеринг..... | 118 |
| 4.5.1.3.3. Редокументација..... | 121 |
| 4.5.1.3.4. Опоравак дизајна..... | 123 |
| 4.5.1.3.5. Реструктурирање..... | 124 |
| 4.5.1.3.6. Софтверски реинжењеринг..... | 124 |
| 4.5.1.4. Традиционални модел софтверског реинжењеринга..... | 125 |
| 4.5.1.5. Аспекти софтверског реинжењеринга..... | 127 |
| 4.5.1.5.1. Економски аспект..... | 127 |
| 4.5.1.5.2. Организациони аспект..... | 128 |
| 4.5.1.5.3. Правни аспект..... | 129 |
| 4.5.1.6. Приступи софтверском реинжењерингу..... | 129 |
| 4.5.2. Облагање..... | 131 |
| 4.5.3. Миграција..... | 133 |
| 4.5.4. Остали приступи..... | 134 |
| 4.5.4.1. Архитектуром вођена модернизација..... | 134 |
| 4.5.4.2. Приступ базиран на микросервисима..... | 136 |

5. СТУДИЈА СЛУЧАЈА НА МОДЕЛУ ЈКП „БЕОГРАДСКИ ВОДОВОД И КАНАЛИЗАЦИЈА“ 137

| | |
|--|-----|
| 5.1. Увод..... | 137 |
| 5.1.1. Опис проблема..... | 137 |
| 5.1.2. Опис случаја..... | 141 |
| 5.1.3. Веза елемената случаја са проблемом и продубљивање проблема..... | 145 |
| 5.1.4. Значај проблема истраживања, погодност студије случаја као методе истраживања и погодност ЈКП БВК као субјекта анализе..... | 146 |
| 5.1.5. Позадина истраживања и резиме претходних истраживања..... | 147 |
| 5.1.6. Начин на који ће студија случаја унапредити ниво знања или пружити нове начине разумевања..... | 148 |
| 5.2. ПРЕГЛЕД РАЗМАТРАЊА..... | 148 |
| 5.2.1. Преглед постојећих разматрања..... | 148 |

| | |
|---|-----|
| 5.2.2. Успостављање новог (проширеног) теоријског оквира | 154 |
| 5.3. МЕТОД | 158 |
| 5.3.1. Посматрање наслеђеног система на случају ЈКП БВК | 159 |
| 5.3.1.1. Развојна перспектива..... | 159 |
| 5.3.1.1.1. Документација није ажурна или не постоји | 166 |
| 5.3.1.1.2. Особље које је радило на развоју система је напустило предузеће..... | 168 |
| 5.3.1.1.3. Новозапослени немају довољно знања и вештина потребних за одржавање система | 168 |
| 5.3.1.1.4. Нарушена почетна архитектурална замисао | 169 |
| 5.3.1.1.5. Редуданса..... | 170 |
| 5.3.1.2. Оперативна перспектива | 171 |
| 5.3.1.2.1. Не постоји подршка произвођача хардвера и/или софтвера | 171 |
| 5.3.1.2.2. Проблеми интеграције система | 172 |
| 5.3.1.3. Организациона перспектива..... | 173 |
| 5.3.1.3.1. Систем ограничава пословање | 173 |
| 5.3.1.3.2. Систем је вредан због подршке главним пословним активностима..... | 174 |
| 5.3.1.4. Стратешка перспектива..... | 175 |
| 5.3.1.4.1. Систем ограничава измену стратегије..... | 175 |
| 5.3.1.4.2. Систем ограничава будућу стратегију..... | 176 |
| 5.3.1.5. Одступања од предложеног теоријског оквира на примеру ЈКП БВК..... | 176 |
| 5.3.2. Анализа ланца вредности на случају ЈКП БВК..... | 179 |
| 5.3.2.1. Ланац вредности водоводних предузећа..... | 180 |
| 5.3.2.2. Ланац вредности ЈКП БВК..... | 181 |
| 5.3.2.2.1. Изворишта воде | 181 |
| 5.3.2.2.2. Дистрибуција сирове воде..... | 181 |
| 5.3.2.2.3. Прерада воде..... | 182 |
| 5.3.2.2.4. Дистрибуција пијеће воде..... | 182 |
| 5.3.2.2.5. Продаја | 183 |
| 5.3.2.2.6. Скупљање канализације | 184 |
| 5.3.2.2.7. Прерада канализације..... | 185 |
| 5.3.2.3. Проблеми и изазови у ланцу вредности ЈКП БВК | 185 |
| 5.3.2.3.1. Проблем контаминације извора..... | 186 |
| 5.3.2.3.2. Проблем воде која не доноси приход..... | 187 |
| 5.3.2.3.3. Мапирање примарних активности у ланцу вредности ЈКП БВК и чиниоца воде која не доноси приход | 193 |
| 5.3.2.3.4. Случај ЈКП БВК - обављање активности које утичу на чиниоце воде која не доноси приход (смањење губитака) | 197 |
| 5.3.2.3.5. Проблем изливања кишне канализације..... | 200 |
| 5.3.2.4. Информационе технологије у решавању проблема и изазова..... | 200 |
| 5.3.2.4.1. Рачунарство у облаку | 200 |
| 5.3.2.4.2. Паметни градови и Интернет ствари..... | 203 |
| 5.3.2.4.3. ARM и AMI технологије | 205 |
| 5.3.2.4.4. Бежичне сензорске мреже | 205 |

| | |
|---|------------|
| 5.3.2.4.5. Велики подаци..... | 206 |
| 5.3.2.4.6. NoSQL базе података..... | 209 |
| 5.3.2.4.7. Детекција аномалија..... | 209 |
| 5.3.2.4.8. Вештачка интелигенција и машинско учење | 210 |
| 5.3.2.5. Примена ИТ у решавању проблема и изазова у ланцу вредности ЈКП..... | 210 |
| 5.3.2.5.1. Могућност примене ИТ на проблему контаминације извора сирове воде..... | 210 |
| 5.3.2.5.2. Могућност примене ИТ на проблему детекцији квара на дистрибутивној мрежи водовода..... | 211 |
| 5.3.2.5.3. Могућност примене ИТ на проблему тачности и учесталости читавања, крађе воде и навика у потрошњи воде | 213 |
| 5.3.2.5.4. Могућност примене ИТ на проблему немогућности провере испуњења обавеза ЈКП БВК | 215 |
| 5.3.2.5.5. Могућност примене ИТ на проблему контаминације пијеће воде..... | 216 |
| 5.3.2.5.6. Могућност примене ИТ на проблем немогућност утврђивања приоритета одржавања сливника на значајну промену атмосферских прилика | 216 |
| 5.3.2.5. Предлог примене технолошких решења на основу анализе ланца вредности на наслеђеном информационом систему ЈКП БВК - Закључак анализе ланца вредности..... | 217 |
| 6. ИСТРАЖИВАЊЕ | 219 |
| 6.1. ОПИС НАЧИНА ИСТРАЖИВАЊА | 219 |
| 6.2. ПРИКУПЉАЊЕ И ОБРАДА ПОДАТАКА | 219 |
| 6.2.1. <i>Опис начина прикупљања података.....</i> | <i>219</i> |
| 6.2.1.1. Први круг истраживања | 219 |
| 6.2.1.2. Други круг истраживања..... | 220 |
| 6.2.2. <i>Узорак истраживања.....</i> | <i>224</i> |
| 6.2.3. <i>Ставови испитаника у вези са наслеђеним информационом системима.....</i> | <i>227</i> |
| 6.2.3.1. Дескриптивна статистика..... | 228 |
| 6.2.3.2. Т-тест..... | 230 |
| 6.2.3.2.1. Тестирање ставова испитаника у Србији и земљама региона..... | 231 |
| 6.2.3.2.2. Тестирање ставова испитаника у односу на старост наслеђеног система..... | 234 |
| 6.2.3.2.3. Тестирање ставова испитаника у односу на програмски језик Progress..... | 238 |
| 6.2.3.2.4. Тестирање ставова испитаника у односу на програмске језике четврте генерације (4GL)..... | 243 |
| 6.2.3.3. Корелација..... | 248 |
| 7. РЕЗУЛТАТИ ИСТРАЖИВАЊА | 259 |
| 7.1. ЕВАЛУАЦИЈА ПРЕДЛОЖЕНОГ ТЕОРИЈСКОГ ОКВИРА | 259 |
| 7.1.1. <i>Развојна перспектива.....</i> | <i>261</i> |
| 7.1.1.1. Анализа одговора на питање П18..... | 261 |
| 7.1.1.2. Анализа одговора на питање П19..... | 262 |
| 7.1.1.3. Анализа одговора на питање П15..... | 264 |
| 7.1.1.4. Анализа одговора на питање П21..... | 266 |
| 7.1.1.5. Анализа одговора на питање П6..... | 268 |

| | |
|--|------------|
| 7.1.1.6. Анализа одговора на питање П9..... | 270 |
| 7.1.1.7. Анализа одговора на питање П11..... | 272 |
| 7.1.1.8. Анализа одговора на питање П20..... | 274 |
| 7.1.1.9. Анализа одговора на питање П16..... | 275 |
| 7.1.1.10. Развојна перспектива - резултат..... | 277 |
| 7.1.2. Оперативна перспектива | 279 |
| 7.1.2.1. Анализа одговора на питање П14..... | 279 |
| 7.1.2.2. Анализа одговора на питање П22..... | 280 |
| 7.1.2.3. Анализа одговора на питање П10..... | 282 |
| 7.1.2.4. Анализа одговора на питање П11..... | 284 |
| 7.1.2.5. Анализа одговора на питање П12..... | 286 |
| 7.1.2.6. Анализа одговора на питање П13..... | 288 |
| 7.1.2.7. Оперативна перспектива - резултат | 290 |
| 7.1.3. Организациона перспектива..... | 292 |
| 7.1.3.1. Анализа одговора на питање П7..... | 292 |
| 7.1.3.2. Анализа одговора на питање П8..... | 294 |
| 7.1.3.3. Анализа одговора на питање П23..... | 296 |
| 7.1.3.4. Организациона перспектива - резултат..... | 298 |
| 7.1.4. Стратешка перспектива..... | 299 |
| 7.1.4.1. Анализа одговора на питање П24..... | 299 |
| 7.1.4.2. Стратешка перспектива - резултат | 300 |
| 7.1.5. Главна хипотеза..... | 302 |
| 7.1.5.1. Анализа одговора на питање П3..... | 302 |
| 7.1.5.2. Анализа одговора на питање П17..... | 303 |
| 7.1.5.3. Анализа одговора на питање П25..... | 305 |
| 7.1.5.4. Главна хипотеза - резултат..... | 306 |

8. РАЗМАТРАЊЕ МОДЕРНИЗАЦИЈЕ НАСЛЕЂЕНОГ СИСТЕМА..... 308

| | |
|---|-----|
| 8.1. ИСПИТИВАЊЕ УПОТРЕБЉИВОСТИ НАСЛЕЂЕНОГ ИНФОРМАЦИОНОГ СИСТЕМА ЈКП „БЕОГРАДСКИ ВОДОВОД И КАНАЛИЗАЦИЈА“ | 308 |
| 8.2. ТУМАЧЕЊЕ РЕЗУЛТАТА | 312 |
| 8.2.1. Тумачење резултата за апликацију АП1..... | 312 |
| 8.2.2. Тумачење резултата за апликацију АП2..... | 314 |
| 8.2.3. Тумачење резултата за апликацију АП3..... | 316 |
| 8.3. АНАЛИЗА ПОРТФОЛИЈА..... | 318 |
| 8.3.1. Процена пословне вредности..... | 318 |
| 8.3.1.1. Процена пословне вредности апликације АП1 | 320 |
| 8.3.1.2. Процена пословне вредности апликације АП2 | 321 |
| 8.3.1.3. Процена пословне вредности апликације АП3 | 322 |
| 8.3.2. Процена техничког квалитета..... | 324 |
| 8.3.3. Разматрање еволуционе стратегије апликација наслеђеног система..... | 325 |
| 8.4. РАЗМАТРАЊЕ ПРИСТУПА МОДЕРНИЗАЦИЈИ АПЛИКАЦИЈА | 326 |

| | |
|--|------------|
| 8.4.1. Разматрање повлачења или замене апликације АПЗ..... | 326 |
| 8.4.2. Разматрање стратегије модернизације апликација АП1 и АП2..... | 327 |
| 8.4.2.1. Разматрање софтверског реинжењеринга као стратегије модернизације апликација АП1 и АП2..... | 331 |
| 8.4.2.2. Разматрање облагања као стратегије модернизације апликација АП1 и АП2..... | 333 |
| 8.4.2.3. Разматрање миграције као стратегије модернизације апликација АП1 и АП2..... | 336 |
| 8.4.2.4. Предлог стратегије модернизације апликација АП1 и АП2..... | 338 |
| 8.5. ОСТАЛИ АСПЕКТИ МОДЕРНИЗАЦИЈЕ..... | 340 |
| 9. ЗАКЉУЧАК..... | 342 |
| 9.1. Доприноси | 342 |
| 9.2. Ограничења истраживања..... | 344 |
| 9.3. Будућа истраживања | 345 |

ЛИТЕРАТУРА

ПРИЛОЗИ

Прилог 1: АНКЕТА ЗА ЕВАЛУАЦИЈУ НОВОГ ТЕОРИСКОГ ОКВИРА

Прилог 2: АНКЕТА ЗА ОЦЕНУ УПОТРЕБЉИВОСТИ

Прилог 3: АНКЕТА ЗА ПРОЦЕНУ ПОСЛОВНЕ ВРЕДНОСТИ

БИОГРАФИЈА

Листа слика и табела

Листа слика

| | |
|---|-----|
| Слика 1. разликовање ИС од других ИТ-дисциплина..... | 14 |
| Слика 2. ПИЗМ модел..... | 16 |
| Слика 3 активности информационог система | 19 |
| Слика 4. окружење организације и главни фактори који утичу | 20 |
| Слика 5. положај ИС у реалном систему..... | 22 |
| Слика 6. Водопад модел | 24 |
| Слика 7. V модел..... | 25 |
| Слика 8. Прототипски модел | 26 |
| Слика 9. Фазе агилног развоја | 28 |
| Слика 10. двосмеран однос организације и ИТ-а..... | 35 |
| Слика 11. техничка дефиниција организације..... | 36 |
| Слика 12. концепт захтева за квалитетом и евалуације | 64 |
| Слика 13. WebQEM методологија | 73 |
| Слика 14 . дистрибуција активности одржавања | 82 |
| Слика 15. S системи | 84 |
| Слика 16. P системи | 85 |
| Слика 17. E системи..... | 86 |
| Слика 18. фазни процес модела еволуције | 88 |
| Слика 19. различите еволутивне активности | 89 |
| Слика 20. елементи наслеђеног система | 95 |
| Слика 21. интеграција од тачке до тачке | 102 |
| Слика 22. решења са одређеним степеном реинжењеринга корисничког интерфејса | 113 |
| Слика 23. односи реинжењеринга и њему сличних појмова..... | 117 |
| Слика 24 .NET Reflector, генерисање изворног кода из склопа | 119 |
| Слика 25. процес реинжењеринга | 125 |
| Слика 26. традиционални модел софтверског реинжењеринга | 127 |
| Слика 27. анализа портфолија | 128 |
| Слика 28. облагање наслеђеног система | 131 |
| Слика 29. пословна и ИТ архитектура..... | 135 |
| Слика 30. настанак новог (проширеног) теоријског оквира | 156 |
| Слика 31. нови (проширени) теоријски оквир | 157 |
| Слика 32. ланац вредности водоводних предузећа према Ofwat-у | 180 |
| Слика 33. ланац вредности ЈКП БВК | 185 |
| Слика 34. зачарани круг воде која не доноси приход | 189 |
| Слика 35. виртуозни круг воде која не доноси приход | 190 |

| | |
|---|-----|
| Слика 36. основна зона билансирања..... | 198 |
| Слика 37. типична четворослојна архитектура | 207 |
| Слика 38. архитектура решења за надзор окружења..... | 211 |
| Слика 39. архитектура за читавање | 214 |
| Слика 40. концептуална архитектура решења..... | 215 |
| Слика 41. архитектура решења за детекцију контаминације..... | 216 |
| Слика 42 решење за праћење количине воде доспеле у атмосферску канализацију..... | 217 |
| Слика 43. корелација између питања број 3 и осталих питања у пакету SPSS..... | 249 |
| Слика 44. корелација између питања број 6 и осталих питања у пакету SPSS..... | 250 |
| Слика 45. корелација између питања број 7 и осталих питања у пакету SPSS..... | 250 |
| Слика 46. корелација између питања број 8 и осталих питања у пакету SPSS..... | 251 |
| Слика 47. корелација између питања број 9 и осталих питања у пакету SPSS..... | 252 |
| Слика 48. корелација између питања број 10 и осталих питања у пакету SPSS..... | 252 |
| Слика 49. корелација између питања број 11 и осталих питања у пакету SPSS..... | 253 |
| Слика 50 корелација између питања број 12 и осталих питања у пакету SPSS..... | 253 |
| Слика 51. корелација између питања број 13 и осталих питања у пакету SPSS..... | 254 |
| Слика 52. корелација између питања број 14 и осталих питања у пакету SPSS..... | 254 |
| Слика 53. корелација између питања број 15 и осталих питања у пакету SPSS..... | 255 |
| Слика 54. . корелација између питања број 16 и осталих питања у пакету SPSS..... | 255 |
| Слика 55. корелација између питања број 17 и осталих питања у пакету SPSS..... | 256 |
| Слика 56. . корелација између питања број 18 и осталих питања у пакету SPSS..... | 257 |
| Слика 57. корелација између питања број 19 и осталих питања у пакету SPSS..... | 257 |
| Слика 58. корелација између питања број 20 и питања број 21 у пакету SPSS..... | 257 |
| Слика 59. корелација између питања број 21 и питања број 22 и питања број 24 у пакету SPSS..... | 258 |
| Слика 60. корелација између питања број 22 и питања број 24 у пакету SPSS..... | 258 |
| Слика 61. корелација између питања број 23 и питања број 24 и питања број 25 у пакету SPSS..... | 259 |
| Слика 62. повезаност елемената перспектива наслеђеног система са питањима упитника | 260 |
| Слика 63. портфолио апликације АП1, АП2 и АП3..... | 326 |
| Слика 64. архитектурална замисао облагања наслеђеног система ЈКП БВК и његово инкорпорирање у ширу архитектуру хибридног облака | 336 |

Листа табела

| | |
|--|-----|
| Табела 1. Декомпозиција употребљивости према одређеним ауторима | 56 |
| Табела 2. Десет критичних наслеђених система федералних агенција | 108 |
| Табела 3. поређење инжењеринга у напред и реверзног инжењеринга..... | 121 |
| Табела 4. инфраструктура ЈКП БВК..... | 142 |
| Табела 5. Процењени физички обим активности у 2019. години | 143 |
| Табела 6. проблеми и изазови у активностима ланца вредности..... | 185 |
| Табела 7. Водни биланс | 190 |
| Табела 8. папирање примарних активности у ланцу вредности ЈКП БВК и чиниоца воде која не доноси приход | 193 |
| Табела 9. објединињени упитник са питањима (25 питања)..... | 221 |
| Табела 10. број испитаници по земљама | 225 |
| Табела 11. број испитаника из земаља из региона | 225 |
| Табела 12. број испитаника из Србије по градовима | 225 |
| Табела 13. испитаници из Босне и Херцеговине по градовима | 226 |
| Табела 14. испитаници из Бугарске по градовима | 226 |
| Табела 15. испитаници из Црне Горе по градовима..... | 226 |
| Табела 16. испитаници из Хрватске по градовима..... | 226 |
| Табела 17. испитаници из Мађарске по градовима | 227 |
| Табела 18. испитаници из Румуније по градовима..... | 227 |
| Табела 19. испитаници из Северне Македоније по градовима | 227 |
| Табела 20. испитаници из Словеније по градовима | 227 |
| Табела 21. табела фреквенција | 228 |
| Табела 22. придружене вредности статистичког обележја у програмском пакету SPSS .. | 229 |
| Табела 23. вредност дескриптивних статистика | 229 |
| Табела 24. дескриптивне статистике за групе испитаника из Србије и из земаља региона | 231 |
| Табела 25. резултат Т-теста којим се тестирају ставови испитаника из Србије и из земаља региона..... | 232 |
| Табела 26. дескриптивне статистике за групе испитаника код којих је наслеђен систем предузећа у којем раде стар до 10 година и испитаника код којих је наслеђен систем предузећа старији од 10 година..... | 235 |
| Табела 27. резултат Т-теста којим се тестирају ставови групе испитаника код којих је наслеђен систем предузећа стар до 10 година и групе испитаника код којих је наслеђен систем предузећа стар преко 10 година | 236 |
| Табела 28. дескриптивне статистике за групе испитаника код којих је наслеђен систем предузећа у којем раде није написан у програмском окружењу Progress и оних код којих је наслеђен систем предузећа у којем раде написан у програмском окружењу Progress | 239 |

| | |
|--|-----|
| Табела 29. резултат Т-теста којим се тестирају ставови групе испитаника код којих је наслеђен систем предузећа није написан у програмском окружењу Progress и групе испитаника код којих је наслеђен систем предузећа је написан у програмском окружењу Progress..... | 240 |
| Табела 30. декриптивне статистике за групе испитаника код којих је наслеђен систем предузећа у којем раде није написан у програмском језику 4. генерације и оних код којих је наслеђен систем предузећа у којем раде је написан у програмском језику 4. генерације | 244 |
| Табела 31. резултат Т-теста којим се тестирају ставови групе испитаника код којих наслеђен систем предузећа није написан у програмском језику 4. генерације и групе испитаника код којих је наслеђен систем предузећа написан у програмском језику 4. генерације..... | 245 |
| Табела 32. резултати анализе одговора на питања који су у вези са Посебном хипотезом 1 истраживања..... | 277 |
| Табела 33. резултати анализе одговора на питања који су у вези са Посебном хипотезом 2 истраживања..... | 290 |
| Табела 34. резултати анализе одговора на питања који су у вези са Посебном хипотезом 3 истраживања..... | 298 |
| Табела 35. резултати анализе одговора на питања који су у вези са Посебном хипотезом 4 истраживања..... | 300 |
| Табела 36. резултати анализе одговора на питања који су у вези са Главном хипотезом истраживања..... | 307 |
| Табела 37. просечне вредности одговора корисника на питања у вези употребљивости апликације АП1 | 310 |
| Табела 38. просечне вредности одговора корисника на питања у вези употребљивости апликације АП2 | 311 |
| Табела 39. просечне вредности одговора корисника на питања у вези употребљивости апликације АП3 | 312 |
| Табела 40. кључни фактори за оцену пословне вредности наслеђеног система..... | 319 |
| Табела 41. просечне вредности одговора на кључна питања за апликацију АП1 | 320 |
| Табела 42. тежинска пословна вредност апликације АП1 | 321 |
| Табела 43. просечне вредности одговора на кључна питања за апликацију АП2 | 321 |
| Табела 44. тежинска пословна вредност апликације АП2 | 322 |
| Табела 45. просечне вредности одговора на кључна питања за апликацију АП3 | 322 |
| Табела 46. тежинска пословна вредност апликације АП3 | 323 |

1. Увод

У модерном пословном окружењу, које је окарактерисано перманентним и брзим променама, предузећа реализују своју визију, мисију и циљеве. Она су са једне стране окренута дневним, оперативним активностима путем којих се њихови циљеви материјализују, а са друге и оним стратешким усмереним на дугорочни опстанак. Информациони системи, као модели стварног система успостављају и одржавају интерне и екстерне везе предузећа разменом података, информација и знања, а при томе остварују своје 3 основне улоге (Marakas & O'Brien, 2012, p. 7). Прва од поменутих улога је подршка пословним операцијама и процесима. Друга улога је подршка приликом доношења одлука од стране запослених и менаџмента. Трећа улога је везана за подршку у циљу остваривања конкурентских предности. Сложеност окружења у коме наступа једно предузеће, одликује се вишеструким испреплетаним интеракцијама са чиниоцима околине. Непосредна околина су купци, добављачи, повериоци, регулаторна тела и конкурентска предузећа. Сви они заједно наступају у једној широј околини коју чине социо-економски трендови, политички услови, технолошке иновације и глобални догађаји (Laudon & Laudon, 2017, p. 46). Кључ успеха је предвидети промене у комплексном окружењу и адекватно пружити одговор на њих, у циљу остварења мисије и дугорочног опстанка предузећа.

Рачунарски засновани информациони системи чине хардвер, софтвер, базе података, мреже, људи и процедуре, који су подешени за скупљање, манипулисање, складиштење и процесирање података у информације (Stair & Reynolds, 2018, p. 6). Они се могу посматрати као скуп информационих технологија (хардвер, софтвер, подаци и комуникационе технологије), људи и процедура (пословних процеса). У развоју неког система, па самим тим и информационог система, постоји скуп фаза које се спроводе и познатије су под именом животни циклус развоја информационог система. Те фазе могу да буду: планирање и идентификовање проблема, анализа, пројектовање (дизајн), реализација, употреба и одржавање, као у случају водопад модела.

Упоредо са употребом система, отпочиње и фаза његовог одржавања, која је уједно и најдужа фаза у оквиру животног циклуса. Током процеса одржавања, услед потребе за вишеструким изменама у самом ИС, а ради потребе за одржавањем његових интерних и екстерних веза, уз очување оперативне и стратешке подршке пословању и несметано спровођење дневних операција, долази до постепеног нарушавања његове структуре и очекиваног понашања, тј. он полако креће да стари. Како систем стари

спроводи се његово одржавање, које има за циљ да отклони грешке, побољша перформансе или се прилагоди променљивом окружењу (Grubb & Takang, 2003, p. 7). Процес одржавања се спроводи до оног тренутка, када се систем више не може прилагодити новонасталим променама, и тада је време за његово повлачење из употребе

Иако делује веома уобичајено, да једна појава има свој почетак, ток и завршетак, пракса је показала да повлачење информационих система из употребе може представљати велики проблем, са којим се информатичка струка среће од почетка увођења првих информационих система у пословање до данас. Поменути проблем се огледа у томе да су наведени информациони системи, временом постали наслеђени системи (енг. *legacy systems*). То су системи развијени у прошлости уз употребу старих или застарелих технологија, пројектовани и имплементирани у складу са ондашњим методолошким поставкама и концептима, најчешће уз употребу тадашњих релевантних програмских језика и усвојених пракси у развоју информационих система. Наслеђени системи су операционализовани системи дизајнирани, имплементирани и инсталирани у радикално другачијем окружењу од текуће ИТ стратегије (Tromp & Hoffman, 2003, p. 2), а при томе не подржавају тренутну ИТ стратегију (Mitleton-Kelly, 2006, p. 55). Њихови проблеми се огледају и у томе да не могу одговоре на нове или измењене захтеве у пословању, које се спроводе кроз пословне процедуре или пословне процесе. Поред тога, они могу и да имају проблеме технолошке природе, јер не могу да користе нове технологије, које настају као последица убрзаног технолошког развоја. За њихово одржавање издвајају се значајна финансијска средства, чија је висина често у несразмеру са оним што руководство предузећа очекује да ће се таквим једним улагањем постићи. ИТ менаџери не могу да их се једноставно реше, јер подржавају главне пословне процедуре и процесе, с'тога се стално намеће питање, шта радити са наслеђеним системима ?

Проблеми који наслеђени системи доносе посебно су изражени у предузећима из јавног сектора из разлога што су она зачетници у увођењу ИТ технологија у своје пословање. Поред тога, постоје и случајеви када су наслеђени системи сопствена (енг. *in-house*) ИТ решења поменутих предузећа, често стара по неколико деценија, прављена у складу са захтевима запослених, где се запослени и до данашњег дана у потпуности ослањају на исте приликом извођења пословних операција.

Један од разлога великог несразмера у висини инвестираних средстава у наслеђене системе и стратегије које предузеће спроводи у вези истих и ефеката спроведних промена, јесте недовољно разумевање наслеђених система. Они су једна сложена појава, састављена из великог броја елемената, као и комплексне интеракције

између поменутих елемената, а корисници их често их перципирају кроз једну димензију, један поглед или једну перспективу. Таквим изолованим посматрањем долази до неразумевања наслеђених информационих система, јер се „види“ само један њихов део, најчешће онај са којим корисник (коју год улогу да има) има интеркацију. Уколико се ради о развојној перспективи, програмери, пројектанти и главне архитекте система углавном виде само оне димензије које су у вези са њиховим делом посла, а при томе апстрахују преостале димензије. На примеру развојне перспективе, запослени се баве програмирањем, пројектовањем и организовањем различитих програмских модула, као и њихових подмодула, а при томе могу да занемаре нпр. пословну перспективу, тј. који све пословни процеси су подржани датим модулима или перспективу сервиса, нпр. да ли је довољно употребљив кориснички интерфејс и да ли је перципирани квалитет на одговарајућем нивоу тако да крајњи корисници су задовољни приликом коришћења система. Са једне стране занемаривање преосталих перспектива је корисно јер се корисници, у односу на своју улогу, фокусирају на оне активности које су им значајне са аспекта њиховог посла, док са друге стране губи се шира слика и сложеност коју један такав систем носи са собом. Да би се спречило изоловано посматрање наслеђеног система, потребно је систем сагледати свеобухватно, тј. из свих његових перспектива: развојне, оперативне, организационе и стратешке и тада размотрити стратегије модернизације које би било погодно применити на основу дубљег разумевања система и његових проблема.

Не само што је потребно да се систем сагледа у садашњости и уоче проблеми који се могу јавити у неколико перспектива тренутно, већ имати у виду и значај да када се усвоји одређена стратегија модернизације она треба да обухвати и део будућих изазова који ће се ставити пред наслеђен систем. Дакле, уколико се чини озбиљан подухват да би се обезбедила дугорочна еволуција система, докле год је то могуће и изводљиво, такав подухват је потребно да садржи и одређену усмереност ка будућности, тј. перспективу. Тако да свеобухватно посматрање наслеђеног система, не само што има потребу да уочи проблеме у садашњости, већ има и за циљ да предвиди одређене проблеме у будућности који неће дозволити примену нових и ефикаснијих пословних модела, чиме ће се нарушити дугорочни опстанак предузећа.

Уз технолошки напредак и потребу да се омогући ефикасније пословање уз повећану интеркацију делова једног система са околином, информациони системи постају део једног ширег и већег система, који се састоји из скупа појединачних система и назива се систем система (енг. *System of Systems*). Саставни системи сарађују међусобно

и сваки од њих омогућава јединствене способности великог систем-система. Они су градивни део Индустрије 4.0 исказане у форми сајбер-физичких система (енг. *cyber-physical systems*), интернета ствари (енг. *Internet of things*), различитих сценарија паметних градова (енг. *smart cities*) и тд. Како наслеђени системи, су ограничени у примени нових технологија, које Индустрија 4.0 намеће, као што су технологије великих података (енг. *Big Data*), технологије рачунарства у облаку (енг. *cloud computing*), интернета ствари, сајбер безбедност и тд., додатно ће бити наглашени њихови проблеми у погледу интеграције, интероперабилности и неадекватне архитектуре.

С тога за свеобухватно сагледавање наслеђеног информационог система потребно је сагледати текуће стање путем сагледавања његових различитих перспектива: развојне, оперативне, организационе и стратешке, као и његову перспективу која има за циљ да утврди одређена ограничења у будућности. На тај начин спречава се изоловано посматрање феномена и ствара се оквир, који пружа дубље разумевање феномена и могућност за одабир најбоље алтернативне из скупа стратегија модернизације наслеђеног система.

1.1. Проблем и предмет истраживања

Наслеђени информациони системи, су и даље у употреби у многим предузећима, јер се путем њих обављају најважније пословне операције и процеси. Са једне стране ИТ менаџери би желели да их повуку из употребе, јер они коче и ограничавају развој пословања, а са друге стране не могу их тако једноставно склонити јер подржавају критичне пословне операције и процесе. Из поменутих разлога организације констатнтно улажу у њихово одржавање. На примеру САД, на основу јавно доступног истраживања агенције GAO (Government Accountability Office), влада САД је планирала да потроши преко 90 милијарди долара у 2019. години на ИТ, где се око 80% троши за одржавање наслеђених система и то 10 критичних наслеђених система федералних агенција (United States Government Accountability Office, 2019).

Да би омогућила да се нови пословни захтеви имплементирају предузећа приступају процесу модернизације наслеђених система. Велики број неуспелих пројеката модернизације доводи се у везу са честом еволуцијом поменутих система, из разлога што (Pressman & Maxim, 2015, p. 8): софтвер мора да се адаптира да би задовољио потребе нове технологије или новог рачунарског окружења, софтвер мора да се унапреди да би имплементирао нове пословне захтеве, софтвер мора да се прошири да би био интероперабилан са неким другим савременим системима или базама података и софтвер

мора да се архитектурално реконструише да би се одржао у рачунарском окружењу које еволуира. Према томе, закони еволуције софтвера у великој мери детерминишу активности које ће се спроводити над информационим системима, па је са аспекта менаџмента важно припремити се за исте и адекватно реаговати.

Наслеђени системи се могу посматрати из различитих тачака гледишта, тј. перспектива. Свака од ових перспектива, било да је развојна, оперативна, организациона или стратешка, нуди низ елемената које одређене улоге, које користе непосредно или не наслеђени информациони систем, перципирају као проблематичне. Те перспективе уобичајено дају поглед у садашњости и указују на проблеме који већ постоје, док стратешка перспектива указује и оне проблеме које ће постојати, па самим тим она обједињује и поглед у будућност, тј. перспективу. Појам Индустрије 4.0 укључује велики број нових технологија и концепата као што су: рачунарство у облацима, Интернет ствари, сајбер-физички системи, технологије великих података и др. Ове промене у технолошком домену, генеришу промене у пословном домену и утичу да се значајно мењају или потпуно напуштају постојећи пословни процеси предузећа. У таквој једном измењеном окружењу, перспектива (поглед у будућност) наслеђених система готово да не постоји услед њихове немогућности да се прилагоде и с'тога је потребно спровести адекватну модернизацију истих да би измењени могли да функционишу као део једног већег система.

Дакле, **проблем истраживања** јесте аспект функционисања и обављања основних улога наслеђених информационих система у све већем и сложенијем окружењу, подстакнутим свеукупним променама окружења, које ставља нове захтеве пред њих, али и оним интерним путем којих се ефективно и ефикасно обављају свакодневне операције.

У теорији информационих система и софтверског инжењерства постоји разне дефиниције и објашњења појма наслеђених система. Броди и Стоунбрејкер су још 1995. године указали да су то системи који се значајну опирају модификацијама и изменама (Hainaut et al., 2010, p. 189). Други аутори указују на проблеме који они стављају пред организацију (Jesus Bisbal et al., 1997, p. 1). Неки аутори, као што је Лиу, истичу немогућност наслеђених система да адекватно испрате пословне потребе (Liu et al., 1998, p. 1). Панг указује да се у њих уложило значајно време и новац, а да су инсталирани на мејнфрејму или минирачунару (Khosrow-Pour, 2007, p. 411). Меки је мишљења да је то информациони систем из ранијег периода технолошког развоја (Khosrow-Pour, 2007, p. 411). Истиче се и став да су они можда засновани на старијим технолојима, али су критични за свакодневне операције (Gartner, n.d.). Менаџери желе да их се реше, али то

не могу јер подржавају критичне пословне операције (Sommerville, 2016, р. 261). Неки аутори наводе и чињеницу да је то систем много већи од наслеђеног софтера, који обухвата: људе, експертизу, хардвер, податке, пословне процесе и приступе у одржавању и развоју софтвера (Brooke & Ramage, 2001, р. 3). Веома је тешко схватити ове системе, па су из то разлога проблем приликом интеграције са другим системима (Vestues & Knut, 2019, р. 2). Из свих наведених дефиниција може се приметити велика дисперзивност у кључним елементима софтверског наслеђа на које поједини аутори стављају акценат.

Поред тога, значајна средства се улажу у пројекте модернизације наслеђених система, а пракса показује да постоји велики број пропалих пројеката модернизације, чак и ако цена инвестиције прелази милијарде долара (Ulrich & Newcomb, 2010, р. 5). И у нашој земљи спроведе се одређене стратегије које имају за циљ да наслеђени систем повуку из употребе, али као и у свету завршавају се неуспешно из разних разлога (Rodić, 2009). Дакле, наслеђени системи су феномен, који није последица примене одређене специфичне праксе предузећа, већ је неминовна појава, зато што је под утицајем неизбежног процеса старења софтвера.

Значајна финансијска средства која се издвајају за одржавање наслеђених система, велики број различитих дефиниција наслеђених система у теорији, велика инвестиције у модернизацију ових система и велики број пропалих пројеката модернизације и поред значајно инвестираних средстава у пракси, указују на чињенице:

- да је феномен наслеђених информационих система изузетно комплексан
- да се наслеђени системи у теорији недовољно (или изоловано) схватају
- да неуспели пројекти модернизације у пракси су последица недовољног разумевања ових система
- да корак ка успешној модернизацији наслеђених система јесте њихово адекватно разумевање са сложеним односима између свих његових елемената, али и са чиниоцима сложеног перманентно променљивог окружења

Да би се спречило изоловано посматрање наслеђених система, којим се указује на само неке његове аспекте или елементе, поставља се питање да ли могуће из више различитих перспектива или гледишта сагледати овај феномен свеобухватно и уколико је то могуће које су то перспективе које ће омогућити довољано исцрпне информације о наслеђеном систему на начин да се сви његови релевантни аспекти сагледају у целиности и тиме омогући његова успешна модернизација.

Предмет истраживања је сагледавање наслеђених информационих система из различитих перспектива у све сложенијем окружењу, где је потребно одговорити на садашње и будуће проблеме (проблеми у перспективи), са фокусом на перспективе наслеђеног информационог система ЈКП „Београдског водовод и канализације“.

1.2. Циљеви истраживања

Добро осмишљен и реализован информациони систем увећава ефикасност пословања, задовољава потребе запослених на оперативним и руководећим местима, утиче на успостављање и одржавање односа са купцима и добављачима и у неком најширем обиму може да утиче и на јавно мњење. Наслеђени системи услед раније поменутих проблема не могу се прилагодити променама и због тога на одређени начин они ограничавају развој пословања, па самим тим не могу се у потпуности задовољити потребе и адекватан ниво сарадње између свих учесника у пословању са предузећем на ефикасан начин. Када се има у виду да предузеће наступа на заостреном тржишту, где је веома важно нове пословне идеје и пословне моделе брзо спровести од њихове замисли до реализације одређене у виду производа и услуга које се нуде у односу на конкуренцију, наслеђени информациони системи представљају значајно ограничење. Њихова инертност онемогућава их да се прилагоде променама, било да оне долазе из ближег или даљег окружења, па постају претња за мисију, визију и циљеве предузећа. Истовремено они подржавају кључне пословне процесе и садрже вредна знања о обављању посла која су се деценијама прикупљала и прилагођавала пословању.

С тога модернизација наслеђеног система је важна у **циљу** испуњавања свих улога које једно предузеће има у оквиру друштва, као и будућих улога и **циљева** за које се може претпоставити да ће бити сложенији и значајнији услед технолошког напретка и нових могућности да се раније познати проблеми у пословању а нерешиви или нерешиви довољно исплативо, реше.

Да би модернизација наслеђених система била могућа, тј. да би била спроведена ефективно и ефикасно, због њихове велике сложености потребно их је прво схватити и разумети дубину промена које ће бити потребно спровести у поступку модернизације. Зато је **циљ** истраживања потпуније разумевање наслеђених информационих система као последица посматрања из више различитих тачки гледишта, тј. перспектива, које би требало да обезбеде целовито посматрање феномена.

Поред тога, анализа модернизације наслеђеног система на примеру ЈКП „Београдски водовод и канализација“ може бити значајна и из разлога да се сачувају

вишедеценијска улагања у поменути систем и да се сачувају вредна знања о познавању пословних операција и процеса, настала у интеракцији људи који су развијали поменути систем и запослених који су давали захтеве за израду. Обзиром да се проблеми наслеђених система, обрађују на примеру ЈКП „Београдски водовод и канализација“ кроз различите перспективе, **циљ** је да се анализирају поменути проблеми на том случају, уз поређење наслеђених система водоводних предузећа у земљи и у региону, да се утврди њихова специфичност и да се заузме мишљење о томе како је потребно водити модернизацију конкретног наслеђеног система, у измењеним условима технолошког напретка. Поред тога, сва предузећа која имају сличне проблеме и сличан наслеђен систем могла би да искористе поменути случај, да дефинишу свој правац модернизације. Тиме би био обухваћен и циљ да одређена класа наслеђених система, као препоруку може користити презентовани правац модернизације. Тиме би био обухваћен и **циљ** да одређена класа наслеђених система, као препоруку може користити одређен правац модернизације уз себи својствена прилагођавања.

1.3. Очекивани резултати (хипотезе)

Очекивани резултати (доприноси) дисертације су:

- могућност потпунијег сагледавања феномена наслеђених система сагледавањем свих његових перспектива
- указивање на нову врсту проблема наслеђених система који се јављају у стратешкој перспективи као последица перцепције улоге ИТ функције од стране топ менаџмента
- анализа ланца вредности која може послужити у дефинисању будућих ИТ решења које ће бити потребно имплементирати, као и указивање на проблеме и ограничења које постојећи систем има да би могао да ради са таквим решењима
- анализа стања наслеђеног система ЈКП „Београдски водовод и канализација“ посматрањем перспектива и предлог адекватне стратегије модернизације
- могућност уопштавања стратегије модернизације на наслеђеним системима израђеним у програмским језицима 4. генерације

Главна (општа) хипотеза истраживања гласи:

Посматрањем различитих перспектива (тачки гледишта) наслеђеног информационог система као и његове перспективе (поглед ка будућности) могуће је

потпуније сагледати наслеђени систем и на основу тога креирати путоказ ка његовој модернизацији.

На основу дефинисаног предмета истраживања, могу се дефинисати следеће **посебне хипотезе**:

1. Посматрањем развојне перспективе наслеђеног система могуће је сагледати проблеме наслеђених система из улоге запослених који га развијају, а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема.

2. Посматрањем оперативне перспективе наслеђеног система могуће је сагледати проблеме наслеђених система из улоге корисника, а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема.

3. Посматрањем организационе перспективе наслеђеног система могуће је сагледати проблеме наслеђених система из улоге менаџера, а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема.

4. Посматрањем стратешке перспективе наслеђеног система могуће је сагледати проблеме наслеђених система из улоге топ менаџмента, а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема.

1.4. План рада

Израда дисертације је реализована кроз 8 фаза:

Прва фаза: Изучавање релевантне литературе за израду дисертације из области информационих система, употребљивости као важног атрибута квалитета софтвера, наслеђених система, модернизације наслеђених система, софтверског реинжењеринга, облагања наслеђених система и миграције.

Друга фаза: Утврђивање недостатака у постојећим теоријским приступима у посматрању наслеђених система. Интегрисање постојећих теоријских приступа у нов теоријски оквир. Отклон од постојећих теоријских приступа додавањем нових елемената у посматрању предмета истраживања у теоријски оквир и проширивање теоријског оквира. Утврђивање подобности ЈКП „Београдски водовод и канализација“ као случаја у истраживању. Посматрање теоријског оквира на случају ЈКП БВК. Анализа ланца вредности ЈКП БВК да би се утврдила ограничења наслеђених система у примени нових

технологија у будућности које имају за циљ да побољшају активности ланца вредности и тиме се врши употпуњавање стратешке перспективе.

Трећа фаза: Креирање упитника за прикупљање података о ставовима испитаника о новом теоријском оквиру. Успостављање комуникације са испитаницима и прикупљање података од стране испитаника како из Србије, тако и из земаља у региону.

Четврта фаза: Спровођење анализе резултата, применом статистичких метода. Анализа ставова испитаника и извођење закључака о подобности теоријског оквира.

Пета фаза: Креирање упитника за прикупљање података о ставовима испитаника (корисника система) у вези употребљивости наслеђеног система ЈКП БВК, према SUS методи. Анализа употребљивости, извођење закључака и оцена техничког квалитета апликација наслеђеног система.

Шеста фаза: Креирање упитника за прикупљање података о ставовима испитаника (менаџера) у вези пословне вредности апликација наслеђеног система ЈКП БВК, према методи Дедекеа (Dedeke, 2012). Анализа резултата, извођења закључака и оцена апликација наслеђеног система.

Седма фаза: Спровођење портфолио анализе апликација наслеђеног информационог система, на основу резултата пете и шесте фазе. Одређивање подобности модернизационих стратегија за апликације наслеђеног система ЈКП БВК. Разматрање подобности софтверског реинжењеринга, облагања и миграције система као модернизационих стратегија за апликације које је потребно модернизовати, на основу свих претходно спроведених анализа, као и специфичности наслеђеног система ЈКП БВК. Дефинисање правца модернизационе стратегије наслеђеног система ЈКП БВК употребом облагања.

Осма фаза: Дискусија добијених резултата истраживања, разматрање ограничења истраживања и дефинисање будућних праваца истраживања.

1.5. Метод и узорак истраживања

Следеће методе истраживања ће се користити приликом реализације ове докторске дисертације:

- аналитичко-синтетичка метода
- метода анализе садржаја
- експертне методе научног испитивања (анкета за сакупљање података)

- методе статистичке обраде података (дескриптивна статистика, t-тест, корелациона анализа)
- метода студије случаја

Истраживање је спроведено на 3 различите групе испитаника. Прва група испитаника су људи који раде у ИТ сектору водоводних предузећа на развоју и одржавању информационог система и ИТ менаџери у водоводним предузећима у Србији и земљама у региону: Босни и Херцеговини, Црној Гори, Северној Македонији, Хрватској, Словенији, Мађарској, Бугарској и Румунији. Најчешћи је случај да постоји једно водоводно предузеће у граду. Укупан број испитаника је 63. За обраду податка је коришћен статистички алат SPSS.

Друга група испитаника су корисници који раде на апликацијама наслеђеног система ЈКП БВК. Укупан број испитаника је 43. Коришћена је SUS метода за оцену употребљивости система.

Трећа група испитаника су менаџери који воде организационе јединице које користе апликације наслеђеног система ЈКП БВК. Укупан број испитаника је 19.

2. Информационе технологије и информациони системи

2.1. Информационе технологије и информациони системи

Веома често у свакодневној комуникацији може се приметити да се појмови информациони системи (енг. *information systems*), скраћено ИС, и информационе технологије (енг. *information technologies*), скраћено ИТ, користе равноправно, али иако постоје одређене сличности међу њима, постоје и специфичне разлике које треба нагласити, уз прихватање чињенице да су појмови веома повезани између себе. Једно од мишљења јесте да су ИТ део ИС-а. Група аутора (Восиј et al., 2015, р. 3) наводи дефиницију Британске академије за информационе системе „Информациони системи су средства путем којих организације и људи, употребом информационих технологија скупљају, процесирају, складиште, користе и преносе (дисеминирају) информације.“ Поменута дистинкција је усвојена и у овом раду.

Информациони системи се састоје од неколико компоненти: хардвера (енг. *hardware*), софтвера (енг. *software*), података (енг. *data*), људи и процеса, при чему прве три спадају у категорију технологија. Док друге две компоненте људи и процеси, су оно што раздваја идеју о информационим системима искључиво као технолошког поља (Bourgeois, 2014, pp. 5–6). Поред тога, у данашњим условима важна је и улога технологија комуникација (енг. *communication*), тако да и она чини технолошку компоненту.

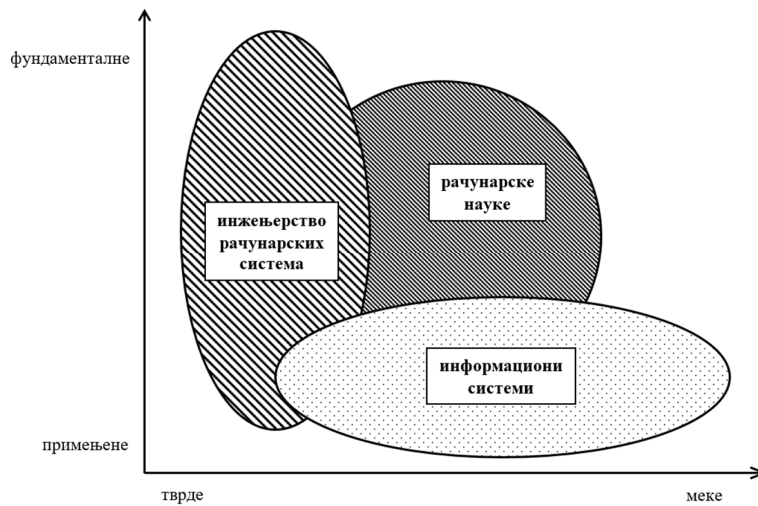
Да би се нагласила функција комуникације, као једна од најважнијих функција, а пре свега употреба глобалне мреже Интернета, појам информационе технологије се проширује, појмом информационо-комуникационе технологије (енг. *information and communications technology*), скраћено ИКТ (енг. *ICT*). Појмови се могу користити равноправно, али има и ставова да је адекватнија употреба ИКТ појма услед тога што се јасно раздвајају компоненте информације и компоненте комуникације. Па се према Чуданову (Čudanov, 2007, р. 13) ИКТ дефинишу као „скуп технологја које се користе у процесу сакупљања, одабира, верификације, обраде, преноса, приказа, објављивања и поделе информација или у процесима комуникације“. У овом раду појмови ИТ и ИКТ, ће се третирати истоветно, што је у складу са предметом истраживања и темом саме дисертације.

Оно што често прави конфузију у дефинисању разлика између појмова ИТ и ИС, јесте да због употребе технологије људи мисле да су сви информациони системи засновани на употреби рачунара, што није случај, јер ИС може бити у реализован нпр. у форми папира, у форми глинених таблица или на камену и тд. Дакле, једна компонента ИТ, технологије чувања информација, не морају да буду реализоване употребом рачунарски заснованих (енг. *computer-based*) технологија, као што су нпр. базе података. Неки од примера ИС још од настанка цивилизације па до данашњих дана могу да буду (Marakas & O'Brien, 2012, p. 6):

- Употреба димних сигнала се користила као пренос информација на одређену удаљност када је прималац информације није у непосредној околини пошиљаоца.
- Претрага каталога у библиотеци употребом картица које садрже основне податке о књигама омогућава да по различитим критеријумима читаоци дођу до жељене литературе.
- Књиговодствена документација је вековима у употреби и омогућава бележење разних финансијских података, трансакција и стања на рачуну клијената.

Из угла данашњице, може се рећи да је фокус информационих система више на подршци пословању превазилажењем граница између људи и технологија употребом пословних процедура и процеса, а да је фокус самих информационих технологија на софтверу, хардверу, умрежавању којима се изграђује информациони систем.

Уважавајући поглед Лија, о томе да област информационих система проучава много више него искључиво технолошке системе, или само друштвене системе, већ проучава феномене који настају из интеракције, Ависон и Елиот дају поређење фокуса информационих система, са фокусом ИТ-повезаних дисциплина: рачунарске науке (енг. *computer science*) и инжењство рачунарских система (енг. *computer systems engineering*) (D. Avison & Elliot, 2006, p. 6). Ово поређење је приказано на слици 1. Они указују да је код ИС много више фокус на примени технологије, него на фундаменталне технологије и теорије. Поред тога фокус је много више усмерен на интеракције између људи и организација (тзв. „меки“ (енг. *soft*) проблеми) и технологије, у односу на саме технологије (тзв. „тврди“ (енг. *hard*) проблеми).



Слика 1. разликовање ИС од других ИТ-дисциплина, преузето (Avison & Eliot, 2006)

2.1.1. Дефинисање информационих система

За дефинисање појма информационих система, потребно је да дефинишемо његове основне чиниоце, а то су појмови система и информација. Широко прихваћена дефиниција система је колекција међуповезаних компоненти које раде заједно у остваривању заједничког циља. За потребе симулације система, аутори дају ширу дефиницију (Раденковић et al., 1999, р. 9): „Под реалним системом подразумевамо уређен, међузависан скуп елемената који формирају јединствену целину и делују заједнички како би остварили задати циљ или функцију, без обзира да ли се ради о природном или вештачком систему, и такође, без обзира да ли тај систем у посматраном тренутку постоји или се његово постојање планира у будућности.“ Системима као научном дисциплином се бави општа теорија система. Систем може да има своје улазе и излазе, а његова функција је да посредством трансформације од улаза произведе излазе. Путем улаза околина делује на систем, док путем излаза систем делује на околину. Уколико вредности излаза зависе само од улаза, тада се може рећи да су то системи без меморије, а ако зависе и од стања, тада су системи са меморијом. Појам стања Петровић дефинише као „сажету представу претходних понашања система, довољно потпуна да нам омогући да на основу улазних дејстава предвидимо каква ће бити излазна дејства и промене самог стања“ (Петровић, 1998, р. 21). Појмови система, његових улаза, излаза и стања су основа за разумевање ИС.

Други основни чинилац јесте информација. Веома често се дешава да се информације поистовећују са подацима, а да се знање поистовећује са информацијама. Односи између ових појмова су сложени и у различитим дисциплинама се третирају на различит начин. Ипак, један од приступа који се често користи у области информационих система и менаџмента знања (енг. *knowledge management*) се издвојио и нагласио да постоји градација у појмовима података (енг. *data*), информација (енг. *information*), знања (енг. *knowledge*) и мудрости (енг. *wisdom*). Поменути приступ се назива хијерархија подаци-информације-знање-мудрост, скраћено ПИЗМ (енг. *DIKW*) хијерархија, познат још под називом „пирамида знања“.

Турбан и Рајнер наводе прва три појма. „Подаци су елементарни описи ствари, догађаја, активности, и трансакција који су забележени, класификовани и сачувани, али нису организовани и не носе никакво конкретно значење. Подаци могу бити бројеви, слова, облици, звуци или слике. Информације чине подаци организовани на такав начин да имају значење и вредност за примаоца. Прималац тумачи значење и доноси закључке и импликације из информације. Знање се састоји од података и/или информација који су организовани и обрађени тако да омогуће примену ранијег разумевања проблема, искуства, акумулираног учења и стручног знања на конкретан пословни проблем“ (Rainer & Turban, 2009, p. 6). Мудрост је способност да се направи добра процена и предузме адекватна акција на основу сакупљеног искуства и знања (Bonwich, n.d.). Поменути приступ, је додатно анализиран и систематизован на већем броју извора, а затим су предложена су одређена проширења и објашењења поменутих појмова.

Аутор (Rowley, 2007) у својој анализи издаваја следећа запажања:

- Већина дефиниција података наглашава шта подацима фали, да ли је то значење или вредност, затим да су неорганизовани и непроцесирани и тиме дају основу за дефинисање информација у односу на њих.
- Информације су подаци организовани или структурирани. Путем ове обраде подаци постају релевантни за одређену сврху или контекст и постају смислени, вредни, корисни и релевантни.
- Знање се може видети као микс информација, разумевања, способности, искуства, вештина и вредности, али све анализирани дефиниције не обухватају све ове чиниоце.

За потребе овог рада усвојићемо следеће дефиниције.

Према групи аутора (Vocij et al., 2015, pp. 7–8) подаци су сирове чињенице или опсервације за које се сматра да имају мало или немају вредности све док се не процесирају и трансформишу у информацију. Када је податак смислен и адекватно интерпретиран од стране примаоца, онда се назива информација и уобичајено је да се користи у неку сврху.

Група аутора (Rainer et al., 2013, p. 15) назначавача да се знање састоји од података и/или информације које су организоване и процесирани да пренесу разумевање, искуство, акумулирано учење и експертизу приликом решавања тренутног пословног проблем. Мудрост је дубоко, богато разумевање и увид који се уобичајено развија путем комбинације екстензивног знања (знати) и личног искуства (радити) током времена. Оно расте узajамним дејством постојећег знања, новог знања (извученог из нових информација) добијено путем учења и комуникације са другим људима који знају, практичним искуствима, размишљањем, и тд (Spitzer, 2007, p. 106). На основу претходних разматрања могуће је успоставити ПИЗМ модел уз разумевање његових појединачних чинилаца.



Слика 2. ПИЗМ модел, преузето (Vocij et al., 2015)

Размотримо различите дефиниције информационих система у литератури.

„Информациони систем може бити било која организована комбинација људи, хардвера, софтвера, комуникационих мрежа, ресурса података и политика и процедура која складишти, враћа, трансформише и распрострањује информације у организацији“ (Marakas & O’Brien, 2010, p. 4).

„Информациони систем је уређење људи, података, процеса и информационих технологија које међусобно дејствују на прикупљању, обради, складиштењу и пружању информација потребних за подршку организацији“ (Whitten & Bentley, 2005, p. 6).

„Информациони систем може бити технички дефинисан као скуп повезаних компоненти које скупљају (или враћају), обрађују, складиште и дистрибуирају информације да би подржале доношење одлука, координацију и контролу у организацији. Поред тога, информациони системи могу такође да помажу менаџерима и радницима да анализирају проблем, визуелизују сложене теме и креирају нове производе“ (Laudon & Laudon, 2017, p. 13).

„Пословни информациони систем је група међусобно повезаних компоненти које раде заједно да обаве улаз, обраду, излаз, складиштење и контролне акције у циљу претварања података у информационе производе које могу да се користе у подршци предвиђања, планирања, контроле, координације, одлучивања и оперативних активности у организацији“ (Bosij et al., 2015, p. 42).

„У рачунарски-заснованом информационом систему, рачунари скупљају, складиште и процесирају податке у информације према инструкцијама које спроводе људи путем рачунарских програма“ (Sousa & Oz, 2014, p. 13).

„Информациони систем је комбинација људи и информационе технологије која креира, скупља, процесира, складишти и дистрибуира корисне податке“ (Valacich & Schneider, 2017, p. 16).

„Информациони систем скупља, процесира, складишти, анализира и шири информације ради посебне намене. Сврха информационог система је била дефинисана као добављање праве информације правим људима у прво време, у одговарајућој количини и у исправном формату“ (Rainer et al., 2013, p. 7).

„Информациони системи су рачунарски засновани системи који могу да приступе различитим рачунарски-добијеним, -складиштеним или -генерисаним подацима, и да одаберу и процесирају те податке да би обезбедили специфичне информације за помоћ пројектантима и менаџменту у њиховом одлучивању“ (Wiederhold, 1992, p. 1).

„Информациони системи су системи које организације користе да их подрже у доношењу одлука. Ови системи обезбеђују информације свим нивоима менаџмента у организацији“ (Leonard & Van Zyl, 2018, p. 787).

„Информациони системи су рачунарски или нерачунарски системи дизајнирани и изграђени да задовоље потребе за информацијама корисника система“ (D’Ambra et al., 2014, p. 2307).

Посматрајући наведене дефиниције, а и многе сличне Оливе (Olivé, 2007, p. 1) износи проблеме и дискусију у вези дефинисања појма информационих система, које се односе на 3 перспективе из којих су дефиниције дате: доприноса које информациони системи дају, њихове структуре и понашања, и функција које извршавају. За дефиниције посматране из перспективе структуре и понашања, исти аутор наводи два важна коментара (Olivé, 2007, p. 3):

- потребно нагласити да су информациони системи пројектовани од стране инжењера, што их разликује од других класа система који имају функције процесирања, као је нпр. људски ум у когнитивним наукама
- већина дефиниција не ставља ограничење о којем типу информација се ради, па се може помислити да чак и факс машина може да се третира као информациони систем, јер прихвата документа, чува их, складишти их, преводи их и шаље путем телефонске линије, зато информациони систем мора да путем информација које обрађује упути на стање одређеног домена (енг. *domain*)

Имајући у виду напред наведене дефиниције које су из области информационих система и менаџмента знања, а при томе уважавајући претходну дискусију, који посматра феномен са аспекта концептуалног моделовања (енг. *conceptual modeling*) за потребе рада ће се корисити следећа дефиниција:

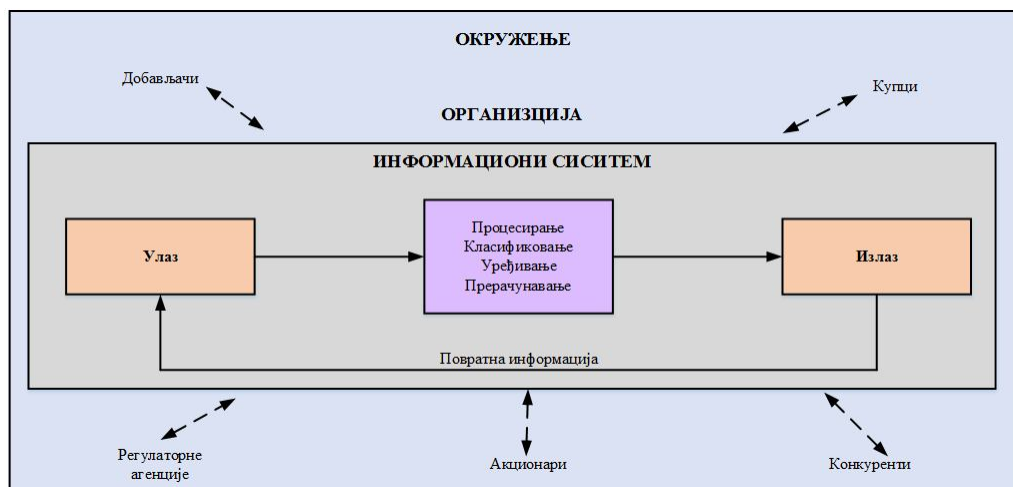
Информациони систем је систем који доприноси пословању организације обављањем свакодневних операција и давањем подршке у одлучивању, путем међусобног дејства информационих технологија, људи и процедура подешених за скупљање, манипулисање, складиштење и процесирање података у информације путем којих се указује, одржава и мења репрезентација стања домена за који је систем пројектован и изграђен.

2.1.2. Компоненте и активности информационих система

Према ауторима (Marakas & O’Brien, 2012, p. 27) информациони систем се састоји из следећих међусобно повезаних компоненти: људи, хардвера, софтвера, периферних уређаја и мрежа, где међу њима постоје јасно дефинисане границе: функције, модули,

типови апликација, одељења и корисничке групе. Навођењем компоненти ИС дефинише се његова структура.

У остваривању своје сврхе, а посредством дејства својих компоненти, информациони систем користи три активности у организацији за коју је прављен и окружења у којем делује, а оне су улаз (енг. *input*), процесирање (енг. *processing*) и излаз (енг. *output*) (Laudon & Laudon, 2018, р. 44). Улазима се означавају прикупљени подаци унутар организације или њеног спољног окружења, где се у процесирању добија форма података која има значење, а затим се те информације прослеђују корисницима. Повратном спрегом (енг. *feedback*) се обезбеђује да се на основу излаза исправе улази. Други аутори такође истичу важност повратне спреге, а при томе додају и контролу (енг. *control*), где заједно увећавају ефективности, а реализују се путем порука о грешкама, дијалога, шифара и управљања правима корисника (Marakas & O'Brien, 2012, р. 28). У пракси је веома важно уврстити принципе повратне спреге, јер на основу њих крајњи корисници имају бољи увид у своју интеракцију са системом и бенефите од његове употребе.



Са слике 3. уочавају се елементи окружења организације, а за потребе овог рада посматраћемо детаљнију класификацију према групи аутора (Wosij et al., 2015, р. 17) који окружењем организације, назива пословним окружењем и дефинише елементе окружења у односу на утицај који имају, где су утицајнији они који су ближе организацији и друге који су даље су мање утицајни. При томе они их стављају неколико класа: специфично

окружење (енг. *specific environment*), генерално окружење (енг. *general environment*) и физичко окружење (енг. *physical environment*).

Организација најпре делује унутар специфичног окружења којег чине: запослени са својим односима, купци, добављачи и конкурентне организације. Након тога организација са својим специфичним окружењем делује унутар генералног окружења кога чине: економски фактори, технологија, правни оквир и јавно мњење. На крају организација са својим специфичним и генералним окружењем делује унутар физичког окружења кога чине: локација и метеоролошки фактор. Описани елементи окружења и односи организације са њима су приказани на слици 4.

Готово увек део информационог система је потребно да реализује одређена ограничења и правила која су део постојећих правних процедура, прописа и закона. Тако нпр. са изменом Закона о камати, потребно је изменити део информационог система који је пре тога обрачунавао камату на један начин и прилагодити га да у новим околностима је обрачунава на други начин.



Слика 4. окружење организације и главни фактори који утичу, преузето (Восиј et al., 2015)

2.1.3. Моделовање информационих система и њихов положај у реалном времену

У процесу моделовања настаје модел, а он је апстракција реалног система која искључује одређене детаље да би се дошло до одговарајућих сазнања релевантних за појаву која се посматра. Аутори (Раденковић et al., 1999, p. 2) наводе да је модел је упрошћена и идеализована слика реалности, он нам омогућава да се суочимо са реалним светом (системом) на поједностављен начин, избегавајући његову комплексност и иреверзибилност, као и све опасности (у најширем смислу те речи) које могу проистећи из експеримента над самим реалним системом. Постоји више врста модела (А. Marković, 2013): ментални (мисаони), вербални, физички, математички, аналогни, структурни (концептуални), рачунарски и симулациону. Честа подела је на: формалне и неформалне.

Неформални опис модела припрема се доста брзо и лако, али он најчешће није конзистентан и јасан, нарочито када су у питању сложени модели. Аномалије које се јављају приликом неформалног описа модела су: некомплетан опис модела, неконзистентан опис модела и нејасан опис модела (Раденковић et al., 1999, p. 5). Формални опис модела треба да обезбеди већу прецизност и потпуност у описивању модела, а понекад и формализацију поступка испитивања аномалија. Употреба формализама је веома важна, зато што се усмеравамо на оне аспекте који су релевантни за проблем (А. Marković, 2013). На тај начин се ствара основа да информациони систем на адекватан начин осликава везе између елемената унутар себе а и везе са елементима окружења, као и потребне интеракције између тих елемената.

Модел као производ процеса моделовања је веома битан алат који се користи приликом развоја информационог система. Он је важан је се путем њега могу описати интерна структура самог система, тз. статички аспект система, као и интеракција елемената система између себе, а и са околином, тзв. динамички аспект система. Моделовање се може користити у неколико различитих фаза пројектовања информационог система, у зависности од одабране методологије које ће се користити у развоју.

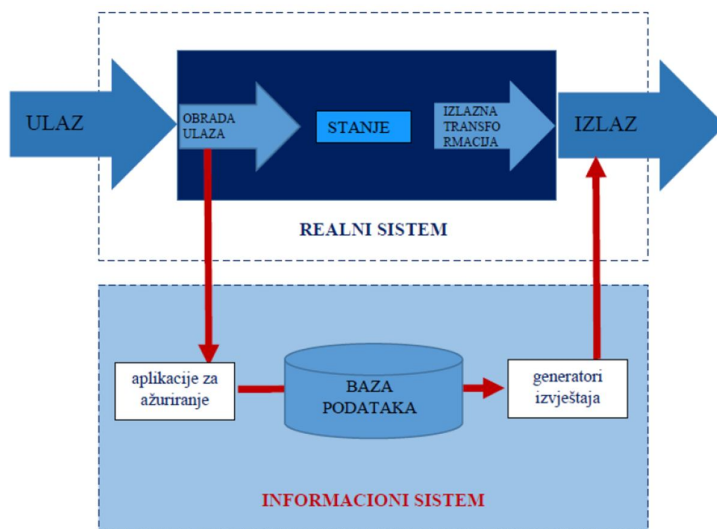
Аутори (Lazarević et al., 1985, p. 11) истичу да је код пројектовања информационих система потребно направити јасно разграничење између логичког и физичког пројектовања:

- логичко пројектовање укључује оне фазе пројектовања које су независне од опреме и других ресурса за развој ИС

- физичко пројектовање је задовољавајућа реализација логичког пројекта на конкретној опреми, односно задатих ресурса

Аутори (Lazarević et al., 1985, p. 15; Lazović & Bulatović, 2019, p. 12) описују положај информационог система у односу на реални систем. На слици 5. паралелно се виде улази, излази и стање система, како у реалном, тако и у информационом систему који представља апстракцију реалног система. Улази у систем мењају стања система, које описује како аутори наводе фундаменталне карактеристике система. У једном тренутку времена стање представља скуп објеката система, скуп њиховим међусобних веза и скуп вредности атрибута објекта у том тренутку времена.

Измасна трансформација дефинише неки начин мерења или посматрања динамичког понашања реалног система, где се генеришу излази на основу улаза и тренутног стања система (LabIS, 2000a, p. 6). На тај начин начин, као и у општој теорији система се успоставља веза између улаза и излаза, која се одвија преко стања.



Слика 5. положај ИС у реалном систему, преузето (Lazović & Bulatović, 2019)

На основу описа датог на слици 5., сваки модел информационог система мора да садржи две компоненте (Lazarević et al., 1985, p. 15):

- Модел података који преко скупа података и њихових међусобних веза приказује стање система у једном тренутку времена.

- Модел процеса (модел функција) који треба да предстаља структурирани скуп процеса који мењају стање система и помоћу којих се формирају излази система.

Модел података је интелектуално средство које служи за опис система у стационарном стању, а сваки модел података има 3 компоненте (LabIS, 2000a, p. 7):

1. структуру модела - скуп концепата за опис објеката система, њихових атрибута и њиховим међусобних веза
2. ограничења - семантичка ограничења на вредности података која у сваком стационарном стању морају бити задовољена (правила интегритета модела података)
3. операције над концептима структуре - преко њих је могуће описати динамику система у моделима процеса

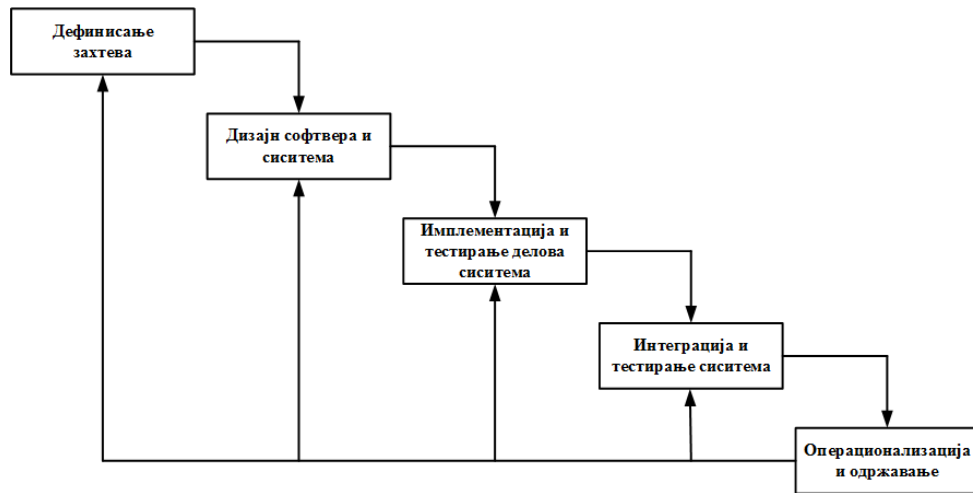
Модел процеса може се посматрати да са једне стране може да одговори на питање „шта систем треба а ради ?“ и тада је он логички модел процеса, док на питање „како систем треба да ради?“ дефинише се физички модел процеса, тј. имплементација логичког модела процеса у изабраном технолошком и организационом окружењу (LabIS, 2000b, p. 26). На тај начин се омогућава јасна сепарација активности који су платформски зависне од оних које су независне, што касније служи као идеја водила у развоју вођеног моделима. Обзиром да се моделовање примењује у једном или више фаза развоја једног информационог система, потребно је сагледати све фазе развоја система од његовог настанка па до његовог повлачења из употребе.

Аутори описују развој система од почетка употребе првих рачунара и идентификују неколико фаза у развоју информационог система, где је животни циклус развоја система, оличен у облику водопад модел, настао у склопу ране методолошке фазе као одговор на пре-методолошку фазу где су се сви напори решавали програмирањем (Aveson & Fitzgerald, 2006, p. 2). Животни циклус развоја система (енг. *Software Development Life Cycle*) је начин за описивање свих активности који се дешавају у развоју система. Поступак развоја једног система је сложен, па постоји потреба да се разбије на фазе, да би лакше извршили реализацију поменутог пројекта.

2.1.3.1. Водопад модел

Водопад модел је представио Ројс 1970. као методологију која прати цео циклус развоја производа или система. Развој је организован тако да се на наредни корак прелази тек пошто је претходни завршен, дакле секвенцијално је уређен и где животни циклус производа има 7 различитих корака, како је оригинално Ројс дефинисао, према (Roуse,

1970, р. 329): системски захтеви, софтверски захтеви, анализа, дизајн програма, кодирање, тестирање и операције. Овај модел је био веома популаран и зато има много варијација и са мање или више корака.



Према Сомервилу (Sommerville, 2016, р. 47), фазе су:

- анализа и дефинисање захтева - описује шта систем ради, ограничења система и циљеве који су постављени у интеракцији са корисницима система
- дизајн софтвера и система - дизајном се одређују захтеви система и архитектура система
- имплементација и тестирање делова система - дизајн софтвера се реализује као скуп програма или програмских делова. Тестирањем делова

Слика 6. Водопад модел, преузето (Sommerville, 2016)

се
проверава

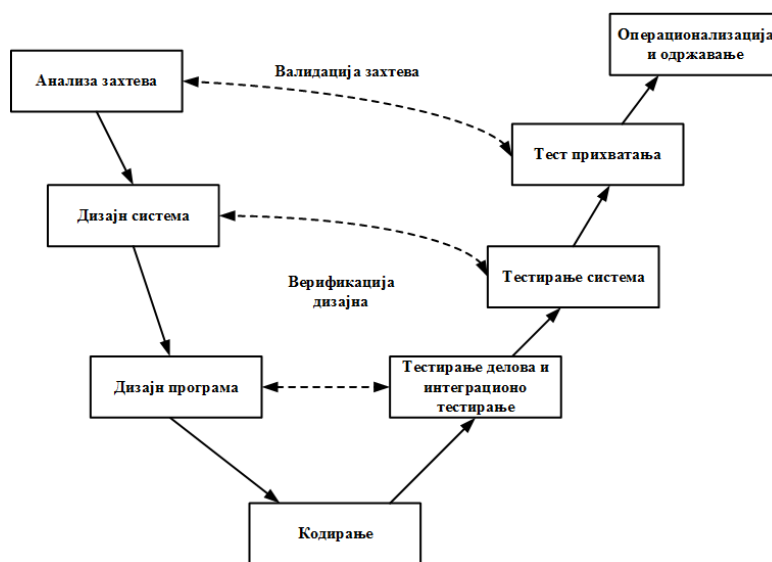
- да ли одговарају дизајну
- интеграција и тестирање система - програмски делови или сами програми су интегрисани и тестирани као комплетан систем да би се уверили да софтвер испуњава захтеве
- операционализација и одржавање - систем је стављен у употребу. Одржавање укључује и исправљање грешака које се нису откриле у ранијим фазама. Уобичајено је најдужа фаза у животном циклусу.

Једна од основних мана овог модела јесте његова секвенцијалност, где уколико настану промене у корацима пре него што посматрамо тренутну фазу, овај модел то не узима у обзир. Иако је имао велику популарност, поменута мана је дошла до изражаја са убрзаним технолошким развојем, као и потребе да се брзо испоручи нов производ (Isaias & Issa, 2015, р. 23). Корисници немају могућност за интеракцијом све до фазе тестирања, па није могућа корекција у грешкама које могу настати у раним корацима, које се веома тешко касније отклањају (Веђејски-Vујаклија, 2008, р. 13). Описани недостаци водопад модела су довели до развоја нових модела у развоју информационих сиситема.

2.1.3.2. V модел

V модел је варијанта водопад модела, уведена да би се обезбедио пораст квалитета кроз обезбеђивање везе свих фаза са њиховим фазама одржавања.

Аутори (Pfleeger & Atlee, 2009, р. 52) наводе да Unit тестови и интеграциони тестови проверавају исправност програма, при чему се сугерише да они могу да се искористе за проверу дизајна, а тестови прихватања проверавају исправност захтева, чиме се омогућава да је систем у потпуности имплементиран пре него што је прихваћен. За случај да је у некој фази, која се налази са десне стране уочен проблем, тада се преусмерава на леву страну овог модела и тамо се врше поправке, чиме се експлицитније исказују неке итерације и преправљања која су у водопаду скривена.



Слика 7. V модел, преузето (Pfleeger & Atlee, 2009)

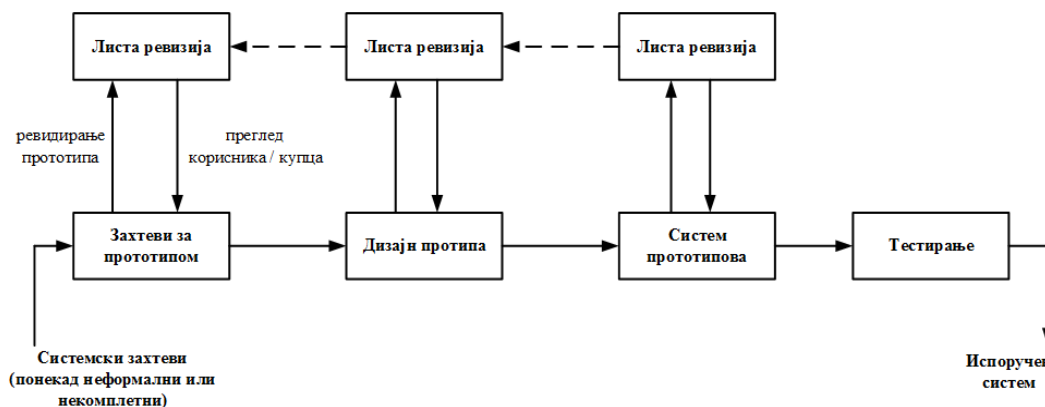
2.1.3.3. Инкрементални модел

Како модел водопада има веома дугачак пут од захтева до тесирања у којем корисници између осталог и евалуирају да ли су им захтеви испуњени, појавио се инкрементални приступ. Уместо да се цео поступак развоја од захтева до пуштања у рад спроводи над целим пројектом, инкрементални модел даје могућност да се део пројекта заврши и веома брзо испоручи корисницима на евалуацију, где они дају повратну информацију и тиме се грешке могу релативно брзо могу отклонити (Isaias & Issa, 2015, р. 25). За то време развија се следећи, део тзв. инкремент, којим се допуњује захтев који је реализован претходним инкрементом и у оквиру њега се поново спроводе секвенцијално дизајн, анализа, кодирање, тестирање.

Овај приступ је погодана за употребу када је: потребно брзо оспособљавање, међувремени производ је расположив за коришћење, систем се природно дели на инкременте, обезбеђивање средстава је инкрементално (Већејски-Vујаклија, 2008, р. 15). Идеје инкременталног модела су нашле своју примену у савременим агилним методологијама развоја.

2.1.3.4. Прототипски модел

Како аутор истиче прототип (енг. *prototype*) је рана верзија система која се користи за приказивање концепата, пробу дизајна и сазнавање више о проблему и његовим могућим решењима (Sommerville, 2016, р. 62).



Слика 8. Прототипски модел, преузето (Pfleeger & Atlee, 2009)

Прототипско моделовање је значајно јер се од самих корисника добијају повратне информације најчешће о томе да ли су захтеви адекватно формулисани или у интеракцији са корисничким интерфејсом да ли систем задовољава испуњава захтеване

функционалности. Уколико не задовољава, тада се врше одговарајуће ревизије (енг. *revision*) у фази за којој има потребе (Pfleeger & Atlee, 2009, p. 54). Када се ревизије нпр. у захтевима спроведу, то може да води и ревизију у дизајну, а затим и у самом кодирању. Комплетан поступак прототипског развоја укључује кориснике у интеракцију са прототиповима да би се релативно рано отклонили ризици и смањиле грешке.

2.1.3.5. Рапидни модел

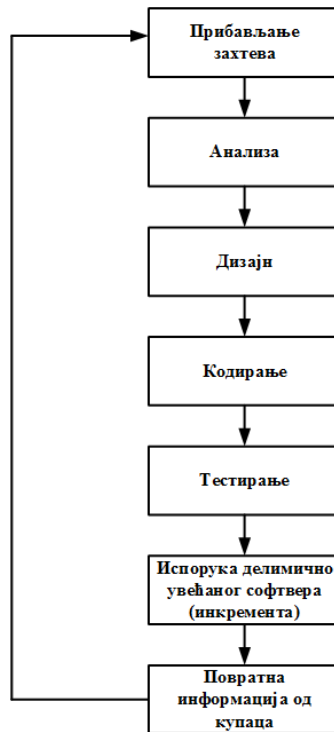
Рапидан развој или како се назива рапидан развој апликација (енг. *rapid application development*) настао је са идејом да је потребно обезбедити брз развој, где досадашњи приступи то нису омогућили. На тржишту, где је заострена борба између конкурената, потребно је остварити предност, а за то је потребан брз развој.

Уобичајено код рапидног модела је да постоје три или четири циклуса у развоју и то су захтеви, планирање, дизајн, конструкција и пресек (Isaias & Issa, 2015, p. 30). Код рапидног развоја је у фокусу брз развој употребом софтверских алата који то омогућавају и стална комуникација са клијентима, док се целокупан производ не заврши. Такође важно је истаћи да уколико је пројекат обимнији, услед тога што се мало времена спроводи у планирању и моделовању система, то може да наруши дизајн система и да уместо да се пројекат заврши на време, заправо да се веома продужи.

2.1.3.6. Агилни модел

Агилне методологије настале су око концепта да је дизајн кључна активност и уважавају сталну промену система и потребне измене, тзв. адаптације, које је потребно спровести услед различитих промена, па је потребно заузети флексибилнији приступ.

Аутори у дискусији наводе да су агилне методе више адаптивне него што су предиктивне, а поред тога теже бржем развоју и већем задовољству корисника, услед већег испуњавања њихових потреба (Isaias & Issa, 2015, p. 31). Група аутора (Sharma et al., 2012, p. 893) је дала уопштени приказ агилног поступка на слици 9. Агилне методологије препознају промену као нужност и у складу са тим не ограничавају кориснике када ће захтевати промене у сиситему, већ теже да дизајн учине довољно флексибилним да чак и у касним фазама развоја, уколико се укаже потреба за променом, оне то могу да подрже. Због тога су нашле своју широку примену у софтверској индустрији.



Слика 9. Фазе агилног развоја (Sharma et al., 2012)

Принципи које су усвојени у оквиру агилног манифеста (енг. *agile manifesto*) су следећи (Beedle et al., n.d.):

- Приоритет је задовољство клијента путем ране и континуалне испоруке софтвера који има вредност.
- Поздрављамо промене захтева, чак и касније у развоју. Агилни процес укључује промену ради клијентове конкурентске предности.
- Испоручујемо софтвер који ради често, за неколико недеља или месеци, префереирањем краћег временског распона.
- Пословни људи и људи који развијају (енг. *developers*) морају да раде заједно на дневном нивоу током целог пројекта.
- Пројекти су организовани око мотивисаних појединаца.
- Дајте им окружење и подржите их и верујте им да ће завршити посао.
- Најефикаснија и најефективнија метода преношења информација ка или унутар тима је конверзација лицем у лице.
- Софтвер који ради је главна мера напретка.

- Агилни процеси промовишу стабилност у развоју. Спонзори, људи који развијају и корисници би требало да непрекидно одржавају константан темпо.
- Континуална усмереност на техничку изврсност и добар дизајн унапређује агилност
- Једноставност - уметност максимизирања количине посла који се не ради - је есенцијална.
- Најбоље архитектуре, захтеви и дизајн настају од само-организованих тимова.
- У редовним интервалима, тим осмишљава како да постане ефикаснији, затим подешава и прилагођава своје понашање према томе.

У наставку ће бити презентовано неколико агилних методологија: Scrum, Kanban и Екстремно програмирање.

2.1.3.6.1. Scrum

Scrum припада агилним методологијама путем које се остварује висок ниво продуктивности, а заснован је на концепту инкременталног моделу развоја. Код Scrum приступа, цео пројекат развоја се састоји из серије итерација које се називају sprint, који максимално може да траје 30 дана (Sharma et al., 2012, p. 895). Корисници дају захтев за израдом софтвера, који не мора бити потпун, већ ће временом додавати одређене функционалности. Од њих се не очекује да у старту знају све што им је потребно. Следећа фаза јесте да се успостави листа свега што је потребно у производу на основу захтева која се назива product backlog. Лице које је означено као Product Owner је одговорно за Product Backlog, укључујући његов садржај, доступност и редослед, тј приоритет (Scrum.org, n.d.). Сваког дана се спроводе састанци у вези Sprint-а и добијају се повратне информације, шта и колико је урађено. Sprint функционише као инкремент, тако да када се сви Sprint-ови заврше добијамо софтвер који ради.

2.1.3.6.2. Kanban

Kanban је методологија за управљање радним током (енг. *workflow management*) осмишљена да визуелизује рад и повећа ефикасност. Иницијално је настала у Јапану у фабрикама Тојоте, а сам појам Kanban значи огласна табла (енг. *billboard*) или картица коју можете видети (Boutin, n.d.). За разлику од приступа у којем се производи одређена количина добара за тражиште, уводи се пракса да је производња заснована на тражњи купаца. Такав приступ је постави основу Lean производње, чија је сврха да смањи непотребне активности без да утиче на продуктивност.

Основни циљ је да се креира већа вредност за клијенте без додатних трошкова (kanbanize.com, n.d.), при чему су основни принципи Kanban приступа следећи:

- Принцип 1: Започни са оним што радиш сада - Kanban може да се укључи на постојеће радне токове и да их временом побољша без опасности да процес рада буде ометен. Не постоји бојазан од културног шока.
- Принцип 2: Сложите се са инкременталном, еволутивном променом - Kanban у својој сржи има мале и еволутивне промене тренутних процеса, које су осмишљене тако да пруже најмањи могући отпор.
- Принцип 3: Поштуј тренутни процес, улоге и одговорности - Kanban поштује постојеће процесе, улоге, одговорности и титуле јер они имају вредност. Није потребно изазивати страх од промена и зато оне морају да буду инкременталне и логичке.
- Принцип 4: Охрабрите лидерство на свим нивоима - Наглашава се улога лидера у тиму. Такође, важно је да сви усвоје принцип континуалног побољшања да би остварили оптималне перформансе на свим нивоима управљања.

Популаран алат за организацију рада по Kanban методологији је Trello, аустралијске корпорације Atlassian.

2.1.3.6.3. Екстремно програмирање

Екстремно програмирање (енг. *extreme programming*) је метода агилног развоја која максимално наглашава укључивање корисника у интеракцију приликом развоја и омогућава корисницима да у било којем тренутку изврше промену захтева (Sharma et al., 2012, p. 894). За ефективну примену овог модела у пракси важно је да се оствари добра комуникација, једноставност, повратне везе и да развојни тим (или чак пар) буде одважан у многобројним стварима које ће морати да мења како пројекат буде текао.

Постоји одређени скуп правила које би требало применити приликом у циљу успешног завршетка пројекта екстремног програмирања (Powell-Morse, n.d.):

- Планирање
 - Корисничке приче (енг. user stories) су написане.
 - Планирање верзије (енг. release) креира распоред верзија.
 - Често креирајте мале верзије.
 - Пројекат је подељен у итерације.
 - Планирање итерација започиње сваку итерацију.

- Управљање
 - Дајте тиму одговарајући радни простор.
 - Успоставите одржив темпо.
 - Спроводите састанке (енг. *stand-up meeting*) сваки дан.
 - Брзина пројекта се мери.
 - Померајте људе.
 - Поправите екстремно програмирање када не функционише.
- Дизајнирање
 - Једноставност.
 - Одаберите метафору система.
 - Употребљавајте CRC картице (енг. *CRC cards*) као брејнсторминг алате током сесије у вези дизајна.
 - Креирајте мале експерименте који треба да дају одговор на проблем (енг. *spike solutions*) да би сте смањили ризик.
 - Функционалности не додајте рано.
 - Спроводите рефакторисање (енг. *refactoring*) кад год је то потребно.
- Кодирање
 - Клијент је увек доступан.
 - Код мора да буде написан у договореном стандарду.
 - Прво правите unit тестове.
 - Цео продукциони код је кодиран у пару (енг. *pair programming*).
 - Само један пар интегрише код у једном тренутку времена.
 - Интегришите често.
 - Поставите наменски рачунар за интеграцију.
 - Користите колективно власништво (енг. *collective ownership*)
- Тестирајте
 - Сав код мора да буде тестиран.
 - Сав код мора да прође све unit тестове пре него што може да буде стављен у продукцији.
 - Кад је пронађен баг тестови се креирају.
 - Тестови прихватања се покрећу често и објављује се њихов резултат.

Овај приступ је добро применити када постоји мали број високо мотивисаних појединаца. Једна од варијација овог приступа је програмирање у пару.

2.1.4. Ресурси информационих система

Као што је раније поменуто, информациони систем је изграђен употребом следећих целина: информационих технологија, људи и процедура. Саме ИТ технологије обухватају (Bourgeois, 2014, р. 6): хардвер, софтвер, базе података и комуникационе технологије (које се понекад изостављају јер су изграђене од хардвера и софтвера). Неки аутори их називају и ресурсима информационих система (Marakas & O'Brien, 2012, р. 29). Треба имати у виду да су могуће и другачије класификације од наведене.

Хардвер представља физички опипљиве делове технологије: стони и преносиви рачунари, мобилни телефони, таблети, електронски читачи, делови за складиштење као што је преносива меморија, затим улазни уређаји као што су тастатура, миш, секенер и излазни уређаји као што су штампачи, звучници.

Софтвер чини скуп наредби који говори хардверу шта да ради, а који се групише на: оперативне системе, који омогућава употребу хардвера и апликациони софтвер, који ради нешто корисно. Сам апликациони софтвер, користи се ради постизања неког циља, а може бити неки специјализовани алат (нпр. антивирусни софтвер) и програмерски софтвер. Такође, важан део су подаци, који се користе употребом база података. Базе података су уређене колекције података, које постоје релативно дуго и које користе и одржавају већи број корисника, односно програма (Lazarević et al., 2006, р. 1). Обзиром на дугорочну примену релационих система за управљање базом података у пракси, пре свега у пословном домену, ове технологије се сматрају једне од најстабилнијим.

Комуникационе технологије су омогућиле размену података са различитих локација. Оне се у предузећима користе у форми локалне мреже (енг. *local area network*) или Интернета. Оне могу садржати следеће мрежне хардверске компоненте као што су: гејтвеј (енг. *gateway*), мост (енг. *bridge*), усмеривач или рутер (енг. *router*), прекидач или свич (енг. *switch*), модем, каблове и тд. За потребе комуникације мрежних уређаја извршавају се мрежни софтвери који омогућавају да се спроведе сложен процес комуникације између различитих врста мрежне опреме и рачунара. Мрежна комуникација претпоставља употребу модела комуникације, где су данас најпознатији референтни модели ISO/OSI модел и TCP/IP модел. Поменути модели се могу посматрати као одређена архитектура која има своје делове који се зову слојеви (енг. *layers*) где се сваки слој може посматрати као услуга (енг. *service*) коју нуди слоју изнад себе. На сваком слоју користе се комуникациони протоколи путем којих се дефинише формат и редослед порука, размењених између најмање две стране које учествују у комуникацији, као и поступци који се предузимају после слања и/или пријема порука или неког другог

догађаја (Kurose & Ross, 2009, p. 9). У надлежности одговарајућег мрежног софтвера су сви слојеви (апликативни (енг. *application layer*), транспортни (енг. *transport layer*), мрежни (енг. *network layer*), слој везе (енг. *data link layer*) осим физичког (енг. *physical layer*) који је задужен за пренос на физички медиј.

Људи су веома важан фактор успешног функционисања свих информационих система. Они могу бити крајњи корисници (енг. *end users*) и ИС специјалисти (енг. *IS specialists*) (Marakas & O'Brien, 2012, p. 29). Крајњи корисници, још се називају клијенти, су сви они људи који користе информациони систем. ИС специјалисти су људи који развијају и управљају информационим системом.

Процедуре су стратегије, политике, методе и правила за употребу информационог система (Stair & Reynolds, 2018, p. 9). Када корисник ради одређени посао он прати одговарајуће кораке да би оствари резултат. Када су процедуре добро документоване крива учења је мања и смањују се додатни трошкови тренинга.

2.1.5. Врсте информационих система

Како се ИС развијају за различита предузећа, чији су послови различити, тако један ИС не може да задовољава потребе свих. У сладу са тим постоји више врста информационих система. Према основној класификацији коју даје Маракас (Marakas & O'Brien, 2012, p. 12) они могу бити информациони системи за подршку операцијама (енг. *operations support systems*) и информациони системи за подршку менаџменту (енг. *management support systems*). За потребе овог рада поменута класификација је одговарајућа, јер је веома примењива на већину пословних система, поготово у онима који обједињују неколико врста, као што су нпр. водоводна предузећа, електродистрибуција, топлане и тд, услед великог варијитета пословних процеса у свом пословању.

2.1.5.1. Информациони системи за подршку операцијама

Процесирања пословних трансакција, контролисање индустријских процеса, подршка комуникацијама и сарадњи и ажурирање база података предузећа су основне активности ове класе информационих система чиме се обезбеђује подршка пословању (Marakas & O'Brien, 2012, p. 13), а они могу бити:

- системи за процесирање трансакција (енг. *transaction processing systems*) који дају подршку пословним трансакцијама, креирају одређене пословне извештаје (документе) и ажурирају стање у бази података. Примери ових

система су: књиговодствени систем, систем продаје и наплате, материјално и магацинско пословање, и тд.

- системи за контролу процеса (енг. *process control systems*) путем којих се врши контрола и надзор индустријских процеса. Примери могу да укључе системе за генерисање струје, прераду нафте и тд.
- системи за сарадњу (колаборацију) предузећа (енг. *enterprise collaboration systems*) путем којих се остварују комуникација и колаборација. Примери су електронска пошта (енг. *e-mail*), системи за размену порука (енг. *chat*), системи за видео конференције итд.

Нека предузећа имају потребу да обједине више напред наведених врста ИС, па је за њих изузетно важан аспект интеграције поменутих система, да би се остварила висока ефикасност пословних процеса.

2.1.5.2. Информациони системи за подршку менаџменту

Улога менаџмент информационих система за подршку менаџменту јесте да се свим менаџерима без обзира на ниво хијерархије омогући подршка у доношењу одлука (Marakas & O'Brien, 2012, p. 14), а они могу бити:

- Менаџмент информациони системи (енг. *management information systems*), скраћено MIS, који прослеђују информације у облику предефинисаних извештаја и на тај начин пружају подршку одлучивању. Примери могу бити: анализа продаје, перформансе у производњи и тд.
- Системи за подршку одлучивању (енг. *decision support systems*), скраћено DSS, који подржавају интерактивно ad hoc извештавање. Примери су: системи за анализу ризика, предвиђање профитабилност и тд.
- Информациони системи извршних директора (енг. *executive information systems*) обезбеђују значајне информације од MIS-а и DSS-а, пословне интелигенције (енг. *business intelligence*), скраћено BI, и других извора путем којих се потребне информације прослеђују извршним директорима. Примери: системи за лак приступ анализама пословних перформанси, акција конкурената и економског развоја ради подршке стратешком планирању.

За успешну примену ИС у процесима доношења одлука изузетно је важно да све поменуте врсте система за подршку менаџменту буду међусобно усаглашене и део

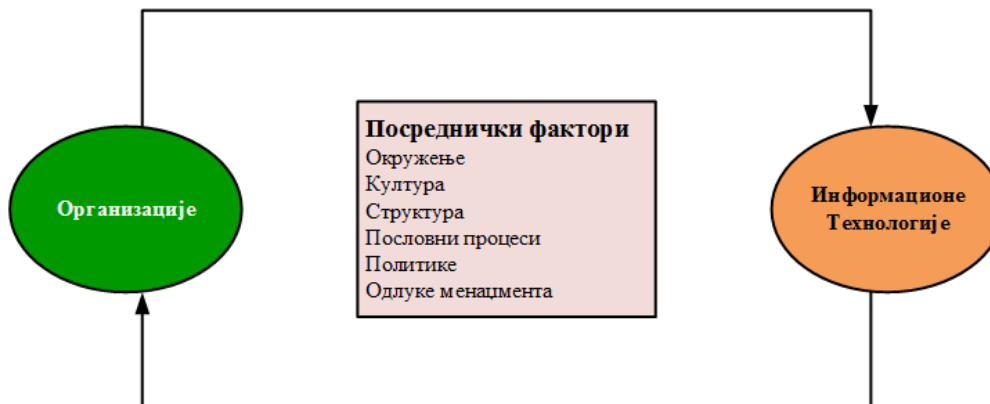
једног већег сиситема, тако одлуке које долазе са различитих нивоа се подржавају међусобно.

2.2. Интеракција информационих система и организације

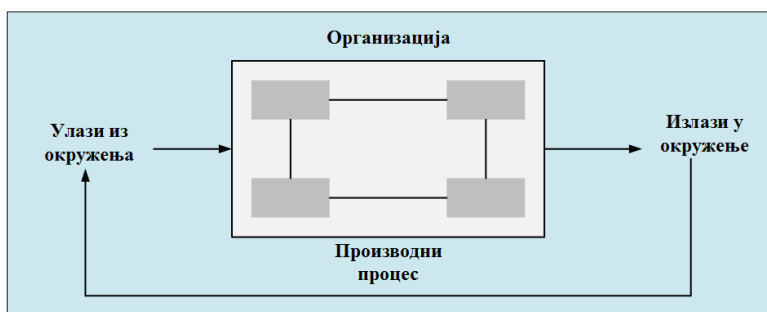
Како информациони системи утичу на организације, тако и организације утичу на информационе системе, а овај узајаман и испреплетан однос уједно је и сложен. Са једне стране организација има своју мисију, визију и циљеве и користе информационе системе да би их остварила, тако што они подржавају свакодневне активности и дају подршку у одлучивању.

Менаџмент организације утиче на развој информационих система, постављањем нових захтева пред ИТ сектор, јер њега види као веома важну карику у својем пословању. Са друге стране, са развојем технологија настају нове могућности које омогућавају ефикасније обављања послова, па је због тога важно са аспекта менаџмента бити отворен у примени нових технологија и омогућити да информациони систем своје улоге испуни још боље.

Сложена интеракција између информационих технологија и организације је под утицајем многобројних фактора укључујући организациону структуру, пословне процесе (енг. *business processes*), политику, културу, околинину и одлуке менаџмента (Laudon & Laudon, 2018, p. 109). Међусобно дејство је приказано на слици 10.



Према ауторима (Laudon & Laudon, 2017, р. 110) са техничког и микроекономског аспекта, организација се посматра слично систему са својим улазима и излазима, па су са тог становишта „улази“ у организацију рад и капитал, где се путем производног процеса („трансформације“) креирају производи и услуге („излази организације“). Како организација путем својих „излаза“ делује на окружење употребом производа и услуга тј. тај „излаз“ делује поново на „улаз“ тј. капитал и рад. Дакле, према овом тумачењу остварује се повратна веза између производа и услуга организације као „излаза“ и рада и капитала као „улаза“. Овај модел организације је приказан на слици 11.



Слика 11. техничка дефиниција организације, преузето (Laudon & Laudon, 2017)

Други аспект посматрања је бихевијорални, где се улази у детаљније објашњење организације и њено структурирање, да би се сви релевантни елементи везани за информациони систем видели, јер не може организација да буде само „трансформација улаза у излазе“. Стога поглед аутора (Laudon & Laudon, 2018, р. 110) на организацију јесте да је она колекција права, привилегија, облигација и одговорности деликатно уравнотежених током времена путем конфликта и решавања истих и издвајају две важне карактеристике:

- структуру - чине хијерархија, подела рада, правила и процедуре, пословни процеси и културу
- процесе - чине права и обавезе, привилегије и одговорности, вредности, норме и људи

Ова два вида посматрања дају основу за адекватно сагледавање сложеног односа интеракције ИС и организације.

2.2.1. Информациони системи и нивои управљања у организацијама

Информациони системи имају улоге на различитим нивоима управљања у организацији у односу на потребне информације које су потребне запосленима и

менаџерима у циљу доношења одлука. Разликујемо следеће нивое (Vocij et al., 2015, p. 20):

- стратешки ниво - менаџери који доносе одлуке на стратешком нивоу креирају дугорочне планове, које најчешће имају значајан утицај на организацију. Такви типови одлука нису честе и немају структурирану форму.
- тактички ниво - менаџери који доносе одлуке на тактичком нивоу креирају средњерочне планове, а приликом остваривања циљева дефинисаним тим плановима тежи се и остварењу дугорочних организационих циљева. Они контролишу буџет, надгледају резултате и перформансе за које су задужени.
- оперативни ниво - менаџери који доносе одлуке на оперативном нивоу креирају краткорочне планове, а приликом остваривања тих циљева ствара се основа за постизање циљева и политика дефинисаним на тактичком нивоу, тј. тежи се остварењу средњерочних циљева. Те одлуке који доносе су по својој природи структуриране. Поред тога, они се баве контролисањем дневних организационих активности.

Одлуке које се доносе на напред наведеним нивоима је потребно да буду усаглашене међусобно, стога ИС треба да понуди интегрално решење које то омогућава.

2.2.2. Нове улоге информационих система у организацијама

Захваљујући технолошком напретку, нове технологије имају могућност у много да промене пословање организација, са једне стране оне могу да утичу на унапређење свакодневних активности које организације обављају, а поред тога оне могу да побољшају доношење одлука. Оба ове улоге информационих система су под утицајем технолошког напретка и на нове и специфичне начине мењају како ефикасност саме организације, нпр. побољшањем производа и услуга који се испоручују купцима, тако утичу и на њене друге организационе елементе, нпр. смањују хијерархију.

Иновације у информационих технологијама довеле су до нових технологија или напретка у истим. Ту се пре свега мисли на рачунарство у облацима, технологије великих података, интернет ствари, напредна аналитика, пораст употребе мобилних технологија и тд. Рачунарство у облацима (енг. *cloud computing*) омогућава да се гомила сервиса која је организација развила са своје клијенте са своје интерне инфраструктуре, пребаци на

облак и да се учини да они буду расположивији, поред тога да се смање трошкови одржавања инфраструктуре, као и да се омогући веома висока доступност сервисима (скоро 24 часа), затим лакше и једноставније је управљати порастом клијената, а поред тога лакше и ефикасније је могуће извршити ажурирање свих верзија софтвера и учинити да последња доступна верзија је увек расположива за клијенте. Рачунарство у облацима има модел „плати колико трошиш“ (енг. *pay as you go*), за разлику од традиционалних рачунарских центара, где је потребно прво расписати набавку за новом опремом, сачекати да она стигне, инсталирати све потребне софтвере, потенцијално је интегрисати са другим деловима система и тек на крају пустити у рад, а при томе може се десити да је њено искоришћење веома мало. У односу на то облак је много прилагодљивији и организација плаћа само оно колико заиста троши. Интернет ствари (енг. *internet of things*) настале под утицајем глобалног смањења цене сензора који се производе и развоја рачунара са малим плочама (енг. *single board computing*) омогућавају већи увид у пословне активности, као и побољшање задовољства купаца. Синергијом ове технологија са технологијом облака настаје много нових могућности за пословање, који до сада нису били могући или нису били исплативи по прихватљивој цени. На примеру сценарија паметних градова (енг. *smart city*) крајњи корисник може да добије детаљнији увид у стање инфраструктуре коју користи: електрична енергија, водовод, гас, и тд. Организација може да понуди нове сервисе који до тог тренутка нису били могући. Нпр. један од сценарија може да буде детекција аномалија у потрошњи електричне енергије, путем синергије интернета ствари, рачунарства у облаку и технологије великих података и напредне аналитике, могуће је направити сервис који анализира потрошњу клијента, и када та потрошња одступи од неке уобичајене, може послати тим људи на терен да провере за чему се ради, при чему они могу да утврде нпр. да се ради о крађи електричне енергије.

Технологије великих података (енг. *big data*) омогућавају да се спроведу напредније управљање подацима у ситуацији када имамо велики обим података, брзину и различитост, где стандардне технологије база података нису у стању да то ураде. На тај начин стварају се могућности да се интегришу многобројни извори података, како они који су доступни унутар организације, тако и оних екстерних. Кроз адекватну анализу коју је могуће спровести заједно са напредном аналитиком, могуће је пословању на свим нивоима управљања дати нове информације, које се могу употребити на стратешком, тактичком и оперативном нивоу. Поред тога што је могуће да ту детаљнију анализу на одређени проблем у неком времену, постоји и могућност да се делови такве анализе учине и доступним у реалном времену, што може бити веома значајно, јер се ствара

простор за брзу реакцију организације, а тиме и могућност да се оствари конкурентска предност.

Напредна аналитика омогућава да се створе бољи увиди у пословање, да се омогући побољшање постојећих начина рада и створи простор за иновације. Организације имају прилику да испитају до сада увојене начине рада, њихове претпоставке и одређене стандарде које се примењују и испитају њихову релевантност за пословни проблем који обрађују. Генерисањем увида у пословање, ствара се и могућност да се оно измени, а све је на менаџменту како ће да употребљава резултате поменутих анализа. Са једне стране они морају да воде рачуна да се иновације усвоје и спроведу због остваривања мисије, визије и циљева организације, док са друге стране они знају да одређене технологије могу да имају тзв. дисруптивни утицај, јер су у стању да потпуно избршу одређене послове, па из разлога да се не створе велики потреси у организацији и тиме пословање буде угрожено, пажљиво треба управљати поменутим.

Пораст употребе мобилних телефона, омогућава организацијама да са адекватно развијеним мобилним апликацијама омогуће готово тренутни приступ свим својим сервисима крајним клијентима, чак и на географски удаљеним локацијама. Једна од апликација која је нашла своју примену, јесте апликација за мобилно банкарство, где се готово све могућности које клијент може да изврши излажу путем функционалности у самој апликацији, од тога да он може да види стање по рачуну, прегледа и креира уплате и исплате, затим да и ка банци упути одређене захтеве нпр. за кредит, поред тога он има могућност да и врши трансфер новчаних средстава са рачуна на рачун и тд. Таквим начином употребе клијент штеди и време и новац, његово корисничко искуство се повећава, ефикасност у продаји производа и услуга се увећава, смањује се папирологија, имплементирају се нове стратегије мобилног маркетинга и одређеног персонализованог садржаја и услуга које се нуде. Поред тога све је то веома доступно путем Интернета, чиме клијенти имају могућности да 24 сата у недељи користе поменуте сервисе.

2.2.3. Значај информационих технологија за земље у развоју

Познато је да информационе технологије значајно могу да утичу на развој, како развијених, тако и земаља у развоју. Како наводе аутори (Stankić et al., 2018, p. 62) за економски раст је важно остварити садејство информационих технологија, људских ресурса и инфраструктуре, а за земље у развоју то је посебно значајно јер их уводи у

трансформацију и процес глобализације. Услед наведеног постоји потреба за дефинисање интегралне стратегије развоја која то може да омогући.

Према аутору Ђурђевић основна функција инфраструктуре је да изврши пренос људи, материјалних добара, информација и енергије, при чему постоји техничка инфраструктура: водовод, канализација, електроенергетика, комуникација, даљинско грејање, гасоводни систем, саобраћај и тд. и друштвена инфраструктура: друштвене и социјалне институције, објекти за боравал деце, библиотеке, ватрогасне станице и тд. (Ђурђевић, 2009, р. 11). Инфраструктура је веома значајна за сваку државу и друштво у целини и стога примена ИТ треба да буде таква да побољша живот становништва, а са друге да испуни одређене интересе које свака држава исказује и да подстицајно делује у многим сегментима у којима је то могуће.

Уколико погледамо утицај информационих технологија на инфраструктуру, може видети да је он веома велики. Пре свега, техничка инфраструктура најчешће користи неколико различитих врста информационих система. Поред система за процесирање трансакција који је уобичајено у основи сваког пословања, јер се путем њега остварују неке кључне функције организације као што су продаја и наплата, организација користи и системе за контролу процеса путем којих се контролишу разни технолошки процеси: нпр. процес прераде воде. Такође ова врста предузећа уједно и користи информационе системе за подршку одлучивању. Јасно је да ефикасност једне инфраструктуре је у директној вези са информационим технологијама, јер се веома ослања на њих. Обзиром да је већина ових информационих система, технолошки разнородних, има различиту функцију унутар неких предузећа, а да је суштина пословања у вредности која се креира за клијенте, јасно је да од тога колико ће добро поменуте функције сарађивати је у директној спрези са тиме колико ће њихови информациони системи као њихови модели заједно функционисати. Из поменутих разлога потребно је посебно обратити пажњу на интеграцију поменутих система, као што су: пословни информациони систем, SCADA као систем за контролу процеса и географски информациони систем, као и сви системи који су подршка одлучивању.

Утицај ИТ на образовање је такође веома велики, без обзира да ли се похађа школски систем, од основног до високог образовања, или се то образовање врши на неформалнији начин путем масивних отворених онлајн курсева (енг. *massive open online course*), разних платформи за видео туторијале, Webinar-а и тд. Према групи аутора најзначајнији ефекат информационих технологија у високом образовању је његов утицај на раст БДП-а, генерисање нових производа и услуга, иновације, отварање нових радних

места, трансформација радне снаге (Stankić et al., 2018, p. 65). Дакле, постоје многи бенефити од примене ИТ у образовању.

Једна од дефиниција е-учења је: „Е-учење је процес у којем наставници презентују одређени наставни садржај применом различитих метода и облика рада (стратегије) и коришћењем информационо комуникационих технологија и на тај начин ученицима обезбеђују сакупљање нових знања“ (Miladinović, 2019).

Према ауторима (Holmes & Gardner, 2006, p. 55) најважније вештине за 21. век су:

- решавање проблема, критичко размишљање и вештине самоусмеравања, а могу се посебно разложити на:
 - критичко и системско размишљање
 - идентификација, формулација и решавање проблема
 - креативност и интелектуална радозналост
 - међуљудске и колаборативне вештине
 - самоусмеравање
 - одговорност и прилагодљивост
 - друштвена одговорност
- ИКТ писменост
 - информациона и медијска писменост
 - комуникационе вештине
 - вештине међуљудских односа и самоусмеравања

Концепти е-образовања као што су паметне учионице, виртуелна реалност, употреба система за управљање учењем (енг. *learning management system*), мобилно рачунарство и друге, су веома важне и у односу на претходно поменуте најважније вештине које су за 21. век покривају добар део. Поред тога, потребно је уочити да тзв. меке вештине (енг. *soft skills*) се не могу изоставити, већ и оне чине важан део поменутог концепта.

2.2.4. Утицај информационих технологија на организационе перформансе

Према теорији организације, организационе перформансе покривају 3 специфичне области излаза предузећа: финансијске перформансе, перформансе на тржишту и вредности за акционаре (Richard et al., 2009, p. 722). За мерење перформанси организација, данас су веома често у употреби индикатори кључних перформанси (енг.

key performance indicators). На одређени начин индикатори кључних перформанси су подскуп метрика, јер обухватају само оне које су релевантне за конкретну организацију (*What is a KPI?*, n.d.). Имајући у виду брзину којом се одвијају промене на тржишту, у условима заоштрене конкуренције, менаџмент организације жели да буде сигуран да ли промена стратегије или пословног модела, даје ефекте и зато су кључни индикатори веома значајни.

Када нека организација усвоји релевантне индикаторе кључних перформанси, онда је потребно они да се прате и да постоји перманента визуелизација истих. Информационе технологије имају кључну улогу у томе, јер употребом различитих технологија база података, као што су трансакциони системи за управљање базама података и технологије складишта података (енг. *data warehouse*), затим употреба одговарајућих софтвера за генерисање извештаја (енг. *reporting software*) омогућава се преглед одређених индикатора кључних перформанси. У данашњем времену, када постоји велика компететивност на тржишту, потребно је ефикасније праћење кључних индикатора перформанси, а то је могуће употребом софтвера за визуелизацију (енг. *visualisation software*), где неки од њих уз одговарајућа софтверска решења у позадини омогућавају праћење у реалном времену. Поред тога, обзиром на захтеве данашњице које један информациони систем је потребно да испоручи својим корисницима, већина софтвера који се користи за визуелизацију је потребно да буде омогућена на различитим типовима корисничких интерфејса и уређаја: од интерфејса на стоном рачунару, преко Web интерфејса до интерфејса на преносивим уређајима, тј. паметним телефонима и таблетима.

Обзиром да одређене организације имају послове који су глобално орјентисани и имају веома велику количину података које је потребно обрадити и презентовати у адекватној форми, оне могу користити технологије рачунарства у облаку, као и технологије великих података за остваривање поменутих циљева. Где резултати тих обрада могу да буду доступни на разним врстама корисничких интерфејса који конкретна организација користи.

Поред тога, визуелизација је веома важан аспект приликом употребе технологија, јер у интеракцији са интерфејсом корисник долази до значајних увида, па произвођачи софтвера за визуелизацију и извештавање воде рачуна да разне типове графикона, командних табли (енг. *dashboard*) и осталих елемената извештавања понуде својим клијентима. Неке организације у пуном капацитету користе те алате и омућавају промену поменутих параметара у реалном времену, преко информационих технологија, па чак иду

и догле да када одређени индикатори падну испод неке вредности, јасним сигналимa на различитим корисничким интерфејсима информишу своје кориснике, што је веома значајно за брзо реаговање.

Један од аспеката о којем је потребно водити рачуна у пројектовању таквих интерфејса и извештаја, јесте когнитивни замор (енг. *cognitive overload*). Когнитивни замор онемогућава корисницима да се фокусирају на посао који је потребно да обаве, јер улажу додатан напор да из неадекватно пројектованог корисничког интерфејса сазнају релевантне податке. Иако нова технологија даје много могућности, у неким случајевима може се показати да најновија решења нису најбоља у контексту њихове употребе јер могу да произведу когнитивни замор (Vesić & Minović, 2015, p. 51). Са друге стране разумевање корисника, њихових менталних модела и пројектовање корисничких интерфејса у складу са тим, може омогућити веома велико искоришћење технологије и доношење правовремених одлука. Поменути аспект је део интеракције човека и рачунара и са посебном пажњом треба приступити при одабиру или развоју поменутих алата, поготово када они треба да визуелизују индикаторе кључних перформанси који су витални за једну организацију. Још један важан аспект који је потребно нагласити је безбедност ИС, а који може утицати на перформансе организације. У јавности је познат случај Stuxnet, где је нанесена штета иранском нуклеарном програму. Треба имати у виду да поред техничких мера организација је потребно да обрати пажњу и на људски фактор да би се претње свеле на прихватљив ниво (Vjelajac & Vesić, 2020). Нови модалитети угрожавања безбедности ИС указују да постоји рањивост информатичко-комуникационе инфраструктуре, самих информација, па чак и људског живота.

3. Карактеристике савременог процеса производње, набавке и употребе софтвера

3.1. Набавка софтверског производа

3.1.1. Уговарање и набавка софтвера

Неко предузеће може да има сопствени развој софтвера, у склопу одређене организационе јединице, тзв. развој софтвера „унутар куће“ (енг. *in-house software development*), а такође може и софтвер набављати од других предузећа које се баве производњом софтвера или продајом лиценци. Када се софтвер набавља од стране других предузећа, обе стране купац и добављач желе да остваре различите интересе у погледу

тога шта се са конкретним купљеним софтверским решењем жели постићи. Наручилац софтвера, жели да за обим средства која је спреман да плати добије што је могуће више у погледу задовољења својих пословних операција, где ће постојати јасни ефекти у погледу ефикасности како се одређено софтверско решење временом све више буде користило, а уједно и задовољство корисника. Са друге стране продавац жели да за испоручен софтвер добије одговарајућу уговорену накнаду, која ће му омогућити да настави са радом и да испуњава своје мисију.

Поред тога, што је потребно да софтвер који се испоручује задовољи функционалности дефинисане захтевима, посао ту није готов, већ постоји и фаза одржавања, као и потребна обука и подршка корисницима, које испоручилац је потребно да обезбеди. Понекад та обука обухвата и пренос специфичних знања које је испоручилац потребно да обезбеди запосленима у ИТ сектору предузећа, тако да одређен број активности у вези са одржавањем се пренесе на само предузеће које је купило софтвер. Зато је важно да се оствари одговарајући ниво квалитета, који ће испоручилац софтвера обезбедити у процесу развоја софтвера, а наручилац ће га препознати и своје обавезе извршити у складу са истим.

Приликом уговорања и набавке софтвера постоји већи број потенцијалних проблема са којима се суочавају и наручиоци и испоручиоци софтвера. Један део тих проблема се односи на проблем разумевања захтева наручиоца софтвера, јер пројектанти и програмери испоручиоца софтвера у комуникацији користе термине које запослени наручиоца не користе, или их не довољно разумеју, јер је већи број тих термина апстрактан и може имати различите конотације. Нпр. уколико неко употреби термин интерфејс (енг. *interface*), он може значити нешто што корисник види на екрану, тј. место интеракције корисника и рачунара, док у контексту пројектовања и разматрања повезаности одређених софтверских модула може да значи функционалност који један модул или компонента обезбеђује другом модулу. Са друге стране корисници имају значајно доменско знање, којима пројектанти и програмери или немају или немају довољно да би разумели односе између разних пословних термина које корисници употребљавају. На одређени начин они не причају „истим језиком“ и ту може доћи до међусобног неразумевања по питању захтева, тј. спецификације по којој одређено софтверско решење треба да се испоручи, прилагоди или набави.

Поред тога, уколико наручилац софтвера је до тренутка набавке користио неко други решење, које је настало приликом узимања захтева у прошлости ранијих добављача софтвера и корисника, може се очекивати да је одређена пословна логика искодирана на

„заједничком језику“ пројектаната и програмера тог решења и корисника. Тада је важно направити разликовање појмова у старом и новом систему, тако да корисници могу да разумеју разлике и да у односу на њих дају захтеве прецизније. У таквој једној комуникацији, важно је испитати све чињенице и појмове и не претпостављати ништа.

Како је узимање захтева најчешће прва фаза у развоју једног софтверског решења, важно је да добављач софтвера има у виду да грешка направљена у овој фази, а при томе уколико се дошло до фазе кодирања или испоруке, софтвер ће имати значајан број проблема јер неће испуњавати функционалност у целости, а то ће захтевати враћање на узимање захтеве у комуникацији са корисницима и поновног пројектовања и развоја, а поред тога и преправке саме пројектне документације. То може значајно утицати да се пробије буџет испоручиоца опредељен за овај пројекат. Исти проблем настаје уколико се надекватно спроводи процес развоја софтвера у некој од наредних фаза, и не поштује се приступ који је усвојен да би се процес развоја привео крају на један систематичан и успешан начин.

Након испоруке софтвера корисници имају прилику да истестирају његове функционалности и да провере у којој мери су њихова очекивања са аспекта разних атрибута квалитета испуњена. Постоји велики број ситуација у којима оспежно тестирање и проверу квалитета није могуће спровести, па се тек касније уочавају проблеми који нису на почетку употребе били видљиви. Корисници очекују да се одређеним одржавањем поменути проблеми могу исправити и софтвер прилагодити да у потпуности одговара захтевима и да се понаша на очекиван начин. Из свих поменутих разлога велики број испоручилаца софтвера приступа системском обезбеђењу квалитета софтвера (енг. *software quality assurance*).

Један битан аспект који наглашава аутор (М. Јовановић, 2010, р. 4) односа у релацији наручиоца и испоручиоца софтвера јесте неразумевање власништва и права на испоручен софтвер. Имајући у виду сложеност у процесу развоја софтвера, где многе ствари до саме реализације нису видљиве за кориснике, а при томе испоручиоци нису до краја сигурни да су на правом путу, постоји потреба да се спроведе опсежна припрема и на страни наручиоца и испоручиоца софтвера и да се пре самог уговарања спроведе адекватан трансфер знања у оба смера како би приближио стране, а процес био у потпуности успешан на обострану корист. Зато је важно да се дефинишу права и обавезе наручиоца и испоручиоца, јасним, прецизним и свеобухватним уговором.

Такође битан аспект је и лиценцирање, тј. право на употребу (коришћење) софтвера одређено време као и остали аспекти везани за употребу софтвера. Постоји

неколико типова лиценци којима се уређује употреба софтвера, а најпознатије су власничка (енг. *proprietary*), слободна (енг. *free*) и отвореног кода (енг. *open source*).

3.1.2. Употреба стандарда при уговарању софтвера

Добра пракса софтверских уговора (енг. *software contracts*), треба да обухвати следеће елементе и одговоре на питања (Law Boutique Lexidy, n.d.):

- права интелектуалне својине - ко чува софтвер, код и позадинску технологију?
- фазе рада - како одредити колико посла треба завршити у свакој фази пројекта?
- плаћање - који услови плаћања су прихватљиви за обе стране?
- материјал заштићен ауторским правима - ко је власник материјала заштићеног ауторским правима који иде уз развој софтвера?
- спецификације - један од најважнијих делова уговора, добро написане спецификације тачно одређују какав ће готов производ бити.

Према мишљењу аутора, један од најбољих начина да све стране заштите своје интересе је дефинисање прецизних уговора, поштовање закона и стандарда, где би се поменута потенцијална дискрепанција у комуникацији на релацији наручиоца и испоручиоца софтвера, смањила уз поштовање одговарајућих стандарда (М. Јовановић, 2010, р. 5). Поред тога, истиче се и њихов значај у томе да су настали на примени добре праксе, да се омогућава евалуација сваке појединачне фазе у развоју софтвера, да се осигурава да су вредности атрибута квалитета испуњени, да се постиже одговарајући степен колаборације на заједничким пројектима већег броја организација, као и основу за размену информације између информационих система.

3.1.3. Стандарди у информационим технологијама

Да би се олакшала комуникација, производња, трговина и мерење између људи и организација у најширем смислу, постављају се стандарди, а стандардизација је добровољна сарадња индустрије, потрошача, државних власти, истраживача и других заинтересованих страна за развој техничких спецификација заснованих на консензусу (Cenelec, n.d.). Према Институту за стандардизацију Србије постоје три врсте економске користи од примене стандарда унутар компанија: унапређење интерних операција, иновирање и повећање обима операција, креирање или улазак на нова тржишта. Током година предложено је много стандарда које су коришћене у информационим

технологијама, а међу првима су се користили у телекомуникацијама (ISS, 2014). Постоје две велике међународне организације које се баве стандардима: међународна организација за стандардизацију (енг. *International Organization for Standardization*, скраћено ISO) и међународна електротехничка комисија (енг. *International Electrotechnical Commission*, скраћено IEC). Ове две организације су удруженим деловањем формирале ISO/IEC JTC 1 технички одбор 1987. године, која развија стандарде у области информационо-комуникационих технологија. У склопу поментутог одбора постоји организациони део који се бави стандардизацијом процеса, алата и технологија за подршку за инжењерство софтверских производа и система, ISO/IEC JTC 1/SC 7 Software and systems engineering.

ISO/IEC JTC 1/SC 7 је дефинисала кључне стандарде у области софтверског инжењерства који се односе на процесе животног века софтвера (енг. *Software life cycle processes*). Први стандард је био ISO/IEC 12207:1995, а затим су уследиле допуне ISO/IEC 12207:1995/AMD 1:2002 и ISO/IEC 12207:1995/AMD 2:2004. Након тога су повучени из употребе, јер је објављен нови стандард ISO/IEC 12207:2008 којим се успоставља општи оквир за процесе животног циклуса софтвера, са јасно дефинисаном терминологијом, који може да буде референциран у софтверској индустрији. Он садржи процесе, активности и задатке који ће бити примењени током процеса аквизиције софтверског производа или сервиса и током набавке, развоја, рада, одржавања и повлачење из употребе софтверског производа (ISO/IEC JTC 1/SC 7, n.d.-b). ISO/IEC/IEEE 12207:2017 је тренутна верзија овог стандарда, која је заменила ISO/IEC 12207:2008 и у чијем дефинисању поред наведених међународних организација је учествовала и IEEE Computer Society, да би ревизија овог стандарда била у вези са ревизијом ISO/IEC/IEEE 15288:2015, *Systems and software engineering – System life cycle processes*. Кроз ове ревизије се хармонизује структура и садржај оба документа, при чему се уважавају захтеви заједница (ISO/IEC JTC 1/SC 7, n.d.-a). То може послужити у давању смерница ка широј заједници људи који су део софтверске индустрије.

Током година ISO/IEC JTC 1/SC 7 је дефинисала неколико стандарда квалитета софтвера, први је био ISO/IEC 9126:1991, којег је наследила ревизија у облику ISO/IEC 9126-1:2001, а затим је настао ISO/IEC 25010:2011, који је и данас актуелан. Поред тога, стандард ISO/IEC 14598-1:1999, је замењен ISO/IEC 25040:2011. ISO/IEC 9126-1:2001 у односу на пређашњу верзију има координацију са ISO/IEC 14598-1:1999 и дефинише моделе квалитет софтвера описан преко општих карактеристика, које имају своје подкарактеристике, које су састављене од атрибута, који се могу мерити употребом

метрике (SIS, 2003). ISO/IEC 14598-1:1999, стандардизује поступак евалуације квалитета софтверског производа.

Пређашњи стандарди ISO/IEC 9126 и ISO/IEC 14598 замењени су и проширени серијом стандарда ISO/IEC 25000, познатијим као System and Software Quality Requirements and Evaluation, скраћено SQuaRE. Поменуто, тренутно актуелна серија стандарда се састоји од 5 делова: ISO/IEC 2500n – Quality Management Division, ISO/IEC 2501n – Quality Model Division, ISO/IEC 2502n – Quality Measurement Division, ISO/IEC 2503n – Quality Requirements Division и ISO/IEC 2504n – Quality Evaluation Division (ISO/IEC JTC 1/SC 7, n.d.-f). Стандарди у склопу ISO/IEC 2500n дефинишу све заједничке моделе, појмове и дефиниције које ће корисити други стандарди у оквиру SQuaRE серије. Стандарди који формирају део ISO/IEC 2501n представљају детаљан модел квалитета за рачунарске системе и софтверске производе, квалитет у употреби и податке. Стандарди дела ISO/IEC 2502n се односе на референтни модел за мерење квалитета софтверског производа, математичке дефиниције мера квалитета и практичне водиче за њихову примену. Стандарди који чине ISO/IEC 2503n помажу да се специфицирају захтеви за квалитетом који се могу користити у процесу подношења захтева за квалитетом софтверског производа који ће се развијати или као улаз у процесу евалуације. Стандарди у склопу ISO/IEC 2504n обезбеђују захтеве, препоруке и смернице за евалуацију софтверског производа (ISO/IEC JTC 1/SC 7, n.d.-f). На основу њих могуће је израдити софтверске алате који могу да евалуирају одређено софтверско решење до неког нивоа и дају препоруке шта је потребно побошљати.

Проширења SQuaRE серије стандарда, од ISO/IEC 25050 до ISO/IEC 25099 су одређени да садрже систем или квалитет софтверског производа међународних стандарда и/или техничких извештаја који апострофирају специфичне апликационе домене или који могу бити коришћени комплментарно са једним или више међународних SQuaRE стандарда (ISO/IEC JTC 1/SC 7, n.d.-e). То може бити корисно за оне софтверске куће које послују унутар одређеног домена.

3.2. Интеракција човек-рачунар (HCI)

3.2.1. HCI - појам, дефиниција и циљ

Интеракција човек-рачунар (енг. *Human Computer Interaction*), скраћено HCI је мултидисциплинарна област која настаје у пресеку психологије и когнитивне науке, ергономије, социологије, рачунарке науке и инжењерства, пословања, графичког дизајна, техничког писања и тд., при чему централно место заузимају рачунарске науке и

системски дизајн (Dix et al., 2004, p. 4). Најопштије речено проучавањем начина на који рачунарске технологије утичу на људски рад и активности, назива се интеракција човек-рачунар (Dix, 2009, p. 1327). Према дефиницији Хјуита од 1992. године, а коју АСМ (Churchill et al., n.d.) цитира гласи: „Интеракција човек-рачунар је дисциплина која се бави дизајном, евалуацијом, и имплементацијом интерактивних рачунарских система које људи користе и проучавањем главних феномена које их окружују“.

Интеракција човек-рачунар проучава: методологије и процесе за дизајнирање интерфејса, методологије за имплементацију интерфејса, технике за евалуацију и поређење интерфејса, развој нових интерфејса и техника за интеракцију и развој дескриптивних и предиктивних модела и теорија интеракције (Tripathi, 2011, p. 1). Из поменутог види се да је основни концепт интеракције човек-рачунар, место на којем се одвија комуникација човека и рачунара, тј. интерфејс (енг. *interface*).

Историјски гледано постоје три периода или правца у интеракцији човек-рачунар (Duarte & Baranauskas, 2016, p. 1):

- први - веома практичан, решава конкретне проблеме са фокусом на људски фактор и ергономију и решавање проблема у интеракцији
- други - теоријски оријентисан, покушава да сагледа људски ум и рачунар као један јединствен процесор, где се пажња усмерава на то како људски ум перципира информације са машине и употребом интерфејса враћа их машини, наглашава се појам контекста
- трећи - заузима се другачије становиште у односу на претходна два правца јер нове технологије, као што је проширена реалност (енг. *augmented reality*), свеprisутно рачунарство (енг. *pervasive computing*), опипљиви интерфејси (енг. *tangible interfaces*) дају нов печат интеракцији, а истовремено вредности и култура, као и поглед на улогу истраживача се уважавају, па су могуће вишеструке интерпретације интеракције

Унапређење интеракције између корисника и рачунара, чинећи рачунаре употребљивијим и пријемчивијим за потребе корисника, је основни циљ интеракције човек-рачунар, а дизајнирање таквих система који умањују препреку између људског когнитивног модела онога што желе да постигну и рачунарског разумевања задатка корисника је дугорочан циљ интеракције човек-рачунар (Riedel et al., 2011, p. 244). Дакле, интеракцијом је потребно управљати на начин да се људима олакша рад.

3.2.2. Проблем интеракције човек-рачунар

У основи интеракције човек-рачунар, је таква употреба технологија која ће омогућити кориснику да једноставно спроведе одређени задатак на одређеном интерфејсу. Иако делује да је то једноставно постићи, у пракси се дешава да веома функционални софтвери, одбијају кориснике, јер у интеракцији корисници осећају одређену фрустрацију приликом извршавања задатка. У тако окарактерисаној интеракцији, настаје когнитивни замор јер се улаже додатни напор и концентрација потребна за извршење неколико задатака у уређеном редоследу (Conklin, 1987, р. 40). Корисници осећају незадовољство при дуготрајној употреби дизајнираних система на начин да не одговара њиховом менталном моделу, а њихов посао постаје им мучан и временом покушавају да га избегну. На одређени начин, несагледавањем и игнорисањем овог проблема, често присутног у пракси програмера и пројектаната, систем одбија кориснике да га користе. У основи поменутог проблема лежи употребљивост (енг. *usability*) која има за циљ да оцени како једноставно корисници употребљавају интерфејс, а такође се односи и на методе за побољшање једноставније употребе у процесу дизајна. Употребљивост је атрибут квалитета. Дакле, уз употребљивост се стварају могућности да корисници користе систем и да у таквој интеракцији не осећају когнитивни замор.

Потребно је разграничити појмове корисности (енг. *utility*) и употребљивости (енг. *usability*), где се корисност односи на пружање потребних карактеристика или функција, а употребљивост на лакоћу и задовољство коришћења ових карактеристика или функција (Nielsen, 2012). Истовремено неки софтверски производ може бити једноставан за коришћење а да нема потребне функције, а са друге стране може да има потребне функције, али да буде веома тежак за коришћење. Истовремено испуњене ове две особине омогућавају корисницима да свој посао заврше без когнитивног замора уз задовољство. Реализација једног потпуног система подразумева испуњење обе особине, што у пракси није лако, па се у науци приступа изучавању оба појма, који су везани за функционалне и нефункционалне софтверске захтеве.

3.3.3. Инжењеринг употребљивости

Да би се побољшала употребљивост интерфејса, примењује се инжењеринг употребљивости (енг. *usability engineering*), где се у различитим фазама развоја дизајн тестира са корисницима и експертима, а у циљу да се систем учини употребљивијим. Потреба за инжењерингом употребљивости је оправдана поготово када постоје фамилије производа, где се уграђено знање и ресурси у једну верзију производа могу пренети ка

осталим верзијама производа, па је оправдано то урадити у раној фази (Nielsen, 1994, p. 1) јер ће се тиме обезбедити ефикасна и ефективна употреба производа уз задовољство. Једном када корисници навикну на одређени дизајн система који је употребљив, веома тешко прелазе на дизајн у новој верзији истог система, па је зато важно да интеракција буде осмишљена тако да корисници имају утисак да се ништа није променило, тј. да њихова интеракција буде што је могућа сличнија интеракцији у старој верзији.

Једна од кључних идеја коју презентује Нилсен (Interaction-Design-Foundation, n.d.-c) јесте да уместо да се размишља о уградњи употребљивости након процеса развоја производа, те активности се могу одвијати током развоја. При томе треба обратити пажњу да се атрибути употребљивости као што су лакоћа учења (енг. *learnability*), ефикасност (енг. *efficiency*), меморисање (енг. *memorability*), грешке (енг. *errors*) и задовољство (енг. *satisfaction*) константно инкорпорирају у том процесу.

За адекватано спровођење инжењеринга употребљивости, потребно је усвојити систематичан приступ, где би се могао спровести приступ по коме се прво креће од организације и њеног утврђивања потребе за употребљивошћу. Обзиром да организација има потребе да се прилагоди крајњим корисницама, а ради остваривања својих циљева, потребно је програмере и пројектанте усмерити да промене свој инцијално осмишљен дизајн на начин да подстакну употребљивост, где важно да менаџмент подржи поменуте процесе. За почетка је потребно обезбедити макар малу количину специфичних ресурса за инжењеринг употребљивости, да се осигура да се замисао не оствари услед притисака који се увећавају код крајњих рокова. У развојном циклусу употребљивост, активности које воде уградњи употребљивости могу се спровести у великом броју фаза. Корисници система је потребно да истестирају све интерфејсе и приликом спровођења овог процеса, потребно је то и осигурати (Nielsen, 1993, p. 21). Једним осмишењим процесом управљања у процесу инжењеринга употребљивости уз свест и деловање од стране менаџмента са усмереним активностима ка извршиоцима (пројектантима, програмерима и дизајнерима) осигурава се инкорпорирање употребљивости у дизајн производа још у раним фазама путем којег се поред ефикасне и ефективне употребе истог од стране корисника, производи и задовољство чиме се ствара добра основа да производ не живи само од прве продаје и употребе од стране корисника, већ од сваке наредне. Као пример може да се наведе компанија Apple, чији корисници уређаја жељно ишчекују сваку нову верзију, да би поред нових карактеристика и функција које нуде нове верзије производа, првенствено уживали и у препознатој интеракцији.

3.3. Употребљивост софтвера

3.3.1. Појам употребљивости софтвера

Уколико посматрамо употребљивост софтвера, можемо рећи да одређење поменутог појма настаје у пресеку интеракције човек-рачунар, софтверског инжењерства са фокусом на један посебан део квалитет софтвера. Потребно је имати у виду да са једне стране постоји изражена мултидисциплинарност у пољу интеракције-човек рачунар, где се поред рачунарске науке, системског дизајна, користи и психологија и когнитивне науке и др., а са друге стране постоји потреба да се обухвате све активности у вези са израдом, употребом и одржавањем софтвера по принципима инжењерских дисциплина кроз софтверско инжењерство, па из тога следи да употребљивост софтвера има велику комплексност и потребно је обратити пажњу на све његове потенцијално наслеђене специфичности настале у поменутом пресеку осталих дисциплина.

На одређени начин употребљивост комбинује науку и уметност (M. Abdelaziz et al., 2016, p. 12), и њена употреба је много шира од примене на специфичним софтверским производима, јер је везана за сваки појединачни део људске интеракције. Поред наведеног, употребљивост софтвера је један специфичан део употребљивости, која не мора да буде везана само за софтвер, већ се односи и на производе, уређаје, системе и тд.

Употребљивост је једна од карактеристика квалитета софтвера, присутна у разним моделима квалитета софтвера. Квалитет према неким ауторима (M. Abdelaziz et al., 2016, p. 11) се исказује као интерни квалитет, који се односи на унутрашње аспекте који нису видљиве крајњем кориснику и као екстерни квалитет кроз корисничко искуство приликом употребе, па је у том смислу интерни квалитет више везан за програмера, док је екстерни квалитет више везан за корисника. Дакле, екстерни квалитет се остварује у процесу интеракције корисника са софтверским решењем.

Одређени аутори, наводе шта се све може подразумевати под појмом употребљивост и да поменути појам значи размишљање о томе како и зашто људи користе производ, па и да употребљивост значи евалуација (процена), да значи много више од „једноставне употребе“ и да значи дизајн усмерен на корисника (енг. *user-centered design*) (Quesenbery, n.d.). Па се зато и постављају питања у вези употребљивости, да ли је то резултат или је процес или је скуп техника или је филозофија дизајнирања ради задовољења људских потреба ?

Имајући у виду све напред наведено и другачијег погледа софтверског инжењерства и интеракције човек-рачунар на употребљивост, потребно је сагледати дефиниције употребљивости ради бољег и потпунијег разумевања појма.

3.3.2. Дефиниције употребљивости

Како су се дисциплине интеракције човек-рачунар и софтверског инжењерства развијале током времена, тако се појам, објашењење и карактеристике употребљивости мењао. Током година предложене су многе дефиниције употребљивости. Неке од њих биће приказане у овом раду.

Према спроведеном истраживању 1985. Батлер је тестирао употребљивост и констатовао је да корисници веома лако уче квалитет производа када је његов кључни атрибут лакоћа учења (енг. *learnability*), која је описана у појмовима времена да се нешто савлада и да се изврши опоравак од грешака или њихово избегавање (Butler, 1985). Поменуто истраживање указује да када су циљеви дизајна чврсто повезани са емпиријском дефиницијом корисничких перформанси, систем ће бити употребљивији.

Шакл указује да када систем има атрибуте да је флексибилан, субјективно пријатан, ефикасан и постоји лакоћа у учењу тада је он употребљив, а процена његове употребљивости се може спроводити кроз све фазе животног циклуса (Dubey & Rana, 2010, p. 4724). Према Гудвин, употребљивост иако није до краја појмовно разјашњена, је комплементаран захтев са функционалношћу једног система који доприноси целокупном систему и његовој ефективности, а то захтева разумевање корисника, њиховог нивоа експертизе, количине времена које ће провести у употреби система и каква ће бити промена њихових потреба када добију искуство (Goodwin, 1987, p. 230). Дакле, употребљивост никако не ограничава функционалност већ утиче на ефективну употребу система.

Нилсен (Nielsen, 1993) подробније описује употребљивост преко атрибута, као што су лакоћа учења (енг. *learnability*), ефикасност (енг. *efficiency*), меморисање (енг. *memorability*), грешке (енг. *errors*) и задовољство (енг. *satisfaction*). Лакоћа учења значи да се систем лако научи, тако да корисник веома брзо може да обавља послове. Ефикасност описује да једном када корисници науче да користе систем могу да остваре високу продуктивност, јер је систем ефикасан. Меморисање се односи на чињеницу да је лако упамтити шта систем ради чак и када се неко време не користи па крене поново да се користи. Грешке указују да систем треба да има низак ниво грешака, а за случај и да се

десе да корисници могу лако да се опораве од њих. Задовољство указује да су корисници задовољни када користе систем .

Аутор такође објашњава да постоји јака релација између прихватања система и употребљивости, на начин да описани атрибути делују на употребљивост, па она заједно са корисношћу (енг. *utility*), делује на „укупну корисност“ (енг. *usefulness*), која опет заједно са трошковима, компатибилношћу, поузданошћу и другим факторима утиче на практичну прихватљивост (енг. *practical acceptability*), која делује заједно са друштвеном прихватљивошћу (енг. *social acceptability*) и заједно утичу на прихватљивост система (енг. *system acceptability*) (Nielsen, 1993). На тај начин се указује на дистинкцију између појмова и њихове односе, чиме се смањује неразумевање и конфузија често присутна у заједници.

Према међународном стандарду ISO 9241-11:2011, а касније и у актуелној верзији стандарда ISO 9241-11:2018, употребљивост се дефинише као степен до којег систем, производ или услуга се може користити од стране одређених корисника да би ефикасно и ефективно остварили одређене циљеве и са задовољством у одређеном контексту употребе (International Organisation for Standardisation (ISO), 2018).

Сада ће бити презентоване још неке дефиниције употребљивости. Из угла посматрања на мобилне трговину и проблема који се јављају приликом дизајнирања интерфејса аутори (S. S. Chan & Fang, 2009, p. 2158) на појам употребљивости кроз то како добро је апликација дизајнирана за кориснике да би извели жељени задатак лако и ефективно. Аутори са аспекта Web сајтова усмерених на кориснике (енг. *user-centered*) дефинишу употребљивост да је потенцијал за ефикасност, ефективност и задовољство корисника, или једноставно, успех корисника и задовољство када имају интеракцију са технологијом (O'Connell & Murphy, 2009, p. 3896). Са становишта софтверских архитектура, које се баве нефункционалним захтевима које је потребно да буду део неког софтверског решења, употребљивост, поред скалабилности, доступности, перформанси и тд. спада у атрибуте квалитета који су пре свега усмерени на кориснике (Gorton, 2011, p. 5). Гледано са аспекта учења и искуства корисника аутори користе дефиницију да је употребљивост могућност употребе производа од стране ученика за његову намеравану сврху ефикасно и без фрустрација (Kenttälä et al., 2017, p. 140). Из свих наведених дефиниција, видимо да је веома тешко дефинисати појам употребљивости због његове сложене природе, која је више везана за сваки пројекат појединачно него за неко опште решење или неку метрику путем које би се рекало да ли је систем употребљив или не.

3.3.3. Декомпозиција употребљивости

Постоји много дефиниција употребљивости и не може се дати јединствена дефиниција, јер постоји велика сложеност која настаје у пресеку интеракције човек-рачунар и софтверског инжењерства и разноврсних домена примене употребљивости, а при томе та сложеност ће бити још већа услед нових технологија које мењају интеракцију човек-рачунар, као што је проширена реалност, свеприсутно рачунарство, употреба интернета ствари, великих података и тд. Из поменутих разлога, многи истраживачи још од раних дана, претпостављајући да ће се са настанком нових технологије променити природа употребљивости уместо да покушавају да дају свеобухватну дефиницију, приступили су анализи исте објашњавајући је преко њених атрибута, карактеристика, принципа, димензија и тд.

Декомпозиција употребљивости на атрибуте у првој посматраној фази иде од приступа МекКола (Jim McCall et al., 1977) који наводи операбилност (енг. *operability*), тренинг (енг. *training*) и комуникативност (енг. *communicativeness*), приступа Боема (Boehm et al., 1978) који означава портабилност (енг. *portability*), одрживост (енг. *maintainability*) и приступа (Reed, 1986) који наводи лакоћу учења (енг. *ease of learn*) и лакоћу употребе (енг. *ease of use*). У овој фази уочавамо већи фокус на сам рачунар, у интеракцији човека и рачунара, па и употребљивости, а аспекти човека се мање спомињу.

У другој фази декомпозиција иде преко Шакла (Shackel, 1981) који наводи атрибуте једноставност употребе (енг. *ease of use*), ефективност (енг. *effectiveness*), лакоћа учења (енг. *learnability*), флексибилност (енг. *flexibility*), став корисника (енг. *user attitude*), затим приступа Нилсена (Nielsen, 1993) лакоћа учења (енг. *learnability*), ефикасност (енг. *efficiency*), меморисање (енг. *memorability*), мањи број грешки (енг. *few errors*), задовољство (енг. *satisfaction*) и Константина (Constantine & Lockwood, 1999) лакоћа учења (енг. *learnability*), ефикасност у употреби (енг. *efficiency in use*), лакоћа памћења (енг. *rememberability*), поузданост у употреби (енг. *reliability in use*), задовољство корисника (енг. *user satisfaction*). Поменута фаза много више укључује аспекте човека у интеракцију.

У трећој фази декомпозиција иде преко Шнајдермана (Shneiderman & Plaisant, 2005) са атрибутима употребљивости као што су време за учење (енг. *time to learn*), брзина перформанси (енг. *speed of performance*), стопа корисничких грешака (енг. *rate of errors by users*), задржавање током времена (енг. *retention over time*), субјективна сатисфакција (енг. *subjective satisfaction*), Тамира (Tamir et al., 2008, p. 47) који наводи лакоћу учења (енг. *learnability*), операбилност (енг. *operability*), разумљивост (енг.

understandability) и Харисона (Harrison et al., 2013, p. 3) ефективност (енг. *effectiveness*), ефикасност (енг. *efficiency*), задовољство (енг. *satisfaction*), лакоћа учења (енг. *learnability*), лакоћа памћења (енг. *memorability*), грешке (енг. *errors*), когнитивно оптерећење (енг. *cognitive load*), корисник (енг. *user*), задатак (енг. *task*), контекст (енг. *context*). Ова фаза је уједно и најсложенија.

За потребе рада издвојићемо неколико декомпозиција употребљивости које делују пријемчиво у односу на посматрани проблем истраживања, приказ је дат у табели:

Табела 1. Декомпозиција употребљивости према одређеним ауторима

| Аутор | Атрибути |
|--|--|
| Нилсен (Nielsen, 1993) | лакоћа учења (енг. <i>learnability</i>), ефикасност (енг. <i>efficiency</i>), меморисање (енг. <i>memorability</i>), мањи број грешки (енг. <i>few errors</i>), задовољство (енг. <i>satisfaction</i>) |
| Луис (Lewis, 1995) | корисност система (енг. <i>system usefulness</i>), квалитет информација (енг. <i>information quality</i>), квалитет интерфејса (енг. <i>interface quality</i>) |
| Констатин (Constantine & Lockwood, 1999) | лакоћа учења (енг. <i>learnability</i>), ефикасност у употреби (енг. <i>efficiency in use</i>), лакоћа памћења (енг. <i>rememberability</i>), поузданост у употреби (енг. <i>reliability in use</i>), задовољство корисника (енг. <i>user satisfaction</i>) |
| Квесенбери (Quesenbery, n.d.) | ефективност (енг. <i>effectiveness</i>), ефикасност (енг. <i>efficiency</i>), ангажовање (енг. <i>engagement</i>), толеранција грешака (енг. <i>error tolerance</i>), лакоћа учења (енг. <i>ease of learning</i>) |
| Блендфорд (Blandford & Buchanan, 2002) | технички (енг. <i>technical</i>), когнитивни (енг. <i>cognitive</i>), социјални (енг. <i>social</i>), орјентисан на дизајн (енг. <i>design-oriented</i>) |

3.3.4. Модели употребљивости

Квалитет софтверског производа је веома значајан нефункционални захтев, који ставља пред изазов софтверско инжењерство која је потребно у пракси да га обезбеди. Употребљивост је један његов важан аспект (квалитет у употреби) или како се још назива атрибут, карактеристика и тд. Много истраживача је током година предложило разне моделе употребљивости, који се састоје од атрибута употребљивост, а које треба да буду смерница софтвер инжењерима, дизајнерима и осталим учесницима у процесу развоја софтвера, како да се направи квалитетан софтверски производ. Тако да се након усвајања модела употребљивост, спроводи његова евалуација да би се утврдило да ли будући софтверски производ задовољава поменуте атрибуте употребљивости. Дакле, како квалитет софтвера има употребљивост као један од својих атрибута, а при томе и употребљивост има своје атрибуте употребљивости, то значи да усвајањем одређеног

модела употребљивости се може значајним делом утицати на крајњи квалитет софтверског производа, уз процес спровођења евалуације тог модела употребљивости.

Услед великог броја атрибута употребљивости, неки експерти у развоју софтвера се опредељују за подскуп који према њиховој процени највише одговара њиховом специфичном софтверком производу и на тај начин заправо креирају свој модел употребљивости (D. Gupta et al., 2017, p. 2). Неки од модела употребљивости су већ поменути у претходној секцији, а овде ће се навести још два који су значајни у теорији.

Хиксов (Nix & Hartson, 1993) модел употребљивости обухвата лакоћу учења (енг. *learnability*), дугорочно задовољство корисника (енг. *long-term user satisfaction*), употребу напредних карактеристика (енг. *advanced feature usage*), први утисак (енг. *first impression*), перформансе (енг. *performance*), одрживост (енг. *retainability*). Аутори (Seffah et al., 2006, p. 168) предлажу QUIM оквир (енг. *framework*) а наглашавају атрибуте ефикасност (енг. *efficiency*), ефективност (енг. *effectiveness*), продуктивност (енг. *productivity*), задовољство (енг. *satisfaction*), лакоћу учења (енг. *learnability*), сиурност (енг. *safety*), поузданост (енг. *trustfulness*), приступачност (енг. *accessibility*), универзалност (енг. *universality*) и корисност (енг. *usefulness*). Преко QUIM оквира, многе неконзистентности које су дефинисане у неким другим постојећим моделима употребљивости и њихове везе пре свега у међусобном односу фактор или критеријум или метрика покушавају да се реше, јер нису довољно јасно дефинисани.

3.3.5. Стандарди употребљивости

Током година Међународна организација за стандардизацију, скраћено ISO објавила је већи број стандарда који се тичу употребљивости. Беван (Bevan et al., 2016, p. 268) наводи промене које су се догодиле, где је ISO 9241-11:1998 замењен ISO 9241-11, затим ISO TR 18529 је замењен ISO 9241-220, поред тога изашао је нови стандард ISO/IEC 25066, а ISO TR 9126-4 замењен је ISO/IEC 25022, а два стандарда ISO/IEC TR 9126-2 и ISO/IEC TR 9126-3 су замењени ISO/IEC 25023 стандардом. Велики број стандарда из ове области указује на значај проблема употребљивости, који је током година све израженији услед општег технолошког развоја.

ISO 9241-11:2018 стандард као што је раније поменуто даје дефиницију употребљивости, где она садржи 3 основна елемента: ефективност (енг. *effectiveness*), ефикасност (енг. *efficiency*) и задовољство (енг. *satisfaction*), као и контекст употребе. Према ISO 9241-210, кориснички центриран дизајн (енг. *human-centred design*) је приступ дизајну и развоју система који тежи да учини интерактивне системе употребљивијим

фокусирањем на употребу система и примену „људских фактора“/ергономије и знање о употребљивости и технике (Bevan et al., 2016). ISO 9241-220 даље разрађује процесе потребне за кориснички центриран дизајн у ISO 9241-210. За извештавање о процени употребљивости користи се CIF (енг. *Common Industry Format*) формат, који је описан у ISO/IEC 25066:2016 (ISO/IEC, n.d.). ISO/IEC 25022, обезбеђује скуп мера за мерење квалитета у употреби предложеног у моделу ISO/IEC 25010 (ISO/IEC JTC 1/SC 7, n.d.-d). ISO/IEC 25023 садржи мере за систем и квалитет софтверског производа, укључујући мере употребљивости као атрибута производа и његова намеравана употреба је у окружењима која се баве развојем софтвера (Bevan et al., 2016). Поменути стандарди делују у синергији на начин да дају смернице у вези употребљивости које је потребно да се задовоље.

3.3.6. Модели квалитета софтвера

Квалитет софтвера (енг. *software quality*) се може дефинисати као ефективан софтверски процес примењен на начин да креира корисне производе који обезбеђују мерљиве вредности за оне које их производе и за оне које их користе (Pressman & Maxim, 2015, p. 414). На тај начин испуњењем вредности атрибута на једној страни деловаће се на испуњење вредности оних атрибута који су значајни за другу страну.

Квалитет софтвера је значајан из више разлога: увећање критичности софтвера, нематеријална природа софтвера и акумулација грешака током развоја софтвера (Suman & Мапој, 2014, p. 5635). Квалитет софтвера се може исказати преко модела квалитета софтвера (енг. *software quality models*), који је заправо најчешће графичка представа карактеристика квалитета и веза између њих. Први модел квалитета софтвера је МекКолов (Jim McCall et al., 1977) модел квалитета, настао 1977. године, који је познат још као Ценерал Електрик модел (General Electrics Model). Његова суштина јесте да покуша да превазиђе јаз између корисника и програмера фокусирајући се на бројне факторе квалитета софтвера који описују приоритете како корисника, тако и програмера, а при томе указује на 3 скупа фактора (B. Singh & Kannoјia, 2013, p. 803):

- рад производа (енг. *product operation*) који укључује: тачност (енг. *correctness*), поузданост (енг. *reliability*), ефикасност (енг. *efficiency*), интегритет (енг. *integrity*), употребљивост (енг. *usability*)
- ревизије производа (енг. *product revision*) који укључује: одрживост (енг. *maintainability*), флексибилност (енг. *flexibility*), могућност тестирања (енг. *testability*)

- транзиције производа (енг. *product transition*) који укључује: портабилност (енг. *portability*), поновна употребљивост (енг. *reusability*) интероперабилност (енг. *interoperability*)

Након тога, 1978. године Боем (Boehm et al., 1978) је предложио модел квалитета који је такође врста хијерархијског модела и сличан је МекКоловом. Организован је у три нивоа хијерархије: карактеристике високог нивоа, карактеристике средњег нивоа и примитивне карактеристике. Сваки од ових нивоа има своје карактеристике. Оно што је добро код Боемовог модела види се подељеност разних карактеристика које учествују у укупном квалитету софтвера, али не и о томе како се те карактеристике мере (Al-Badareen et al., 2011, p. 48).

FURPS модел је предложен од стране Грејдија 1987. године и класификује карактеристике на функционалне и нефункционалне захтеве. FURPS је акроним од функционалности (енг. *functionality*), употребљивости (енг. *usability*), поузданости (енг. *reliability*), перформансе (енг. *performance*), могућности подршке (енг. *supportability*) (Suman & Manoj, 2014, pp. 5635–5636). Касније је проширен на FURPS+ модел.

Друмијев модел квалитета је настао 1995. године и усмерен је на везу између атрибута квалитета и податрибута да би повезао својства софтверских производа са атрибутима квалитета софтвера. Такође износи став да је се процес евалуације квалитета разликује за сваки производ и да би требало развити такав општи процес за различите системе. Постоји 7 атрибута квалитета: функционалност (енг. *functionality*), поузданост (енг. *reliability*), одрживост (енг. *maintainability*), ефикасност (енг. *efficiency*), поновна употребљивост (енг. *reusability*), портабилност (енг. *portability*), употребљивост (енг. *usability*) (B. Singh & Kannoja, 2013, p. 803) .

Представљени су класични модели квалитета софтвера који се јављају у литератури, а поред њих постоје и други, као што су СММИ, QMOOD, SATC, Казманов модел и многи други. Поред тога, постоје ISO стандарди и њихов преглед и компаративна анализа се може наћи код Ал-Бадарина (Al-Badareen et al., 2011) и Сумана (Suman & Manoj, 2014). Заснивају се на давањем препорука у вези модела квалитета, као и све атрибуте које квалитет софтвера има.

3.4. Методе за евалуацију употребљивости софтвера

3.4.1. Основни концепти евалуације употребљивости

3.4.1.1. Циљеви евалуације

Како је употребљивост мера колико једноставно корисници користе одређен производ и колико су задовољни приликом употребе поменутог производа, онда је евалуација употребљивости процес који треба да то и процени.

Уколико се путем евалуације утврди да производ није довољно једноставан за употребу, тиме се пружају информације развојном тиму који треба да измене дизајн и да се након тога поново спроведу евалуација све док се проблеми употребљивости који се јављају не поправе. Дакле, евалуација употребљивости је прво потребно да идентификује проблеме и након тога предложи решења да би се унапредила употребљивост и тај предлог решења треба да буде јасан и конкретан, јер недовољно јасан предлог нема вредност (Brinck et al., 2002, p. 407). Приликом развоја софтвера, најчешће услед додатних трошкова или недовољно остављеног времена да се поменути поступак спроведе, често се одустаје од евалуације. Неки истраживачи из области интеракције човек-рачунар никако се не могу сложити са оваквим поступањем јер тиме не постоји начин да се утврди да ли је систем добар и не може да се разуме како је потребно да се нешто издизајнира боље да би се систем побољшао, тј. употребљивост унапредила. Из тог разлога њима су разлози потпуно очигледни за спровођење евалуације употребљивости (Karat, 1997, p. 690) и није им јасно како је могуће другачије поступање. Такође исти аутор истиче да је употребљивост атрибут интеракције са производом у контексту употребе. Дакле, разумевање контекста употребе је веома важно када спроводимо евалуацију употребљивости, је се она не може спроводити изоловано од контекста употребе.

Постоје разне дефиниције контекста, од тога да је само физичка локација контекст, преко становишта да контекст чине локација корисника, окружење, идентитет и време (Ryan et al., 1998), док неки други аутори (Maguire, 2001, p. 453) наводе културни, а постоје и организациони, технолошки и друштвени контекст и тд. Из поменутих разлога Триведи и Канум, покушавају да свеобухватно посматра контекст па тиме и даје дефиницију према којој контекст представља било шта што може да утиче на људско понашање (Trivedi & Khanum, 2012, p. 72). У раду ће се усвојити поменута дефиниција услед своје свеобухватности разматрати, јер се њоме не елиминише ни један важан фактор, нити се фаворизују одређени који могу утицати на евалуацију употребљивости.

3.4.1.2. *Стилови евалуације*

Евалуација употребљивости одређеног производа или специфичног софтверског решења може се спроводити у лабораторијским условима, што више одговара традиционалном HCI приступу, а поред тога се може спроводити и на самом терену (lat. *in situ*) где се налазе корисници. Обе поставке имају својих предности и недостатака и у великој мери зависи од тога шта се спроведеном евалуацијом унутар одабране поставке жели постићи као и од врсте производа који се тестира. Нпр. није исто спроводити евалуацију употребљивости апликације за мобили телефон која може да буде неког забавног садржаја и апликацију на стоном рачунару која се бави одређеном рачуноводственом проблематиком, а обе су у суштини апликације. Сада ћемо погледати предности и недостатке обе поставке.

Тестирање употребљивости у лабораторијски контролисаним условима спроводи се да би се добили тачни подаци над којима могу да се спроведу адекватне анализе. Такође у лабораторијским условима постоји једноставност у скупљању података, једноставно регрутовања учесника као и могућности употребе разне опреме, која се једноставно може подесити да снима експресије лица корисника док користе одређена софтверска решења, као и др. (Razak et al., 2010, p. 105). На тај начин се могу пратити и синхронизовати операције корисника на интерфејсу и одговарајуће експресије које настају у интеракцији са апликацијом. Касније се поменути подаци о интеракцији могу интерпретирати уз експертско знање и извршити одговарајућа евалуација софтверског производа.

Са друге стране уместо контролисаног окружења каква је лабораторија може се корисити поставка у коме се одлази у окружење у којем партиципант истраживања ради и на тој локацији се обавити истраживање, где је потребно имати одговарајућу опрему која је преносива. Оно што је проблематично приликом спровеђења евалуације у реалном окружењу је велика количина дистракције која се јавља приликом рада, од разних социјалних контаката, буке произведене од стране телекомуникационих уређаја и других могућих канцеларијских апарата, свега специфичног што једно окружење носи (Oztoprak & Erbug, 2006). Поменуте специфичности са једне стране, колико утичу на смањење продуктивности у процесу евалуације употребљивости, са друге дају много боље податке, јер се врши спровођење експеримента у реалном окружењу, па су информације везане за контекст употребе релевантније у односу на лабораторијску поставку.

Да би се поменуте контекстуалне информације адекватно пренеле у лабораторију, у лабораторији се могу увести неке промене у окружење да би се боље симулирао реалан

контекст. Тако на пример, аутори наводе да је за потребе евалуације одређене мобилне апликације, да би се симулирала мобилност корисника и покретљивост приликом употребе уређаја уводе се покретне траке (Sun & May, 2013, p. 2) .

Имајући у виду претходно наведено, потребно је одредити се за поставку која је погоднија за конкретно истраживање.

3.4.1.3. Процес евалуације квалитета

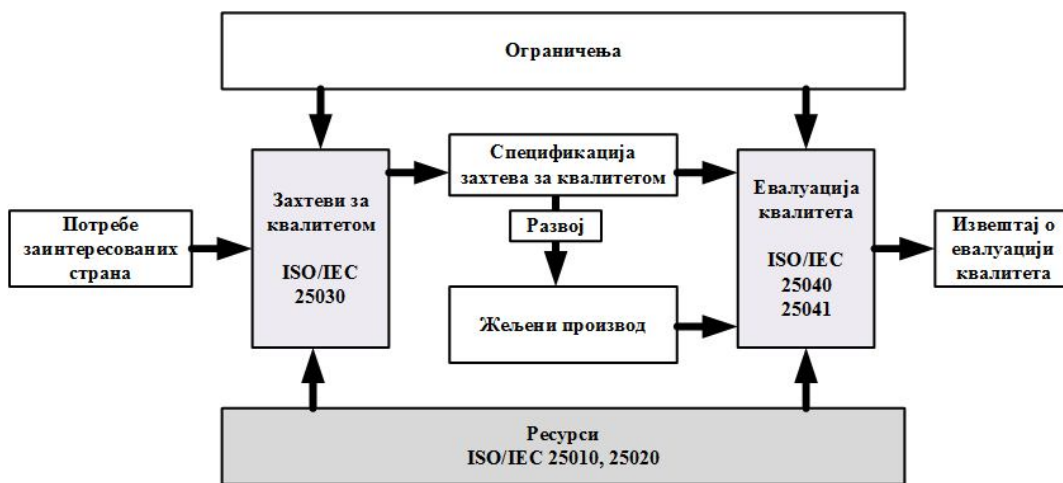
У склопу серије стандарда ISO/IEC System and software Quality Requirement and Evaluation, скраћено названо SQuaRE, специфицира се модел квалитета, мерења, захтеви и евалуациони процес. Евалуациони процес је специфициран у оквиру дивизије ISO/IEC 2504n. Поменути процес у односу са осталим дивизијама поменуте серије стандарда, је дао Есаки (Esaki, 2013), као што се види на слици 12. На наведеној слици се уочава како одређени стандарди делују на друге, да би се евалуација квалитета спровела на прави начин и у сагласности са дефинисаним захтевима.

На основу захтева препорука и водича за евалуацију квалитета софтверског производа у склопу стандарда ISO/IEC 25040 процес евалуације квалитета се састоји из неколико активности ((ISO/IEC JTC 1/SC 7, n.d.-f), (ISO/IEC JTC 1/SC 7, n.d.-e)):

1. Успостављање захтева за евалуацијом, која се састоји из 4 задатка:
 - a. утврђивање сврхе евалуације - документује се сврха због које организација жели да процени квалитет софтверског производа
 - b. прибављање захтева за квалитетом софтверског производа - дефинишу се актери софтверског производа (програмер, преузималац, независни проценитељ (евалуатор), корисник, лице који одржава, добављач и тд.) и да се употребом модела квалитета специфицирају захтеви за квалитетом софтверског производа
 - c. идентификовање делова производа који ће бити укључени у евалуацију - сви делови производа који су укључени у евалуацију потребно је да буду идентификовани и документовани
 - d. дефинисање уверљивости (строгости) евалуације - строгост оцењивања треба да буде дефинисана у циљу да се обезбеди поверење у квалитет софтверског производа у складу са његовом наменом и сврхом евалуације.
2. Специфицирање евалуације, која има 3 задатка:

- a. одабир мера квалитета (евалуационих модула) - евалуатор бира мере квалитета (евалуационе модуле) да би покрио све захтеве евалуације квалитета, што заправо значи да процедуре мерења треба да мере одговарајуће карактеристике (или подкарактеристике) квалитета довољно прецизно, дивизија стандарда за мерење квалитета ISO/IEC 2502n може да се користи
 - b. дефинисање критеријума одлучивања за мере квалитета - за сваку одабрану појединачну меру требало би дефинисати критеријум одлучивања са његовим прагом који треба да укаже на ниво поверења
 - c. дефинисање критеријума одлучивања за евалуацију - евалуатор треба да припреми процедуру за даље резимирање, са одвојеним критеријумима за различите карактеристике квалитета, што ће бити основа за процену квалитета софтверског производа
3. дизајнирање (пројектовање) евалуације - која има 1 задатак:
- a. планирање активности евалуације - идентификоване активности евалуације квалитета софтверског производа треба да буду заказане, да узму у обзир доступност ресурса као што су особље, софтверски алати и рачунари
4. спровођење евалуације - обухвата 3 задатка:
- a. спровођење мерења - одабране мере квалитета софтверског производа примењују се на софтверски производ и компоненте, у складу са планом евалуације, што као резултат има вредности на скалама мерења
 - b. примена критеријума одлучивања за мере квалитета - на измерене вредности примењују се критеријуми одлучивања за мере квалитета софтверског производа
 - c. примена критеријума одлучивања за евалуацију - скуп критеријума за одлучивање треба да буде сумиран у подкарактеристике и карактеристике и да процени резултате у форми исказа до којег нивоа софтверски производ испуњава захтеве за квалитетом
5. закључивање евалуације - састоји се од 4 задатака:
- a. прелед евалуационих резултата - евалуатор и подносилац захтева заједнички прегледају резултате евалуације

- b. креирање евалуационих извештаја - када се резултати прегледају креира се извештај о евалуацији који укључује захтеве евалуације, резултате мерења и спроведених анализа, уочених ограничења, као и евалуаторе и њихове квалификације
- c. преглед евалуације квалитета и обезбеђивање повратне информације организацији - евалуатор ће прегледати резултате евалуације и ваљаности процеса евалуације, показатеље и примењене мере, а повратне информације рецензије требало би да се искористе да побољшају процес евалуације и евалуационе технике (модуле)
- d. стављање на располагање евалуационих података - након завршеног процеса евалуације, подаци и предмети евалуације се смештају у складу са захтевима подносиоца, враћају, архивирају или се уништавају на сигуран начин у зависности од типа података



Слика 12. концепт захтева за квалитетом и евалуације, преузето (Esaki, 2013)

Стандардном су дефинисане препоруке, задаци и подзадаци које упућују да је то један комплексан поступак који служи за проверу квалитета.

3.4.2. Методе и технике за процену употребљивости софтвера

Употребљивост софтвера је веома важна карактеристика једног софтверског производа, која утиче ефикасност, ефективност и задовољство корисника који користи тај производ у одговарајућем контексту употребе. Током година развијано је велики број метода за процену, тј. евалуацију употребљивости софтвера (енг. *Usability Evaluation Method*). Све поменуте методе имају за циљ да процене употребљивост, где би се након поменуте повратне информације, укупан дизајн и употребљивост могла да промени, тако да утиче на побољшање софтверског производа и перцепцију тог побољшања од стране крајњих корисника. То је веома важно за мисију, визију и циљеве једног предузећа, а поготово у ситуацији када је потребно остварити конкурентску предност у заоштреној тржишној утакмци. Дакле, употреба метода и техника за процену употребљивости софтвера утиче директно на квалитет софтверског производа, дајући увид које аспекте једног софтверског решења је потребно исправити и у ком смеру да би се побољшало укупно задовољство.

У складу са поменутиим (Ivory & Hearst, 2001, р. 470) систематизују 3 основне активности општег карактера у циљу евалуације употребљивости:

- снимање - прикупљање података о употребљивости као што су времена извршених задатака, грешке, кршење водича и субјективни рејтинзи
- анализа - интерпретирање података о употребљивости да би се идентификовали проблеми употребљивости на интерфејсу
- критика - предлагање решења или унапређења ради превазилажења проблема

Једна од класификација метода за евалуацију употребљивости која је широко усвојена, класификује поменуте у три целине (S. Gupta, 2015, р. 103):

- методе прегледања (енг. *inspection methods*)
- методе тестирања (енг. *testing methods*)
- методе испитивања (енг. *inquiry methods*)

У наставку ћемо описати сваку од поменутих класа метода за евалуацију употребљивости са својим специфичностима и указати на неке од појединачних метода које припадају свакој од наведених класа.

3.4.2.1. Методе прегледања

Приликом примене методе за тестирање употребљивости, могу се написати тзв. водичи (енг. *guidelines*). У неком тренутку евалуатори имају за циљ да провере усклађеност корисничког интерфејса, и самог водича и то је основна суштина већине метода које се баве прегледањем, тј. инспекцијом (S. Gupta, 2015, p. 103). На основу поменутог прегледа утврђује се да ли постоји неки проблем употребљивости корисничког интерфејса.

3.4.2.1.1. Метода когнитивног пролаза

Метода когнитивног пролаза (енг. *cognitive walkthrough*) или когнитивне шетње спроводи се тако што се симулира понашање корисника у интеракцији са интерфејсом приликом решавања одређеног задатка (Plantak Vukovac & Orehovački, 2010, p. 174). Постоји секвенца задатака, где евалуатор коментарише сваки проблем употребљивости који се појављује у тој секвенци, тј. унапред дефинисаном сценарију.

Предност ове методе јесте што се може релативно брзо спровести уз ниске трошкове и постоји могућност њене примене пре фазе развоја софтвера, током пројектовања, тј. дизајна. Тиме се значајно смањује могућност развоја комплетног производа који је потпуно неупотребљив, јер нема вредност за крајњег корисника. Такође добра ствар у употреби поменуте методе јесте што се релативно брзо може доћи до повратних информација о употребљивости корисничког интерфејса путем којих се може унапредити дизајн производа и решити проблем који може постојати у употребљивости (Interaction-Design-Foundation, n.d.-b). Један од недостатака поменуте методе јесте што је она не даје оцене проблема употребљивости који су се појавили и веома је ограничена искуством и вештином евалуатора чији је рад поприлично напоран у спровођењу ове методе.

3.4.2.1.2. Хеуристичка евалуација

Хеуристичка евалуација (енг. *heuristic evaluation*) је неформална метода процене употребљивости, где мали број евалуатора испитује интерфејс и оцењује његову усклађеност са признатим принципима (или критеријумима) употребљивости, тзв. „хеуристикама“ како их називају Нилсен и Молик (Nielsen, 1994, p. 152; Nielsen & Molich, 1990, p. 249). Уобичајено пре самог поступка спровођења јесте да се усвоји скуп хеуристика који ће се користити за евалуацију употребљивости интерфејса. На крају процеса добија се листа проблема употребљивости интерфејса. Ова метода је веома исплатива са становишта употребе, као и зато што се може применити рано у циклусу развоја производа и добити се повратна информација о дизајну производа што је важно за

превенцију проблема употребљивости. Један од недостатака може бити мали број у потпуности обучених евалуатора који имају потребна специфична знања и искуство у спровођењу поменуте методе у потпуности.

За сваки елемент интерфејса врши се анализа његове усклађености од стране малог броја евалуатора и то са најмање два пролаза, јер се први пут евалуатори упознају са корисничким интерфејсом, а у другом пролазу се фокусира на функционалности корисничког интерфејса и његове елементе и оцењују их у складу са хеуристикама (Plantak Vukovac & Orehovački, 2010, p. 173). У пракси се показало да је довољно да постоји неколико евалуатора, јер су постојала мишљења на почетку њене примене да је потребно већи број њих, али се показало да чак и око пет евалуатора моће да открије до 75% проблема употребљивости (Interaction-Design-Foundation, n.d.-a). То је погодност описане методе, поготово у ситуацији када организације није спремна да определи већа средства.

3.4.2.1.3. Инспекција конзистентности

Приликом употребе неког софтверског производа веома је важно да постоји усклађеност између већег броја интерфејса које користи крајњи корисник. То је посебно изражено код Web апликација или Web страница где је потребно обезбедити један главни стил, који се користи на свим поменутим страницама, то се у пракси решава главном технологијом за презентацију на Web-у, познатијом као каскадни стилови (енг. *Cascading Style Sheets, CSS*). Уколико се посутпа другачије, постоји велика вероватноћа да ће се код крајњих корисника изазвати конфузија и когнитивни замор, а поготово ако се ради о групи корисника која није у довољној мери информатички писмена.

Најчешће се ради о старијој популацији која није имала могућност да у склопу свог образовања усвоји основе рада на рачунару, јер тада рачунари нису били у широкој употреби као данас, већ су били скоцентрисани на употребу у војним комплексима и научним лабораторијама. Уз наведено, организације у којима претходно наведени део популације ради недовољно и несистематски се баве поменутом проблематиком, а ни држава много не чини на овом пољу.

Инспекција конзистентности (енг. *Consistency Inspection*) је метода која се користи да утврди да ли је интерфејс конзистентан и униформан заједно са функционалностима које се нуде када корисник има интеракцију (Wixon et al., 1994). Уобичајено је да се користи као додатна метода уз друге.

У литератури постоје и друге методе прегледања или инспекције, а детаљан преглед је дат код Вилсона (Wilson, 2014). Неке од њих су инспекција карактеристика (енг. *Feature Inspection*), плуралистички пролаз (енг. *Pluralistic Walkthrough*), инспекција заснована на перспективи (енг. *Perspective based Inspection*), формална инспекција употребљивости (енг. *Formal Usability Inspection*), и инспекција стандарда (енг. *Standards Inspection*).

3.4.2.2. Методе тестирања

Када се спроводи евалуација употребљивости путем метода тестирања (енг. *testing methods*), корисници употребљавају систем или прототип да заврше скуп предефинисаних задатака док евалуатор (тестер) или софтвер снима резултате учесника тестирања (S. Gupta, 2015, p. 104). Ове методе паралелном анализом система и понашања корисника покушавају да открију проблеме употребљивости и могу да имају више доступних релевантних информација него неке друге методе.

3.4.2.2.1. Размишљање наглас

Размишљање наглас (енг. *thinking aloud*) је метода (још се назива и протокол) тестирања употребљивости која охрабрује партиципанта да подели своје мисли док ради са производом, а то може да обогати увид у корисничко искуство додавањем додатних димензија као што су њихове мисли, реакције, задовољство, одређени бол и тд. (Barnum, 2010, p. 19). Разумевањем поменутих додатних димензија се побољшава разумевање корисничког менталног модела и самом његове интеракције са производом.

Постоји неколико варијанти ове методе, а најпознатије су две, а то су конкурентно размишљање наглас (енг. *concurrent think aloud*) и ретроспективно размишљање наглас (енг. *retrospective thinkaloud*) (Bergstrom, n.d.):

- Конкурентно размишљање наглас је техника која служи за разумевање мисли учесника док раде са производом, где модератор користи само једноставне вербалне инструкције да охрабри учеснике да дају одговарајуће повратне информације и емотивне одговоре у реалном времену. На тај начин они се боље разумеју. Проблем који може настати јесте утицање на метрике употребљивости као што је тачност и време за задатак.
- Ретроспективно размишљање наглас функционише тако што корисници заврше своју сесију а тада их модератори питају за њихово мишљење, најчешће уз употребу видео снимка који приказује њихове акције поново.

У односу на конкурентно размишљање наглас не утиче се на метрике употребљивости, али може се догодити да се сесија знатчајно продужи и да корисници не могу да се сете својих мисли, што лоше утиче на увиде у проблеме употребљивости.

Потребно је проценити коју варијанту је погодније користити у конкретном случају и према томе и поступити.

3.4.2.2. Метода праћења ока

Метода праћења ока (енг. *eye tracking*) је специфична метода тестирања употребљивост, јер се праћењем покрета очију корисника који користе Web апликацију утврђује когнитивно процесирање (Plantak Vukovac & Orehovački, 2010, p. 174). Ова метода може да послужи да се утврди где корисници гледају и колико дуго, како се фокусирају од елемента до елемента на Web страници, који делови интерфејса нису прегледа и тд.

Уређај за праћење прати зеницу ока корисника који гледа Web страницу, где су важне тачке на које се корисник фокусира, тзв. „фиксације“ (енг. *fixation*) и генерише податке у облику топлотних мапа (енг. *heat maps*) које су репрезентација фокуса погледа корисника и сакадних путева (енг. *saccade pathways*) преко којих се прати померај ока између области фокуса (usability.gov, n.d.). Ову метода се користи уз неку другу методу провере употребљивости, јер она самостално не може да објасни зашто нешто корисници раде, она показује шта они раде. Поред тога, потребно је размотрити да ли је вредно инвестирање у средстава, као и у експерте који могу да објасне и презентују резултате ове методе.

3.4.2.3. Методе испитивања

Да би се добиле повратне информације од корисника у процесу провере употребљивост користе се методе испитивања. Путем њих корисници исказују свој субјективан став о елементима корисничког интерфејса који су од интереса за истраживаче, да ли им се свиђају или не. Могу се спровести вербално, а могу и у писаној форми.

3.4.2.3.1. Интервју

Интервју је техника где интервјуер прикупља информације које га занимају од корисника, које он бележи. Може се организовати као структуриан и неструктуриан интервју. Структурирани интервју има унапред дефинисан распоред питања и усмеравање интервјуа у једном правцу. Неструктурирани интервју је организован тако да интервјуер

добије што је могуће више информација о корисничком искуству, као и да испита каква су очекивања корисника од система. Могу се примењивати у било којој фази развоја производа.

3.4.2.3.2. Упитник

Упитник се користи да прикупи одговоре корисника у вези корисничког интерфекса и њиховог задовољства и одређених ставова који они имају. Најлешће се спроводи попуњавањем упитника који се налазе на Web-у, где се путем неке платформе одговори корисника памете и могу се након тога додатно обрадити. Предност оваквог начина рада је да је могуће за релативно кратко време прикупити велики број одговора и спровести одговарајуће статистичке анализе.

Остале методе за евалуацију употребљивости, уз оне које су презентоване овде дате су у кратким цртама у раду аутора (Plantak Vukovac & Orehovački, 2010). Категоризоване су на сличан начин који је изложен овде.

3.4.3. Методе за евалуацију квалитета Web апликација

Web технологије од како су се постале доступне становништву, након њиховог настанка у војној индустрији па након тога и у истраживачким центрима пре него што су постале доступне глобалној популацији, бележе константан раст са аспекта употребе. Оне су данас део свакодневног живота, од провере електронске поште, претраживања садржаја, употребе социјалних мрежа, употребе мобилних апликација, електронске трговине, као и електронског пословања и тд. Готово да нема области људског живота у којој Web технологије нису укључене на одређени начин.

Развој Web технологија и програмерских парадигми које функционишу у његовој позадини се одвијао доста интензивно од самих почетака и употребе CGI (енг. *Common Gateway Interface*) модела програмирања, Java аплета (енг. *Applets*), затим обезбеђивања понашања преко JavaScript технологије на страни клијентског дела апликације, преко класичних клијент-сервер Web апликација, употребе Flash технологије и додавање додатака (енг. *Plug-In*) у претраживаче, стварање утиска да су Web апликације сличне desktop апликацијама путем концепата богатих Internet апликација (енг. *Rich Internet Applications*, RIA) којима се нуди боље корисничко искуство, употребе Ajax технологије у стварању богатијег корисничког искуства асинхроним разменом података, смањивање саобраћаја и величине размењених докумената између клијентског и серверског дела апликације, уз употребу JSON формата, као и концепта апликације на једној страници (енг. *Single-Page Applications*) и тд.

Уз напредак временом је дошло до померања са концепта Web 1.0, на концепт Web 2.0, где се Internet посматра као платформа која омогућава већу интерактивност (Vesić & Minović, 2015, p. 48). Поменути тренд посматрања Internet-а као платформе је утицао и на развој хибридних апликација, које треба да обједине најбоље карактеристике мобилних и Web апликација (Vesić, 2016, p. 47), као што су прогресивне Web апликације (енг. *Progressive Web Apps*, PWA). Интеракција корисника са Web апликацијама у концепту 2.0 је значајно интензивнија и са пажњом треба приступити у одабиру технологије и начина њене имплементације да би се задовољиле карактеристике употребљивости, а когнитивни замор свео на најмању могућу меру (Yuuki & Kashimura, 2010).

Све поменуте технолошке промене које су настале током година су у битној мери промениле однос људи који дизајнирају и изграђују Web апликације према интеракцији корисника и Web интерфејса, са акцентом на употребљивости Web апликација. Како је употребљивост једна од значајних карактеристика квалитета софтвера, јасно је зашто постоје разне методе евалуације употребљивости да би се осигурао поменути квалитет Web апликација. У наставку ће бити приказане одабране методе за евалуацију квалитета Web апликација.

3.4.3.1. *WebQEM*

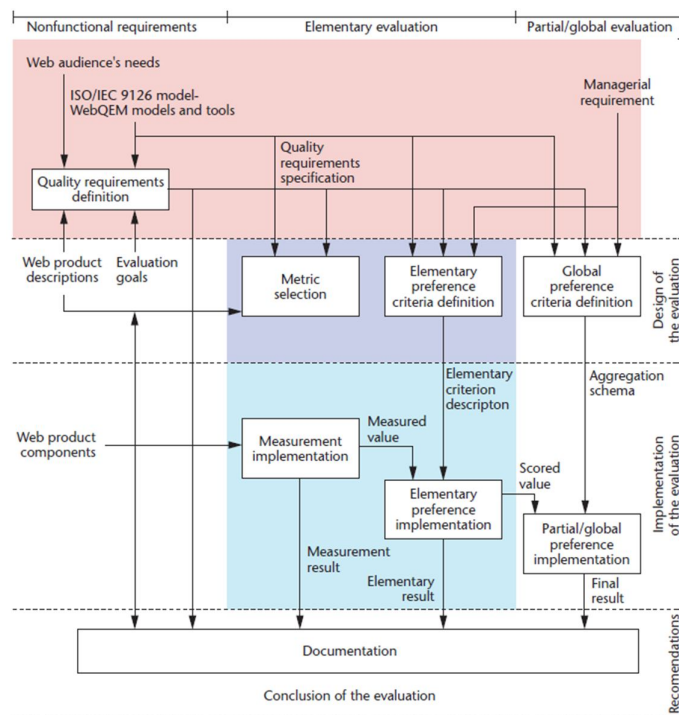
WebQEM је поступак евалуације квалитета Web апликације, којим се мере одговарајући индикатори квалитета и на основу тих мера могуће је унапредити и побољшати Web апликацију (Olsina & Rossi, 2002, p. 20). Модел квалитета коришћен у овој методи користи карактеристике дефинисне у ISO 9126-1 стандарду: употребљивост (енг. *usability*), функционалност (енг. *functionality*), поузданост (енг. *reliability*), ефикасност (енг. *efficiency*), портабилност (енг. *portability*) и одрживост (енг. *maintainability*).

WebQEM метода је коришћена у евалуацији квалитета различитих Web сајтова, као на примеру нафтне индустрије Ирана и разних Web портала министарства у Ирану (Shahvari, 2019, p. 252). Затим је спровођена евалуација квалитета сајтова светских финансијских институција, затим финансијских институција у Румунији, као и неких универзитетских сајтова. Поред тога вршена је евалуација квалитета Web сајтова ресторана у Малезији.

Поступак евалуације у WebQEM методи се спроводи у четири фазе (Olsina & Rossi, 2002, p. 20):

- дефинишу се захтеви за квалитетом и спецификација - прво дефинишу циљеви евалуације, одабере модел квалитета, идентификују Web компоненте и њихов значај у односу на намераване групе корисника, затим дефинишу се групе корисника и након тога карактеристике, подкарактеристике и атрибути квалитета чиме се заправо креира спецификација
- спроводи се елементарна евалуација у 2 корака: дизајн елементарне евалуације где се дефинише функција елементарних критеријума која ће дати елементарни индикатор или вредност преференце, импеметација елементарне евалуације примењује одабране метрике на Web апликацији
- спроводи се глобална евалуација такође у 2 корака: дизајн и имплементација парцијалне и глобалне евалуације квалитета, где се у фази дизајна агрегирају критеријуми и скор модел којима ће се касније срачунати парцијална и глобална евалуација
- износи се закључак, тј. предлози за повољшање - компоненте Web производа, захтеви за квалитетом, метрике и критеријуми, као и сачувани елементарни и крајњи резултати, затим се спроводи анализа и узврђују снаге и слабости производа у односу на циљ у кориснички поглед и дају одређене препоруке и предлози за побољшање

На приказан процес.



слици је WebQEM

Слика 13. WebQEM методологија, презето (Olsina & Rossi, 2002)

3.4.3.2. WAMMI

WAMMI (Website Analysis and Measurement Inventory) је алат који служи да измери корисничко искуство Web сајта на основу реакција посетиоца употребом упитника који има 20 питања, а поред тога може и да изврши поређење Web сајта са другим Web сајтовима из базе и да генерише једноставне извештаје корисне за менаџмент. Друге опције укључују анализу квалитативних коментара и реакција на на ваш Web сајт од стране корисника и интерпретацију квантитативних и квалитативних података у циљу да се одреди шта тачно да се побољша (WAMMI (*Website Analysis and Measurement Inventory*), n.d.). Основна сврха овог алата јесте да на основу очекивања корисника, утврди њихово корисничко искуство постављањем питања. Одговори корисника се пореде сада подацима из базе података која има у себи преко 320 анкета и затим се дају одговори који могу да укажу на корисничко задовољство. Још једна специфичност одликује WAMMI алат, а то је да он у позадини користи психометријске технике и веома је поуздан (WAMMI, n.d.). Услед својих добрих карактеристика, користи се и као научни аналитички сервис.

У литератури WAMMI сервис се користио у пракси да измери употребљивост на Намик Кемал универзитету у Турској где су одабране карактеристике употребљивости атрактивност (енг. *attractiveness*), управљивост (енг. *controllability*), корисност (енг. *helpfulness*), ефикасност (енг. *efficiency*) и лакоћа учења (енг. *learnability*) (Ahmet Mentis & Turan, 2012, p. 63). Поред примене у домену образовања, WAMMI се користио и у анализи употребљивости за Web сајт електронске трговине, где је истраживање показало да је атрактивност најважнији фактор који је потребно да привуче кориснике, омогући добро корисничко искуство и испуни мисију једног предузећа у том домену. Поменуто истраживање није користило опцију поређења уграђену у WAMMI са другим Web сајтовима за електронску трговину, па је то одређена мањкавост спроведеног истраживања (Ahmad et al., 2017, p. 142). На основу изложених примера види се да није везана само за један домен.

Корисници на упитник могу да дају своје мишљење, које може да се креће од става да се у потпуности слажу до тога да се не слажу са изнетим тврдњама, нека питања (ставови) на пример могу да буду (WAMMI, n.d.):

- Странице на овом Web сајту су веома атрактивне.
- Могу да брзо пронађем оно што ми је потребно на овом Web сајту
- Овај Web сајт ми се чини логичан.
- Тешко ми је да запамтим где сам раније био на овом Web сајту.
- Осећам се ефикасно када користим овај Web сајт.

Анализом ставова испитаника, добија се оцена корнисчког искуства одређеног Web сајта за кога се постављају питања, тј. износе ставови.

4. Наслеђени системи

4.1. Старење софтвера

Софтвер је веома сличан људима када га посматрамо из угла старења, јер као што људи старе, тако и софтвер стари, како је приметио Парнас један од зачетника области софтверског инжењерства (Parnas, 1994). Старење софтвера (енг. *software aging*) је неминовност и појава која траје од увођења софтвера у експлоатацију до тренутка када се повлачи из употребе.

Како је софтвер део ширег информационог система једне организације, а сам информациони систем модел реалног система, промене које долазе из окружења и које делују на организацију делују на начин да је потребно изменити информациони систем, па самим тим и софтвер који је веома важан део њега. Уколико се то не учини софтвер постаје застарео и неупотребљив. Уколико посматрамо одређени финансијски софтвер путем којег се обрачунава камата употребом конформне методе обрачуна затезне камате може се догодити да је посматрани софтвер био у употреби од стране организације дужи низ година. Изменом Закона о Затезној камати, изменио се начин обрачуна затезне камате са конформне на пропорционалну методу у Републици Србији. Уколико финансијски

софтвер предузећа није измењен тако да подржава нови прописани начин обрачуна камате он застарева и постаје неупотребљив.

Други део посматраног феномена је везан за нарушавање иницијално осмишљене структуре и понашања програма, које се догађа услед вишеструких промена, а поготово у ситуацији када је софтвер комплексан. Да би се измена у софтверу спровела на адекватан начин потребно је његово детаљно познавање, што у пракси значи познавање програмског кода на начин да је промену потребно спровести само у секцији програмског кода која је задужена за реализацију жељене функционалности. Није редак случај да људи који су у почетку радили на развоју софтвера напусте организацију, а за одржавања софтвера задуже се људи који не разумеју иницијално осмишљени дизајн. Ситуација се погоршава уколико документација не постоји или није ажурна, јер тада не постоји трансфер знања о систему. Измене над програмским кодом које се спроводе да би се одговорило на промене које долазе из окружења, доводе до нарушавања иницијално осмишљеног дизајна и нарушава се структура програма. Када се спроведе већи број оваквих промена догађа се да структура софтвера постане неразумљива, а он веома тежак, ризичан и скуп за одржавање и само је питање времена када ће нова измена довести до отказа и нових грешака у програму (енг. *bugs*). Дакле, старење софтвера је неизбежно и како аутор наводи у његовој основи су два различита узрока (Parnas, 1994, р. 280):

- први је последица неуспеха у модификацији софтвера тако да он може да одговори на промене
- други је резултат промена које су направљене које воде деградацији структуре софтверског производа

На сличне појмове али са фокусом на софтверску архитектуру указују аутори који разликују ерозију архитектуре (енг. *architecture erosion*) и архитектурално скретање (енг. *architectural drift*) (Perry & Wolf, 1992, р. 43). Ерозија архитектуре се односи на нарушавање архитектуре система које води значајним проблемима система увећавајући његову крхкост. Архитектурално скретање односи се на неосетљивост архитектуре које води ка немогућности прилогагођавања, а затим и до смањења повезаности у систему и јасноће његове форме. Уобичајено настаје када имплементација програма надиђе његову сврху и иницијалну пројектантску замисао.

Постоји неколико индустријских примера који указују на ерозију архитектуре која проузрокује поновни развој одређених делова система уз велике напоре, при чему

аутори (Van Gurp & Bosch, 2002) наводе неколико примера. Први је развој Mozilla Web претраживача када је компанија Netscape одлучила да након губитка тржишне битке са компанијом Microsoft, програмски код свог претраживача учини отвореним кодом (енг. *source code*). Неколико верзија дела програмског кода је повучено и поново прављено услед тога што се нису поштовале оптималне одлуке по питању дизајна, а неки делови програмског кода су у потпуности повучени још пре завршетка целог система, јер је у међувремену дошло до промене захтева.

Други пример је на шведска компанија Axis AB која је након неколико година употребе свог власничког оперативног система за производе које је тада развијала прешла на верзију Linux оперативног система. Поменуга транзиција је трајала око 2 године, а све време је власнички оперативни систем одржаван у паралели.

Трећи пример се односи на развој кернела Linux оперативног система, који је захтевао огроман напор да се реструктурира програмски код са верзије 2.2 и верзије 2.3 на верзију 2.4. Аутори указују да успех поменугих пројеката није у потпуности, јер поновни развој поред тога што је изискивао много ресурса, он је уједно био и делимично спроведен, а при томе су га спроводиле само оне компаније које су идентификовале ране знаке ерозије (Van Gurp & Bosch, 2002, р. 106). Сви наведени примери указују да је ерозија архитектуре сложен проблем и да неадекватне одлуке по питању архитектуре могу на дугорочно да произведу значајне проблеме, које је тешко у целости исправити уколико у старту нису исправне.

Последице старења софтвера доводе до губитка тржишног дела колача, јер се не могу испратити континуалне промене у потребама корисника софтверских решења које конкурентни успевају да задовоље. Са новим променама на софтверу настају грешке које додатно смањују задовољство корисника, где у неким случајевима они може имати и губитке у пословању произведене нетачним извршавањем одређених операција или вишеструким отказима. Перципирани квалитет софтвера опада, јер је иницијална пројектантска замисао нарушена, а то најчешће води да корисник уочава пад одређених перформанси и само је питање времена када ће напустити постојеће и прећи на ново софтверско решење.

Старење софтвера је неизбежно и иако постоје многе технике и приступи који смањују брзину, тј. интензитет старења као што су рефакторисање кода (енг. *code refactoring*) или употреба пројектних шаблона (енг. *design patterns*) или принципа раздвајања одговорности (енг. *separation of concerns*) и друге, потребно је припремити се за тренутак када ће се софтвер повући из употребе и завршити свој животни циклус.

4.2. Одржавање софтвера

Када се одређено софтверско решење испоручи крајњим корисницима и стави у употребу, тиме се означава прекид његовог процеса развоја и тада наступа процес одржавања (енг. *maintenance*). Најопштије речено, одржавање подразумева све оне активности које се спроводе у циљу измене система након његовог увођења у експлоатацију (Pfleeger & Atlee, 2009, p. 535). Да би се измене спровеле на адекватан начин потребно је детаљно познавање структуре посматраног система и разумевање његовог понашања.

Документација има значајну улогу у процесу одржавања, јер програмерима који раде на пословима одржавања скраћује време у упознавању софтвера над којим треба да спроводе активности одржавања, поготово у ситуацији када га нису иницијално развијали. Поред чињенице да је важно да документација постоји, потребно је имати и у виду да је битно да она буде ажурна на начин да описује актуелну верзију софтвера. Уколико то није случај, њен значај и улога у процесу одржавања се смањује и време и напор који је потребно да програмери уложе у разумевање софтвера који одржавају значајно расте. Документација настаје у процесу развоја софтвера и уобичајено је да она садржи захтеве, дизајн, документован код, водиче и упутства за крајње кориснике и тд. Приликом процеса одржавања потребно је ажурирати документацију тако да се све спроведене промене у систему прикажу и у самој документацији. На поменути начин одржава се њена ажурност.

Процес одржавања софтвера је специфичан са становишта програмера, јер од њих захтева значајно познавање истог, а знање настаје као последица вишегодишњег њиховог ангажовања на поменутом проблему. Обзиром да у пракси постоје софтвери без којих организације не могу да обављају своје пословање, улога програмера је по својој природи интензивнија у односу на процес развоја софтверског решења од почетка, тзв. *greenfield project* (енг. *greenfield project*). Самим тим, програмер који ради на одржавању у великој мери је потребно да познаје разна ограничења које одређени софтвер има у себи, да сагледа нове разноврсне захтеве који су пред њим и да спроведе све потребне активности у циљу очувања функционалност или превенције потенцијалних проблема и отказа.

Оправданост одржавања софтвера је вишеструка (Grubb & Takang, 2003). У пракси постоје тако уређена пословања организација да одређени софтвер је критичан за обављање неких активности и готово је неодвојив од њих и он једноставно мора да настави да буде у експлоатацији. Пример таквог софтвера може бити софтвер који

управља одређеном инфраструктуром или је задужен за њено надледање, као што је нпр. употреба SCADA система у електродистрибуцији или у водоводима. Уколико софтвер који је потребно да процени количину одређеног хемијског елемента у води која се спрема за процес пречишћавања и да након тога покрене аларм у систему када се пређе горња граница концентрације тог елемента, откаже тада последице могу бити велике по здравље грађана. Специфичност описаних система јесте да уколико откажу или постоје неидентификоване грешке, цена је веома висока, као што је губитак људских живота, значајно угрожавање здравља или безбедности и спречавање испоруке најважнијих ресурса или услуга.

Други разлог настаје због тога што софтвер функционише као део веће целине информационог система организације, а сама та организација обавља своје пословање у променљивом окружењу, па промене из окружења могу условити промене у информационом систему и активности у вези са одржавањем софтвера. Један од примера може бити промена стопе пореза на додатну вредност за општу стопу ПДВ-а са 18% на 20%, на основу Закона о порезу на додату вредност.

Сваки програмер програмира софтвер којег неко употребљава и један од кључних захтева јесте да се софтвер прилагоди крајњим корисницима, који када увиде његове функционалности и могућности које им доноси, крећу да га користе. Недуго затим иако су задовољни неким функционалностима, проналазе да друге функционалности не задовољавају у потпуности њихове потребе, а негде проналазе да је потребно обезбедити нове боље функционалности. Њихови захтеви нису искључиво функционалног карактера, тзв. функционални захтеви, већ могу бити нефункционални захтеви, када је потребно побољшати перформансе система да би њихово корисничко искуство било боље. Један од примера може бити прилагођавање софтвера тако да одговара одређеном региону или народу одређеног говорног подручја, а који је другачији од оног у испорученом софтверу, што може значити креирање могућност промене језика на корисничком интерфејсу. Поред тога, захтев ка побољшању перформанси може бити такав да се омогући брже време одговора софтверског система. Анализом базе података и прегледом кода програмер може да утврди да постојећи програм не користи ефикасно систем за управљање базом података, па може креирати индексе путем којих бих се омогућио бржи приступ табелама у релационој бази података. Као последицу његове измене у бази података и промене софтверског кода, корисник може брже вршити операције приказа, креирања, измене или брисања у бази података, што може бити важно уколико су корисници оператери контакт

центра и који је потребно да приликом комуникације са купцима брзо, прецизно и поуздано им пруже информације.

У условима савременог пословања, тежи се да одређени производ, па тиме и софтвер, што пре буде на тржишту. У таквој тржишној утакмици могуће је жртвовати многе принципе и праксе доброг развоја софтвера да би се постигао жељени ефекат. Уколико је такав испоручени софтвер у употреби, њему предстоје значајне активности на његовом одржавању. У неком тренутку значајне измене над базом података и програмским кодом постају извесне. На описани начин трошкови одржавања могу да неконтролисано порасту и како систем све више стари и потребно је чинити све више промена, а трошкови још више расту, док у неком тренутку постане неисплативо одржавати га. Уколико се ради о системима који су критични за пословање или су веома значајни да би њихово повлачење, без адекватне замене, проузроковало катастрофу по становништво или довело до фаталних последица, организације су приморане да издвајају велика средства да одржавају поменути софтвер. Није редак случај да се организације дугорочно вежу за неког произвођача власничког софтвера, без могућности да једноставно пређу на нека нова и савременија решења. У неким организацијама се доносе одлуке, које се тичу евалуације иницијалног дизајна, па чак и промена у дизајну или коду уколико се утврди да је то оправдано са аспекта дугорочне употребе и одржавања поменутог софтверског система. Добро дефинисана архитектура система у значајној мери може олакшати одржавање.

Према ISO/IEC 14764:2006 стандарду постоји неколико врста одржавања (ISO/IEC JTC 1/SC 7, n.d.-c):

- корективно одржавање (енг. *corrective maintenance*)
- адаптивно одржавање (енг. *adaptive maintenance*)
- перфективно одржавање (енг. *perfective maintenance*)
- превентивно одржавање (енг. *preventive maintenance*)

Корективно одржавање се спроводи када настану грешке и откази након испоруке софтвера који могу бити грешке у дизајну (енг. *design errors*), логичке грешке (енг. *logic errors*) и грешке у коду (енг. *coding errors*) (Osborne, 1985). Спроводи се измене да би се уочени проблеми исправили и софтвер вратио у исправно стање. Одликује се активностима мањег обима које се фрквентно спроводе. Обзиром да корисници софтверских решења ће веома брзо уочити неправилности у функционалностима које

спроводе на дневном или периодичном нивоу, са аспекта програмера је важно брзо одреаговати и вратити систем у редован начин рада.

Грешке у дизајну најчешће настају у пракси као последица лошег разумевања захтева за изменом софтвера у комуникацији на релацији тима пројектаната и програмера и крајњих корисника. Промене које је потребно спровести захтевају и значајно разумевање постојећег система од којег се креће па ће пројектна замисао која је потребна да буде спроведена бити резултат знања и умећа пројектанта и програмера, уз познавање система и адекватано схваћене захтеве. Документација је значајна у овом случају јер уколико је добро израђена и ажурна значајно може олакшати рад. Уколико се поменути процес не спроведе адекватно могу настати бројне грешке, које се после теже отклањају.

Логичке грешке могу бити вишеструке. Када се креира спецификација софтверског решења потребно је да имплементација прати поменути спецификацију, што у пракси не мора бити случај. Услед да програмери не разумеју или не поступају по спецификацији или да сама спецификација није ажурна довешће до нетачне имплементације. Други вид ових грешака је везан за грешке које настају као последица нетачних тестова, који воде до погрешних закључака. Трећи вид је нетачна логика, која проузрокује неисправност операција.

Грешке у коду су последица нетачне имплементације, што води ка неочекиваном понашању софтвера или немогућности његовог извршавања па чак и опадања перформанси система.

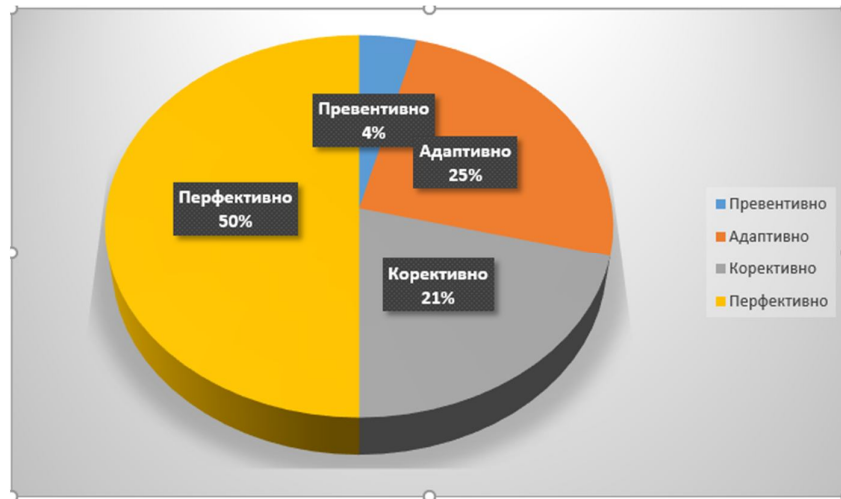
Адаптивно одржавање настаје као последица свеукупних промена у окружењу, где се чине сви напори потребни да софтверски систем настави да функционише у том окружењу (James McCall et al., 1985, p. 10). Окружење може бити дефинисано веома широко, а оно што га одликује јесте да су поменути утицаји ван контроле људи који одржавају софтвер. То су пре свега промене у разним софтверским и хардверским платформама, али и промене у пословним правилима, као и промене у политикама влада и уредбама које уређују пословање. Поред тога нове безбедносне претње из сајбер простора такође могу утицати да се спроводе одређене активности у вези одржавања којим би се поменуте спречиле.

Перфективно одржавање је везано за проширивање функционалности постојећег система захтеваних од стране корисника, које се односе модификовање, побољшавање, убацивање нових и уклањање постојећих. У пракси се уобичајено реализује проширивањем или додавањем нових програмских модула који омогућавају

функционалности које очекују крајњи корисници. Ова врста одржавања обухвата измену неких аспеката система (Pfleeger & Atlee, 2009, p. 544) чак и када они нису последица грешака, па се због тога уједно и назива одржавање ради усавршавања. Потребно је имати у виду да промене могу да буду неочекиване за крајње кориснике и могу произвести незадовољство, тако да је важно да се са њима обави адекватна комуникација о томе какве ће промене наступити у систему у којем они у дугом низу година обављају свој посао. У складу са тим можда ће бити потребно и планирати нове обуке где би са на новим или измењеним програмским функционалностима приказале предочене измене.

Превентивно одржавање спроводи се да уочене грешке након испоруке, а које нису још значајно утицале на нарушавање функционалност система се отклоне на време, да би се спречиле последице (ISO/IEC JTC 1/SC 7, n.d.-c). Улога није да унапреди основне функционалности, већ да се будуће одржавање учини једноставнијим. Програмер спроводи овај тип одржавања најчешће без знања корисника, анализом тренутног софтвера и уочавања потенцијалних проблема или слабих тачака које ће касније исправити. На поменути начин превентивно одржавање је одређени вид припреме за спровођење будућег одржавања.

Све врсте одржавања корективно, адаптивно, перфективно и превентивно су заступљене у организацијама, при чему постоји дистинкција у њиховом уделу у целокупном одржавању. Корективно одржавање је заступљено нешто мање од четвртине 21%, адаптивно има удео једне четвртине, перфективно има највећи део тачно једну половину, док је најмањи удео превентивног 4%, што се може видети на слици (van Vliet, 2007, p. 17). Исте резултате које презентује Ван Влиет, наводе и аутори Флигер и Атли који се позивају на обимно истраживање Линца и Свансона у вези управљања софтвером и одржавања софтвера крајем 70-их (Pfleeger & Atlee, 2009, p. 545). Истраживање је спроведено анкетањем 487 менаџера (Lientz & Swanson, 1981). Носек и Палвиа спроводе ново истраживање деценију након тога и долазе до сличних резултата (Nosek & Palvia, 1990). Имајући у виду напредак у методолошким принципима пројектовања, као и техникама примене, уз модерније софтверске алате поменути резултат није било за очекивати.



Слика 14 . дистрибуција активности одржавања, преузето (van Vliet, 2007)

У пракси је чест случај да трошкови одржавања постојећег софтвера надилазе трошкове развоја новог софтвера. Разни аутори предлажу различите односе између одржавања постојећег и развоја новог софтвера (Hanby, n.d.). Разлике у резултатима су последица различитих софтверских пројеката на којима се посматрају поменуте активности, као и непостојања јединствене методологије по којој би се утврдили поменути односи. Неки од аутора као што је Коскинен (Koskinen, 2015) указују да одржавање чини преко 90% укупно опредељених средстава, док рецимо Шаш наводи 67% (Schach, n.d.). Галорат истиче да тај однос иде и до 75% у корист одржавања (Galorath, 2014). Сви презентовани односи између одржавања и развоја новог система указују да је обим средстава за одржавање значајнији, а то је познато и у пракси јер већина софтверских кућа не остварује приход на првој продаји софтвера, већ на дугогодишњем одржавању.

Високи трошкови одржавања су производ великог броја фактора који утичу на њих. Аутори наводе техничке факторе као што су сложеност софтвера (енг. *software complexity*), развој људског капацитета (енг. *development of human capacity*), квалитет документације (енг. *documentation quality*), технологија за управљање конфигурацијом (енг. *configuration management technology*), модерна програмерска спецификација (енг. *modern programming specifications*) и величина базе података (енг. *database size*) (Ren et al., 2011, p. 3). Нетехнички фактори су апликационо искуство (енг. *application experience*), стабилност запослених (енг. *staff stability*), примена током времена (енг. *application of time*), екстерно окружење (енг. *external environment*), подршка за окружење (енг. *support*

environment) и потребе корисника (енг. *user needs*) (C. Singh et al., 2019, p. 279). Дејством техничких и нетехничких фактора трошкови одржавању си високи и претња су да угуше развој новог система или неких нових његових делова. То је посебно изражено када је цео информациони систем настао у предузећу, тј. када је решење у кући (енг. *in-house*), јер развојни тим увек има обе врсте активности на различитим пројектима.

4.3. Еволуција софтвера

Све већа употреба рачунарски заснованих информационих система допринела је развоју софтверских решења у разне сврхе. Временом дошло је до употребе појма софтверско инжењерство као одговора на проблеме у пракси које су се јавили у развоју сложених система. Ови проблеми су уобличени у појави познатијом под именом софтверска криза и она указује на потребу да процес развоја софтверског система буде организован, поуздан и исплатив.

Упоредо са овим догађајима представљени су први концепти у којима се процес развоја софтверског система спроводи кроз фазе када је Ројс представио водопад модел животног циклуса развоја система. У пракси овај модел се показао као нефлексибилан јер иако раздвајање развоја софтверског система на фазе има своје предности, проблеми настају услед промена које се могу јавити у било којој фази животног циклуса. Из поменутог разлога можда је потребно вратити се на неке раније фазе у развоју софтверског система и допунити га новим информацијама које су раније биле недоступне. У софтверској пракси није редак случај да након узимања програмског захтева и реализације првих софтверских функционалности корисници софтвера након што имају прву интеракцију са системом увиђају да су им потребне још неке функционалности, па они рedefинишу своје захтеве тако што их допуњују или мењају. Корисници упоређују доменско знање које поседују настало путем вишегодишњег рада на одређеном домену и функционалности које пружа софтверско решење које обрађује поменути домен.

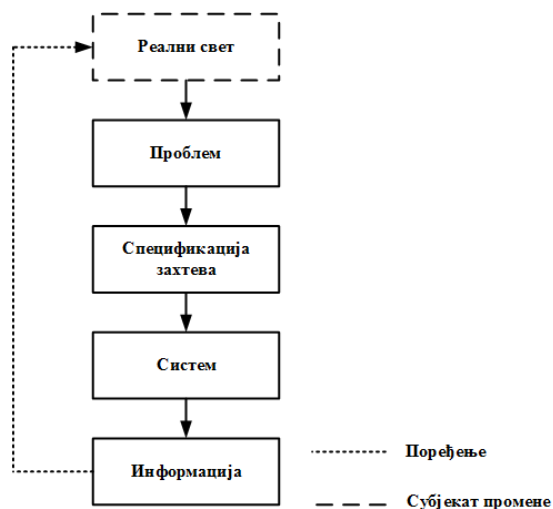
Секвенцијална природа водопад модела се показује као нефлексибилна и одступа од процеса развоја софтвера у стварности па су многи даљи приступи у развоју софтвера инкорпорирани могућност повратне информације из каснијих фаза развоја софтвера у раније. У овом моделу процес одржавања софтвера се посматра као једна од фаза животног циклуса развоја система након што је он ушао у употребу.

Још један проблем у концепцији водопад модела животног циклуса развоја система јесте да он занемарује старење софтвера као неизбежну појаву у животном веку

софтвера. Да би се направила дистинкција између појма одржавања софтвера који се у периоду 70-их посматрао као процес једноставног уклањања грешака (Mens, 2008, р. 2), Леман је дефинисао појам еволуције софтвера (енг. *software evolution*) као потребу да се систем мења током свог животног циклуса. Корисници мењају своје захтеве или настају друге промене у окружењу и тада систем има потребу да се прилагоди тим променама.

У односу на промене из реалног света које је потребно превести у апстрактни свет софтвера и програма који су део информационог система, као упрошћеног модела реалног система, Леман разликује 3 класе програма: S, P и E програме (називају се и системима) (Meir M. Lehman, 1980). Поменуте класе система су основа за разумевања еволуције софтвера и потребе да се ИС прилагоди променама.

S системи (енг. *S-systems*) су формално дефинисани системи изведени из спецификације, код којих су све околности неког проблема над којим се примењује систем познате (Pfleeger & Atlee, 2009, р. 536). S системи су приказани на слици 15.

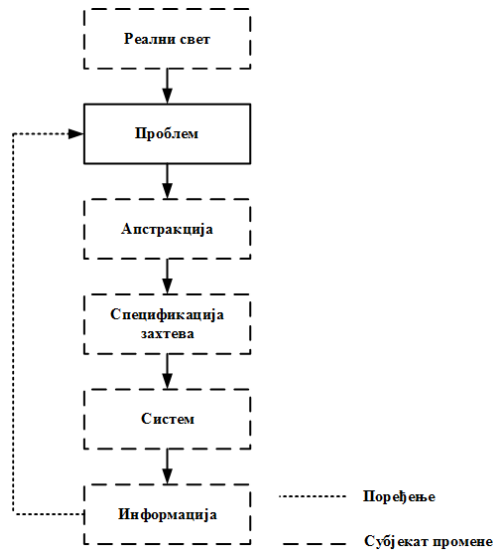


Слика 15. S системи, преузето (Pfleeger & Atlee, 2009)

Спецификација усмерава и контролише програмера ка жељеном решењу чији је једини посао да води рачуна о коректности и ефикасности имплементације, а са променом у реалном систему дефинише се нов проблем за који се креира новог решење (Meir M. Lehman, 1980). Један од примера може бити имплементација одређене математичке функције, где се познати проблем, решава познатим решењем.

Друга класа система су P системи (енг. *P-systems*) и они су засновани на апстракцији проблема, уместо на у потпуности дефинисаној спецификацији (Pfleeger &

Atlee, 2009, p. 537). Решење зависи од окружења у којем систем функционише и ако информација која се добија из решења не одговара реалном систему, апстракција тог проблема се мења. Р системи су приказани на слици 16.

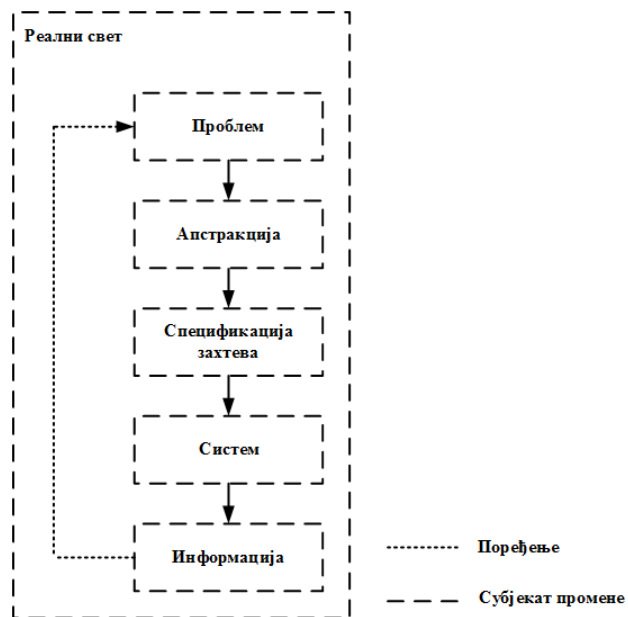


Слика 16. Р системи, преузето (Pfleeger & Atlee, 2009)

Један од примера Р система (или програма) би била игра шаха, где теоретски постоји могућност да се одигра савршена партија, али имплементационо то није изводљиво услед великог броја комбинација које постоје.

Трећа класа су Е системи (енг. *E-systems*) и они су од интереса за овај рад јер информациона системи и софтвер који је саставни део њега одговара овој класи система.

Леман указује да је програм постао део света који моделује, јер је уграђен у њега (Meir M. Lehman, 1980). Е системи су приказани на слици 17. За разлику од претходне две класе система у којима се окружење не мења тј. статичко је, Е системи се мењају заједно са окружењем у којем функционишу. Над системима који функционишу у реалном свету спроводе се континуалне исправке, унапређења и проширења да би обезбедило да одговара стању реалног система.



Слика 17. Е системи, преузето (Pfleeger & Atlee, 2009)

Током рада на IBM OS 360 оперативном систему, Леман је започео дефинисање закона еволуције софтвера, које је допунио током наредних 20 година. Он дефинише 8 закона еволуције софтвера (М. М. Lehman, 1996):

1. Стална промена (енг. *continuing change*)
2. Увећање комплексности (енг. *increasing complexity*)
3. Саморегулација (енг. *self regulation*)
4. Конзервација организационе стабилности (енг. *conservation of organisational stability*)
5. Конзервација фамилијарности (енг. *conservation of familiarity*)
6. Континуирани раст (енг. *continuing growth*)
7. Опадање квалитета (енг. *declining quality*)
8. Систем повратне спреге (енг. *feedback system*)

Стална промена се односи да софтвер којим се презентује нека друга стварност континуално се мења и прогресивно је све мање користан (Stefanović et al., 2016, p. 541). Тада га је исплативије заменити новим, јер толико деградирао да је неисплативо одржавати га.

Увећање комплексности указује да са еволуцијом комплексност програма се увећава уколико се не предузму активности које имају за циљ да је смање и одрже на

прихватљивом нивоу. Како еволуција тече везе између елемената, интеракције и зависности у систему се увећавају што води ка већој ентропији система, па је потребно изнаћи начине да се комплексност ограничи (М. М. Lehman, 1996, р. 1). Примери добре праксе указују да се комплексност може смањити применом принципа лабаве повезаности, адекватном употребом шаблона за дизајн, употребом SOLID принципа и тд.

Саморегулација означава да је процес еволуције саморегулативан и приближан нормалној дистрибуцији у мерама производа и атрибутима процеса а то значи да су величина, време између две верзије и број грешака су приближни за сваку верзију истог софтвера (М. М. Lehman, 1980, р. 216). Овај закон се назива и фундаментални закон еволуције софтвера.

Конзервација организационе стабилности указује да у неком тренутку ресурси и излази достижу оптималан ниво и да додавање нових ресурса не мења значајно излаз (Stefanović et al., 2016, р. 541). Назива се још и непроменљива стопа рада.

Конзервација фамилијарности указује да је садржај верзије (промене, додавања, брисања) наредних верзија програма који еволуира статистички непроменљив, тј. инваријантан (М. М. Lehman, 1980, р. 216). Овај закон указује да је потребно држати одређени степен фамилијарности између претходне и наредне верзије, јер превелика промена доводи до тога да знање о систему може да опадне, па један део развојног тима не може да је испрати. Временом долази до деградације квалитета.

Континуирани раст објашњава да је потребно да се функционалности програма које су подржане одређеним кодом континуално увећавају да би се увећало задовољство корисника током животног века софтвера (М. М. Lehman, 1996, р. 2). Дакле, постоји релација између функционалности које систем обезбеђује и задовољства корисника.

Опадање квалитета указује да Е системи ће се у будућности перципирати као системи којима опада квалитет уколико се ригорозно не одржавају и прилагођавају променљивом окружењу у којем функционишу (М. М. Lehman, 1996, р. 3). То је и логично, обзиром да је старење софтвера неминовност.

Систем повратне спреге означава да је процес програмирања Е система вишеслојан са повратном спрегом и петљама и мора да се третира као такав да би успешно могао да се модификује и унапређује (Stefanović et al., 2016, р. 542). До промена захтева може да дође у било којем тренутку, тако да рад треба да буде методолошки и организационо уређен тако да може да подржи такав вид промене.

Фазни модел еволутивног процеса је приказан на слици и он у себи експлицитно садржи наизбежан проблем старења софтвера (Mens, 2008, р. 2).



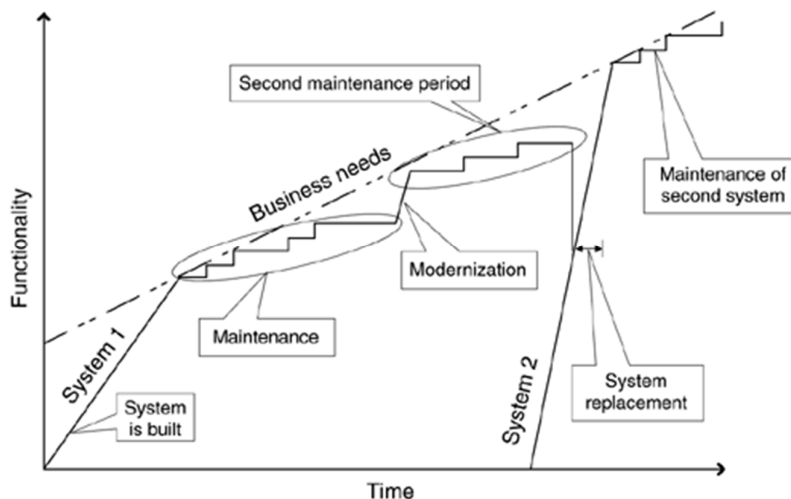
Слика 18. фазни процес модела еволуције (Mens, 2008)

Са слике се види да фаза еволуције омогућава спровођење промена над софтвером. То је могуће спроводити до тренутка када се не нарушава осмишљена структура система, у супротном долази до пропадања (енг. *decay*) јер се на тај начин губи могућност еволуирања (Mens, 2008, р. 3). Трошкови задржавања система у употреби могу бити веома високи па је потребно сваки пут доносити одлуку да ли наставити са његовом употребом или га повући из употребе и прећи на неко ново решења.

Такође потребно је имати у виду да систем може да има значајну вредност за организацију и она не може да га једноставно замени новим решењем без обзира на то што су његова иницијално осмишљена структура и његово понашање значајно деградирани у квалитету.

Иако постоје одређена неслагања и преклапања у појмовима одржавања система и еволуције, постоје одређени аутори који указују да је целокупна еволуција система подржана са више различитих активности, у који се разликују: одржавање, модернизација (енг. *modernization*) и замена (енг. *replacement*) система (Weiderman et al., 1997). Активности се разликују према степену дискрепанције између онога што систем нуди и потреба пословања у томе тренутку, па ће се у различитим фазама системе применити различите активности. У том смислу активности модернизације захтевају више времена и напора у односу на одржавање (Seacord et al., 2003). На слици 19. је приказан животни циклус једног информационог система, где како систем стари, а пословне потребе расту функционалности система не могу да задовоље те потребе. Након што је систем ушао у употребу, веома брзо када се појави одступање спроводи се одржавање система, које има за циљ да смањи одступање. Обзиром на променљиво ближе и даље окружење организације и информационог система као модела исте, појављују се промене које воде до еволуције система, па је потребно да еволуција буде пресликана и на информациони

систем. Када је та промена већег интензитета, тада се спроводи модернизација, где се функционалност система увећава и доводи у равн са пословним потребама.



Слика 19. различите еволутивне активности, преузето (Seacord et al., 2003)

Након тога, поново се спроводи процес одржавања, који има за циљ да смањи та мала одступања. У једном тренутку дикрепанца је толико велика, да ни процес модернизације не може да је смањи или не може да се спроведе на исплатив начин за предузеће и тада је време за повлачење система из употребе. Тиме се завршава његов животни век, а започиње експлоатација новог система

4.4. Наслеђени информациони системи

4.4.1. Појам и дефиниције наслеђених система

Још од појаве првих информационих система, отпочео је неизбежан процес старења софтвера као једног њиховог чиниоца. Временом услед промена које су наступиле из интерног или екстерног окружења организације, информациони систем је морао да се прилагођава да би могао да одговори на поменуте промене. Процес прилагођавања у најширем смислу омогућио је информационим системима као моделима стварних система да се прилагоде променама и да наставе да несметано спроводе дневне активности, као и да наставе да пружају довољно добру основу за доношење одлука које су од животне важности за организацију. Дакле, за све промене које наступају у реалном свету потребно је спровести и у информационом систему да би он наставио да представља модел реалног система и тиме омогући подршку организацији да остварује

своје визије, мисије и циљеве. Након развоја информационог система наступа процес његовог одржавања, или његове еволуције како неки аутори истичу, да би се разликовали од одржавања, јер одржавање посматрају само кроз поправке ситних грешака у програму након испоруке (корективно одржавање). За потребе овог рада користиће се појам одржавања и ту ће се подразумевати све његове врсте: корективно, адаптивно, перфективно и превентивно, а појам еволуције ће се корисити када желимо да означимо промене које долазе из окружења, па је потребно систем прилагодити истим. Дакле, еволуција ће се посматрати као потребна карактеристика система путем које се систем прилагођава интерним и екстерним променама.

Током процеса одржавања може се догодити да информациони систем и софтвер који се извршава као део њега значајно се измени, тако да не може да одговори на захтеване промене било да су оне настале из интерног или екстерног окружења. Ситуација може бити додатно отежана уколико дође до великог померања у односу на инцијалну пројектну замисао система, тј. његову архитектуру.

Када информациони систем више не може да се прилагодити променама он се повлачи из употребе, јер је прошао његов животни век и тада неки нови систем заузима то место и отпочиње своју фазу експлоатације унутар животног циклуса. Како се софтверски производ може посматрати као било који други производ који има свој животни циклус, повлачење информационог система из употребе може деловати поприлично једноставно, али је пракса показала да је то много већи проблем. Поменути проблем настао је још од времена првих информационих система и њихове комерцијалне употребе па све до данашњих дана и уобличен је у форми наслеђених информационих система или краће наслеђених система (енг. *legacy system*). Наслеђени системи су системи развијени у прошлости, најчешће на старој или застарелој технологији, пројектовани и имплементирани у складу са тадашњим методолошким поставкама и праксама развоја (Vesić, 2020, p. 668). То су системи дизајнирани, имплементирани и инсталирани у радикално другачијем окружењу од текуће ИТ стратегије (Tromp & Hoffman, 2003, p. 2), а при томе не подржавају тренутну ИТ стратегију (Mitleton-Kelly, 2006, p. 55).

Једну од раних дефиниција наслеђеног информационог система су дали Броди и Стоунбрејкер 1995. године по којој је поменути апликација која интензивно употребљава податке, као што је пословни систем заснован на стотинама или хиљадама фајлова података или табела који се значајно опире модификацијама и измени (M. Brodie & Stonebraker, 1995).

Приликом изучавања стратегија миграције, о којима ће касније бити реч, група аутора не даје једну дефиницију наслеђених система, већ указује да критични наслеђени системи организацију стваљају пред многе и важне проблеме, а међу њима најважнији су (Jesus Bisbal et al., 1997):

- ови системи уобичајено раде на застарелом хардверу који спор и скуп за одржавање
- одржавање софтвера је финансијски захтевно, пређење грешака је скупо и временски захтевно обзиром на непостојање документације и генералног недостатка разумевања интерног рада система
- интеграциони напори су поприлично скрајнути услед одсуства јасних интерфејса¹
- наслеђени системи су веома тешки, па чак и немогући за проширење

Алдерсон и Шах, експерти из области дају своје техничко мишљење и указују на чињеницу да је свест о значају наслеђених система и реинжењеринга порасла након проблема Y2K (Year 2000), познатијег под називом „миленијумска буба“ (енг. *millennium bug*) (Alderson & Shah, 1999). „Миленијумска буба“² (миленијумска грешка) је проблем о којем се интензивно говорило у јавности³ 1999. године, где се очекивало да ће бити великих проблема у информационим системима након датума 31.12.1999. године. Већина тадашњих система је користила датум у формату у којој се година памти само преко две последње цифре, па се сматрало да када се пређе у нови миленијум проблеми могу наступити услед тога што ће бити присутне 2 нуле. Логика разних софтверских решења, а најчешће оних финансијских вршила је разна прерачунавања употребом ових датума, тако да није чудно што је у том периоду настала велика забринутост од стране јавности. Једна од најпознатијих финансијских кућа у области ИТ консалтинга Gartner је дао процену да је „цена“ миленијумског бага између 300 и 600 милијарди долара (Halton, n.d.). Иако је постојало јавно негодовање уз велику медијску пажњу, у стварности се није много тога догодило.

¹ посматра се са аспекта пројектовања система, и односи се на део система који је намењен интеракцији са другим системом, а не кориснички интерфејс као место сучељавања корисника и система

² појам се такође користи да би описао специфичну појаву из области истраживања језика и контакта енглеског и српског језика, видети рад аутора Прћића (Prčić, 2000), док ће се у овом раду напред наведени појам користити у контексту наслеђених система

³ уз велики број иностраних извора и у нашим медијима се говорило о томе, као нпр. у чланку аутора Павловића (Pavlović, 1999) у „Гласу јавности“, 1999. године

Може се уочити велика дисперзивност у изнетим дефиницијама и разматрањима појма наслеђених система, а то ће се додатно нагласити изношењем још неких дефиниција наслеђених система, које су у складу са предметом истраживања.

„Да би остало конкурентно, пословање мора да континуално мења своје процесе, понекад радикално, али чешће инкрементално, да би се изборило са својим променљивим окружењем. Као резултат, ИТ системи постају неадекватни при рефлектовању пословних потреба, било оперативно или економично па због тога постају наслеђени системи“ (Liu et al., 1998).

„Наслеђени систем је типично систем за управљање базом података у који су организације инвестирале значајно време и новац и који је инсталиран на мејнфрејму или минирачинару” (Khosrow-Pour, 2007, p. 411).

„Наслеђени систем је софтверски систем који наставља да се користи због цене његове замене или редизајнирања иако је слабе конкурентности и компатибилности у односу на модерне еквиваленте“ (OECD, 2005, p. 146).

„Наслеђени систем је застерели систем критичан за пословање који је подржавао пословне функције много година, али се више не сматра да је последња реч технологије и има своја ограничења у дизајну у употреби, Унутар организација, велика већина њих је замењена пре 2000. године ERP системима” (Khosrow-Pour, 2007, p. 411).

„Наслеђени системи су старије рачунарске технологије које организације временом користе. Они изводе пословне критичне функције и моће да се интегрише са новим системима“ (Yoon et al., 2005).

„Наслеђени системи су стари рачунарски системи, потенцијално тешки за одржавање и могу да се покрећу на старом хардверу који се можда тешко поправља. Често наслеђени системи нису добро схваћени, тако да проширивање система и интегрисање са новим системима може да буде тешко“ (Vestues & Knut, 2019, p. 2).

„Наслеђени систем је систем, апликација или технологија која је застарела, али наставља да се употребљава зато што адекватно спроводи потребне функционалности“ (Ayanso, 2015, p. 309).

„Наслеђени систем је застерели рачунарски систем, програмски језик или апликативни софтвер који се користи уместо доступних надограђених верзија“ (Birgün & Altan, 2019, p. 266).

„Наслеђени систем је информациони систем који је можда базиран на застарелим технологијама, али је критичан за свакодневне операције. Замена наслеђених апликација и система са системима заснованим на новим или различитим технологијама је један од највећих изазова за професионалце из области информационих система. Како предузећа побољшавају или мењају њихове технологије, они морају да осигурају компатибилност са старим системима и форматима податка који су и даље у употреби“ (Gartner, n.d.).

У пракси се често мешају појмови наслеђених система (енг. *legacy systems*) и наслеђеног софтвера (енг. *legacy software*), што није исправно јер се тиме занемарују остали важни аспекти наслеђених система. Наслеђен софтвер је само део наслеђеног система као ширег појма, на сличан начин као што је софтвер део информационог система. Наслеђени софтвер, где се у пракси може користити и термин наслеђен код (енг. *legacy code*), је софтвер који је тежак за измену, тежак за одржавање и скуп за реструктурирање. Са друге стране наслеђен систем је много више од наслеђеног софтвера јер укључује: људе, експертизу, хардвер, податке, пословне процесе и приступе у одржавању и развоју софтвера, при чему је нагласак на интеракцији ових елемената (Brooke & Ramage, 2001, p. 12). Дакле, наслеђен систем је шири појам од наслеђеног софтвера, што се често занемарује и води ка неоправданом редуковању његове сложености на само техничку представу система, док је у реалности веома комплексан феномен.

Наслеђени системи су стари информациони системи, често имплементирани у застарелом програмском језику и технологији, а и даље су у употреби, јер су корисни и подржавају критичне пословне операције (Sommerville, 2016, p. 261). ИТ менаџери са једне стране желе да их се реше, јер наслеђени системи им задају велике проблеме у ефективном и ефикасном управљању, али то не могу урадити једноставно и без ризика, док са друге стране знају да су поменути системи корисни јер подржавају најважније пословне операције и чувају вишедеценијска вредна знања о томе како се одређени посао обавља. Из поменутих разлога ИТ менаџери су стално у дилеми шта да раде са наслеђеним системима, поготово зато што трошкови њиховог одржавања са сваком новом изменом све више расту. Тежина и комплексност у управљању наслеђеним системима долази из чињенице да они не укључују само хардвер и софтвер, већ и наслеђене процесе и процедуре - „старе начине рађења ствари“ који су тешки за промену јер се ослањају на наслеђен софтвер (Sommerville, n.d.).

Они су настали у једном тренутку у времену, када су се пословне операције обављале на одређени начин, које су крајњи корисници у интеракцији са програмерима и

пројектантима система превели у захтеве, спецификацију, дизајн и имплементацију система, без обзира да ли организација има свој интерни развој или је софтвер добила од стране друге организације. Дакле, програмери и пројектанти система су преузели знање о послу од крајњих корисника и имплементирали их у систем. У међувремену над тим системом је наступио низ промена где се поменуто знање изменило и најчешће надоградило. Оно је некада изражено у пословним правилима које је могуће разумети уколико је документација постоји и ажурна је, што најчешће није случај. Знања о систему, како Ван Хауел (Willem-Jan, 2009, р. 21) указује нису само експлицитна знања о пословним процесима, пословим политикама и подацима који су искодирани на заједничком језику (корисника и програмера), већ и тацитна знања која омогућавају несметано обављање дневних пословних операција. Неки од поменутих система, као што је нпр. систем контроле летења, су критични за обављање посла и уколико откажу чак и у кратком временском интервалу то може да има катастрофалне последице.

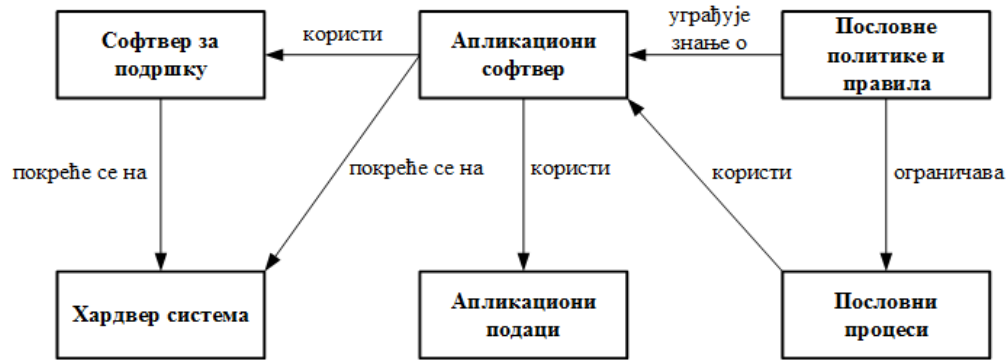
Имајући у виду све поменуте дефиниције, разматрања и мишљења у вези наслеђених система може се констатовати да су наслеђени системи:

- системи развијени у прошлости
- састоје се од наслеђеног хардвера, наслеђеног софтвера, наслеђених пословних процедура и правила
- подржавају кључне пословне процесе
- тешко се разумеју
- тешко користе најмодерније технологије
- опиру се променама
- ограничавају пословање
- њихово одржавање је скупо
- ризични су за замену
- производе незадовољство на свим нивоима управљања у предузећу
- могу да имају неадекватну софтверску архитектуру
- могу да имају малу могућност интеграције са другим системима
- један од највећих изазова за професионалце у области информационих система

Дефиниција наслеђених система која ће бити коришћена у овом раду, са мишљу да се спречи изоловно посматрање феномена и да се сви важни чиниоци њега као предмета посматрања обухвате је дата у наредним редовима.

Наслеђени системи су системи развијени у прошлости који се тешко разумеју, тешко користе најмодерније технологије, опирају се променама, ограничавају пословање и скупи су за одржавање, организација би их заменила али је то веома ризично, поготово зато што подржавају кључне пословне процесе, при чему производе незадовољство на свим нивоима управљања у предузећу. Уобичајено се састоје од од наслеђеног хардвера, наслеђеног софтвера и наслеђених пословних процедура и правила, а при томе могу да имају неадекватну софтверску архитектуру и малу могућност интеграције са другим системима. Представљају један од највећих изазова за професионалце у области информационих система.

Неки аутори наглашавају да се наслеђени системи не могу посматрати једнострано, нпр. само са становишта софтвера, већ да они чине шири социотехнички систем који укључује хардвер, софтвер, библиотеке и други софтвер потребан као подршка, пословне процесе, као и везе између њих (Sommerville, 2016, р. 263). Односи елемената наслеђеног система су приказани на слици 20.



Слика 20. елементи наслеђеног система, преузето (Sommerville, 2016)

Може се догодити да хардвер система (енг. *system hardware*) који користи наслеђени систем више не постоји, јер произвођач хардвера је можда престао да постоји или је повукао из производње хардвер који је купљен, а нова верзија је некомпатибилна на старом.

Софтвер потребан као подршка (енг. *support software*) је такође део наслеђеног система, јер он служи да омогући апликативном софтверу да ради на оперативном систему и да користи додатне помоћне програме које прави произвођач хардвера.

Компајлер је пример таквог софтвера који преводи инструкције програмског језика на језик који одређена хардверска платформа може да разуме, а он може да буде написан за специфичну хардверску платформу. Уколико није подржан од стране произвођача или је застарео то може правити додатне проблеме у наслеђеном систему.

Апликативни софтвер (енг. *application software*) је део наслеђеног система, јер се путем његовог извршавања на хардверу и уз употребу хардвера, помоћних програма за подршку и података спроводе пословни процеси којима се креира нова вредност за предузеће, чиме се реализују његови пословни циљеви. Уобичајено је да се апликативни софтвер састоји из скупа програма. Неке од напред наведених програма су развијали програмери у прошлости, а у међувремену су напустили организацију.

Апликативни софтвер користи податке (енг. *application data*) који се најчешће памте у систему за управљање базом података. Поред тога постоје и софтверски системи који користе систем фајлова (енг. *file system*) да упамте апликационе податке. Неки програми истовремено користе већи број извора и могу наићи на проблем уколико постоји редуданса, тј. понављање података у различитим системима.

Извршавањем пословних процеса (енг. *business processes*) организација остварује своју мисију, визију и циљеве. Апликациони софтвер подражава њихово извршавање. Наслеђени систем извршава оне пословне процесе који су описани захтевима приликом његовог развоја у прошлости и може се догодити да уколико постоји потреба за изменом истих то се не може једноставно реализовати.

Пословна политике и правила (енг. *business policies and rules*) су такође део наслеђеног система, јер описују како се пословање спроводи. Интегрални део наслеђа чине и политике и правила која су постављена и која се извршавају, тј. евалуирају кроз систем.

Као што је приказано између елемената наслеђеног система постоји велики број међусобних веза. Када су те везе архитектурално решене на начин да постоји тесна повезаност (енг. *tight coupling*) између њих или унутар њих, неминовност је да ће промене учињене у једном делу система да се пропагирају ка другим деловима система. У пракси због овакве природе наслеђених система велики део програмера заузима став „ако ради, не дирај“, што може имати дугорочне последице на смањење животног века система, јер је процес старења софтвера неминовност и доводи до опадања квалитета система, па ће се он можда повући из употребе пре времена.

Велика финансијска средства се улажу у одржавање наслеђених система, као и у спровођење пројеката модернизације путем којих би они требало да наставе свој животни век најчешће у неком другом облику. Како су промене које се спроводе над напред наведеним системима ризичне и често неконтролисане, пракса је показала да постоји велики број пропалих пројеката модернизације чак и након уложених значајних средстава (Ulrich & Newcomb, 2010, p. 5):

- Компанија Signa, утрошила је 1 милијарду долара за ИС у осигурању, где се здравствено осигурање признавало у неодговарајућим случајевима.
- Неуспела имплементација ЕРП решења у Hershey Foods, довела је до проблема у дистрибуцији и губитка дела тржишта од 27%.
- Неуспела имплементација ЕРП решења фирме FoxMeуer Drug, довела је фирму до пропасти.
- Нови систем пореске администрације, IRS систем, омогућио је неадекватно рефундирање у износу од 200 милиона долара и стављен је ван употребе.
- Портовање софтвера компаније Oregon DMV трајало је 8 година, а уз негодовање јавности пројекат је прекинут.
- Систем социјалне помоћи државе Флорида приказивао је велики број грешака након имплементације, иако је преплаћен у износу од 260 милиона долара.
- Корпорације ARM, Hilton, Budget Rent A Car и Merriot, су током 4 године инвестирале у пројекат, у износу од 125 милиона долара, који је на крају пропао.
- Након повлачења претходног система из употребе и инвестирања од 456 милиона фунти у нови информациони систем агенције Child Support Agency у Великој Британији, нови систем је мање ефикасан него претходни.
- IRS пројекат је имао пројектоване трошкове од 8 милијарди долара када се заврши, а премашио је буџет 40%, при чему је било потребна још 1 милијарда долара да би се пројекат завршио у договореном року.
- SAIC корпорација, испоручила је нефункционалан систем, са много програмских грешака, који је довео до прекида пројекта за који је планирано 170 милиона долара, где постоји велики део неупотребљивог кода у износ од 105 милиона долара.

- Обустављена је имплементација SAP решења у износу од 70 милиона долара у америчкој федералној агенцији.
- Међународна телекомуникациона компанија је прекинула пројекат замене главног система уз губитак од 80 милиона долара.

Наслеђени системи функционишу у окружењу које се стално мења и где је потребно остати конкурентан на тржишту, па из поменутих разлога организације значајно инвестирају у њихово одржавање. Како процес одржавања тече, долази до нарушавања структуре и функционалности система, када у једном тренутку постане потпуно неисплативо одржавати их и тада организација одлучује да их повуче из употребе. Поред тога што су трошкови замене ових система велики, они нису коначни јер добар део трошкова настаје услед поновног откривања акумулираног знања о пословним правилима и процесима (Jesús Bisbal et al., 1999, p. 104). У пракси то значи да када долази до замене система, менаџери често не схватају да добар део знања одлази са тим системима, а поготово ако су наслеђени јер најчешће немају документацију.

4.4.2. Перспективе наслеђених система

У претходној секцији презентоване су многобројне дефиниције наслеђених система, при чему не постоји јединствено мишљење или став о томе шта је наслеђен систем. То води ка конфузији, и даље ка томе да у теорији и пракси постоји велика шанса да неки аспект софтверског наслеђа буде занемарен, што може да доведе до ситуације да све активности које се спроводе у циљу побољшања наслеђеног система не буду ефективне и ефикасне, јер одређени његови аспекти нису узети у обзир. Дакле, постоји озбиљан ризик да наслеђени системи, као сложени социо-технички системи, буду изловано посматрани и да значајно уложена средства у враћање њихове еволутивне способности не дају очекиване ефекте. Постоји неколико разлога зашто овакви сложени системи морају да еволуирају (Pressman & Maxim, 2015, p. 8):

- софтвер мора да се адаптира да би задовољио потребе нове технологије или новог рачунарског окружења
- софтвер мора да се унапреди да би имплементирао нове пословне захтеве
- софтвер мора да се прошири да би био интероперабилан са неким другим савременим системима или базама података
- софтвер мора да се архитектурално реконструише да би се одржао у рачунарском окружењу које еволуира

Да би се спречило изоловано посматрање наслеђених система кроз изучавање само неких њихових аспеката, многи аутори наводе различите перспективе наслеђених система да би остварили један свеобухватан поглед на посматрани феномен. У литератури се веома често спомињу две димензије, два гледишта или два аспекта наслеђених система, технички и пословни, при чему неки аутори наводе да наслеђени систем треба да „помири“ та два аспекта (Brooke & Ramage, 2001). Бојић и Велашевић указују на три перспективе: перспективу произвођача софтвера, перспективу технолошког развоја и пословну перспективу (Bojić & Velašević, 1999). Група аутора је истраживала перцепцију наслеђених система од стране професионалаца у софтверској индустрији и упоређивала њихову перспективу у односу на перспективу академске заједнице и донела одређене закључке по том питању употребом посебног вида методологије за прикупљање и анализу података, ГТ (енг. *grounded theory*) методологија (Khadka et al., 2014).

За потребе овог рада, а са становишта аутора да је довољно за свеобухватно посматрање појаве, користиће се класификација перспектива наслеђених система према Алдерсону и Шах, где постоје четири перспективе или тачке гледишта: развојна (енг. *developmental*), оперативна (енг. *operational*), организациона (енг. *organizational*) и стратешка (енг. *strategic*) (Alderson & Shah, 1999). У односу на раније презентоване приступе, ово разматрање даје увид у наслеђени систем са неколико различитих позиција и организационих нивоа у предузећу, па се ствара могућност да се процес еволуције система потпуније сагледа.

4.4.2.1. Развојна перспектива

Развојна перспектива је поглед на наслеђен систем из угла људи који развијају и касније одржавају тај систем. То су аналитичари система, пројектанти, архитекте, програмери, тестери, инжењери који се брину за квалитет и тд. Свака од ових улога на различити начин гледа на систем и ради са другим, најчешће различитим концептима система у односу на неку другу улогу, али оно што је свима њима заједничко јесте перцепција старења система. Услед тога, свака од улога има схватање да старење система, тј. пут ка наслеђеном систему започиње када се фаза на којој они раде заврши. Сваки концепт система, захтев, дизајн или код, са аспекта одржавања је подложен промени. Те промене могу да проузрокују тешкоће у процесу одржавања.

Проблем је утолико већи уколико не постоји документација или она није довољно ажурна, што је у пракси чест случај. На тај начин смањује се разумевање система, па самим тим и одржавање тог система, јер је веома тешко нешто одржавати уколико се не

разуме и уколико не може у потпуности да се сагледају промене које је потребно извршити. Тиме се отвара простор да измена у једном делу система неконтролисано води ка променама у другим деловима, познатијим као ефекат таласа. Такође процес одржавања се усложњава уколико особље које је радило на развоју система више не ради. Старији запослени, који најчешће имају много више искуства у раду са старијим технологијама, можда су променили предузеће или су отишли у пензију, тако да се још више смањује знање о систему.

Нови запослени, који долазе са факултета ретко када поседују знања и вештине које су потребне за одржавање старијих система. Поред тога, проблематика наслеђених система је веома мало заступљена у плану и програму студија, јер се много више фаворизују знања у форми испита и курсева која усмеравају студенте ка развоју новог система, него на одржавање. То је контрадикторно у односу на праксу софтверског инжењерства, где се у пракси око 80% уложених средства троши у одржавање постојећих система. Други проблем са новим запосленим јесте недостатак личне мотивације, јер немају могућности да примене она знања за која су се школовали, што од њих захтева „враћање у прошлост“ и усвајање старијих знања. На тај начин они дугорочно не виде рад на наслеђеним системима као вредно искуство, јер неће развијати модерна знања и вештине из области софтверског инжењерства. То их чини додатно немотивисаним, јер неће поседовати вештине и знања која су у много већем обиму тражена на тржишту.

Као последица дугорочног одржавања, долази до нарушавања архитектуре система. Чест је случај да услед кратких рокова за које је потребно спровести одређене измене, недовољно пажње се посвећује архитектури система, већ се све покушава решити програмским кодом. Иако је за програмере и руководиоце ИТ пројеката важно да што пре заврше одређени део посла, таквим начином рада губи се из вида да недовољно времена проведено на осмишљавању архитектуре и дизајна, као и активности усмерених ка њиховом очувању може да проузрокује последице у будућности. То се често догађа, а манифестује се многобројним ефектима таласа и несагледивим променама у деловима система у којима промене у одржавању нису иницијално чињене. Писање новог кода никако не може да буде замена за добро осмишљен дизајн, који поштује одређене принципе, као што је лабава повезаност (енг. *loose coupling*).

Поред чињенице да је архитектура наслеђеног система сложена и тешко разумљива, у зависности од одабраног имплементационог окружења и технологија које су коришћене, као и архитектуралних одлука из прошлости може се догодити да је архитектура монолитна, тј. да је цео систем пројектован у једном слоју (енг. *layer*).

Уколико је то случај одржавање може бити веома компликовано, јер ће промене које је потребно учинити изискивати много више времена него што би то био случај у системима који имају софтверску архитектуру организовану у више слојева (енг. *multilayered architecture*), где се измене много лакше спроводе. Трослојна архитектура има три слоја: презентациони слој (енг. *presentation layer*), слој пословне логике (енг. *business logic layer*) и слој који приступа подацима (енг. *data access layer*). Сваки слој има своје задужење, нпр. кроз презентациони слој обавља се интеракција са корисником где корисник позива неке функције система, слој пословне логике описује пословну логику и пословна правила, док слој који приступа подацима ради са базама података и брине се да се све операције везане за приказ и ажурирање базе података се спроводе. Када је потребно спровести неку измену, то је могуће урадити релативно једноставно, на неколико различитих слојева, уколико има потребе за тим. А када је архитектура монолитна, где се све налази на једном месту, то ствара проблеме у одржавању, јер програмери не могу једноставно изменити програмски код, јер није довољно разумљив, као што је случај код софтвера који су организовани у више јасно раздвојених слојева.

Један од проблема који је такође заступљен је редуванса, односно вишеструко памћење истих података у систему, чиме се усложњава систем и производе аномалије у ажурирању података.

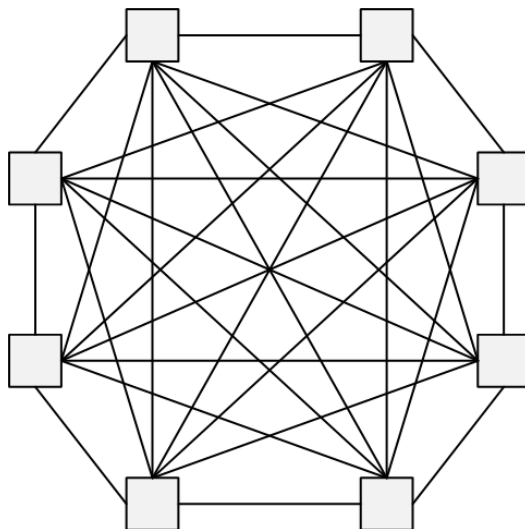
4.4.2.2. *Оперативна перспектива*

Оперативна перспектива наглашава да је за систем потребно да обезбеди ефикасан сервис и то пре свега онима који га користе. Ефикасан сервис је повезан и са временом одговора система на разне функционалности које је он потребно да обавља, тако да крајњи корисници могу у интеракцији са њим да реализују жељене функционалности.

Проблеми се могу појавити када наслеђен систем користи наслеђен хардвер или софтвер за који више не постоји подршка од оргиналних произвођача. Произвођачи временом услед технолошког развоја напуштају рад на хардверу и софтверу који су развијали, јер је то захтевно а такође можда не могу са понуђеним досадашњим решењима да наставе да спроводе своју мисију, визију и циљеве, па то може бити велики проблем за предузеће које има систем реализован у таквом једном окружењу. Уколико не постоји одређени део опреме, може се догодити да цео систем престане са радом. Поред тога, додатни проблеми могу настати уколико се не спроводе редовна ажурирања система, услед тога што је произвођач напустио развој, јер се могу јавити разне претње по сигурност система. Такође проблем који може да постоји то је уколико не постоји

компатибилности између верзија система, јер се на тај начин онемогућава систему да корисни нове и савременије програмске карактеристике и могућности. Када се изврши прелаз са једне верзије на следећу, може се догодити да програмске карактеристике употребљене у ранијој верзији можда не могу да раде у новој, јер су превазиђене, тј. застареле (енг. *deprecated*).

Наслеђен систем може да има и проблеме везане за интеграцију са другим системима, што утиче на то да се одређена пословна сценарија не могу извршавати у потпуности. Некада је то последица застарелог хардвера и софтвера, који немају могућности за употребу одређене врсте технологије, а некада је последица и неадекватне архитектуре и избора да се интеграција спроводи из тачке у тачку (енг. *point-to-point*). Одржавати такав један систем је веома компликовано, јер је потребно одржавати велики број веза и интерфејса између њих.



Слика 21. интеграција од тачке до тачке

Уколико се систем састоји из 8 апликација, при чему је свака са сваком повезана, онда број потребних интерфејса и веза за одржавање је 28. Када се спроведе нека промена у једној апликацији, то је потребно спровести на још 7 места (за све остале које су повезане са њом) што значајно усложњава одржавање, као што се види на слици 21.

Један од веома честих разлога за настанак овакве архитектуре јесте брзина којом се тражи развој новог софтверског решења. Уместо да се софтверско решење и дизајн система адекватно осмисли и повеже са осталим решењима на начин да има адекватан дизајн, спроводи се ad-hoc интеграција, којом се занемарује значај дизајна. Таква решења

називају се силоси⁴ (енг. *silos*) или острва аутоматизације (енг. *islands of automation*). Она могу бити дефинисана на нивоу једног система (енг. *stovepipe system*), где подсистеми спроводе описани вид интеграције, али могу и бити на нивоу целог предузећа (енг. *stovepipe enterprise*) где сваки од система је повезан са другим системом на описани начин. Оба начина су лоши примери софтверске праксе, тј. анти-шаблони (енг. *anti-pattern*) и проузрокују велики број проблема, од којих су недостатак истоветности (енг. *commonality*) која кочи интероперабилност система, спречава поновну употребу (енг. *reuse*) и увећава трошкове, а при томе се нарушава и адаптивност система (Sourcemaking.com, n.d.-b, n.d.-a). Проблем силоса јесте што значајно усложњава рад тима који ради на одржавању система, константно одржавајући интерфејсе и водећи рачуна о томе да ли ће се промена унутар једног интерфејса пропагирати на све остале.

4.4.2.3. Организациона перспектива

Организациона перспектива се односи на меру у којој наслеђени систем подржава пословање организације, тј. колико се спровођењем пословних процеса реализује пословна стратегија предузећа. Ова је посматрање наслеђених система из перспективе менаџмента. Овај однос је вишеструк.

Са једне стране наслеђени системи се виде као кочничари промена, јер услед своје инертности и нефлексибилности не могу брзо да одговоре изменама у пословним процесима предузећа. На одређени начин они ограничавају пословање, поготово у ситуацији када је потребно спровести реинжењеринг пословних процеса. Услед тога, менаџери су фрустрирани и виде их као једну од препрека у остваривању нове пословне стратегије. Са друге стране они имају у виду да наслеђени системи подржавају главне (енг. *core*) пословне процесе и функције, па су незаменљиви. Иако су стари, скупи за одржавање, инертни и често неразумљиви они су критични за пословање.

4.4.2.4. Стратешка перспектива

Стратешки ниво одређује основни правац кретања предузећа, па је за усвајање нове или измену постојеће стратегије, успостављање нових пословних модела или других активности на стратешком нивоу, важно да те замисли буду прихваћене и спроведене на оперативном нивоу. Како су наслеђени информациони системи подршка у спровођењу измењених пословних активности при чему постоји извесни јаз између онога што наслеђени систем нуди и онога што активности стратешког нивоа захтевају, спроводи се

⁴ такође се користи и термин цилиндри (енг. *stovepipes*)

евалуација и оцена изгубљених могућности за све оне пословна сценарија проистекла из стартегије која наслеђен систем не може да их подржи.

Трошкови одржавања система, лиценцирања, могућност и трошкови ангажовања нових људи који ће радити на систему и задржавања постојећих, ограничена подршка произвођача хардвера и софтвера, са једне стране се упоређују са губитком клијената и тржишта са друге и доносе се дугорочне одлуке шта радити.

4.4.3. Трошкови одржавања постојећих наслеђених система

Одржавање наслеђеног система у употреби и одлагање његове замене, на одређени начин заобилази ризик који би настао, али доводи до раста трошкова. Разлога има неколико.

Пре свега, постојећи изворни код система је посебно тешко прочитати и схватити мануелно, јер не постоји конзистентан стил програмирања, јер је сваки тим имплементирао различити део наслеђеног система. Поред тога, дешава се да није коришћен пример добре праксе у писању коментара који би објаснио одређене програмске функционалности и помогао да се одређене секције кода једноставније разумеју, тако што би се друге секције које су коментарисане се апстраховале у тренутном посматрању. Програмер је осуђен на мануелни, мукотрпни рад који му не гарантује да ће разумети систем, поготово уколико је потребно да спроведе промене над истим у кратком временском року. „Шпагети код“ посебно прави проблеме, јер превелики број употребе go to наредби безусловног скока производе велику конфузију коју ретко ко може да разуме.

Следећи разлог је везан за величину програмерске заједнице и доступну радну снагу за програмски језик у којем је написан наслеђени систем. Млади програмери који су завршили факултет или високу школу најчешће не могу да се ангажују за ове типове пројеката, јер не поседују знања за поменуто застарела програмска окружења, па их је тешко обучити за иста, поготово када веома мали број људи у предузећу познаје поменуто окружење. Трошкови ангажовања искусних програмера са тржишта рада може да буде веома финансијски захтевно, па је потребно направити одговарајућу анализу исплативости. Нпр. иако је COBOL програмски језик који се користио пре неколико деценија, његова интензива употреба у том периоду, поготово у банкарском сектору у USA допринела је да и данас постоји одређени број позиција за програмере који поседују ове вештине. Уобичајено је да су они старосне доби око 50 година и да то можда није у

складу са политикама ангажовања кадрова у поменутиим предузећима, па менаџери људских ресурса гледају са негодовањем. Такође неки од њих су завршили свој радни век, па фирме их ангажују да поново ступе у радни однос или да наплаћују своје услуге по дану ангажовања, где накнада може бити и у висини од 1000 долара. Истраживање Ројтерса указује на масовност наслеђених система насписаних у COBOL програмском језику (Cassel, n.d.):

- 43% банкарских система је изграђено на COBOL-у
- 80% персоналних трансакција користи COBOL
- 95% апарата за картице се ослањају на COBOL
- 220 милијарди линија кода у COBOL-у се користи данас

Дакле, поступак повлачења пословних информационих система из употребе, који су ослонац главног бизниса у банкарском сектору је веома спор, па се константно трага за другачијим стратегијама, јер повлачењем се уводи револутивна промена која може да угрози систем, па и пословње у целости. У том смислу еволутивне промене делују подобније у наведеном контексту.

4.4.4. Ризици и трошкови замене наслеђених система

Постоје многи ризици и несигурности за предузеће које се одлучи да замени наслеђени систем. Разлога има неколико.

Развој новог софтвера може да потраје дуже времена у односу на планирано и да произведе велике трошкове, јер је и сам ризичан процес и на том путу се могу појавити многи неочекивани проблеми. Проблем неразумевања наслеђеног система може продужити тај пројекат, поготово уколико га је потребно опонашати делимично или у целости. Корисници који буду користили нови систем је потребно да се навикну на нове интерфејсе, који можда неће бити употребљиви толико као у наслеђеном систему па процес рада можда неће бити довољно ефикасан.

Наслеђени систем подржава пословање предузећа и веома често има пословну логику и правила која су повезана са пословним процесима које организација обавља. Може се догодити да са заменом наслеђеног система процес рада се промени, јер ће можда постојати другачији ток њиховог извршавања. Уколико промена није сагледана у потпуности, лако се може догодити да се један део потребног посла превиди, што може да има последица на пословне циљеве предузећа, јер тај пословни процес треба да произведе вредност за крајњег потрошача који ће бити ускраћен па можда неће ни користити производ предузећа.

Са заменом наслеђеног система не могу се сагледати промене које ће да наступе, јер ретко када постоји документација која описује систем у потпуности и са свих аспеката. Из поменутог разлога ће бити тешко направити нови систем, јер не постоји спецификација по којој је он потребно да ради, већ ће његови пословни аналитичари и пројектанти анализом долазити до тога шта систем треба да ради. О самом наслеђеном систему може се знати веома мало, поготово у ситуацији када програмери не раде у тимовима, већ је због њиховог малог броја организација посла направљена тако да свако ради све, али искључиво на својим пројектима, тј. деловима система. Можда само једна особа има потпуно знање о систему, али пракса често показује да ни то није случај.

Многи делови наслеђеног система су искодирани на заједничком језику програмера и његових корисника који су дали захтев за програмом. Кроз тај програмски захтев, о коме најчешће не постоји писани траг или документација могу се имплементирати важна пословна правила која је потребно да буду укључена приликом извршавања одређених пословних функционалности. Путем поменутих правила, ограничава се у којим случајевима ће се одређени пословни сценарио спроводити. Нпр. претпоставимо да је потребно повући из употребе наслеђени информациони систем предузећа у којем је имплементирано пословно правило да када потрошач изврши плаћање рачуна до датума валуте шест месеци узастопно одобрава му се попуст од 5%, уколико уплаћује редовно у року девет месеци тада има попуст од 6%, а уколико годину дана плаћа у валути тада има попуст од 7%. Уколико нов систем који је потребно да замени наслеђени не имплементира ово правило, може се догодити да потрошачи не добију попусте који су им загарантовани пословним политиком и уобличене су у одређеним законским актима, па могу покренути судски поступак, тужити предузеће и добити спор. Такође, финансијска инспекција може приметити поменуту мањкавост у новом систему, па написати одређени приговор или изрећи казну.

ИТ менаџери имају подељен однос у вези наслеђених информационих система, са једне стране препознају њихову пословну вредност, а са друге увиђају високе трошкове одржавања и ризике њихове замене, али размишљају о одређеној врсти модернизације. Поменута подељеност настаје и из специфичности које сваки наслеђени систем има, јер садржи целокупну историју одржавања од свог настанка па до тренутка посматрања, па је тешко говорити о генералном начину решавања њихових проблема.

То показује и истраживање спроведено 2002. међу ИТ менаџерима, где је сваки од њих одговорио на низ питања (Ulrich, 2005):

1. Колико су корисни постојећи системи предузећа у подршци постизању стратешких циљева ?
 - a. веома корисни / нешто корисни (63,7%)
 - b. нешто ограничавајући / веома ограничавајући (36,3%)
2. Који су главни разлози да задржавање постојећих система у употреби ?
 - a. и даље су способни да подрже пословне процесе (54,4%)
 - b. и даље су поуздани (49,7%)
 - c. постоје запослени који могу да их одржавају (44,3%)
 - d. и даље су финансијски исплативије од алтернатива (41,6%)
 - e. немамо буџет да би смо направили било какву промену (36,9%)
 - f. и даље могу да подрже стратешке циљеве (36,9%)
3. Уколико тренутно спроводите миграцију или уколико планирате миграцију било којег од ваших постојећих система наредне године, шта су ваши главни разлози ?
 - a. остваривање нових стратешких циљева (65,2%)
 - b. постојећи системи не могу да подрже пословне процесе (59,9%)
 - c. постојећи системи не могу да подрже тренутне стратешке циљеве (56,1%)
 - d. нови системи би били више финансијски исплативи (48,5%)
 - e. постојећи системи имају слабу интероперабилност (41,7%)

На основу резултата поменутог истраживања може се закључити:

- наслеђени системи су и даље важни
- наслеђени системи су и даље релевантни и поуздани
- постоје планови за миграцију или модернизацију система

4.4.5. Актуелност проблема

Наслеђени системи проузрокују велики број проблема за предузећа и готово да не постоји област у којој то није случај. Они су често присутни у јавном сектору, чија су предузећа велики системи и тешко се мењају, па ни информациони системи нису изузетак у томе. Софтверско наслеђе јавног сектора има свој корен у томе што су поменута предузећа најчешће прва уводила рачунарски засноване информационе системе у свој рад, за које су била потребна значајна финансијска средства у прошлости и сем института, војних установа и других великих система ретко ко је могао да их приушти. Велики број ових предузећа је кренуо да користи информационе системе још у ери

мејнфрејма и до данашњих дана многи су опстали у изворном или делимично измењеном облику. За одржавање поменутих система у употреби улажу се значајна финансијска средства.

На примеру САД, на основу јавно доступног истраживања агенције GAO (Government Accountability Office), влада САД је планирала да потроши преко 90 милијарди долара у 2019. години на ИТ, где се око 80% троши за одржавање наслеђених система (United States Government Accountability Office, 2019). Извештај приказује 10 критичних наслеђених система федералних агенција, са исказаном потребом за модернизацијом при чему се наводе и године старости система, хардвера, као и оцене перципиране критичности система и сигурносних ризика према агенцији. Приказ је дат у табели.

Табела 2. Десет критичних наслеђених система федералних агенција, преузето (United States Government Accountability Office, 2019).

| Агенција | Назив система | Старост система (у годинама) | Најстарији хардвер (у годинама) | Критичност система | Сигурносни ризик |
|------------------------------------|---------------|------------------------------|---------------------------------|--------------------|------------------|
| Министарство одбране | Систем 1 | 14 | 3 | Умерено висок | Умерен |
| Министарство образовања | Систем 2 | 46 | 3 | Висок | Висок |
| Министарство здравља | Систем 3 | 50 | непознато | Висок | Висок |
| Министарство унутрашње безбедности | Систем 4 | 8-11 | 11 | Висок | Висок |
| Министарство унутрашњих послова | Систем 5 | 18 | 18 | Висок | Умерено висок |
| Министарство финансија | Систем 6 | 52 | 4 | Висок | Умерено низак |
| Министарство саобраћаја | Систем 7 | 35 | 7 | Висок | Умерено висок |
| Канцеларија за управљање кадровима | Систем 8 | 34 | 24 | Висок | Умерено низак |
| Администрација за мала предузећа | Систем 9 | 17 | 10 | Висок | Умерено висок |
| Администрација за социјалну | Систем 10 | 45 | 4 | Висок | Умерен |

| | | | | | |
|------------|--|--|--|--|--|
| безбедност | | | | | |
|------------|--|--|--|--|--|

Ситуација у јавном сектору Србије није много боља (Vesić, 2020). Већина јавних предузећа има неколико деценија старе системе, најчешће изграђене од власничког софтвера, што за последицу има дугорочну везаност за одређеног произвођача који може диктирати услове пословања као и могућности у погледу проширивања и технолошког оспособљавања система за рад са савременим информационим технологијама.

У граду Београду постоје сва велика јавна комунална предузећа „Инфостан технологије“ и „Београдски водовод и канализација“ који имају наслеђене системе старе неколико деценија, који подржавају кључне функционалности предузећа чији би откази довели до великих проблема у пословању предузећа. Не само што потенцијално користе наслеђен хардвер и софтвер, велики део пословних правила и пословне логике се налази у овим системима, тако да се је потребно спровести адекватан процес модернизације који би сачувао вредна знања о обављању посла и годинама инвестирана средства у развој поменутих система, при чему је процес њихове модернизације потребно сагледати из свих перспектива. На тај начин ће се смањити изоловано посматрање феномена софтверског наслеђа и минимизираће се могућност да будуће инвестиције пропадне, што захтева заузимање адекватног правца модернизације и стратегије која то треба и да обезбеди.

4.5. Начини решавања проблема наслеђених система

Обзиром да сваки наслеђени информациони систем је тренутна представа система као која је настала од прве његове верзије, људи, знања и праксе које су важиле тада, уз сва измене које је претрпео, активности модернизације и активности одржавања, тешко да постоје два наслеђена система која су идентична. Сваки наслеђени систем је посебан и захтева такав приступ уколико је битно да се модернизује, спроведе реинжењеринг или повуче из употребе и зато је веома тешко дати јединствен методолошки поступак који би могао да обједини и покрије све случајеве који се јављају у пракси. Иако је овај задатак веома тежак, посматрајући наслеђене системе, разни аутори су предложили различите начине и стратегије у вези истих.

Према Сомервилу постоје 4 основне стратегије (Sommerville, 2016, p. 266):

- избацили систем комплетно из употребе - применити када постоји јаз између пословних процеса у пракси и пословних процеса које нуди наслеђен систем

- оставити систем непромењен и наставити са регуларним одржавањем - применити када је систем стабилан и постоји мали број измена
- спровести реинжењеринг система ради увећања његове одрживости - применити када квалитет система значајно опадне, а постоје промене које је потребно спровести у циљу омогућавања рада са другим системима
- заменити све делове система новим системом - применити када систем не може да ради, а замена је исплатива

Аутор Јовановић предлаже следеће начине за разрешавање софтверског наслеђа (В. Јовановић, 2007, р. 311):

- развој новог система од почетка
- преписати - прерадити делове система очувајући његово понашање
- пребацити систем на другу платформу
- стратегија миграције са коришћењем делова постојећег система

У теорији и пракси постоји већи број приступа у решавању проблема наслеђених система. Неки приступи ће у наставку бити поменути, а за потребе овог рада фокусираћемо се на три приступа за које сматрамо да су релевантни:

- софтверски реинжењеринг (енг. *software reengineering*)
- облагање (енг. *wrapping*)
- миграција (енг. *migration*)

4.5.1. Софтверски реинжењеринг

4.5.1.1. Софтверски инжењеринг

Појам софтверско инжењерство (енг. *software engineering*) или софтверски инжењеринг је предложен на конференцији коју је организовао НАТО 1968. и 1969. године, као одговор на „софтверску кризу“ (енг. *software crisis*). Софтверска криза је термин који означава потешкоће у развоју великих и сложених софтверских система, где је велики број софтверских пројеката пропао услед тога што није постојао одговарајући методолошки начин за израду којим би се на одређени начин уредио процес израде софтверских решења.

Потреба за одговарајућим методолошким приступом је настала из више различитих разлога од којих се издвајају да пројекти нису испоручивани у договореним временским роковима и често су превазилазили планом опредељена средства чиме

организације задужене за развој нису могле да остваре своје мисије, визије и циљеве. У пракси тадашњег времена документација је често била неажурна услед велике количине промена над самим софтвером и није могла да се користи у циљу његовог лакшег одржавања. Поред тога, није редак случај био да захтеви нису у потпуности схваћени од стране програмера, који су се више бавили интерном организацијом софтверских пројеката, покушавајући да избегну неминовност тадашње праксе - нарушен дизајн, него што су били усмерени на кориснике, па самим тим захтеви нису у потпуности реализовани. Из истог разлога, квалитет софтвера је био на ниском нивоу иако је значајан број програмера радио на одржавању једног таквог решења и што је оно било старије сама структура му је била нарушенија, па је све већи број људи одржавао такво решење. У пракси се најчешће примењивао приступ да се нешто испрограмира, тј. напише се одговарајући програмски код, затим се то евалуира од стране корисника и након тога се врше преправке над тим програмским кодом и то се понавља све док се у коначници не задовоље захтеви. Поменути најстарији приступ у изради софтвера је познат под именом кодирање и преправљање (енг. *code and fix*). Он је временом довео до све тежег одржавања софтвера, где једна мала преправка захтева веома велико ангажовање, јер она може да произведе значајан број несагледивих промена у другим деловима система, тј. догађа се ефекат таласа (енг. *ripple effect*).

Према стандарду ISO/IEC 2382-1:1993 софтверско инжењерство се дефинише као систематична примена научног и технолошког знања, метода и искуства у дизајну, имплементацији, тестирању и документацији софтвера да би се оптимизовала његова производња, подршка и квалитет.

Слична дефиниција је дата у Водичу кроз корпус знања софтверског инжењерства где је софтверско инжењерство примена систематичног, дисциплинованог, мерљивог приступа у развоју, раду и одржавању софтвера, тј. примена инжењерства на софтверу (IEEE, 2014). Дефиниције софтверског инжењерства су дате из разлога да би могли у потпуности да сагледамо процесе који су заступљени у софтверском реинжењерингу, као и које су њихове додирне тачке са софтверским инжењерингом.

4.5.1.2. Сврха и циљеви софтверског реинжењеринга

Дугорочни процес одржавања софтвера доводи до промене у његовом коду, дизајну и све већем издвајању финансијских средстава за његово спровођење, што ставља велики притисак на пројектне менаџере и софтверски тим који је задужен за реализацију. Када услед великог броја имплементираних промена иницијално осмишљен софтверски пројекат више не може да се разуме да би се спровеле нове промене, затим не може да се

исконтролише и утврди да ли ће та нова промена нарушити његово функционисање и не може да се уреди на начин да се раније осмишљене функционалност на једноставан начин изнова употребљавају тада је време за његов реинжењеринг (енг. *reengineering*).

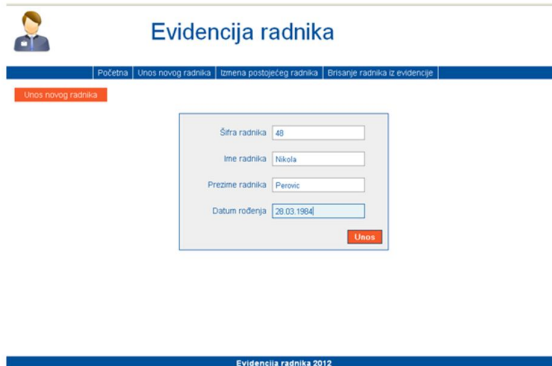
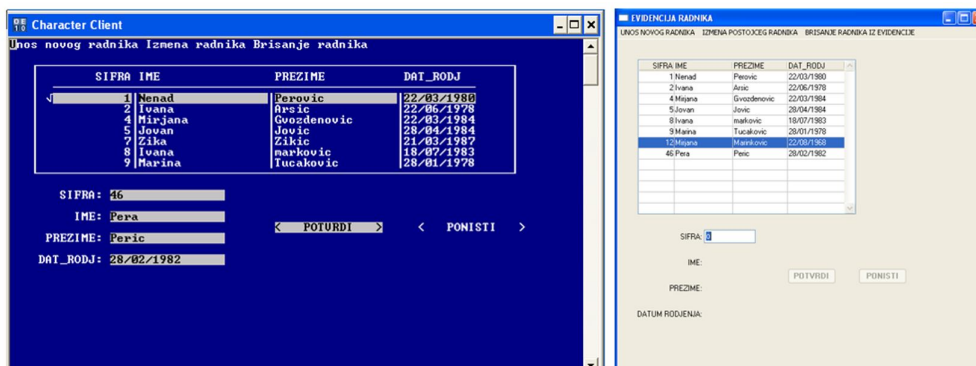
Наслеђени софтверски систем у пракси најчешће нема пројектну документацију или ако је има она је застарела, па је тешко схватити захтеве који су крајњи корисници испоставили у почетној фази животног циклуса развоја софтвера, као и у следећој фази која се односи на дизајн. Поред тога, може се догодити да иницијални захтеви и дизајн постоје, али је у међувремену дошло до многих промена услед одржавања, при чему повратне информације из тог процеса нису инкорпориране у документацију, па је немогуће из документације утврдити које функционалности постоје у систему.

Потреба да се реше проблеми наслеђених софтверских система је довела до појаве реинжењеринга, којим се испитује, анализира и мења постојећи систем са циљем да се реконструише у новој форми (Chikofsky & Cross, 1990, p. 15). Добро осмишљене, испланиране и имплементирани активности реинжењеринга могу да у многоме продуже животног века система.

Многи софтвери који су и данас у употреби, настали су пре појаве Web технологија, а неки су настали и много пре тога где је реализација интеракције запослених са софтверским системом била реализована путем карактерног корисничког интерфејса (енг. *character user interface*, скраћено CUI или CHUI). Одређени број произвођача софтверских развојних окружења, је омогућио да се уз одговарајуће реструктурирање постојећег софтверског кода које је потребно спровести и уз комбиновање са могућностима нове верзије платформе и новим власничким технологијама коју они нуде, добије се новије и побољшано решење уз нове функционалности.

Уколико се посматра OpenEdge платформа произвођача Progress Software Corporation као пример, видећемо да је то случај. Наиме, у напред наведеној платформи имплементирају се софтверска решења заснована на програмском језику ABL, раније познатијим као Progress 4GL уз употребу релационе базе података Progress. Прве верзије поменутог окружења, настале средином 80-их, користе искључиво карактерни кориснички интерфејс, док почетком 90-их настаје и верзија која може да користи графички кориснички интерфејс (енг. *graphical user interface*, скраћено GUI). Могућност рада са Web технологијама у поменутом окружењу настаје 1997. године, где се уз употребу претраживача (енг. *browser*) корисницима омогућава употреба и Web интерфејса. Да би се софтвер написани за карактерни кориснички интерфејс, прерадио у

софтвер са графичким корисничким интерфејсом потребно је одређено реструктурирање кода у случају да програмер у потпуности познаје претходно написан програм, тако да може да ради у новој графичкој форми. Уколико програмер не познаје програмски код тада је прво потребно његово испитивање и анализа, па тек након тога спровођење жељених измена. По сличном принципу, потребан је реинжењеринг постојећег софтвера да би се софтвер написан за карактерни или графички кориснички интерфејс уподобио да ради са Web технологијама. Одређени делови програмског кода је потребно да се прераде, тако да могу да служе као позадински део (енг. *back-end*) Web апликације који извршава пословну логику на Web серверу и комуникацију са системом за управљање базом податка. Елементи карактерног и графичког корисничког интерфејса се не могу искористити за Web интерфејс, па је потребно креирати нове уз употребу HTML, CSS и JavaScript технологија. Упоредне верзије софтверског решења које је прошло одређени степен реинжењеринга од карактерног, преко графичког до Web интерфејса можемо видети на слици 22.



На примеру видимо да софтверски реинжењеринг полази од претпоставке да неки софтверски систем већ постоји, па га је прво потребно разумети, а након тога трансформисати на начин да би се очувале његове главне функционалности. Софтверски реинжењеринг је потребан када документација система не постоји, а код је је нејасан, неструктуриран и веома га је тешко разумети. Програмери који су радили његовој изради више нису расположиви у организацији, па не постоји нико код би објаснио како систем функционише. Програмски код је једини извор информација. Може се догодити да је подршка за хардвер и софтвер застарела, па не може да прати организационе промене и такмичарски настројено окружење. Постоји велики ризик измене у наслеђеним системима уз високе трошкове њиховог одржавања (Majthoub et al., 2018, p. 266). Из поменутог разлога, потребно је са великом пажњом приступити модернизационим процесима, а поготово реинжењерингу, јер он спада у инванзивне стратегије модернизације.

У литератури наводе се 4 генерална циља реинжењеринга (Rosenberg, n.d.):

- припрема за унапређење функционалности
- побољшање одрживости
- миграција
- побољшање поузданости

Структура и логика наслеђеног система је најчешће нарушена као последица измена насталих вишегодишњим одржавањем. Уколико је документација неажурна постоји проблем да се сагледа систем од нивоа захтева, преко дизајна па до нивоа имплементације. Функционалност наслеђеног система је потребно да се побољша, па се реинжењеринг може искорисити као припрема за побољшање, јер ће се креирати спецификација тренутног софтверског система чиме ће се створити основа за разумевање његових кључних карактеристика. Када се утврди спецификација жељеног софтверског система и његове кључне карактеристике, постоји могућност упоређивања ове две спецификације и планирање корака које је потребно спровести да би се систем какав јесте, превео у жељено стање.

Током времена да би се софтверски систем прилагодио стално променљивом окружењу, отклонио постојеће грешке и спроводио своје функционалност, спроводе се активности одржавања у најширем смислу. Временом систем све више стари и његово одржавања је све теже. Реинжењерингом се омогућава да се смањи напори који се улажу у одржавање, јер систем постаје одрживији. Временом долази до промена у ИТ

индустрији, па се може догодити да верзија програмског језика, оперативног система или хардвера више не буде подржана. То наслеђени систем ставља у проблем, јер је сада потребно извршити миграцију са једног окружења на ново, што ће се реинжењерингом операционализовати. Такође веома је важно водити рачуна у остваривању компатибилности старих система са новим.

У пракси је веома чест случај да се наслеђени систем не разуме у целости. Свака промена на таквом једном систему, може да произведе ефекат таласа, што се често и дешава када постоји јака повезаност међу деловима софтверског система. Такође, услед кратких рокова за које је потребно омогућити одређене нове функционалности или изменити постојеће у систему, расте фреквентност ефеката таласа и временом систем постаје поприлично непоуздан. Реинжењинг се спроводи да би поменуто непоузданост свео на прихватљиву меру.

4.5.1.3. Основни појмови софтверског реинжењеринга

За опсежније сагледавање реинжењеринга постоји потреба да се направи јасна дистинкција између појмова реинжењеринг (енг. *reengineering*), реверзни инжењеринг (енг. *reverse engineering*) и инжењеринг у напред (енг. *forward engineering*), као и сродних појмова као што су редокументација (енг. *redocumentation*), опоравак дизајна (енг. *design recovery*) и реструктурирање (енг. *restructuring*), јер неретко постоји конфузија о томе на шта се који појам односи, какви су односи између њих и да ли један искључују другог или су пак део веће целине.

Међу првим истраживачима области реинжењеринга Чикофски и Крос посматрају животни циклус развоја софтвера кроз неколико фаза: захтеви, дизајн и имплементација (Chikofsky & Cross, 1990, р. 14). Овакво гледиште је базирано на три концепта: моделу животног циклуса развоја софтвера, постојање предметног система и могућности идентификације нивоа апстракције. Развој софтвера је веома сложен, а када се узме у обзир његова нематеријална природа која указује на значајно разликовање у односу на производњу осталих индустријских производа најчешће материјалне природе, исказује се потреба да се упростити његов развој. Такво упрошћавање води до креирања разних модела животног циклуса развоја софтвера, путем којих се процес развоја софтвера дели на фазе. Излази из једне фазе се улази у другу фазу, негде постоје и могућности повратних веза из тренутне фазе у ону која јој претходи, као и одређене итеративне фазе које се извршавају до испуњења неког услова, нпр. завршетка израде једног модула или целокупног софтверског система.

Предметни систем је најопштије речено систем, програм, програмски модул, неколико повезаних програма који се извршавају у уређеном току, део кода и тд., над којим се врши посматрање и над којим се спроводе одређене операције уређене процесима инжењеринга у напред, реверзног инжењеринга и тд. За инжењеринг у напред предметни систем је софтверски систем који је потребно развити, док за реверзни инжењеринг предметни систем је наслеђени систем од којег се полази и над којим се спроводе процеси реверзног инжењеринга.

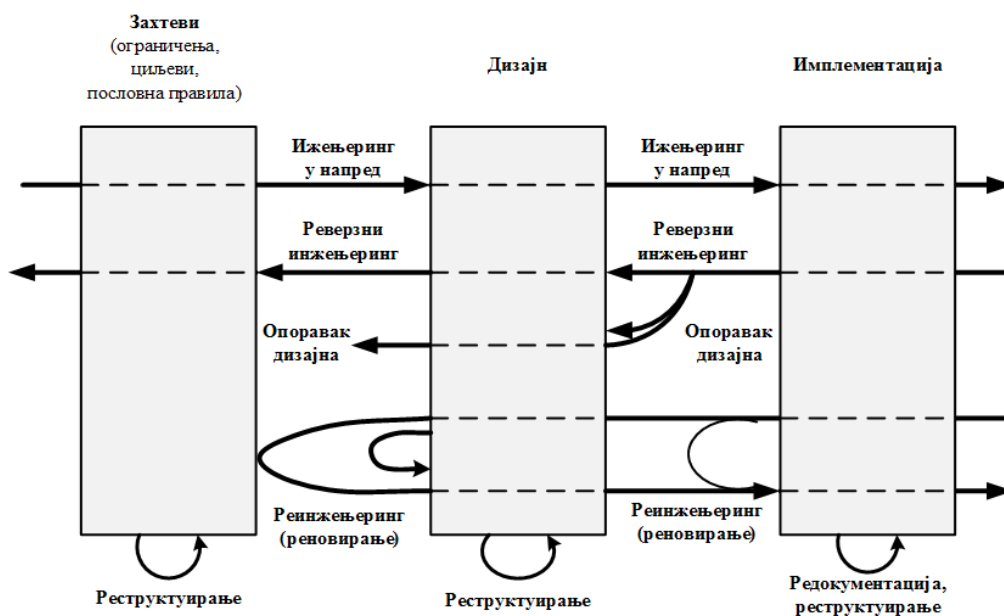
„Апстракција је контролисано и постепено укључивање детаља у опис система, „сакривање“ детаља у описивању система, односно извлачење и приказивање општих, а одлагање описивања детаљних особина неког система“ (Lazarević et al., 2006, p. 15). Модел животног циклуса развоја софтвера на у својим почетним фазама има концепте који веома апстрактни и имплементационо независни. Ти концепти описују пословање предузећа на високом нивоу, уз пословна правила која описују начин и логику реализације одређених пословних операција, а која су касније потребна да буду имплементирана у крајњи производ, тј. софтвер. Како се крећемо ка каснијим фазама тако се ниво апстракције смањује, полако се укључују се имплементациони детаљи и поменуте операције добијају конкретније форме како напредујемо из фазе у фазу.

Дакле, између фаза у моделу животног циклуса развоја софтвера и нивоа апстракције постоји мапирање где се у једној фази фокусирамо на важне концепте везане за ту фазу, а занемарујемо оне детаље који нису релеватни за ту фазу и тај ниво апстракције, док их у наредној фази развоја разрађујемо и тада се фокусирамо на њих. Нпр. претпоставимо да смо у фази креирања дизајна система где структуру система, његове елементе и везе између њих можемо представити UML дијаграмом класа и тада игноришемо имплементационе детаље. Имплементациони детаљи ће бити реализовани у наредној фази где ћемо се фокусирати на њих, као и на њихови реализацију у одређеном програмској језику и на концепте које нуди тај језик.

Захтеви (енг. *requirements*) нам говоре о томе шта крајњи корисници желе, тј. шта софтверски систем је потребно да ради, који проблем систем треба да реши. Поред тога, садржи спецификацију система, пословна правила и ограничења и тд.

Дизајн (енг. *design*) креће од корисничких захтева и спецификације и преводи их у форму која је прилагођена програмерима који је потребно да имплементирају систем. Дизајном се на високом нивоу описује како ће кориснички захтеви бити остварени.

Имплементација (енг. *implementation*) је реализација софтверског решења, где се након фазе дизајна елементи дизајна конкретизују у програмски код, тестирају да ли решење испоручује специфициране функционалности. Када је све то задовољено систем се ставља у продукцију. Однос поменуте три фазе животног циклуса развоја софтвера са различитима нивоима апстракције је приказан на слици 23., где се приказује уједно и однос међу појмовима инжењерства у напред, инверзног инжењерства, реинжењеринга, редокументације, реструктурирања и обнове дизајна.



4.5.1.3.1. Инжењеринг у напред

Инжењеринг у напред (енг. *forward engineering*), је само назив за традиционални развој софтвера, тј. софтверски инжењеринг да би се учинила јасна дистинкција у односу на реверзни тј. обрнути инжењеринг. На почетку се креће од захтева, преко концептуалног дизајна, детаљног дизајна и долази се до имплементације софтвера који је потребно направити. Дакле, то је поступак којим се креће са високог нивоа апстракције, од захтева и имплементационо независног дизајна до конкретног софтверског решења у изабраном имплементационом окружењу и програмском језику, тј. најнижег нивоа апстракције.

Инжењеринг у напред са сваком својом фазом смањује ниво апстракције система и води га од апстрактних до конкретних информација у вези софтверског пројекта. Као крајњи производ је софтверско решење које треба у потпуности да испуњава захтеве корисника. Ипак у неким изворима се наглашава да постоји разлика између софтверског инжењеринга и инжењеринга у напред која се огледа у томе да инжењеринг у напред користи елементе софтвера који су добијени из наслеђеног система и да уз нове захтеве креира нови жељени систем (JLC/CRM, n.d.). Тиме се указује на различите полазне основе инжењеринга у напред и софтверског инжењеринга.

4.5.1.3.2. Реверзни инжењеринг

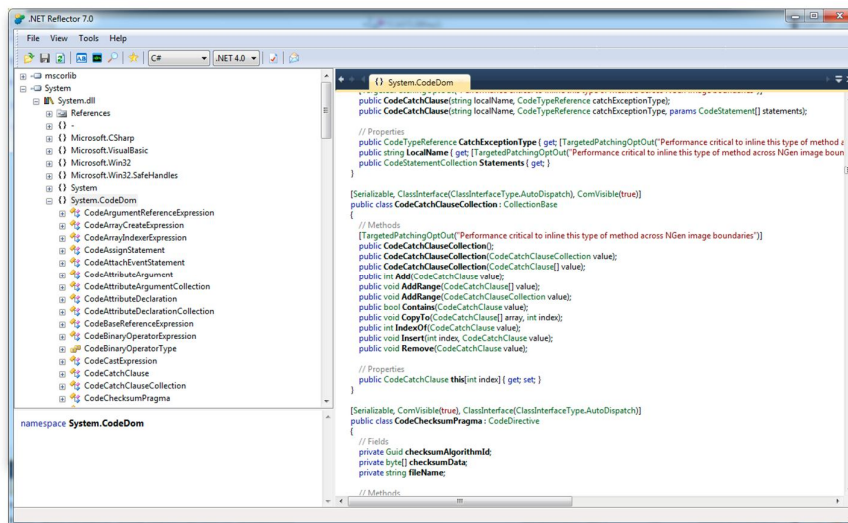
Реверзни инжењеринг⁵ (енг. *reverse engineering*), који се још назива и обрнути инжењеринг или инжењеринг у назад (енг. *backward engineering*) је обрнути процес од раније описаног инжењеринга у напред. Розенберг истиче да је предмет анализе реверзног инжењеринга постојећи систем у којем је потребно да се утврде његове компоненте, као и релације између тих компоненти, са идејом да се креирају репрезентације система на вишем нивоу апстракције, тј. у другој форми (Rosenberg, n.d.). Реверзни инжењеринг креће од изворног кода (енг. *source code*) предметног система, а за случај да не постоји онда покушава да га поврати из извршне верзије софтвера, најчешће специјалним алатима.

Узмимо као пример Microsoft .NET платформу и програмски језик C# (енг. *csharp*) који се користи као основни програмски језик за разне типове софтвера које је могуће правити у .NET платформи. Прво ћемо укратко објаснити процес компајлирања у .NET окружењу. Употребом програмског језика C#, можемо креирати апликацију са графичким корисничким интерфејсом у технологији WindowsForms. Процес компајлирања ове апликације се спроводи у две фазе. У првој фази се из изворног C# кода добија међујезик (енг. *Intermediate Language*, скраћено IL) и он је смештен унутар склопа (енг. *assembly*). Обзиром да смо одабрали WindowsForms технологију за креирање апликације, склоп има екстензију фајла .exe. Након тога се покреће други компајлер, JIT (енг. *just-in-time*) компајлер који узима међујезик добијен у претходном кораку и од њега прави нативни (енг. *native code*), тј. машински код који се извршава од стране оперативног система. Описани поступак JIT компајлирања је у надлежности виртуелног окружења које се назива CLR (енг. *Common Language Runtime*) и које за .NET платформу има сличну функцију као JVM (енг. *Java Virtual Machine*) за програмски језик Java.

⁵ често се употребљава и назив реверзибилни инжењеринг

Претпоставимо сада да програмери желе да изврше реверзни инжењеринг поменутог C# кода и да креирају дизајн из тог изворног кода. Уколико не постоји изворни код у пракси из било ког разлога, постоје специјални софтверски алати који могу да из извршне верзије програма врате изворни код. Један од таквих алата за .NET платформу је .NET Reflector, који има могућност да прочита фајл, у конкретном случају склоп, који у себи садржи међукод и да на основу тога реконструише изворни C# код (Gumoz, n.d.). Поменути алат је један од многих који омогућава описану функционалност.

Реверзни инжењеринг креће од најнижег нивоа апстракције на којем се налази изворни код, врши извлачење дизајна из кода и тиме уопштава информације добијене из кода, занемаривањем неких делова. На тај начин помера се на виши ниво апстракције који је мање имплементационо зависан, а након тога утврђује захтеве на највишем нивоу апстракције. Он нема циљ да изврши неку промену над софтверским системом, већ да анализира и испита постојећи систем и да га припреми за касније промене које ће уследити (Majthoub et al., 2018, p. 267).



Слика 24 .NET Reflector, генерисање изворног кода из склопа, преузето (Gumoz, 2010)

Уколико је документација наслеђеног система не постоји или је неажурна, реверзним инжењерингом ће се извући дизајн, структура и захтеви система, али и одређена пословна правила и процеси која су инкорпорирани у том систему. Реверзним инжењерингом се постојећи систем боље разуме, па је он драгоцен како за будући реинжењеинг, тако и за процесе одржавања, поготово превентивног.

Услед брзих промена у технологијама које доводе до застаревања софтверских система, реверзни инжењеринг има додатне циљеве које испуњава (Chikofsky & Cross, 1990, p. 16):

- суочавање са сложености
- генерисање алтернативних погледа
- враћање (реконструкција) првобитне информације
- детектовање нежељених ефеката
- синтетизовање виших апстракција
- олакшавање поновне употребе

Као што је раније речено, сложеност софтверског система у великој мери утиче на процесе који имају циљ да остваре његово побољшање, јер га је тешко разумети. Реинжењеринг би требало то да омогући, пре свега на начин да постоји што мање рада програмера и пројектаната који би се састојао из анализирања кода. Самим тим би требало омогућити нове алате који ће служити као подршка у решавању овог типа проблема.

Један од најбољих начина да се оствари приказ софтверског система на вишем нивоу апстракције и олакша његово разумевање је постојање алата које омогућавају визуелну репрезентацију тог система. Добра пракса би била да се разни типови дијаграма могу приказати, тако да се омогући програмерима и пројектантима да сагледају систем из више различитих углова, а не уско само из једног угла као што већина алата омогућава. Нпр. поред приказа дијаграма тока којим се описује одређено понашање, добро било ако може да се генерише и UML дијаграм класа или дијаграм објекти-везе који би описао статичке елементе система и везе између њих. Када документација не постоји или није ажурна услед вишегодишњег неодржавања и неспровођења промена на њој тако да осликава последњу верзију софтверског система, реверзни инжењеринг је кључан за разумевања система и интеркације елемената система (програма, класа, објеката и тд.).

Уз реверзни инжењеринг могу се лако уочити мањкавости система настале иницијалним узимањем захтева, дизајном и кодирањем као и бројним узастопним променама које нису документоване. На поменути начин може се предупредити да систем се не понаша у складу са захтевима и потребама корисника. За реверзни инжењеринг је важно да оставари висок ниво апстракције, па су потребне методе и технике које ће то омогућити.

Поновна употреба одређеног дела софтверског кода, је веома значајна са аспекта ефикасности софтверског решења, као и једноставне кастомизације постојећег кода различитим крајњим корисницима. Уколико постоје целине истог кода, реверзни инжењеринг их може препознати и предложити као кандидате за поновно употребљиве софтверске компоненте. На тај начин се побољшава структура и олакшава разумевање постојећег система, а поготово када се планирају будуће измене које могу користити поменуте компоненте и када је потребно водити рачуна да систем настави да исправно функционише.

Поређење између инжењеринга у напред и реверзног инжењеринга је дат у табели (GeeksforGeeks, n.d.):

Табела 3. поређење инжењеринга у напред и реверзног инжењеринга., преузето (GeeksforGeeks, 2020)

| инжењеринг у напред | реверзни инжењеринг |
|---|---|
| апликације се развијају на основу задатих захтева | информације су прикупљене из дате апликације |
| професионалне вештине високог нивоа | нижи ниво професионалних вештина |
| више времена да се развије апликација | мање времена да се развије апликација |
| природа је заснована на правилима | природа је прилагодљива |
| производња се покреће за задатим захтевима | производња се покреће узимањем постојећег производа |

Редокументација (поновна документација) и опоравак дизајна су две популарне технике реверзног инжењеринга.

4.5.1.3.3. Редокументација

Редокументација (енг. *redocumentation*) је једноставна техника обрнутог инжењеринга која има за циљ да за наслеђени систем који нема документацију или чија је документација неажурна изврши њено поновно генерисање на начин да она садржи све релевантне информације о систему чиме ће се увећати његово укупно разумевање прилагођено људима.

У литератури постоји већи број предложених приступа процеса редокументације. Тили указује да је потребно направити разлику између процеса редокументације малих и великих софтверских система, јер није исто посматрати софтвер који има неколико хиљада линија кода и софтвер који има неколико стотина хиљада линија кода или чак неколико милиона (Tilley, 1993, p. 1083). Основна дистинкција између ове две групе софтверских система јесте да је код малих софтверских система у документацији акценат на опису система алгоритмом над одређеном структуром података, док код великих система фокус је на архитектури која треба да сагледа елементе система и везе између

тих елемената. Код великих софтверских система опис делова система није пожељан јер то води ка изолованој слици о целокупном систему, без могућности да се он у потпуности сагледа. Из тог разлога, важна је укупна структура тог система са свим софтверским артефактима на основу којих програмери могу да формирају ментални модел, тј. представу о систему. Артефакти су софтверске компоненте, процедуре, модули, подсистеми и интерфејси, са зависностима између њих приказаних наслеђивањем или контролим током и тд. У приступу који поменути аутор истиче јесте акценат на креирању оперативне архитектуре система уз употребу неког окружња за реверзно инжњерство, и структуре која омогућава вишеструке апстрактне погледе на тај систем.

Аутори заговарају приступ структурне редокументације којим се постижу архитектурални аспекти софтвера уз примену алата Rigi који су развили у ту сврху (Wong et al., 1995). Поменути алат омогућава пренос апстрактних информација о софтверској архитектури који је у складу са менталним моделом програмера који одржавају софтвер. Поменти алат се састоји од 3 компоненте: RigiReverse, RigiServer и RigiEdit. RigiReverse је подсистем за парсирање путем којег се извлаче информације из изворног кода. RigiServer служи да тако извучене информације сачува у репозиторијуму. RigiEdit омогућава приказ различитих репрезентација програма у форми графа. Сагледавањем графа програмери који раде на одржавању и који ће вршити реинжњеринг предметног система употребом алата могу да сагледају његову структуру.

Рајлич приказује стратегију инкременталне редокументације употребом Partitioned Annotations of Software, скраћено PAS алата (Rajlich, 2000). Када неки тим ради на редокументацији одређеног наслеђеног система он користи PAS алат да запамти свако разумевање дела софтверског система од стране програмера у формату хипертекста. Како анализира код програмер полако креће да разуме одређене његове целине и сваку своју опсервацију на било којем нивоу апстракције може да забележи у PAS употребом аотација. Свака аотација се може описати са више различитих нивоа апстракције и углова посматрања, што је значајно за архитектурално разумевање система.

Група аутора предлаже спровођење процеса редокументације уз потребу алата Redoc (Anquetil et al., 2005). Процес редокументације се састоји од три главне фазе и седам активности. Главне фазе су: фаза припреме, фаза планирања и фаза редокументације. Фаза припреме врши анализу стања софтвера и дотадашње документације. У фази планирања се одлучује који делови система ће бити редокументовани на почетку. Редокументација је главни фаза у којој се креирају различита документа. Фаза припреме има две активности инвентарисање система и

процену система. Инвентарисањем се добијају основне информације о величини проблема и активностима које ће наступити након ње. У процени система се одређује ниво поверења у изворе информација о систему, као што су код, тренутна документација и тд. Фаза планирања редокументације дефинише начин на који ће се процес спроводити и проритети у том процесу. Фаза редокументације има четири активности: дефиниција на високом нивоу, извлачење повезаности, дефиниције подсистема и документацију ниског нивоа. Прво се на документују погледи на систем на високом нивоу као што су функционалности и интеракција са другим системима и тд. Затим се идентификују повезаности рутине са рутином, рутине са подацима, подаци са подацима и функционалности са рутинама. Након тога се систем посматра од врха ка дну, где се уочавају подсистеми, компоненте и тд. На крају се врши коментарисање сваког елемента који је идентификован. Потребно је рећи да се неке од наведених активности одвијају на потпуно аутоматизован начин, неке на полу-аутоматизован начин а неке се и даље обављају мануелно.

Сви наведени приступи у процесу редокументације теже да што боље опишу елементе система, везе између њих и да створе могућност програмерима који раде на његовом одржавању да имају бољу представу о том систему. У свим приступима је потребно корисити алате у процесу редокументације, поготово за велике наслеђене системе које у себи имају много програмског кода, да би се успешније представила његова структура и понашање.

4.5.1.3.4. Опоравак дизајна

Опоравак дизајна (енг. *design recovery*) служи да би се идентификовале апстракције система вишег нивоа. Изворни код често није довољан да би се рекреирао дизајн система, па су потребни додатни извори информација који ће потпомоћи поменути процес. Према Бигерстафу идеја је да се међусобним дејством документације постојећег система (уколико постоји), програмског кода, знања о проблему и апликационом домену, као и искуством тима који ради на томе (поготово аналитичара система) поново креирају апстракције дизајна (Biggerstaff, 1989, p. 36). Такође процес мора да буде прилагођен људима којима је намењен, на начин да особа у потпуности разуме све релевантне информације о систему. При корак јесте да се софтвер разуме кроз своје елементе као што су модули, софтверски артефакти и тд. Након тога, врши се попуњавање библиотека за поновну употребу и опоравак, чиме се ствара основа за генерализацију компоненти и њихову поновну употребу. На крају је потребно применити резултате опоравка дизајна.

4.5.1.3.5. Реструктурирање

Реструктурирање (енг. *restructuring*) у свом изворном облику подразумева мењање структуре кода једног софтверског решења у бољу форму којом се омогућава једноставније одржавање. Посматрано из поменутог угла појам реструктурирања је сличан појму рефакторисања (енг. *refactoring*), али постоје разлике у поменутим појмовима. Рефакторисање је више везано за промену у коду која треба да допринесе побољшању нефункционалних карактеристика и везано је углавном за објектно-орјентисане програмске језике, док реструктурирање је више усмерено на побољшање модуларности.

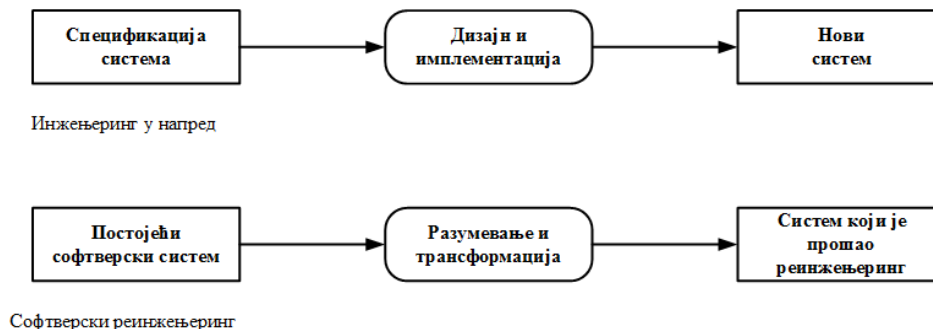
Ипак у односу на свој изворни облик реструктурирање је шири појам који се односи на трансформацију из једне форме репрезентације у другу на истом нивоу апстракције, уз очување екстерног понашања (Chikofsky & Cross, 1990, p. 15) . У том смислу промене не морају да се односе само на ниво апстракције на којем се налази програмски код већ могу да се укључе и одређена преобликовања модела података и тд. Највише се користи да би се софтвер једноставније схватио и променио уз смањење могућности грешке, па на тај начин чини процес одржавања једноставнији и то пре свега превентивног одржавања (Arnold, 1989, p. 607). Реструктурирањем се може смањити одређена комплексност и побољшати модулартност и боље схватити постојеће решење, а то је предуслов за његову промену.

4.5.1.3.6. Софтверски реинжењеринг

Софтверски реинжењеринг (енг. *software reengineering*) у литератури познат и под именом обнова или реновирање (енг. *renovation*) повезује и комбинује готово све претходно наведене појмове и методе као што су реверзно инжењерство, редокументација, реструктурирање, транслација (превођење) и инжењерство у напред (Maithoub et al., 2018, p. 268). Прелазећи на више нивое апстракције од имплементације ка дизајну и спецификацији покушавамо да разумемо постојећи наслеђени систем и створимо свеобухватну слику о њему, тј. сагледавамо систем какав јесте (енг. *as-is*). Након што га разумемо вршимо његову поновну имплементацију да би смо добили систем какав желимо да буде (енг. *to-be*). Као резултат тог процеса имамо побољшан систем са аспекта функционалних и нефункционалних захтева, жељених карактеристика система које одговарају захтевима, као и његове интерне структуре.

Основна разлика између инжењеринга у напред и софтверског реинжењеринга, јесте да се код инжењеринга у напред креће од спецификације система и долази се до новог система, преко дизајна и имплементације, док се код реинжењеринга започиње са постојећим системом и уз његово разумевање и трансформацију добија се преправљен

систем (Sommerville, 2006, p. 527). Ова дистинкција графички је представљена на слици 25.



4.5.1.4. Традиционални модел софтверског реинжењеринга

За разумевање традиционалног, познатог под другим називима класични или општи модел, софтверског реинжењеринга, потребно је да се направи дистинкција између почетних истраживања Чиковског и Кроса (Chikofsky & Cross, 1990) и истраживања Линде Розенберг (Rosenberg, n.d.) у погледу на систем са различитог броја нивоа

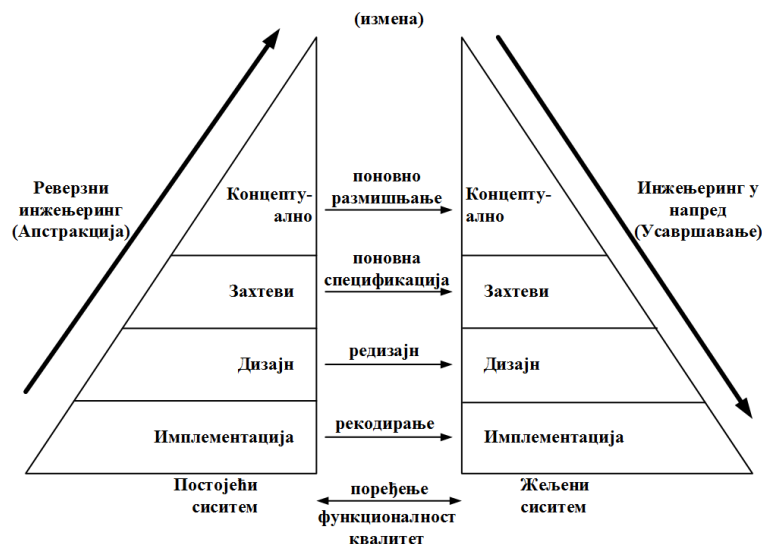
Слика 25. процес реинжењеринга, преузето (Sommerville, 2006)

апстракције. Розенберг проширује нивое апстракције у односу на Чиковског и Кроса, који су се више усмерили на таксономију предметне области, где се нивои апстракције као што су ниво апстракције имплементације (енг. *implementation*), ниво апстракције дизајна (енг. *design*) и ниво апстракције захтева (енг. *requirements*) проширују са још једним нивоом апстракције, концептуалним (енг. *conceptual*) нивоом, тј. највишим нивоом.

Концептуални ниво апстракције је највиши ниво у хијерарји и систем је у њему описан на најапстрактнијем нивоу, где су његове функционалности само назначене. Ниво апстракције захтева детаљно описује функционалне карактеристике које систем поседује. На нивоу апстракције дизајна систем је описан својом архитектуром, компонентама, интерфејсима између компоненти, процедурама и структурама података. На нивоу апстракције имплементације, најнижем нивоу апстракције систем је описан на начин да га рачунар разуме путем имплементационих карактеристика. Последња два нивоа апстракције пружају информације о интерној структури система, у односу на прва два нивоа која описују систем без залажења у његове детаље. Дакле, са опадањем хијерархије

ниво детаља је све већи и већи, тј. концепти система су мање апстрактни а више конкретни, док са напретком у хијерархији концепти система су све више апстрактнији. Оба поменута процеса имају своје место у традиционалном моделу софтверског реинжењеринга.

Код традиционалног модела софтверског реинжењеринга, као што је раније поменуто, на најнижем нивоу апстракције налази се имплементација, тј. програмски код са којим започиње процес реинжењеринга. У зависности од величине промене које је потребно спровести, тј. од тога шта се жели постићи са постојећим системом зависи и обим промена које је потребно спровести. У неком најједоставнијем сценарију, потребно је само пребацити из одређеног програмског језика постојећи систем у жељени програмски језик и тада је фокус искључиво на тој активности. Сложенија сценарија захтевају измене на апстрактнијим нивоима у хијерархији, када је потребно изменити дизајн или изменити захтев и тд. Тада је број активности може бити далеко већи. Традиционални модел софтверског реинжењеринга је приказан на слици 26. У основи софтверског реинжењеринга постоје три главна принципа: апстракција (енг. *abstraction*), измена (енг. *alteration*) и усавршавање (енг. *refinement*) (Rosenberg, n.d.). Апстракција одговара процесу реверзног инжењеринга, где се кретањем од имплементације до концептуалног нивоа увећава ниво апстракције система, тј. концепти система су све више апстрактнији, а детаљи се занемарију. Изменом се врше измене у систему на појединачном нивоу апстракције, где се врше одређене измене, брисања и модификовања концепата, а при томе се задржавају функционалности. Усавршавање одговара процесу инжењеринга у напред, где се од концептуалног до имплементационог нивоа, смањује ниво апстракције у опису концепата система увођењем детаља.



Слика 26. традиционални модел софтверског реинжењеринга, преузето (Rosenberg, n.d.)

акле, процесом реверзног инжењеринга се увећава ниво апстракције у опису карактеристика софтверског система, тј. ниво детаља се смањује, док се процесом инжењеринга у напред смањује ниво апстракције у опису карактеристика софтверског система, тј. ниво детаља се увећава и опис карактеристике постаје конкретнији. Превођење, тј. рекодирање софтверског решења из једног језика у други не захтева спровођење реверзног инжењеринга, али измене на било којем другом нивоу апстракције захтевају реверзни инжењеринг.

4.5.1.5. Аспекти софтверског реинжењеринга

За успешно спровођење софтверског реинжењеринга, размотрићемо следеће његове аспекте: економски, организациони, правни и технички (Војић & Велашевић, 1999). Технички аспект је у суштини преглед стратегија модернизације, јер се у тренутку посматрања проблема софтверског реинжењеринга аутори Бојић и Велашевић стратегије модернизације посматрали као стратегије реинжењеринга, док је у тези заузет став да се софтверски реинжењеринг посматра као само једна од стратегија модернизације.

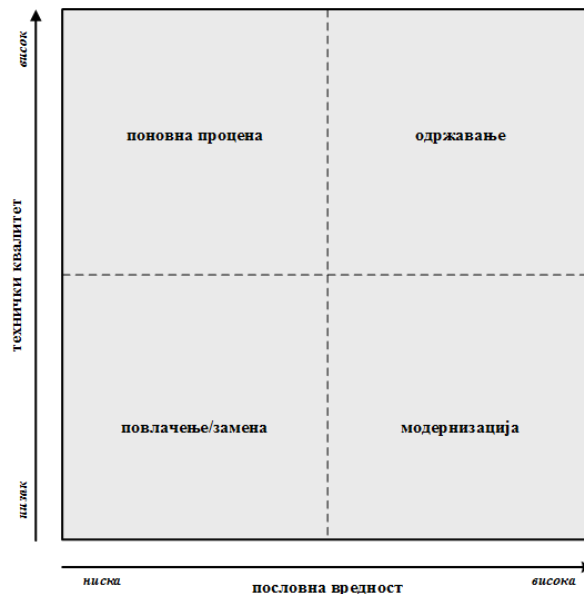
4.5.1.5.1. Економски аспект

Економски аспект треба да одговори на питање да ли је реинжењеринг оправдан у поређењу са осталим стратегијама решавања софтверског наслеђа, при чему се састоји од следећих активности (Sneed, 1995): оправданост пројекта, анализа портфолија, процена трошкова, анализа трошкова и користи и уговарање. На сваком ниову постоји скуп активности које је потребно спровести да би се утврдила економска димензија система.

Оправданост пројекта (енг. *project justification*) укључује анализу постојећег софтверског решења, процес одржавања и пословну вредност апликације. Потребно је да се заштити повраћај уложених средстава (енг. *return of investment, ROI*), до којег нивоа ће се повећати квалитет софтвера, како ће се процес одржавања побољшати и које пословне вредности ће бити унапређене ?

Анализа портфолија (енг. *portfolio analysis*) служи да се утврде које су то апликације приоритет за процес реинжењеринга у односу на свој технички квалитет и пословну вредност. Према Нолану Нортону ове две димензије имају по два сегмента: високе и ниске вредности па се формирају 4 квадранта, од којих сваки указује на одређену групу активности које је потребно предузето (Willem-Jan, 2009, p. 28). Анализа портфолија је приказана на слици 27. Процена трошкова (енг. *cost estimation*) се спроводи

идентификовањем и оцењивањем које све софтверске компоненте је потребно да прођу реинжењеринг. Анализа трошкова и користи (енг. *cost-benefit analysis*) упоређује трошкове реинжењеринга са очекиваним умањеним трошковима одржавања и увећање вредности.



Слика 27. анализа портфолија, преузето (Willem-Jan, 2009)

Уговарање (енг. *contracting*) се односи на тип уговора који ће се закључити за послове реинжењеринга, где Снид указује да је боље да буду уговорени по учинку, тј. резултатима за разлику од уговора за развој софтвера код којих је мање детаљ познато па су уговорени да се спроведу у одређеном временском року (Sneed, 1995). У том смислу он упућује да трошкове треба узети за мерљиви учинак над сиситемом, а не за време који је неко спреман да наплати.

4.5.1.5.2. Организациони аспект

Бојић и Велашевић указују на припремене активности које се могу спровести према Центру за подршку софтверским технологијама (Војић & Велашевић, 1999):

1. дефинисање циљева и праваца
2. формирање тима за реинжењеринг
3. провера постојећег процеса одржавања/развоја софтвера
4. одабир скупа софтверских метрика
5. анализа наслеђеног софтвера
6. дефинисање процеса спровођења реинжењеринга
7. развијање/допуњавање тестних примера

8. анализа и избор реинжењерских алата
9. обука чланова тима

Реинжењеринг је дакле један велики подухват који захтева опсежне организаионе припреме пре него што ступи у реализацију.

4.5.1.5.3. Правни аспект

Обзиром да је реверзни инжењеринг, део процеса софтверског реинжењеринга у којем се стиче знање о систему на различитим нивоима апстракције, потребно је водити рачуна да се не развије сувише сличан производ неком оргиналном, јер тада може да дође до нарушавања ауторских права. У складу са тим потребно је повести рачуна о инжењерингу у напред који се спроводи, тако да у одређеној мери осликава жељено решење, а при томе не доведе до кршења ауторских права. Иако је у овом раду фокус на реинжењеринг наслеђених система, треба имати у виду и овај аспект који је веома значајан у условима заоштрене конкуренције, па је упутно ангажовати правну помоћ која може бити добар консултант по овом питању и пружити адекватан савет (Behrens & Levary, 1998).

4.5.1.6. Приступу софтверском реинжењерингу

Постоји пет основних приступа у реинжењерингу софтверског решења, који се се међусобно разликују у односу на циљеве пројекта, доступност ресурса, тренутно стање система над којим ће бити спроведен реинжењеринг и ризике у пројекту реинжењеринга (Tripathy & Naik, 2015, p. 144) :

- приступ „велики прасак“
- инкрементални приступ
- парцијални приступ
- итеративни приступ
- еволутивни приступ

Приступ „велики прасак“ (енг. *big bang approach*) је такав да је цео систем одједном замењен и из тога разлога се назива и “Lump Sum” приступ (приступ одједном). Користи се када је потребно да се у софтверском пројекту проблем реши одмах, а реинжењеринг не може да се спроведе у деловима, као што је пребацивање на другу архитектуру (Majthoub et al., 2018, p. 268). Цео систем одједном наставља да функционише у новом окружењу. Није погодан за велике системе јер може да се искористе сувише велика средства, а да се напори реинжењеринга не примећују све док се не успостави жељени систем.

Инкрементални приступ (енг. *incremental approach*), познат још под називом постепени престанак употребе (енг. *phase-out*) или додатак (енг. *additive*), заговара да се над деловима система спроводи реинжењеринг и такви прерађени делови се инкрементално додају као нове верзије система при чему задовољавају нове циљеве (Rosenberg, n.d.). Уколико се процес реинжењеринга спроведе на овај начин крајњи корисници система виде постепени напредак и измене у систему, па се лакше прилагођавају него што је у случају када се систем одједном промени. За измену система у целости потребно је више времена обзиром да се део по део мења, а при томе је потребно водити рачуна о међусобним везама и верзијама тих компоненти (путем система за верзионирање кода), тако да се систем не наруши. Недостатак овог приступа је да се иницијална структура система не може изменити, јер се промена одвија искључиво на начин да се стари делови система мењају новим, а при томе се не мења број делова система (додавањем нових или уклањањем постојећих компоненти) и везе између њих. Због тога што се реинжењеринг спроводи део по део, приступ је мање ризичан од приступа „великог праска“.

Парцијални приступ (енг. *partial approach*) се користи када се спроводи реинжењеринг дела система и затим интегрише са остатком система над којим није спроведен реинжењеринг (Tripathy & Naik, 2015, p. 145). На одређени начин целокупан посао реинжењеринга се смањује и прилагођава потребама организације, јер и оне имају ограничена средства која желе да искористе за поменуте послове. Када неки део система постане толико проблематичан, да је његове отказе немогуће више одржавати, тј. да је компликовано спроводити реактивно одржавање или су трошкови сувише велики, тада се парцијални приступ може показати као исправна одлука, јер се смањује време и трошкови реорганизације система. Модификације компоненти не укључују делове кода који су везани за интегрисање истих са другим компонентама, што може бити проблематично.

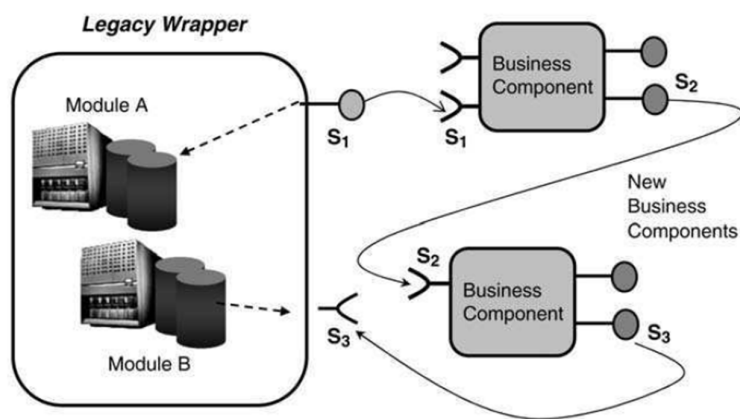
Итеративни приступ (енг. *iterative approach*) се примењује на неколико процедура истовремено, где свака операције реинжењеринга траје кратко времена (Tripathy & Naik, 2015, p. 145). Неколико типова компоненти у систему истовремено постоји током реинжењеринга, а то су старе компоненте, нове компоненте над којима је спроведен реинжењеринг, оне над којима се тренутно спроводи реинжењеринг и нове компоненте које су додате у систем. Добре стране овог приступа јесте што се на одређени начин чува сличност система чиме се олакшава одржавање, а корисници имају исто или слично корисничко искуство, а друго јесте да систем функционише и током реинжењеринга.

Потребно је омогућити да се испрате и исконтролише потенцијално велика количина разнородних компоненти система, што може представљати проблем, па је потребно корисити софтвер за верзионирање кода (енг. *software versioning*).

Еволутивни приступ (енг. *evolutionary approach*) има одређене сличности са инкременталним приступом, а разлика је да замена компоненти постојећег система са компонентама новог система не спроводи тако да се очува структура, већ се то ради засновано на функционалности (Majthoub et al., 2018, p. 268). Примена овог приступа је погодна када се реинжењеринг спроводи тако да систем настави да функционише у објектно-орјентисаним технологијама, подстиче се модуларност чиме се подстиче кохезиван (енг. *cohesive*) дизајн и смањује се опсег компоненте (Rosenberg, n.d.). Недостатак се огледа да уколико постоје сличне функционалности система оне се морају уочити, а затим прерадити тако да од њих постане једна функција, што не значи да је то једноставан задатак.

4.5.2. Облагање

Један од начина за превазилажење проблема наслеђених система јесте да се користи техника облагања⁶ (енг. *wrapping*) када не желимо да извршимо реинжењеринг система или да систем заменимо комплетно. На тај начин се омогућава да наслеђена апликација или део наслеђеног система функционише са другим апликацијама и извршава заједничке пословне процесе (Willem-Jan, 2009, p. 41), као на слици 28. Облагањем се постиже могућност постепене замене наслеђеног система или неког његовог модула, апликације, процедуре и тд.



Слика 28. облагање наслеђеног система, преузето (Willem-Jan, 2009)

⁶ назива се и техника омотавања

Када имамо апликацију која није организована као скуп компонената облагањем постижемо њену модуларизацију и енкапсулацију, тако да се може корисити као било која друга компонента. Постоји неколико врста омотача (Orfali et al., 1995):

- омотачи базе података - обезбеђују приступ наслеђеној бази података из неког модерног, најчешће објектно-орјентисаног програмског језика
- омотачи системских сервиса - обезбеђују приступ системским сервисима без познавања интерних интерфејса
- омотачи апликација - омогућавају облагање система са пакетном обрадом или неког другог система, при чему омогућавају другим компонентама да их позову и спроведу одређене операције
- омотачи функција - омотавају појединачне функције унутар програма и омогућавају њихов позив из клијентских апликација

Омотач омогућава да се обави размена порука између наслеђене компоненте и клијента који жели да позове ту компоненту, тј. да јој проследи неке параметре. Задатак омотача јесте да преформатира поруку клијента у формат који одговара наслеђеној компоненти, модулу, систему или програму и да их проследи, где након тога чека одговор од наслеђене компоненте и затим га реформатира и шаље назад до клијента (Sneed, 2000). Поменута комуникација се остварује путем апликационог програмског интерфејса (енг. *application program interface*, API) омотача.

Једна од често спомињаних идеја у литератури јесте облагање наслеђених апликација, тако да оне могу да постану део шире сервисно-орјентисане архитектуре (енг. *service-oriented architecture*, SOA), тако што ће се облагањем омогућити да наслеђене апликације буду Web сервиси. Web сервиси омогућавају да се раније имплементирана софтверска решења сачувају и да се њихове функционалности искористе у новом систему, где ће они постати сервиси. На тај начин ће се наслеђен систем учинити поново употребљив (енг. *reusable*) и компатибилан са стандардима који важе у индустрији. У том смислу потребно је спровести следеће (Sneed, 2006):

- сачувати наслеђен код
- обложити наслеђен код
- учинити код доступан као Web сервис

Поменути начин није инвазиван у смислу да ће нарушити постојећи систем, већ ће омогућити посредством омотача да се наслеђени систем интегрише у ширу целину, тако да се ствара начин да се проблеми интеграције реше.

4.5.3. Миграција

Миграција (енг. *migration*) је појам који је дуго у употреби у информационим технологијама може да означава више ствари: миграцију података, миграцију схеме базе података, миграцију софтвера, миграцију садржаја, миграцију виртуалне машине и тд. Сваки од ових појмова подразумева пребацивање одређених ентитета у нову форму и наставак њиховог рада. Када говоримо о наслеђеним системима најчешће се мисли на миграцију софтвера (енг. *software migration*) или миграцију података (енг. *data migration*). Такође постоје и схватања у теорији и пракси по којима је миграција назив за све технике решавања проблема наслеђених система. У овом раду миграција ће се посматрати као само једна од техника за решавање проблема наслеђених система. У односу да ли се миграција спроводи целокупна или само део разликујемо комплетну миграцију (енг. *complete migration*), инкременталну миграцију (енг. *incremental migration*) и парцијалну миграцију (енг. *partially migration*) (Althani & Khaddaj, 2018, p. 155). То је основна подела која упућује на обим активности који ће се спровести над наслеђеним системом или неким његовим модулом.

Комплетна миграција подразумева да се систем поновно развија од почетка, укључујући све његове функционалности, интерфејсе, податке и програме. Идеја је да се нови систем има нову архитектуру или се извршава на новом хардверу или софтверској платформи. Пре него што се стари систем повуче из употребе и нови пусти у експлоатацију, потребно је податке пребацивати и спровести тестирање система. Cold Turkey је један од примера овог приступа у којем се цео систем изнова „преписује“ у другу платформу (M. L. Brodie & Stonebraker, 1993). Ова стратегија носи са собом велике ризике и временски је захтевна, при чему захтева стручно особље и познавање система. Доноси многобројна побољшања системе, готово у свим његовим сегментима, а уједно и смањује трошкове одржавања. Није лако испунити предуслове познавања система и квалификованог особља које је потребно да испуни задатак, па је због тога можда потребно применити реверзни инжењеринг и додатне обуке да би се задатак испунио у целости. Чак ни тада се не елиминишу веома велики ризици које ова стратегија доноси.

Инкрементална миграција се обавља у више малих инкременталних корака да би се циљ остварио у дугом року. Постоји много мања алокација ресурса у односу на комплетну миграцију, при чему уколико један корак откаже, он се може спровести

поново чиме се умањују ризици. Промене је потребно да буду укључене на техничком и организационом нивоу. Броди и Стоунбрејкер дефинишу 11 генеричких корака у спровођењу Chicken Little приступа, при чему је потребно да постоје делови који омогућава комуникацију и спровођење операције на различитим нивоима између наслеђеног система и новог система и називају се „пролази“ (енг. *gateways*) (Wu et al., 1997). Ова стратегија дуго траје и може бити поприлично скупа, при чему је потребно да се систем у потпуности познаје и захтева људе које имају вештине и о старом и новом систему. Њене предности су у томе што може да значајно смањи јаз између функционалности које нуди наслеђен систем и онога што само пословање захтева, поред тога што смањује трошкове одржавања.

Парцијална миграција укључује миграцију само неких делова система, тј. апликација у ново окружење. Постоје одређене потребе пословња због којег је потребно да се миграција спроведе, нпр. пребацивање апликације да ради на облаку (Althani & Khaddaj, 2018, p. 155). Са једне стране обезбеђује релативно брзо технолошко решење са мање ризика, док са друге може се применити само на одређеној групи апликација, уз трошкове додатног прилагођавања. На тај начин наслеђени систем је замењен готовим комерцијалним софтверским компонентама или пакетима (енг. *Commercial off-the-shelf, COTS*). То су пре свега ERP системи (уобичајено покривају финансије, производњу и неке друге функције), CRM системи (управљање односима са клијентима), SCM системи (управљање ланцима снабдевања) и тд. (Vesić, 2020). Такође једна од његових предности је то што се обезбеђује модуларност, али потенцијални недостатак су проблеми интеграције који могу настати на делу система који спаја остатак наслеђеног система и новог модула (апликације) (Војић & Velašević, 1999). Због тога претхода анализа потребних интерфејса је веома значајна да би се могло установити шта је то што новодобављена компонента је потребно да понуди окружењу у којем се интегрише, као и то што само окружење нуди компоненти.

4.5.4. Остали приступи

У овој секцији описаћемо два приступа за која сматрамо да су релевантни у теорији и пракси, иако их има много више.

4.5.4.1. Архитектуром вођена модернизација

Наслеђени системи могу да буду веома сложени састављени од наслеђеног хардвера, наслеђеног софтвера, разних комерцијалних софтверских компоненти, технологије средњег слоја и тд. и их чини веома тешким за разумевање, па самим тим и

модернизацију. Један од приступа који има за циљ да се избори са великом комплексношћу је архитектуром вођена модернизација (енг. *architecture-driven modernization, ADM*). Object Management Group (OMG) дефинише архитектуром вођену модернизацију као процес разумевања и еволуције постојећих софтверских средстава у циљу побољшања софтвера, измена, интероперабилности, рефакторисања, реструктурирања, поновне употребе, портовања, миграције, превођења из једног у други програмски језик, интеграције, сервисно-орјентисане архитектуре и миграције моделом вођеног развоја (OMG-ADMTF, n.d.). OMG је оформила ADM радну групу у циљу креирања спецификација и промоције индустријског консензуса у вези модернизације постојећих апликација.



Слика 29. пословна и ИТ архитектура (Ulrich & Newcomb, 2010)

Модернизација се посматра из два домена: пословног домена (енг. *business domain*) који садржи пословну архитектуру (енг. *business architecture*) и ИТ домена (енг. *IT domain*) који садржи архитектуру апликације и податка (енг. *application and data architecture*) и техничку архитектуру (енг. *technical architecture*). На слици 29. је приказано са леве стране постојеће (енг. *as-is*) решење, а са десне жељено (енг. *to-be*) решење (Ulrich & Newcomb, 2010, p. 17). Како пословна и ИТ архитектура еволуирају временом, постојеће решење се трансформише у жељено решење. Како пословна и ИТ архитектура еволуирају временом, постојеће решење се трансформише у жељено решење.

Свака трансформација има 3 елемента (Khusidman & Ulrich, 2007, p. 3):

1. откривање знања (енг. *knowledge discovery*) постојећег решења - спроводи се на неколико нивоа апстракције
2. дефинисање жељене архитектуре - креира се архитектура жељеног решења у које ће се постојеће трансформисати
3. спроводе се кораци трансформације да би се систем трансформисао из постојећег у жељено стање - спроводи се на неколико нивоа од физичког (нпр. превођење програмског језика) до апстрактнијег нивоа (нпр. мапирање пословних правила)

OMG дефинише спецификацију метамодела откривања знања (енг. *knowledge discovery metamodel*, KDM) путем којег се знање о постојећем систему презентује у форми модела. Поред поменутог модела, користи се и метамодел апстрактног синтаксног стабла (енг. *abstract syntax tree metamodel*, ASTM) који омогућава репрезентацију софтвера на ниском нивоу процедуралне логике, дефиниције података, као и композицију тока извршавања (Ulrich & Newcomb, 2010, p. 52). ADM приступ користи извлачење одозго на горе архитектуралних модела које након тога прати њихова поновна употреба одозго на доле у MDA процесу и сценаријима током модернизације наслеђених система. На поменути начин ADM користи принципе моделом вођеног развоја (енг. *model-driven architecture*, MDA) и мора да буде у сагласности на OMG MOF (енг. *Meta-Object Facility*) стандардом у MDA приступу (Newcomb, 2005). Као и у MDA и у ADM на највишем нивоу апстракције се налази налази са којим су сви концепти у сагласности. AMD је један од приступа који користи MDA принципе да би омогућио комплетну трансформацију на нивоу архитектуре. У литератури постоје и другачија разматрања која користе MDA принципе и у којима се предлаже и обухвата само један део наслеђеног система (MacDonald et al., 2005) или само један аспект софтвера као што је интероперабилност (Vesić et al., 2016). Сви поменути приступи базирају се на моделима, њиховој трансформацији и интерпретацији.

4.5.4.2. Приступ базиран на микросервисима

Технолошки напредак оличен у технологијама виртуализације, бржем приступу глобалној мрежи - Интернету и спремности све већег броја предузећа да константно улажу у своју информатичку инфраструктуру довело је до појаве рачунарства у облаку (енг. *cloud computing*). Неке апликације, као што су у почетку биле апликације е-трговине на глобалном нивоу, нпр. Amazon.com, условиле су измене у нефункционалним захтевима, међу којима је скалабилност веома важна. Платформа облака, услед могућности да хоризонтално скалира се показала као погодна за омогућавање потребних

обима свих ресурса који су неопходни апликацији. За неке апликације скалабилност тренутно није важан захтев, али софтверска архитектура у виду микросервиса (енг. *microservices*) се може показати као веома занимљива за пројектанте и архитекте система, поготово са аспекта одржавања. У односу на традиционални приступ⁷, микросервиси услед својег мањег кода и јаке изолације компоненти, могу се показати као веома добар приступ у решавању проблема наслеђених систем (Knoche & Hasselbring, 2018). Уобичајено се користе са технологијама облака.

5. Студија случаја на моделу ЈКП „Београдски водовод и канализација“

5.1. Увод

5.1.1. Опис проблема

Процес старења софтвера је неминован и настаје због два разлога који су повезани међусобно. Први разлог се односи на потребу да се софтвер прилагоди

⁷ често називан монолитан, када се говори о микросервисима

променама које долазе из околине, а други се односи на деградирање структуре софтвера које је настало као последица учињених промена.

Из поменутих разлога, а у циљу да софтвер задовољи функционалности које пословање ставља пред њега, потребно је спроводити редовно одржавање софтвера. Одржавање софтвера, као део информационог система, се види као активност која настаје након фазе пуштања информационог система у експлоатацију. Софтвер како временом стари, процес одржавања има за циљ да његове деградације умањи, али то не може да спроведе у потпуности.

Временом, како процес одржавања тече и како постоји све већи јаз између онога што је потребно да систем обавља и онога што он стварно обавља, потребно је спровести активности које имају за циљ да овај све већи јаз смање. Скуп тих активности једним именом се назива модернизација и по својој природи је интензивнији процес од одржавања. Обе активности и одржавање и модернизација имају за циљ да врате способност еволуције систему, јер према Леману информациони системи и софтвер као неизоставни део њих имају потребу да се мењају са променом окружења у којем функционишу. Он то означава као класу Е система и они су уграђени у свет који моделују, па самим тим еволуирају као део њега. Након тога, поново се спроводи процес одржавања, који има за циљ да смањи та мала одступања. У једном тренутку јаз између онога што систем нуди и оно шта се од њега захтева је толико велик, да ни процес модернизације не може да га редукује, или начин на који је то потребно спровести није финансијски исплатив и тада је дошло веме да се систем повуче из употребе. Тада је крај његовог животног века, а отпочиње се са употребом новог система.

Проблем истраживања су наслеђени информациони системи као системи који се опиру и тешко прилагођавају променама, веома често састављени не само од наслеђеног софтвера, већ и од наслеђеног хардвера, као и наслеђених пословних процедура и пословних правила која су уграђена у њих. Они ограничавају пословање, јер спречавају једноставну промену или увођење нових пословних процеса, па тиме производе незадовољство међу менаџерима, који су са друге стране пак свесни да они имају вредна знања и подржавају кључне пословне процесе. Њихово одржавање је веома скупо, а процес њихове замене је ризичан. У пракси постоји велики број пропалих пројеката модернизације, чак и након значајних финансијских улагања. Тешко могу да користе најмодерније технологије, јер су развијени у старим технологијама и знањима која су некада „била у тренду“.

Веома често имају деградирану архитектуру система, да ли у форми ерозије архитектуре која указује на крхкост система или у форми архитектуралног скретања који указује на значајно одступање од иницијалне пројекте замисли што их чини веома проблематичним. Поред тога, веома тешко се интегришу са другим системима и постају део већег система, који свих корисницима треба да понуди потребне сервисе и информације.

У теорији информационих система и софтверског инжењерства наслеђени системи су презентовани преко многобројних дефиниција, где се указује на њихову инертност и опирање модификацијама и изменама (Stonebraker & Brodie, 1995), става да већина менаџера жели да их повуче из употребе, али не могу јер подржавају главне пословне операције које чине срж пословања предузећа (Sommerville, 2016). То су системи који обухватају: људе, експертизу, хардвер, податке, пословне процесе и приступе у одржавању и развоју софтвера (Brooke & Ramage, 2001, p. 12). Не могу да на адекватан начин испрате пословне потребе (Liu et al., 1998) и тд.

Значајна средства се улажу у њихово одржавање, где према јавно доступном истраживању агенције GAO, влада САД је планирала да потроши преко 80% уложених 90 милијарди долара у ИТ на одржавање 10 критичних наслеђених система федералних агенција (United States Government Accountability Office, 2019). Поред тога, значајна средства се улажу у пројекте модернизације, а пракса показује да постоји велики број пропалих пројеката модернизације, чак и ако цена инвестиције прелази милијарде долара (Ulrich & Newcomb, 2010).

Ситуација у Србији није изолована случај по питању овог феномена у односу на остатак света, јер се значајна средства троше на одржавање наслеђених информационих система, а постоје и неуспели покушаји да се они модернизују. Тако на примеру једног значајног привредног субјекта у граду Београду имамо случај обимног улагања у одржавање наслеђеног система и након тога куповине новог решења које није заживело, а при томе су лиценце за окружење у којем је развијен наслеђен систем истекле и нису се обновили, при чему софтвер и подаци нису у потпуности портовани на нову платформу, како указује медијска мрежа Антидот (Antidot, n.d.-a, n.d.-b). Пореска управа је још један пример који указује на немогућност наслеђених система да у потпуности испрате пословање и подрже измене путем којих би се увећала наплата пореза, како наводи један званичник за портал Б92 (B92, n.d.).

Значајна финансијска средства која се издвајају за одржавање наслеђених система, велики број различитих дефиниција наслеђених система у теорији, велика

инвестиције у модернизацију ових система и велики број пропалих пројеката модернизације и поред значајно инвестираних средстава у пракси, указују на:

- сложеност феномена наслеђених информационих система;
- њихово недовољно (изоловано) схватање у теорији;
- неуспелост пројеката модернизације као последица недовољног схватања и да путоказ ка успешној модернизацији јесте адекватно разумевање и посматрање овог сложеног феномена са свим његовим елементима.

Како постоје различита схватања наслеђених система услед многобројних дефиниција и не постоји јединствен став о томе шта су они, оставља се један недефинисан простор између теорије и праксе, и могућност да се одређеном дефиницијом изоставе неки важни елементи наслеђених система, а посматрање наслеђених система буде нецеловито. Дакле, постоји озбиљан ризик да наслеђени системи, као сложени социо-технички системи, буду изоловано посматрани и да значајно уложена средства у враћање њихове еволутивне способности не дају ефекте за које се очекује. Из поменутих разлога Алдерсон и Шах су за потребе свеобухватног посматрања појма наслеђених система, а на основу различитих и неконзистентних тумачења наведеног појма од стране експерата из праксе на конференцијама и семинарима, 1999. године дефинисали тачке гледишта или перспективе наслеђених информационих система: развојна, оперативна, организациона и стратешка (Alderson & Shah, 1999). Као потребу да истраже перцепцију о наслеђеним системима и њиховој модернизацији у индустрији аутори су спровели истраживање које је указало на значајне разлике у схватању наслеђених система у теорији и пракси (Khadka et al., 2014).

Повезивањем претходно поменутог извештаја и истраживања, као и података из праксе може се створити теоријски оквир који има за циљ да обезбеди боље и дубље разумевање наслеђених информационих система, што је први корак ка њиховој модернизацији. Такав оквир према мишљењу аутора рада би требало да омогући једно свеобухватно посматрање наслеђених система, а то поред тога што значи да се сагледају тренутни проблеми из различитих перспектива, уједно и значи да се обрати пажња на будуће проблеме који се могу антиципирати. Такав један приступ омогућава да се спроведе поступак модернизације који би имао за циљ да на дужи временски период усклади функционалности које систем нуди, као и оне које се стварно захтевају у пракси. На тај начин се добија да наслеђени информациони систем наставља да живи унутар новог система и да испуњава улогу коју има, при чему се штите значајно уложена

средства у његов развој, као и вичедеценијска знања која су инкорпорирана у њега, кроз интеракције људи који развијају систем и корисника тог система.

5.1.2. Опис случаја

Субјекат анализе, тј. случај (енг. *case*) који је одабран за ову студија случаја (енг. *case study*) је Јавно Комунално Предузеће „Београдски водовод и канализација“, скраћено ЈКП БВК. То је београдско јавно комунално предузеће које се бави производњом воде и услугама одвођења отпадних вода на територији града Београда. Визија, мисија и циљеви предузећа су дати на Web порталу ЈКП БВК (JKPBVK, n.d.).

Мисија предузећа је:

- Јасна оријентација ка крајњем кориснику кроз пружање квалитетне услуге дистрибуције воде која ће задовољити квалитетом, квантитетом и здравственом исправношћу.
- Одвођење отпадних вода према највишим стандардима заштите животне средине.
- Планиран и стабилан развој, уз сталну имплементацију иновативних идеја и уважавање еколошких принципа.

Визија предузећа је:

Савремено, успешно Предузеће и лидер у области водоснабдевања, одвођења и пречишћавања отпадних вода уз обавезу очувања животне средине и ширења свести о заштити животне средине.

Циљеви предузећа су:

- Одржив развој водоизворишта и примарних објеката за прераду и дистрибуцију воде
- Повећана поузданост снабдевања водом и одвођења отпадних вода кроз санацију, реконструкцију и модернизацију постојећих система
- Производња и дистрибуција воде са што мање губитака на мрежи
- Реализација Стратешког плана развоја система водоснабдевања и канализације насеља на широј територији града Београда до 2025. године
- Финансијски независно пословање предузећа на одржив начин и по тржишним принципима
- Смањење оперативних трошкова

- Стручно усавршавање и оспособљавање запослених у циљу пружања што квалитетније услуге корисницима.

Према програму пословања предузећа за 2020. годину ЈКП БВК, означавају се 3 главне групе ресурса (ЈКРВВК, 2019):

- људски ресурси
- инфраструктура
- информационо-комуникационе технологије

Предузеће има 2590 запослених, који су распоређени на обављању пословних активности по следећим целинама:

- водоводни систем
- канализациони систем
- развој, пројектовање и инвестиције
- пословно-техничка подршка
- економско-финансијски, кадровски и општи послови
- правни послови и послови набавке
- безбедност и квалитет

Инфраструктура је предузећа приказана у следећој табели.

Табела 4. инфраструктура ЈКП БВК, преузето (ЈКРВВК, 2019)

| Редни број | Назив елемента | ЈМ | количина |
|-----------------------------|---|-----|----------|
| Београдски водоводни систем | | | |
| 1. | Бунари (99 бунара са хоризонталним дренажама и 51 цевасти бунар) | ком | 150 |
| 2. | Постројења за пречишћавање (ПП Макиш и ПП Макиш 2, ПП Беле воде, ПП Баново брдо, ПП Бежанија, ПП Винча) | ком | 5 |
| 3. | Резервоари 38 резервоара + 2 прекидна комора + 2 водоторња (10 Р+ 1 ПК + 1 ВТ нису у функцији са погона ЦСР) | ком | 42 |
| 4. | Црпне станице 19 ЦС + 20 ХС + 2 Бустер (2 ЦС, 6 ХС и 2 бустера са погона ЦСР нису у функцији) | ком | 49 |
| 5. | Мерила протицаја | ком | 239 |
| 6. | Затварачи | ком | 22350 |
| 7. | Регулатори притиска | ком | 108 |

| | | | |
|---------------------------------|---|-----|---------|
| 8. | Хидранти | КОМ | 15546 |
| 9. | Водоводна мрежа | КМ | 3958 |
| | Сирова вода | КМ | 126 |
| | Чиста вода | КМ | 3832 |
| 10. | Број корисничких регистра Број регистара главних, локалних, процентуалних и паушалних корисника | КОМ | 220011 |
| 11. | Станице за дохлорисавање | КОМ | 10 |
| Београдски канализациони систем | | | |
| 1. | Ретензије (2 надземне и 1 подземна) | КОМ | 3 |
| 2. | Црпне станице | КОМ | 49 |
| 3. | Постројења за прераду отпадних вода | КОМ | 5 |
| 4. | Колектори | КМ | 447,9 |
| 5. | Цеви | КМ | 1682,77 |
| 6. | Прикључци | КОМ | 54929 |
| 7. | Сливници | КОМ | 35706 |

Процењени физички обим активности у 2019. години на нивоу предузећа је дат следећом табелом.

Табела 5. Процењени физички обим активности у 2019. години, преузето (ЈКРВК, 2019)

| Активност | Јединица мере | План | Процена | Индекс |
|------------------------------------|------------------|-------------|-------------|--------|
| | | 2019. | 2019. | 4/3 |
| 1 | 2 | 3 | 4 | 5 |
| Производња воде | м ³ | 191.400.000 | 191.200.000 | 99,9 |
| Сопствена потрошња | м ³ | 15.300.000 | 18.100.000 | 118,3 |
| Потис у систем | м ³ | 176.100.000 | 173.900.000 | 98,8 |
| Електрична енергија | kwh | 167.000.000 | 162.500.000 | 97,3 |
| Фактурисана вода | м ³ | 116.758.269 | 116.758.269 | 100,0 |
| Фактурисана канализација | м ³ | 92.284.223 | 91.150.143 | 98,8 |
| Чишћење сливника и сливничких веза | КОМ | 73.000 | 72.000 | 98,6 |
| Чишћење цевне мреже | М | 150.000 | 162.000 | 108,0 |
| Чишћење колектора | м ³ | 4.800 | 6.700 | 139,6 |

Сектор за информационо комуникационе технологије спроводи разне активности које се тичу развоја, одржавања и интеграције постојећег информационог система и комплетне информационо-комуникационе инфраструктуре. Према програму пословања предузећа за 2020. годину ЈКП БВК (ЈКРВК, 2019):

- у систему постоји 1200 рачунара, 400 мрежних и 50 локалних штампача

- постоји 35 физичких и 80 виртуелних сервера
- број физичких локација је 21 која су умрежене у локалну мрежу
- везе се остварују највише оптичким, па затим бежичним путем, а употребљава се и технологија frame relay
- постоји велики број сервера и неколико кластера сервера: сервери за уређење и контролу мреже по моделу активног директоријума, сервери за електронску пошту, сервери који се користе за базе података, сервери за географски информациони систем, терминал сервери, сервери за пословно извештавање путем IBM Cognos пакета, Web сервери, сервери за Backup података, сервери за антивирусни софтвер и тд.
- трансакционе базе података садрже око 100 GB података
- оперативни системи у употреби су: Windows, Solaris и Linux
- технологије виртуелизације у употреби су: Hyper-V и Vmware
- постоји локација и план опоравка у случају катастрофе (енг. *disaster recovery*)

ЈКП БВК има наслеђен пословни информациони систем стар око 20 година, написан у програмском језику Progress ABL, познатијим под именом Progress 4GL и базом података Progress. Поменути наслеђени систем обухвата следеће програмске модуле и апликације (ЈКРВВК, 2019):

- Главна књига
- Аналитике купаца и добављача
- Аналитика камата и тужби
- Магацинско и материјално књиговодство
- Фактурисање воде и услуга
- Евиденција водомера са заменама,
- Обрачун личних доходака
- Евиденција потрошње по мобилним телефонима
- Web апликација за преглед личних доходака
- Благајна
- Шалтери (пријем докумената и рекламација, водомера на баждарење)
- Ликвидатура
- Основна средства
- Погонско књиговодство

- Кадровска евиденција
- Ангажовање радника изван БВК
- Лична заштитна средства
- Модул за картице за превоз
- Модул за картице за гориво
- Пакет за народну одбрану
- Планирање и администрирање набавки
- Праћење реализације јавних набавки
- Праћење реализације уговора
- Опомене са радним налозима за искључење
- Праћење наплате са формирањем тужби, протокола плаћања и налога за искључења
- Модул за Службу заступања
- Пријава кварова и евиденција радова на водоводној и канализационој мрежи, контакт центар, командно-контролни центар, диспечерски центар
- Мобилна апликација за читавање водомера
- Web апликација - портал за потрошача
- Web апликација - телефонски именик
- Апликација за одмаралиште у Врњачкој Бањи
- Апликација за пријем нових захтева и захтева за измену у постојећим апликацијама
- Апликација за пријаву кварова на рачунарској и телекомуникационој опреми као и системском софтверу

Све наведене чињенице упућују да је ЈКП БВК један сложен систем, којег одликује разноврсност пословних процеса и потреба за различитим врстама информационих система који је потребно да подрже поменуто разноврсност.

5.1.3. Веза елемената случаја са проблемом и продубљивање проблема

Наслеђен информациони систем ЈКП БВК у пракси има доста проблема који се могу сагледати из угла програмера, пројекатаната и архитеката система, као и из угла корисника система, и из угла менаџера у разним одељењима, службама и секторима које користе наслеђен систем, као и из угла топ менаџмента који се бори за остваривање визије, мисије и циљева предузећа.

Посматрањем ових проблема на случају ЈКП БВК и повезивањем са перспективама наслеђених система и перспективом (погледом у будућност) који су дефинисане у склопу одређених истраживања, ствара се један оквир који омогућава да се наслеђених системи посматрају свеобухватно и да се боље сагледају, боље разумеју како у садашњости, тако и у будућности. На тај начин се спречава њихово изоловано посматрање, као и могућности да се занемаре одређени њихови аспекти, што би за последицу имало да се отворе оне алтернативе у стратегијама модернизације које нису подобне за конкретан случај. На тај начин потенцијална модернизација наслеђеног система ЈКП БВК би била само једна у низу неуспелих модернизација са значајно инвестираним средствима.

Поред тога, изучавањем перспектива наслеђених система на примеру ЈКП БВК може да укаже на неке проблеме који се раније нису јавили у пракси или теорији наслеђених информационих система, па се на тај начин обогаћује и практични и теоријски фонд (научни фонд), поготово уколико се одређене појаве објасне са већ познатих теоријских позиција, али из друге научне области.

5.1.4. Значај проблема истраживања, погодност студије случаја као методе истраживања и погодност ЈКП БВК као субјекта анализе

Сагледавање наслеђених система из развојне, оперативне, организационе и стратешке перспективе омогућава бољи увид у проблеме наслеђеног информационог система, па самим тим и олакшава дизајнирање процеса његове модернизације јер се избегава изоловано посматрање феномена. Сагледавање проблема наслеђених система У ЈКП БВК и одабира одговарајуће модернизационе стратегије може да помогне и осталим предузећима из истог домена пословања (свим осталим водоводима) које имају сличне наслеђене системе. Такође како се ради о јавном комуналном предузећу, претпоставка је да би и друга јавно комунална предузећа која имају сличну или мању сложеност пословања, сличан или мање сличан систем могли да искористе начин посматрања наслеђених система и тиме подпомогну процесе модернизације својих наслеђених система.

Студија случаја је метода истраживања која се може користити и у експлораторним и у дескриптивним и у експланаторним истраживањима, а погодна је јер може да комбинује квалитативне и квалитативне методе истраживања. Она има за циљ да посматрањем одређене особе, места, догађаја или феномена утврди кључне теме и

результате које помажу да се предвиде будући трендови, осветле претходно скривени проблеми а који се могу применити у пракси и / или да се обезбеди разумевање истраживачког проблема са већом јасноћом. Она може да обезбеди нове начине за разумевање проблема истраживања који нису раније били изложени у пређашњим истраживањима и да открије нове и важне импликације у пракси.

ЈКП БВК је погодан као субјекат анализе (случај) из неколико разлога. Пре свега, предузеће има наслеђен информациони систем стар око две деценије. Такође предузеће се одликује разноврсним пословним процесима и чија сложеност је значајно већа него у другим мањим системима. Постоји велики број проблема у вези са наслеђеним системима који се јављају у склопу свих различитих перспектива на случају ЈКП БВК. Поред тога аутор истраживања скоро целу деценију ради у поменутом предузећу, на позицији пројектанта пословног информационог система и кроз рад и искуство је спознао проблеме и специфичности тог наслеђеног система.

5.1.5. Позадина истраживања и резиме претходних истраживања

У теорији постоји много различитих дефиниција наслеђених система, као што је раније поменуто, а већина њих сагледава наслеђене информационе системе са једног или највише два аспекта, најчешће технолошког и пословног, при чему се перспективе које су везане за угао посматрања од стране корисника система, као и угао посматрања од стране топ менаџера, се често игноришу и тиме се не сагледава феномен у потпуности.

Алдерсон и Шах увиђају да постоје значајне разлике у посматрању наслеђених система у теорији и пракси и дају идеју о посматрању наслеђених система кроз 4 перспективе које су повезане међусобно: развојна, оперативна, организациона и стратешка (Alderson & Shah, 1999).

Такође другачије посматрање на наслеђене системе у односу на теорију дају Кадка и остали, чије се истраживање заснива на резултатима добијеним из интервјуа, конструисањем одређених теорија путем Grounded Theory методе истраживања и након тога евалуације теорије на путем упитника (Khadka et al., 2014). Истраживање указује да корисници система, веома цене своје наслеђене системе, а изазови са којима се они суочавају нису само техничке природе како се често наглађава у теорији, већ и пословне и организационе природе.

5.1.6. Начин на који ће студија случаја унапредити ниво знања или пружити нове начине разумевања

Случај ЈКП БВК ће бити користан јер ће моћи да потврди до сада утврђене чињенице, или да укаже на неке нове, као и неке нове проблеме који се јављају у пракси.

Такође омогућиће да се интегралним посматрањем перспектива као и проблема наслеђених система који се јављају у склопу њих оствари боље сагледавање наслеђеног система, у односу на теорију која посматра овај феномен изоловано.

5.2. Преглед разматрања

5.2.1. Преглед постојећих разматрања

У литератури постоји велики број различитих дефиниција, описа, тумачења и становишта о томе шта су наслеђени информациони системи или наслеђени системи током последње три деценије. Оне веома варијају између себе и не постоји једно јединствено мишљење шта овај сложен феномен обухвата.

Највећи број аутора указује на неке њихове особине, пре свега описује да су у питању проблематични системи, дуго у употреби, тешки за промену, да имају одређене недостатке који се могу огледати у већем броју ствари, да су у питању застарела технолошка решења која не могу да прате промене у окружењу, тешки и скупи за одржавање и тд. Неки од њих указују и на вредности које они имају у подржавању критичних пословних процеса.

Насиф и Мичусон 1993. говоре о томе да су то системи које је веома скупо и тешко одржавати, а при томе не могу да се испрате одређени захтеви пословања (Nassif & Mitchusson, 1993). Брук их види као системе у којима постоји јаз између могућности система и захтева пословања (Brooke, 1994). Бенет описује да је то информациони систем који се може разумети само кроз изворни код, где је документација оскудна, а структура не постоји (Bennett, 1995). Броди и Стоунбрејкер истичу да се ради о било којем информационом систему који се значајно опире модификацијама и еволуцији да би достигао нове и константно променљиве пословне захтеве (Stonebraker & Brodie, 1995). Адолф истиче недостатке и да су то стари системи који имају проблем у имплементацији нових програмских функционалности који корисници желе, јер постоје ограничене перформансе (Adolph, 1996). Група аутора описује да они уобичајено раде на застарелом хардверу који је скуп за одржавање, као и на софтверу који је тежак за разумевање и скуп

за одржавање, а при томе могућност сарадње са другим системима је значајно смањена (Jesus Bisbal et al., 1997). Робсон их види као информационе системе настале као последица претходне аквизиције која не испуњава тренутне потребе (Robson, 1997). Синг их перципира као системе који једном када се инсталирају и ставе у продукцију, више се не мењају (V. Singh, 1997).

Ким има другачија становишта у односу на ауторе који указују на њихове недостатке и устиче да њихова вредност лежи у акумулацији пословних правила, политика, експертизе и know-how-a (Kim, 1997). Тејлор описује одређене специфичности у вези са људима који раде на његовом развоју, а то је да су они системи креирани од стране људи са скупом вештине које су дефицитарне на тржишту или недоступне у организацији и њеном окружењу (Taylor et al., 1997).

Друга група аутора указује на њихову особину да су питању инертни сложени системи настали услед широко распрострањене употребе рачунарске технологије током неколико деценије, који се опира даљим променама и еволуцији (Wu et al., 1997). Лиу и остали их такође виде као системе са одређених недостацима, зато што не могу да прате пословне потребе које се мењају јер систем функционише у окружењу које се континуално мења (Liu et al., 1998). Ворен их такође перципира преко недостатака и повезују их са застарелешћу, високим трошковима одржавања, лошом документацијом и недостатком техничке подршке (Warren, 1999). Аутори указују да не могу да подрже нове пословне процесе (Laudon & Laudon, 1999). Брук их види као рачунарске системе, који се налази у одређеном организационом окружењу који више не задовољава захтеве тог окружења (Brooke, 2002). Улрих указује њихов утицај на пословање, јер се ради о застарелом хардверу, софтверу и архитектури података која спречава организацију да у потпуности искористи рачунаре и вредност који они доносе организацији, њеним купцима и партнерима (Ulrich, 2002). Аутор објашњава да наслеђеност информационог система лежи у јазу између потреба пословања и могућности технологије (Brooke & Ramage, 2001). Аутори их посматрају са становишта система за управљање базама података и према њима то су системи који развијени током последње три деценије, чији су наслеђени програми и логика изграђени око наслеђене структуре података која је навигациона или хијерархијска кроз старе технологије база података као што су IDMS, CODASYL, NETWORK и тд. (Rahgozar & Oroumchian, 2002). Панг је схватања да су то системи који су инсталирани на мејнфрејму, а у чије се одржавање значајно инвестира (Khosrow-Pour, 2007). Група аутора такође истиче неке њихове недостатке указујући да је цена замене ових система је веома велика. али они и даље настаљају да се користе иако

нису компатибилни са другим системима (Aversano & Tortorella, 2004). Аутор Маки указује на њихову временску димензију, јер то су системи настали у прошлости када се употребљавала другачија технологија (Khosrow-Pour, 2007).

Сомервил их види као социо-техничке рачунарски засноване системе развијене у прошлости најчешће употребом старе или застареле технологије, које укључују наслеђен хардвер, наслеђен софтвер, али и наслеђене процесе и процедуре, тј. старе начине рађења ствари, који су тешки за промену због тога што се ослањају на наслеђен софтвер (Sommerville, 2006). Аутор их описује преко старих програмских језика као што је COBOL и истиче да нису интероперабилни са другим апликацијама (Colosimo et al., 2007). Група аутора указује и на њихову вредност јер су мишљења да су то старији системи који имају велику вредност за организације које их користе, али креирају проблеме у одржавању како старе (Stehle et al., 2008). Аутори их такође види преко недостатака, јер су они стари системи тешки за разумевање, одржавање, проширивање и интегрисање, који могу да се извршавају на застарелом хардверу (Vestues & Knut, 2019).

Група аутора их описује на начин да су у питању системи који није дефинисани само старошћу ИТ система (нпр. 20 година), јер постоји много система који су пројектовани за континуалну надоградњу, већ у појмовима који се фокусирају на елементе као што је подршка, ризик и агилност укључујући доступност подршке за хардвер и софтвер и могућност да се интерно или екстерно ангажује особље, опрема или техничка подршка (National Association of State Chief Information Officers (NASCIO), 2008). Новаковски и остали истичу њену способност еволуције, тако што описује да је то апликација која је у продукцији годинама и константно еволуира да би се прилагодила пословним променама које су последица промене потреба тржишта, као и нових технологија које пружају нове пословне могућности (Nowakowski et al., 2012). Консултантска кућа истиче да они успешно обављају свакодневне операције, али то је у основи информациони система базиран на застарелим технологијама (Gartner, n.d.). Група аутора такође их перципира као старе системе, па је према њима опсежан и сложен софтверски систем који ради већ дуже време (више од 20 година) (Meng et al., 2013). Група аутора истиче неке њихове недостатке према којима су то постојећи велики софтверски системи који никада нису изграђени да би могли да испуне тренутне пословне захтеве услед лошег кодирања, пројектних структура, логике и документације (Srinivas et al., 2016). Сличан став имају и Кроти и Хорокс, према којима су то системи који су критични за пословање, при томе могу да су стари, написани на застарелим програмским језицима, имају мало или готово да немају документације, имају неадекватно

управљање подацима, деградирану структуру, могућност за ограничену подршку, растуће трошкове одржавања и недостатак архитектуре потребан за еволуцију (Crotty & Horrocks, 2017). Сомервил их види као системе имплементираних у застарелој технологији или превазиђеном програмском језику, а при томе подржавају критичне пословне операције (Sommerville, 2016).

Значајно мањи број аутора има другачије ставове у односу на преовладавајуће ставове по којима су то претежно проблематични системи који су дуго у употреби, где се често опажају њихови недостаци, а још мањи број аутора указује на вредности које они пружају.

Тако нпр. група аутора их посматра мало шире и указују да су то системи сачињени од техничких компоненти и социјалних фактора (софтвер, људи, вештине, пословни процеси) који више не испуњавају захтеве пословног окружња (Brooke & Ramage, 2001). Посматрањем њиховог значаја Тромп и Хофман дају мало другачије тумачење на које се друга група аутора надовезује, а то је да су то су системи дизајнирани, имплементирани и инсталирани у радикално другачијем окружењу од текуће ИТ стратегије (Tromp & Hoffman, 2003), а при томе не подржавају тренутну ИТ стратегију, како други аутора проширује појмом из угла еволуције система (Mittleton-Kelly, 2006).

Имајући у виду све поменуте дефиниције, мишљења, дескрипције и ставове можемо уочити велики број варијација између њих, при чему ни једно становиште није свеобухватно, већ указује на појединачна појављивања сложене појаве. Тако посматран феномен наслеђених система може да доведе до изолованог тумачења поменутог појма и до несагледавања свих његових видова, као и разних нивоа. Из поменутих разлога други аутори су покушали да дају другачији поглед на овај феномен.

Аутори Алдерсон и Шах уводе елемент праксе у досадашње преовладавајуће теоријске интерпретације наслеђених информационих система које они опажају на конференцијама и семинарима (Alderson & Shah, 1999). На основу тога они развијају начин посматрања наслеђених информационих система из различитих валидних гледишта или перспектива, које обухватају 4 перспективе за које они верују да су довољне да отклоне конфузију у посматрању сложеног феномена:

- развојна (енг. *developmental*)
- оперативна (енг. *operational*)
- организациона (енг. *organizational*)

- стратешка (енг. *strategic*)

Аналитичари система, пројектанти и архитекте система, програмери, тестери, инжењери који су задужени да осигурају квалитет имају специфичан поглед на наслеђени информациони систем, тзв. развојну перспективу. Зато се јављају и разне тумачења наслеђених система као нпр. „било који систем који је напустио развој“ или „чим је први захтев узет“ које можда делују чудно, али они желе да нагласе своју улогу и њихове проблеме које произилазе из многобројних промена које је потребно да спроведу.

Оперативна перспектива указује да је потребно обезбедити адекватан сервис, који корисници морају да препознају, било да се он састоји у ефикасном спровођењу операција или сарадњи са другим системима. Поред тога је потребно обезбедити и одређене нефункционалне захтеве, као што је нпр. безбедност.

Менаџерима је важно како наслеђен систем утиче на пословање и да ли подржава и у којој мери пословне процесе и то се назива организациона перспектива. Они уобичајено имају две супротне перцепције о томе. Са једне стране виде предности јер подржавају критичне операције без којих предузеће не може да обавља своју визију, мисију и циљеве, док са друге стране уочавају мане када је потребно извршити одређене измене или реинжењеринг пословних процеса, јер се тада наслеђени информациони систем показује као неко ко спречава поменуте измене, тј. они их перципирају као „кочничаре“.

Топ менаџмент настоји да учини све да би се мисија, визија и циљеви предузећа испунили, па у том смислу доноси одлуке о предузимању одређених стратешких активности које то треба да обезбеде. Када је на тржишту јака конкуренција, усвајање нових пословних модела је веома битно, па стратешка перспектива разматра трошкове изгубљених могућности (опортунитета) када наслеђен систем не може да подржи жељене пословне могућности, штавише онемогућава организацију да заради од те пословне могућности за све оне пословна сценарија проистекла из стратегије која наслеђен систем не може да их подржи.

Разматрајући реинжењеринг наслеђених система Бојић и Велашевић (Војић & Velašević, 1999) указују на важност посматрања наслеђених система из различитих перспектива. Они истичу три аспекта (перспективе) веома сличне, али и донекле различита аспекта у односу на Алдерсона и Шах, а то су: аспект произвођача софтвера, аспект технолошког развоја и аспект пословања. Аспект произвођача софтвера указује да се ради о сложеним и скупим системима развојеним у застарелим програмским језицима

(као пример наводе асемблер или Cobol), при чему слабо документовани, тешки за одржавање јер људи који су развијали систем су напустили предузеће, при чему постоје многобројне измене које су размекшале архитектуру. Аспект технолошког развоја указује да уколико систем користи застареле технологије иако је потпуно функционалан може бити наслеђен. Промене у домену пословања онемогућавају систему да се прилагоди и тада је он наслеђен са аспекта пословања

Аутор Лајт указује да је потребно дати алтернативну теорију која указује на више димензија и признаје динамизам наслеђених система (Light, 2003). У том смислу он презентује теорију наслеђених информационих система која се састоји из концепата временских ефеката (енг. *temporal effects*), интерпретација (енг. *interpretations*) и карактеристика (енг. *characteristics*). Карактеристике се односе на природу наслеђеног информационог система, које он класификује са аспекта дуговечности (енг. *longevity*), функционалности (енг. *functionality*), перцепције технологије (енг. *perceptions of technology*) и структуре (енг. *structure*). Све ове категорије зависе од интерпретације и од конкретне ситуације наслеђеног система. Интерпретације се баве тиме како се карактеристике наслеђеног информационог система доживљавају, чиме се наглашава његова субјективна и интерпретативна природа. На карактеристике које чине наслеђени систем може се гледати различито. Временски ефекти указују да наслеђени информациони систем зависе од времена, јер ће се његове карактеристике временом изменити услед измене програмског кода, проширења функционалности и промене запослених, па ће се тиме покренути другачија интерпретација карактеристика.

Услед тога што у академској заједници преовлађују схватања по којима су наслеђени системи претежно технолошке природе, а да истовремено доприносе пословним операцијама предузећа, група аутора је одлучила да поменуте ставове испита ставове у пракси, као и ставове у вези са модернизацијом поменутих система (Khadka et al., 2014). Спроведено је истраживање, које је прво користило интервју над 26 учесника да испита ставове у вези наслеђеног система и модернизације, а затим је налазе интервјуа потврдила са анкетом од 198 учесника. Резултати су показали да иако су системи наслеђени ону су веома вредни њихови корисници их веома цене, а поред техничких постоје пословни и организациони аспекти који су веома важни. На сличне ставове у вези тога да наслеђени системи у традиционалној литератури се преодминантно фокусирају на њихов технички аспект, док постоји значајне везе које указују у на „нетехничке“ аспекте који се друштвених и организационих карактеристика указује и претходно наведени Лајт.

Резултати до којих су Кадка и остали дошли у свој истраживању су следећи (Khadka et al., 2014):

- око 50% испитаника се слаже да наслеђени систем може да се утврди уколико је информациони систем написан у застарелом програмском језику
- већина њих су део главног (енг. *core*) пословања са животним веком око 20 до 30 година
- користи које они доносе су: чињеница да су критични за пословање, да су написани у доказаној технологији, да су поуздани и перформансе
- мотивација за њихову модернизацију укључује: високе трошкове одржавања, недостатак знања, могућност да се остане агилан у односу на промене и склоност кваровима
- изазови модернизације су: сложена архитектура система, тежина за извлачење и давање приоритета у пословној логици, недостатак знања, миграција података, отпор од стране организације, временско ограничење у којем је потребно завршити пројекат модернизације, означавање меких фактора модернизације

Сва презентована разматрања послужиће се успостављање новог (проширеног) теоријског оквира, који има за циљ да продуби разумевање наслеђених система.

5.2.2. Успостављање новог (проширеног) теоријског оквира

За сада у прегледу постојеће литературе велика већина радова у теорији посматра наслеђене системе најчешће са технолошког аспекта, а при томе указује и на њихову вредност у пословању. Радови и истраживања која обухватају праксу указују и на друге димензије сем технолошке или пословне, при чему постоје одређена неслагања у начин на који се гледа на одређене карактеристике наслеђених система.

У покушају да се проблем сагледа у целости и спречи изоловано посматрање феномена наслеђених система, користиће се проширени теоријски оквир, који се у својој основи заснива на раду Алдерсона и Шах, о томе да наслеђене системе је потребно посматрати кроз њихове 4 перспективе: развојну, оперативну, организациону и стратешку.

Новина у односу на Алдерсона и Шах, јесте да ће се проблеми наслеђених система посматрати тако да припадају једној од поменутих перспектива. На тај начин сваки проблем не само што је сагледан као део наслеђеног система, он је уједно и класификован у једну од перспектива, што омогућава да се поменути проблем накнадно боље истражи, јер је могуће организовати истраживања која имају за циљ да испитају одређену групу запослених у вези конкретних проблема, за које можда друга група запослених, који су усмерени на неку другу перспективу, не могу да знају. На пример, уколико желимо да утврдимо да ли програмски језик који је коришћен за развој наслеђеног информационог система представља проблем за наслеђени систем јер нпр. не поседује одређене карактеристике модерних језика, усмерићемо се на развојну перспективу, а самим тим на програмере, пројектанте и архитекте који у свом послу имају додир са поменутих проблемом да би добили исцрпне информације о томе.

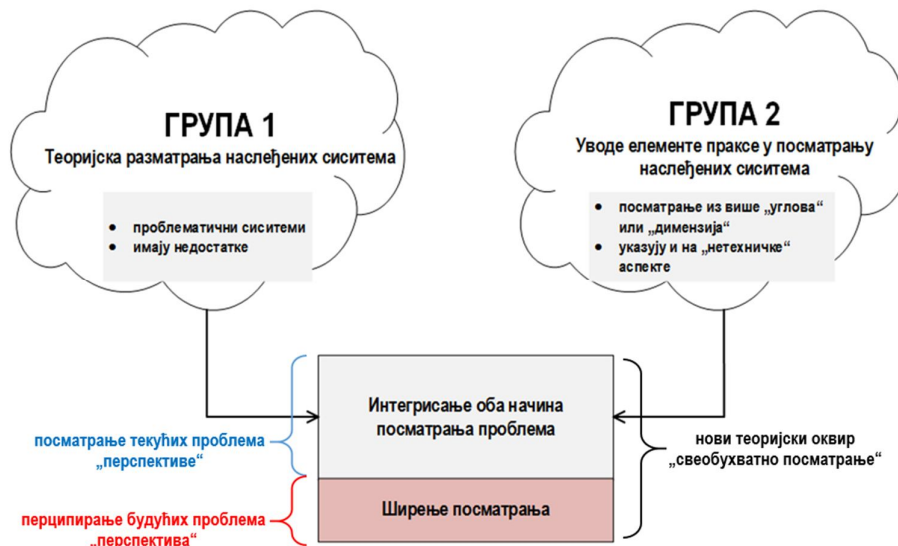
Предложен оквир такође има за циљ да интегрише до сада изложене дефиниције, описе и перцепције о наслеђеним системима који постоје у теорији и практична сазнања која су настала кроз рад Алдерсона и Шах, и истраживања Кадке.

У односу на рад Алдерсона и Шах, као и на истраживања Кадке, предложени теоријски оквир ће унутар себе инкорпорирати перспективу (поглед у будућност) наслеђеног информационог система као део стратешке перспективе. То се спроводи јер је аутор истраживачког рада уочио да сви приступи посматрају проблем наслеђених система у садашњости и на основу тога износе своја запажања. Улога стратешке перспективе према мишљењу аутора истраживања јесте и да једним делом антиципира промене у будућности, а поготово оне које могу да утичу на промену пословног модела и измену пословних процеса у домену пословања, што је важно за остваривање мисије, визије и циљева предузећа.

Уколико је потребно вршити модернизацију, која је захтеван посао и са аспекта финансија, менаџмента људских ресурса али и времена које је потребно посветити тим активностима, уз велики ризик који носи, антиципирање будућих потреба система може бити корисно и исплатљивије уколико у старту у стратегију модернизације се уграде будућа побољшања система, ако је то могуће и финансијски оправдано. То се може показати боље него спроводити модернизацију без поменутих побољшања, па за кратко време након тога поново спроводити опсежан процес модернизације у циљу да се омогуће поменута побољшања. Описани поступак настанка новог (проширеног) теоријског оквира је дат на слици 30.

Нови (проширени) теоријски оквир се састоји у томе да наслеђени информациони система има 4 перспективе, као што је раније речено:

1. развојна перспектива
2. оперативна перспектива
3. организациона перспектива
4. стратешка перспектива



Слика 30. настанак новог (проширеног) теоријског оквира

Развојна перспектива обухвата следеће елементе, тј. проблеме наслеђених информационих система:

- документација није ажурна или не постоји
- особље које је радило на развоју система је напустило предузеће
- новозапослени немају довољно знања и вештина потребних за одржавање система
- нарушена почетна архитектурална замисао
- редуванса

Оперативна перспектива обухвата следеће елементе, тј. проблеме наслеђених информационих система:

- не постоји подршка произвођача хардвера и/или софтвера
- проблеми интеграције система

Организациона перспектива обухвата следеће елементе, тј. проблеме наслеђених информационих система:

- систем ограничава пословање
- систем је вредан због подршке главним пословним активностима

Стратешка перспектива обухвата следеће елементе, тј. проблеме наслеђених информационих система:

- систем ограничава нову или измењену стратегију
- систем ограничава будућу стратегију

Нови (проширени) теоријски оквир је приказан на слици 31.



Слика 31. нови (проширени) теоријски оквир

Односи између ових перспектива су такви да уколико дође до измене стратегије предузећа услед константно променљивог окружења у којем функционише наслеђен систем, као модел реалног система, те промене ће се спустити на организациони ниво у

виду нпр. измене пословних процеса, затим ће се спровести на оперативном нивоу тако да се обезбеде сви потребни сервиси, а при томе на развојном нивоу морају се изменити системи тако да подржавају све друге перспективе. Дакле, промене које спроводе унутар једне перспективе, пропагирају се на промене које се морају спровести унутар друге перспективе и то у смеру од врха ка дну, тј. од стратешке до развојне. На тај начин се измена модела пословања и стратегија предузећа спроводи до најнижег нивоа, где је потребно промене инкорпорирати у постојећи наслеђени систем.

Предложени теоријски оквир прво ће бити приказан и анализиран на случају ЈКП БВК, чиме ће се утврдити перспективе наслеђеног информационог система, као и проблеми повезани са поменутиим перспективама. Током тог поступка можда ће се показати и потреба да се након тога спроведу нова истраживања уколико се за истим укаже потреба, а са циљем да утврде и осветле неке неочекиване налазе или да се утврде дубље релације између елемената проблема.

5.3. Метод

Сада ћемо да се осврнути на наслеђени систем ЈКП БВК и на њему посматрати како се реализује предложени теоријски оквир. Посматраћемо сваку од перспектива које су део предложеног теоријског оквира: развојну, оперативну, организациону и стратешку. Унутар сваке од ових стратегија постоје проблеми који су везани за ту перспективу и на случају ЈКП БВК ће бити презентовани. На тај начин створиће се једна детаљна слика о наслеђеном информационом систему ЈКП БВК. Све наведене перспективе је потребно да детаљно опишу проблеме које наслеђени информациони систем ЈКП БВК има у садашњем тренутку.

Након тога ће бити дата анализа ланца вредности ЈКП БВК. Поменута анализа ће указати на активности у ланцу вредности које су од кључног значаја за пословање ЈКП БВК. Такође, извршиће се анализа технолошких решења која могу да унапреде активности које су од кључног значаја за ЈКП БВК. Размотриће се функционисање наслеђеног информационог система ЈКП БВК и нових ИТ решења која имају за циљ да унапреде активности у ланцу вредности које су од кључног значаја за ЈКП БВК, као и проблеми које је потребно да наслеђени информациони систем ЈКП БВК превазиђе да би омогућио интегрисање постојећег наслеђеног система и нових ИТ решења.

Након тога, обједињавајући сазнања из посматрања случаја ЈКП БВК и анализе ланца вредности ЈКП БВК, предложени теоријски оквир ће се евалуирати у наредним поглављима на предузећима који имају исту доменску проблематику као и ЈКП БВК, тј.

водоводним предузећима. Употребљивост наслеђеног система ЈКП БВК ће се такође накнадно испитати.

На тај начин све перспективе (начини посматрања), које указују на проблеме наслеђеног информационог система ЈКП БВК у садашњости, заједно са перспективом (погледом у будућност), која указује на проблеме рада са технологијама које имају за циљ да унапреде кључне активности унутар ланца вредности, ће се евалуирати у пракси и утврдити да ли су релевантне за предмет истраживања.

Након тога, ће се дискутовати о стратегији модернизације наслеђеног информационог система ЈКП БВК, у односу на његове специфичности.

5.3.1. Посматрање наслеђеног система на случају ЈКП БВК

5.3.1.1. Развојна перспектива

У циљу подробнијег објашњавања развојне перспективне, потребно је у потпуности сагледати развојно окружење у којем је развијен наслеђен систем ЈКП БВК. Да би посматрање било адекватно и у вези са постојећим теоријским фондом, потребно је да сагледамо генерације програмских језика и да утврдимо место програмског језика у којем је развијен наслеђени информациони систем, као и предности и недостатке програмског језика Progress ABL, познатијег под именом Progress 4GL⁸ и базом податка Progress. Опис је потребан за боље разумевање развојне перспективе.

Најједноставније речено програмски језик (енг. *programming language*) је језик који се користи да се напише скуп инструкција које говоре рачунару шта да ради. Потребно је да буду прецизни па због тога имају своју форму, која се још назива и синтакса (енг. *syntax*) и значење, тј. семантику (енг. *semantics*). Синтакса програмских језика је форма његових израза, наредби и програмских јединица, док је семантика значање тих израза, наредби и програмских јединица (Sebesta, 2016).

Уобичајена подела програмских језика јесте на програмске језике нижег нивоа (енг. *low-level programming languages*) и програмске језике вишег нивоа (енг. *high-level programming languages*). Програмски језици нижег нивоа, се често називају и машински зависни програмски језици јер су зависни од машине, тј. рачунара на којем се извршавају, док програмски језици вишег нивоа називају се и програмским језицима независним од

⁸ у наставку ће се користити назив Progress 4GL услед јасне асоцијације са програмским језицима четврте генерације

рачунара на којем се извршавају. Нижи програмски језици су „ближи машини“, а виши програмски језици су „ближи човеку“.

Друга честа подела програмских језика јесте на генерације програмских језика. Постоје прва, друга, трећа и четврта генерација програмских језика и у односу на претходну поделу прва и друга генерација програмских језика припадају програмским језицима нижег нивоа, док трећа и четврта генерација програмских језика припада вишим програмским језицима. Генерације програмских језика се могу посматрати као померање нивоа апстракције од нивоа ближих машини, ка нивоима који су ближи човеку.

Програмски језици прве генерације називају се машинским језицима или машинским кодом (енг. *machine language*) и њихова одлика јесте да их машина директно разуме без интерпретације. Програми су написани у бинарној нотацији и везани су за машину на којој се извршавају. Са једне стране програми се извршавају веома брзо јер су оптимални и написани за процесор за који су намењени, а да друге стране су веома тешки за разумевање јер се свака инструкција састоји од низа јединица и нула, при чему програмер у потпуности треба да познаје како функционише рачунар (Byte-notes, n.d.). Инструкције су се задавале померањем прекидача, бушеним картицама или на неки други начин.

Програмски језици друге генерације настали су да програмирање учине једноставнијим у односу на претходну генерацију програмских језика. Називају се асемблерским језицима (енг. *assembly languages*) и такође захтевају детаљно познавање хардвера. Поред тога потребно је превести их у машински језик да би процесор могао да разуме, а то ради асемблер (енг. *assembler*) који узима једну по једну линију асемблерског кода и преводи га у машински код, а поступак се назива асемблирање. Код ове генерације програмских језика, као и код претходне проблем је портабилност, тј. пребацавање на другу машину.

Програмски језици треће генерација за разлику од претходне две генерације су више окренути људима, јер су њиховом употребом програмер не води много рачуна о детаљима ниског нивоа машине, који се апстрахују. Обзиром да било која програмска инструкција је потребно да се изврши на нивоу машине, потребно је да постоји процес у којем ће се инструкције програмског језика вишег нивоа превести на инструкције машинског језика. Поменути процеси се називају компајлирање или превођење и интерпретирање, а спроводе се уз употребу преводиоца (енг. *compiler*) и интерпретера (енг. *interpreter*). Обзиром да су независни од машине, могуће их је пребацити на другу

машину и извршавати. Примери програмских језика ове генерације су: Fortran, ALGOL, COBOL, ADA, Pascal, C и тд.

Једноставнији су за употребу од претходне генерације асемблерских језика, али имају одређене потешкоће (Grubb & Takang, 2003):

- Писање и дебаговање програма је споро, тешко и скупо. То може успори процес развоја, па софтвер можда не буде испоручен на време.
- Ефективно их користе само професионални програмери. Уколико на тржишту рада не постоје кадрови са тим вештинама пројекат може да буде успорен и рок његовог завршетка продужен.
- Трошкови одржавању су високи јер је на сложеном систему измена је спора и тешка.
- Потребно је написати неколико линија кода да би се решио веома мали проблем, што може отежати продуктивност програмера.

Програмски језици четврте генерације (енг. *fourth generation of programming languages*), скраћено 4GL, настали су током 80-их са идејом да је потребно што већи број људи укључити у програмирање. У том тренутку је постојао релативно мали број програмера на тржишту и они су користили програмске језике треће генерације, тако да је развој софтверских решења за којима је појавила све већа потреба на тржишту значајно порасла, па је било потребно дизајнирати нову врсту програмских језика који су једноставни за коришћење. Идеја водилца је била смањити време потребно за програмирање софтверских решења, тако што ће се креирати програмски језици, најчешће са релативно малим бројем кључних речи и програмских инструкција, а блиски језику људи, тако да већи део популације без формалног образовања у софтверском инжењерству може да започне да се бави програмирањем. Програми се креирају брзо, софтверски производи се завршавају и продају, чиме произвођачи софтверских решења остварују своје планиране циљеве.

Постоје разне врсте 4GL језика које се могу користити у различите сврхе, а неки од концепата поменуте генерације језика су следећи (Fisher, 1986):

- интегрисани систем за управљање базом података
- подскуп непроцедуралног језика прилагођен за крајње кориснике који нису техничка лица
- подскуп непроцедуралног и процедуралног програмског језика прилагођен за професионалце који се баве процесирањем података

- генератор екрана
- до 10 пута побољшања у продуктивности програмирања, измерен преко количине кода коју програмер пише у односу на трећу генерацију програмских језика
- генератор извештаја
- упитни језик (енг. *query language*)

Једна веома заступљена врста ове генерације програмских језика су језици са концептом табелом вођеног програмирања (енг. *table-driven programming*). Програмерско окружење је направљено тако да корисник може без и једне линије програмског кода (енг. *codeless programming*) да направи програм, тј. екранску форму која подржава све операције над базом података, приказа података, унос новог, измена постојећег и брисање, у пракси познатије као CRUD (Create, Read, Update, Delete) операције. Програмер је вођен корисничким интерфејсом да одабере операције и елементе екранске форме које програм садржи.

Познати 4GL програмски језици су: Clarion, Clipper, FoxPro, PowerBuilder, Informix 4GL, Ingress 4GL, Progress 4GL, SQL, ABAP, PL/SQL и тд. Неки програмски језици четврте генерације су непроцедурални програмски језици, где корисници шта програм треба да ради, али не и то како да ради. Најчешћи примери таквих језика је писање упита над базом података или дизајнери (генератори) екранских форми. Неки програмски језици четврте генерације комбинују карактеристике и процедуралних и непроцедуралних језика.

Обзиром да одржавање софтвера представља велики удео у животном циклусу развоја софтвера и да је финансијски захтевно, предности програмских језика четврте генерације су: смањење трошкова, једноставно разумевање, аутоматска документација и смањење оптерећења програмера (Grubb & Takang, 2003).

Смањење трошкова везано је за време проведено у целокупног процесу развоја софтвера употребом одређеног софтверског окружења, где је нагласак на фазу развоја и фазу одржавања. Такође требало би имати у виду и део у вези тестирања готовог софтверског решења. Апликације написане у програмском језику четврте генерације имају у себи велики део програмског кода који је већ генерисан и укупно гледано програмер потроши мање времена да заврши апликацију, у односу на остале програмске језике треће генерације. Нпр. апликација се састоји из неколико слојева, најчешће корисничког интерфејса, пословне логике и комуникације са базом података. Програмски

језици четврте генерације усмеравају програмера да уз употребу софтверског окружења, које у позадини генерише много програмског кода који се не види, напишу програмски код са веома мали број линија који ће чинити апликацију, при томе они могу да имају одређену мешавину дела кода који се односи на кориснички интерфејс, дела кода који чини пословну логику и дела кода који ради са базом. Код програмских језика треће генерације, програмер је потребно да потроши више времена да сваки од поменутих слојева засебно напише, а затим их организује у складу целину. Веома често постоје велике разлике у брзини развоја, јер неки програмски језици четврте генерације омогућавају рапидан развој софтвера.

Једноставно разумевање програмског језика је веома значајно за програмере поготово у ситуацији када је потребно да раде на апликацији коју су преузели од другог колеге који је напустио предузеће. Такође уколико програмски језика има релативно мали број програмских инструкција, као што је често случај са програмским језицима четврте генерације, обзиром на њихову специфичност за одређени домен, као што је нпр. пословни домен, тада програмери који одржавају код потребно је да много мање времена проводе над истим да би га разумели и на крају да би извршили тражене промене. Уколико се ради о пословном домену, команде могу да садрже кључне речи које раде са табелама из базе података, спроводе операције ажурирања над базом података, преузимају податке са корисничког интерфејса када их корисник унесе, и друге програмске конструкције. Број тих инструкција је значајно мањи него што је у случају конвенционалних програмских језика као што су: Pascal, COBOL и C.

Генерисање аутоматске документације је такође један од значајних карактеристика програмских језика четврте генерације, где унапред постоје познате секције и готови програмски шаблони који омогућавају да на веома једноставан начин се креира документација. Секције предефинисаног програмског кода које нуди окружење су познате, као и секције корисници дефинисаног програмског кода су познате, па је веома лако извршити генерисање документације из дела кода које је писао корисник, при чему се првенствено мисли на документовани код, док нека напреднија имају и могућност приказа одређених аспеката дизајна.

Смањење оптерећења на запослене је веома важан аспект у свим пословима, па тако и у програмирању. Када постоји веома мала заједница програмера, онда је тешко пронаћи специјализовану радну снагу и завршити пројекте, чиме се увећава притисак на сваког програмера који ради на апликацијама. Када је окружење организовано тако да је потребно много мање програмског кода написати, тада се увећава продуктивност

програмера и смањује се његово оптерећење. Поред тога, уколико окружење је једноставно за учење тада је претпоставка да је могуће већи број нових људи, који су спремни да се преквалификују, запослити и смањити притисак са већ постојећих запослених. Слична ситуација се односи и на оне запослене који имају вештине у другим програмским језицима или су тек завршили факултете или више школе.

Недостаци програмских језика четврте генерације огледају се у следећем (Grubb & Takang, 2003):

- апликациона специфичност - многи програмски језици четврте генерације су пројектовани за специфичне домене употребе и то их чини тешким у коришћењу у другим доменима у односу на оне за које су оригинално дизајнирани. Овај недостатак може да проузрокује неекономски или пак немогућ напор да се дода нека функционалност у програм, јер није омогућена језиком, за разлику од програмских језика треће генерације, који подржавају цео спектар апликација.
- власништво - многи 4GL језици нису компатибилни са другим језицима. Они су власнички софтвер, а нису отвореног кода. Једном када је 4GL систем купљен од продавца, организација је забринута због тога што не може једноставно да мигрира на сличне системе или да поново употребљава код из других окружења без великих трошкова. То се одражава на дугорочну еволуцију система.
- сензационалистичко размишљање по којем су 4GL језици веома једноставни - има много тврдњи у којима се наводи да су програмски језици четврте генерације дизајнирани за непрофесионалне програмере. Ниједан 4GL језик сам за себе није замена добрим основама у теоријама софтверског инжењерства. Овај појам једноставног коришћења може заправо да доведе до неодрживог програмског кода.
- лош дизајн - термин „једноставно коришћење“ значи да системи могу да буду развијени од стране људи без експертизе или основе у софтверском инжењерству, или од људи који немају потпуно разумевање у решавању пословних проблема. Ово може да води до лошег дизајнираног и имплементираног система са мало или без документације. Такви системи ће убрзо постати неодрживи.

У ЈКП БВК у употреби је Progress 4GL програмски језик који је настао почетком 80-их година прошлог века, намењен развоју пословних апликација, а који је у

власништву корпорације Progress Software Corporation, скраћено PSC. Комбиновањем језика четврте генерације и релационе базе података Progress 4GL омогућава употребу рапидног развоја софтвера. Постоје три основне врсте наредби у Progress 4GL-у: процедуралне наредбе, наредбе за приступ бази података и наредбе за кориснички интерфејс. Progress 4GL је доменски специфичан језик (енг. *Domain Specific Language*, скраћено DSL) намењен развоју пословних апликација. Он је процедурални програмски језик, веома сличан структурираном упитном језику (енг. *Structured Query Language*, скраћено SQL).

Употребом Progress платформе апликације се извршавају дистрибуирано према архитектури клијент-сервер, где клијентски део апликације шаље захтев за извршењем серверском делу апликације, који прихвата захтев, обрађује захтев и након тога прослеђује одговор клијентском делу апликације. У Progress платформи, постоје две врсте клијента који се могу корисити у овој архитектури, то су карактерни и графички, а серверска компонента се назива AppServer. Овај тип програмског модела је радио годинама и на тај начин су корисници извршавали своје задатке. Како је Progress настао пре појаве World Wide Web-а његова архитектура није могла да обради комуникацију путем HTTP протокола уз употребу претраживача и Web Server-а.

Да би поменути проблем био превазиђен PSC корпорација је креирала WebSpeed технологију и проширила постојећу архитектуру уз употребу WebSpeed Transaction Server технологије. Ова технологија има за циљ омогући да се преко технологије средњег слоја омогући динамички HTML, јер је Progress 4GL веома тешко поново употребити, па је потребно било израдити такав слој који ће умети да процесира HTTP захтеве. На тај начин се показује да иницијална Progress-ова погодност да омогући рапидан развој апликација, тада када су технолошке околности биле промењене постала је фактор ограничења, јер не постоји јасна сепарација задужења између слојева, где би се на пример омогућило да се слој пословне логике поново употреби. WebSpeed је власничко решење истог произвођача софтвера, којим се омогућава да се иницијална архитектура решења радикално не мења, већ да се технологијом средњег слоја омогући начин њеног функционисања са Web технологијама, где је потребно изнова написати решење, а где се само део постојећег кода може искорисити и прерадити у потпуно другу форму, што захтева веће радно ангажовање у односу на типичне поновно употребљиве софтверске компоненте. У одређеним ситуацијама, то ангажовање је веома обимно и често превазилази време и напор уложено у развој desktop апликација, јер је потребно додати и

програмски код написан у HTML, CSS i JavaScript технологији уз прераду постојећег кода.

5.3.1.1.1. Документација није ажурна или не постоји

Иако има велики број израђених програмских апликација и модула, наслеђени систем ЈКП БВК највећим делом нема документацију, док неколико апликација и модула имају документацију, али за неку од претходних верзија те апликације, која у тренутку посматрања није релевантна. Рад на наслеђеном информационом систему је организован тако да сваки програмер или пројектант обавља комплетан посао развоја софтвера: узимање захтева од корисника, пројектовање, програмирање, пуштање софтвера у продукцију и одржавање. Најчешће се ради о преузимању већ постојећих пројеката од програмера који су напустили предузеће. Овакав начин рада из много разлога није добар, пре свега не постоји раздвајање фаза у развоју и тестирању софтвера, могућности да се доносе другачије пројектно решење као последица различитих мишљења и дубинске анализе, као и много других бенефита који проистичу из тимског рада.

Једно од оправдања зашто је посао организован тако, барем по менаџменту ИТ сектора јесте да се они константно боре са мањком запослених, па је свако од програмера или пројектаната система преоптерећен са бројем пројеката који развија и одржава. Када постоји већи обим посла, потребно је да се утврде проритети у његовом обављању. Организација пословања као у софтверским фирмама је немогућа јер не постоји довољан број запослених који би се одвојено бавили узимањем програмских захтева, засебно пројектовањем и архитектуром система, засебно програмирањем, засебно тестирањем и осигуравањем квалитета кода и тд.

У таквој организацији посла пракса у вези са документацијом јесте да је она непотребан посао који програмери и пројектанти система треба да обаве, јер се увек жури да се апликације измене или креирају нове што пре. Поред тога, како је организација посла уређена тако да сваки пројекат има само једног програмера или пројектанта који ради на њему, међу њима влада мишљење да је креирање документације непотребан посао, јер и овако само они раде на том делу система. Не постоји потреба да се изради документација јер нико други неће да је користи сем њих самих. Како они већ имају одређено знање о том делу система, сматрају да је то непотребан посао на који би они само изгубили време. При томе ИТ менаџер их не притиска да израђују документацију, јер и овако има проблема да се постигну рокови у вези са одржавањем и изменом постојећих функционалности система, а веома малим уделом развоја новог дела система. Однос одржавања и развоја новог, је такав да се на одржавање троши можда и преко 90%

времена и ресурса. Такав начин рада је присутан још од увођења ИТ технологија у ЈКП БВК, дакле почетком 90-их и када се користио COBOL као главни програмски језик. Овакав начин рада је проблематичан из много разлога и доводи до недостатка документације које проузрокује даље проблеме.

Пре свега, постоји **значајно смањено знање о целокупном наслеђеном систему**, при чему се веома мало разуме његова структура и његово понашање, јер ни једна особа не разуме и не сагледава систем у потпуности. То је проблематично, јер се има угисак да систем није довољно вредан, па се лако може донети погрешна одлука у вези модернизације система. У таквој ситуацији се може куповати ново решење, под одређеним изговором често сагледавањем само једне карактеристике, а при томе да то ново решење обавља исте или чак мање функционалности у односу на оно што већ постоји. Један од примера у ЈКП БВК који то показује јесте куповина софтвера за архиву, иако је у склопу наслеђеног система постојала апликација која је радила исте функционалности.

Поред тога проблематично је и то да је **знање концентрисано код програмера и пројектаната и то само о делу система на којем раде**. Када не постоји документација коју би они консултовали, тада су у позији да донесу лошије пројектне одлуке, које ће додатно ослабити структуру система. Такође то води у понављање великог дела кода који већ постоји, прављења тзв. дупликата. Типичан пример у ЈКП БВК је рад са шифарником пословних партнера, било да су они купци или добављачи. Готово сваки програмер у својој апликацији има потребу макар да прочита податке о пословном партнеру и прикаже их кориснику. Уместо да постоји један програм који би то радио, постоји много дупликата који то раде. Треба имати у виду да елиминисање дупликата и прављења делова кода који су поново употребљиви (енг. reusable) је један од важних принципа који се примењују у пракси софтверског инжењерства.

Велики проблем је и то да се **тешко и споро преносе знања између програмера и пројектаната у ситуацији када је потребно интегрисати два или више подсистема**. Услед тога што документација не постоји сваки од програмера мора да вербално опише шта његов код ради и да на тај начин изврши трансфер знања о делу система на којем ради другом програмеру да би могли да интегришу подсистема, тј. апликације. На примеру развоја функционалности потребне за опредељивање планираних средстава по активним уговорима, која се користи за креирање финансијског плана, програмер који је радио на креирању те функционалности, морао је да више пута да консултује другог програмера који је радио на уговорима, наруџбеницама и фактурама

добављача да би могао да реализује поменућу функционалност, што се одужило на месец дана рада иако је било предвиђено само 2 недеље.

5.3.1.1.2. Особље које је радило на развоју система је напустило предузеће

У ЈКП БВК у периоду од 2010. у Служби за пројектовање и програмирање пословних апликација, која је задужена за развој и одржавање наслеђеног система је радио 10 запослених, а у 2020. их је 6, при чему су у последње 3-4 године фирму напустили 3 запослених са искуством у збиру преко 40 година рада. Сваки пут када би неко напустио предузеће, поготово уколико је радио/радила на већем броју апликација те активности би се прерасподелиле на преостале програмере и тиме би се увећао обим посла преосталих програмера.

Када програмер или пројектант напусти предузеће, сво знање о делу система на којем ради, заједно са доменским знањем одлази заједно са њим/њом, што је у пракси чест случај. У таквом ситуацији када документација не постоји, предузеће мора изнова да улаже у сазнавање о систему, функционалностима које тај део система обавља а самим тим и систем у целини, као и о реализацији тих функционалности у имплементационом окружењу. На примеру апликације за пријављивање кварова на водоводној и канализационој мрежи, програмер је радио 10 година и затим је напустио предузеће, а други програмер који у том тренутку ништа није знао о том делу система, је преузео да одржава и прилагођава тај део система у односу на вишеструке промене које су тражене током година. Време које је програмер који је преузео да одржава пројекат морао да потроши за спознају система је значајно, а за све то време предузеће је исплаћивало зараду, дакле предузеће је за истоветне активности које се тичу знања о систему више пута имало трошкове.

5.3.1.1.3. Новозапослени немају довољно знања и вештина потребних за одржавање система

Новозапослени који долазе са факултета у пракси ретко када поседују знања и вештине за одржавање старијих система. У плану и програму студија је ретко када заступљена проблематика наслеђених система, одржавање или еволуција софтвера, јер се универзитетски програми⁹ углавном фокусирају на проблематику развоја новог система, па су предмети и курсеви организовани око фаза у животном циклусу развоја система до пуштања система у рад. То је веома контрадикторно у односу на реалност и праксу у којој се деценијама показује да је много веће учешће одржавања, како и са аспекта

⁹ један од ретких примера у Србији је Електротехнички факултет у Београду, где је на мастер академским студијама организован предмет Еволуција софтвера

времемена, тако и са аспекта ресурса у односу на развој нових система и то у односу отприлике 80:20.

На примеру ЈКП БВК, новозапослени програмери 2010. године су завршили факултет, при чему су највише додиром имали са објектно-орјентисаним програмским језицима Java и C# који су 15 и више година млађи од Progress 4GL-а и припадају програмским језицима треће генерације у односу на Progress 4GL. Поред тога, они су програмски језици опште намене и подржавају објектно-орјентисану парадигму, за разлику од Progress 4GL-а који је доменски специфичан језик за пословну примену, где се размишља у појмовима табела што је скучено посматрање у односу на објекте који могу да представљају било шта у реалном свету. На одређен начин они су морали да се „одуче“ од принципа енкапсулације, наслеђивања и полиморфизма из ООП и усвоје начин процедуралног размишљања који је у теорији програмских језика био добар разлог уопште да настане објектно-орјентисано програмирање. На поменути начин они нису могли да користе знања за која су се школовали и обучавали, која су углавном модерна и услед тога на одређени начин они су у старту имали губитак мотивације. Из поменутог разлога они нису видели рад на наслеђеним системима као вредно искуство, па се тиме и нису ангажовали довољно и предано, зато што кроз рад они не развијају модерна знања и вештине из области софтверског инжењерства која су тражена на тржишту у већем обиму.

5.3.1.1.4. Нарушена почетна архитектурална замисао

Услед дугорочног одржавања временом долази до нарушавања почетне архитектуралне замисли. Како се од програмера тражи да брзо учине промене на постојећем софтверском решењу, често се дешава да услед те брзине архитектура испашта. Неодовљно времена пороведено на осмишљавању архитектуре, може да произведе дугорочне последице по софтверско решење и да доведе до ефекта таласа у виду промена које нису могле бити сагледане, а настале су у потпуном другом делу система.

Једна од грешака којих се често прави јесте да се компоненте или програмске целине учине тесно повезаним (енг. *tight coupling*) па постоје проблеми ефеката таласа, активности одржавања су веће услед међузависности и теже је вршити поновну употребу. У примеру ЈКП БВК постоји веза између апликације за повезивање књиговодственог дела система са осталим апликацијама реализована преко тзв. апликације за пренос код које је дефинисан фајл одређеног формата који апликација прихвата. Услед потребе да се програм који ради са тим форматом измени, то је довело до тога да су програмери морали

да прилагођавају све своје програме који креирају фајлове у траженом формату, неке од њих су апликација за попусте и апликација за креирање књижних одобрења.

Пример добре праксе који се дуго година користи у индустрији је трослојна архитектура која има презентациони слој (енг. *presentation layer*), слој пословне логике (енг. *business logic layer*) и слој који приступа подацима (енг. *data access layer*). Раздвојене целине чине да је укупно одржавање једноставније, јер се тачно зна за шта је који слој задужен и коју функцију и одговорност има.

Код Progress 4GL-а начин рада је такав да један програмски код комбинује и презентацију кориснику и пословну логику и приступ бази података, па када је потребно вршити измене, понекада су оне једнаке развоју новог. На примеру апликације за командно контролни центар, примарни кључ базе података је проширен за још једну колону, а то је проузроковало у апликацији да се у преко 15 програма и подпрограма морају спровести измене да би се делом задржала логика која је била пре тога присутна. То се радило на тај начин да корисници не би осетили значајне измене у њима већ познатом току програма, а то је проузроковало ситуацију да неки програми су морали да буду поново написани (испрограмирани) од почетка.

5.3.1.1.5. Редуданса

Редуданса, тј. вишеструко памћење истих података је нежељена карактеристика која спроводи аномалије у ажурирању базе података. На примеру ЈКП БВК готово свака апликација користи шифарник радника приликом логовања, као и приликом евидентирања који радник је последњи ажурирао одређени ред у табелама у базама података. Уместо да постоји један шифарник радника, у неколико различитих база података постоје шифарници. Само у апликацији кадровске евиденције се ажурира на које радно место је радник распоређен у ком тренутку и такође из поменуте табеле се види ко је коме надређен у организационој структури.

Неке од апликација имају унутар себе уграђен вид контроле да након уноса података од стране радника у базу података, његов надређени исконтролише тај унос и провери пословни документ на који се тај унос односи или означава да је сагласан са неким предлогом или да одобрава реализацију одређеног пословног документа и тд. На примеру апликације која ради са набавкама догодило се да је одређеном раднику промењен руководиоца, а он не може да настави да ради јер нови руководиоца то није одобрио, услед тога што постоји засебна евиденција распореда радника у набавци, а промена у кадровској евиденцији је спроведена. То се често појављивало као проблем док се нису додатни програми написали који су кренули да поменуту редудансу држе под

контролом и да се на промене у табелама у кадровској евиденцији аутоматски промене спроведу и у апликацији за набаку.

5.3.1.2. Оперативна перспектива

5.3.1.2.1. Не постоји подршка произвођача хардвера и/или софтвера

У пракси се догађа да када је систем развијен у старијим технологијама, можда је за њега **престала да постоји подршка за хардвер на којем се извршава или софтвер**. Уколико наслеђен систем треба још годинама да ради, то може да представља велики проблем.

У ЈКП БВК проблем који се јавио у вези софтвера, јесте да је произвођач Progress OpenEdge окружења PSC за верзију 10 окружења повукао из употребе софтвер за извештавање Report Builder који се годинама користио. Сваки програмер који треба да направи нови или измени постојећи извештај потребно је да паралелно покрене ранију верзију Progress окружења и да у њему измени извештај, па тек онда да га стави у продукцију. Други део програмера који је одлучио да се склони са поменутог софтвера за извештавање, и пређе на Crystal Reports XI морао је да све извештаје преради у поменутом алату.

Сличан проблем се јавио у вези подршке разних ActiveX контрола које се користе у развоју софтвера, а које Progress окружење у основи не нуди, јер је ограничено само на онај скуп контрола које оно има у себи (а које је произвођач раних дана предвидео). Ради се о софтверу који је написан у другом програмском језику, а изложен је тако да може да се користи уз неке функционалности које нуди оперативни систем. Типичан пример такве контроле је софтвер који омогућава да се изврши приказ скенираних докумената. Када је промењена верзија развојног окружења и верзија оперативног система над којим се извршава, напред наведене ActiveX контроле су престале да раде јер су написане тако су компатибилне са ранијом верзијом, али не и са новом и то је проузроковало много додатних проблема.

Други проблем који може да постоји јесте ако се **не спроводе одговарајућа ажурирања система од стране произвођача** јер је напустио развој или још није креирао одређену закрпу. Тако на примеру ЈКП БВК у верзији Progress OpenEdge 10.2В да би се спречило могућност читавања Java програма у AdminServer компоненту потребно је било да се изврши измена конфигурационог фајла. Док то није урађено систем је био у сигурносном ризику и постојала је могућност одређених малициозних активности

усмерених према информационом систему предузећа у виду неколико типичних сигурносних напада.

Још један проблем који постоји јесте **проблем компатибилности између верзија система**, односно развојног окружења у којем је наслеђени систем написан. На примеру ЈКП БВК показало се да одређене програмске конструкције Progress 4GL језика, као што је нпр. TEMP TABLE, не могу да се користе у новијој верзији Progress OpenEdge решења, верзије 11 у који су инвестирана значајна средства, иако произвођач софтвера не указује на поменуте проблеме. Због тога, програми написани у старој верзији који користе напред наведени концепт привремене табеле не могу да се компајлирају у новој верзији система. На поменути начин се блокира пребацивање програма на новију верзију која има веће могућности, а у коју су инвестирана значајна средства.

5.3.1.2.2. Проблеми интеграције система

Развој информационог система у ЈКП БВК се одвијао тако што се правио један део система, програмски модул или апликација, па се затим развијао наредни део система и тд. То се радило на основу захтева других организационих јединица, тако да након фазе развоја организациона јединица је добијала свој програмски модул или апликацију. Дакле, уместо да се са развојем информационог система крене одозго на доле и да се прво систем осмисли у целости, а затим да се ради на његовом развоју, кренуло се приступом одозго на горе тако да се за сваку организациону јединицу која је то тражила развија засебна апликација, а затим да се те апликације међусобно интегришу.

Треба имати у виду да ИТ сектор никада није могао да утиче на пословање других организационих јединица, јер су они увек били „власници“ својих пословних процедура, а улога ИТ-а је била да само то информатички подржи. Описани начин рада довео је до формирања силоса, одн. острва аутоматизације података. Када је уочено да ти подсистеми, модули или апликације је потребно да буду повезани међусобно, тада се приступило њиховом интегрисању. Овакав вид ad-hoc интеграције довео је до тога да се проблеми у вези са архитектуром и тесном везом додатно нагласе. Интегрисање се одвија преко базе података, тако што се из једног дела система програмским процедурама, које се покрећу ван радног времена, подаци се пребацују у други део система.

Проблем који се дешавао у пракси јесте интеграција са програмским пакетом EDAMS, који служи за управљање средствима инфраструктурне мреже, а који је користи SQL Server као базу података. У апликацији за пријаву кварова, постоји потреба да када се квар на водоводној или канализационој мрежи распореди на одређен погон и када се уђе у реализацију тог квара из EDAMS систем се одштампа радни налог. То је

реализовано тако што се у Progress 4GL-у на одређен догађај на корисничком интерфејсу, подаци прослеђивали у SQL Server. Поменута интеграција је била проблематична јер се веома често догађало да подаци нису уписани у SQL Server базу података, а када су се касније нови подаци преносили у Progress постојао је проблем да или нису пренесени или су дуплирани и тд. То је захтевало константну интервенцију програмера на оба система.

ЈКП БВК је до пре неколико година имао своју апликацију за архиву, која скенира сва улазна и излазна документа из предузећа. Ова апликација је била тесно повезана са многим другим апликацијама, које су користиле њене функционалности. Апликација за архиву која је развијена у Progress платформи је замењена другом апликацијом развијеној у Java платформи, док је интеграција урађена тако да се у већ раније осмишљене табеле у бази које су се користиле за стару апликацију архиве подаци пребаце из нове базе података која ради са новом апликацијом.

Поменуто нова апликација је развијана ван ЈКП БВК и процес ETL није урађен како треба, тако да је настао читав низ проблема, дупликати у бази података, недостајући скенирани документи. То је произвело многе проблеме у апликацијама које су користиле податке из архиве (ефекат таласа), запослени нису могли да обављају свој посао, од програмера из ЈКП БВК је тражено додатно ангажовање да би се проблеми превазишли.

5.3.1.3. Организациона перспектива

5.3.1.3.1. Систем ограничава пословање

Наслеђен систем извршава одређене функционалности и на тај начин подржава пословне операције. Успостављањем наслеђеног система много година креиране су и тадашње пословне операције (процеси), при чему су се они временом мењали, па је било потребе да се и наслеђени систем мења. Некада те промене се не могу спровести у року који се захтева, а некада су толико велике да захтевају репрограмирање одређених делова система. У том смислу наслеђени систем ограничава пословање и менаџери који су одговорни за припадајуће пословне операције их виде као сметњу.

На примеру ЈКП БВК, предвиђено је да програм који врши пренос одређених пословних докумената у финансијски део система може да пренос једне те исте фактуре уради само једном. У пракси се појавила потреба да се рачуни за воду и канализацију умање више од 1 пута, путем књижних одобрења. Програм за пренос је спречавао да се пренесе два пута фактура која је ушла у два различита књижна одобрења и након тога прокњижи. То је решено тако што су референти који фактуришу воду и канализацију направили књижно писмо ручно и проследили га надлежној финансијској служби на

ручно књижење. Дакле, они су документима у папирној форми заобишли део наслеђеног система који их је ограничавао.

На случају апликације за планирање набавки, креиран је одређени ток програма који кориснику омогућава да након спроведених одређених корака, нпр. одабира типа набавке, типа поступка, уноса предмета набавке, уноса процењене вредности и тд. се креира планирана набавка. Сектор набавке је желео да увођњем нове пословне активности, креирање техничке спецификације, измени поменути начин рада тако да се корисници прво креирају техничку спецификацију, а затим из ње набавку. То није било могуће спровести услед раније дефинисаног тока програма, којег је немогуће проментити. Обзиром да би реализација поменутог захтева подразумевала програмирање од почетка целе екранске форме, услед других приоритета и рокова у којим је потребно завршити пројекат, одлучено је да се то не ради. Иако је реч о промени која није суштинска и не мења пословну операцију значајно, претходни начин рада на апликацији и проблематично развојно окружење су и такву промену малог интензитета онемогућили.

Поред пословног информационог система ЈКП БВК има и систем који служи пословном одлучивању који је реализован у IBM Cognos технологији. Из постојећег наслеђеног система и базе података Progress подаци се трансформишу и пребацују у структуре које одговарају софтверу за пословно одлучивање. Примећено је да менаџмент средњег слоја, као и топ менаџмент у малој мери користи систем за пословно одлучивање. Имајући у виду да су значајна средства уложена за набавку поменутог решења, као и за обуке запослених, намеће се питање који су разлози да се такво једно решење не користи. Један од одговора на то питање може да буде и у перформансама који поменути систем остварује у раду са наслеђеним системом из којег препуњава податке. Корисници нису задовољни перформансама или неким другим аспектима поменутог решења, па га не користе у довољној мери.

5.3.1.3.2. Систем је вредан због подршке главним пословним активностима

Наслеђен систем се годинама развијао и одржавао и садржи вредна знања за обављање посла, као и многобројна пословна правила која су настали у интеракцији програмера и корисника приликом узимања софтверских захтева. Поменута правила и знања су прошли године тестирања у пракси и могу послужити да пословне операције доведу на висок ниво ефикасности. То је посебно изражено тамо где постоје неке пословне операције које нису толико заступљене у пракси и нису понуђене кроз интегрисана софтверска решења.

На примеру ЈКП БВК програмски модул за обрачун личних доходака је један такав пример, где садржи све потребне функционалности за решавање свих питања у вези обрачуна зарада запослених, поготово уколико се има у виду да Град Београд прописује одређену новчану масу коју ЈКП БВК може да искористи. На основу многобројног уграђеног знања, и ограничења из закона и правилника о зарадама, као и вишегодишње праске, сваки од радника добија свој постотак од новчане масе, тј. то је његов лични доходак. Све то је део програмског модула који је вредан за пословање.

Програмски модул за фактурисање воде и услуга је такође незаменљив, јер је део примарних активности у ланцу вредности ЈКП БВК, где се обрачунава потрошња и издају рачуни за воду, као и за многе услуге које предузеће омогућава, као што је нпр. уградња водомера. На поменути начин најважнији животни ресурс – воду, потрошачи могу да спроведу до својих кућа и станова, да је користе и да за то плаћају одређену надокнаду кроз фактуре за воду и услуге. Уклањањем поменутог модула из предузећа, ЈКП БВК не би могао да функционише, јер не би имао приходе.

5.3.1.4. Стратешка перспектива

5.3.1.4.1. Систем ограничава измену стратегије

Топ менаџмент може да донесе нову одлуку да се промени стратегија пословања, јер је то важно за предузеће приликом постизања његове мисије, визије и циљева. То може да покрене реинжењеринг пословних процеса, да би се измењена стратегија спровела. Оно што може да се догоди јесте да наслеђени информациони систем не може да подржи потребну измену стратегије, јер је креиран у неком претходом времену када је пословање било уређено на друачији начин.

На примеру ЈКП БВК уколико би стратегија била да се спроведе измена у систему фактурисања воде тако да се омогући тарифирање потрошње воде по вишој и нижој тарифи, то не би могло да се спроведе. Наслеђени информациони систем услед платформе у којој је реализован има сиромашне начине интеграције система и излагања својих функционалности ка другим системима, као и употребу функционалности од других система па је то проблем за реализацију измењеног пословног модела.

Једно сложено решење које би подразумевало паметне читаче, као и анализу података са пумпи које служе да потисну воду у мрежу, као и анализу профила потрошње потрошача имало би потребе да се интегрише са наслеђеним информационим системом ЈКП БВК, а он као такав нема тих могућности. 10.2В верзија Progress OpenEdge платформе не подразумева рад са REST сервисима, који би служили да се спроведе

интеграција једног оваквог комплексног софтверског решења, које је сачињено из више мањих софтверских решења са наслеђеним софтверским системом, који је потребно да податке прихвати у формату и на начин који је радио до сада.

Поред тога, што је тренутно овакво решење неизводљиво, потребно је имати у виду чињеницу да топ менаџмент има задатак да перципира финансијске бенефите од конкретног подухвата и да их упореди са трошковима који ће они произвести и да на основу тога одлучи да ли је исплативо спроводити поменути измену стратегије. Процес одлучивања треба да буде организован тако да топ менаџмент може да има увид у све трошкове које је потребно сагледати, што често може бити веома проблематично уколико се ради измена постојећег наслеђеног система.

Поред тога што ће бити потребно додатног рада да би се наслеђеном систему омогућило да има интеграцију са новим технологијама, постоји и много рада који ће условити промене у организационој перспективи, као што је нпр. реинжењеринг одређених пословних процеса који се тичу фактурисања воде и услуга. Затим, то ће да произведе промене у оперативној перспективи где је потребно да корисници који раде на апликацијама за фактурисање имају увид у поменуте промене, као и у нов начин рада, при чему могу да буду незадовољни одзивом система који је интегрисан и неконзистентности у достави података. На крају то ће произвести промене на развојној перспективи која је потребно да све поменуто спроведе над наслеђеним информационом системом.

5.3.1.4.2. Систем ограничава будућу стратегију

Да би смо боље сагледали ограничења која ће будућа стратегија ЈКП БВК имати потребно ће бити да анализирамо ланац вредности (одељак 5.3.2.), у којем ће се идентификовати кључне активности, као и технолошка решења која треба да побољшају поменуте активности, затим ћемо погледати како наслеђен информациони систем ограничава примену поменутих решења.

5.3.1.5. Одступања од предложеног теоријског оквира на примеру ЈКП БВК

Током фазе посматрања и описивања реализације теоријског оквира на случају ЈКП БВК уочен је један изузетак (проблем) који је у вези са стратешком перспективом, а који није део теоријских полазних основа.

Поменути проблем односи се на повезаност улоге која организациона јединица задужена за информациони систем, ИТ сектор, има у предузећу, и како се та улога перципира од стране топ менаџмента.

У ЈКП БВК ИТ сектор не може да утиче у потпуности на стратегију развоја ИС (није независан у одлучивању по питању информационог система предузећа, па самим тим и наслеђеног система) јер са других организационих нивоа, најчешће виших у хијерархији намеће се употреба софтверских решења која производе многе проблеме и у свим другим перспективама наслеђеног система: организационој, оперативној и развојној.

Поменути проблем није само део теорије информационих система и софтверског инжењерства већ је повезан и са теоријом менаџмента.

Рос и Фини су истраживали улогу и кредибилитет директора сектора за ИТ током историје развоја информационих система и учили да његова улога и моћ расту у односи на пораст организационог учења (Ross & Feeny, 1999). У мејнфрејм ери, од 60-их до раних 80-их, улога директора сектора за информационе технологије се своди на испуњавање захтева и обећања. У дистрибуираној ери, од краја 70-их, улога директора сектора за информационе технологије је промењена у члана извршног тима, где је потребно да се ускладе ИТ функције са пословањем. У Web-заснованој ери, од средине 90-их, улога директора сектора за информационе технологије је да да оријентише стратегију и да развија нове пословне моделе које ће искористити Web технологије.

Чун и Муни су спровели истраживање 25 година касније, потврдили његову валидност и указали на чињеницу да је дошло до раздвајање функције директора сектора за информационе технологије, на две функције: директора сектора за информационе технологије и директора за иновације (Chun & Mooney, 2009).

На случају ЈКП БВК може се приметити да је улога директора сектора за информационе технологије се налази негде између прве и друге фазе, тј. између мејнфрејм и дистрибуиране ере, што је зачуђујуће обзиром да случај посматрамо скоро 40 година после. То указује да брзина организационог учења у ЈКП БВК је значајно мања у односу на трендове који су познати у теорији. Навешћемо два примера на случају ЈКП БВК у којима се то манифестује.

На примеру већ ранијег поменутог проблема архиве код оперативне перспективе, где је замењена апликација архиве наслеђеног система новом, што је проузроковало много проблема у раду, донесена је одлука изван ИТ сектора да се набави поменути програмски пакет. На тај начин ИТ сектор није имао прилику да учествује у одлучивању да ли је одлука оправдана, да ли је исправна у односу на стање наслеђеног система и анализа да ли ће постојати користи од нове апликације и колике ће оне бити, као и да ли

ће постојати неки проблеми приликом њеног увођења, односно повлачења апликације из наслеђеног система.

Такође уопште се није обратила пажња на то да је раније уложен рад програмера у апликацију наслеђеног система, који је фирма већ исплатила. Поред тога није се ни извршила анализа да ли ће висина трошкова увођења новог система и прилагођавања постојећег наслеђеног система бити мања у односу на бенефите које доноси поменуто нова апликација. Да ли нова апликација нешто суштински мења у раду архиве или је само „новије“ технолошко решење.

Такав проблем у стратешкој перспективи, који је настао услед изостављања ИТ менаџера из одлучивања, је генерисао проблеме и у свим осталим перспективама, који су месецима исправљани, а ни до данашњег дана нису у потпуности елиминисани.

Други пример је увођење ERP решења које треба да има за циљ да замени многе апликације и финасијске програмске модуле који постоје у наслеђеном систему, где се без укључивања ИТ сектора одлучује о томе да ли је поменуто стратегија модернизације дела наслеђеног система адекватна за решавање проблема наслеђеног информационог система, као и потенцијалних проблема који ће настати уколико поменуто решење уђе у употребу.

Услед стања наслеђеног система ЈКП БВК интегрисано решење, обзиром да у целости не покрива све апликације и програмске модуле које ЈКП БВК има, а и пословне процедуре произвешће проблеме и у организационој и у оперативној и у развојној перспективи наслеђеног информационог система. Промене се састоје у: реинжењерингу пословних процедура (процеса), реорганизацији рада служби, интеграцији поменутог готовог решења са свим наслеђеним апликацијама и програмским модулима, неко време рад оба система у паралели, трошкове одржавања оба система у паралели, праћење измена и промена оба система у паралели (верзионирање), реинжењеринг апликација наслеђеног система и многе друге.

Трошкови спровођења такве одлуке су вишеструки, понегде и скривени, а никада нису у целости сагледани и упоређени са осталим алтернативама модернизације наслеђеног система као што су облагање система или софтверски реинжењеринг. Такође, према Росу и Финију ERP системи одговарају другој технолошкој ери (између 80-их и 90-их), па се поставља питање како је то интегрисано софтверско решење оправдано увести у организацију која би када би требало да буде на нивоу организационог учења 25 до 30 година после и да исказује значајно друге карактеристике, које би барем подразумевале

да поред директора сектора за ИТ, постоји и директор за иновације који би усмеравао одређену стратегију иновација.

5.3.2. Анализа ланца вредности на случају ЈКП БВК

Усвајањем и применом конкурентске стратегије организација жели да створи конкурентску предност. За повећање конкурентности, Портер је дефинисао модел сила конкуренције (енг. *competitive forces model*) који обухвата 5 основних сила који могу побољшати или угрозити позицију неке организације у одређеној грани привреде (Rainer & Turban, 2009), а оне су:

- опасност од појаве нових конкурената
- преговарачка моћ добављача
- преговарачка моћ купца
- опасност од појаве замене за производе и услуге
- ривалство између фирми у једној привредној грани

Основне стратегије за стицање конкурентске предности су (Rainer & Turban, 2009):

- стратегија најнижих трошкова - производе се производи или услуге уз најмање трошкове у датој грани индустрије
- стратегија диференцијације - нуде се различити производи, услуге или карактеристике производа
- стратегија иновације - уводе се нови производи и услуге, додаје се нова карактеристика постојећим производима и услугама и развија нов начин производње
- стратегија ефективности пословања - побољшава начин на који се извршавају унутрашње пословне активности у организацији, тако да компанија обавља исте или сличне активности боље од својих ривала
- стратегија орјентисаности ка купцима - концентрише се на то да задовољи купце

За одређивање специфичне стратегије коју нека организације је потребно да примени, користи се модел ланца вредности (енг. *value chain model*). Ланац вредности је серија повезаних активности које додаје вредност производима или сервисима организације (Bosij et al., 2015, p. 52). Постоје 2 врсте активности примарне активности и активности подршке (Stair & Reynolds, 2018, p. 47). Примарне активности директно

утичу на производњу и дистрибуцију фирминих производа и услуга и уобичајено за производњу обухватају: улазну логистику, операције, излазну логистику, маркетинг и продају и сервисе. Активности подршке укључују 4 основне области активности подршке: технолошка инфраструктура, менаџмент људских ресурса, рачуноводство и финансије и набавка.

5.3.2.1. Ланац вредности водоводних предузећа

Ofwat је регулаторно тело основано 1989. године када је извршена приватизација индустрије водовода и канализације Енглеске и Велса. Према Ofwat ланац вредности предузећа водовода и канализације изгледа као на слици 32.

Ланац вредности предузећа водовода и канализације садржи следеће важне активности (PwC, 2016):

- извори воде
- дистрибуција сирове воде
- прерада воде
- дистрибуција пијеће воде
- продаја
- скупљање канализације
- прерада канализације
- прерада муља
- одлагање муља



Слика 32. ланац вредности водоводних предузећа према Ofwat-у, преузето (PwC, 2016)

Овако дефинисан ланац вредност, указује на све важне активности за једно водоводно предузеће, где нека водоводна предузећа не морају да имају све побројане активности.

5.3.2.2. Ланац вредности ЈКП БВК

ЈКП БВК организује и спроводи своје пословање, где је основни производ вода, а при томе поторшачи користе и услуге одвођења отпадних вода, тј. канализације.

Вода која ЈКП БВК обезбеђује и наплаћује својим потрошачима било да су они домаћинства или привреда, потребно је да задовољи одговарајући прописан квалитет, који се огледа у бактериолошко, хемијској и микробиолошкој исправности воде за пиће. Поред тога, испоручена вода на течећим местима потрошача је потребно да има одговарајуће вредности хидрауличких параметара, као што су притисак и проток, тако да потрошачи могу да је користе на адекватан начин.

Услуга одвођења атмосферских и отпадних вода (канализација) је такође веома важан сегмент пословања ЈКП БВК, где се потрошачима након испоручене и употребљене воде омогућава њено одвођење.

Анализом ланца вредности предложеног по Ofwat-у и упоређивањем са активностима које служе у циљу обезбеђивања основног производа пијеће воде и услуге одвођења отпадних вода, може се закључити да предложене активности по Ofwat-у су примарне активности ЈКП БВК. Сада ћемо их укратко описати.

5.3.2.2.1. Изворишта воде

Изворишта воде служе за добављање сирове воде. Она могу бити подземна и надземна. Надземне воде се захватају из река Саве и то већим делом и из Дунава мањим делом. Подземне воде се добијају путем рени бунара. При захвату воде спроводе се и процеси везани за уклањање грубих нечистоћа.

Активности које се спроводе у вези разуђеног изворишта су пре свега ширење изворишта ради добијања нових потребних капацитета који перманентно расте упоредо са растом самог града Београда, изградњом нових и регенерацијом постојећих рени бунара, као и активности везане за одржавање захвата речне воде и планираних повећања капацитета. Све то има за циљ да се постигне одговарајућа тражња за водом, која је променљива и стога је употреба изворишта сирове воде и њена операционализација почетни део ланца вредности ЈКП БВК.

5.3.2.2.2. Дистрибуција сирове воде

Дистрибуција сирове воде има за циљ да од изворишта сирове воде изврши њен транспорт до постројења за прераду воде. Она се састоји из скупа цеви, црпних станица и

осталих хидрауличких елемената који раде у синергији и потребно је да допреме сирову воду од изворишта воде до фабрике за њену прераду.

Задатак водовода је да обезбеди несметано допремање сирове воде до фабрика за прераду, па је потребно спроводити активности одржавања, као и изградње и реконструкције одређених делова мреже за дистрибуцију сирове воде.

5.3.2.2.3. Прерада воде

У постројењима за прераду воде, врши се технолошки поступак добијања чисте пијаће воде прерадом сирове воде. Технолошки поступак се разликује од извора који је одабран, обзиром да различити извори имају различит степен нечистоће који је потребно отклонити у технолошком процесу. Уобичајено је да технолошки поступак се састоји из неколико фаза, а неке од њих су: преозонизација, бистрење, озонизација, филтрација употребом двослојних филтера, филтрација употребом активног угља, хлорисање, УВ дезинфекција и тд. Након тога, се врши складиштење пијеће воде у резервоарима. Контрола квалитета воде се спроводи тако што се обавља неколико нивоа испитивања квалитета воде и то физичко-хемијска, микробиолошка и биолошка.

Процес контроле додатно врши Градски завод за јавно здравље. Контрола се спроводи након сваке фазе прераде воде, а такође се спроводи и над сировом водом. Процес узорковања се спроводи перманентно са различитом динамиком поједине врсте узорка, па се тако узорковање на производним погонима спроводи свакодневно, узорковање воде из резервоара 1 недељно, а са тачећих места потрошача 45 узорака дневно према засебно утврђеној динамици. Поред тога, на територији града узимају се узорци са 190 тачећих места која се контролишу 2 пута месечно.

5.3.2.2.4. Дистрибуција пијеће воде

Дистрибуција пијеће воде има за циљ да допреми пијећу воду од фабрике за прераду воде до крајњих потрошача. То је један сложен хидраулички систем састављен из цеви, вентила, пумпи и резервоара.

Обзиром да је део Београда изграђен на брдовитом терену, у процесу дистрибуције воде је потребно савладати разлику у висину на којој се налази насеље са потрошачима и висине на којој се налази резервоар пијеће воде добијену процесом прераде воде. Град Београд је подељен на неколико висинских зона, које се налазе на различитим надморским висинама. Пијећа вода се препумпава од фабрике за прераду воде до резервоара који се налазе на I висинској зони употребом црпне станице. Уобичајено је да се резервоар пијеће воде постави нешто изнад I зоне и затим да се вода

од њега путем силе гравитације допреми до потрошача који се налазе на висинама унутар I зоне. За потрошаче који се налазе унутар II зоне потребно је извршити препумпавање у резервоар намењен тој зони где се путем силе гравитације из резервоара вода допрема до њих. То се врши употребом црпне станице која препумпава воду са резервоара I зоне у резервоар II зоне. Исти поступак се примењује и на остале зоне више. Оваквим постепеним препумпавањем воде штеди се електрична енергија потребна за савладавање висинских разлика, у односу на начин да се у старт врши препумпавање на највише зоне (Ђурђевић, 2009).

Обавеза ЈКП БВК јесте да одржава дистрибутивну мрежу у исправном стању да би могла да испоручи здравствено исправну пијећу воду потрошачима. Такође ЈКП БВК је у обавези да трајно и континуирано врши испоруку воде, као и да се обезбеди одговарајући притисак и то 24 часа дневно, под једнаким условима за све потрошаче (Grad Beograd - Sekretarijat za informisanje, 2005). За случај да дође до прекида у водоснабдевању потребно је да се ангажују службе које раде на отклањању кварова и других разлога у прекиду водоснабдевања.

5.3.2.2.5. Продаја

Крајњи потрошачи, домаћинства и привреда, употребљавају воду, која се наплаћује у процесу продаје. Очитавањем мерног уређаја, водомера, добија се увид у очитано стање. Поређењем очитаног стања са водомера, и претходно очитаног стања утврђује се колико је потрошач утрошио воде и она се наплаћује по цени коју прописује ЈКП БВК уз одобрење Града Београда. ЈКП БВК је сходно Одлуци о пречишћавању и дистрибуцији воде, дужно да обезбеди очитавање водомера 2 пута годишње. У случају када из било којих разлога очитавање водомера није могуће, тада се потрошња наплаћује према просечној дневној потрошњи између два очитавања (Grad Beograd - Sekretarijat za informisanje, 2005). За потрошаче који се налазе у зградама, где више станара користи један водомер ЈКП БВК врши очитавање водомера, док саму расподелу по становима обавља ЈКП „Инфостан технологије“ и то према броју чланова домаћинства или према квадратури. Тако да потрошачи унутар зраде плаћају удео у укупној потрошњи.

Са досељавањем становништва у град и другим процесима који утичу на урбанизацију града почињу да се формирају градска и приградска насеља. Један број тих насеља је извршио изградњу своје водоводне дистрибутивне мреже, а која није у складу са стандардима и нормама које прописује ЈКП БВК и то су вишедеценисјки проблеми које ЈКП БВК покушава да реши. За таква насеља се на улазу у насеље уградио водомер, који се назива блок водомер и преко њега се очитава целокупна потрошња насеља. Поред

тога сами потрошачи могу да уграде своје контролне водомере и да им се потрошња наплаћује према читавањима контролних водомера, при чему треба нагласити да је то само један део потрошачевог рачуна. Потрошња обрачуната преко блок водомера и збира свих контролних водомера може да се разликује и то се назива растур воде. Поменути растур се распоређује на све кориснике који се снабдевају водом са блок водомера, тако да други део потрошачевог рачуна је удео у укупном растуру воде. Растур настаје из неколико разлога, пре свега мрежа која снабдева потрошаче није урађена по стандарду и нормама који прописује ЈКП БВК па могу настати разне врсте неконтролисаног отицања воде. Други разлог јесте што та мрежа коју су изградили потрошачи најчешће пре неколико деценија је потребно да се одржава, а они то у пракси најчешће не чине. Трећи разлог јесте што можда постоје нелегални потрошачи прикључени на мрежу.

Такође, ЈКП БВК је у обавези да одржава одређени део водомера, углавном уграђених у зграде, и да врши њихову калибрацију и редовну замену у прописаном року, да би се обезбедила тачна мерења утрошка воде, као и наплате по истим. Као што је раније поменуто, обавеза ЈКП БВК је да одржава дистрибутивну мрежу и то до самог доводног вентила, док је у обавези потрошача да одржава свој водомер и мрежу од одводног вентила па до њихових точећих места. На тај начин се регулише чија су задужења у погледу одржавања мреже.

5.3.2.2.6. Скупљање канализације

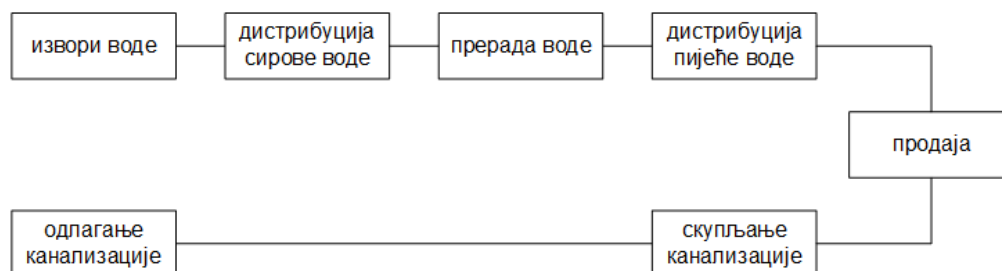
Вода након употребе одлази у канализацију, која има задатак да скупља атмосферске и отпадне воде у процесу њиховог скупљања, да их пречишћава и да их доводи са територије града и да их испушта у пријемнике воде. Атмосферске воде настају путем падавина као што су снег, киша, град. У отпадне воде се убраја она вода које су корисници искористили, као и воде настале у индустријским процесима и производним делатностима, затим вода за прање улица и тд. У процесу скупљања воде користи се мрежа канала (тзв. секундарна канализациона мрежа), преливи за рестерећење мреже или ретенциони басени за смањивање осцилација протицаја воде кроз канала, каскаде и друго. Поред тога користе се колектори, који су канали већег профила и они су елементи тзв. примарне канализационе мреже. Такође могу се користити и црпне станице које имају улогу да атмосферске и отпадне воде допреме до постројења за прераду ових вода.

ЈКП БВК има задатак и обавезу да одржава канализацију (Grad Beograd - Sekretarijat za informisanje, 2010). Ове активности укључују редовно, ванредно и инвестиционо одржавања. Поред наведеног, спроводе се хитне интервенције у циљу

отклањања тренутних и будућих дисфункционалности и активности у вези проширења капацитета.

5.3.2.2.7. Прерада канализације

Прерада атмосферских и отпадних вода је корак у којем се прикупљена и допремљена вода прерађује и смањује се њен лош утицај на екологију и тиме се спроводи адекватна заштита животне средине. Тренутно ЈКП БВК не спроводи поменути корак, јер он захтева значајна улагања у инфраструктуру. У јануару 2020. године, потписан је уговор између кинеске компаније China Machinery Engineering Corporation за реализацију фабрике за пречишћавање вода која се налази у Великом Селу, са чијом изградњом би био решен недостајући део канализационе инфраструктуре града Београда. Реализација поменутог пројекта је веома значајна и због Поглавља 27 Европске Уније која покрива област заштите животне средине и климатских промена. За сада атмосферска и отпадна вода се излива у Дунав путем испуста, а тиме се максимално шити Сава која се већински употребљава у процесу захватања сирове воде. Ланац вредности ЈКП БВК се донекле разликује у односу на ланац вредности водовода предложених по Ofwat-у и приказан је на слици 33.



Слика 33. ланац вредности ЈКП БВК

5.3.2.3. Проблеми и изазови у ланцу вредности ЈКП БВК

На основу анализе ланца вредности ЈКП БВК и сагледавања ризика одређених активности у ланцу вредности презентованих у извештају PwC-а, указује се на проблеме који се могу јавити у одређеним примарним активностима из ланца вредности ЈКП БВК, презентованих у табели 6.

Табела 6. проблеми и изазови у активностима ланца вредности

| | Активности | Проблеми и изазови |
|--|------------|--------------------|
| | | |

| Извор воде (извориште) | Извлачење сирове воде | Контаминација извора |
|--------------------------|---|---|
| Дистрибуција сирове воде | Пренос сирове воде до фабрике за прераду | Прекид транспорта сирове воде услед квара |
| | | Контаминација сирове воде |
| Дистрибуција пијеће воде | Пренос пијеће воде од резервоара до потрошача | Прекид транспорта пијеће воде услед квара |
| | | Контаминација пијеће воде |
| | | Крађа воде |
| Продаја | Очитавање водомера | Тачност и учесталост очитавања |
| | Утврђивање растура у потрошњи | Растур код блок водомера |
| | Класификовање потрошача у категорије | Провера да ли потрошач одговара својој категорији |
| | Задовољство потрошача обрачунатом потрошњом и надзор испуњења обавеза ЈКП БВК | Немогућност провере испуњења обавеза ЈКП БВК |
| Скупљање канализације | Одржавање атмосферске канализационе инфраструктуре | Немогућност утврђивања приоритета одржавања сливника на значајну промену атмосферских прилика |
| | Раздвајање атмосферске и отпадне канализације | Немогућност контроле раздвајања атмосферске и отпадне канализације |

5.3.2.3.1. Проблем контаминације извора

Као што је раније описано захватањем воде са изворишта започињу примарне активности ланца вредности ЈКП БВК. Да би се процес прераде сирове воде спровео адекватно и да би се крајњим корисницима испоручила пијећа воде прописаног квалитета, важно је спроводити контролу воде на изворишту, која је се налази на улазу процеса прераде воде. Потребно је перманентно вршити проверу квалитета сирове воде и уколико се утврди њена контаминација, спровести активности у спречавању прераде такве воде или увести додатне активности у прераду такве воде.

Из поменутих разлога ЈКП БВК, врши узорковање и проверу квалитета и сирове воде. То се спроводи на мануелни начин у лабораторији. Такав начин рада одузима много времена, а такође његов недостатак јесте што не постоји систем раног упозоравања који би се омогућило да се спроведу активности у превенцији потенцијално великих последица на заједницу.

5.3.2.3.2. Проблем воде која не доноси приход

Поменути проблем се односи на пијећу воду од које предузеће не може да приходује, а састављена је скупа проблема, где се неки односе на цурење у дистрибутивној мрежи водовода, затим у читавању самих водомера, крађи воде и тд. Поменути проблеми доводе до тога да постоје губици произведене воде која се испоручује у дистрибутивни систем, а који расту годинама.

Водоводи произведу одређену количину воде за пиће, али због проблема и аномалија у транспорту кроз дистрибутивни систем долази до губитка исте. Губици пијеће воде су изузетно високи и према истраживању спроведеном 2014. године они у Републици Србији износе око 32% и имају тренд пораста (Јовановић et al., 2014). Према поменутом истраживању, у Републици Србији губици су већи него у земљама Европске Уније, где су у Немачкој 6,8%, Шведској 17%, Шпанији и Великој Британији 22%, а у Словачкој 27%, Румунији 31% и Бугарској чак 50%. Такође, области са највећим губицима су Рашка и Колубарска област са преко 50%, а као пример је Нови Пазар са губицима 72% и Лајковац 66%, док Златиборска и Јабланичка област имају губитке од 21% до 30%, а веома високе вредности губитака имају Бајина Башта 64% и Бојник чак 82%. Према поменутом истраживању губици ЈКП БВК износе око 30%.

Погледајмо један пример који илуструје поменути проблем. Претпоставимо да желимо да утврдимо колико воде истиче услед квара чији је пречник 6 mm, на цеви под притиском дугој 60 m. За један сат истекне 1.8 m^3 , а 1300 m^3 за месец дана, што значи да ће количина воде која истиче за мање од 2 месеца напунити олимпијски базен, чиме се задовољају дневне потребе за водом 317 људи (Ziegler et al., 2009).

Иако може деловати да су губици воде искључиво инжењерски проблем или економски или проблем менаџмента и операционализације активности, пре би се рекло да они настају као последица много комбинованих фактора и међусобних дејстава поменутих, па проблем захтева мултидисциплинарно посматрање, што је учила и Светска здравствена организација, сумирајући их у 4 групе као: ресурсе, стање инфраструктуре, однос институција и политика контроле цурења (Farley, 2001).

У свакој земљи различито се перципира однос према води као кључном ресурсу који постоји у ограниченим количинама, где одређене земље или регије имају оскудицу воде и пажљиво приступају при употреби ресурса, док неке друге имају изобиље и немају у потпуности рационалан однос приликом потрошње, па њихове операције не подстичу извршност у управљању овим ресурсом.

Обзиром да вода пролази кроз дистрибутивни систем, који један сложен хидраулички систем састаљен од разних инфраструктурних елемената, јасно је да од степена текућег и инвестиционог одржавања зависи колики ће ниво губитака воде бити у систему. Одговорно управљање, које је суштински ограничено горе поменутим ресурсима требало би да садржи компоненту одлучивања, која може помоћи доносиоцима одлука, тј. менаџменту водовода где, када и како ће инвестирати ограничена средства у инфраструктуру.

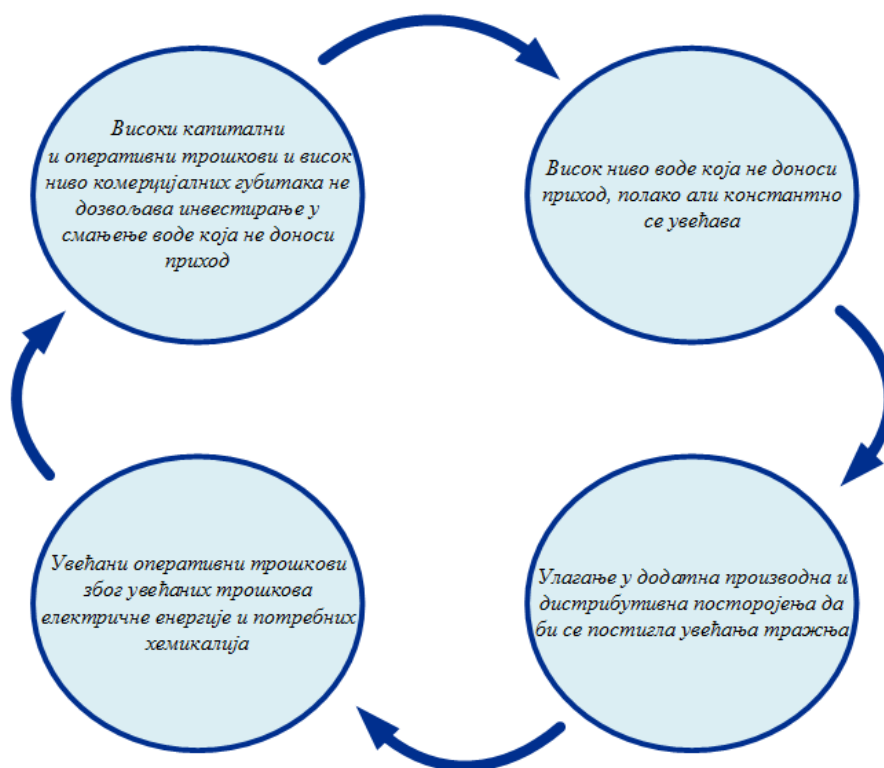
Проблем који посматрамо надилази водоводе јер се ради о најзначајнијем ресурсу без којег је живот немогућ и нечињење у смеру његовог очувања и одрживог развоја може да има разорне последице, па се решава у кооперацији активности на неколико нивоа, на нивоу државе, локалне заједнице и водовода. Није довољно да водоводи покажу извршеност операција и спремност да реше поменуте проблеме који захтевају и значајна улагања (која водоводи самостално не могу у потпуности да спроведу), али и деловање на свест потрошача о рационалној употреби пијеће воде. Из поменутих разлога део овог проблема не само што треба да буде подржана до нивоа државе, већ и да буде инкорпориран у форми која јасно, прецизно и конкретно уређује поменуту област.

Када се поставе јасне смернице од највишег нивоа до оперативног, за који су задужени водоводна предузећа, потребно је спровести одговарајуће активности у циљу решавања проблема губитака воде. Обзиром да је проблем сложен и да се састоји од скупа међусобно повезаних проблема, које је потребно засебно сагледати, а мере и активности усмерене ка њиховом решавању или минимизирању уредити на интегрални начин, планирање, оранизовање, координација и контрола активности је од пресудног значаја. Приступ који се усвоји треба да буде довољно флексибилан, у погледу разумевања проблема и потребе за свеобухватним деловањем зато што разни делови водоводне дистрибутивне мреже имају различиту динамику, па је потребно разумети ту динамику, прилагодити се и спровести специфичне активности којима ће се на адекватан начин проблем решити. Информационе технологије имају кључну улогу у реализацији овог приступа, јер се њиховом употребом могу не само надледати делови мреже засебно и као целина, из прикупљених података разумети њихово понашање, већ се и може предвидети са одређеним степеном сигурности, где ће проблеми у будућности настати. Поменути сервиси могу да дају увиде који имају кључну улогу у усмеравању активности у решавању проблема.

Водоводна предузећа спроводе разне активности у циљу да смање губитке воде. Као последица недостатка детаљаног, структурираног и свеобухватног приступа у погледу

деловања на губитке воде, чија је динамика значајно другачија од динамике спровођење изолованих а недовољно ефективних активности, догађа се да се инвестирају значајна финансијска средства, а да њихов ефекат је веома мали и често непрепознатљив.

Поменути проблем, назива се и зачарани круг воде која не доноси приход (енг. *Vicious NRW Cycle*), а односи се на ситуацију када водоводна предузећа инвестирају у увећање производње воде повећавањем захвата воде из изворишта и развој дистрибутивне мреже, а те инвестиције нису ефективне у погледу смањења губитака воде (Frauendorfer & Liemberger, 2010, pp. 9–11). Зачарани круг воде се може видети на слици 34.



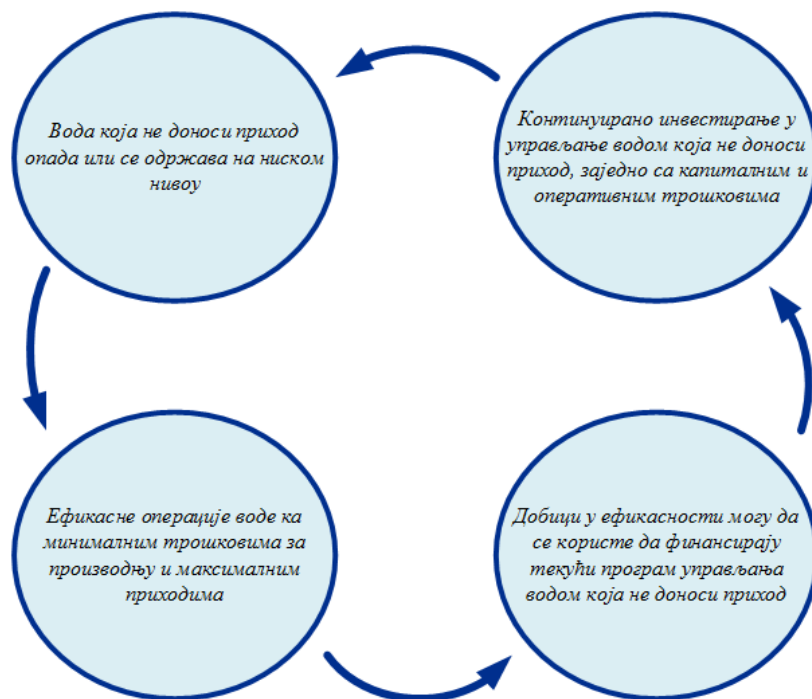
Слика 34. зачарани круг воде која не доноси приход, преузето (Frauendorfer & Liemberger, 2010)

Ситуација временом постаје све гора у погледу све веће количине воде која не доноси приход, која настаје као последица зачараног круга у који водоводи упадају, јер не решавају свој основни проблем.

Како аутори наводе „ни један посао не може опстати дуго уколико изгуби значајан део производа који продаје, али то је тачно оно што се догађа многим водоводним предузећима“ и према њима излаз из поменутог ситуације лежи у трансформисању зачараног круга у виртуозни круг воде која не доноси приход (енг.

Virtuous NRW Cycle) (Frauendorfer & Liemberger, 2010, pp. 9–11). Виртуозни круг се може видети на слици 35.

За детаљно разумевање структуре губитака воде, тј. воде која не доноси приход (енг. *non revenue water*, NRW), користи се водни биланс, предложен од стране Међународне асоцијације за воду (*International Water Association*, IWA), а који је прихваћен од стране великог броја водовода (Babić & Djukić, 2011).



Слика 35. виртуозни круг воде која не доноси приход, преузето (Frauendorfer & Liemberger, 2010)

Табела 7. Водни биланс, преузето (Babić & Djukić, 2011)

| | | | | |
|-----------------------------------|------------------|------------------------|-----------------------------------|----------------------------|
| Доток воде у дистрибутивни систем | Легална потрошња | Фактурисана потрошња | Фактурисана измерена потрошња | Вода која доноси приход |
| | | | Фактурисана неизмерена потрошња | |
| | | Нефактурисана потрошња | Нефактурисана измерена потрошња | Вода која не доноси приход |
| | | | Нефактурисана неизмерена потрошња | |
| | Губици воде | Привидни губици | Нелегална потрошња | |
| | | | Грешке у мерењу потрошње | |
| Грешке у обради и руковању | | | | |

| | | | |
|--|--|----------------|--|
| | | | подацима |
| | | Стварни губици | Процуривања у транспортним и дистрибуционим цевима |
| | | | Процуривања и преливања из резервоара |
| | | | Процуривања на прикљчцима до места водомера |

Елементи водног биланса су описани у наставку (Ziegler et al., 2009). Доток воде у дистрибутивни систем, је количина измерене вода која се након процеса производње испоручује у водоводну дистрибутивну мрежу, а за коју водоводно предузеће претпоставља да ће у највећој мери приходовати.

Део воде у дистрибутивној мрежи водовода који се допрема потрошачима назива се легална потрошња. Легалну потрошњу чине легална фактурисана потрошња и легална нефактурисана потрошња. Легална фактурисана потрошња може бити измерена потрошња и неизмерена потрошња и заправо то је количина воде која доноси приход, а за коју потрошачи плаћају водоводном предузећу. Фактурисана измерена потрошња настаје, тако што се читава мерни инструмент, водомер и по тој потрошњи се формирају фактуре, док у ситуацији када водомер из било којих разлога не може да се прочита тада се испоставља рачун најчешће по дневној просечној потрошњи и то је заправо нефактурисана измерена потрошња. Дакле, фактурисана измерена потрошња и фактурисана неизмерена потрошња чине воду која доноси приход, при чему неки водоводи приходују и од извезене воде, уколико је тако нешто део њиховог пословања.

Легална нефактурисана потрошња, такође може бити измерена потрошња и неизмерена потрошња и то је део воде која не доноси приход. Водовод може испоручити воду одређеним странама која за то неплаћају рачун. Примери укључују воду која се користи са хидраната, где ватрогасне службе користе део воде, као и службе градске чистоће и градско зеленила за обављање својих активности. Такође само водоводно предузеће користи воду за неке своје оперативне активности, као што је нпр. испирање цевовода или прање филтера.

Део воде која се налази у дистрибутивној мрежи водовода, а која није допремљена до потрошача називају се губици воде и они су део воде која не доноси приход. Уобичајена њихова подела је на стварне и привидне губитке. Стварни губици,

који се још називају и физички губици су део воде која не стигне до потрошача, јер се изгуби приликом транспорта у дистрибутивном систему. Процуривања у транспортним и дистрибутивним цевоводима, процуривања и преливања из резервоара, као и процуривања на прикључцима до места водомера чине стварне губитке. Привидни или комерцијални губици такође имају удела укупним губицима воде и њих чине нелегална потрошња (крађа воде), грешке које настају у процесу мерења, као и грешке које настају приликом обраде и руковања подацима. Дакле, вода која не доноси приход састоји се од губитака воде и нефактурисане потрошње, а заједно са водом која доноси приход чине целокупну воду која се налази у дистрибутивној мрежи водоводног предузећа.

Водни биланс може да се сумира у следећој формули:

Вода која не доноси приход = Укупна вода која је ушла у дистрибутивни систем - Вода која доноси приход (фактурисана потрошња) = Нефактурисана потрошња + Привидни губици + Стварни губици

Водни биланс је уједно и први корак које свако водоводно предузеће је потребно да учини да би кренуло у стратешку борбу против воде која не доноси приход, али важно је спровести и све наредне кораке, у једном интегрисаном приступу смишљених, детаљно испланираних и спроведених активности које треба да обухвати и делује на све елементе воде која не доноси приход, тј. нефактурисану потрошњу, привидне губитке и стварне губитке. Изоловани приступ неће бити довољно ефикасан и ефикасан по овом веома често занемареном или недовољно схваћеном питању, које мимоилази чињеницу да је пијећа вода најважнији ограничени ресурс.

Стварни губици настају услед кварова на цевима, спојевима и затварачима, као и услед преливања и процуривања на резервоарима, а могу се класификовати према величини и дужини трајања на: пријављена или видљива процуривања, непријављена или скривена процуривања и позадинска процуривања (Ziegler et al., 2009). Код пријављеног, тј. видљивог процуривања, вода избија на површину кратко траје и човеково око може детектовати квар, па најчешће грађани пријаве квар, а водоводно предузеће пошаље тим људи да га оправи. У поменутом случају грађани учествују у процесу пријаве квара и они су први корак у његовој детекцији. Непријављено, тј. скривено процуривање, не може се регистровати голим оком, јер се вода не појављује на површини. Водоводно предузеће најчешће ангажује своју службу, која је опремљена специјалном опремом за детекцију процуривања и употребом опреме долази се до идентификовања квара. Позадинска процуривања су обично малог обима и тешко их је детектовати, а по својој природи трају дуго, па се њихово отклањање најчешће спроводи приликом замене водоводне мреже.

Привидни губици као што смо већ рекли чине нелегална потрошња воде, грешке које се јављају приликом мерења и грешке у руковању и обради података (Ziegler et al., 2009). Нелегална потрошња је веома честа у земљама у развоју, код којих регулатива у том погледу није дефинисана. Комунално предузеће има и циљ да услугу испоручи под једнаким условима свим потрошачима, при чему нелегална потрошња нарушава поменути принцип, а уједно и предузеће не може да приходује, или може да приходује као у случају блок водомера при чему терет нелегалне потрошње сноси сви регистровани потрошачи који се налазе иза блок водомера. У том случају потрошач иако није потрошио стварну количину воде, добија да плати део потрошње нелегалног потрошача који производи растур (који може да увећа постојећи растур саме мреже узрокован разним процуривањима а које не одржава водовод).

Водомери су мерни уређаји који служе да измере количну потрошене воде и они могу да имају грешку, тј. да не мере довољно добро и прецизно количину воде коју потрошачи искористе. Из поменутих разлога водоводна предузећа спроводе поступак баждарења водомера, а на одређени временски период морају да изврше њихову замену.

Грешке у руковању подацима могу да настану, тако што запослени у водоводном предузећу чији је задатак да очита водомер, може да погреша приликом читавања водомера и та грешка може утицати на величину износа на рачуну за воду потрошача.

Такође, износ на рачуну потрошача може да буде неправилно исказан, уколико се поступак фактурисања не спроведе како треба. Поменути поступак се обавља најчешће употребом рачунара и дела пословног информационог система предузећа.

5.3.2.3.3. Мапирање примарних активности у ланцу вредности ЈКП БВК и чиниоца воде која не доноси приход

За успешно смањивање воде која не доноси приход, водоводна предузећа је потребно да спроведу интегрисани скуп активности који се односи на смањење стварних и привидних губитака. Како су оне повезане са активностима примарног ланца вредности, тј. део су њих, потребно је утврдити које активности унутар ланца вредности је потребно побољшати, односно изменити, тако да се оствари смањење воде која не доноси приход. Поменуто мапирање, је дато у табели 8.

Табела 8. тапирање примарних активности у ланцу вредности ЈКП БВК и чиниоца воде која не доноси приход

| | Активност | Проблеми и изазови | Губици |
|--------------------------|--|--|---|
| Дистрибуција сирове воде | Пренос сирове воде до фабрике за прераду | Предкид транспорта сирове воде услед квара | <ul style="list-style-type: none"> видљива, невидљива и скривена процуривања |

| | | | |
|---------------------------------|---|---|---|
| Дистрибуција пијеће воде | Пренос пијеће воде од резервоара до потрошача | Прекид транспорта пијеће воде услед квара | <ul style="list-style-type: none"> видљива, невидљива и скривена процуривања |
| | | Краја воде | <ul style="list-style-type: none"> нелегална потрошња |
| Продаја | Очитавање водомера | Тачност и учесталост очитавања | <ul style="list-style-type: none"> грешке у мерењу потрошње водомера грешке у руковању подацима |
| | Утврђивање растура у потрошњи | Растур код блок водомера | <ul style="list-style-type: none"> нелегална потрошња |

Дакле, деловањем на смањење стварних и привидних губитака воде, делује се на побољшање примарних активности унутар ЈКП БВК и тим се ефективно и ефикасно обављају пословне активности, које доводе до смањења удела воде која не доноси приход, а увећања удела воде која доноси приход у укупној количини воде коју испоручује водоводна дистрибутивна мрежа.

Уколико водоводно предузеће не предузме активности у смањењу губитака воде стварају се проблеми који утичу на нарушавање активности унутар ланца вредности. Губицима воде не нарушава се само активност дистрибуције или продаје воде, већ улазне и излазне вредности свих ранијих активности ланца вредности се губе и постају ништавне, а такође се утиче и на активности које се налазе након поменутих у ланцу вредности. Дакле, у губицима воде, поред тога што сам потрошач не добија довољну количину воде или има испрекидану испоруку, садржани су трошкови хемијских материјала, енергетски трошкови, трошкови људске снаге и трошкови радног искоришћења производних и дистрибутивних погона, као и додатни енергетски трошкови, обзиром да већина губитака заврши у канализацији, па је потребно веће искоришћење њеног капацитета.

Такође уколико је неко предузеће у својем ланцу вредности, користи воду као улаз (нпр. производња леда), то значи да прекиди у водоснабдевању узроковани губицима неће само нарушити ланац вредности самог водоводног предузећа, већ ће деловати и на ланце вредности других предузећа, а они ће опет деловати на ланце вредности предузећа које користе њихове производе и услуге као свој улаз (нпр. то је ледара, за ресторане).

5.3.2.3.3.1. Последице губитака воде

Постоје разне димензије губитака воде, а аутори наглашавају економске, техничке, друштвене и еколошке (Ziegler et al., 2009). Све ове димензије указују да су губици воде веома велики проблем и да имају многобројне негативне реперкусије.

Економска димензија је везана за трошкове који се издвајају за поправку елемената водоводне дистрибуционе мреже, као и потенцијално уништене околне инфраструктуре или друге учињене штете. Понекад се дешава да превозна средства која се налазе на путу буду оштећена услед хаварије која узрокује пуцање и пропадање асфалта и пута или неких других делова саобраћајне инфраструктуре. Поред поменутог, постоји много трошкова који су обухваћени губицима воде, а то су: трошкови приликом извлачења воде, њене прераде и транспорта и спречавање водоводног предузећа да оствари приход за пијећу воду.

Техничка димензија се огледа у немогућности испуњења параметара под којим је потребно испоручити воду потрошачима, где долази до феномена испрекидане испоруке воде, што проузрокује улажење ваздуха у цеви и стварање додатних оперативних трошкова у одржавању остатка инфраструктуре. Тражњу за водом је немогуће задовољити, па се уједно и смањује досег и покривеност мреже.

Друштвена димензија је везана за здравствене ризике који настају у испрекиданој испоруци, могућности да вода из канализације контаминира постојеће водоводне цеви и произведе здравствене проблеме потрошачима. Поред тога, нарушава се и једнакост снабдевања и задовољство потрошача услед смањених вредности притиска и протока, као и повремених прекида у снабдевању.

Еколошка димензија одређена је и тиме да су водоводна предузећа веома велики потрошачи, некада и највећи у граду, електричне енергије потребне за рад постројења и црпних станица. Када се проблем губитака третира тако да уместо да се губици санирају на један прихватљив ниво, већ се улаже у повећање производних капацитета, а и увећања извлачења воде, то доприноси већој емисији угљен диоксида услед потребне додатне енергије. Уз поменуто изворишта се додатно црпе јер се извлачи већа количина воде него што је потребна.

5.3.2.3.3.2. Одржив развој

Једна веома важна тема јесте јака повезаност одрживог развоја и губитака воде, где је овај однос обрнуто пропорционалан. Наиме, вода покрива 71% наше планете, а само 1% воде је пијећа вода, надземна и подземна. Око 40% људи на планети земљи живи у

оскудици воде. Према предвиђањима УН свака четврта особа ће 2050. бити погођена несташницом воде. Услед величине и озбиљности проблема, а све у циљу да се заштити овај најважнији ограничен ресурс, УН су у склопу Агенде одрживог развоја 2030. у Циљ 12 ставиле одговорну потрошњу и производњу воде, док се део тог циља односи на одрживу и ефикасну употребу природних ресурса (Radenković et al., n.d.). Одржив развој се може дефинисати као развој који задовољава потребе садашњости без угрожавања способности будућих генерација да задовоље своје потребе (Environment Engineering Group, n.d.). Глобалне промене под утицајем раста популације, климатских промена, урбанизације, ширења инфраструктуре, миграција, конверзије земљишта и загађења смањују доступност воде постављају све веће притиске у управљању воденим ресурсима (UNESCO, 2011). Према пројекцији УН, око 70% људи ће живети у урбаним зонама, у односу на данашњих 55%, што води све већем искоришћењу инфраструктуре водоводних предузећа (United Nations, n.d.). Све наведене чињенице указују да су губици воде једна од највећих претњи по одржив развој и због тога је потребно спровести опсежне мере како би се поменути проблем умањено.

5.3.2.3.3.3. Потреба за управљањем губицима воде

Смањење губитака воде је циљ сваког водоводног предузећа и води ка побољшању економске и еколошке ефикасности и бољим услугама за потрошача (Thornton et al., 2008).

Посматрано из угла водоводних предузећа, постоји најмање 10 разлога који могу да оправдају увећане трошкове управљања губицима воде (Ziegler et al., 2009):

1. ефикасност оперативних трошкова - добро одржаван систем за дистрибуцију воде ће захтевати мање поправки, имаће мање трошкове производње и спречиће плаћање компензација
2. ефикасност трошкова капитала - недостатак одржавања и повремених радова увећава хабање цеви, затварача и мерача. Побољшано снабдевање ће продужити живот компоненти система и водити ка смањеним фиксним трошковима водоводног предузећа на дуги рок.
3. побољшано мерење и наплата - мање цурења и побољшано снабдевање такође могу да имају позитиван ефекат на смањење привидних губитака јер ваздух унутар дистрибутивног система може да проузрокује грешке у мерењу
4. смањени здравствени ризици - канализација и други загађивачи могу да доспеју у ценовод путем цурења и да проузрокују болести настале од воде у

системима са смањеним притиском или у случају повремених радних операција.

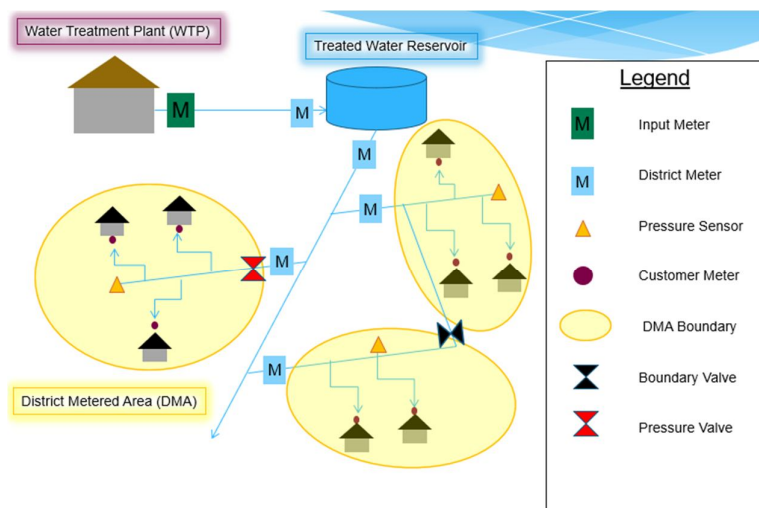
5. побошана сигурност снабдевања - добро одржавани систем са мање цурења и пуцања ће побољшати загарантовано снабдевање
6. мања оштећења инфраструктуре - цурење може да креира празнине испод земље која могу да воде до уручавања путева и колапса грађевина
7. смањено оптерећење канализације - губици воде доспевају у канализацију и тиме додатно оптерећују канализациони цевовод и фабрике за прераду отпадних вода
8. побољшано задовољство потрошача - као додатак лошем квалитету воде, неадекватној количини и здравственим ризицима, цурења такође доводе до опадања притиска на тачећим местима потрошача. Унапређење услуге снабдевања ће побољшати задовољство купаца и њихову спремност да плате.
9. публицитет и спремност за плаћање - мање кварова, побољшана сигурност снабдевања и хигијенски услови ће унапредити јавну перцепцију водоводног предузећа. Ово такође може да има позитивне ефекте на спремност потрошача да плаћају.
10. Смањени еколошки стрес - развој стратегија за смањење водоводних губитака има смисла са еколошке тачке гледишта. У случају оскудице или претераног искоришћења водених ресурса, губици воде би требало да буду смањени да би умањили притисак на ове ресурсе.

Сви наведени разлози указују да је потребно омогућити ефикасније начине управљања губицима воде, него што су до сада у пракси примењивани.

5.3.2.3.4. Случај ЈКП БВК - обављање активности које утичу на чиниоце воде која не доноси приход (смањење губитака)

Размотрићемо начин на који ЈКП БВК обавља своје активности у циљу смањивања губитака. На сваки пријављени квар од стране потрошача у контакт центру, утврђује се приоритет квара и квар се распоређује одређеном погону који уобичајено покрива један део града или општину, некада и део општине. Затим ЈКП БВК шаље тим људи који санира поменути квар. Уобичајено је да они прво затворе одређени део дистрибутивне мреже на основу шеме затварача која се добија из ГИС-а, а затим се ради на санацији квара и користе се људска снага и веома често потребна механизација. Када санација траже дуже од 12 сати, БВК обезбеђује пијећу воду путем привременог

мобилног резервоара у виду цистерне, преко којих потрошачи могу да се снабдеју пијећом водом. Након завршетка поменутог процеса поправке, измене на дистрибутивној мрежи се уносе у ГИС. За потребе да се утврди стварни губици, поред редовне оправке кварова, спроводи се поступак поделе дистрибутивне мреже на основне зоне билансирања (енг. *district metered area*). Основна зона билансирања, служи да се одређена просторна област дистрибутивне мреже изолује од остатка мреже да би могло да се утврди да ли постоје губици воде у том делу.



Слика 36. основна зона билансирања, преузето (Gannon, 2016)

Она се креира најчешће уз употребу затварача, мада некад може бити и физичким одвајањем цеви (Ziegler et al., 2009). Количина воде која улази у основну зону билансирања и која излази се мери и периодично анализира да би се утврдили губици. Поделом на зоне билансирања стварају се услови да се изврши боља контрола губитака воде, као што је приказано на слици 36. (Gannon, n.d.).

Након што су се успоставиле основне зоне билансирања у ЈКП БВК, задужене службе имају задатак да утврде потенцијална цурења. То се спроводи тако што се опремом за детекцију кварова спроводи испитивање унутар зоне билансирања првенствено да се пронађу кварови који нису пријављени или они који нису видљиви. Опрема за детекцију је најчешће заснована на ултразвучној детекцији квара. Радник са опремом за детекцију испитује цевовод зпотребом опреме и уколико постоји цурење он га бележи.

Када се утврди да одређени део дистрибутивне мреже има много цурења, тада се спроводи реконструкција и санација тог дела дистрибутивне мреже. На тај начин водовод мења инфраструктурну мрежу и утиче на смањење губитака. То је инвестициони подухват.

Водовод има обавезу да читава водомере потрошача 2 пута годишње, али то се спроводи најчешће 4 пута годишње. Процес читавања водомера се спроводи, тако што читач чита стање на водомеру и након тога путем апликације унесе стање на водомеру путем PDA уређаја. Уколико не може да прочита водомер, услед тога што му је онемогућен приступ (пас се налази у дворишту или је возило паркирано изнад шахте и многи други разлози), тада он бележи ту информацију, а процес фактурисања се спроводи на основу просечне потрошње, која може да се разликује од стварне потрошње.

Водомер је механички уређај који мери потрошњу у воде и временом се квари, па је потребно да предузме спроведе активности калибрације и замене водомера. Такође дешава се из неколико разлога да водомер не мери тачно потрошњу воде, већ приликом мерења постоји одређена грешка, а како водомер стари та грешка може да се увећава и тиме може да се произведене велики комерцијални губитак. Водовод има своју баждарницу, која служи за поправку водомера и њихово баждарење.

Такође један од проблема који се јавља су нелегални потрошачи, чија потрошња такође утиче на привидне губитке. ЈКП БВК има службу која проверава и контролише поменуте потрошаче и уз сарадњу са државним органима уводи их у легалне потрошаче.

На основу анализе активности које утичу на чиниоце воде која не доноси приход (смањење губитака) може се закључити следеће:

ЈКП БВК улаже напоре да се избори са водом која не доноси приход, оправком кварова, реконструкцијом дела мреже, креирањем основних зона билансирања и постављања мерне опреме, спровођењем мануелне детекције кварова употребом опреме, набавком нових и поправком старих водомера, увођењем нелегалних потрошача у систем. Ове активности спроведе различите организационе јединице у предузећу, најчешће изоловано од других активности које треба да утичу на смањење стварних и привидних губитака.

На описани начин не може се у целости сагледати проблем, састављен од мањих и испреплетаних проблема, па самим тим и спровести опсежан план активности који би требало да делује на све његове саставне елементе (чиниоце). С'тога једно од решења можда треба потражити у интегралном приступу, којим се

пре свега омогућава надзор делова система, боље разумевање свих елемената воде која не доноси профит, ствара оквир за доношење одлука и ефективно и ефикасно искоришћење ограничених средстава. У таквом једном решењу информационо-комуникационе технологије имају кључну улогу, а интегрисањем појединачних решења може се обезбедити основ за спровођење интегралног (свеобухватног) приступа у решавању проблема губитака.

5.3.2.3.5. Проблем изливања кишне канализације

За кишну канализацију је значајно да се ниво атмосферске воде прати у сливницима, да би могло да се спречи изливање те воде на улицу и потенцијално угрожавање безбедности људи у саобраћају. Често се догађа да долази до зачепљења сливника, услед несавених грађана који бацају разне форме отпада у њих, као и изливања сливника услед огромних количина падавина. У ситуацији када постоје велика количина падавина, може се лако догодити да се догоди нежељени догађај у виду изливања воде на улицу чиме се доводе у опасност људски животи, имовина и остала инфраструктура.

5.3.2.4. Информационе технологије у решавању проблема и изазова

5.3.2.4.1. Рачунарство у облаку

„Рачунарство ће једног дана можда бити организовано као јавни сервис баш као што је телефонски систем јавни сервис.“ су речи Џона Мекартија 1961. године, професора Масачусетског технолошког института (Jadeja & Modi, 2012).

Традиционални начин у испоруци ИТ услуга у почетку је реализован тако да се комплетна опрема налази унутар рачунарског центра објекта предузећа тзв. (енг. *on-premise*). Умрежавањем се могу бавити запослени унутар предузећа или се ангажује посебна предузећа којима је то посао, тзв. пружаоци мрежних услуга (енг. *network service provider*). Опрема које је потребна се наручује, а она се плаћа једном и таква врста улагања производи трошкове основних средстава. Када стигне она се уграђује у рачунски центар, уобичајено у раф (енг. *rack*), затим се инсталира и конфигурише. Након тога се исти поступак понавља за нову опрему.

Обзиром да су трошкови оваквог начина рада високи и да захтевају куповину опреме одједном, временом су уведени колокациони центри (енг. *colocation center*) који су локације рачунског центра које власник тог објекта жели да изнајми спољним корисницима. Такви центри обезбеђују напајање, хлађење, мрежну опрему и тд. У предузећима се и даље налазе рачунари корисника, док се веза између њих и колокационог центра остварује посредством мрежних услуга коју реализују пружаоци

мрежних услуга. Сада је опрема и даље део трошкова основних средстава, а поред тога постоје и трошкови ренте колокационог центра и они се обично реализују на месечном нивоу и спадају у текуће трошкове. Уместо да трошкове опреме плати одједном, предузеће месечно плаћа текуће трошкове, док су сви послови одржавања рачунског центра у надлежности колокационог центра.

Економски гледано рачунарство у облацима се може описати као „претварање трошкова основних средстава у текуће трошкове“, док аутори (Armbrust et al., 2009, 2010) сматрају да је фраза „плати колико потрошиш“ (енг. *pay as you go*), прикладнија. Исти аутори додају да рачунарство у облацима се односи на апликације испоручене као услуге преко Интернета и хардвер и системски софтвер у центрима података који обезбеђују ове услуге. По NIST-у (Mell & Grance, 2011): „Рачунарство у облацима је модел који омогућава свеприсутан, одговарајући, на захтев доступан мрежни приступ дељеним конфигурисаним рачунарским ресурсима (нпр. мреже, сервери, складишта, апликације и сервиси) који се брзо могу обезбедити и ослободити са минималним напорима у управљању или интеракцијом провајдера услуге. Овај модел се састоји од пет главних карактеристика, три модела услуге и четири модела развоја“.

Хардверски гледано 3 аспекта су нова у рачунарству у облацима (Armbrust et al., 2009, 2010):

1. Илузија неограничених рачунарских ресурса доступних на захтев елиминишу потребу корисника услуга рачунарства у облаку да планирају далеко испред ради доступности поменутих.
2. Уклањање плаћања унапред корисницима рачунарства у облаку, омогућава предузећима да започну са малим и повећавају хардверске ресурсе само онда када постоји увећање њихових потреба.
3. Могућност да се плати на основу употребе рачунарских ресурса у кратком року уколико је потребно (нпр. процесори по сату и складиштење по дану) и да се ослободe када је потребно, чиме се подстиче чување, дозвољавајући машинама и складиштима да се уклоне када више нису потребни.

Главне карактеристике рачунарства у облацима су (Mell & Grance, 2011):

- самоуслуживање на захтев (енг. *on-demand self-service*) - корисник сам обезбеђује рачунарске ресурсе без потребе за интеракцијом са пружаоцем услуга

- широки мрежни приступ (енг. *broad network access*) - сервиси су доступни путем стандардних платформи мобилних телефона, таблета, преносивих рачунара и радних станице
- удруживање ресурса (енг. *resource pooling*) - пружалац услуга омогућава складиштење (енг. *storage*), процесирање (енг. *processing*), меморију (енг. *memory*) и мрежни проток (енг. *network bandwidth*) спајањем различитих физичких и вирuellних ресурса који се динамички додељују и отпуштају корисницима
- брза еластичност (енг. *rapid elasticity*) - у зависности од потребе корисника рачунарске могућности се еластично прилагођавају
- измерена услуга (енг. *measured service*) - употреба ресурса може се надгледати, контролисати и извештавати, па се на тај начин обезбеђује транспарентност за кориснике и пружаоце услуга

Модели услуга рачунарства у облацима су (Mell & Grance, 2011):

- софтвер као услуга (енг. *Software as a Service, SaaS*) - могућност да корисник да користи апликације пружаоца услуге покренуте на инфраструктури облака
- платформа као услуга (енг. *Platform as a Service, PaaS*) - кориснику је омогућено да инсталира на инфраструктуру облака кориснички креиране или купљене апликације креиране употребом програмских језика, библиотека, сервиса или алата које подржава пружалац услуге
- инфраструктура као услуга (енг. *Infrastructure as a Service, IaaS*) - могућности које су пружене кориснику се односе на обезбеђивање процесирања, складиштења, мрежних и других основних рачунарских ресурса где је кориснику омогућено да инсталира и покреће произвољан софтвер, који може да укључи оперативне системе и апликације

Модели развоја рачунарства у облаку су (Mell & Grance, 2011):

- приватни облак (енг. *private cloud*) - инфраструктура облака је обезбеђена искључиво за употребу од стране једне организације
- облак заједнице (енг. *community cloud*) - инфраструктура облака је обезбеђена искључиво за специфичну заједницу корисника
- јавни облак (енг. *public cloud*) - инфраструктура облака је намењена јавној употреби

- хибридни облак (енг. *hybrid cloud*) - инфраструктура облака је састављена од две или више различитих инфраструктура (приватне, заједничке, или јавне)

Постоје многоструке користи која предузећа могу да имају употребом инфраструктуре облака: исплатљивија употреба ресурса, глобална дистрибуција, оптималније искоришћене ресурса и тд.

5.3.2.4.2. Паметни градови и Интернет ствари

Визија модерног града, где је инфраструктура повезана електронски и његови становници партиципирају у сервисима који су омогућени, назива се паметни град (енг. *smart city*) (Musa, 2016). Не постоји јединствена дефиниција и назив (Cocchia, 2014; Hollands, 2008), а поред тога појам је посматран и изучаван са неколико димензија и мењао се током времена (Albino et al., 2015). Најопштије речено основна идеја јесте да се побољша квалитет људи, кроз боље и ефикасније услуге којима се задовољавају њихове потребе, а то је омогућено употребом технологије. Разна технолошка решења служе бољем управљању инфраструктуром и она побољшавају услуге локалне управе, комуналних служби, библиотека, болница, јавног транспорта, електрана и тд.

За остварење визије паметног града веома је важан појам интернет ствари (енг. *Internet of Things*, IoT), који се још назива и Интернет интелигентних уређаја. Може се рећи да интернет ствари су радикална еволуција тренутног Интернета у мрежу међусобно повезаних објеката који не само што преузимају информације из окружња (енг. *sensing*) и имају интеракцију са физичким светом (енг. *actuation/command/control*), већ користе постојеће Интернет стандарде да обезбеде услуге за пренос информација, аналитику и комуникације (Gubbi et al., 2013). Према ИТУ: „IoT се може видети као глобална инфраструктура информационог друштва, која омогућава напредне сервисе међусобним повезивањем (физичких и виртуелних) ствари заснованим на постојећим и растућим интероперабилним информационо-комуникационим технологијама“ (Zennaro, 2016).

Како ИТУ даље објашњава ствари (енг. *things*) је потребно да могу да се идентификују и интегришу у комуникациону мрежу, и да са једне стране оне могу бити објекти физичког света (физичке ствари), а да друге објекти информатичког света (виртуелне ствари), тако да физичке ствари постоје у физичком свету и могу да се осете, актуирају и повежу, док виртуелне ствари постоје у информационом свету и могу да се сачувају (складиште), процесирају и да им се приступи (Zennaro, 2016). Разлика међу њима јесте да виртуелна ствар може да постоји без репрезентације у физичком свету, док физичка ствар може да се репрезентује једном или више ствари у виртуелном свету,

процесом мапирања. На пример физичке ствари могу да буде део животне средине која нас окружује, индустријски роботи, добра и електрична опрема, док примери виртуелних ствари су мултимедијални садржај и апликациони софтвер.

Према ИТУ уређај (енг. *device*) је део опреме са обавезним могућностима комуникације и опционе могућности да осете, актуирају, преузму податке, складиште их и процесирају их. Са једне стране они могу да скупљају податке и прослеђују их мрежи за даље процесирање, а са друге могу да извршавају операције на основу информација добијених путем мреже (Zennaro, 2016). Сензори, актуатори, микроконтролери и рачунари на једној плочи чине интернет ствари.

Сензор (енг. *sensor*), детектор или давач је уређај који реагује на неки физички подстицај, као што је топлота, светлост, звук, притисак, влажност, проток и тд. и производи мерљиви електрични сигнал (Sitharama Iyengar et al., 2010).

Актуатори (енг. *actuators*) користи електрични улаз и конвертује га у неки други облик енергије, где се делује на окружење из система путем актуације.

Микроконтролер (енг. *microcontroller*) је мали рачунар са микропроцесором, меморијом и улазно / излазним портovima. Улазни портави скупљају податке са сензора, док излазни портави подржавају било коју потребну актуацију. Развој микропроцесора је једноставнији када се микропроцесор, меморија и улази/излази интегришу, а самим тим њихова цена опада (Dunko et al., 2017). Они имају ограничене могућности у виду операција које могу спроводити.

Са друге стране рачунари који су на једној плочи (енг. *single board computers*) имају много веће могућности у односу на микроконтролере, као и да подрже инсталацију одређеног оперативног система. Један од примера је Raspberry Pi.

За разлику од седмослојног ISO/OSI референтног модела комуникације, где су протоколи груписани у седам слојева, где сваки слој има одређене функционалности које нуди слојевима изнад себе, код интернет ствари ситуација је другачија.

Једно од предложених груписања је следеће (Somasundaram, n.d.):

- инфраструктура (нпр.: 6LowPAN, IPv4/IPv6, RPL)
- идентификација (нпр.: EPC, uCode, IPv6, URIs)
- комуникација / транспорт (нпр.: Wifi, Bluetooth, LPWAN)
- откривање (нпр.: Physical Web, mDNS, DNS-SD)
- протоколи за податке (нпр.: MQTT, CoAP, AMQP, Websocket, Node)

- управљање уређајем (нпр.: TR-069, OMA-DM)
- семантички (нпр.: JSON-LD, Web Thing Model)
- више-нивоски оквири (нпр.: Alljoyn, IoTivity, Weave, Homekit)

Можемо уочити велику хетерогеност која се испољава код комуникационих протокола који користе интернет ствари.

5.3.2.4.3. ARM и AMI технологије

Технологија аутоматског читавања (енг. *automated meter reading*, AMR), служи за читавање потрошње која је потребна ради испостављања рачуна потрошачима у комуналним предузећима. Када се користи ова технологија нема потребе за мануелним читавањем бројила, које је подложно људским грешкама у читавању, већ се врши конверзија аналогног у дигитални сигнал. Подаци могу да се пошаљу од бројила до комуналног предузећа путем телефонске линије, струјне линије, сателитом, каблом или путем радио фреквенција (techtarget, n.d.). Свако читавање се памти у меморији уређаја и када дође читач он може да преузме та читавања. Један од начина преузимања јесте да се читавање са бројила проследи путем радио фреквенција уређају примаоцу и који може да буде инсталиран у возило и тада се возилом прође зоном читавања где се налазе бројило и преузму се читавања са бројила. Уобичајено да је да су AMR-ови једносмерни уређаји, где комуникација иде од уређаја до примаоца.

Напредна мерна инфраструктура (енг. *Advanced Metering Infrastructure*, AMI) је напреднија верзија аутоматског читавања, која омогућава сложенији скуп функционалности, а неке од њих су везане за контролу уређаја, као и управљање тражњом (Cramer & Deconinck, 2010). AMI се називају и паметна бројила (енг. *smart meter*).

5.3.2.4.4. Бежичне сензорске мреже

Бежичне сензорске мреже (енг. *Wireless Sensor Networks*, WSN) у својој најједноставнијој форми могу да се дефинишу као мрежа (по могућству мале величине и мале сложености) уређаја који су означени као чворови (енг. *nodes*) који могу да прикупе податке из окружења и да их пренесу, из области која се надзире путем бежичних веза, где се подаци прослеђују по могућству путем вишеструких скокова (енг. *hops*) прослеђују синку (енг. *sink*) (који се понекад означава као контролер или монитор) и који може да их користи локално или да их повеже са другим мрежама (нпр. Интернетом) путем гејтвеја (енг. *gateway*) (Verdone et al., 2008). Основна намена WSN-а је прикупљање и достављање података и информација о окружењу мреже, у складу са потребама корисника мреже, а

једна од њених добрих карактеристика је могућност самосталног формирања ad-hoc бежичних мрежа (G. Marković & Dukić, 2009). Бројне примене укључујући детекцију поплава, биолошке и хемијске детекције, надзор агрокултура и разне друге примене у надзору окружења (Akyildiz et al., 2002). Посматрано са аспекта надзора технологија бежичних сензорских мрежа је веома важна и може унапредити ову врсту активности.

5.3.2.4.5. Велики подаци

Под термином велики подаци (енг. *big data*) обухвата се употребу технологија за прихватање, процесирање, анализирање и визуелно презентовање потенцијално великих скупова података у разумном временском периоду, који се не може спровести стандардним ИТ технологијама (NESSI, 2012). Технологије великих података се најчешће описују преко особина које се популарно називају В-ови (енг. *V's*). Постоји више поменутих особина, у почетку их је било 3 (Jagadish et al., 2014; Manogaran et al., 2017; Papadokostaki et al., 2017), затим 4 и 5, а неки извори наводе чак 10 (Manogaran et al., 2017).

Основне су (Papadokostaki et al., 2017):

- обим (енг. *volume*) - обим података стално расте, јер су подаци генерисани из различитих извора и стандардне базе података не могу да их обраде
- разноврсност (енг. *variety*) - различите врсте података се скупљају путем сензора, паметних телефона, друштвених мрежа и тд, и могу да буду текст, слике, видео записи у структурираном, полуструктурираном или неструктурираном формату
- брзина (енг. *velocity*) - односи се на брзину којом се врши аквизиција података, као и брзину путем које је потребно да они буду процесирани и анализирани

Постоје 2 начина обраде великих података (Kleppmann, 2016):

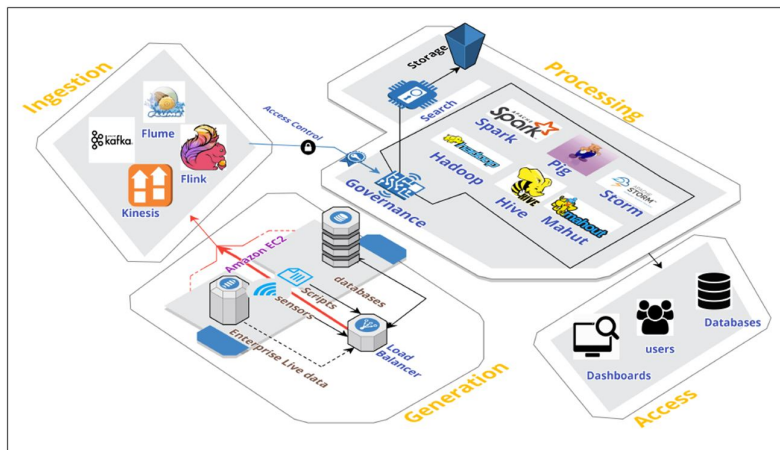
- систем са пакетним процесирањем (енг. *batch processing system*) - Систем са пакетним процесирањем, тј. обрадом узима велику количину улазних података, покреће задатке (енг. *job*) и производи неке излазне податке. Задаци трају, од неколико минута до неколико дана, тако да корисник не чека да се задатак заврши. Уместо тога, пакетни задаци су често заказани да буду покренути периодично, нпр. једном дневно.
- систем за процесирање токова (енг. *stream processing system*) - Процесирање токова се назива процесирање блиско реалном времену (енг.

near-real-time processing). Процесор тока, као код пакетне обраде узима улазе и производи излазе, али његов задатак се извршава кратко након што су се догађаји догодили, а главна разлика је да се задатак код пакетног процесирања се извршава на фиксираним скупу улазних података. Ова разлика омогућава системима за процесирање токова да имају мање кашњење него системи са пакетним процесирањем.

Уобичајено је да архитектура великих података има 4 слоја:

- узимање (енг. *ingestion*) - служи за прихватање података из различитих извора података који могу бити мобилни телефони, Интернет ствари, подаци са социјалних мрежа и тд.
- процесирање (енг. *processing*) - служи за процесирање података
- складиштење (енг. *storage*) - служи за складиштење података
- визуелизација (енг. *visualization*) - служи за приказ података у виду различитих графичких елемената

Не мора свака апликација да имплементира све поменуте слојеве (Pal et al., 2018), као на слици 37.



Слика 37. типична четворослојна архитектура, (Pal et al., 2018)

На сваком од поменутих нивоа постоје велики број софтверских решења и алата који се нуди, као и велики број понуђача услуга као што су: Amazon, Microsoft, Google, IBM, Cloudera и тд. Решења омогућавају да се користи и ламбда (енг. *Lambda Architecture*) и капа (енг. *Kappa Architecture*) архитектурални стилови, где за потребе овог рада само ће се споменути, а за детаљније видети извор (Forgeat, n.d.).

Apache Hadoop је софтверска платформа отвореног кода која омогућава дистрибуирано складиштење и дистрибуирано процесирање на кластерима рачунара који су изграђени од хардвера који је доступан на тржишту (енг. *commodity hardware*). у његовој основи налази се HDFS (енг. *Hadoop Distributed File System*), где се простор доступан за складиштење на свим рачунарима посматра као један велики фајл систем. Поред тога постоји одређени број копија који може да се конфигурише, тако да се на дистрибуираном систему подаци не чувају само на једном месту, већ постоји неколико њихових копија. Уколико дође до отказа, могуће је податке повратити и наставити са радом. На HDFS се инсталира YARN (енг. *Yet Another Resource Negotiator*) који омогућава да се управља ресурсима кластера машина. Поред тога он служи и за покретање задатака (енг. *job*), тј. програма. Следећа важна компонента Hadoop екосистема је MapReduce. То је програмски модел који омогућава масивно паралелно процесирање података преко целог кластера и састоји се од две функције: мапера (енг. *mapper*) и редукера (енг. *reducer*). Hadoop се првенствено користи за пакетну обраду података. Навешћемо још неколико технологија које се могу користити у Hadoop екосистему, али и ван њега.

Apache Spark функционише отприлике на сличном нивоу као и MapReduce, а може да се програмира у неколико програмских језика као што су: Scala, Python, Java и SQL. SQL је посебно важан јер омогућава да људима који желе да спроводе анализу са великим бројем података, не морају да пишу скрипте у програмским језицима, већ користе стандардни упитни језик са којим су они много фамилијарнији. Apache Spark омогућава спровођење и пакетне обраде, али и обраде токова података, што га чини јединственим оквиром, где није потребно користити многе друге технологије које су раније биле заступљене у Hadoop екосистему пре него што је Spark заживео. Има могућност да процесира податке у меморији, што га чини поприлично бржим од Hadoop модела.

Apache Kafka је платформа отвореног кода за процесирање токова података, која омогућава да се узме и процесира велика количина података у еалном времену. Добре карактеристике ове платформе су да је скалабилна, брза и да омогућава трајно складиштење података.

Обезбеђује три главне функције корисницима (Amazon, n.d.):

- функције Publish и Subscribe токова података

- ефективно складиштење токова податка у редоследу у којем су подаци генерисани
- процесирање података у реалном времену

Apache Flink је скалабилна платформа за дистрибуирано процесирање токова података. Омогућава високу пропусност и мало кашењење, као и високу расположвост, што је веома важно за процесирање токова. Када постоји велики број интернет ствари и када је ток података непрекидан треба имати у виду неколико чињеница: подаци су неограничени, интервали доласка нових података је немогуће предвидети и подаци могу да дођу ван редоследа којим су послати. Apache Flink омогућава да се поменути проблеми реше.

5.3.2.4.6. NoSQL базе података

У односу на релационе базе података, када се ради са великим подацима користе се NoSQL (not only SQL) базе података. На одређен начин оне напуштају ACID (Atomicity, Consistency, Isolation, Durability) особине типичне за релационе системе за управљање базама података и користе BASE (Basically, Available, Soft state, Eventually consistent) особине да би биле високо доступне и скалабилне, што је веома важно у рачунарству у облаку.

NoSQL базе примењују CAP (Consistency, Availability, Partition tolerance) теорему која указује да дистрибуирани систем може да испоручи само две од три жељене карактеристике: конзистентност, доступност и толеранција раздвајања. Дистрибуирани систем је мрежа која складишти податке на више од једног чвора било да је он физички или вирутелни у истим тренутку (IBM Cloud Education, n.d.).

Постоји неколико различитих типова ових база података:

- базе података засноване на концепту документа (нпр. CouchDB, MongoDB, Cosmos DB)
- базе податка кључ-вредност (нпр. Dynamo, Apache Ignite, FoundationDB)
- базе података са широкик колонама (нпр. Cassandra, HBase)
- базе података графова (нпр. Neo4J, Apache Giraph)

5.3.2.4.7. Детекција аномалија

Детекција аномалија (енг. *anomaly detection*) је проблем проналажења патерна у подацима који нису у складу са очекиваним понашањем, која може да укаже да ће се неки догађај догдити или се већ догодио. Ови неусуглашени обрасци често се називају

аномалије (енг. *anomalies*), аутлејери (енг. *outliers*), неусклађене опсервације (енг. *discordant observations*), изузеци (енг. *exceptions*), аберације (енг. *aberrations*), изненађења (енг. *surprises*), специфичности (енг. *peculiarities*) или загађивачи (енг. *contaminants*), у најразличитим областима примене (Chandola et al., 2009).

Употреба детекције аномалије је широка област, примењена је на разном доменима и комбинује знања из статистике, машинског учења, откривања законитости у подацима и тд. Детекција аномалије се може корисити у разне сврхе: од детектовања упада, превара, детектовања отказа или кварова, праћење здравља, детекције догађаја у сензорним мрежама, детекције догађаја у нарушавању екосистема и тд.

Технике детекције аномалије се уобичајено сврставају у 3 категорије: ненадгледана детекција аномалије (енг. *unsupervised anomaly detection*), надгледана детекција аномалије (енг. *supervised anomaly detection*) и полунадгледана детекција аномалије (енг. *semi-supervised anomaly detection*). У ненадгледане алгоритме за детекцију аномалије спадају: кластеровање употребом к-средњих вредности (енг. *k-means*), аутоенкодери (енг. *autoencoders*), анализа главних компоненти (енг. *principal component analysis* – PCA) и тд. У надгледане алгоритме за детекцију аномалије спадају надгледане неуронске мреже (енг. *supervised neural networks*), стабла одлучивања (енг. *decision trees*), алгоритам к-најближих суседа (енг. *k-nearest neighbors*) и тд.

5.3.2.4.8. Вештачка интелигенција и машинско учење

Вештачка интелигенција (енг. *artificial intelligence*) и машинско учење (енг. *machine learning*) су део рачунарске науке и веома су повезани међусобно. Иако се у последње време због велике популарности оба термина користе наизменично од стране великог броја људи, требало би имати у виду да нису исто. Вештачка интелигенција је шири концепт по којем је циљ креирање интелигентних машина који могу да симулирају способности људи да размишљају и да се понашају, док је машинско учење подскуп вештачке интелигенције који омогућава машинама да уче из података без потребе да се експлицитно програмирају (JavaTpoint, n.d.). Обе поља су веома значајна за концепт паметних градова, очување инфраструктуре и интелигентније управљање средствима.

5.3.2.5. Примена ИТ у решавању проблема и изазова у ланцу вредности ЈКП

5.3.2.5.1. Могућност примене ИТ на проблему контаминације извора сирове воде

Напредак у сензорским технологијама и технологијама бежичне комуникације омогућава да се реализује решење којем би се реализовало надзор квалитета воде на

изворишту. Као што Paska наводи, тренути проблем да се овакво решење усвоји јесте велика цена сензора којим би се омогућиле квалитетније анализе, обзиром да је за поменуто решење потребно поставити велики број таквих сензора (Paska, 2018). За сада једна од опција јесте да се користе сензори који мере неке једноставније параметре као што су: температура, проводљивост, рН вредност и растворени кисеоник.

Исти аутор наводи да је град Стокхолм спровео истраживање током 2017. и 2018. године у сарадњи са академском заједницом и применио поменуто решење на језеру Меларен, где су се на мерили основни параметри као што су: рН, температура, проводљивост, растворени кисеоник и редокс-потенцијал (Paska, 2018).



Слика 38. архитектура решења за надзор окружења, преузето (Paska, 2018)

Употребом LTE мреже сензори су комуницирали са Ericsson IoT платформом заснованој на технологији облака. Резултат прве фазе поменутог истраживања јесте да се веома тешко спровело закључивање на основу података са сензора и да се одговори да ли да је дошло до контаминације. Одлучено је да у следећој фази пројекта се примени мултивариациона анализа и вештачка интелигенција (енг. *artificial intelligence*, AI) које би омогућиле реализацију решења за надзор квалитета воде у потпуности. Архитектура решења је дата на слици 38.

ИТ технологије које омогућавају ово решење су: мрежа сензора, рачунарство у облацима као и аналитика омогућена статистичким методама и вештачком интелигенцијом, а такође је важан део и визуелизација.

5.3.2.5.2. Могућност примене ИТ на проблему детекцији квара на дистрибутивној мрежи водова

ЈКП БВК, користи SCADA (engl. *Supervisory Control And Data Acquisition*) систем који се користи у надзору разних параметара водоводног система и распоређен је да надгледа процесе од изворишта до процеса слања воде у дистрибуцију. Велики део водоводне дистрибутивне мреже се не надзире, не спроводи се континуирано праћење релевантних параметара и не анализира се да ли је дошло до неке нерегуларности. Један од разлога поменуге ситуације и тај што SCADA системи користе велике сензоре који

захтевају повезивање на напајање жицом и успостављају везу са централизованим системом, што отежава њихову уградњу и додаје значајне трошкове које потенцијално реализован систем за надзор дистрибутивне водоводне мреже треба да има (Berry et al., 2005). На тај начин водоводна дистрибутивна мрежа се не надзире у целисти и највише се поступа реактивно, врше се поправке кварова по пријави грађана.

У сврху решавања поменутог проблема могу се искористити технологије бежичних сензорских мрежа, технологије великих података и рачунарства у облаку, као и неки приступи у детекцији аномалија, да се креира такав систем који може да изврши детекцију квара. На тај начин се може створити могућност за бржу и прецизнију детекцију квара у односу на време пријаве од стране корисника, као и могућност детекције невидљивих цурења која до сада нису ни била пријављивана, као и краће време које је потребно за његово отклањање. На тај начин смањују се стварни губици, као и количина воде која не доноси приход.

Једна реализација поменутог сценарија може да архитектурално буде осмишљена на следећи начин. У циљу преузимања података о водоводној дистрибутивној мрежи, користиће се бежична сензорска мрежа, где ће се по један њен чвор уградити у елемент дистрибутивне мреже, нпр. цев. Поменути чвор, мери вредности притиска и протока у задатком временском интервалу и те вредности прослеђује путем главног чвора, локалном gateway-у, који даље прослеђује вредности локалној станици, или Fog кластеру.

Обзиром да ЈКП БВК већ има уређене основне зоне билансирања, један Fog кластер би могао да покрива поменуту зону и да срачунава да ли је унутар основне зоне билансирања, надгледањем параметра настала аномалија и уколико јесте да се креира догађај којим ће се крајњи корисници (радници на мрежном погону) обавестити о томе и да је потребно да предузму акције у циљу провере и отклањања потенцијалног квара.

Унутар Fog кластера спроводи се процесирање блиско реалном времену, употребом неког Big Data оквира, као што је нпр. Apache Spark, у којем је имплементиран алгоритам за детекцију аномалије. Приликом процесирања блиском реалном времену, процесор тока анализира вредности неког параметра и тек ако је у неком временском интервалу, задовољен услов којиме је описана аномалија, тада се креира догађај којим се обавештава да је аномалија настала.

Подаци заједно са аномалијама се прослеђују на облак (Cloud), где се памте и визуелизују, заједно са подацима и аномалијама из свих осталих основних зона билансирања преко њихових Fog кластера. Након настанка аномалије постоји услов за

предузимање одређених контролних радњи на мрежи и евентуалне санације квара или утврђивање нелегалне потрошње. У раду се неће наводити алгоритми детекције аномалија, обзиром да нису од већег интереса за истраживање наслеђених система, али их у теорији и пракси има доста. Преглед области је дат у (Т. К. Chan et al., 2018), а слично архитектурално решење као што је предложено овде преко локалних индикатора просторне аутокорељације, са временским проширењем је дато у (Difallah et al., 2013). Треба рећи да се сличан сценарио се највероватније може применити у детекцији квара на цевоводима сирове воде.

ИТ технологије које омогућавају ово решење су: сензори, интернет ствари, рачунарство у облацима, Fog рачунарство, велики подаци, статистичке методе, методе машинског учења, вештачком интелигенција, а такође је важан део и визуелизација.

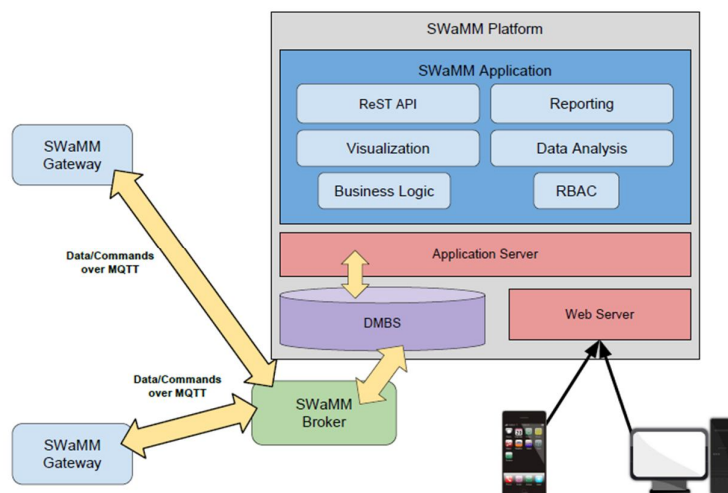
5.3.2.5.3. Могућност примене ИТ на проблему тачности и учесталости читавања, крађе воде и навика у потрошњи воде

ЈКП БВК користи водомере да би измерило количину угрошене воде. Водомер је уређај којем временом услед употребе смањује тачност мерења угрошене количине воде и мора да се промени на одређени временски рок од 5 година. Водомер се читава од 2 до 4 пута годишње. Читачи читавају водомер, тако што уносе стање на водомеру у ПДА уређај када могу да га читају. Када не могу да га читају потрошња се обрачунава преко просечне потрошње.

Описани проблеми везани за тачност и учесталост читавања, се могу решити увођењем паметних водомера (енг. *smart water meter*). Паметни водомери омогућавају да се читавање водомера спроводи на аутоматизован начин, без мануелног читавања, које је подложно људској грешки. Поред тога омогућава се већа учесталост читавања, па самим тим и бољи и прецизнији рачуни за потрошаче, јер се њиховом употребом смањује начин обрачуна преко просечне потрошње. Додатна погодност је и то што се елиминишу разни разлози нечитавања, да је пун шахт воде, да је возило паркирано изнад шахте и тд.

Једно од предложених решења засновано је дато у (Alvisi et al., 2019). Подаци о потрошњи се читавају путем паметног водомера. Након тога са шаљу у локални gateway, који је заправо са паметним водомером чини Edge целину. Након што се скупе подаци са околних водомера, gateway прослеђује податке на Cloud у интеракцији са компонентом брокера, који је у суштини технологија средњег слоја која са једне стране може да се обраћа Edge делу, а са друге да комуницира са базом података и осталим компонентама пре свега пословног софтвера које се налазе на Cloud. На описани начин могуће је аутоматски прикупљати податке о читавањима.

Поменути систем са одређеним проширењима и прилагођавањима се може користити и да аутоматски прикупља податке о потрошњи унутар основне зоне билансирања. Уколико је на улазу у зону билансирања уграђен паметни водомер, као и код свих регистрованих потрошача, поређењем потрошње са водомера на улазу у зону и потрошње са свих водомера потрошача унутар зоне, може се утврдити да унутар саме зоне постоје одређени проблеми и да негде настаје губитак воде. Такав губитак може да буде стварни губитак, јер се догађа неко цурење воде. Поред тога, може да се ради и о нелегалној потрошњи.



Слика 39. архитектура за читавање, преузето (Alvisi et al., 2019)

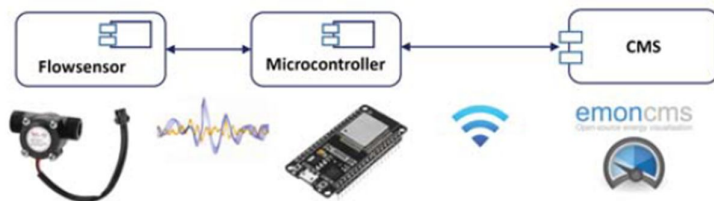
Овакво решење, такође се може користити као основа за утврђивање понашања потрошача. Наиме, сваком потрошачу приликом регистравања може се одредити одређени профил потрошње у односу на то којег је потрошач типа. Нека основна подела јесте на домаћинства и привреду. Након неког времена, може се упоредити потрошња са паметног водомера и профил који одговара том типу потрошача, и створити се основа за другачије тарифирање уколико потрошач значајно одступа од свог профила потрошње. Са једне стране описана ситуација може да се објасни тако да постоји неки непријављени потрошач или је потребно да се спроведе пренамена профила, нпр. из домаћинства у привреду. Поред тога, поменути обрасци у потрошњи потрошача, могу се искористити да се на нивоу потрошача изврши провера да ли постоје нека додатна цурења, па се потрошња показује као увећана. То може да буде веома корисно тамо где постоје већи број потрошача који користи воду са једног заједничког прикључка, као што је нпр. у зградама.

Подаци и навике о потрошњи стварају основу да се предвиђа тражња за водом, па самим тим постоји могућност да се оптимизује процес рада пумпи чиме би се смањили трошкови од захвата воде, до њене дистрибуције. ИТ технологије које омогућавају ово решење су: интернет ствари, рачунарство у облацима, Edge рачунарство, Rest Web сервиси и тд.

5.3.2.5.4. Могућност примене ИТ на проблему немогућности провере испуњења обавеза ЈКП БВК

Потрошачи желе да имају бољи увид у своју потрошњу и они могу да одлуче да уграде решења који нуде разни произвођачи опреме и софтвера, а која може да мери потрошњу воде унутар њихових објеката, станова и кућа. Таква решења имају за циљ да потрошачи имају увид у своју потрошњу као и да брзо спроведу интервенције уколико неке од њихових унутрашњих инсталација се покваре.

Нека од поменутих решења су дата у (Di Mauro et al., 2019), где постоји неколико архитектуралних елемената, то су: паметни мерач, локални gateway и Cloud платформа која са својим елементима има могућност да прихвата податке од локалног gateway-а. Уобичајено је да се потрошачима омогући путем мобилне апликације да надзиру потрошњу на свим местима у која су уградили паметне мераче. На описани начин могуће је аутоматски прикупљати податке о читавањима. Концептуална архитектура једног од решења је приказана на слици. Са оваквим системом, ствара се могућност да се подаци о потрошњи унутар њиховог стана или куће, упореде са подацима о потрошњи воде која је испоручена потрошачу кроз паметни водомер и на тај начин се има увид да ли је ЈКП БВК испунио своју обавезу.



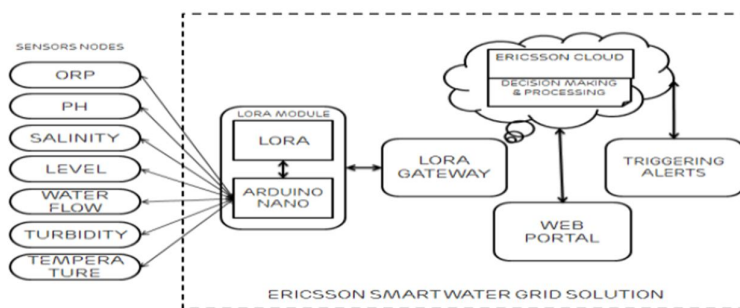
Слика 40. концептуална архитектура решења, преузето (Di Mauro et al., 2019)

У случају већих одступања разлог се може тражити у инсталацијама од водомера до потрошача или евентуалном отказу паметног мерача на страни водовода, описаног у претходном сценарију. Дакле, овакав сценарио захтева одређени вид интеграције потенцијалног решења ЈКП БВК и решења произвођача опреме и софтвера који нуди

услугу крајњим корисницима. ИТ технологије које омогућавају поменута решења су: паметни мерачи, рачунарство у облацима и тд.

5.3.2.5.5. Могућност примене ИТ на проблему контаминације пијеће воде

Један од важних параметара која вода која се испоручи потрошачу треба да испуни јесте да буде одговарајућег квалитета. То значи да је потребно да параметри који се надзиру имају одговарајуће вредности. Нарушавање вредности тих параметра може настати просласком кроз већи број цеви, од којих потенцијално нека од њих може да буде контаминирана. Уобичајено је да се мере pH, conductivity, turbidity, temperature и тд. Архитектура решења које надзире квалитет воде, може да буде веома слична архитектури описаној у надзирању изворишта. Поред тога, сличан опис је дат и у (Saravanan et al., 2017), који се састоји од одоварајућих сензора, затим LoRA модула који користи Arduino и очитане параметре пролеђује LoRA gateway-у, који затим прослеђује Ericsson Cloud-у, који податке процесира, примењујући одређене алгоритме машинског учења и визуелизује их на Web интерфејсу. Архитектура решња је приказана на слици 41.



Слика 41. архитектура решења за детекцију контаминације, преузето (Saravanan et al., 2017)

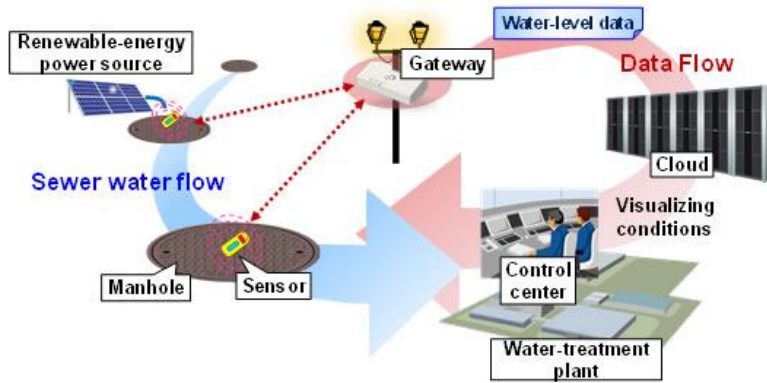
Ово решење може да буде веома значајно у циљу спречавања контаминације веће дела популације. Сличан сценарио, уз одговарајуће измене може се применити на детекцији одређене контаминације у цевоводима сирове воде.

ИТ технологије које омогућавају поменута решења су: интернет ствари, рачунарство у облацима, вештачка интелигенција, машинско учење и тд.

5.3.2.5.6. Могућност примене ИТ на проблем немогућност утврђивања приоритета одржавања сливника на значајну промену атмосферских прилика

Да би се спречило изливање атмосферске воде на улицу из канализационих сливника могуће је уградити сензор на одређеној висини сливника, чиме би се пратило колики је ниво воде унутар сливника и када пређе одређену границу, затим да се тај

тогађај проследи локалном gateway-у, који би даље послао информацију на облак. Једно такво решење, нуди Fujitsu (Fujitsu, n.d.). Архитектура решња је приказана на слици 42. На тај начин могуће је да се изврши аутоматско креирање радног налога за тим људи који је потребно да правовремено отклони проблем у сливнику.



Слика 42 решење за праћење количине воде доспеле у атмосферску канализацију, преузето (Fujitsu, 2015)

ИТ технологије које омогућавају поменута решења су: интернет ствари, рачунарство у облацима, Rest Web сервиси и тд.

5.3.2.5. Предлог примене технолошких решења на основу анализе ланца вредности на наслеђеном информационом систему ЈКП БВК - Закључак анализе ланца вредности

На основу анализе могућности примене ИТ технологија које делују на пословне активности ланца вредности, може се рећи да је потребно обезбедити следеће технологије:

- интернет ствари и паметни мерачи
- технологије облака, Fog-а и Edge-а
- технологије великих података
- вештачка интелигенција, машинско учење и детекција аномалија

Свако од ових решења захтева, одређене податке које се налазе инутар наслеђеног ИС, а то ће се илустровати на неколико примера.

На примеру детекција квара захтева да се на основу података из система за детекцију квара на дистрибутивној мрежи ЈКП БВК интегрише са делом наслеђеног ИС који обрађује кварове, апликацијом за пријаву кварова и командно контролним центром.

Софтвер за читавање вредности потрошње са паметних мерача, захтева упису у базу дела система фактурисања наслеђеног информационих система.

Поред тога, подаци о прочитаним стањима, уколико је потребно да се упореде са постављеним профилима потрошача, је потребно прво да се упишу у базу података наслеђеног информационог система, а то су програмски модули за фактурисање и за водомере.

Имајући у виду, да ЈКП БВК има наслеђен систем написан на Progress 4GL програмском језику, а за побољшавање примарних активности у ланцу вредности је потребно омогућити рад са технологијама: интернет ствари, паметних мерача, технологијама облака, Fog-а и Edge-а, технологијама великих података и технологијама које омогућавају рад са вештачком интелигенцијом, машинским учењем и детекцијом аномалија, закључује се да наслеђен систем не може да се интегрише са поменутим технологијама.

У верзији 10.2В Progress OpenEdge платформе коју користи ЈКП БВК није могућа употреба REST Web сервиса из окружења. Технолошко решење које би могло да обухвати употребу сервиса из технологије облака и описаних решења, који би се интегрисали са наслеђеним информационим системом подразумевало би употребу хибридног облака, које комбинује on-premise решење (наслеђен систем) и технологије које се користе на облаку. Једна од технологија који би се користила подразумевала би употребу REST API-а, а коју Progress OpenEdge платформа у верзији 10.2В не може да подржи.

6. Истраживање

6.1. Опис начина истраживања

Након дефинисања теоријског оквира и спроведене студије случаја на моделу ЈКП „Београдски водовод и канализације“, а након тога и анализе ланца вредности ЈКП „Београдски водовод и канализација“, на реду је спровести истраживање које има за циљ да евалуира предложени теоријски оквир у пракси. Циљ је да се обезбеди спољна валидност студије случаја, као основа за могућности уопштавања резултата студије случаја.

Та фаза истраживања ће се спровести путем упитника, у којој ИТ менаџери, људи који раде на развоју система (програмери, пројектанти, архитекте, администратори базе података и тд.), као и менаџери средњег нивоа, исказују своје ставове о наслеђеним информационим системима.

6.2. Прикупљање и обрада података

6.2.1. Опис начина прикупљања података

6.2.1.1. Први круг истраживања

У првом кругу истраживања спроведена је анкета (упитник) од 18 питања, која су приказана у наставку:

- Назив државе у којој се налази Ваше предузеће
- Назив града у којем се налази Ваше предузеће
- Ваше предузеће има стари (застарели) информациони систем (ИС) или софтвер
- Стари (застарели) ИС или софтвер је стар
- Програмски језик (или језици) у којима је кодиран стари ИС (или софтвер/и) [можете написати и окружење или језик базе података]
- У старом ИС (или софтверу/има), нарушен је иницијално постављен дизајн (пројектна замисао) услед великог броја измена и дугорочног одржавања
- Стари ИС је користан, тиме што и даље доприноси пословању (и даље се извршавају одређене пословне процедуре и процеси)

- У старом ИС релативно једнаставно се врше измене, у погледу измене пословних процедура или пословних процеса, а у складу са захтевима
- Ако се изврши измена у једном делу старог ИС, та промена се пропагира ка другим деловима система
- Стари ИС има апликације (развијене за неку организациону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама
- Заменом старе апликације или дела старог ИС потенцијално новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим
- Интеграција старог ИС, са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS, и тд.) је једноставна.
- Постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин
- Стари ИС се извршава на застарелом хардверу
- Програмери (пројектанти, software developer-и, software engineer-и) могу да примене нановија знања методологије, концепте и алате из области софтверског инжењерства, без проблема у старом ИС
- Стари ИС је развијен од стране кадрова и средстава самог предузећа (in-house)
- Стари ИС је потребно модернизовати
- Стари ИС, такав какав је (без измена) може се лако интегрисати са елементима Индустије 4.0 (Cloud Computing, Internet of Things, Big Data и тд.)

6.2.1.2. Други круг истраживања

У другом кругу истраживања спроведена је допуна претходне анкете (упитника) са додатних 7 питања, над истим испитаницима. То је урађено зато што је аутор сматрао да ће се тиме добити потпунији и јаснији резултати истраживања, који нису могли да буду довољно истражени са претходно конципираном анкетом од 18 питања. Постојала је потреба да се додатно испитају још неки елементи чиниоца предмета истраживања, као и да се направи директнија веза између перспектива наслеђених система и проблема наслеђених система, а са циљем да истраживање буде обухватније него што би било у случају да је остала анкета од 18 питања.

Аутор је одлучио да допуни анкету (упитник) да би се омогућило боље разумевање елемената предмета истраживања, односа између њих као и закључака који ће бити изведени, а поред тога и потпуније одређивање будуће стратегије модернизације наслеђеног информационог система.

Додатних 7 питања је приказано у наставку:

- Документација старог ИС постоји и ажурна је
- Велики је одлазак запослених који у радили на старом ИС
- Редунданса у систему је изражена
- Новозапослени имају знање и вештине да у старту раде са старим ИС
- Постоји подршка приликом одржавања хардвера и софтвера од стране произвођача
- Модернизациом старог ИС могу створити бољи услови за подршку одлучивању него супротно.
- Налазим да овакав упитник може боље усмерити модернизацију система, уместо посматрања само из једног угла (нпр. технолошког)

На основу првог и другог круга истраживања дошло се до прикупљања података, као да се ради у једној интегралној анкети од 25 питања. Број питања је 25, од тога број отворених питања је 3, а број затворених питања са могућношћу избора једне опције 22. Затворена питања су уређена према Ликертовој скали од 5 одговора.

Од 22 питања, на 21 питање (на питање број 3 и на питања од броја 6 до броја 25) је могуће дати одговор: у потпуности се слажем, делимично се слажем, подељено, делимично се не слажем и у потпуности се не слажем. На питање број 4 везано за старост система могуће је дати одговор: мање од 10 година, између 10 и 15 година, између 15 и 20 година, између 20 и 25 година и преко 25 година.

Обједињена (интегрална) анекта је приказана са питањима, њиховим редном бројевима и могућим одговорима у табели 9.

Табела 9. обједињени упитник са питањима (25 питања)

| Редни број питања | Питање | Тип питања | Могући одговори |
|-------------------|---|------------|-----------------|
| 1. | Назив државе у којој се налази Ваше предузеће | отворено | |
| 2. | Назив града у којем се налази Ваше | отворено | |

| | предузеће | | |
|-----|---|-----------|--|
| 3. | Ваше предузеће има стари (застарели) информациони систем (ИС) или софтвер | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 4. | Стари (застарели) ИС или софтвер је стар | затворено | <ul style="list-style-type: none"> • мање од 10 година • између 10 и 15 година • између 15 и 20 година • између 20 и 25 година • преко 25 година |
| 5. | Програмски језик (или језици) у којима је кодиран стари ИС (или софтвер/и) [можете написати и окружење или језик базе података] | отворено | |
| 6. | У старом ИС (или софтверу/има), нарушен је иницијално постављен дизајн (пројектна замисао) услед великог броја измена и дугорочног одржавања | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 7. | Стари ИС је користан, тиме што и даље доприноси пословању (и даље се извршавају одређене пословне процедуре и процеси) | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 8. | У старом ИС релативно једставно се врше измене, у погледу измене пословних процедура или пословних процеса, а у складу са захтевима | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 9. | Ако се изврши измена у једном делу старог ИС, та промена се пропагира ка другим деловима система | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 10. | Стари ИС има апликације (развијене за неку организациону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 11. | Заменом старе апликације или дела старог | затворено | <ul style="list-style-type: none"> • у потпуности се слажем |

| | | | |
|-----|--|-----------|--|
| | ИС потенцијално новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим | | <ul style="list-style-type: none"> • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 12. | Интеграција старог ИС, са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS, и тд.) је једноставна. | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 13. | Постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 14. | Стари ИС се извршава на застарелом хардверу | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 15. | Програмери (пројектанти, software developer-и, software enginner-и) могу да примене нановија знања методологије, концепте и алате из области софтверског инжењерства, без проблема у старом ИС | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 16. | Стари ИС је развијен од стране кадрова и средстава самог предузећа (in-house) | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 17. | Стари ИС је потребно модернизовати | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 18. | Документација старог ИС постоји и ажурна је | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 19. | Велики је одлазак запослених који у радили на старом ИС | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем |

| | | | |
|-----|--|-----------|--|
| | | | <ul style="list-style-type: none"> • подељено • делимично се не слажем • у потпуности се не слажем |
| 20. | Редуданса у систему је изражена | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 21. | Новозапослени имају знање и вештине да у старту раде са старим ИС | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 22. | Постоји подршка приликом одржавања хардвера и софтвера од стране произвођача | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 23. | Модернизациом старог ИС могу створити бољи услови за подршку одлучивању него супротно. | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 24. | Стари ИС, такав какав је (без измена) може се лако интегрисати са елементима Индустије 4.0 (Cloud Computing, Internet of Things, Big Data и тд.) | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |
| 25. | Налазим да овакав упитник може боље усмерити модернизацију система, уместо посматрања само из једног угла (нпр. технолошког) | затворено | <ul style="list-style-type: none"> • у потпуности се слажем • делимично се слажем • подељено • делимично се не слажем • у потпуности се не слажем |

6.2.2. Узорак истраживања

Одговори на питања су прикупљени од испитаника који раде у водоводним предузећима у Србији и региону и то је учињено путем online упитника. Са испитаницима се комуницирало: телефонским путем, употребом e-mail-а и коришћењем социјалне мреже LinkedIn. Називи предузећа нису захтевани у упитнику, обзиром да су градски водоводи једина предузећа тог типа у граду, а испитаници раде у њима.

Упитник је за испитанике у Мађарској, Бугарској и Румунији креиран на енглеском језику. Одговори на упитник су након тога обједињени.

Истраживање је спроведено на узорку од 63 испитаника који раде у водоводним предузећима у Србији и региону. У табели 10. је приказан број испитаника.

Табела 10. број испитаници по земљама

| Земља | Број испитаника |
|--------|-----------------|
| Србија | 34 |
| Регион | 29 |

У табели 11. је приказан број испитаника у земљама у региону који раде у водоводним предузећима.

Табела 11. број испитаника из земаља из региона

| Земље из региона | Број испитаника |
|---------------------|-----------------|
| Босна и Херцеговина | 7 |
| Бугарска | 2 |
| Црна Гора | 3 |
| Хрватска | 6 |
| Мађарска | 3 |
| Румунија | 2 |
| Северна Македонија | 3 |
| Словенија | 3 |

У табели 12. је приказан број испитаника из Србије по градовима у којима се налазе водоводна предузећа у којима су они запослени.

Табела 12. број испитаника из Србије по градовима

| Земља | Град | Број испитаника |
|--------|-----------------|-----------------|
| Србија | Крагујевац | 1 |
| | Краљево | 2 |
| | Лесковац | 2 |
| | Мионица | 2 |
| | Горњи Милановац | 1 |
| | Лазаревац | 1 |
| | Инђија | 1 |
| | Стара Пазова | 1 |
| | Ваљево | 1 |
| | Нови Сад | 3 |
| | Ниш | 3 |
| | Суботица | 2 |
| | Зрењанин | 2 |

| | | |
|--|-------------------|---|
| | Смедерево | 2 |
| | Шабац | 1 |
| | Чачак | 1 |
| | Крушевац | 1 |
| | Сремска Митровица | 2 |
| | Вршац | 2 |
| | Врање | 2 |
| | Пожаревац | 1 |

У табели 13. је приказан број испитаника из Босне и Херцеговине по градовима у којима се налазе водоводна предузећа у којима су они запослени.

Табела 13. испитаници из Босне и Херцеговине по градовима

| Земља | Град | Број испитаника |
|---------------------|-----------------|-----------------|
| Босна и Херцеговина | Козарска Дубица | 1 |
| | Сарајево | 3 |
| | Добој | 1 |
| | Бијељина | 2 |

У табели 14. је приказан број испитаника из Бугарске по градовима у којима се налазе водоводна предузећа у којима су они запослени.

Табела 14. испитаници из Бугарске по градовима

| Земља | Град | Број испитаника |
|----------|---------|-----------------|
| Бугарска | Софија | 1 |
| | Пловдив | 1 |

У табели 15. је приказан број испитаника из Црне Горе по градовима у којима се налазе водоводна предузећа у којима су они запослени.

Табела 15. испитаници из Црне Горе по градовима

| Земља | Град | Број испитаника |
|-----------|-----------|-----------------|
| Црна Гора | Никшић | 1 |
| | Бар | 1 |
| | Подгорица | 1 |

У табели 16. је приказан број испитаника из Хрватске по градовима у којима се налазе водоводна предузећа у којима су они запослени.

Табела 16. испитаници из Хрватске по градовима

| Земља | Град | Број испитаника |
|----------|-----------|-----------------|
| Хрватска | Загреб | 1 |
| | Дубровник | 1 |
| | Осијек | 1 |
| | Огулин | 1 |

| | | |
|--|-------|---|
| | Бузет | 1 |
| | Јелса | 1 |

У табели 17. је приказан број испитаника из Мађарске по градовима у којима се налазе водоводна предузећа у којима су они запослени.

Табела 17. испитаници из Мађарске по градовима

| Земља | Град | Број испитаника |
|----------|------------|-----------------|
| Мађарска | Будимпешта | 1 |
| | Мишколц | 1 |
| | Дебрецин | 1 |

У табели 18. је приказан број испитаника из Румуније по градовима у којима се налазе водоводна предузећа у којима су они запослени.

Табела 18. испитаници из Румуније по градовима

| Земља | Град | Број испитаника |
|----------|----------|-----------------|
| Румунија | Букурешт | 1 |
| | Темишвар | 1 |

У табели 19. је приказан број испитаника из Северне Македоније по градовима у којима се налазе водоводна предузећа у којима су они запослени.

Табела 19. испитаници из Северне Македоније по градовима

| Земља | Град | Број испитаника |
|--------------------|----------|-----------------|
| Северна Македонија | Скопље | 1 |
| | Куманово | 1 |
| | Кочани | 1 |

У табели 20. је приказан број испитаника из Словеније по градовима у којима се налазе водоводна предузећа у којима су они запослени.

Табела 20. испитаници из Словеније по градовима

| Земља | Град | Број испитаника |
|-----------|---------|-----------------|
| Словенија | Пиран | 1 |
| | Љубљана | 1 |
| | Марибор | 1 |

6.2.3. Ставови испитаника у вези са наслеђеним информационим системима

На питање под редним бројем 4 које је везано за старост ИС одговори могу бити да је наслеђени ИС стар

- до 10 година

- између 10 и 15 година
- између 15 и 20 година
- између 20 и 25 година
- преко 25 година

Вредностима поменутих обележја су у SPSS алату придружене вредности, тако да до 10 година узима вредност 1, да између 10 и 15 година узима вредност 2, да између 15 и 20 година узима вредност 3, да између 20 и 25 година узима вредност 4 и да преко 25 година узима вредност 5. Табела фреквенција је приказана испод, према којој се види учешће одређене групе у узорку.

Табела 21. табела фреквенција

| Starost_softvera | | | | | |
|------------------|-------|-----------|---------|---------------|--------------------|
| | | Frequency | Percent | Valid Percent | Cumulative Percent |
| Valid | 1 | 30 | 47,6 | 47,6 | 47,6 |
| | 2 | 17 | 27,0 | 27,0 | 74,6 |
| | 3 | 9 | 14,3 | 14,3 | 88,9 |
| | 4 | 3 | 4,8 | 4,8 | 93,7 |
| | 5 | 4 | 6,3 | 6,3 | 100,0 |
| | Total | 63 | 100,0 | 100,0 | |

На питање под редним бројем 5, корисници су одговорали тако што су уписивали следеће програмске језик: SQL, Oracle, Dynamics NAV, MS Access, Visual FoxPro, Cobol, Java, VB (Visual Basic), C#, MS SQL, Progress, SQL Server, PL2, .NET, Delphi, Harbour, MySQL, C++, IAF, ASP, PHP, ABAP (SAP) и C. Уочава се да поред програмских језика су уносили и називе платформи, језика базе података и програмска окржења.

За потребе статистичке анализе коришћен је статистички пакет SPSS. Спроведене су следеће анализе:

- дескриптивна статистика
- т-тест
- корелација

6.2.3.1. Дескриптивна статистика

Аритметичка средина (просек) је најчешће коришћена мера централне тенденције. У случају негруписаних података, аритметичка средина се добија дељењем збира свих вредности са бројем вредности у тој серији података (Man, 2009). Аритметичка средина

израчуната за податке узорка се обележава са \bar{x} . Број вредности узорка се обележава са n . Варијабла се обележава са X , а збир свих вредности варијабле X са $\sum x$.

$$\text{Аритметичка средина узорка: } \bar{X} = \frac{\sum x}{n}$$

Стандардна девијација је најчешће коришћена мера дисперзије. Вредност стандардне девијације показује колико су близу вредности серије података груписане око аритметичке средине (Man, 2009). Стандардна девијација се добија узимањем позитивног квадратног корена варијансе. Варијанса узорка се означава са s^2 , а стандардна девијација узорка са s .

$$\text{Варијанса узорка: } s^2 = \frac{\sum (x - \bar{x})^2}{n}$$

Величина $x - \bar{x}$ у наведеној формули се назива девијација (одступање) вредности x од аритметичке средине.

Коефицијент варијације, је однос између стандардне девијације и аритметичке средине (Man, 2009). Најчешће се изражава у процентима, док у софтверским статистичким пакетима, то не мора да буде случај.

$$CV = \frac{s}{\bar{x}}$$

Ставовима испитаника, на питање број 3 и на питања од броја 6 до броја 25 придружене су вредности статистичког обележја на следећи начин:

Табела 22. придружене вредности статистичког обележја у програмском пакету SPSS

| | |
|---------------------------|---|
| У потпуности се не слажем | 1 |
| Делимично се не слажем | 2 |
| Подељено | 3 |
| Делимично се слажем | 4 |
| У потпуности се слажем | 5 |

Вредност дескриптивних статистика је приказана у следећој табели, која је приказује излазне резултате из SPSS-а:

Табела 23. вредност дескриптивних статистика

Descriptive Statistics

| | N | Mean | Std. Deviation | Koeficijent varijacije |
|----------|----|------|----------------|------------------------|
| Pitanje3 | 63 | 3,16 | 1,273 | 0,40 |
| Pitanje6 | 63 | 3,52 | 1,189 | 0,34 |

| | | | | |
|--------------------|----|------|-------|------|
| Pitanje7 | 63 | 3,95 | 1,156 | 0,29 |
| Pitanje8 | 63 | 2,59 | 1,227 | 0,47 |
| Pitanje9 | 63 | 3,65 | 1,138 | 0,31 |
| Pitanje10 | 63 | 3,24 | 1,316 | 0,41 |
| Pitanje11 | 63 | 3,87 | 1,143 | 0,30 |
| Pitanje12 | 63 | 2,24 | 1,160 | 0,52 |
| Pitanje13 | 63 | 3,52 | 1,148 | 0,33 |
| Pitanje14 | 63 | 2,87 | 1,431 | 0,50 |
| Pitanje15 | 63 | 2,76 | 1,292 | 0,47 |
| Pitanje16 | 63 | 2,40 | 1,397 | 0,58 |
| Pitanje17 | 63 | 4,08 | 1,168 | 0,29 |
| Pitanje18 | 63 | 2,56 | 1,074 | 0,42 |
| Pitanje19 | 63 | 3,89 | 0,764 | 0,20 |
| Pitanje20 | 63 | 3,03 | 1,107 | 0,37 |
| Pitanje21 | 63 | 2,67 | 1,244 | 0,47 |
| Pitanje22 | 63 | 3,10 | 0,995 | 0,32 |
| Pitanje23 | 63 | 4,25 | 0,803 | 0,19 |
| Pitanje24 | 63 | 1,95 | 1,069 | 0,55 |
| Pitanje25 | 63 | 4,21 | 0,626 | 0,15 |
| Valid N (listwise) | 63 | | | |

6.2.3.2. T-тест

За потребе истраживања спроведено је неколико Т-тестова, према процедури Independent-Samples Т-тест. Овај тест примењује се за тестирање хипотезе о једнакости аритметичких средина посматраног обележја за два скупа када имамо независне узорке (Soldić-Aleksić, 2011).

Нулта хипотеза т-теста је: $H_0: \mu_1 = \mu_2$, тестира се против алтернативне хипотезе: $H_1: \mu_1 \neq \mu_2$. Т статистика се рачуна као количник разлике две узорачке средине посматраног обележја и стандардне грешке разлике између средина:

$$t = \frac{\bar{x}_1 - \bar{x}_2}{s\bar{x}_1 - s\bar{x}_2}$$

Овај тест има два случаја у пракси. Један случај је када су варијансе у скуповима једнаке, а други случај је када варијансе у скуповима нису једнаке. За проверу те једнакости, тј. хомогеност варијансе у IBM SPSS користи се Левенеов тест (Pallant, 2017).

Код Левенеовог теста гледамо вредност р, тј. вредност Sig. (значајност, тј. вероватноћа) коју поредимо са нивоом значајности од 5%. Када је та вредност већа од нивоа значајности, тј. од 0,05 закључујемо да су варијансе једнаке, а у супротном да нису

једнаке. Од тог резултата зависи који резултат р вредности (Sig 2-tailed) читамо за закључак Independent-Samples T-тест-а.

6.2.3.2.1. Тестирање ставова испитаника у Србији и земљама региона

У SPSS алату креирају се 2 групе:

- група 1 – којој одговара Србија (вредност 1,00)
- група 2 – којој одговарају земље у региону (вредност 2,00)

T-тест нам служи да упоредимо да ли се ставови по одређеним питањима у вези наслеђених система у просеку разликују између испитаника који раде у водоводним предузећима у Србији и земљама у региону.

У SPSS алату прво се добијају дескриптивне статистике за обе групе, табела 24., на основу којих може да се наслути какви ће бити каснији резултати теста.

Табела 24. дескриптивне статистике за групе испитаника из Србије и из земаља региона

Group Statistics

| | Drzava1 | N | Mean | Std. Deviation | Std. Error Mean |
|-----------|---------|----|------|----------------|-----------------|
| Pitanje3 | 1,00 | 34 | 3,56 | 1,236 | ,212 |
| | 2,00 | 29 | 2,69 | 1,168 | ,217 |
| Pitanje6 | 1,00 | 34 | 3,71 | 1,219 | ,209 |
| | 2,00 | 29 | 3,31 | 1,137 | ,211 |
| Pitanje7 | 1,00 | 34 | 3,85 | 1,077 | ,185 |
| | 2,00 | 29 | 4,07 | 1,252 | ,232 |
| Pitanje8 | 1,00 | 34 | 2,24 | 1,103 | ,189 |
| | 2,00 | 29 | 3,00 | 1,254 | ,233 |
| Pitanje9 | 1,00 | 34 | 3,74 | 1,163 | ,199 |
| | 2,00 | 29 | 3,55 | 1,121 | ,208 |
| Pitanje10 | 1,00 | 34 | 3,59 | 1,234 | ,212 |
| | 2,00 | 29 | 2,83 | 1,311 | ,243 |
| Pitanje11 | 1,00 | 34 | 4,24 | ,923 | ,158 |
| | 2,00 | 29 | 3,45 | 1,242 | ,231 |
| Pitanje12 | 1,00 | 34 | 2,06 | 1,205 | ,207 |
| | 2,00 | 29 | 2,45 | 1,088 | ,202 |
| Pitanje13 | 1,00 | 34 | 3,68 | 1,121 | ,192 |
| | 2,00 | 29 | 3,34 | 1,173 | ,218 |
| Pitanje14 | 1,00 | 34 | 3,06 | 1,324 | ,227 |
| | 2,00 | 29 | 2,66 | 1,542 | ,286 |
| Pitanje15 | 1,00 | 34 | 2,47 | 1,285 | ,220 |
| | 2,00 | 29 | 3,10 | 1,235 | ,229 |

| | | | | | |
|-----------|------|----|------|-------|------|
| Pitanje16 | 1,00 | 34 | 2,47 | 1,331 | ,228 |
| | 2,00 | 29 | 2,31 | 1,491 | ,277 |
| Pitanje17 | 1,00 | 34 | 4,26 | ,994 | ,171 |
| | 2,00 | 29 | 3,86 | 1,329 | ,247 |
| Pitanje18 | 1,00 | 34 | 2,56 | 1,078 | ,185 |
| | 2,00 | 29 | 2,55 | 1,088 | ,202 |
| Pitanje19 | 1,00 | 34 | 4,12 | ,769 | ,132 |
| | 2,00 | 29 | 3,62 | ,677 | ,126 |
| Pitanje20 | 1,00 | 34 | 3,15 | 1,184 | ,203 |
| | 2,00 | 29 | 2,90 | 1,012 | ,188 |
| Pitanje21 | 1,00 | 34 | 2,29 | 1,268 | ,217 |
| | 2,00 | 29 | 3,10 | 1,081 | ,201 |
| Pitanje22 | 1,00 | 34 | 3,03 | 1,029 | ,177 |
| | 2,00 | 29 | 3,17 | ,966 | ,179 |
| Pitanje23 | 1,00 | 34 | 4,21 | ,729 | ,125 |
| | 2,00 | 29 | 4,31 | ,891 | ,165 |
| Pitanje24 | 1,00 | 34 | 1,76 | 1,075 | ,184 |
| | 2,00 | 29 | 2,17 | 1,037 | ,193 |
| Pitanje25 | 1,00 | 34 | 4,21 | ,687 | ,118 |
| | 2,00 | 29 | 4,21 | ,559 | ,104 |

Након тога се спроводи Independent-Samples T-тест, где је резултат приказан у табели 25.

Табела 25. резултат T-теста којим се тестирају ставови испитаника из Србије и из земаља региона

| Independent Samples Test | | | | | | | | | | |
|--------------------------|-----------------------------|---|------|------------------------------|--------|-----------------|-----------------|------------------|---|-------|
| | | Levene's Test for Equality of Variances | | t-test for Equality of Means | | | | | | |
| | | F | Sig. | t | df | Sig. (2-tailed) | Mean Difference | Std. Error Diff. | 95% Confidence Interval of the Difference | |
| | | | | | | | | | Lower | Upper |
| Pitanje 3 | Equal variances assumed | ,004 | ,950 | 2,853 | 61 | ,006 | ,869 | ,305 | ,260 | 1,478 |
| | Equal variances not assumed | | | 2,866 | 60,326 | ,006 | ,869 | ,303 | ,263 | 1,476 |
| Pitanje 6 | Equal variances assumed | ,806 | ,373 | 1,324 | 61 | ,191 | ,396 | ,299 | -,202 | ,993 |

| | | | | | | | | | | |
|------------|-----------------------------|-------|------|--------|--------|------|-------|------|--------|-------|
| | Equal variances not assumed | | | 1,331 | 60,487 | ,188 | ,396 | ,297 | -,199 | ,990 |
| Pitanje 7 | Equal variances assumed | ,153 | ,697 | -,737 | 61 | ,464 | -,216 | ,293 | -,803 | ,370 |
| | Equal variances not assumed | | | -,728 | 55,684 | ,470 | -,216 | ,297 | -,811 | ,379 |
| Pitanje 8 | Equal variances assumed | ,232 | ,631 | -2,576 | 61 | ,012 | -,765 | ,297 | -1,358 | -,171 |
| | Equal variances not assumed | | | -2,550 | 56,332 | ,014 | -,765 | ,300 | -1,365 | -,164 |
| Pitanje 9 | Equal variances assumed | ,052 | ,820 | ,635 | 61 | ,528 | ,184 | ,289 | -,395 | ,762 |
| | Equal variances not assumed | | | ,637 | 60,060 | ,527 | ,184 | ,288 | -,393 | ,760 |
| Pitanje 10 | Equal variances assumed | ,093 | ,762 | 2,370 | 61 | ,021 | ,761 | ,321 | ,119 | 1,403 |
| | Equal variances not assumed | | | 2,358 | 58,131 | ,022 | ,761 | ,323 | ,115 | 1,406 |
| Pitanje 11 | Equal variances assumed | 3,012 | ,088 | 2,880 | 61 | ,005 | ,787 | ,273 | ,241 | 1,333 |
| | Equal variances not assumed | | | 2,814 | 51,002 | ,007 | ,787 | ,280 | ,226 | 1,349 |
| Pitanje 12 | Equal variances assumed | ,020 | ,887 | -1,337 | 61 | ,186 | -,389 | ,291 | -,972 | ,193 |
| | Equal variances not assumed | | | -1,348 | 60,779 | ,183 | -,389 | ,289 | -,967 | ,189 |
| Pitanje 13 | Equal variances assumed | ,220 | ,640 | 1,146 | 61 | ,256 | ,332 | ,289 | -,247 | ,910 |
| | Equal variances not assumed | | | 1,141 | 58,481 | ,258 | ,332 | ,291 | -,250 | ,913 |
| Pitanje 14 | Equal variances assumed | 2,709 | ,105 | 1,118 | 61 | ,268 | ,404 | ,361 | -,318 | 1,126 |
| | Equal variances not assumed | | | 1,105 | 55,638 | ,274 | ,404 | ,365 | -,329 | 1,136 |
| Pitanje 15 | Equal variances assumed | ,533 | ,468 | -1,984 | 61 | ,052 | -,633 | ,319 | -1,271 | ,005 |
| | Equal variances not assumed | | | -1,990 | 60,103 | ,051 | -,633 | ,318 | -1,269 | ,003 |
| Pitanje 16 | Equal variances assumed | ,626 | ,432 | ,451 | 61 | ,654 | ,160 | ,356 | -,551 | ,871 |

| | | | | | | | | | | |
|------------|-----------------------------|-------|------|--------|--------|------|-------|------|--------|-------|
| | Equal variances not assumed | | | ,447 | 56,763 | ,657 | ,160 | ,359 | -,558 | ,879 |
| Pitanje 17 | Equal variances assumed | 2,462 | ,122 | 1,373 | 61 | ,175 | ,403 | ,293 | -,184 | ,989 |
| | Equal variances not assumed | | | 1,342 | 51,212 | ,185 | ,403 | ,300 | -,199 | 1,005 |
| Pitanje 18 | Equal variances assumed | ,099 | ,754 | ,026 | 61 | ,979 | ,007 | ,274 | -,540 | ,555 |
| | Equal variances not assumed | | | ,026 | 59,266 | ,979 | ,007 | ,274 | -,541 | ,555 |
| Pitanje 19 | Equal variances assumed | ,312 | ,578 | 2,699 | 61 | ,009 | ,497 | ,184 | ,129 | ,865 |
| | Equal variances not assumed | | | 2,727 | 60,931 | ,008 | ,497 | ,182 | ,133 | ,861 |
| Pitanje 20 | Equal variances assumed | ,566 | ,455 | ,894 | 61 | ,375 | ,251 | ,280 | -,310 | ,811 |
| | Equal variances not assumed | | | ,905 | 60,999 | ,369 | ,251 | ,277 | -,303 | ,804 |
| Pitanje 21 | Equal variances assumed | 2,020 | ,160 | -2,701 | 61 | ,009 | -,809 | ,300 | -1,409 | -,210 |
| | Equal variances not assumed | | | -2,735 | 61,000 | ,008 | -,809 | ,296 | -1,401 | -,218 |
| Pitanje 22 | Equal variances assumed | ,101 | ,752 | -,565 | 61 | ,574 | -,143 | ,253 | -,649 | ,363 |
| | Equal variances not assumed | | | -,568 | 60,415 | ,572 | -,143 | ,252 | -,646 | ,360 |
| Pitanje 23 | Equal variances assumed | ,228 | ,635 | -,512 | 61 | ,611 | -,104 | ,204 | -,513 | ,304 |
| | Equal variances not assumed | | | -,504 | 54,165 | ,616 | -,104 | ,207 | -,520 | ,311 |
| Pitanje 24 | Equal variances assumed | ,016 | ,899 | -1,525 | 61 | ,132 | -,408 | ,267 | -,942 | ,127 |
| | Equal variances not assumed | | | -1,529 | 60,038 | ,131 | -,408 | ,267 | -,941 | ,126 |
| Pitanje 25 | Equal variances assumed | ,567 | ,454 | -,006 | 61 | ,995 | -,001 | ,160 | -,320 | ,318 |
| | Equal variances not assumed | | | -,006 | 60,887 | ,995 | -,001 | ,157 | -,315 | ,313 |

6.2.3.2.2. Тестирање ставова испитаника у односу на старост наслеђеног система

У SPSS алату креирају се 2 групе:

- група 1 – којој одговара да старост застарелог ИС или софтвера је до 10 година (вредност 1,00)
- група 2 – којој одговара да старост застарелог ИС или софтвера је преко 10 година (вредност 2,00)

T-тест нам служи да упоредимо да ли се ставови по одређеним питањима у вези наслеђених система у просеку разликују између испитаника који раде у водоводним предузећима где је ИС стар до 10 година у односу на ставове испитаника који раде у водоводним предузећима где је ИС стар преко 10 година.

У SPSS алату прво се добијају дескриптивне статистике за обе групе.

Табела 26. дескриптивне статистике за групе испитаника код којих је наслеђен систем предузећа у којем раде стар до 10 година и испитаника код којих је наслеђен систем предузећа старији од 10 година

Group Statistics

| | Starost_softvera1 | N | Mean | Std. Deviation | Std. Error Mean |
|-----------|-------------------|----|------|----------------|-----------------|
| Pitanje3 | 1,00 | 30 | 2,50 | 1,137 | ,208 |
| | 2,00 | 33 | 3,76 | 1,091 | ,190 |
| Pitanje6 | 1,00 | 30 | 2,97 | 1,217 | ,222 |
| | 2,00 | 33 | 4,03 | ,918 | ,160 |
| Pitanje7 | 1,00 | 30 | 4,03 | 1,245 | ,227 |
| | 2,00 | 33 | 3,88 | 1,083 | ,188 |
| Pitanje8 | 1,00 | 30 | 2,87 | 1,196 | ,218 |
| | 2,00 | 33 | 2,33 | 1,216 | ,212 |
| Pitanje9 | 1,00 | 30 | 3,27 | 1,143 | ,209 |
| | 2,00 | 33 | 4,00 | 1,031 | ,179 |
| Pitanje10 | 1,00 | 30 | 2,70 | 1,317 | ,240 |
| | 2,00 | 33 | 3,73 | 1,126 | ,196 |
| Pitanje11 | 1,00 | 30 | 3,57 | ,971 | ,177 |
| | 2,00 | 33 | 4,15 | 1,228 | ,214 |
| Pitanje12 | 1,00 | 30 | 2,57 | 1,040 | ,190 |
| | 2,00 | 33 | 1,94 | 1,197 | ,208 |
| Pitanje13 | 1,00 | 30 | 2,90 | 1,062 | ,194 |
| | 2,00 | 33 | 4,09 | ,914 | ,159 |
| Pitanje14 | 1,00 | 30 | 2,57 | 1,478 | ,270 |
| | 2,00 | 33 | 3,15 | 1,349 | ,235 |
| Pitanje15 | 1,00 | 30 | 3,27 | 1,202 | ,219 |
| | 2,00 | 33 | 2,30 | 1,212 | ,211 |
| Pitanje16 | 1,00 | 30 | 1,73 | 1,081 | ,197 |

| | | | | | |
|-----------|------|----|------|-------|------|
| | 2,00 | 33 | 3,00 | 1,392 | ,242 |
| Pitanje17 | 1,00 | 30 | 3,53 | 1,383 | ,252 |
| | 2,00 | 33 | 4,58 | ,614 | ,107 |
| Pitanje18 | 1,00 | 30 | 3,03 | 1,066 | ,195 |
| | 2,00 | 33 | 2,12 | ,893 | ,155 |
| Pitanje19 | 1,00 | 30 | 3,63 | ,809 | ,148 |
| | 2,00 | 33 | 4,12 | ,650 | ,113 |
| Pitanje20 | 1,00 | 30 | 2,53 | 1,042 | ,190 |
| | 2,00 | 33 | 3,48 | ,972 | ,169 |
| Pitanje21 | 1,00 | 30 | 3,23 | 1,104 | ,202 |
| | 2,00 | 33 | 2,15 | 1,149 | ,200 |
| Pitanje22 | 1,00 | 30 | 3,57 | ,817 | ,149 |
| | 2,00 | 33 | 2,67 | ,957 | ,167 |
| Pitanje23 | 1,00 | 30 | 4,17 | ,913 | ,167 |
| | 2,00 | 33 | 4,33 | ,692 | ,120 |
| Pitanje24 | 1,00 | 30 | 2,33 | 1,155 | ,211 |
| | 2,00 | 33 | 1,61 | ,864 | ,150 |
| Pitanje25 | 1,00 | 30 | 4,00 | ,643 | ,117 |
| | 2,00 | 33 | 4,39 | ,556 | ,097 |

Након тога се спроводи Independent-Samples T-тест, где је резултат приказан у табели 27.

Табела 27. резултат T-теста којим се тестирају ставови групе испитаника код којих је наслеђен систем предузећа стар до 10 година и групе испитаника код којих је наслеђен систем предузећа стар преко 10 година

Independent Samples Test

| | Levene's Test for Equality of Variances | | t-test for Equality of Means | | | | | | |
|-------------------------------------|---|------|------------------------------|--------|------------------------|--------------------|--------------------------|---|-------|
| | F | Sig. | t | df | Sig. (2- tailed) | Mean Difference | Std. Error Difference | 95% Confidence Interval of the Difference | |
| | | | | | | | | Lower | Upper |
| Pitanje3 Equal variances assumed | ,800 | ,375 | -4,479 | 61 | ,000 | -1,258 | ,281 | -1,819 | -,696 |
| Equal variances not assumed | | | -4,470 | 59,851 | ,000 | -1,258 | ,281 | -1,820 | -,695 |
| Pitanje6 Equal variances assumed | 2,805 | ,099 | -3,938 | 61 | ,000 | -1,064 | ,270 | -1,604 | -,524 |
| Equal variances not assumed | | | -3,886 | 53,728 | ,000 | -1,064 | ,274 | -1,612 | -,515 |

| | | | | | | | | | | |
|-----------|-----------------------------|-------|------|--------|--------|------|--------|------|--------|-------|
| Pitanje7 | Equal variances assumed | ,204 | ,653 | ,527 | 61 | ,600 | ,155 | ,293 | -4,32 | ,741 |
| | Equal variances not assumed | | | ,523 | 57,814 | ,603 | ,155 | ,295 | -4,37 | ,746 |
| Pitanje8 | Equal variances assumed | ,363 | ,549 | 1,752 | 61 | ,085 | ,533 | ,304 | -,075 | 1,142 |
| | Equal variances not assumed | | | 1,754 | 60,612 | ,085 | ,533 | ,304 | -,075 | 1,142 |
| Pitanje9 | Equal variances assumed | 1,054 | ,309 | -2,678 | 61 | ,009 | -,733 | ,274 | -1,281 | -,186 |
| | Equal variances not assumed | | | -2,665 | 58,676 | ,010 | -,733 | ,275 | -1,284 | -,183 |
| Pitanje10 | Equal variances assumed | 1,328 | ,254 | -3,337 | 61 | ,001 | -1,027 | ,308 | -1,643 | -,412 |
| | Equal variances not assumed | | | -3,312 | 57,375 | ,002 | -1,027 | ,310 | -1,648 | -,406 |
| Pitanje11 | Equal variances assumed | ,572 | ,452 | -2,082 | 61 | ,042 | -,585 | ,281 | -1,146 | -,023 |
| | Equal variances not assumed | | | -2,106 | 59,901 | ,039 | -,585 | ,278 | -1,140 | -,029 |
| Pitanje12 | Equal variances assumed | ,381 | ,540 | 2,210 | 61 | ,031 | ,627 | ,284 | ,060 | 1,195 |
| | Equal variances not assumed | | | 2,225 | 60,883 | ,030 | ,627 | ,282 | ,063 | 1,191 |
| Pitanje13 | Equal variances assumed | ,038 | ,846 | -4,783 | 61 | ,000 | -1,191 | ,249 | -1,689 | -,693 |
| | Equal variances not assumed | | | -4,749 | 57,555 | ,000 | -1,191 | ,251 | -1,693 | -,689 |
| Pitanje14 | Equal variances assumed | ,646 | ,425 | -1,642 | 61 | ,106 | -,585 | ,356 | -1,297 | ,127 |
| | Equal variances not assumed | | | -1,635 | 58,927 | ,107 | -,585 | ,358 | -1,301 | ,131 |
| Pitanje15 | Equal variances assumed | ,326 | ,570 | 3,165 | 61 | ,002 | ,964 | ,304 | ,355 | 1,572 |
| | Equal variances not assumed | | | 3,167 | 60,524 | ,002 | ,964 | ,304 | ,355 | 1,572 |
| Pitanje16 | Equal variances assumed | 2,389 | ,127 | -4,005 | 61 | ,000 | -1,267 | ,316 | -1,899 | -,634 |
| | Equal variances not assumed | | | -4,054 | 59,593 | ,000 | -1,267 | ,312 | -1,892 | -,642 |

| | | | | | | | | | | |
|-----------|-----------------------------|--------|------|--------|--------|------|--------|------|--------|-------|
| Pitanje17 | Equal variances assumed | 20,355 | ,000 | -3,928 | 61 | ,000 | -1,042 | ,265 | -1,573 | -,512 |
| | Equal variances not assumed | | | -3,802 | 39,181 | ,000 | -1,042 | ,274 | -1,597 | -,488 |
| Pitanje18 | Equal variances assumed | 2,141 | ,149 | 3,693 | 61 | ,000 | ,912 | ,247 | ,418 | 1,406 |
| | Equal variances not assumed | | | 3,662 | 56,833 | ,001 | ,912 | ,249 | ,413 | 1,411 |
| Pitanje19 | Equal variances assumed | 3,608 | ,062 | -2,650 | 61 | ,010 | -,488 | ,184 | -,856 | -,120 |
| | Equal variances not assumed | | | -2,623 | 55,660 | ,011 | -,488 | ,186 | -,861 | -,115 |
| Pitanje20 | Equal variances assumed | ,269 | ,606 | -3,750 | 61 | ,000 | -,952 | ,254 | -1,459 | -,444 |
| | Equal variances not assumed | | | -3,738 | 59,372 | ,000 | -,952 | ,255 | -1,461 | -,442 |
| Pitanje21 | Equal variances assumed | ,553 | ,460 | 3,802 | 61 | ,000 | 1,082 | ,285 | ,513 | 1,651 |
| | Equal variances not assumed | | | 3,809 | 60,800 | ,000 | 1,082 | ,284 | ,514 | 1,650 |
| Pitanje22 | Equal variances assumed | 1,126 | ,293 | 3,993 | 61 | ,000 | ,900 | ,225 | ,449 | 1,351 |
| | Equal variances not assumed | | | 4,023 | 60,773 | ,000 | ,900 | ,224 | ,453 | 1,347 |
| Pitanje23 | Equal variances assumed | ,251 | ,618 | -,821 | 61 | ,415 | -,167 | ,203 | -,573 | ,239 |
| | Equal variances not assumed | | | -,810 | 53,896 | ,421 | -,167 | ,206 | -,579 | ,246 |
| Pitanje24 | Equal variances assumed | 2,035 | ,159 | 2,847 | 61 | ,006 | ,727 | ,255 | ,216 | 1,238 |
| | Equal variances not assumed | | | 2,809 | 53,472 | ,007 | ,727 | ,259 | ,208 | 1,247 |
| Pitanje25 | Equal variances assumed | 3,195 | ,079 | -2,608 | 61 | ,011 | -,394 | ,151 | -,696 | -,092 |
| | Equal variances not assumed | | | -2,589 | 57,646 | ,012 | -,394 | ,152 | -,699 | -,089 |

6.2.3.2.3. Тестирање ставова испитаника у односу на програмски језик Progress

У SPSS алату креирају се 2 групе:

- група 1 – којој одговара да је стари ИС написан у окружењу од којих ни једно није Progress (вредност 1,00)

- група 2 – којој одговара да је стари ИС написан у Progress окружењу (вредност 2,00)

Групи 2 припадају сви они ставови испитаника, који су на питање под редним бројем 5 одговорили Progress. Т-тест нам служи да упоредимо да ли се ставови по одређеним питањима у вези наслеђених система у просеку разликују између испитаника који раде у водоводним предузећима који имају Progress окружење у склопу тог система у односу на испитане из предузећа који немају Progress окружење унутар наслеђеног система.

У SPSS алату прво се добијају дескриптивне статистике за обе групе.

Табела 28. дескриптивне статистике за групе испитаника код којих је наслеђен систем предузећа у којем раде није написан у програмском окружењу Progress и оних код којих је наслеђен систем предузећа у којем раде написан у програмском окружењу Progress

Group Statistics

| | Programskijezik | N | Mean | Std. Deviation | Std. Error Mean |
|-----------|-----------------|----|------|----------------|-----------------|
| Pitanje3 | 1,00 | 59 | 3,08 | 1,277 | ,166 |
| | 2,00 | 4 | 4,25 | ,500 | ,250 |
| Pitanje6 | 1,00 | 59 | 3,47 | 1,209 | ,157 |
| | 2,00 | 4 | 4,25 | ,500 | ,250 |
| Pitanje7 | 1,00 | 59 | 3,92 | 1,179 | ,153 |
| | 2,00 | 4 | 4,50 | ,577 | ,289 |
| Pitanje8 | 1,00 | 59 | 2,64 | 1,242 | ,162 |
| | 2,00 | 4 | 1,75 | ,500 | ,250 |
| Pitanje9 | 1,00 | 59 | 3,61 | 1,160 | ,151 |
| | 2,00 | 4 | 4,25 | ,500 | ,250 |
| Pitanje10 | 1,00 | 59 | 3,20 | 1,349 | ,176 |
| | 2,00 | 4 | 3,75 | ,500 | ,250 |
| Pitanje11 | 1,00 | 59 | 3,88 | 1,146 | ,149 |
| | 2,00 | 4 | 3,75 | 1,258 | ,629 |
| Pitanje12 | 1,00 | 59 | 2,24 | 1,165 | ,152 |
| | 2,00 | 4 | 2,25 | 1,258 | ,629 |
| Pitanje13 | 1,00 | 59 | 3,47 | 1,165 | ,152 |
| | 2,00 | 4 | 4,25 | ,500 | ,250 |
| Pitanje14 | 1,00 | 59 | 2,86 | 1,420 | ,185 |
| | 2,00 | 4 | 3,00 | 1,826 | ,913 |
| Pitanje15 | 1,00 | 59 | 2,81 | 1,306 | ,170 |
| | 2,00 | 4 | 2,00 | ,816 | ,408 |
| Pitanje16 | 1,00 | 59 | 2,37 | 1,413 | ,184 |
| | 2,00 | 4 | 2,75 | 1,258 | ,629 |

| | | | | | |
|-----------|------|----|------|-------|------|
| Pitanje17 | 1,00 | 59 | 4,05 | 1,195 | ,156 |
| | 2,00 | 4 | 4,50 | ,577 | ,289 |
| Pitanje18 | 1,00 | 59 | 2,61 | 1,083 | ,141 |
| | 2,00 | 4 | 1,75 | ,500 | ,250 |
| Pitanje19 | 1,00 | 59 | 3,86 | ,776 | ,101 |
| | 2,00 | 4 | 4,25 | ,500 | ,250 |
| Pitanje20 | 1,00 | 59 | 3,00 | 1,114 | ,145 |
| | 2,00 | 4 | 3,50 | 1,000 | ,500 |
| Pitanje21 | 1,00 | 59 | 2,73 | 1,257 | ,164 |
| | 2,00 | 4 | 1,75 | ,500 | ,250 |
| Pitanje22 | 1,00 | 59 | 3,15 | ,979 | ,127 |
| | 2,00 | 4 | 2,25 | ,957 | ,479 |
| Pitanje23 | 1,00 | 59 | 4,24 | ,817 | ,106 |
| | 2,00 | 4 | 4,50 | ,577 | ,289 |
| Pitanje24 | 1,00 | 59 | 2,00 | 1,083 | ,141 |
| | 2,00 | 4 | 1,25 | ,500 | ,250 |
| Pitanje25 | 1,00 | 59 | 4,20 | ,637 | ,083 |
| | 2,00 | 4 | 4,25 | ,500 | ,250 |

Након тога се спроводи Independent-Samples T-тест, где је резултат приказан у табели 29.

Табела 29. резултат T-теста којим се тестирају ставови групе испитаника код којих је наслеђен систем предузећа није написан у програмском окружењу Progress и групе испитаника код којих је наслеђен систем предузећа је написан у програмском окружењу Progress

Independent Samples Test

| | Levene's Test for Equality of Variances | | t-test for Equality of Means | | | | | | |
|----------------------------------|---|------|------------------------------|-------|-----------------|-----------------|-----------------------|---|-------|
| | F | Sig. | t | df | Sig. (2-tailed) | Mean Difference | Std. Error Difference | 95% Confidence Interval of the Difference | |
| | | | | | | | | Lower | Upper |
| | | | | | | | | | |
| Pitanje3 Equal variances assumed | 3,425 | ,069 | -1,804 | 61 | ,076 | -1,165 | ,646 | -2,457 | ,126 |
| Equal variances not assumed | | | -3,881 | 6,177 | ,008 | -1,165 | ,300 | -1,895 | -,436 |
| Pitanje6 Equal variances assumed | 4,275 | ,043 | -1,268 | 61 | ,210 | -,775 | ,612 | -1,998 | ,447 |

| | | | | | | | | | | |
|-----------|-----------------------------------|-------|------|--------|-------|------|-------|------|--------|-------|
| | Equal variances not assumed | | | -2,625 | 5,800 | ,041 | -,775 | ,295 | -1,504 | -,047 |
| Pitanje7 | Equal variances assumed | 1,884 | ,175 | -,979 | 61 | ,332 | -,585 | ,597 | -1,780 | ,610 |
| | Equal variances not assumed | | | -1,789 | 4,915 | ,135 | -,585 | ,327 | -1,430 | ,260 |
| Pitanje8 | Equal variances assumed | 3,619 | ,062 | 1,422 | 61 | ,160 | ,894 | ,629 | -,363 | 2,151 |
| | Equal variances not assumed | | | 3,003 | 5,983 | ,024 | ,894 | ,298 | ,165 | 1,623 |
| Pitanje9 | Equal variances assumed | 4,302 | ,042 | -1,090 | 61 | ,280 | -,640 | ,587 | -1,814 | ,534 |
| | Equal variances not assumed | | | -2,191 | 5,550 | ,075 | -,640 | ,292 | -1,369 | ,089 |
| Pitanje10 | Equal variances assumed | 4,694 | ,034 | -,801 | 61 | ,426 | -,547 | ,682 | -1,911 | ,817 |
| | Equal variances not assumed | | | -1,789 | 6,609 | ,119 | -,547 | ,306 | -1,278 | ,185 |
| Pitanje11 | Equal variances assumed | ,008 | ,930 | ,221 | 61 | ,826 | ,131 | ,595 | -1,059 | 1,321 |
| | Equal variances not assumed | | | ,203 | 3,346 | ,851 | ,131 | ,647 | -1,811 | 2,074 |
| Pitanje12 | Equal variances assumed | ,095 | ,758 | -,021 | 61 | ,983 | -,013 | ,604 | -1,221 | 1,196 |
| | Equal variances not assumed | | | -,020 | 3,358 | ,985 | -,013 | ,647 | -1,953 | 1,928 |
| Pitanje13 | Equal variances assumed | 4,079 | ,048 | -1,315 | 61 | ,193 | -,775 | ,590 | -1,955 | ,404 |

| | | | | | | | | | | |
|---------------|-----------------------------------|-------|------|--------|-------|------|-------|------|--------|-------|
| | Equal variances not assumed | | | -2,652 | 5,575 | ,041 | -,775 | ,292 | -1,504 | -,047 |
| Pitanje1 4 | Equal variances assumed | ,607 | ,439 | -,182 | 61 | ,856 | -,136 | ,745 | -1,626 | 1,355 |
| | Equal variances not assumed | | | -,146 | 3,251 | ,893 | -,136 | ,931 | -2,974 | 2,703 |
| Pitanje1 5 | Equal variances assumed | 2,378 | ,128 | 1,224 | 61 | ,226 | ,814 | ,665 | -,516 | 2,143 |
| | Equal variances not assumed | | | 1,840 | 4,125 | ,137 | ,814 | ,442 | -,400 | 2,027 |
| Pitanje1 6 | Equal variances assumed | ,937 | ,337 | -,519 | 61 | ,605 | -,377 | ,726 | -1,829 | 1,075 |
| | Equal variances not assumed | | | -,575 | 3,534 | ,600 | -,377 | ,655 | -2,296 | 1,542 |
| Pitanje1 7 | Equal variances assumed | ,811 | ,371 | -,741 | 61 | ,461 | -,449 | ,606 | -1,660 | ,762 |
| | Equal variances not assumed | | | -1,370 | 4,975 | ,229 | -,449 | ,328 | -1,293 | ,395 |
| Pitanje1 8 | Equal variances assumed | 4,478 | ,038 | 1,568 | 61 | ,122 | ,860 | ,549 | -,237 | 1,957 |
| | Equal variances not assumed | | | 2,997 | 5,185 | ,029 | ,860 | ,287 | ,130 | 1,590 |
| Pitanje1 9 | Equal variances assumed | ,781 | ,380 | -,976 | 61 | ,333 | -,386 | ,395 | -1,176 | ,404 |
| | Equal variances not assumed | | | -1,430 | 4,054 | ,225 | -,386 | ,270 | -1,130 | ,359 |
| Pitanje2 0 | Equal variances assumed | ,071 | ,790 | -,873 | 61 | ,386 | -,500 | ,573 | -1,646 | ,646 |

Група 2 покрива све оне испитанике који су на питање под редним бројем 5 одговорили да је у питању програмски језик четврте генерације, тј. Progress, ABAP, SAP, Visual FoxPro, FoxPro, Harbour и Delphi.

У односу на претходно спроведени т-тест, овде је група обухватила програмске језике четврте генерације, јер Progress је заступљен веома мало, само 4 пута, па се намећу питања о ваљаности статистичког закључивања, јер је друга група има других 59 вредности истог статистичког обележја. С'тога у циљу бољег закључивања, проширујемо групу на програмске језике четврте генерације, којима Progress припада.

Т-тест нам служи да упоредимо да ли се ставови по одређеним питањима у вези наслеђених система у просеку разликују између испитаника који раде у водоводним предузећима који имају програмски језик четврте генерације у склопу тог система у односу на испитане из предузећа који немају програмски језик четврте генерације унутар наслеђеног система.

У SPSS алату прво се добијају дескриптивне статистике за обе групе.

Табела 30. дескриптивне статистике за групе испитаника код којих је наслеђен систем предузећа у којем раде није написан у програмском језику 4. генерације и оних код којих је наслеђен систем предузећа у којем раде је написан у програмском језику 4. генерације

Group Statistics

| | Programski_jezik2 | N | Mean | Std. Deviation | Std. Error Mean |
|-----------|-------------------|----|------|----------------|-----------------|
| Pitanje3 | 1,00 | 52 | 3,00 | 1,268 | ,176 |
| | 2,00 | 11 | 3,91 | 1,044 | ,315 |
| Pitanje6 | 1,00 | 52 | 3,46 | 1,212 | ,168 |
| | 2,00 | 11 | 3,82 | 1,079 | ,325 |
| Pitanje7 | 1,00 | 52 | 3,88 | 1,182 | ,164 |
| | 2,00 | 11 | 4,27 | 1,009 | ,304 |
| Pitanje8 | 1,00 | 52 | 2,67 | 1,264 | ,175 |
| | 2,00 | 11 | 2,18 | ,982 | ,296 |
| Pitanje9 | 1,00 | 52 | 3,58 | 1,144 | ,159 |
| | 2,00 | 11 | 4,00 | 1,095 | ,330 |
| Pitanje10 | 1,00 | 52 | 3,19 | 1,284 | ,178 |
| | 2,00 | 11 | 3,45 | 1,508 | ,455 |
| Pitanje11 | 1,00 | 52 | 3,88 | 1,114 | ,155 |
| | 2,00 | 11 | 3,82 | 1,328 | ,400 |
| Pitanje12 | 1,00 | 52 | 2,29 | 1,177 | ,163 |
| | 2,00 | 11 | 2,00 | 1,095 | ,330 |
| Pitanje13 | 1,00 | 52 | 3,38 | 1,140 | ,158 |

| | | | | | | |
|-----------|------|--|----|------|-------|------|
| | 2,00 | | 11 | 4,18 | ,982 | ,296 |
| Pitanje14 | 1,00 | | 52 | 2,87 | 1,415 | ,196 |
| | 2,00 | | 11 | 2,91 | 1,578 | ,476 |
| Pitanje15 | 1,00 | | 52 | 2,87 | 1,284 | ,178 |
| | 2,00 | | 11 | 2,27 | 1,272 | ,384 |
| Pitanje16 | 1,00 | | 52 | 2,31 | 1,435 | ,199 |
| | 2,00 | | 11 | 2,82 | 1,168 | ,352 |
| Pitanje17 | 1,00 | | 52 | 4,00 | 1,221 | ,169 |
| | 2,00 | | 11 | 4,45 | ,820 | ,247 |
| Pitanje18 | 1,00 | | 52 | 2,62 | 1,051 | ,146 |
| | 2,00 | | 11 | 2,27 | 1,191 | ,359 |
| Pitanje19 | 1,00 | | 52 | 3,83 | ,785 | ,109 |
| | 2,00 | | 11 | 4,18 | ,603 | ,182 |
| Pitanje20 | 1,00 | | 52 | 2,98 | 1,057 | ,147 |
| | 2,00 | | 11 | 3,27 | 1,348 | ,407 |
| Pitanje21 | 1,00 | | 52 | 2,79 | 1,242 | ,172 |
| | 2,00 | | 11 | 2,09 | 1,136 | ,343 |
| Pitanje22 | 1,00 | | 52 | 3,27 | ,866 | ,120 |
| | 2,00 | | 11 | 2,27 | 1,191 | ,359 |
| Pitanje23 | 1,00 | | 52 | 4,17 | ,834 | ,116 |
| | 2,00 | | 11 | 4,64 | ,505 | ,152 |
| Pitanje24 | 1,00 | | 52 | 2,04 | 1,120 | ,155 |
| | 2,00 | | 11 | 1,55 | ,688 | ,207 |
| Pitanje25 | 1,00 | | 52 | 4,17 | ,648 | ,090 |
| | 2,00 | | 11 | 4,36 | ,505 | ,152 |

Након тога се спроводи Independent-Samples T-тест, где је резултат приказан у табели 31.

Табела 31. резултат T-теста којим се тестирају ставови групе испитаника код којих наслеђен систем предузећа није написан у програмском језику 4. генерације и групе испитаника код којих је наслеђен систем предузећа написан у програмском језику 4. генерације

Independent Samples Test

| | Levene's Test for Equality of Variances | | t-test for Equality of Means | | | | | | | |
|--|---|------|------------------------------|----|--------------------|--------------------|--------------------------|---|-------|--|
| | F | Sig. | t | df | Sig. (2-tailed) | Mean Difference | Std. Error Difference | 95% Confidence Interval of the Difference | | |
| | | | | | | | | Lower | Upper | |
| Pitanje3 Equal variances assumed | 1,454 | ,232 | -2,220 | 61 | ,030 | -,909 | ,410 | -1,728 | -,090 | |

| | | | | | | | | | | |
|-----------|--------------------------------|-------|------|--------|--------|------|-------|------|--------|-------|
| | Equal variances not assumed | | | -2,520 | 16,886 | ,022 | -,909 | ,361 | -1,670 | -,148 |
| Pitanje6 | Equal variances assumed | ,978 | ,327 | -,902 | 61 | ,371 | -,357 | ,395 | -1,147 | ,434 |
| | Equal variances not assumed | | | -,974 | 15,833 | ,345 | -,357 | ,366 | -1,133 | ,420 |
| Pitanje7 | Equal variances assumed | ,605 | ,440 | -1,012 | 61 | ,316 | -,388 | ,384 | -1,155 | ,379 |
| | Equal variances not assumed | | | -1,123 | 16,383 | ,278 | -,388 | ,346 | -1,119 | ,343 |
| Pitanje8 | Equal variances assumed | 1,430 | ,236 | 1,211 | 61 | ,230 | ,491 | ,406 | -,320 | 1,302 |
| | Equal variances not assumed | | | 1,428 | 17,811 | ,171 | ,491 | ,344 | -,232 | 1,214 |
| Pitanje9 | Equal variances assumed | 1,395 | ,242 | -1,122 | 61 | ,266 | -,423 | ,377 | -1,177 | ,331 |
| | Equal variances not assumed | | | -1,155 | 14,986 | ,266 | -,423 | ,366 | -1,204 | ,358 |
| Pitanje10 | Equal variances assumed | ,533 | ,468 | -,597 | 61 | ,553 | -,262 | ,439 | -1,140 | ,616 |
| | Equal variances not assumed | | | -,537 | 13,243 | ,600 | -,262 | ,488 | -1,315 | ,790 |
| Pitanje11 | Equal variances assumed | ,211 | ,648 | ,174 | 61 | ,863 | ,066 | ,382 | -,698 | ,831 |
| | Equal variances not assumed | | | ,155 | 13,142 | ,879 | ,066 | ,429 | -,860 | ,993 |
| Pitanje12 | Equal variances assumed | 1,616 | ,209 | ,747 | 61 | ,458 | ,288 | ,386 | -,484 | 1,061 |
| | Equal variances not assumed | | | ,783 | 15,303 | ,446 | ,288 | ,368 | -,495 | 1,072 |
| Pitanje13 | Equal variances assumed | ,941 | ,336 | -2,153 | 61 | ,035 | -,797 | ,370 | -1,538 | -,057 |
| | Equal variances not assumed | | | -2,376 | 16,263 | ,030 | -,797 | ,336 | -1,508 | -,087 |
| Pitanje14 | Equal variances assumed | ,526 | ,471 | -,091 | 61 | ,928 | -,044 | ,479 | -1,001 | ,914 |
| | Equal variances not assumed | | | -,085 | 13,611 | ,934 | -,044 | ,515 | -1,151 | 1,063 |
| Pitanje15 | Equal variances assumed | ,054 | ,817 | 1,393 | 61 | ,169 | ,593 | ,425 | -,258 | 1,443 |

| | | | | | | | | | | |
|-----------|-----------------------------|-------|------|--------|--------|------|-------|------|--------|-------|
| | Equal variances not assumed | | | 1,402 | 14,640 | ,182 | ,593 | ,423 | -,311 | 1,496 |
| Pitanje16 | Equal variances assumed | 1,283 | ,262 | -1,103 | 61 | ,275 | -,510 | ,463 | -1,436 | ,415 |
| | Equal variances not assumed | | | -1,262 | 17,072 | ,224 | -,510 | ,404 | -1,364 | ,343 |
| Pitanje17 | Equal variances assumed | ,318 | ,575 | -1,176 | 61 | ,244 | -,455 | ,386 | -1,227 | ,318 |
| | Equal variances not assumed | | | -1,517 | 20,677 | ,144 | -,455 | ,300 | -1,078 | ,169 |
| Pitanje18 | Equal variances assumed | ,054 | ,817 | ,960 | 61 | ,341 | ,343 | ,357 | -,371 | 1,056 |
| | Equal variances not assumed | | | ,884 | 13,493 | ,392 | ,343 | ,388 | -,491 | 1,177 |
| Pitanje19 | Equal variances assumed | 1,245 | ,269 | -1,410 | 61 | ,164 | -,355 | ,252 | -,858 | ,148 |
| | Equal variances not assumed | | | -1,675 | 18,005 | ,111 | -,355 | ,212 | -,800 | ,090 |
| Pitanje20 | Equal variances assumed | 2,627 | ,110 | -,793 | 61 | ,431 | -,292 | ,368 | -1,029 | ,445 |
| | Equal variances not assumed | | | -,676 | 12,727 | ,511 | -,292 | ,432 | -1,228 | ,644 |
| Pitanje21 | Equal variances assumed | ,641 | ,427 | 1,715 | 61 | ,091 | ,698 | ,407 | -,116 | 1,511 |
| | Equal variances not assumed | | | 1,819 | 15,500 | ,088 | ,698 | ,383 | -,117 | 1,513 |
| Pitanje22 | Equal variances assumed | 3,647 | ,061 | 3,239 | 61 | ,002 | ,997 | ,308 | ,381 | 1,612 |
| | Equal variances not assumed | | | 2,632 | 12,331 | ,021 | ,997 | ,379 | ,174 | 1,819 |
| Pitanje23 | Equal variances assumed | 1,140 | ,290 | -1,769 | 61 | ,082 | -,463 | ,262 | -,987 | ,060 |
| | Equal variances not assumed | | | -2,425 | 23,358 | ,023 | -,463 | ,191 | -,858 | -,068 |
| Pitanje24 | Equal variances assumed | 2,226 | ,141 | 1,400 | 61 | ,166 | ,493 | ,352 | -,211 | 1,197 |
| | Equal variances not assumed | | | 1,904 | 22,948 | ,070 | ,493 | ,259 | -,043 | 1,029 |
| Pitanje25 | Equal variances assumed | ,011 | ,915 | -,916 | 61 | ,363 | -,191 | ,208 | -,607 | ,226 |

| | | | | | | | | | |
|--------------------------------|--|--|--------|--------|------|-------|------|-------|------|
| Equal variances not assumed | | | -1,078 | 17,783 | ,295 | -,191 | ,177 | -,562 | ,181 |
|--------------------------------|--|--|--------|--------|------|-------|------|-------|------|

6.2.3.3. Корелација

Корелација је степен везе између променљивих које се посматрају и она се испитује корелационом анализом.

Мера корелације се изражава коефицијентом корелације који може узети вредности у интервалу од -1 до 1.

Корелација се може поделити према:

- смеру (позитивна или негативна)
- броју посматраних променљивих (проста или сложена)
- облику везе (линеарна или нелинеарна)

Када испитујемо везу између две променљиве ради се о простој корелацији, а уколико испитујемо везу између више од две променљиве реч је о вишеструкој корелацији.

Позитивна корелација јесте:

- ако X расте, онда расте и Y
- ако X опада, онда опада и Y

Негативна корелација јесте:

- ако X расте, онда опада Y
- ако X опада, онда расте Y

Одговори на питања под редним бројем 3 и на питања од редног броја 6 до редног броја 25 су мерени ординалном скалом, па из тог разлога користимо Спирменов коефицијент корелације.

Коефицијент корелације може узети вредности између -1 и 1, чиме се указује на јачину везе између две променљиве. Вредност 0 показује да никаква веза (корелација) не постоји међу променљивим. Вредност 1 показује да је корелација потпуна и позитивна, док вредност -1 показује да је корелација потпуна и да је негативна.

Према Коену, важе следеће смернице (Soldić-Aleksić, 2011) :

- мала или слаба корелација је тамо где је вредност параметра r између 0,10 и 0,29

- средња корелација је тамо где је вредност параметра r између 0,30 и 0,49
- велика или јака корелација је тамо где је вредност параметра r између 0,5 и 1

Знак испред коефицијента корелације показује смер везе. Обзиром да постоји велики број корелационих веза, указаћемо на неке које су значајне за ово истраживање, пре свега код оних питања код којих се јављају средња и велика корелација између одговора на питања.

Корелација између питања број 3 и осталих питања

Корелација између одговора испитаника на питање број 3 и питања број 6, питања број 8, питања број 10, питања број 11, питања број 13, питања број 17, питања број 20 и питања број 25, је приказана на слици 57., која одговара резултату из SPSS алата.

| | | Correlations | | | | | | | | | |
|----------------|-----------|--------------|----------|----------|-----------|-----------|-----------|-----------|-----------|-----------|--|
| | | Pitanje3 | Pitanje6 | Pitanje8 | Pitanje10 | Pitanje11 | Pitanje13 | Pitanje17 | Pitanje20 | Pitanje25 | |
| Spearman's rho | Pitanje3 | 1,000 | ,668** | -,368** | ,422** | ,551** | ,531** | ,405* | ,542** | ,321* | |
| | | | ,000 | ,003 | ,001 | ,000 | ,000 | ,001 | ,000 | ,010 | |
| | | | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | |
| | Pitanje6 | ,668** | 1,000 | -,571** | ,686** | ,509* | ,704** | ,614** | ,625** | ,297* | |
| | | ,000 | ,000 | ,000 | ,000 | ,000 | ,000 | ,000 | ,000 | ,018 | |
| | | | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | |
| | Pitanje8 | -,368** | -,571** | 1,000 | -,532** | -,233 | -,520** | -,381** | -,491** | -,174 | |
| | | ,003 | ,000 | ,000 | ,000 | ,066 | ,000 | ,002 | ,000 | ,172 | |
| | | | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | |
| | Pitanje10 | ,422** | ,686** | -,532** | 1,000 | ,388** | ,604** | ,618** | ,649** | ,237* | |
| | ,001 | ,000 | ,000 | ,000 | ,002 | ,000 | ,000 | ,000 | ,062 | | |
| | | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | | |
| Pitanje11 | ,551** | ,509* | -,233 | ,388** | 1,000 | ,368** | ,436** | ,385** | ,279* | | |
| | ,000 | ,000 | ,066 | ,002 | ,000 | ,003 | ,000 | ,002 | ,027 | | |
| | | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | | |
| Pitanje13 | ,531** | ,704** | -,520** | ,604** | ,368** | 1,000 | ,584** | ,575** | ,267* | | |
| | ,000 | ,000 | ,000 | ,000 | ,003 | ,000 | ,000 | ,000 | ,035 | | |
| | | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | | |
| Pitanje17 | ,405* | ,614** | -,381** | ,618** | ,436** | ,584** | 1,000 | ,586** | ,310* | | |
| | ,001 | ,000 | ,002 | ,000 | ,000 | ,000 | ,000 | ,000 | ,013 | | |
| | | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | | |
| Pitanje20 | ,542** | ,625** | -,491** | ,649** | ,385** | ,575** | ,586** | 1,000 | ,306* | | |
| | ,000 | ,000 | ,000 | ,000 | ,002 | ,000 | ,000 | ,000 | ,015 | | |
| | | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | | |
| Pitanje25 | ,321* | ,297* | -,174 | ,237 | ,279* | ,267 | ,310* | ,306* | 1,000 | | |
| | ,010 | ,018 | ,172 | ,062 | ,027 | ,035 | ,013 | ,015 | ,000 | | |
| | | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | | |

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Слика 43. корелација између питања број 3 и осталих питања у пакету SPSS

Корелација између питања број 6 и осталих питања

Корелација између одговора испитаника на питање број 6 и питања број 8, питања број 10, питања број 12, питања број 17, питања број 18, питања број 19 и питања број 24, је

| | | | Correlations | | | | | | | |
|----------------|-----------|-------------------------|--------------|----------|-----------|-----------|-----------|-----------|-----------|-----------|
| | | | Pitanje6 | Pitanje8 | Pitanje10 | Pitanje12 | Pitanje17 | Pitanje18 | Pitanje19 | Pitanje24 |
| Spearman's rho | Pitanje6 | Correlation Coefficient | 1,000 | ,571** | ,686** | -,415** | ,614** | -,617** | ,502** | -,428** |
| | | Sig. (2-tailed) | . | ,000 | ,000 | ,001 | ,000 | ,000 | ,000 | ,000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje8 | Correlation Coefficient | -,571** | 1,000 | -,532** | ,585** | -,381** | ,417** | -,397** | ,472** |
| | | Sig. (2-tailed) | ,000 | . | ,000 | ,000 | ,002 | ,001 | ,001 | ,000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje10 | Correlation Coefficient | ,686** | -,532** | 1,000 | -,350** | ,618** | -,690** | ,482** | -,392** |
| | | Sig. (2-tailed) | ,000 | ,000 | . | ,005 | ,000 | ,000 | ,000 | ,001 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje12 | Correlation Coefficient | -,415** | ,585** | -,350** | 1,000 | -,328** | ,278* | -,479** | ,531** |
| | | Sig. (2-tailed) | ,001 | ,000 | ,005 | . | ,009 | ,027 | ,000 | ,000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje17 | Correlation Coefficient | ,614** | -,381** | ,618** | -,328** | 1,000 | -,573** | ,370** | -,250** |
| | | Sig. (2-tailed) | ,000 | ,002 | ,000 | ,009 | . | ,000 | ,003 | ,048 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje18 | Correlation Coefficient | -,617** | ,417** | -,690** | ,278* | -,573** | 1,000 | -,426** | ,169 |
| | | Sig. (2-tailed) | ,000 | ,001 | ,000 | ,027 | ,000 | . | ,001 | ,185 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje19 | Correlation Coefficient | ,502** | -,397** | ,482** | -,479** | ,370** | -,426** | 1,000 | -,207 |
| | | Sig. (2-tailed) | ,000 | ,001 | ,000 | ,000 | ,003 | ,001 | . | ,104 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje24 | Correlation Coefficient | -,428** | ,472** | -,392** | ,531** | -,250** | ,169 | -,207 | 1,000 |
| | | Sig. (2-tailed) | ,000 | ,000 | ,001 | ,000 | ,048 | ,185 | ,104 | . |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |

** Correlation is significant at the 0.01 level (2-tailed).

* Correlation is significant at the 0.05 level (2-tailed).

Слика 44. корелација између питања број 6 и осталих питања у пакету SPSS

приказана на слици 58., која одговара резултату из SPSS алата.

Корелација између питања број 7 и осталих питања

Корелација између одговора испитаника на питање број 7 и питања број 10, питања број 13, питања број 16, питања број 18 и питања број 20, је приказана на слици 59., која одговара резултату из SPSS алата.

| | | | Correlations | | | | | |
|----------------|-----------|-------------------------|--------------|-----------|-----------|-----------|-----------|-----------|
| | | | Pitanje7 | Pitanje10 | Pitanje13 | Pitanje16 | Pitanje18 | Pitanje20 |
| Spearman's rho | Pitanje7 | Correlation Coefficient | 1,000 | -,327** | -,303* | -,352** | ,317* | -,341** |
| | | Sig. (2-tailed) | . | ,009 | ,016 | ,005 | ,011 | ,006 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje10 | Correlation Coefficient | -,327** | 1,000 | ,604** | ,369** | -,690** | ,649** |
| | | Sig. (2-tailed) | ,009 | . | ,000 | ,003 | ,000 | ,000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje13 | Correlation Coefficient | -,303* | ,604** | 1,000 | ,383** | -,553** | ,575** |
| | | Sig. (2-tailed) | ,016 | ,000 | . | ,002 | ,000 | ,000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje16 | Correlation Coefficient | -,352** | ,369** | ,383** | 1,000 | -,514** | ,300* |
| | | Sig. (2-tailed) | ,005 | ,003 | ,002 | . | ,000 | ,017 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje18 | Correlation Coefficient | ,317* | -,690** | -,553** | -,514** | 1,000 | -,671** |
| | | Sig. (2-tailed) | ,011 | ,000 | ,000 | ,000 | . | ,000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje20 | Correlation Coefficient | -,341** | ,649** | ,575** | ,300* | -,671** | 1,000 |
| | | Sig. (2-tailed) | ,006 | ,000 | ,000 | ,017 | ,000 | . |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 |

** Correlation is significant at the 0.01 level (2-tailed).

* Correlation is significant at the 0.05 level (2-tailed).

Слика 45. корелација између питања број 7 и осталих питања у пакету SPSS

Корелација између питања број 8 и осталих питања

Корелација између одговора испитаника на питање број 8 и питања број 10, питања број 12, питања број 13, питања број 14, питања број 15, питања број 17, питања број 21 и питања број 24, је приказана на слици 60., која одговара резултату из SPSS

| | | Correlations | | | | | | | | | |
|----------------|-----------|-------------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|---------|
| | | Pitanje8 | Pitanje10 | Pitanje12 | Pitanje13 | Pitanje14 | Pitanje15 | Pitanje17 | Pitanje21 | Pitanje24 | |
| Spearman's rho | Pitanje8 | Correlation Coefficient | 1,000 | -.532** | .585** | -.520** | -.383** | .603** | -.381** | .425** | .472** |
| | | Sig. (2-tailed) | . | .000 | .000 | .002 | .000 | .002 | .001 | .000 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje10 | Pitanje10 | Correlation Coefficient | -.532** | 1,000 | -.350** | .604** | .683** | -.483** | .618** | -.505** | -.392** |
| | | Sig. (2-tailed) | .000 | . | .005 | .000 | .000 | .000 | .000 | .000 | .001 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje12 | Pitanje12 | Correlation Coefficient | .585** | -.350** | 1,000 | -.468** | -.202 | .575** | -.328** | .405** | .531** |
| | | Sig. (2-tailed) | .000 | .005 | . | .000 | .112 | .000 | .009 | .001 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje13 | Pitanje13 | Correlation Coefficient | -.520** | .604** | -.468** | 1,000 | .434** | -.541** | .584** | -.550** | -.547** |
| | | Sig. (2-tailed) | .000 | .000 | .000 | . | .000 | .000 | .000 | .000 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje14 | Pitanje14 | Correlation Coefficient | -.383** | .683** | -.202 | .434** | 1,000 | -.279* | .496** | -.141 | -.225 |
| | | Sig. (2-tailed) | .002 | .000 | .112 | .000 | . | .027 | .000 | .271 | .077 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje15 | Pitanje15 | Correlation Coefficient | .603** | -.483** | .575** | -.541** | -.279* | 1,000 | -.300* | .788** | .481** |
| | | Sig. (2-tailed) | .000 | .000 | .000 | .000 | .027 | . | .017 | .000 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje17 | Pitanje17 | Correlation Coefficient | -.381** | .618** | -.328** | .584** | .496** | -.300* | 1,000 | -.300* | -.250* |
| | | Sig. (2-tailed) | .002 | .000 | .009 | .000 | .000 | .017 | . | .017 | .048 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje21 | Pitanje21 | Correlation Coefficient | .425** | -.505** | .405** | -.550** | -.141 | .788** | -.300* | 1,000 | .453** |
| | | Sig. (2-tailed) | .001 | .000 | .001 | .000 | .271 | .000 | .017 | . | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje24 | Pitanje24 | Correlation Coefficient | .472** | -.392** | .531** | -.547** | -.225 | .481** | -.250* | .453** | 1,000 |
| | | Sig. (2-tailed) | .000 | .001 | .000 | .000 | .077 | .000 | .048 | .000 | . |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |

** Correlation is significant at the 0.01 level (2-tailed).

* Correlation is significant at the 0.05 level (2-tailed).

Слика 46. корелација између питања број 8 и осталих питања у пакету SPSS

алата.

Корелација између питања број 9 и осталих питања

Корелација између одговора испитаника на питање број 9 и питања број 13, питања број 19 и питања број 20, је приказана на слици 61., која одговара резултату из SPSS алата.

| | | Correlations | | | | |
|----------------|-----------|-------------------------|-----------|-----------|-----------|--------|
| | | Pitanje9 | Pitanje13 | Pitanje19 | Pitanje20 | |
| Spearman's rho | Pitanje9 | Correlation Coefficient | 1,000 | .396** | .311* | .343** |
| | | Sig. (2-tailed) | . | .001 | .013 | .006 |
| | | N | 63 | 63 | 63 | 63 |
| Pitanje13 | Pitanje13 | Correlation Coefficient | .396** | 1,000 | .491** | .575** |
| | | Sig. (2-tailed) | .001 | . | .000 | .000 |
| | | N | 63 | 63 | 63 | 63 |
| Pitanje19 | Pitanje19 | Correlation Coefficient | .311* | .491** | 1,000 | .480** |
| | | Sig. (2-tailed) | .013 | .000 | . | .000 |
| | | N | 63 | 63 | 63 | 63 |
| Pitanje20 | Pitanje20 | Correlation Coefficient | .343** | .575** | .480** | 1,000 |
| | | Sig. (2-tailed) | .006 | .000 | .000 | . |
| | | N | 63 | 63 | 63 | 63 |

** Correlation is significant at the 0.01 level (2-tailed).

* Correlation is significant at the 0.05 level (2-tailed).

Слика 47. корелација између питања број 9 и осталих питања у пакету SPSS

Корелација између питања број 10 и осталих питања

Корелација између одговора испитаника на питање број 10 и питања број 12, питања број 13, питања број 16, питања број 17, питања број 18, питања број 20 и питања број 24, је

| | | | Correlations | | | | | | | |
|----------------|-----------|-------------------------|--------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| | | | Pitanje10 | Pitanje12 | Pitanje13 | Pitanje16 | Pitanje17 | Pitanje18 | Pitanje20 | Pitanje24 |
| Spearman's rho | Pitanje10 | Correlation Coefficient | 1,000 | -.350** | .604** | .369** | .618** | -.690** | .649** | -.392** |
| | | Sig. (2-tailed) | . | .005 | .000 | .003 | .000 | .000 | .000 | .001 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje12 | | Correlation Coefficient | -.350** | 1,000 | -.468** | -.302* | -.328** | .278* | -.380** | .531** |
| | | Sig. (2-tailed) | .005 | . | .000 | .016 | .009 | .027 | .002 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje13 | | Correlation Coefficient | .604** | -.468** | 1,000 | .383** | .584** | -.553** | .575** | -.547** |
| | | Sig. (2-tailed) | .000 | .000 | . | .002 | .000 | .000 | .000 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje16 | | Correlation Coefficient | .369** | -.302* | .383** | 1,000 | .453** | -.514** | .300* | -.198 |
| | | Sig. (2-tailed) | .003 | .016 | .002 | . | .000 | .000 | .017 | .120 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje17 | | Correlation Coefficient | .618** | -.328** | .584** | .453** | 1,000 | -.573** | .586** | -.250* |
| | | Sig. (2-tailed) | .000 | .009 | .000 | .000 | . | .000 | .000 | .048 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje18 | | Correlation Coefficient | -.690** | .278* | -.553** | -.514** | -.573** | 1,000 | -.671** | .169 |
| | | Sig. (2-tailed) | .000 | .027 | .000 | .000 | .000 | . | .000 | .185 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje20 | | Correlation Coefficient | .649** | -.380** | .575** | .300* | .586** | -.671** | 1,000 | -.184 |
| | | Sig. (2-tailed) | .000 | .002 | .000 | .017 | .000 | .000 | . | .148 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje24 | | Correlation Coefficient | -.392** | .531** | -.547** | -.198 | -.250* | .169 | -.184 | 1,000 |
| | | Sig. (2-tailed) | .001 | .000 | .000 | .120 | .048 | .185 | .148 | . |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Слика 48. корелација између питања број 10 и осталих питања у пакету SPSS

приказана на слици 62., која одговара резултату из SPSS алата.

Корелација између питања број 11 и осталих питања

Корелација између одговора испитаника на питање број 11 и питања број 12, питања број 13, питања број 15, питања број 17, питања број 19, питања број 21 и питања број 22, је

на слици

одговара
из SPSS

приказана
63., која
результату
алата.

| | | | Correlations | | | | | | | |
|----------------|-----------|-------------------------|--------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| | | | Pitanje11 | Pitanje12 | Pitanje13 | Pitanje15 | Pitanje17 | Pitanje19 | Pitanje21 | Pitanje22 |
| Spearman's rho | Pitanje11 | Correlation Coefficient | 1,000 | -.391** | .368** | -.334** | .436** | .374** | -.399** | -.334** |
| | | Sig. (2-tailed) | . | .002 | .003 | .007 | .000 | .003 | .001 | .007 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje12 | Correlation Coefficient | -.391** | 1,000 | -.468** | .575** | -.328** | -.479** | .405** | .218 |
| | | Sig. (2-tailed) | .002 | . | .000 | .000 | .009 | .000 | .001 | .086 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje13 | Correlation Coefficient | .368** | -.468** | 1,000 | -.541** | .584** | .491** | -.550** | -.519** |
| | | Sig. (2-tailed) | .003 | .000 | . | .000 | .000 | .000 | .000 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje15 | Correlation Coefficient | -.334** | .575** | -.541** | 1,000 | -.300* | -.469** | .788** | .299 |
| | | Sig. (2-tailed) | .007 | .000 | .000 | . | .017 | .000 | .000 | .017 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje17 | Correlation Coefficient | .436** | -.328** | .584** | -.300* | 1,000 | .370** | -.300* | -.369** |
| | | Sig. (2-tailed) | .000 | .009 | .000 | .017 | . | .003 | .017 | .003 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje19 | Correlation Coefficient | .374** | -.479** | .491** | -.469** | .370** | 1,000 | -.467** | -.283* |
| | | Sig. (2-tailed) | .003 | .000 | .000 | .000 | .003 | . | .000 | .025 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje21 | Correlation Coefficient | -.399** | .405** | -.550** | .788** | -.300* | -.467** | 1,000 | .428** |
| | | Sig. (2-tailed) | .001 | .001 | .000 | .000 | .017 | .000 | . | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje22 | Correlation Coefficient | -.334** | .218 | -.519** | .299* | -.369** | -.283* | .428** | 1,000 |
| | | Sig. (2-tailed) | .007 | .086 | .000 | .017 | .003 | .025 | .000 | . |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Слика 49. корелација између питања број 11 и осталих питања у пакету SPSS

К

орелација између питања број 12 и осталих питања

Корелација између одговора испитаника на питање број 12 и питања број 13, питања број 15, питања број 17 и питања број 24, је приказана на слици 64., која одговара резултату из SPSS алата.

| | | | Correlations | | | | |
|----------------|-----------|-------------------------|--------------|-----------|-----------|-----------|-----------|
| | | | Pitanje12 | Pitanje13 | Pitanje15 | Pitanje17 | Pitanje24 |
| Spearman's rho | Pitanje12 | Correlation Coefficient | 1,000 | -.468** | .575** | -.328** | .531** |
| | | Sig. (2-tailed) | . | .000 | .000 | .009 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 |
| | Pitanje13 | Correlation Coefficient | -.468** | 1,000 | -.541** | .584** | -.547** |
| | | Sig. (2-tailed) | .000 | . | .000 | .000 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 |
| | Pitanje15 | Correlation Coefficient | .575** | -.541** | 1,000 | -.300* | .481** |
| | | Sig. (2-tailed) | .000 | .000 | . | .017 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 |
| | Pitanje17 | Correlation Coefficient | -.328** | .584** | -.300* | 1,000 | -.250* |
| | | Sig. (2-tailed) | .009 | .000 | .017 | . | .048 |
| | | N | 63 | 63 | 63 | 63 | 63 |
| | Pitanje24 | Correlation Coefficient | .531** | -.547** | .481** | -.250* | 1,000 |
| | | Sig. (2-tailed) | .000 | .000 | .000 | .048 | . |
| | | N | 63 | 63 | 63 | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Слика 50 корелација између питања број 12 и осталих питања у пакету SPSS

Корелација између питања број 13 и осталих питања

Корелација између одговора испитаника на питање број 13 и питања број 15, питања број 17, питања број 18, питања број 20, питања број 21, питања број 22 и питања број 24 је приказана на слици 65., која одговара резултату из SPSS алата.

| | | Correlations | | | | | | | |
|----------------|-----------|-------------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| | | Pitanje13 | Pitanje15 | Pitanje17 | Pitanje18 | Pitanje20 | Pitanje21 | Pitanje22 | Pitanje24 |
| Spearman's rho | Pitanje13 | Correlation Coefficient | 1,000 | -.541** | .584** | -.553** | .575** | -.550** | -.547** |
| | | Sig. (2-tailed) | . | .000 | .000 | .000 | .000 | .000 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje15 | Pitanje15 | Correlation Coefficient | -.541** | 1,000 | -.300* | .442** | -.514** | .788** | .299* |
| | | Sig. (2-tailed) | .000 | . | .017 | .000 | .000 | .000 | .017 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje17 | Pitanje17 | Correlation Coefficient | .584** | -.300* | 1,000 | -.573** | .586** | -.300* | -.250* |
| | | Sig. (2-tailed) | .000 | .017 | . | .000 | .000 | .017 | .003 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje18 | Pitanje18 | Correlation Coefficient | -.553** | .442** | -.573** | 1,000 | -.671** | .448** | .565** |
| | | Sig. (2-tailed) | .000 | .000 | .000 | . | .000 | .000 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje20 | Pitanje20 | Correlation Coefficient | .575** | -.514** | .586** | -.671** | 1,000 | -.449** | -.450** |
| | | Sig. (2-tailed) | .000 | .000 | .000 | .000 | . | .000 | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje21 | Pitanje21 | Correlation Coefficient | -.550** | .788** | -.300* | .448** | -.449** | 1,000 | .428** |
| | | Sig. (2-tailed) | .000 | .000 | .017 | .000 | .000 | . | .000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje22 | Pitanje22 | Correlation Coefficient | -.519** | .299* | -.369** | .565** | -.450** | .428** | 1,000 |
| | | Sig. (2-tailed) | .000 | .017 | .003 | .000 | .000 | .000 | . |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| Pitanje24 | Pitanje24 | Correlation Coefficient | -.547** | .481** | -.250* | .169 | -.184 | .453** | .355** |
| | | Sig. (2-tailed) | .000 | .000 | .048 | .185 | .148 | .000 | .004 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Слика 51. корелација између питања број 13 и осталих питања у пакету SPSS

Корелација између питања број 14 и осталих питања

Корелација између одговора испитаника на питање број 14 и питања број 17, питања број 18 и питања број 22 је приказана на слици 66., која одговара резултату из SPSS алата.

| | | Correlations | | | |
|----------------|-----------|-------------------------|-----------|-----------|-----------|
| | | Pitanje14 | Pitanje17 | Pitanje18 | Pitanje22 |
| Spearman's rho | Pitanje14 | Correlation Coefficient | 1,000 | .496** | -.480** |
| | | Sig. (2-tailed) | . | .000 | .000 |
| | | N | 63 | 63 | 63 |
| Pitanje17 | Pitanje17 | Correlation Coefficient | .496** | 1,000 | -.573** |
| | | Sig. (2-tailed) | .000 | . | .000 |
| | | N | 63 | 63 | 63 |
| Pitanje18 | Pitanje18 | Correlation Coefficient | -.480** | -.573** | 1,000 |
| | | Sig. (2-tailed) | .000 | .000 | . |
| | | N | 63 | 63 | 63 |
| Pitanje22 | Pitanje22 | Correlation Coefficient | -.464** | -.369** | .565** |
| | | Sig. (2-tailed) | .000 | .003 | .000 |
| | | N | 63 | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

Слика 52. корелација између питања број 14 и осталих питања у пакету SPSS

Корелација између питања број 15 и осталих питања

Корелација између одговора испитаника на питање број 15 и питања број 16, питања број 18, питања број 19, питања број 20, питања број 21 и питања број 24 је

| | | | Correlations | | | | | | |
|----------------|-----------|-------------------------|--------------|-----------|-----------|-----------|-----------|-----------|-----------|
| | | | Pitanje15 | Pitanje16 | Pitanje18 | Pitanje19 | Pitanje20 | Pitanje21 | Pitanje24 |
| Spearman's rho | Pitanje15 | Correlation Coefficient | 1,000 | -,382** | ,442** | -,469** | -,514** | ,788** | ,481** |
| | | Sig. (2-tailed) | . | ,002 | ,000 | ,000 | ,000 | ,000 | ,000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje16 | Correlation Coefficient | -,382** | 1,000 | -,514** | ,295* | ,300* | -,371** | -,198 |
| | | Sig. (2-tailed) | ,002 | . | ,000 | ,019 | ,017 | ,003 | ,120 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje18 | Correlation Coefficient | ,442** | -,514** | 1,000 | -,426** | -,671** | ,448** | ,169 |
| | | Sig. (2-tailed) | ,000 | ,000 | . | ,001 | ,000 | ,000 | ,185 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje19 | Correlation Coefficient | -,469** | ,295* | -,426** | 1,000 | ,480** | -,467** | -,207 |
| | | Sig. (2-tailed) | ,000 | ,019 | ,001 | . | ,000 | ,000 | ,104 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje20 | Correlation Coefficient | -,514** | ,300* | -,671** | ,480** | 1,000 | -,449** | -,184 |
| | | Sig. (2-tailed) | ,000 | ,017 | ,000 | ,000 | . | ,000 | ,148 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje21 | Correlation Coefficient | ,788** | -,371** | ,448** | -,467** | -,449** | 1,000 | ,453** |
| | | Sig. (2-tailed) | ,000 | ,003 | ,000 | ,000 | ,000 | . | ,000 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje24 | Correlation Coefficient | ,481** | -,198 | ,169 | -,207 | -,184 | ,453** | 1,000 |
| | | Sig. (2-tailed) | ,000 | ,120 | ,185 | ,104 | ,148 | ,000 | . |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Слика 53. корелација између питања број 15 и осталих питања у пакету SPSS

приказана на слици 67., која одговара резултату из SPSS алата.

Корелација између питања број 16 и осталих питања

Корелација између одговора испитаника на питање број 16 и питања број 17, питања број 18, питања број 19, питања број 20 и питања број 21 је приказана на слици 68., која

| | | | Correlations | | | | |
|----------------|-----------|-------------------------|--------------|-----------|-----------|-----------|-----------|
| | | | Pitanje16 | Pitanje17 | Pitanje18 | Pitanje20 | Pitanje21 |
| Spearman's rho | Pitanje16 | Correlation Coefficient | 1,000 | ,453** | -,514** | ,300* | -,371** |
| | | Sig. (2-tailed) | . | ,000 | ,000 | ,017 | ,003 |
| | | N | 63 | 63 | 63 | 63 | 63 |
| | Pitanje17 | Correlation Coefficient | ,453** | 1,000 | -,573** | ,586** | -,300* |
| | | Sig. (2-tailed) | ,000 | . | ,000 | ,000 | ,017 |
| | | N | 63 | 63 | 63 | 63 | 63 |
| | Pitanje18 | Correlation Coefficient | -,514** | -,573** | 1,000 | -,671** | ,448** |
| | | Sig. (2-tailed) | ,000 | ,000 | . | ,000 | ,000 |
| | | N | 63 | 63 | 63 | 63 | 63 |
| | Pitanje20 | Correlation Coefficient | ,300* | ,586** | -,671** | 1,000 | -,449** |
| | | Sig. (2-tailed) | ,017 | ,000 | ,000 | . | ,000 |
| | | N | 63 | 63 | 63 | 63 | 63 |
| | Pitanje21 | Correlation Coefficient | -,371** | -,300* | ,448** | -,449** | 1,000 |
| | | Sig. (2-tailed) | ,003 | ,017 | ,000 | ,000 | . |
| | | N | 63 | 63 | 63 | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Слика 54. . корелација између питања број 16 и осталих питања у пакету SPSS

одговара резултату из SPSS алата.

Корелација између питања број 17 и осталих питања

Корелација између одговора испитаника на питање број 17 и питања број 18, питања број 20, питања број 21, питања број 22 и питања број 25 је приказана на слици 69, која

| | | | Correlations | | | | | |
|----------------|-----------|-------------------------|--------------|-----------|-----------|-----------|-----------|-----------|
| | | | Pitanje17 | Pitanje18 | Pitanje20 | Pitanje21 | Pitanje22 | Pitanje25 |
| Spearman's rho | Pitanje17 | Correlation Coefficient | 1,000 | -,573** | ,586** | -,300* | -,369** | ,310* |
| | | Sig. (2-tailed) | . | ,000 | ,000 | ,017 | ,003 | ,013 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje18 | Correlation Coefficient | -,573** | 1,000 | -,671** | ,448** | ,565** | -,273* |
| | | Sig. (2-tailed) | ,000 | . | ,000 | ,000 | ,000 | ,030 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje20 | Correlation Coefficient | ,586** | -,671** | 1,000 | -,449** | -,450** | ,306* |
| | | Sig. (2-tailed) | ,000 | ,000 | . | ,000 | ,000 | ,015 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje21 | Correlation Coefficient | -,300* | ,448** | -,449** | 1,000 | ,428** | -,333** |
| | | Sig. (2-tailed) | ,017 | ,000 | ,000 | . | ,000 | ,008 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje22 | Correlation Coefficient | -,369** | ,565** | -,450** | ,428** | 1,000 | -,145 |
| | | Sig. (2-tailed) | ,003 | ,000 | ,000 | ,000 | . | ,257 |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 |
| | Pitanje25 | Correlation Coefficient | ,310* | -,273* | ,306* | -,333** | -,145 | 1,000 |
| | | Sig. (2-tailed) | ,013 | ,030 | ,015 | ,008 | ,257 | . |
| | | N | 63 | 63 | 63 | 63 | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

*. Correlation is significant at the 0.05 level (2-tailed).

Слика 55. корелација између питања број 17 и осталих питања у пакету SPSS

одговара резултату из SPSS алата.

Корелација између питања број 18 и осталих питања

Корелација између одговора испитаника на питање број 18 и питања број 19, питања број 20 и питања број 21 је приказана на слици 70, која одговара резултату из SPSS алата.

| | | | Correlations | | | |
|----------------|-----------|-------------------------|--------------|-----------|-----------|-----------|
| | | | Pitanje18 | Pitanje19 | Pitanje20 | Pitanje21 |
| Spearman's rho | Pitanje18 | Correlation Coefficient | 1,000 | -,426** | -,671** | ,448** |
| | | Sig. (2-tailed) | . | ,001 | ,000 | ,000 |
| | | N | 63 | 63 | 63 | 63 |
| | Pitanje19 | Correlation Coefficient | -,426** | 1,000 | ,480** | -,467** |
| | | Sig. (2-tailed) | ,001 | . | ,000 | ,000 |
| | | N | 63 | 63 | 63 | 63 |
| | Pitanje20 | Correlation Coefficient | -,671** | ,480** | 1,000 | -,449** |
| | | Sig. (2-tailed) | ,000 | ,000 | . | ,000 |
| | | N | 63 | 63 | 63 | 63 |
| | Pitanje21 | Correlation Coefficient | ,448** | -,467** | -,449** | 1,000 |
| | | Sig. (2-tailed) | ,000 | ,000 | ,000 | . |
| | | N | 63 | 63 | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

Слика 56. . корелација између питања број 18 и осталих питања у пакету SPSS

Корелација између питања број 19 и осталих питања

Корелација између одговора испитаника на питање број 19 и питања број 20, питања број 21 и питања број 25 је приказана на слици 71., која одговара резултату из SPSS алата.

| | | | Pitanje19 | Pitanje20 | Pitanje21 | Pitanje25 |
|----------------|-----------|-------------------------|-----------|-----------|-----------|-----------|
| Spearman's rho | Pitanje19 | Correlation Coefficient | 1,000 | ,480** | -,467** | ,410** |
| | | Sig. (2-tailed) | . | ,000 | ,000 | ,001 |
| | | N | 63 | 63 | 63 | 63 |
| | Pitanje20 | Correlation Coefficient | ,480** | 1,000 | -,449** | ,306* |
| | | Sig. (2-tailed) | ,000 | . | ,000 | ,015 |
| | | N | 63 | 63 | 63 | 63 |
| | Pitanje21 | Correlation Coefficient | -,467** | -,449** | 1,000 | -,333** |
| | | Sig. (2-tailed) | ,000 | ,000 | . | ,008 |
| | | N | 63 | 63 | 63 | 63 |
| | Pitanje25 | Correlation Coefficient | ,410** | ,306* | -,333** | 1,000 |
| | | Sig. (2-tailed) | ,001 | ,015 | ,008 | . |
| | | N | 63 | 63 | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

*. Correlation is significant at the 0.05 level (2-tailed).

Слика 57. корелација између питања број 19 и осталих питања у пакету SPSS

Корелација између питања број 20 и питања број 21

Корелација између одговора испитаника на питање број 20 и питања број 21, је приказана на слици 72., која одговара резултату из SPSS алата.

| | | | Pitanje20 | Pitanje21 |
|----------------|-----------|-------------------------|-----------|-----------|
| Spearman's rho | Pitanje20 | Correlation Coefficient | 1,000 | -,449** |
| | | Sig. (2-tailed) | . | ,000 |
| | | N | 63 | 63 |
| | Pitanje21 | Correlation Coefficient | -,449** | 1,000 |
| | | Sig. (2-tailed) | ,000 | . |
| | | N | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

Слика 58. корелација између питања број 20 и питања број 21 у пакету SPSS

Корелација између питања број 21 и питања број 22 и питања број 24

Корелација између одговора испитаника на питање број 21 и питања број 22, питања број 24 је приказана на слици 73., која одговара резултату из SPSS алата.

| Correlations | | | Pitanje21 | Pitanje22 | Pitanje24 |
|----------------|-----------|-------------------------|-----------|-----------|-----------|
| Spearman's rho | Pitanje21 | Correlation Coefficient | 1,000 | ,428** | ,453** |
| | | Sig. (2-tailed) | . | ,000 | ,000 |
| | | N | 63 | 63 | 63 |
| | Pitanje22 | Correlation Coefficient | ,428** | 1,000 | ,355** |
| | | Sig. (2-tailed) | ,000 | . | ,004 |
| | | N | 63 | 63 | 63 |
| | Pitanje24 | Correlation Coefficient | ,453** | ,355** | 1,000 |
| | | Sig. (2-tailed) | ,000 | ,004 | . |
| | | N | 63 | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

Слика 59. корелација између питања број 21 и питања број 22 и питања број 24 у пакету SPSS

Корелација између питања број 22 и питања број 24

Корелација између одговора испитаника на питање број 22 и питања број 24 је приказана на слици 74., која одговара резултату из SPSS алата.

| Correlations | | | Pitanje22 | Pitanje24 |
|----------------|-----------|-------------------------|-----------|-----------|
| Spearman's rho | Pitanje22 | Correlation Coefficient | 1,000 | ,355** |
| | | Sig. (2-tailed) | . | ,004 |
| | | N | 63 | 63 |
| | Pitanje24 | Correlation Coefficient | ,355** | 1,000 |
| | | Sig. (2-tailed) | ,004 | . |
| | | N | 63 | 63 |

** . Correlation is significant at the 0.01 level (2-tailed).

Слика 60. корелација између питања број 22 и питања број 24 у пакету SPSS

Корелација између питања број 23 и питања број 24 и питања број 25.

Корелација између одговора испитаника на питање број 23 и питања број 24 и питања број 25 је приказана на слици 75., која одговара резултату из SPSS алата.

| | | | Pitanje23 | Pitanje24 | Pitanje25 |
|----------------|-----------|-------------------------|-----------|-----------|-----------|
| Spearman's rho | Pitanje23 | Correlation Coefficient | 1,000 | -,412** | ,517** |
| | | Sig. (2-tailed) | . | ,001 | ,000 |
| | | N | 63 | 63 | 63 |
| | Pitanje24 | Correlation Coefficient | -,412** | 1,000 | -,212 |
| | | Sig. (2-tailed) | ,001 | . | ,096 |
| | | N | 63 | 63 | 63 |
| | Pitanje25 | Correlation Coefficient | ,517** | -,212 | 1,000 |
| | | Sig. (2-tailed) | ,000 | ,096 | . |
| | | N | 63 | 63 | 63 |

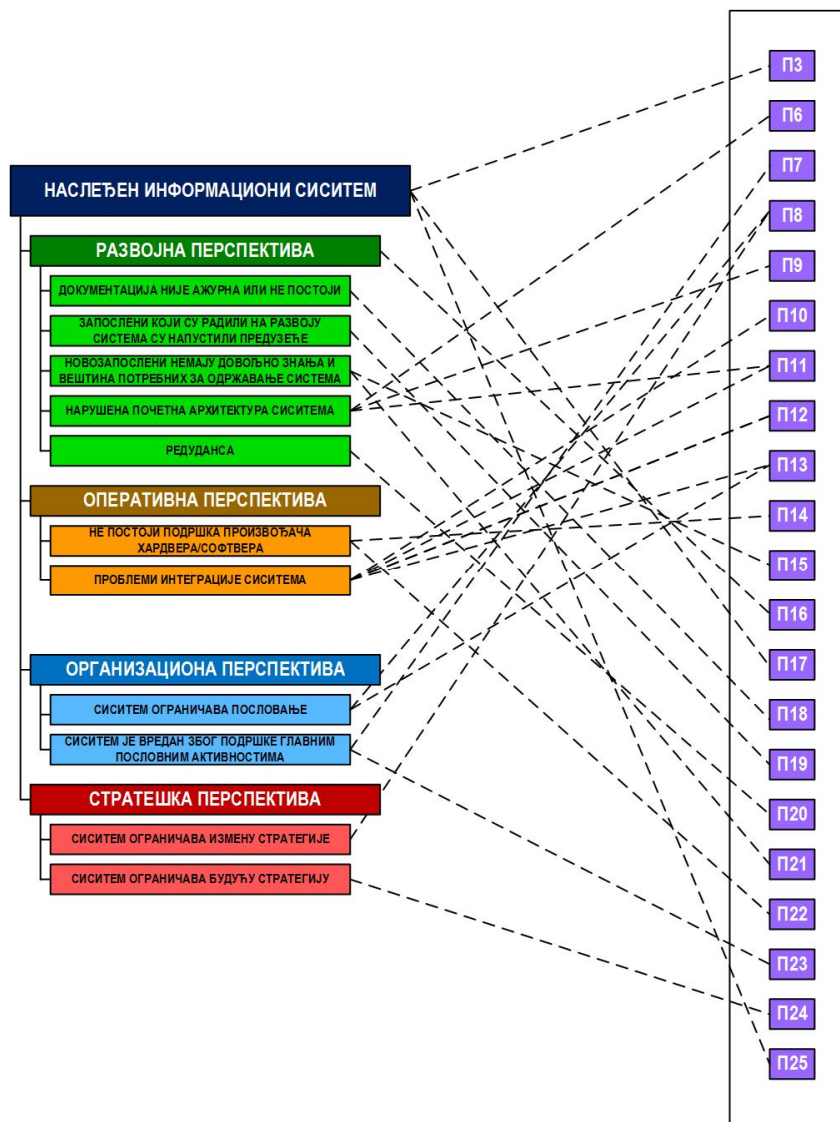
** . Correlation is significant at the 0.01 level (2-tailed).

Слика 61. корелација између питања број 23 и питања број 24 и питања број 25 у пакету SPSS

7. Резултати истраживања

7.1. Евалуација предложеног теоријског оквира

Резултати ове фазе истраживања треба да покажу да ли је предложени теоријски оквир применљив и изван ЈКП „Београдски водовод и канализација“ на предузећима која се баве производњом воде и услугама одвођења отпадних вода, тј. водоводима. У ту сврху је креирана слика 62. која показује повезаност елемената наслеђеног система и њихових перспектива са питањима упитника.



Слика 62. повезаност елемената перспектива наслеђеног система са питањима упитника

7.1.1. Развојна перспектива

7.1.1.1. Анализа одговора на питање П18

Према резултатима дескриптивне статистике ставови испитаника на питање П18, имају **аритметичку средину 2,56**, стандардну девијацију 1,074 и коефицијент варијације 42%. Из поменутих података може се закључити да **испитаници сматрају да за наслеђени ИС документација не постоји или није ажурна, у просеку**. Поред тога, може се уочити да постоји већа варијација у њиховим одговорима.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 2,56, док испитаници из земаља у региону имају аритметичку средину од 2,55. Из тога се закључује да **испитаници из Србије и испитаници из земаља у региону имају ставове који се слажу да наслеђени ИС нема документацију или она није ажурна, у просеку**.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При томе, ставови испитаника код којих је ИС стар мање од 10 година имају аритметичку средину 3,03 док испитаници код којих је ИС њиховог предузећа старији од 10 година имају аритметичку средину 2,12. Из тога се закључује да **испитаници чије предузеће има ИС стар више од 10 година имају ставове по којима се слажу да наслеђени ИС нема документацију или да она није ажурна, у просеку**.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа није реализован у програмском језику 4. генерације и испитаника чији је ИС предузећа реализован у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника код којих ИС предузећа није реализован у програмском језику 4. генерације имају аритметичку средину 2,62, а ставови испитаника код којих ИС предузећа јесте реализован у програмском језику 4. генерације имају нешто мању аритметичку средину од 2,27. Из тога се закључује да **испитаници код којих ИС предузећа није реализован у**

програмском језику 4. генерације и они код којих је ИС предузећа реализован у програмском језику 4. генерације имају ставове по којима се слажу да наслеђени ИС нема документацију или да она није ажурна, у просеку, при чему је став испитаника код којих је ИС предузећа реализован у програмском језику 4. генерације нешто израженији.

Корелациона анализа указује на јаку негативну корелациону везу од $-0,671$ са нивоом значајности од 1% између питања П18 и П20, што значи да ставови испитаника који виде да документација наслеђеног ИС постоји и ажурна је уједно и сматрају да редуданса у систему није изражена. Такође, треба истаћи да питање П20 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.

Корелациона анализа указује на средњу позитивну корелациону везу од $0,448$ са нивоом значајности од 1% између питања П18 и П21, што значи да ставови испитаника који виде да је документација наслеђеног ИС постоји и ажурна је уједно и сматрају да новозапослени имају знање и вештине да у старту раде са наслеђеним ИС. Такође, треба истаћи да питање П21 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.

7.1.1.2. Анализа одговора на питање П19

Према резултатима дескриптивне статистике ставови испитаника на питање П19, имају аритметичку средину **3,89**, стандардну девијацију $0,764$ и коефицијент варијације 20%. Из поменутих података може се закључити да испитаници сматрају да постоји велики одлазак запослених који су радили на наслеђеном ИС, у просеку.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од $4,12$, док испитаници из земаља у региону имају аритметичку средину од $3,62$. Из тога се закључује да испитаници из Србије и испитаници из земаља у региону имају различите ставове да постоји велики одлазак запослених који су радили на наслеђеном ИС, где се обе групе испитаника слажу да постоји велики одлазак запослених, при чему испитаници из Србије имају израженије слагање са тим ставом, у просеку.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 3,63, док испитаници чији ИС предузећа је стар више од 10 година имају аритметичку средину од 4,12. Из тога се закључује да **испитаници чији ИС предузећа је стар мање од 10 година имају различите ставове, у односу на ставове испитаника чији ИС предузећа је старији од 10 година у вези одласка запослених који су радили на наслеђеном ИС, где се обе групе испитаника слажи да постоји велики одлазак запослених, при чему испитаници чији ИС предузећа је старији од 10 година имају израженије слагање са тим ставом, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа није реализован у програмском језику 4. генерације и испитаника чији је ИС предузећа реализован у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника код којих ИС предузећа није реализован у програмском језику 4. генерације имају аритметичку средину 3,83, а ставови испитаника код којих ИС предузећа јесте реализован у програмском језику 4. генерације имају нешто већу аритметичку средину од 4,18, чиме исказују веће слагање са ставом да постоји велики одлазак запослених који су радили на наслеђеном ИС. Из тога се закључује да **испитаници код којих ИС предузећа није реализован у програмском језику 4. генерације и они код којих је ИС предузећа реализован у програмском језику 4. генерације имају ставове по којима се слажу да постоји велики одлазак запослених који су радили на наслеђеном ИС.**

Корелациона анализа указује на средњу позитивну корелациону везу од 0,480 са нивоом значајности од 1% између питања П19 и П20, што значи да **ставови испитаника по којима се слажу да постоји велики одлазак запослених који су радили на наслеђеном ИС уједно и сматрају да је редуванса у систему изражена. Такође треба истаћи да питање П20 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.**

Корелациона анализа указује на средњу позитивну корелациону везу од 0,480 са нивоом значајности од 1% између питања П19 и П25, што значи да **ставови испитаника**

по којима се слажу да постоји велики одлазак запослених који су радили на наслеђеном ИС уједно и сматрају да овакав упитник може боље усмерити модернизацију система уместо посматрање само из једног угла. Такође треба истаћи да питање П25 подржава Главну хипотезу истраживања.

Корелациона анализа указује на средње негативну корелациону везу од $-0,467$ са нивоом значајности од 1% између питања П19 и П21, што значи да ставови испитаника по којима се слажу да постоји велики одлазак запослених који су радили на наслеђеном ИС уједно и сматрају да новозапослени немају знање и вештине да у старту раде са наслеђеним ИС. Такође треба истаћи да питање П21 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.

7.1.1.3. Анализа одговора на питање П15

Према резултатима дескриптивне статистике ставови испитаника на питање П15, имају аритметичку средину **2,76**, стандардну девијацију **1,292** и коефицијент варијације **47%**. Из поменутих података **не може се закључити да ли програмери могу да примене најновија знања, методологије, концепте и алате из области софтверског инжењерства, без проблема у наслеђеном ИС**, а може се видети да постоји већа варијација у њиховим одговорима.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају нешто мању аритметичку средину од **2,47**, у односу на испитанике из земаља у региону који имају аритметичку средину од **3,10**. Из тога се закључује да **испитаници из Србије имају став према којем се не слажу да програмери могу да примене најновија знања, методологије, концепте и алате из области софтверског инжењерства, без проблема у наслеђеном ИС**, у просеку, док испитаници из земаља у региону имају подељене ставове у вези тога.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају

аритметичку средину од 3,27, док испитаници чији ИС предузећа је стар више од 10 година имају аритметичку средину од 2,30. Из тога се закључује да **испитаници чији ИС предузећа је стар мање од 10 година имају различите ставове, у односу на ставове испитаника чији ИС предузећа је старији од 10 година у вези тога да ли програмери могу да примене најновија знања, методологије, концепте и алате из области софтверског инжењерства, без проблема у наслеђеном ИС. Испитаници чији је ИС старији од 10 година више нагињу ка томе да програмери не могу да промене најновија знања, методологије, концепте и алате из области софтверског инжењерства, у просеку док испитаници чији је ИС стар мање од 10 година имају подељено мишљење по том питању.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа није реализован у програмском језику 4. генерације и испитаника чији је ИС предузећа реализован у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника код којих ИС предузећа није реализован у програмском језику 4. генерације имају аритметичку средину 2,87, а ставови испитаника код којих ИС предузећа јесте реализован у програмском језику 4. генерације имају нешто мању аритметичку средину од 2,27, чиме исказују веће неслагање са ставом да програмери могу да примене најновија знања, методологије, концепте и алате из области софтверског инжењерства, без проблема у наслеђеном ИС. Из тога се закључује да **испитаници код којих ИС предузећа реализован у програмском језику 4. генерације имају ставове по којима се не слажу да програмери могу да примене најновија знања, методологије, концепте и алате из области софтверског инжењерства, без проблема у наслеђеном ИС.**

Корелациона анализа указује на јаку негативну корелациону везу од -0,514 са нивоом значајности од 1% између питања П15 и П20, што значи да **ставови испитаника по којима се слажу да програмери могу да примене најновија знања, методологије, концепте и алате из области софтверског инжењерства, без проблема у наслеђеном ИС уједно и сматрају да редуванса у систему није изражена. Такође треба истаћи да питање П20 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.**

Корелациона анализа указује на јаку позитивну корелациону везу од 0,788 са нивоом значајности од 1% између питања П15 и П21, што значи да **ставови испитаника по којима се слажу да програмери могу да примене најновија знања, методологије,**

концепте и алате из области софтверског инжењерства, без проблема у наслеђеном ИС уједно и сматрају да новозапослени имају знање и вештине да у старту раде са старим ИС. Такође треба истаћи да питање П21 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.

Корелациона анализа указује на средњу негативну корелациону везу од -0,382 са нивоом значајности од 1% између питања П15 и П16, што значи да ставови испитаника по којима се слажу да програмери могу да примене најновија знања, методологије, концепте и алате из области софтверског инжењерства, без проблема у наслеђеном ИС уједно и сматрају да је наслеђени ИС није развијен од стране кадрова и средстава самог предузећа (in-house). Такође треба истаћи да питање П16 је у вези је са посебном хипотезом 1.

Корелациона анализа указује на средњу позитивну корелациону везу од 0,481 са нивоом значајности од 1% између питања П15 и П24, што значи да ставови испитаника по којима се слажу да програмери могу да примене најновија знања, методологије, концепте и алате из области софтверског инжењерства, без проблема у наслеђеном ИС уједно и сматрају да наслеђени ИС може се лако интегрисати са елементима Индустије 4.0 (Cloud Computing, Internet of Things, Big Data, i td.). Такође треба истаћи да питање П24 припада стратешкој перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 4.

7.1.1.4. Анализа одговора на питање П21

Према резултатима дескриптивне статистике ставови испитаника на питање П21, имају **аритметичку средину 2,67**, стандардну девијацију 1,244 и коефицијент варијације 47%. Из поменутих података **не може се закључити да новозапослени имају знање и вештине да у старту раде са наслеђеним ИС**, а може се видети да постоји већа варијација у њиховим одговорима.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају нешто мању аритметичку средину од 2,29, у односу на испитанике из земаља у региону који имају аритметичку средину од 3,10. Из тога се закључује да **испитаници из Србије имају ставове да новозапослени немају знање и вештине да у старту раде са старим ИС, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 3,23, док испитаници чији ИС предузећа је стар више од 10 година имају аритметичку средину од 2,15. Из тога се закључује да **испитаници чији је ИС старији од 10 година имају ставове да новозапослени немају знање и вештине да у старту раде са старим ИС, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа није реализован у програмском језику 4. генерације и испитаника чији је ИС предузећа реализован у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника код којих ИС предузећа није реализован у програмском језику 4. генерације имају аритметичку средину 2,79, а ставови испитаника код којих ИС предузећа јесте реализован у програмском језику 4. генерације имају мању аритметичку средину од 2,09. Из тога се закључује да **испитаници код којих ИС предузећа реализован у програмском језику 4. генерације имају ставове по којима се не слажу да новозапослени имају знање и вештине да у старту раде са старим ИС, у просеку.**

Корелациона анализа указује на средњу позитивну корелациону везу од 0,428 са нивоом значајности од 1% између питања П21 и П22, што значи да **ставови испитаника по којима се слажу да новозапослени имају знање и вештине да у старту раде са старим ИС уједно и сматрају да постоји подршка приликом одржавања хардвера и софтвера од стране произвођача. Такође треба истаћи да питање П22 припада оперативној перспективи према предложеном теоријском оквиру је у вези је са посебном хипотезом 2.**

Корелациона анализа указује на средњу позитивну корелациону везу од 0,453 са нивоом значајности од 1% између питања П21 и П24, што значи да **ставови испитаника по којима се слажу да новозапослени имају знање и вештине да у старту раде са старим ИС уједно и сматрају да наслеђени ИС може се лако интегрисати са елементима Индустије 4.0 (Cloud Computing, Internet of Things, Big Data, i td.). Такође треба истаћи да питање П24 припада стратешкој перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 4.**

7.1.1.5. Анализа одговора на питање П6

Према резултатима дескриптивне статистике ставови испитаника на питање П6, имају **аритметичку средину 3,52**, стандардну девијацију 1,189 и коефицијент варијације 34%. Из поменутих података може се закључити да **испитаници сматрају да у наслеђеном ИС нарушен је иницијално постављен дизајн (пројектна мисао) услед великог броја измена и дугорочног одржавања, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају нешто већу аритметичку средину од 3,71, у односу на испитанике из земаља у региону који имају аритметичку средину од 3,31. Из тога се закључује да **испитаници из Србије имају ставове да у наслеђеном ИС нарушен је иницијално постављен дизајн (пројектна мисао) услед великог броја измена и дугорочног одржавања, у просеку, док су ставови испитаника из региона подељени по том питању, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 2,97, док испитаници чији ИС предузећа је стар више од 10 година имају аритметичку средину од 4,03. Из тога се закључује да **испитаници чије је ИС старији од 10 година имају ставове да у наслеђеном ИС нарушен је иницијално постављен дизајн (пројектна мисао) услед великог броја измена и дугорочног одржавања, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику Progress 4GL и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику Progress 4GL може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При чему ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику Progress 4GL имају аритметичку средину од 3,47, док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику Progress 4GL имају аритметичку средину од 4,25. Из тога се закључује да **испитаници чији наслеђени ИС**

предузећа јесте написан у програмском језику Progress 4GL имају ставове да у наслеђеном ИС нарушен је иницијално постављен дизајн (пројектна мисао) услед великог броја измена и дугорочног одржавања, у просеку.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При чему ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику четврте генерације имају аритметичку средину од 3,46, док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику четврте генерације имају аритметичку средину од 3,82. Из тога се закључује да испитаници чији наслеђени ИС предузећа јесте написан у програмском језику четврте генерације **имају ставове да у наслеђеном ИС нарушен је иницијално постављен дизајн (пројектна мисао) услед великог броја измена и дугорочног одржавања, у просеку.**

Корелациона анализа указује на јаку негативну корелациону везу од -0,571 са нивоом значајности од 1% између питања П6 и П8, што значи да **ставови испитаника по којима се слажу да у наслеђеном ИС је нарушен иницијално постављен дизајн (пројектна замисао) услед великог броја измена и дугорочног одржавања уједно и не сматрају да се у наслеђеном ИС релативно једноставно врше измене у погледу измене пословних процедура или пословних процеса, а у складу са захтевима. Такође треба истаћи да питање П8 припада и организационој и стратешкој перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 3 и посебном хипотезом 4.**

Корелациона анализа указује на јаку позитивну корелациону везу од 0,686 са нивоом значајности од 1% између питања П6 и П10, што значи да **ставови испитаника по којима се слажу да у наслеђеном ИС је нарушен иницијално постављен дизајн (пројектна замисао) услед великог броја измена и дугорочног одржавања уједно и сматрају да стари ИС има апликације (развијене за неку организациону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама. Такође треба истаћи да питање П10 припада оперативној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 2.**

Корелациона анализа указује на средњу негативну корелациону везу од -0,415 са нивоом значајности од 1% између питања П6 и П12, што значи да **ставови испитаника по којима се слажу да у наслеђеном ИС је наручен иницијално постављен дизајн (пројектна замисао) услед великог броја измена и дугорочног одржавања уједно и не сматрају да је интеграција старог ИС са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS и тд.) једноставна. Такође треба истаћи да питање П12 припада оперативној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 2.**

Корелациона анализа указује на јаку позитивну корелациону везу од 0,614 са нивоом значајности од 1% између питања П6 и П17, што значи да **ставови испитаника по којима се слажу да у наслеђеном ИС је наручен иницијално постављен дизајн (пројектна замисао) услед великог броја измена и дугорочног одржавања уједно и сматрају да је стари ИС потребно модернизовати. Такође треба истаћи да је питање 17 у вези са Главном хипотезом истраживања.**

Корелациона анализа указује на средње негативну корелациону везу од -0,428 са нивоом значајности од 1% између питања П6 и П24, што значи да **ставови испитаника по којима се слажу да у наслеђеном ИС је наручен иницијално постављен дизајн (пројектна замисао) услед великог броја измена и дугорочног одржавања уједно и не сматрају да стари ИС такав какав је (без измена) се може лако интегрисати са елементима Индустије 4.0 (Cloud Computing, Internet of Things, Big Data и тд.). Такође треба истаћи да питање П24 припада стратешкој перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 4.**

7.1.1.6. Анализа одговора на питање П9

Према резултатима дескриптивне статистике ставови испитаника на питање П9, имају **аритметичку средину 3,65**, стандардну девијацију 1,138 и коефицијент варијације 31%. Из поменутих података може се закључити да **испитаници сматрају да уколико се изврши измена у једном делу старог ИС, та промена се пропагира ка другим деловима система, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају нешто већу аритметичку средину од 3,74, у односу на испитанике из земаља

у региону који имају аритметичку средину од 3,55. Из тога се закључује да **испитаници из Србије и испитаницу из земаља региона имају ставове да уколико се изврши измена у једном делу старог ИС, та промена се пропагира ка другим деловима система, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 3,27, док испитаници чији ИС предузећа је стар више од 10 година имају аритметичку средину од 4,00. Из тога се закључује да **испитаници чије је ИС старији од 10 година имају ставове да ако се изврши измена у једном делу старог ИС, та промена се пропагира ка другим деловима система, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При чему ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику четврте генерације имају аритметичку средину од 3,58, док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику четврте генерације имају аритметичку средину од 4,00. Из тога се закључује да **испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације, заједно са испитаницима чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају ставове да у ако се изврши измена у једном делу старог ИС, та промена се пропагира ка другим деловима система, у просеку.**

Корелациона анализа указује на средњу позитивну корелациону везу од 0,396 са нивоом значајности од 1% између питања П9 и П13, што значи да **ставови испитаника по којима се слажу да ако се изврши измена у једном делу старог ИС, та промена се пропагира ка другим деловима система уједно и сматрају да постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизован или полу-аутоматизован начин. Такође треба истаћи да питање П13 је у вези са**

оперативном перспективом и организационом перспективом према предложеном теоријском оквиру, односно посебном хипотезом 2 и посебном хипотезом 3.

Корелациона анализа указује на средњу позитивну корелациону везу од 0,311 са нивоом значајности од 1% између питања П9 и П19, што значи да **ставови испитаника по којима се слажу да ако се изврши измена у једном делу старог ИС, та промена се пропагира ка другим деловима система уједно и сматрају да је велики одлазак запослених који су радили на старом ИС. Такође треба истаћи да питање П19 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.**

Корелациона анализа указује на средњу позитивну корелациону везу од 0,343 са нивоом значајности од 1% између питања П9 и П20, што значи да **ставови испитаника по којима се слажу да ако се изврши измена у једном делу старог ИС, та промена се пропагира ка другим деловима система уједно и сматрају да је редуванса у систему изражена. Такође треба истаћи да питање П20 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.**

7.1.1.7. Анализа одговора на питање П11

Према резултатима дескриптивне статистике ставови испитаника на питање П11, имају **аритметичку средину 3,87**, стандардну девијацију 1,143 и коефицијент варијације 30%. Из поменутих података може се закључити да **испитаници се слажу да заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 4,24, док испитаници из земаља у региону имају аритметичку средину од 3,45. Из тога се закључује да **испитаници из Србије и испитаници из земаља у региону имају ставове који се слажу да заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом

значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При томе, ставови испитаника код којих је ИС стар мање од 10 година имају аритметичку средину 3,57 док испитаници код којих је ИС њиховог предузећа старији од 10 година имају аритметичку средину 4,15. Из тога се закључује да **испитаници чије предузеће има ИС стар мање од 10 година, као и испитаници чије предузеће има ИС старији од 10 година имају ставове по којима се слажу да заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим, у просеку, при чему су ставови израженији код испитаника код којих је ИС старији од 10 година.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При чему ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику четврте генерације имају аритметичку средину од 3,88, док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику четврте генерације имају аритметичку средину од 3,82. Из тога се закључује да **испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације, заједно са испитаницима чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају ставове којима се слажу да заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим, у просеку.**

Корелациона анализа указује на средњу негативну корелациону везу од -0,391 са нивоом значајности од 1% између питања П11 и П12, што значи да **ставови испитаника по којима се слажу да заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим, а при томе уједно и не сматрају да интеграција старог ИС, са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS и тд.) је једноставна. Такође треба истаћи да питање П12 припада оперативној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 2.**

Корелациона анализа указује на средњу позитивну корелациону везу од 0,368 са нивоом значајности од 1% између питања П11 и П13, што значи да **ставови испитаника по којима се слажу да заменом старе апликације или дела старог ИС потенцијалним**

новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим, при чему уједно и сматрају да постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовано или полу-аутоматизовани начин. Такође треба истаћи да питање П13 је у вези са и са оперативном перспективом и организационом перспективом према предложеном теоријском оквиру, односно посебном хипотезом 2 и посебном хипотезом 3.

Корелациона анализа указује на средњу позитивну корелациону везу од 0,436 са нивоом значајности од 1% између питања П11 и П17, што значи да ставови испитаника по којима се слажу да заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим, уједно и сматрају да је стари ИС потребно модернизовати. Такође треба истаћи да је питање П17 у вези са Главном хипотезом истраживања.

7.1.1.8. Анализа одговора на питање П20

Према резултатима дескриптивне статистике ставови испитаника на питање П20, имају аритметичку средину **3,03**, стандардну девијацију 1,107 и коефицијент варијације 37%. Из поменутих података **не може се закључити да ли испитаници сматрају да редуданса у систему је изражена, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 3,15, док испитаници из земаља у региону имају аритметичку средину од 2,90. Из тога се закључује да **испитаници из Србије и испитаници из земаља у региону имају подељене ставове по питању да ли је редуданса у систему изражена, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При томе, ставови испитаника код којих је ИС стар мање од 10 година имају аритметичку средину 2,53 док испитаници код којих је ИС њиховог предузећа старији од 10 година имају аритметичку средину 3,48. Из тога се закључује да **испитаници чије предузеће има ИС стар мање од 10 година, не сматрају да је редуданса у систему изражена, док**

испитаници чије предузеће има ИС старији од 10 година имају ставове по којима се слажу да је редуанса у систему изражена, у просеку.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При чему ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику четврте генерације имају аритметичку средину од 2,98, док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику четврте генерације имају аритметичку средину аритметичку средину од 3,27. Из тога се закључује да испитаници чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације **имају подељене ставове по питању да ли је редуанса у систему изражена, у просеку.**

Корелациона анализа указује на средњу негативну корелациону везу од -0,449 са нивоом значајности од 1% између питања П20 и П21, што значи да **ставови испитаника по којима се слажу да је редуанса у систему изражена уједно и не сматрају да новозапослени имају знање и вештине да у старту раде са старим ИС. Такође, треба истаћи да питање П21 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.**

7.1.1.9. Анализа одговора на питање П16

Према резултатима дескриптивне статистике ставови испитаника на питање П16, имају **аритметичку средину 2,40**, стандардну девијацију 1,397 и коефицијент варијације 58%. Може се закључити да **се испитаници не слажу са ставом да стари ИС је развијен од стране кадрова и средстава самог предузећа (in-house ИС).** Такође може се уочити већа варијација у одговорима испитаника.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 2,47, док испитаници из земаља у региону имају аритметичку средину од 2,31. Из тога се закључује да испитаници из Србије имају исте ставове као и испитаници земаља у региону, а то је да се **да се испитаници не слажу са**

ставом да стари ИС је развијен од стране кадрова и средстава самог предузећа (in-house ИС) и међу њима нема значајнијих разлика, у просеку.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови испитаника чији ИС предузећа је стар мање од 10 година разликују у просеку од ставова испитаника чији је ИС предузећа стар више од 10 година. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 1,73, док испитаници чији је ИС предузећа стар више од 10 година имају аритметичку средину од 3,00. Из тога се закључује да испитаници чији ИС предузећа је стар мање од 10 година **се не слажу са ставом да стари ИС је развијен од стране кадрова и средстава самог предузећа (in-house ИС), у просеку док су испитаници чије је ИС стар више од 10 година подељених ставова по овом питању, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају аритметичку средину 2,31 док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају аритметичку средину 2,82. Из тога се закључује да **ставови испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације и ставови испитаника чији ИС није написан у програмском језику 4. генерације се не слажу са ставом да стари ИС је развијен од стране кадрова и средстава самог предузећа (in-house ИС), у просеку, при чему је то неслагање мање код испитаника чији наслеђени ИС је написан у програмском језику 4. генерације.**

Корелациона анализа указује на средњу позитивну корелациону везу од 0,453 са нивоом значајности од 1% између питања П16 и П17, што значи да **се ставови испитаника који се слажу са ставом да стари ИС је развијен од стране кадрова и средстава самог предузећа (in-house ИС), уједно и налазе да је стари ИС потребно модернизовати. Такође треба истаћи да је питање П17 у вези са Главном хипотезом истраживања.**

Корелациона анализа указује на јаку негативну корелациону везу од -0,514 са нивоом значајности од 1% између питања П16 и П18, што значи да се ставови испитаника који се слажу са ставом да стари ИС је развијен од стране кадрова и средстава самог предузећа (in-house ИС), уједно и налазе да је докуменација старог ИС постоји и ажурна је. Такође, треба истаћи да питање П18 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.

7.1.1.10. Развојна перспектива - резултат

На основу анализе одговора на питања П18, П19, П15, П21, П6, П9, П11, П20 и П16, креирана је табела 32., у којој су представљени резултати различитих статистичких показатеља у односу на ставове испитаника са којима они могу да сложе, да имају подељено мишљење и да се не сложе. Такође је приказан и одговор који иде у прилог Посебној хипотези 1 дисертације.

Табела 32. резултати анализе одговора на питања који су у вези са Посебном хипотезом 1 истраживања

| | | П18 | П19 | П15 | П21 | П6 | П9 | П11 | П20 | П16 |
|---|--|-----|-----|-----|-----|----|----|-----|-----|-----|
| | дескриптивна | Н | С | П | Н | С | С | С | П | Н |
| т-тест 1 | Србија | Н | С | Н | Н | С | С | С | П | Н |
| | земље региона | Н | С | П | П | П | С | С | П | Н |
| т-тест 2 | једнако или мање од 10 година | П | С | П | П | П | П | С | Н | Н |
| | више од 10 година | Н | С | Н | Н | С | С | С | С | П |
| т-тест 3 | систем није развијан у језику 4. генерац. | Н | С | П | П | С | С | С | П | Н |
| | систем јесте развијан у језику 4. генерац | Н | С | П | Н | С | С | С | П | П |
| т-тест 4 | систем није развијан у Progress платформи | | | | | С | | | | |
| | систем јесте развијан у Progress платформи | | | | | С | | | | |
| | одговор који иде у прилог посебној хипотези 1. | Н | С | Н | Н | С | С | С | С | Н |
| Легенда: <ul style="list-style-type: none"> • С - слажем се • П - подељено • Н - не слажем се | | | | | | | | | | |

Прво ћемо посматрати ставове испитаника у вези питања П18, П19, П21, П6, П9, П11, П16. Дескриптивна статистика је значајан показатељ, и резултати анализе на горе

наведена питања показују да се одговори испитаника поклапају са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 1. Поред тога, на основу резултата т-тестова уочава се да постоје специфична поклапања. У првом реду постоје поклапања одговора испитаника из Србије са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 1. Након тога, постоје поклапања одговора испитаника код којих је ИС старији од 10 година са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 1. Затим, постоје поклапања одговора испитаника код којих је наслеђени ИС развијен у програмском језику 4. генерације са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 1.

Друго ћемо посматрати ставове испитаника у вези питања П15 и П20. Декриптивна статистика је значајан показатељ, и резултати анализе на ова два питања указују да су ставови испитаника подељени, у односу на одговоре који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 1. На основу резултата т-тестова уочава се да постоје одређена специфична поклапања. У првом реду постоје поклапања одговора испитаника из Србије за питање П15 са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 1. Након тога, постоје поклапања одговора испитаника за оба питања код којих је ИС старији од 10 година са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 1. Посматрањем корелационих веза може се уочити да питање П15 има корелационе везе са питањима: П18 и то средњу позитивну корелациону везу, П19 и то средњу негативну корелациону везу, П21 и то јаку позитивну корелациону везу, а одговори испитаника на та три питања се поклапају са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 1. Посматрањем корелационих веза може се уочити да питање П20 има корелационе везе са питањима: П6 и то јаку позитивну корелациону везу, П9 и то средњу позитивну корелациону везу, П11 и то средњу позитивну корелациону везу, а одговори испитаника на та три питања се поклапају са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 1.

Треће и најважније, посматрамо одговоре испитаника на питање П25, које је уједно и најзначајније питање у упитнику у вези предмета истраживања. Декриптивна статистика указује да постоји позитиван став испитаника у вези целокупног упитника, чиме се потврђује релевантност свих питања која су део упитника.

На основу свега наведеног, може се закључити да је први део ПОСЕБНЕ ХИПОТЕЗЕ 1: „Посматрањем развојне перспективе наслеђеног система могуће је сагледати проблеме наслеђених система из улоге запослених који га развијају“, доказан.

Други део, ПОСЕБНЕ ХИПОТЕЗЕ 1: „а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема.“, ће бити обрађен у наредном поглављу.

7.1.2. Оперативна перспектива

7.1.2.1. *Анализа одговора на питање П14*

Према резултатима дескриптивне статистике ставови испитаника на питање П14, имају **аритметичку средину 2,87**, стандардну девијацију 1,431 и коефицијент варијације 50%. Из поменутих података **не може се закључити да ли испитаници сматрају да се стари ИС извршава на застарелом хардверу, тј. постоје подељена размишљања**. Поред тога, може се видети да постоји већа варијација у њиховим одговорима.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 3,06, док испитаници из земаља у региону имају аритметичку средину од 2,66. Из тога се закључује да **испитаници из Србије и испитаници из земаља у региону имају подељене ставове по питању да се стари ИС извршава на застарелом хардверу, у просеку**.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника код којих је ИС стар мање од 10 година имају аритметичку средину 2,57 док испитаници код којих је ИС њиховог предузећа старији од 10 година имају аритметичку средину 3,15. Из тога се закључује да **испитаници чије предузеће има ИС стар мање од 10 година и испитаници чије предузеће има ИС старији од 10 година имају подељене ставове по питању да се стари ИС извршава на застарелом хардверу, у просеку**.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При чему ставови

испитаника чији наслеђени ИС предузећа није написан у програмском језику четврте генерације имају аритметичку средину од 2,87, док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику четврте генерације имају аритметичку средину аритметичку средину од 2,91. Из тога се закључује да **испитаници чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају подељене ставове по питању да се стари ИС извршава на застарелом хардверу, у просеку.**

Корелациона анализа указује на средњу позитивну корелациону везу од 0,496 са нивоом значајности од 1% између питања П14 и П17, што значи да **ставови испитаника по којима се слажу да стари ИС се извршава на застарелом хардверу уједно и сматрају да стари ИС је потребно модернизовати. Такође треба истаћи да је питање П17 у вези са Главном хипотезом истраживања.**

Корелациона анализа указује на средњу негативну корелациону везу од -0,480 са нивоом значајности од 1% између питања П14 и П18, што значи да **ставови испитаника по којима се слажу да стари ИС се извршава на застарелом хардверу уједно се не слажу да документација старог ИС постоји и да је ажурна. Такође, треба истаћи да питање П18 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.**

Корелациона анализа указује на средњу негативну корелациону везу од -0,464 са нивоом значајности од 1% између питања П14 и П22, што значи да **ставови испитаника по којима се слажу да стари ИС се извршава на застарелом хардверу уједно се не слажу да постоји подршка приликом одржавања хардвера и софтвера од стране произвођача. Такође, треба истаћи да питање П22 припада оперативној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 2.**

7.1.2.2. Анализа одговора на питање П22

Према резултатима дескриптивне статистике ставови испитаника на питање П22, имају **аритметичку средину 3,10**, стандардну девијацију 0,995 и коефицијент варијације 32%. Из поменутих података **не може се закључити да ли испитаници сматрају да постоји подршка приликом одржавања харвера и софтвера од стране произвођача.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се

закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 3,03, док испитаници из земаља у региону имају аритметичку средину од 3,17. Из тога се закључује да **испитаници из Србије и испитаници из земаља у региону имају подељене ставове по питању да постоји подршка приликом одржавања харвера и софтвера од стране произвођача, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При томе, ставови испитаника код којих је ИС стар мање од 10 година имају аритметичку средину 3,57 док испитаници код којих је ИС њиховог предузећа старији од 10 година имају аритметичку средину 2,67. Из тога се закључује да **испитаници чије предузеће има ИС стар мање од 10 година слажу се са ставом да постоји подршка приликом одржавања харвера и софтвера од стране произвођача, у просеку, што се не може рећи за испитанике чије предузеће има ИС стар више од 10 година који имају подељене ставове по том питању, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При томе, ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају аритметичку средину 3,27 док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају аритметичку средину 2,27. Из тога се закључује да **испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације не слажу се са ставом да постоји подршка приликом одржавања харвера и софтвера од стране произвођача, у просеку.**

Корелациона анализа указује на средње позитивну корелациону везу од 0,355 са нивоом значајности од 1% између питања П22 и П24, што значи да **ставови испитаника по којима се слажу да постоји подршка приликом одржавања харвера и софтвера од стране произвођача уједно и сматрају да стари ИС такав какав је (без измена) се може лако интегрисати са елементима Индустрије 4.0 (Cloud Computing, Internet of**

Things, Big Data и тд.). Такође треба истаћи да питање П24 припада стратешкој перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 4.

7.1.2.3. Анализа одговора на питање П10

Према резултатима дескриптивне статистике ставови испитаника на питање П10, имају **аритметичку средину 3,24**, стандардну девијацију 1,316 и коефицијент варијације 41%. Из поменутих података **не може се закључити да ли испитаници сматрају да стари ИС има апликације (развијене за неку организаиону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама.** Поред тога, може се видети да постоји већа варијација у њиховим одговорима.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 3,59, док испитаници из земаља у региону имају аритметичку средину од 2,83. Из тога се закључује да испитаници из Србије се слажу са ставом да **стари ИС има апликације (развијене за неку организаиону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При томе, ставови испитаника код којих је ИС стар мање од 10 година имају аритметичку средину 2,70 док испитаници код којих је ИС њиховог предузећа старији од 10 година имају аритметичку средину 3,73. Из тога се закључује да **испитаници чије предузеће има ИС стар више од 10 година се слажу да стари ИС има апликације (развијене за неку организаиону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у

програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају аритметичку средину 3,19 док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају аритметичку средину 3,45. Из тога се закључује да **ставови испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације иако се значајно не разликују од ставова испитаника чији ИС није написан у програмском језику 4. генерације ипак више нагињу ка ставу да се слажу да стари ИС има апликације (развијене за неку организаиону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама, у просеку.**

Корелациона анализа указује на јаку позитивну корелациону везу од 0,604 са нивоом значајности од 1% између питања П10 и П13, што значи да **ставови испитаника по којима се слажу да стари ИС има апликације (развијене за неку организаиону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама уједно и сматрају да постоји проблем у размени података старог ИС са неким другим системом другог предузећа на аутоматизовани или полу-аутоматизовани начин. Такође треба истаћи да питање П13 припада оперативној перспективи и организационој перспективи и према предложеном теоријском оквиру у вези је са посебном хипотезом 2 и са посебном хипотезом 3.**

Корелациона анализа указује на јаку позитивну корелациону везу од 0,618 са нивоом значајности од 1% између питања П10 и П17, што значи да **ставови испитаника по којима се слажу да стари ИС има апликације (развијене за неку организаиону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама уједно и сматрају да је наслеђени ИС потребно модернизовати. Такође треба истаћи да је питање 17 у вези са Главном хипотезом истраживања.**

Корелациона анализа указује на јаку негативну корелациону везу од -0,690 са нивоом значајности од 1% између питања П10 и П18, што значи да **ставови испитаника по којима се слажу да стари ИС има апликације (развијене за неку организаиону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама уједно и се и не слажу са ставом да документација старог ИС постоји и да је ажурна. Такође, треба истаћи да питање**

П18 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.

Корелациона анализа указује на средњу негативну корелациону везу од -0,392 са нивоом значајности од 1% између питања П10 и П24, што значи да **ставови испитаника по којима се слажу да стари ИС има апликације (развијене за неку организациону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама уједно и се и не слажу са ставом да стари ИС, такав какав је (без измена) може се лако интегрисати са елементима Индустрије 4.0 (Cloud Computing, Internet of Things, Big Data, и тд.). Такође треба истаћи да питање П24 припада стратешкој перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 4.**

7.1.2.4. Анализа одговора на питање П11

Према резултатима дескриптивне статистике ставови испитаника на питање П11, имају **аритметичку средину 3,87**, стандардну девијацију 1,143 и коефицијент варијације 30%. Из поменутих података **може се закључити да испитаници сматрају да заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 4,24, док испитаници из земаља у региону имају аритметичку средину од 3,45. Из тога се закључује да испитаници из Србије се слажу са ставом да **заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 3,57, док ставови испитаника чији је ИС предузећа стар више од 10 година имају аритметичку средину од 4,15. Из тога се закључује да испитаници чији

ИС предузећа је стар мање од 10 година и испитаници чији је ИС предузећа стар више од 10 година се слажу са ставом да **заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим, у просеку, при чему испитаници чији је ИС предузећа стар више од 10 година исказују веће слагање.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају аритметичку средину 3,88 док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају аритметичку средину 3,82. Из тога се закључује да **ставови испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. и ставови испитаника чији ИС није написан у програмском језику 4. генерације се слажу да заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим, у просеку.**

Корелациона анализа указује на средњу негативну корелациону везу од -0,391 са нивоом значајности од 1% између питања П11 и П12, што значи да **ставови испитаника по којима се слажу да заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим уједно се и не слажу са ставом да је интеграција старог ИС, са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS, и тд.) једноставна. Такође треба истаћи да питање П12 припада оперативној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 2.**

Корелациона анализа указује на средњу позитивну корелациону везу од 0,436 са нивоом значајности од 1% између питања П11 и П17, што значи да **ставови испитаника по којима се слажу да заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим уједно се слажу са ставом да је стари ИС потребно модернизовати. Такође треба истаћи да је питање П17 у вези са Главном хипотезом истраживања.**

Корелациона анализа указује на средњу позитивну корелациону везу од 0,374 са нивоом значајности од 1% између питања П11 и П19, што значи да ставови испитаника по којима се слажу да заменом старе апликације или дела старог ИС потенцијалним новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим уједно се слажу са ставом да је велики одлазак запослених који су радили на старом ИС. Такође, треба истаћи да питање П19 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.

7.1.2.5. Анализа одговора на питање П12

Према резултатима дескриптивне статистике ставови испитаника на питање П12, имају аритметичку средину 2,24, стандардну девијацију 1,160 и коефицијент варијације 52%. Из поменутих података може се закључити да се испитаници не слажу са ставом да је интеграција старог ИС са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS и тд.) једноставна. Такође, присутна је велика варијација у ставовима испитаника.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 2,06, док испитаници из земаља у региону имају аритметичку средину од 2,45. Из тога се закључује да испитаници из Србије и земаља у региону се не слажу ставом да је интеграција старог ИС са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS и тд.) једноставна, у просеку.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 2,57, док ставови испитаника чији је ИС предузећа стар више од 10 година имају аритметичку средину од 1,94. Из тога се закључује да испитаници чији ИС предузећа је стар мање од 10 година и испитаници чији је ИС предузећа стар више од 10 година се не слажу ставом да је интеграција старог ИС са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS и тд.) једноставна, у просеку,

при чему испитаници чији је ИС предузећа стар више од 10 година исказују веће неслагање.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају аритметичку средину 2,29 док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају аритметичку средину 2,00. Из тога се закључује да **ставови испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. и ставови испитаника чији ИС није написан у програмском језику 4. генерације се не слажу да је интеграција старог ИС са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS и тд.) једноставна, у просеку.**

Корелациона анализа указује на средњу негативну корелациону везу од -0,468 са нивоом значајности од 1% између питања П12 и П13, што значи да **ставови испитаника по којима се слажу да је интеграција старог ИС са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS и тд.) једноставна уједно се и не слажу са ставом да постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин. Такође треба истаћи да питање П13 припада оперативној перспективи и организационој перспективи и према предложеном теоријском оквиру у вези је са посебном хипотезом 2 и са посебном хипотезом 3.**

Корелациона анализа указује на јаку позитивну корелациону везу од 0,575 са нивоом значајности од 1% између питања П12 и П15, што значи да **ставови испитаника по којима се слажу да је интеграција старог ИС са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS и тд.) једноставна уједно се и слажу са ставом да програмери могу да примене најновија знања, методологије, концепте и алате из области софтверског инжењерства, без проблема у старом ИС. Такође, треба истаћи да питање П15 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.**

Корелациона анализа указује на средњу негативну корелациону везу од -0,328 са нивоом значајности од 1% између питања П12 и П17, што значи да **ставови испитаника по којима се слажу да је интеграција старог ИС са неким другим системима у**

оквиру предузећа (GIS, SCADA, EDAMS и тд.) једноставна уједно се и не слажу са ставом да је стари ИС потребно модернизовати. Такође треба истаћи да је питање П17 у вези са Главном хипотезом истраживања.

Корелациона анализа указује на јаку позитивну корелациону везу од 0,575 са нивоом значајности од 1% између питања П12 и П24, што значи да ставови испитаника по којима се слажу да је интеграција старог ИС са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS и тд.) једноставна уједно се и слажу са ставом да стари ИС, такав какав је (без измена) може се лако интегрисати са елементима Индустије 4.0 (Cloud Computing, Internet of Things, Big Data, и тд.). Такође треба истаћи да питање П24 припада стратешкој перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 4.

7.1.2.6. Анализа одговора на питање П13

Према резултатима дескриптивне статистике ставови испитаника на питање П13, имају аритметичку средину 3,52, стандардну девијацију 1,148 и коефицијент варијације 33%. Из поменутих података може се закључити да се испитаници слажу са ставом постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 3,68, док испитаници из земаља у региону имају аритметичку средину од 3,34. Из тога се закључује да испитаници из Србије се слажу са ставом постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин, у просеку.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 2,90, док ставови испитаника чији је ИС предузећа стар више од 10 година имају аритметичку средину од 4,09. Из тога се закључује да испитаници чији ИС предузећа је стар више од 10 година се слажу са ставом постоји проблем у размени

података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин, у просеку.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику Progress 4GL и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику Progress 4GL може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При чему ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику Progress 4GL имају аритметичку средину од 3,38, док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику Progress 4GL имају аритметичку средину од 4,18. Из тога се закључује да испитаници чији наслеђени ИС предузећа јесте написан у програмском језику Progress 4GL **се слажу са ставом постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При томе, ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају аритметичку средину 3,38 док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају аритметичку средину 4,18. Из тога се закључује да **ставови испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. се слажу са ставом постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин, у просеку.**

Корелациона анализа указује на јаку позитивну корелациону везу од 0,584 са нивоом значајности од 1% између питања П13 и П17, што значи да **ставови испитаника по којима се слажу да постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин, уједно се и слажу са ставом да је стари ИС потребно модернизовати. Такође треба истаћи да је питање П17 у вези са Главном хипотезом истраживања.**

Корелациона анализа указује на јаку негативну корелациону везу од -0,553 са нивоом значајности од 1% између питања П13 и П18, што значи да **ставови испитаника**

по којима се слажу да постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин, уједно се и не слажу са ставом да документација старог ИС постоји и да је ажурна. Такође, треба истаћи да питање П18 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.

Корелациона анализа указује на јаку негативну корелациону везу од -0,519 са нивоом значајности од 1% између питања П13 и П22, што значи да ставови испитаника по којима се слажу да постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин, уједно се и не слажу са ставом да постоји подршка приликом одржавања хардвера и софтвера од стране произвођача. Такође, треба истаћи да питање П22 припада оперативној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 2.

Корелациона анализа указује на јаку негативну корелациону везу од -0,547 са нивоом значајности од 1% између питања П13 и П24, што значи да ставови испитаника по којима се слажу да постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин, уједно се и не слажу са ставом да стари ИС, такав какав је (без измена) може се лако интегрисати са елементима Индустрије 4.0 (Cloud Computing, Internet of Things, Big Data и тд.). Такође треба истаћи да питање П24 припада стратешкој перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 4.

7.1.2.7. Оперативна перспектива - резултат

На основу анализе одговора на питања П14, П22, П10, П11, П12 и П13, креирана је табела 33., у којој су представљени резултати различитих статистичких показатеља у односу на ставове испитаника са којима они могу да се сложе, да имају подељено мишљење и да се не сложе. Такође је приказан и одговор који иде у прилог Посебној хипотези 2 дисертације.

Табела 33. резултати анализе одговора на питања који су у вези са Посебном хипотезом 2 истраживања

| | | П14 | П22 | П10 | П11 | П12 | П13 |
|----------|-------------------------------|-----|-----|-----|-----|-----|-----|
| | дескриптивна | П | П | П | С | Н | С |
| т-тест 1 | Србија | П | П | С | С | Н | С |
| | земље региона | П | П | П | С | Н | П |
| т-тест 2 | једнако или мање од 10 година | Н | С | П | С | Н | П |

| | | | | | | | |
|---|--|---|---|---|---|---|---|
| | више од 10 година | П | П | С | С | Н | С |
| т-тест 3 | систем није развијан у језику 4. генерац. | П | П | П | С | Н | П |
| | систем јесте развијан у језику 4. генерац. | П | Н | С | С | Н | С |
| т-тест 4 | систем није развијан у Rprogress платформи | | | | | | П |
| | систем јесте развијан у Rprogress платформи | | | | | | С |
| одговор који иде у прилог посебној хипотези 2. | | С | Н | С | С | Н | С |
| Легенда: | | | | | | | |
| <ul style="list-style-type: none"> • С - слажем се • П - подељено • Н - не слажем се | | | | | | | |

Прво ћемо посматрати ставове испитаника у вези питања П11, П12 и П13. Дескриптивна статистика је значајан показатељ, и резултати анализе на горе наведена питања показују да се одговори испитаника поклапају са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 2.

Поред тога, на основу резултата т-тестова уочава се да постоје специфична поклапања. У првом реду постоје поклапања одговора испитаника из Србије са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 2. Након тога, постоје поклапања одговора испитаника код којих је ИС старији од 10 година са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 2. Затим, постоје поклапања одговора испитаника код којих је наслеђени ИС развијен у програмском језику 4. генерације са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 2.

Друго ћемо посматрати ставове испитаника у вези питања П14, П22 и П10. Дескриптивна статистика је значајан показатељ, и резултати анализе на ова два питања указују да су ставови испитаника подељени, у односу на одговоре који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 2. На основу резултата т-тестова уочава се да постоје одређена специфична поклапања. У првом реду постоје поклапања одговора испитаника из Србије за питање П10 са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 2. Након тога, постоје поклапања одговора испитаника на питање П10 код којих је ИС старији од 10 година са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 2. Затим, постоје поклапања одговора испитаника на питања П22 и П10 која иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 2. Посматрањем корелационих веза може се уочити да питање П14 има корелационе везе са питањима: П13 и то средњу позитивну корелациону везу, П17 и то

средњу позитивну корелациону везу, а одговори испитаника на та два питања се поклапају са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 2. Посматрањем корелационих веза може се уочити да питање П22 има корелационе везе са питањима: П10 и то јаку негативну корелациону везу, П11 и то средњу негативну корелациону везу, П13 и то јаку позитивну корелациону везу, П17 и то средњу негативну корелациону везу, а одговори испитаника на та четири питања се поклапају са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 2. Посматрањем корелационих веза може се уочити да питање П10 има корелационе везе са питањима: П12 и то средњу позитивну корелациону везу, П13 и то јаку позитивну корелациону везу, П13 и то јаку позитивну корелациону везу, а одговори испитаника на та два питања се поклапају са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 2.

Треће и најважније, посматрамо одговоре испитаника на питање П25, које је уједно и најзначајније питање у упитнику у вези предмета истраживања. Дескриптивна статистика указује да постоји позитиван став испитаника у вези целокупног упитника, чиме се потврђује релевантност свих питања која су део упитника.

На основу свега наведеног, може се закључити да је први део ПОСЕБНЕ ХИПОТЕЗЕ 2: „Посматрањем оперативне перспективе наслеђеног система могуће је сагледати проблеме наслеђених система из улоге корисника“, доказан.

Други део, ПОСЕБНЕ ХИПОТЕЗЕ 2: „а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема.“, ће бити обрађен у наредном поглављу.

7.1.3. Организациона перспектива

7.1.3.1. Анализа одговора на питање П7

Према резултатима дескриптивне статистике ставови испитаника на питање П7, имају **аритметичку средину 3,95**, стандардну девијацију 1,156 и коефицијент варијације 29%. Из поменутих података може се **закључити да се испитаници слажу са ставом да је стари ИС користан тиме што и даље доприноси пословању (и даље се извршавају одређене пословне процедуре и процеси).**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују

у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 3,85, док испитаници из земаља у региону имају аритметичку средину од 4,07. Из тога се закључује да испитаници из Србије и испитаници земаља у региону **се слажу са ставом да је стари ИС користан тиме што и даље доприноси пословању (и даље се извршавају одређене пословне процедуре и процеси), у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови испитаника чији ИС предузећа је стар мање од 10 година не разликују у просеку од ставова испитаника чији је ИС предузећа стар више од 10 година. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 3,85, док испитаници чији је ИС предузећа стар више од 10 година имају аритметичку средину од 4,07. Из тога се закључује да испитаници чији ИС предузећа је стар мање од 10 година и испитаници чији је ИС предузећа стар више од 10 година **се слажу са ставом да је стари ИС користан тиме што и даље доприноси пословању (и даље се извршавају одређене пословне процедуре и процеси), у просеку.**

Корелациона анализа указује на средњу негативну корелациону везу од -0,327 са нивоом значајности од 1% између питања П7 и П10, што значи да ставови испитаника **по којима се слажу да стари ИС је користан, тиме што и даље доприноси пословању (и даље се извршавају одређене пословне процедуре и процеси), уједно се и не слажу са ставом да стари ИС има апликације (развијене за неку организациону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама. Такође треба истаћи да питање П10 припада оперативној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 2.**

Корелациона анализа указује на средњу негативну корелациону везу од -0,303 са нивоом значајности од 1% између питања П7 и П13, што значи да ставови испитаника **по којима се слажу да стари ИС је користан, тиме што и даље доприноси пословању (и даље се извршавају одређене пословне процедуре и процеси), уједно се и не слажу да постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полуаутоматизовани начин. Такође, треба истаћи да питање П13 припада и оперативној и организационој перспективи, према**

предложеном теоријском оквиру, а тиме је и у вези са посебном хипотезом 2 и посебном хипотезом 3.

Корелациона анализа указује на средњу негативну корелациону везу од -0,352 са нивоом значајности од 1% између питања П7 и П16, што значи да **ставови испитаника по којима се слажу да стари ИС је користан, тиме што и даље доприноси пословању (и даље се извршавају одређене пословне процедуре и процеси), уједно се и не слажу да је стари ИС развијен од стране кадрова и средстава самог предузећа (in-house). Такође треба истаћи да питање П17 је у вези са развојном перспективом према предложеном теоријском оквиру, а према томе и са посебном хипотезом 1.**

7.1.3.2. Анализа одговора на питање П8

Према резултатима дескриптивне статистике ставови испитаника на питање П7, имају **аритметичку средину 2,59**, стандардну девијацију 1,227 и коефицијент варијације 47%. %. Из поменутих података може се **закључити да се испитаници не слажу са ставом да се у старом ИС релативно једноставно врше измене, у погледу измене пословних процедуре или пословних процеса, а у складу са захтевима.** Такође може се уочити већа варијација у одговорима испитаника.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 2,24, док испитаници из земаља у региону имају аритметичку средину од 3,00. Из тога се закључује да испитаници из Србије **се не слажу са ставом да се у старом ИС релативно једноставно врше измене, у погледу измене пословних процедуре или пословних процеса, а у складу са захтевима, у просеку, док у земљама из региона ставови су подељени по овом питању, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови испитаника чији ИС предузећа је стар мање од 10 година не разликују у просеку од ставова испитаника чији је ИС предузећа стар више од 10 година. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 2,87, док испитаници чији је ИС предузећа стар више од 10 година имају аритметичку средину од 2,33. Из тога се закључује да испитаници чији ИС

предузећа је стар више од 10 година се не слажу са ставом да се у старом ИС релативно једноставно врше измене, у погледу измене пословних процедуре или пословних процеса, а у складу са захтевима, у просеку, док су испитаници чији је ИС стар мање од 10 година подељени по овом питању, у просеку.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају аритметичку средину 2,67 док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају аритметичку средину 2,18. Из тога се закључује да ставови испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације не слажу са да се у старом ИС релативно једноставно врше измене, у погледу измене пословних процедуре или пословних процеса, а у складу са захтевима, у просеку.

Корелациона анализа указује на јаку негативну корелациону везу од -0,532 са нивоом значајности од 1% између питања П8 и П10, што значи да се ставови испитаника који се слажу са ставом да се у старом ИС релативно једноставно врше измене, у погледу измене пословних процедуре или пословних процеса, а у складу са захтевима, уједно и не слажу са ставом да стари ИС има апликације (развијене за неку организациону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама. треба истаћи да питање П10 припада оперативној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 2.

Корелациона анализа указује на јаку позитивну корелациону везу од 0,585 са нивоом значајности од 1% између питања П8 и П12, што значи да се ставови испитаника који се слажу са ставом да се у старом ИС релативно једноставно врше измене, у погледу измене пословних процедуре или пословних процеса, а у складу са захтевима, уједно се и слажу са ставом да је интеграција старог ИС са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS и тд.) једноставна. Такође треба истаћи да питање П12 припада оперативној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 2.

Корелациона анализа указује на средње негативну корелациону везу од -0,381 са нивоом значајности од 1% између питања П8 и П17, што значи да се ставови испитаника који се слажу са ставом да се у старом ИС релативно једноставно врше измене, у погледу измене пословних процедуре или пословних процеса, а у складу са захтевима, уједно и не слажу са ставом да је стари ИС потребно модернизовати. Такође треба истаћи да је питање П17 у вези са Главном хипотезом истраживања

Корелациона анализа указује на средње позитивну корелациону везу од 0,585 са нивоом значајности од 1% између питања П8 и П24, што значи да се ставови испитаника који се слажу са ставом да се у старом ИС релативно једноставно врше измене, у погледу измене пословних процедуре или пословних процеса, а у складу са захтевима, уједно се и слажу да стари ИС, такав какав је (без измена) може се лако интегрисати са елементима индустрије 4.0 (Cloud Computing, Internet of Things, Big Data, и тд.). Такође треба истаћи да питање П24 припада стратешкој перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 4.

7.1.3.3. Анализа одговора на питање П23

Према резултатима дескриптивне статистике ставови испитаника на питање П23, имају аритметичку средину 4,25, стандардну девијацију 0,803 и коефицијент варијације 19%. Из поменутих података може се закључити да се испитаници слажу са ставом да модернизацијом старог ИС могу се створити бољи услови за подршку одлучивању него супротно.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 4,21, док испитаници из земаља у региону имају аритметичку средину од 4,31. Из тога се закључује да испитаници из Србије и испитаници из земаља у региону слажу са ставом да модернизацијом старог ИС могу се створити бољи услови за подршку одлучивању него супротно, у просеку.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови испитаника чији ИС предузећа је стар мање од 10 година не разликују у просеку од ставова испитаника чији је ИС предузећа стар више од 10

година. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 4,17, док испитаници чији је ИС предузећа стар више од 10 година имају аритметичку средину од 4,33. Из тога се закључује да испитаници чији ИС предузећа је стар мање од 10 година и испитаници чији је ИС предузећа стар више од 10 година се слажу са ставом да модернизацијом старог ИС могу се створити бољи услови за подршку одлучивању него супротно, у просеку.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају аритметичку средину 4,17 док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају аритметичку средину 4,64. Из тога се закључује да ставови испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације и ставови испитаника чији ИС није написан у програмском језику 4. генерације се слажу са ставом да модернизацијом старог ИС могу се створити бољи услови за подршку одлучивању него супротно, у просеку.

Корелациона анализа указује на средње негативну корелациону везу од -0,412 са нивоом значајности од 1% између питања П23 и П24, што значи да се ставови испитаника који се слажу са ставом да модернизацијом старог ИС могу се створити бољи услови за подршку одлучивању него супротно, уједно се и не слажу са ставом да се стари ИС, такав какав је (без измена) може лако интегрисати са елементима Индустрије 4.0 (Cloud Computing, Internet of Things, Big Data, и тд.). Такође треба истаћи да питање П24 припада стратешкој перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 4.

Корелациона анализа указује на јаку позитивну корелациону везу од 0,517 са нивоом значајности од 1% између питања П23 и П25, што значи да се ставови испитаника који се слажу са ставом да модернизацијом старог ИС могу се створити бољи услови за подршку одлучивању него супротно, уједно и налазе да овакав упитник може боље усмерити модернизацију система, уместо посматрања само из једног угла (нпр. технолошког). Такође треба истаћи да је питање П25 подржава Главну хипотезу истраживања.

7.1.3.4. Организациона перспектива - резултат

На основу анализе одговора на питања П7, П8, П23 и П13, креирана је табела 34., у којој су представљени резултати различитих статистичких показатеља у односу на ставове испитаника са којима они могу да се сложе, да имају подељено мишљење и да се не сложе. Такође је приказан и одговор који иде у прилог Посебној хипотези 3 дисертације.

Табела 34. резултати анализе одговора на питања који су у вези са Посебном хипотезом 3 истраживања

| | | П7 | П8 | П23 | П13 |
|---|--|----|----|-----|-----|
| | дескриптивна | С | Н | С | С |
| т-тест 1 | Србија | С | Н | С | С |
| | земље региона | С | П | С | П |
| т-тест 2 | једнако или мање од 10 година | С | П | С | П |
| | више од 10 година | С | Н | С | С |
| т-тест 3 | систем није развијан у језику 4. генерац. | С | П | С | П |
| | систем јесте развијан у језику 4. генерац. | С | Н | С | С |
| т-тест 4 | систем није развијан у Progress платформи | | | | П |
| | систем јесте развијан у Progress платформи | | | | С |
| одговор који иде у прилог посебној хипотези 3. | | С | Н | С | С |
| Легенда: <ul style="list-style-type: none"> • С - слажем се • П - подељено • Н - не слажем се | | | | | |

Прво ћемо посматрати ставове испитаника у вези питања П7, П8, П23 и П13. Дескриптивна статистика је значајан показатељ, и резултати анализе на горе наведена питања показују да се одговори испитаника поклапају са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 3. Поред тога, на основу резултата т-тестова уочава се да постоје специфична поклапања. У првом реду постоје поклапања одговора испитаника из Србије са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 3. Након тога, постоје поклапања одговора испитаника код којих је ИС старији од 10 година са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 3. Затим, постоје поклапања одговора испитаника код којих је наслеђени ИС развијен у програмском језику 4. генерације са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 3.

Друго и најважније, посматрамо одговоре испитаника на питање П25, које је уједно и најзначајније питање у упитнику у вези предмета истраживања. Дескриптивна статистика указује да постоји позитиван став испитаника у вези целокупног упитника, чиме се потврђује релевантност свих питања која су део упитника.

На основу свега наведеног, може се закључити да је први део ПОСЕБНЕ ХИПОТЕЗЕ 3: „Посматрањем организационе перспективе наслеђеног система могуће је сагледати проблеме наслеђених система из улоге менаџера“, доказан.

Други део, ПОСЕБНЕ ХИПОТЕЗЕ 3: „а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема.“, ће бити обрађен у наредном поглављу.

7.1.4. Стратешка перспектива

7.1.4.1. Анализа одговора на питање П24

Према резултатима дескриптивне статистике ставови испитаника на питање П24, имају **аритметичку средину 1,95**, стандардну девијацију 1,069 и коефицијент варијације 55%. Из поменутих података **може се закључити да се испитаници не слажу са ставом да стари ИС, такав какав је (без измена) може се лако интегрисати са елементима Индустрије 4.0 (Cloud Computing, Internet of Things, Big Data, и тд.)**. Такође може се уочити већа варијација у одговорима испитаника.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 1,76, док испитаници из земаља у региону имају аритметичку средину од 2,17. Из тога се закључује да испитаници из Србије и испитаници из земаља у региону **не слажу са ставом да стари ИС, такав какав је (без измена) може се лако интегрисати са елементима Индустрије 4.0 (Cloud Computing, Internet of Things, Big Data, и тд.)**, у просеку.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови испитаника чији ИС предузећа је стар мање од 10 година

разликују у просеку од ставова испитаника чији је ИС предузећа стар више од 10 година. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 2,33, док испитаници чији је ИС предузећа стар више од 10 година имају аритметичку средину од 1,61. Из тога се закључује да испитаници чији ИС предузећа је стар мање од 10 година и испитаници чији је ИС предузећа стар више од 10 година се не слажу са ставом да стари ИС, такав какав је (без измена) може се лако интегрисати са елементима Индустије 4.0 (Cloud Computing, Internet of Things, Big Data, и тд.), у просеку. При томе треба имати у виду да је неслагање групе испитаника чији је ИС предузећа стар више од 10 година веће у односу на испитанике чији ИС предузећа је стар мање од 10 година.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају аритметичку средину 2,04 док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају аритметичку средину 1,55. Из тога се закључује да испитаници чији наслеђени ИС предузећа није написан у програмском језику 4. генерације, као и испитаници чији наслеђени систем јесте написан у програмском језику 4. генерације се не слажу са ставом да стари ИС, такав какав је (без измена) може се лако интегрисати са елементима Индустије 4.0 (Cloud Computing, Internet of Things, Big Data, и тд.), у просеку. При томе треба имати у виду да је неслагање групе испитаника наслеђени систем јесте написан у програмском језику 4. генерације веће у односу на другу групу испитаника.

7.1.4.2. Стратешка перспектива - резултат

На основу анализе одговора на питања П24 и П8, креирана је табела 35., у којој су представљени резултати различитих статистичких показатеља у односу на ставове испитаника са којима они могу да се сложе, да имају подељено мишљење и да се не сложе. Такође је приказан и одговор који иде у прилог Посебној хипотези 4 дисертације.

Табела 35. резултати анализе одговора на питања који су у вези са Посебном хипотезом 4 истраживања

| | П24 | П8 |
|--------------|-----|----|
| дескриптивна | Н | Н |

| | | | |
|---|---|---|---|
| т-тест 1 | Србија | Н | Н |
| | земље региона | Н | П |
| т-тест 2 | једнако или мање од 10 година | Н | П |
| | више од 10 година | Н | Н |
| т-тест 3 | систем није развијан у језику 4. генерац. | Н | П |
| | систем јесте развијан у језику 4. генерац. | Н | Н |
| т-тест 4 | систем није развијан у Progress платформи | | |
| | систем јесте развијан у Progress платформи | | |
| одговор који иде у прилог посебној хипотези 4. | | Н | Н |
| Легенда: | | | |
| <ul style="list-style-type: none"> • С - слажем се • П - подељено • Н - не слажем се | | | |

Прво ћемо посматрати ставове испитаника у вези питања П24 и П8. Дескриптивна статистика је значајан показатељ, и резултати анализе на горе наведена питања показују да се одговори испитаника поклапају са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 4. Поред тога, на основу резултата т-тестова уочава се да постоје специфична поклапања. У првом реду постоје поклапања одговора испитаника из Србије са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 4. Након тога, постоје поклапања одговора испитаника код којих је ИС старији од 10 година са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 4. Затим, постоје поклапања одговора испитаника код којих је наслеђени ИС развијен у програмском језику 4. генерације са одговорима који иду у прилог ПОСЕБНОЈ ХИПОТЕЗИ 4.

Друго и најважније, посматрамо одговоре испитаника на питање П25, које је уједно и најзначајније питање у упитнику у вези предмета истраживања. Дескриптивна статистика указује да постоји позитиван став испитаника у вези целокупног упитника, чиме се потврђује релевантност свих питања која су део упитника.

На основу свега наведеног, може се закључити да је први део ПОСЕБНЕ ХИПОТЕЗЕ 4: „Посматрањем стратешке перспективе наслеђеног система могуће је сагледати проблеме наслеђених система из улоге топ менаџмента“, доказан.

Други део, ПОСЕБНЕ ХИПОТЕЗЕ 4: „а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема.“, ће бити обрађен у наредном поглављу.

7.1.5. Главна хипотеза

7.1.5.1. Анализа одговора на питање ПЗ

Према резултатима дескриптивне статистике ставови испитаника на питање ПЗ, имају аритметичку средину 3,16, стандардну девијацију 1,273 и коефицијент варијације 40%. Из поменутих података **не може се закључити да ли испитаници перципирају ИС свог предузећа као наслеђен**, а може се видети да постоји већа варијација у њиховим одговорима.

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 3,56, док испитаници из земаља у региону имају аритметичку средину од 2,69. Из тога се закључује да **испитаници из Србије имају ставове који се слажу да је ИС предузећа у којем раде наслеђен, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При томе, ставови испитаника код којих је ИС стар мање од 10 година имају аритметичку средину 2,5 док испитаници код којих је ИС њиховог предузећа старији од 10 година имају аритметичку средину 3,76. Из тога се закључује да **испитаници чије предузеће има ИС стар мање од 10 година имају ставове по којима се не слажу да је ИС предузећа у којем раде наслеђен, у просеку, а да испитаници чије предузеће има ИС стар више од 10 година имају ставове по којима се слажу да је ИС предузећа у којем раде наслеђен, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку разликују. При томе, ставови

испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају аритметичку средину 3,00 док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају аритметичку средину 3,91. Из тога се закључује да **испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације се слажу са ставом да њихово предузеће има стари (застарели) информациони систем (ИС) или софтвер, док испитаници чији ИС предузећа није написан у програмском језику 4. генерације имају подељено мишљење, у просеку.**

Корелациона анализа указује на јаку позитивну корелациону везу од 0,668 са нивоом значајности од 1% између питања П3 и П6, што значи да **ставови испитаника који виде ИС предузећа као наслеђен уједно и сматрају да је нарушена иницијално постављена архитектура, у просеку. Такође, треба истаћи да питање П6 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.**

Корелациона анализа указује на јаку позитивну корелациону везу од 0,531 са нивоом значајности од 1% између питања П3 и П13, што значи да **ставови испитаника који виде ИС предузећа као наслеђен уједно и сматрају да наслеђен систем има проблема у размени података са неким другим предузећем на аутоматизован или полуаутоматизован начин. Такође, треба истаћи да питање П13 припада и оперативној и организационој перспективи, према предложеном теоријском оквиру, а тиме је и у вези са посебном хипотезом 2 и посебном хипотезом 3.**

Корелациона анализа указује на средњу позитивну корелациону везу од 0,321 са нивоом значајности од 5% између питања П3 и П25, што значи да **ставови испитаника који виде ИС предузећа као наслеђен уједно и сматрају да овакав упитник може боље усмерити модернизацију система уместо посматрање само из једног угла. Такође треба истаћи да је питање П25 подржава Главну хипотезу истраживања.**

7.1.5.2. Анализа одговора на питање П17

Према резултатима дескриптивне статистике ставови испитаника на питање П17, имају **аритметичку средину 4,08**, стандардну девијацију 1,168 и коефицијент варијације 29%. Може се закључити да **се испитаници слажу са ставом да је стари ИС потребно модернизовати.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се

закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од 4,26, док испитаници из земаља у региону имају аритметичку средину од 3,86. Из тога се закључује да испитаници из Србије имају исте ставове као и испитаници земаља у региону, а то је **да се слажу са ставом да је стари ИС потребно модернизовати, у просеку. При томе, став испитаника из Србије је израженији од става испитаника из земаља у региону.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови испитаника чији ИС предузећа је стар мање од 10 година разликују у просеку од ставова испитаника чији је ИС предузећа стар више од 10 година. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 3,53, док испитаници чији је ИС предузећа стар више од 10 година имају аритметичку средину од 4,58. Из тога се **закључује да испитаници чији ИС предузећа је стар мање од 10 година се слажу са ставом да је стари ИС потребно модернизовати, као и испитаници чији је ИС предузећа стар више од 10 година, при чему је став испитаника чији је ИС предузећа стар више од 10 година, израженији, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају аритметичку средину 4,00 док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају аритметичку средину 4,45. Из тога се закључује да **испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације, као и испитаници чији наслеђени ИС предузећа није написан у програмском језику 4. генерације се слажу са ставом да њихово предузеће има стари (застарели) информациони систем (ИС) или софтвер, у просеку.**

Корелациона анализа указује на јаку позитивну корелациону везу од 0,586 са нивоом значајности од 1% између питања П17 и П20, што значи **да се ставови испитаника који се слажу са ставом да је стари ИС потребно модернизовати, уједно**

и слажу да је редуанса у систему изражена. Такође треба истаћи да питање П20 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.

Корелациона анализа указује на средњу негативну корелациону везу од $-0,300$ са нивоом значајности од 1% између питања П17 и П21, што значи да **се ставови испитаника који се слажу са ставом да је стари ИС потребно модернизовати, уједно и не слажу са ставом да новозапослени имају знање и вештине да у старту раде са старим ИС. Такође, треба истаћи да питање П21 припада развојној перспективи према предложеном теоријском оквиру и у вези је са посебном хипотезом 1.**

Корелациона анализа указује на средњу негативну корелациону везу од $-0,369$ са нивоом значајности од 1% између питања П17 и П22, што значи да **се ставови испитаника који се слажу са ставом да је стари ИС потребно модернизовати, уједно и не слажу са ставом да постоји подршка приликом одржавања хардвера и софтвера од стране произвођача. Такође треба истаћи да питање П22 је у вези је са посебном хипотезом 1.**

Корелациона анализа указује на средњу позитивну корелациону везу од $0,310$ са нивоом значајности од 1% између питања П17 и П25, што значи да **се ставови испитаника који се слажу са ставом да је стари ИС потребно модернизовати, уједно и налазе да овакав упитник може боље усмерити модернизацију система, уместо посматрања само из једног угла (нпр. технолошког). Такође треба истаћи да је питање П25 подржава Главну хипотезу истраживања.**

7.1.5.3. Анализа одговора на питање П25

Према резултатима дескриптивне статистике ставови испитаника на питање П25, имају **аритметичку средину 4,21**, стандардну девијацију $0,626$ и коефицијент варијације 15%. Из поменутих података **може се закључити да се испитаници слажу са ставом да овакав упитник може боље усмерити модернизацију система, уместо посматрања само из једног угла (нпр. технолошког).**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника из Србије и групе испитаника из земаља региона може се закључити са нивоом значајности од 5% да се ставови испитаника из Србије не разликују у просеку од ставова испитаника из земаља у региону. При чему ставови испитаника из Србије имају аритметичку средину од $4,21$, док испитаници из земаља у региону имају аритметичку средину од $4,21$. Из тога се закључује да испитаници из Србије имају исте

ставове као и испитаници земаља у региону, а то је да се **слажу са ставом да овакав упитник може боље усмерити модернизацију система, уместо посматрања само из једног угла (нпр. технолошког), у просеку и међу њима нема значајнијих разлика.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији ИС предузећа је стар мање од 10 година и групе испитаника чији је ИС предузећа стар више од 10 година може се закључити са нивоом значајности од 5% да се ставови испитаника чији ИС предузећа је стар мање од 10 година не разликују у просеку од ставова испитаника чији је ИС предузећа стар више од 10 година. При чему ставови испитаника чији ИС предузећа је стар мање од 10 година имају аритметичку средину од 4,00, док испитаници чији је ИС предузећа стар више од 10 година имају аритметичку средину од 4,39. Из тога се закључује да испитаници чији ИС предузећа је стар мање од 10 година и испитаници чији је ИС предузећа стар више од 10 година се **слажу са ставом да овакав упитник може боље усмерити модернизацију система, уместо посматрања само из једног угла (нпр. технолошког), у просеку. При томе треба имати увиду да испитаници код којих је ИС стар више од 10 година имају веће слагање по овом питању, у просеку.**

Када се посматра резултат т-теста у којем се испитују ставови 2 групе испитаника: групе испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације и испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације може се закључити са нивоом значајности од 5% да се ставови између ове две групе испитаника у просеку не разликују. При томе, ставови испитаника чији наслеђени ИС предузећа није написан у програмском језику 4. генерације имају аритметичку средину 4,17 док испитаници чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације имају аритметичку средину 4,36. Из тога се закључује да **ставови испитаника чији наслеђени ИС предузећа јесте написан у програмском језику 4. генерације и ставови испитаника чији ИС није написан у програмском језику 4. генерације се слажу са ставом да овакав упитник може боље усмерити модернизацију система, уместо посматрања само из једног угла (нпр. технолошког), у просеку и међу њима нема значајнијих разлика.**

7.1.5.4. Главна хипотеза - резултат

На основу анализе одговора на питања П3, П17 и П25, креирана је табела 36., у којој су представљени резултати различитих статистичких показатеља у односу на ставове испитаника са којима они могу да се сложе, да имају подељено мишљење и да се не сложе. Такође је приказан и одговор који иде у прилог Главној хипотези дисертације.

Табела 36. резултати анализе одговора на питања који су у вези са Главном хипотезом истраживања

| | | ПЗ | П17 | П25 |
|---|--|----|-----|-----|
| | дескриптивна | С | С | С |
| т-тест 1 | Србија | С | С | С |
| | земље региона | П | С | С |
| т-тест 2 | једнако или мање од 10 година | Н | С | С |
| | више од 10 година | С | С | С |
| т-тест 3 | систем није развијан у језику 4. генерац. | П | С | С |
| | систем јесте развијан у језику 4. генерац. | С | С | С |
| т-тест 4 | систем није развијан у Progress платформи | | | |
| | систем јесте развијан у Progress платформи | | | |
| одговор који иде у прилог посебној главној хипотези | | С | С | С |
| Легенда: <ul style="list-style-type: none"> • С - слажем се • П - подељено • Н - не слажем се | | | | |

Прво ћемо посматрати ставове испитаника у вези питања ПЗ, П17 и П25. Дескриптивна статистика је значајан показатељ, и резултати анализе на горе наведена питања показују да се одговори испитаника поклапају са одговорима који иду у прилог ГЛАВНОЈ ХИПОТЕЗИ. Поред тога, на основу резултата т-тестова уочава се да постоје специфична поклапања. У првом реду постоје поклапања одговора испитаника из Србије са одговорима који иду у прилог ГЛАВНОЈ ХИПОТЕЗИ. Након тога, постоје поклапања одговора испитаника код којих је ИС старији од 10 година са одговорима који иду у прилог ГЛАВНОЈ ХИПОТЕЗИ. Затим, постоје поклапања одговора испитаника код којих је наслеђени ИС развијан у програмском језику 4. генерације са одговорима који иду у прилог ГЛАВНОЈ ХИПОТЕЗИ.

Друго, први део ПОСЕБНЕ ХИПОТЕЗЕ 1, је доказан, а она је саставни део ГЛАВНЕ ХИПОТЕЗЕ.

Треће, први део ПОСЕБНЕ ХИПОТЕЗЕ 2, је доказан, а она је саставни део ГЛАВНЕ ХИПОТЕЗЕ.

Четврто, први део ПОСЕБНЕ ХИПОТЕЗЕ 3, је доказан, а она је саставни део ГЛАВНЕ ХИПОТЕЗЕ.

Пето, први део ПОСЕБНЕ ХИПОТЕЗЕ 4, је доказан, а она је саставни део ГЛАВНЕ ХИПОТЕЗЕ.

На основи свега наведеног може се закључити да је први део ГЛАВНЕ ХИПОТЕЗЕ: „Посматрањем различитих перспектива (тачки гледишта) наслеђеног информационог система као и његове перспективе (поглед ка будућности) могуће је потпуније сагледати наслеђени систем“, доказан.

Други део, ГЛАВНЕ ХИПОТЕЗЕ: „и на основу тога креирати путоказ ка његовој модернизацији.“ ће бити обрађен у наредном поглављу.

На основу свега наведеног може се закључити да је предложени теоријски оквир дефинисан у 5. поглављу уз Студију случаја на моделу ЈКП „Београдски водовод и канализација“, као и Анализу ланца вредности на случају ЈКП „Београдски водовод и канализација“ евалуиран позитивно и да се може корисити за потпуно сагледавање наслеђеног информационог система.

8. Разматрање модернизације наслеђеног система

8.1. Испитивање употребљивости наслеђеног информационог система ЈКП „Београдски водовод и канализација“

Постоји велики број метода и техника за тестирање и проверу употребљивости као једне од важније карактеристике (атрибута) квалитета софтвера. За потребе истраживања одабрана је метода System Usability Scale, скраћено SUS. SUS метода тестирања употребљивости настала је још 80-их година у склопу тадашње познате корпорације Digital Equipment Corporation, скраћено DEC, а њен аутор је Џон Брук. Ова метода има за циљ да повери ставове о употребљивости корисника система и тиме припада методама испитивања за евалуацију употребљивости (UIUXtrend, n.d.).

SUS метода је веома популарна метода, коришћена у многим истраживањима, обухваћена је у око 1300 чланака и публикација, где се користи за анализу

употребљивости и према неким изворима она је постала индустријски стандард. Њене предности су (Usability.gov, n.d.):

- веома је једноставна за скалирање и администрирање
- може се користити и на малим величинама узорака са поузданим резултатима
- валидна је - може ефикасно направити разлику између употребљивих и неупотребљивих система

Приликом употребе SUS методе, учесници износе оцене на следећих десет ставова који су приказани заједно са својим редним бројевима:

1. Мислим да бих желео да овај систем користим често
2. Налазим да је систем непотребно сложен
3. Мислим да је систем једноставан за коришћење
4. Мислим да ће ми бити потребна подршка техничке особе да бих могао да користим систем
5. Налазим да су различите функције у овом систему добро интегрисане
6. Мислим да је много неконзистентности у систему
7. Замишљам да би већина људи научила да систем користи брзо
8. Налазим да је систем незгодан за коришћење
9. Осећао сам се веома поуздано приликом коришћења система
10. Потребно је било да научим много ствари пре него што бих могао да кренем са овим системом

У склопу ове методе користи се Ликертова скала са пет опција и сваки учесник може да одабере једну од пет оцена за сваки од ставова:

- у потпуности се не слажем
- не слажем се
- неутралан сам
- слажем се
- у потпуности се слажем

Свака од оцена се вреднује, тако да избор оцене „у потпуности се не слажем“ одговара 1 боду, избор оцене „не слажем се“ одговара 2 бода, избор оцене „неутралан сам“ одговара 3 бода, избор оцене „слажем се“ одговара 4 бода, док избор оцене „у потпуности се слажем“ одговара 5 бодова. На поменути начин дефинисане су вредности

статистичких обележја, према упутству за примену SUS методе за оцену ставова о употребљивости система.

Прерачунавање SUS резултата се врши на следећи начин:

- вредност променљиве X се добија тако што се саберу оцене испитаника на непарна питања (питања под бројевима 1, 3, 5, 7 и 9) и затим се од тог збира одузме 5
- вредност променљиве Y се добија тако што се саберу оцене испитаника на парна питања (питања под бројевима 2, 4, 6, 8, 10), а затим се од броја 25 одузме тај збир
- на крају се збир променљиве X и променљиве Y помножи са 2.5 и на тако се добије SUS резултат

Да би испитали употребљивост наслеђеног информационог системе, спроведена је анкета (упитник) међу 43 корисника наслеђеног система ЈКП БВК, где је свако од њих давао своју субјективну оцену на десет постављених питања (ставова) према SUS методи.

Корисници су оцене давали за 3 одабране апликације, које су означене са АП1, АП2 и АП3¹⁰. За апликацију АП1 прикупљени су одговори на питања од стране 20 корисника. За апликацију АП2 прикупљени су одговори на питања од стране 17 корисника, док на питања везаних за апликацију АП3 постоје одговори од стране 6 корисника.

SUS скор за апликацију АП1 износи **44.25** и добијен је прерачунавањем просечне вредности појединачних SUS скорова од стране 20 корисника система. Просечна вредност параметра X износи **9.95**, а параметра Y **7.75**.

Поред тога, просечна вредност обележја за ставове који су корисници оцењивали по питањима за апликацију АП1 је дата у следећој табели:

Табела 37. просечне вредности одговора корисника на питања у вези употребљивости апликације АП1

| Рбр. | Питање | Просечна вредност |
|------|---|-------------------|
| 1. | Мислим да бих желео да овај систем користим често | 3.65 |
| 2. | Налазим да је систем непотребно сложен | 3.30 |
| 3. | Мислим да је систем једноставан за коришћење | 3.25 |

¹⁰ разлози чувања безбедности података, који се могу подвући под пословну тајну узроковали су да праве називе апликација у њиховом изворном облику не наведем, већ сам их означао на горе описан начин

| | | |
|-----|---|------|
| 4. | Мислим да ће ми бити потребна подршка техничке особе да бих могао да користим систем | 2.50 |
| 5. | Налазим да су различите функције у овом систему добро интегрисане | 1.90 |
| 6. | Мислим да је много неконзистентности у систему | 4.20 |
| 7. | Замишљавам да би већина људи научила да систем користи брзо | 3.55 |
| 8. | Налазим да је систем незгодан за коришћење | 3.00 |
| 9. | Осећао сам се веома поуздано приликом коришћења система | 2.60 |
| 10. | Потребно је било да научим много ствари пре него што бих могао да кренем са овим системом | 4.25 |

SUS скор за апликацију АП2 износи **46.77** и добијен је прерачунавањем просечне вредности појединачних SUS скорова од стране 17 корисника система.

Просечна вредност параметра **X** износи **9.65**, а параметра **Y** **9.06**.

Поред тога, просечна вредност обележја за ставове који су корисници оцењивали по питањима за апликацију АП2 је дата у следећој табели:

Табела 38. просечне вредности одговора корисника на питања у вези употребљивости апликације АП2

| Рбр. | Питање | Просечна вредност |
|------|---|-------------------|
| 1. | Мислим да бих желео да овај систем користим често | 3.59 |
| 2. | Налазим да је систем непотребно сложен | 2.59 |
| 3. | Мислим да је систем једноставан за коришћење | 3.35 |
| 4. | Мислим да ће ми бити потребна подршка техничке особе да бих могао да користим систем | 2.59 |
| 5. | Налазим да су различите функције у овом систему добро интегрисане | 1.82 |
| 6. | Мислим да је много неконзистентности у систему | 4.29 |
| 7. | Замишљавам да би већина људи научила да систем користи брзо | 3.41 |
| 8. | Налазим да је систем незгодан за коришћење | 2.65 |
| 9. | Осећао сам се веома поуздано приликом коришћења система | 2.47 |
| 10. | Потребно је било да научим много ствари пре него што бих могао да кренем са овим системом | 3.82 |

SUS скор за апликацију АП3 износи **44.15** и добијен је прерачунавањем просечне вредности појединачних SUS скорова од стране 6 корисника система. Просечна вредност параметра **X** износи **8.83**, а параметра **Y** **8.83**.

Поред тога, просечна вредност обележја за ставове који су корисници оцењивали по питањима за апликацију АП3 је дата у следећој табели:

Табела 39. просечне вредности одговора корисника на питања у вези употребљивости апликације АПЗ

| Рбр. | Питање | Просечна вредност |
|------|---|-------------------|
| 1. | Мислим да бих желео да овај систем користим често | 3.17 |
| 2. | Налазим да је систем непотребно сложен | 2.00 |
| 3. | Мислим да је систем једноставан за коришћење | 2.83 |
| 4. | Мислим да ће ми бити потребна подршка техничке особе да бих могао да користим систем | 2.50 |
| 5. | Налазим да су различите функције у овом систему добро интегрисане | 1.00 |
| 6. | Мислим да је много неконзистентности у систему | 4.33 |
| 7. | Замишљам да би већина људи научила да систем користи брзо | 4.00 |
| 8. | Налазим да је систем незгодан за коришћење | 3.33 |
| 9. | Осећао сам се веома поуздано приликом коришћења система | 2.83 |
| 10. | Потребно је било да научим много ствари пре него што бих могао да кренем са овим системом | 4.00 |

8.2. Тумачење резултата

8.2.1. Тумачење резултата за апликацију АП1

На основу вредности SUS скора за апликацију АП1 наслеђеног система ЈКП БВК од **44.25**, закључује се да систем (апликација АП1) није употребљив, тј. да је његова употребљивост на ниском нивоу, односно испод нормалне вредности. Да би употребљивост система била нормална, потребно је да се оствари SUS скор од 68.

Овакав резултат упућује да је потребно предузети такве кораке да се решили значајни проблеми употребљивости са којима се суочавају крајњи корисници. Корисници апликације АП1 не перципирају апликацију као употребљиву, јер имају напор приликом употребе исте. Због тога они не могу да ефективно и ефикасно обављају своје радне задатке.

Оваква вредност SUS скора посредно указује да је квалитет софтвера у апликацији АП1 наслеђеног информационог система на ниском нивоу, што даље указује на низ проблема који наслеђени систем има, а које корисници перципирају.

Просечна вредност одговора испитаника на прво питање од **3.65** указује да корисници желе да овај систем користе често. То може да значи да **проналазе вредност у овој апликацији и да им она помаже да изврше своје радне задатке**. Извршавањем радних задатака корисници спроводе одређене пословне активности предузећа, путем којих се остварују циљеви предузећа, као његова визија и мисија.

Просечна вредност одговора испитаника на друго питање од **3.30** указује да корисници **имају подељене ставове о томе да ли је систем непотребно сложен**. Један део корисника апликације АП1 сматра да постоји сложеност, док други део корисника има потпуно опречне ставове.

Просечна вредност одговора испитаника на треће питање од **3.25** описује да је корисници **имају подељена мишљења о томе да ли је систем једноставан за коришћење**.

Просечна вредност одговора испитаника на четврто питање од **2.50** указује да су корисници става да не сматрају да ће им бити потребна подршка техничке особе да би могли да користе систем, тј. апликацију АП1. Дакле, они перципирају да **самостално могу да обављају задатке**.

Одговори корисника на пето питање у просеку износе од **1.90**, што указује да корисници **перципирају наслеђен систем (апликацију АП1) као систем који добро не интегрише одређене функционалности, за које са они сматрају да су релевантне приликом обављања посла**. На тај начин они **посредно указују на проблеме интеграције између апликација АП1 и осталих делова система**, што се може искорисити приликом одабира стратегије модернизације.

Испитаници су на шесто питање давали одговоре и оформили просечну оцену од **4.20** чија вредност указује да **корисници перципирају одређену неконзистентност у систему (апликацији АП1)**. Неконзистентност може бити **последница редувансе у систему** али и резултат вишеструких ефеката таласа који указују на **неадекватну архитектуру система (апликације АП1)**.

Просечна вредност одговора испитаника на седмо питање од **3.55**, указује да корисници апликације АП1 **сматрају да би већина људи научила да брзо користи систем (апликацију АП1)**. Дакле, апликација је тако осмишљена да корисници могу да је лако савладају.

Одговори корисника на осмо питање у просеку износе **3.00**, што указује да постоје **подељена мишљења међу корисницима у вези тога да ли је систем незгодан за коришћење**. Неки корисници наилазе на проблеме у употреби, док други не налазе на неке веће проблеме приликом интеракције са апликацијом АП1 које их спречавају да је користе, сем оних које су исказали.

Просечна вредност одговора испитаника на девето питање од **2.60**, описује да корисници **имају подељене ставове о томе да се осећају поуздано приликом коришћења система** (апликације АП1). Код једног дела корисника апликација АП1 се понаша на начин да одговара њиховим очекивањима, па они налазе да је употреба поуздана, док други део корисника има опречне ставове и према укупном резултату ови ставови су наглашенији.

Испитаници су на десето питање давали одговоре и оформили просечну оцену од **4.25**, што указује да они **сматрају да је потребно да науче много ствари пре него што би кренули да раде са системом** (апликацијом АП1). Лакоћа приликом учења како се нека апликација користи је једна од најважнијих чинилаца употребљивости, према многим ауторима. Због тога, је потребно посебно обратити пажњу на ову оцену корисника, и предузети мере како би се систем (апликација) учинила употребљивијом.

8.2.2. Тумачење резултата за апликацију АП2

На основу вредности SUS скорa за апликацију АП2 наслеђеног система ЈКП БВК од **46.77**, закључује се да систем није употребљив, тј. да је његова употребљивост на ниском нивоу, односно испод нормалне вредности. Да би употребљивост система била нормална, потребно је да се оствари SUS скор од 68.

Овакав резултат упућује да је потребно предузети такве кораке да се решили значајни проблеми употребљивости са којима се суочавају крајњи корисници. Корисници апликације АП2 не перципирају апликацију као употребљиву, јер имају напор приликом употребе исте. Због тога они не могу да ефективно и ефикасно обављају своје радне задатке.

Оваква вредност SUS скорa посредно указује да је квалитет софтвера у апликацији АП2 наслеђеног информационог система на ниском нивоу, што даље указује на низ проблема који наслеђени систем има, а које корисници перципирају.

Просечна вредност одговора испитаника на прво питање од **3.59** указује да корисници желе да овај систем користе често. То може да значи да **проналазе вредност у овој апликацији и да им она помаже да изврше своје радне задатке**. Извршавањем радних задатака корисници спроводе одређене пословне активности предузећа, путем којих се остварују циљеви предузећа, као његова визија и мисија.

Просечна вредност одговора испитаника на друго питање од **2.59** указује да **корисници сматрају да систем није непотребно сложен**. Дакле, они виде да је систем (апликација АП2) једноставна.

Просечна вредност одговора испитаника на треће питање од **3.35** описује да је корисници **имају подељена мишљења о томе да ли је систем једноставан за коришћење**.

Просечна вредност одговора испитаника на четврто питање од **2.59** указује да су корисници имају подељене ставове о томе да ли ће им бити потребна подршка техничке особе да би могли да користе апликацију, при чему су ставови корисника који то не сматрају израженији. Дакле, они перципирају да **самостално могу да обављају задатке**.

Одговори корисника на пето питање у просеку износе од **1.82**, што указује да корисници **перципирају наслеђен систем (апликацију АП2) као систем који добро не интегрише одређене функционалности, за које са они сматрају да су релевантне приликом обављања посла**. На тај начин они **посредно указују на проблеме интеграције између апликација АП2 и осталих делова система**, што се може искорисити приликом одабира стратегије модернизације.

Испитаници су на шесто питање давали одговоре и оформили просечну оцену од **4.29** чија вредност указује да **корисници перципирају одређену неконзистентност у систему (апликацији АП2)**. Неконзистентност може бити **последница редувансе у систему** али и резултат вишеструких ефеката таласа који указују на **неадекватну архитектуру система (апликације АП2)**.

Просечна вредност одговора испитаника на седмо питање од **3.41**, указује да корисници апликације АП2 **имају подељене ставове о томе да би већина људи научила да брзо користи систем (апликацију АП2)**.

Одговори корисника на осмо питање у просеку износе **2.65**, што указује да **корисници имају подељене ставове о томе да је систем незгодан за коришћење**. Један део њих исказује да је систем незгодан за коришћење, али други део њих указује да они немају тај проблем и њихови ставови су израженији.

Просечна вредност одговора испитаника на девето питање од **2.47**, описује да корисници **имају став да се осећају поуздано приликом коришћења система (апликације АП2)**. Они сматрају да се апликација АП2 понаша на начин да одговара њиховим очекивањима.

Испитаници су на десето питање давали одговоре и оформили просечну оцену од **3.82**, што указује да они **сматрају да је потребно да науче много ствари пре него што би кренули да раде са системом** (апликацијом АП2). Лакоћа приликом учења како се нека апликација користи је једна од најважнијих чинилаца употребљивости, према многим ауторима. Због тога, је потребно посебно обратити пажњу на ову оцену корисника, и предузети мере како би се систем (апликација) учинила употребљивијом.

8.2.3. Тумачење резултата за апликацију АП3

На основу вредности SUS скора за апликацију АП3 наслеђеног система ЈКП БВК од **44.15**, закључује се да систем (апликација АП3) није употребљив, тј. да је његова употребљивост на ниском нивоу, односно испод нормалне вредности. Да би употребљивост система била нормална, потребно је да се оствари SUS скор од 68.

Овакав резултат упућује да је потребно предузети такве кораке да се решили значајни проблеми употребљивости са којима се суочавају крајњи корисници. Корисници апликације АП3 не перципирају апликацију као употребљиву, јер имају напор приликом употребе исте. Због тога они не могу да ефективно и ефикасно обављају своје радне задатке.

Оваква вредност SUS скора посредно указује да је квалитет софтвера у апликацији АП3 наслеђеног информационог система на ниском нивоу, што даље указује на низ проблема који наслеђени систем има, а које корисници перципирају.

Просечна вредност одговора испитаника на прво питање од **3.17** указује да корисници имају **подељене ставове о томе да желе да овај систем (апликацију АП3) користе често**. Један део њих жели да користи често апликацију, док други има другачији став. Дакле, неки од њих проналазе вредност у овој апликацији која им помаже да изврше своје радне задатке, а други не. Извршавањем радних задатака корисници спроводе одређене пословне активности предузећа, путем којих се остварују циљеви предузећа, као његова визија и мисија.

Просечна вредност одговора испитаника на друго питање од **2.00** указује да корисници **не сматрају да је систем (апликација АП3) непотребно сложен**.

Просечна вредност одговора испитаника на треће питање од **2.83** описује да је корисници **имају подељена мишљења око тога да је систем (апликација АП3) једноставан за коришћење**.

Просечна вредност одговора испитаника на четврто питање од **2.50** указује да су корисници става да не сматрају да ће има бити потребна подршка техничке особе да би могли да користе систем, тј. апликацију АПЗ. Дакле, они перципирају да **самостално могу да обављају задатке**.

Одговори корисника на пето питање у просеку износе од **1.00**, што указује да корисници **перципирају наслеђен систем (апликацију АПЗ) као систем који добро не интегрише одређене функционалности, за које са они сматрају да су релевантне приликом обављања посла**. На тај начин они **посредно указују на изражене проблеме интеграције између апликације АПЗ и осталих делова система**, што се може искорисити приликом одабира стратегије модернизације.

Испитаници су на шесто питање давали одговоре и оформили просечну оцену од **4.33** чија вредност указује да **корисници перципирају одређену неконзистентност у систему (апликацији АПЗ)**. Неконзистентност је веома изражена и она може бити **последица редувансе у систему** али и резултат вишеструких ефеката таласа који указују на **неадекватну архитектуру система (апликације АПЗ)**.

Просечна вредност одговора испитаника на седмо питање од **4.00**, указује да корисници апликације АПЗ **сматрају да би већина људи научила да брзо користи систем (апликацију АПЗ)**. Дакле, апликација је тако осмишљена да корисници могу да је лако савладају.

Одговори корисника на осмо питање у просеку износе **3.33**, што указује да корисници **имају подељене ставове о томе да ли је систем незгодан за коришћење**. Један део њих има ставове да систем јесте незгодан за коришћење, док други део њих има другачији став и не наилазе на неке веће проблеме приликом интеракције са апликацијом АПЗ који их спречавају да је користе, сем оних које су већ исказали.

Просечна вредност одговора испитаника на девето питање од **2.83**, описује да корисници имају **подељене ставове око тога да је систем (апликација АПЗ) поуздан приликом коришћења**. Дакле, један део њих сматра да се апликација АПЗ понаша у складу са њиховим очекивањима, док други део корисника има опречне ставове.

Испитаници су на десето питање давали одговоре и оформили просечну оцену од **4.00**, што указује да они **сматрају да је потребно да науче много ствари пре него што би кренули да раде са системом (апликацијом АПЗ)**. Лакоћа приликом учења како се нека апликација користи је једна од најважнијих чинилаца употребљивости, према

многим ауторима. Због тога, је потребно посебно обратити пажњу на ову оцену корисника, и предузети мере како би се систем (апликација) учинила употребљивијом.

8.3. Анализа портфолија

Да би се одредило коју стратегију применити на одређену појединачну апликацију, потребно је спровести портфолио анализу. Портфолио анализа служи за процену апликација и њихову евалуацију, ради утврђивња да ли су оне и даље у равни са пословним процесима и техничким захтевима (Willem-Jan, 2009). У том контексту говори се о две димензије: техничким вредностима и пословним вредностима. Неки од аутора, као што је Снид имају слична разматрања и указују на посматрање апликација са становишта техничког квалитета и пословне вредност (Sneed, 1995).

На основу поменуте две димензије формирају се 4 квадранта, где имамо системе или апликације са ниском пословном вредношћу и ниским техничким квалитетом, апликације са високом пословном вредношћу и ниским техничким квалитетом, затим апликације са ниском пословном вредношћу и високим техничким квалитетом и апликације са високом пословном вредношћу и високим техничким квалитетом. У односу на четири поменута квадранта, могуће је одредити које стратегије применити у односу на наслеђени систем, тј. апликације (Willem-Jan, 2009) :

- када је пословна вредност ниска и технички квалитет низак, тада је потребно апликацију повући из употребе или заменити је новом
- када је пословна вредност висока и технички квалитет низак, тада је потребно спровести модернизацију
- када је пословна вредност ниска и технички квалитет висок, тада је потребно је спровести поновну процену
- када је пословна вредност висока и технички квалитет висок, тада је потребно апликацију одржавати

Дакле, анализа портфолија може послужити у одабиру адекватне модернизационе стратегије на основу оцене пословне вредности и техничког квалитета.

8.3.1. Процена пословне вредности

Проценити пословну вредност апликације, тј. колико апликација доприноси пословању, поготово када се разматра шта радити са апликацијама наслеђеног система, може бити веома сложен задатак. Један од једноставних приступа које заговара Снид, јесте да пословни аналитичар спроведе процену, где је потребно укључити вредност на

тржишту, допринос профиту и значај информација сваке од апликација која се подвргава процени (Sneed, 1995). На основу поменутих атрибута, свака појединачна апликација се може проценити у односу на остале апликације и онда говоримо о њиховој релативној пословној вредности.

Идеју Снида, додатно је разрадио Дедеке (Dedeke, 2012) који је идентификовао 6 кључних фактора за оцену пословне вредности система:

- компететивна предност (енг. *competitive advantage*)
- утицај на профитабилност (енг. *impact on profitability*)
- потенцијал за раст (енг. *growth potential*)
- стандардизација и усаглашавање (енг. *standardization and compliance*)
- међузависност система (енг. *systems interdependency*)
- безбедност система (енг. *systems security*)

Кључни фактори могу се проценити за наслеђени систем или апликације, на основу одговора испитаника, на начин да сваки од кључних фактора може да се провери са једним питањем. У табели 40., је дат приказ питања у односу на кључне факторе:

Табела 40. кључни фактори за оцену пословне вредности наслеђеног система

| Кључни фактор | Питање |
|--------------------------------|--|
| компететивна предност | Систем помаже организацији да оствари нове могућности |
| утицај на профитабилност | Параметри трошкова система су унутар буџета |
| потенијал за раст | Софтвер се може прилагодити променама у пословним захтевима и има подршку произвођача |
| стандардизација и усаглашавање | Систем може да задовољи нове стандардизационе профиле, укључујући регулаторне, индустријске и корпоративне стандарде за ИТ платформе, податке и апликације |
| међузависност система | Организација се у потпуности ослања на наслеђен систем |
| безбедност система | Систем је робустан у односу на сигурносне претње |

За сваки од кључних фактора дефинише се релативна важност фактора за један наслеђен систем или апликацију. Збир релативних важности фактора мора да буде 1.00. У склопу ове методе користи се и Ликертова скала са пет опција, где сваки учесник (испитаник) може да одабере једну од пет оцена за сваки од ставова тј. питања:

- у потпуности се не слажем
- не слажем се

- неутралан сам
- слажем се
- у потпуности се слажем

Сваком од одговора придружује се одговарајућа вредност обележја, тако да за одговор „у потпуности се не слажем“ придружује се вредност обележја 1, за одговор „не слажем се“ придружује се вредност обележја 2, за одговор „неутралан сам“ придружује се вредност обележја 3, за одговор „слажем се“ придружује се вредност обележја 4 и за одговор „у потпуности се слажем“ придружује се вредност обележја 5. На основу одговора испитаника, одређује се просечна вредност ставова испитаника за свако од 6 питања. Затим се просечна вредност одговора испитаника на одређено питање множи са релативном тежином фактора и добијају се тежински резултати. Сумирањем тежинских резултата добија се процењена пословна вредност апликације, односно система.

На примеру три наслеђене апликације, које су део наслеђеног информационог система ЈКП БВК, спроведена је процена пословне вредности наслеђених апликација, где менаџери који у свом послу користе те апликације су давали субјективне ставове за свако од 6 питања. Анкета је спроведена међу 19 менаџера нижег и средњег нивоа који раде са апликацијама у наслеђеном систему ЈКП БВК.

За апликацију АП1 7 менаџера је изнело своје ставове, за апликацију АП2 6 менаџера је изнело своје ставове, а за апликацију АП3 такође 6 менаџера је изнело своје ставове. Свако од њих је дао субјективну оцену на 6 кључних питања.

8.3.1.1. *Процена пословне вредности апликације АП1*

За апликацију АП1, 7 менаџера је изнело своје ставове, тако да су просечне вредности одговора на 6 кључних питања приказани у табели 41.

Табела 41. просечне вредности одговора на кључна питања за апликацију АП1

| Рбр. | Питање | Просечна вредност |
|------|--|-------------------|
| 1. | Систем помаже организацији да оствари нове могућности | 2.43 |
| 2. | Параметри трошкова система су унутар буџета | 4.86 |
| 3. | Софтвер се може прилагодити променама у пословним захтевима и има подршку произвођача | 2.57 |
| 4. | Систем може да задовољи нове стандардизационе профиле, укључујући регулаторне, индустријске и корпоративне стандарде за ИТ платформе, податке и апликације | 2.00 |
| 5. | Организација се у потпуности ослања на наслеђен систем | 4.86 |

| | | |
|----|--|-------------|
| 6. | Систем је робустан у односу на сигурносне претње | 4.00 |
|----|--|-------------|

За свако од кључних питања одређени су тежински кофицијенти који представљају релативну значајност фактора за апликацију АП1 наслеђеног информационог система:

- питању 1, придружен је тежински кофицијент 0.15
- питању 2, придружен је тежински кофицијент 0.15
- питању 3, придружен је тежински кофицијент 0.25
- питању 4, придружен је тежински кофицијент 0.2
- питању 5, придружен је тежински кофицијент 0.15
- питању 6, придружен је тежински кофицијент 0.1

На основу просечне вредности ставова менаџера у вези кључних фактора, као и придружених тежинских кофицијената сваком од фактора, срачунава се тежински резултат, који је приказан у табели. Сумирањем тежинских резултата одређује се тежинска пословна вредност апликације АП1.

Табела 42. тежинска пословна вредност апликације АП1

| Фактор | Просечна вредност | Тежински кофицијент | Тежински резултат |
|--|-------------------|---------------------|-------------------|
| компетитивна предност | 2.43 | 0.15 | 0.36 |
| утицај на профитабилност | 4.86 | 0.15 | 0.73 |
| потенијал за раст | 2.57 | 0.25 | 0.64 |
| стандардизација и усаглашавање | 2.00 | 0.2 | 0.40 |
| међузависност система | 4.86 | 0.15 | 0.73 |
| безбедност система | 4.00 | 0.1 | 0.4 |
| тежинска пословна вредност апликације АП1 | | | 3.26 |

8.3.1.2. Процена пословне вредности апликације АП2

За апликацију АП2, 6 менаџера је изнело своје ставове, тако да су просечне вредности одговора на 6 кључних питања приказани у табели 43.

Табела 43. просечне вредности одговора на кључна питања за апликацију АП2

| Рбр. | Питање | Просечна вредност |
|------|---|-------------------|
| 1. | Систем помаже организацији да оствари нове могућности | 2.83 |
| 2. | Параметри трошкова система су унутар буџета | 4.67 |
| 3. | Софтвер се може прилагодити променама у пословним захтевима и има подршку произвођача | 2.33 |

| | | |
|----|--|------|
| 4. | Систем може да задовољи нове стандардизационе профиле, укључујући регулаторне, индустријске и корпоративне стандарде за ИТ платформе, податке и апликације | 2.17 |
| 5. | Организација се у потпуности ослања на наслеђен систем | 5.00 |
| 6. | Систем је робустан у односу на сигурносне претње | 4.00 |

За свако од кључних питања одређени су тежински кофицијенти који представљају релативну значајност фактора за апликацију АП2 наслеђеног информационог система:

- питању 1, придружен је тежински кофицијент 0.1
- питању 2, придружен је тежински кофицијент 0.15
- питању 3, придружен је тежински кофицијент 0.25
- питању 4, придружен је тежински кофицијент 0.25
- питању 5, придружен је тежински кофицијент 0.1
- питању 6, придружен је тежински кофицијент 0.15

На основу просечне вредности ставова менаџера у вези кључних фактора, као и придружених тежинских кофицијената сваком од фактора, срачунава се тежински резултат, који је приказан у табели. Сумирањем тежинских резултата одређује се тежинска пословна вредност апликације АП2.

Табела 44. тежинска пословна вредност апликације АП2

| Фактор | Просечна вредност | Тежински кофицијент | Тежински резултат |
|--|-------------------|---------------------|-------------------|
| компетитивна предност | 2.83 | 0.1 | 0.28 |
| утицај на профитабилност | 4.67 | 0.15 | 0.70 |
| потенијал за раст | 2.33 | 0.25 | 0.58 |
| стандардизација и усаглашавање | 2.17 | 0.25 | 0.54 |
| међузависност система | 5.00 | 0.1 | 0.5 |
| безбедност система | 4.00 | 0.15 | 0.6 |
| тежинска пословна вредност апликације АП2 | | | 3.20 |

8.3.1.3. Процена пословне вредности апликације АП3

За апликацију АП3, 6 менаџера је изнело своје ставове, тако да су просечне вредности одговора на 6 кључних питања приказани у табели.

Табела 45. просечне вредности одговора на кључна питања за апликацију АП3

| Рбр. | Питање | Просечна |
|------|--------|----------|
|------|--------|----------|

| | | вредност |
|----|--|-----------------|
| 1. | Систем помаже организацији да оствари нове могућности | 2.33 |
| 2. | Параметри трошкова система су унутар буџета | 4.50 |
| 3. | Софтвер се може прилагодити променама у пословним захтевима и има подршку произвођача | 2.00 |
| 4. | Систем може да задовољи нове стандардизационе профиле, укључујући регулаторне, индустријске и корпоративне стандарде за ИТ платформе, податке и апликације | 1.67 |
| 5. | Организација се у потпуности ослања на наслеђен систем | 4.17 |
| 6. | Систем је робустан у односу на сигурносне прегње | 3.00 |

За свако од кључних питања одређени су тежински кофицијенти који представљају релативну значајност фактора за апликацију АПЗ наслеђеног информационог система:

- питању 1, придружен је тежински кофицијент 0.3
- питању 2, придружен је тежински кофицијент 0.1
- питању 3, придружен је тежински кофицијент 0.2
- питању 4, придружен је тежински кофицијент 0.1
- питању 5, придружен је тежински кофицијент 0.2
- питању 6, придружен је тежински кофицијент 0.1

На основу просечне вредности ставова менаџера у вези кључних фактора, као и придружених тежинских кофицијената сваком од фактора, срачунава се тежински резултат, који је приказан у табели. Сумирањем тежинских резултата одређује се тежинска пословна вредност апликације АПЗ.

Табела 46. тежинска пословна вредност апликације АПЗ

| Фактор | Просечна вредност | Тежински кофицијент | Тежински резултат |
|--|--------------------------|----------------------------|--------------------------|
| компететивна предност | 2.33 | 0.3 | 0.70 |
| утицај на профитабилност | 4.50 | 0.1 | 0.45 |
| потенијал за раст | 2.00 | 0.2 | 0.40 |
| стандардизација и усаглашавање | 1.67 | 0.1 | 0.17 |
| међузависност система | 4.17 | 0.2 | 0.83 |
| безбедност система | 3.00 | 0.1 | 0.3 |
| тежинска пословна вредност апликације | | | 2.85 |

8.3.2. Процена техничког квалитета

Процена техничког квалитета апликације је веома комплексан задатак. Постоје различити приступи који могу да се користе, где неки од њих употребљавају одређене метрике квалитета, при чему постоји велика дебата о томе које метрике применити, како им дефинисати минималне и максималне вредности. Такав приступ подразумева употребу специјализованих софтверских алата и оквира који омогућавају анализу овог типа. Такође, један од важних фактора примене овог приступа у процени техничког квалитета јесте обучено особље, које у потпуности влада са знањима потребним да спроведе процену на одговарајући начин. Дакле, примена оваквог систематичног приступа изискује време и значајне финансијске и људске ресурсе.

Један од проблема који се може јавити у пракси, јесте да за одређена софтверска окружења такви алати нису доступни, као што је нпр. Progress OpenEdge окружење, које ЈКП БВК користи. Такође, уколико имамо нестандардног произвођача софтвера, који има свој власнички програмски језик, као што је Progress Software Corporation, поставља се питање које одговарајуће метрике је потребно применити на таквим специфичним окружењима.

Услед свега наведеног, постоји потреба да се изнађе начин да се процени квалитет софтверског решења када имамо нестандардна софтверска окружења у којима је имплементиран наслеђени информациони систем. Један од начина за процену техничког квалитета апликације, може бити посредним путем преко анализе употребљивости. Употребљивост је један важан аспект квалитета софтвера (квалитет у употреби) или како се још назива атрибут, карактеристика и тд. Она утиче на ефикасност, ефективност и задовољство корисника који употребљава тај производ у одговарајућем контексту.

За потребе овог рада искористићемо резултате анализе употребљивости презентоване у одељку 8.1 и 8.2 овог поглавља, где се употребљавала SUS метода. На основу добијених резултата ћемо проценити технички квалитет апликација наслеђеног система које у развијене у програмском језику 4. генерације, Progress 4GL. Према томе:

- апликација АП1, има SUS скор **44.25**
- апликација АП2, има SUS скор **46.77**
- апликација АП3, има SUS скор **44.15**

8.3.3. Разматрање еволуционе стратегије апликација наслеђеног система

На основу процене техничког квалитета и пословне вредности апликација АП1, АП2 и АП3 за наслеђени информациони систем ЈКП БВК, могуће је одредити коју стратегију еволуције применити на сваку од поменутих апликација, анализом портфолија.

Пројектовањем вредности техничког квалитета апликација на вертикалну осу и пројектовањем пословне вредности апликација на хоризонталну осу, сваку од апликација стављамо у један од 4 квадранта. Потребно је да имамо у виду да тежинску пословну вредност апликације је потребно да пројектујемо на оси која има вредности од 0 до 100. Пропорционалним рачуном добићемо вредности:

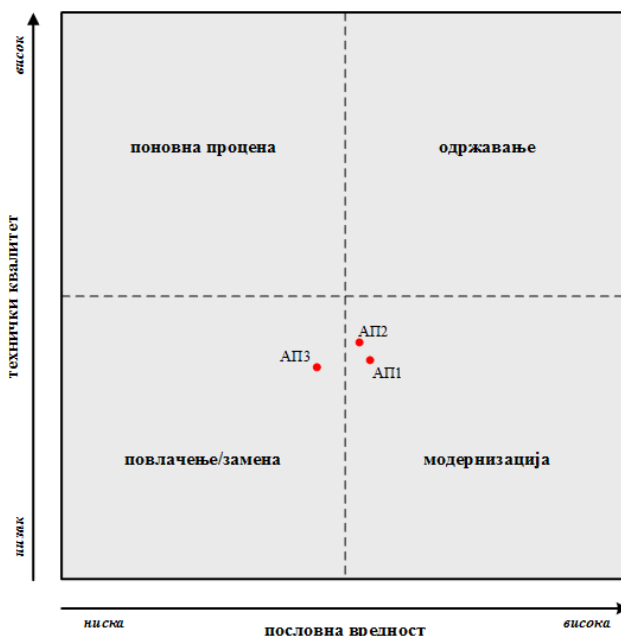
- апликација АП1, **54**
- апликација АП2, **53**
- апликација АП3, **47.5**

На слици 63. можемо видети позицију сваке од 3 наведене апликације АП1, АП2 и АП3 и којем квадранту припада.

Технички квалитет апликације АП1 је ближе нижим вредностима, док је њена пословна вредност ближе вишим вредностима, па апликација АП1 припада доњем десном квадранту. **Апликацију АП1 је потребно модернизовати.**

Технички квалитет апликације АП2 је ближе нижим вредностима, док је њена пословна вредност ближе вишим вредностима, па апликација АП2 припада доњем десном квадранту. **Апликацију АП2 је потребно модернизовати.**

Технички квалитет апликације АП3 је ближе нижим вредностима, док је њена пословна вредност ближе вишим вредностима, па апликација АП3 припада доњем десном квадранту. **Апликацију АП3 је потребно модернизовати.**



вредност ближе
вредностима, па
припада доњем
квадранту.
АП3 је
повући или

Слика 63. портфолио апликације АП1, АП2 и АП3

8.4. Разматрање приступа модернизацији апликација

8.4.1. Разматрање повлачења или замене апликације АП3

Апликација АП3 нема задовољавајућу пословну вредност и технички квалитет, па је потребно повући је из употребе или је заменити. Уколико се апликација АП3 повуче из употребе, потребно је имати у виду да пословне операције које је апликација подржавала неће више бити део информационог система, тј. мораће да се спроведе мануелно.

Један од могућих приступа у замени апликације АП3 може да буде куповина готовог софтверског решења, тј. одређеног комерцијалног пакета који је потребно да адекватно замени и побољша све функционалности које је постојећа апликација извршавала. Такође, потребно би било обратити пажњу на аспект интеграције уколико се спроведе набавка одређеног готовог комерцијалног пакета, јер остале апликације наслеђеног система је потребно да имају одређену размену података са новом купљеном апликацијом, као и покретање пословних операције које се пропадају кроз више апликација. То може бити проблематично јер постојећи наслеђени систем ЈКП БВК нема адекватан ниво модуларности, па се одабир ове алтернатива може показати као погрешан. Додатни проблем је тесна повезаност, која доводи до ефеката таласа, па је потребно додатно бити обазрив. Обзиром да цена готових комерцијалних пакета може да буде значајно већа у односу на функционалности које се нуде, као и дуг период њиховог прилагођавања док не заживе у потпуности унутар информационог система, поставља се

питање да ли би развој новог решења из сопствених извора могао да буде боља алтернатива.

Имајући у виду да ЈКП БВК има могућност сопственог развоја, без ослањања на спољног добављача софтвера, могуће је дефинисати програмски захтев у којем је потребно реализовати нову апликацију која је потребно да замени апликацију АПЗ наслеђеног система, при чему у том програмском захтеву је потребно обрадити и нове функционалности тако да се увећа пословна вредност те нове апликације. Како ЈКП БВК располаже интегрисаним развојним окружењем Visual Studio, та нова апликација се може развити у Microsoft .NET технологији. Архитектура решења може бити организована на начин да се омогући једноставна поновна употреба функционалности апликације дефинисањем API-а, где ће друге апликације моћи да комуницирају и размењују податке са новом (уз одређена промене, тј. начине како да се то омогући) посредством API-а. Такво решење може да има значајно мање трошкове одржавања него постојећа апликација са једне стране, док са друге стране захтева пуно ангажовање пројектаната и програмера. Такође, оно што је потребно размотрити јесу и трошкови обуке које је потребно организовати, тако да развојни тим усвоји потребна знања да би успешно могао да реализује нову апликацију.

Зато је потребно упоредити све важне аспекте са становишта ИТ менаџмента приликом одабира једне од поменутих алтернатива, како са технолошког аспекта, тако и са економског аспекта, као и са аспекта менаџмента људских ресурса. Такође један важан аспект који је потребно да буде инкорпориран у нову апликацију је и могућност еволуције нове апликације, у складу са променама које долазе са стратешког и пословног нивоа. У том смислу не би требало само посматрати садашња ограничења које намеће наслеђени информациони систем, а која треба превазићи, већ имати у виду и будуће промене које су неминовне.

Из свих наведених разлога је потребно спровести анализу по више критеријума и тек тада се одлучити који приступ предузети у замени апликације АПЗ, тако да буде одабрано најбоље могуће решење.

8.4.2. Разматрање стратегије модернизације апликација АП1 и АП2

На основу свих претходних анализа и закључака, потребно је одредити модернизационе правце за апликације АП1 и АП2 наслеђеног информационог система ЈКП БВК.

Стратегија модернизације може да настане у пресеку:

- новог теоријског оквира насталог посматрањем наслеђеног информационог система на ЈКП БВК и анализе ланца вредности ЈКП БВК (студија случаја)
- испитивања употребљивости наслеђеног система ЈКП БВК
- процене пословне вредности апликација АП1 и АП2 наслеђеног информационог система ЈКП БВК
- постојећих разматрања експерата из области у вези програмских језика 4. генерације

Према посматрању наслеђеног система ЈКП БВК на основу предложеног теоријског оквира, могу се сагледати следећи његови проблеми у кратким цртама:

- развојна перспектива
 - смањено знање о систему, тешко преношење знања и чињеница да знање није доступно на једном месту
 - систем развијен унутар организације (in-house)
 - нарушена архитектура
 - учестали ефекти таласа
 - значајно прилагођавање приликом замене апликације
 - не постоји раздвајање слојева у програмском окружењу (не постоји раздвајање у једном програму презентације на корисничком интерфејсу, пословне логике и комуникације са базом података)
- оперативна перспектива
 - застарели хардвер/софтвер
 - лоша подршка произвођача, проблем компатибилности верзија, као и мале могућности едукације и мала заједница у односу на решења отвореног кода
 - проблеми у интеграцији, услед апликационих силоса
 - тешка интеграција са другим а за ЈКП БВК значајним софтверским окружењима (ГИС - географски информациони систем, EDAMS - за управљање средствима инфраструктуре и SCADA)
 - отежана, неаутоматизована или полу-аутоматизована размена података са другим организацијама
- организациона перспектива

- ограничава пословање и промене у пословању
- ограничава подршку у одлучивању, услед неадекватног функционисања са системима за пословно извештавање
- стратешка перспектива
 - ограничава измену стратегије
 - ограничава рад са технологијама које су елементи индустрије 4.0. (а које су значајне ради побољшања ефикасности и ефикасности операција које се налазе у ланцу вредности ЈКП БВК: извлачење сирове воде, дистрибуција сирове воде, дистрибуција пијеће воде, продаја, сакупљање канализације):
 - интернет ствари и паметни мерачи
 - технологија облака, Fog и Edge рачунарство
 - технологија великих података
 - вештачка интелигенција, машинско учење и детекција аномалија (као посебан део машинског учења)

На основу испитивања употребљивости наслеђеног система уочени су следећи проблеми:

- констатује се да је употребљивост система на ниском нивоу
- указује се на смањену могућност за учење система
- указује се да су функционалности лоше интегрисане у систему
- указује се да постоји неконзистентност у систему, као резултат редувансе и лоше архитектуре

На основу процене пословне вредности апликација АП1 и АП2, уочено је следеће:

- апликације не подржавају искоришћавање нових могућности
- апликације нису прилагођене на промене у пословању
- апликације не користе стандарде ИТ индустрије, корпоративне стандарде, као и поновну употребљивости
- ЈКП БВК се у потпуности ослања на апликације наслеђеног система
- трошкови апликације и система су унутар буџета

Експерти из области модернизације износе своја мишења у вези модернизације наслеђених система написаних у програмским језицима 4. генерације.

Снид указује да једна од важних чињеница коју треба размотрити приликом модернизације наслеђеног система јесте да се наслеђени софтвер, који је део наслеђеног система може поделити у 3 основне категорије у односу на степен зависности од његовог окружења (Sneed, 2006):

- програми који су независни од својег окружења
- програми који су парцијално зависни од својег окружења
- програми који су у потпуности зависни од својег окружења

Програмски језици четврте генерације, међу којима је и Progress 4GL у којем је написан наслеђен информациони систем ЈКП БВК, припадају трећој групи и они захтевају специфично окружење у којем је потребно да се извршавају. Како Снид истиче, овакав софтер не може да се поново користи у другом окружењу, јер је зависан од окружења и једини начин да се овакви софтвери користе јесте држати их у њиховом нативном окружењу, а изградити одређене везе према том окружењу (Sneed, 2006).

Разматрајући ову класу система, постоје ставови експерата из области према којима програмски језици четврте генерације везују предузеће које га користи за произвођача, који није стандардан и не обезбеђују исти ниво флексибилности као програмски језици треће генерације или објектно-орјентисани језици (Linthicum, 1994).

Обзиром да програмски језици четврте генерације нису стандардизовани и често програмери немају до краја знање о томе шта се дешава у позадини, јер постоји доста генерисаног програмског кода од стране самог окружења, није лако схватити како ови програмски језици раде и како спроводе своје операције (D. E. Avison & Shah, 1992).

Груб и Таканг, наводе основне њихове недостатке: апликациона специфичност, власништво, сензационалистичко размишљање и лош дизајн (Grubb & Takang, 2003).

Разматрања експерата из области у вези програмских језика 4. генерације, па самим тим и Progress 4GL окружења у којем је израђен наслеђени информациони систем ЈКП БВК могу се сублимирати:

- немогућност да се користи у другом окружењу сем изворног
- систем (апликације) оставити у том окружењу, а изградити везе ка њима
- језик је доменски специфичан и направљен за развој апликација које одговарају пословном домену, а све концепте изван тога је веома тешко корисити

- рапидан развој апликација у овом окружењу је такав да не постоји јасно раздвајања на слојеве, као на примеру трослојних апликација: слој презентације, слој пословне логике и слој комуникације са базом података, већ комбиновање свега у једном програмског коду - што отежава процес одржавања и будућих измена
- језик је нефлексибилан у односу на модерне објектно-орјентисане језике
- везаност за произвођача софтвера, PSC Corporation, који условљава развој тог окружења и стално додаје одређена власничка проширења да би се омогућио рад са новим технологијама
- непознавање до краја начина рада самог програмског окружења, јер окружење у позадини додаје одређен програмски код путем којег се омогућава функционисање апликација у том окружењу

Приступу које ћемо размотрити у модернизацији наслеђеног система су:

- софтверски реинжењеринг
- облагање
- миграција

8.4.2.1. Разматрање софтверског реинжењеринга као стратегије модернизације апликација АП1 и АП2

Имајући у виду нови теоријски оквир, испитивање употребљивости наслеђеног система, процену пословне вредности апликација АП1 и АП2 наслеђеног информационог система ЈКП БВК, као и постојећих разматрања експерта из области у вези програмских језика 4. генерације, може се рећи да спровођење процеса софтверског реинжењеринга је проблематично из неколико разлога који ће бити наведени.

Софтверски реинжењеринг укључује реверзни инжењеринг (апстракција), трансформација (измена) и инжењеринг у напред (усавршавање). У случају Progress 4GL платформе софтверски реинжењеринг укључује доста неаутоматизованог рада програмера, пројектаната и архитеката система, јер је из програмског кода потребно добити апстракције вишег нивоа, нпр. дизајн и архитектуру система, а при томе не постоје софтверски алати који би омогућили да се то спроведе на аутоматизован начин и да се резултат понуди у визуелној форми. Програмери, пројектанти и архитекте система све самостално морају да ураде без ослањања на софтверске алате, који су им недоступни.

Поред тога, Progress 4GL услед тога што омогућава рапидан развој софтвера, апликације наслеђеног система су развијене без јасног раздајања слојева, као и задужења које би ти слојеви требало да имају у укупном систему, а ради лакшег одржавања. Због тога, готово сваки програм у једној програмској процедури садржи програмске инструкције које приказују резултате на графичком корисничком интерфејсу, затим извршавају одређену пословну логику и поред тога комуницирају са базом. Зато није једноставно реконструисати структуру апликације (нпр. у форми UML дијаграма класа) и понашање апликације (нпр. у форми UML дијаграма секвенци), па је потребно уложити много мануелног рада да би се повратило знање о систему, које не постоји у некој форми, јер сама документација не постоји. Дакле, редокументација захтева много мануелног рада, услед недостатка знања и специјализованих алата који би то олакшали.

Софтверски реинжењеринг подразумева фазу трансформације у којој се на одређеном нивоу апстракције, врши трансформација из старог у ново окружење. У случају Progress 4GL програмског језика и OpenEdge платформе то је веома тешко спровести, јер Progress 4GL у позадини додаје одређени код који програмер не види, а да би нормално радио у својем окружењу. Дакле, не само што је потребно водити рачуна о томе како транслирати програмски код, већ је потребно повести рачуна и како транслирати програмске инструкције које аутоматски додају у Progress 4GL окружењу да би могао нормално да ради. Софтверски алат који би требало да омогући поменути трансформацију, морао би да поменуте опције има уграђене у себе.

Софтверски реинжењеринг укључује фазу инжењеринга у напред, тзв. усавршавања и у оквиру ове фазе би потребно било да се измени архитектура постојећих апликација, на начин да се апликације наслеђеног система инкорпорирају у нову архитектуру. То би подразумевало уклањања тренутног начина остваривања интеграције између апликација, као и доста активности које треба да изложе трансформисане апликације у новом систему, преко новедефинисаних интерфејса интеграције. То може бити веома захтева активност, поготово уколико софтверски алати који треба да подпомогну процес реинжењеринга не постоје. Обзиром да је то веома тешко или скупо спровести, проблеми које би остали отворени се тичу и развојне и оперативне перспективе. Дакле, архитектура би и даље била нарушена, не би се спречили учестали ефекти таласа јер и даље постоје чврсте везе међу апликацијама и базама података. Поред тога, проблеми интеграције не би били решени, како са интерним системима ЈКП БВК, тако и са системима екстерних организација.

Делимично би се решили проблеми организационе перспективе, јер би апликација могла једноставније да промени своју интерну структуру уколико су принципи дизајна примењени адекватно. Део проблема који би се задржао унутар ове перспективе тиче се спровођења заједничких пословних операција са другим системима.

Делимично би се решили проблеми стратешке перспективе који се тичу измене стратегије, док би се део проблема који се тиче архитектуралних измена задржао. Не би се решио проблем рада са технологијама који су елементи индустрије 4.0, уколико се не спроведу архитектуралне измене и проширење постојећих апликација.

Имајући у виду све наведено, софтверски реинжењеринг као стратегија модернизације апликација АП1 и АП2 које су део наслеђеног информационог система ЈКП БВК написаног у Progress 4GL програмском решењу је окарактерисана:

- непостојањем алата који омогућавају једноставније спровођење реинжењеринга у односу на мануелни рад пројектног тима
- великим и дуготрајним радним ангажовањем и трошковима на задатку реинжењеринга
- нерешавање проблема софтверске архитектуре
- нерешавање проблема интеграције
- додатно ангажовање око решавања проблема архитектуре и интеграције, уз додатно ангажовање око процеса инжењеринга у напред тако да се апликације прошире и уклопе у нову софтверску архитектуру
- побољшање интерне структуре апликација
- након спровођења реинжењеринг на апликацијама АП1 и АП2, поступак би потенцијално могао да се примени и на остатку наслеђеног система, што може бити значајан подухват

8.4.2.2. Разматрање облагања као стратегије модернизације апликација АП1 и АП2

На основу разматрања експерата из области модернизације у вези програмских језика 4. генерације, проблема наслеђених система исказаних преко различитих перспектива новог теоријског оквира, испитивања употребљивости наслеђеног иситема, као и процене пословне вредности апликација АП1 и АП2, облагање апликација наслеђеног система може се посматрати као адекватна стратегија модернизација за ЈКП БВК. Разлози ће бити наведени.

Облагање нам даје могућност да сукцесивно вршимо модернизацију једне по једне апликације, тако да она може да учествује у извршавању пословних операција, без прекидања и нарушавања њеног досадашњег начина рада. Она може да настави да функционише унутар окружења у којој је инцијално развијена, али такође омогућава јој се да учествује у извршавању пословних операција заједно са другим апликацијама, јер се увећава њена модуларност.

Спровођење облагања се може спровести на нивоу базе података, тако што би се употребом ODBC конекције, повезали са SQL Engine-ом Progress базе података и могли да спроводимо SQL упите. На поменути начин би се заобишло извршавање постојеће апликације наслеђеног система, која је написана у Progress 4GL програмском језику. API за наслеђену апликацију би могао да буде реализован у Microsoft ASP.NET WEB API технологији (или некој сличној), тако да би се позивањем његових функција, из Progress базе података извлачили подаци употребом ADO.NET технологије рада са базама података у .NET окружењу и прослеђивали било коме које би позивао API. Програмски код у којем би био реализован API би путем ODBC конекције вршио операције над Progress базом података, чиме би се енкапсулирала наслеђена апликација и створила могућност да постане део шире сервисно-орјентисане архитектуре. Функционалности наслеђене апликације би путем API-а било изложене остатку система, тако да би по потреби могле да буду позване.

На описани начин створила би се могућност да се апликације интегришу у нову софтверску архитектуру употребом специјализоване технологије средњег слоја, као што је нпр. пословна сабирница (енг. Enterprise Service Bus). Одређеним изменама програмског кода у наслеђеним апликацијама би се створила могућност да се интеграција не обавља преко база података и процедура које се извршавају ван радног времена, а које преносе податке из једне базе у другу, већ да се посредством API-а позивају функционалности апликација и да преко технологије средњег слоја размењују податке.

Тиме би се решили проблеми развојне перспективе у погледу архитектуре и учесталих ефеката таласа у склопу развојне перспективе.

На нивоу оперативне перспективе решили би се проблеми интеграције, јер би се омогућило да ка свим другим системима посредством технологије средњег слоја се изврши размена података и покрену одређене аутоматизоване операције.

На нивоу организационе перспективе постојала би могућност да се реше проблеми који би захтевале измене на нивоу пословних процеса, као што је

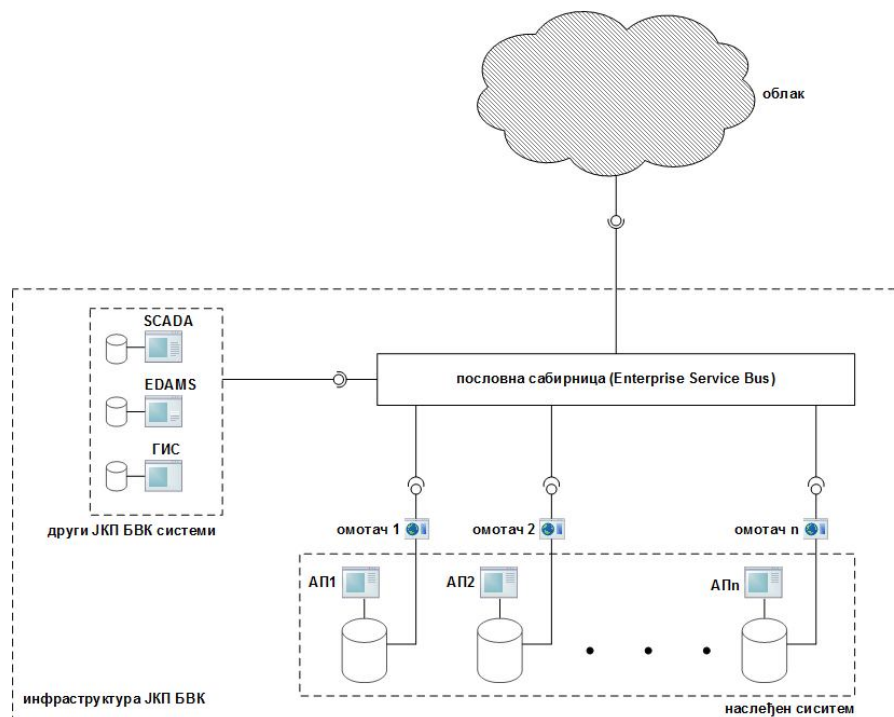
реинжењеринг пословних процеса. Део проблема у вези измене саме апликације би остао нерешен, али би се створила могућност, да у односу на приоритет једна по једна апликација из старог окружења по потреби се пребаци у неко ново окружење, при чему би интерфејси ка осталим деловима система били јасно дефинсани.

На нивоу стратешке перспективе делимично би се решили проблеми који се тичу измене стратегије на нивоу појединачне апликације, али би се створила могућност да се она пребаци у ново окружење по потреби, без нарушавања рада осталих апликација. Такође, омогућила би се интеграција функционалности наслеђених апликација са елементима индустрије 4.0, тако што би наслеђене апликације могле да постану део хибридног облака.

Имајући у виду све наведено, облагање као стратегија модернизације апликација АП1 и АП2 које су део наслеђеног информационог система ЈКП БВК написаног у Progress 4GL програмском решењу је окарактерисана:

- ангажовањем програмера на развоју омотача у .NET окружењу који би користио ODBC конекцију
- употреба постојећег лиценцираног од стране ЈКП БВК интегрисаног развојног окружења Visual Studio
- ангажовањем програмера на делу измене постојећих апликација наслеђеног система ради уклањања веза ка осталим апликацијама
- ангажовањем пројектаната и архитеката да уклопе омотане апликације у нову архитектуру
- могућност да се сачувају функционисања наслеђених апликација и изложе да други делови система могу да их користе, а при томе да се не троше ресурси на њихов поновни развој
- могућност за једноставнијим решењем проблема интеграције како са интерним системима ЈКП БВК, тако и са екстерним системима других предузећа
- могућност за једноставније уклапање у нову архитектуру
- могућност за једноставније интегрисање и рад са технологијама индустрије 4.0 уз уклапање у архитектуру хибридног облака
- немање утицаја на интерну структуру апликације
- стварање могућности да се постепено пребаце апликације у ново окружење према исказаном приоритету у пословању (поготово оних код којих су промене чешће и интензивније)

На слици 78. је приказана архитектурална замисао поменутог решења, које у себе укључује облагање апликација, као стратегију модернизације наслеђеног система ЈКП БВК. Уз употребу омотача, омогућава се да апликације размењују поруке са другим апликацијама наслеђеног система, али и са другим системима у оквиру ЈКП БВК, као што су ГИС, EDAMS и SCADA. Такође на поменути начин се омогућава да апликације наслеђеног система могу да комуницирају са апликацијама које се налазе у облаку, као и да позивају њихове функције путем API-а, и да размењују податке са њима. Поред тога ствара се могућност да се изврши постепена замена једне по једне апликације наслеђеног система.



Слика 64. архитектурална замисао облагања наслеђеног система ЈКП БВК и његово инкорпорирање у ширу архитектуру хибридног облака

8.4.2.3. Разматрање миграције као стратегије модернизације апликација API1 и API2

Имајући у виду нови теоријски оквир, испитивање употребљивости наслеђеног система, процену пословне вредности апликација API1 и API2 наслеђеног информационог система ЈКП БВК, као и постојећих разматрања експерта из области у вези програмских језика 4. генерације, може се рећи да спровођење миграције је веома захтевно и потенцијално неисплативо из неколико разлога који ће бити наведени.

Миграција (миграција софтвера) би захтевала да се апликација АП1 и АП2 развију од почетка, као и да се уклопе у нову софтверску архитектуру коју је потребно осмислити.

Наслеђени информациони систем ЈКП БВК развијен из сопствених средстава, па и поред великог одласка запослених ван предузећа, одређено знање о систему се задржало код ангажованих програмера и пројектанта. На тај начин је лакше доћи до функционалности које обавља наслеђен систем, него да је неко развијао споља, али и даље остаје рад на поновном сазнавању функционалности које систем обавља. Дакле, пре него што се крене у миграцију и развој нове архитектуре и апликација, потребно је урадити посао редокументације да би се вратила знања о функционалностима постојећег наслеђеног система, а то је прва фаза реверзног инжењеринга.

Тренутно разматрамо миграцију само 2 апликације, али када причамо о наслеђеном систему ЈКП БВК апликација има око 40, што значи да је за сваку апликацију потребно спровести захтеван посао миграције.

Са развојем нове апликације, стара апликација се повлачи из употребе и на тај начин предузеће отписује вишегодишња уложена средства у њен развој и одржавање.

Приликом спровођења миграције, обзиром да је тешко очекивати да се она одједном спроведе, према Cold Turkey приступу, потребно је одржавати бројне везе апликације са остатком наслеђеног система, па је потребан додатни рад око ангажовања на том захтеву и то тако све док последња апликација немигрира на нову платформу.

Имајући у виду све наведено, миграција као стратегија модернизације апликација АП1 и АП2 које су део наслеђеног информационог система ЈКП БВК написаног у Progress 4GL програмском решењу је окарактерисана:

- дугорочним ангажовањем и трошковима развоја нових апликација
- ангажовањем око развоја нове архитектуре и укалапања апликација
- дугорочним ангажовањем око сазнавања функционалности наслеђеног система и његовим поновним документовањем - 1. фаза реверзног инжењеринга
- повлачење из употребе средстава у које је годинама инвестирано
- дугорочним ангажовањем и трошковима развоја веза са наслеђеним системом док се све апликације наслеђеног система не пребаце на нову платформу

- потенцијална могућност да се проблеми наслеђеног система превазиђу

8.4.2.4. Предлог стратегије модернизације апликација АП1 и АП2

За апликације АП1 и АП2 размотрене су три различите стратегије модернизације: софтверски реинжењеринг, облагање апликација и миграција. Из свега наведеног закључујемо да је облагање апликација најбоља алтернатива, која се може спровести у најкраћем времену, уз ограничена финансијска средства и ограничене људске ресурсе, а уз постојеће софтверске алате који су већ у поседу ЈКП БВК. Такође предност ове стратегије јесте и то што је неинвазивна, што значи да су смањене могућности да постојећи наслеђени систем се наруши док се спроводи његово облагање.

На поменути начин апликације наслеђеног система су обложене и постају део нове архитектуре, која има могућност да функционише са технологијама облака, интернет ствари, великих података и тд. На тај начин могуће је превазићи све проблеме који су презентовани у развојној, оперативној, организационој и стратешкој перспективи.

Одабиром облагања као стратегије модернизације апликација, стварају се могућности да се реше проблеми развојне перспективе и то нарушена архитектура система, учестали ефекти таласа, редуанса у систему и тд. Такође ствара се основа да се систем поново документује, да би се све модернизационе активности спровеле.

Можемо закључити да други део ПОСЕБНЕ ХИПОТЕЗЕ 1: „ а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема“ је доказан.

Дакле, ПОСЕБНА ХИПОТЕЗА 1: „Посматрањем развојне перспективе наслеђеног система могуће је сагледати проблеме наслеђених система из улоге запослених који га развијају, а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема“ је доказана у целости. (први део ПОСЕБНЕ ХИПОТЕЗЕ 1, доказан је у претходном поглављу).

Одабиром облагања као стратегије модернизације апликација, стварају се могућности да се реше проблеми оперативне перспективе и то застарели софтвер и проблеми у компатибилности верзија, проблеми у интеграцији међу апликацијама, проблеми у интеграцији са другим а за ЈКП БВК значајним системим (ГИС, EDAMS и SCADA), проблеми размене податка са системима других организација.

Можемо закључити да други део ПОСЕБНЕ ХИПОТЕЗЕ 2: „ а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема“ је доказан.

Дакле, ПОСЕБНА ХИПОТЕЗА 2: „Посматрањем оперативне перспективе наслеђеног система могуће је сагледати проблеме наслеђених система из улоге корисника, а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема“ је доказана у целости. (први део ПОСЕБНЕ ХИПОТЕЗЕ 2, доказан је у претходном поглављу).

Одабиром облагања као стратегије модернизације апликација, стварају се могућности да се реше проблеми организационе перспективе и то проблеми у којима се ограничава пословање и промене у пословању, као и да се побољша подршка у одлучивању.

Можемо закључити да други део ПОСЕБНЕ ХИПОТЕЗЕ 3: „ а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема“ је доказан.

Дакле, ПОСЕБНА ХИПОТЕЗА 3: „Посматрањем организационе перспективе наслеђеног система могуће је сагледати проблеме наслеђених система из улоге менаџера, а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема“ је доказана у целости. (први део ПОСЕБНЕ ХИПОТЕЗЕ 3, доказан је у претходном поглављу).

Одабиром облагања као стратегије модернизације апликација, стварају се могућности да се реше проблеми стратешке перспективе и то проблеми у којима се ограничава измена стратегије и ограничава рад са технологијама које су елементи индустрије 4.0.: рачунарство у облацима, интернет ствари, велики подаци и тд.

Можемо закључити да други део ПОСЕБНЕ ХИПОТЕЗЕ 4: „ а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема“ је доказан.

Дакле, ПОСЕБНА ХИПОТЕЗА 4: „Посматрањем стратешке перспективе наслеђеног система могуће је сагледати проблеме наслеђених система из улоге топ менаџмента, а на основу тога и креирати такав путоказ ка модернизацији система који треба да обухвати решавање ових проблема.“ је доказана у целости. (први део ПОСЕБНЕ ХИПОТЕЗЕ 4, доказан је у претходном поглављу).

На основу тога што други део ПОСЕБНЕ ХИПОТЕЗЕ 1 је доказан, затим други део ПОСЕБНЕ ХИПОТЕЗЕ 2 је доказан, затим други део ПОСЕБНЕ ХИПОТЕЗЕ 3 је доказан и други део ПОСЕБНЕ ХИПОТЕЗЕ 4 је доказан, закључује се да други део ГЛАВНЕ ХИПОТЕЗЕ: „и на основу тога креирати путоказ ка његовој модернизацији“ је доказан.

Дакле, ГЛАВНА ХИПОТЕЗА: „Посматрањем различитих перспектива (тачки гледишта) наслеђеног информационог система као и његове перспективе (поглед ка будућности) могуће је потпуније сагледати наслеђени систем и на основу тога креирати путоказ ка његовој модернизацији“ је доказана у целости. (први део ГЛАВНЕ ХИПОТЕЗЕ, доказан је у претходном поглављу).

8.5. Остали аспекти модернизације

Да би се у потпуности сагледао поступак модернизације, као један веома захтеван поступак, потребно је размотрити и нетехничке аспекте који предмет проучавања ове тезе, али су у пракси веома значајни и требало би обратити пажњу на њих. Пре свега мисли се на економске и организационе аспекте.

Економски аспект треба да понуди одређне економске показатеље који би указали да ли је поступак спровођења модернизације оправдан или није, а уколико је оправдан какве користи се очекују од спровођење одабране стратегије. Пре свега пројекат је потребно да буде оправдан, а то би значило да се детаљно извржи анализа постојећег софтверског решења, као и процеса одржавања, где је потребно применити одговарајуће метрике које би указале на интензитет одржавања, просечни број радних сати који запослени проведу на решавању одређених проблема у одржавању, као и финансијске показатеље тренутног процеса одржавања. Ово је веома озбиљан задатак и захтева пажљиву анализу потребних метрика које је потребно одабрати, ангажовање специјализованог техничког и обученог особља, као и добављање специјализованих алата које имају могућности да спроведу жељене анализе. Након тога је потребно оценити у којој мери ће апликације кандидати за модернизацију ће утицати на побољшање профита, као и на смањење одржавања. Након тога је потребно проценити и вредност подухвата модернизације. На крају упоредити и утврдити да ли се исплати спроводити модернизацију, тако да трошкови модернизације буду мањи у односу на повећање профита и смањења трошкова одржавања. За сваку од поменутих активности је потребно спровести детаљану анализу пре него што се крене у подухват модернизације.

Организациони аспекти се тичу пре свега ангажовања тима који је потребно да спроведе пројекат модернизације. Као што пракса показује да је најлешће боље да људи који нису развијали софтверско решење, врше његово тестирање и оцену квалитета, тако је боље да тим који ће спровести модернизацију није радио на развоју решења, да би постојала потребна објективност у поступку. У почетку је важно дефинисати јасне циљеве који је потребно да тим за модернизацију спроведе. Некада је то само евалуација одређеног софтверског решења, некада је и предлог модернизације постојећег система, као и реализација будућег решења. Посебну пажњу би требало обратити на знања и вештине које би чланови тима требало да имају, тако да могу да користе алате намењене за ову сврху, метрике које су потребне за евалуацију, као и јасне методологије коју је потребно да примене. Поред тога, требало би да имају одређено искуство рада са наслеђеним системом, тако да би методе и алате могли да искористе на најбољи могући начин. Такође је битно да успоставе адекватну сарадњу са менаџментом који користи наслеђени систем, као и да буду упознати са одређеним стратешким правцима које дефинише топ менаџмент да би модернизација била успешна. Дакле, обухватањем свих перспектива једног наслеђеног система и дефинисање јасних планова и циљева који треба да реше проблеме дефинисане на неком од ових нивоа, могу јасније усмерити модернизацију.

9. Закључак

9.1. Доприноси

Наслеђени системи су системи развијени у прошлости, најчешће у старој или застарелој технологији, при чему се значајно опиру променама и модификацијама које је потребно спровести да би информациони систем могао да одговори на промене у пословању. Значајна средства која се издвајају за одржавање наслеђених система, велики број пропалих пројеката модернизације, као и различита схватања у теорији указују да је овај феномен комплексан и да је недовољно схваћен. Дакле, да би процеси модернизације наслеђених система били успешни потребно је пре свега адекватно разумети наслеђене системе са свим њиховим елементима.

Успостављањем новог теоријског оквира, као главног научног **доприноса** дисертације, који интегрише различите приступе у посматрању наслеђених система, ствара се могућност за њихово свеобухватно посматрање, на начин да се они посматрају кроз перспективе: развојну, оперативну, организациону и стратешку. Тиме се обезбеђује да сви проблеми које наслеђени систем има, посматрано из угла развојног тима, затим корисника система, менаџера и топ менаџмента, се сагледају и спречи посматрање само једног дела њих. На напред наведен начин избегава се изоловано посматрање наслеђеног система као феномена и указује се на све његове проблеме које он има тренутно, а које перципира развојни тим кроз развојну перспективу, корисници кроз оперативну перспективу, менаџери кроз организациону перспективу и топ менаџмент кроз стратешку перспективу.

Данашње пословање се одвија у једном веома променљивом окружењу где постоји изражена конкуренција на тржишту, па се од предузећа захтева да антиципирају будуће промене, припреме се на измењене околности и реагују на њих на прави начин, тако да осигурају реализацију своје мисије, визије и циљева. Процес модернизације наслеђеног система треба да укључи и могућност да се реагује на промене овог типа и зато је важно стратешку компоненту проширити и обогатити перспективом, тј. погледом у будућност. На тај начин се обезбеђује да значајно инвестирана средства у процес модернизације буду искоришћена ефективно и ефикасно и да се спречи да предузеће дође у ситуацију да након спровођења модернизационог подухвата, убрзо мора да спроведе нови. Анализом ланца вредности на случају ЈКП БВК, уочавају се које су то примарне активности које могу да се побољшају у ланцу вредности, као и технолошка решења која

могу да донесу описана побољшања, што је додатни **допринос** тезе. Утврђивањем могућности наслеђених система да раде са новим технолошким решењима као део једног ширег система, додатно указује на њихове проблеме из стратешке перспективе.

Предложени теоријски оквир инкорпорира и перспективе (начине гледања) на наслеђени систем и то развојну, оперативну, организациону и стратешку, као и перспективу (поглед у будућност) и даје свеобухватно сагледавање наслеђених система. Са сваком перспективом дефинисан је скуп проблема који могу да постоје у наслеђеном систему. У оквиру развојне перспективе документација није ажурна или не постоји, запослени који су радили на развоју система су напустили предузеће, новозапослени немају довољно знања и вештина потребних за одржавање система, нарушена је почетна архитектура система и постоји редуванса. Оперативна перспектива истиче да не постоји подршка произвођача хардвера и/или софтвера, као и проблеме интеграције система. У организационој перспективи са једне стране систем ограничава пословање, док са друге стране је вредан због подршке главним пословним активностима. Код стратешке перспективе наслеђен систем ограничава измену стратегије и ограничава будућу стратегију.

Уочавањем напред наведених проблема на конкретном наслеђеном систему који се повезују са одговарајућим перспективама, могуће је размотрити различите модернизационе стратегије и усмерити одабир на алтернативу која је најбоља за посматрани случај, што је показано на студији случаја ЈКП БВК, а такође је **допринос** тезе. Сагледавањем наслеђеног система из свих перспектива и проблема који су у вези са перспективама, елиминише се изоловано посматрање феномена чиме се смањује ризик да процес модернизације пропадне. Уколико се има у виду да се многа данашња пословања предузећа у потпуности ослањају на примену информационо-комуникационих технологија и информационих система, то значи да се опстанак предузећа директно повезује са успехом модернизационог процеса.

Предложена модернизациона стратегија која се може применити на ЈКП БВК, се може показати као адекватна за сва она предузећа која имају наслеђен систем написан у програмским језицима 4. генерације. Тако да је то још један **допринос** тезе.

Још један **допринос** дисертације је уочено одступање од предложеног теоријског оквира на примеру ЈКП БВК, где се уочавају проблеми који су у вези са стратешком перспективом наслеђених система у пракси, а нису део теоријских полазних основа из области информационих система и софтверског инжењерства, већ су део менаџмента. Наиме, проблем је повезаност улоге коју организациона јединица задужена за

информациони систем, најчешће ИТ сектор, има у предузећу и како се та улога перципира од стране топ менаџмента. Изостављањем ИТ менаџера из одлучивања у вези креирања ИТ стратегије предузећа која води до набавке софтверских решења који нису део модернизационе стратегије настале у ИТ сектору, делује се на стартешку перспективу наслеђених система, генерише се велики број проблема у свим осталим перспективама и доводи се у опасност постојећа модернизациона стратегија ИТ сектора. Такође поменуто одступање се не може генерализовати у потпуности на посматрани феномен, али би требало обратити пажњу, код спровођења поступака модернизације наслеђених система у јавном сектору. Како су привредни субјекти јавног сектора од великог друштвеног значаја за државу Србију, требало би размотрити додатне мере и правце деловања који би утицали на смањење поменуте појаве, што може бити један од праваца будућих истраживања.

9.2. Ограничења истраживања

Истраживањем је покривен домен пословања водоводних предузећа, јер се за њих спроводи анализа ланца вредности на случају ЈКП БВК. Теоријски оквир може послужити анализи наслеђених система водоводних предузећа. Да би се применио и у другим пословним доменима, потребно је спровести анализу ланца вредности жељеног домена, а затим утврдити који су то технологије које у будућности могу да допринесу побољшању активности у том ланцу. У том смислу предложени теоријски оквир се може модификовати да буде примењен и на друге домene пословања, тако што ће стратешка перспектива предузећа другог домена, бити допуњена потребним информацијама о могућности функционисања наслеђеног система са технологијама које активности ланца вредности чине оптималнијим. Дакле, са поменутом адаптацијом теоријски оквир је опште применљив.

Презентовано решење модернизационе стратегије може се показати погодно за наслеђене системе написане у програмском језику 4. генерације имајући у виду специфичност ових језика и окружења која се пре свега односе да они не могу да се користе у другим окружењима сем изворног, да су доменски специфични језици направљени за развој апликација за специфичан домен (нпр. пословни), да користе рапидан развој софтвера са циљем да што пре се дође до верзије софтверског решења које одговара захтевима, нефлексибилност програмског језика, везаност за произвођача софтвера и непознавање до краја начина рада самог програмског окружења.

Приликом одабира стратегије модернизације, потребно је спровести анализу портфолија и за сваку апликацију одредити технички квалитет (вредност) и пословну вредност. У теорији је познато да анализа портфолија има своја ограничења: тешко одређивање техничког квалитета и пословне вредности, делови система се понекад не могу изоловано посматрати као апликације или модули и недостатак будућих организационих промена. Из наведених разлога, свака организација треба да процени да ли ће користити анализу портфолија на презентовани начин или ће приступити другачијем виду евалуације пословне вредности и техничког квалитета апликација, нпр. коју предлажу други аутори (Ulrich, 2002). Који год приступ да искористити, предложени теоријски оквир може послужити у усмеравању одабира будуће стратегије модернизације, јер анализира проблеме система из неколико перспектива и даје потпунију слику о конкретном наслеђеном систему у пракси. Када се познаје тренутно стање система, и када је одређено жељено будуће стање, тада је могуће и дефинисати све потребне кораке које једна стратегија модернизације (може да се састоји из неколико других) треба да спроведе да систем преведе, тј. трансформише из тренутног у жељено стање

Такође на случају ЈКП БВК, презентован је поступак разматрања модернизационе стратегије за 3 апликације, да би се показао начин на који теоријски оквир може усмерити модернизацију и одабир погодне модернизационе стратегије, док у уопштем случају, потребно је проценити све апликације наслеђеног система.

9.3. Будућа истраживања

Будућа истраживања могу се кретати у неколико праваца. Први правац је да се теоријски оквир провери на другим доменима пословања у односу на водоводна предузећа, при томе очекујући да могу да постоје значајне разлике у стратешкој перспективи. Други правац би био да се на основу теоријског оквира, утврди метрика или метрике која би требало да оцене стање наслеђеног система у односу на његове елементе и да их квантитативно искаже. Након тога, могло би да се у односу на одређене вредности метрике и специфичности програмског окружења у којем је израђен наслеђени систем утврде погодне модернизационе стратегије.

Литература

- Adolph, W. S. (1996). Cash cow in the tar pit: reengineering a legacy system. *IEEE Software*, 13(3). <https://doi.org/10.1109/52.493019>
- Ahmad, A., Hussain, A., Flayyih, O. H., Abdulwahab, W., & Sabri, M. I. (2017). Utilizing WAMMI components to evaluate the usability of E-commerce website. *Journal of Telecommunication, Electronic and Computer Engineering*, 9(2–11), 139–143.
- Ahmet Mentis, S., & Turan, A. H. (2012). Assessing the usability of university websites: An empirical study on Namik Kemal University. *Turkish Online Journal of Educational Technology*, 11(3), 61–69.
- Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, 38(4), 393–422. [https://doi.org/https://doi.org/10.1016/S1389-1286\(01\)00302-4](https://doi.org/https://doi.org/10.1016/S1389-1286(01)00302-4)
- Al-Badareen, A. B., Selamat, M. H., Jabar, M. A., & Din, J. (2011). Software Quality Models □: A Comparative Study Software Quality Models □: A Comparative Study. In J. M. Zain (Ed.), *Communications in Computer and Information Science ICSECS 2011* (pp. 46–55). <https://doi.org/10.1007/978-3-642-22170-5>
- Albino, V., Berardi, U., & Dangelico, R. M. (2015). Smart Cities: Definitions, Dimensions, Performance, and Initiatives. *Journal of Urban Technology*, 22, 3–21. <https://doi.org/10.1080/10630732.2014.942092>
- Alderson, A., & Shah, H. (1999). Technical Opinion: Viewpoints on legacy systems. *Communications of the ACM*, 42(3), 115–116. <https://doi.org/10.1145/295685.295722>
- Althani, B., & Khaddaj, S. (2018). Systematic Review of Legacy System Migration. *Proceedings - 2017 16th International Symposium on Distributed Computing and Applications to Business, Engineering and Science, DCABES 2017, 2018-Septe*, 154–157. <https://doi.org/10.1109/DCABES.2017.41>
- Alvisi, S., Casellato, F., Franchini, M., Govoni, M., Luciani, C., Poltronieri, F., Riberto, G., Stefanelli, C., & Tortonesi, M. (2019). Wireless middleware solutions for smart water metering. *Sensors (Switzerland)*, 19(8), 1–25. <https://doi.org/10.3390/s19081853>
- Amazon. (n.d.). *What is Apache Kafka?* Retrieved June 1, 2020, from <https://aws.amazon.com/msk/what-is-kafka/>

- Anquetil, N., Oliveira, K. M., Dos Santos, A. G. M., Da Silva, P. C. S., De Araujo, L. C., & Vieira, S. D. C. F. (2005). Software re-documentation process and tool. *Proceedings of the CAiSE'05 Forum*, 161(January), 95–100.
- Antidot. (n.d.-a). *Nabavka softvera dell i maxbill*. Retrieved June 15, 2020, from <https://www.anti.media/uploads/files/01-nabavka-softvera-dell-i-max-bill-informacija.pdf>
- Antidot. (n.d.-b). *Ugovor o softverskoj licenci*. Retrieved June 15, 2020, from <https://www.anti.media/uploads/files/02-dopis-advokata-ugovor-o-softverskoj-licenci-od-15.10.2012..pdf>
- Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R. H., Konwinski, A., Lee, G., Patterson, D. A., Rabkin, A., Stoica, I., & Zaharia, M. (2009). *Above the Clouds: A Berkeley View of Cloud Computing*.
- Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R. H., Konwinski, A., Lee, G., Patterson, D. A., Rabkin, A., Stoica, I., & Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*, 53(4).
<https://doi.org/https://doi.org/10.1145/1721654.1721672>
- Arnold, R. S. (1989). Software Restructuring. *Proceedings of the IEEE*, 77(4), 607–617.
<https://doi.org/10.1109/5.24146>
- Aversano, L., & Tortorella, M. (2004). An assessment strategy for identifying legacy system evolution requirements in ebusiness context. *Journal of Software Maintenance and Evolution*, 16(4–5), 255–276. <https://doi.org/10.1002/smr.296>
- Aveson, D., & Fitzgerald, G. (2006). Methodologies for developing information systems: A historical perspective. In D. Avison, S. Elliot, J. Krogstie, & J. Pries-Heje (Eds.), *The Past and Future of Information Systems: 1976–2006 and Beyond* (The Past a, pp. 27–38). Springer. https://doi.org/https://doi.org/10.1007/978-0-387-34732-5_3
- Avison, D. E., & Shah, H. U. (1992). From third generation to fourth generation applications development: A case study. *Information Technology & People*, 6(4), 233–248.
<https://doi.org/10.1108/EUM0000000003552>
- Avison, D., & Elliot, S. (2006). Scoping the Discipline of Information Systems. In J. L. King & K. Lyytinen (Eds.), *Information Systems: The State of the Field* (pp. 3–18). John Wiley & Sons Ltd.
- Ayanso, A. (2015). Business and Technology Trends in Social CRM. In H.-R. Kaufmann (Ed.),

Handbook of Research on Managing and Influencing Consumer Behavior. IGI Global.
<https://doi.org/DOI: 10.4018/978-1-4666-6547-7.ch013>

- B92. (n.d.). *I nasleđeni softver štiti tajkune?* Retrieved February 19, 2020, from
https://www.b92.net/biz/vesti/srbija.php?yyyy=2014&mm=10&dd=30&nav_id=917809
- Babić, B., & Djukić, A. (2011). Estimation of Water Balance and Water Losses in Water Utilities-Experiences from the Belgrade Waterworks. *Water Research and Management*, 1(4), 27–34.
- Barnum, C. M. (2010). *Usability Testing Essentials: Ready, Set...Test!* (1st ed.). Morgan Kaufmann.
- Bečejski-Vujaklija, D. (2008). *Uvod u informacione sisteme: priručnik za pripremu prijemnog ispita za Master studije*. Fakultet organizacionih nauka.
- Beedle, M., Cockburn, A., Fowler, M., Martin, R. C., Sutherland, J., Schwaber, K., & Cunningham, W. (n.d.). *Principles behind the Agile Manifesto*. Retrieved March 27, 2020, from <https://agilemanifesto.org/principles.html>
- Behrens, B. C., & Levary, R. R. (1998). Practical Legal Aspects of Software Reverse Engineering. *Communications of the ACM*, 41(2), 27–29.
<https://doi.org/10.1145/269012.269017>
- Bennett, K. H. (1995). Legacy systems: coping with success. *IEEE Software*, 12(1).
<https://doi.org/10.1109/52.363157>
- Bergstrom, J. R. (n.d.). *Moderating Usability Tests*. Retrieved May 17, 2020, from
<https://www.usability.gov/get-involved/blog/2013/04/moderating-usability-tests.html>
- Berry, J. W., Hart, W. E., Phillips, C. A., Uber, J. G., & Walski, T. M. (2005). Water Quality Sensor Placement in Water Networks with Budget Constraints. *World Water and Environmental Resources Congress 2005: Impacts of Global Climate Change*.
- Bevan, N., Carter, J., Earthy, J., Geis, T., & Harker, S. (2016). Human-Computer Interaction. Theory, Design, Development and Practice. In M. Kurosu (Ed.), *Proceedings, Part I, of the 18th International Conference on Human-Computer Interaction. Theory, Design, Development and Practice* (Vol. 9731, pp. 268–278). Springer-Verlag.
<https://doi.org/10.1007/978-3-319-39510-4>
- Biggerstaff, T. J. (1989). Design Recovery for Maintenance and Reuse. *Computer*, 22(7), 36–49. <https://doi.org/10.1109/2.30731>

- Birgün, S., & Altan, Z. (2019). A Managerial Perspective for the Software Development Process: Achieving Software Product Quality by the Theory of Constraints. In *Agile Approaches for Successfully Managing and Executing Projects in the Fourth Industrial Revolution* (pp. 243–266). IGI Global. <https://doi.org/DOI: 10.4018/978-1-5225-7865-9.ch013>
- Bisbal, Jesús, Lawless, D., Wu, B., & Grimson, J. (1999). Legacy information systems: issues and directions. *IEEE Software*, 16(5), 103–111. <https://doi.org/10.1109/52.795108>
- Bisbal, Jesus, Lawless, D., Wu, B., Grimson, J., Wade, V., Richardson, R., & O'Sullivan, D. (1997). Overview of legacy information system migration. *Proceedings of the Asia-Pacific Software Engineering Conference and International Computer Science Conference, APSEC and ICSC*, 529–530. <https://doi.org/10.1109/apsec.1997.640219>
- Bjelajac, Ž. Đ., & Vesić, S. L. (2020). Security of information systems. *Pravo - Teorija i Praksa*, 37(2), 63–76. <https://doi.org/10.5937/ptp2002063b>
- Blandford, A., & Buchanan, G. (2002). Usability for digital libraries. *JCDL '02: Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries*.
- Bocij, P., Greasley, A., & Hickie, S. (2015). *Business Information Systems: Technology, Development and Management for the E-Business* (5th ed.). Pearson Education Limited.
- Boehm, B., Brown, J. ., Kapar, H., Lipow, M., McLeod, G., & Merrit, M. (1978). *Characteristics of Software Quality*. North Holland Publishing.
- Bojić, D., & Velašević, D. (1999). Softversko reinženjerstvo - pregled stanja. *Infoteh*.
- Bonwich, B. (n.d.). *5 ESSENTIAL TIPS TO SIMPLIFIED KNOWLEDGE MANAGEMENT*. Retrieved April 17, 2020, from <https://www.clydebankmedia.com/blog/technology/information-technology/knowledge-management>
- Bourgeois, D. (2014). *Information Systems for Business and Beyond*. Saylor Foundation. <https://resources.saylor.org/wwwresources/archived/site/textbooks/Information Systems for Business and Beyond.pdf>
- Boutin, C. (n.d.). *A Beginner's Guide to Kanban Methodology*. Retrieved March 30, 2020, from <https://unito.io/blog/beginners-guide-to-kanban-methodology/>
- Brinck, T., Gergle, D., & Wood, S. D. (2002). *Usability for the Web: Designing Web Sites that Work* (1st ed.). Morgan Kaufmann.

- Brodie, M. L., & Stonebraker, M. (1993). *DARWIN: On the Incremental Migration of Legacy Information Systems*.
- Brodie, M., & Stonebraker, M. (1995). *Migrating legacy systems*. Morgan Kaufmann.
- Brooke, C. (1994). Information Technology and the Quality Gap. *Employee Relations*, 16(4).
- Brooke, C. (2002). A framework for evaluating Legacy IS. In D. Remenyi & A. Brown (Eds.), *Make or Break Issues in IT Management: A Guide to 21st Century Effectiveness* (1st ed., pp. 108–124). Butterworth-Heinemann. <https://doi.org/10.1017/CBO9781107415324.004>
- Brooke, C., & Ramage, M. (2001). Organisational scenarios and legacy systems. *International Journal of Information Management*, 21(5), 365–384. [https://doi.org/10.1016/S0268-4012\(01\)00023-8](https://doi.org/10.1016/S0268-4012(01)00023-8)
- Butler, K. A. (1985). Connecting Theory and Practice: A Case Study of Achieving Usability Goals. In *Proceedings of CHI 85*, 85–88.
- Byte-notes. (n.d.). *Generation Of Programming Languages*. Retrieved June 30, 2020, from <http://byte-notes.com/generation-of-programming-languages/>
- Cassel, D. (n.d.). *COBOL Is Everywhere. Who Will Maintain It?* Retrieved May 30, 2020, from <https://thenewstack.io/cobol-everywhere-will-maintain/>
- Cenelec. (n.d.). *The Importance of Standards*. Retrieved April 13, 2020, from <https://www.cenelec.eu/research/tools/ImportanceENs/Pages/default.aspx>
- Chan, S. S., & Fang, X. (2009). Interface Design Issues for Mobile Commerce. In M. Khosrow-Pour (Ed.), *Encyclopedia of Information Science and Technology* (2nd ed., pp. 2153–2158). Idea Group Reference.
- Chan, T. K., Chin, C. S., & Zhong, X. (2018). Review of Current Technologies and Proposed Intelligent Methodologies for Water Distributed Network Leakage Detection. *IEEE Access*, 6, 78846–78867.
- Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly Detection: A Survey. *ACM Computing Surveys (CSUR)*, 41(3). <https://doi.org/10.1145/1541880.1541882>
- Chikofsky, E. J., & Cross, J. H. (1990). Reverse engineering and design recovery: a taxonomy. *IEEE Software*, 7(1), 13–17. <https://doi.org/10.1109/52.43044>
- Chun, M., & Mooney, J. (2009). CIO roles and responsibilities: Twenty-five years of evolution and change. *Information Systems and Technology Management*, 46(6), 323–334.

<https://doi.org/https://doi.org/10.1016/j.im.2009.05.005>

- Churchill, E., Bowser, A., & Preece, J. (n.d.). *Teaching and Learning Human-Computer Interaction*. Retrieved April 20, 2020, from <https://interactions.acm.org/archive/view/march-april-2013/teaching-and-learning-human-computer-interaction>
- Cocchia, A. (2014). Smart and Digital City: A Systematic Literature Review. In Renata Paola Dameri & Camille Rosenthal-Sabroux (Eds.), *Smart City How to Create Public and Economic Value with High Technology in Urban Space* (Progress i, pp. 13–43). Springer International Publishing. https://doi.org/https://doi.org/10.1007/978-3-319-06160-3_2
- Colosimo, M., Lucia, A. De, Tortora, G., & Scanniello, G. (2007). *Introducing Legacy System Migration Technologies in an Academic Context: a Controlled Experiment*. https://pdfs.semanticscholar.org/26e6/da86a9a1707e40ca4d6d4c46e3671d970be0.pdf?_ga=2.3993394.348807532.1599855710-2035203312.1585030258
- Conklin, J. (1987). Hypertext: An Introduction and Survey. *Computer*, 20(9), 17–41. <https://doi.org/10.1109/MC.1987.1663693>
- Constantine, L. L., & Lockwood, L. A. D. (1999). *Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design*. Addison-Wesley Professional.
- Craemer, K. De, & Deconinck, G. (2010). Analysis of State-of-the-art Smart Metering Communication Standards. *Proceedings of the 5th Young Researchers Symposium*, 1–6. <https://doi.org/10.1109/TSG.2012.2218834>
- Crotty, J., & Horrocks, I. (2017). Managing legacy system costs: A case study of a meta-assessment model to identify solutions in a large financial services company. *Applied Computing and Informatics*, 13(2), 175–183. <https://doi.org/10.1016/j.aci.2016.12.001>
- Čudanov, M. (2007). *Projektovanje organizacije i IKT*. Zadužbina Andrejević.
- D’Ambra, J., Wilson, C. S., & Akter, S. (2014). Evaluating the Perceived Fit Between E-Books and Academic Tasks. In *Encyclopedia of Information Science and Technology* (3rd ed., Issue January 2015, pp. 2298–2307). IGI Global. <https://doi.org/10.4018/978-1-4666-5888-2.ch223>
- Dedeke, A. (2012). Improving Legacy-System Sustainability: A Systematic Approach. *IT Professional*, 14(1), 38–44. <https://doi.org/10.1109/MITP.2012.10>
- Di Mauro, A., Di Nardo, A., Santonastaso, G. F., & Venticinque, S. (2019). An IoT system for

- monitoring and data collection of residential water end-use consumption. *2019 28th International Conference on Computer Communication and Networks (ICCCN)*, 1–6.
<https://doi.org/10.1109/ICCCN.2019.8847120>
- Difallah, D. E., Cudre-Mauroux, P., & McKenna, S. A. (2013). Scalable anomaly detection for smart city infrastructure networks. *IEEE Internet Computing*, *17*(6), 39–47.
<https://doi.org/10.1109/MIC.2013.84>
- Dix, A. (2009). Human-Computer Interaction. In L. Liu & M. T. Özsu (Eds.), *Encyclopedia of Database Systems* (pp. 1327–1332). Springer US.
- Dix, A., Finlay, J., Abowd, G. D., & Beale, R. (2004). *Human-Computer Interaction* (3rd ed.). Pearson Education Limited.
- Duarte, E. F., & Baranauskas, M. C. C. (2016). Revisiting the three HCI waves: A preliminary discussion on philosophy of science and research paradigms. *IHC '16*.
<https://doi.org/10.1145/3033701.3033740>
- Dubey, S. K., & Rana, A. (2010). Analytical Roadmap to Usability Definitions and Decompositions. *International Journal of Engineering Science and Technology*, *2*(9), 4723–4729.
- Dunko, G., Misra, J., Robertson, J., & Snyder, T. (2017). *A Reference Guide to the Internet of Things*.
- Đurđević, M. (2009). *Komunalna infrastruktura* (2. izdanje). Visoka građevinsko-geodetska škola.
- Environment Engineering Group. (n.d.). *Održivi razvoj*. Retrieved May 14, 2020, from <https://www.activity4sustainability.org/odrzivi-razvoj/>
- Esaki, K. (2013). System Quality Requirement and Evaluation Importance of application of the ISO/IEC 25000 series. *Communications in Computer and Information Science*, *2*(2), 52–59.
- Farley, M. (2001). Leakage Management and Control: A BEST PRACTICE TRAINING MANUAL. In *World Health Organization*.
- Fisher, G. E. (1986). *A Functional Model for Fourth Generation Languages*.
- Forgeat, J. (n.d.). *Data processing architectures - Lambda and Kappa*. Retrieved June 30, 2020, from <https://www.ericsson.com/en/blog/2015/11/data-processing-architectures--lambda->

and-kappa

- Frauentorfer, R., & Liemberger, R. (2010). *The Issues and Challenges of Reducing Non-Revenue Water*. Asian Development Bank.
- Fujitsu. (n.d.). *Fujitsu Develops Technology for Low-Cost Detection of Potential Sewer System Overflows*. Retrieved June 30, 2020, from <https://www.fujitsu.com/global/about/resources/news/press-releases/2015/0210-03.html>
- Galorath, D. D. (2014). *Software Total Ownership Costs: Development Is Only Job One*. https://galorath-old.lovestheir.site/wp-content/uploads/2014/08/software_total_ownership_costs-development_is_only_job_one.pdf
- Gannon, K. (n.d.). *Managing Resources & Capability Building to deliver Customer Benefits*. Retrieved June 10, 2020, from <https://slideplayer.com/slide/12149191/>
- Gartner. (n.d.). *Legacy Application Or System*. Gartner Glossary. Retrieved May 5, 2020, from <https://www.gartner.com/en/information-technology/glossary/legacy-application-or-system>
- GeeksforGeeks. (n.d.). *Difference between Forward Engineering and Reverse Engineering*. Retrieved June 11, 2020, from <https://www.geeksforgeeks.org/difference-between-forward-engineering-and-reverse-engineering/>
- Goodwin, N. C. (1987). Functionality and usability. *Communications of the ACM*, 30(3), 229–233.
- Gorton, I. (2011). Essential Software Architecture. In *Essential Software Architecture* (2nd ed.). Springer. <https://doi.org/10.1007/978-3-642-19176-3>
- Grad Beograd - Sekretarijat za informisanje. (2005). Odluka o prečišćavanju i distribuciji vode. *Službeni List Grada Beograda*, 23. https://www.bvk.rs/wp-content/uploads/2020/01/Odlukao-odvodjenju-i-preciscavanju_2019-min.pdf
- Grad Beograd - Sekretarijat za informisanje. (2010). Odluka o odvođenju i prečišćavanju atmosferskih i otpadnih voda na teritoriji grada Beograda. *Službeni List Grada Beograda*, 6. https://www.bvk.rs/wp-content/uploads/2020/01/Odlukao-odvodjenju-i-preciscavanju_2019-min.pdf
- Grubb, P., & Takang, A. (2003). *Software Maintenance: Concepts and Practice* (2nd ed.). World Scientific Publishing Co. Pte. Ltd.

- Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7), 1645–1660. <https://doi.org/10.1016/j.future.2013.01.010>
- Gumoz. (n.d.). *reflect and unpack, the use of .net reflector and some unpacking techniques*. Retrieved June 5, 2020, from reflect and unpack, the use of .net reflector and some unpacking techniques
- Gupta, D., Ahlawat, A. K., & Sagar, K. (2017). Usability Prediction & Ranking of SDLC Models Using Fuzzy Hierarchical Usability Model. *Open Engineering*, 7(1), 161–168. <https://doi.org/10.1515/eng-2017-0021>
- Gupta, S. (2015). A Comparative study of Usability Evaluation Methods. *International Journal of Computer Trends and Technology*, 22(3), 103–106. <https://doi.org/10.14445/22312803/ijctt-v22p121>
- Hainaut, J.-L., Henrard, J., Roland, D., Hick, J.-M., & Englebert, V. (2010). Database Reverse Engineering. In V. Ferraggine, J. Doorn, & L. Rivero (Eds.), *Handbook of Research on Innovations in Database Technologies and Applications: Current and Future Trends*. Information Science Reference. <https://doi.org/10.4018/978-1-60566-242-8>
- Halton, C. (n.d.). *What Was Y2K?* Retrieved May 27, 2020, from <https://www.investopedia.com/terms/y/y2k.asp>
- Hanby, J. (n.d.). *Software Maintenance: Understanding and Estimating Costs*. Retrieved May 21, 2020, from <http://blog.lookfar.com/blog/2016/10/21/software-maintenance-understanding-and-estimating-costs/>
- Harrison, R., Flood, D., & Duce, D. (2013). Usability of mobile applications: literature review and rationale for a new usability model. *Journal of Interaction Science*, 1(1), 1–16. <https://doi.org/10.1186/2194-0827-1-1>
- Hix, D., & Hartson, H. R. (1993). *Developing user interfaces: Ensuring usability through product & process* (1st ed.). Wiley.
- Hollands, R. G. (2008). Will the real smart city please stand up? Intelligent, progressive or entrepreneurial? *City*, 12(3), 303–320. <https://doi.org/10.1080/13604810802479126>
- Holmes, B., & Gardner, J. (2006). *e - Learning: Concepts and Practice*. SAGE Publications.
- IBM Cloud Education. (n.d.). *CAP Theorem*. Retrieved June 2, 2020, from <https://www.ibm.com/cloud/learn/cap-theorem>

- IEEE. (2014). *Guide to the Software Engineering Body of Knowledge, Version 3.0* (P. Bourque & R. E. Fairley (eds.)). IEEE Computer Society. www.swebok.org
- Interaction-Design-Foundation. (n.d.-a). *Heuristic Evaluation*. Retrieved May 12, 2020, from <https://www.interaction-design.org/literature/topics/heuristic-evaluation>
- Interaction-Design-Foundation. (n.d.-b). *How to Conduct a Cognitive Walkthrough*. Retrieved May 5, 2020, from <https://www.interaction-design.org/literature/article/how-to-conduct-a-cognitive-walkthrough>
- Interaction-Design-Foundation. (n.d.-c). *Usability Engineering*. Retrieved April 30, 2020, from <https://www.interaction-design.org/literature/topics/usability-engineering>
- International Organisation for Standardisation (ISO). (2018). Ergonomics of human-system interaction - Part 11: Usability: Definitions and Concepts. In *ISO 9241-11:2018(E)*.
- Isaias, P., & Issa, T. (2015). *High Level Models and Methodologies for Information Systems*. Springer Science+Business Media. <https://doi.org/10.1007/978-1-4614-9254-2>
- ISO/IEC. (n.d.). *Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - Common Industry Format (CIF) for Usability - Evaluation Report*. Retrieved May 1, 2020, from https://www.vde-verlag.de/iec-normen/preview-pdf/info_isoiec25066%7Bed1.0%7Den.pdf
- ISO/IEC JTC 1/SC 7. (n.d.-a). *ISO/IEC/IEEE 12207:2017 - Systems and software engineering - Software life cycle processes*. Retrieved April 8, 2020, from <https://www.iso.org/standard/63712.html>
- ISO/IEC JTC 1/SC 7. (n.d.-b). *ISO/IEC 12207:2008 - Systems and software engineering - Software life cycle processes*. Retrieved April 7, 2020, from <https://www.iso.org/standard/43447.html>
- ISO/IEC JTC 1/SC 7. (n.d.-c). *ISO/IEC 14764:2006(en): Software Engineering Software Life Cycle Processes - Maintenance*. Retrieved May 19, 2020, from <https://www.iso.org/obp/ui/#iso:std:iso-iec:14764:ed-2:v1:en>
- ISO/IEC JTC 1/SC 7. (n.d.-d). *ISO/IEC 25022:2016 Systems and software engineering — Systems and software quality requirements and evaluation (SQuaRE) — Measurement of quality in use*. Retrieved May 1, 2020, from <https://www.iso.org/standard/35746.html>
- ISO/IEC JTC 1/SC 7. (n.d.-e). *SQuaRE extension (ISO/IEC 25050 to ISO/IEC 25099)*. Retrieved April 10, 2020, from <https://iso25000.com/index.php/en/iso-25000->

standards?limit=4&start=4

ISO/IEC JTC 1/SC 7. (n.d.-f). *The ISO/IEC 25000 series of standards*. Retrieved April 9, 2020, from <https://iso25000.com/index.php/en/iso-25000-standards>

ISS. (2014). *Ekonomске koristi od primene standarda*.

Ivory, M. Y., & Hearst, M. A. (2001). The state of the art in automating usability evaluation of user interfaces. *ACM Computing Surveys*, 33(4), 470–516.
<https://doi.org/10.1145/503112.503114>

Jadeja, Y., & Modi, K. (2012). Cloud Computing - Concepts, Architecture and Challenges. *2012 International Conference on Computing, Electronics and Electrical Technologies, ICCEET 2012*, 877–880. <https://doi.org/10.1109/ICCEET.2012.6203873>

Jagadish, H. V., Gehrke, J., Labrinidis, A., Papakonstantinou, Y., Patel, J. M., Ramakrishnan, R., & Shahabi, C. (2014). Big data and its technical challenges. *Communications of the ACM*, 57(7), 86–94.

javaTpoint. (n.d.). *Difference between Artificial intelligence and Machine learning*. Retrieved June 30, 2020, from <https://www.javatpoint.com/difference-between-artificial-intelligence-and-machine-learning>

JKPBVK. (n.d.). *Osnovni podaci*. Retrieved June 20, 2020, from <https://www.bvk.rs/osnovni-podaci/>

JKPBVK. (2019). *Program poslovanja za 2020. godinu*. <https://www.bvk.rs/wp-content/uploads/2020/01/program-poslovanja-2020-god-min.pdf>

JLC/CRM. (n.d.). *Reengineering Definitions*. Retrieved June 3, 2020, from <https://www2.informatik.hu-berlin.de/swt/projekt98/lehre/define.htm>

Jovanović, B. (2007). Problemi softverskog nasleđa u našem okruženju. In M. Čangalović & M. Suknović (Eds.), *SYM-OP-IS 2007* (pp. 311–314). Fakultet organizacionih nauka.

Jovanović, M. (2010). Upravljanje ugovorima za softver oslanjanjem na standarde – način dostizanja kvaliteta i sprečavanja zloupotreba. *Savetovanje o Zloupotrebi Informacionih Tehnologija i Zaštiti ZITEH '10*, 1–13.

kanbanize.com. (n.d.). *What Is Kanban? Definition and Brief Introduction*. Retrieved March 30, 2020, from <https://kanbanize.com/kanban-resources/getting-started/what-is-kanban>

Karat, J. (1997). User-Centered Software Evaluation Methodologiess. In M. Helander, T. K.

- Landauer, & P. Prabhu (Eds.), *Handbook of Human-Computer Interaction* (2nd ed., pp. 689–704). Elsevier Science.
- Kenttälä, V., Rousi, R., & Kankaanranta, M. (2017). Towards the Learning Experience Technology Usability Framework. In T. Kidd & L. R. Morris (Eds.), *Handbook of Research on Instructional Systems and Educational Technology* (pp. 128–140). IGI Global.
- Khadka, R., Batlajery, B. V., Saeidi, A. M., Jansen, S., & Hage, J. (2014). How do professionals perceive legacy systems and software modernization? *ICSE 2014: Proceedings of the 36th International Conference on Software Engineering*, 36–47.
<https://doi.org/10.1145/2568225.2568318>
- Khosrow-Pour, M. (2007). Dictionary of Information Science and Technology. In *British Journal of Educational Technology* (Issue 1). Idea Group Reference.
<https://doi.org/10.4018/978-1-59904-385-2>
- Khusidman, V., & Ulrich, W. (2007). *Architecture-driven modernization: Transforming the enterprise*. <https://www.omg.org/cgi-bin/doc?admtf/07-12-01.pdf>
- Kim, Y.-G. (1997). Improving Legacy Systems Maintainability. *Improving Legacy Systems Maintainability*, 14(1).
- Kleppmann, M. (2016). *Designing Data-Intensive Applications* (1st ed.). O'Reilly Media.
- Knoche, H., & Hasselbring, W. (2018). Using Microservices for Legacy Software Modernization. *IEEE Software*, 35(3), 44–49. <https://doi.org/10.1109/MS.2018.2141035>
- Koskinen, J. (2015). *Software Maintenance Costs*.
<https://ocw.unican.es/pluginfile.php/1408/course/section/1805/SMCOSTS.pdf>
- Kurose, J., & Ross, K. (2009). *Umrežavanje računara: Od vrha ka dnu* (4. izd.). Računarski fakultet.
- LabIS. (2000a). *Prošireni model objekti-veze (skripta)*.
http://download.tutoriali.org/Tutorials/Baze_podataka/prosireni_model_objekti-veze.pdf
- LabIS. (2000b). *Strukturna sistemska analiza (skripta)*.
https://www.puskice.org/download/upravljanje_kvalitetom_dokumentacije/SSA_Skripte.pdf
- Laudon, K. C., & Laudon, J. P. (1999). *Management Information Systems: Organization and*

Technology in the Networked Enterprise. Pearson.

Laudon, K. C., & Laudon, J. P. (2017). *Essentials of Management Information Systems* (12th ed.). Pearson.

Laudon, K. C., & Laudon, J. P. (2018). *Management Information Systems: Managing the Digital Firm* (15th-glo ed.).

Law Boutique Lexidy. (n.d.). *Why software development contracts are important*. Retrieved April 15, 2020, from <https://www.lexidy.com/why-software-development-contracts-are-important/>

Lazarević, B., Jovanović, V., & Vučković, M. (1985). *Projektovanje informacionih sistema, I deo*. Naučna knjiga.

Lazarević, B., Marjanović, Z., Aničić, N., & Babarogić, S. (2006). *Baza podataka* (3rd ed.). Fakultet organizacionih nauka.

Lazović, V., & Bulatović, B. (2019). *Informacioni sistemi u funkciji revizije - priručnik*. <http://www.dri.co.me/1/doc/Prirucnik - Informacioni sistem u funkciji revizije.pdf>

Lehman, M. M. (1980). On understanding laws, evolution, and conservation in the large-program life cycle. *The Journal of Systems and Software*, 1, 213–221.

Lehman, M. M. (1996). Laws of software evolution revisited. In C. Montangero (Ed.), *EWSPT '96: Proceedings of the 5th European Workshop on Software Process Technology*. Springer. <https://doi.org/10.1007/BFb0017737>

Lehman, Meir M. (1980). *Programs, life cycles, and laws of software evolution*. 68(9), 1060–1076. <https://doi.org/10.1109/PROC.1980.11805>

Leonard, A. C., & Van Zyl, D. H. (2018). Social Issues in IT Project Teams. In M. Khosrow-Pour (Ed.), *Encyclopedia of Information Science and Technology* (4th ed., pp. 777–787). IGI Global.

Lewis, J. R. (1995). IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction*, 7(1), 57–78. <https://doi.org/https://doi.org/10.1080/10447319509526110>

Lientz, B. P., & Swanson, E. B. (1981). Problems in application software maintenance. *Communications of the ACM*, 24(11). <https://doi.org/https://doi.org/10.1145/358790.358796>

- Light, B. (2003). An Alternative Theory of Legacy Information Systems. *ECIS 2003 Proceedings*.
- Linthicum, D. (1994). 4GLs: No easy answers. *Object Magazine*, 4(5), 38–42.
- Liu, K., Alderson, A., Sharp, B., Shah, H., & Dix, A. (1998). Using Semiotic Techniques to Derive Requirements from Legacy Systems. *First SEBPC Legacy Workshop*.
- M. Abdelaziz, T., M. Maatuk, A., & Rajab, F. (2016). An Approach to Improvement The Usability in Software Products. *International Journal of Software Engineering & Applications*, 7(2), 11–18. <https://doi.org/10.5121/ijsea.2016.7202>
- MacDonald, A., Russell, D., & Atchison, B. (2005). Model-driven development within a legacy system: An industry experience report. *ASWEC '05: Proceedings of the 2005 Australian Conference on Software Engineering*, 14–22. <https://doi.org/10.1109/ASWEC.2005.32>
- Maguire, M. (2001). Context of use within usability activities. *International Journal of Human Computer Studies*, 55(4), 453–483. <https://doi.org/10.1006/ijhc.2001.0486>
- Majthoub, M., Qutqui, M. H., & Odeh, Y. (2018). Software Re-engineering: An Overview. *2018 8th International Conference on Computer Science and Information Technology, CSIT 2018*, 266–270. <https://doi.org/10.1109/CSIT.2018.8486173>
- Man, P. (2009). *Uvod u statistiku*. Centar za izdavačku delatnost Ekonomskog fakulteta.
- Manogaran, G., Thota, C., Lopez, D., Vijayakumar, V., Abbas, K. M., & Sundarsekar. Revathi. (2017). Big Data Knowledge System in Healthcare. In Chintan Bhatt, Nilanjan Dey, & Amira S. Ashour (Eds.), *Internet of Things and Big Data Technologies for Next Generation Healthcare*. Springer.
- Marakas, G., & O'Brien, J. (2010). *Management Information Systems* (10th ed.). McGraw-Hill Education.
- Marakas, G., & O'Brien, J. (2012). Introduction to Information Systems. In *Introduction to Information Systems* (16th ed.). McGraw-Hill/Irwin. <https://doi.org/10.1007/978-1-137-10325-3>
- Marković, A. (2013). *Modeliranje i simulacija*. <http://sipo.fon.bg.ac.rs/wp-content/uploads/2013/08/Modeliranje-i-simulacija-2.pdf>
- Marković, G., & Dukić, M. (2009). Bežične senzorske mreže, I deo: Osnovna arhitektura, karakteristike i primene. *Telekomunikacije*, 3.

http://www.telekomunikacije.rs/arhiva_brojeva/treci_broj/mr_goran_b_markovic,_prof_dr_miroslav_l_dukic:_bedjicne_senzorske_mredje,_i_deo:_osnovna_arhitektura,_karakteristike_i_primene.169.html

- McCall, James, Herndon, M., & Osborne, W. M. (1985). *Software Maintenance Management*.
- McCall, Jim, Richards, P., & Walters, G. (1977). *Factors in Software Quality: Concept and Definitions of Software Quality*.
- Mell, P., & Grance, T. (2011). *The NIST Definition of Cloud Computing*.
<https://doi.org/10.6028/NIST.SP.800-145>
- Meng, F., Qu, Z., & Guo, X. (2013). Refactoring Model of Legacy Software in Smart Grid based on Cloned Codes Detection. *International Journal of Computer Science Issues*, 10(1), 296–303.
- Mens, T. (2008). Introduction and Roadmap: History and Challenges of Software Evolution. In T. Mens & S. Demeyer (Eds.), *Software Evolution* (pp. 1–11). Springer-Verlag.
<https://doi.org/10.1007/978-3-540-76440-3>
- Miladinović, T. (2019). *Učenje na daljinu, E-učenje*. <https://vts.edu.rs/wp-content/uploads/2019/04/e-ucenje-1.pdf>
- Mitleton-Kelly, E. (2006). IT Legacy Systems: Enabling Environments That Reduce the Legacy Problem: A Complexity Perspective. In N. Madhavji, J. Fernandez-Ramil, & D. Perry (Eds.), *Software Evolution and Feedback: Theory and Practice* (1st ed.). John Wiley & Sons Ltd.
- Musa, S. (2016). Smart Cities - A Roadmap for Development. *Journal of Telecommunications System & Management*, 5(3). <https://doi.org/10.4172/2167-0919.1000144>
- Nassif, R., & Mitchusson, D. (1993). Issues and approaches for migration/cohabitation between legacy and new systems. *SIGMOD '93: Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*.
- National Association of State Chief Information Officers (NASCIO). (2008). *Digital States at Risk! Modernizing Legacy Systems*. <https://www.nascio.org/wp-content/uploads/2019/11/NASCIO-DigitalStatesAtRisk.pdf>
- NESSI. (2012). *Big Data: A New World of Opportunities, NESSI White Paper*.
- Newcomb, P. H. (2005). *Abstract Syntax Tree Metamodel Standard ASTM Tutorial 1.0*.

- Nielsen, J. (1993). *Usability Engineering*. Morgan Kaufmann.
- Nielsen, J. (2012). *Usability 101: Introduction to Usability*.
<https://www.nngroup.com/articles/usability-101-introduction-to-usability/>
- Nielsen, J. (1994). Enhancing the explanatory power of usability heuristics. *Conference on Human Factors in Computing Systems - Proceedings*, 152–158.
<https://doi.org/10.1145/191666.191729>
- Nielsen, J., & Molich, R. (1990). Heuristic evaluation of user interfaces. *Conference on Human Factors in Computing Systems - Proceedings*, 249–256.
<https://doi.org/10.1145/97243.97281>
- Nosek, J. T., & Palvia, P. (1990). Software maintenance management: Changes in the last decade. *Journal of Software Maintenance: Research and Practice*, 3, 157–174.
- Nowakowski, W., Śmiałek, M., Ambroziewicz, A., Jarzębowski, N., & Straszak, T. (2012). Recovery and Migration of Application Logic From Legacy Systems. *Computer Science*, 13(4). <https://doi.org/10.7494/csci.2012.13.4.53>
- O’Connell, T. A., & Murphy, E. D. (2009). Usability Engineering of User-Centered Web Sites. In M. Khosrow-Pour (Ed.), *Encyclopedia of Information Science and Technology* (2nd ed., pp. 3890–3896). Idea Group Reference.
- OECD. (2005). IT infrastructure: use of learning management system (LMS) and other applications. In *E-learning in Tertiary Education: Where Do We Stand?* Organization for Economic.
- Olivé, A. (2007). *Conceptual Modeling of Information Systems*. Springer-Verlag.
- Olsina, L., & Rossi, G. (2002). Measuring Web application quality with WebQEM. *IEEE Multimedia*, 9(4), 20–29. <https://doi.org/10.1109/MMUL.2002.1041945>
- OMG-ADMTF. (n.d.). *Architecture-Driven Modernization*. Retrieved June 20, 2020, from <https://www.omg.org/adm/>
- Orfali, R., Harkey, D., & Edwards, J. (1995). *The Essential Distributed Objects Survival Guide*. Wiley.
- Osborne, W. M. (1985). Executive Guide to Software Maintenance. In *NBS Special Publication 500-130*.
- Oztoprak, A., & Erbug, C. (2006). Field versus laboratory usability testing: A first Comparison.

- In D. de Waard, K. A. Brookhuis, & A. Toffetti (Eds.), *Developments in Human Factors in Transportation, Design, and Evaluation* (pp. 205–212). Shaker Publishing.
<https://doi.org/10.1145/1088463.1088482>
- Pal, G., Li, G., & Atkinson, K. (2018). Multi-agent big-data lambda architecture model for E-commerce analytics. *Data*, 3(4), 1–15. <https://doi.org/10.3390/data3040058>
- Pallant, J. (2017). *SPSS: priručnik za preživljavanje, prevod 6. izdanja*. Mikro knjiga.
- Papadokostaki, K., Mastorakis, G., Panagiotakis, S., Mavromoustakis, C. X., Dobre, C., & Mongay Batalla, J. (2017). Handling Big Data in the Era of Internet of Things (IoT). In Constandinos X. Mavromoustakis, George Mastorakis, & Ciprian Dobre (Eds.), *Advances in Mobile Cloud Computing and Big Data in the 5G Era* (1st ed., pp. 3–22). Springer.
<https://doi.org/10.1007/978-3-319-45145-9>
- Parnas, D. L. (1994). Software Aging. *ICSE '94 Proceedings of the 16th International Conference on Software Engineering*, 279–287.
- Paska, D. (2018). Digitalized water and smart cities - how can telecommunication networks be used for environmental resilience? *ITU Journal: ICT Discoveries*, 2, 1–8.
<https://xilirprojects.com/wp-content/uploads/2019/11/ITU2018-3.pdf>
- Pavlović, B. (1999). Bez brige, zaostali smo u razvoju. *Glas Javnosti*, 21.
<https://brankopavlovic.rs/documentation/glasnedelje7.pdf>
- Perry, D. E., & Wolf, A. L. (1992). Foundations for the study of software architecture. *ACM SIGSOFT Software Engineering Notes*, 17(4), 40–52.
<https://doi.org/10.1145/141874.141884>
- Pfleeger, S. L., & Atlee, J. M. (2009). *Software Engineering: Theory and Practice* (4th ed.). Pearson.
- Plantak Vukovac, D., & Orehovački, T. (2010). Metode vrednovanja web upotrebljivosti. *CASE 22 - Metode i Alati Za Razvoj Poslovnih i Informatičkih Sustava*, 171–182.
- Powell-Morse, A. (n.d.). *Extreme Programming: What Is It And How Do You Use It?* Retrieved April 18, 2020, from <https://airbrake.io/blog/sdlc/extreme-programming>
- Prčić, T. (2000). O sindromu milenijumske bube i jeziku anglosrpskom. *Južnoslovenski Filolog*, LVI, 867–873. <http://dais.sanu.ac.rs/bitstream/id/20894/3238.pdf>
- Pressman, R., & Maxim, B. (2015). *Software Engineering: A Practitioner's Approach* (8th ed.).

McGraw-Hill Education.

- PwC. (2016). *Balance of Risk: Risk and reward across the water and sewerage value chain*.
http://www.ofwat.gov.uk/wp-content/uploads/2015/12/rpt_com201512pwcrisk.pdf
- Quesenbery, W. (n.d.). *What Does Usability Mean: Looking Beyond 'Ease of Use.'* Retrieved March 28, 2020, from <https://www.wqusability.com/articles/more-than-ease-of-use.html>
- Radenković, M., Glogonjac, D., & Probić, J. (n.d.). *ODRŽIVI RAZVOJ - CILJ BROJ 12 UN AGENDE ODRŽIVOG RAZVOJA 2030: ODGOVORNA POTROŠNJA I PROIZVODNJA VODE*. Retrieved May 14, 2020, from <https://www.adriamediagroup.com/sr/odrzi-razvoj-cilj-broj-12-un-agende-odrzi-vog-razvoja-2030-odgovorna-potrosnja-i-proizvodnja-vode/>
- Rahgozar, M., & Oroumchian, F. (2002). A Practical Approach for Modernization of Legacy Systems. *First Eurasia Conference on Advances in Information and Communication Technology-ICT2002*.
- Rainer, K., Prince, B., & Cegielski, C. (2013). *Introduction to Information Systems: Supporting and Transforming Business* (5th ed.). John Wiley & Sons, Inc.
<https://doi.org/10.1007/978-1-137-08467-5>
- Rainer, K., & Turban, E. (2009). *Uvod u informacione sisteme - prevod drugog izdanja* (2nd ed.). DATA STATUS.
- Rajlich, V. (2000). Incremental redocumentation using the Web. *IEEE Software*, 17(5), 102–106. <https://doi.org/10.1109/52.877875>
- Razak, F. H. A., Hafit, H., Sedi, N., Zubaidi, N. A., & Haron, H. (2010). Usability testing with children: Laboratory vs field studies. *Proceedings - 2010 International Conference on User Science and Engineering, i-USER 2010*, 104–109.
<https://doi.org/10.1109/IUSER.2010.5716733>
- Reed, P. (1986). Usability Testing in the Real World. *In Proceedings of CHI 86*.
- Ren, Y., Xing, T., Chen, X., & Chai, X. (2011). Research on software maintenance cost of influence factor analysis and estimation method. *2011 3rd International Workshop on Intelligent Systems and Applications, ISA 2011 - Proceedings*.
<https://doi.org/10.1109/ISA.2011.5873461>
- Richard, P. J., Devinney, T. M., Yip, G. S., & Johnson, G. (2009). Measuring organizational performance: Towards methodological best practice. *Journal of Management*, 35(3), 718–

804. <https://doi.org/10.1177/0149206308330560>

Riedel, R., Fransoo, J., Wiers, V., & Fischer, K. (2011). Building Decision Support Systems for Acceptance. In J. C. Fransoo, T. Wafler, & J. R. Wilson (Eds.), *Behavioral Operations in Planning and Scheduling* (pp. 231–294). Springer-Verlag. <https://doi.org/10.1007/978-3-642-13382-4>

Robson, W. (1997). *Strategic Management and Information Systems: An Integrated Approach*. FT Publishing International.

Rodić, A. (2009). „Infostan“ bez potrebe troši 150 miliona.
<https://www.blic.rs/vesti/beograd/infostan-bez-potrebe-trosi-150-miliona/eefen11>

Rosenberg, L. H. (n.d.). *Software Re-engineering*.
<https://pdfs.semanticscholar.org/d6b2/89019f9fef924fd3300d1094f29c8bc079f9.pdf>

Ross, J., & Feeny, D. (1999). *The Evolving Role of the CIO*.
<https://dspace.mit.edu/bitstream/handle/1721.1/2758/SWP-4089-43797710-CISR-308.pdf?sequence=1&isAllowed=y>

Rowley, J. (2007). The wisdom hierarchy: Representations of the DIKW hierarchy. *Journal of Information Science*, 33(2), 163–180. <https://doi.org/10.1177/0165551506070706>

Royce, W. W. (1970). Managing the Development of Large Software Systems. *Proceedings of IEEE WESCON*, 26, 328–338.

Ryan, N. S., Pascoe, J., & Morse, D. R. (1998). Enhanced Reality Fieldwork: the Context-aware Archaeological Assistant. In V. Gaffney, M. van Leusen, & S. Exxon (Eds.), *Computer Applications in Archaeology*.

Saravanan, M., Das, A., & Iyer, V. (2017). Smart water grid management using LPWAN IoT technology. *2017 Global Internet of Things Summit (GIoTS)*.
<https://doi.org/10.1109/GIOTS.2017.8016224>

Schach, R. (n.d.). *Software Life Cycle Models*. Retrieved May 22, 2020, from
<http://courses.cs.vt.edu/csonline/SE/Lessons/LifeCycle/Lesson.html#refs>

Scrum.org. (n.d.). *What is a Product Backlog?* Retrieved March 27, 2020, from
<https://www.scrum.org/resources/what-is-a-product-backlog>

Seacord, R. C., Plakosh, D., & Lewis, G. A. (2003). *Modernizing Legacy Systems: Software Technologies, Engineering Processes, and Business Practices*. Addison-Wesley

Professional.

- Sebesta, R. W. (2016). *Concepts of Programming Languages* (Global Edi). Pearson.
- Seffah, A., Donyaee, M., Kline, R. B., & Padda, H. K. (2006). Usability measurement and metrics: A consolidated model. *Software Quality Journal*, *14*(2), 159–178.
<https://doi.org/10.1007/s11219-006-7600-8>
- Shackel, B. (1981). The concept of usability. *Proceedings of IBM Software and Information Usability Symposium*.
- Shahvari, K. (2019). Evaluating the portals of Iran ministries using the Standard WebQEM. *Journal of Management & Technology*, *19*, 249–260.
- Sharma, S., Sarkar, D., & Gupta, D. (2012). Agile Processes and Methodologies: A Conceptual Study. *International Journal on Computer Science & Engineering*, *4*, 892–898.
- Shneiderman, B., & Plaisant, C. (2005). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison Wesley.
- Singh, B., & Kannoja, S. P. (2013). A review on software quality models. *Proceedings - 2013 International Conference on Communication Systems and Network Technologies, CSNT 2013*, 801–806. <https://doi.org/10.1109/CSNT.2013.171>
- Singh, C., Sharma, N., & Kumar, N. (2019). Analysis of software maintenance cost affecting factors and estimation models. *International Journal of Scientific and Technology Research*, *8*(9), 276–281.
- Singh, V. (1997). Systems integration □ coping with legacy systems. *Integrated Manufacturing Systems*, *8*(1).
- SIS. (2003). *ISS-SO/IEC 9126-1 Software Engineering - Product Quality*.
<https://www.sis.se/api/document/preview/33592/>
- Sitharama Iyengar, S., Parameshwaran, N., Phoha, V. V, Balakrishnan, N., & Okoye, C. D. (2010). *Fundamentals of Sensor Network Programming: Applications and Technology* (1st ed.). Wiley.
- Sneed, H. M. (1995). Planning the Reengineering of Legacy Systems. *IEEE Software*, *12*(1), 24–34. <https://doi.org/10.1109/52.363168>
- Sneed, H. M. (2000). Encapsulation of legacy software: A technique for reusing legacy software components. *Annals of Software Engineering*, *9*(1–4), 293–313.

<https://doi.org/10.1023/a:1018989111417>

- Sneed, H. M. (2006). Integrating legacy software into a service oriented architecture. *Conference on Software Maintenance and Reengineering (CSMR'06)*.
<https://doi.org/10.1109/CSMR.2006.28>
- Soldić-Aleksić, J. (2011). *Primenjena analiza podataka: rad u programima za statističku analizu i tabelarna izračunavanja*. Ekonomski fakultet.
- Somasundaram. (n.d.). *IoT Standards and Protocols*. Retrieved July 20, 2020, from
<https://www.ilovephd.com/iot-standards-and-protocols/>
- Sommerville, I. (n.d.). *Legacy systems*. Software Engineering. Retrieved May 12, 2020, from
<https://ifs.host.cs.st-andrews.ac.uk/Books/SE9/Web/LegacySys/index.html>
- Sommerville, I. (2006). *Software Engineering* (8th ed.). Addison Wesley.
- Sommerville, I. (2016). *Software Engineering* (10th ed.). Pearson Education Limited.
- Sourcemaking.com. (n.d.-a). *Stovepipe Enterprise*. Retrieved May 28, 2020, from
<https://sourcemaking.com/antipatterns/stovepipe-enterprise>
- Sourcemaking.com. (n.d.-b). *Stovepipe System*. Retrieved May 28, 2020, from
<https://sourcemaking.com/antipatterns/stovepipe-system>
- Sousa, K., & Oz, E. (2014). *Management Information Systems* (7th ed.). Cengage Learning.
- Spitzer, D. (2007). *Transforming Performance Measurement: Rethinking the Way We Measure and Drive Organizational Success*. AMACOM.
- Srinivas, M., Ramakrishna, G., Rajasekhara Rao, K., & Suresh Babu, E. (2016). Analysis of legacy system in software application development: A comparative survey. *International Journal of Electrical and Computer Engineering*, 6(1), 292–297.
<https://doi.org/10.11591/ijece.v6i1.8367>
- Stair, R. M., & Reynolds, G. W. (2018). *Principles of Information Systems* (13th ed.). Cengage Learning.
- Stankić, R., Jovanović-Gavrilović, B., & Soldić-Aleksić, J. (2018). Informatičko-komunikacione tehnologije u obrazovanju kao podsticaj ekonomskom razvoju. *Ekonomski Horizonti*, 20(1), 61–73. <https://doi.org/10.5937/ekonhor1801061s>
- Stefanović, M., Ristić, S., Pržulj, Đ., & Vukmanović, M. (2016). Lehman-ovi zakoni evolucije

- softvera i Open Source Software. *INFOTEH-JAHORINA*, 15, 541–545.
- Stehle, E., Piles, B., Max-Sohmer, J., & Lynch, K. (2008). *Migration of Legacy Software to Service Oriented Architecture*.
<https://www.cs.drexel.edu/~bmittchell/course/cs575/WorkshopF0708/2T7Paper.pdf>
- Stonebraker, M., & Brodie, M. L. (1995). *Migrating Legacy Systems: Gateways, Interfaces & the Incremental Approach* (1st ed.). Morgan Kaufmann Pub.
- Suman, & Manoj, W. (2014). A Comparative Study of Software Quality Models. *International Journal of Computer Science and Information Technologies*, 5(4), 5634–5638.
<https://doi.org/10.1109/ICAETR.2014.7012958>
- Sun, X., & May, A. (2013). A comparison of field-based and lab-based experiments to evaluate user experience of personalised mobile devices. *Advances in Human-Computer Interaction*, 1–9. <https://doi.org/10.1155/2013/619767>
- Tamir, D., Komogortsev, O. V., & Mueller, C. J. (2008). An effort and time based measure of usability. *WoSQ '08: Proceedings of the 6th International Workshop on Software Quality*, 47–52. <https://doi.org/10.1145/1370099.1370111>
- Taylor, M., Moynihan, E., & Wood-Harper, T. (1997). Knowledge for software maintenance. *Journal of Information Technology*, 12(2).
<https://doi.org/https://doi.org/10.1080/026839697345161>
- techtarget. (n.d.). *automated meter reading (AMR)*. Retrieved May 30, 2020, from
<https://whatis.techtarget.com/definition/automated-meter-reading-AMR>
- Thornton, J., Sturm, R., & Kunkel, G. (2008). *Water Loss Control* (2nd ed.). McGraw-Hill Education.
- Tilley, S. R. (1993). Documenting-in-the-large vs. documenting-in-the-small. *CASCON '93: Proceedings of the 1993 Conference of the Centre for Advanced Studies on Collaborative Research: Distributed Computing - Volume 2*, 1083–1090.
- Tripathi, K. (2011). A Study of Interactivity in Human Computer Interaction. *International Journal of Computer Applications*, 16(6), 1–3.
- Tripathy, P., & Naik, K. (2015). Software evolution and maintenance: A practitioner's approach. In *Software Evolution and Maintenance: A Practitioner's Approach* (Vol. 9780470603). <https://doi.org/10.1002/9781118964637>

- Trivedi, M. C., & Khanum, M. A. (2012). Role Of Context In Usability Evaluations: A Review. *Advanced Computing: An International Journal*, 3(2), 69–78.
<https://doi.org/10.5121/acij.2012.3208>
- Tromp, H., & Hoffman, G. (2003). Evolution of legacy systems: strategic and technological issues, based on a case study. *Proceedings of ELISA, International Workshop on Evolution of Large-Scale Industrial Software Applications*, 147–153.
- UIUXtrend. (n.d.). *Measuring and Interpreting System Usability Scale (SUS)*. Retrieved July 20, 2020, from <https://uiuxtrend.com/measuring-system-usability-scale-sus/>
- Ulrich, W. M. (2002). *Legacy Systems: Transformation Strategies*. Pearson.
- Ulrich, W. M. (2005). Architecture - Driven Modernization 101: Concepts, Strategies & Justification. *Architecture-Driven Modernization Workshop*.
<https://www.omg.org/news/meetings/workshops/adm-2005.htm>
- Ulrich, W. M., & Newcomb, P. H. (2010). Information Systems Transformation: Architecture-Driven Modernization Case Studies. In *Information Systems Transformation*. Morgan Kaufmann OMG Press Titles. <https://doi.org/10.1016/C2009-0-19987-7>
- UNESCO. (2011). *The Impact of Global Change on Water Resources*.
- United Nations. (n.d.). *68% of the world population projected to live in urban areas by 2050*. Retrieved May 12, 2020, from <https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html>
- United States Government Accountability Office. (2019). *GAO-19-471 - INFORMATION TECHNOLOGY - Agencies Need to Develop Modernization Plans for Critical Legacy Systems*. <https://www.gao.gov/assets/700/699616.pdf>
- usability.gov. (n.d.). *Eye Tracking*. Retrieved May 18, 2020, from <https://www.usability.gov/how-to-and-tools/methods/eye-tracking.html>
- Usability.gov. (n.d.). *System Usability Scale (SUS)*. Retrieved July 25, 2020, from <https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>
- Valacich, J., & Schneider, C. (2017). *Information Systems Today: Managing the Digital World* (8th ed.). Pearson.
- Van Gurp, J., & Bosch, J. (2002). Design erosion: Problems and causes. *Journal of Systems and*

- Software*, 61(2), 105–119. [https://doi.org/10.1016/S0164-1212\(01\)00152-2](https://doi.org/10.1016/S0164-1212(01)00152-2)
- van Vliet, H. (2007). *Software Engineering: Principles and Practice* (3rd ed.). Wiley.
- Verdone, R., Dardari, D., Mazzini, G., & Conti, A. (2008). *Wireless Sensor and Actuator Networks: Technologies, Analysis and Design* (1st ed.). Academic Press.
- Vesić, S. L. (2016). Progressivne web aplikacije - između nativnih i mobilnih Web aplikacija. *InfoM*, 15(60), 43–49.
- Vesić, S. L. (2020). Uticaj razvoja IT tehnologija na prevazilaženje problema nasleđenih sistema. *Kultura Polisa*, XVII(42), 667–678.
- Vesić, S. L., Aničić, N., & Babarogić, S. (2016). Model-Driven Approach to The Implementation of Web Service Interfaces For 4GL System. *XV International Scientific Symposium SymOrg2016*, 805–813.
- Vesić, S. L., & Minović, M. (2015). Single-Page Applications - trend ili budućnost. *InfoM*, 14(55), 48–54.
- Vestues, K., & Knut, R. (2019). Making Digital Infrastructures More Generative Through Platformization and Platform- driven Software Development: An Explorative Case Study. *Tenth Scandinavian Conference on Information Systems (SCIS2019)*, 1–15. <https://aisel.aisnet.org/scis2019/4>
- WAMMI. (n.d.). *What is WAMMI*. Retrieved May 21, 2020, from <http://www.wammi.com/whatis.html>
- WAMMI (*Website Analysis and Measurement Inventory*). (n.d.). Retrieved May 22, 2020, from <https://www.allaboutux.org/wammi-website-analysis-and-measurement-inventory>
- Warren, I. (1999). *The Renaissance of Legacy Systems*. Springer.
- Weiderman, N. H., Bergey, J. K., Smith, D. B., & Tilley, S. R. (1997). Approaches to Legacy System Evolution. In *CMU/SEI-97-TR-014*.
- What is a KPI?* (n.d.). Retrieved April 2, 2020, from <https://www.klipfolio.com/resources/articles/what-is-a-key-performance-indicator>
- Whitten, J., & Bentley, L. (2005). *Analysis and Design Methods*. (7th ed.). McGraw-Hill/Irwin.
- Wiederhold, G. (1992). The roles of artificial intelligence in information systems. *Journal of Intelligent Information Systems*, 1, 35–55.

- Willem-Jan, van den H. (2009). *Aligning Modern Business Processes and Legacy Systems: A Component-Based Perspective*. The MIT Press. <https://doi.org/978-0262513463>
- Wilson, C. (2014). *User Interface Inspection Methods: A User-Centered Design Method*. Morgan Kaufmann.
- Wixon, D., Jones, S., Tse, L., & Casaday, G. (1994). Inspections and design reviews: Framework, history, and reflection. In J. Nielsen & R. L. Mack (Eds.), *Usability Inspection Methods* (pp. 77–103). John Wiley & Sons.
- Wong, K., Tilley, S. R., Müller, H. A., & Storey, M. A. D. (1995). Structural Redocumentation: A Case Study. *IEEE Software*, 12(1), 46–54. <https://doi.org/10.1109/52.363166>
- Wu, B., Lawless, D., Bisbal, J., Grimson, J., Wade, V., O’Sullivan, D., & Richardson, R. (1997). Legacy systems migration-a method and its tool-kit framework. In IEEE (Ed.), *Proceedings of Joint 4th International Computer Science Conference and 4th Asia Pacific Software Engineering Conference* (pp. 312–320). IEEE. <https://doi.org/10.1109/apsec.1997.640188>
- Yoon, V., Aiken, P. H., & Guimaraes, T. (2005). Applying a Metadata Framework to Improve Data Quality. In *Encyclopedia of Information Science and Technology* (1st ed., pp. 146–151).
- Yuuki, H., & Kashimura, K. (2010). Cognitive aging and rich internet applications: Usability problems of Ajax based on the empirical study of older adults. *Japanese Psychological Research*, 52(3), 216–226. <https://doi.org/https://doi.org/10.1111/j.1468-5884.2010.00439.x>
- Zennaro, M. (2016). *Intro to Internet of Things*. [https://www.itu.int/en/ITU-D/Regional-Presence/AsiaPacific/SiteAssets/Pages/Events/2016/Dec-2016-IoT/IoTtraining/IoT Intro-Zennaro.pdf](https://www.itu.int/en/ITU-D/Regional-Presence/AsiaPacific/SiteAssets/Pages/Events/2016/Dec-2016-IoT/IoTtraining/IoT%20Intro-Zennaro.pdf)
- Ziegler, D., Sorg, F., Fallis, P., & Hübschen, H. (2009). *Guidelines for water loss reduction: A focus on pressure management*.
- Јовановић, Љ., Јовановић, Д., Вељковић, Н., Савић, А., & Станојевић, Д. (2014). *Спровођење протокола о води и здрављу у републици Србији - анализа стања*. http://www.sepa.gov.rs/download/Protokol_o_vodi_i_zdravlju_Analiza_stanja.pdf
- Петровић, Б. (1998). *Теорија система*. Факултет организационих наука.
- Раденковић, Б., Станојевић, М., & Марковић, А. (1999). *Рачунарска симулација*. Факултет

организационих наука, Саобраћајни факултет.

Прилози

Прилог 1: Анкета за евалуацију новог теоријског оквира

1. Назив државе у којем се налази Ваше предузеће _____
2. Назив града у којем се налази Ваше предузеће _____
3. Ваше предузеће има стари (застарели) информациони систем (ИС) или софтвер:
 - у потпуности се слажем
 - делимично се слажем
 - подељено
 - делимично се не слажем
 - у потпуности се не слажем
4. Стари (застарели) ИС или софтвер је стар:
 - мање од 10 година
 - између 10 и 15 година
 - између 15 и 20 година
 - између 20 и 25 година
 - преко 25 година
5. Програмски језик (или језици) у којима је кодиран стари ИС (или софтвер):

6. У старом ИС (или софтверу/има), нарушен је иницијално постављен дизајн (пројектна замисао) услед великог броја измена и дугорочног одржавања:
 - у потпуности се слажем
 - делимично се слажем
 - подељено
 - делимично се не слажем
 - у потпуности се не слажем
7. Стари ИС је користан, тиме што и даље доприноси пословању (и даље се извршавају одређене пословне процедуре и процеси):
 - у потпуности се слажем
 - делимично се слажем
 - подељено
 - делимично се не слажем
 - у потпуности се не слажем

8. У старом ИС релативно једноставно се врше измене, у погледу измене пословних процедура или пословних процеса, а у складу са захтевима:
- у потпуности се слажем
 - делимично се слажем
 - подељено
 - делимично се не слажем
 - у потпуности се не слажем
9. Ако се изврши измена у једном делу старог ИС, та промена се пропагира ка другим деловима система:
- у потпуности се слажем
 - делимично се слажем
 - подељено
 - делимично се не слажем
 - у потпуности се не слажем
10. Стари ИС има апликације (развијене за неку организациону јединицу), које се извршавају изоловано од других апликација и веома тешко „сарађују“ са другим апликацијама:
- у потпуности се слажем
 - делимично се слажем
 - подељено
 - делимично се не слажем
 - у потпуности се не слажем
11. Заменом старе апликације или дела старог ИС потенцијално новим решењем, постоји значајно прилагођавање старог ИС да би радио без прекида са новим:
- у потпуности се слажем
 - делимично се слажем
 - подељено
 - делимично се не слажем
 - у потпуности се не слажем
12. Интеграција старог ИС, са неким другим системима у оквиру предузећа (GIS, SCADA, EDAMS, и тд.) је једноставна:
- у потпуности се слажем
 - делимично се слажем
 - подељено

- делимично се не слажем
- у потпуности се не слажем

13. Постоји проблем у размени података старог ИС, са неким системом другог предузећа на аутоматизовани или полу-аутоматизовани начин:

- у потпуности се слажем
- делимично се слажем
- подељено
- делимично се не слажем
- у потпуности се не слажем

14. Стари ИС се извршава на застарелом хардверу:

- у потпуности се слажем
- делимично се слажем
- подељено
- делимично се не слажем
- у потпуности се не слажем

15. Програмери (пројектанти, software developer-и, software enginner-и) могу да примене нановија знања методологије, концепте и алате из области софтверског инжењерства, без проблема у старом ИС:

- у потпуности се слажем
- делимично се слажем
- подељено
- делимично се не слажем
- у потпуности се не слажем

16. Стари ИС је развијен од стране кадрова и средстава самог предузећа (in-house):

- у потпуности се слажем
- делимично се слажем
- подељено
- делимично се не слажем
- у потпуности се не слажем

17. Стари ИС је потребно модернизовати

- у потпуности се слажем
- делимично се слажем
- подељено
- делимично се не слажем

- у потпуности се не слажем

18. Документација старог ИС постоји и ажурна је

- у потпуности се слажем
- делимично се слажем
- подељено
- делимично се не слажем
- у потпуности се не слажем

19. Велики је одлазак запослених који у радили на старом ИС

- у потпуности се слажем
- делимично се слажем
- подељено
- делимично се не слажем
- у потпуности се не слажем

20. Редунданса у систему је изражена

- у потпуности се слажем
- делимично се слажем
- подељено
- делимично се не слажем
- у потпуности се не слажем

21. Новозапослени имају знање и вештине да у старту раде са старим ИС

- у потпуности се слажем
- делимично се слажем
- подељено
- делимично се не слажем
- у потпуности се не слажем

22. Постоји подршка приликом одржавања хардвера и софтвера од стране произвођача

- у потпуности се слажем
- делимично се слажем
- подељено
- делимично се не слажем
- у потпуности се не слажем

23. Модернизациом старог ИС могу створити бољи услови за подршку одлучивању него супротно.

- у потпуности се слажем

- делимично се слажем
- подељено
- делимично се не слажем
- у потпуности се не слажем

24. Стари ИС, такав какав је (без измена) може се лако интегрисати са елементима Индустије 4.0 (Cloud Computing, Internet of Things, Big Data и тд.):

- у потпуности се слажем
- делимично се слажем
- подељено
- делимично се не слажем
- у потпуности се не слажем

25. Налазим да овакав упитник може боље усмерити модернизацију система, уместо посматрања само из једног угла (нпр. технолошког):

- у потпуности се слажем
- делимично се слажем
- подељено
- делимично се не слажем
- у потпуности се не слажем

Прилог 2: Анкета за оцену употребљивости

1. Мислим да бих желео да овај систем користим често
 - у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем
2. Налазим да је систем непотребно сложен
 - у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем
3. Мислим да је систем једноставан за коришћење
 - у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем
4. Мислим да ће ми бити потребна подршка техничке особе да бих могао да користим систем
 - у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем
5. Налазим да су различите функције у овом систему добро интегрисане
 - у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем

6. Мислим да је много неконзистентности у систему
- у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем
7. Замишљај да би већина људи научила да систем користи брзо
- у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем
8. Налазим да је систем незгодан за коришћење
- у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем
9. Осећао сам се веома поуздано приликом коришћења система
- у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем
10. Потребно је било да научим много ствари пре него што бих могао да кренем са овим системом
- у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем

Прилог 3: Анкета за процену пословне вредности

1. Систем помаже организацији да оствари нове могућности
 - у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем
2. Параметри трошкова система су унутар буџета
 - у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем
3. Софтвер се може прилагодити променама у пословним захтевима и има подршку произвођача
 - у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем
4. Систем може да задовољи нове стандардизационе профиле, укључујући регулаторне, индустријске и корпоративне стандарде за ИТ платформе, податке и апликације
 - у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем
5. Организација се у потпуности ослања на наслеђен систем
 - у потпуности се не слажем
 - не слажем се
 - неутралан сам
 - слажем се
 - у потпуности се слажем

6. Систем је робустан у односу на сигурносне претње

- у потпуности се не слажем
- не слажем се
- неутралан сам
- слажем се
- у потпуности се слажем

Биографија

Славимир Весић рођен је 11.02.1984. године у Београду. Завршио је Прву београдску гимназију. Основне академске студије на студијском програму „Информациони системи и технологије“ на Факултету организационих наука у Београду завршио је 2010. године (просечна оцена 8,2). Дипломске академске (мастер) студије на смеру за Информационе системе и технологије, програмско подручје (модул) Информационе технологије, на Факултету организационих наука у Београду завршио је 2012. године (просечна оцена 9,57). Докторске академске студије на Факултету организационих наука, студијски програм Информациони системи и менаџмент, изборно подручје Информациони системи, уписује 2013. године. Положио је свих девет, програмом предвиђених, испита на докторским студијама са просечном оценом 10. У току трајања докторских студија прелази на Факултет за економију и инжењерски менаџмент, где је са успехом положио разлику испита са просечном оценом 10.

Од децембра 2010. година ради у Сектору за информационо комуникационе технологије, ЈКП „Београдски водовод и канализација“. Радио је на пословима програмирања, пројектовања, израде пословних извештаја, израде Web апликација и тд., на којима је и даље ангажован.