

TARTU ÜLIKOOL
MATEMAATIKA-INFORMAATIKATEADUSKOND
Arvutiteaduse instituut
Informaatika eriala

Rain Ojavee

Tervise infosüsteemiga liidestumine haigla infosüsteemi LIISA näitel

Bakalaureusetöö (6 EAP)

Juhendaja: Vambola Leping

Autor: “.....“ mai 2013

Juhendaja: “.....“ mai 2013

Lubada kaitsmisele

Professor: “.....“ mai 2013

TARTU 2013

Sisukord

Sisukord.....	2
Mõisted	4
Sissejuhatus.....	5
1 Liidestatavad infosüsteemid	7
1.1 Tervise infosüsteem	7
1.1.1 Tervise infosüsteemi ülevaade.....	7
1.1.2 Tervise infosüsteemi turvalisus.....	8
1.1.3 Tervise infosüsteemi publitseeritud standardid	9
1.2 Haigla infosüsteem LIISA	10
1.2.1 LIISA ülesehitus.....	10
1.2.2 LIISA tehniline kirjeldus	11
1.2.3 LIISA turvalisus	12
2 Sõnumite standardid	13
2.1 HL7 versioon 3.....	13
2.2 CDA dokument	13
2.2.1 Objekti identifikaator	14
3 Terviseandmeid sisaldavate dokumentide edastamine Digilukku.....	16
3.1 Ülevaade dokumendi edastamisest Digilukku	16
3.2 Dokumendi moodustamine.....	17
3.2.1 CDA dokumendi andmete päring andmebaasist.....	17
3.2.2 Transformatsioonifaili koostamine	20
3.2.3 Andmete transformeerimine CDA dokumendi kujule.	21
3.2.4 Viide XSL failile	23
3.3 Dokumendi allkirjastamine	23
3.4 Dokumendi saatmine Digilukku	24
3.4.1 Dokumendi edastamiseks kasutatav sõnum.....	24
3.4.1.1 Dokumendi ümbris.....	25
3.4.2 Ümbrise lisamine dokumendile.....	25
3.4.3 Dokumendi saatmine	26
4 Dokumendi vaatamine Digiloost infosüsteemis LIISA.....	28
4.1 Dokumentide vaatamise protsess.....	28
4.2 Liides dokumentide nimekirja vaatamiseks Digiloost	29

4.3	Dokumentide nimekirja päring	29
4.4	Dokumentide nimekirja päringu vastus	30
4.5	Tervikdokumendi päring Digiloost	31
4.6	Tervikdokumendi kuvamine.....	31
	Kokkuvõte.....	33
	Summary	34
	Kasutatud kirjandus.....	35
	Lisad.....	39
	Lisa 1 – Uuringu vastuse CDA dokument XML kujul.....	39
	Lisa 2 – Uuringu vastuse CDA dokument inimloetaval kujul.....	42

Mõisted

- **XML** (*Extensible Markup Language*) – tekstipõhine informatsiooni struktureeritud esitamise ja talletamise viis, mis on nii inim- kui ka masinloetav. XML on SGML (Standard Generalized Markup Language) kitsendatud kuju. Tegu on ühe kõige laialdasemalt kasutatava struktureeritud andmete jagamise viisiga arvutite ja andmevõrkude vahel. [1]
- **XSL** (*Extensible Stylesheet Language*) – keel, mille abil saab XML dokumenti kasutajale kuvamiseks kujundada. Kujundamise eesmärk on dokumendi loetavust parandada. [2]
- **XSLT** (*XSL Transformations*) – keel XML dokumendi või selle osade muundamiseks teise kujuga XML dokumendiks. XSLT on XSL osa. [3]
- **XSD** (*XML Schema Definition*) – XML skeemi definitsioon, mis määrab ära XML dokumendi struktuuri ning kasutatavad elemendid, nende tüübid ja kohustuslikkuse. [4]
- **SOAP** (*Simple Object Access Protocol*) – XML põhine protokoll vahetamiseks andmeid üle HTTP. [5]
- **BASE64** – andmete kodeerimismeetod, mis teisendab binaarandmed US-ASCII tekstiks. Teisendatud andmeid on võimalik binaarkujule tagasi teisendada. [6]
- **MIME** (*Multipurpose Internet Mail Extensions*) – standard, mis defineerib sõnumi sisu ja võimaldab lisaks US-ASCII vormingus tähemärkide saata ka teistes vormingutes tähemärke. Lisaks võimaldab MIME saata ka heli-, pildi- ja videovormingus sõnumeid ja faile. [7]
- **HL7** (*Health Level Seven International*) – ANSI (*American National Standards Institute*) poolt akredeeritud mittetulundusorganisatsioon, kes tegeleb tervishoiualase andmevahetuse standardite väljatöötamisega. [8]
- **Mall** (*Template*) – dokument või fail, millel on ettemääratud formaat. [9]
- **Haiguslugu** – haige kohta peetav haiguse kulu ja ravi kirjeldus. [10]

Sissejuhatus

Veel mõned aastad tagasi hoiti inimese tervise kohta käivaid andmeid erinevates raviasutustes, kelle teenuseid inimesed kasutasid. Kui patsient läks arsti vastuvõtule, siis arstidel puudus ülevaade patsiendi teistes haiglates olevatest haiguslugudest. Andmed olid küll olemas, kuid asusid teiste raviasutuste vahel laiali. Seega oli ühtse tervikpildi saamine patsiendi tervisest raskendatud ning palju vajalikku ja tihti ka aegkriitilist infot (allergiad, immuniseerimised) jäi kättesaamatuks. Selleks, et tervishoiusüsteemi tõhustada ning tervishoiuteenuste arendamist kvaliteetsemaks ja patsiendisõbralikumaks muuta, arendati Sotsiaalministeeriumi tellimusel välja Tervise infosüsteem.

Käesolevas töös käsitletakse haigla infosüsteemi ja Tervise infosüsteemi liidestamist ja omavahelist andmevahetust. Töös keskendutakse andmete vastuvõtmise ja edastamise tehnoloogilistele aspektidele meditsiinitarkvara poolel ning andmete töötlemisele. Näidatakse, kuidas andmebaasist päringu teel saadud andmed transformeeritakse saatmiseks sobivasse formaati ning kirjeldatakse tegevusi, mis on vajalikud andmete edastamiseks. Samuti demonstreeritakse, kuidas Tervise infosüsteemi saadetud dokumente on võimalik vaadata. Eesmärgiks on tutvustada ja analüüsida kogu andmetöötlus- ja andmeedastusahelat ning kasutatavaid tehnoloogiaid selleks, et luua selgem pilt haigla infosüsteemi liidestamisest Tervise infosüsteemiga.

Töös ei käsitleta ainult teooriat, vaid tuuakse näiteid reaalselt toimivast süsteemist, mis omakorda peaks looma parema ettekujutuse kogu süsteemist ning andma ideid asjaga seotud isikutele. Tulemuseks on kirjeldav analüüs Eesti digitaalsest Tervise infosüsteemist ja selle tervishoiuasutustega liidestamisest haigla infosüsteemi LIISA näitel. Töö võiks esmajoones pakkuda huvi neile, kes hakkavad või juba on tervishoiuasutuse infosüsteemi liidestanud Tervise infosüsteemiga, kuid sooviksid tutvuda ka teiste loodud lahendustega. Antud töö vastu võiksid huvi tunda ka inimesed, kes ei ole seotud tervishoiuasutustele tarkvara arendamisega, kuid tahavad teada, kuidas ja mis teid pidi haiguslugudega seotud info liigub. Eesmärgi saavutamiseks kasutab käesoleva töö autor suuremas osas elektroonilist materjali. Praktilised näited ning loodud joonised tuginevad haigla infosüsteemi sellele osale, mis on autori enda poolt programmeeritud.

Esimeses peatükis antakse ülevaade Tervise infosüsteemist ja haigla infosüsteemist LIISA, seletatakse lahti Tervise infosüsteemi olemus ning selle osad ja kirjeldatakse publitseeritud dokumentide standardeid. Samas tutvustatakse haigla infosüsteemi LIISA ülesehitust ja tehnilist poolt.

Teises peatükis tutvustatakse põgusalt andmete edastamiseks ja vastuvõtmiseks ning dokumentide loomiseks kasutatavaid standardeid.

Kolmandas peatükis käsitletakse haigla infosüsteemi seda osa, mis on seotud andmete edastamisega Tervise infosüsteemi. Kirjeldatakse LIISA arhitektuuri ja andmete ettevalmistust Tervise infosüsteemi saatmiseks ning selleks kasutatavaid tehnoloogiaid. Näidatakse, kuidas andmebaasist loetakse dokumendi koostamiseks vajalikud andmed ning transformeeritakse need ühtseks dokumendiks, samuti tutvustatakse saatmise ahelat ning andmete edastamist ja vastuvõtmist.

Neljandas peatükis kirjeldatakse dokumentide vaatamist Tervise infosüsteemist. Näidatakse, kuidas on haigla infosüsteemis lahendatud vastuvõetud andmete töötlemine ja kuvamine kasutajale loetavas vormis.

Töö lisa on toodud näide Tervise infosüsteemi saadetavast dokumendist uuringu vastuse näitel.

1 Liidestatavad infosüsteemid

1.1 Tervise infosüsteem

Suurem osa Eestis tervishoiuteenust pakkuvaid asutusi kasutab oma töös toimivat infosüsteemi või infotehnoloogilisi lahendusi. Kuna neid süsteeme on mitmeid ning nende vahel puudub liidestatus, ei suuda tervishoiuteenuse osutajad omavahel informatsiooni vahetada. Selle probleemi lahendamiseks on loodud riiklik Tervise infosüsteem. [11]

Tervise infosüsteem (TIS) on riigi infosüsteemi kuuluv digitaalne andmekogu, milles töödeldakse tervishoiuvaldkonnaga seotud andmeid. [12] TIS on keskseks struktuuriks meditsiiniinformatsiooni vahendamisel erinevate tervishoiuteenuse osutajate vahel. [11]

1.1.1 Tervise infosüsteemi ülevaade

TIS kujutab endast inimese haiguslugude, uuringute, visiitide ja muid tervist puudutavaid andmeid koondavat andmekogu. Selle vastutav töötleja on Sotsiaalministeerium ja volitatud töötleja on Eesti E-tervise Sihtasutus, kes haldab ja arendab Tervise infosüsteemi ning töötleb selle andmeid. [12]

Tervise infosüsteemi kesksed osadeks on:

- Digilugu – koondab patsiendi terviseandmed ja tänu sellele võimaldab arstil saada patsiendi tervisest parema ülevaate. Patsiendiportaali kaudu saab ka patsient oma terviseandmeid igal ajal vaadata.
- Digiregistratuur – võimaldab patsiendil ja arstil saada ülevaate vabast vastuvõtuaegadest teistes tervishoiuasutustes ning vastuvõtuaega broneerida, muuta ja tühistada.
- Digipilt – võimaldab vaadata ühtsesse pildiarhiivi koondatud röntgenülesvõtteid ja muid pildidiagnostika uuringuid kõigil raviasutustel üle Eesti.
- Digiretsept – elektrooniline retsept patsiendile väljakirjutatud ravimi kättesaamiseks talle kõige sobilikumast apteegist. [13]

Digilugu koondab patsiendi kogu haigusloo. Raviasutused üle Eesti edastavad sinna terviseandmeid. Edastatud andmed on patsiendi raviarstidele kättesaadavad ning neid on võimalik kasutada inimese edaspidise ravi määramiseks ja tema tervisliku seisundi hindamiseks. Andmete koondatuse tõttu on arstil ühtne ülevaade patsiendi haiguslugudest

ning see aitab patsiendi ravimist efektiivsemaks muuta. Digilugu võimaldab raviarstil tutvuda aegkriitiliste andmetega nagu veregrupp, varasemalt esinenud allergiad ja eelnevad haigused. Digiloo olulisus seisneb ka selles, et patsiendi perearst saab kiirema ligipääsu patsiendi uuringutulemustele. Lisaks saab edastatud andmeid kasutada statistilistel eesmärkidel, arstiabi tõhustamiseks ja ravikvaliteedi hindamiseks, samuti teadusuuringuteks. Raviajalugu, mis on kantud Tervise infosüsteemi, saavad läbi patsiendiportaali vaadata ka patsiendid ise, seda arstist sõltumatult. Lisaks näevad patsiendid patsiendiportaalist, kes on nende haiguslugusid vaadanud. [13]

TIS osalejaks on raviasutus või muu juriidiline isik, kes on sõlminud osalemislepingu Eesti E-tervise Sihtasutusega ning omab õigust Digiloo andmete kasutamiseks ja terviseinfo vahetamiseks läbi selle süsteemi. [14]

1.1.2 Tervise infosüsteemi turvalisus

Kuna TIS näol on tegu delikaatsete isiku- ja terviseandmete koondamisega ühte füüsilisse kogumisse, siis pannakse erilist rõhku turvalisusele ning andmekaitsele. Tervise infosüsteemi kesksüsteemis on turve hajutatud, mis võimaldab ühe või mitme elementaarründe korral ära hoida reaalse kahju tekkimist.

Rakendatavad meetmed turvalisuse tõstmiseks Tervise infosüsteemis:

- Kõikide kasutajate autentimiseks kasutatakse kompleksautentimist (ID-kaart ja selle parool või mobiil-ID).
- Infosüsteemil ei ole administraatorit, kes pääseks andmetele piiranguteta ligi.
- Talletatud andmed on krüpteeritud ja kodeeritud ning isiku- ja terviseandmed hoitakse andmebaasis eraldi.
- Kõik tegevused (andmete lisamine, muutmine ja vaatamine) talletatakse olenemata kasutaja juurdepääsuastmest. Infosüsteemil pole superadministraatorit, kes saaks jälgi jätmata tegutseda.
- Võimaliku ründe tuvastamiseks analüüsitakse logitud tegevusi. [15]
- Kõik meditsiinidokumendid digiallkirjastatakse, mis tagab nende tõestusväärtuse ja välistab hilisema muutmise.

1.1.3 Tervise infosüsteemi publitseeritud standardid

Tervise infosüsteemiga liitumiseks on vajalik kliendi poolel realiseerida sõnumite ja dokumentide edastamine ning vastuvõtmine. Selleks on Eesti E-tervise Sihtasutus vajamineva dokumentatsiooni, klassifikaatorid, loendid ja standardid avaldanud Publitseerimiskeskuses. Tegu on tsentraalse meditsiiniklassifikaatorite ja -standardite avaldamise keskkonnaga, mida haldab Eesti E-tervise Sihtasutus. [16]

Seisuga 09.05.2013 on standarditest publitseeritud versioon 3.0, [17] kuid arendamisel on versioon 4.0 standardid. [18]

Tänaseks on avaldatud alljärgnevate Sotsiaalministeeriumi õigusaktidega kinnitatud meditsiinidokumentide sõnumite versioon 3.0 standardid:

1. Ambulatoorne epikriis
2. Statsionaarne epikriis
3. Koduõenduseepikriis
4. Päevastatsionaarne epikriis
5. Statsionaarse haigusloo koondandmete teatis
6. Ambulatoorse haigusloo koondandmete teatis
7. Päevastatsionaarse haigusloo koondandmete teatis
8. Tervisekontrolli arengu teatis
9. Tervisekontrolli kasvamise teatis
10. Tervisekontrolli immuniseerimise teatis
11. Tervisekontrolli immuniseerimise kõrvalnähtude teatis
12. Tervisekontrolli läbivaatuse teatis
13. Tervisekontrolli nõustamise teatis
14. Sünniepikriis
15. Retsept
16. Retsept apteegist (retsepti andmeid täiendatakse apteegis väljaostetud ravimi andmetega)
17. Rasedakaardi visiidi teatis
18. Rasedakaardi üldandmete teatis
19. Rasedakaart (dokument pannakse Digiloos kokku rasedakaardi visiidi teatiste ja rasedakaardi üldandmete teatise põhjal, seega rasedakaarti saab Digiloost ainult vaadata)

20. Tervisekontrolli kaart (dokument pannakse Digiloos kokku tervisekontrolli arengu, kasvamise, immuniseerimise, immuniseerimise kõrvalnähtude, läbivaatuse ja nõustamise teatiste põhjal, seega tervisekontrolli kaarti saab Digiloost ainult pärida)
21. Aegkriitiliste andmete teatis
22. Patsiendi volitus usaldusisikule
23. Tahteavaldus tervishoiuteenuse osutamiseks
24. Pildiviit
25. Saatekiri
26. Vastus saatekirjale [17]

TIS on kaasaegne infosüsteem, mis muudab meditsiinilise abi infovahetuse lihtsamaks nii arstile kui patsiendile. Tänu kiiremale kommunikatsioonile on raviteenus muutunud mugavamaks ja tõhusamaks. Kuna TIS arvestatavaks teguriks on delikaatsed isikuandmed ja iga süsteemi arendamisega käivad kaasas ohud, on talletatud andmed krüpteeritud ja andmebaasis hajutatud. Digitaalsed dokumendid allkirjastatakse, mis tagab nende tõestusväärtuse. Vaadates meditsiinidokumentide standardite loendit, mis on Sotsiaalministeeriumi poolt kinnitatud, tõestab see, kui mitmekülgne on loodud süsteem. Järgmises alapeatükis analüüsitakse Tervise infosüsteemiga liidestatud haigla infosüsteemi LIISA.

1.2 Haigla infosüsteem LIISA

TIS on andmekogu, mis talletab, edastab ja tagastab patsiendi terviseandmeid. Nende tegevuste rakendamiseks on vajalik teiste infosüsteemide liidestumine. Üheks liidestatavaks infosüsteemiks on käesolevas töös käsitletav haigla infosüsteem LIISA.

Haigla infosüsteem LIISA on terviklahendus tervishoiuteenuseid pakkuvale ettevõttele. Tänu oma modulaarsusele on seda võimalik kasutada erineva suurusega raviasutustel, võimaldades valida ainult tööks vajalikud ning sobivamad moodulid. Infosüsteem LIISA on tänaseks päevaks kasutusel 29 raviasutuses, kellest suurimad on Lääne-Tallinna Keskhaigla, Kuressaare Haigla ja SA Tallinna Lastehaigla.[19]

1.2.1 LIISA ülesehitus

Haigla infosüsteem LIISA koosneb moodulite süsteemist, mis võimaldab neid vajadusel kombineerida ühtseks programmiks.

LIISA tähtsamad moodulid jagunevad:

- Registratuur – patsientide broneerimine vastuvõttele ja uuringutele ning visiitidatuste ja tasuliste teenuste arveldus.
- Eriarst – patsiendi terviseandmete sisestamine ja meditsiinidokumentide koostamine, haigete liikumise haldamine osakondades ja palatites, erinevate toimingute teostamine.
- Diagnostika – uuringute ajakavade koostamine, teenuste ja uuringuvastuste sisestamine.
- Raviarveldus – koondarvete koostamine ja arvete edastamine elektroonsel kujul Haigekassale.
- Labor – laborianalüüside tellimuste vastuvõtmine, vastuste kontrollimine ja kinnitamine.
- Arhiiv – dokumentide arhiivi haldamine.
- Aruanded – aruannete koostamine.
- Apteek – meditsiinikaupade laovarvestus.[19]

Lisaks moodulitele on loodud abiprogrammid e-kirjade saatmiseks, dokumentide allkirjastamiseks ja edastamiseks Digiloole. Samuti on loodud LIISAGA ühilduv veebiregistratuur, mis võimaldab sellega liidestunud asutuse arstide juurde aega broneerida otse internetibrauseris.

1.2.2 LIISA tehniline kirjeldus

LIISA on arendatud programmeerimiskeeles Delphi ja töötab ainult keskkonnas Windows. Kuna paljudel klientidel on kasutusel 32-bit Windows, siis on ka LIISA programmid 32-bitised. LIISA kasutab Microsoft SQL Serveri andmebaasi.

LIISA on liidestatud Tervise infosüsteemi mitmete osadega:

- Digilooga – võimaldab saata patsiendi terviseandmeid sisaldavaid dokumente, neid Digiloost vaadata ning saadetud dokumente kustutada.
- Digiregistratuuriga – vabade aegade päring teistest tervishoiuasutustest. Broneerimine, broneeringute muutmine ja tühistamine. Vabade aegade ressursside üleslaadimine kesksüsteemi ning võimalus teistel infosüsteemidel vabasid aegu pärida, broneerida või broneeringuid tühistada.

- Digipildiga – radioloogiliste piltide aadressandmete saatmine, mille alusel saab pöörduda PACS (*Picture Archiving and Communication System*) pildiarhiivide poole tegelike piltide vaatamiseks. [20]
- Digiretseptiga – digiretsepti ja meditsiiniseadme kaardi koostamine, koostatud ja väljastatud retseptide pärimine ning kasutamata jäänud retseptide annulleerimine.

Andmevahetus Tervise infosüsteemiga nõuab liidestatust X-teega. [21] Andmete edastamine toimub üle X-tee kasutatakse HL7 standardile vastavaid sõnumeid. X-tee kaudu toimub ka kasutajate autentimine. Andmevahetuse realiseerimiseks on kaks võimalust:

- Turvaserver – füüsiliselt eraldiseisev arvuti koos sellele paigaldatud eritarkvaraga, mis võimaldab juurdepääsu X-teele. [22]
- MISP (*mini-infosüsteem-portaal*) – portaal, läbi mille saab kasutada asutusele avatud X-tee teenuseid. [23]

1.2.3 LIISA turvalisus

Delikaatsete isiku- ja terviseandmetega töötamine nõuab turvalist keskkonda, et minimeerida andmete kuritarvitamist ja sattumist valede isikute kätte. Turvalisuse tõstmiseks on infosüsteemis LIISA kasutusel järgnevad meetmed:

- Windowsi domeeni logitakse ID-kaardiga
- Kasutajate ligipääs erinevatele moodulitele ja akendele on konfigureeritav
- Kõik tegevused logitakse
- Arstid saavad vaadata ainult nende tööga seotud patsientide haiguslugusid, seejuures peab patsiendil olema aktiivne haigusjuht

Eelpool nimetatud turvalisusmeetmete kasutamise tõttu on infosüsteem LIISA tunnustatud Andmekaitse Inspektsiooni poolt turvaliseks ning selle kasutamine on lubatud kõikides Eesti raviasutustes.

2 Sõnumite standardid

Digilugu kasutab andmevahetuseks teiste infosüsteemidega HL7 (*Health Level Seven*) poolt koostatud standardeid. [24] HL7 on organisatsioon, kes tegeleb meditsiinistandardite väljatöötamisega ning kellel on üle 2300 liikme. [8] Standardid on loodud spetsiaalselt tervishoiuvaldkonna jaoks ning arvestavad meditsiinivaldkonna keerukust ja eripärasid. [25] HL7 standardite näol on tegemist spetsiaalse tervishoiusüsteemi jaoks loodud keelega, mis aitab meditsiinidokumente hoida üheselt mõistetavatena. [26]

2.1 HL7 versioon 3

Digilugu kasutab andmevahetuseks HL7 V3 (versioon 3) standardeid. [24] V3 standardite näol on tegemist tervishoiualase infovahetuse spetsifikatsiooniga, mis põhineb RIM (*Reference Information Model*) mudelil. RIM on objektorienteeritud mudel, kus informatsioon on jagatud klassidesse. Klassid sisaldavad atribuute ning on omavahel seotud. [27] HL7 V3 määrab ära sõnumid, andmetüübid ja terminoloogiad, mille alusel tervishoiuasutused saavad infot vahetada. Sõnumid baseeruvad XML süntaksil. [28]

2.2 CDA dokument

CDA (*Clinical Document Architecture*) on XML baseeruv dokumendi standard, mis määrab kliinilise dokumendi struktuuri ja semantika. CDA on HL7 V3 alamstandard ja seda arendab samuti HL7. Praegu on kasutusel ANSI poolt kinnitatud standardi versioon 2.0. Arendamisel on versioon 3. [29]

CDA dokumenti iseloomustab terviklikkus, inimloetavus, masinloetavus ja selle tõestusväärtust on võimalik tagada allkirjaga. Dokument võib sisaldada teksti, pilte, heli ja ka videot. Dokument edastatakse kasutades HL7 andmevahetusstandardile vastavaid sõnumeid, kuid see eksisteerib sõltumatult ka väljaspool edastamiseks kasutatavat sõnumit. [30]

Digiloo raames kasutatakse CDA standardit digitaalsel kujul dokumentide säilitamiseks.

2.2.1 Objekti identifikaator

Dokumentide üheseks mõistmiseks defineeritakse kõik dokumentides kasutatavad elemendid üheselt, kasutades selleks OID (*object identifier*) süsteemile vastavaid objekte. [31] Need objektid koosnevad kahest atribuudist:

- globaalselt unikaalsest objekti identifikaatorist (OID)
- väärtusest, mille määrab OID valdaja. [32]

OID on objekti identifikaator, järjestatud positiivsete täisarvude loend, mis määrab üheselt objekti OID puus. [33] HL7 sõnumites kirjutatakse OID järgmisel kujul [34]:

```
<juur>.<tase1>.<tase2>...<taseN>
```

OID registreerimine ja haldamine toimub õiguste delegeerimise põhimõttel. OID identifitseerib asutuse, kes on vastutav järgnevate OIDide registreerimise eest. Asutus, kes kontrollib mingi taseme identifikaatorit, võib järgnevate tasemete identifikaatorite haldamise ja registreerimise õigused edasi anda ning seda tegevust võib korrata kuitahes palju. Asutus peab enda hallatavatele OIDidele sõne kujul nimed andma ning need nimed peavad taseme piires unikaalsed olema. [33]

Ettevõtete OIDide väljastamise eest vastutab organisatsioon IANA (*Internet Assigned Numbers Authority*). [35] Digiloo OIDide haldamise õigus on Eesti E-tervise Sihtasutusel. [36]

Allpool on toodud näide sektsiooni *<section>* olemuse määramisest:

```
<section>  
  <code code="PROC" codeSystem="1.3.6.1.4.1.28284.6.2.2.11.1"  
  codeSystemName="Sektsiooni kodeering" displayName="Teostatud uuringud"/>  
</section/>
```

Märgise *<code>* atribuudi *codeSystem* väärtus „1.3.6.1.4.1.28284.6.2.2.11.1“ määrab, et tegu on sektsiooni sisu kirjeldava märgisega ja atribuudi *code* väärtus „PROC“ määrab sektsiooni sisu liigi, milleks on „Teostatud uuringud“.

Standardid määravad reeglid ja meetodid, mille alusel infosüsteemid omavahel suhtlevad. Standardite järgimine tagab infovahetuse erinevate infosüsteemide vahel. Käesolevas peatükis kirjeldatud standardeid on vaja, et oleks võimalik Tervise infosüsteemi ja sellega liidestunud erinevate infosüsteemide vaheline suhtlus. Järgmises peatükis vaatame, kuidas

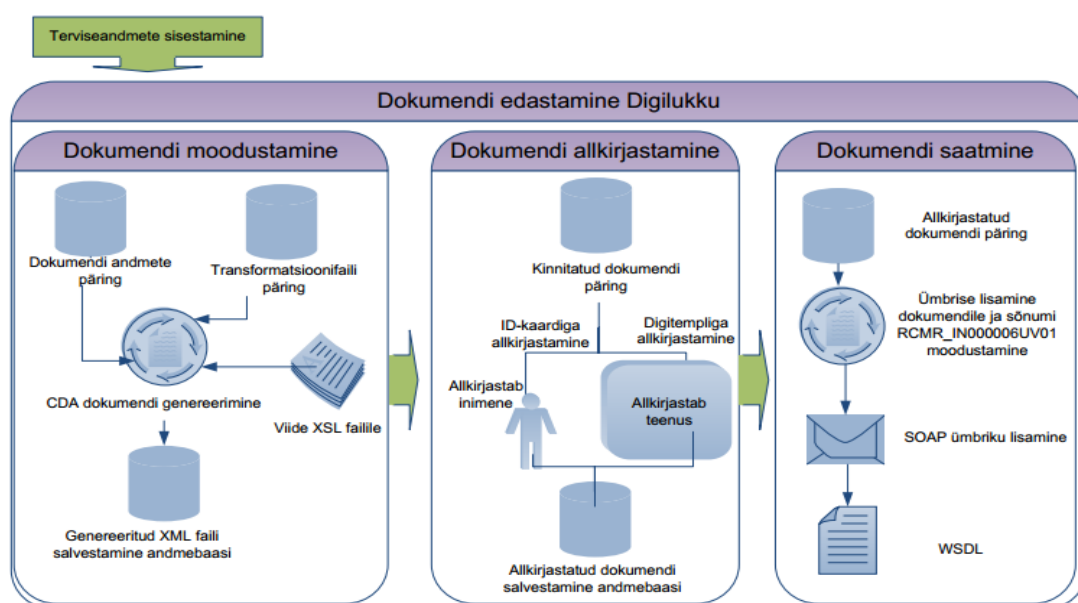
toimub infosüsteemis LIISA dokumentide edastamine Digilukku neid standardeid kasutades.

3 Terviseandmeid sisaldavate dokumentide edastamine Digilukku

LIISA Digilooga liidestatus võimaldab seda kasutavatel arstidel kiiresti ja mugavalt patsiendi terviseandmeid sisaldavaid dokumente edastada. Tänu digitaalsele saatmisele on need kohe kättesaadavad teistele erialaarstidele ja perearstile. See aitab ravi efektiivsust tõsta, hoiab aega kokku ja on keskkonnasõbralik. Samuti saab patsient ise neid dokumente patsiendiportaalist vaadata. Selles peatükis vaatame, kuidas infosüsteemis LIISA dokumente moodustatakse ja Digilukku saadetakse.

3.1 Ülevaade dokumendi edastamisest Digilukku

Infosüsteemi LIISA sisestatakse pidevalt patsiendi seisundiga seotud meditsiinilisi andmeid, nendeks on näiteks diagnoosid, uuringute tulemused, anamnees. Kusjuures laborianalüsaatorite väljundid liiguvad automaatselt LIISA andmebaasi. Sisestatud andmete põhjal genereeritakse erinevaid dokumente, mida on võimalik edastada Digilukku. Mõningateks LIISA infosüsteemist saadetavateks dokumentideks on saatekiri, tervisekontrolli teatis, digiretsept, epikriis (kokkuvõte patsiendi raviloost) ning uuringu vastus. Alljärgneval joonisel (joonis 1) on näidatud, mis sammud tehakse dokumendi moodustamiseks ja saatmiseks infosüsteemis LIISA. Järgnevates peatüki 3 alapeatükkides seletatakse lahti dokumendi moodustamine infosüsteemis LIISA, dokumendi allkirjastamine ja saatmine.



Joonis 1. Dokumentide edastamine Digilukku infosüsteemis LIISA

3.2 Dokumendi moodustamine

Kui dokumendi koostamiseks vajalikud patsiendi terviseandmed on programmi sisestatud ja kontrollitud ning enam neid ei muudeta, siis andmed kinnitatakse. Andmete kinnitamine tähendab sisuliselt CDA dokumendi genereerimist sisestatud andmete põhjal ning selle salvestamist andmebaasi.

Kuna kõigi CDA dokumentide moodustamine käib infosüsteemis LIISA sarnaselt, siis käesolevas peatükis vaatleme Digilukku saadetava uuringu vastuse dokumendi genereerimist, mille esimeseks sammuks on vajalike terviseandmete päring andmebaasist. Moodustatav CDA dokument vastab standardile „vastus saatekirjale“. [37]

3.2.1 CDA dokumendi andmete päring andmebaasist

Kuigi uuemates Microsoft SQL Serveri versioonides on XML töötamise tugi hea, siis suuremate ja keerukamate XML genereerimise korral peab arvestama serveri jõudlusega. Lisaks on HL7 standardile vastavad XML dokumendid piisavalt keerukad, et nende moodustamine andmebaasist ei pruugi olla võimalik.

Dokumendi moodustamiseks tagastatakse selle jaoks vajalikud terviseandmed andmebaasi päringuga võimalikult lihtsal kujul XML formaadis. Selleks on loodud erinevate dokumentide jaoks vastavad protseduurid (*stored procedure*), lõplik dokument pannakse kokku Delphi XE keskkonnas.

Dokumendi moodustamise näitena vaatame uuringu vastuse dokumendi genereerimist. Esimese sammuna kutsutakse Delphi XE keskkonnas välja SQL andmebaasi protseduur *pr_KoostaSaateKirjaVastusCDA*, millega tagastatakse CDA uuringu dokumendi koostamiseks vajalikud andmed XML osadena. Joonisel 2 on päring *pr_KoostaSaateKirjaVastusCDA* käivitatud Microsoft Management Studio liideses, kus on näha tagastatavad XML osad tunnuste kaupa. Programmis loetakse need osad kokku ühtseks XML failiks.

	(No column name)
1	<?xml version="1.0" encoding="UTF-8"?>
	(No column name)
1	<cda_SaatekirjaVastus xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="">
	XML_F52E2B61-18A1-11d1-B105-00805F49916B
1	<METADATA typeIdRoot="2.16.840.1.113883.1.3" typ...
	XML_F52E2B61-18A1-11d1-B105-00805F49916B
1	<RECORDTARGET recordTargetTC="RCT" realmCodeC="PA...
	XML_F52E2B61-18A1-11d1-B105-00805F49916B
1	<AUTHOR authorTC="AUT" assignedAuthorCC="ASSIGNE...
	XML_F52E2B61-18A1-11d1-B105-00805F49916B
1	<AUTHOR ORGANIZATION representedOrganizationCC="D...
	XML_F52E2B61-18A1-11d1-B105-00805F49916B
1	<CUSTODIAN custodianTC="CST" IDExtension="100476...
	XML_F52E2B61-18A1-11d1-B105-00805F49916B
1	<COMP_PROC componentTC="COMP" sectionCC="DOCSEC...
	XML_F52E2B61-18A1-11d1-B105-00805F49916B
1	<COMP_PROC_ENTRY entryTC="COMP" procedureCC="PRO...
	XML_F52E2B61-18A1-11d1-B105-00805F49916B
1	<COMP_QTSUS componentTC="COMP" sectionCC="DOCSEC...
	XML_F52E2B61-18A1-11d1-B105-00805F49916B
1	<COMP_MARKUS componentTC="COMP" sectionCC="DOCSEC...
	(No column name)
1	</cda_SaatekirjaVastus>

Joonis 2. Microsoft SQL Management Studio liideses käivitatud protseduuri pr_KoostaSaateKirjaVastusCDA tulemus.

Andmete päring on teostatud vastavalt koostatava CDA dokumendi struktuurile. Uuringu vastuse CDA dokumendis oleva sektsiooni `<recordTarget/>` andmete päring on järgmine:

```
SELECT * FROM (
    SELECT * FROM #PatsiendiAndmed
) RECORDTARGET FOR XML AUTO
```

Antud päringus on `#patsiendiAndmed` ajutine tabel (*temporary table*), kuhu patsiendi andmed on juba eelnevalt koondatud. SQL käsk `FOR XML AUTO` tagastab andmed XML formaadis [38] ja seda kasutatakse kõigi tagastatavate andmete puhul. Selle päringu tulemus on kujul:

```
<RECORDTARGET recordTargetTC="RCT" realmCodeC="PAT" patientRoleC="PAT"
Isikukood="45603210000" IsikukoodOID="1.3.6.1.4.1.28284.6.2.2.1"
patientCC="PSN" patientDC="INSTANCE" Eesnimi="GALINA" Perenimi="PÕDER"
Sugu="N" SuguCS="1.3.6.1.4.1.28284.6.2.3.16.1" SuguCSN="Sugu"
SuguDN="naine" Synniaeg="19560321" useVal="PHYS" Riik="EST"
Maakond="IDA-VIRUMAA" Vald="ASERI V." Aadress="a" Asukood="44154"
Telefon=""/>.
```

Keerulisemate või mahult suuremate andmekoosseisude tagastamine on jagatud mitmeks osaks. Allpool toodud uuringu vastuse CDA dokumendi sisuosa on jagatud kolmeks osaks. Eraldiseisva päringuga tagastatakse otsus *COMP_OT SUS*, märkus *COMP_MARKUS* ja teostatud uuringud, mis omakorda on jagatud veel kaheks. *COMP_PROC* sisaldab uuringu üldisi andmeid ning *COMP_PROC_ENTRY* uuringu üksikuid teenuseid:

```
<!-- SISUOSA ALGUS -->
  <component typeCode="COMP">
    <structuredBody classCode="DOCBODY" moodCode="EVN">
      <!-- TEOSTATUD UURINGUD ALGUS -->
      <component typeCode="COMP">
        <section classCode="DOCSECT" moodCode="EVN">
          ...
          <!-- uuringud ja protseduurid algus-->
          <!-- protseduur 1 algus-->
          <entry typeCode="COMP">
            ...
          </entry>
          <!-- protseduur 1 lõpp-->
          <!-- uuringud ja protseduurid lõpp-->
        </section>
      </component>
      <!-- UURINGUD LÕPP-->
      <!-- OTSUS ALGUS-->
      <component typeCode="COMP">
        ...
      </component>
      <!-- OTSUS LÕPP-->
      <!-- MÄRKUSED ALGUS-->
      <component typeCode="COMP">
        ...
      </component>
      <!-- MÄRKUSED LÕPP-->
    </structuredBody>
  </component>
<!-- SISUOSA LÕPP -->
```

Sellisel moel tagastatakse kõigi koostatavate CDA dokumentide andmed. Loodavate dokumentide sisuosa jagatakse loogiliselt seotud lõikudeks ning nende lõikude andmed tagastatakse andmebaasist eraldiseisvate XML osadena.

3.2.2 Transformatsioonifaili koostamine

Dokumendi koostamiseks andmebaasist tagastatud andmed töödeldakse lõpliku dokumendi kujule, kasutades selleks XSLT transformatsiooni. Muundefail XSLT on mõeldud XML faili transformeerimiseks teisele kujule. [3]

Käesolevas töös nimetatud XSLT failide loomiseks on kasutatud programmi Altova Mapforce. [39] See võimaldab vastendada ühe XML faili struktuuri teise struktuuriga ja genereerida selle põhjal muundefail.

Programmile Mapforce antakse ette kaks XML faili, milledest esimene on andmebaasist tagastatud andmete põhjal koostatud fail ja teine CDA dokument. Antud juhul on üheks XML failiks päringu *pr_KoostaSaateKirjaVastusCDA* vastuse põhjal koostatud XML fail ja teiseks „Vastus saatekirjale“ standardile vastav CDA dokument. [37]

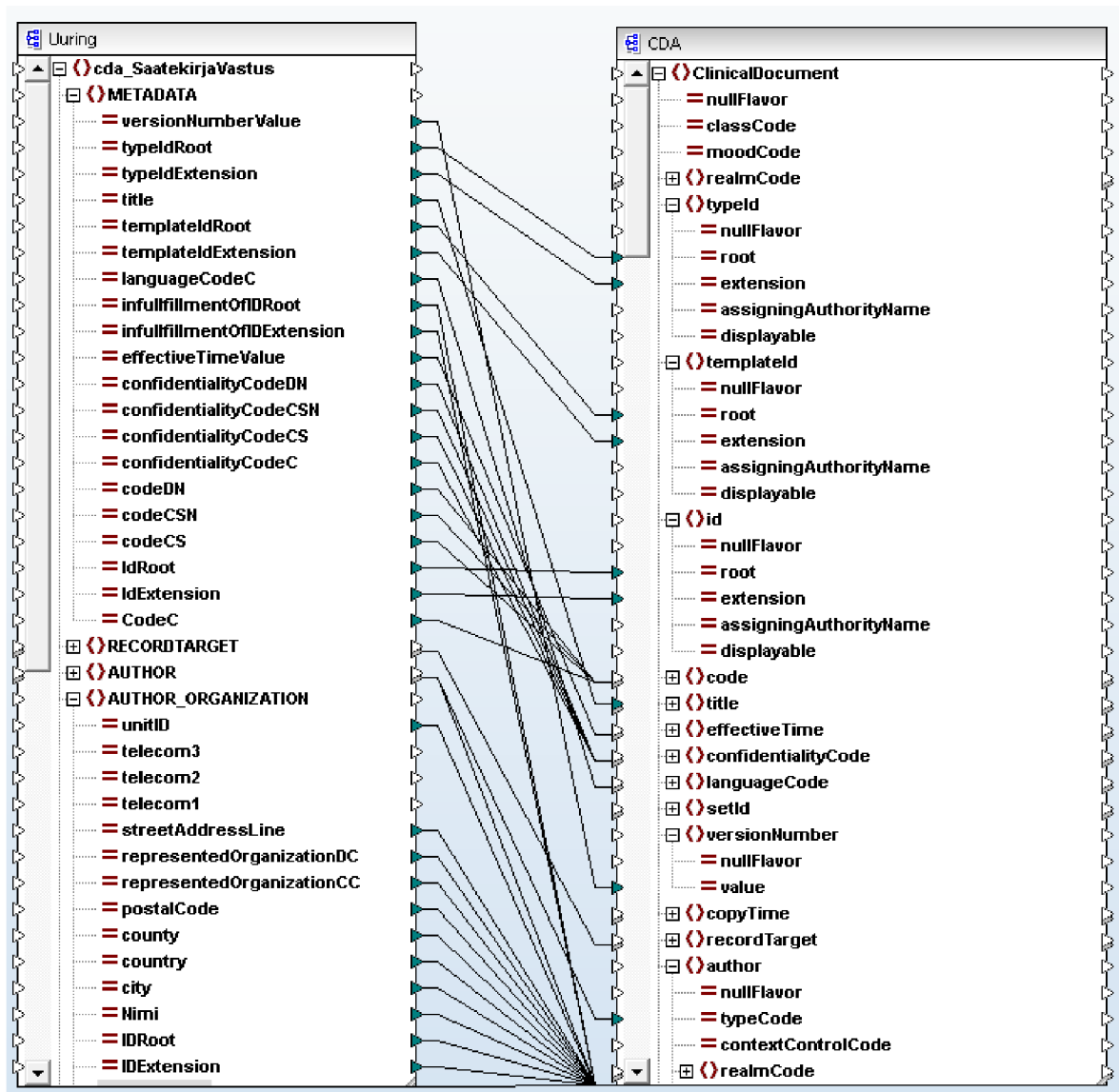
Mõlemad failid peavad sisaldama viidet XSD failidele, millega kirjeldatakse XML failide struktuur. Nii XML kui ka XSD failid on vajalikud programmi Mapforce kasutamiseks. Uuringu vastuse CDA dokumendi asukoha viide XSD failile on järgmine:

```
xsi:schemaLocation="urn:h17-org:v3 http://pub.e-  
tervis.ee/standards2/Schema/V3/HL7-ORG-V3-2005-  
NORMATIVE/infrastructure/cda/CDA.xsd"
```

Kuna protseduuri *pr_KoostaSaateKirjaVastusCDA* päringu vastusel pole selle struktuurile vastavat XSD faili, koostatakse see kasutades programmi Altova XMLSpy. [40] XMLSpy võimaldab skeemi definitsiooni faili automaatset genereerimist XML dokumendi struktuuri alusel. Loodud XSD faili asukoha viide lisatakse XML faili päisesse.

Genereeritud XSLT fail lisatakse andmebaasi, kust see dokumendi kinnitamise käigus välja kutsutakse.

Joonisel 3 on näidatud protseduuri *pr_KoostaSaateKirjaVastusCDA* tulemuse vastendamine CDA dokumendi struktuuriga kasutades programmi Mapforce.



Joonis 3. XML failide vastendamise.

3.2.3 Andmete transformeerimine CDA dokumendi kujule

Andmete transformeerimine toimub Delphi XE keskkonnas. Uuringu vastuse dokumendi koostamiseks loetakse andmebaasi protseduuri *pr_KoostaSaateKirjaVastusCDA* päringu vastus voogu (*stream*) *strmResults*. Seejärel luuakse XML dokumendi objekt *fxml* ning sellele omistatakse voo *strmResults* sisu:

```
fxml := CreateOLEObject('MSXML2.DOMDocument');
fxml.Async := false;
fxml.LoadXML(strmResults.ReadText);
```

fxml objektis olevad andmed viiakse CDA standardile vastavale kujule kasutades muundefaili. Uuringu vastuse dokumendi koostamiseks vajalik transformatsioonifail

xsltfail tagastatakse andmebaasi päringuga ning laetakse Delphi XE keskkonnas objekti *fxslt*:

```
fxslt := CreateOLEObject('MSXML2.FreeThreadedDOMDocument');
fxslt.Async := false;
fxslt.LoadXML(xsltfail);
```

Allpool on toodud lähteandmete muundamine CDA dokumendi kujule Delphi XE keskkonnas:

```
Template := CreateOLEObject('MSXML2.XSLTemplate');
Template.StyleSheet := fxslt;
ATStreamClass := TMemoryStream.Create;
Adapt := TStreamAdapter.Create(ATStreamClass);
Adapt.Seek(0, 0, tPos);
OutputStream := Adapt as IStream;
Processor := Template.CreateProcessor;
Processor.Input := fxml;
Processor.Output := OutputStream;
Processor.Transform;
docCDA := ComsDOMDocument.Create;
OutputStream.Seek(0, 0, tPos);
docCDA.load(OutputStream);
```

Kuna kõik andmed ja muundefail saadakse andmebaasist, siis vea leidmise korral andmete päringus või muundefailis võimaldab see lähenemine nende kiiret parandamist, samuti standardi muutuse korral muudatuste tegemist. Seega saab parandusi realiseerida otse andmebaasis, kliendile märkamatu ja programme välja vahetamata.

Näiteks *fxml* sektsiooni *<recordTarget/>* andmete kuju omandab pärast muundefailiga *fxslt* töötlemist HL7 CDA standardile vastava järgmise kuju:

```
<recordTarget typeCode="RCT">
  <realmCode code="PAT"/>
  <patientRole classCode="PAT">
    <id root="1.3.6.1.4.1.28284.6.2.2.1"
extension="45603210000"/>
    <addr use="PHYS">
      <country>EST</country>
      <state>ASERI V.</state>
      <county>IDA-VIRUMAA</county>
      <streetAddressLine>a</streetAddressLine>
      <unitID>44154</unitID>
    </addr>
  <patient classCode="PSN" determinerCode="INSTANCE">
    <name>
      <family>PÕDER</family>
      <given>GALINA</given>
    </name>
```

```

        <administrativeGenderCode code="N"
codeSystem="1.3.6.1.4.1.28284.6.2.3.16.1" codeSystemName="Sugu"
displayName="naine"/>
        <birthTime value="19560321"/>
    </patient>
</patientRole>
</recordTarget>

```

Transformatsiooni tulemusel saadud XML formaadis andmed moodustavad tervikliku CDA dokumendi, kuid see dokument on ainult masinloetav. Saadud tulemus on toodud välja bakalaureusetöö lisas 1. Inimloetavuse tagamiseks lisatakse dokumendi algusesse viide XSL failile.

3.2.4 Viide XSL failile

XML dokumendi loetavaks muutmiseks lisatakse faili algusesse viide XSL stiililehele. XSL fail peab olema avalikult kättesaadav ning seetõttu on koostatavale dokumendile soovitatav lisada viide E-tervise standardsele stiililehele:

```

<?xml-stylesheet type="text/xsl" href="http://pub.e
tervis.ee/standards2/Standards/3.0/XSL/main.xsl"?>

```

Uuringu vastuse kinnitamise käigus koostatud CDA dokument kodeeritakse UTF-8 vormingusse ja salvestatakse andmebaasi XML vormingus. Loodud dokument vaadatuna internetibrauseris on toodud välja bakalaureusetöö lisas 2.

3.3 Dokumendi allkirjastamine

Koostatud CDA dokumendile tuleb tagada tõestusväärus. Selleks on vaja dokument allkirjastada. Haigla infosüsteemis LIISA on selleks realiseeritud kaks moodust:

- ID-kaardiga – kasutaja kasutab dokumendi allkirjastamiseks isiklikku ID-kaarti. Nimetatud moodust kasutavad väiksemad asutused. Puudusteks võib pidada pidevat PIN-koodi sisestamist.
- Digitempliga – digitaalne tempel on mõeldud asutustele. Digitempliga allkirjastatud dokument kinnitab, et just konkreetne asutus on dokumendi autor. [41] Digitempel võimaldab allkirjastamist automatiseerida ja on seega kasutajale mugav.

Kui ID-kaardiga allkirjastamise tugi on integreeritud LIISA moodulitesse, siis digitempliga allkirjastamiseks on kirjutatud eraldi Windowsi teenus (*Windows service*). Teenus sooritab

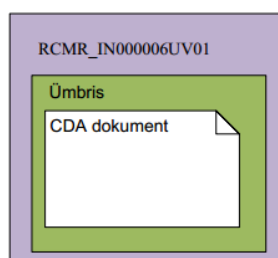
kindla intervalli tagant andmebaasi päringuid, kontrollides kinnitatud olekus dokumentide olemasolu ning leidmise korral allkirjastab need. Seega digitemplit kasutavate asutuste kasutajad ei pea ise allkirjastamisega vaeva nägema, allkirjastamine toimub automaatselt.

Allkirjastamiseks tehakse päring andmebaasi ning tagastatakse kinnitamise käigus koostatud dokument, kusjuures andmete kaitsmise huvides dokumenti arvuti kõvakettale ei salvestata. Tõestusväärtuse andmine LIISAs on realiseeritud DigiDoc COM teegi abil. [42] Windowsi teenus kasutab DigiDoc Java teeki *jdigidoc*. [43] Allkirjastatud dokument salvestatakse LIISA andmebaasi, asendades varem kinnitatud dokumendi.

3.4 Dokumendi saatmine Digilukku

Pärast dokumendi allkirjastamist saadetakse see Digilukku. Kui asutusel on kasutusel turvaserver ja digitaalne tempel, siis saatmine on automatiseeritud ning sellega tegeleb Windowsi teenus.

3.4.1 Dokumendi edastamiseks kasutatav sõnum



Joonis 4. Dokumendi lisamine sõnumisse *RCMR_IN000006UV01*.

Digilukku edastamiseks pakitakse dokument ümbrise sisse ja see omakorda konteiner-sõnumi sisse (joonis 4). Konteinersõnumina kasutatakse standardi „Staatuse muutmine koos sisuga“ sõnumit *RCMR_IN000006UV01* [44]:

```
<RCMR_IN000006UV01>  
  <templateId/>  
  <id/>  
  <creationTime/>  
  <versionCode/>  
  <interactionId/>  
  <processingCode/>  
  <processingModeCode/>  
  <acceptAckCode/>  
  <sequenceNumber/>
```



```

    <receiver/>
    <sender/>
    <controlActProcess>
        <authorOrPerformer/>
        <dataEnterer/>
        <subject>
            ... Dokumendi ümbris ...
        </subject>
    </controlActProcess>
</RCMR_IN000006UV01>

```

3.4.1.1 Dokumendi ümbris

Dokumendi edastamiseks Digilukku lisatakse sellele ümbris. Digilugu kasutab ümbrist dokumendi kohta info saamiseks. Allpool on toodud dokumendi saatmisel kasutatava ümbrise mall:

```

<clinicalDocument>
    <templateId/>
    <id/>
    <code/>
    <text>
        ... Dokument MIME vormingus ...
    </text>
    <statusCode/>
    <effectiveTime/>
    <availabilityTime/>
    <confidentialityCode/>
    <languageCode/>
    <versionNumber/>
    <recordTarget/>
    <author/>
    <custodian/>
</clinicalDocument>

```

Ümbrise lisamiseks dokumendile peab dokument olema MIME vormingus ja BASE64 kodeeringus. [44]

3.4.2 Ümbrise lisamine dokumendile

Andmebaasi sooritatakse päring, mis tagastab juba varem allkirjastatud XML formaadis dokumendi. See laetakse *IXMLDOMDocument* tüüpi objekti:

```

doc := ComsDOMDocument.Create;
doc.loadXML(xml);

```

Saatmiseks Digilukku lisatakse dokumendile ümbris, kasutades selleks andmebaasist laetud dokumentide edastamise sõnumi *RCMR_IN000006UV01* malli:

```

xmlTemp := ComsDOMDocument.Create;
xmlTemp.LoadXML(mall);

```

Delphi XE keskkonnas on ümbrise lisamiseks dokumendile koostatud meetod *LisaSonumileClinicalDocumentYmbriseAndmed*, mis saab parameetritena ette nii saadetava dokumendi *doc* kui ka malli *xmlTemp*. Kuna ümbrise päise osa on suures osas sama, mis CDA dokumendi päis, saab andmeväljad täita kopeerimise teel. Näiteks autori andmete kopeerimine on teostatud järgnevalt:

```
node:=
xmlTemp.documentElement.selectSingleNode('controlActProcess/authorOrPerformer');
node1 := node.selectSingleNode('assignedPerson/id');
node2:=
doc.documentElement.selectSingleNode('author/assignedAuthor/id').cloneNode(true);
nodeR := node.selectSingleNode('assignedPerson');
nodeR.replaceChild(node2, node1);
```

Kui päise andmed on kopeeritud, siis allkirjastatud CDA dokument kodeeritakse UTF-8 vormingusse ja seejärel BASE64 kodeeringusse ning lisatakse ümbrisesse:

```
xmlTemplate.documentElement.selectSingleNode('controlActProcess/subject/clinicalDocument/text').text:=
    #13#10 +
    'MIME-Version: 1.0'+ #13#10 +
    'content-type: ' + SisuTyyp + #13#10 +
    'content-transfer-encoding: base64' + #13#10 + #13#10 +
    MimeEncodeString(UTF8Encode(doc.xml));
```

Dokument on lisatud ümbrisesse ning saadetavat sõnumit enne Digilukku saatmist rohkem ei muudeta.

3.4.3 Dokumendi saatmine

Dokumendi sisaldav sõnum edastatakse Digilukku läbi turvaserveri, mida on vaja andmete vahetamiseks üle X-tee. Turvaserveriga suhtlemiseks pannakse saadetav sõnum SOAP ümbrikusse. [45] Kasutatav SOAP ümbriku näidis on saadaval veebiaadressil http://www.e-tervis.ee/images/stories/est/arendajatele/soap_naidis.pdf.

Käesolevas töös dokumentide saatmiseks arenduskeskkonda kasutatav SOAP ümbrik:

```
<SOAP-ENV:Envelope xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header xmlns:xtee="http://x-tee.riik.ee/xsd/xtee.xsd">
    <xtee:asutus>10047646</xtee:asutus>
    <xtee:allasutus>10047646</xtee:allasutus>
```

```

    <xtee:andmekogu>digilugu-xtee5</xtee:andmekogu>
    <xtee:isikukood>38700000000</xtee:isikukood>
    <xtee:id>9426BB48-4F24-4564-A39A-DA931997491B</xtee:id>
    <xtee:nimi>digilugu-xtee5.hl7.v1</xtee:nimi>
    <xtee:toimik/>
    <xtee:asynkroonne/>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <h17 xmlns="http://producers.digilugu-
xtee5.xtee.riik.ee/producer/digilugu-xtee5">
      <keha>
        <h17InputMessage xmlns="">
          ... Saadetak sõnum ...
        </h17InputMessage>
      </keha>
    </h17>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

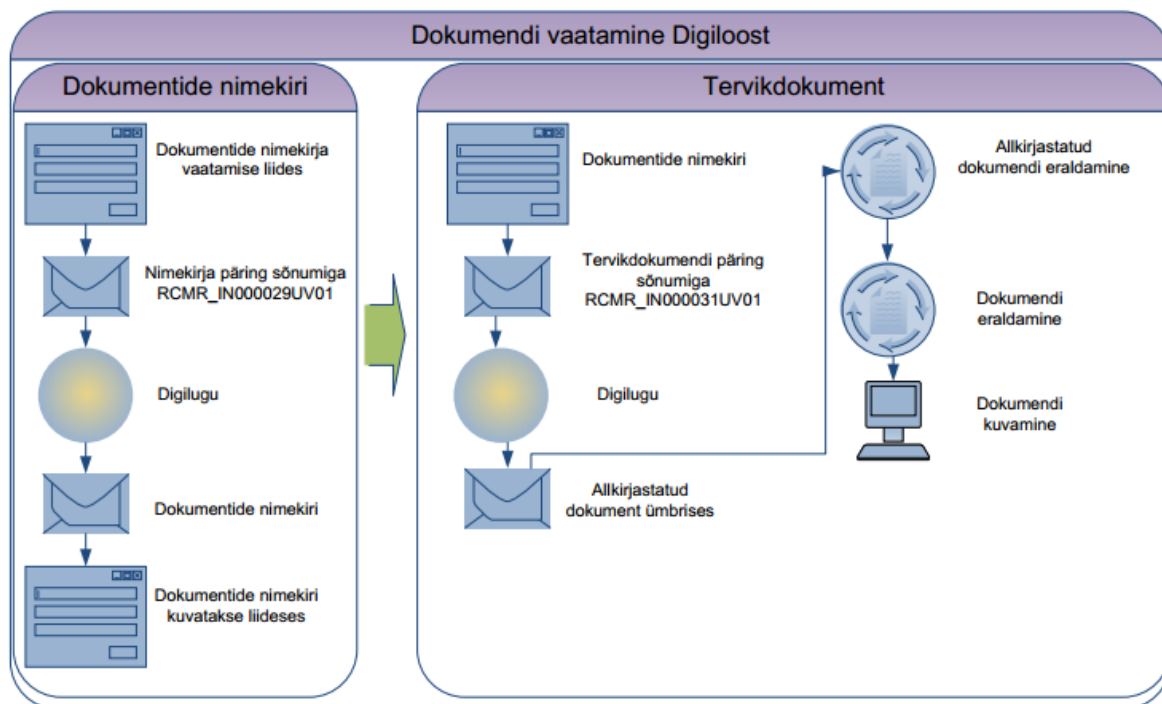
Digilugu saadab vastussõnumi, mis sisaldab teavet dokumendi aktsepteerimisest või tagasilükkamisest. Tagasilükkamise korral on kirjeldatud selle põhjus. LIISA andmebaasis märgitakse sõnumi olek vastavalt vastussõnumi sisule. Kui tagasilükkamise põhjuseks on tõrge või ei saada Digilooga üldse ühendust, siis üritatakse konkreetset sõnumit hiljem uuesti saata.

4 Dokumendi vaatamine Digiloost infosüsteemis LIISA

Infosüsteemis LIISA on võimalik tervishoiutöötajatel vaadata ka Digilukku saadetud dokumente, kuid ainult nende patsientide dokumente, kelle haiguslooga ollakse seotud. Vaadata saab siiski ainult neid dokumente, millele juurdepääsu pole patsient Patsiendiportaalis piiranud. Kõik päringud logitakse ning kõik vaatamised on näha Patsiendiportaalis. Käesolevas peatükis tutvume dokumentide vaatamise protsessiga arsti vaatevinklist, kasutades infosüsteemi LIISA.

4.1 Dokumentide vaatamise protsess

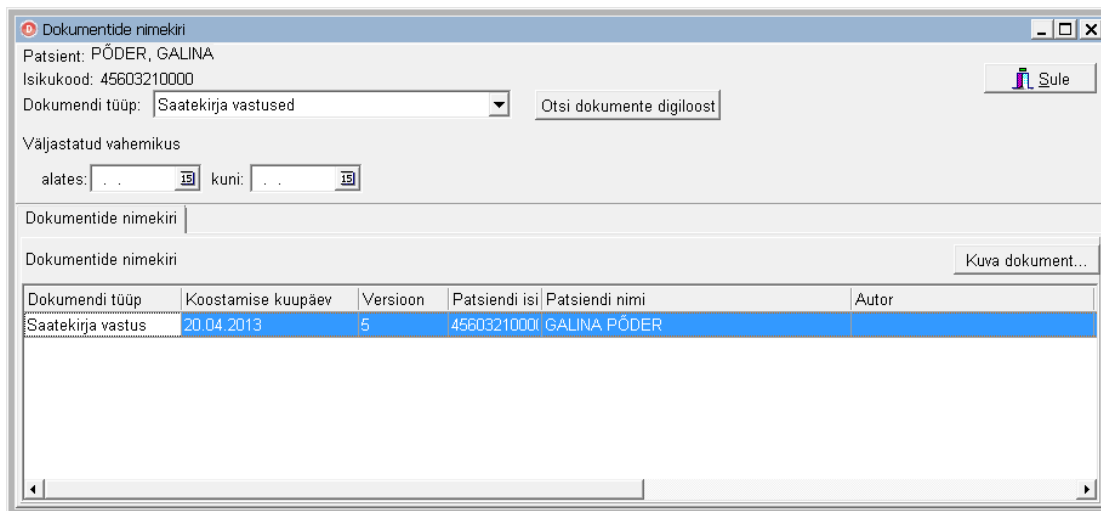
Dokumendi vaatamiseks tehakse kõigepealt Digilukku päring, mille vastus sisaldab dokumentide nimekirja. Seejärel valitakse nimekirjast sobiv kanne ning sooritatakse päring, mille vastus sisaldab konkreetset dokumenti. Viimase sammuna kuvatakse tagastatud dokument arvuti ekraanil. Joonis 5 kirjeldab protsessi tervikdokumendi vaatamiseks.



Joonis 5. Dokumendi vaatamise protsess infosüsteemis LIISA

4.2 Liides dokumentide nimekirja vaatamiseks Digiloost

Digilukku saadetud dokumentide nimekirja vaatamiseks infosüsteemis LIISA on eraldi liides (joonis 6).



Joonis 6. Liides Digiloo dokumentide pärimiseks.

Kuna kõikide dokumentide päring töötab sarnaselt, vaatleme ainult uuringu vastuse dokumentide vaatamist Digiloost. Joonisel 6 kujutatud liideses valitakse kõigepealt dokumendi tüüp „Saatekirja vastused“. Soovi korral määratakse ka dokumentide väljastamise ajavahemik. Seejärel vajutatakse nuppu „Otsi dokumente digiloost“. Programm kontrollib kasutaja õigust dokumentide nimekirja päringu teostamiseks ning ravisuhte olemasolu patsiendiga. Positiivse tulemuse korral teostatakse dokumentide nimekirja päring.

4.3 Dokumentide nimekirja päring

Dokumentide nimekirja päring on realiseeritud „dokumentide päring“ standardile vastavat sõnumit *RCMR_IN000029UV01* kasutades, [46] millega on võimalik pärida kõiki Digilukku saadetud dokumente. Uuringu dokumentide nimekiri tagastatakse sõnumis määratud parameetrite järgi:

```
<RCMR_IN000029UV01>
  ...
  <controlActProcess>
    <authorOrPerformer>
      ...
    </authorOrPerformer>
```

```

        <dataEnterer>
            ...
        </dataEnterer>
        <queryByParameter
xsi:type="ext:RCMR_MT00003UV_Extension.QueryByParameter" ext:HL7-
Domain="RCMR_RM000000" ext:realmCode="EE">
            <queryId root="1.3.6.1.4.1.28284.8.6.2.8"
extension="DBF7E4BF-C894-4721-A8D9-07DA8E1D0C23"/>
            <clinicalDocument.code>
                <value code="64"
codeSystem="1.3.6.1.4.1.28284.6.2.1.3.1"/>
            </clinicalDocument.code>
            <patient.id>
                <value root="1.3.6.1.4.1.28284.6.2.2.1"
extension="45603210000"/>
            </patient.id>
            <serviceEvent.code>
                <value code="clinical_document"
codeSystem="1.3.6.1.4.1.28284.6.2.2.32.1"/>
            </serviceEvent.code>
        </queryByParameter>
    </controlActProcess>
</RCMR_IN000029UV01>

```

Päring saadetakse Digilukku punktis 3.4.3 toodud SOAP ümbrikus.

4.4 Dokumentide nimekirja päringu vastus

Nimekirja päringu vastussõnumis on loetletud patsiendi uuringu vastuste dokumendid

<ns:subject> märgiste vahel:

```

<ns:RCMR_IN000030UV01>
    ...
    <ns:controlActProcess>
        <ns:subject>
            <ns:clinicalDocument>
                ...
                <ns:id root="1.3.6.1.4.1.28284.8.6.2.2.2"
extension="379"/>
                <ns:code code="64"
codeSystem="1.3.6.1.4.1.28284.6.2.1.3.1" codeSystemName="Dokumendi tüüp"
displayName="Saatekirja vastus"/>
                ...
                <ns:recordTarget>
                    <ns:patient>
                        <ns:id
root="1.3.6.1.4.1.28284.6.2.2.1" extension="45603210000"/>
                    ...
                </ns:patient>
            </ns:recordTarget>
        ...
    </ns:controlActProcess>

```

```

        </ns:clinicalDocument>
    </ns:subject>
    <ns:subject>
        ...
    </ns:subject>
        ...
    </ns:controlActProcess>
</ns:RCMR_IN000030UV01>

```

Vastussõnumis olevad dokumendid loetakse massiivi ja kuvatakse joonisel 6 olevasse tabelisse.

4.5 Tervikdokumendi päring Digiloost

Tervikdokumendi päring teostatakse „tervikdokumendi päring“ standardit ja sõnumit *RCMR_IN000031UV01* kasutades. Konkreetse dokumendi päringu sooritamiseks valitakse joonisel 6 olevas tabelis dokument ja vajutatakse nupule „*Kuva dokument...*“. Koostatavas sõnumis määratakse sõnumi *RCMR_IN000029UV01* vastusega saadud dokumendi number ja OID, patsiendi isikukood ja OID ning viidaregistri kande tüüp. [47] Päring saadetakse Digilukku punktis 3.4.3 toodud SOAP ümbrikus.

4.6 Tervikdokumendi kuvamine

Tervikdokumendi päringu vastus tagastatakse „tervikdokumendi päringu vastus“ standardile [48] vastavas sõnumis *RCMR_IN000032UV01*. [49] Dokument tagastatakse punktis 3.4.1.1 kirjeldatud ümbrikus:

```

<ns:RCMR_IN000032UV01>
    ... Sõnumi päis ...
    <ns:controlActProcess>
    <ns:subject>
        ... Dokumendi ümbris ...
    </ns:subject>
    <ns:queryAck/>
    <ns:queryByParameter/>
    </ns:controlActProcess>
</ns:RCMR_IN000032UV01>

```

Delphi XE keskkonnas loetakse vastussõnum *IXMLDOMDocument* objekti *sVastusDok*. Kuna allkirjastatud dokument on ümbrikes BASE64 kodeeringus, siis dekodeeritakse see järgnevalt:

```
allkirjastatudDok:=
dekodeeriMime(sVastusDok.documentElement.selectSingleNode('controlActPro
cess/subject/clinicalDocument/text').text);
```

```
Function dekodeeriMime(mimeString : WideString) : WideString;
begin
    DoubleCarriageReturnPos := pos(WideString(#10#10),MimeString);
    Base64Position := Pos('base64', Ansistring(Mimestring));

    if (DoubleCarriageReturnPos > 0) and (Base64Position <
    DoubleCarriageReturnPos) then
        Result := MimeDecodeString(
            AnsiString(
                Copy(MimeString,DoubleCarriageReturnPos + 2,
                    length(Mimestring))
            )
        )
    else
        Result := MimeDecodeString(
            Trim(
                copy(
                    Ansistring(Mimestring),
                    Base64Position + 6,
                    Length(Ansistring(Mimestring))-1)
            )
        );
end;
```

Objekt *allkirjastatudDok* sisaldab allkirjastatud dokumenti:

```
<SignedDoc>
  <DataFile>
    ... CDA dokument ...
  </DataFile>
  <Signature>
    ... Allkiri ...
  </Signature>
</SignedDoc>
```

Kuna CDA dokument on BASE64 kodeeringus, dekodeeritakse see funktsiooni *dekodeeriMime* abil:

```
cdaDok := dekodeeriMime
(deMimeDoc.documentElement.selectSingleNode('DataFile').text);
```

Dokument salvestatakse arvuti kõvakettale ja kuvatakse vaikumisi määratud XML vaatamise programmis.

Kokkuvõte

Käesoleva töö eesmärgiks oli kirjeldada tervishoiuasutuse infosüsteemi liidestamist Tervise infosüsteemiga. Töö aluseks on autori poolt realiseeritud liidestumise lahendus.

Kui enne liidestamist Tervise infosüsteemiga puudus terviklik ülevaade patsiendi haiguslugudest, siis tänu liidestuse realiseerimisele sai infosüsteemi LIISA kasutavatel tervishoiuasutustel võimalikuks tutvuda ka teistes tervishoiuasutustes haigusloo raames koostatud dokumentidega. Samuti sai võimalikuks dokumente ise Tervise infosüsteemi edastada. Dokumentide edastamine Tervise infosüsteemi parandas aegkriitilise informatsiooni kättesaadavust. Samuti avanes patsiendil võimalus Patsiendiportaalil enda kohta kirjutatud ravidokumentidega tutvuda.

Töö tulemuseks on ülevaade infosüsteemi LIISA ja Tervise infosüsteemi liidestamise teostamisest. Töös toodi välja selle realiseerimiseks kasutatud tehnoloogiad ning programmeerimisvõtted. Kirjeldati samm-sammult dokumentide edastamise toiminguid, alustades selle moodustamisest kuni saatmiseni Tervise infosüsteemi. Kuna kõikide dokumentide edastamise ahel on LIISAs ühelaadne, siis kirjeldati ainult uuringu vastuse dokumendi edastamist. Samuti kirjeldati saadetud dokumendi pärimist Tervise infosüsteemist ning selle kasutajale kuvamist.

Liidestamise realiseerimise üheks raskemaks osaks oli selle teostamisest tervikpildi saamine. Seda raskendas asjaolu, et autor polnud varem kasutanud tehnoloogiaid, mida kasutatakse andmete edastamiseks üle interneti. Samuti osutus aeganõudvaks HL7 standarditest aru saamine mahuka dokumentatsiooni tõttu.

Tervise infosüsteemi arendamine ei ole lõppenud. Avaldatakse uusi dokumentide standardeid ning muudetakse olemasolevaid. Samuti arendatakse edasi infosüsteemi LIISA, realiseerides avaldatud standardite uusi võimalusi.

Interfacing hospital information system LIISA with the Health Information System

Bachelor thesis

Rain Ojavee

Summary

The aim of this bachelor thesis is to give an overview of interfacing hospital information system with the Health Information System. This work is based on actual implementation done by the author.

The result of this work is a detailed overview of interfacing the Health Information System and hospital information system LIISA. It describes how documents are generated and prepared in LIISA and sent to the Health Information System. It also shows how sent documents are queried and displayed to users.

First part of this bachelor thesis is purely theoretical and gives small overview of the Health Information System and information system LIISA. It also provides a brief introduction to object identifiers and HL7 standards, which are used for generating documents as well as sending and receiving data.

Second part is more practical and includes explicit examples from LIISA. It presents an overview of how documents are generated and forwarded to the Health Information System as well as how they are queried and displayed to the user. Documents generation and forwarding process begins by querying data from database followed by data transformation. After that it is described how adding a digital signature and sending the document to the Health Information System works in LIISA. Querying and displaying documents begins by creating a query to receive a list of sent documents. In the end it is shown how to query a certain document and display it to the user.

Neither development of the Health Information System nor LIISA is finished. Many new standards are published and changes made to current ones, which makes implementing them to LIISA a work in progress.

Kasutatud kirjandus

1. Extensible Markup Language (XML) 1.0 (Fifth Edition)
<http://www.w3.org/TR/2008/REC-xml-20081126> [01.05.2013]
2. Extensible Stylesheet Language (XSL) Version 1.1
<http://www.w3.org/TR/2006/REC-xsl11-20061205> [01.05.2013]
3. XSL Transformations (XSLT) Version 2.0
<http://www.w3.org/TR/2007/REC-xslt20-20070123> [01.05.2013]
4. XML Schema
<http://www.w3.org/standards/xml/schema> [01.05.2013]
5. SOAP
http://www.w3schools.com/w3c/w3c_soap.asp [01.05.2013]
6. The Base16, Base32, and Base64 Data Encodings
<http://tools.ietf.org/html/rfc4648> [01.05.2013]
7. Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies
<http://tools.ietf.org/html/rfc2045> [01.05.2013]
8. About HL7
<http://www.hl7.org/about/index.cfm?ref=common> [04.05.2013]
9. Template
<http://www.thefreedictionary.com/template> [04.05.2013]
10. Haiguslugu
<http://www.eki.ee/dict/qs/index.cgi?Q=haiguslugu&F=M> [04.05.2013]
11. Eesti üleriigilised e-tervise projektid
http://www.digilugu.ee/dl/pdf/etervis_12_11a.pdf [04.05.2013]
12. Tervise infosüsteemi põhimäärus
<https://www.riigiteataja.ee/akt/13015769> [04.05.2013]
13. E-tervise projektid
<http://www.digilugu.ee/portal/page/portal/Digilugu/ETerviseProjektid> [04.05.2013]
14. Digitaalse tervise loo infosüsteem
http://www.e-tervis.ee/images/stories/est/visioonidokumendid/dl_is_tutvustus_06.02.14.pdf
[04.05.2013]

15. Digiloo süsteemi turvalisus
<http://www.digilugu.ee/portal/page/portal/Digilugu/TehnilineLahendus/SusteemiTurvalisus> [04.05.2013]
16. Standardite ja klassifikaatorite publitseerimiskeskus
<http://pub.e-tervis.ee> [04.05.2013]
17. Digiloo, Digipildi ja Digiregistratuuri standardid versioon 3.0
<http://pub.e-tervis.ee/standards2/Standards/3.0> [04.05.2013]
18. Digiloo, Digipildi ja Digiregistratuuri arendatavad standardid
<http://pub.e-tervis.ee/devstandard/devStandards/4.0> [04.05.2013]
19. Haigla infosüsteem LIISA
<http://www.medisoft.ee/index.php?id=9> [04.05.2013]
20. Pildiviit
http://pub.e-tervis.ee/standards2/Standards/3.0/DP/ST_Pildiviit.doc [04.05.2013]
21. Tervise infosüsteemi tehniline informatsioon
<http://www.e-tervis.ee/index.php/et/2012-07-22-09-20-40/tehniline-informatsioon>
[04.05.2013]
22. Infosüsteemide andmevahetuskiht
<https://www.riigiteataja.ee/akt/12956835?leiaKehtiv> [04.05.2013]
23. Andmevahetuskiht X-tee tarkvara
<https://www.ria.ee/x-tee-tarkvara/> [04.05.2013]
24. Digiloo sõnumivahetus
<http://www.digilugu.ee/portal/page/portal/Digilugu/TehnilineLahendus/Sonumivahetus> [04.05.2013]
25. Introduction to HL 7 Standards
<http://www.hl7.org/implement/standards/index.cfm?ref=nav> [04.05.2013]
26. HL7 explained
<http://www.ehi.co.uk/insight/analysis/89/hl7-explained> [04.05.2013]
27. The HL7V3.0 Primer
<http://healthcareinformatics3000feet.blogspot.com/2008/01/hl7v30-primer.html>
[04.05.2013]
28. HL7 Version 3 Product Suite
http://www.hl7.org/implement/standards/product_brief.cfm?product_id=186
[04.05.2013]

29. CDA Release 2
http://www.hl7.org/implement/standards/product_brief.cfm?product_id=7
[04.05.2013]
30. HL7 CDA
<http://iehr.eu/standards/hl7-cda> [04.05.2013]
31. OID standardi kasutamine Eesti tervishoius
<http://pub.e-tervis.ee/manuals/ISO%20OID/1.0/1.0.pdf> [04.05.2013]
32. HL7 Implementation Guide: For Simple CDA Release 2 Documents
<http://wiki.siframework.org/file/view/CDAQuickStart.doc/331063378/CDAQuickStart.doc> [04.05.2013]
33. Information technology – Procedures for the operation of object identifier registration authorities: General procedures and top arcs of the international object identifier tree
http://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-X.660-201107-I!!PDF-E&type=items [04.05.2013]
34. The String Representation of Standard Attribute Syntaxes
<http://www.ietf.org/rfc/rfc1778.txt> paragrahv 2.15 [04.05.2013]
35. Internet Assigned Numbers Authority
<http://www.iana.org> [04.05.2013]
36. Standardite ja klassifikaatorite publitseerimiskeskuse OID-sektsioon
<http://pub.e-tervis.ee/oids.py> [04.05.2013]
37. Digiregistratuur (DR) Vastus saatekirjale standard
<http://pub.e-tervis.ee/standards2/Standards/3.0/DR/doc/DR-KI-6%20Vastus%20saatekirjale.doc> [04.05.2013]
38. Use AUTO Mode with FOR XML
<http://msdn.microsoft.com/en-us/library/ms188273.aspx> [04.05.2013]
39. Altova MapForce
<http://www.altova.com/mapforce.html> [04.05.2013]
40. Altova XMLSpy
<http://www.altova.com/xmlspy.html> [04.05.2013]
41. Digitaalne tempel
<https://www.sk.ee/teenused/digitempli-teenus> [04.05.2013]
42. DigiDoc COM teek
<http://www.id.ee/index.php?id=30392> [04.05.2013]

43. Digidoc Java teek jdigidoc
<http://www.id.ee/index.php?id=30393> [04.05.2013]
44. Staatuse muutmine koos sisuga standard
http://pub.e-tervis.ee/standards2/Standards/3.0/DL/ST_Staatuse%20muutmine%20koos%20sisuga.doc [04.05.2013]
45. X-tee nõuded infosüsteemidele ja adapterserveritele Versioon 8.7
http://x-road.ee/docs/est/nouded_infosusteemidele_ja_adapterserveritele.pdf
[04.05.2013]
46. Dokumentide päringu standard
http://pub.e-tervis.ee/standards2/Standards/3.0/DL/ST_Dokumentide%20paring.doc
[04.05.2013]
47. Tervikdokumendi päringu standard
http://pub.e-tervis.ee/standards2/Standards/3.0/DL/ST_Tervikdokumendi%20paring.doc
[04.05.2013]
48. Tervikdokumendi päringu vastuse standard
http://pub.e-tervis.ee/standards2/Standards/3.0/DL/ST_Tervikdokumendi%20paringu%20vastus.doc [04.05.2013]
49. Tervikdokumendi päringu vastuse RCMR_IN000032UV01 näidissõnum
http://pub.e-tervis.ee/standards2/Standards/3.0/DL/XML/RCMR_IN000032UV01_tervikdokumendi_paringu_vastus.xml [04.05.2013]

Lisad

Lisa 1 – Uuringu vastuse CDA dokument XML kujul

```
<?xml version="1.0"?>
<ClinicalDocument xsi:schemaLocation="urn:hl7-org:v3 http://pub.e-
tervis.ee/standards2/Schema/V3/HL7-ORG-V3-2005-
NORMATIVE/infrastructure/cda/CDA.xsd" xmlns="urn:hl7-org:v3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <typeId root="2.16.840.1.113883.1.3" extension="POCD_HD000040"/>
  <templateId root="1.3.6.1.4.1.28284.6.1.1"
extension="1.3.6.1.4.1.28284.6.1.1.64.1.1"/>
  <id root="1.3.6.1.4.1.28284.8.6.2.2.2" extension="379"/>
  <code code="64" codeSystem="1.3.6.1.4.1.28284.6.2.1.3.1"
codeSystemName="Dokumendi tüüp" displayName="Saatekirja vastus"/>
  <title>Saatekirja vastus</title>
  <effectiveTime value="20130420195801"/>
  <confidentialityCode code="N" codeSystem="2.16.840.1.113883.5.25"
codeSystemName="Konfidentsiaalsus" displayName="Normaalne">
    <translation code="N"
codeSystem="1.3.6.1.4.1.28284.6.2.2.37.1"
codeSystemName="Konfidentsiaalsus eestkostjale"
displayName="Normaalne"/>
    <translation code="N"
codeSystem="1.3.6.1.4.1.28284.6.2.2.39.1"
codeSystemName="Konfidentsiaalsus arstile" displayName="Normaalne"/>
  </confidentialityCode>
  <languageCode code="EST"/>
  <versionNumber value="1"/>
  <recordTarget typeCode="RCT">
    <realmCode code="PAT"/>
    <patientRole classCode="PAT">
      <id root="1.3.6.1.4.1.28284.6.2.2.1"
extension="45603210000"/>
      <addr use="PHYS">
        <country>EST</country>
        <state>ASERI V.</state>
        <county>IDA-VIRUMAA</county>
        <streetAddressLine>a</streetAddressLine>
        <unitID>44154</unitID>
      </addr>
      <patient classCode="PSN" determinerCode="INSTANCE">
        <name>
          <family>PÖDER</family>
          <given>GALINA</given>
        </name>
        <administrativeGenderCode code="N"
codeSystem="1.3.6.1.4.1.28284.6.2.3.16.1" codeSystemName="Sugu"
displayName="naine"/>
        <birthTime value="19560321"/>
      </patient>
    </patientRole>
  </recordTarget>
</ClinicalDocument>
```

```

</recordTarget>
<author typeCode="AUT">
  <time value="201304201958"/>
  <assignedAuthor classCode="ASSIGNED">
    <id root="1.3.6.1.4.1.28284.6.2.4.9"
extension="D09999"/>
    <code code="DOCTOR"
codeSystem="1.3.6.1.4.1.28284.6.2.2.15.1"
codeSystemName="Tervishoiutöötajate tüüp" displayName="arst"/>
    <assignedPerson classCode="PSN"
determinerCode="INSTANCE">
      <name>
        <family>OJAVEE</family>
        <given>RAIN</given>
      </name>
    </assignedPerson>
    <representedOrganization classCode="ORG"
determinerCode="INSTANCE">
      <id root="1.3.6.1.4.1.28284.4"
extension="10047646"/>
      <name>Medisoft Test</name>
      <telecom value="mailto:ltk@ltk.eeeee"/>
      <telecom value="fax:+372 650 8686"/>
      <telecom value="tel:6977-111"/>
      <addr use="PHYS">
        <country>EST</country>
        <county>HARJUMAA</county>
        <city>AEGVIIDU AS.</city>
        <postalCode>10611</postalCode>
        <streetAddressLine>Sõle 16,
Tallinn</streetAddressLine>
        <unitID>37112</unitID>
      </addr>
    </representedOrganization>
  </assignedAuthor>
</author>
<custodian typeCode="CST">
  <assignedCustodian>
    <representedCustodianOrganization>
      <id root="1.3.6.1.4.1.28284.4"
extension="10047646"/>
      <name>Medisoft Test</name>
    </representedCustodianOrganization>
  </assignedCustodian>
</custodian>
<component typeCode="COMP">
  <structuredBody classCode="DOCBODY" moodCode="EVN">
    <component typeCode="COMP">
      <section classCode="DOCSECT" moodCode="EVN">
        <code code="PROC"
codeSystem="1.3.6.1.4.1.28284.6.2.2.11.1" codeSystemName="Sektsooni
kodeering" displayName="Uuringud ja protseduurid"/>
        <title>Uuringud ja protseduurid</title>
        <text>

```



```

                <paragraph>
                    <content> Kuupäev: 10.04.2013
11:44 Kood ja nimetus: 7903 Röntgeniülesvõte rindkere piirkonnast
(üks ülesvõte)

Põhjendus: J20.9 Täpsustamata äge bronhiit</content>
                </paragraph>
            </text>
            <entry typeCode="COMP">
                <procedure classCode="PROC"
moodCode="EVN">
                    <code code="7903"
codeSystem="1.3.6.1.4.1.28284.6.2.1.6.1" codeSystemName="Haigekassa
hinnakirjakoodid" displayName="Röntgeniülesvõte rindkere piirkonnast
(üks ülesvõte)"/>
                    <text>Põhjendus: J20.9
Täpsustamata äge bronhiit
                </text>
                <effectiveTime
value="20130410"/>
            </procedure>
        </entry>
    </section>
</component>
<component typeCode="COMP">
    <section classCode="DOCSECT" moodCode="EVN">
        <code code="DEC"
codeSystem="1.3.6.1.4.1.28284.6.2.2.11.1" codeSystemName="Sektsiooni
kodeering" displayName="Otsus"/>
        <title>Otsus</title>
        <text styleCode="formatted">
            <paragraph ID="DECSUM">
                <content>Kirjeldus: RÖ
RINDKEREST OTSESUUNAS
Süda eale vastava konfiguratsiooniga, kompensatsioonis. Ülamediastiinum
tavalise laiusega. Diafragmakuplid kumerad ja selgepiirilised. Kopsude
õhustatus tavaline.

Järeldus: Infiltratiivseid muutusi ei ole.</content>
            </paragraph>
        </text>
    </section>
</component>
</structuredBody>
</component>
</ClinicalDocument>

```

Lisa 2 – Uuringu vastuse CDA dokument inimloetaval kujul

Saatekirja vastus nr 379 versioon 1

Tervishoiuasutus Medisoft Test

Registreerimiskood 10047646

Aadress Sõle 16, Tallinn, AEGVIIDU AS., 10611 HARJUMAA, Eesti

Telefon 6977-111

E-post Itkh@Itkh.eeeee

Faks +372 650 8686

Koostaja

Nimi RAIN OJAVEE Kood D09999

Patsient

Isikukood 45603210000

Eesnimi GALINA

Perekonnanimi PÖDER

Sünniaeg 21.03.1956

Sugu naine

Elukoht a, IDA-VIRUMAA, Eesti

Saatekirja tellimuse nr.

Uuringud ja protseduurid

Kuupäev	HK Hinnakirja kood
10.04.2013	7903 - Röntgeniülesvõtte rindkere piirkonnast (üks ülesvõtte)
Põhjendus: J20.9 Täpsustamata äge bronhiit	

Otsus

Kirjeldus: RÖ RINDKEREST OTSESUUNAS Süda eale vastava konfiguratsiooniga, kompensatsioonis. Ülamediaastiinum tavalise laiusega. Diafragma kuplid kumerad ja selgepiirilised. Kopsude õhustatus tavaline. Järeldus: Infiltratiivseid muutusi ei ole.

Dokument koostatud 20.04.2013 19:58:01

Dokumendi keel EST

Konfidentsiaalsus:

Patsiendile Normaalne

Eestkostjale/usaldusisikule Normaalne

Meditsiinitöötajale Normaalne

Vastutaja

Asutuse/isiku nimi Medisoft Test

Kood 10047646

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina Rain Ojavee _____
(*autori nimi*)
(sünnikuupäev: 19.05.1987 _____)

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose
Tervise infosüsteemiga liidestumine haigla infosüsteemi LIISA näitel

(*lõputöö pealkiri*)

mille juhendaja on Vambola Leping _____
(*juhendaja nimi*)

- 1.1.reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace-is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
 - 1.2.üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace'i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.
 3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tartus, **09.05.2013**