Design Patterns for Smart Home Systems Development

Assist. Prof. PhD Mariya Armyanova
University of Economics - Varna, Varna, Bulgaria
armianova@ue-varna.bg

Abstract

The information technology is increasingly entering the household and allowing the different devices integration into smart home systems. IoT (Internet of things) provides interoperability and the ability to control devices through a global network. The "smart" devices used in the environment shaping for smart home systems utilize large amounts of data that need storage, distribution and computation. There are numerous problems with the processing and security of large data sets. The design patterns can be used to solve the whole variety of emerging issues. The purpose of the report is to identify and classify the design patterns that allow different processes and devices collaboration on smart home systems.

Keywords: IoT, Smart home, Design patterns, Security, Interactions, Communication

JEL Code: C88; doi:10.36997/IJUSV-ESS/2019.8.2.56

Въведение

Информационните технологии все повече навлизат в бита и позволяват обединяване на различни устройства в smart home системи. Понятието smart home системи се появява, като следствие на концепцията за интелигентна среда и сграда. IoT (Internet of Things) осигурява взаимодействието и възможността устройствата да се управляват с помощта на глобалната мрежа. Включва технология за мрежова комуникация, интелигентни уреди и обзавеждане за дома, компютърна, аудио и видео технология, и технология за сигурност. Smart home системите се стремят да направят дома по-комфортен, сигурен и екологичен. Може да се обобщи, че при създаването на smart home системи се използват нова концепция за дизайн и метод за проектиране, които се базират на технологията IoT (Nawaz et al., 2014), но я специализират за интелигентния контрол над дома.

1. Същност и цели на интелигентния дом

Smart home системите преследват различни цели, като енергоспестяване, безопасност, комфорт, дори опазване на околната среда. Отделните устройства се управляват от потребителите посредством мрежовата среда, като за целта се създава прост и удобен интерфейс. В същото време данните от сензорите се събират и обработват, за да позволят на база на натрупания опит по-ефективно управление и контрол над устройствата. Обработката на данните се реализира в облачните платформи, като се основава на технологиите big data, data lake, интелигентни анализи и изкуствен интелект. Разработването на smart home система изисква задълбочени, експертни познания в множество технологии, за да се постигнат всички поставени цели и задачи. От тях се очаква да реализират интелигентен контрол над различни части от съвременните жилища. Задачите на този клас системи са разнообразни и са свързани с осигуряване на комфортна, екологична и безопасна среда за живот. Групите приложения на новите технологии могат да се обобщят като:

- Контрол на осветлението;
- Контрол над електрическите уреди;
- Осигуряване на сигурността на дома.

Една от основните задачи на интелигентния контрол над дома е осигуряване на интелигентен контрол върху осветлението и завесите. Той се реализира с централизирано управление, което е дистанционно и позволява персонални настройки. Така се постига и по-голяма сигурност на дома.

Съвременните уреди, като телевизори, климатици, прахосмукачки, бойлери, позволяват интелигентен контрол чрез дистанционно управление и персонални настройки. Те се управляват през глобалната мрежа, като изпращат събраните данни в различни облачни системи, които откриват най-ефективния начин за работа на устройството.

Сигурността са постига с автоматично уведомление при възникване на сигнал от сензорите за наблюдение над събития, като кражба, грабеж, пожар, изтичане на газ или друга аварийна ситуация. Освен това интелигентно се управляват самите уреди, които могат да се включат или изключат при възникване на аварийната ситуация. Така се постига по-голяма сигурността както над собствеността, така и над личната безопасност.

От основните приложения на Smart home системите могат да се определят основните изисквания към тях. Те се свързани с осигуряване на удобството, сигурността, лесното им развитие и поддръжка, и потребителски ориентирана визуализация на данните.

Концепцията за интелигентния дом възниква с идеята за осигуряване на по-комфортна, безопасна, удобна и ефективна среда за живот. Smart home системите се стремят да осигурят спокойствие на потребителите, което се постига и въз основа на някои психологически прийоми. На потребителите не се предава информацията от сензорите, която може да е объркваща и неясна, а се използва интуитивен графичен интерфейс, който лесно да се възприема и използва. Освен това на потребителите се осигурява възможност за непрекъснат достъп до данните от уредите, както и възможност за непрекъснато им управление. Потребителите сами могат да избират начина и количеството на информацията, които желаят да получават.

Важна характеристика на smart home системите е способността им да работят без прекъсване двайсет и четири часа в денонощието. Целта е да се гарантира постоянно надеждното им функциониране, като така се гарантира постигането на определно ниво на сигурност. Всички системи следва да имат предвидени мерки за работа при възникване на извънредни ситуации, като липса на захранване или други промени в заобикалящата ги среда. Следва да се прави резервно копие на системата, така че тя да се възстанови при срив. Затова се предприемат мерки за гарантиране на непрекъснатата нормална и безопасна работа, качество и производителност на системата. В същото време прехвърлянето на големите данни, генерирани от сензорите, през глобалната мрежа се сблъсква със сериозни заплахи за сигурността им. Затова следва smart home системите да се придържат към следните принципи, които гарантират безопасността на събраните потребителски данни:

- Системите следва непрекъснато да подобряват нивото на безопасност и защита на данните;
- Непрекъснато следва да се подобрява технологията за защита на данните от хардуерни и софтуерни заплахи;
- Следва да се създаде система за чувствителен мониторинг върху използването на събраните данни;
- Придържане към стандарти за мрежова сигурност при различни обстоятелства;
- Работа с организациите и органите, които прилагат стандартите за сигурност и гарантират постигане на определени нива на сигурност.

Друга важна характеристика на smart home системите е поддържането на възможностите за лесно им разширяване и промяна, посредством осигуряването на стандартизацията в използваната технология и архитектура. За да се гарантира взаимосвързаността и съвместимостта между различните уреди, участващи в системата, всички производители се придържат към TCP/IP мрежовата технология. Всички устройства, формиращи front end архитектурата на системата са разработени съгласно универсални, отворени стандарти, позволяващи лесно мащабиране (Nawaz et al., 2014). Така се гарантира, че системата е в състояние да се свърже и да си взаимодейства и с нови бъдещи устройства, създадени от други производители.

Поддръжка на smart home системата може да се осъществи дистанционно. Освен това непрекъснатото проследяване на натоварването на инсталацията, отстраняването на грешки и актуализацията на системата, е нереализируемо без възможностите на съвременните технологиите, които гарантират автоматичното поддържане на многобройните smart home системи на клиентите. Затова при изграждането на системите се осигурява възможност за автоматично диагностициране на повреди чрез дистанционната ѝ проверка (Chen et al., 2014). Актуализацията и настройката на новите версии на системата също се извършват дистанционно, което улеснява поддържането на системата, намалява времето за реакция и разходите за поддръжка.

Използването на big data и data lake технологиите позволяват работата с големите информационни масиви да остане скрита за потребителя. На потребителите се предоставя обобщено сечение на данните, което е лесно за възприемане и подпомага вземането на решения, така че да се подобри работата на системата и да се удовлетворят максимално желанията на потребителите. Визуализацията на данните следва да се адаптира към особеностите на различните обекти, участващи в системата и така лесно да се включват нови устройства. Данните се предоставят на потребителите многоизмерно, така че всеки потребител сам може да подбере начинът и обемът на данните, които са му най-ясни. Постепенно се надгражда познанието за системата, което позволява лесното усвояване на начина на работата с нея.

2. Тенденции в развитието на шаблоните за smart home системите

Създаването на smart home системи се базира на IoT технологиите. Internet of Things (IoT) се разглежда като концепция за обвързването на различни устройства с Интернет така, че да се осигури възможността им да бъдат разпознавани от други устройства. IoT обединява водещи технологии, като машинно обучение, изкуствен интелект, big data, data lake и интелигентни анализи. Всяка една технология поставя нови предизвикателства пред осигуряване на сигурността на системата. Например технологията data lake е добър подход за съхраняване на big data, но при неправилен дизайн и употреба носи много рискове свързани с качеството, сигурността и контрола на достъпа и използването им (Sulova, 2019). Затова разработването на smart home системи е съпроводено с решаването на голям брой разнообразни проблеми и прилагането на шаблоните за проектиране би могло да подпомогне работата на разработчиците. Често отделните подсистеми се разработват от независими екипи от разработчици и обединяването им в единна smart home система изисква съобразяване с възможността от възникване на множество потенциални проблеми. Едно възможно решение е използването на шаблоните за проектиране, които да позволят многократно използване на опита на разработчиците при решаването на даден проблем. Те могат да се разглеждат като описания на често срещани проблеми, техните абстрактни решения и последствията им. Използването на шаблоните до голяма предотвратява внасянето на неочаквани последствия в кода на системата при решаването на конкретния проблем.

Шаблоните за проектиране имат различни цели. Едни са свързани с реализиране на определена функционалност в системата, а други със съпровождането ѝ. За да позволят лесното развитие на smart home системите, те следва да отразяват тенденциите в развитието им. Шаблоните за проектиране, създадени за smart home системите, следва да поддържат очакваната им бъдеща функционалност. Очаква се основната посока на интелигентния дизайн на дома да бъде проектирането на операционна система, интегрирането на устройства на различни производители и включване на нови средствата за контрол и управление (Changhua, 2018).

Възниква необходимостта всяка подсистема да работи и самостоятелно без необходимост от непрекъснато управление от интерактивната платформа. Платформата,

осигуряваща интеграцията на smart home системата, има за задачи да събира данните от отделните подсистеми и да осигури взаимодействието им. Различните подсистеми, като алармата за сигурност, контрола на електрическите уреди и др. се очаква да могат да работят и при откъсване от Интернет мрежата. Следва да се гарантира, че и когато системите нямат достъп до платформата, те продължават да изпълняват поставените им задачи.

Чрез платформата следва да се осигури взаимодействието между устройствата на различните производители, които използват различни мрежови протоколи и информационни технологии. Платформата следва да осигури единен начин на потребителите да управляват различните подсистеми, като им осигури общи възможности за настройка. Чрез нея потребителят реализира дистанционен контрол и управление. Различните системи обменят през платформата данни, за да изпълнят ефективно поставените задачи от потребителя. Това налага тя да изпълнява и функцията на шлюз за различните подсистеми и устройства.

Smart home системите предоставят различни варианти за контрол и управление. За да се осигури непрекъснатият достъп до информация, е необходимо осигуряване на възможности за включване на различни интелигентни устройства, като личен цифров помощник, интелигентен дистанционен контролер, сензорен екран, мобилен телефон, таблет, компютър.

3. Критерии за класификация на шаблони за проектиране, подпомагащи разработването на Smart home системите

Smart home системите е динамично развиваща се област и затова непрекъснато се описват нови шаблони, които решават възникналите проблеми при разработването им. Все още не е направен опит за класификация на шаблоните, които решават проблемите на този клас системи, но съществуват опити да се класифицират шаблоните в областта на IoT. Проблемът е, че концепцията за IoT се използва в различни области, под понятието се разбират различни системи, като Smart Homes, Smart Office, Smart Grids, или концепцията Smart City, Индустриалния Интернет и др.

Шаблоните за проектиране, подпомагащи разработването на smart home системи, са малка част от всички шаблони, които се използват при разработването на системи. Те са подмножество получено при класифициране на шаблоните по критерия предназначение от общата схема за класификация (Armiyanova, 2018). Затова предлагаме за основен критерии за класификацията им критерият цел. Подходящ е и критерият слой на архитектурата, тъй като той подразделя шаблоните според етапите на разработка и при подобна класификация разработчикът лесно открива подходящ шаблон от съответното ниво.

Критерият цел, показва конкретния проблем, към чието решаване е насочен шаблона. Целта се определя от елементите на системата, на които влияе шаблонът: интеграция, функционалност, комуникация, достъп, сигурност и др. Това разделение се отнася не само до шаблоните за проектиране, а и до архитектурните шаблони, които също могат да добавят цели. Но тъй като описаните до момента архитектурни шаблони за само този тип системи – smart home не са толкова много, засега не е необходимо тяхното класифициране.

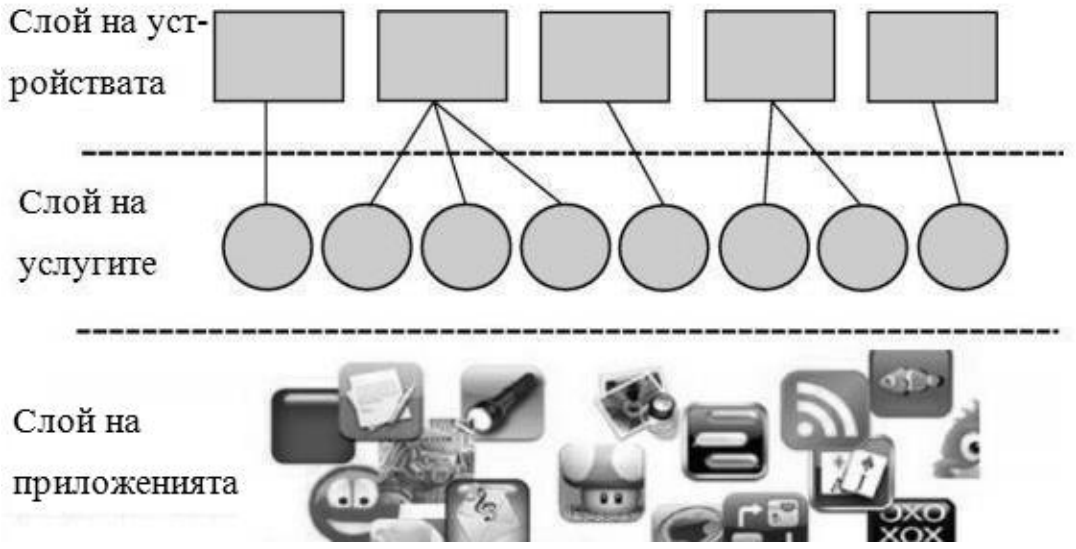
Съществуват класификации по критерия цел, но те се отнасят за цялото множество от IoT шаблони (Koster, 2014; Chandra et al., 2016). Обаче предложените от авторите категории са много, тъй като се класифицират всички IoT шаблони, подходящи за използване в IoT системите, а не само предназначените за smart home системите. Затова има някои категории, които могат да отпаднат, като например шаблоните за обработка на информацията. Те не са специфични и могат да се използват и при изграждането на различни облачни системи.

Шаблоните за smart home системите могат да се групират в две основни групи – отговарящи за потребителския интерфейс и за back-end нивото на архитектурата на системите. Според критерия цел шаблоните следва да се разделят според основните групи проблеми, които решават. Проблемите се обобщят в следните групи:

- Избор на подходяща архитектура за IoT приложенията, която да позволи получаване и обработване на огромно количество данни от сензорите едновременно;
- Изграждане на интуитивен, гъвкав и лесен за поддържане потребителски интерфейс;
- Комбиниране на различни устройства, някои, от които не поддържат комуникационния протокол или технология, в обща IoT система или интегриране на множество различни протоколи в IoT платформа;
- Осигуряване на сигурността и надеждността на комуникацията с устройствата, както и физическия достъп до тях;
- Осигуряване на възможности за енергоспестяване и в същото време гарантиране на незабавно преминаване в работен режим на устройството при необходимост.

Архитектурните шаблони определят начина на организация и взаимодействие на елементите на системата. Те въвеждат базовите характеристики и поведение на системата, принципите и многослойната структура в архитектура на системата. Smart home системите се базират на IoT системите. Затова те използват IoT архитектурните шаблони, които предполагат изграждане на единна система в една хетерогенна среда. При необходимост в системата се въвеждат компоненти, които да позволят създаването на различен тип интеграция и комуникация на компонентите ѝ. Те включват комуникация между устройства (Device to Device D2D), между мрежи (Networking to Networking N2N), данните и приложенията (Middleware to Middleware (MW2MW), между приложения (Application&Services to Application&Services AS2AS), обмен на данни (Data&Semantics to Data&Semantics DS2DS) и защита от различни нива на архитектурата, която включва различни механизми от автентификация на идентификационни данни, използване на маркери за удостоверяване или слой за сигурни сокети (SSL). Включването на едни или други архитектурни елементи се влияе и от предназначението на системата, тъй като системите са разнородни. Архитектурни шаблони са предложени от различни автори (Weyrich et al., 2016, Aziz, 2013, Alreshidi et al., 2019).

Слоевете на архитектурата на smart home системите могат да се използват като отправна точка за класификация на шаблоните за проектиране, който позволява всяка група разработчици по-лесно да открие подходящ шаблона за разработвания елемент от системата. Слоевете на архитектурата са три (Aziz, 2013): слой на устройствата, слой на услугите и слой на приложенията (фиг. 1).



Фигура 1. Общ вид на слоевете на архитектурата на smart home система.

Слоят на устройствата се реализира основно на хардуерно ниво. За шаблоните за smart home системи, този слой е важен по отношение на начините за управление на устройствата така, че да се постигнат различни цели, като енерго-спестяване. Слоят на услугите се реализира посредством облачните технологии, а за приложенията е от изключително значение управлението и мониторинга на потребителя. Затова тези два слоя биха могли да се разделят от гледната точка на взаимодействието с потребителя на слой на потребителски интерфейс (или frontend слой) и backend ниво. Smart home системите следват принципите на IoT, тъй като са представител на този клас системи. Но има и някои различия, като основното се състои в реализацията на диалога с потребителя. За разлика от Индустриалния Интернет на нещата при системите за интелигентен дом, диалога с потребителя следва да е многопластов, за да е подходящ за голямото разнообразие от потребители с различни цели и ниво на познания. За да се отговори на различните им нужди се създават разнообразни приложения, за които следва да се разработи подходящ потребителски интерфейс.

Като се обобщят проблемите, които решават шаблоните за проектиране могат да се разделят в следните основни групи, определени от основните категории проблеми: шаблони за потребителски интерфейс, шаблони за интеграция, шаблони за комуникация, шаблони за сигурност и шаблони за енергоспестяване. Като към шаблоните за проектиране се добави и групата на архитектурните шаблони се получава класификация, която да позволи лесно откриване на шаблоните при необходимост (фиг. 2).

Фигура 2. Общ вид на слоевете на архитектурата на smart home система. (Aziz, 2013)

4. Класифициране на шаблоните за проектиране в Smart home системи и особености на различните групи шаблони

Шаблоните за взаимодействие с потребителя реализират потребителския интерфейс. Те се разделят според групите операции, които изпълняват. Използват се за извличане на данни, изпълняване на действия или обработване на събития. Първата разновидност на шаблоните за потребителския интерфейс са шаблоните за предизвикване на действие от устройството. Тя обединява шаблони, които позволяват на потребителя да изпрати на устройството набор от операции, които да изпълни. Такива шаблони са предложени от Брамбила (Brambilla et al., 2017). Това са One device one operation, One Device

More Operations, More Devices One Operation, More Devices More Operations, One Device One Program и One Category More Operations. Други шаблони за потребителския интерфейс са предназначени за извличане на данни. Те са шаблони за взаимодействие, които извличат информация от устройства или програми. Техни представители са Get Details of a Device, Get State of the Device, Get Information from the Device, Get Information for One Category, Search Device, и Nearby Devices (Brambilla et al., 2017). Разновидност на шаблоните за интерфейса са предназначени и за обработка на настъпването на конкретно събитие, като проверка за постъпването на нови данни от устройствата. Те подпомагат енергоспестяването и биха могли да се отнесат и към тази група. Такива са Pull Information, Application Launch и Push Information (Brambilla et al., 2017).

Групата на шаблоните за взаимодействие с потребителя са предназначени да подпомогнат разработването на интерфейса, като цяло. Те се типични шаблони, подпомагащи организацията на интерфейса, съдържанието и навигацията му. Предложените шаблони (Brambilla et al., 2014) не са строго специализирани за smart home системи, а са предназначени за организация на интерфейса на различни уеб и мобилни приложения. Подходящи са и шаблоните, които се отнасят за конфигуриране на IoT системи. Част от тях подпомагат осигуряването на сигурността чрез конфигуриране на разрешенията и правата за достъп. Използват се CRUD патентите на потребители или групите им и им се присвояват правата за достъп. Такива са шаблоните User management, Master details and multi-details, Multi-level master details и др. (Brambilla et al., 2014). За контрола на достъпа също има предложени шаблони за автентикация и вход на потребителите: Location-aware search, Login.

Друга голяма група шаблони са шаблоните за синхронизация на данните. За тази цел има дефинирани голямо разнообразие от шаблони, като особено подходящи за smart home системите са шаблоните, които се базират на принципите на IoT. Те се прилагат съвместно с шаблоните за потребителския интерфейс, за да предоставят необходимата информация. Такива са шаблоните Asynchronous data synchronizations, Synchronous data synchronization, Partial storage, Complete storage, Full transfer, Timestamp transfer, Mathematical transfer (Brambilla et al., 2014).

Голяма група шаблони, които са строго специфични за smart home системите са предложени от Вега (Vega-Barbas et al., 2017). Той предлага шаблон за взаимодействие с потребителя, които имат за цел да го информират за действията на системата. Това са Greeting Pattern, Farewell Pattern, Action-Reaction Pattern, Conversation Pattern, Exploration Pattern. За да може потребителят да следи работата на системата, следва да получи обратна връзка за работата й. Тези шаблони имат за цел да предоставят на потребителя пълна и точна информация за избраната от тях услуга, така че потребителите да могат да вземат адекватно решение. Например шаблонът Greeting следва да информира потребителите дали желаната услуга може да се реализира в средата на конкретната система или каква е причината, ако е невъзможно. Шаблонът Farewell Pattern информира потребителя за възможността да се изтрие или запази събраната информация за конкретна услуга. Шаблонът Action-Reaction Pattern информира потребителят за последствията от избраната от него услуга или за невъзможността на изпълнението ѝ. Шаблонът Conversation Pattern следва да информира потребителя в достъпна за него форма за цялата информация, която се получава при изпълнението на услугата. Шаблонът Exploration Pattern докладва за цялата събрана информация от сензорите на устройствата.

Шаблоните за сигурност имат за цел да предотвратят случайното вмъкване на уязвимости в кода и да смекчат последствията от тези уязвимости. Те обединяват познанията за сигурността и системната структура, като дават възможност за развитие и усъвършенстване на софтуера. Позволяват да се интегрира политиката на сигурност с проектирането и разработването на софтуера. Шаблоните за сигурност са многобройни. Нивата им варират от архитектурни шаблони, включващи проектиране на ниво цялостна

система, до шаблони на ниво на изпълнение, отнасящи се до реализацията на отделни функции или методи в системата. Част от проблемите по сигурността са общи за всички системи. Затова при разработката на IoT система могат да се използват както специализирани шаблони за IoT, така и шаблони, които реализират един или друг аспект на сигурността при всички системи. Например шаблона Whitelisting Firewall на (Villarreal et al., 2013) въвежда защитна стена, която да контролира комуникацията и да я филтрира само до разрешените събеседници без да я забавя. Подобни шаблони предлага и Фернандес (Fernandez, 2013; Fernandez et al., 2014). Той представя два шаблона за изграждане на защитна стена с цел контрол на мрежовия достъп до облачните ресурси. Подходящи са и шаблоните, които осигуряват сигурността в облачната среда на Хафиз (Hafiz, 2013), шаблоните за уеб приложенията на Шумахер (Schumacher et al., 2006), шаблоните за симетрично и асиметрично кодиране на достъпа на Дара (Dara, 2014) или шаблоните на Романовски (Romanosky, 2001) и на Кинзъл (Kienzle et al., 2002).

Освен общите шаблони са създадени и специфични за нуждите на IoT системите. Такива са шаблоните на Рейнфърт (Reinfurt et al., 2017). Той предлага шаблоните за проектиране Trusted Communication Partner, Outbound-Only Connection, Permission Control, Personal Zone Hub, Whitelist и Blacklist. Шаблонът Trusted Communication Partner има за цел да предотврати достъпа до устройството или неговата мрежа на случайни комуникационни партньори чрез предоставяне на възможност за комуникация с устройството единствено на списък от разрешени предварително известни партньори. Друго решение за предотвратяване на неоторизиран достъп е предложено в шаблона Outbound-Only Connection, които позволява отговор само на тези входящи заявки за комуникация, които са в отговор на връзка, инициирана от устройството. Шаблонът Permission Control гарантира сигурността на данните на устройството чрез запазване на backend сървър на правата за достъп до данните на комуникационните партньори. Personal Zone Hub предполага създаване на хъб, който да позволи управлението на правата, споделянето на данните и контрола на всички устройства на даден потребител. Шаблонът Whitelist предполага създаване на списък с потребители, които имат права за достъп, като никой извън този списък няма достъп. Blacklist е на обратния принцип. Предполага създаването на списък с ненадеждни партньори, на които се отказва правото за достъп.

Другата основна група **шаблони са за интегриране**, които осигуряват комбинирането на устройствата, разположени на различни крайни възли с различни протоколи в IoT платформа. Костер предлага шаблони за интегриране на устройства, които използват различни технологии и мрежови протоколи. Те се отнасят до проблемите с инфраструктурата на IoT (Koster, 2014). Неговите решения обаче не са описани, като типични шаблони. Кенбъри (Qanbari et al., 2016) също предлага предлага шаблони за комбиниране на устройствата, които се отнасят за възлите от периферията: Edge Provisioning Pattern, Edge Code Deployment Pattern, Edge Orchestration Pattern, Edge Diameter Of Things (DOT) Pattern. Edge Provisioning Pattern осигурява контрол върху голям брой труднодостъпни разпръснати крайни устройства, като осигурява възможност за улесненото им преконфигуриране, а също и за лесно включване на нови. Edge Code Deployment Pattern осигурява възможност за лесно поддържане на софтуера на устройствата чрез децентрализиран Git контрол на версиите, разположен на специализиран сървър. Edge Orchestration Pattern разпределя функционалността така, че остави възможност на крайните възли сами да контролират, конфигурират и управляват крайните си устройства, да следят състоянието им, да откриват нужните им услуги. Edge DOT Pattern изисква създаване на сървър за измерване, чрез който да се уеднакви начина на измерване на извършените услуги в различните крайни възли. Това се налага, тъй като различните доставчици могат да използват различни шаблони за използване, като например базирани на събития или базирани на време.

Следващата голяма група са **шаблони са за комуникация.** Питър (Peter, 2016)

предлага такива шаблони: Request/Response, Event Subscription, Asynchronous Messaging, Reliable Messaging, Multicasting, Publish/Subscribe, Queues, Message Brokers, Federation, Discovery and Delegation of Trust, които обаче не са описани според строгата дефиниция за документация на шаблон.

Reinfurt също предлага шаблони за решаване на проблемите с комуникацията (Reinfurt et al., 2019). Device Gateway, Device Shadow, Rules Engine, Device Wakeup Trigger, Remote Lock and Wipe, Delta Update, Remote Device Management, Visible Light Communication. Шаблонът Device Gateway се използва в случаите, когато част от устройствата не поддържат комуникационната технология или протокол на мрежата. Шаблонът предполага използване на посредник, като Gateway, които да превежда протоколите от и към устройството. Device Shadow се използва за изпращане на заявки към устройства, които не са постоянно включени в мрежата. Шаблонът предполага създаване на виртуално обръщение към устройството и обикновено пряко си взаимодейства с шаблоните за интеграция. Rules Engine позволява на потребителите да настройват системата чрез съвкупност от прости правила. Device Wakeup Trigger предполага изпращане на съобщение по комуникационния канал до управляващия блок на устройството, което не е постоянно включено в мрежата. Remote Lock and Wipe гарантира сигурност на данните съхранявани на устройството, като позволява при отключване на устройството от системата данните му да се изтриват. Delta Update подпомага намаляването на мрежовия трафик чрез изпращане само на променените от последното изпращане данни на устройството. Remote Device Management позволява локално управление на устройството чрез клиент инсталиран на устройството, който интерпретира сървърните команди. Visible Light Communication предполага изпращане на светлини сигнали за комуникация до или от отдалечено устройство.

Последната група **шаблоните за енергоспестяване** набират все по-голяма актуалност. Рейнфърт (Reinfurt et al., 2017) също предлага шаблони за тази категория, като ги разделя на два типа. Първите описват различните типове устройства според изискванията им на енергия Mains-Powered Device, Period Energy-Limited Device, Lifetime Energy-Limited Device, Energy-Harvesting Device и затова не представляват интерес за софтуерното обезпечаване на smart home системите. Вторият тип описват начини за комуникация с устройствата, така че да се спести енергия: Always-On Device и Normally-Sleepping Device. Шаблонът Always-On Device се отнася до случаите, когато пестенето на енергия е неефективно. Другият шаблон Normally-Sleepping Device се използва в случаите, когато не е необходимо устройството да работи непрекъснато и се реализира чрез деактивиране на всички му енергозависими елементи. При необходимост от работата на такова устройство управляващият блок подава енергия и събужда устройството. Възможно е дори управляващият блок да не е постоянно подключен в мрежата, а да се включва на определени интервали, за да провери за необходимост от събуждане на устройството. Шаблоните, с чиято помощ се реализира комуникацията с подобни устройства от разгледаните са Device Shadow, Device Wakeup Trigger.

Разгледаните групи шаблони по критерия цел са обобщени в таблица 1.

Таблица 1. Списък на smart home шаблоните според целта.

Група шаблони	Примери
Архитектурни шаблони	Шаблон на Алрашиди, на Азис, на Уейрик
Шаблони за взаимодействие с потребителя:	
• Шаблони за предизвикване на действие от устройството	One device one operation, One Device More Operations, More Devices One Operation, More Devices More Operations, One Device One Program, One Category More Operations

• Шаблони за потребителския интерфейс	User management, Master details and multi-details, Multi-level master details
• Шаблони за синхронизация на данните	Asynchronous data synchronizations, Synchronous data synchronization, Partial storage, Complete storage, Full transfer, Timestamp transfer, Mathematical transfer
Шаблони за сигурност:	
• Шаблони за сигурност за облачните системи	Шаблони на Хафиз, на Шумахер, на Романовски, на Кинзъл, шаблони за симетрично и асиметрично кодиране на достъпа
• Шаблони за сигурност специфични за IoT системите	Trusted Communication Partner, Outbound-Only Connection, Permission Control, Personal Zone Hub, Whitelist и Blacklist
Шаблони за интегриране	Шаблони на Koster, на Кенбъри, на Рейнфърт
Шаблони за комуникация	Request/Response, Event Subscription, Asynchronous Messaging, Reliable Messaging, Multicasting, Publish/Subscribe, Queues, Message Brokers, Federation, Discovery and Delegation of Trust, Device Gateway, Device Shadow, Rules Engine, Device Wakeup Trigger, Remote Lock and Wipe, Delta Update, Remote Device Management, Visible Light Communication.
Шаблони за енергоспестяване:	
• Шаблони за типовете устройства	Mains-Powered Device, Period Energy-Limited Device, Lifetime Energy-Limited Device, Energy-Harvesting Device
• Шаблони за комуникация с устройствата	Always-On Device, Normally-Sleepping Device

Заключение

Smart home системите осигуряват удобство и сигурност, потребителски ориентирана визуализация на данните от сензорите, спестяват енергия и намаляват разходите. Но разработването им е съпроводено с множество разнообразни проблеми. Една ефективна възможност за решаването им се предлага с помощта на шаблоните за проектиране. Но откриването на подходящ шаблон за конкретен проблем е трудно, особено при условие, че областта се развива динамично и все още не е предложена система за класификацията им. Подходящ критерий за класификацията им е целта. Разработчиците се специализират в конкретна проблематика и разделението на шаблоните според критерия цел позволява те да се фокусират върху нея. С подобна класификация при използване на top-down подход за разработка се улеснят разработчиците в откриването на подходящ шаблон.

References

1. Alreshidi, A. and Ahmad, A. (2019) *Architecting Software for the Internet of Thing Based Systems.* MDPI, Basel, Switzerland [Online] Available from: https://www.mdpi.com/1999-5903/11/7/153/pdf [Accessed 12/08/2019].
2. Armiyanova, M. (2018) Patterns Classification Criteria and Scheme. *Eastern Academic Journal*, Bourgas, Vol. 3, September 2018, pp.70-89.

3. Aziz, M., (2013) Service-Oriented Layered Architecture for Smart Home. *International Journal of Smart Home* Vol.7, No.6 (2013), pp.409-418.
4. Brambilla, M, Fraternali, P. (2014) *Interaction Flow Modeling Language: Model-Driven UI Engineering of Web and Mobile Apps with IFML.* Morgan Kaufmann Publishers Inc., USA.
5. Brambilla, M., Umuhoza, E., Acerbis, R. (2017) Model-driven development of user interfaces for IoT systems via domain-specific components and patterns. *Journal of Internet Services and Applications. Internet Serv Appl* (2017) 8: 14. Available from: https://doi.org/10.1186/s13174-017-0064-1 [Accessed 12/10/2019]
6. Changhua, C. (2018) Discussions on the Pattern Development of Smart Home Design. *2018 International Conference on Social Sciences*, Education and Management (SOCSEM 2018).
7. Chen, Y. C., Chen, C. C., Peng, W. C., et al. (2014) Mining Correlation Patterns among Appliances in Smart Home Environment, *Pacific-Asia Conference on Knowledge Discovery and Data Mining,* April 2015, Advances in Knowledge Discovery and Data Mining. pp. 222-233.
8. Dara, S. (2014) Privacy Patterns in Public Clouds. *Proceedings of the Indian Conference on Pattern Languages of Programs (GuruPLoP).*
9. Fernandez, E. (2013) *Security Patterns in Practice: Designing Secure Architectures Using Software Patterns.*Wiley.
10. Fernandez, E. B., Yoshioka, N. and Washizaki, H. (2014) Patterns for Cloud Firewalls. Proceedings of the *Asian Conference on Pattern Languages of Programs* (AsianPLoP).
11. Hafiz, M. (2013) *Security Pattern Catalog*. [Online] Available from: http://www.munawarhafiz.com/securitypatterncatalog/index.php [Accessed 12/08/2019].
12. Kienzle, D., Elder, M., Tyree, D. and Edwards-Hewitt, J. (2002) *Security Patterns Repository* Version 1.0. [Online] Available from: http://www.scrypt.net/~celer/securitypatterns/repository.pdf [Accessed 12/08/2019]
13. Koster, M. (2014) *Design Patterns for an Internet of Things.* [Online] Available from: http://community.arm.com/groups/internet-of-things/blog/2014/05/27/design-patterns-for-an-internet-of-things Available from: http://iot-datamodels.blogspot.com/2014/05/design-patterns-for-internet-of-things.html [Accessed 12/08/2019]
14. Nawaz, A., Helbostad, J. L., Skjæret, N., et al. (2014) Designing Smart Home Technology for Fall Prevention in Older People, *International Conference on Human-Computer Interaction.* HCI International 2014 - June 2014, Crete, Greece, Springer International Publishing, 485-490. Available from: http://farseeingresearch.eu/wp-content/uploads/2014/06/Nawaz-et-al.-2014-Poster.pdf [Accessed 07/10/2019]
15. Peter W. (2016) *Communication Patterns for the Internet of Things*, June 1, 2016. [Online] Available from: https://software.intel.com/en-us/articles/communication-patterns-for-the-internet-of-things [Accessed 12/08/2019].
16. Qanbari, S., Pezeshki, S., Raisi, R., Mahdizadeh, S., Rahimzadeh, R., Behinaein, N., Mahmoudi, F., Ayoubzadeh, S., Fazlali, P., Roshani, K., Yaghini, A., Amiri, M., Farivarmoheb, A., Zamani, A., and Dustdar, S. (2016) IoT Design Patterns: Computational Constructs to Design, Build and Engineer Edge Applications. *IEEE First International Conference on Internet-of-Things Design and Implementation* (IoTDI), pp. 277-282.
17. Reinfurt, L., Breitenbücher, U., Falkenthal, M., Leymann, F. and Riegg, A. (2019) *Internet of Things Patterns for Communication and Management*. Springer-Verlag. [Online] Available from: https://www.iaas.uni-stuttgart.de/publications/ART-2019-08-Internet-of-Things-Patterns-for-Communication-and-Management.pdf [Accessed 12/08/2019].
18. Reinfurt, L., Breitenbücher, U., Falkenthal, M., Leymann, F. and Riegg, A. (2017) Internet of Things Patterns for Devices [Online] Available from: https://www.iaas.uni-stuttgart.de/publications/INPROC-2017-15-Internet-of-Things-Patterns-for-Devices.pdf [Accessed 12/08/2019].
19. Reinfurt, L., Breitenbücher, U., Falkenthal, M., Fremantle, P., and Leymann, F. (2017) Internet of Things Security Patterns. HILLSIDE *Proc. of Conf. on Pattern Lang. of Prog*. [Online]

Available from: https://www.hillside.net/plop/2017/papers/proceedings/papers/20-reinfurt.pdf [Accessed 12/08/2019].

20. Romanosky, S., (2001) Security Design Patterns, Technical report, [Online] Available from: http://www.cgisecurity.com/lib/securityDesignPatterns.pdf [Accessed 12/08/2019]
21. Schumacher, M., Fernandez, E. B., Hybertson, D., Buschmann, F., and Sommerlad, P., (2006) *Security Patterns: Integrating Security and Systems Engineering*. John Wiley & Sons.
22. Sulova, S., (2019) The Usage of Data Lake for Business Intelligence Data Analysis, *International Conference Information and communication technologies in business and education*, 18 October. UE-Varna, University publishing house "Science and economics", pp. 135-144.
23. Vega-Barbas, M., Pau, I., Augusto, J., Seoane, F. (2017) Interaction Patterns for Smart Spaces: *A Confident Interaction Design Solution for Pervasive Sensitive IoT Services*. November 2017 *IEEE Access, 6* . pp. 1126-1136. ISSN 2169-3536
24. Villarreal, N., Fernandez, E., Larrondo-Petrie, M. and Hashizume, K. (2013) A Pattern for Whitelisting Firewalls (WLF). PLoP 13. [Online] Available from: http://www.laccei.org/LACCEI2013-Cancun/ExtendedAbstracts/EA055.pdf [Accessed 12/08/2019].
25. Weyrich,M., Ebert, C. (2016) *Reference Architectures for the Internet of Things.* IEEE Softw. 33, pp. 112–116.