

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Informaatikainstituut

Infosüsteemide õppetool

Kasutajasõbraliku liidese analüüs ja väljatöötamine targa maja lahendusele

Bakalaureusetöö

Üliõpilane: Martin Talimets

Üliõpilaskood: 120729IABB

Juhendaja: Enn Õunapuu

Tallinn
2016

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

(kuupäev)

(allkiri)

Annotatsioon

Antud töös on analüüsitud ja väljatöötatud koduomanikule orienteeritud targa maja liides.

Antud bakalaureusetöö probleemiks oli erinevaid targa maja lahendusi ühendava tarkvara kasutajasõbraliku liidese puudumine. Kasutajaliideses on arendatud lõpptarbijale mõeldud liides, kus on võimalik seadistada majaplaani, lisada automatiseerimisreegleid ning ühendada süsteemiga füüsilisi seadmeid. Loodud liidest on lõpptarbijal mugav ja lihtne kasutada nii mobiilsetel seadmetel kui ka tavaarvutitel.

Töös kirjeldatakse loodava liidese toimimist ning analüüsi tulemuste põhjal on arendatud lõpptarbijale mõeldud kasutajaliides. Analüüsi käigus on vaadeldud olemasolevaid targa maja kasutajaliideseid ning kajastatud tulevikuarendusi, mis on vajalikud enne liidese lõpptarbijatele kasutamiseks andmist.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 35 leheküljel, 4 peatükki, 13 joonist, 2 tabelit.

Abstract

In this present thesis, there is analyzed and developed a smart home interface which is oriented for home owners.

The problem of this bachelor thesis was that there was not a user friendly interface to connect different smart home solutions. The user interface is developed for end users, where users can configure house plans, add automation rules and connect physical devices with the system. The created interface is convenient and easy to use for end users on mobile devices as well as computers.

The operation of the interface is described in this thesis and based on the analysis' results, the author has developed a user interface for end users. During the analysis, existing smart home user interfaces are examined and reflected future developments that are necessary before the interface is released for end users.

The thesis is in Estonian and contains 35 pages of text, 4 chapters, 13 figures, 2 tables.

Lühendite ja mõistete sõnastik

REST

Representational State Transfer

Tarkvaraarhitektuuri stiil

API

Application Program Interface

Reeglistik olemasoleva valmisprogrammiga suhtlemiseks

JSON

JavaScript Object Notation

Lihtsustatud andmevahetusvorming

IoT

Internet of Things

Asjade internet on igapäevaste seadmete võrgustik, mis oskab läbi interneti enda kohta infot edastada ja käskke vastu võtta.

Jooniste nimekiri

Joonis 1. IoT valdkonna mudeli osa [1]	10
Joonis 2. openHAB kasutajaliides	16
Joonis 3: HABmin kasutajaliides	17
Joonis 4: Yoga Systems kasutajaliidese majaplaani seadistamise vaade	30
Joonis 5: Yoga Systems kasutajaliidese reeglite lisamise vaade	30
Joonis 6: Fibaro kasutajaliides.....	31
Joonis 7: Fibaro seadme lisamine	31
Joonis 8. Majaplaani loomise vaade	32
Joonis 9. Avalehe vaade	32
Joonis 10. Seadme lisamise vaade	33
Joonis 11. Seadme muutmine	33
Joonis 12. Seadmega ühendamise vaade	34
Joonis 13. Reeglite lisamise vaade	34

Tabelite nimekiri

Tabel 1: Kasutajaliidese vaated	20
Tabel 2: Kasutatavad REST otspunktid	21

Sisukord

1. Sissejuhatus	9
1.1 Taust ja probleem	9
1.1.1 Targa maja olemusest	10
1.2 Ülesande püstitus	12
1.3 Metoodika	13
1.4 Ülevaade tööst	13
2. Lahendus.....	14
2.1 Olemasolevate targa maja lahenduste kasutajaliideste analüüs.....	14
2.1.1 Telia kodujuhtimine.....	14
2.1.2 Fibaro.....	15
2.1.3 OpenHAB	15
2.1.4 HABmin	16
2.2 Kasutatavate tehnoloogiate valik.....	18
2.2.1 JSON.....	18
2.2.2 Bootstrap.....	19
2.3 Kasutajaliidese toimimine ja võimalused	19
2.4 Kasutajaliidese vaated	20
2.4.1 REST otspunktid	21
2.4.2 Avaleht	21
2.4.3 Majaplaani lisamine.....	22
2.4.4 Majaplaani vaade.....	23
2.4.5 Seadmega ühendamise vaade	24
2.4.6 Reeglite lisamise vaade	24
3. Tulemuste analüüs	26
3.1 Kasutajaliidese vastavus püstitatud nõuetele.....	26
3.2 Võimalikud tulevikuarendused.....	26
4. Kokkuvõte	27
Summary.....	28
Kasutatud kirjandus	29
Lisa 1	30
Lisa 2	35

1. Sissejuhatus

Targa maja mõistet kasutatakse tänapäeval palju. Selle all mõeldakse enamasti kodu, kus kasutatakse automatiseeritud seadmeid. Üldjuhul on eesmärgiks lisaks nutifunktsionaalsusele hoida kokku tarbitavat energiat, pakkuda turvalisust ning lisaks on sellel ka meelelahutuslik pool. Tarka maja on võimalik seadistada erinevate andurite abil saavutamaks maksimaalset mugavust. Näiteks paigutades temperatuuriandurit, mis on seotud küttesüsteemiga, on võimalik reguleerida kodu sisetemperatuuri vastavalt eelnevalt seadistatud normidele. Lisaks on võimalik paigaldada targa kodu süsteemile erinevaid liikumisandureid, niiskusandureid, tuletõrjealarme, erinevaid turvasüsteemide elemente jms. Igal ühel on võimalik muuta oma kodu täpselt nii targaks ja mugavaks, nagu parajasti soove ja võimalusi on, kuna kindlaid norme ja reeglistikke targa maja olemusele seatud ei ole. Terve süsteemi kasutamiseks on vajalik liidese olemasolu, mis pakub kasutajale võimalust oma maja seadistada vastavalt soovile. Käesolevas lõputöös pakub töö autor välja targa maja kasutajaliidese lahenduse lõpptarbijale, kelle soov on kiirelt ja mugavalt seadistada oma kodu.

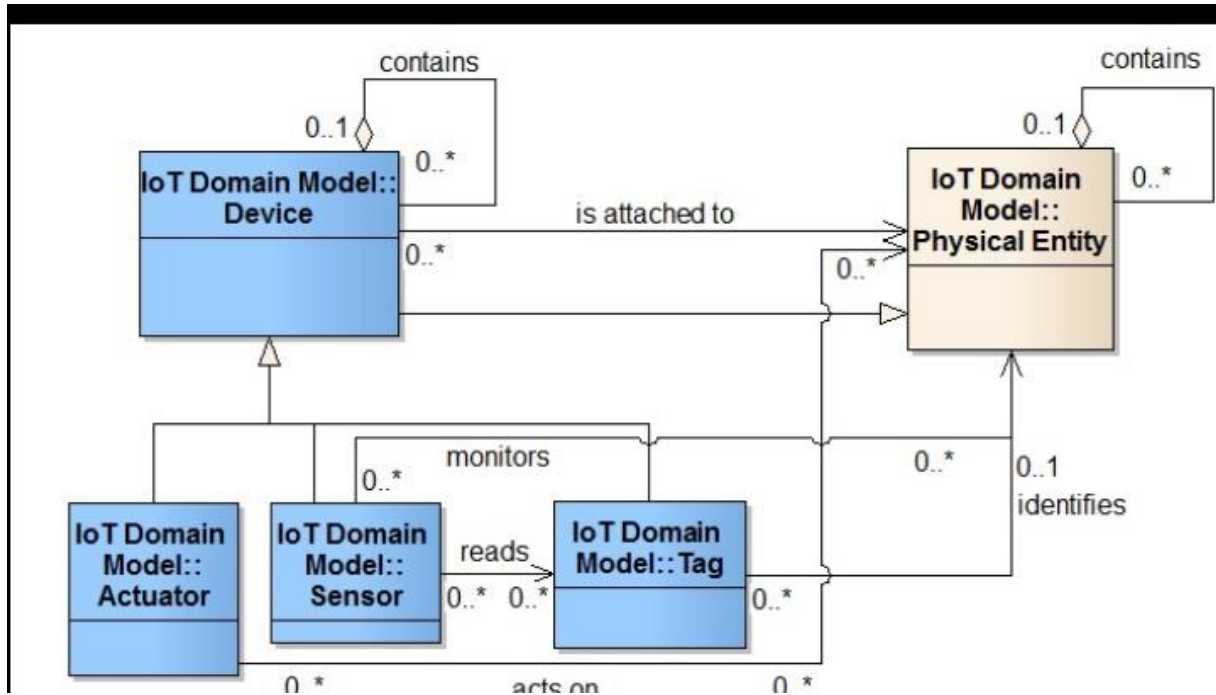
1.1 Taust ja probleem

Antud bakalaureusetöö on mõeldud lõpptarbijatele, kes tahavad oma kodu nutikaks muuta ja seda võimalikult lihtsalt seadistada. Praegused olemasolevad targa kodu lahendused ei ühildu teiste sarnaste väljatöötatud lahendustega või on nende kasutajaliideseid lõpptarbijate jaoks liiga keerulised. Liideseid on keerulised, kuna seadistamine toimub erinevates vaadetes ning kogu vajalik info ei ole kergesti leitav.

Antud töö lahendatavaks probleemiks on kõiki targa kodu süsteeme ühendava tarkvara kasutajaliidese ebasobivus lõpptarbijajaoks. Olemasolevad liideseid on arusaadavad ja loogilised programmeerijatele, kuid mitte tavakasutajatele. Töö kirjutamise hetkel ei olnud veel loodud openHAB tarkvara jaoks lihtsat ja mugavat kasutajaliidest, mis sobiks kasutamiseks lõpptarbijatele, kes ei ole sellega varasemalt kokku puutunud. Seetõttu arendab töö autor lõputöö käigus kasutajasõbraliku liidese eelnevalt nimetatud tarkvarale.

1.1.1 Targa maja olemusest

Tihti mõistetakse asjade interneti all seda, et on mingisugused seadmed, mis on ühendatud internetiga. Tegelikult ei ole interneti ühendatud seadme füüsiline olemus, vaid selle küljes olevad andurid, mis on seotud virtuaalse olemusega. Järgnev joonis tutvustab asjade interneti valdkonna mudelit, millel kujutatakse seadme füüsilise olemuse jagunemist.



Joonis 1. IoT valdkonna mudeli osa [1]

Seadmed koosnevad väga paljudest osadest ja tihti aetakse neid segamini anduritega. *Sensor* ehk andur on vaid osa seadmest, mis tegeleb seadme info edastamisega. *Actuator* on näiteks seadme küljes olev lüliti, millega on võimalik muuta seadme käitumist. *Tag-i* abil toimub seadme identifitseerimine, mis määrab ära kasutaja õigused seadmele ligipääsemiseks. Seadme füüsilise olemusega on seotud selle virtuaalne olemus, mida kasutaja näeb liidese kaudu. Virtuaalne olemus kujutab endast füüsilist seadet, mille kaudu nähakse seadme kohta käivat infot või saadetakse seadmele käsklusi.

Tark maja on ehitis, mille sisse on integreeritud spetsiaalsed seadmed, mida on võimalik käskudega automaatselt juhtida. Näiteks on koduomanikul võimalik juhtida turvasüsteemi, muuta temperatuuri, seadmeid sisse/välja lülitada, kodukinosüsteemi juhtida ja palju muud teha kasutades ainult oma nutitelefoni. Kodu automatiseerimise valdkond areneb ühe kiiremini koos

tehnoloogia arenguga ja hõlmab endast meelelahutust, turvalisust, mugavust, side- ja infosüsteeme [2].

Täna sel päeval tähendab tark maja ennekõike energiasäästu, turvalisust, mugavust ja kõike seda peab olema võimalik juhtida ühtse maja juhtimistarkvara abil. Kui nn. tavapärasel majal on enamasti mitmeid lokaalseid automaatikasüsteeme nagu näiteks küttesüsteemi, valvesüsteemi, ventilatsiooni, kodutehnika ja -seadmete juhtimine siis targa maja peamine ülesanne on kõik seadmed saada tsentraalse juhtimise alla, mis võimaldab programmeerida erinevate sündmuste korral tegevusi, mis kõige paremini maja ja omaniku soovidega kokku sobivad.

Tark maja muutub väärtuslikuks ainult siis, kui selle kasutamise abil realselt ka elukvaliteeti on võimalik tõsta. Targaks ei tee maja mitte seadmed iseenesest vaid ikkagi tarkvara. Tarkvara abil luuakse eelised, näiteks tarbimisharjumuste analüüsi abil optimeerides maja süsteemide säästlikumat kasutamist. Tõenäoliselt areneb maja juhtimistarkvara just sellest suunas, et inimese poolt süsteemi reguleerimisvajadus väheneb või kaob hoopiski ning inimese rolliks jääb ülevaate saamine ja süsteemi töö aeg-ajalt kontrollimine.

Kindlasti on oluline teada, et suurimat efekti annavad intelligentsed targa maja süsteemid ennekõike suurtes elamutes, büroo- ja korterelamutes, kus ka väikesed muudatused näiteks kütetemperatuuri või ventilatsiooni juhtimisel annavad suurt kokkuhoidu. Eramajade puhul on kindlasti samuti võimalik saavutada kokkuhoidu, kuid väiksemas mahus ja pigem on eramaja puhul esmatähtis ennekõike turvalisus ja mugavus ning see on just see, mida me oma kodult ootame [3].

Üks põhilisi targa maja kasutegureid tuleb välja tema energiasäästlikkuse võimest. Küttesüsteemid ja ventilatsioonisüsteemid, mis töötavad vastavalt ilmastikule, toovad kodukasutajale aastas arvestatava kokkuhoiu. Lisaks valgustuse ja elektrienergia optimaalne juhtimine, mis tagab mittevajalike valgustite ja seadmete automaatse väljalülitamise ajal kui neid ei vajata [3].

Automaatne energia salvestamine soodsal hetkel on üks kõige olulisemaid targa kodu omadusi. Tarkvara, mis salvestab elektrit ainult siis, kui see on võimalikult madala hinnaga, võimaldab energiakulusid suuresti kokku hoida. Tarkvara paneb kokku statistika energia hindadest ja laeb akusid võimalikult madala hinna korral. Tavalised kodumasinad kasutavad seda akudesse laetud energiat, mis on kogutud odavama energia hinnaga.

Turvasüsteeme on ka võimalik automatiseerida ja ühildada teiste majas olevate süsteemidega. Oma kodu turvalisuse tagamiseks võib uksi lukustada hääle, sõrmejälgede, koodide või kiipide abil. Neid süsteeme saab sisse või välja lülitada kaugjuhtimise abil. Samuti võimaldab tark maja seadmete automaatset regulaarset monitooringut, mis teavitab kõrvalekalletest kohe, kui need tekivad või juhib ennetavalt tähelepanu, kui selline olukord on tekkimas [3].

Kodust eemal viibides saab maja turvasüsteemidel silma peal hoida. Kogu info maja turvasüsteemide kohta laetakse reaalajas veebi üles ja on kaugelt hallatav. Turvakaamerate pilt ja alarmide info on koduomanikule olenemata asukohast alati ligipääsetav ja võimalik süsteeme tööle panna või välja lülitada. Aktiveeritud alarmide info jõuab koheselt koduomanikule ja nii on ohule reageerimisaeg kiirem [3].

Automatiseeritud meelelahutuskeskus laseb kasutajal ennast sobivalt seadistada. Puudub vajadus mitmete pultide järgi, millega seadmeid juhtida. Kogu juhtimine toimub ühest kohast ja on võimalik automatiseerida. Automatiseeritud meelelahutuskeskus paneb vastavalt seadistusele iga kasutaja kohta kokku temale meeldivad saated, muusika või filmid.

Tark maja jälgib kasutaja liikumist ruumides. Vastavalt sellele hoiab tark maja meelelahutust kasutajaga igas ruumis kaasas. Erinevate ruumide vahel liikudes ei ole vaja puldiga seal asuvaid seadmeid sisse või välja lülitada, seda teeb automatiseeritud meelelahutuskeskus kasutaja eest ise. Samamoodi kasutaja liikumist jälgides lülitab tark maja automaatselt sisse lambid, televiisorid, avab kardinad või isegi ukсед. Samuti lülitab tark maja kõik ebavajalikud seadmed välja, kui kedagi majas pole. Peale mugavuse on see ka energiasäästlik. Nutitelefoniga on võimalik ka enne koju jõudmist seadmeid juhtida, näiteks küttesüsteeme käivitada või garaažiukse avada.

1.2 Ülesande püstitus

Käesoleva töö esmaseks eesmärgiks on luua lõpptarbijale mugav kasutajaliides, mis töötaks enimlevinud veebilehitsejates. Põhieesmärgi saavutamiseks analüüsib töö autor erinevaid olemasolevaid targa kodu lahendusi. Kogu targa maja seadistamine peab toimuma võimalikult väheste ressurssidega, ehk kiirelt ja lihtsalt, et tavakasutaja saaks sellega iseseisvalt hakkama. Seadistamisele ei tohiks kuluda liiga palju aega, ning kogu protsess peab lõpptarbijale olema arusaadav.

Hästi loodud kasutajaliides võiks ideaalis võimaldada olemasolevate seadmete otsimist, mitte nende käsitsi sisestamist. Sellisel juhul ei peaks kasutaja seadme ühendamiseks sisestama infot, vaid kogu protsess peaks toimuma automaatselt. Sellisel viisil ühendamine on kasutajale eelkõige ajasäästlik ja mugav, ning lisaväärtusena ei vaja see instruksioone.

1.3 Metoodika

Püstitatud eesmärkide saavutamiseks analüüsib töö autor esmalt Eestis kasutatavaid kasutajaliideseid. Eestis on kasutusel umbes kümnekond erinevat targa maja lahendust. Antud töös on analüüsitud nendest lahendustest Telia Nutikodu, Fibaro, openHAB-i ja HABmin-i kasutajaliideseid. Kasutajaliideste analüüsimiseks vaatleb töö autor liideste mugavust ning lihtsust lõpptarbija jaoks.

Tehtud analüüsi põhjal pakub töö autor välja lõpptarbija jaoks mõeldud kasutajaliidese. Kasutajaliidese arendamisel on arvestatud eelkõige selle lihtsusega lõpptarbija jaoks. Järgitud on põhimõtet, et mida kiiremini ja vähemate liigutustega saab seadistusi teha, seda mugavam ja parem on liidest kasutada.

1.4 Ülevaade tööst

Teises peatükis analüüsib töö autor olemasolevaid kasutajaliideseid. Autor tutvustab arenduseks kasutatavaid tehnoloogiaid ja põhjendab, miks just need on valitud. Samuti tutvustab väljapakutud kasutajaliidest ja selle toimimist.

Kolmandas peatükis uurib töö autor kasutajaliidese vastavust püstitatud nõuetele ja pakub välja lisaarendusi, mida kasutajaliidesele tulevikus lisada tuleks.

2. Lahendus

Selles peatükis analüüsib töö autor olemasolevaid targa maja süsteemide kasutajaliideseid ning pakub välja lõpptarbijale mõeldud kasutajaliidese openHAB tarkvara jaoks.

2.1 Olemasolevate targa maja lahenduste kasutajaliideste analüüs

Maju ei tee targaks seadmed, vaid neid juhtiv tarkvara. Seadmete juhtimine toimub läbi selle jaoks loodud liidese. Liides võimaldab reeglite abil muuta seadmete käitumist teatud tingimustes. Neid tingimusi järgides oskavad seadmed erinevates olukordades käituda ning seeläbi saavutataksegi tark maja.

Targa maja lahendusi on loodud väga palju. Kuigi erinevaid lahendusi on palju, ei ole nad enamasti omavahel ühendatavad. Samuti on enamus neist kuutasulised ning süsteemi töötamine sõltub teenusepakkujast. Töö autor on valinud mõned Eestis enimkasutatavad juhtmevabad targa maja liidesed ning analüüsinud nende kasutajasõbralikkust.

2.1.1 Telia kodujuhtimine

Telia nutikodu kasutab Yoga Systems-i poolt pakutavaid seadmeid. Antud toote jaoks on loodud pilveteenusel põhinev IoT platvorm. See tähendab seda, et nutikodu lahendust pakutakse teenuse näol ning on kuutasu põhine. Yoga Systems ei ole avatud lähtekoodiga ega suuda ühendada erinevaid protokolle kasutavaid targa kodu lahendusi [5]. Selle süsteemi veel üheks nõrkuseks on see, et ühele Yoga Systems keskseadmele on maksimaalselt võimalik lisada vaid 20 andurit. Suuremate majade korral võib sellest väheks jääda ning mitme keskseadme omamine ei ole lõpptarbija jaoks mugav.

Telia kodujuhtimise kasutajaliidest on mugav kasutada. Majaplaani seadistamise vaatel (Lisa 1, Joonis 4) kuvatakse kasutajale olemasolevad majaplaanid ning ka uue majaplaani lisamise nupp on kohe nähtaval. Majaplaanile vajutades avaneb menüü, kust on võimalik erinevaid seadeid muuta. Näiteks on seal võimalik lisada seadmeid, reegleid, kasutajaid jpm.

Seadmetele reeglite lisamine (Lisa 1, Joonis 5) on antud tarkvara poolt kasutaja jaoks väga mugavaks tehtud. Kasutaja ei pea reeglite lisamiseks muutma lähtekoodi, mille alusel seadmed töötavad. Liidese jaoks on loodud mugavad menüüid, kus on loetletud erinevad tingimused, tegevused ja seadmed. Kogu reeglite lisamise protsess on lihtne ja kasutajale arusaadav ning mugav kasutada.

2.1.2 Fibaro

Fibaro targa kodu süsteem on iseseisev ja ei vaja töötamiseks väliseid servereid. Sarnaselt Telia kodujuhtimisele, ei ole võimalik Fibaro süsteemiga ühildada teisi seadmeid, peale tootja enda pakutavate või sama protokolliga kasutatavate.

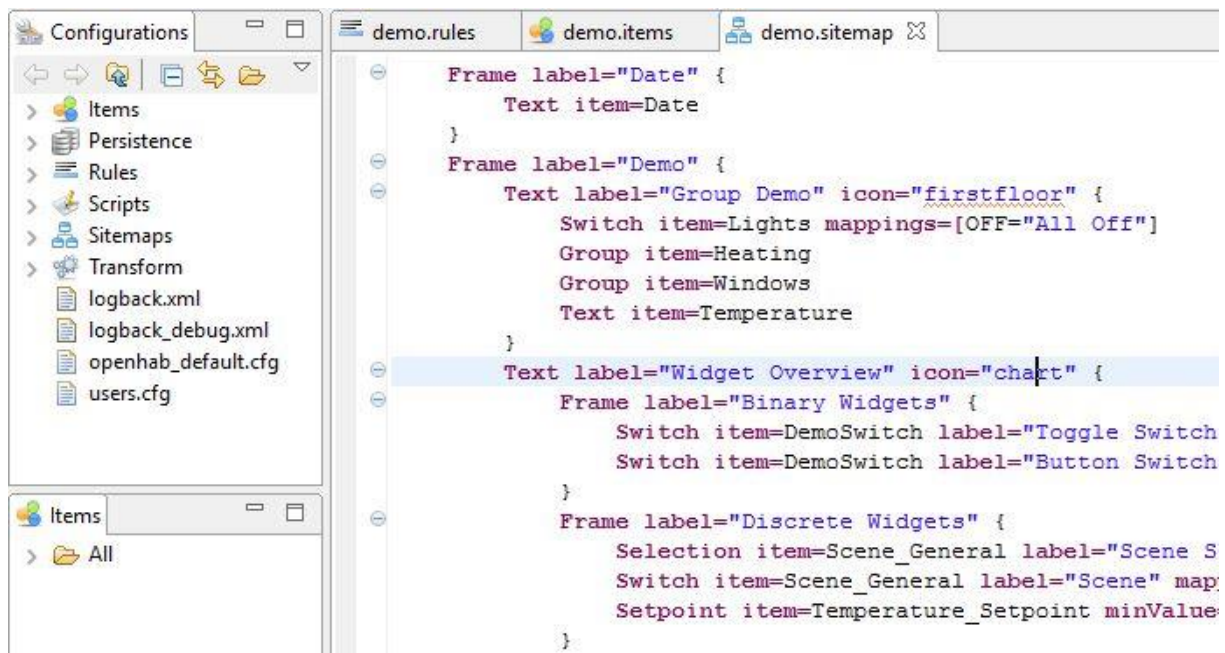
Antud süsteemi kasutajaliides (Lisa 1, Joonis 6) nõuab kiireks kasutamiseks mõningat harjumist. Seadistamiseks on valikus väga palju erinevaid võimalusi. Küll aga on majaplaani seadistamine üsna lihtne. Kõigepealt lisatakse majaplaanile korruseid ning samas vaates on võimalus kohe ka korrusele tube lisada. Samamoodi on lõpptarbija jaoks ka seadmete lisamine (Lisa 1, Joonis 7) tubadele võimalikult lihtsaks tehtud. Seadme lisamiseks tuleb käivitada otsingu funktsioon, mis leiab kõik saadavalolevad seadmed üles. Seejärel tuleb kasutajal need ükshaaval tubade külge lisada.

2.1.3 OpenHAB

OpenHAB on tarkvara, mis ühendab erinevaid targa maja lahendusi üheks tervikuks. Selle tarkvara suureks eeliseks on see, et see on vabavaraline ehk kõigile tasuta kättesaadav. Tarkvara arendamine ei ole ainult valitud arendajate kättes, vaid neid võib olla lõpmata palju. Sellepärast on openHAB turvalisem, töökindlam ja pidevas arengus. Samuti tähendab see ka seda, et openHAB-i on kõigil võimalik oma äranägemise järgi kohandada, kas siis funktsionaalsust juurde lisades või seda vähendades [6].

OpenHAB töötab ainult selle seadme peal, kuhu see on paigaldatud. See tähendab seda, et openHAB ei vaja toimimiseks väliseid servereid ning kogu info salvestatakse seadme enda kettale. Kuna antud tarkvara töötab täiesti iseseisvana ja on vabavaraline, siis puuduvad ka igakuised teenustasud.

OpenHAB-i kasutamiseks on vajalik Java programmeerimiskeele oskus ja eraldi installeerida openHAB Designer Eclipse-i programm, mille kaudu tehakse vajalikud seadistused seadmete ühendamiseks ning reeglite määramiseks. Antud programm ei ole küll kohustuslik, kuid siiski soovitatav, sest see oskab pakkuda kasutajale vajalikke käske ja parandab koodis esinevaid vigu. Selle programmi asemel võib kasutada ka muid tekstiredaktoreid, millega seadistusi sisaldavaid lähtefaili muuta.



Joonis 2. openHAB kasutajaliides

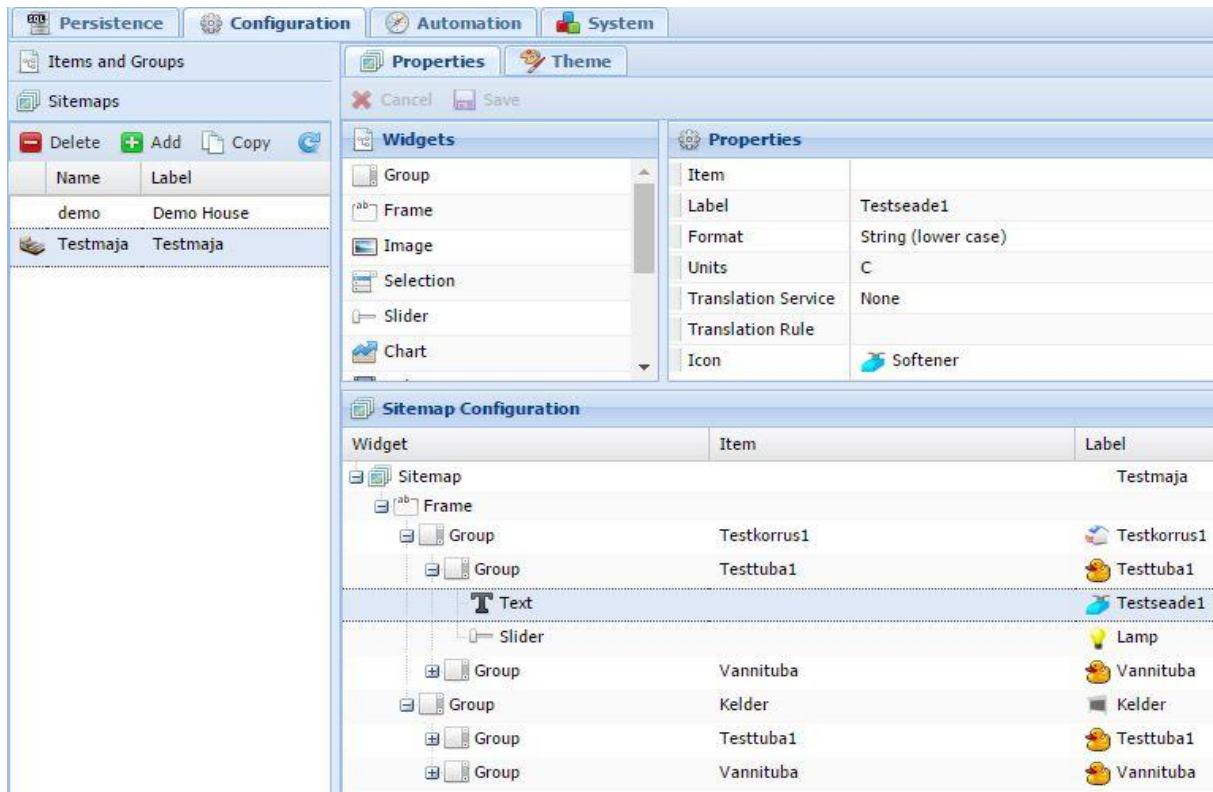
OpenHAB Designer kasutajaliides (vt Joonis 2) ei näe lõpptarbijale, kellel puuduvad kokkupuuted programmeerimisega, arusaadav välja. Oma majaplaan tuleb sisestada käsitsi koodifaili muutes ja teades korrektset süntaksit. Sama kehtib ka seadmete ja automatiseerimisreeglite lisamise kohta. Targa kodu seadistamiseks vajalikku süntaksit teadmata ei oska lõpptarbija sellise programmiga midagi teha. Lõpptarbija jaoks on see ebamugav ja keeruline. Esmasel kasutamisel võib kogu seadistuse tegemine aega võtta mitmeid tunde, mida oleks võimalik kasutajasõbraliku kasutajaliidesega vältida.

Antud töös on arendatud kasutajaliides openHAB-i jaoks. Peamine põhjus, miks töö autor openHAB-i valis, on see, et openHAB on avatud lähtekoodiga, vabavaraline ja võimaldab erinevad targa maja seadmed üheks tervikuks kokku liita. OpenHAB toetab paljusid erinevaid protokolle, nagu näiteks Z-Wave, ZigBee, Wi-Fi või Bluetooth. Samuti töötab openHAB igal platvormil, millel on peal Java virtuaalmasin.

2.1.4 HABmin

HABmin on veebiliides openHAB tarkvara jaoks. Selle eesmärk on pakkuda openHAB-ile kasutajaliidest, et teha openHAB tarkvara administreerimine tavakasutajale lihtsamaks. HABmin võimaldab kasutajal ilma programmeerimiskusteta oma kodu seadistada. Lisaks seadistamisele saab kasutaja näha ka graafilist statistikat seadmete andmete kohta.

Antud tarkvara on loodud tarkvaraarhitektuuri REST stiilis. Andmete pärimine, lisamine, uuendamine ja kustutamine käib läbi REST veebiteenuse. Lisaks on võimalik defineerida XML formaadis faile, mis kirjeldavad openHAB-i konfigureeritavaid seadistusi. Nendes failides on defineeritud seadmed ja nendega sidumiseks vajalikud seadistused.



Joonis 3: HABmin kasutajaliides

Automatiseerimisreeglite jaoks on HABmin-il graafiline liides (Vt Joonis 3). Liidest saab kasutada erinevaid plokkide lohistades ja omavahel ühendades. Nende plokkide sees määratakse ära seadmed, tingimused ja täidetavad tegevused, mida tehakse kui vajalikud tingimused on täidetud. Selles liidises saab kokku panna lihtsamaid reegleid, mille alusel seadmed töötavad [7].

HABmin-i kasutajaliides ei näe välja kasutajasõbralik ja esmapilgul tundub keeruline (Lisa 1, Joonis 2). Lõpptarbijale, kes soovib kiirelt ja mugavalt erinevaid seadmeid ühest kohast korruga juhtida, on liiga palju ebavajalikke seadistusi. Oma majaplaani seadistamiseks tuleb korruste ja tubade lisamiseks kõigepealt need eraldi seadmete lisamise vaates luua. Kui korrused ja toad on seadmete vaates loodud, tuleb need siiski veel uuesti majaplaani koostamisel luua. Selline topelt lisamine tekitab segadust ja nõuab ebavajalikke liigutusi lõpptarbijalt.

HABmin liides täidab küll oma eesmärgi, sest majaplaani saab koostada ja seadmeid sinna lisada, kuid see ei ole siiski kasutajasõbralik. Majaplaani seadistamine ning reeglite ja seadmete lisamine on kogematu tavakasutaja jaoks liialt keeruline. Samuti tekitab segadust osade elementide topelt lisamise vajadus.

Töö autor kasutab kasutajaliidese arendamiseks HABmin tarkvara võimalusi. Antud tarkvara on kasutatud sellepärast, et HABmin-i REST API võimaldab kätte saada kogu vajaliku info openHAB-ist ja seda kasutajaliideses kuvada. HABmin tarkvara pakub openHAB-ile kasutajaliidest, milles on antud tööks vajalikke osi juba realiseeritud. Seetõttu sobib antud tarkvara autori töösse hästi, sest juba olemasolevaid andmete päringuid openHAB-ist ei ole vaja uuesti välja töötada.

2.2 Kasutatavate tehnoloogiate valik

Kasutajaliidese realiseerimiseks vajaminev kood on kirjutatud HTML, Javascript ja CSS keeles, mille jaoks on kasutatud Notepad++ tekstiredaktorit. Notepad++ võimaldab muuta lähtekoodi ja toetab paljusid koodisüntakseid. Antud tekstiredaktor on valitud sellepärast, et see on tasuta saadaval, kiire ning lihtne ja mugav kasutada.

Veebiliidese testimiseks kasutatud veebilehitsejad:

- Chrome
- Mozilla
- Microsoft Edge
- Internet Explorer

2.2.1 JSON

JSON on lihtsustatud andmevahetus formaat, mis baseerub JavaScript programmeerimiskeele alamhulgal. Seda on inimestel lihtne lugeda ja kirjutada. Masinatel on seda lihtne parsida ja genereerida. JSON on tekstiformaat, mis on täielikult sõltumatu programmeerimiskeelest. See on rajatud kahele struktuurile:

- Nime ja väärtuse paarid
- Organiseeritud väärtuste jada

Need on universaalsed andmestruktuurid. Praktiliselt kõik kaasaegsed programmeerimiskeeled toetavad neid ühes või teises formaadis. On mõistlik, et andmeformaadid, mis on sõltumatu programmeerimiskeelest, põhineb nendel struktuuridel.

Antud töös kasutatakse JSON-it andmete salvestamisel ja saatmiseks REST API-le. Andmete pärimisel saadakse vastus JSON kujul, mis salvestatakse JavaScript-i objekti. Seda objekti kasutatakse andmete kuvamiseks kasutajaliideses ja hoitakse mälus kuni veebileheküljelt lahkumiseni.

2.2.2 Bootstrap

Bootstrap on populaarne raamistik rakenduste ning *responsive* ja mobiilisõbralike veebilehtede arendamiseks. See raamistik sisaldab kõrgekvaliteedilist CSS-i, HTML-i ja JavaScript-i [8]. Bootstrap sisaldab eeldefineeritud stiile ja elemente. Antud raamistikuga on võimalik veebileht struktureerida nii, et see töötaks kõikidel seadmetel ja enim-kasutatavatel brauseritel, näiteks Google Chrome, Mozilla Firefox, Internet Explorer või Safari.

Antud töös on valitud just Bootstrap raamistik, sest sellega on võimalik kiirelt luua kasutajaliides, mis oleks kasutatav võimalikult paljudel seadmetel. Loodava kasutajaliidese üheks nõudeks on selle mugav kasutus ka mobiilsetel seadmetel, mida Bootstrap võimaldab realiseerida.

2.3 Kasutajaliidese toimimine ja võimalused

Kõik anduritena käituvad seadmed on ühendatud ühe keskseadme külge. Keskseadme külge ühendatud seadmed suhtlevad ainult keskseadmega ning ei ole iseseisvalt internetiga ühenduses. Internetiga on ühenduses vaid keskseade, mille kaudu pääsetakse ligi kõikidele ühenduses olevatele seadmetele. Ilma keskseadme külge ühendamata, seadmetele ligi ei pääse.

Loodud kasutajaliidese toimimiseks on vajalik openHAB ja HABmin. Enne kasutajaliidese kasutamist, tuleb OpenHAB-i juurkaustas käivitada start.bat (Windows juhul) või start.sh (Linux või Mac juhul) fail, millega käivitatakse server, laetakse seadistusi sisaldavad lähtefailid ja käivitatakse REST API-d. Serveri töötamisel tuleb veebilehitsejas minna aadressile

http://<openHAB address või arvuti nimi (nt. localhost)>:8080/habmin/veeb/. Sellelt aadressilt saab seadistada majaplaani ning lisada andureid.

Loodud kasutajaliides pakub lõpptarbijale kiire ja mugava seadistamise võimaluse. Seadistamiseks kuvatakse vaid selleks vajalikku infot, et kogu protsess oleks lõpptarbijale võimalikult lihtne. Välja pakutud veebiliides võimaldab kasutajal seadistada oma targa kodu lahendus. Kasutajal on võimalik lisada majaplaane. Igale majaplaanile on võimalik lisada piiramatult korruseid, millele omakorda lisatakse tube. Samuti saab lisada igale toale seal asuvaid seadmeid. Kasutajaliides võimaldab ka neid kõiki muuta või kustutada. Seadmetele on võimalik lisada ka tingimusi, mille alusel nad töötavad. Näiteks saab lisada seadmele reegli, et kui temperatuuri andur näitab vähemalt 22°C, lülitatakse kütteseade välja.

Töö autor on lõputöö kirjutamise käigus loonud reeglite lisamiseks ja füüsiliste seadmetega ühendamiseks vaid kasutajaliidese poole. Seetõttu ei ole võimalik liidesest reaalse seadmetega ühendusi luua ega neid etteantavate reeglitega juhtida. Liides on loodud eesmärgiga välja pakkuda lahendus, milline võiks välja näha targa maja seadistamise protsess.

2.4 Kasutajaliidese vaated

Kasutajasõbralikkuse tagamiseks on loodud kasutajaliides, mis koosneb erinevatest vaadetest. Vaadete kaudu tehakse päringuid ja tagastatakse info lähtefailidest või salvestatakse nendesse.

URL	Kirjeldus
/index.html	Kasutajaliidese avaleht, kus kuvatakse olemasolevate majaplaanide viited
/loomine.html	Kuvab uue majaplaani loomise vormi
/majaplaan.html	Kuvad loodud majaplaani seadistused
/seade.html	Kuvab virtuaalse seadme füüsilise seadmega ühendamise vaate
/reeglid.html	Kuvab seadmetele reeglite lisamise vaate

Tabel 1: Kasutajaliidese vaated

Igal vaatel on olemas menüüriba. Menüürealtil on võimalik alati liikuda tagasi avalehele või alustada uue majaplaani loomist.

2.4.1 REST otspunktid

Kasutajaliides saab kõik oma andmed kasutades päringuid REST API-le. REST API-i tagastab päringule vastuseks JSON formaadis vastuse, mis kuvatakse kasutajaliideses. Samamoodi saadetakse ka andmete salvestamiseks JSON kujul info REST API-le.

URL HTTP meetod	Kirjeldus	Saadetav info
/services/habmin/config/sitemap GET	Tagastab kõik olemasolevad majaplaanid.	-
/services/habmin/config/sitemap/{id} GET	Tagastab majaplaani seadistused.	{id} – majaplaani nimi.
/services/habmin/config/sitemap/{id} PUT	Lisab uue majaplaani.	{id} – majaplaani nimi.
/services/habmin/config/sitemap/{id} DELETE	Kustutab olemasoleva majaplaani	{id} – majaplaani nimi.
/services/habmin/config/items/{id} GET	Tagastab info seadme kohta.	{id} – seadme nimi.
/services/habmin/config/items/{id} PUT	Lisab uue seadme.	{id} – seadme nimi.

Tabel 2: Kasutatavad REST otspunktid

2.4.2 Avaleht

Veebiliidese käivitamisel tehakse päring HABmin-i REST API-le, et leida olemasolevate majaplaanide seadistused. Majaplaanide olemasolu korral tagastab päring seadistused, mis salvestatakse JavaScript-i objekti. Süsteem kuvab saadud info põhjal majaplaanide loetelu (Lisa 1, Joonis 8). Majaplaanile vajutades avaneb menüü, millel on võimalik nii kustutada kui ka vaadata majaplaani ülevaadet ning lisada reegleid ja seadmeid. Majaplaanide puudumise korral kuvatakse kasutajale viide uue majaplaani loomiseks, millele vajutades avaneb majaplaani lisamise vaade.

Vajutades majaplaani vaatamisele, suunatakse kasutaja edasi valitud majaplaani vaatele, kus on võimalik näha majaplaani ülevaadet ning teha muudatusi. Majaplaani eemaldamise korral saadetakse päring HABmin-i REST API-le, pärast mida kustutatakse majaplaaniga seotud seadistuste failid failisüsteemist ning eemaldatakse valitud majaplaani viide avalehelt. Nii reeglite kui ka seadete lisamistele vajutades avanevad vastavad vaated seadistuste tegemiseks.

2.4.3 Majaplaani lisamine

Majaplaani lisamise vaates (Lisa 1, Joonis 9) alustab kasutaja oma maja kirjeldamist. Kõigepealt sisestatakse majaplaanile nimetus, mille järgi seda teistest eristada. Samuti on võimalus lisada majaplaani iseloomustav ikoon. Peale vormi salvestamist saadetakse info REST API-le, mille tulemusena luuakse failisüsteemi <majaplaani nimetus>.sitemap seadistusi sisaldav fail. Õnnestunud salvestuse korral suunatakse kasutaja edasi loodud majaplaani vaatesse, kus tehakse edasised seadistused.

Olles suunatud majaplaani vaatesse, kuvatakse kasutajale kõigepealt menüü, kus on kirjas hetkel olemasolev majaplaan. Majaplaanile vajutades on võimalik seda kas muuta või alustada korruse lisamisega. Ühtlasi on võimalik ka pooleli olev majaplaan kohe kustutada.

Korruse lisamise vormil saab kasutaja varem loodud majaplaanile korruseid lisada. Korruse lisamiseks peab kasutaja sisestama vaid korruse nimetuse ning võib valida korrust iseloomustava kujutise. Süsteem lisab loodava korruse automaatselt valitud majaplaani külge. Samuti lisatakse seadistused REST API kaudu <majaplaani nimetus>.sitemap faili. Pärast andmete salvestamist uuendatakse majaplaani menüüd. Kasutaja võib koheselt lisada juurde uusi korruseid või valida menüüst loodud korruse. Korruse valimisel kuvatakse kasutajale nupp „Lisa tuba“, millele vajutades avaneb korruse lisamise vorm.

Võrreldes HABmin kasutajaliidesega, ei pea kasutaja enam korruseid majaplaaniga käsitsi ühendama. Korruste ühendamine majaplaaniga toimub automaatselt. Samuti ei pea kasutaja kõigepealt looma korrust seadmete lisamise vaates ja siis uuesti majaplaani lisamisel.

Toa lisamine on samuti loodud kasutajaliidesega lihtsam. Uue toa lisamisel on kohustuslik täita vaid toa nimi ning soovi korral võimalus valida tuba iseloomustav kujutis. Toa lisamine korruse külge toimub samamoodi automaatselt nagu ka korruse lisamine majaplaani külge. Niimoodi tehes peab kasutaja vähem liigutusi tegema, mis omakorda lihtsustab ja mugavdab kogu seadistamise protsessi. Toa andmete salvestamisel lisatakse REST API kaudu seadistused

<majaplaani nimetus>.sitemap faili ning uuendatakse majaplaani menüüd. Loodud toa valimisel menüüst, kuvatakse nupule „Lisa seade“ vajutades kasutajale seadme lisamise vorm.

Seadme lisamise vaade (Lisa 1, Joonis 10) avaneb, kui toa lisamine on tehtud, soovitakse lisada varasemalt loodud toale seadmeid juurde või tahetakse olemasolevaid seadmeid muuta. Antud vaates on võimalik sisestada lisatava seadme info või seda muuta. Kuvataval vormil on järgnevad väljad:

- Nimi – seadme nimetus;
- Tüüp – seadme olemus;
 - Lüliti
 - Number
 - Värvivalik
 - Liugur
 - Tekst
- Formaat – seadme poolt tagastatavate andmete formaat;
- Ühik – seadme poolt tagastatavate andmete ühik;
- Ikoon – seadet iseloomustav kujutis;

Pärast vormi salvestamist lisatakse uus seade valitud toa külge ning läbi REST API uuendatakse ka <majaplaani nimetus>.sitemap faili ja luuakse <seadme nimi>.items fail. Õnnestunud salvestamise korral uuendatakse majaplaani menüüd ning kasutajal on võimalus koheselt uusi seadmeid juurde lisada. Valides menüüst loodud seadme, kuvatakse kasutajale lisaks „Kustuta“ ja „Uuenda“ nuppudele „Ühenda seade“ nupp, mis suunab kasutaja edasi seadmega ühendamise vaatesse.

2.4.4 Majaplaani vaade

Antud vaates kuvatakse kogu info salvestatud majaplaani kohta. Erinevate majaplaanide nägemiseks saadetakse URL parameetrina majaplaani nimetus: [http://<openHAB aadress või arvuti nimi \(nt. localhost\)>:8080/habmin/veeb/majaplaan.html?id=<majaplaani nimetus>](http://<openHAB aadress või arvuti nimi (nt. localhost)>:8080/habmin/veeb/majaplaan.html?id=<majaplaani nimetus>).

URL-i parameeter „maja“ väärtus saadetakse Ajax päringuga REST API-le, mis tagastab vastusena küsitud majaplaani seadistused.

Majaplaani vaates kuvatakse kasutajale menüü, milles on välja toodud majaplaani struktuur. Antud struktuurist on võimalik valida majaplaaniga seotud korruseid, tube ja seadmeid. Iga valiku kohta kuvatakse kasutajale võimalikud seadistused. Seadistusi on võimalik kohe muuta (Lisa 1, Joonis 11) ning salvestada nupule „Uuenda“ vajutamisega. Seadete uuendamisel saadetakse andmed REST API-le, mille tulemusena muudetakse lähtefaile vastavalt valitud seadistustele.

Valitud elemente on võimalik ka kustutada, vajutades nupule „kustuta“. Nupule vajutamisega saadetakse päring REST API-le, mille tulemusena eemaldatakse valitud element nii majaplaani küljest kui ka muudetakse seadistust sisaldavat lähtefaili. Majaplaani kustutamise korral suunatakse kasutaja edasi avalehele, kust on võimalik alustada uue majaplaani loomist või muuta olemasolevaid.

Samuti saab antud vaatest lisada juurde korruseid, tube või seadmeid. Vastavalt sellele, milline element on valitud, on võimalik lisada juurde selle alla kuuluvaid elemente. Sellisel juhul on võimalik igal hetkel olemasolevat majaplaani täiendada, kui kasutaja peaks tahtma majale seadmeid juurde lisada või nende asukohta majas muuta.

2.4.5 Seadmega ühendamise vaade

Avalehel olevatelt majaplaanidelt on võimalik minna seadmete ühendamise vaatesse (Lisa 1, Joonis 12). Antud vaates saab kasutaja valida menüüst virtuaalse seadme, millega tahetakse ühendada füüsilist seadet. Menüüs on vaid valitud majaplaaniga seotud seadmed, mis on saadud läbi REST API-le saadetud päringu.

Seadme ühendamiseks tuleb vajutada nupule „Otsi“, millega alustatakse seadmega ühendamise protsessi. Sellega lülitatakse keskseade vastuvõtu režiimi. Keskseade üritab 30 sekundi jooksul leida füüsilisi seadmeid, mis on lülitatud enda tutvustamise režiimi. Tutvustamise režiim tuleb igal füüsilisel seadmel otsimise hetkel eraldi sisse lülitada. Tutvustamise ja vastuvõtu režiim on turvaline ühendamise meetod, sest nii välistatakse võõrastele seadmetele info edastamine.

2.4.6 Reeglite lisamise vaade

Antud vaates (Lisa 1, Joonis 13) on võimalik lisada reegleid, mille alusel seadmed töötavad. Seadmete valiku menüüsse kuvatakse need seadmed, mis on valitud majaplaaniga seotud.

Selleks saadetakse päring REST API-le, mille käest küsitakse valitud majaplaani info. Saadud info põhjal leitakse majaplaaniga seotud seadmed ning kuvatakse need seadmete valiku menüüsse.

Kasutaja saab valida erinevate tingimuste vahel, mis omakorda kutsuvad välja teatud tegevused. Kuvatud menüüdest on võimalik valida tingimus ning sisestada väärtus. Näiteks valida tingimuste menüüst „Temperatuur suurem kui“ ning lahtrisse sisestada soovitud temperatuuri väärtus. Tegevuste menüüst on võimalik valida sündmus, mis toimub kui tingimus on täidetud. Samuti tuleb valida seade, mille puhul peab tingimus täidetud olema ning seade, mis valitud tegevust täidab.

3. Tulemuste analüüs

Antud peatükis analüüsib autor loodud kasutajaliidese vastavust nõuetele ning pakub välja tulevikuarendused, mis oleksid vajalikud enne liidese pakkumist lõpptarbijatele kasutamiseks.

3.1 Kasutajaliidese vastavus püstitatud nõuetele

Loodud kasutajaliides vastab püstitatud nõuetele. Kasutajaliidese käivitamiseks sobivad kõik enimkasutatavad veebilehitsejad. Samuti on liides kasutatav nii nutiseadmetes kui ka tavaarvutites. Loodud liides on lihtne, arusaadav ja mugav lõppkasutaja jaoks.

Kasutajal on võimalik vähese ajaga koostada ja kustutada majaplaane, korruseid, tube ning seadmeid. Seadmetega ühendamiseks peab kasutaja vajutama otsingu nuppu ning kogu ülejäänud ühendamine peab toimuma süsteemi enda poolt. Sama lihtsalt on reeglite lisamine tehtud valikmenüüdega, et kasutaja ei peaks tundma koodisüntaksit, millega seadmetele käske anda.

3.2 Võimalikud tulevikuarendused

Kasutajaliidese tulevikuarendustest kõige olulisem oleks füüsilise seadmega ühendamise realiseerimine. Füüsilise seadmega ühendus võimaldaks kasutajatel reaalselt süsteemi kasutada ning oma kodus rakendada. Samuti annab see võimaluse arendada edasi ka seadmete juhtimist reeglite abil. Reeglite lisamise arendamine muudab süsteemi terviklikuks ning seda oleks võimalik lõpptarbijatele pakkuda.

Tulevikuarenduste hulka võiks lisada ka seadmete juhtimise lisamise. Antud töös käsitletakse kasutajaliidese loomist targa maja seadistamiseks, mitte selle juhtimiseks. Loodud kasutajaliides ei paku võimalust seadmeid juhtida või nendelt infot saada. Seadmete juhtimise lisamine kasutajaliidesesse eemaldaks vajaduse kasutada lisa kasutajaliidest.

Kasutajaliidese kujundust tuleks muuta paremaks ning kasutajatele atraktiivsemaks. Et kasutajaliidest tarbijatele pakkuda, peab see usaldusväärne ning esinduslik välja nägema. Antud töö ei keskendunud kasutajaliidese atraktiivsusele, vaid selle mugavusele, mistõttu vajaks liides ümberkujundamist.

4. Kokkuvõte

Antud töö põhieesmärgiks oli targa maja lahenduse kasutajaliidese arendus lõpptarbijale, toetudes töö autori eelnevalt tehtud olemasolevate liideste analüüsile. Läbiviidud analüüsi tulemustest järeldus, et kogu vajalik funktsionaalsus on realiseeritud, kuid enamasti ei ole olemasolevad liidesed lõpptarbijaja jaoks kasutajasõbralikud.

Töö tulemusena valminud kasutajaliides võimaldab mugavalt ja lihtsalt targa maja lahendust seadistada. Loodud liideses on mõeldud kasutajasõbralikkusele. Lõpptarbijaja saab mugavalt ühest kohast seadistada kogu majaplaani ilma erinevate vaadete vahel liikumiseta. Töö autori loodud liidese lõpptarbijatel on võimalus lisada majaplaane, seadmeid ning neile omakorda automatiseerimisreegleid. Liides toimib kõikides enimlevinud veebilehitsejates ja on mobiilisõbralik.

Liidese edasiseks arenduseks võiks realiseerida füüsilise seadmega ühendamise. Loodud liideses on valmis tehtud seadmega ühendamise vaade, kuid edasistes arendustes tuleb sinna lisada ka reaalse ühenduse loomine. Selle arenduse tulemusena on võimalik edasi arendada ka automatiseerimisreeglite lisamist, mis on vajalik liidese terviklikkuse saavutamiseks. Sellisel juhul saaksid seadmed omavahel suhelda ning sooritada automaatseid tegevusi.

Summary

The main aim of this thesis was the development of smart home user interface for the end user, based on the author's analysis of the existing interfaces. The result of the analysis concluded that all of the necessary functionality was realized, but in most cases, these interfaces were not user friendly for end users.

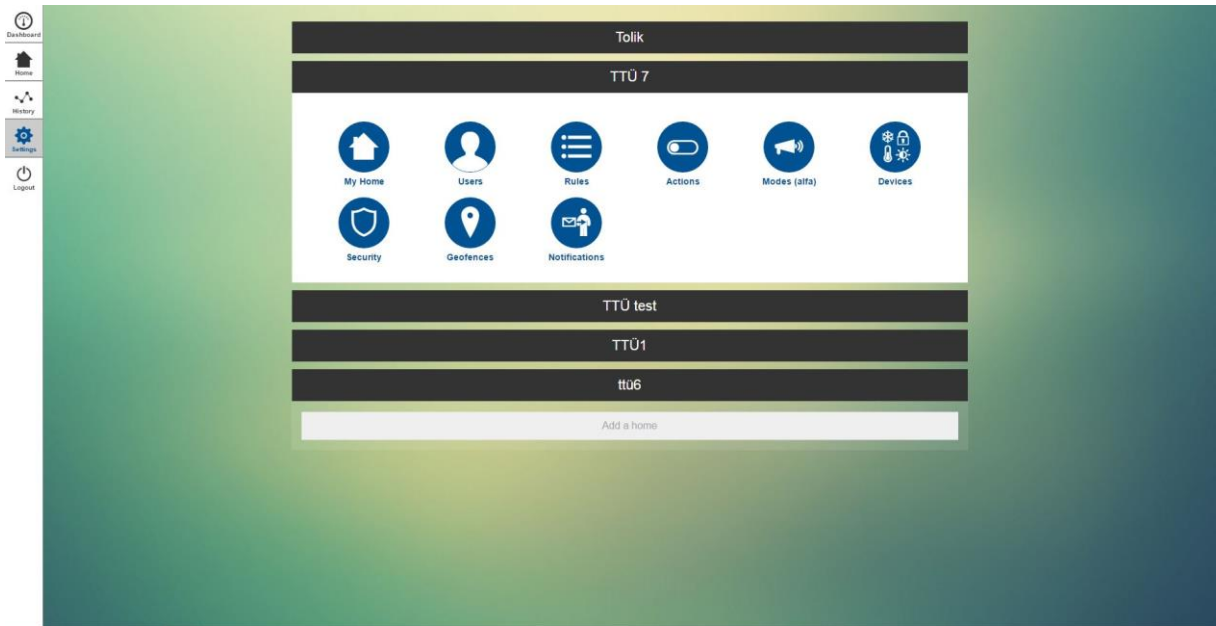
The resulting user interface allows convenient and easy configuration of smart home solution. Created interface is designer to be user friendly. The end user can comfortably set up the whole house plan without the need to switch between different views. The author's created interface allows end users to add house plans, devices and add automation rules to them. The interface functions in all of the most popular browsers and is mobile-friendly.

Further development of the created interface could involve the development of the realization of connecting a physical device to the interface. The interface includes the view of connecting physical device, but further development should realize the actual connection also. As a result of that, further development of adding automation rules is also necessary to make this interface a complete package. In this case, the devices can communicate with each other and perform automated actions.

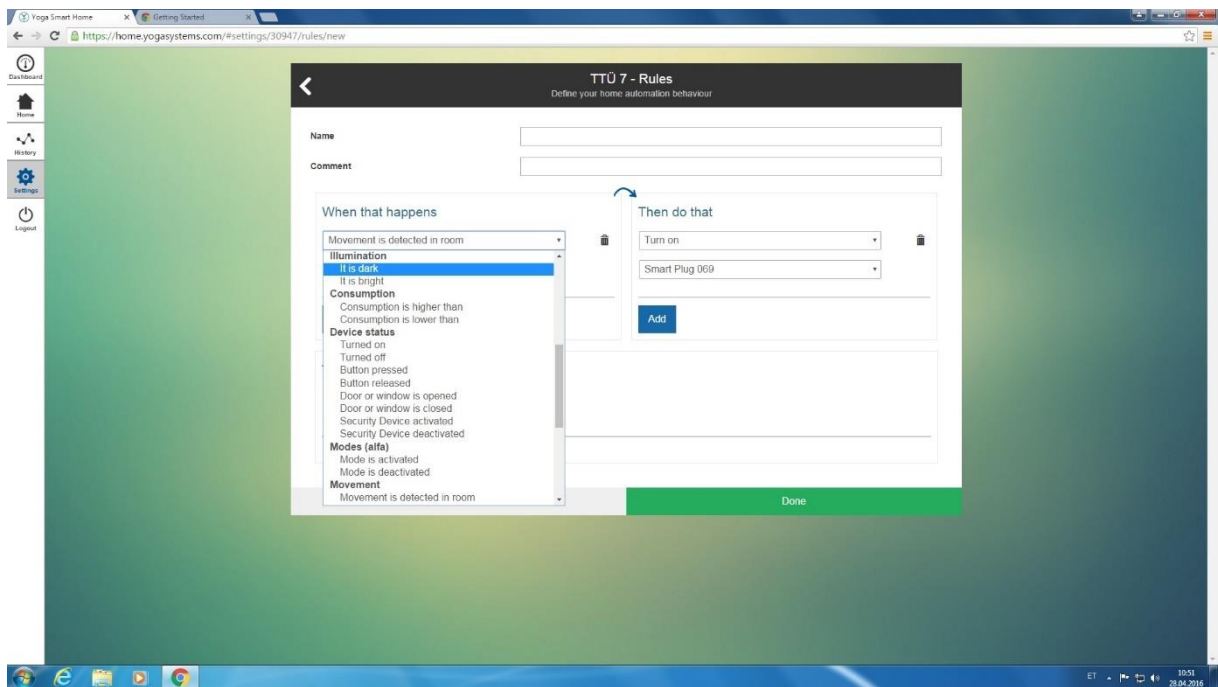
Kasutatud kirjandus

- [1] M. Bauer, S. Haller, A. Serbanati ja F. Carrez, „A Domain Model for the Internet of Things,“ 2013.
- [2] „TechTarget,“ [Võrgumaterjal]. Available: <http://internetofthingsagenda.techtarget.com/definition/smart-home-or-building>. [Kasutatud Mai 2016].
- [3] „Majaehitaja,“ [Võrgumaterjal]. Available: <http://www.majaehitaja.ee/mis-asi-on-tark-maja/>. [Kasutatud Aprill 2016].
- [4] J. L. Fern´andez, D. . P. Losada ja E. P. Domonte, „An Integrated and Low Cost Home“.
- [5] „Yoga Systems,“ [Võrgumaterjal]. Available: <https://www.yogasystems.com/about-us/>. [Kasutatud April 2016].
- [6] „OpenHAB,“ [Võrgumaterjal]. Available: <http://www.openhab.org/features/introduction.html>. [Kasutatud Aprill 2016].
- [7] „Github,“ [Võrgumaterjal]. Available: <https://github.com/cdjackson/HABmin>. [Kasutatud 2016].
- [8] „Bootstrap,“ [Võrgumaterjal]. Available: <http://v4-alpha.getbootstrap.com/getting-started/introduction/>. [Kasutatud Aprill 2016].

Lisa 1



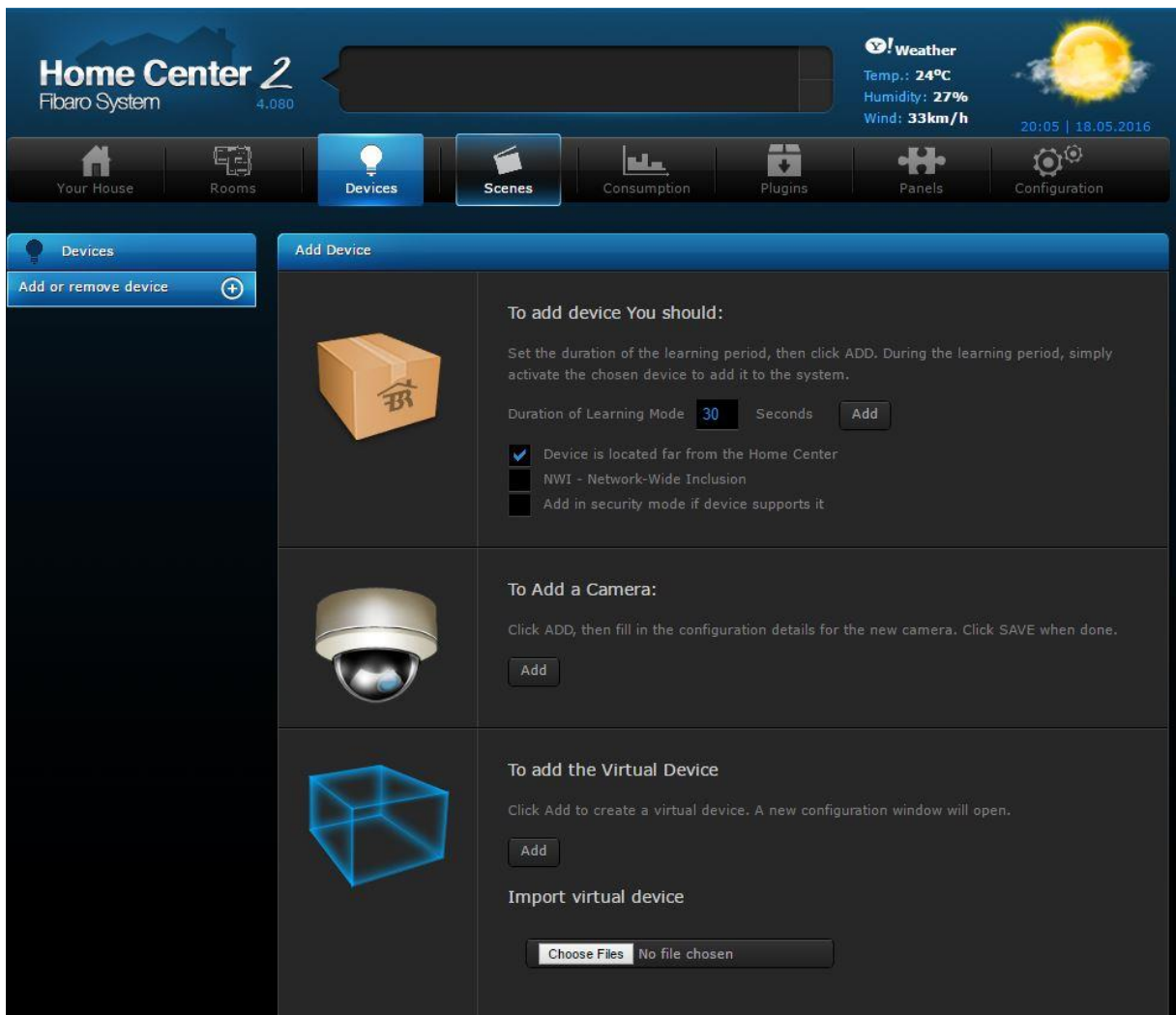
Joonis 4: Yoga Systems kasutajaliidese majaplaani seadistamise vaade



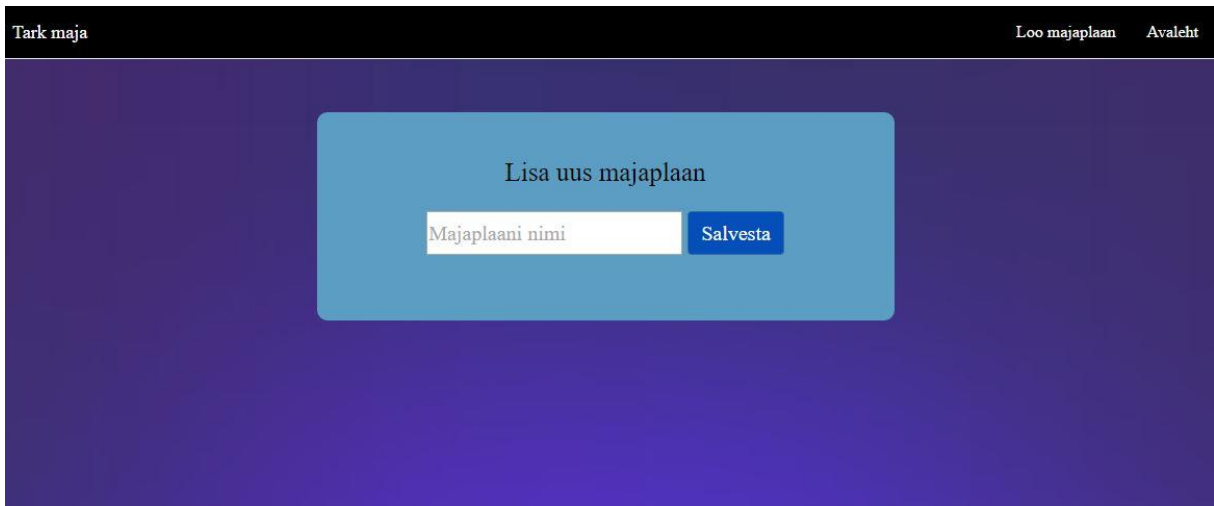
Joonis 5: Yoga Systems kasutajaliidese reeglite lisamise vaade



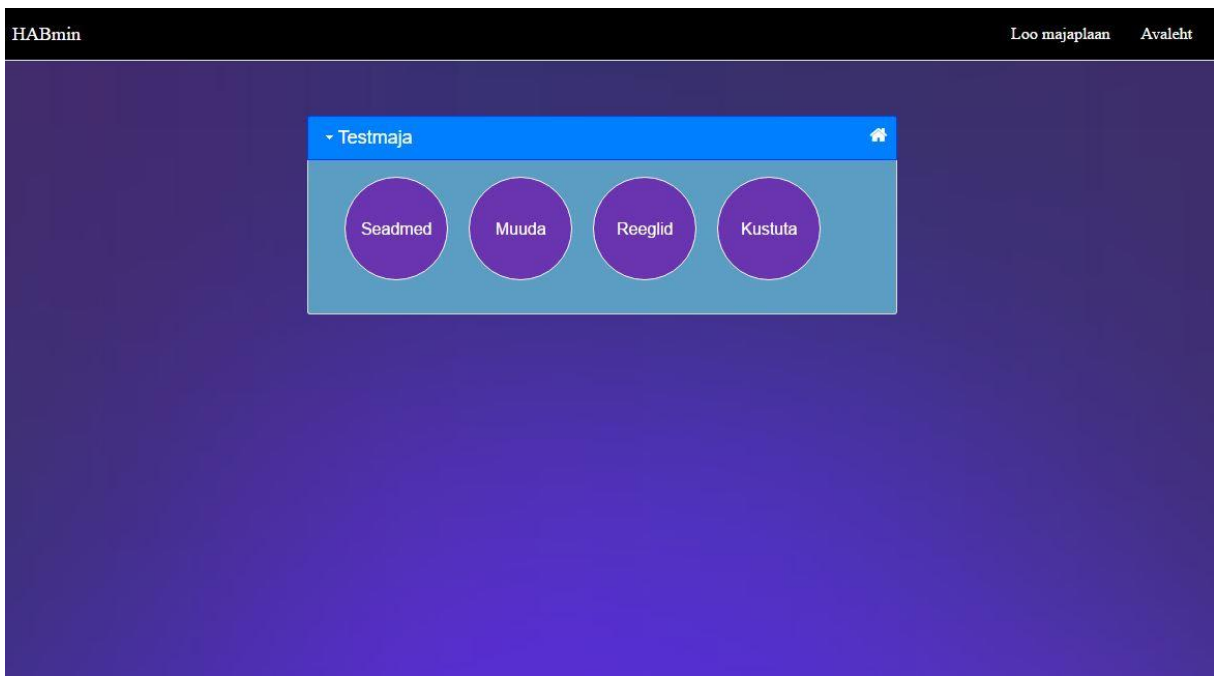
Joonis 6: Fibaro kasutajaliides



Joonis 7: Fibaro seadme lisamine



Joonis 8. Majaplaani loomise vaade




Joonis 9. Avalehe vaade

HABmin Loo majaplaan [Avaleht](#)

- Testmaja
- Testkorrus1
- Testtuba1
- Testseade1
- Lamp
- Vannituba
- Kelder

Muuda tuba

Ikoon:

Lisa seade
Uuenda
Kustuta

Lisa uus seade


Nimi


Tüüp

Ühik

Formaat

Ikoon:

Salvesta 

Joonis 10. Seadme lisamise vaade

HABmin Loo majaplaan [Avaleht](#)


- Testmaja
- Testkorrus1
- Testtuba1
- Testseade1
- Lamp
- Vannituba
- Kelder

Muuda seadet

Ühik

Formaat

Ikoon:

Uuenda
Kustuta

Joonis 11. Seadme muutmine

Tark maja Loo majaplaan Avaleht

Vali ühendatav seade

Vali...

Füüsilise seadme otsimine

Otsi...

Tagasi Salvesta

Joonis 12. Seadmega ühendamise vaade

Tark maja Loo majaplaan Avaleht

Reegli nimi

Reegli nimi

Vali tingimust täitev seade

Seade 1

Tingimus

Vali...

Väärtus

Vali tegevust teostav seade

Seade 1

Tegevus

Vali...

Tagasi Salvesta

Joonis 13. Reeglite lisamise vaade

Lisa 2

Programmikood on üles laetud aadressile:

<https://www.dropbox.com/sh/x35tfhp51mmseub/AADyDOUq7dHsztL2EoR1QFlpa?dl=0>