

TARTU ÜLIKOOL
Matemaatika-informaatikateaduskond
Arvutiteaduse instituut
Informaatika õppekava

**Projekti “Household of the future virtual reality” arendus
ja juhtimine Scrum meetodil COVID-19 pandeemia ajal**

Bakalaureusetöö (9 EAP)

Hanna-Marii Kaljas
Juhendaja: Anne Villems, MSc (informaatika)

Tartu 2020

Projekti “Household of the future virtual reality” arendus ja juhtimine Scrum meetodil COVID-19 pandeemia ajal

Lühikokkuvõte:

Bakalaureusetöö eesmärk on teostada projekti “*Household of the future virtual reality*” arendus ja juhtimine Scrum meetodil. Töös antakse ülevaade agiilsest projektijuhtimisest ja selle kolme levinumast meetodist, kaasa arvatud lõputöö projektis rakendatud Scrum meetodist. Töös tutvustatakse projekti “*Household of the future virtual reality*” probleemi, projekti tellinud kliente ja projekti planeerimise etappi. Kirjeldatakse töös kasutatud uurimismeetodeid, kliendi poolt esitatud funktsionaalseid ja mittefunktsionaalseid nõudeid tootele ning nõuetel põhinevat stsenaariumite disaini. Viimaks antakse ülevaade COVID-19 poolt põhjustatud organisatoorses probleemidest, Scrum meetodi ja projekti tulemuslikkusest ning töö lõpus tutvustatakse võimalikke edasiarendusvõimalusi.

Võtmesõnad:

Agiilne projektijuhtimine, Scrum, videosimulatsioon, “Tark maja”

CERCS: P170 Arvutiteadus, arvanalüüs, süsteemid, kontroll

Project “Household of the future virtual reality” development and management through Scrum method turning the COVID-19 pandemy

Abstract:

The purpose of this Bachelor’s thesis was to develop and manage the project "Household of the future virtual reality" using the Scrum method. Thesis describes agile project management and its methods. Also a Scrum method, which was implemented in the project. In addition, this thesis gives an overview of the project problem, introduces the clients and the project planning process. Research methods, functional requirements, non-functional requirements and scenario design is thoroughly described. Finally, the developing process is shown in detail and all emerged problems are highlighted. In the end further development opportunities are described.

Keywords:

Agile project management, Scrum, video simulation, “Smart House”

CERCS: P170 Arvutiteadus, arvanalüüs, süsteemid, kontroll

Sisukord

Sissejuhatus	5
1. Projektijuhtimine	7
1.1 Projekti ja projektijuhtimise mõiste	7
1.2 Agiilne projektijuhtimine	7
1.2.1 Agiilne manifest	7
1.3 Agiilsed projektijuhtimise meetodid	9
1.3.1 Ekstreemprogrammeerimine	9
1.3.2 Kanban	13
1.3.3 Scrum	15
2. Projekt “Household of the future virtual reality”	21
2.1 Probleemipüstitus	21
2.2 Kliendid	23
2.3 Planeerimine	24
2.3.1 Ajakava	24
2.3.2 Rollid ja vastutus	25
2.3.3 Platvorm, programmeerimiskeel ja riistvara	26
3. Projekti stsenaariumite arendus	29
3.1 Uurimismeetodid	29
3.2 Funktsionaalsed ja mittefunktsionaalsed nõuded	29
3.3.1 ”Tulevikukodu” projekti funktsionaalsed nõuded	30
3.3.2 Mittefunktsionaalsed nõuded	32
3.3 Simulatsiooni kvalitatiivne uuring	33
3.3.1 Koduautomaatika	35
3.4 Stsenaariumite disain	38
3.4.1 “Pimedas vannituppa” stsenaarium	40
3.4.2 Stsenaarium “Infrapunaküte”	42
4. Projekti tulemuslikkus	45
4.1 Tekkinud probleemid	45
4.2 Projekti protsess	46
4.2.1 Nõuete täitmine	55
4.3 Edasiarendusvõimalused	59
Kokkuvõte	61
Viidatud kirjandus	62
Lisa 1. Kliendi intervjuu küsimused	67
Lisa 2. Pildid BAM-i majast	69

Lisa 3. LED-tulede energia ja maksumuse efektiivsus	71
Lisa 4. Infrapunakütte energia ja maksumuse efektiivsus	73
Lisa 5. Projekti mõttekaart	74
Lisa 6. “Household of the future virtual reality” manual	75
Lisa 7. Menüü ja karakterite valik simulatsioonis	81
Lisa 8. LED-tulede ja infrapuna küttepaneelid simulatsioonis	82
Lisa 9. Automatiseeritud LED-tulede paigaldamise maksumus BAMi poolt	84
Lisa 10. “Tulevikukodus” lõpptarne kliendile	87

Sissejuhatus

Projekt on ühekordne kindla eesmärgi nimel, konkreetse ajavahemiku sees ja piiritletud ressurssidega läbi viidud tegevuste teostamise protsess. Projektijuhtimine koosneb planeerimisest, organiseerimisest ja ressursside haldamisest, sihiga saavutada seatud eesmärk [1]. Tarkvaraarenduses kasutatakse sagedasti agiilseid projektijuhtimismeetodeid ehk välearendusmeetodeid [2], mille eesmärgiks on väljastada kõrge kvaliteediga toode kliendile võimalikul kiiresti ja efektiivselt [3]. Antud töös käsitletud projekti “*Household of the future virtual reality*” (edaspidi “Tulevikukodu”) juhtimismeetodina rakendatakse välearendusmeetodit Scrum.

Projekt viidi läbi Hollandis, Hanze Rakenduskõrgkoolis programmi “Tark Energia” (*Smart Energy*) raames viie kuu vältel 2020. aasta kevadel. Projekt oli pühendatud rahvastiku vananemise probleemile, mille tõttu kasvab pensionäride arvu ühiskonnas. Hollandis ennustatakse, et 2050. aastaks on 50 ja vanemate inimeste arv ühiskonnas ligi 8 miljonit, mis on ligi pool Hollandi elanikkonnast [4]. See omakorda suurendab tervishoiuteenuste nõudlust ning hooldekodude täituvust. Mitmed Hollandi ettevõtted ja organisatsioonid otsivad seetõttu innovatiivseid lahendusi energiatõhusateks elamuteks, mis soodustaksid vanurite iseseisvat toimetulekut kodus ja tõstaks nende elukvaliteeti. Fookus on suunatud Targa maja (*Smart House*) arendustele, mille abil peaksid eakad võimalikult hilja oma isiklikust kodust hoolekandekeskustesse kolima. Projekti “Tulevikukodu” tellijateks on neli Hollandi organisatsioon - Groningeni linnavalitsus, Hollandi ehitusettevõtte BAM, Hollandi telekom ettevõtte KPN ja Hanze Rakenduskõrgkooli energia teaduslabor, kelle sooviks on luua virtuaalreaalsusel põhinev simulatsioon Targas majast, mis sisaldab tervisliku vananemise ja energiatõhususe aspekte. Simulatsiooni läbimise järel peaksid eakad mõistma koduautomaatika seadmete ja süsteemide kasulikkust nii energiaefektiivsuse kui ka terviseriskide vältimise vallas. Selle bakalaureusetöö eesmärgiks on teostada projekti “Tulevikukodu” arendus ja juhtimine Scrum meetodil.

Bakalaureusetöö koosneb neljast osast. Esimeses osas tutvustatakse agiilset projektijuhtimist ja selle kolme levinumat meetodit, kaasa arvatud lõputöö projektis rakendatud Scrum meetodit. Teises osas antakse ülevaade projekti “Tulevikukodu” probleemist, projekti tellinud klientidest ja projekti planeerimise etapist. Kolmandas osas kirjeldatakse töös kasutatud

uurimismeetodeid, kliendi poolt esitatud funktsionaalseid ja mittefunktsionaalseid nõudeid tootele ning nõuetel põhinevat stsenaariumite disaini. Neljandas osas antakse ülevaade COVID-19 poolt põhjustatud organisatoorsetes probleemidest, Scrum meetodi ja projekti tulemuslikkusest ning töö lõpus tutvustatakse võimalikke edasiarendusvõimalusi.

1. Projektijuhtimine

Selleks, et saavutada lõputöös käsitletava projekti eesmärk on vaja valida teostuseks sobiv projektijuhtimismeetod. Meetodi valiku langetamiseks on tarvis teada projektijuhtimise põhialuseid ja meetodite sisu. See peatükk keskendub kolme enim levinud agiilse projektijuhtimismeetodi uurimisele.

1.1 Projekti ja projektijuhtimise mõiste

Projekt on ainulaadsete, keerukate ja ühendatud tegevuste jada, millel on üks eesmärk ja mis tuleb konkreetse aja, eelarve piires ja vastavalt spetsifikatsioonile lõpule viia [5].

Projektijuhtimise puhul on tegemist tegevustega, mis on suunatud konkreetse eesmärgi saavutamiseks piiratud aja ning ressursside piires [6]. Tegevuse käigus kaasatakse pidevalt projekti tellijat ehk klienti eesmärgiga saada ülevaade tema vajadustest ja protsessi lõppedes projekti lõpptulemusega kasvatada organisatsiooni väärtust [5]. Projektijuhtimine koosneb peamiselt neljast etapist: teostatavuse etapp ehk käivitamine; planeerimine; teostamine ehk elluviimine; lõpetamine [6]. Järgmistes peatükkides selgitatakse agiilse projektijuhtimise metoodika ja selle populaarsemad meetodid.

1.2 Agiilne projektijuhtimine

Projektijuhtimisel on mitmeid erinevaid metoodikaid, mille valikust oleneb palju projekti õnnestumine. Antud lõputöö keskendub agiilsete projektijuhtimismeetodite (*Agile development*) ehk välearendusmeetodite [7] uurimisele, mis on üks populaarsemaid tarkvara arenduse lähenemisviise. Välearenduse eesmärgiks on väljastada kõrge kvaliteediga toode kliendile kiirelt ja efektiivselt. Arendusprotsessi jooksul on nii kliendil kui ka arendusmeeskonnal lai ja paindlik ülevaade projekti arengust. Välearendus on kirjeldatud agiilse manifestiga (*Manifesto for Agile Software Development*) [3]. Järgmises peatükis kirjeldab autor agiilse manifesti väärtusi ja printsiipe.

1.2.1 Agiilne manifest

Vajadus luua manifest sai alguse üheksakümnendatel, kui kasvas personaalarvutite levik ning sellega koos ettevõtete soov luua uusi rakendusi. Peamiseks probleemiks osutus aga

projektide suurus ja mahukad nõudmised, mille elluviimine võttis tohutu aja. Tuli luua uusi viise tarkvaraarenduseks [8]. Võeti kasutusele objektorienteeritud programmeerimine, mille eesmärk oli kiirendada tarkvaraarendusprojektide läbiviimise aega. Erinevate ettevõtete meeskonnad hakkasid katsetama tööde ja nõuete tükeldamist, et jõuda kiiremini koodi kirjutamiseni. Aega ei kulutatud enam nii palju planeerimiseks ja dokumenteerimiseks, mille tulemusena oli töövoog ja tulemus saavutamise kordades kiirem [7].

2001. aastal kogunes grupp maailma tarkvaraarendajaid Park City linna Utahi osariigis USA-s, et arutada seni katsetatud meetodite üle ning koondada nende sarnasused ja erinevused. Ühiseks nimetuseks sai *agile* ning sündis „Agiilne Manifest“ [3].

Agiilse manifest neli peamist põhimõtet on:

1. Inimeste omavahelist suhtlust ja koostööd on rohkem kui protsesse ja vahendeid;
2. Töötavat tarkvara on rohkem, kui põhjalikku projekti dokumentatsiooni;
3. Koostööd kliendiga on rohkem, kui läbirääkimisi lepingute üle;
4. Jooksvat reageerimist muudatustele on rohkem, kui algse plaani järjepidevat järgimist.

[7]

Nendele neljale põhimõttele toetuvad manifesti 12 printsiipi, mis on omased agiilsele projektijuhtimisele:

1. Kõige olulisem on tagada kliendi rahulolu tarnides talle vajalik tarkvara võimalikult varakult;
2. Muutuvaid olusid tuleb arvestada ka arenduse lõppjärgus. Agiilsed meetodid suudavad muuta sellised muutused kliendi konkurentsieelisteks;
3. Tarkvara tarnida nii tihti kui võimalik. Soovituslikult iga paari nädala kuni paari kuu tagant. Eelistada tuleks lühemat ajavahemikku;
4. Valdakonna spetsialistid ja tarkvaraarendajad peavad töötama igapäevaselt koos kogu projekti vältel;
5. Projekti edukuse aluseks on motiveeritud inimesed. Tuleb luua neile meeldiv ja toetav töökeskkond ning meeskonnaliikmeid tuleb usaldada, et nad saavad oma tööga hakkama;
6. Kõige tõhusam ja tulemuslikum viis info jagamiseks on meeskonnaga näost näkku vestlus;
7. Edu peamiseks mõõtjaks on töötav tarkvara;
8. Agiilse tarkvaraarenduse protsessid soodustavad jätkusuutlikku arendust. See tähendab, et nii sponsorid, arendajad ja kliendid saavad projektiga püsivas tempos jätkata määramata aja jooksul;

9. Hoides tehnilist täiuslikkust ja head disaini pideva tähelepanu all tagatakse tarkvaraarenduse kiirus ja paindlikkus;
10. Lihtsus – ebavajaliku töö tegematajätmise kunst on väga oluline;
11. Parimad arhitektuurilised lahendused, nõuded ja disain tekivad iseorganiseeruvates meeskondades;
12. Meeskond otsib regulaarselt võimalusi saamaks veelgi tõhusamaks ja muudab end vastavalt vajadusele.

[3]

Põhimõtted ja printsiibid on eelkõige soovitusel ja väärtused agiilsete meetodite rakendamisel, mida saab lähtudes organisatsiooni vajadustest ümber kohandada [7]. Järgmises peatükis kirjeldab töö autor kolme üldlevinud agiilset meetodit.

1.3 Agiilsed projektijuhtimise meetodid

Agiilseid meetodeid on kirjeldatud mitmeid kuid antud lõputöös käsitleb autor kolme enim levinud meetodit - ekstreemprogrammeerimine (XP), Kanban ja Scrum. Iga meetodi puhul vaadeldakse peamisi väärtusi ja praktikaid.

1.3.1 Ekstreemprogrammeerimine

Ekstreemprogrammeerimine (XP) võeti kasutusele vajadusest muuta pikaajalise arenduse tsüklid kiiremaks. Sõna “ekstreem” meetodis tuleb omadusest, et agiilse manifesti põhimõtteid ja printsiipe võetakse ekstreemselt tõsiselt [9]. See põhineb väärtustel nagu lihtsus, kommunikatsioon, tagasiside, julgus ja austus [10].

Kommunikatsioon - iga meeskonna liige on osa tervikust meeskonnast, kelle eesmärgiks on leida parim lahendus probleemile. Meeskonnasisene suhtlus toimub näost-näku igapäevaselt ning koostöö toimub kogu protsessi vältel nõuetest koodi kirjutamiseni [11].

Lihtsus – meeskond teeb nii palju kui on küsitud, aga mitte rohkem, et vältida ressursside ja aja raiskamist. Tuleb tegeleda ainult nende nõuetega, mille kohta on informatsioon ja vajadus. See maksimeerib investeeringu väärtuse. Areng toimub väikeste sammudena, mis soodustab tekkinud probleeme esilekerkimist [11].

Tagasiside – pidev tagasiside aitab meeskonnal tuvastada kitsaskohti projekti juures ning aitab luua paremaid tooteid ja teenuseid [12]. Toodet demostreeritakse varakult ja tihti, et teha koheselt vajalikud muudatused [11].

Julgus – julgus tunnistada protsessi reaalselt kulgu ja tuleviku hinnangulist arengut. Meeskonnaliikmed ei karda põruda, sest kõik töötavad üksteist toetavalt ühise eesmärgi nimel [11]. Julgus on vajalik, et lõpetada tegevused, mis ei toimi ning proovida midagi uut. See aitab aktsepteerida negatiivset tagasisidet, isegi kui seda on raske teha [12].

Austus - meeskond peab üksteist austama, et toimuks avatud ja efektiivne suhtlus. Tagasiside andmine on vajalik, et luua väärtuslikke tooteid ja teenuseid [12].

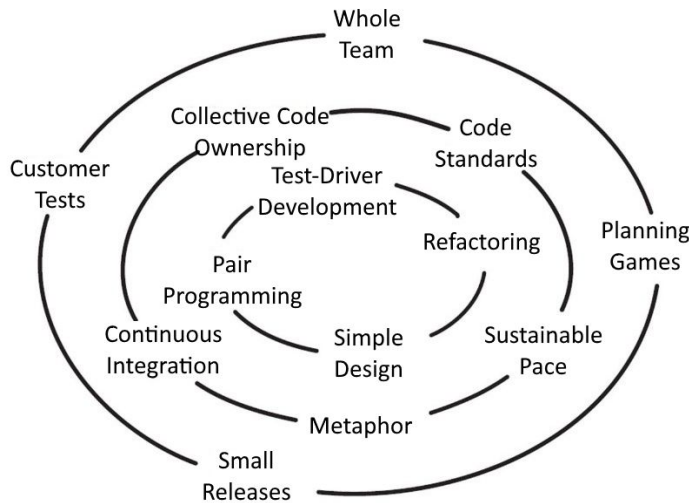
Järgnevalt on lühidalt kirjeldatud ekstreemprogrammeerimise peamisi praktikaid, mis on välja toodud joonisel 1:

Terve meeskonna (*Whole Team*) puhul on igaüks meeskonna liige ja panustab arendusprojekti valmimisse. Meeskond teeb kliendiga tihedalt koostööd ja klient on vajadusel meeskonnale distantsilt kergesti kättesaadav. Kliendi ja meeskonna kohtumisel pannakse paika kliendi nõuded, mida meeskond peab täitma. Planeerimise mängu (*Planning Game*) puhul kasutavad ekstreemprogrammeerimise meeskonnad lihtsasti mõistetavat planeerimise ja töö jälgimise vormi. Selle eesmärgiks on planeerida kliendi nõudmiste täitmise plaan ehk omada ülevaadet olukorrast, mida tuleb teha järgmiseks ning kui kaua mingi töö aega võib võtta. Toodet või teenust arendatakse lühikeste arendus seeriatega kaupade (*Small Releases*), mida testitakse kliendi poolt tihedalt ette määratud kriteeriumite järgi (*Customer Tests*).

Ekstreemprogrammeerimise meeskond jagab ühist visiooni sellest, milline arendatav tarkvara välja hakkab nägema (*Metaphor*). Toimub pidev integratsioon kliendi ja meeskonna vahel (*Continuous Integration*). Meeskond kirjutab koodi ühiste reeglite ja kodeerimis standardite järgi (*Coding Standards*) ning koodi hallatakse ühiselt (*Collective Code Ownership*). Töö käib ühtlases ja normaalses tempos (*Sustainable Pace*), mida on võimalik säilitada ehk XP ei poolda ületöötamist ja ületunde.

Ekstreemprogrammeerijad programmeerivad paarides või gruppides (*Pair Programming*), mis hoiab süsteemi pidevalt töökorras ning teeb selle loetavaks ja lihtsasti mõistetavaks. Luuakse koos lihtne disain (*Simple Design*), mille töökindluste eest vastutatakse ühiselt. Koodi testitakse pidevalt ehk arendus on testimisele toetuv (*Test-Driven Development*). Nii

hoitakse toode või teenust täpselt hetke nõuete kohaselt, pidevalt selle disaini (*Design Improvement*) ja koodi täiendades (*Refactoring*) [7].



Joonis 1. Ekstreemprogrammeerimise praktikad [7]

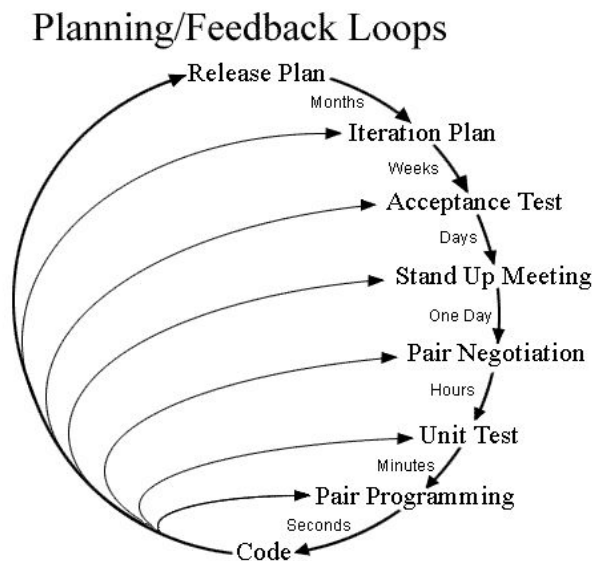
Ekstreemprogrammeerimise praktikaid rakendades jälgitakse reegleid, mis jagunevad viieks:

1. **Planeerimise reeglid** - meeskond kirjeldab projekti alguses kasutajalood (*User story*), mis on koostatud kliendi nõudmiste põhjal. Kasutajalood kirjutatakse lahti iteratsioonideks ehk väiksemateks ülesanneteks. Meeskond alustab iteratsioonidega töötamist peale tarne ajakava koostamist. Toode/teenust tuleb väljastada kliendile arenduse käigus tihedalt, et saada tagasiside vajalikeks parandusteks [13].
2. **Juhtimise reeglid** - meeskonnale tuleb tagada avatud töökeskkond, kus on võimalik maksimaalselt tööle pühenduda. Määratakse ära jätkusuutlik töötempo. Igat päeva alustab püstijala koosolek, kus arutatakse juba tehtut, mis on veel tegemisel ning mis on seni esile kerkinud probleemid. Projekti käigus tuleb mõõta projekti kiirust, et anda reaalne hinnang kasutajalugude realiseerimisvõimekusele. Meeskonnasiseselt liigutatakse arendajaid ringi, et iga programmeerija oleks võimalikult palju kursis kirjutatava koodiga. Meeskond peab jälgima XP reegleid, et projekti eesmärk saaks saavutatud [13].
3. **Disain** - arendatava projekti disain tuleks hoida võimalikult lihtsana. Kui disainis on midagi keerulist, tuleks see võimaluse korral asendada lihtsama lahendusega. Koodi lihtsuses otsustab meeskond ühiselt koodi hinnates. Lisafunktsionaalsused lisada nii hilja kui võimalik, olles kindel, et seda on kindlasti programmis vaja.

Refaktoreerimine toimub pidevalt läbi projekti elutsükli, tõstes teenuse/toote kvaliteeti ja säästes aega [13].

4. **Kodeerimine** - üheks peamiseks nõudeks ekstreemprogrammeerimise juures on kliendi kättesaadavus. Kõigis arendusetappides on vaja hoida kliendiga sidet. Kodeerimine toimub kokkulepitud standardi järgi. Tellitud tootele programmeeritakse testid enne loodavat lahendust, mis on justkui nõuete kirjapanemine. Kogu toodangu/teenuse kood on paarisprogrammeeritud. Ainult üks paar modifitseerib loodavat rakendust või rakenduse osa korraga. Igal programmeerijal on kogu koodi juurde võrdne ligipääs [13].
5. **Testimine** - iga koodi osa peab omama testi ning kogu kood peab läbima testid enne toote või teenuse tarnimist. Kui avastatakse viga (*bug*), siis luuakse sellele veale koheselt test. Aktsepteerimistestide tehakse pidevalt ja tulemused avalikustatakse [13].

Kõiki eelmainitud reegleid rakendatakse projekti elutsükli (Joonis 2), kus kuudeks koostatakse tarneplaan, nädalateks iteratsioonide täitmise plaan, päevadeks luuakse aktsepteerimistestid, igapäevaselt viiakse läbi püstijala kohtumisi, iga tunni jooksul suheldaks kaasprogrammeerijatega (paarisprogrammeerimine), minutite jooksul kontrollitakse kirjutatud programmi automaattestidega ja iga sekund kodeeritakse uut koodi [14].



Joonis 2. Ekstreemprogrammeerimise elutsükkel [14]

XP puhul suhtlevad programmeerijad pidevalt oma kliendi ja kaasprogrammeerijatega. Arenduse disain hoitakse lihtsa ja puhtana. Tagasiside saadakse tarkvara testimisega

igapäevaselt. Toode tarnitakse kliendini nii vara kui võimalik, et viia ellu vajalikud muudatused. Iga väike saavutus oleneb meeskonna austaval suhtumisel üksteise panuse vastu. Sellele põhinevalt on XP meetodit rakendavad meeskonnad hästi valmis kohanema nõuete ja tehnoloogia muutustega [11].

1.3.2 Kanban

Kanban sai alguse Jaapanis kus see tähendab tõlkes “visuaalne register” või “visuaalne osa” [15]. 1940ndatel avastas Jaapani autotootjafirma Toyota parema tootmisprotsessi, mis määrab laoseisu vastavalt nõudmisele ja püüdleb kõrgema kvaliteedi poole. 2000. aastate keskel kohandas Microsofti töötaja David Anderson Kanbani tarkavaraarenduses kasutatavaks meetodiks [7]. Kanban rõhutab “täpselt õigeaks ajaks” teenuse või toote tarnimist kliendile. Kanbani põhirõhk on võimalikult detailselt määratleda, mis tööd tuleb ära teha ja millal need tuleb teostada. Selleks tähtsustatakse ülesandeid, määratletakse töövoog ning tarnimiseks kuluv aeg. Kanbani protsess kirjeldab selgesõnaliselt kõige olulisemaid ülesandeid, mis vajavad kõige rohkem tähelepanu, et vähendada nende mittetäitmise riski ja suurendada projekti muude ülesannete hulgas paindlikkust [16].

Kanabanil on neli peamist printsiipi:

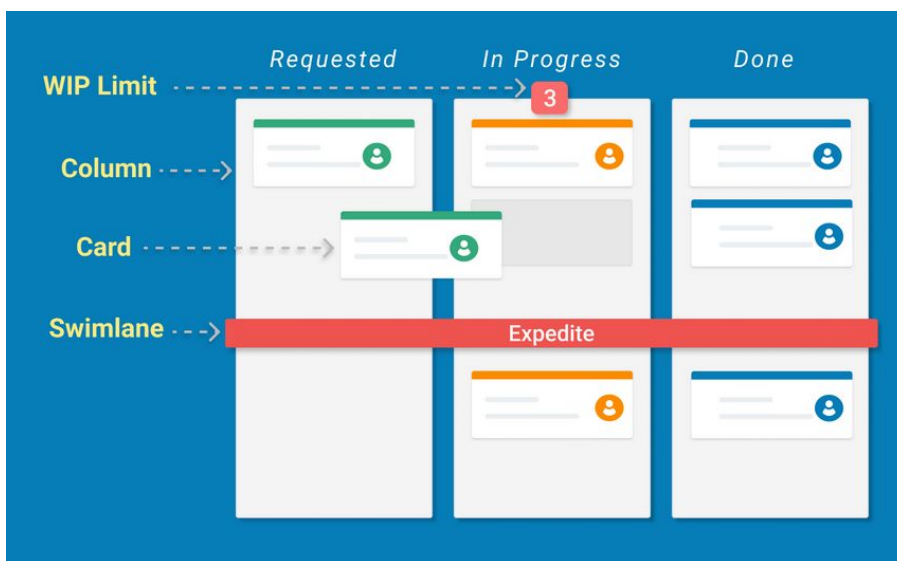
Alusta sealt, kus sa hetkel oled – Kanbani meetodit on lihtsalt võimalik üle kanda olemasolevatele töövoogudele, süsteemidele ja protsessidele, häirimata juba tehtavat. See toob esile probleemid, millega tuleb tegeleda ja aitab muudatusi hinnata ja kavandada, nii et nende rakendamine oleks võimalikult häirimatu. Kanban on hõlpsasti rakendatav igat tüüpi organisatsioonides, kuna algselt pole vaja suuri muudatusi teha [17].

Järgi inkrementaalset ehk järk-järgulist lähenemist – Kanbani meetod on disainitud eesmärgiga tekitada minimaalset vastumeelsust ja sellest tulenevalt tehakse toimivas protsessis muudatusi väikeste tükkide kaupa ehk inkrementaalselt. Kanban ei poolda suuri ja järske muutusi, mis võivad kaasa tuua vastumeelsust ja ebakindlust [17].

Austa olemasolevat protsessi, rolle ja kohustusi – Kanban meetod väärtustab olemasolevaid protsesse, rolle ja vastutusi ning proovib neid säilitada. Muutusi tuleks läbi viia järk-järgult ja loogiliselt, mis ei tekitaks meeskonnas hirmu nende ees [17].

Julgusta juhtimist igal tasandil – parim juhtimine tuleb igapäevase meeskonna kaasamisega. Meeskond peaks olema ühise eesmärgi eest väljas, et tarnida parim võimalik toode [17].

Kanban nõuab täielikku töö läbipaistvust ning suutlikkust suhelda reaalselt. Tööülesanded visualiseeritakse kanbani laual (*Kanban board*), kus need on igal hetkel kättesaadavad meeskonna liikmetele. Töövoog liigub tahvil vasakult paremale. Liigutatakse Kanban kaarte, millel on ülesande informatsioon - lõpptähtaeg, ülesannet täitev isik ja ülesande kirjeldus. Kanban tahvil on kolm tulp (Joonis 3). Esimeseks on soovitud tellimused (*Requested*), siis töös olevad ülesanded (*WIP - Work in Progress*) ning viimaks tehtud ülesanded (*Done*) [18].



Joonis 3. Kanban Tahvel [18]

Kanban meetodi edukaks rakendamiseks on David Anderson defineerinud kuus peamist praktikat:

Visualiseeri töökeskkond - esimene samm on mõista, mis tegevusi tuleb teha, et tarnida kliendile soovitud toode. Selleks tuleb luua eelpool mainitud Kanban tahvel koos ülesandekaartidega. Tööd alustades tõmmatakse ülesanne *WIP* tulpa ja kui ülesanne on sooritatud, siis tõmmatakse see tehtud tööde tulpa [17].

Piira töös olevaid ülesandeid - liiga palju tööülesandeid võib kahjustada tööprotsessi, sest soodustab mitme asjaga korraga tegelemist. See soodustab ebaefektiivsust ja ressursside raiskamist. Kanbani idee on kitsendada töös olevaid ülesandeid nii palju, et iga meeskonnaliige omaks enda võimete piires korraga tehtavaid ülesandeid. Kui töös olevad ülesanded ei ole piiratud, siis ei praktiseerita Kanbani meetodit [17].

Halda töövoogu- Kanbani peamine eesmärk on luua sujuv ja kiire töövoog, mis tähendab tööülesande liikumist alguse sammust kuni lõpetamiseni. Töövoogu juhtimine on ülesannete juhtimine mitte inimeste juhtimine. Tuleb jälgida kuidas saada ülesanded Kanban tahvlil kõige kiiremini ja sujuvalt tellimuse tulbast tehtud tööde tulpa. Nii loob süsteem kliendile pidevalt väärtust ning minimaliseerib viivitustest tulenevaid kulutusi [17].

Määra selge tegutsemiseviis - keeruline on arendada asju, millest ei ole võimalik aru saada. Tähtis on, et töötamise protsess oleks selgelt defineeritud ja üheselt mõistetav, et meeskond võtaks protsessi omaks. Meeskond töötab ühiste väärtuste ja eesmärgi nimel [17].

Pidev tagasiside - Kanbani puhul viiakse läbi regulaarseid “püsti jala” (*Stand up*) kohtumisi Kanban tahvli ees, kus iga meeskonna liige annab ülevaate tehtust ja mida on plaanis teha järgnevalt. See ühendab meeskonda ja hoiab nad üksteise tegemistega kursis. Samuti on olemas tarne ülevaate, toimingute ülevaate ja riskiülevaate koosolekud. Need on regulaarsed, kindla ajaga ja kokkuvõtlikud. Püstijala kohtumiste keskmine pikkus peaks olema 10-15 minutit, kuid see oleneb meeskonna suuruselt, sest tähtis on, et iga meeskonna liige saaks sõna [17].

Arendada koostööd mudelite ja meetodite abil – selleks, et meeskonnasiseselt toimuks pidev areng peab töögrupp jagama ühist visiooni ja eesmärke. Meeskonnad, kellel on ühised arusaamad töövoost, protsessist ja riskiteooriatest, saavad suurema tõenäosusega probleemidest üle ning leiavad konsensusse [17].

Kanbani meetod keskendub sellele, et arendajate oskuste komplekti arvestades oleks õige töö õigel ajal tehtud. Arendajatel võivad olla erinevad oskused ja töökiirus. Arendajad ei arenda tarbetuid funktsioone, ei kirjuta rohkem spetsifikatsioone, kui nad suudavad kodeerida, ei kirjuta rohkem koodi, kui nad suudavad testida, ega testi rohkem koodi, kui nad suudavad juurutada. Kanbani meetod kõrvaldab kõik mittevajalikud tegevused ja sobib hästi tarkvaraarenduse projektide jaoks [16].

1.3.3 Scrum

Scrum on agiilne projektijuhtimise meetod tarkvara arendamiseks. Scrum kasutab iteratsiooni ja inkrementatsiooni [19]. Eelnevalt mainitud mõisteid käsitletakse tarkvaraarenduses järgnevalt: inkrementatsioon tähendab, et uut funktsionaalsust lisatakse tootele või teenusele

väikeste osade kaupa. Iteratsioon tähendab, et uut funktsionaalsust lisatakse tarkvarale kindla ajavahemiku tagant [7]. Scrum on disainitud eesmärgiga hallata kiiresti muutuvaid projektinõudeid, parandades suhtlemist projekti arendajate, projekteerijate ja teiste meeskonnaliikmete vahel. 1986. aastal nimetasid Hirotaka Takeuchi ja Ikujiro Nonaka Scrumi uueks tootearenduse standardiks, mis on iseloomulik auto- ja tarbekaupu tootvatele ettevõtetele. Scrumi on defineeritud kui esimese agiilne meetod. 1993. aastal kasutasid Jeff Sutherland, John Scumniotales ja Easel Company Jeff McKenna esimest korda tarkvara arendusprojektide jaoks Scrumi [20]. 2002. aastal kirjutasid Schwaber ja Beedle raamatu "Agile with Scrum", kirjeldades Scrumi metoodikat [19].

Läbipaistvus, kontroll ja kohanemine on Scrumi kolm olulist alussammast, mida on ülioluline rakendada projekti erinevates arenguetappides.

Läbipaistvus - arendusprotsess, töövoog ja progress peab olema nähtav kõigile, kes on projekti kaasatud.

Kontroll - et tagada Scrumi läbipaistvus, peab pidevalt toimuma ülevaatlik kontroll arendusest ja meeskonnatööst, et tuvastada projekti probleemid varajases faasis.

Kohandamine - tänu tihedale kontrollile tuvastab meeskond kiiresti arenduse seisakud ja saab enda tegevusi või toote/teenuse disaini vastavalt kohandada [19].

Läbipaistvus, kontroll ning kohanemine nõuab tiimi avatust ja usaldust. Selleks on kirjeldatud Scrumi viis väärtust:

1. **Pühendumine** - iga meeskonnaliige pühendab ennast individuaalselt täielikult projekti eesmärgi saavutamisele.
2. **Julgus** - meeskonnal on julgus ületada ühiselt konfliktid ja projekti käigus esinevad väljakutsed.
3. **Fookus** - meeskonnaliikmed pühenduvad igas sprindis eranditult projekti eesmärgile ning töötavad ainult ettenähtud ülesannete kallal.
4. **Avatus** - meeskond ja klient lubavad olla läbipaistvad nii tehtud tööd kui ka tekkivate probleemide osas.
5. **Respekterimine** - meeskond austab üksteise oskusi ja võimekust.

[21]

Neid väärtusi tuleb järgida iga Scrumi protsessi juures. Scrum koosneb Scrumi meeskonnast, Scrumi tahvlist ja Scrumi sündmustest. Scrumi alguses pannakse kliendi poolt paika toote või

teenuse nõuded, mis on olulised eelmainitud elementide sidumiseks projekti jooksul. Järgmistes lõikudes selgitatakse üksikasjalikult Scrumi meeskonda, tahvlit ja elutsükli:

Scrumi meeskond koosneb tooteomanikust (*Product Owner*), Scrumi meistrist ehk projektijuhist (*Scrum Master*) ja arendusmeeskonnast. Meeskond on iseseisev ehk organiseerib oma tegevusi suuresti ise. Seetõttu on neil kontroll projekti üle ja nad teavad, kuidas eesmärged saavutada ilma meeskonnast väljaspool olevate inimeste juhiste lootmata. Meeskond tarnib tooteid nii iteratsioonalselt kui ka inkrementatsioonalselt, maksimeerides tagasisidet, mida nad kliendilt saavad iga sprindi lõpus. **Sprint** on kindel ajavahemik (tavaliselt 2-4 nädalat), mille lõppedes esitab meeskond ülevaate seni valminud tootest [19].

Allpool kirjeldatakse erinevaid Scrumi meeskonna rolle:

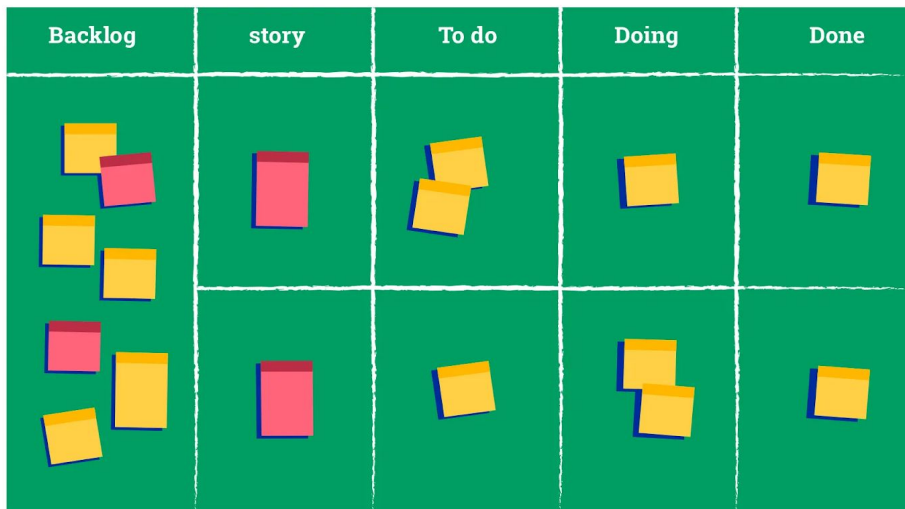
Tooteomanik (*Product Owner*) on vastutav tegemata tööde nimekirja (*Product Backlog*) haldamise ja projekti väärtuse maksimeerimise eest. Tema rollide hulka kuulub ka ülesannete selgitamine ning arenduseesmärgi esitlemine arendusmeeskonnale. Ta tagab, et meeskond mõistaks neid eesmärged ja teostaks arendust nõutud tasemel [19].

Scrumi meister (*Scrum Master*) ehk projektijuht haldab toote tegemata tööde nimekirja ja juhendab arendusmeeskonda loomaks sinna selgeid ülesandeid. Scrumi meister vastutab, et meeskond mõistaks projekti pikaajalisi plaane ja eesmärged. Üldiselt on projektijuht vastutav toote tarnimise eest kliendile [19].

Arendusmeeskond vastutab tarnitava toote esitamise eest iga sprindi lõpus. Tarkvara arenduse raames on meeskonnas näiteks programmeerijad, kasutajaliidese disainerid ja testijad. Tähtis on see, et meeskond oleks 100% pühendunud töös olevale ülesandele, sest mitme ülesandega korraga töötamine piirab meeskonna jõudlust ning segab keskendumist [7]. Arendusmeeskonna liikmed organiseerivad ise oma tööd ja neid ei jaotata alarühmadesse.

Meeskonna suurus on oluline detail scrumi juures - väike meeskond võib kannatada puudulike oskuste käes, samas suur meeskond võib kannatada arenduse liialt keerukaks muutmise tõttu. On leitud, et arendusmeeskonna soovitatav suurus on seitse liiget [19].

Scrumi tahvel on Scrum meetodi haldamise vahendeid. See koosneb kõikide tegemata tööde nimekirjast (*Product backlog*), sprindi jaoks valitud tegemata tööde nimekirjast (*Sprint backlog*, joonisel *Story*), hetkel sprindis töös olevate tööde nimekirjast (*Doing*) ja seni valmis saanud töödest (*Done*) (Joonis 4).



Joonis 4. Scrum tahvel [22]

Tegemata tööde nimekiri sisaldab nõuete ja funktsioonide loendit, toote täiustusi ja parandusi. See näitab toote funktsionaalsust tehnilistest ja ärilistest vaatenurkadest. Tooteomanik vastutab loendi koostamise ja meeskonnale ülesannete jagamise eest [19].

Sprindi tegemata tööde nimekiri on tegemata tööde kuhjast olevate üksuste loend, mis on valitud konkreetse sprindi jaoks. Arendusmeeskond selgitab toote funktsioone, mida arendatakse järgmisel sprindil ja selleks arenduseks vajaminevat tööd. Sprindi tegemata tööde nimekirja prioritseerib samuti tooteomanik [19].

Scrumi elustsükkel algab kliendi kohtumisega, kus pannakse paika soovitud toote või teenuse kliendipoolsed nõuded. Nõuete alusel koostab meeskond ühiselt tegemata tööde nimekirja, mille järjekorra prioritseerib tooteomanik. Peale seda toimub projekti arendus sprintides. Iga sprint kestab kindla aja (tavaliselt 2-4 nädalat). Sprindi ajal toimub neli meeskonnasisest tegevust. Nende eesmärk on kontrollida arengut, rakendada vajadusel uusi meetodeid projekti probleemide lahendamiseks, suurendada läbipaistvust ja tagada järelevalvet arendusprotsessis. Nendeks sündmusteks on sprindi planeerimine (*Sprint planning*), igapäevane Scrum (*Daily Scrum*), sprindi ülevaade (*Sprint review*) ja sprindi tagasivaade (*Sprint retrospective*) (Joonis 5) [19].

Sprindi planeerimine on scrum protsessi süda. Sprindi planeerimise koosolekul otsustatakse, mis on sprindi eesmärk ning mis ülesandeid hakatakse tegema. Sprindi ja arendusmeeskonna arengueesmärke ei tohiks sprindi jooksul muuta. Kuid tooteomanik ja arendusmeeskond võivad projekti ulatust vastavalt vajadusele uuesti määratleda.

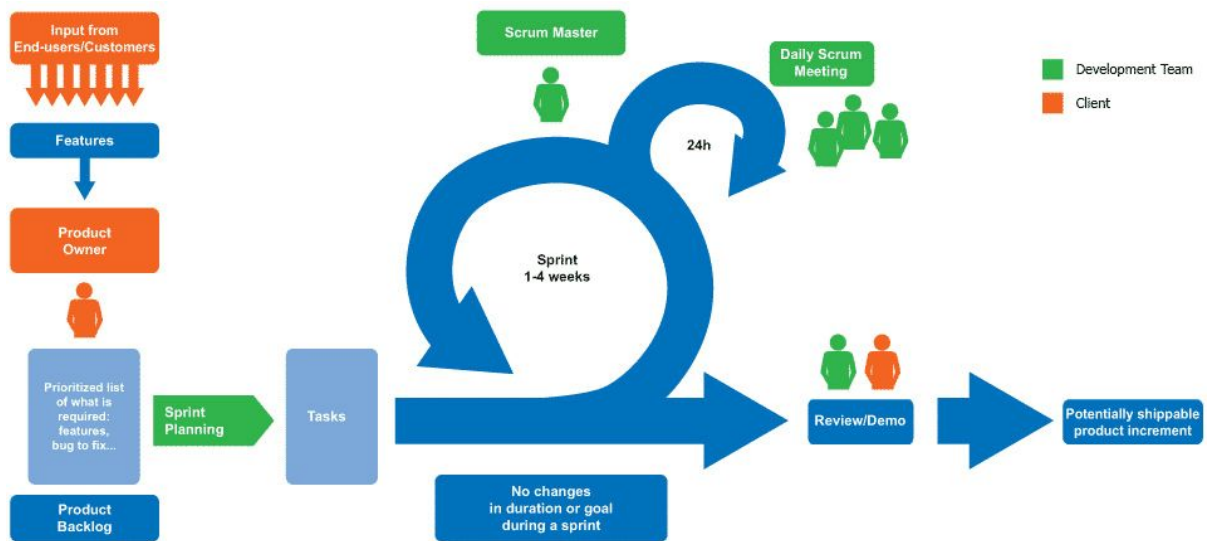
Tooteomanik saab sprinti tühistada juhul, kui ettevõtte tegevussuunas, turuvajaduses või tehnoloogias ilmnevad muudatused. Iga sprinti üldine eesmärk on luua kasutatav ja potentsiaalselt taaskasutatav toode, mida tuntakse toote „Valmis” nime all. Sprinti planeerimise koosolek toimub üks kord enne igat sprinti ning kestab tavaliselt mitu tundi [19].

Igapäevane Scrum toimub iga päev umbes 15-minutit, kus kogu meeskond annab kokkuvõtliku ülevaate üksteise edusammudest, oma järgmistest eesmärkidest ja seni tekkinud raskustest.

Sprinti ülevaade toimub iga sprinti lõpus, kus antakse sprinti ülevaade. Selle eesmärk on arutada, mida iga meeskonnaliige toote arendamise iteratsiooni ajal tegi. See kohtumine on tavaliselt tooteomaniku või kogu meeskonna tootesitlus kliendile.

Sprinti tagasivaade toimub peale sprinti ülevaadet ja enne järgmist Sprinti. Analüüsitakse kuidas viimasel Sprintil läks suhtluse, inimressursside, protsesside ja seni rakendatud töömeetodite osas. Samuti tehakse kindlaks tulevaste Sprintide võimalikud parandused. See kohtumine võtab tavaliselt mitu tundi [19].

Peale viimast sprinti tarnitakse lõpptoode kliendile. Tavaliselt teeb meeskond selleks kliendile lõppettekande, kus annab ülevaate arendusprotsessi tulemuslikkusest ja tarnib toote kliendile. Scrumi elutsükel on kujutatud joonisel 5.



Joonis 5. Scrumi elutsükel [23]

Scrum on tuntud selle poolest, et suudab kohaneda sagedaste muutustega tarkvaraarendusprojektides (nii tehnoloogia kui ka nõuete seisukohast) ning võimaldab

projektides kiiresti tulemusi saada. Scrumi peamine eripära on paindlikkus ja samal ajal pakub see projekti toimivuse kontrollimise ja parandamise mehhanisme. Samuti aitab Scrumi meetod praeguses teadmispõhises ühiskonnas hoida tähtsat informatsiooni organisatsioonide sees. Meetod sobib väga hästi väikeste arendusmeeskondadega projektideks (alla 10 arendaja, kuid soovitatav on vähemalt 4-5 arendajat). Eelistatav on meetodit kasutada projektide korral sisemise kliendi või väga kättesaadava väliskliendiga - kliendi kättesaadavus mõjutab otseselt projekti edukust [24]. Neid tunnuseid ja omadusi arvesse võttes kasutati lõputöös analüüsitavas projektis "Tulevikukodu" Scrum meetodit. Järgnevates peatükkides antakse ülevaade projektist ja selle arendusest.

2. Projekt “Household of the future virtual reality”

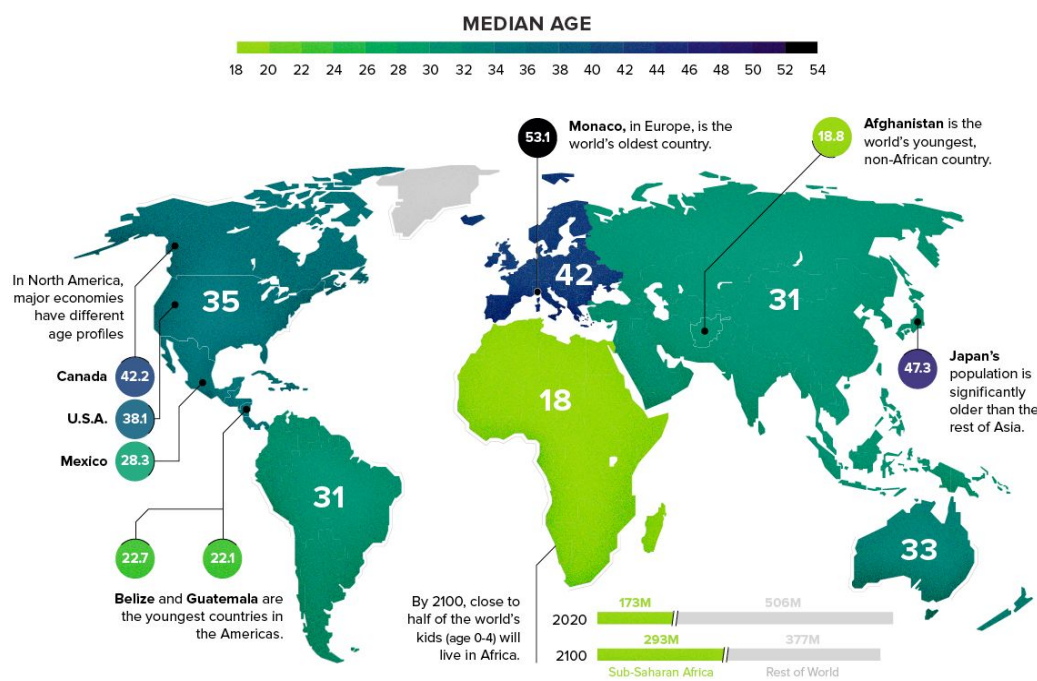
Selles peatükis annab autor ülevaate projekti probleemist, selle eesmärgist ja Scrum meetodil põhineval lahenduse planeerimise etapist. Probleemi püstitus on Hollandi näitel, sest projekt viidi läbi Hollandis, Hanze Rakenduskõrgkoolis (*Hanze University of Applied Sciences*) “Tark Energia” (*Smart Energy*) programmi raames. Programmi eesmärgiks oli ühendada energiasektor ja arvutiteadus läbi üliõpilaste koostöö erinevate Hollandi infotehnoloogia ja energiasektori ettevõtete ja organisatsioonide. Programmis moodustati tudengitest meeskonnad, kes kogunud õppejõu juhendamisel ja kliendi nõudmistel leidsid lahendust hetkel antud valdkonnas esinevale probleemile. Peatükis kirjeldatakse probleemi konteksti, olulisust ja projekti tellinud kliente. Viimase osana antakse ülevaade sellest, kuidas projekti “Tulevikukodu” meeskond plaanib pakkuda püstitatud probleemile lahendust võttes arvesse agiilsel meetodikal põhinevat Scrum meetodit.

2.1 Probleemipüstitus

Probleem: Rahvastiku vananemine, kui üks globaalse demograafia suurtrendidest, mõjutab lähitulevikus suuresti ühiskonna ülesehitust [25].

Maailma Terviseorganisatsiooni (*World Health Organisation - WHO*) andmetel on 2020. aasta seisuga 60-aastaseid ja vanemaid inimesi maailmas rohkem kui 5-aastaseid ja nooremaid lapsi. Lisaks ennustab WHO, et 60-aastased ja vanemad inimesed moodustavad 2050. aastaks umbes 22% kogu maailma elanikkonnast. 2015. aastal oli eakaid 900 miljonit kogu maailma rahvastikust, aga 2050 aastaks tõuseb see arv ennustuste kohaselt 2 miljardini ehk pea kahekordistub [25].

Rahvastiku vananemine on suur trend ka Euroopa mandril, mis on hiljuti määratletud maailma vanimaks mandriks, mille keskmine vanus (42) on suurim kõigist mandritest (Joonis 6). Uuel reaalsusel on aga kogu mandrile tervikuna oma majanduslikud tagajärjed. Maailma Majandusfoorum (*World Economic Forum - WEF*) on rõhutanud, et Euroopa vananeva elanikkonna kogumaksumus moodustab 2070. aastaks umbes 27% Euroopa Liidu sisemajanduse kogutoodangust [26].



Joonis 6. Maailma mandrite mediaanvanus Allikas: [26]

Rahvastiku vananemise probleem on ka Hollandis, kus ennustatakse, et 2050. aastaks on eakate arv kasvanud 8 miljonini, mis on ligi pool Hollandi elanikkonnast, mis tõstab pensionäride arvu ühiskonnas ja kasvatab tervishoiteenuste nõudlust [4]. Lisaks on eelnimetatud väljakutse täiendav külg aga see, et eakad Hollandi kodanikud eelistavad elada võimalikult kaua isiklikes eluruumides, mitte eakate hooldekodudes. Seda eelistust on toetanud ka poliitika, mille on kehtestanud Hollandi valitsus. Selle tulemusel elab 80% 80-aastastest ja vanematest inimestest oma isiklikes kodudes [27].

Rahvastiku vananemise probleemile tähelepanu pöörates otsivad mitmed Hollandi ettevõtted ja organisatsioonid innovatiivseid lahendusi ehitistes, et need oleksid energiatõhusad ja sisaldaks endas jätkusuutlikku tervisliku vananemise võimalusi. Projekti “Tulevikukodu” eesmärgiks on välja tuua Targa kodu (*Smart House*) eelised ja kasulikkus. Täpsemalt põhineb projekt virtuaalse reaalsuse (*Virtual Reality*) simulatsioonil, kus sihtotstarbelised lõppkasutajad (vanurid vanuses 50–85 aastat ja puudega inimesed) saavad võimaluse kogeda eeliseid, mida tuleviku Tark kodu pakub - näha selle mõju oma elukvaliteedi tõstmiseks ja iseseisva toimetuleku soodustamisel.

Projekti viivad ellu Hanze Rakenduskõrgkooli “Tark Energia” programmi üliõpilased koostöös järgmiste klientidega.

2.2 Kliendid

Projekti viiakse läbi koostöös nelja kliendiga - Groningeni linnavalitsus, Hollandi ehitusettevõtte BAM, Hollandi telekom ettevõtte KPN ja Hanze Rakenduskõrgkooli energia teadusuuringute keskus EnTranCe. Järgnevalt annab autor kokkuvõtliku ülevaate kõigist neljast organisatsioonist:

Groningeni linnavalitsus - teostatav projekt on tellitud Groningeni linna poolt. Groningeni linn on Groningeni maakonnakeskus, mis on kõige põhjapoolsem maakond Hollandis. Linnas elab umbes 230 tuhat elanikku, neist 60 tuhat (26,08%) on üliõpilased. Groningen eesmärgiks on olla Hollandis tuntud kui "energialinn", muutes linna energianeutraalseks 2035-ks aastaks. [28].

BAM - on tulude põhjal suurim ehitusettevõtte Hollandis ja seda juba alates 1869. aastast. Hollandis on BAMil kaks peamist ärivaldkonda: avaliku sektori ehitus ning tsiviilehitus [29]. Ettevõtte 2020. aasta visiooniks on, et organisatsiooni tunnustatakse kui ühte Euroopa juhtivat jätkusuutlikku ja innovaatilist ehitusettevõtet. BAM julgustab oma kliente ja koostööpartnereid leidmaks maailmale jätkusuutlikke, CO2-neutraalseid lahendusi [30].

KPN - KPN on Hollandi suurim telekommunikatsiooniteenuse pakkuja. Firma asutati 1893. aastal, algselt pakkudes posti ja telegraafi teenust. KPN on üks maailma “rohelisemaid” telekommunikatsiooniettevõtteid, mis on olnud kliimaneutraalne ettevõtte alates 2015. aastast. Nad taaskasutavad modemeid ja telefone. Alates 2011. aastast kasutab KPN 100% rohelist energiat [31]. KPN Cegeka Consulting osakond teeb koostööd erinevate ettevõtete ja ülikoolidega, et välja arendada tehisintellektil põhinevaid Targa maja tehnoloogiasid ja sensoreid [32].

EnTranCe - on Hanze Rakenduskõrgkoolis (*Hanze University of Applied Sciences*) tegutsev energia teaduslabor, mis keskendub ühiskonna üleminekule puhtale, taastuvale ja taskukohasele energiale. Teadlased, üliõpilased, ettevõtted, asutused ja sotsiaalsed organisatsioonid teevad koostööd, et välja töötada erinevaid innovatiivseid lahendusi ning rakendada neid praktikas [33].

Kõigi neljal kliendil on omad nõudmised, ootused ja ettepanekud projekti “Tulevikukodu” simulatsiooni loomisel. Projekti eesmärgi saavutamiseks, tuleb valida sobiv projektijuhtimismeetod ja planeerida projekti etapid.

2.3 Planeerimine

“Tulevikukodu” projekti rahvusvaheline meeskond koosnes neljast liikmest, kus iga liige oli pärit eri riigist (Poola, Eesti, Jamaica, Holland). Meeskonna siseselt oli välearendusmeetoditega kokku puutunud vaid üks töögrupi liige. Valikuks osutus Scrum, sest ühel liikmel oli meetodiga juba eelnev kogemus, meetod sisaldab kindlaid rolle ja on üles ehitatud detailse elutsükliks, mis teeb selle kogemusteta meeskonna jaoks esmakordse rakendamise puhul lihtsasti omandatavaks. Samuti tagab meetod sidusrühmade pideva kaasamise projekteerimis- ja arendusprotsessi. Seetõttu ajendab selline lähenemisviis kaasloomet, paindlikku töötamist ja hõlbustab muudatusi, mis võivad olla vajalikud projekti vältel. Scrumi haldamise platvormiks valiti Trello [34]. Trello on veebipõhine Kanban tahvli stiilis ülesannete haldamise tahvel, mis annab ülevaate projekti tulemuslikkusest. Projekt kestis 5 kuud, selle ajakava selgitatakse järgmises punktis.

2.3.1 Ajakava

“Tulevikukodu” projekt oli osa “Tark Energia” programmist, mis toimus 2020 veebruarist algusest kuni 2020 juuli alguseni Hollandis, Hanze Rakenduskõrgkoolis (*Hanze University of Applied Sciences*). Projekt koosnes 20 nädalast, mille jooksul tuli neljaliikmelisel töögrupil tarnida eelmainitud 4 kliendile töötav simulatsioon “Targast majast”. Järgnevatel lõikudes kirjeldatakse projekti algselt planeeritud ajakava.

Esimesele kolmele nädalale planeeriti projekti sissejuhatus ja ettevalmistus. Tuleb moodustada meeskond, jagada Scrumi rollid ja korraldada esimene meeskonnakohtumine. Töögrupp valmistab ette ja viib läbi intervjuu kliendiga, et kaardistada ära nõuded ja need lahti kirjutada Trellos tööülesannete nimekirjana. Ülikool annab töögrupile üle virtuaalreaalsuse labori, kus meeskond hakkab 5 kuu jooksul arendust läbi viima.

Neljandal nädalal algab esimene 3-nädalane sprint ning sellega kaasnevad punktis 1.3.4 välja toodud sprindi tegevused. Samuti alustab meeskond dokumentatsiooniga ja alustab tegevuskava (*Plan of Attack*) väljatöötamisega kliendile, mida esitletakse sprindi lõpus koos

esimese toote demonstratsiooniga. Selle sprindi käigus valib meeskond projekti jaoks kõige sobivama uurimisstrateegia ja andmekogumismeetodi.

Seitsmendal nädalal algab teine 3-nädalane sprint. Saab arvestada kliendi poolt antud tagasiside. Viiakse läbi esimene grupisisene tagasiside andmine (*Peer assessment*), kus meeskonnaliikmed hindavad üksteise panust seni projekti jooksul. Sprindi alguses alustatakse tegevusaruande (*Progress report*) kirjutamisega, mis esitatakse sprindi lõpus kliendile koos toote teise demonstratsiooniga.

Kümnendal nädalal algab kolmas 4-nädalane sprint, kus rakendatakse kliendi poolt saadud tagasisidet. Sprindi käigus koostatakse uurimuslik küsimustik sihtgrupile testimise jaoks Groningen hooldekodudes.

Neljateistkümnendal nädalal algab neljas 3-nädalane sprint, kus toimub aktiivne stsenaariumite arendus. Selle sprindi lõpus testitakse toodet ka sihtgrupi peal Groningeni hooldekodudes.

Seitsmeteistkümnendal nädalal algab viies, projekti viimane, 3-nädalane sprint. Viimases sprindis rakendatakse nii kliendi kui ka sihtrühma poolt antud tagasisidet seni valminud tootele ning viiakse ellu viimased vajalikud parandused. Kirjutatakse lõpparuanne. Samuti toimub teine meeskonnasisene tagasiside andmine (*Peer assessment*). Toode tarnitakse 19ndal nädalal kliendile ning lõpparuanne esitletakse 20ndal nädalal.

2.3.2 Rollid ja vastutus

Agiilse Scrum meetodi rakendamiseks ja ajakavas püsimiseks on vaja ära jaotada meeskonna rollid. Järgnevas tabelis on kirjeldatud neljaliikmelise meeskonna rollijaotust “Tulevikukodu” projekti juures.

Meeskonnaliige	Roll/funktsioon	Vastutus
Lech Bialek (õppejõud)	Juhendaja	Annab juhiseid ja tuge toote tarnimiseks. Ootamatute olukordade korral tagab võimalused projekti õnnestumiseks. Meeskonna mentor.
Hanna-Marii	Scrumi meister	Kontrollib lõpptoote väljatöötamise juures kogu

Kaljas (autor)	ehk projektijuht	protsessi. Motiveerib meeskonnaliikmeid, optimeerib projekti toiminguid ja vastutab lõpptarne eest.
Dante Baker (kaastudeng)	Tooteomanik	Hoiab klienti pidevalt kursis tootearendusega. On kliendi ja meeskonna vahendajaks. Määrab ära sprindi ülesannete nimekirja prioriteetsed ülesanded.
Piotrek Pawlowski (kaastudeng)	Arendusmeeskond	Pakub projekti lõpuleviimiseks vajalikku tehnilist tuge (tootearendus).
Willem Pepping (kaastudeng)	Arendusmeeskond	Pakub projekti lõpuleviimiseks vajalikku tehnilist tuge (tootearendus).

Peale rollide jaotamist ja selgeks tegemist, tuli meeskonnal end kurssi viia arendamiseks vajalike rakenduste ja vahenditega, mille abil tarniti nõutud projekti klientidele.

2.3.3 Platvorm, programmeerimiskeel ja riistvara

Nõutud lahenduse väljatöötamiseks valis meeskond välja vajaliku arendusplatvormi, programmeerimiskeele ja kasutatava riistvara. Vastavateks valikuteks osutusid Unity, C# ja Oculus Go. Järgnevatel lõikudes põhjendab autor tehtud valikuid.

Unity - valik tehti kahe mänguarendusplatvormi vahel: Unity ja Unreal Engine. Unreal Engine'i kasutatakse eelkõige väga kõrge kvaliteediga visuaalidega mängude tootmise juures ehk programmi jooksumine nõuab suure võimsusega arvuteid. Projekti meeskonnale tagati küll ülikooli poolt mänguarenduseks mõeldud lauarvuti, aga arendusmeeskonnal oli vaja simulatsiooni väljatöötamiseks vähemalt kahte sellist arvutit. Unity lubab jookutada keerulisi ja mahukaid projekte ka keskmise võimsusega arvutites, mis sai selle projekti juures üheks määravaks teguriks - kõik meeskonna liikmed said rakendust oma isiklikes arvutites kasutada. Lisaks on Unreal Engine projektide arendamiseks vaja mitme arendajaga

meeskonda, sest programm on keeruline ja tihtipeale raskesti hallatav paari arendaja poolt. Alustajatele on Unityga tutvumine aga piisavalt lihtne ja see pakub palju funktsioone, mis võimaldavad arendajatel rakendust hõlpsasti muuta ja hooldada. Antud projekti juures polnud ükski arendaja varasemalt ühegi mänguarendusplatvormiga kokku puutunud ehk valikuks osutus kergemini omandatav Unity. Samuti saab platvormi kasutusele võtta paljudes keskkondades: lauaarvutites, mobiilides, veebis ja konsoolides. See on suureks eeliseks olukorras, mil simulatsiooni tahetakse testida hooldekodudes sihtgrupi peal, kus simulatsioon peab olema lihtsasti kaasavõetav, kergesti ülespandav ja kiiresti jooksutatav.

C# - algses programmeerimiskeelte valikus olid C# ja JavaScript. Kuna arendusmeeskonnas oskasid mõlemad arendajad eelkõige C#, osutus valituks just vastav keel. Samuti JavaScript ei toeta Unity keelt sajabrotsendiliselt vaid sellele analoogset UnityScripti, mida arendusmeeskond oleks pidanud esmakordselt õppima. Unity toetab C# aga täielikult. Eelnevatele põhjustele tuginedes langetas meeskond valiku C# programmeerimiskeele kasuks.

Oculus Go - Oculus on alates turule tulekust 2016. aastal maailma üks juhtivaid virtuaalreaalsuse riistvara tootjaid. Ettevõtte toodab hetkel kolme erinevat mudelit - Oculus Rift S, Oculus Go ja uusim Oculus Quest [35]. Meeskonna valikuks osutus teise põlvkonna Oculus Go, sest projekti võõrustav Hanze Rakenduskõrgkool omas just Oculus Go komplekti (Joonis 7). Komplekti eelisteks on, et see on juhtmevaba ja iseseisev ehk sellel on oma ekraan, protsessor, mälu, salvestusruum ja aku. Riistvara iseseisvus teeb selle lihtsasti transporditavaks ja testitavaks sihtrühma peal.



Joonis 7. Oculus Go komplekt [36]

Projekti jooksul lubas ülikool hankida uusima viimase põlvkonna Oculus Quest riistvara (Joonis 8). Oculus Quest on parema jõudluse, resolutsiooni (**1600 x 1440px & 1600 x 1440px** vs 2560 x 1440px) ja kiirema ekraani värskendamise sagedusega (**72Hz** vs 60Hz) [37], mis tagaks simulatsioonis kasutajale realistlikuma kogemuse. Samuti võib kahe juhtimispuldi olemasolu teha kasutajale programmis orienteerumise hõlpsamaks.



Joonis 8. Oculus Quest komplekt [38]

Peamiseks koodijagamise rakendusena planeeris meeskond kasutada GitHubi, mis on üks enim levinud platvorm tarkvaraversioonide hoidmiseks ja meeskonnaliikmetega jagamiseks. Rakendus aitab arendusprotsessi ajal simulatsiooni säilitada, hallata ja värskendada [39].

Peale projekti “Tulevikukodu” planeerimise etappi tuli meeskonnal alustada nõutud stsenaariumite arendusega, mida kirjeldab järgmine peatükk.

3. Projekti stsenaariumite arendus

Selles peatükis antakse põhjalik ülevaade kliendi poolt nõutud stsenaariumite arendusest. Kirjeldatakse uurimismeetodeid, koostatakse kliendi poolt esitatud funktsionaalsed- ja mittefunktsionaalsed nõuded, antakse ülevaade uurimistulemustest ja nende põhjal tehtud stsenaariumite disainist.

3.1 Uurimismeetodid

Kliendile nõuetekohaste stsenaariumite tarnimiseks kasutas meeskond kvalitatiivne uuringu meetodit, mida olemuselt kirjeldatakse mittemõõdetavate omaduste abil [40].

Kvalitatiivse uuringuna viidi läbi intervjuu klientidega, et koostada projekti “Tulevikukodu” funktsionaalsed- ja mittefunktsionaalsed nõuded. Peale nõuete kaardistamist tehti kirjanduslik uuring nende nõuete teostamiseks ja rakendamiseks stsenaariumite disainis. Meeskonna ja kliendi eesmärgiks oli valminud simulatsiooni testida ka sihtgrupi peal Groningeni hooldekodudes, et saada kogemuslikku tagasisidet stsenaariumite täiendamiseks või edasiarendusteks tuleviku.

3.2 Funktsionaalsed ja mittefunktsionaalsed nõuded

Klientide ootuste ja nõudmiste kaardistamiseks viidi läbi intervjuu (Lisa 1). Intervjuu vastuste põhjal kaardistati klientide soovid kahte kategooriasse: funktsionaalsed ja mittefunktsionaalsed nõuded. Funktsionaalsed nõuded määratlevad ära, mida projekti käigus tarnitud toode peab või ei pea tegema ehk kirjeldab ära süsteemi põhikäitumise. Mittefunktsionaalsed nõuded kirjeldavad ära kuidas süsteem seda tegema peaks, aga need nõuded ei mõjuta süsteemi põhifunktsioone. Isegi kui mittefunktsionaalseid nõudeid ei täideta, täidab süsteem ikkagi oma põhieesmärgi [41]. Nõuete prioritseerimiseks on kasutatud MoSCoW-analüüs tehnikat, kus nõuded sildistatakse “Peab olema” (*Must have*), “Peaks olema” (*Should have*), “Võiks olla” (*Could have*), “Seekord pole selleks aega” (*Won't have this time*) [42].

Groningeni linnavalitsuse nõudmistel tuli toode turustada ja luua spetsiaalselt sihtkliendile - üle 50-aastastele inimestele või puudega inimesele, kes näeksid tootes “Targa Maja” eeliseid seoses jätkusuutliku ja tervisliku vananemisega. BAMi ja EnTranCe peamiseks nõudmisteks

oli, et simulatsioon sisaldaks energia säästmise aspekti. Targa Maja tehnoloogia ja süsteemide eeliseid soovis välja tuua KPN.

Järgnevalt on välja toodud intervjuude põhjal koostatud 8 funktsionaalset ja 6 mittefunktsionaalset nõudmist.

3.3.1 "Tulevikukodu" projekti funktsionaalsed nõuded

Autor esitab projekti funktsionaalsed ja mittefunktsionaalsed nõuded kaartidena, kus on välja toodud nõude ID, kirjeldus, nõude esitanud klient/kliendid ja MoSCow analüüsi põhjal tehtud nõude prioritseering.

ID	Funktsionaalne nõue 1
Kirjeldus	Iga simulatsioonis mängitav stsenaarium peab sisaldama energia säästmise aspekti
Klient	EnTranCe, BAM
Prioritseering	Peab olema

ID	Funktsionaalne nõue 2
Kirjeldus	Kasutaja peab saama simulatsioonis mängida infrapunakütet sisaldavat stsenaariumi
Klient	EnTranCe, BAM, KPN
Prioritseering	Peab olema

ID	Funktsionaalne nõue 3
Kirjeldus	Kasutaja peab saama simulatsioonis mängida stsenaariumi, mille eesmärgiks on öine tualeti külastus
Klient	Groningeni linnavalitsus
Prioritseering	Peab olema

ID	Funktsionaalne nõue 4
Kirjeldus	Iga simulatsioonis mängitav stsenaarium peab sisaldama mõnda vanuritel esinevat terviseprobleemi
Klient	Groningeni linnavalitsus
Prioritiseering	Peab olema

ID	Funktsionaalne nõue 5
Kirjeldus	Simulatsiooni peab olema võimalik mängida karakteriga, kellel pole ühtegi terviseprobleemi
Klient	Groningeni linnavalitsus, EnTranCe, BAM, KPN
Prioritiseering	Peab olema

ID	Funktsionaalne nõue 6
Kirjeldus	Simulatsiooni peaks olema võimalik mängida nägemisvaegusega karakteriga
Klient	Groningeni linnavalitsus
Prioritiseering	Peaks olema

ID	Funktsionaalne nõue 7
Kirjeldus	Simulatsiooni peaks olema võimalik mängida liikumisvaegusega karakteriga
Klient	Groningeni linnavalitsus
Prioritiseering	Peaks olema

ID	Funktsionaalne nõue 8
Kirjeldus	Simulatsiooni läbi mängides peaks kasutaja mõistma Targa maja eeliseid
Klient	EnTranCe, BAM, KPN, Groningeni linnavalitsus
Prioritiseering	Peaks olema

3.3.2 Mittefunktsionaalsed nõuded

ID	Mittefunktsionaalne nõue 1
Kirjeldus	Simulatsiooni võiks olla võimalik projekteerida suurele ekraanile, et seda saaks demonstreerida suurele publikule
Klient	EnTranCe, BAM, KPN, Groningeni linnavalitsus
Prioritiseering	Võiks olla

ID	Mittefunktsionaalne nõue 2
Kirjeldus	Kasutajal peab olema lihtne ja arusaadav orienteeruda kasutajaliideses
Klient	EnTranCe, BAM, KPN, Groningeni linnavalitsus
Prioritiseering	Peab olema

ID	Mittefunktsionaalne nõue 3
Kirjeldus	Arendajatel peaks olema lihtne luua uus stsenaarium (ava/sule printsiip)
Klient	EnTranCe, Groningeni linnavalitsus, BAM
Prioritiseering	Peaks olema

ID	Mittefunktsionaalne nõue 4
Kirjeldus	Simulatsiooni kood peab olema lihtsalt ja üle võetav uute arendajate poolt
Klient	EnTranCe, Groningeni linnavalitsus, BAM
Prioritiseering	Peab olema

ID	Mittefunktsionaalne nõue 5
Kirjeldus	Simulatsiooni võiks olla mängitav Oculus Quest riistvaraga
Klient	EnTranCe, BAM, KPN, Groningeni linnavalitsus
Prioritiseering	Võiks olla

ID	Mittefunktsionaalne nõue 6
Kirjeldus	Simulatsioon peaks madalate näitajatega süsteemis säilitama umbes 90 kaadrit sekundis oleku, et vältida merehaiguse teket ja tagada võimalikult realistlik kogemus
Klient	EnTranCe, BAM, KPN, Groningeni linnavalitsus
Prioritiseering	Peaks olema

Eelmainitud funktsionaalsete ja mittefunktsionaalsete nõuete täitmiseks ja implementeerimiseks simulatsioonis, tuli projekti meeskonnal koguda eelnevat informatsiooni kasutatud allikatest, mida kirjeldatakse järgmises punktis.

3.3 Simulatsiooni kvalitatiivne uuring

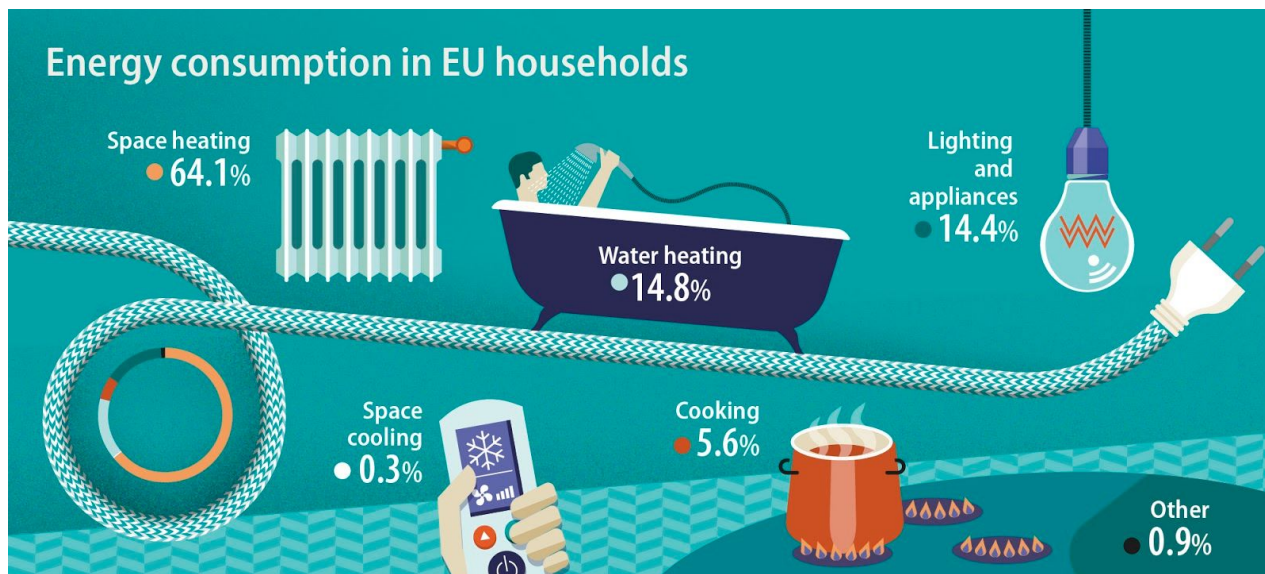
Peale nõuete kaardistamist tehti kirjanduslik uuring nõuete teostamiseks ja rakendamiseks stsenaariumite disainis. Kirjanduslik uuring jaotati kahte kategooriasse: tervisliku vananemisega seotud väljakutsed ja energiatõhusus. Neid kahte kategooriat võeti peamiselt arvesse sobilike rakendatavate koduautomaatika seadmete valimisel ja stsenaariumite disainis.

Tervislikku vananemise mõistet defineerib Maailma Terviseorganisatsioon (*WHO*) kui „protsess, mis arendab ja võimaldab heaolu säilitamist vanemas eas“. Selle kontseptsiooni eesmärk on pakkuda eaka inimese elukvaliteedile teatavat lisaväärtust. WHO andmetel on kõige enam eakatega seoses levinud terviseriskid kuulmise ja nägemise kaotus, luu-lihaskonna vaevused, hingamisteede haigused, diabeet, vaimuhaigused ja vigastused [25]. WHO prognooside kohaselt moodustavad 2030. aastal mittenakkuslikud haigused 87% eakate elanikkonna haiguskoormusest sõltumata sotsiaalmajanduslikust taustast [43]. Üks suurimatest vigastuste tekitajatest on kukkumine. Igal aastal juhtub kukkumiste tõttu hinnanguliselt 646 000 surma, mis teeb selle peale liiklusõnnetuste teiseks tahtmatute vigastuste surma põhjustajaks [44]. Euroopa Liidus juhtub hinnanguliselt 64% vanuritega juhtunud vigastustest kodus [45]. Kodus esinevatest vanurite terviseriskidest on uuringute kohaselt kõige levinumad juhuslikud kukkumised, mürgitus, kokkamise ja tulega seotud õnnetused. Nendest enim esineb kukkumisi [45, 46]. Igal aastal kukub umbes 30% üle 65-aastastest inimestest ja üle 75-aastaste seas on see määr kõrgem. 20–30% kukkunutest saab vigastusi, mis vähendavad liikuvust ja iseseisvust ning suurendavad enneaegse surma

riski [45]. Ameerika Ühendriikide erakorralise meditsiini osakondade andmete põhjal tehtud analüüsist 2001-2008 selgus, et enamik vanurite kukumisi (73%) juhtub kodustes tingimustes. Kukkumisohvrites moodustasid ligi 80% naised ja kõige levinum kukkumisvigastuste koht oli vannituba (35,7%). Teisteks vigastuste tekkimise põhjusteks olid treppide halb kujundus ja lagunemine, ebapiisav valgustus, komistamine, libedad põrandad, kinnitamata matid [47].

Energiatõhusus on energia raiskamise kõrvaldamine, kasutades sama ülesande täitmiseks vähem energiaressursse [48]. Viimase 30 aasta jooksul on kasvuhoonegaaside heitkogused maailmas kasvanud enam kui 40%. Energiatarbimine on suurim inimtegevusest põhjustatud kasvuhoonegaaside allikas, põhjustades kogu maailma heitkogusest 73%. Sellest 30% moodustab soojuse ja elektri tootmine. See probleem on suunanud riikide valitsusi liikuma ressursside säästlikuma kasutamise suunas [49].

Eurostati andmetel moodustasid 2017. aastal Euroopa Liidus tarbijad umbes 27% energia lõpptarbimisest (energia, mis jõuab lõppkasutajani). Sellest ligikaudu 17,5% tuli taastuvatest energiaallikatest. Alltoodud joonis 9 annab ülevaate Euroopa Liidu keskmise majapidamise energiatarbimisest. Levinumad kodumajapidamiste energiatarbimise allikad on küte (nii toa kui ka vee jaoks, vastavalt 64.1% ja 14.8%), valgustus ja seadmed (vastavalt 14.4%). Kokku moodustavad nad ligi 93% kogu majapidamise energiatarbimisest [50].



Joonis 9. Energia tarbimine Euroopa Liidu majapidamistes Allikas: [50]

Tervislikku vananemist ja energiatõhusust silmas pidades ning kahe oma valdkonna pikaajalise kogemustega kliendi soovitusi arvesse võttes (KPN ja BAM), valis projekti meeskond simulatsiooni sobilikud koduautomaatika seadmed, mida rakendati stsenaariumite disainis. Tehtud valikutest annab ülevaate järgmine punkt 3.3.1.

3.3.1 Koduautomaatika

“Tulevikukodu” projekti simulatsiooni koduautomaatika seadmeteks valiti LED-tuled ja infrapunaküte. Mõlemad seadmed käivituvad sensorite ajendil. Sensorid on ühenduses Targa maja haldava üldsüsteemiga. Valik langetati nii kliendi soovitude põhjal kui ka tervisliku vananemise ja energiatõhususe aspekte silmas pidades.

Sensorid - koduseadmete automatiseerimise ja andmeside hõlbustamiseks on koduste mõõtmis- ja tuvastussensorite kasutamine hädavajalik. Riikliku Teadusnõukogu definitsiooni järgi on sensor seade, mis saab keskkonnast väljundi (signaali) ja levitab selle juhtmega või traadita võrgus täiturmehhanismini vastusena konkreetsele muutujale. Täiturmehhanism on aga mehaaniline seade, mis võtab väljundi vastu, kontrollib ja muundab signaali interneti võrgu ja seadmete kaudu vastavaks vastureaktsiooniks. Koostöös sensorite ja täiturmehhanismidega moodustub Tarka maja haldav süsteem [51].

Simulatsioonis on eesmärk rakendada liikumisandurit öise valgustuse sisselülitamiseks ning infrapuna paneelide aktiveerimiseks. Liikumisandur muundab ruumis liikuva objekti asukoha

või asendi muutuse elektrisignaali [52]. Samuti kasutatakse simulatsioonis sensoreid, mis tuvastavad akente lahtioleku.

LED-tuled

LED-lambid ehk valgusdiodlambid on tõhusad valgusallikad, milles esineb kvantmehaaniline protsess, mille käigus energia muutub ühest olekust teise. Erinevalt hõõglampidest (traditsioonilistest lambipirnidest) ja gaasilampidest on LED-tuledel suurem valgusviljakus ja oluliselt pikem eluiga. Kui hõõglambid kiirgavad suure osa oma energiast keskkonda soojusena, siis LED-lambid eraldavad enamus oma energiast valgusena [53, 54]. Energy Star kvalifikatsiooniga (energiatõhusate tarbekaupade rahvusvaheline standard) LED-lamp tarbib vähemalt 75% vähem energiat ja kestab 25 korda kauem kui hõõglamp [55], mis õigustab nende kasutamist projekti simulatsioonis energiatõhususe aspekti silmas pidades. Automatiseeritud valgustus võib olla abistav faktor eakate inimestele, kellel on nägemise või liikumisprobleemid. On leitud, et nutikas (automatiseeritud) öövalgustus võib aidata vähendada eakate seas koduste vigastuste riski. Samuti on sensorite abil võimalik seadistada LED-tulede temperatuuri erinevalt päevast ja ööst, mis soodustab nägemise kohanemist pimedas olukorras [56, 57].

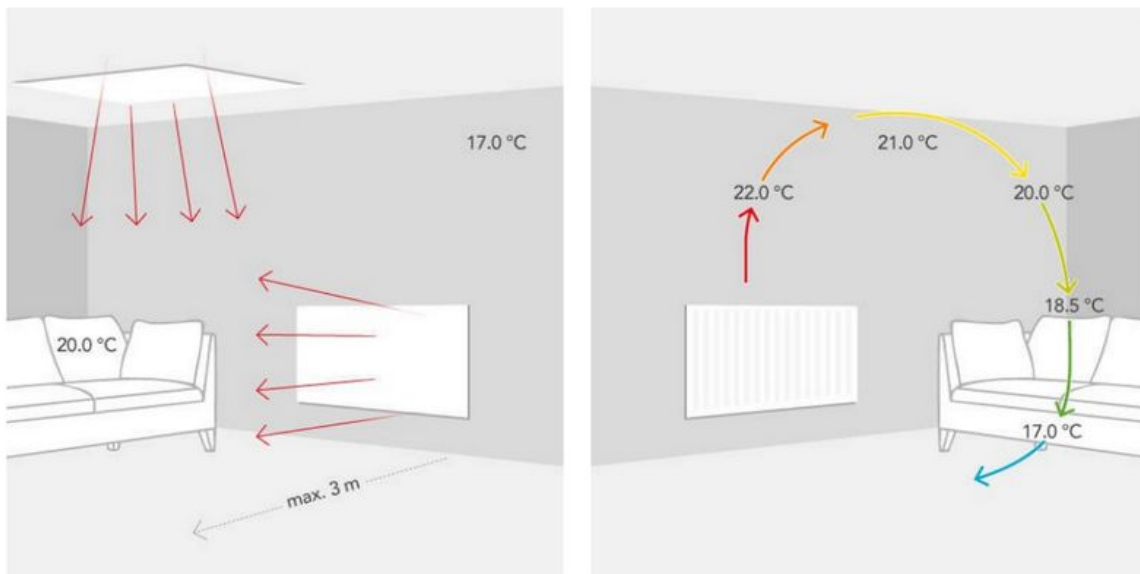
Infrapunaküte

Infrapunakütte algne levik küttesüsteemide maailmas toimus peale teist maailmasõda. Peale kaug-infrapunaküttepaneeli (Joonis 10) turule tulekut on seda hakatud eelitsama teistele küttesüsteemidele, sest on energiasäästlikum, paindlikumate paigaldamisvõimalustega ning tervisele ohutum [58]. Eelnevaid väiteid põhjendavad järgmised lõigud.



Joonis 10. Kaug-infrapunaküttepaneel [59]

Meie suurim infrapunakütteallikas on päike, mis võib isegi külmal talvapäeval soojana tunduda. Sama põhimõtet kasutatakse infrapunakütte tehnoloogias. Erinevalt tavalisest radiaatoritest, mis soojendavad kogu ruumi õhku, soojendavad infrapunapaneelid erinevaid objekte ruumis. Konveksioonkütte (valdav soojuse ülekandumine toimub ruumi madalama temperatuuriga õhu kokkupuutel kõrgema temperatuuriga küttekeha pinnaga [60]) peamiseks probleemiks on see, et köetud soe õhk on kergem kui külm õhk. Selle tulemusel tõuseb soe õhk laeni, jahedam õhk püsib põrandale lähemal. Lühikese aja jooksul jahtub soe õhk, mida tuleb uuesti soojendada. See tähendab, et sama õhu soojana hoidmiseks peab ruumi pidevalt kütma ning et soe ja jahe õhk jaotub ruumis ebahõltselt. Infrapunaküte töötab aga kiirguse abil. Kiirgus soojendab õhu asemel objekte ja inimesi kuni kolme meetri kaugusel paneelist. Soojendatud objektid säilitavad kuumust kauem kui õhk säilitab soojust, nii eraldavad objektid soojust ka teistele objektidele, mis hoiab lõpuks kogu ruumi soojana (Joonis 11). Infrapunaküttekehad suudavad optimaalse küttemperatuuri saavutada 30 sekundi jooksul ning jaotavad soojuse üle terve ruumi ühtlaselt. Neid saab paigaldada igasse ruumi vabasse seinale. On hinnatud, et infrapunaküttepaneel on võrreldes tavalise elektriradiaatoriga 30-40% energiasäästlikum [61].



Joonis 11. Infrapunaküte (vasakul) ja konveksioonküte (paremal) [62]



Konveksioonküte õhku soojendades samuti kuivatab seda. Liialt kuiv õhk soodustab ruumis viiruste levikut, kuivatab meie nahka ja meie limaskestasid ehk oleme bakteritele ja viirustele vastuvõtlikumad. Infrapunaküte hoiab aga stabiilselt tervisele soodsat 30-40% õhuniiskust. Õhu liikumise puudumise tõttu ei ringle ruumis ka tolm ja õietolmu, mis teeb selle heaks küttelahenduseks allergikutele. Infrapunaküte pärsib ka hallituse teket soojendades objekte sealhulgas ka seinu, mis püsivad kuivana [62].


Kolmandas peatükis uuritud vanurite seas ilmnevaid terviseriske, koduautomaatika seadmeid ja nende energiatõhusust rakendati stsenaariumite disainis, mida kirjeldab järgmine punkt.

3.4 Stsenaariumite disain

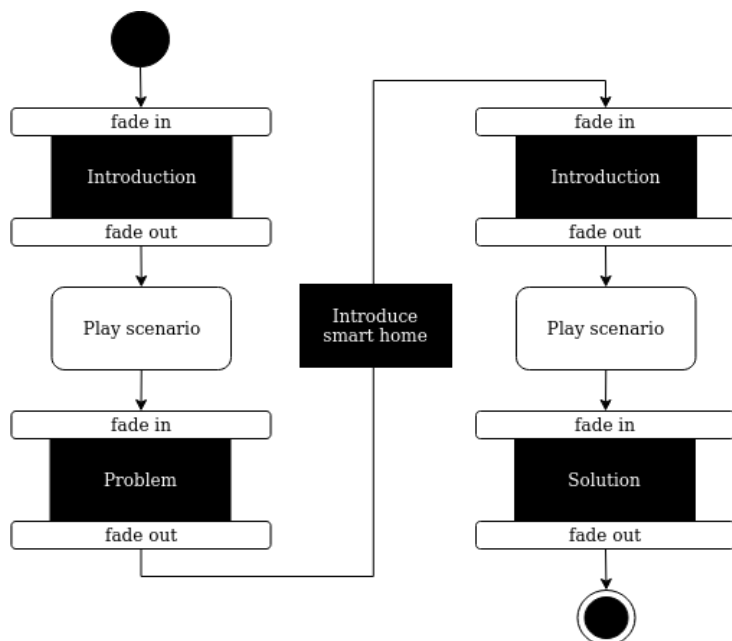
Töögrupp töötas välja kaks kliendi nõudmistel põhinevat stsenaariumit, mis sisaldavad tervisliku vananemise ja energiatõhususe aspekte ning meeskonna poolt koostöös kliendiga välja valitud koduautomaatika seadmeid ja süsteeme.

Põhinedes punkti 3.3 toodud peamistele eakate seas esinevatele terviseprobleemidele ja projekti arendamise võimalustele Unity keskkonnas, otsustati luua kolm vanuri isikut kolme erineva terviseseisundiga: terve, halva silmanägemisega ja liikumispuudega eakas.

	<p>Nimi: Marieke Weerts</p> <p>Sugu: Naine</p> <p>Vanus: 68</p> <p>Terviseprobleem: Puudub</p>		<p>Nimi: Karin Veenstra</p> <p>Sugu: Naine</p> <p>Vanus: 65</p> <p>Terviseprobleem: Halb silmanägemine</p>
---	--	--	--

	<p>Nimi: Aart Dubbink</p> <p>Sugu: Mees</p> <p>Vanus: 78</p> <p>Terviseprobleem: Liikumispuue</p>
---	---

Stsenaariumite eesmärgiks on simuleerida võimalikult realistlikult eakate kodus igapäevaselt ettetulevaid olukordi. Proovitakse kasutajas tekitada võimalikult suur äratundmis moment. Mõlema stsenaariumi läbimäng käib kahes osas. Esimene simulatsiooni pool toob välja uuringu käigus selgunud vanurite seas levinud terviseprobleemi ja/või energiaraiskamise majas, kus puutuvad koduautomaatika seadmed ja süsteemid antud igapäevases olukorras. Stsenaariumi esimese poole lõpus rõhutatakse kasutaja poolt just läbi mängitud ja kogetud tavamaja kitsaskohti ning juhatatakse sisse “Tulevikukodu” stsenaariumi pool. Kasutaja mängib nüüd sama situatsiooni läbi uues, innovatiivses keskkonnas. Teine stsenaariumi pool esitab võimalikke Targa maja poolt pakutud lahendusi eelmises stsenaariumi pooles esile kerkinud takistustele. Kasutaja läbib situatsiooni sujuvalt ja turvaliselt ilma probleemideta. Läbimängu lõppedes antakse kokkuvõtlik ülevaade Targa maja kasulikkusest seoses tervisliku vananemise ja energiasäästlikkusega antud probleemi juures (Joonis 12).



Joonis 12. “Tulevikukodu” stsenaariumi ülesehitus Allikas: Töö autor

Simulatsioon toimub BAMi poolt disainitud 3D majas (Lisa 2). Majal on kaks korrust ja ta pakub realistlikku Hollandi kodu meenutavat atmosfääri. Majas on magamistuba, elutuba, vannituba ja wc. Maja on möbleeritud ja sinna on lisatud interaktiivsed 3.4.1 punktis välja toodud Targa maja koduautomaatika seadmed.

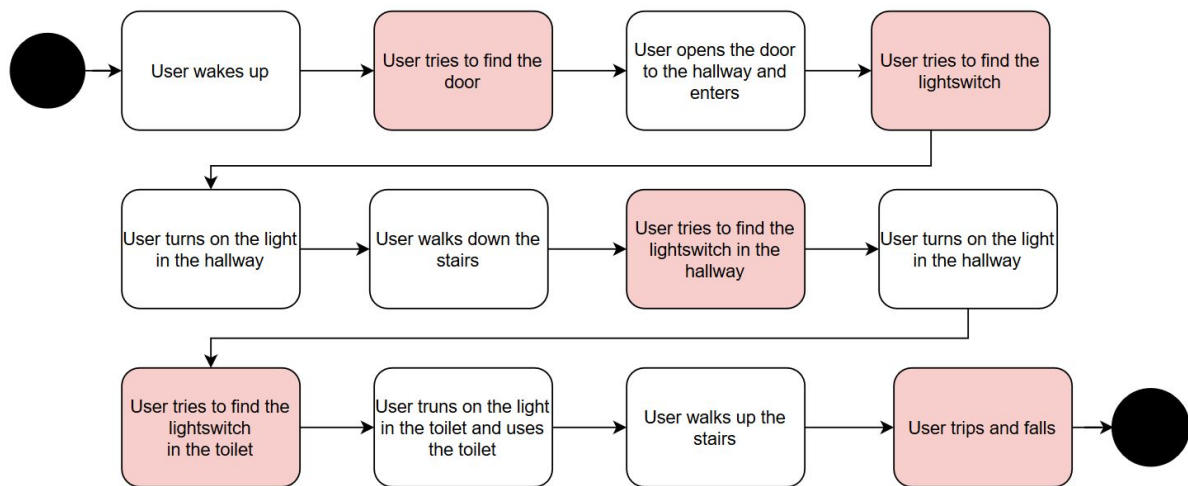
Järgnevalt kirjeldatakse kahte väljatöötatud stsenaariumit detailsemalt. Esitatakse ka vastavad joonised, kus punaste kastiga on märgistatud läbimängu probleemid ja roheliste kastidega võimalik lahendused nendele.

3.4.1 “Pimedas vannituppa” stsenaarium

“Pimedas vannituppa” stsenaariumis on läbimängija eesmärgiks käia öösel tualetis ja naasta magamistuppa. Stsenaarium käsitleb vanurite seas levinud kukkumise teel saadud vigastuste terviseriski ning öösel esinevat halva nähtavuse probleemi manuaalse valgustusüsteem puhul.

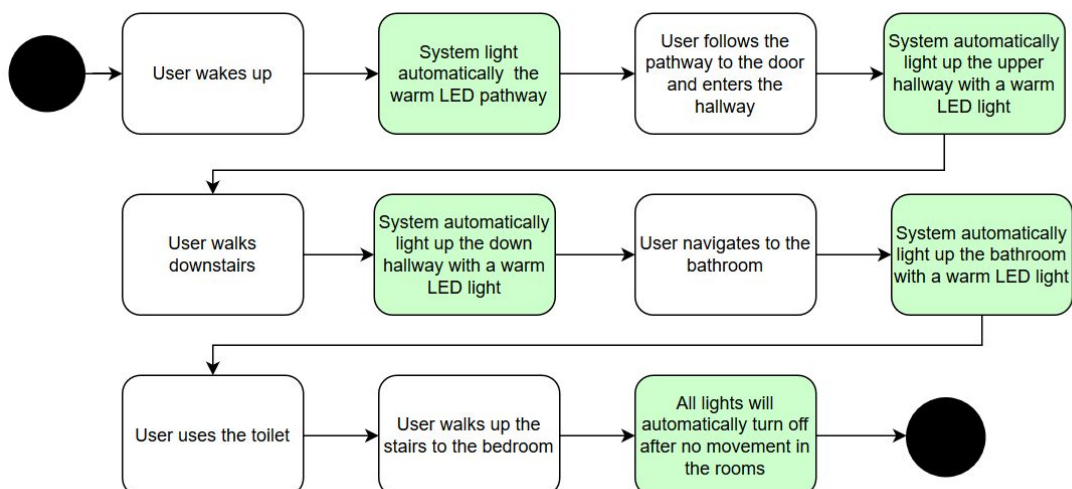
Probleemi läbimängu kirjeldus: Kasutaja ärkab öösel üles ning proovib pimedas jõuda ukseni. Kuna on pime, siis on tegevus raskendatud. Kasutaja ei taha tulesid magamistoas sisse lülitada, sest see võib üles äratada tema endiselt magava partneri. Magamistoast väljudes otsib ta koridori valguslülitit. Valguse sisselülitamisel pimestab see hetkeks kasutaja nägemise. Kasutaja kõnnib trepist alla koridori ning proovib leida ka seal lülitit. Vanitoas

lõpetades kustutab ta alumise korruse tuled. Trepist üles kõndides komistab ja kukub (Joonis 13).



Joonis 13. Probleem läbimäng “Pimedas vannituppa” stsenaariumis

Probleemi lahenduse läbimängu kirjeldus: Kasutaja ärkab üles Targas majas. Koduautomaatika sensorid märkavad liikumist ning omavad informatsiooni öisest kellaajast. Selle tulemusena lülitavad sensorid sisse sooja temperatuuriga LED-ribade valgustuse, mis ei ärataks üles kasutaja partnerit, aga samas teeksid magamistoas liikumise hästi nähtavaks. Magamistoast väljudes kustuvad magamistoa tuled automaatselt ning LED-tuled süttivad automaatselt koridoris. Sama protsess toimub trepil, all koridoris ja vannitoas (Joonis 14).



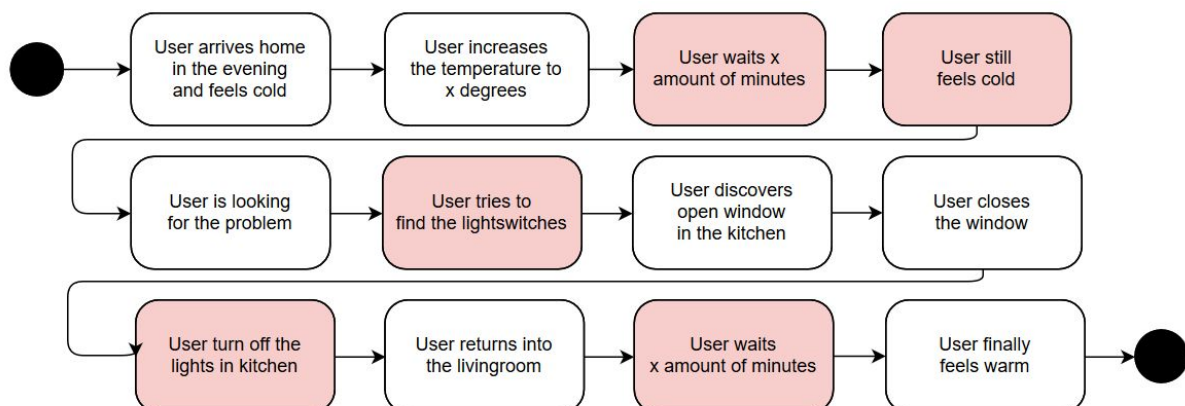
Joonis 14. Probleem lahenduse läbimäng “Pimedas vannituppa” stsenaariumis

Kasutaja ei pea ühtegi lülitit ise vajutama ning saab ohutult oma kodus tualetti liikuda. See on abistavaks faktoriks nägemis- ja liikumisraskustega inimeste jaoks. Valgustid põlevad ainult kasutaja liikumise korral ehk energiat tarbitakse efektiivselt. Samuti on LED-tuled 70% energiasäästlikumad ehk stsenaarium sisaldab energiatõhusust. Stsenaariumi läbi mängides ilmub lõpustseenis ekraanile palju säästeti antud stsenaariumi korral LED-tuledega elektrit ja raha võrreldes hõõglampidega. Lisaks antakse ülevaade, palju on nii võimalik säästa energiat ja raha aasta jooksul (Lisa 3). Tuuakse välja punktis 3.3 uuritud kukkumiste statistika eakate seas ning rõhutatakse üle Targa maja eelis kukkumiste vältimiseks.

3.4.2 Stsenaarium “Infrapunaküte”

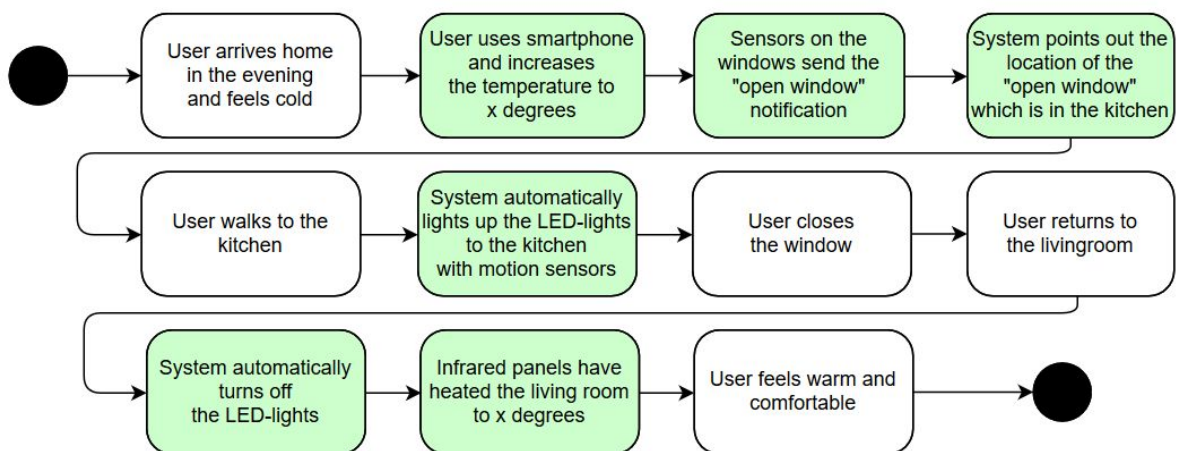
“Infrapunakütte” stsenaarium näitab infrapunakütte eeliseid konvektsioonkütte ees. Selles stsenaariumis rõhutatakse Targa maja infrapunakütte energiatõhusust, sensorite suhtlust Targa maja süsteemiga ja automaatvalgustuse eeliseid turvalise liikumise soodustamiseks.

Probleemi läbimängu kirjeldus: Kasutaja on õhtuselt poeskäigult koju saabunud. Õues on pakane ehk kodu on jahe. Kasutaja otsustab kütte termostaadil sisse lülitada ning istub diivanile telekat vaatama. Maja soojenemisega läheb aega. Minutid mööduvad, aga kasutajal on endiselt külm. Kasutaja kõnnib probleemi otsides majas veidi ringi ja avastab, et alumise korruse köögis on aken lahti jäetud. Teekonnal kööki peab ta otsima valgustite lüliteid, mis suurendab komistamise ja kukkumise riski. Akna sulgemise järel kustutab kasutaja tuled ja naaseb diivanile telekat vaatama. Möödub teatud ajavahemik ning kogu kodu on soojaks kõetud (Joonis 15).



Joonis 15. Probleem läbimäng “Infrapunaküte” stsenaariumis

Probleemi lahenduse läbimängu kirjeldus: Kasutaja saabub poeskäigult koju ja soovib kodus telekat vaadata. Ta tahab koheselt jahedas majas kütte sisse lülitada. Kasutaja võtab välja nutitelefoni ja lülitab infrapunaküttepaneeli püsivalt sisse elutoas diivani juures. Nutitelefoni abil saab ta kütte sisse lülitada korraga kogu majas, kontrollida individuaalseid paneele või lülitada sisse liikumisanduritel põhinev küttesüsteem. Kütte sisselülitamise järel saadavad akendel olevad sensorid nutitelefoni teate, et üks aken on majas lahti. Samuti on täpsustatud, et lahti on köögi aken. Kasutaja läheb sulgeb akna. Teekonnal kööki ja tagasi elutuppa toimib majas automatiseeritud LED-valgustussüsteem ehk kasutaja ei pea ise ühtegi lülitit sisse ega välja lülitama. Elutuppa naastes on infrapunaküte toa juba mugavalt soojaks kütnud. Kasutaja istub diivanile telekat vaatama (Joonis 16) .



Joonis 16. Probleem lahenduse läbimäng “Pimedas vannituppa” stsenaariumis

Stsenaariumi lõpus ilmub ekraanile informatsioon stsenaariumi energiasäästust - kui palju kiirem, odavam ja energiatõhusam oli infrapunaküte võrreldes probleemi läbimängus esineva konvektsioonküttega (Lisa 4). Tark maja kütab soovi korral majas ainult kasutuses olevaid ruume ning ruum soojeneb soovitud temperatuurini vaid mõnekümne sekundiga. See on suur energiasäästu punkt arvestades punktis 3.3 välja toodud statistikat, et küttele läheb majapidamises keskmiselt umbes 64% kogu kasutatavast energiast. Informatsiooni slaidil tuuakse välja ka LED-tulede efektiivsus võrreldes hõõglampidega sarnaselt “Pimedas vannituppa” stsenaariumiga. Samuti on automatiseeritud valgustus abistavaks omaduseks majas pimedas liikumisel. See vähendab kodus kukkumise riski, mis kaasneb tihti pimedas ruumis valguslülitite otsimisega.

Järgnevalt rakendati stsenaariumite disaini mänguarendusplatvormis Unity, et välja töötada reaalselt toimiv virtuaalreaalsuse simulatsioon. Disaini- ja arendusprotsess koostöös klientidega viidi läbi Scrum meetodil. Scrum meetodi ja projekti “Tulevikukodu” arenduse tulemuslikkusest annab ülevaate järgmine peatükk.

4. Projekti tulemuslikkus

Peatükk kirjeldab projekti protsessi viie kuu vältel. Tuuakse välja peamised töö jooksul tekkinud takistused, ettevalmistavad protseduurid enne Scrumi sprinte, viis projekti jooksul läbi viidud sprinti ja toote lõpptarne kliendile. Samuti antakse ülevaade klientide esitatud nõuete täitmisest ning soovitatakse võimalikke edasiarendusi tulevikus.

4.1 Tekkinud probleemid

“Tulevikukodu” projekti käigus tekkis mitmeid probleeme, mis mõjutasid suuresti punktis 2.3.1 kirjeldatud algselt planeeritud arenduse ajakava. Peamiseks probleemide allikaks oli märtsis Euroopasse jõudnud COVID-19 ehk koroonaviirushaiguse pandeemia, mille tõttu pidi meeskond tarkvaraarenduses ja projektijuhtimises tegema mitmeid muudatusi. Järgnevates lõikudes kirjeldatakse peamiseid koroonaviirusest tingitud probleeme.

Keelati ära kontaktkohtumised - 11. märtsil klassifitseeris Maailma Tervishoiuorganisatsioon koroonaviiruse pandeemikaks, mil surnute arv ületas 4000 piiri [63]. Euroopa riigid võtsid kasutusele mitmeid ühiskonna tavaelu mõjutavaid meetmeid, kaasa arvatud Hollandi valitsus, kelle nõudmisel pidid kõik Hollandi haridusasutused kontakttunnid määramata ajaks ära jätma. Seda tegi ka Hanze Rakenduskõrgkool ehk alates 13. märtsist polnud projektimeeskonnal enam võimalust üksteise, kleintide ja projekti juhendajaga füüsiliselt kohtuda.

Kadus ligipääs virtuaalreaalsuse laborile - peale piirangute kehtestamist ei saanud meeskond enam kasutada virtuaalreaalsuse laborit tarkvara testimiseks ja arenduseks.

Kadus võimalus testida simulatsiooni sihtrühma peal - algselt oli planeeritud valminud toodet testida hooldekodudes eakate seas, et näha, kas toode täitis oma eesmärgi ning saada vajalikku tagasisidet parandusteks. COVID-19 viiruse üks peamisi riskigruppe olid vanurid ehk kontakt hooldekodu elanikega oli keelatud.

Projekti rahvusvahelise meeskonna kaks liiget pidid naasma koduriikidesse - kõik projekti meeskonnaliikmed olid eri rahvusest (Holland, Poola, Eesti ja Jamaica). Pandeemia saabudes kutsusid enamik välisministeeriume oma riigi kodanikke koju. Seetõttu kolisid Scrumi meister Hanna-Marii Kaljas ja arendaja Piotr Pawlowski projekti ajal tagasi oma

koduriikidesse (Eestisse ja Poola), mis mõjutas teatud ajavahemikul nende kaasatust projekti arendusse.

Kaks klienti astusid ajutiselt projektist tagasi - koroonaviirus tekitab mitmeid probleeme ja väljakutseid Hollandi ettevõtetele ja omavalitsustele. Groningeni linnavalitsus ja KPN otsustasid märtsi keskel kaheks kuuks “Tulevikukodu” projektiarendusest kõrvale astuda ning pühenduda täielikult olukorra stabiliseerimisele oma organisatsiooni siseselt.

Projekti juhendaja poole koormusega kohalolek - juuni algusest kuni projekti lõpuni töötas meeskonna juhendaja Lech Bialek oma perearsti nõudmistel poole koormusega, sest tal esinesid läbipõlemise sümptomid. See vähendas tema kohalolu meeskonna mentorina viimasel projekti kuul.

Tooteomaniku tagasiastumine juuni lõpus - meeskonna tooteomanik (*Product owner*) Dante Baker otsustas juuni lõpus enne toote tarnimist projektist isiklikel põhjustel välja astuda.

Jäi ära tuur BAM'i näidiseks ehitatud Tarka majja - üks projekti klientidest BAM planeeris meeskonnale tutvustava ekskursiooni Groningenis asuvasse näidis Tarka majja, mis sisaldab mitmeid koduautomaatika seadmeid ja süsteeme. Tuur pidi toimuma 13. märtsil, aga koroonapiirangute tõttu jäi sündmus ära.

Kõik eelnevad probleemid mõjutasid Scrum meetodi rakendamist ja kliendi esitatud nõuete täitmist projekti jooksul. Järgmises peatükis kirjeldatakse projekti reaalselt protsessi ja tulemusi ning mil määral rahuldati tootele määratud kliendi poolset nõudeid.

4.2 Projekti protsess

Selles peatükis kirjeldatakse projekti “Tulevikukodu” Scrum meetodi rakendamise protsessi. Järgmistes lõikudes antakse ülevaade sprintidele eelnevatest tegevustest ja igast toimunud sprindist eraldi kuni toote lõpptarneni klientidele.

Tegevused enne sprinti (1.-3. nädal)

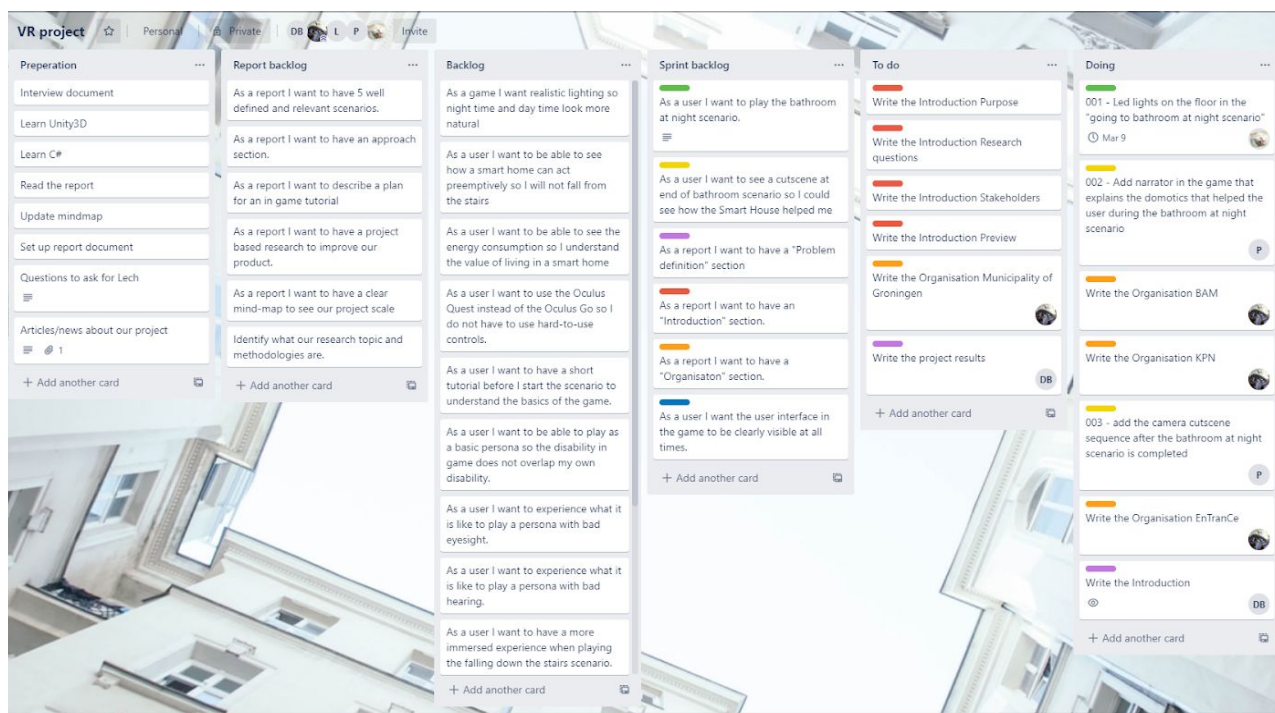
Projekt toimus Hanze Rakenduskõrgkoolis programmi “Tark Energia” raames, mille eesmärgiks on ühendada energiasektor ja arvutiteadus. Esimesel programmi nädalal toimus viiepäevane intensiivne koolitus koos kaasavate töötubadega. Nädala jooksul anti üliõpilastele ülevaade Hollandi ja maailma energiasektori hetkeolukorrast ning arutati, kuidas

võiks toimuda inimkonna energia üleminek fossiilsetelt kütustelt taastuenergiale. Selle nädala eesmärgiks oli üliõpilased võimalikult palju kurssi viia programmis esitatud projektide valdkonnaga. Teisel programmi nädalal tutvustati üliõpilastele “Tark Energia” programmi projekte ning kõik kursusel osalejad pidid esitama oma valikueelistused. Programmi juht Lech Bialek moodustas eelistustele ja tudengite oskustele põhinedes neli erinevat projekti meeskonda. Töö autor valiti projekti “Tulevikukodu” meeskonna liikmeks, mille probleemi kirjeldab käesoleva töö punkt 2.1. Järgnevalt tutvustati tudengitele erinevaid agiilse tarkvaraarenduse meetodeid k.a. Scrum (Punkt 1.3). Teisel nädalal toimus esimene meeskonnakohtumine. Meeskond otsustas projektijuhtimismeetodiks valida Scrum meetodi selle detailse organiseerituse, lihtsa kasutuselevõtu ja klientide tiheda kaasatuse tõttu. Kohtumisel tutvuti üksteisega ja jaotati ära Scrum rollid (Punkt 2.3.2). Neljast meeskonna liikmest vaid üks oli eelnevalt Scrum meetodit kasutanud (Willem Pepping). Teistele kolmele oli meetod uus ning vajab põhjalikumat sisseelamist. Kolmandal nädalal kohtus töögrupp kolmel korral. Moodustati projekti mõttekaart (Lisa 5), et kaardistada enda jaoks projekti ulatus. Ideekaardi eesmärgiks oli ideede ja mõistete kaardistamine, seoste loomine ja tegevuste planeerimine. Peale mõttekaardi loomist koostati projekti eeldatav ajakava (Punkt 2.3.1) ja valmistati ette küsimused klientide intervjuuks (Lisa 2). Kolmanda nädala lõpus toimus intervjuu projekti klientidega. Intervjuul osalesid kõigi nelja kliendi esindajad (Punkt 2.2). Tutvustati Scrum meetodit ning kliendi kaasatust sellesse planeeritud viie sprinti jooksul. Kliendid olid esitletud ajakavaga rahul. Intervjuu käigus kaardistati ära klientide funktsionaalsed ja mittefunktsionaalsed nõuded (Punkt 3.2). Peale esimest kohtumist klientidega algas projekti esimene sprint.

Esimene sprint (4.-7. nädal)

Neljandal nädalal algas esimene 4-nädalane sprint, mis oli algselt planeeritud kolme nädalaseks. Koroonaviiruse saabumisega viienda nädala lõpus nihkus sprinti esitlus kliendile nädala võrra edasi, sest kõik organisatsioonid kohanesid ootamatult tekkinud olukorraga. Esimese sprinti peamine eesmärk oli tutvuda arenduskeskkondadega, uurida vanurite seas peamisi esinevaid terviseprobleeme, uurida erinevate koduautomaatika süsteemide ja seadmete kohta, alustada stsenaariumite disainiga ning esimese sprinti lõpus esitleda kliendile esimest algset stsenaariumit.

Sprindi alguses koostati Trello klientide nõuetel põhinev tegemata tööde nimekiri (*Product backlog*) kasutajalugu (*User stories*), mille prioritiseeris tooteomanik Dante Baker. Prioritiseering põhines nõutetele määratud MoSCoW analüüsil (Punkt 3.2). Seejärel moodustati kuuest esimesest ülesandest sprindi tegemata ülesannete nimekiri (*Sprint backlog*). Iga sprindi tegemata tööde nimekirja ülesanne märgistati ära värviga ning tükeldati veel väiksemateks ülesanneteks “teha vaja” (*To do*) veergu. Seal jagas Scrumi meister ülesanded meeskonna liikmetele laiali “tegemisel” listi (*Doing*). Sama protsess kordus iga sprindi algul. Samuti kirjeldati ettevalmistavad tegevused (*Preparation*). Nendeks olid tegevused, mis on eelduseks nõute täitmisel: arendustiim viis end kurssi Unity, Oculus Go ja C# programmeerimisega süsteemis Unity (Punkt 2.3.3); täiendati mõttekaarti; alustati projekti dokumenteerimisega (Joonis 17).



Joonis 17. Esimese sprindi Scrumi tahvel Trello keskkonnas.

Neljanda nädala alguses sai meeskond enda kasutusse Hanze Rakenduskõrgkoolis olevad virtuaalreaalsuse labori, mis hõlmas endas Oculus Go komplekti ja mänguarenduseks sobiliku võimsusega lauarvutit. Laboris kohtuti teisipäevast-reedeni igal hommikul kell 10.00 püstijalakoosolekuteks. Kokkusaamistel andsid töögrupi liikmed üksteisele kokkuvõtliku ülevaade juba tehtust, mida tehakse järgneva päeva jooksul ning mis on seni esile kerkinud peamised takistused. Scrumi meister oli vastutav, et tekkinud probleemidele

leitakse igapäevaselt lahendus. Selleks kas jaotati ülesandeid ringi või muudeti lähenemisviisi. Projekti vältel toimus vaid 6 füüsilist püstijalakohtumist, sest viienda nädala lõpus keelati ülikoolis COVID-19 tõttu ära kontakttunnid. Seetõttu pidi meeskond alates kuuendast nädalast kohtumisi jätkama Discordi vahendusel, mis on häälvestlusi võimaldav platvorm internetis. Vaatamata esilekerkinud probleemidele sujus koostöö libedalt, sest meeskonna suhtlus omavahel oli aus, läbipaistev ja üksteist toetav. Samuti toimus iga nädal kohtumine juhendaja Lech Bialekiga, kes aitas ja suunas tekkinud takistuste puhul. Kohtumine mentoriga toimus füüsiliselt kolmel korral ja edaspidi kord nädalas Microsoft Teams platvormi vahendusel. Kogu meeskonnatöö viidi üle ajutiselt üle veebi. Kõik dokumentatsiooni hoiti Google Drive keskkonnas, meeskonna sisene kirjalik suhtlus toimus WhatsAppis, sest seal olid kõik meeskonnaliikmed kättesaadavad nii telefoni kui arvuti teel. Koodi jagati GitHubi keskkonnas. Kõik kohtumised - mentoriga kohtumised, sprindi esitlused kliendile, hommikused püstijala kohtumised märgiti ära projekti jaoks eraldi loodud Google kalendrisse. Kuuendal ja seitsmendal nädalal toimusid püstijala kohtumised veebis üle päeva kolmel korral.

Tänu juhendaja loale sai kuuendal nädalal meeskonna arendaja Willem Pepping endale ülikooli Oculus Go riistvara koduseks kasutamiseks, et testida riistvara virtuaalreaalsuses. Seitsmenda nädala lõpus esitles meeskond kliendile esimese sprindi tulemusi. Olude sunnil toimus esitlus BlackBoard keskkonnas, mida Hanze Rakenduskõrgkool kasutab oma veebipõhise õppekeskkonnana. Tehti kokkuvõtlik ülevaade simulatsiooni kvalitatiivse uuringu tulemustest seoses tervisliku vananemisega (Punkt 3.3) ja anti ülevaade, milliseid erinevaid koduautomaatika süsteeme tarkades majades kasutatakse. Kuna koroonaviiruse tõttu jäi ära ekskursioon BAMi tarka majja Groningenis, siis pidi meeskond leppima ainult teoreetilise infoga koduautomaatika vallas. Sprindi ülevaate kohtumisel esitleti kliendile ka esialgseid “Pimedas vannituppa” ja “Infrapunakütte” stsenaariumeid. Arendajad näitasid Unitys valminud “vaba liikumise” režiimi (*free walking mode*) BAMi 3D mudelmajas. Peamiseks tagasisideks oli, et stsenaariumid disain sisaldasid liialt palju koduautomaatikat, mis pidavat stsenaariumid eakatele liialt eksitavaks muutma. Kuna enamus sihtgrupi jaoks on virtuaalreaalsus uus asi, siis ei tohiks stsenaariumeid informatsiooniga üle paisata. Need peaksid olema lihtsasti läbi mängitavad ja mõistetavad. Samuti puudus stsenaariumites veel energiatõhususe aspekt. Kliendid olid rahul simulatsiooni lisatud nägemisvaeguse ja liikumispuudega karakteritega ning kodus vanurite seas sagedasti esineva kukkumise

terviseriski väljatoomisega. Samuti oldi rahul BAMi majas loodud Hollandi kodule omase atmosfääriga. Klient soovitas stsenaariumid disainida vormis, et algselt mängitakse läbi probleem ning sellele järgnevalt probleemile loodud Targa maja lahendus (Punkt 3.4).

Teine sprint (8.-10. nädal)

Kaheksas nädal algas meeskonna sprindi tagasivaatega. Meeskond tõdes, et näost-näku kohtumiste veebi viimine ja töölabori puudumine raskendab koostööd. Arutati tekkinud muresid ning Scrumi meister otsustas püstijala kohtumised taas igapäevaseks teha. Samuti koostati uus kolmenädalase sprindi tegemata tööde nimekiri. Endiselt lootis meeskond, et koroonal olukord taandub ning varsti saab tööd jätkata tavaviisil. Selle sprindi eesmärgiks oli lõpetada stsenaariumite disain, et arendusmeeskond saaks järgmine sprinti alustada nende täieliku implementeerimisega Unity keskkonnas. Selle sprindi lõpus oli Scrumi meistri kohustuseks klientidele esitada tegevusaruanne (*Progress report*) projekti sisust, seni toimunud tegevustest ja tuleviku eesmärkidest (Punktid 2.1 - 3.3.2). Arendusmeeskond alustas mängu menüü loomisega ning karakterite rakendamisega “vaba liikumise” režiimis (Lisa 7). Püstijala kohtumised toimusid sprindi vältel neli korda nädalas kell 10.00 hommikul Discordis ning kõik liikmed töötasid projekti kallal kodustes tingimustes. Üheksandal nädalal tabas meeskonda tagasilöökk, sest KPN ja Groningeni linnavalitsus teatasid projektist ajutiselt tagasi astumisest COVID-19 olukorra tõttu. See tähendas nende mitteosalemist järgmisel sprindi esitlusel. Samuti naases Piotr Pawlowski üheksandal nädalal tagasi oma koduriiki Poola ning oli nädal projekti arenduses eemal. Seetõttu pidi Scrumi meister arendajate ülesanded sellel nädalal ümber jagama. Programmeerimise koormus läks ajutiselt ainult Willem Peppingu peale. Meeskond hakkas ajakavast maha jääma ning toimus motivatsiooni langus ebaselge tuleviku tõttu. Sprindi esitlusel kümnenda nädala lõpus saavutati rahuldavalt küll kõik sprindi algul seatud eesmärgid ja esitati projekti tegevusaruanne, aga ruumi oli mitmeteks edasiarendusteks. Kliendile sobis stsenaariumite läbimängu skeem, aga endiselt polnud kasutajale selgesti välja toodud energiatõhususe aspekt. Kliendid soovitasid lisada energiatõhususe aspekti läbi kalkulatsioonide simulatsiooni lõpus, mis arvutavad välja stsenaariumi läbimängu jooksul nii LED-tulede efektiivsuse võrreldes hõõglampidega kui ka infrapunakütte energiatõhususe võrreldes konvektsioonküttega (Punktid 3.4.1 - 3.4.2). Kliendi soovitude kohaselt pidi meeskond lisama ka energiatõhususe fakte stsenaariumi lõppu. Arutati ka sihtrühma peal testimise probleemi, mille läbiviimine oli võimatu

kehtestatud liikumispiirangute tõttu. Otsustati koostada küsimustik Google Formsis, mis uurib sihtrühma teadlikkust, eelistusi ja murekohti seoses Targa maja seadmete ja süsteemidega. Tulemusi saab võimaluse korral rakendada stsenaariumite disainis või tuleviku edasiarendustes.

Kolmas sprint (11.-13. nädal)

Üheteistkümnenda nädala alguses tehti järjekordselt sprindi tagasivaade ja pandi paika uue kolmenädalase sprindi eesmärgid. Arvestati kliendi tagasisidega ning seati sprindi eesmärgiks välja töötada küsimustik, seda jagada sihtrühma seas ning tulemusi võimaluse korral rakendada stsenaariumite disainis. Küsimustiku jagas iga meeskonnaliige oma tutvusringkonna eakate seas. Samuti jätkas meeskond kvalitatiivset uuringuga LED-lampide ja infrapunakütte energiatõhususe kohta, et lisada need stsenaariumitesse. Arendustiim implementeeris simulatsiooni LED-tuled, infrapunapaneelide 3D mudelid (Lisa 8) ning alustas stsenaariumite disaini teostamisega. Koodijagamine viidi GitHubist üle Stack keskkonda, sest failid muutusid mahult liiga suureks GitHubis jagamiseks, kuna simulatsioonis oli mitmeid 3D-mudeleid ja funktsioone. Üheteistkümnenda nädala alguses viidi läbi ka esimene grupisisene tagasiside andmine (*Peer assessment*), kus meeskonnaliikmed hindasid üksteise panust. Iga tiimiliige hindas üksteist kuues kategoorias (Tabel 1) ja skaalal -1 kuni 3 (-1 hoiab meeskonda tagasi eesmärgi saavutamisest; 0 temast pole abi meeskonnatöös; 1 veidi madalam panus kui rühma keskmine; 2 rühma keskmine; 3 parem kui ülejäänud).

	Dante	Hanna-Marii	Piotr	Willem
1. Õppimisvõime: omandab uut informatsiooni kiirelt ja oskab seda rakendada	8	9	7	9
2. Motivatsioon tulemust saavutada: meeskonnaliikmel on projekti eesmärk selge ja suur tahe seda teostada	6	8	6	8
3. Panus: tegelik panus lõpptoote arendusse	8	9	7	9
4. Probleemianalüüs: suudab tuvastada probleeme, võime tuvastada probleemide võimalikke põhjuseid	9	9	8	9
5. Koostöö: aitab ühisele tulemusele kaasa, isegi kui käsitletav teema ei paku hetkel isiklikku huvi	9	9	9	9
6. Teistega arvestamine: arvestab teiste meeskonnaliikmete vajaduste ja soovidega	8	9	7	9
Kokku	48	53	44	53

Tabel 1. Meeskonnasisene esimene tagasiside

Kõige parema tagasiside meeskonnakaaslaste poolt said Scrumi meister Hanna-Marii ja arendaja Willem. Madalamat motiveeritust oli märgata Dante ja Piotri poolt. Projektijuht korraldas nendega kolmeteistkümnendal nädalal individuaalsed kokkusaamised Discordis, et tõsta liikmete motivatsiooni ning arvestada rohkem nende vajadustega. Liikmed oli suuresti mõjutatud koroonaviiruse olukorrast, töökeskkonna puudumisest ja faktist, et meeskond ei saa omavahel füüsiliselt kohtuda. Meeskond teadvustas olukorda püstijalakohtumistel ning üritas olukorrast parimat võtta. Kaheteistkümnendal nädalal naases koju ka Scrumi meister Hanna-Marii Kaljas, mille tõttu oli ta protsessist paar päeva eemal. Sprindi esitlusel näidati kliendile seni algset “Pimedas vannituppa” stsenaariumit. Esimese versiooni kohaselt kukkus kasutaja probleemi läbimängus koheselt trepist alla vannituppa suundudes. See aga tegi energiatarbimise lühikese aja tõttu peaaegu olematuks. Kliendi ettepanek oli stsenaariumi pikendada. Samuti palus klient lisada laadimisvaate stsenaariumite käivitamisel. Klient soovis, et kõik simulatsiooni kirjalikud käsud oleksid salvestatud peale loetud häälena. Seda menüüs persoon ja stsenaariumi valiku juures kui ka “vaba liikumise” režiimis interaktiivsetele objektidele klikkides. Meeskond ei jõudnud küsimustiku tulemusi selle sprindi esitluseks analüüsida ning ülevaade lükkus edasi järgmisesse sprinti. Sprindi ülevaate lõpus otsustasid kliendid ja meeskond üheskoos, et olukorrast tingitult ja ajanappuse tõttu

arendatakse lõpptooteni ainult üks stsenaarium. Selleks sai “Pimedas vannituppa” stsenaarium. “Infrapunakütte” stsenaariumi puhul sooviti ainult teoreetilist disaini ja energiatõhususe kalkulatsioone tuleviku edasiarenduseks.

Neljas sprint (14.-16. nädal)

Neljateistkümnendal nädalal algas neljas 3-nädalane sprint. Eelmine sprint lühendati nelja asemel kolmenädalaseks sprindiks, mis hoidis meeskonda algselt püstitatud ajakavas. Eelmise sprindi tagasiside põhjal oli selle sprindi eesmärgiks arenduses täielikult keskenduda “Pimedas vannituppa” stsenaariumile. Tooteomanik korraldas seetõttu ümber Scrumi tahvli tegemata ülesannete prioritseeringu ja Scrumi meister jagas ülesanded uuesti laiali. Lõpetati küsimustiku analüüs. Küsimustikule vastas 42 inimest vanuses 40-95 ehk tulevased ja praegused eakad. Selgus, et vastanutest üle kolmveerandi olid kuulnud Targa maja kontseptsioonist ja üle poole vastanutest olid kokku puutunud Targa maja süsteemide ja seadmetega. See näitab, et simulatsioon võib olla lahenduseks esmakordse ligilähedase kogemuse pakkumiseks sihtrühmale. Peamiste eelistena toodi välja maja mugavus, keskkonnasõbralikkus ja abistav funktsioon. Murekohtadeks olid süsteemide installeerimise ja koduautomaatika kallidus, kahtlused Targa maja turvalisuses ja privaatsuses, informatsiooni puudulikkus ning süsteemide keerukas kasutuselevõtt (tundmatu tehnoloogia). Tulemusi esitleti klientidele ning projekti osapooled nentisid turvalisuse ja privaatsus aspekti puudumist stsenaariumites. Samuti otsustati küsimustiku tulemustele põhinedes lisada järgmisel sprindil stsenaariumi lõppu automatiseeritud LED-tulede paigaldamise maksumus. Selleks saatis BAM omapoolse oletatava paigaldamise hinna (Lisa 9), mis paluti viimasel sprindil stsenaariumisse lisada. Teistele välja toodud murekohtadele peaks simulatsioon eduka valmimise korral lahendusi pakkuma. Arendusmeeskond oli sprindi esitluseks lisanud laadimiskuva ja tekste pealelugeva audio.

Viies sprint ja tarne (17.-20. nädal)

Seitsmeteistkümnendal nädalal algas viies, projekti viimane, 3-nädalane sprint. Viimasel sprindil rakendati kliendi poolt antud tagasisidet seni valminud tootele ning viidi ellu viimased vajalikud parandused. Arendusmeeskond lisas “vaba liikumise” režiimi nutitelefonil väljavõtmise võimaluse, millega saab Targas majas tulesid iseseisvalt sisse ja välja lülitada.

Seda funktsiooni saab tulevikus kasutada “Infrapunaküte” stsenaariumi arenduses (Joonis 18).



Joonis 18. Simulatsiooni lisatud telefoni kasutamise võimalus

Sprindi alguses teatas meeskonna juhendaja, et peab perearsti nõudmistel aja maha võtma ning on kättesaadav kahel päeval nädalas. Meeskond kohanes küll olukorraga, aga see tekitas siiski pingeid. Kaheksateistkümnendal nädalal teatas ootamatult tooteomanik Dante Baker projektist tagasiastumisest. Põhjuseks tõi ta välja isikliku elu raskused põhjustatud COVID-19st ning sellest tulenev otsus ülikool pooleli jätta. Ühe meeskonnaliikme taganemise tõttu pidi Scrumi meister ülesanded ümber jaotama ning koormus kasvas kõigi allesjäänud liikmete jaoks. Scrumi meister võttis enamuse tooteomaniku viimastest kohustustest enda peal. Üheksateistkümnendal nädalal teatasid KPN ja Groningen linnavalituses esindajad, et nad osalevad toote lõpptarne kohtumisel BlackBoard keskkonnas. Kahekümnendal nädalal toimus viimane meeskonnasisene tagasiside andmine (*Peer assessment*) kolmekesi (Tabel 2). Scrumi meistri motivatsioon oli kõige kõrgem, sest temal oli suurim vastutus klientidele nõutele vastav lõpptoode tarnida. Arendaja Willemi kõrge motivatsioon oli tingitud isiklikust suurest huvist Unity ja antud projekti vastu. Arendaja Piotr oli kõige rohkem mõjutatud esile kerkinud probleemidest ning seetõttu oli ta projekti lõpus kõigist liikmetes kõige vähem motiveeritud.

	Hanna-Marii	Piotr	Willem
1. Õppimisvõime: omandab uut informatsiooni kiirelt ja oskab seda rakendada	5	4	6
2. Motivatsioon tulemust saavutada: meeskonnaliikmel on projekti eesmärk selge ja suur tahe seda teostada	6	4	4
3. Panus: tegelik panus lõpptoote arendusse	6	6	6
4. Probleemianalüüs: suudab tuvastada probleeme, võime tuvastada probleemide võimalikke põhjuseid	6	4	6
5. Koostöö: aitab ühisele tulemusele kaasa, isegi kui käsitletav teema ei paku hetkel isiklikku huvi	6	5	6
6. Teistega arvestamine: arvestab teiste meeskonnaliikmete vajaduste ja soovidega	6	4	4
Kokku	35	27	32

Tabel 2. Meeskonnasisene teine tagasiside

Kahekümnendal nädalal tarniti kliendile lõpptoode. Enne simulatsiooni demonstratsiooni esitleti klientidele ettekanne “Tulevikukodu” projekti probleemist, nõuete täitmisest, stsenaariumite disainist ja anti soovitusi edasiarendamiseks tulevikus. Demonstratsioonis mängiti läbi “Pimedas vannituppa” stsenaarium ning näidati “vaba liikumise” režiimis Targa maja interaktiivseid elemente (Lisa 10). Järgmistes peatükkides annab autor ülevaate kliendile esitatud nõuete täitmisest ja edasiarendusvõimalustest.

4.2.1 Nõuete täitmine

Kliendile tarnitud simulatsioon Targast majast rahuldab seitsmest “peab olema” nõudest kuus, viiest “peaks olema” nõudest neli ja kahest “võiks olla” nõudest ühe. Järgnevalt on vaadeldud iga funktsionaalse ja mittefunktsionaalse nõude täitmist eraldi. Täidetud nõuded on tähistatud rohelisega ja täitmata nõuded punasega.

FN-1	Iga simulatsioonis mängitav stsenaarium peab sisaldama energia säästmise aspekti
Täitmine	<ul style="list-style-type: none"> - Simulatsioon sisaldab “Pimedas vannituppa” stsenaariumit, mis omab energiatõhususe aspekti. Stsenaarium toob välja LED-tulede efektiivsuse võrreldes hõõglampidega.

FN-2	Kasutaja peab saama simulatsioonis mängida infrapunakütet sisaldavat stsenaariumi
Täitmine	<ul style="list-style-type: none"> - Kasutajal ei ole võimalik simulatsioonis läbida “Infrapunekütte” stsenaariumit, sest COVID-19 põhjustatud probleemide tõttu ei jätkunud meeskonna ressursse selle täielikuks teostamiseks. - Küll aga on simulatsiooni implementeeritud infrapuna küttepaneelid ning meeskond töötas välja “Infrapunakütte” stsenaariumi disaini.

FN-3	Kasutaja peab saama simulatsioonis mängida stsenaariumi, mille eesmärgiks on öösel tualeti külastus
Täitmine	<ul style="list-style-type: none"> - Kasutaja saab läbida “Pimedas vannituppa” stsenaariumi.

FN-4	Iga simulatsioonis mängitav stsenaarium peab sisaldama mõnda vanuritel esinevat terviseprobleemi
Täitmine	<ul style="list-style-type: none"> - “Pimedas vannituppa” stsenaarium sisaldab kodus komistamise/kukkumise probleemi. - Simulatsiooni saab läbida nägemisvaegusega ja liikumispuudega karakteriga, mis on ühed peamised vanurite seas esinevad terviseprobleemid.

FN-5	Simulatsiooni peab olema võimalik mängida karakteriga, kellel pole terviseprobleemi
Täitmine	- Kasutaja saab simulatsiooni läbida karakteriga, kellel pole ühtegi terviseprobleemi.

FN-6	Simulatsiooni peaks olema võimalik mängida nägemisvaegusega karakteriga
Täitmine	- Kasutajal on võimalik läbida simulatsioon nägemisvaegusega karakteriga.

FN-7	Simulatsiooni peaks olema võimalik mängida liikumisvaegusega karakteriga
Täitmine	- Kasutajal on võimalik läbida simulatsioon liikumisvaegusega karakteriga.

FN-8	Simulatsiooni läbi mängides peaks kasutaja mõistma Targa maja eeliseid
Täitmine	<ul style="list-style-type: none"> - Stsenaariumid on disainitud nii, et rõhutada Targa maja eeliseid ja kasulikkust. - Eelised on välja toodud järgnevatel viisidel: <ol style="list-style-type: none"> 1) stsenaariumi alguses mängib kasutaja läbi probleemi ning sellele järgneb läbimäng kuidas võiks Tark maja probleemile lahendust pakkuda; 2) stsenaariumi lõpus tuuakse välja Targa maja energiatõhusus; 3) Targas majas on interaktiivsed koduautomaatika objektid, millele klikkides kuuleb paari informatiivset lauset selle eseme kasulikkusest ja efektiivsusest.

MFN-1	Simulatsiooni võiks olla võimalik projekteerida suurele ekraanile, et seda saaks demonstreerida arvukamale publikule publikule
Täitmine	- Simulatsiooni on võimalik projekteerida suurele ekraanile.

MFN-2	Kasutajal peab olema lihtne ja arusaadav orienteeruda kasutajaliideses
Täitmine	<ul style="list-style-type: none"> - Simulatsiooni testiti meeskonnasiseselt ja klientide poolt, kellele oli kasutajaliides lihtne ja arusaadav. Eakate poolt testimist ei olnud võimalik läbi viia ehk sihtrühma kasutajaliidese rahulolu ei olnud võimalik hinnata.

MFN-3	Arendajatel peaks olema lihtne luua uus stsenaarium (ava/sule printsiip)
Täitmine	<ul style="list-style-type: none"> - Arendajatel on lihtne luua uusi stsenaariume. Testitud juhendaja Lech Bialeki poolt.

MFN-4	Simulatsiooni kood peab olema hallatav ja üle võetav uute arendajate poolt
Täitmine	<ul style="list-style-type: none"> - Simulatsiooni koodile loodi ülevõtmise manual (Lisa 6), mis võimaldab uutel arendajatel koodi ülevõtmise.

MFN-5	Simulatsiooni võiks olla mängitav Oculus Quest riistvaraga
Täitmine	<ul style="list-style-type: none"> - Simulatsioon ei ole mängitav Oculus Quest riistvaraga, sest arenduse ajal oli võimalik testida simulatsiooni ainult Oculus Go riistvaral.

MFN-6	Simulatsioon peaks madalate näitajatega süsteemis säilitama umbes 90 kaadrit sekundis oleku, et vältida merehaiguse teket ja tagada võimalikult realistlik kogemus
Täitmine	<ul style="list-style-type: none"> - Simulatsioon on võimalik madalate näitajatega süsteemis säilitama umbes 90 kaadrit sekundis oleku. - Selle võimekust testis meeskond, kliendid ja juhendaja.

Arvestades mitmete projekti käigus esile kerkinud probleemidega jäi klient lõpptootelega rahule. Meeskond rahuldab 80% kliendi poolt esitatud nõuetest. Ruumi jäi ka mitmeteks edasiarendusteks, mis tuuakse välja järgmises peatükis.

4.3 Edasiarendusvõimalused

Koroonaviirusest tingitud probleemide, aja- ja ressursipiirangute tõttu jäi tarnitavas tootes ruumi veel mitmeteks edasiarendusvõimalusteks. Järgnevates lõikudes on vastavaid soovitusi täpsemalt kirjeldatud.

Implementeerida “Infrapunakütte” stsenaarium simulatsiooni - projekti meeskond viis läbi küll kvalitatiivse uuringu infrapunakütte omadustest ja eelistest, disainis vastava stsenaariumi, mis sisaldas tervisliku vananemise ja energiatõhususe aspekte, sisestas simulatsiooni 3D mudelid infrapuna küttepaneelidest, aga meeskond ei jõudnud välja töötatud stsenaariumi implementeerimiseni simulatsiooni.

Simulatsiooni testimine sihtrühma hulgas - COVID-19 piirangute tõttu ei olnud võimalik arendatud toodet testida eakate hulgas, et saada vajalikku kogemustel põhinevat tagasisidet parandusteks. Toote jätkuarenduseks on see kindlasti vajalik, sest toode peab olema vastuvõetav sihtrühmale ja disainitud vastavalt sihtrühma nõudmistele.

Simulatsiooni virtuaalreaalsuse riistvaraks Oculus Quest - praegune toode on arendatud Oculus Go riistvaral, aga tulevikus võiks simulatsioon olla läbi mängitav Oculus Quest riistvaral. Questil jooksvatav programm annab kasutajale oma paremate tehniliste näitajate tõttu realistlikuma ja sujuvama virtuaalreaalsuse kogemuse (Punkt 2.3.3).

Stsenaariumid peaksid sisaldama turvalisuse ja privaatsuse aspekte - eakate seas läbi viidud uuringus selgus, et turvalisus ja privaatsus on suurimad murekohad Targa maja kasutuselevõtu juures. Praeguses simulatsioonis neid kahte märksõna välja ei tooda, aga tulevikus peaks neid aspekte stsenaariumite disaini juures kindlasti arvesse võtma.

Simulatsioon koduautomaatika seadmete ja süsteemide lisamine - turul on veel mitmeid koduautomaatika seadmeid ja süsteeme, mis teevad kodus elamise energiatõhusamaks ja soodustavad tervislikku vananemist, aga mida projektis arendatud simulatsioonis veel rakendada ei jõutud. Nendeks näiteks võivad olla kodudoktori süsteemi olemasolu, targa maja tugi hädaolukorras, automaatselt avanuvad ukSED, robottolmuimeja ja -niiduk, sensorid

aias põllul (mulla niiskus, temperatuur, mulla Ph tase) jne. Simulatsiooni saab lisada veel stsenaariumeid lisaks “Pimedas vannituppa” ja “Infrapunaküte” stsenaariumitele.

Kasutajaliides võiks olla võimalikult puhas, et vältida segadust kasutajas - arendajate soovitusel võiks kõik mängus sisselülitatavas funktsioonid olla aktiiveeritavad läbi simulatsiooni nutitelefonil. Samuti kõik tekstid menüüs ja stsenaariumis peaksid olema kergesti loetavad ja ette loetud *audiona*.

Projekti “Tulevikukodu” meeskond tarnis kliendile lõpptootena virtuaalreaalsuse simulatsiooni Targast majast, mis sisaldab tervisliku vananemise ja energiatõhususe aspekte, kuid koroonaviirusest tingitud probleemide tõttu jäi 20% kliendi esitatud nõuetest täitmata. Seetõttu on projektil mitmeid edasiarendusvõimalusi, mida saab arvesse võtta järgnevate arenduste juures.

Kokkuvõte

Käesoleva bakalaureusetöö eesmärgiks oli teostada projekti “Tulevikukodu” arendus ja juhtimine Scrum meetodil COVID-19 pandeemia ajal.

Lõputöö andis ülevaate peamistes agiilsete projektijuhtimismeetoditest nagu ekstreemprogrammeerimine, Kanban ja Scrum, millest meeskond valis antud projekti arenduseks välja Scrum meetodi. Meetod sisaldab kindlaid rolle, kaasab pidevalt klienti protsessi ja on üles ehitatud detailse elutsükliks, mis teeb selle varasema kogemusega meeskonna jaoks esmakordse rakendamise puhul lihtsasti omandatavaks.

Projekti tellinud klientide nõuded kaardistati projekti algul intervjuu teel ning kirjeldati funktsionaalsete ja mittefunktsionaalsete nõuetena. Nõuete teostus ja projekti arendus toimus Scrumi sprintidena. Sprinte oli viis ehk projekti käigus esitati kliendile toodet viiel korral. Lõpptarnena valmis Tarka maja imiteeriv simulatsioon, kuhu algselt planeeritud kahe stsenaariumi asemel jõuti implementeerida üks “Pimedas vannituppa” stsenaarium. Simulatsioon tõi välja koduautomaatika seadmete/süsteemide energiaefektiivsuse ja abistava omaduse vanurite seas esinevate terviseriskide vältimiseks. Projekti nõuete täieliku täitmise peamiseks takistuseks sai projekti algfaasis Euroopasse saabunud koroonaviirus, mis keelas ära kontaktkohtumised ehk meeskond pidi kogu kommunikatsiooni üle viima veebi. Samuti astusid kaks projekti tellinud klienti tagasi, kaks meeskonnaliiget pidi naasma kodumaale ning toodet ei saadud testida sihtrühma peal. Uutest tingimustest tulenevalt pidi meeskond pidevalt kohanema ootamatute olukordadega. See omakorda vähendas töögrupi ajaressurssi, vähendas meeskonna motivatsiooni ja langetas kliendi ootusi lõpptoote suhtes. Tänu eeskujulikule Scrum meetodi rakendamisele ja järgimisele, tarnis meeskond aga peale viit kuud arendust klientidele rahuldava toote, millele algsetest nõuetest oli täidetud 80%.

Selle bakalaureusetöö raames valmis virtuaalreaalsusel põhinev simulatsioon Targast majast, mis sisaldab tervisliku vananemise ja energiatõhususe aspekte. Simulatsiooni arendas neljaliikmeline meeskond Scrum meetodil. Töös kirjeldati agiilse projektijuhtimise tuntumad meetodid (k.a Scrum), anti ülevaade projekti probleemist, arenduses ja Scrum meetodi rakendamisest. Koroonaviiruse tõttu esile kerkinud probleemide tõttu on valminud tootel mitmeid edasiarendusvõimalusi.

Viidatud kirjandus

- [1] Wikipedia Projektijuhtimine <https://et.wikipedia.org/wiki/Projektijuhtimine#Projekt> (14.07.2020)
- [2] Andmekaitse ja infoturbe leksikon <https://akit.cyber.ee/term/2148>
- [3] Beedle M., Bennekum A., Cockburn A., Cunningham W., Fowler M., Highsmith J., Hunt, A., Jeffries R., Kern J., Marick B., Martin R., Schwaber K., Sutherland J., Thomas D. Manifesto for Agile Software Development, 2001. https://moodle2019-20.ua.es/moodle/pluginfile.php/2213/mod_resource/content/2/agile-manifesto.pdf (07.08.2020)
- [4] Half of Dutch adults will be over 50 in 2019, CBS, 2014. <https://www.cbs.nl/en-gb/news/2014/38/half-of-dutch-adults-will-be-over-50-in-2019> (16.07.2020)
- [5] Wysocki R. Effective Project Management - Traditional, Agile, Extreme. Ameerika Ühendriigid: John Wiley & Sons. 2014. <http://1.droppdf.com/files/vcAsT/wiley-effective-project-management-traditional-agile-extreme-7th-edition-2014.pdf>
- [6] Orav I. Projektijuhtimine. Loov Eesti. <https://www.looveesti.ee/arenda-ettevotet/juhtimine/projektijuhtimine/> (16.07.2020)
- [7] Kiil K. Agiilsete metoodikate rakendamine andmeaida arenduses finantsorganisatsiooni regulatiivse raporteerimise osakonna rüpi ja Baltimaade meeskonna näitel. Tallinna Tehnikaülikool infotehnoloogia teaduskonna magistritöö. 2019. [agiilsete metoodikate rakendamine andmeaida arenduses](#)
- [8] Sergeev A. A brief history of agile methodology. Hygger, 2016. <https://hygger.io/blog/a-brief-history-of-agile-methodology/> (17.07.2020)
- [9] Abrahamsson P., Salo O., Ronkainen J., Warsta J. Agile software development methods: Review and analysis. Finland: VTT. 2002. <https://arxiv.org/ftp/arxiv/papers/1709/1709.08439.pdf>
- [10] Cobb G.C. Making sense of agile project management: balancing control and agility. Kanada: John Wiley & Sons. 2011.

<https://www.pdfdrive.com/making-sense-of-agile-project-management-balancing-control-and-agility-d156665069.html>

[11] Wells D. The values of extreme programming. *Extreme Programming*, 2009. <http://www.extremeprogramming.org/values.html> (20.07.2020)

[12] Extreme programming. *Agile Alliance*. <https://www.agilealliance.org/glossary/xp/#q=~> (20.07.2020)

[13] Wells D. The rules of extreme programming. *Extreme Programming*, 1999. <http://www.extremeprogramming.org/rules.html>

[14] Gordon D. The extreme programming model. 2017. <https://www.slideshare.net/DamianGordon1/the-extreme-programming-xp-model> (20.07.2020)

[15] Rahman N.A.A., Sharif S.M., Esa M.M. Lean manufacturing case study with Kanban system implementation. 2013. <https://www.sciencedirect.com/science/article/pii/S2212567113002323>

[16] Lei H., Ganjeizadeh F., Jayachandran P.K., Ozcan P. A statistical analysis of the effects of Scrum and Kanban in software development projects. 2017. https://www.sciencedirect.com/science/article/pii/S0736584515301599?casa_token=fo5bVHYuI3QAAAAA:4viL83CoFEvdmhQry2UdQ8jwr2k9D-_zluZ20xw62Ylc9TJN_IzgI4yfSMpjBl7wKDcFNlqRQ

[17] What is Kanban? *Kanbanize*. <https://kanbanize.com/kanban-resources/getting-started/what-is-kanban> (20.07.2020)

[18] What is Kanban board? *Kanbanize*. <https://kanbanize.com/kanban-resources/getting-started/what-is-kanban-board> (20.07.2020)

[19] <https://www.sciencedirect.com/science/article/abs/pii/S0736584515301599>

[20] Krishnamurthy V. brief history of Scrum. *Techwell*, 2012. <https://www.techwell.com/techwell-insights/2012/10/brief-history-scrum> (20.07.2020)

[21] Scrum values. *Scrum Alliance*. <https://www.scrumalliance.org/about-scrum/values> (20.07.2020)

[22] What is a Scrum Board and how can you use it best? *Zoho Sprints*. <https://www.zoho.com/sprints/what-is-a-scrum-board.html> (20.07.2020)

[23] TechBlue and scaled agile framework (SAFe). *TechBlue*, 2019. <http://www.techblue.com/?p=3319> (21.07.2020)

- [24] Ionel N. Critical analysis of the Scrum project management methodology. *Majandus Akadeemia* Bukarestis. lk 435-441. <http://steconomiceuoradea.ro/anale/volume/2008/v4-management-marketing/077.pdf>
- [25] Ageing and health. *World Health Organization*, 2018. <https://www.who.int/news-room/fact-sheets/detail/ageing-and-health> (22.07.2020)
- [26] Fleming S. Cash for babies: How Europe is tackling its falling birthrate. *World Economic Forum*, 2020. <https://www.weforum.org/agenda/2020/02/europe-ageing-population-migration-birthrate/> (22.07.2020)
- [27] Jong B.M., Wynia K., Geluk-Bleumink A. Ageing better in the Netherlands. *Intechopen*, 2017. <https://www.intechopen.com/books/gerontology/ageing-better-in-the-netherlands> (22.07.2020)
- [28] Informatie gemeente Groningen: Bekijk informatie over meer dan 100 onderwerpen voor de gemeente Groningen. *Alle Cijfers*. <https://allecijfers.nl/gemeente/groningen/> (22.07.2020)
- [29] Company profile. *BAM kodulehekülg*. <https://www.bam.com/en/about-bam> (22.07.2020)
- [30] Business principles. *BAM kodulehekülg*. <https://www.bam.com/en/about-bam/business-principles> (22.07.2020)
- [31] Everyone Connected: KPN Integrated annual report. *KPN kodulehekülg*, 2017. https://web.archive.org/web/20190126001013/https://jaarverslag2017.kpn/app/uploads/KPN_IR-2017_Spread_navigation.pdf
- [32] Outsourcing & managed services. *Cegeka*. <https://www.cegeka.com/cs/in-close-cooperation/outsourcing-and-managed-services> (23.07.2020)
- [33] About EnTranCe. *EnTranCe kodulehekülg*. <https://www.en-tran-ce.org/en/over-entrance/> (23.07.2020)
- [34] Trello. *Wikipedia*, 2020. <https://en.wikipedia.org/wiki/Trello> (23.07.2020)
- [35] Oculus VR. *Wikipedia*, 2020. https://en.wikipedia.org/wiki/Oculus_VR (23.07.2020)
- [36] Davenport C. Oculus discontinues the Gear VR-based Go headset. *Androidpolice*, 2020. <https://www.androidpolice.com/tags/vr/>
- [37] Oculus Go vs Oculus Quest. *Versus*. <https://versus.com/en/oculus-go-vs-oculus-quest> (23.07.2020)

- [38] Oculus Quest Png, Transparent Png. *KindPNG*.
https://www.kindpng.com/imgv/hbiTmTh_oculus-quest-png-transparent-png/ (23.07.2020)
- [39] GitHub kodulehekülg. <https://github.com/> (23.07.2020)
- [40] Uurimistöö alused: Uurimusliku tegevuse liigitus meetodi järgi. *Saaremaa Ühisgümnaasium*. https://www.syg.edu.ee/~peil/ut_alused/meetodid.html (25.07.2020)
- [41] Functional vs non-functional requirements: the definitive guide. *QRA kodulehekülg*, 2019. <https://qracorp.com/functional-vs-non-functional-requirements/> (25.07.2020)
- [42] 10 MoSCoW prioritisation. *Agile Business Consortium*, 2019. https://www.agilebusiness.org/page/ProjectFramework_10_MoSCoWPrioritisation (25.07.2020)
- [43] Global Health and Aging. *World Health Organization*, 2011. https://www.who.int/ageing/publications/global_health.pdf?ua (26.07.2020)
- [44] Falls. *World Health Organization*, 2018. <https://www.who.int/news-room/fact-sheets/detail/falls> (26.07.2020)
- [45] Falls among older adults in the EU-28: Key facts from the available statistics. *EuroSafe*, 2015. https://www.eurosafe.eu.com/uploads/inline-files/POLICY%20BRIEFING%2020_Facts%20on%20falls%20in%20older%20adults%20in%20EU.pdf (26.06.2020)
- [46] Policy briefing 14: Injuries among older people. European innovation partnership, 2010. http://82.196.12.102/library/policy-briefing-14-injuries-among-older-people_en (26.06.2020)
- [47] Alshammari S.a., Alhassan A.M., Aldawsari M.A., Bazuhair F.O., Alotaibi F.K., Aldakhil A.A., Abdulfattah F.W. Falls among elderlt and its relation with their health problems and surroundings environmental factors in Riyadh. *Journal of Family & Community Medicine*, 2018, 25(1), lk 29-34. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5774040/#ref11> (28.07.2020)
- [48] Energy Efficiency. *Environmental and Energy Study Institute*. <https://www.eesi.org/topics/energy-efficiency/description> (28.07.2020)
- [49] Ge M., Friedrich J., 4 charts explain greenhouse gas emission by countries and sectors. *World Resources Institute*, 2020. <https://www.wri.org/blog/2020/02/greenhouse-gas-emissions-by-country-sector> (28.07.2020)

- [50] Energy consumption in households. Eurostat Statistics Explained, 2020. https://ec.europa.eu/eurostat/statistics-explained/index.php/Energy_consumption_in_households (28.07.2020)
- [51] Byers C. IoT Actuators and Sensors. *Mouser Electronics*, 2019. <https://www.mouser.se/blog/iot-actuators-and-sensors> (29.07.2020)
- [52] Liikumisandur. Wikipedia, 2018. <https://et.wikipedia.org/wiki/Liikumisandur> (29.07.2020)
- [53] Wilson B. Introduction to physical electronics. Rice Ülikool, 2008. <http://www.freeinfosociety.com/media/pdf/2822.pdf>
- [54] Incandescent light bulb. *Energy education*. https://energyeducation.ca/encyclopedia/Incandescent_light_bulb (01.08.2020)
- [55] LED lamp. *Wikipedia*, 2020. https://en.wikipedia.org/wiki/LED_lamp#Energy_Star_qualification (01.08.2020)
- [56] Unintentional Injuries among the Elderly in Rural Areas and Their Related Behaviors. *Korean Journal of Family Medicine* 2019, 40(2), lk 80-86 <https://www.kjfm.or.kr/journal/view.php?doi=10.4082/kjfm.17.0124> (01.08.2020)
- [57] Figueiro M.G., Gras L.Z., Rea M.S., Plitnick B., Rea M.S. Lighting for improving balance in older adults with and without risk for falls. *Age and Ageing*, number 41, 2012, lk 392–395. <https://academic.oup.com/ageing/article/41/3/392/31467> (02.08.2020)
- [58] History of infrared heating panels. *The Renewable Energy Hub*. <https://www.renewableenergyhub.co.uk/main/infrared-heating-information/history-of-infrared-heating/> (02.08.2020)
- [59] Infrared Heating Panels – Quick And Effective Heating Technology. *Helvetic*, 2018. <https://helvetic.ie/infrared-heating-panels-quick-and-effective-heating-technology/> (02.08.2020)
- [60] Rant A. Küttesüsteemid. *Tallinna Tehnikaülikool*. https://old.taltech.ee/public/p/projektid/BuildEst/Kuttetesusteemid_Aivar_-Rant.pdf
- [61] The running costs of infrared heating. *The GreenAge*, 2015. <https://www.thegreenage.co.uk/the-running-costs-of-infrared-heating/> (02.08.2020)
- [62] Kütetehnoloogias. *Hea Soojus kodulehekülj*. <https://heasoojus.ee/kuttetehnoloogias/> (02.08.2020)
- [63] COVID-19. *Wikipedia*, 2020. <https://et.wikipedia.org/wiki/COVID-19> (02.08.2020)

Lisa 1. Kliendi intervjuu küsimused

Interviewers Willem Pepping, Hanna-Marii Kaljas, Piotr Pawlowski, Dante Baker

Date 03-03-2020

Location Groningen

Duration ± 45 minutes

Subject Requirement gathering for a VR game.

Respondent Klaas Van Den Berg (Municipality of Groningen), Albert Hofman (BAM),
Lech Bialek (EnTranCe), Wico Mulder (KPN)

Client

1. Tell us about yourself.
2. Tell us about your company.

Assignment

1. What is the reason for the assignment?
2. What is the problem that this project must solve?
3. Could you tell us specifically what the goal is of this project?
4. Who are the end users? (demography: age/sex/nationality/language/location/disabilities)
5. What are the deliverables?

Functional

1. Could you describe the software in detail?
2. What are the fundamental and crucial needs of this software?
3. What functionalities must this product have?
4. Is there something that you specifically want to avoid in the system?
5. What are your concerns about this type of system?
6. What do you NOT want to see in the system?

Scenarios

1. How many scenarios would you like to see in the product?
2. Are there any scenarios you would like to see added?
3. Are we free to think of and create scenarios?
4. What is your opinion on the current implementation of scenario ...

- a. Falling down stairs? (no animation/ambulance immediately available → not realistic)
 - b. “Healthy light”?
 - c. Bathroom at night (Not found in the game)?
5. Should one or more scenarios be removed or changed?

Personas

1. What is your opinion on the current three personas?
2. Do they fit the description of the target audience?
3. Should their disabilities affect gameplay? (It currently does not, the report says it does).

Energy

1. How do you envision the use of renewable energy in the simulation?
 - a. Energy consumption charts?
2. What do you think of the current implementation with the information points? (interacting with it will prompt a voice based information snippet about energy usage).

Mobile phone

1. What is your opinion on the mobile phone in the game? (The phone allows you to turn light on/off or open/close curtains).

Etc

1. Does this product need to use any online services? Should it function offline?
2. Should this product need to come with a user-manual? (vs. in-game tutorial)

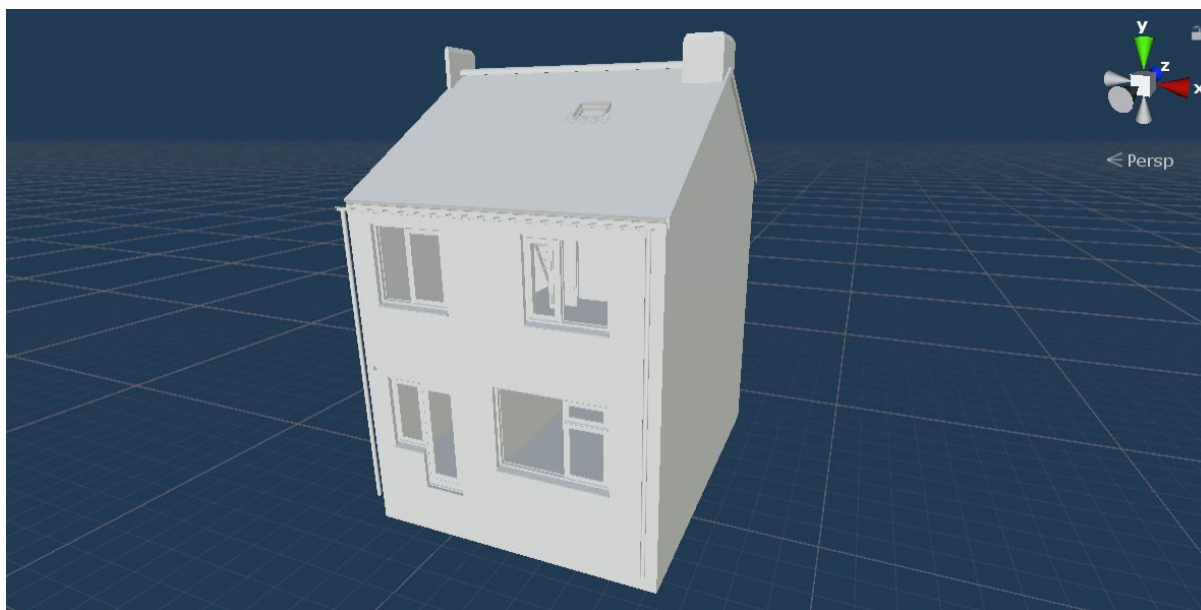
Non-functional

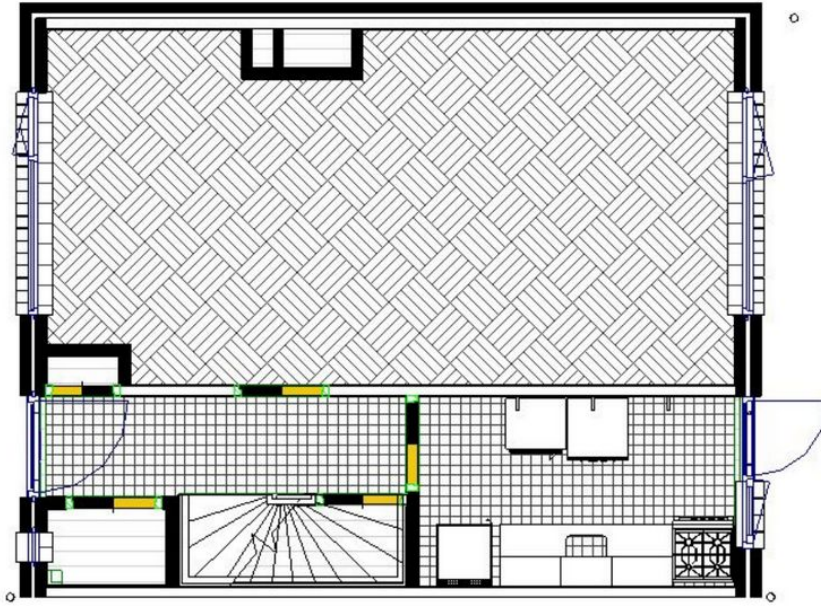
1. How important is performance? (speed/loading times)
2. How important is realism? (Bugs: walking through walls)

End

1. Who do we keep in contact with for further questions?

Lisa 2. Pildid BAM-i majast





Lisa 3. LED-tulede energia ja maksumuse efektiivsus

Energy and Cost Efficiency LED bulbs

Energy consumption:

1K lumens would require a 13-watt LED bulb. In addition, the hallway light is generally used mostly in the evenings. Moreover, depending on the approach to energy consumption this may amount to approximately 5 hours per day, however, during the winter it is generally assumed to be approximately 6 hours per day.

Therefore, using the data above, the annual cost of the LED bulb: (13 watts) x (6 hours per day) = 78-watt hours per day. $78 \text{ Wh/day} \div 1000$ (converting Watt hours to Kilowatt hours—kWh) = 0.078 kWh/day. **The hallway LED bulb would use approximately 28kWh/year** (0.078×365).

Lastly, the average cost of electricity per kWh in the Netherlands is approximately 21 Euro cents (Statista, 2020). Therefore, **the annual cost of using the LED bulb in the hallway would be €5.88** ($28\text{kWh/yr} \times 21$ Euro cents). In comparison, an Incandescent light bulb would require 75-watts—therefore, utilizing approximately 480% more energy to provide lighting for the same space/hallway (McBride, n.d.).

Cost efficiency:

Using the same method of calculation above, an incandescent light bulb would use $75 \times 6\text{h/day} \div 1000 \times 365 = 164\text{kwh/year}$. Therefore, **the total annual cost per an incandescent light bulb for the hallway would cost approximately €34**, which is approximately **500% more expensive** than an LED bulb on an annual basis.

Energy and Cost Efficiency LED stripes

Energy consumption:

Therefore, using the data above, the annual cost of the LED stripes: (15 watts) x (5 hours per day) = 75 watt hours per day for one meeter of LED stripes. $75 \text{ WH/day} \div 1000$ (converting it to kilowatt hours) = 0.075 KWH/day. The 1 meeter of LED stripe would use approximately

0.075*365=27.4 kwh/year. Incandescent light will use for 1500 lumens about 100W of energy (see table 5). That means $100W * 5h/day \div 1000 * 365 = 182.5$ kwh/year.

Cost efficiency:

The average cost per KWH in the Netherlands is 21 euro cents (the world average is 12 euro cents). **LED stripes $27.4*0.2\text{€} = 5.75\text{€/year}$ comparing to incandescent light bulbs $182.5*0.21=36.3\text{€/year}$.** That makes **incandescent light bulbs 6.6 times more expensive than LED stripes.**

Lisa 4. Infrapunakütte energia ja maksumuse efektiivsus

Energy and Cost Efficiency of infrared heating

Energy consumption:

Typically an infrared heater just 40% size of a traditional convection radiator heater will produce the same amount of energy. In other words a 1kW convection heater will produce as much energy as a 0.4kW infrared heater. Let's assume we use heating for the whole year and the average amount while it is on is 5 hours per day. Let us compare the energy use between Infrared Heater and Convection radiator heater.

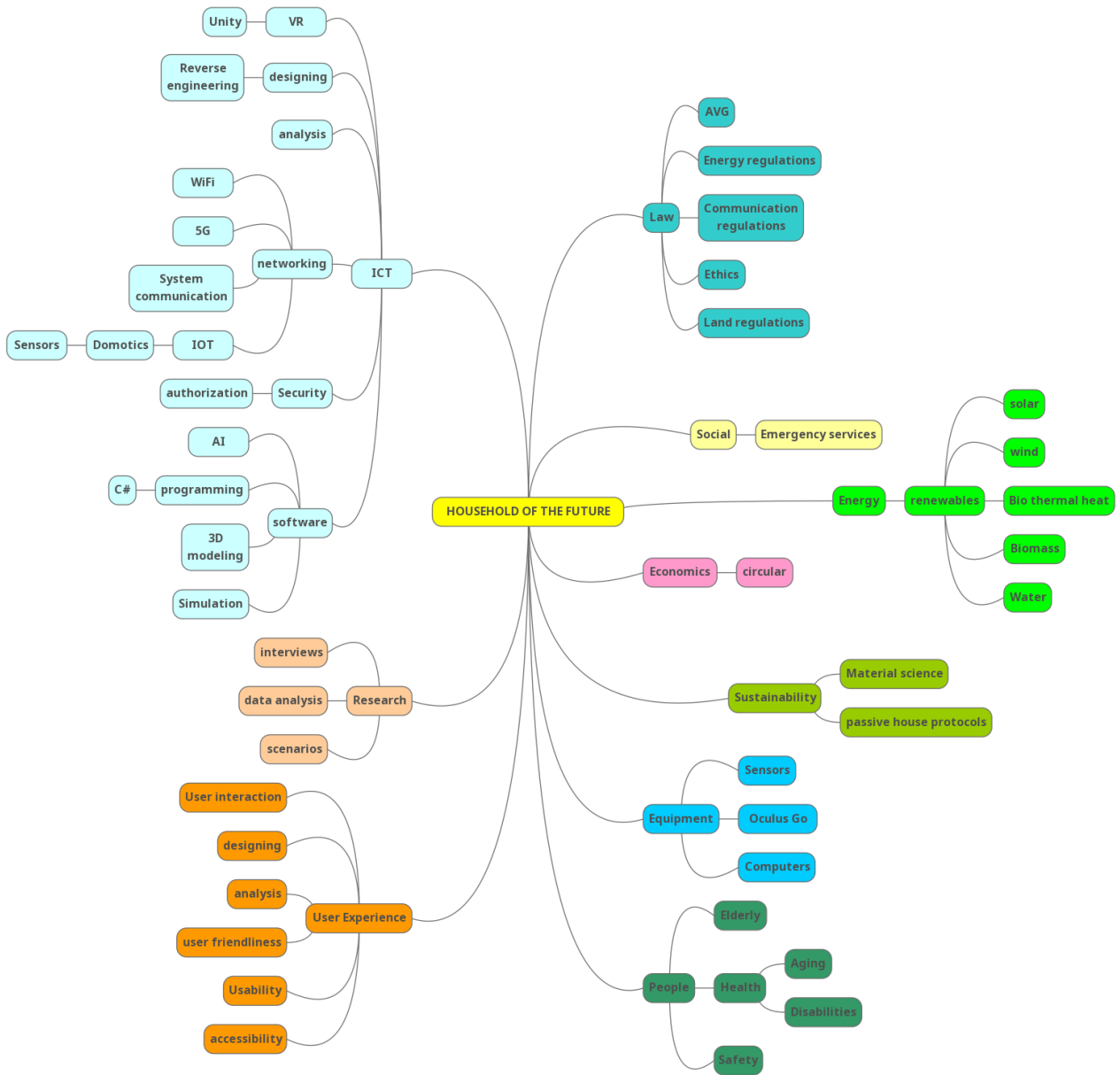
Therefore, using our assumptions we can calculate the annual amount of energy spent in both cases: Infrared heater: 400 watts x 5 (hours per day) = 2000 watts per day. It is 2 kWh x 365 = **730 kWh/year**. Convection heater: 1000 watts x 5 (hours per day) = 5000 watts per day. It takes 5 kWh x 365 = **1825 kWh/year** which is 2.5 times higher energy consumption.

Cost efficiency:

The average cost of kWh in the Netherlands is 0.22 euro cents. The annual costs are following: Infrared heater: 730 x €0.2 = **€146 / year** in comparison to convection heater: 1825 x €0.2 = **€365 / year**. This shows infrared heater can save approximately **€200** each year.

Talking about costs of installation, for basic model, the convection heater cost varies between €35 - €60, but €50 - €120 for infrared heater. This means installation costs for infrared heaters are a bit higher, but the difference is reduced while we take into account the running costs which are €200 per year lower in favour of infrared heaters.

Lisa 5. Projekti mōttekaart



Lisa 6. “Household of the future virtual reality” manual

1. Introduction

The manual will provide you with detailed information about the “House of the future VR” project. The certain steps must be taken to open and run the simulation. Also, you will see how to set up devices properly. In order to develop and work on the application you will be given the several tips.

2. Installation and run

This section describes the components required for this project and the general use.

2.1. Files

All files and source code are in Lech’s repository.

2.2. System specification

The minimum requirements for Unity.

2.2.1 Unity Editor systems requirements

This section lists all the minimum requirements to run the Unity Editor. The performance and rendering may actually vary depending on the complexity of the project.

Minimum requirements	Windows	macOS	Linux (Support in Preview)
Operating system version	Windows 7 (SP1+) and Windows 10, 64-bit versions only.	Sierra 10.12.6+	Ubuntu 16.04, Ubuntu 18.04, and CentOS 7
CPU	X64 architecture with SSE2 instruction set support	X64 architecture with SSE2 instruction set support	X64 architecture with SSE2 instruction set support
Graphics API	DX10, DX11, and DX12-capable GPUs	Metal-capable Intel and AMD GPUs	OpenGL 3.2+ or Vulkan-capable, Nvidia and AMD GPUs.
Additional requirements	Hardware vendor officially supported drivers	Apple officially supported drivers	Gnome desktop environment running on top of X11 windowing system, Nvidia official proprietary graphics driver or AMD Mesa graphics driver. Other configuration and user environment as provided stock with the supported distribution (Kernel, Compositor, etc.)
For all operating systems, the Unity Editor is supported on workstations or laptop form factors, running without emulation, container or compatibility layer.			

2.2.2 Unity Player systems requirements

This section lists all the minimum requirements to build and run the Unity Player. The performance and rendering may actually vary depending on the complexity of the project.

Operating system	Windows	Universal Windows Platform	macOS	Linux
Operating system version	Windows 7 (SP1+) and Windows 10	Xbox One, Hololens	Sierra 10.12+	Ubuntu 16.04 and Ubuntu 18.04
CPU	x86, x64 architecture with SSE2 instruction set support.	x86, x64 architecture with SSE2 instruction set support, ARM, ARM64.	x64 architecture with SSE2.	x64 architecture with SSE2 instruction set support.
Graphics API	DX10, DX11, DX12 capable.	DX10, DX11, DX12 capable GPUs.	Metal capable Intel and AMD GPUs	OpenGL 3.2+, Vulkan capable.
Additional requirements	Hardware vendor officially supported drivers.	Hardware vendor officially supported drivers. For development: Windows 10 (64-bit), Visual Studio 2015 with C++ Tools component or later and Windows 10 SDK.	Apple officially supported drivers.	Gnome desktop environment running on top of X11 windowing system Other configuration and user environment as provided stock with the supported distribution (such as Kernel or Compositor) Nvidia and AMD GPUs using Nvidia official proprietary graphics driver or AMD Mesa graphics driver.

2.3 Installation

To perform these steps you need a developer account Oculus. The link above has a number of distinct steps. The most important information is listed below.

<https://ez-360.com/knowledge-base/how-to-enable-developer-mode-oculus-go-quest/>

1. Make sure your phone has Bluetooth enabled.
2. Open the Oculus app on your mobile device.
3. In the Settings menu, select your Oculus Go headset that you want to use
4. Select More Settings.
5. Turn Developer Toggle Mode on. If you're asked to create an Oculus Organization, follow these steps:
 - a. On your phone or a PC, login to your account and navigate to:
<https://dashboard.oculus.com/>
 - b. Create a new organization
 - c. Enter your organization name and click submit.

d. Review and accept the terms.

6. Go back to your mobile device and follow the instructions to toggle from Oculus Developer Mode on.

2.3.1. Install APK files

Certain steps must be taken before starting the project. Firstly, we have to set up the Oculus Go. This can be done by placing 'bulid1.apk' file in the project files. The following link gives provides tutorial to do this properly:

<https://headjack.io/tutorial/sideload-install-app-apk-oculus-go-quest/>

2.3.2. Building project

To open and install the project on the Oculus Go you will have to build time within the project itself. To do this properly you will need following programs:

- Unity
- Android Debug Bridge (ADB)
- Android Studio

Then, follow these steps (steps about creating a project can be ignored):

<https://medium.com/inborn-experience/how-to-build-an-app-for-the-oculus-go-from-start-to-finish-with-unity-cb72d931ddae>

Tips:

- Pay attention about required versions.
- Make sure you set the correct Environment Variables.
- Remember to turn on the Oculus Go app in Developers Modes
- If a problem with build time occurs you will have to uninstall the app (type in Command Prompt: "adb uninstall com.GemeenteGroningen.HuisvanMorgenVR", this name may vary, it can be found in Oculus Go library)

2.4. Unity Version

Project was created in version 2019.3.8f1 and further developed on 2019.2.10f1. For next groups we recommend using the latest version to avoid problems with version control and keep everything updated.

1. Project development

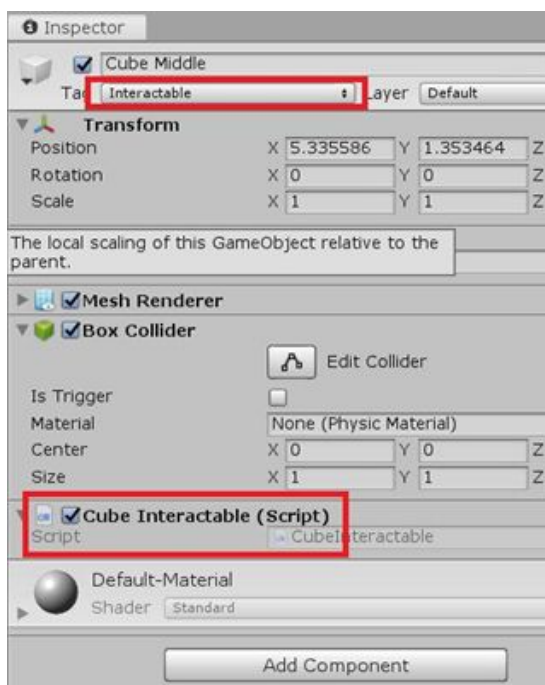
The first crucial element is to figure out how Unity works. If you have no experience, we recommend spending some time getting acquainted with Unity. To get a quick overview of the project you open Demoscene (assets/scenes/DemoScene.unity). You can see what objects are there.

3.1. Player and interaction with objects

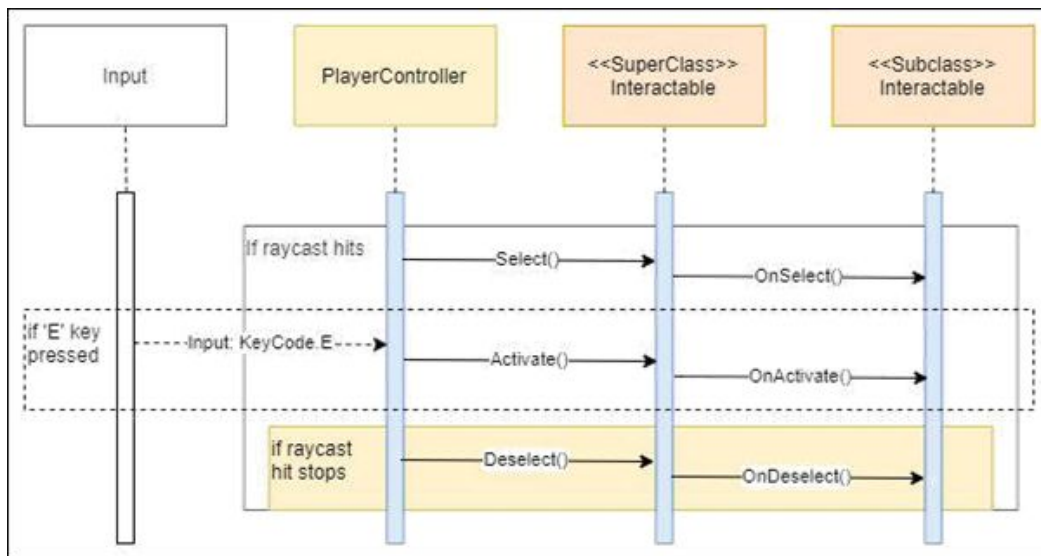
The Player Controller class is responsible for choosing and setting the correct player.

Player uses raycast in both cases, to determine where you are looking at or to point to the controller. If you want to make an object interactively, you use the Interactable class. Follow these two actions:

- Put the tag “Interactable”
- Add “Interactable” component to the object:



If “Interactable” is selected during playing, you can select it in-game using ‘e’ button in editor.

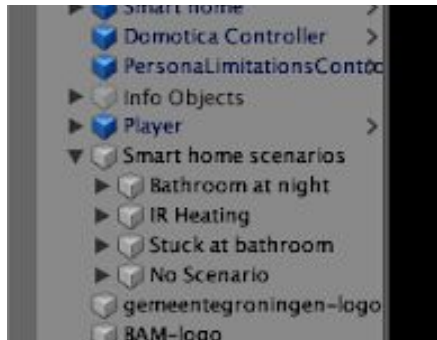


The above sequence diagram shows the communication between the various classes. It is important that the Select(), Activate() and Deselect() functions belong to the superclass. These are constantly being invoked. Then, the onSelect(), OnActivate() or OnDeSelect () function is called from the subclass. This ensures that for example the outline of the Select() function can be turned on. It should be done in all Interactable types. “On-functions" can then be added to the specific functionalities, this makes it scalable. The interaction between the Input and Player Controller is fairly abstract held in this diagram. Here it is chosen because this interaction is not relevant to the interactables. In reality the Player Controller controls or the "E" key is pressed. If something is selected, it is activated.

3.2. Creating a new scenario

In order to implement a new scenario in the game, adjustments must be made to both the smart home scene and the regular home scene. The following steps are the same for both scenes, but the content should not be identical.

1. Open one of the home scenes. In the following image, we view the contents of the smart home scenarios.

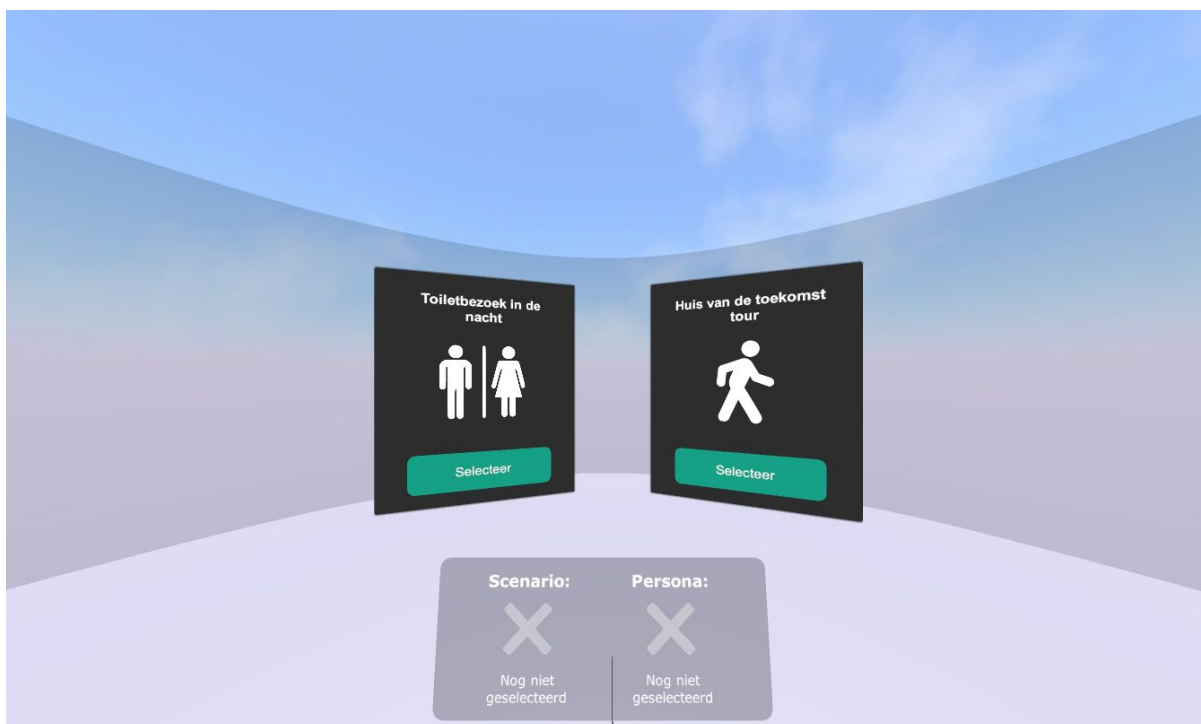


2. To add a new scenario, right click on the Smart home scenarios object and create a new empty object.
3. Assign the Scenario script to this object.
4. Create a Starting point object, an Intro object and an Outro object.
5. Create the steps of the scenario.
6. Assign all objects to the scenario script of step 3.
7. Finally, add the Scenario object to the Scenario Controller object.

The scenario is now testable using the Debug Mode and Debug Active Scenario settings in the Scenario Controller object. Setting the Debug Mode will allow the developer to test the scenario without going through the home menu first. The Debug Active Scenario should contain the index of the scenario to test, based on the index in the same script, in case Debug Mode is activated.

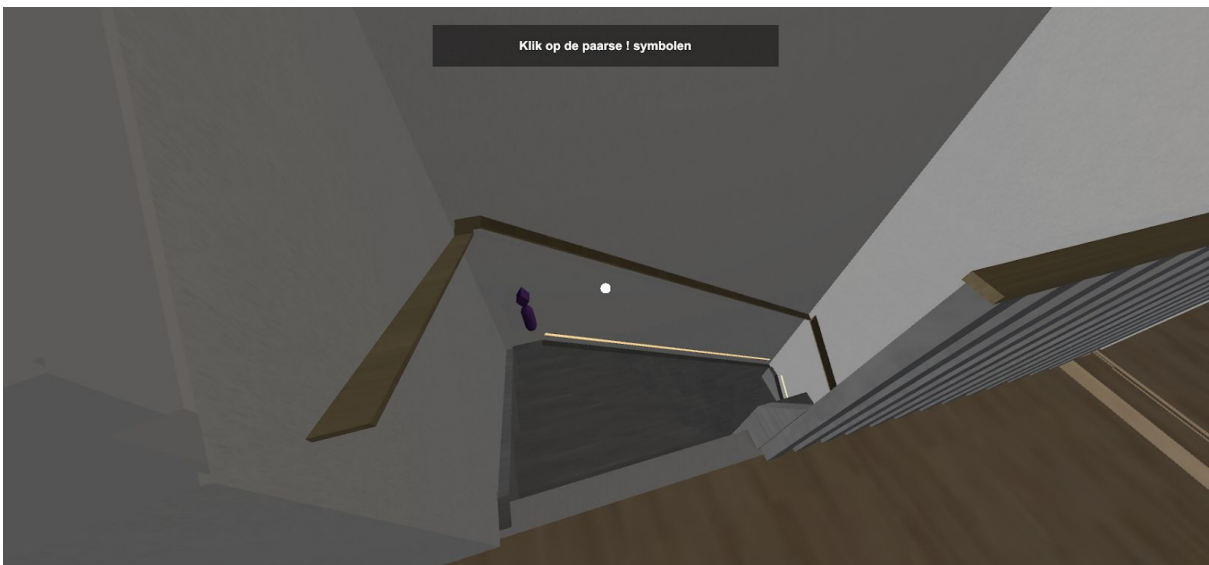
To play the scenario from the menu, a new concrete ScenarioModel class must be created and added to the repository class. See the folder Scenario > Concrete scenarios for examples.

Lisa 7. Menüü ja karakterite valik simulatsioonis



Lisa 8. LED-tulede ja infrapuna küttepaneelid simulatsioonis





Lisa 9. Automatiseeritid LED-tulede paigaldamise maksumus BAMi poolt



BAM Bouw en Techniek – Noord(grn)
Algemene projectgegevens
ELEKTROTECHNISCHE INSTALLATIES

UTIL

Project naam	Huis van Morgen
Plaats project	Groningen
Betreft	Elektrotechnische installatie

Project nummer	Huis van Morgen		
Calculator	J. Peterson		
Datum begroting	8 juni 2020		
Datum opgeslagen	08 juni 2020	Tijd	13:28:32

Opdrachtgever	Huis van Morgen
Projectleiding	J.Peterson

LOOPTIJD

Aanvang (opdracht)	ntb
Oplevering project	ntb

Opmerkingen:	

Proj. Nr.	<i>Huis van Morgen</i>			
Proj. Naam	<i>Huis van Morgen</i>			
Betreft	<i>Elektrotechnische installatie</i>		totale	totaal
Aantal	Ehd	Omschrijving	montage	netto mat.
			(m.u.)	
1: MONTAGE:				
<u>Uitgangspunten:</u>				
- Het leveren en aanbrengen van LED strip 20+12 meter				
- Het leveren en plaatsen van 3 bewegingsmelders				
- Gebruik maken van een bestaande e-voeding				
- Verlichting geschakeld via aanwezigheidsmelders				
Leidingwerk / bekabeling				
25 st		(2850207004) D11271NN DON YMK DCA D GY5G2,5 MTR		58
20 st		(2700110341) POLVALIT VSV PIP SLAGV.BUIS VSV LF 19MM 4M		21
200 st		(2700212007) EC19 DONKERGRIJS SCH EURO-CLIP 3/4" DONKERGRYS		52
4 st		Lasdozen		50
Ledstrip + toebehoren				
32 st		(2700456343) 2.01.0625 LMP ALU LED HOEKPROF 16MM		255
32 st		(2700450998) 2.01.0619 LMP AFSCHERMING OPAAL		255
32 st		(2700456345) 2.01.0628 LMP EINDKAP GESLOTEN		36
7 st		(2850387975) 2.21.1154 LMP LEDSTRIP EXTRO 24V ROL 5M		347
4 st		(2700283728) SP-200-24 MW PSU N CLOSED 24V/8.4A		249
1 st		Montage		
Algemeen				
3 st		Aanwijsmelders		255
1 st		Klein- en bevestigingsmateriaal		65
1 st		Montage	32,00	
			TOTAAL	32,00
				1 644
Totaal materiaal:			€	1 644

Project nummer
Project naam
Betreft

Huis van Morgen
Huis van Morgen
Elektrotechnische installatie

8-jun-2020

Eindcalculatie UTILITEIT

Netto materiaal			€	1 644
Toeslag materiaal	10% van €	1 644	€	164
				Bruto materiaal € 1 808
<u>Arbeid: tarief code 0 t/m 19</u> <i>Elektrotechnische montage</i>		32 mu x factor 1,00 x bpl 1,00 x € 62,25	€	1 992
				Subtotaal (1) € 3 800
Projectbegeleiding			€	525
Winst & Risico		5,00 %	€	190
				Totaal (excl. BTW) € 4 515

Verkoop : €

excl. BTW

Accoord :

Lisa 10. “Tulevikukodus” lõpptarne kliendile

Valminud simulatsiooni “Pimedas vannituppa” stsenaariumi ja Targas maja vabalt ringi kõndimise režiimi läbimängu videod on kättesaadavad järgmistelt linkidelt:

- **“Pimedas vannituppa” stsenaarium:**

<https://drive.google.com/file/d/1TxoNAfXOKjs5qQtEUB3pSyR3DbabZszH/view?usp=sharing>

- **Targas maja vabalt ringi kõndimise režiim:**

<https://drive.google.com/file/d/1Md8dy37IKFEh7roudAL6htktNC59O3e2/view?usp=sharing>

Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Hanna-Marii Kaljas,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose

Projekti “Household of the future virtual reality” arendus ja juhtimine Scrum meetodil COVID-19 pandeemia ajal,

mille juhendaja on Anne Villems,

reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.
3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.
4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Hanna-Marii Kaljas

10.08.2020