

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Anastasiia Oprysk 206823IABB

Marija Talašmanova 206494IABB

# **GovTech'i ettevõtete veebikataloogi arendus**

Bakalaureusetöö

Juhendaja: Bahdan Yanovich  
BSc

Kaasjuhendaja: Alena Labanova  
MSc

Tallinn 2024

## **Autorideklaratsioon**

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

Autorid: Anastasiia Oprysk, Marija Talašmanova

20.05.2024

## **Annotatsioon**

Bakalaureusetöö eesmärk on arendada veebirakendus-kataloog GovTech'i sektoris tegutsevatele ettevõtete jaoks.

Platvorm on suunatud kasuliku ja asjakohase teabe pakkumisele ettevõtete kohta. See võimaldab tavakasutajal paremini navigeerida GovTech'i valdkonnas ning avab uusi võimalusi analüütiliste uuringute ja statistika jälgimiseks erinevates valdkondades. See võib tugevdada suhtlust ettevõtete ja nende potentsiaalsete klientide, partnerite, investorite vahel ning on samuti oluline panus teadusmaailma.

Tänapäeval on selles valdkonnas olemas valmislahendused, millel on aga kas piiratud funktsionaalsus või ebamugav liides ning mis ei hõlma kõiki riike. Bakalaureusetöö autorid arvestasid olemasolevate platvormide puudustega, luues mugava veebirakenduse, pöörates erilist tähelepanu liidese visuaalsele atraktiivsusele, muutes selle mitte ainult funktsionaalseks, vaid ka visuaalselt nauditavaks.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 57 leheküljel, 5 peatükki, 40 joonist, 3 tabelit.

## **Abstract**

### **Development of a web catalogue for GovTech companies**

The aim of the bachelor thesis is to develop an online application-catalogue for companies operating in the GovTech sector.

The platform is aimed to provide useful and relevant information about companies. It allows an average user to better navigate the GovTech field and provides new opportunities for tracking analytical research and statistics in related fields. It can strengthen communication between companies and their potential customers, partners, investors and is also an important contribution to the scientific world.

There are existing solutions in this field, but they either have limited functionality or an inconvenient interface which does not cover all countries. The authors of the bachelor thesis took into account the disadvantages of the existing platforms, creating a convenient web application, paying special attention to the visual appeal of the interface, making it not only functional, but also picturesque.

The thesis is in Estonian language and contains 57 pages of text, 5 chapters, 40 figures, 3 tables.

## Lühendite ja mõistete sõnastik

GovTech	Government Technology ehk valitsuse tehnoloogia
XP	Extreme Programming ehk ekstreemprogrammeerimine
IKT	Info- ja kommunikatsioonitehnoloogia
SDG	Sustainable Development Goals ehk säästva arengu eesmärgid
PK	Primary key ehk primaarvõti
FK	Foreign key ehk välisvõti
MVC	Model-View-Controller ehk mudel-vaade-kontroller, tarkvara arhitektuurimuster
CRUD	Create, read, update, delete ehk loo, loe, uuenda, kustuta, neli olulist toimingut püsivate andmeelementide loomiseks ja haldamiseks
HTTP	Hypertext Transfer Protocol ehk hüperteksti edastusprotokoll, protokoll teabe edastamiseks arvutivõrkudes
HTML	Hypertext Markup Language ehk hüperteksti märgistuskeel
SQL	Structured Query Language ehk struktuurpäringukeel
CSS	Cascading Style Sheets ehk kaskaadlaadistik
DevSecOps	Development, Security, Operations ehk arendus, turvalisus, operatsioonid
NPM	Node Package Manager, JavaScript'i programmeerimiskeele jaoks loodud paketi haldur
JSON	JavaScript Object Notation, JavaScript'il põhinev tekstipõhine andmevahetusformaat
API	Application Programming Interface ehk rakendustarkvara liides, tarkvaraprojektides määratluste, protokollide ja tööriistade komplekt rakendustarkvara ehitamiseks

# Sisukord

1 Sissejuhatus .....	12
1.1 Üldine taust.....	12
1.2 Probleem.....	13
1.3 Eesmärk .....	13
1.4 Töö edasine struktuur .....	14
2 Metoodika.....	15
2.1 Objekti detailne kirjeldus.....	15
2.2 Tööriistad.....	16
2.2.1 Arenduskeskkond .....	16
2.2.2 Serveripoolne liides ja andmebaas .....	16
2.2.3 Kasutajaliides .....	17
2.2.4 Projekti juurutamine .....	18
2.2.5 Projekti haldamine.....	18
2.2.6 Ajalogimine .....	19
2.2.7 Suhtlus .....	19
2.2.8 Muud tööriistad .....	19
2.3 Tööprotsessi kirjeldus.....	21
2.3.1 Kasutatud meetodikad .....	21
2.3.2 Rollid .....	22
2.3.3 Tegevused.....	22
3 Töö tulemused .....	24
3.1 Olemasolevad lahendused .....	24
3.1.1 IPS-X (Innovative Public Services Explorer) [34].....	24
3.1.2 Interoperable Europe (Joinup) [36] .....	26
3.1.3 Startup Estonia [38].....	28
3.1.4 Kokkuvõtte olemasolevatest lahendustest .....	29
3.2 Nõuded.....	30
3.2.1 Funktsionaalsed nõuded .....	30
3.2.2 Mittefunktsionaalsed nõuded.....	31
3.2.3 Kasutuslood .....	31
3.3 Arhitektuur.....	32

3.3.1 Andmebaasi arhitektuur.....	33
3.3.2 Tagarakenduse arhitektuur .....	34
3.3.2.1 Mudelid .....	34
3.3.2.2 Vaated.....	36
3.3.2.3 Kontrollerid .....	38
3.3.2.4 API Kontrollerid.....	40
3.3.3 Kasutajaliidese arhitektuur .....	40
3.4 Testimine .....	44
3.5 Rakenduse juurutamine .....	44
3.5.1 Docker .....	44
4 Analüüs ja järeldused.....	47
4.1 Tööriistade ja tehnoloogiate valik .....	47
4.1.1 Nõuded .....	48
4.1.2 Arhitektuur .....	49
4.1.2.1 Andmebaasi arhitektuur .....	49
4.1.2.2 Tagarakenduse arhitektuur .....	49
4.1.2.3 Kasutajaliidese arhitektuur.....	49
4.1.3 Testimine .....	50
4.2 Veebirakenduse vastavus nõuetele .....	51
4.2.1 Funktsionaalsetele nõuetele vastavus .....	51
4.2.2 Mittefunktsionaalsetele nõuetele vastavus .....	59
4.3 Võrdlus olemasolevate lahendustega.....	60
4.4 Hinnang projekti teostamise protsessi kohta .....	62
4.5 Probleemid.....	64
4.6 Edasiarendus .....	64
4.7 Ajalogid ja meeskonnaliikmete panus.....	65
4.7.1 Anastasiia ajalogide kokkuvõte.....	65
4.7.2 Marija ajalogide kokkuvõte .....	66
5 Kokkuvõte .....	68
Kasutatud kirjandus .....	69
Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks .....	73
Lisa 2 – Anastasiia Oprysk eneseanalüüs.....	74

Lisa 3 – Marija Talašmanova eneseanalüüs .....	76
Lisa 4 – Kaasjuhendaja Alena Labanova tagasiside .....	78



## Jooniste loetelu

Joonis 1. Projekti arendamine Miro's .....	20
Joonis 2. IPS-X Cases.....	25
Joonis 3. IPS-X Statistics .....	26
Joonis 4. Interoperable Europe. Catalogue of GovTech initiatives.....	27
Joonis 5. Interoperable Europe. Kaart infoga.....	28
Joonis 6. Startup Estonia .....	29
Joonis 7. Startup Estonia. Ettevõtte info .....	29
Joonis 8. Kasutuslood.....	32
Joonis 9. Üldine arhitektuur.....	32
Joonis 10. Olemi-suhte diagramm.....	34
Joonis 11. Mudelid .....	35
Joonis 12. Kood Solution mudelist.....	36
Joonis 13. Vaated.....	36
Joonis 14. Kood Solution vaadest .....	38
Joonis 15. Kontrollerid .....	38
Joonis 16. Kood SolutionsController kontrollerist.....	40
Joonis 17. Kasutajaliidese arhitektuur .....	41
Joonis 18. CompanyGrid komponent .....	41
Joonis 19. Home lehe koodinäide.....	42
Joonis 20. Router funktsioon.....	43
Joonis 21. Külgriba välimuse kood .....	43
Joonis 22. Tagarakenduse Dockerfile.....	45
Joonis 23. Kasutajaliidese Dockerfile .....	46
Joonis 24. Docker'i konteiner.....	46
Joonis 25. Administraatori vaade sisselogimisel.....	51
Joonis 26. Ettevõtte loomise lehe vaade administraatori jaoks .....	52
Joonis 27. Ettevõtte redigeerimislehe vaade administraatori jaoks.....	53
Joonis 28. Ettevõtte kustutamise lehe vaade administraatori jaoks.....	54
Joonis 29. Ettevõtte nimekirja vaade administraatori jaoks .....	54
Joonis 30. Ettevõtte infolehe vaade administraatori jaoks .....	55
Joonis 31. Ettevõtete lehe vaade kasutaja jaoks .....	55

Joonis 32. Ettevõtte infolehe vaade kasutaja jaoks .....	55
Joonis 33. Ettevõtete filtreerimine kasutaja jaoks .....	56
Joonis 34. Lahenduste filtreerimine kasutaja jaoks .....	56
Joonis 35. Ettevõtte otsinguvõimalus kasutaja jaoks .....	56
Joonis 36. Lahenduse otsinguvõimalus kasutaja jaoks.....	57
Joonis 37. Statistika ettevõtete ja nende lahenduste kohta kasutaja jaoks.....	57
Joonis 38. Maailmakaardi vaade kasutaja jaoks.....	58
Joonis 39. Google'i vorm uute kandidaatide registreerimiseks kasutajate jaoks.....	58
Joonis 40. Meeskonna ajalogid.....	65

## **Tabelite loetelu**

Tabel 1. GovTech Catalogue võrdlus olemasolevate lahendustega .....	60
Tabel 2. Anastasiia ajalogide kokkuvõte .....	65
Tabel 3. Marija ajalogide kokkuvõte .....	66

# 1 Sissejuhatus

## 1.1 Üldine taust

Maailma liidrid tulid 2015. aastal kokku ja andsid ajaloolise lubaduse tagada kõigi inimeste õigused ja heaolu tervel ja õitsval planeedil, võttes säästva arengu tegevuskavas vastu 17 säästva arengu eesmärki aastani 2030 [1].

GovTech'i ettevõtted mängivad võtmerolli säästva arengu eesmärkide saavutamisel, arendades ja rakendades tehnoloogilisi lahendusi, mis viivad eesmärkide saavutamisele lähemale. Nende innovatsioonid aitavad kaasa parandustele erinevates valdkondades alates haridusest ja tervishoiust kuni majandusarenguni, mis koos võimaldavad saavutada ülemaailmseid eesmärke.

Valitsuse tehnoloogia (GovTech) kasvu kaasaegses maailmas seostatakse valitsusasutuste üleminekuga digikeskkonda ja suurenenud vajadusega kaasaegsete tehnoloogiliste lahenduste järele. See protsess stimuleerib GovTech'i ettevõtete arengut, mis on spetsialiseerunud uuenduslike tehnoloogiate arendamisele ja riigi klientidele digitaalteenuste pakkumisele. Kaasaegsete digitaaltehnoogiatega kasutuselevõtt võimaldab mitte ainult parandada kliendikogemust, vaid võib aidata ka valitsustel teenuseid kaasavamaks muuta [2].

GovTech'i ettevõtete roll avaliku sektori moderniseerimisel on hindamatu. Nad tutvustavad uuenduslikke tehnoloogiaid erinevates valdkondades, näiteks tervishoid, haridus ja transport, mis aitavad parandada pakutavate riigiteenuste tõhusust. GovTech'i ettevõtetel on võtmeroll valitsusasutuste kohandamisel digiajastu nõudmistega ja ülemineku võimaldamisel kaasaegsematele ning tõhusamatele tööviisidele [3].

Arvestades kõike ülaltoodut, on GovTech'i ettevõtete kataloogiga veebiplatvormi loomine eriti oluline. Selline platvorm võib saada väärtuslikuks ressursiks riigiasutustele, ettevõtjatele ja investoritele, pakkudes lihtsat ligipääsu infole valdkonna tehnoloogiliste lahenduste, arengute ning uuenduste kohta. Hoolimata sellest, et GovTech'i ettevõtete

kohta on juba olemas veebilehed infoga, ei ole siiski veel lahendust, mis koondaks kõikide riikide GovTech'i ettevõtete infot mugavas ja kasutajasõbralikus vormis.

## **1.2 Probleem**

Teave GovTech'i ettevõtete kohta on leitav erinevatel veebisaitidel, kuid see pole alati täielik ega hõlpsasti kasutatav. Puudub ühtne lahendus kataloogi näol, mis koondaks ülemaailmse teabe GovTech'i ettevõtetest arusaadavas ja kasutajasõbralikus formaadis, mistõttu on seda teemat uurivatel teadlastel andmete kogumine väga keeruline. Sellise platvormi puudumine muutub probleemiks valitsusasutuste, ettevõtjate ja investorite jaoks, kuna neil on raske hõlpsasti ligi pääseda ulatuslikule teabele GovTech'i ettevõtete kohta, mis piirab nende võimalusi teadlikkuse tõstmiseks ning tehnoloogiaprojektide arendamiseks.

## **1.3 Eesmärk**

Bakalaureusetöö eesmärgiks on arendada GovTech'i ettevõtete veebikataloog.

Veebirakendus võib salvestada teavet GovTech'i ettevõtete kohta üle maailma ja see teave on esitatud meeldivas ning kasutajasõbralikus liideses. Veebirakendus on avalikult kättesaadav ja sellele pääseb ligi lingi kaudu, mis võimaldab kõigil kogutud infole hõlpsasti ligi pääseda.

Veebirakenduse eesmärgiks on võimaldada järgmisi funktsioone:

- GovTech'i ettevõtete kohta info lisamine ja muutmine administraatori rollis;
- GovTech'i ettevõtete info vaatamine, filtreerimine kategooriate kaupa ja erinevaid andmeid sisaldavate statistiliste tabelitega tutvumine tavakasutaja rollis;
- maailmakaart riigi ettevõtete arvu vaatamise võimalusega;
- ettevõtte saab esitada oma kandidatuuri kataloogi lisamiseks.

## **1.4 Töö edasine struktuur**

Töö koosneb seitsmest osast. Sissejuhatuses kirjeldatakse töö üldist tausta, põhiprobleemi ja eesmärki. Metoodika osas kirjeldatakse nii objekti, kasutamiseks valitud tööriistu ja meetodeid kui ka ülesande täitmise protsessi. Tulemuste osa sisaldab tehnilist dokumentatsiooni ja ärianalüüsi. Analüüs annab ülevaate tehtud tööst ja annab hinnangu projekti erinevatele osadele. Kokkuvõttes esitatakse tulemused ja tehakse järeldused. Kasutatud kirjandus sisaldab töö kirjutamisel kasutatud allikaid. Lisad sisaldavad lihtlitsentsi, autorite eneseanalüüsi ja kaasjuhendaja tagasisidet.

## 2 Metoodika

Selles peatükis keskendutakse objektile ja töö käigus kasutatud tööriistadele ning kirjeldatakse kasutatud arendusprotsesse ja -põhimõtteid.

### 2.1 Objekti detailne kirjeldus

Lõputöö käigus töötati välja veebipõhine GovTech'i ettevõtete kataloogirakendus, mis on mõeldud selleks, et pakkuda nende ettevõtete kohta kasulikku ja väärtuslikku teavet, mis hõlmab erinevaid aspekte.

Rakenduse vajaliku funktsionaalsuse ning kasutajaliidese arendasid autorid koostöös lõputöö juhendaja ja kaasjuhendajaga. Ühistöö raames analüüsiti nii turul olemasolevate analoogide puudusi kui ka eeliseid. See aitas kindlaks teha, milliseid elemente saaks kohandada ja lisada arendatavasse veebirakendusse ning millised funktsioonid ei vastanud projekti nõuetele. Analüüs oli GovTech'i ettevõtete kasutajasõbraliku ja funktsionaalse kataloogi loomisel võtmetähtsusega..

Lõputöö käigus loodi rakendus, kasutades JetBrains Rider'i [4] integreeritud arenduskeskkonda tagarakendusena ja Visual Studio Code'i [5] kasutajaliidesena. Tagarakendus rakendati C# programmeerimiskeeles, kasutades ASP.NET Core MVC raamistikku [6], tagarakenduse arhitektuuri ülesehitamiseks kasutati MVC mustrit. See muster võimaldab ärioloogika, andmehalduse ja kasutajaliidese selget eraldamist, mis lihtsustab rakenduse arendamist ning hilisemat hooldust. Andmete säilitamiseks kasutati SQLite andmebaasi [7]. Esirakendust arendati JavaScript programmeerimiskeeles, kasutades React.js [8] raamistiku, mis muutis kasutajaliidese kaasaegseks ja hõlpsasti kasutatavaks.

Projekti tagarakendus, sealhulgas andmebaas, ja kasutajaliides juurutati edukalt Docker'i tarkvara [9] abil, mis tagas arendus- ja testimisprotsesside järjepidevuse ning rakenduse kõrge turvalisuse taseme.

Kasutatud lahenduste valikul lähtuti nii olemasolevatest kogemustest kui ka projekti tehnilistest nõuetest, seega rakendasid autorid mitte ainult kogutud teadmisi ja oskusi, vaid laiendasid oluliselt oma erialast silmaringi.

## **2.2 Tööriistad**

### **2.2.1 Arenduskeskkond**

Autorid kasutasid veebirakenduse tagarakenduse arendamiseks JetBrains Rider'i integreeritud tarkvaraarenduskeskkonda. Otsus kasutada JetBrains Rider'it põhines nii positiivsetel varasematel kogemustel selle arenduskeskkonnaga kui ka selle muljetavaldavatel tehnilistel omadustel, mis vastasid kõigile nõuetele.

Projekti kasutajaliidese arendamiseks otsustati kasutada Visual Studio Code'i lähtekoodiredaktorit, millega oli autoritel ka varasem kokkupuude. Lähtekoodiredaktor toetab arendamist React.js raamistikul ja vastavalt sellele ka JavaScript programmeerimiskeelele, mis oli kasutajaliidese loomiseks ideaalne.

### **2.2.2 Serveripoolne liides ja andmebaas**

GovTech'i sektori ettevõtetele mõeldud veebikataloogirakenduse loomise tehniline osa algas tagarakenduse arendamisega. Selleks valiti ASP.NET Core MVC raamistik ja rakendamine toimus C# programmeerimiskeeles.

Tagarakenduse arendus viitab veebirakenduse serveripoolse loomisele, mis on peamine vahend kasutajate päringute käsitlemiseks, andmebaasi ja muude süsteemidega suhtlemiseks, et tagada rakenduse toimimine. See rakenduse osa, mis hõlmab serverit ja andmebaasi, vastutab põhifunktsioonide täitmise eest serveris ning toiminguteks vajaliku loogika eest [10].

Otsuse kirjutada tagarakenduse kood C# programmeerimiskeeles langetasid autorid antud keele tundmise põhjal, mida kasutati aktiivselt kogu õppeprotsessi vältel. Lisaks oli oluliseks teguriks ka asjaolu, et C# on ideaalne veebirakenduste serveripoolse arendamiseks, pakkudes usaldusväarsust ja tõhusust vajaliku tagarakenduse funktsionaalsuse rakendamisel. Üksikute koodikomponentide puhul kasutati ka Razor'i



märgendussüntaksit, mis võimaldab C# keeles väljendeid arvutada ja neid HTML-väljundis visualiseerida.

Autorid kasutasid andmete säilitamiseks ja haldamiseks SQLite'i andmebaasi, mis on iseseisev ning serverivaba SQL-andmebaasi haldussüsteem, mis ei vaja seadistamist ja toetab tehinguid. Taolist teeki kasutatakse laialdaselt manustatud andmebaasina kohalike andmete säilitamiseks rakendustes nagu veebibrauserid, aga ka muudes rakendustes, kus on vaja kergelt manustatud andmebaasi. SQLite on populaarne oma lihtsa integreeritavuse ja usaldusväärsuse tõttu [11]. Arvestades kõiki selle süsteemi omadusi ja arendatava veebirakenduse nõudeid, jõudsid arendajad järeldusele, et SQLite vastab nõuetele ning on ülesande täitmiseks sobiv.

### **2.2.3 Kasutajaliides**

Projekti kasutajaliidese loomiseks valiti React.js raamistik ja programmeerimiskeel JavaScript, mis võimaldasid muuta kasutajaliidese dünaamiliseks ning paindlikuks.

Mugava, reageeriva ja visuaalselt meeldiva kasutajaliidese arendamine on projekti oluline osa, sest esiteks oli niisuguse liidese olemasolu kaasjuhendaja ühest nõudmistest ja teiseks suurendab see veebirakenduse vastu huvi ning võimaldab seda tulevikus edaspidi arendada vastavalt vajadustele.

Eesrakenduse tööriistade valimisel arvestasid autorid erinevate teguritega, näiteks nende populaarsuse, õppematerjalide olemasolu ja oma isikliku huviga. Valik seisnes Vue.js [12] ja React.js vahel ning lõpuks otsustasid autorid kasutada React.js raamistikku, kuna see osutus populaarsemaks kasutamiseks [13].

React raamistiku valik tähendas ka seda, et kasutajaliidese koodi kirjutamiseks kasutavad autorid programmeerimiskeelt JavaScript, mida kasutatakse veebirakenduse dünaamiliste ja interaktiivsete elementide loomiseks [14].

Samuti kasutavad autorid veebirakenduse erinevate visuaalsete efektide, näiteks font, värv ja teksti suurus, haldamiseks CSS'i võimalusi, mis võimaldavad veebirakenduse kasutajaliidest hõlpsasti ning ilma liigse koodita arendada [15].

#### **2.2.4 Projekti juurutamine**

Meeskond kasutas veebirakenduse tagarakenduse, integreeritud andmebaasi ja kasutajaliidese juurutamiseks Docker'i platvormi.

Docker on avatud lähtekoodiga tarkvaraplatvorm, mida kasutatakse ühisel operatsioonisüsteemil põhinevate virtualiseeritud rakenduskonteinerite loomiseks, kasutuselevõtuks ja haldamiseks ning sellega seotud tööriistade ökosüsteemi loomiseks [16].

Docker kiirendab ja optimeerib tarkvarakomponentide arendamise, testimise ning kasutuselevõtmise protsessi, võimaldades arendajatel töötada rakenduse eri osadega paralleelselt. Iga konteiner sisaldab kõiki vajalikke elemente komponendi ehitamiseks ja seejärel käivitamiseks. Docker tagab ka nende konteinerite teisaldatavuse erinevate serverite või keskkondade vahel [16].

Docker pakendab ja majutab konteinereid, muutes konteinerite tehnoloogia kättesaadavaks operatsioonisüsteemi kaudu. Iga konteiner sisaldab teenust või rakendusefunktsiooni koos kõigi vajalike teekide, konfiguratsioonifailide, sõltuvuste ja muude komponentidega. Kõik konteinerid kasutavad sama aluseks olevat operatsioonisüsteemi. Docker'i kujutised sisaldavad kõiki sõltuvusi, mis on vajalikud konteineri sees oleva koodi täitmiseks ning tagavad, et konteinerid töötavad sujuvalt, kui liigutakse erinevate Docker'i keskkondade vahel sama operatsioonisüsteemiga [16].

#### **2.2.5 Projekti haldamine**

Autorid kasutasid GitLab platvormi [17], et hallata ja salvestada projekti serveripoolset kasutajaliidest ning graafilist kasutajaliidest. Meeskond valis selle tööriista koodi rakendamiseks, sest sellega oli neil varasemaid kogemusi ja nad olid teadlikud platvormi eelistest. GitLab sobis hästi koodiversioonide haldamiseks ja muudatuste jälgimiseks. Lisaks pakkus GitLab eduka koostöö korraldamiseks ulatuslikku ülesannete ajakava loomise võimalust ja tähtaegade ning prioriteetide seadmist.

GitLab on DevSecOps platvorm, mis võimaldab organisatsioonidel maksimeerida tarkvaraarenduse üldist tulu, pakkudes tarkvara kiiremini ja tõhusamalt, tugevdades samal ajal turvalisust ning vastavust. GitLab'iga saab iga meeskond organisatsioonis ühiselt planeerida, ehitada, turvata ja juurutada tarkvara, et saavutada äritulemusi

kiiremini ning saavutada täielik läbipaistvus, järjepidevus ja jälgitavus kogu DevSecOps'i elutsükli jooksul [18].

### **2.2.6 Ajalogimine**

Autorid kasutasid interaktiivset Toggl Track keskkonda [19], et registreerida lõputööle kulunud aeg ja jälgida edusamme. See võimaldas meeskonnal süstemaatiliselt jälgida kulunud aega, tõhusalt planeerida tulevast tööd ja hinnata iga arendusetapi tootlikkust, pakkudes struktureeritud ning sihipärast lähenemist projekti elluviimisele.

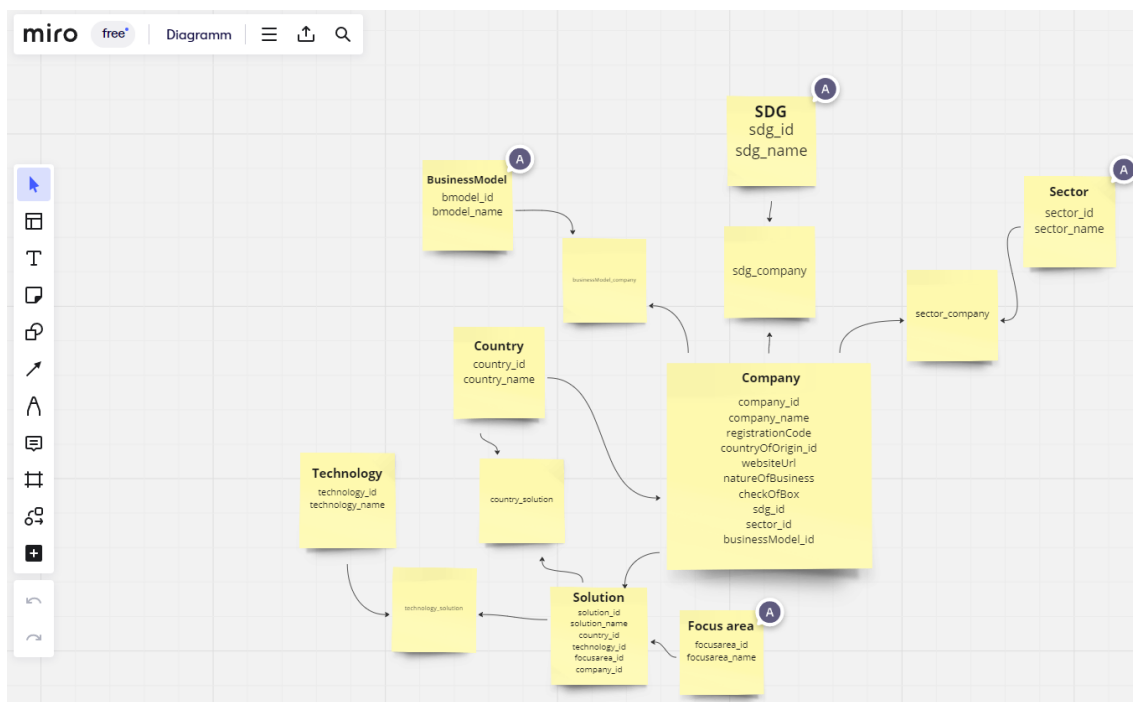
### **2.2.7 Suhtlus**

Autorite ja juhendaja ning kaasjuhendaja vaheline suhtlus toimus iganädalaselt Microsoft Teams'i töökeskkonnas [20]. Nende kohtumiste käigus demonstreeriti tööd, arutati nii projekti positiivseid kui ka negatiivseid aspekte ning seati eesmärgid järgnevateks nädalateks. Autorid eelistasid omavahelise suhtlusvahendina kasutada Facebook Messenger'it [21] ja Discord'it [22], mis tagas kiire ja tõhusa meeskonna suhtluse kogu arenduse vältel.

### **2.2.8 Muud tööriistad**

Lõputöö koostamise käigus kasutasid töö autorid aktiivselt erinevaid abivahendeid, mis parandasid oluliselt töö kvaliteeti ja mugavust.

Üheks töövahendiks oli Miro platvorm [23], mis on visuaalne tööruum innovatsiooni jaoks, kus igas suuruses meeskonnad saavad projekte planeerida, kujundada ja ühiselt töötada [24]. Platvorm võimaldas meeskonnal ühiselt kavandada ja kujundada tulevase rakenduse arhitektuuri juba arenduse varajases etapis. Tänu sünkroonse suhtluse võimalusele sai iga meeskonnaliige teha projektile ettepanekuid ja muudatusi, mis aitasid kaasa kavandatud ideede kiiremale ning tõhusamale elluviimisele ja samaaegselt kiirendada üldist arendusprotsessi (Joonis 1).



Joonis 1. Projekti arendamine Miro's

Veebirakenduse arendamise protsessis kasutas meeskond Lucidchart [25] veebiplatvormi olemi-suhte ja kasutuslugude diagrammide loomiseks. Varasem positiivne kogemus mängis olulist rolli platvormi kasutuselevõtul. Lucidchart'il on mugav kasutajaliides ja see pakub kõiki vajalikke funktsioone diagrammide tõhusaks loomiseks. See tagas meeskonnale mugava ja produktiivse töö, mis tegi ülesande palju lihtsamaks.

Tekstide tõlkimiseks valis meeskond kaasaegse Deepl tõlkija [26], mis töötab neuronivõrkude alusel, mis on omakorda koolitatud suure hulga tõlgete töö põhjal. Selline lähenemisviis võimaldab Deepl tõlkijal paremini ja täpsemalt töötada kui alternatiivsed tõlkelahendused [27].

ChatGPT-4 [28] on pakkunud protsessi käigus toetust läbi tekstide redigeerimise ja nõuannete andmisel tööriistadega, nagu Docker. Tekkivatele probleemidele pakkus platvorm võimalikke lahendusi ja juhiseid edaspidisteks sammudes, kuigi tihti esines tagasilööke programmi piiratud võimete tõttu.

## 2.3 Tööprotsessi kirjeldus

### 2.3.1 Kasutatud metoodikad

Töö algul kaardistati erinevad lüngad veebirakenduse funktsioonide osas, keskendudes funktsioonide vajadusele ning visuaalsele tulemusele. Seetõttu otsustasid autorid toetuda Scrum agiilse arendusmeetodi põhimõtetele [46]. Paremate tulemuste saavutamiseks nii meeskonnatöö protsessis kui ka arendatavas veebirakenduses otsustati kasutada ka Extreme Programming (XP) metoodika mõningaid põhimõtteid [47].

Scrum metoodika on tüüpiliselt kasutusel suuremates meeskondades, kuid esinevad kohandamise võimalused kaheliikmelise meeskonna jaoks, säilitades selle põhiprintsiibid ja jäädes samaaegselt paindlikuks. Meeskond tegi koostööd tööde planeerimisel, ülesannete valimisel ja eesmärkide püstitamisel. Iganädalased koosolekud aitasid jälgida edusamme, plaane ja takistusi, säilitades samal ajal järjepidevuse. Mõlemad autorid jagasid koostöös ülesandeid, muutes prioriteete ja lisades uusi elemente, olenevalt püstitatud eesmärkidest ning juba tehtud tööst [46]. Kaheliikmelise meeskonna Scrum metoodika rõhutab koostööd, iteratiivset edenemist ja pidevat täiustamist.

Extreme Programming (XP) metoodikast lähtudes järgisid autorid järgmisi põhimõtteid:

- Paarisprogrammeerimine - autorid teevad koostööd ühe kooditüki kallal, lihtsustades pidevat koodi ülevaatamist, ühist probleemide lahendamist ja teadmiste jagamist. See aitab tuvastada ja parandada vigu, parandades koodi kvaliteeti;
- Kollektiivne koodi omand - autorid jagavad kogu koodibaasi omandiõigust, võimaldades igal meeskonnaliikmel koodi mis tahes osa muuta. See soodustab koostööd ja tagab projekti tervikliku mõistmise;
- Kliendiga pidev suhtlemine tagasiside saamiseks - XP rõhutab tihedat koostööd kliendiga kogu projekti vältel. Suhtlus annab kohest tagasisidet edenemise kohta, tagab projekti vastavuse ärivajadustele ja aitab kliendi rahulolu tagamiseks funktsioone prioritseerida.

Ülaltoodud meetodid loovad dünaamilise koostööpõhise arenduskeskkonna, mille tulemuseks on kvaliteetsem tarkvara ja paremad projektitulemused.

### **2.3.2 Rollid**

Rollide jagamisel toetusid autorid meeskonnatöö kogemusele. Paindliku protsessi eesmärgil otsustati määratud ülesannete täitmisel teha koostööd ja välistati piiratud rollide jaotus. Selline lähenemine aitas kaasa projekti edukale lõpuleviimisele, kuna autorid said tööd võrdselt jagada ja vajadusel üksteist eesmärkide saavutamisel aidata.

Arvestada tuli aga ka autorite tugevate külgede ja huvidega, mille tulemusena sai tagarakenduse vastutus Anastasiiale ja kasutajaliidese vastutus Marijale. Töö kirjutamise ajal ei takistanud see vastutuse jaotus igal autoril osalemast veebirakenduse mõlema osa väljatöötamises.

### **2.3.3 Tegevused**

Projekti arendustegevused olid järgmised:

1. Iganädalased koosolekud - kord nädalas kohtusid autorid, bakalaureusetöö juhendaja ja kaasjuhendaja Microsoft Teams'i vahendusel, et arutada eelmise nädala tulemusi ning seada järgmise nädala eesmärgid;
2. Veebirakenduse arendus - pärast iganädalast koosolekut tegid autorid tavaliselt omavahel täiendava kõne, et arutada nädala kava ja ülesanded omavahel ära jagada. Seejärel asusid autorid oma nädala eesmärgi täitma ning vajadusel pidasid Facebook Messenger'is kirjavahetust, et küsimusi täpsustada või teineteist aidata;
3. Paarisprogrammeerimine - võimaluse korral leppisid autorid kokku kas veebi teel või reaalajas kohtumised paarisprogrammeerimiseks. Praktika on näidanud häid tulemusi varasemates ainetes, parandades tehtud töö kvaliteeti ja lihtsustades suhtlust autorite vahel;
4. React raamistiku õppimine - projekt eeldas kasutajaliidese arendamist, mistõttu oli vaja võimalikult palju õppida sobivat programmeerimiskeelt. Autorid valisid React'i, sest see on üks populaarsemaid ja sagedamini kasutatavaid raamistikku. React'i õppimisest sai Marija põhiülesanne ja sel ajal, kui Anastasiia tagarakenduse arendas, kasutas Marija raamistiku õppimiseks YouTube'i

platvormi [29], [30] ja w3schools.com veebilehte [31]. Anastasiia on pühendanud aega ka kasutajaliidese õppimisele väiksemas mahus, valides YouTube'i platvormi videokursused [29], [32].

## **3 Töö tulemused**

Selles peatükis esitatakse autorite tehtud töö tulemused. Esmalt esitatakse olemasolevate lahenduste analüüs, seejärel loetletakse veebirakendusele esitatavad funktsionaalsed ja mittefunktsionaalsed nõuded. Järgnevalt kirjeldatakse töös kasutatud arhitektuuri ja tehnilist lahendust. Lõpuks selgitatakse koodi ja kirjeldatakse valitud testimismeetodit.

### **3.1 Olemasolevad lahendused**

GovTech'i ettevõtete turg laieneb ja areneb pidevalt, mis peegeldab selle teema kasvavat tähtsust tänapäeva maailmas [33]. Tänapäeval pöörduv üha rohkem avaliku ja erasektori organisatsioone tehnoloogiliste lahenduste poole, et optimeerida oma tööd ja pakkuda uuenduslikke teenuseid. Seepärast on tekkimas veebirakendused, mis võimaldavad kasutajatel mitte ainult tutvuda GovTech'i ettevõtete tegevusega, vaid ka teada saada, milliseid eesmärke nad järgivad, kuidas nende lahendused võivad muuta erinevaid eluvaldkondi ja aidata kaasa säästva arengu eesmärkide saavutamisele.

Allpool on esitatud kolm uuenduslikku lahendust, mis on tihedalt seotud GovTech'i ettevõtete tegevusega. Lõputöö raames analüüsisid autorid koostöös juhendaja ja kaasjuhendajaga neid veebirakendusi üksikasjalikult. Analüüsid hõlmasid mitte ainult parimate tavade ja ideede uurimist, vaid ka olemasolevate lahenduste põhiprobleemide ning puuduste tuvastamist. See võimaldas üle võtta edukaid lähenemisviise ja tuvastada parendusnõudeid ning tõhusamate lahenduste loomist. Analüüsi tulemusena saadi põhjalik ülevaade sellest, millised funktsioonid on kõige nõutumad ja millised probleemid vajavad lahendamist.

#### **3.1.1 IPS-X (Innovative Public Services Explorer) [34]**

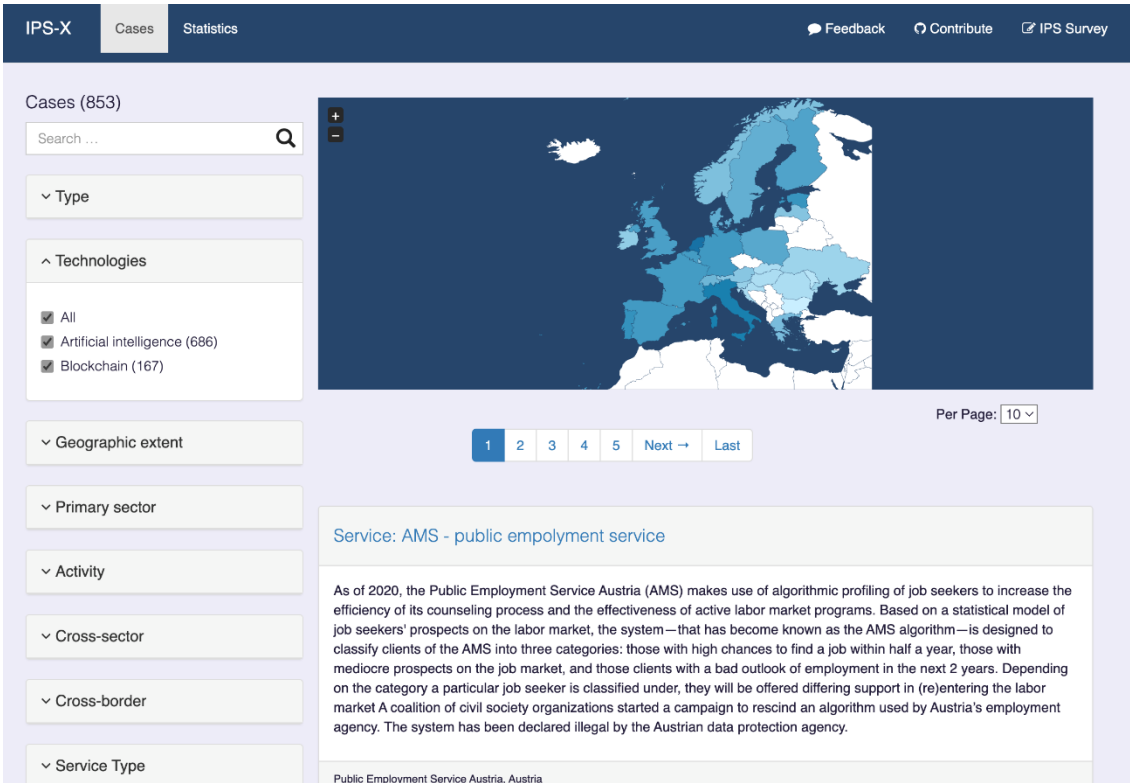
Innovatiivsete avalike teenuste programmi "Innovative Public Services Explorer" eesmärk on luua terviklik ülevaade avalikest teenustest, kasutades uusi ja murrangulisi tehnoloogiaid [34].



IPS Explorer'i veebisaidi arendamist alustati 2019. aasta septembris, mida juhtisid Andrea Perego ja Marco Combetto [35]. Veebiressurs on avatud kõigile kasutajatele, pakkudes põhjalikku teavet erinevate GovTech'i kasutusjuhtumite kohta. Veebisait võimaldab tutvuda lahenduste erinevate aspektidega, sealhulgas tehnoloogia, geograafiline asukoht, esmane sektor, tegevuse liigid, sektoriülesed ja piiriülesed projektid, teenuste liigid ning nende staatus (Joonis 2).

Eraldi vahekaardil on eri aspekte hõlmav statistika. Näiteks, statistika tehnoloogiajaotuse kohta kogutud juhtudel, esmatasandi sektorite ja ka haldustaseme kohta ning palju muud. Veebilehel on ka interaktiivne kaart, mis näitab GovTech'i lahenduste geograafilist jaotust, tuues tumedamate toonidega esile kõige aktiivsemad piirkonnad (Joonis 3).

Meeskond jõudis järeldusele, et lahendus on informatiivne ja lihtne kasutamiseks, kuid probleem seisneb selles, et see puudutab ainult Euroopa projekte ja keskendub peamiselt tehisintellektile (AI), kuigi GovTech ei piirdu sellega..



IPS-X Cases Statistics Feedback Contribute IPS Survey

Cases (853)

Search ...

Type

Technologies

All

Artificial intelligence (686)

Blockchain (167)

Geographic extent

Primary sector

Activity

Cross-sector

Cross-border

Service Type

Per Page: 10

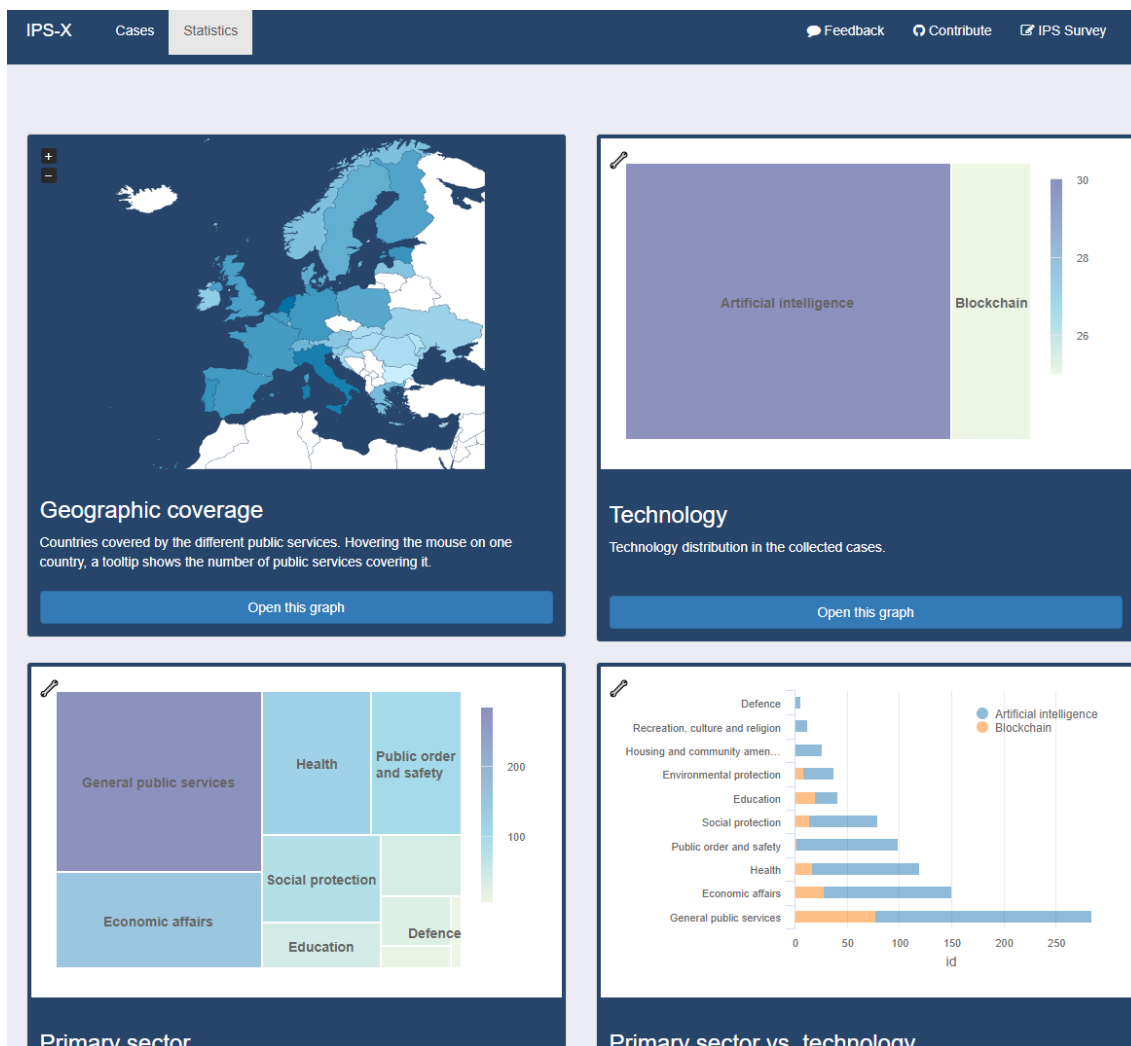
1 2 3 4 5 Next → Last

Service: AMS - public employment service

As of 2020, the Public Employment Service Austria (AMS) makes use of algorithmic profiling of job seekers to increase the efficiency of its counseling process and the effectiveness of active labor market programs. Based on a statistical model of job seekers' prospects on the labor market, the system—that has become known as the AMS algorithm—is designed to classify clients of the AMS into three categories: those with high chances to find a job within half a year, those with mediocre prospects on the job market, and those clients with a bad outlook of employment in the next 2 years. Depending on the category a particular job seeker is classified under, they will be offered differing support in (re)entering the labor market. A coalition of civil society organizations started a campaign to rescind an algorithm used by Austria's employment agency. The system has been declared illegal by the Austrian data protection agency.

Public Employment Service Austria, Austria

Joonis 2. IPS-X Cases



Joonis 3. IPS-X Statistics

### 3.1.2 Interoperable Europe (Joinup) [36]

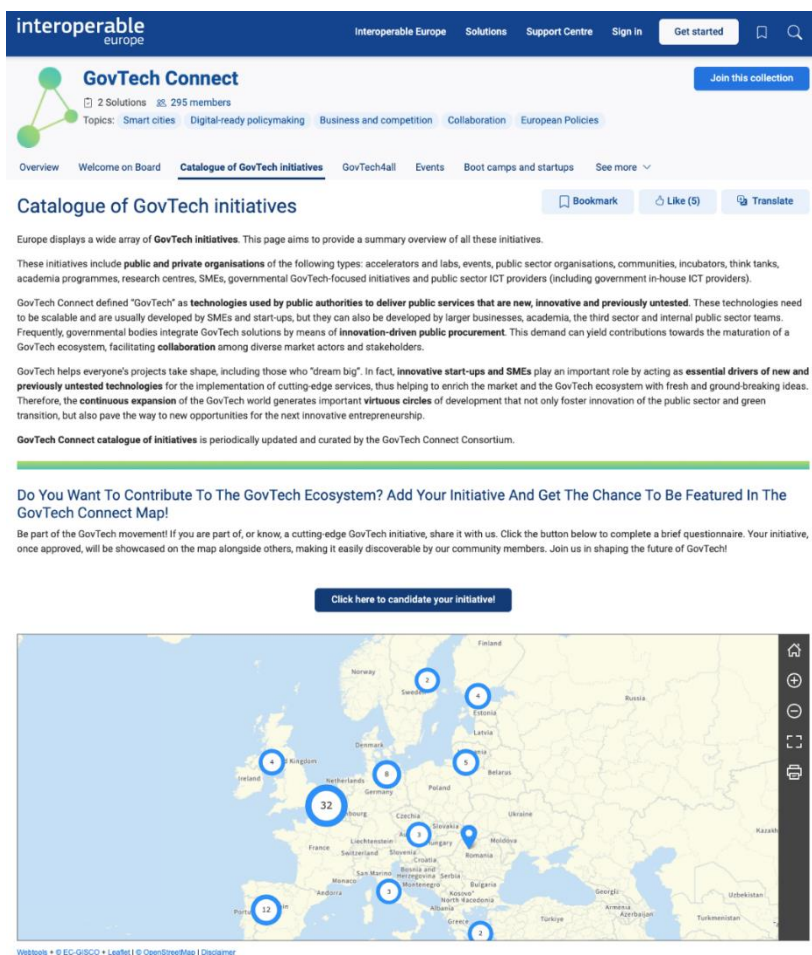
Joinup on Euroopa Komisjoni lahendus koostalitlusvõimeliste, avatud ja tasuta digivalitsemise IKT-lahenduste jaoks ning veebiruum, kus E-valitsuse spetsialistid ja entusiastid saavad jagada teavet ning õppida tundma digivalitsemise teenuseid ja algatusi [37].

Alates 2021. aastast on Joinup'ist saanud kodu Interoperable Europe jaoks, mis on Euroopa Komisjoni algatus koostalitlusvõime poliitika tugevdamiseks, tutvustades valitsuse koosvõimega seotud uudiseid ja sündmusi, samuti algatusi ning parimaid tavasid, mis toetavad Euroopa Liidu valitsuse digitaliseerimist [37].

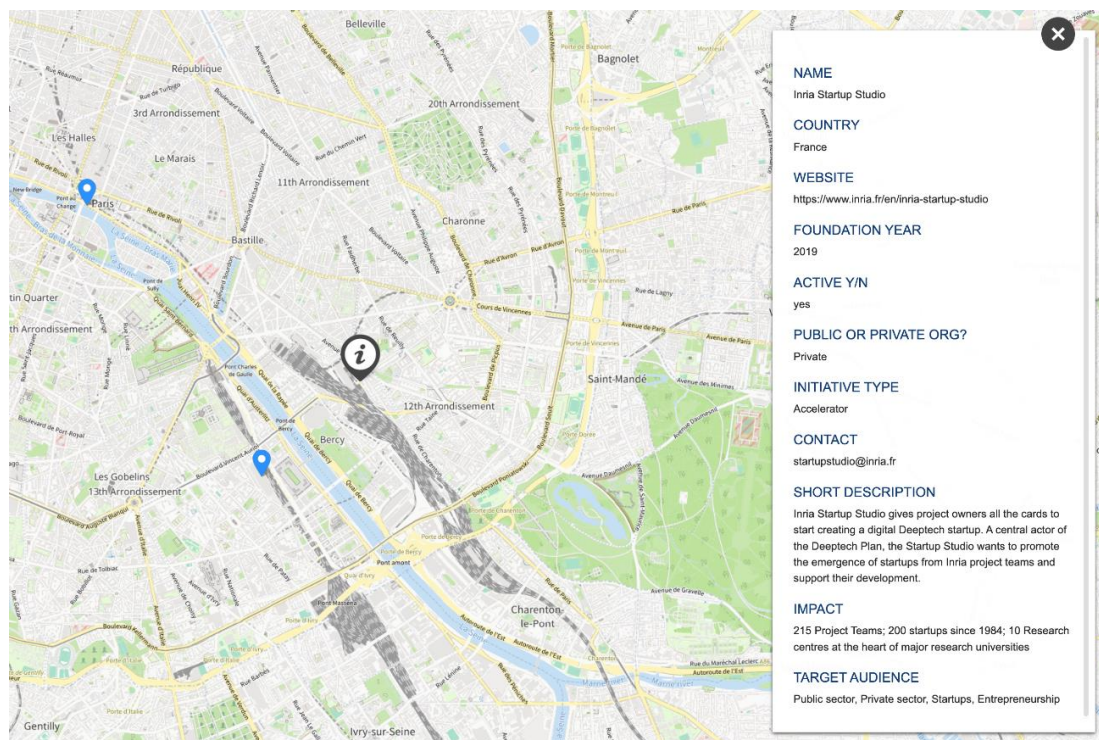
Joinup veebilehelt leitakse teavet GovTech'i algatuste kohta. Autoreid huvitav osa, mis sisaldab GovTech'i algatuste kataloogi, sisaldab kolme peamist funktsiooni (Joonis 4).

Esimene osa annab üldise kirjelduse selle kohta, mida GovTech'i algatused ja nende liikmesorganisatsioonid endast kujutavad, teine funktsionaalne osa võimaldab registreerida oma algatuse kataloogi lisamiseks. Kolmas osa on kataloog ise, kuid see on esitatud kaardi kujul, millele on märgitud kataloogi algatused. Kui valitakse algatus, avaneb kaardil teave nagu nimi, riik, veebisait, kirjeldus ja muu (Joonis 5).

Veebikataloogi nõrkuseks on see, et puudub filtreerimise, sorteerimise ja otsimise võimalus, mis raskendab vajaliku info leidmist. Samuti pole võimalust vaadata kõiki lisatud algatuste loendit, mis ei anna täielikku ülevaadet. Lisaks eelnevale annab see lahendus (sarnaselt eelmisega punktis 3.1.1) teavet ainult Euroopas registreeritud algatuste kohta.



Joonis 4. Interoperable Europe. Catalogue of GovTech initiatives



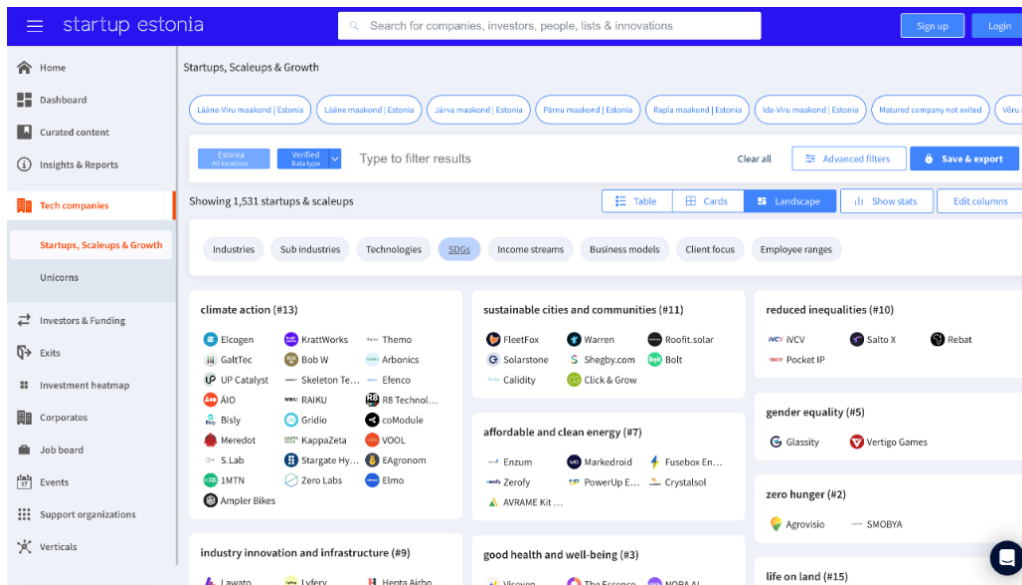
Joonis 5. Interoperable Europe. Kaart infoga

### 3.1.3 Startup Estonia [38]

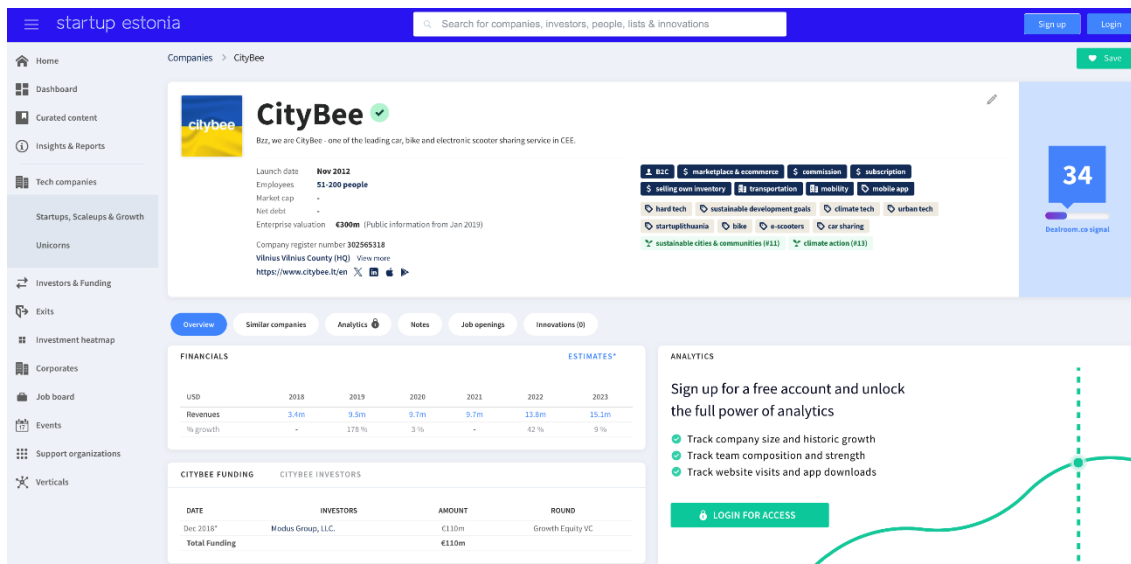
Startup Estonia on valitsuse algatus Eesti Ettevõtluse ja Innovatsiooni Sihtasutuse raames. Organisatsioon teeb koostööd erinevate sidusrühmadega, et ühendada erinevad sektorid idufirmade kogukonnaga. Startup Estonia programmi rahastab Euroopa Regionaalarengu Fond [39].

Kuigi Startup Estonia'l puudub GovTech'i algatustega seotud ettevõtete kataloog, on siiski võimalik leida infot Eesti idufirmade kohta, kelle eesmärgid ja lahendused on seotud SDG eesmärkidega, mis omakorda on tihedalt seotud GovTech'i eesmärkidega (Joonis 6).

Autorid ei pea Startup Estonia veebilehe täiesti alternatiivseks lahenduseks, kuna selle põhieesmärk on teistsuguse tähendusega ja see ei sisalda eraldi GovTech'i ettevõtete nimekirja. SDG eesmärkide järgi idufirmade sorteerimise olemasolu võimaldab aga vaadata ettevõtteid, kes on olulised elukvaliteedi ja inimeste heaolu tõstmiseks digiteenuste pakkumise arendamisel. Lisaks eeltoodule said autorid inspiratsiooni Startup Estonia veebilehe kasutajaliidesest, kuna see osutus meeldivaks ja hõlpsasti kasutatavaks (Joonis 7).



Joonis 6. Startup Estonia



Joonis 7. Startup Estonia. Ettevõtte info

### 3.1.4 Kokkuvõte olemasolevatest lahendustest

Pärast olemasolevate lahenduste põhjalikku uurimist jõudsid juhendaja, kaasjuhendaja ja autorid ühisele järeldusele, et olemasolevatel lahendustel on omad miinused ja plussid, kuid puudub lahendus, mis täidaks kõiki vajalikke funktsioone. Seetõttu otsustasid töö autorid proovida ühendada olemasolevate lahenduste eelised uues veebirakenduses, mis võimaldaks kombineerida infot GovTech'i ettevõtete kohta üle kogu maailma.

## 3.2 Nõuded

Selles peatükis esitatakse bakalaureusetöö osana välja töötatud projekti funktsionaalsed ja mittefunktsionaalsed nõuded. Projektis osalejad määratlesid rakendusele ja süsteemile tervikuna esitatavad nõuded, uurides ning arutades olemasolevaid lahendusi. Nõuete väljatöötamisel arvestati eelkõige kaasjuhendaja soove, aga ka lõppkasutajate arvu suurendamiseks vajalikke funktsioone [48].

### 3.2.1 Funktsionaalsed nõuded

Funktsionaalsed nõuded kirjeldavad konkreetseid käitumisviise ja funktsioone, mida süsteem peab oma eesmärkide saavutamiseks täitma. Need nõuded määratlevad, mida süsteem peab tegema, nagu protsessid, andmetöötlus, kasutaja interaktsioon ja muud funktsioonid, mida lõppkasutaja ja administraator süsteemilt ootavad. See selgus aitab tagada, et lõplik süsteem töötab ettenähtud viisil [48].

Veebirakenduse arendamise käigus tuvastati järgmised funktsionaalsed nõuded:

- Administraator saab logida sisse ja ainult pärast muuta, uuendada, lisada või kustutada andmeid.
- Administraator saab lisada süsteemi GovTech'i ettevõtete või lahendustega seotud infot.
- Administraator saab muuta GovTech'i ettevõtete või lahendustega seotud andmeid.
- Administraator saab kustutada GovTech'i ettevõtet ja lahendust kataloogist.
- Kasutaja ja administraator saavad vaadata GovTech'i ettevõtetega seotud infot.
- Kasutaja saab filtreerida GovTech'i ettevõtete kataloogi eelmääratletud kategooriate järgi.
- Kasutaja ja administraator saavad huvitava GovTech'i ettevõtte või lahenduse leidmiseks kasutada otsinguriba.

- Kasutaja saab vaadata erinevate kategooriate andmeid sisaldavaid statistilisi tabeleid.
- Kasutaja saab vaadata maailmakaardist GovTech'i ettevõtete arvu riigiti.
- Kasutaja võib esitada oma ettevõtte kandidatuuri GovTech'i ettevõtete kataloogi lisamiseks.

### **3.2.2 Mittefunktsionaalsed nõuded**

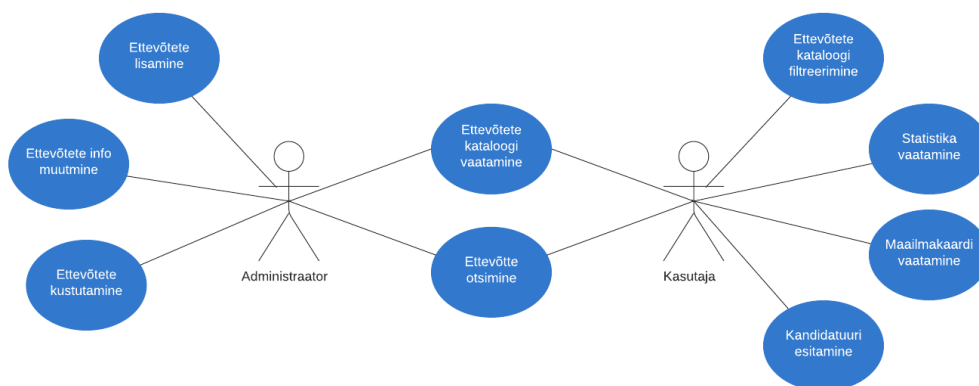
Mittefunktsionaalsed nõuded määratlevad süsteemi omadusi nagu jõudlus, kasutatavus, töökindlus ja turvalisus. Erinevalt funktsionaalsetest nõuetest, mis kirjeldavad, mida süsteem peaks tegema, kirjeldavad mittefunktsionaalsed nõuded kuidas süsteem peaks erinevates tingimustes toimima ja käituma. Need nõuded on süsteemi projekteerimisetapi lahutamatu osa tagamaks, et süsteem vastaks nii äri- kui ka tehnilise kvaliteedinõutele [48].

Veebirakenduse arendamise käigus tuvastati järgmised mittefunktsionaalsed nõuded:

- Veebirakenduse kasutajaliides on lihtne ja kasutajale arusaadav.
- Veebirakenduse kasutajaliides on kasutatav erinevates veebibrauserites.
- Veebirakenduse kasutajaliides on sama hästi rakendatud nii arvuti, sülearvuti kui ka nutitelefoniga jaoks.

### **3.2.3 Kasutuslood**

Koos funktsionaalsete nõuetega koostasid autorid kasutuslugude diagrammi, et selgemalt näidata veebirakenduse kasutamise võimalusi administraatori ja kasutaja jaoks (Joonis 8).



Joonis 8. Kasutuslood

### 3.3 Arhitektuur

Veebirakendus koosneb kolmest komponendist (Joonis 9):

- **Andmebaas** - salvestab andmeid ettevõtete ja nende lahenduste kohta ning vajadusel otsib neid välja.
- **Tagarakendus** - vastab kasutajaliidesest saadud päringutele, töötleb neid ja tagastab vajalikud andmed andmebaasist või operatsioonide tulemused.
- **Kasutajaliides** - rakenduse visuaalne osa, millega kasutaja suhtleb veebibrauseri kaudu, kasutades mitmesuguseid juhtelemete vajaliku teabe hankimiseks.



Joonis 9. Üldine arhitektuur

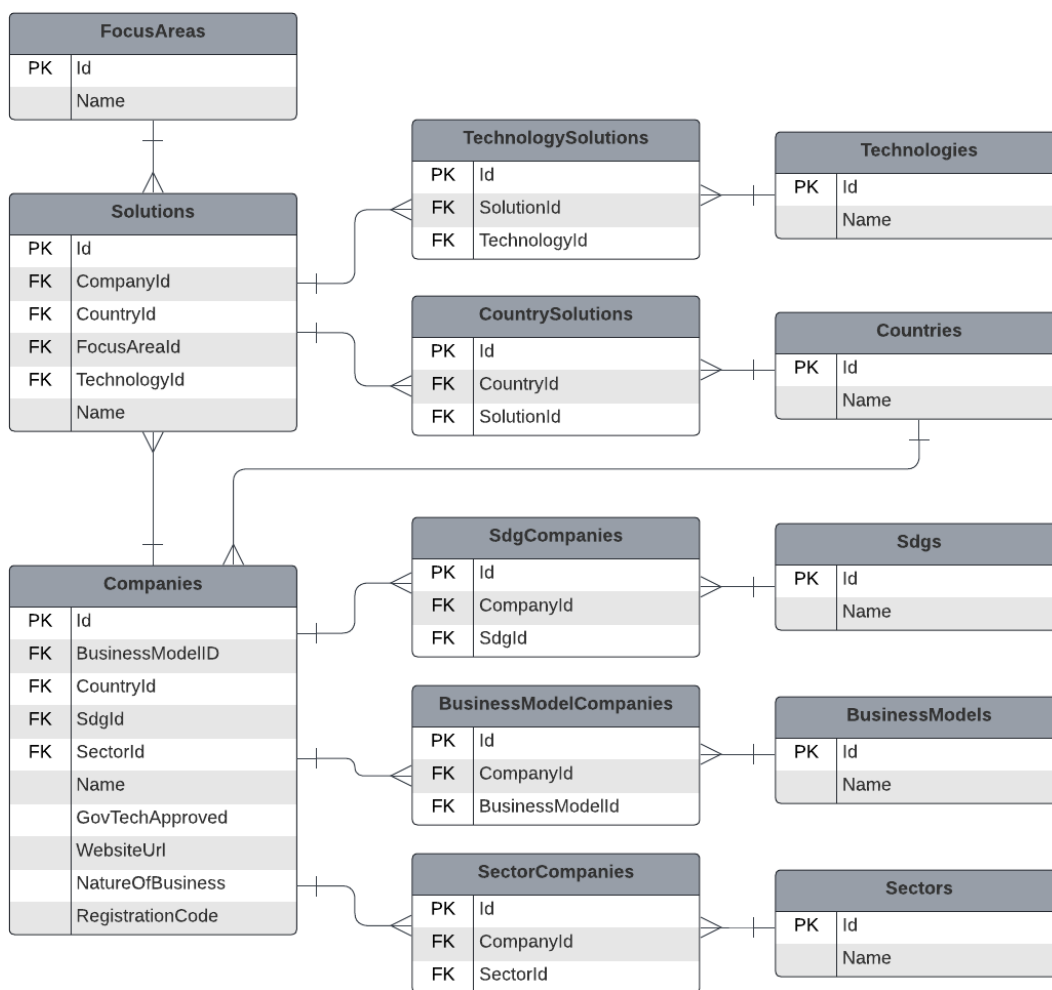


### 3.3.1 Andmebaasi arhitektuur

Autorid kasutasid andmete säilitamiseks ja haldamiseks SQLite'i andmebaasi.

Kokku on andmebaasis 8 tabelit ja 5 ühendustabelit (Joonis 10):

- Companies – tabel sisaldab andmeid registreeritud GovTech'i ettevõtete kohta;
- Solutions – tabel sisaldab lahendusi, mida GovTech'i ettevõtted pakuvad;
- FocusAreas – tabelisse on sisestatud fookusvaldkonnad, mille järgi on liigitatud GovTech'i ettevõtted;
- Technologies – tabelisse on salvestatud tehnoloogiad, mida GovTech'i ettevõtted kasutavad;
- Countries – tabel sisaldab kõikide riikide loendit;
- Sdgs – tabelisse on sisestatud kõik 17 säästva arengu eesmärki;
- BusinessModels – tabel sisaldab kõikvõimalikke ärimudelite loendit;
- Sectors – tabelisse on sisestatud sektorid, mille järgi saab GovTech'i ettevõtteid klassifitseerida;
- SectorCompanies - ühendustabel sektorite ja ettevõtete vahel;
- BusinessModelCompanies - ühendustabel ärimudelite ja ettevõtete vahel;
- SdgCompanies - ühendustabel säästva arengu eesmärkide ja ettevõtete vahel;
- CountrySolutions - ühendustabel riikide ja lahenduste vahel;
- TechnologySolutions - ühendustabel tehnoloogiate ja lahenduste vahel.



Joonis 10. Olemi-suhte diagramm

### 3.3.2 Tagarakenduse arhitektuur

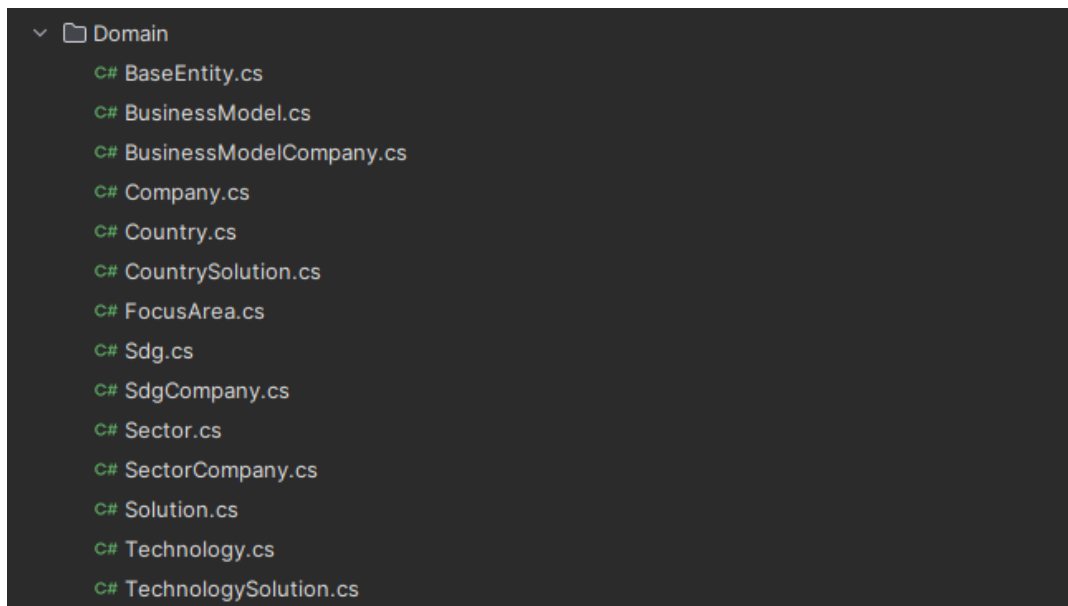
Tagarakendus on rakendatud C# programmeerimiskeeles, kasutades ASP.NET Core MVC raamistikku.

Projekti tagarakenduse arhitektuur on üles ehitatud MVC (Model-View-Controller) põhimõttel. Mudel jagab rakenduse kolmeks põhikomponendiks: mudel, vaade ja kontrolleri, mis annavad võimaluse neid sõltumatult muuta, lihtsustades seega veebirakenduste arendamist ja toetamist [40].

#### 3.3.2.1 Mudelid

Mudel on komponent, mis sisaldab rakenduse andmeid. Sellel ei ole vaadet kasutajaliidesele. Mudel vastutab domeeniloogika (tegelikud ärireeglid) käsitlemise ja andmebaasi ning võrgukihiga suhtlemise eest [40].

Projektis kasutatakse 14 mudelit, millest igaühel on rakenduse arhitektuuris oma roll (Joonis 11). Need hõlmavad ärioloogika ja andmehalduse erinevaid aspekte, sealhulgas andmeid ettevõtete, riikide, ärimudelite, majandussektorite, jätkusuutlikkuse eesmärkide ja tehnoloogiate kohta. Mudelite vahelised ettenähtud seosed tagavad, et andmete sõltuvused ja andmete terviklikkus on tõhusalt korraldatud ja hallatud.



Joonis 11. Mudelid

Iga mudel on rakendatud C#-klassi abil, mis määrab selle omadused ja andmetüübid. Klassi atribuudid hõlbustavad valideerimist ja suhete haldamist teiste mudelitega. Mudelite ühendamise struktuur hõlmab üks-mitmele, mitu-mitmele ja üks-ühele suhteid, mida rakendatakse navigeerimisomaduste ja kogumite kaudu.

Näitena on võetud kood Solution mudelist (Joonis 12).

```
( ) namespace WebApp.Domain;

22 usages anopry 2 exposing APIs More...
public class Solution: BaseEntity
{
    24 usages
    public string Name { get; set; } = default!;

    16 usages
    public ICollection<CountrySolution>? CountrySolutions { get; set; }

    8 usages
    public Guid CompanyId { get; set; }

    12 usages
    public Company? Company { get; set; }

    8 usages
    public Guid FocusAreaId { get; set; }

    12 usages
    public FocusArea? FocusArea { get; set; }

    16 usages
    public ICollection<TechnologySolution>? TechnologySolutions { get; set; }
}
```

Joonis 12. Kood Solution mudelist

### 3.3.2.2 Vaated

Vaade (View) - on kasutajaliidese kiht, mis sisaldab ekraanil nähtavaid komponente. Lisaks sellele pakub see mudelisse salvestatud andmete visualiseerimist ja kasutajale interaktsiooni [40]. See ei sisalda äriloogikat ega andmetöötlusoperatsioone, vaid keskendub teabe visualiseerimisele kasutajasõbralikus vormis.

Projektis kasutatakse 15 vaadet (Joonis 13).

```
Views
├── BusinessModelCompanies
├── BusinessModels
├── Companies
│   ├── Create.cshtml
│   ├── Delete.cshtml
│   ├── Details.cshtml
│   ├── Edit.cshtml
│   └── Index.cshtml
├── Countries
├── CountrySolutions
├── FocusAreas
├── Home
├── SdgCompanies
├── Sdgs
├── SectorCompanies
├── Sectors
├── Shared
├── Solutions
├── Technologies
└── TechnologySolutions
```

Joonis 13. Vaated

Iga vaade sisaldab viit põhifaili:

- Create.cshtml - lehekülge uue elemendi loomiseks, kus kasutaja täidab nõutavad väljad ja saadab selle andmebaasi salvestamiseks.
- Delete.cshtml - lehekülge, kus kasutaja näeb kogu informatsiooni elemendi kohta, mida ta kavatses kustutada. Seda lehekülge kasutatakse elemendi kustutamise kinnitamiseks, et vältida andmete soovimatut kustutamist.
- Details.cshtml - lehekülge näitab kogu vajalikku informatsiooni valitud elemendi kohta, mis võib sisaldada kõiki konkreetse elemendi kohta olemasolevaid andmeid.
- Edit.cshtml - lehekülge, mis võimaldab kasutajal muuta olemasoleva elemendi andmeid. See kujutab endast vormi, mis on eelnevalt täidetud elemendi hetkeandmetega, mida saab muuta ja salvestada.
- Index.cshtml - lehekülge, mis sisaldab kõigi loodud või muudetud elementide nimekirja, kus kuvatakse ka selle olemasolevad andmed. See sisaldab võimalust otsida elemente nimekirjast, samuti järjestikuse lehekülje nummerdamist.

Iga vaade kirjutati Razoni süntaksiga, mis omakorda võimaldas C# koodi otse HTML-märgistusse põimida, muutes vaate paindlikumaks ja kasutajasõbralikumaks.

Seega mängivad selles projektis vaated võtmerolli tagarakendusel välja töötatud administraatori liideses. Need kuvavad andmeid selgelt ja struktureeritult, salvestavad administraatori tegevused rakenduses edasiseks töötlemiseks, võimaldavad navigeerimist veebilehe sektsioonides ning osalevad otseselt andmete loomisel, muutmisel, kustutamisel ja vaatamisel, andes administraatorile täieliku kontrolli veebirakenduse funktsionaalsuse üle.

Näitena on võetud kood Solution vaatest (Joonis 14).

```
model WebApp.Domain.Solution

@{
    ViewData["Title"] = "Create";
}

<h1>Create</h1>

<h4>Solution</h4>
<hr />
<div class="row">
    <div class="col-md-4">
        <form asp-action="Create">
            <div asp-validation-summary="ModelOnly" class="text-danger"></div>
            <div class="form-group">
                <label asp-for="Name" class="control-label"></label>
                <input asp-for="Name" class="form-control" />
                <span asp-validation-for="Name" class="text-danger"></span>
            </div>
            <div class="form-group">
                <label asp-for="CompanyId" class="control-label">Company</label>
                <select asp-for="CompanyId" class="form-control" asp-items="ViewBag.CompanyId"></select>
            </div>
            <div class="form-group">
                <label asp-for="FocusAreaId" class="control-label">FocusArea</label>
                <select asp-for="FocusAreaId" class="form-control" asp-items="ViewBag.FocusAreaId"></select>
            </div>
        </form>
    </div>
</div>
```

Joonis 14. Kood Solution vaadest

### 3.3.2.3 Kontrollerid

Kontroller – komponent, mis loob ühenduse vaate ja mudeli vahel. See sisaldab rakenduse põhiloogikat ja töötleb kasutajalt saadud andmeid mudeli vastavaks uuendamiseks [6].

Projekt sisaldab 14 kontrollerit (Joonis 15).

```
Controllers
  C# BusinessModelCompaniesController.cs
  C# BusinessModelsController.cs
  C# CompaniesController.cs
  C# CountriesController.cs
  C# CountrySolutionsController.cs
  C# FocusAreasController.cs
  C# HomeController.cs
  C# SdgCompaniesController.cs
  C# SdgsController.cs
  C# SectorCompaniesController.cs
  C# SectorsController.cs
  C# SolutionsController.cs
  C# TechnologiesController.cs
  C# TechnologySolutionsController.cs
```

Joonis 15. Kontrollerid

Kontrolleritel on järgmised komponendid:

- HTTP GET meetodid:

- Search(string searchString, int page = X, int pageSize = X) - teostab otsingu, kasutades parameetri searchString kaudu määratud nime, ja korraldab tulemused nummerdatud lehekülgedeks. Andmed laaditakse andmebaasist, võttes arvesse seoseid teiste tabelitega ning tulemused kuvatakse vaates "Index".
  - Index() - kuvab kõigi elementide nimekirja. Laeb andmed andmebaasist, sealhulgas seotud üksused ja kuvab need loeteluna.
  - Details(Guid? id) - kuvab üksikasjalikku informatsiooni elemendi kohta identifikaatori järgi. Kui elementi ei leita, tagastatakse veateade „NotFound“.
  - Create() - tagastab vaatluse uue elemendi loomiseks. Valmistab ette andmed vormil kasutatavate rippmenüüde jaoks, kui need on olemas.
  - Edit(Guid? id) - tagastab vaate olemasoleva elemendi muutmiseks selle ID järgi. Kui elementi ei leita, tagastatakse veateade „NotFound“.
  - Delete(Guid? id) - tagastab vaate elemendi kustutamise kinnitamiseks. Kuvab üksikasjaliku teabe kustutatava elemendi kohta. Kui elementi ei leita, tagastatakse veateade „NotFound“.
- HTTP POST meetodid:
    - Create(Mudel X) - võtab vastu vormiandmed ja loob andmebaasis uue elemendi. Kui edastatud andmed on korrektsed, salvestab elemendi ja selle seosed, vastasel juhul tagastab vaate koos vormi ja veateadega.
    - Edit(Guid id, Mudel X) - uuendab olemasolevat elementi ja selle seoseid andmebaasis vormiandmete põhjal. Kui andmed on korrektsed, teostab uuenduse ja suunab elementide loendisse, vastasel juhul tagastab vaate koos vormi ja veateadega.
    - DeleteConfirmed(Guid id) - kustutab elemendi andmebaasist identifikaatori järgi pärast kustutamise kinnitamist. Suunab pärast kustutamist tagasi elementide loendisse.

Seega pakuvad meetodid täielikku komplekti CRUD-operatsioone (create, read, update, delete), et hallata elemente andmebaasis, samuti otsingu- ja paginatsioonifunktsioone kasutaja mugavuse tagamiseks.

Näitena on võetud kood SolutionsController kontrolleriist (Joonis 16).

```
public class SolutionsController: Controller
{
    private readonly ApplicationDbContext _context;

    #anopry
    public SolutionsController(ApplicationDbContext context)
    {
        _context = context;
    }
    [HttpGet]
    #!usage #anopry *
    public IActionResult Search(string companyName, string focusArea, string searchString, int page = 1, int pageSize = 10)
    {
        var solutions :IQueryable<Solution> = _context.Solutions // DbSet<Solution>
            .Include(navigationPropertyPath: s :Solution => s.Company) // IncludableQueryable<Solution,Company?>
            .Include(navigationPropertyPath: s :Solution => s.FocusArea) // IncludableQueryable<Solution,FocusArea?>
            .Include(navigationPropertyPath: s :Solution => s.CountrySolutions).ThenInclude(cs :CountrySolution => cs.Country) // IncludableQue
            .Include(navigationPropertyPath: s :Solution => s.TechnologySolutions).ThenInclude(ts :TechnologySolution => ts.Technology) // In
            .AsQueryable();
        if (!string.IsNullOrEmpty(companyName))
        {
            solutions = solutions.Where(c :Solution => c.Company.Name.Contains(companyName));
        }
        if (!string.IsNullOrEmpty(focusArea))
        {
            solutions = solutions.Where(c :Solution => c.FocusArea.Name.Contains(focusArea));
        }
        if (!string.IsNullOrEmpty(searchString))
        {
            solutions = solutions.Where(c :Solution => c.Name.Contains(searchString));
        }
        var totalSolutions :int = solutions.Count();
        solutions = solutions.Skip((page - 1) * pageSize).Take(pageSize);
        ViewBag.CompanyName = companyName;
        ViewBag.FocusArea = focusArea;
        ViewBag.SearchString = searchString;
        ViewBag.CurrentPage = page;
        ViewBag.TotalPages = (int)Math.Ceiling((double)totalSolutions / pageSize);
        return View("Index", model: solutions.ToList());
    }
}
```

Joonis 16. Kood SolutionsController kontrolleriist

### 3.3.2.4 API Kontrollerid

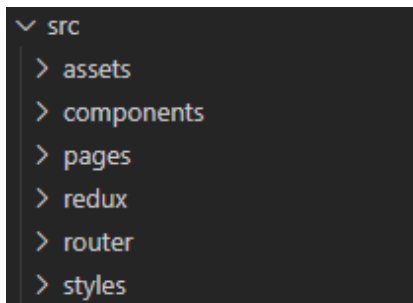
Sarnaselt eelmistele kontrolleriistele on välja töötatud API-kontrollerid, mis tagastavad andmeid JSON-formaadis, millega kasutajaliides seejärel suhtleb.

### 3.3.3 Kasutajaliidese arhitektuur

Kasutajaliidese projekt loodi kasutajale mugava ja meeldiva kasutajaliidese pakkumiseks, kuna liidese olemasolu oli üks kaasjuhendaja nõudmistest.



Kasutajaliides oli kirjutatud JavaScript programmeerimiskeeles ning selleks kasutati React raamistikku. Kasutajaliidese arhitektuur koosneb erinevatest kihtidest, mis täidavad erinevaid ülesandeid (Joonis 17).



Joonis 17. Kasutajaliidese arhitektuur

- Assets - kaust toimib veebirakenduses kasutatavate staatiliste varade keskse hoidlana. See on koht kuhu salvestatakse loodud komponentides viidatud pildid, ikoonid, fondid või muud mittekoodifailid. Ressursside korraldamine muudab nende haldamise ja värskendamise kogu projekti vältel lihtsamaks.
- Components - kaust määratleb korduvkasutatavad ehitusplokid, mis moodustavad veebirakenduse liidese. Komponentid võivad olla igasugused, alates lihtsatest nuppudest või vormidest kuni keerukate interaktiivsete elementideni. Korduvkasutatavate komponentide loomine parandab koodi hooldatavust ja veebirakenduste disaini järjepidevust. Joonis 18 näitab CompanyGrid komponenti, mis on kasutatud ettevõtete lehel. CompanyGrid ise kasutab CompanyBox komponenti, kus on määratletud ettevõtte individuaalse kaardi välimus.

```
function CompanyGrid({companies}) {  
  return (  
    <Grid container spacing={2}>  
      {companies.map((company) => (  
        <Grid key={company.id} item xs={12} sm={6} md={3} className={"items-center"}>  
          <CompanyBox company={company} />  
        </Grid>  
      )}  
    </Grid>  
  );  
}  
  
export default CompanyGrid;
```

Joonis 18. CompanyGrid komponent

- Pages - iga kausta fail on veebirakenduse eraldi jaotis (leht). Lehed koondavad komponendid, et luua üldine kasutajakogemus. Näiteks, on võimalus luua eraldi lehed esilehe, tooteloendi või kontaktivormi jaoks. Kokku on veebirakenduses neli lehte: Home (koduleht), Companies (ettevõtete leht), Company (ühe ettevõtte leht selle lahendustega), Statistics (leht statistiliste tabelitega). Allpool saab näha näidet Home lehe koodist, kus määratletakse, kuidas käitub filtrite valimine (Joonis 19).

```

const handleCheckboxChange = (category, value) => {
  setFilters((prevFilters) => {
    const updatedFilters = {
      ...prevFilters,
      [category]: {
        ...prevFilters[category],
        [value]: !prevFilters[category]?.[value],
      },
    };

    // If "All" checkbox is checked, set all other checkboxes in the category to checked
    if (value === 'all' && updatedFilters[category][value]) {
      Object.keys(updatedFilters[category]).forEach((filterValue) => {
        if (filterValue !== 'all') {
          updatedFilters[category][filterValue] = true;
        }
      });
    } else if (value !== 'all' && !updatedFilters[category][value]) {
      // If any checkbox other than "All" is unchecked, uncheck the "All" checkbox
      updatedFilters[category]['all'] = false;
    }

    return updatedFilters;
  });
};

```

Joonis 19. Home lehe koodinäide

- Redux - kui rakendus haldab keerulisi andmevooge, pakub Redux võimalust rakenduse oleku salvestamiseks ja haldamiseks, hõlbustades muudatuste analüüsimist ja kasutajaliidese komponentide sünkroonimist. See toimib veebirakenduse keskse andmehoidlana.
- Router - kaust vastutab selle eest, kuidas kasutajad veebirakenduse erinevate lehtede vahel navigeerivad. Router jälgib URL-i muutusi ja kuvab soovitud tee alusel sobiva lehekomponeendi. Router funktsioon määrab veebirakenduse jaoks marsruutide komplekti, kus iga marsruut vastab konkreetsele renderdatavale lehele (Joonis 20).

```
function Router() {
  return (
    <Routes Routes history={history}>
      <Route path="/" element={<Home />} />
      <Route path="/companies" element={<Companies />} />
      <Route path="/company/:companyId" element={<Company />} />
      <Route path="/statistics" element={<Statistics />} />
    </Routes>
  );
}
```

Joonis 20. Router funktsioon

- Styles - kaust salvestab kõik CSS-failid, mis määravad veebirakenduse komponentide ja lehtede välimuse. Siia kirjutatakse stiilid, et juhtida paigutust, värve ja fonte. Salvestades stiile, saab kogu rakenduses säilitada ühtse visuaalse kogemuse. Näiteks määratletakse projektis kasutatava külgriba menüü välimust CSS-failis (Joonis 21).

```
.sidebarBox {
  background-color: var(--dark-blue-color) !important;
  border-radius: 8px !important;
  cursor: pointer;
  padding: 10px;
  width: 100%;
  font-weight: bold;
  display: flex;
  flex-direction: column;
  justify-content: center;
  height: fit-content !important;
}
```

Joonis 21. Külgriba välimuse kood

Projekti maailmakaardi ja statistiliste tabelite lisamiseks kasutati erinevaid olemasolevaid raamistikke, et integreerida projekti vajalikud komponendid. Maailmakaardi rakendamiseks kasutati NPM-paketihaldurist võetud Jvectormap raamistikku [41], statistiliste tabelite lisamiseks aga võeti Highcharts'ist [42], Treemap'i [43] ja Heatmap'i [44].

### **3.4 Testimine**

Tagamaks veebirakenduse nii tagarakenduse kui ka kasutajaliidese kvaliteeti, viidi läbi manuaalsed testid. Ettevõtete lisamise funktsionaalsuse testimisel kontrolliti kõigi vajalike väljade olemasolu ja nendevaheliste seoste järgimist. Samuti analüüsiti ettevõtete redigeerimist ja kustutamist, et tagada nende õige toimimine.

Erilist tähelepanu pöörati lahenduste loomise ja ettevõtetesse andmete lisamise testimisele. See hõlmas lahenduste õigeaegset lisamist ja nende sidumist ettevõtetega.

Administratiivse osa jaoks kontrolliti, et ainult administraatoril oleks juurdepääs andmete muutmise funktsioonidele. See tagas süsteemi turvalisuse ja usaldusväärsuse.

Kasutajaliides läbis põhjaliku testimise. Kasutajastsenaariumid hõlmasid kõikide nuppude ja funktsioonide kontrollimist ning veebisaidi mugavuse hindamist erinevatest veebibrauseritest ja seadmetest. Lisaks hinnati kasutajaliidese tõhusust kasutajate tagasiside ja kolmandate osapoolte kasutajate kaasamise põhjal.

### **3.5 Rakenduse juurutamine**

#### **3.5.1 Docker**

Veebirakenduse tagarakendus ja kasutajaliides paigutati Docker'i konteinerisse ja kohalikud andmebaasifailid jäeti projektist välja tänu failile „gitignore“, et võimaldada koodi edastamise koormuse vähendamist. See lähenemisviis tähendab, et andmed on rakendusest eraldatud. See parandab turvalisust, sest tundlikke andmebaasiandmeid ei levitata koos rakenduskoodiga. Selline lähenemisviis lihtsustab ka rakenduse paindlikumat ja mugavamat kasutuselevõttu, suurendades selle skaleeritavust ja hõlbustades testimist, sest muudatusi andmebaasis saab teha rakenduse enda arendamisest sõltumatult.

Dockerfile'i arendamisel nii kasutajaliidesele kui ka tagarakendusele oli projekt „itb2203 2023 Workout App“ suurepäraseks lähtepunktiks [49].

Kasutatud Dockerfaili sisu on nähtav allpool toodud joonisel (Joonis 22).

```

FROM mcr.microsoft.com/dotnet/aspnet:8.0 AS base
USER $APP_UID
WORKDIR /app
EXPOSE 8080
EXPOSE 8081

FROM mcr.microsoft.com/dotnet/sdk:8.0 AS build
ARG BUILD_CONFIGURATION=Release
WORKDIR /src

COPY ["WebApp.csproj", "./"]
RUN dotnet restore "WebApp.csproj"

COPY . .

RUN dotnet build "WebApp.csproj" -c $BUILD_CONFIGURATION -o /app/build

FROM build AS publish
RUN dotnet publish "WebApp.csproj" -c $BUILD_CONFIGURATION -o /app/publish /p:UseAppHost=false

FROM base AS final
WORKDIR /app
COPY --from=publish /app/publish .
ENTRYPOINT ["dotnet", "WebApp.dll"]

```

Joonis 22. Tagarakenduse Dockerfile

Kasutajaliidese kood Docker'i konteineri loomiseks on nähtav allpool toodud joonisel (Joonis 23).

```
# Stage 1: Build the React + Vite application
FROM node:20.9-alpine3.17 as builder
RUN npm install -g pnpm

WORKDIR /app

# Copy package.json and lock file
COPY ./package.json ./pnpm-lock.yaml ./
RUN pnpm install --force

# Copy all files and build the project
COPY . .
RUN pnpm build

# Stage 2: Serve the built application with Nginx
FROM nginx:1.25.3-alpine

# Copy the Nginx configuration file
COPY ./nginx.conf /etc/nginx/nginx.conf

# Copy the build output from the builder stage
COPY --from=builder /app/dist /usr/share/nginx/html

# Expose port 80
EXPOSE 80

# Run Nginx in the foreground
CMD ["nginx", "-g", "daemon off;"]
```

Joonis 23. Kasutajaliidese Dockerfile

Docker'i konteiner on nähtav allpool toodud joonisel (Joonis 24).

<input type="checkbox"/>	 <b>govtech_backend</b> 2dcca72a2c85	<a href="#">webapp</a>	Running
<input type="checkbox"/>	 <b>govtech</b>		Running (1/1)
<input type="checkbox"/>	 <b>react-app-1</b> 82be2bb93069	<a href="#">govtech-react-app</a>	Running

Joonis 24. Docker'i konteiner

## 4 Analüüs ja järelused

Selles osas analüüsitakse projekti raames tehtud peamisi otsuseid ja tehakse vastavad järelused. Fookus pööratakse projektis kasutatavate tööriistade ja tehnoloogiate valiku põhjendamisele, samuti kirjeldatakse tööriistade ja tehnoloogiate vastavust autorite ootustele. Analüüsitakse ka rakenduse arhitektuuri ja tehnilist lahendust. Lisaks antakse ülevaade tehtud tööst ja esitatakse ideid projekti edasiseks arendamiseks.

### 4.1 Tööriistade ja tehnoloogiate valik

Tööriistade ja tehnoloogiate valimisel projekti realiseerimiseks tuginesid autorid eelnevale kogemusele ning samuti, nagu kasutajaliidese realiseerimiseks tööriistade valikul, ka isiklikele eelistustele selles osas, milliseid oskusi autorid sooviksid arendada.

Tagarakenduse arendamiseks valisid autorid programmeerimiskeele C# ja raamistiku ASP.NET Core MVC, mille põhjuseks oli varasem kogemus väikeste veebirakenduste arendamisel ülikooli kursuste raames. Autorid jäid valikuga rahule, kuna puudus vajadus uue programmeerimiskeele omandamiseks ja tekkis võimalus kasutada varasemalt omandatud teadmisi. Lisaks on see keel ja raamistik üsna populaarsed valikud veebirakenduste arendamise tööriistade valimisel, mis tähendas, et vajaliku info ja vihjete leidmine tagarakenduse arendamiseks oli sujuv protsess. Lisaks sellele, valitud arenduskeskkond JetBrains Rider jättis autoritele hea kogemuse, kuna see oli väga mugav ja kasutajasõbralik ning võimaldas autoritel tõhusalt töötada.

Veebirakenduse kasutajaliidese arendamisel pidid autorid omandama mitmeid tööriistu, sealhulgas programmeerimiskeele JavaScript ja React'i raamistiku. JavaScript'i õppimine sai kasutajaliidese funktsionaalsuse loomise aluseks ning JavaScript'il põhinev React'i raamistik lihtsustas kasutajaliidese loomise protsessi. React pakkus ka tööriistu dünaamiliste ja tundlike liideste loomiseks, mis kuvatakse korrektselt erinevatel seadmetel. Autorid on valitud tööriistadega rahul ja usuvad, et need oskused võivad tulevikus kasulikud olla.

Arenduskeskkonnana kasutati Visual Studio Code'i ning tänu mugavale liidesele, arvukatele laiendustele ja autorite varasemale kogemusele täitis Visual Studio Code oma ülesande.

Docker'i tööriista kasutamine projekti tagarakenduse, andmebaasi ja kasutajaliidese juurutamiseks osutus väga heaks lahenduseks ja jättis positiivse mulje. Kuid projektide juurutamisega oli seotud ka raskusi, mis tekkisid vähesest Docker'i kasutamise kogemusest ja see protsess võttis plaanitud rohkem aega. Positiivsest küljest rääkimas need raskused võimaldasid töö autoritel parandada oma oskusi Docker'i kasutamisel ja projekti juurutamisprotsessi mõistmisel. Tänu Docker'i laialdasele populaarsusele professionaalses arendajate kogukonnas oli suhteliselt lihtne leida vajalikku teavet ja ressursse tekkivate küsimuste lahendamiseks, mis võimaldas saavutada projektis soovitud tulemusi.

GitLab'i kasutamine platvormina loodud veebirakenduse koodi majutamiseks oli lõputöö autorite jaoks esimene valik. Tänu varasemale positiivsele kogemusele olid autorid selle funktsionaalsuse ja funktsioonidega kursis. GitLab sai ootuspäraselt suurepäraselt hakkama kõigi ülesannetega, mis tegi projekti elluviimise protsessi meeskonna jaoks palju lihtsamaks. Platvorm pakkus vajalikke vahendeid tõhusaks versioonikontrolliks, koostööks ja edusammude jälgimiseks, mis muutsid töö produktiivseks ja tulemuslikuks.

#### **4.1.1 Nõuded**

Algusest peale ja kogu protsessi vältel olid projekti nõuded selgelt ja täpselt määratletud, vältides muudatusi äriloogikas ja tagades, et töö lõpetati kõrge standardiga, ilma et oleks vaja teha muudatusi. Meeskond koos juhendaja ja kaasjuhendajaga püüdis objektiivselt hinnata autorite tehnilisi võimeid ja võtta arvesse olemasolevat aega, seades kõige olulisemad ülesanded tähtsuse järjekorda. Kõikide nõuete realiseerimine osutus siiski keerulisemaks ja aeganõudvamaks kui algselt eeldati, mistõttu tuli kasutada alternatiivseid, kuigi madalama kvaliteediga funktsionaalseid lahendusmeetodeid, näiteks ettevõtte kataloogi lisamiseks kandideerimise registreerimise loogikat ei saanud veebilehel realiseerida ning seetõttu leidsid autorid probleemile lahenduse lisades veebilehel lingi Google'i vormile, mille kaudu saab esitada ettevõtte kandidatuuri. Lõpuks viisid autorid kõige kriitilisemad ülesanded edukalt lõpule, samas kui madalama prioriteediga ülesanded lükati edasi edasiseks täpsustamiseks tulevikus.



## **4.1.2 Arhitektuur**

### **4.1.2.1 Andmebaasi arhitektuur**

Tänu ärioloogika nõuetekohasele kavandamisele algstaadiumis olid kõik võtmesuhted täpselt määratletud vastavalt projekti nõuetele. Andmebaasi struktuur ja objektide vahelised seosed jäid muutumatuks, mis säästis meeskonda muudatuste tegemisest edasise arenduse käigus. See hoidis ära liigse ajakulu ja ressursside raiskamise, võimaldades arendajatel keskenduda projekti muude aspektide täiustamisele ja optimeerimisele.

### **4.1.2.2 Tagarakenduse arhitektuur**

MVC-mustri valimine projekti arhitektuurilahenduseks osutus arendusmeeskonna jaoks mõistlikuks otsuseks. Esiteks oli projekti autoritel juba varasem kogemus selle mustriga töötamisel, mis võimaldasid neil arendustegevust kiirendada. Teiseks vastas MVC kõigile projekti nõuetele, kuna rakendusloogika, kasutajaliides ja andmehaldus on selgelt eraldatud. Selline eraldamine tegi rakenduse hooldamise ja täiustamise protsessi lihtsamaks ning suurendas ka meeskonna tootlikkust.

MVC kasutamine võimaldas meeskonnal maksimeerida selle mustri eeliseid, näiteks koodi taaskasutamist ja lihtsustatud integratsiooniprotsesse, mis aitasid kaasa tõhusamale tööprotsessile. Arhitektuur toetas muudatuste lihtsat tegemist süsteemi üksikutes komponentides, minimeerides riski, et rakenduse teiste osade funktsionaalsus kannatab. Selle tulemusena suutis meeskond tagada projekti parema juhitavuse ja kontrolli, mis oli ülesannete edukaks täitmiseks väga oluline.

Lisaks sellele hõlbustas MVC kasutamine tõhusat koostööd meeskonnas, kuna muster võimaldas tööd jagada mooduliteks, mille kallal võisid eri meeskonnaliikmed samaaegselt töötada. Seega MVC mitte ainult vastas projekti tehnilistele nõuetele, vaid parandas ka meeskonnasiseseid protsesse, muutes projektitöö organiseeritumaks ja produktiivsemaks.

### **4.1.2.3 Kasutajaliidese arhitektuur**

Veebirakenduse kasutajaliidese projekti arhitektuur, mis kasutab React raamistikku, kujutab endast hästi korraldatud struktuuri. Kuna autoritel puudus varasem kogemus tolle raamistikuga, kerkisid esile mõningad raskused. Raskuste hulka kuulusid: ajakulu

erinevate osade õppimisele algtasemest, sõltuvuste lisamine NPM-pakettide kaudu ja projekti juurutamine läbi Docker'i. Kõige keerulisem oli aga kasutajaliidese projektis filtreerimisfunktsiooni arendamine, et see toimiks õigesti erinevaid filtreerimise kategooriaid kasutades.

Komponendid `assets`, `components` ja `pages` olid hästi struktureeritud ja kergesti hoomatavad. Komponentide jagamine võimaldas autoritel loogiliselt grupeerida funktsionaalsust ja kasutada koodi korduvalt, mis parandas projekti hooldatavust. Nende komponentide abil oli liidese loomine ja kohandamine lihtsam.

Redux kui olekuhalduse tööriist osutus võimsaks vahendiks andmete töötlemisel rakenduses. See võimaldas tõhusalt korraldada ja värskendada ülemaailmset olekut, tagades ühtse juurdepääsu andmetele erinevates rakenduse osades.

Router tagas kasutajaliidese navigeerimise, võimaldades hõlpsalt hallata marsruute ja lehtede vahel liikumist, mis võimaldas luua kasutajasõbraliku ja arusaadava kasutajaliidese.

Teiselt poolt aga tekitasid mõned arhitektuuri aspektid raskusi. Näiteks komponentide kujundamine võttis rohkem aega, kuna kogemuse puuduse tõttu kirjutati igale komponendile stiilid eraldi. See võiks olla tõhusam teatud stiiliraamistikku kasutamisel.

Üldiselt osutus React'i projekti arhitektuur efektiivseks ja funktsionaalseks, kuigi mõned aspektid ei olnud täielikult ära kasutatud kogemuse puudumise tõttu, mis tõi kaasa rohkem ajakulu kasutajaliidese arendamiseks kui oodatud.

#### **4.1.3 Testimine**

Kogu veebirakenduse testimise meetodiks valitud manuaalsel testimisel olid omad plussid ja miinused. Ühest küljest võimaldas see meeskonnal säästa aega, mis oleks kulunud automaatsete testiskriptide väljatöötamisele. Teisest küljest kulus palju vaeva kõigi rakenduse funktsioonide käsitsi testimiseks, mis suurendas testimisele kuluvat aega. Kuigi käsitsi testimine ei ole kõige tõhusam viis, kuna on oht, et mõned detailid jäävad tähelepanuta, osutus see lähenemisviis väikese projekti puhul vastuvõetavaks. Sellest hoolimata tunnistab meeskond, et isegi sel juhul ei ole see kõige parem variant.

Oluline samm oli väliste kasutajate kaasamine rakenduse testimisse, mis võimaldas saada objektiivset tagasisidet otsestelt kasutajatelt. See testimismeetod aitas välja selgitada kasutajate tegeliku arusaama ja reaktsiooni veebirakendusele, mis, nagu tulemused näitasid, võeti tänu oma lihtsusele, informatiivsusele ja atraktiivsele kujundusele hästi vastu.

## 4.2 Veebirakenduse vastavus nõuetele

Selles peatükis selgitatakse, millisel määral vastab arendatud veebirakendus 3.2 peatükis määratletud funktsionaalsetele ja mittefunktsionaalsetele nõuetele.

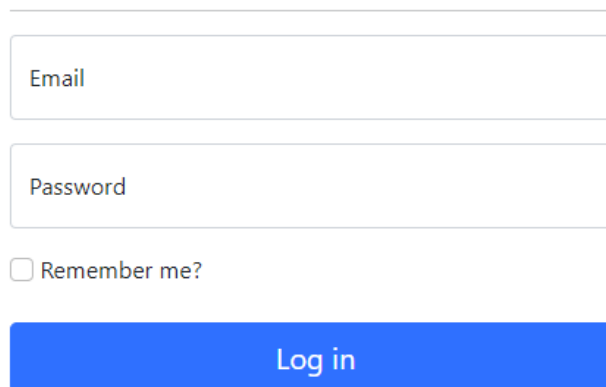
### 4.2.1 Funktsionaalsetele nõuetele vastavus

Täidetud said järgmised nõuded:

- Administraator saab logida sisse ja ainult pärast muuta, uuendada, lisada või kustutada andmeid (Joonis 25).

## Log in

Use a local account to log in.



Email

Password

Remember me?

Log in

Joonis 25. Administraatori vaade sisselogimisel

- Administraator saab lisada süsteemi GovTech'i ettevõtete või lahendustega seotud infot (Joonis 26).

## Create

### Company

Name  
New company

RegistrationCode  
153354

WebsiteUrl  
www.new-company.com

NatureOfBusiness  
The new company is the newest company of all the cor

GovTechApproved

Country Name  
Japan

Business Models  
C2C  
C2B  
B2G  
B2C2G

Sdgs  
No Poverty  
Zero hunger  
Good health and well-being  
Quality education

Sectors  
Agriculture and farming  
Banking and finance  
Construction and engineering  
Education and training

[Create](#)

[Back to List](#)

Joonis 26. Ettevõtte loomise lehe vaade administraatori jaoks

- Administraator saab muuta GovTech'i ettevõtete või lahendustega seotud andmeid (Joonis 27).

# Edit

## Company

Name

New company

RegistrationCode

153354

WebsiteUrl

www.new-company.com

NatureOfBusiness

The new company is the newest company of all the con

GovTechApproved

Country Name

Japan

Business Models

B2B

B2C

C2C

C2B

Sdgs

No Poverty

Zero hunger

Good health and well-being

Quality education

Sectors

Agriculture and farming

Banking and finance

Construction and engineering

Education and training

Save

[Back to List](#)

Joonis 27. Ettevõtte redigeerimislehe vaade administraatori jaoks

- Administraator saab kustutada GovTech'i ettevõtet ja lahendust kataloogist (Joonis 28).

## Delete

Are you sure you want to delete this?

Company

<b>Name</b>	New company
<b>RegistrationCode</b>	153354
<b>WebsiteUrl</b>	www.new-company.com
<b>NatureOfBusiness</b>	The new company is the newest company of all the companies
<b>GovTechApproved</b>	<input checked="" type="checkbox"/>
<b>Country</b>	Japan
<b>BusinessModelCompanies</b>	<ul style="list-style-type: none"><li>C2B</li></ul>
<b>SdgCompanies</b>	<ul style="list-style-type: none"><li>Zero hunger</li><li>No Poverty</li></ul>
<b>SectorCompanies</b>	<ul style="list-style-type: none"><li>Education and training</li><li>Agriculture and farming</li></ul>
<b>Solutions</b>	

[Delete](#) | [Back to List](#)

Joonis 28. Ettevõtte kustutamise lehe vaade administraatori jaoks

- Kasutaja ja administraator saavad vaadata GovTech'i ettevõtetega seotud infot (Joonis 29, Joonis 30, Joonis 31, Joonis 32).

## Companies

[Create New](#)

Find by name:  [Search](#) | [Back to full List](#)

Name	RegistrationCode	WebsiteUrl	NatureOfBusiness	GovTechApproved	Country	Business Model	Sdg	Sector	Solutions
New company	153354	www.new-company.com	The new company is the newest company of all the companies	<input checked="" type="checkbox"/>	Japan	<ul style="list-style-type: none"><li>C2B</li></ul>	<ul style="list-style-type: none"><li>Zero hunger</li><li>No Poverty</li></ul>	<ul style="list-style-type: none"><li>Education and training</li><li>Agriculture and farming</li></ul>	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Company5	4444	company5.ee	comp5	<input type="checkbox"/>	Israel	<ul style="list-style-type: none"><li>B2B</li></ul>	<ul style="list-style-type: none"><li>No Poverty</li></ul>	<ul style="list-style-type: none"><li>Banking and finance</li><li>Solution10</li></ul>	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>

1 2

Joonis 29. Ettevõtte nimekirja vaade administraatori jaoks

## Details

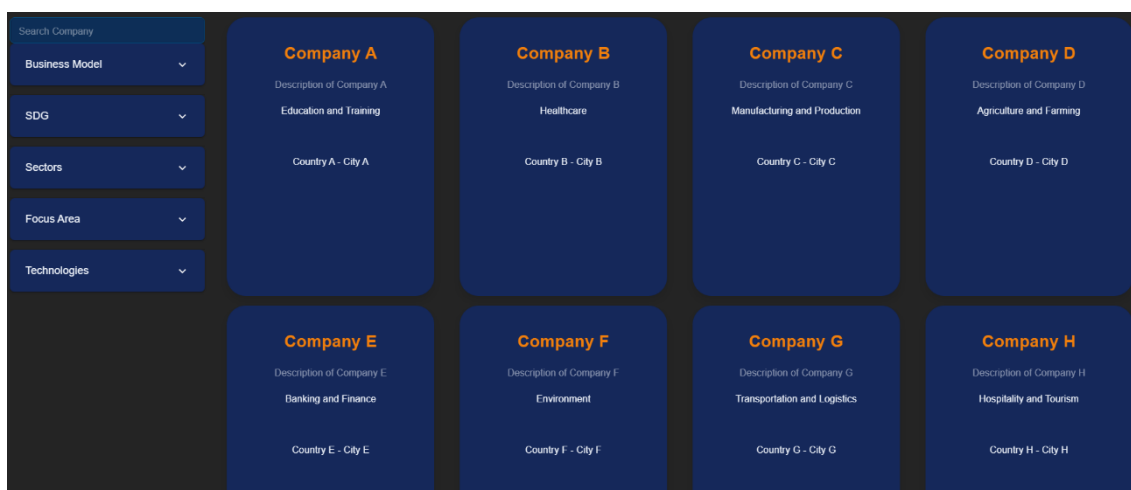
### Company

<b>Name</b>	New company
<b>RegistrationCode</b>	153354
<b>WebsiteUrl</b>	www.new-company.com
<b>NatureOfBusiness</b>	The new company is the newest company of all the companies
<b>GovTechApproved</b>	<input type="checkbox"/>
<b>Country</b>	Japan
<b>BusinessModelCompanies</b>	<ul style="list-style-type: none"> <li>C2B</li> </ul>
<b>SdgCompanies</b>	<ul style="list-style-type: none"> <li>Zero hunger</li> <li>No Poverty</li> </ul>
<b>SectorCompanies</b>	<ul style="list-style-type: none"> <li>Education and training</li> <li>Agriculture and farming</li> </ul>

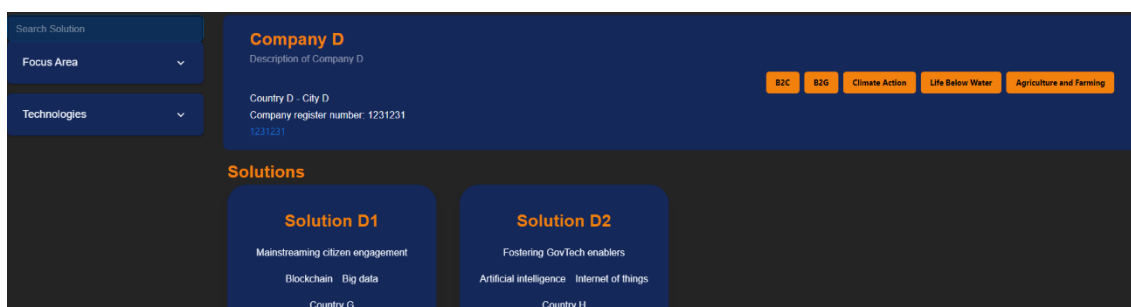
#### Solutions

[Edit](#) | [Back to List](#)

Joonis 30. Ettevõtte infolehe vaade administraatori jaoks



Joonis 31. Ettevõtete lehe vaade kasutaja jaoks

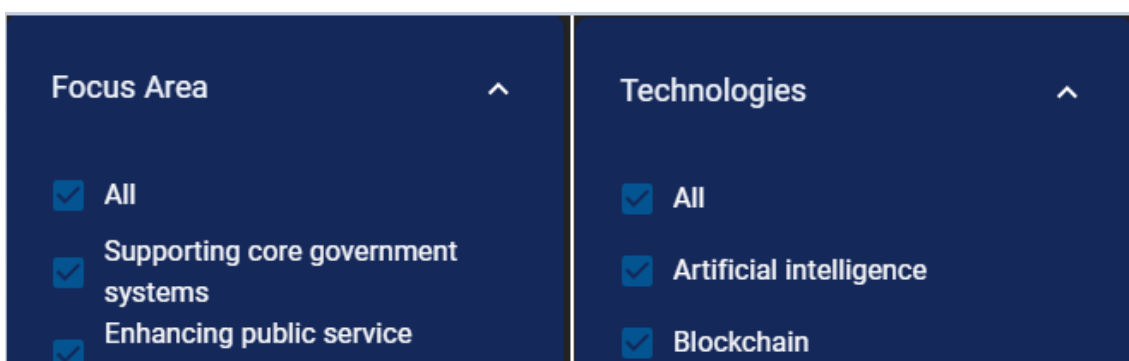


Joonis 32. Ettevõtte infolehe vaade kasutaja jaoks

- Kasutaja saab filtreerida GovTech'i ettevõtete kataloogi eelmääratletud kategooriate järgi (Joonis 33, Joonis 34).

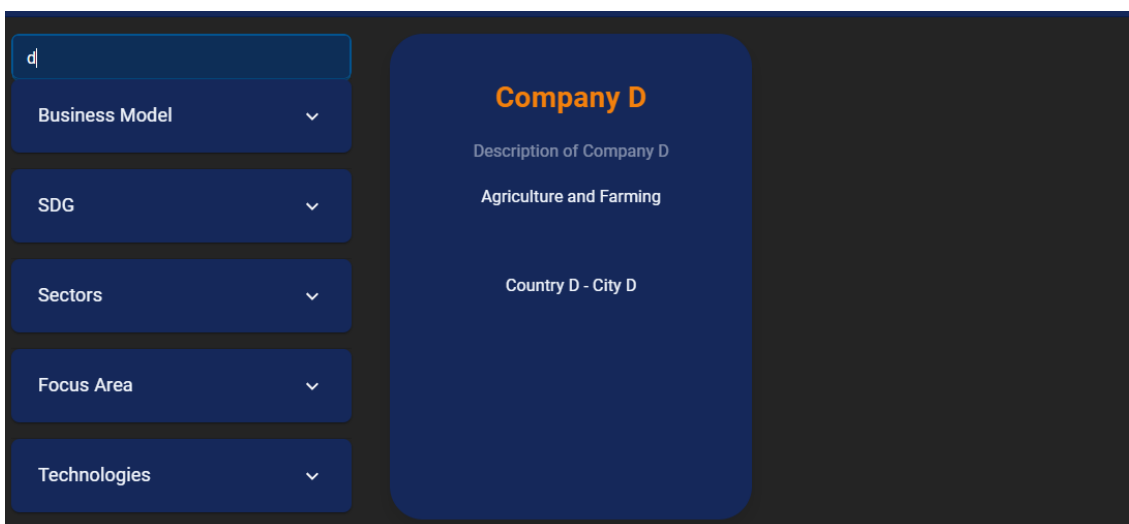


Joonis 33. Ettevõtete filtreerimine kasutaja jaoks



Joonis 34. Lahenduste filtreerimine kasutaja jaoks

- Kasutaja ja administraator saavad huvitava GovTech'i ettevõtte või lahenduse leidmiseks kasutada otsinguriba (Joonis 35, Joonis 36).



Joonis 35. Ettevõtte otsinguvõimalus kasutaja jaoks





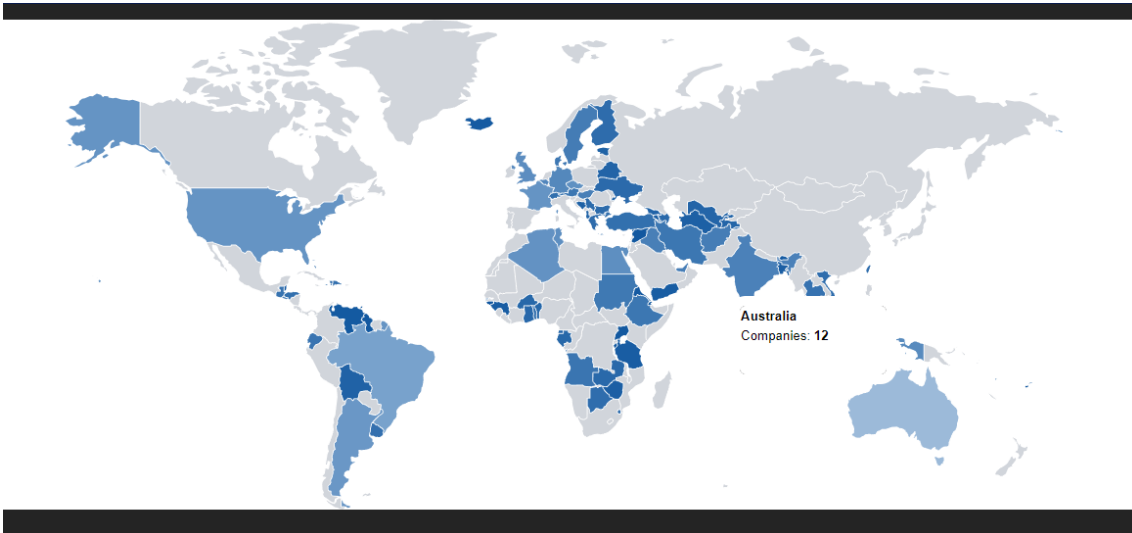
Joonis 36. Lahenduse otsinguvõimalus kasutaja jaoks

- Kasutaja saab vaadata erinevate kategooriate andmeid sisaldavaid statistilisi tabelleid (Joonis 37).



Joonis 37. Statistika ettevõtete ja nende lahenduste kohta kasutaja jaoks

- Kasutaja saab vaadata maailmakaardist GovTech'i ettevõtete arvu riigiti (Joonis 38).



Joonis 38. Maailmakaardi vaade kasutaja jaoks

- Kasutaja võib esitada oma ettevõtte kandidatuuri GovTech'i ettevõtete kataloogi lisamiseks (Joonis 39).

## GovTech Catalogue

Do you have a candidate that isn't in the GovTech catalogue yet?

Please fill out the form and we will review your application.

Edenemise salvestamiseks [logige Google'isse sisse](#). [Lisateave](#)

\* Viitab kohustuslikule küsimusele

Company name \*

Teie vastus

Company website \*

Teie vastus

Contact person

Teie vastus

Additional information

Teie vastus

Saada ära
Tühjenda vorm

Joonis 39. Google'i vorm uute kandidaatide registreerimiseks kasutajate jaoks

Projekti nõuete täitmine esitas meeskonnale mitmeid väljakutseid. Kõige lihtsamalt olid administraatori poolel andmetöötlusega seotud ülesanded, millega autorid said edukalt ja ilma raskusteta hakkama. Otsingufunktsioonid olid samuti hõlpsasti rakendatavad.

Raskem ülesanne oli aga statistika kuvamine maailmakaardil ja graafikute abil, sest meeskond ei olnud varem selliste ülesannetega kokku puutunud ning võimalike rakendusmeetodite uurimine nõudis märkimisväärset pingutust ja aega. Lisaks sellele läks kõige rohkem aega filtreerimise rakendamiseks.

Samuti ei olnud ideaalselt täidetud nõue, et kasutajad saaksid taotleda oma ettevõtte lisamist GovTech'i ettevõtete kataloogi. Soovituslikult oleks tulnud luua spetsiaalne küsimustik otse veebirakenduse sees ja saadud vastused oleks tulnud salvestada andmebaasi. Kuna see ülesanne ei olnud aga esmatähtis, otsustasid autorid jätta ajutiselt olemasoleva lahenduse, mida saaks tulevikus vajadusel täiustada. Otsustati lisada kasutajaliidesele link Google'i vormile, kuhu saab jätta ettevõtte kandidatuuri kataloogi lisamiseks.

Lõpuks said kõik funktsionaalsed nõuded täidetud, kuid administraatori võimalus lisada, muuta ja kustutada ettevõtteid jäi kasutajaliidese projekti lisamata. Autorid ei pea seda puudust eriti tõsiseks ega veebirakenduse funktsionaalsust häirivaks, kuid administraatori kasutajaliidese olemasolu kasutajaliidese projektis oleks olnud eelistatav, et pakkuda administraatoritele mugavamalt ja meeldivamat kasutajakogemust.

#### **4.2.2 Mittefunktsionaalsetele nõuetele vastavus**

Täidetud said järgmised mittefunktsionaalsed nõuded:

- Veebirakenduse kasutajaliides on lihtne ja kasutajale arusaadav.
- Veebirakenduse kasutajaliides on kasutatav erinevates veebibrauserites.
- Veebirakenduse kasutajaliides on sama hästi rakendatud nii arvuti, sülearvuti kui ka nutitelefoni jaoks.

Mittefunktsionaalsete nõuete efektiivsust hinnati veebirakenduse kasutajaliidese kasutuselevõtu ja kolmandate osapoolte kasutajate tagasiside kogumise teel. Saadud tulemused näitasid, et veebirakendus on hästi kasutatav erinevates veebibrauserites ja

seadmetes ning kasutajaliidese võeti kasutajate seas hästi vastu. Samuti hinnati kõrgelt selle lihtsust, informatiivsust ja atraktiivset disaini.

### 4.3 Võrdlus olemasolevate lahendustega

Tabel 1. GovTech Catalogue võrdlus olemasolevate lahendustega

	<b>GovTech Catalogue</b>	<b>IPS-X</b>	<b>Interoperable Europe</b>	<b>Startup Estonia</b>
<i>Otsinguvõimalus</i>	Jah	Jah	Ei	Jah
<i>Filtrid</i>	Jah	Jah	Ei	Jah
<i>Sarnasuse % teiste lahendsutega</i>	Ei	Jah	Ei	Ei
<i>Kaardi saadavus</i>	Jah, maailma kaart	Jah, Euroopa kaart	Jah, Euroopa kaart, aga ebamugav kasutamiseks	Jah, näitab ettevõtte asukohta
<i>Statistika näitamine kaardil</i>	Jah, erinevates toonides	Jah, erinevates toonides	Jah, numbritega	Ei
<i>Statistika</i>	Jah, diagrammide kaudu	Jah, diagrammide kaudu	Ei	Ei
<i>Võimalus kandideerida</i>	Jah, kõik kasutajad saavad Google'i vormi täita	Jah	Jah	Jah, kui on olemas konto
<i>Ettevõtte päritolu</i>	Kogu maailm	Euroopa	Euroopa	Piiratud riikide arv
<i>Kasutajaliides</i>	Heal tasemel	Keskmisel tasemel	Keskmisel tasemel	Heal tasemel

Järgnevalt analüüsitakse tabelis (Tabel 1) esitatud andmeid, võrreldes saadud tulemust teiste olemasolevate lahendustega.

IPS-X lahenduses on otsingufunktsioon rakendatud kõige kõrgemal tasemel: kasutajad saavad otsida teavet mitte ainult juhtumi nime, vaid ka paljude seotud andmete, näiteks tüübi, kasutatud tehnoloogiate, geograafilise asukoha ja muude parameetrite järgi. StartUp Estonia pakub ka ulatuslikke otsinguvõimalusi, mis võimaldavad kasutajatel

leida teavet ettevõtte, investori, kasutajate nimede, nimekirjade ja uuenduste järgi, muutes otsingu väga paindlikuks. Kuigi autorite väljatöötatud lahendus piirdub otsinguga ainult ettevõtte nime või lahenduste järgi, mis on kitsam lähenemisviis kui teised analoogid, pakub see siiski põhilisi otsingufunktsioone, erinevalt Interoperable Europe'ist.

StartUp Estonia pakub kõige täiustatumaid filtreerimisvõimalusi, võimaldades kasutajatel valida ettevõtteid 14 erinevas kategoorias. Sellele järgneb IPS-X lahendus, mis pakub filtreerimist kümnes kategoorias, võimaldades seega head paindlikkust. Kolmandal kohal on GovTech Catalogue, mis pakub kasutajatele valikut viie kategooria vahel, mis samuti lihtsustab kataloogis navigeerimist, kuid vähemal määral. Interoperable Europe ei kasuta filtreid, mis võib piirata kasutaja kogemust asjakohase teabe otsimisel.

Funktsioon, mille abil saab võrrelda protsentuaalseid vasteid teiste lahendustega, on saadaval ainult IPS-X'i veebisaidil, mis on selle veebirakenduse oluline eelis. Kuigi autorid ei kavatsenud praeguses etapis seda funktsiooni oma lahendusse lisada, tasub selle rakendamise võimalust tulevikus kaaluda kui võimalikku täiustamist.

IPS-X'i ja GovTech Catalogue'i lahendustes kasutatud kaardid on nii visuaalselt kui ka funktsionaalselt sarnased: need näitavad statistikat eri toonides ning kui kursor asetatakse riigi kohale, kuvatakse selle nimi ja numbrilised näitajad. GovTech Catalogue'i lahenduse peamine erinevus ja eelis on siiski maailmakaardi kasutamine, erinevalt IPS-X'i piiratud Euroopa katvusest. Kuigi Interoperable Euroopa lahendus sisaldab ka statistikat kuvavat kaarti, ei ole see kasutusmugavusega iseloomulik. StartUp Estonia näitab ainult valitud ettevõtte asukohta.

Kõigist vaadeldud lahendustest pakuvad ainult IPS-X ja GovTech Catalogue GovTech'i lahendustega seotud statistikat. Mõlemad teevad seda diagrammide abil: IPS-X kasutab andmete üksikasjalikuks esitamiseks 17 erinevat diagrammi, GovTech Catalogue piirdub aga kolmeaga. See muudab IPS-X'i lahenduse informatiivsemaks ja funktsiooniderikkamaks võrreldes autorite loodud lahenduse tagasihoidlikuma lähenemisega.

Kandidaatide esitamise parim rakendamine on esitatud Interoperable Europe'i veebisaidil, kus kasutaja saab vormi täita otse veebisaidil. Autorite lahenduses toimub kandidaatide esitamine Google'i vormi täitmise kaudu, mis on vähem mugav võrreldes vormi

täitmisega otse veebisaidil. IPS-X'i kandidaadi esitamise lehekülg avab lingi projekti GitHub'ile koos esitamise juhistega, kus kasutaja peab ise looma kaks CSV-faili ja tegema pull request'i, mis võib olla keskmise kasutaja jaoks keeruline ja ebamugav [45]. StartUp Estonia'le taotluse esitamiseks on vaja olla registreeritud kasutaja, mis piirab selle funktsiooni kättesaadavust.

StartUp Estonia kasutajaliides on kvaliteetne, mugav ja ilus; iga detail on hoolikalt läbi mõeldud, mis näitab arendajate professionaalset lähenemist. Ka GovTech Catalogue lahenduse autorid on kasutajaliidese rakendanud heal tasemel, pöörates erilist tähelepanu visuaalsele komponendile ja püüdes muuta veebirakenduse kasutamist võimalikult arusaadavaks. IPS-X'i kasutajaliides on küll hõlpsasti kasutatav, kuid näeb ebahuvitatav välja, samas kui Interoperable Europe on küll atraktiivsema välimusega, kuid kaotab kasutatavuse lihtsuse osas.

GovTech Catalogue'i veebirakenduse ainulaadne omadus on selle ülemaailmne suunitlus: erinevalt ainult Euroopale mõeldud vastavatest rakendustest hõlmab see rakendus ettevõtteid üle kogu maailma.

Kuigi GovTech Catalogue ei ületa teisi lahendusi iga üksiku parameetri osas, on see oma omaduste kogumis optimaalne valik autorite seatud eesmärgi saavutamiseks. Ülemaailmne katvus, lihtne andmete visualiseerimine maailmakaardil ning integreeritud otsingu- ja filtreerimisfunktsioonid teevad sellest mitmekülgse tööriista, mis ühendab turul saadaolevate lahenduste parimad omadused. See muudab GovTech Catalogue'i parimaks lahenduseks kasutajatele, kes on huvitatud tõhusast ja mugavast juurdepääsust teabele GovTech'i ettevõtete kohta kogu maailmas.

#### **4.4 Hinnang projekti teostamise protsessi kohta**

Projekti elluviimine oli algusest lõpuni hoolikalt kavandatud ja teostatud. Lõputöö autorid pidasid tihedas koostöös juhendaja ja kaasjuhendajaga regulaarselt iganädalasi kohtumisi Teams'i rakenduses, kus nad seadsid eesmärgid järgmiseks nädalaks. See ajavahemik osutus piisavaks mitte ainult püstitatud ülesannete täitmiseks, vaid ka uute küsimuste ja ideede kogumiseks, mida seejärel koosolekul arutati. Koosolekute aja ja päeva valikul võeti alati arvesse kõigi osalejate mugavust ning kui keegi puudus, otsiti võimalust täiendavaks aruteluks.

Kohtumistel valitses sõbralik õhkkond, mis muutis koosolekud meeldivaks ja efektiivseks. Iga kohtumise alguses tutvustasid autorid saavutatud tulemusi, arutasid õnnestumisi ja parandamist vajavaid punkte. Seejärel andsid juhendajad oma hinnanguid ja soovitusi. Kaasjuhendaja pakkus välja ideid, kuidas parandada projekti ärioloogikat, sõnastades selgelt lõpptulemuse visiooni ja prioriteetidid, mis aitasid kujundada järgmiste nädalate kava.

Pärast ärioloogika arutamist jäid autorid põhijuhendajaga üksi tehnilise lahenduse üksikasjade arutlemiseks. Nende arutelude käigus planeeriti edasist tööd ja arutati projekti tehnilisi aspekte.

Igapäevaseks suhtlemiseks kasutasid autorid Messenger'i tekstisõnumite saatmiseks ja Discord'i veebikohtumisteks, säilitades tugevaid suhteid ja koostööd meeskonnas. Kohustuste jaotamisel võeti arvesse iga osaleja tugevusi, mis aitasid kaasa üldisele rahulolule ja tõhususele. Kuigi iga osaleja pühendas rohkem aega oma ülesannetele, kasutas meeskond sageli paarisprogrammeerimise tehnikaid, mis muutsid projekti kallal töötamise palju lihtsamaks.

Kõik püstitatud nõuded said täidetud projekti käigus, kuid mõned neist said aga ajutise lahenduse ja vajavad täiustamist, mis viitab mõningatele puudustele protsessi korralduses. Projekti autorid alahindasid vajaliku töö mahtu, mis ületas olemasolevate teadmiste lihtsat rakendamist. Oluline osa tööst hõlmas uute tehnoloogiate ja tehnikate õppimist ja integreerimist, mis osutus algselt eeldatust aeganõudvamaks. See oli peamine põhjus, miks meeskond ei suutnud kõiki kavandatud ülesandeid ettenähtud viisil ellu viia. Kogemus rõhutab, kui oluline on hoolikam planeerimine ja võimalike raskuste arvestamine uute teadmiste ja oskuste omandamise protsessis, mis võivad oluliselt mõjutada projekti edenemist.

Üldiselt oli meeskond projekti korralduse ja tulemustega rahul, kuid mõned nõuded said ajutise lahenduse. See ei võimalda nimetada protsessi täiuslikuks, kuid peegeldab paljude ülesannete edukat täitmist ja kvaliteetse toote loomist.

## 4.5 Probleemid

Projekti kirjutamisel ei puutunud autorid kokku tõsiste probleemidega. Meeskonnatööd arvestades tuleb märkida, et autorid on juba mitu korda koos meeskondades töötanud, mistõttu ei tekkinud konflikte ega arusaamatusi. Tööprotsess kulges sujuvalt ja jagunes autorite vahel õiglaselt. Suurimaks probleemiks osutus oma jõudude ja oskuste mitte just kõige parem jaotus, mille tõttu autoritel ei õnnestunud kõiki projekti nõudeid täita soovitud viisil. Lisaks tekkis autoritel projekti arendusprotsessi käigus probleeme, mis olid seotud React'i ja JavaScript'i kasutamise kogemuse puudumisega. Kõige rohkem aega võttis kasutajaliidese projektis filtreerimisfunktsiooni arendamine, sest tuli jälgida, et erinevatest kategooriatest filtreid valides oleksid kõik kategooriad arvesse võetud ja filtreerimistulemuste kuvamisel lingitud.

## 4.6 Edasiarendus

Loodud projekt täidab põhifunktsioone, mis on vajalikud veebirakenduse kasutamise alustamiseks. See on aga alles esialgne versioon, mis nõuab täiendusi, mis parandaksid veebirakenduse kasutuskogemust ja lisaksid projektile kõik vajalikud funktsioonid.

Vaatamata asjaolule, et autoritel õnnestus täita kõik esitatud nõuded, sai üks nõuetest vaid ajutise lahenduse. Projekti edasiarendamiseks ja mugavamaks funktsionaalsuseks on vaja luua kasutajaliidese rakendatud võimalus oma ettevõttega kandideerida, ilma et peaks täitma Google'i vormi.

Lisaks ülaltoodud võimalusele projekti edasiseks arendamiseks märgivad autorid ka vajadust viia administraatori vaated üles esiotsa kasutajaliidese projekti, mis muudaks administraatoripoolse andmete lisamise ja muutmise protsessi oluliselt lihtsamaks ja mugavamaks. Teine oluline aspekt on automaattestide lisamine, kuna autorite loodud versioon kasutab ainult käsitsi testimist.

Üldiselt hindavad autorid saadud tulemust hea alusena täieliku veebirakenduse arendamiseks, millel on suur potentsiaal olla kasulik GovTech'i valdkonnas huvitatud kasutajatele ja teadlastele.



## 4.7 Ajalogid ja meeskonnaliikmete panus

Projekti autorite vahelise suhtluse ja ülesannete jaotuse hindamisel andsid meeskonnaliikmed hindeks 0, mis tähendab, et igatühe jõupingutused ja ajakulu on võrdsed. Hoolimata sellest, et üks osalejatest vastutas rakenduse tagarakenduse eest ja teine tegeles kasutajaliidesega, oli nende töö kogu projekti jooksul tihedalt seotud. Nad kasutasid aktiivselt paariprogrammeerimise taktikat, mis võimaldas neil koos töötada probleemide lahendamisel ja üksteise toetamisel. Selline koostöö mitte ainult tugevdas meeskonnatööd, vaid tagas ka kõigi võrdse osaluse ja panuse kõigis projekti aspektides, hõlbustades teadmiste ja kogemuste jagamist osalejate vahel.

Allpool saab leida meeskonna Toogl Track'i kaudu saadud ajalogide kokkuvõtet (Joonis 40).



Joonis 40. Meeskonna ajalogid

### 4.7.1 Anastasiia ajalogide kokkuvõte

Anastasiia Oprysk'i tegevused nädala kaupa on kirjeldatud järgnevas tabelis (Tabel 2).

Tabel 2. Anastasiia ajalogide kokkuvõte

Õppenädal	Tehtud töö
12.02-18.02	Lõputöö teema otsimine ja valimine.
19.02-25.02	Lõputöö teema otsimine ja valimine.
26.02-03.03	Lõputöö teema formuleerimine ja ülesandepüstituse koostamine.

04.03-10.03	Tutvumine lõputöö teemaga, materjali otsimine ja analüüsimine.
11.03-17.03	Tutvumine lõputöö teemaga, materjali otsimine ja analüüsimine, projekti elluviimise läbimõtlemine (tööriistade valimine).
18.03-24.03	Teema arutamine koos lõputöö juhendajaga, eesmärkide ja nõuete väljatöötamine.
25.03-31.03	Koosolek juhendaja ja kaasjuhendajaga. Tutvumine teemaga. Olemasolevate lahenduste otsimine ja analüüsimine.
01.04-07.04	Koosolek juhendaja ja kaasjuhendajaga. Arutelu projekti funktsionaalsete nõuete üle, diagrammide koostamine.
08.04-14.04	Koosolek juhendaja ja kaasjuhendajaga. Projekti loomine GitLab'is, töö alustamine tagarakendusega, paarisprogrammeerimine. Kasutajaliidesega tutvumine.
15.04-21.04	Koosolek juhendaja ja kaasjuhendajaga. Töö tagarakendusega. Kasutajaliidese õppimine.
22.04-28.04	Dokumentatsioon. Töö tagarakendusega. Kasutajaliidese õppimine. Tagarakenduse ja andmebaasi juurutamine Docker'is.
29.04-05.05	Dokumentatsioon. Töö tagarakendusega, paarisprogrammeerimine. Kasutajaliidese õppimine.
06.05-12.05	Koosolek juhendaja ja kaasjuhendajaga. Töö tagarakenduse ja kasutajaliidese, paarisprogrammeerimine. Dokumentatsioon.
13.05-19.05	Koosolek juhendaja ja kaasjuhendajaga. Töö tagarakenduse ja kasutajaliidese, paarisprogrammeerimine. Dokumentatsioon. Projekti tagarakenduse ja kasutajaliidese ühendamise. Projekti testimine.

#### 4.7.2 Marija ajalogide kokkuvõte

Marija Talašmanova tegevused nädala kaupa on kirjeldatud järgnevas tabelis (Tabel 3).

Tabel 3. Marija ajalogide kokkuvõte

Õppenädal	Tehtud töö
-----------	------------

12.02-18.02	Lõputöö teema otsimine ja valimine.
19.02-25.02	Lõputöö teema otsimine ja valimine.
26.02-03.03	Lõputöö teema formuleerimine ja ülesandepüstituse koostamine.
04.03-10.03	Tutvumine lõputöö teemaga, materjali otsimine ja analüüsimine.
11.03-17.03	Tutvumine lõputöö teemaga, materjali otsimine ja analüüsimine, projekti elluviimise läbimõtlemine (tööriistade valimine).
18.03-24.03	Teema arutamine koos lõputöö juhendajaga, eesmärkide ja nõuete väljatöötamine.
25.03-31.03	Koosolek juhendaja ja kaasjuhendajaga. Tutvumine teemaga. Olemasolevate lahenduste otsimine ja analüüsimine.
01.04-07.04	Koosolek juhendaja ja kaasjuhendajaga. Arutelu projekti funktsionaalsete nõuete üle, diagrammide koostamine. Kasutajaliidesega tutvumine.
08.04-14.04	Koosolek juhendaja ja kaasjuhendajaga. Tagarakenduse projekti loomine GitLab'is, töö alustamine tagarakendusega, paarisprogrammeerimine. Kasutajaliidese õppimine.
15.04-21.04	Koosolek juhendaja ja kaasjuhendajaga. Kasutajaliidese õppimine. Dokumentatsioon.
22.04-28.04	Dokumentatsioon. Kasutajaliidese projekti loomine GitLab'is, töö alustamine kasutajaliidesega.
29.04-05.05	Töö kasutajaliidesega, paarisprogrammeerimine. Dokumentatsioon.
06.05-12.05	Koosolek juhendaja ja kaasjuhendajaga. Töö kasutajaliidesega, paarisprogrammeerimine. Dokumentatsioon.
13.05-19.05	Koosolek juhendaja ja kaasjuhendajaga. Töö kasutajaliidesega, paarisprogrammeerimine. Dokumentatsioon. Kasutajaliidese juurutamine. Projekti testimine.

## 5 Kokkuvõte

GovTech'i ettevõtted mängivad võtmerolli säästva arengu eesmärkide saavutamisel, arendades ja rakendades tehnoloogilisi lahendusi, mis viivad eesmärkide saavutamisele lähemale. GovTech'i ettevõtete kohta teavet leitakse erinevatest veebisaitidest, kuid see pole alati lihtne ning leitud teave pole täielik ja hõlpsasti kasutatav.

Lõputöö eesmärk oli töötada välja veebirakendus, mis oleks aluseks GovTech'i ettevõtete kataloogile kogu maailmas. Projekti peamine nõue oli, et veebirakendus peaks olema selge ja kergesti kasutatav. Lisaks oli oluline ülesanne luua praktiline ja sisult väärtuslik ressurss, mis suudaks tõhusalt teenindada GovTech'i valdkonnast huvitatud kasutajaid.

Eesmärgi saavutamiseks töötati välja veebirakendus, mis koosnes kolmest põhikomponendist: veebirakenduse tagarakendus, mis on kirjutatud C# keeles, kasutades ASP.NET Core MVC raamistikku, SQLite'i andmebaas ja rakenduse kasutajaliides, mis on rakendatud JavaScriptis, kasutades React.js raamistikku. Tulemuslikumaks tööks kasutasid autorid paarisprogrammeerimise metoodikat, mis kiirendas projekti elluviimise protsessi ja muutis lõpptulemuse kvaliteetsemaks.

Töö tulemusena suutis meeskond luua veebirakenduse, mis on toimiv GovTech'i ettevõtete kataloog. Kuigi kõiki esialgseid nõudeid ei suudetud täielikult täita, oli lõpptulemuseks kvaliteetne toode, mis on juba praegu kasutajate jaoks väärtuslik ressurss. Sellel projektil on suur potentsiaal edasiseks arendamiseks ja täiustamiseks, mis avab võimalused uute funktsioonide kasutuselevõtuks ja funktsionaalsuse laiendamiseks tulevikus. Lõppkokkuvõttes täideti püstitatud ülesanne ja saavutati eesmärk.

Meeskond oli rahul tehtud töö kõigi etappidega: alates olemasolevate lahenduste analüüsist kuni valmis veebirakenduse rakendamiseni. See projekt tõi osalejatele palju väärtuslikke teadmisi ja kogemusi, mis on kahtlemata kasulikud nende edasistes kutsetegevustes.

## Kasutatud kirjandus

- [1] Martin, 'The Sustainable Development Agenda', United Nations Sustainable Development. [Võrgumaterjal]. Available: <https://www.un.org/sustainabledevelopment/development-agenda/>. [Kasutatud 30.04.2024].
- [2] 'How Digitization Can Make Government More Equitable', GovTech. [Võrgumaterjal]. Available: <https://www.govtech.com/gov-experience/how-digitization-can-make-government-more-equitable>. [Kasutatud 26.04.2024].
- [3] 'The Future of Govtech', ISACA. [Võrgumaterjal]. Available: <https://www.isaca.org/resources/isaca-journal/issues/2022/volume-2/the-future-of-govtech>. [Kasutatud 30.04.2024].
- [4] 'Rider: The Cross-Platform .NET IDE from JetBrains', JetBrains. [Võrgumaterjal]. Available: <https://www.jetbrains.com/rider/>. [Kasutatud 10.03.2024].
- [5] 'Visual Studio Code - Code Editing. Redefined'. [Võrgumaterjal]. Available: <https://code.visualstudio.com/>. [Kasutatud 28.04.2024].
- [6] 'Overview of ASP.NET Core MVC'. [Võrgumaterjal]. Available: <https://learn.microsoft.com/en-us/aspnet/core/mvc/overview?view=aspnetcore-8.0>. [Kasutatud 08.03.2024].
- [7] 'SQLite | JetBrains Rider', JetBrains Rider Help. [Võrgumaterjal]. Available: <https://www.jetbrains.com/help/rider/SQLite.html>. [Kasutatud 10.03.2024].
- [8] 'React'. [Võrgumaterjal]. Available: <https://react.dev/>. [Kasutatud 20.04.2024].
- [9] 'Docker: Accelerated Container Application Development'. [Võrgumaterjal]. Available: <https://www.docker.com/>. [Kasutatud 24.04.2024].
- [10] 'Backend Development Complete Guide', GeeksforGeeks. [Võrgumaterjal]. Available: <https://www.geeksforgeeks.org/backend-development/>. [Kasutatud 30.04.2024].
- [11] 'Introduction to SQLite', GeeksforGeeks. [Võrgumaterjal]. Available: <https://www.geeksforgeeks.org/introduction-to-sqlite/>. [Kasutatud 30.04.2024].
- [12] 'Vue.js'. [Võrgumaterjal]. Available: <https://vuejs.org/>. [Kasutatud 10.05.2024].

- [13] 'Stack Overflow Developer Survey 2023', Stack Overflow. [Võrgumaterjal]. Available: [https://survey.stackoverflow.co/2023/?utm\\_source=social-share&utm\\_medium=social&utm\\_campaign=dev-survey-2023](https://survey.stackoverflow.co/2023/?utm_source=social-share&utm_medium=social&utm_campaign=dev-survey-2023). [Kasutatud 06.05.2024].
- [14] 'JavaScript Tutorial', GeeksforGeeks. [Võrgumaterjal]. Available: <https://www.geeksforgeeks.org/javascript/>. [Kasutatud 08.04.2024].
- [15] K. Thorndyke, 'What Is CSS Used For?', Codecademy Blog. [Võrgumaterjal]. Available: <https://www.codecademy.com/resources/blog/what-is-css-used-for/>. [Kasutatud 30.04.2024].
- [16] H.-P. Cameron, J. B. Stephen, ja C. Meredith, 'What is Docker and How Does It Work?', IT Operations. [Võrgumaterjal]. Available: <https://www.techtarget.com/searchitoperations/definition/Docker>. [Kasutatud 30.04.2024].
- [17] 'The most-comprehensive AI-powered DevSecOps platform'. [Võrgumaterjal]. Available: <https://about.gitlab.com/>. [Kasutatud 08.05.2024].
- [18] 'Unify the DevSecOps lifecycle with GitLab'. [Võrgumaterjal]. Available: <https://about.gitlab.com/stages-devops-lifecycle>. [Kasutatud 08.05.2024].
- [19] 'Toggl Track: Time Tracking Software for Any Workflow'. [Võrgumaterjal]. Available: <https://toggl.com/>. [Kasutatud 19.05.2024].
- [20] 'Sign In | Microsoft Teams'. [Võrgumaterjal]. Available: <https://www.microsoft.com/en-us/microsoft-teams/log-in>. [Kasutatud 20.05.2024].
- [21] 'Messenger'. [Võrgumaterjal]. Available: <https://www.messenger.com/t/3519127708135333/>. [Kasutatud 20.05.2024].
- [22] 'Discord - A New Way to Chat with Friends & Communities', Discord. [Võrgumaterjal]. Available: <https://discord.com/login>. [Kasutatud 20.05.2024].
- [23] 'Miro | The Visual Workspace for Innovation', Miro. [Võrgumaterjal]. Available: <https://miro.com/>. [Kasutatud 17.03.2024].
- [24] 'What is Miro? Get To Know Our Visual Workspace For Innovation', Miro. [Võrgumaterjal]. Available: <https://miro.com/what-is-miro/>. [Kasutatud 12.05.2024].
- [25] 'Intelligent Diagramming', Lucidchart. [Võrgumaterjal]. Available: <https://www.lucidchart.com>. [Kasutatud 18.04.2024].
- [26] 'DeepL Translate: The world's most accurate translator'. [Võrgumaterjal]. Available: <https://www.deepl.com/translator>. [Kasutatud 20.05.2024].

- [27] ‘How does DeepL work?’ [Võrgumaterjal]. Available: <https://www.deepl.com/en/blog/how-does-deepl-work>. [Kasutatud 15.05.2024].
- [28] ‘ChatGPT’. [Võrgumaterjal]. Available: <https://chatgpt.com>. [Kasutatud 19.05.2024].
- [29] Evgeni Paromov, ‘React JS 18 - Самый актуальный полный курс - YouTube’, (2023). [Video YouTube]. Available: <https://www.youtube.com/playlist?list=PLMlifixDLpB1CAAtLxKmie5FfGZkNx5K5XX>. [Kasutatud 17.04.2024].
- [30] IT-KAMASUTRA, ‘Курс “React JS - путь самурая 1.0”, уроки, практика - YouTube’, (2020). [Video YouTube]. Available: <https://www.youtube.com/playlist?list=PLcvhF2Wqh7DNVy1OCUpG3i5lyxyBWhGZ8>. [Kasutatud 26.04.2024].
- [31] ‘React Tutorial’. [Võrgumaterjal]. Available: <https://www.w3schools.com/REACT/DEFAULT.ASP>. [Kasutatud 25.04.2024].
- [32] Vladilen Minin, ‘React JS с Нуля - Курс для начинающих БЕЗ ВОДЫ [2024]’, (2023). [Video YouTube]. Available: <https://www.youtube.com/watch?v=kz23xxukY5s>. [Kasutatud 22.04.2024].
- [33] ‘GovTech Market Size, Share, Trends | Growth Report 2031’. [Võrgumaterjal]. Available: <https://www.businessresearchinsights.com/market-reports/govtech-market-102878>. [Kasutatud 30.04.2024].
- [34] ‘IPS-X - Innovative Public Services Explorer’. [Võrgumaterjal]. Available: <https://ipsoeu.github.io/ips-explorer/>. [Kasutatud 19.05.2024].
- [35] ‘Contributors to ipsoeu/ips-explorer’, GitHub. [Võrgumaterjal]. Available: <https://github.com/ipsoeu/ips-explorer>. [Kasutatud 10.05.2024].
- [36] ‘Catalogue of GovTech initiatives | Joinup’. [Võrgumaterjal]. Available: <https://joinup.ec.europa.eu/collection/govtechconnect/catalogue-govtech-initiatives>. [Kasutatud 19.05.2024].
- [37] ‘Joinup | Joinup’, Interoperable Europe. [Võrgumaterjal]. Available: <https://joinup.ec.europa.eu/collection/joinup>. [Kasutatud 19.05.2024].
- [38] ‘Startups & scaleups | Estonia Innovation Ecosystem’. [Võrgumaterjal]. Available: <https://ecosystem.startupestonia.ee/custom.startups>. [Kasutatud 19.05.2024].
- [39] ‘our story’, Startup Estonia. [Võrgumaterjal]. Available: <https://startupestonia.ee/about-us/our-story/>. [Kasutatud 16.05.2024].

- [40] 'MVC (Model View Controller) Architecture Pattern in Android with Example', GeeksforGeeks. [Võrgumaterjal]. Available: <https://www.geeksforgeeks.org/mvc-model-view-controller-architecture-pattern-in-android-with-example/>. [Kasutatud 18.04.2024].
- [41] 'react-jvectormap', npm. [Võrgumaterjal]. Available: <https://www.npmjs.com/package/react-jvectormap>. [Kasutatud 13.05.2024].
- [42] 'Highcharts - Interactive Charting Library for Developers', Highcharts Blog | Highcharts. [Võrgumaterjal]. Available: <https://www.highcharts.com/blog/homepage21may/>. [Kasutatud 28.05.2024].
- [43] 'Treemap | Highcharts'. [Võrgumaterjal]. Available: <https://highcharts.com/docs/chart-and-series-types/treema>. [Kasutatud 28.04.2024].
- [44] 'Heatmap | Highcharts'. [Võrgumaterjal]. Available: <https://highcharts.com/docs/chart-and-series-types/heatmap>. [Kasutatud 28.04.2024].
- [45] 'ipsoeu/ips-explorer'. Innovative Public Services Observatory. [Võrgumaterjal]. Available: <https://github.com/ipsoeu/ips-explorer>. [Kasutatud 19.05.2024].
- [46] Putu Adi Guna Permana, 'Scrum Method Implementation in a Software Development Project Management', (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 6, No. 9, 2015. [Võrgumaterjal]. Available: [https://eresearch.stikom-bali.ac.id/admin/files/publikasi\\_files/406b9aa48840ed8d357af87f8bb62530.pdf](https://eresearch.stikom-bali.ac.id/admin/files/publikasi_files/406b9aa48840ed8d357af87f8bb62530.pdf).
- [47] K. Beck ja C. Andres, Extreme Programming Explained Second Edition, Addison-Wesley Professional, 2004, pp. 13-55.
- [48] Jun Ying Zhou, 'Functional Requirements and Non-Functional Requirements: A Survey'. M.S thesis, Concordia University, Montreal, Quebec, Canada, April 2004 [Võrgumaterjal]. Available: <https://spectrum.library.concordia.ca/id/eprint/7991/1/MQ91163.pdf>
- [49] Tarvo Treier, 'itb2203 2023 Workout App', GitLab. [Võrgumaterjal]. Available: <https://gitlab.cs.taltech.ee/tarvo.treier/itb2203-2023-workout-app>. [Kasutatud 24.04.2024].



## **Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks<sup>1</sup>**

Autorid Anastasiia Oprysk ja Marija Talašmanova

1. Annavad Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „GovTech’i ettevõtete veebikataloogi arendus“, mille juhendaja on Bahdan Yanovich
  - 1.1. reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
  - 1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Oleme teadlikud, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitame, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

20.05.2024

---

<sup>1</sup> Lihtlitsents ei kehti juurdepääsupiirangu kehtivuse ajal vastavalt üliõpilase taotlusele lõputööle juurdepääsupiirangu kehtestamiseks, mis on allkirjastatud teaduskonna dekaani poolt, välja arvatud ülikooli õigus lõputööd reprodutseerida üksnes säilitamise eesmärgil. Kui lõputöö on loonud kaks või enam isikut oma ühise loomingu tegevusega ning lõputöö kaas- või ühisautor(id) ei ole andnud lõputööd kaitsvale üliõpilasele kindlaksmääratud tähtjaks nõusolekut lõputöö reprodutseerimiseks ja avalikustamiseks vastavalt lihtlitsentsi punktidele 1.1. ja 1.2, siis lihtlitsents nimetatud tähtaja jooksul ei kehti.

## Lisa 2 – Anastasiia Oprysk eneseanalüüs

Töö valmis koostöös Marija Talašmanova'ga ning panus projekti kõikidesse etappidesse oli võrdne. Minu peamine panus projekti oli järgmine:

- Olemasolevate lahenduste analüüsimine
- Nõuete määratlemine
- Kood kõikide rakenduse osades
- Tagarakenduse ja andmebaasi paigutamine Dockeri konteinerisse
- JavaScript'i ja React'i õppimine
- Projekti kõigi osade ühendamise
- Dokumentatsioon

Projekti teema, mida juhendaja koos kaasjuhendajaga pakkus, oli minu jaoks täiesti uus ja tundmatu. Enne lõputööga tegelemist ei olnud ma kunagi kokku puutunud mõistega "GovTech'i ettevõtte", kuigi olin tuttav säästva arengu eesmärkidega. See projekt tutvustas mulle palju uut ja kasulikku teavet, mis laiendas oluliselt minu teadmisi ja arusaamist GovTech'i ettevõtete tähtsusest tänapäeva maailmas. Mõistmine, et minu jõupingutused võivad lisada tõelist väärtust, andis mu tööle erilise tähenduse ja motivatsiooni.

Marija Talašmanova valimine lõputöö kirjutamise paariliseks oli minu jaoks ilmselge otsus. Olles klassikaaslased, töötasime sageli grupitöodes koos, viies läbi nii väikseid kui ka suuremaid projekte. Töö Marijaga on alati olnud võrdne partnerlus, mis põhineb aususel ja vastastikusel austusel, ning meie lõputöö ei olnud erandiks. Kogu protsessi vältel olime kontaktis, toetasime üksteist, küsisime nõu, tulime üksteisele vajadusel appi. Töö lõputöö kallal toimus lihtsas ja sõbralikus õhkkonnas, mis lõi ideaalsed tingimused produktiivseks koostööks.

Veebirakenduse rakendamine osutus keerulisemaks ja aeganõudvamaks ülesandeks, kui alguses eeldasin. Vaatamata minu teadmistele tagarakenduse arendusest ei tundunud

ülesanne lihtne, kuid lõpptulemusega jäin siiski rahule. Suurimad raskused tekkisid mitte niivõrd uute kasutajaliidese arenduse vahendite õppimisega, vaid nende praktilise rakendamisega. Siiski muutis paariprogrammeerimise metoodika koodi kirjutamise protsessi palju lihtsamaks, mis viis lõpuks hea tulemuseni. Minu jaoks olid kõige lihtsamad ülesanded olemasolevate lahenduste analüüs ja projekti nõuete määratlemine, sest juhendaja ja kaasjuhendaja aitasid meid palju. Vaatamata koodi kirjutamisega seotud raskustele oli see etapp lõputööst minu jaoks kõige põnevam.

Tööprotsessi käigus sain kindlasti kasu õpingute käigus saadud teadmistest. Need oskused mängisid võtmerolli rakenduse tagarakenduse loomisel, projekti selgete nõuete määratlemisel ja dokumentatsiooni kirjutamisel. Samas puudusid teadmised veebirakenduse kasutajaliidese rakendamisel, mis tegi töö raskemaks, kui see oleks võinud olla.

Kokkuvõttes jäin lõpptulemusega ja oma panusega projekti rahule. Lõputööga töötamine andis mulle väärtuslikke kogemusi, mitte ainult uute teadmiste omandamise, vaid ka nende praktilise rakendamise osas. Projekt osutus minu jaoks äärmiselt väärtuslikuks kogemuseks, sest sain minna ideest kuni toimiva veebirakenduse loomiseni. Kuna me ei seadnud teineteise ees piire, sain end proovida erinevates rollides ning mõista oma eelistusi ja ametialaseid püüdlusi.

## Lisa 3 – Marija Talašmanova eneseanalüüs

Bakalaureusetöö kallal tegin koostööd Anastasiia Opryskiga ning minu panus oli projekti järgmistes osades:

- Olemasolevate lahenduste analüüsimine
- Nõuete määratlemine
- Diagrammide koostamine
- JavaScript'i ja React'i õppimine
- Kood kõikide rakenduse osades
- Kasutajaliidese projekti juurutamine
- Dokumentatsioon

Valitud teema oli minu jaoks täiesti uus, kuna ma polnud varem kokku puutunud GovTech'i definitsiooniga ega teadnud selle algatuse olemasolust. See avas minu jaoks täiesti uue ja olulise teadmise ülemaailmsetest arenguprogrammidest. Oli huvitav selle teemaga tutvuda ja arendada veebirakendust, mis on küll väike, aga siiski osa GovTech'i eesmärkide saavutamise edusammudest.

Koostöö Anastasiia Oprysk'iga oli mugav ja tulemuslik, mida oli oodata, arvestades, et oleme ülikoolis paljudes kursustes samas grupis olnud. Projekti väljatöötamisel oli meeskonnas tunda vastastikust abivalmidust ja moraalselt tuge, vastastikust mõistmist ja vastutustunnet täidetavate ülesannete eest, mistõttu olen rahul oma paarilise valikuga bakalaureusetöös.

Veebirakenduse nullist arendamine osutus keeruliseks ülesandeks, mida raskendas meeskonnal kogemuste puudumine kasutajaliidese arendamises. Selle raskusega toimetulekuks otsustasime Anastasiiaga jagada vastutust projekti tagarakenduse ja kasutajaliidese vahel, tänu millele Anastasiia keskendus tagarakenduse arendamisele, mina aga õppisin kasutajaliidese jaoks valitud tööriistu. Seda osa oma tööst hindan kõige keerulisemaks ja aeganõudvamaks, kuna kogemuste puudumisel ei osutunud lihtsaks

arendada kasutajaliidest, mis vastaks seatud nõuetele. Projekti lihtsaimaks osaks nimetaksin suhtlust kliendiga, olemasolevate lahenduste analüüsi ja meeskonnatööd, aga kõige huvitavamaks osaks oli uue programmeerimiskeele õppimine.

Töö käigus osutusid loomulikult kasulikuks ülikoolis omandatud teadmised, mis aitasid mul panustada tagarakenduse ja diagrammide koostamisse, ning mis mitte vähem oluline, oskus õppida ja omandatud oskusi praktikas rakendada.

Kasutajaliidese arendamise oskust pean üheks oma bakalaureusetöö peamiseks tulemuseks, aga veelgi olulisem on kogemus veebirakenduse nullist arendamisel ja suhtlemisel kliendiga, mis on väga oluline oskus, kui soovin edaspidi töötada arendajana.

Üldiselt hindan oma panust bakalaureusetöösse positiivselt. Projekti vältel sain proovida end erinevates rollides ja näha veebirakenduse arendamist erinevatest külgedest, mis omas omakorda tähtsust minu kindlaksmääramisel, millises IT-valdkonnas ma sooviksin arenedada.

## **Lisa 4 – Kaasjuhendaja Alena Labanava tagasiside**

Technology companies have more actively developed technological solutions tailored for the public sector in recent years. This situation has even led to the idea of creating a single GovTech marketplace.

An initial step to understand the current state of the entrepreneurial landscape in GovTech is creating a catalogue of GovTech companies and their solutions, which has been created within the current bachelor's thesis project. It allows for searching the companies based on the industry, technology types, SDGs and business models. In this way, specialized agencies can find potentially relevant solutions for their needs. Also, the catalogue generates statistics, which can present interest for researchers specialized in digital government, innovation and entrepreneurship.

The important mission of the project has been recognized by e-Estonia Briefing Centre, which has published an article mentioning the development of the GovTech catalogue. The article has also been shared on social media (e-Estonia LinkedIn profile)<sup>1</sup> and e-Estonia newsletter<sup>2</sup>.

This is, to the best of my knowledge, the first attempt to understand and categorise private sector solutions meant for public administration needs on the global scale. Existing attempts to map GovTech companies are either geographically (only the EU) and/or technologically limited (only AI).

I am satisfied with the outcome (the final version of the website) as well as with the collaboration process. The students have worked consistently, considering my feedback. The weekly progress sessions have made the feedback loop between the students and me very short, which has contributed to the quality of the result. In addition to addressing my vision of the website, the students were proactive in the sense of offering their own ideas

---

<sup>1</sup>[https://www.linkedin.com/posts/e-estonia\\_govtech-for-sustainable-development-goals-activity-7191073633979584512-UiNt/](https://www.linkedin.com/posts/e-estonia_govtech-for-sustainable-development-goals-activity-7191073633979584512-UiNt/)

<sup>2</sup> <https://e-estonia.com/govtech-for-sustainable-development-goals>

of features. The result of the students' work is going to be used not only for my own research, but hopefully will serve as a platform for governments to look for technological solutions as well as for companies to understand the competitors.

Alena Labanava, Early Stage Researcher, Next Gen Digital State research group

20.05.2024