

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Mariliis Sinivee 193885IABB

Reia Rõõmus 193701IABB

ETTEVÕTTESISESE TARKVARA KASUTAJALIIDESE TÄIENDAMINE

Bakalaureusetöö

Juhendaja: Liisa Jõgiste

MSc

Tallinn 2022

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: Mariliis Sinivee, Reia Rõõmus

18.05.2022

Annotatsioon

Meeskonnaprojekti eesmärgiks oli kuvada Pipedrive OÜ rakendusele Missions Tool igale arendusüksusele tema kohta käiv statistika. Antud veebirakendus on ainult ettevõtte siseseks kasutamiseks.

Antud lõputöö eesmärgiks on varasemalt autorite poolt arendatud veebirakendust edasi arendada ja lõpetada statistikavaade. Teiseks eesmärgiks on luua kasutajaliides, mille abil saab arendusprojekti looja uut projekti luua. Kolmandaks eesmärgiks on Missions Tool rakendusse luua mugav lahendus arendusprojekti inimeste kutsumiseks.

Lõputöö tulemusel valmis kasutajaliides uue projekti loomiseks ja inimeste kutsumiseks. Uus projekt salvestatakse projekti looja sisestatud andmete põhjal andmebaasi. Kõikidele projektidele loodi võimalus inimesi antud projekti kutsuda läbi Missions Tool rakenduse. Lisaks lõpetati ITB1706 aine raames seatud eesmärk kuvada iga arendusüksuse statistikat nende planeerimisvaates.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 62 leheküljel, 9 peatükki, 43 joonist, 8 tabelit.

Abstract

The authors began with the project as a team project internship and the goal was to improve the company's internally used software's front-end.

The aim of this thesis is to further develop the web application that was started, to display statistics about each development unit in Pipedrive OÜ on the Missions Tool application.

The main objectives of this thesis are to display certain statistics about each development unit, create a user interface that allows creating a new project in the Missions Tool application, and create a convenient solution in the Missions Tool application to invite developers to participate in the projects.

The reason for the development is to make Pipedrive's resource management easier and quicker. From the statistics it is possible to see when the best time for a new project is. Creating a new project in Missions Tool is crucial to making resource planning easier, it is better to have all the data in one place rather than multiple places. Inviting developers to the projects from Missions Tool is convenient if the project manager does not know the developers that are available.

As a result of the thesis, the authors completed all three objectives. They managed to meet the goal set within the team project internship to display all the necessary statistics and in addition finished the user interface for creating a new project and also for inviting developers to the projects.

The thesis is in Estonian and contains 62 pages of text, 9 chapters, 43 figures, 8 tables.

Lühendite ja mõistete sõnastik

<i>CRM</i>	<i>customer relationship management</i> ehk kliendisuhete juhtimine on integreeritud infosüsteem, mida kasutavad ettevõtted, et oma müügieelset- ja järgset tegevust planeerida ning juhtida [1]
<i>Department</i>	Osakond
Eesrakendus	<i>Front-end</i>
Eesrakendusearendaja	<i>Front-end developer</i>
<i>Git</i>	DevOps'i ehk tarkvaraarenduse kultuuri tööriist, avatud lähtekoodiga versioonihaldussüsteem [2]
ITB1706	Tallinna Tehnikaülikooli äriinfotehnoloogia õppekava õppeaine Infosüsteemide arendamise meeskonnaprojekt: tellimus (2021/2022 sügis) ainekood
Kasutajalugu	Kindla kasutaja vaatenurgast esitatud süsteemi omaduse või funktsionaalsuse kirjeldus [3]
Kompleksarendaja	<i>Full-Stack developer</i> , arendaja, kes on võimeline välja töötama süsteemi kõiki kihte või osi
<i>Launchpad</i>	Ettevõttes Pipedrive viibivad <i>launchpad</i> 'is arendajad, kes tegelevad igapäevaste tööülesannetega. Nendel arendajatel puuduvad arendusprojektid
<i>Mission</i>	Ettevõttes Pipedrive nimetatakse

	arendusprojekte <i>Mission</i> 'ideks, mis tuleneb inglise keelsest sõnavarast
Missions Tool	Veebirakendus, mida kasutatakse arendusprojektide planeerimisel
<i>On-call</i>	Arendajad, kes on <i>on-call</i> , tegelevad ööpäevaringselt rakenduse toimimises tekkivate probleemidega
Püsiprogrammeeritud	<i>Hard-coded</i> , tarkvaras teostatud koodiosa, mida lõppkasutajal pole võimalik muuta
Renderdamine	Andmemudelist kujutise loomise protsess arvutiprogrammide abil
Tagarakendus	<i>Back-end</i> , Tagarakendus
<i>Tribe</i>	Arendusüksus
URL	<i>Uniform Resource Locator</i> , üldine infoallika asukohamääraja ehk internetiaadress

Sisukord

1 Sissejuhatus	12
2 Ettevõtte ja veebirakenduse tutvustus	13
2.1 Ettevõtte tutvustus.....	13
2.2 Veebirakendus Missions Tool	13
3 Probleem ja projekti eesmärk	18
3.1 Probleemi kirjeldus ja lõputöö eesmärk.....	18
3.2 Metoodika	19
4 Loodava kasutajaliidese analüüs	20
4.1 Nõuete kaardistamine.....	20
4.1.1 Töötuba.....	20
4.1.1.1 Mariliis'i ideed	21
4.1.1.2 Reia ideed	25
4.1.2 Funktsionaalsed nõuded kasutajalugudena	30
4.1.2.1 Arendusüksuse statistika vaatamine	30
4.1.2.2 Arendusüksuse statistika ridade selgituste vaatamine.....	30
4.1.2.3 Projekti staatuse vaatamine	31
4.1.2.4 Projektide hoiatuse sisu vaatamine	31
4.1.2.5 Projekti detailse sisu vaatamine	31
4.1.2.6 Projekti loomine	32
4.1.2.7 Projekti inimeste kutsumine	32
4.1.2.8 Inimese detailse info vaatamine	33
4.1.3 Mittefunktsionaalsed nõuded	33
4.2 Eesrakenduse disain	33
4.2.1 Statistika prototüüp	33
4.2.2 Projekti lisamise vormi prototüüp.....	34
4.2.3 Projekti inimeste kutsumise prototüüp.....	36
4.3 Arendusmetoodika valik	37
4.3.1 Ekstreemprogrammeerimine	38
4.3.2 Scrum	40
4.4 Tehnoloogia valik	41
4.4.1 Raamistiku valik: React vs Vue	41

4.4.2	Programmeerimiskeele valik.....	44
4.4.3	Arenduskeskkonna valik	45
4.4.4	Koodihalduskeskkonna valik	46
4.5	Arhitektuur.....	47
4.6	Kasutatud mustrid	50
4.6.1	OOP - Pärilikkus	50
5	Töö tulemus	53
5.1	Planeerimise vaates statistika kuvamine.....	53
5.2	Uue projekti lisamine.....	58
5.3	Inimeste kutsumine	60
5.4	Võimalikud edasiarendused	63
5.4.1	Statistika vaate edasiarendused	63
5.4.2	Projekti loomine	64
5.4.3	Inimeste kutsumine projekti	64
5.4.4	Projekti muutmine	65
5.5	Tulu teenimisvõimalus.....	65
5.6	Hinnang projekti teostamisele	65
6	Testimine	68
6.1	Integratsioonitestimine.....	68
6.2	SonarQube	68
6.3	Snyk	69
7	Teostatud tööde logi	71
7.1	Mariliis Sinivee logi.....	71
7.2	Reia Rõõmuse logi.....	72
8	Tellija tagasiside	73
9	Kokkuvõte	74
	Kasutatud allikad	75
	Lisa 1 – Enda panuse kirjeldus ja eneseanalüüs, Mariliis Sinivee	80
	Lisa 2 – Enda panuse kirjeldus ja eneseanalüüs, Reia Rõõmus	82
	Lisa 3 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks	84

Jooniste loetelu

Joonis 1. "Missions" vaade	14
Joonis 2. "Tribe" vaade.....	15
Joonis 3. Arendusüksuse liikmete info kalendri vaates	16
Joonis 4. "People" vaade	17
Joonis 5. Veebilehe näide filtreerimisest [7]	22
Joonis 6. Veebilehe näide ressursside kuvamisest [8].....	23
Joonis 7. Veebilehe näide värvigammast [9].....	23
Joonis 8. Töötoa viimane etapp: lahenduse skitseerimine.....	25
Joonis 9. Veebilehe näide projektide ja ressursside eraldatusest [10].....	26
Joonis 10. Veebilehe näide värvide järgi kategoriseerimisest. [11]	27
Joonis 11. Veebilehe näide projektide ja ressursside eraldatusest [11].....	28
Joonis 12. Töötoa viimane etapp: Reia lahenduse skitseerimine	29
Joonis 13. Statistikaandmete vaate prototüüp [12].....	34
Joonis 14. Projekti lisamisvormi prototüüp [12]	35
Joonis 15. Planeerimisvaade peale projekti lisamist [12].....	36
Joonis 16. Projekti inimeste kutsumise prototüüp [12]	37
Joonis 17. Vue komponentide näide.....	42
Joonis 18. Vue faili näide	43
Joonis 19. Programmeerimiskeelte kasutuse võrdlus [31]	44
Joonis 20. 2012. aasta koodihaldus keskkondade kasutamise võrdlus [38]	47
Joonis 21. Kolmetasandiline arhitektuur.	48
Joonis 22. Failide ülesehitus kaustades.	49
Joonis 23. KalendriInfo komponent.	51
Joonis 24. Põhikalendri komponent	51
Joonis 25. Kalendrimalli komponent.....	51
Joonis 26. Kalendri vaate komponent.....	52
Joonis 27. Statistika väljad ja andmed nädalate kaupa.	54
Joonis 28. Statistika veeru kirjeldus.	55
Joonis 29. Punane värvus, kui arendajate arv on üks või null.....	55
Joonis 30. ITB1706 aine planeeritud „Planning” vaade	56

Joonis 31. Tegevuste stiili määramise koodinäide disaini loomiseks	57
Joonis 32. Ühtse rea kuvamise koodi näide disaini loomiseks.....	57
Joonis 33. Kastideks lõikamise koodinäide disaini loomiseks	58
Joonis 34. Loodud vorm uue projekti lisamiseks koos lisatud andmetega.....	58
Joonis 35. Vaikimisi inimeste kutsumise leht.	60
Joonis 36. Filtri rakendamine arendusüksuste järgi filtreerimiseks.	61
Joonis 37. Inimeste nimekiri valitud inimestega.	61
Joonis 38. Slack'i sõnumikast	62
Joonis 39. Statistika vaade koos “ <i>Loaned out</i> ” osaga.....	63
Joonis 40. Kalender projekti toimumisaja valimiseks.	64
Joonis 41. Slack’i korrektne sõnumikast.	65
Joonis 42. Mariliis Sinivee Toggle logi.....	71
Joonis 43. Reia Rõõmus Toggle logi.....	72

Tabelite loetelu

Tabel 1. Arendusüksuse statistika vaatamine	30
Tabel 2. Arendusüksuse statistika ridade selgituste vaatamine	30
Tabel 3. Projekti staatuse vaatamine	31
Tabel 4. Projektide hoiatuse sisu vaatamine.....	31
Tabel 5. Projekti detailse sisu vaatamine	31
Tabel 6. Projekti loomine	32
Tabel 7. Projekti inimeste kutsumine	32
Tabel 8. Inimese detailse info vaatamine	33

1 Sissejuhatus

Antud bakalaureusetöö on välja kasvanud ITB1706 infosüsteemide arendamise meeskonnaprojektist: tellimus (2021/2022 sügis) ainek, kui kuus tudengit, Mariliis Sinivee, Reia Rõõmus, Aleksei Kadeikin, Annabel Matkur, Sergei Vassiljev ja Lii Saluvere, asusid tööle ettevõttes Pipedrive OÜ. Tudengite eesmärgiks oli arendada edasi ettevõtte sisest veebirakendust Missions Tool.

Peale aine lõppemist otsustati jätkata töö tegemist lõputöö raames ning sellele saadi ka heakskiit nii kooli kui ka ettevõtte poolt. Koos asuti lahendama probleemi, kuidas muuta uute arendusprojektide planeerimine paremaks ning lihtsustada ressursside planeerimist nendeks. Hetkel puudub ettevõttel selleks ühtne tööriist ning iga arendusüksus kasutab uue projekti loomiseks endale meelepärast vahendit.

Meeskonnas on jaotatud ära ametipositsioonid ja tööülesanded. Annabel Matkur ja Lii Saluvere on tootejuhid, Aleksei Kadeikin ja Sergei Vassiljev on tagarakenduse arendajad ning antud lõputöö autorid Mariliis Sinivee ja Reia Rõõmus on eesrakenduse arendajad. Üheskoos töötati välja võimalik lahendus probleemile, mida hakati teostama. Töö autorite eesmärgiks on luua kasutajaliides tuginedes tootejuhtide loodud prototüübile.

Töös käsitletav probleem ja sellele lahenduse loomine on vajalik töötajatele uue projekti loomise protsessi lihtsustamiseks ning uue projekti planeerimisel aja kokkuhoidmiseks. Lisaks näevad lõputöö autorid ka antud rakenduse äritulu teenimisvõimalust, seda väljaspoole ettevõtet müües.

Antud bakalaureusetöö on jaotatud kuueks suuremaks peatükiks. Esimeses neist on kirjeldatud ettevõtet, kus rakenduse arendamine toimus ja kirjeldatakse rakendust Missions Tool, millele edasiarendust tehakse. Teises peatükis kirjeldatakse sügavamalt probleemi ning tuuakse välja lõputöö eesmärgid. Kolmandas peatükis analüüsitakse loodava eesrakenduse jaoks kasutatavaid tehnoloogiaid, kirjeldatakse nõudeid lahendusele ning tuuakse välja rakenduse arhitektuur. Neljandas peatükis näidatakse millisele tulemusele jõuti arenduses ning viiendas peatükis kirjeldatakse rakenduse testimist.

2 Ettevõtte ja veebirakenduse tutvustus

Antud peatükis kirjeldatakse ettevõtet Pipedrive'i, kus viidi läbi antud lõputöö arendus. Lisaks selgitatakse lahti, mis on rakendus Missions Tool, milleks seda kasutatakse ja milliseid andmeid see kuvab.

2.1 Ettevõtte tutvustus

CRM (*customer relationship management*) ehk kliendisuhete haldamise platvorm on integreeritud infosüsteem, mida kasutavad ettevõtted, et oma müügieelset ja müügijärgset tegevust planeerida ning juhtida. Pipedrive OÜ on esimene CRM-platvorm [1], mis on asutatud 2010. aastal. Pipedrive tegeleb müügiplatvormi teenuse võimaldamise ja arendamisega ehk CRM-lahenduse pakkumisega. Ettevõttel on 10 kontorit kaheksas riigis ning üle 900 töötaja 50 erinevast riigist ja nende teenust kasutab ligi 100 000 ettevõtet 175 riigis. Pipedrive on saanud mitmeid tunnustusi:

- 2016-2019 on Pipedrive saanud Eesti Unistuste Tööandja poolt Parima tööandja tiitli erasektoris.
- Financial Times on öelnud, et Pipedrive on üks Euroopa kiiremini kasvavaid ettevõtteid.
- Inc Magazine on öelnud, et see on üks parimaid ettevõtteid naistöötajate jaoks. [4]

2.2 Veebirakendus Missions Tool

Missions Tool on veebirakendus, mis on loodud ettevõtte Pipedrive arendajate poolt. Rakenduse eesmärgiks on kuvada ülevaadet Pipedrive'i töötajate ja nende arendusüksuste kohta ning anda infot käimasolevatest ja tulevastest arendusprojektidest. Rakendus kasutab andmete kättesaamiseks kahte olemasolevat veebiplatvormi: Bamboo [5] ja Confluence [6]. Platvorm Bamboo omab andmeid ettevõtte kõikide töötajate kohta ning platvorm Confluence omab andmeid käimasolevate ja tulevaste arendusprojektide kohta. Nende kahe veebilehe kaudu edastatakse andmed edasi Missions Tool'i, mis on jaotatud üheksaks vaateks. Nendeks on „Missions”, „Tribes”, „People”, „Programs”, „Ideas”, „Exploration”, „Badges”, „Statistics” ja „Hackathon”.

Järgnevalt tuuakse ülevaade Missions Tool'i kasutatavatest funktsionaalsustest, mida kasutasid lõputöö autorid oma töös.

Vaates „Missions” kuvatakse nimekiri kõikidest arendusprojektidest, mida Pipedrive'is nimetatakse missiooniks, tabelina. Antud tabelis on iga arendusprojekti kohta välja toodud järgnev info:

- arendusüksus ehk *tribe*, kes projekti eest vastutab ja sellega tegeleb,
- *mission lead* ehk arendusprojekti juht, kes vastutab kogu projektiga seotud tegevuste eest,
- *product manager* ehk tootejuht,
- projekti algus- ja lõppkuupäev,
- projekti hetkeolek, mis näitab millises järgus projekti käivitamisel ollakse. Olekuks võib olla mustand, ettevalmistamisel, teoksil, lõpetatud, lõpetamas, tagasilükatud, läbikukkunud või tühistatud.

Vaates „Missions” on võimalik igale arendusprojekti nimele klikkides näha hüpikaknas täiendavat infot projekti kohta, kus on eelnevalt mainitule välja toodud kõik arendusprojektis osalevad liikmed, projekti lühikirjeldus ja link Confluence'i [6], kus on olemas kogu info vastava projekti kohta. (Joonis 1)

The screenshot shows the 'Missions' tab in a software interface. At the top, there are navigation tabs: Ideas, Exploration, Missions (selected), Tribes, People, Planning, Badges, and Hackathon. Below the tabs, it says 'Showing 67 missions'. A table lists missions with columns for Mission, Focus area, Tribe, and a list icon. The table includes entries for Flo.Terry, Betsy_Boyer4, Nikolas.Batz65, Carmelo.Homenick, Dagmar.Swift81, Gerard38, Brett55, Linwood_Kuhlman56, Kelsie_Bernier68, and Delilah_Lubowitz. A detailed view for Brett55 is shown on the right, including the name, tribe (reprehenderit), location (Tallinn), dates (1 Dec - 22 Feb 2022), status (IN PREPARATION), and a list of team members: Rosina Reinger (PM), Johan Boyle (Developer), and Linwood Kihn (Developer). At the bottom of the detailed view are 'Interested' and 'Apply' buttons.

Mission	Focus area	Tribe	List icon
Flo.Terry	FUGA	reprehenderit	3
Betsy_Boyer4	AUTEM	repellendus	9
Nikolas.Batz65	AUTEM	consequatur	9
Carmelo.Homenick	AUT	accusamus	20
Dagmar.Swift81	AUTEM	voluptates	21
Gerard38	REPREHENDERIT	voluptatem	10
Brett55		reprehenderit	1
Linwood_Kuhlman56		blanditiis	1 Nov 2021
Kelsie_Bernier68		commodi	8 Nov 2021
Delilah_Lubowitz		reprehenderit	1 Nov 2021

Joonis 1. "Missions" vaade

Vaates „Tribe” kuvatakse tabelina nimekiri kõikidest ettevõtte arendusüksustest. Ettevõttes Pipedrive on igale arendusüksusele määratud nimi ja need kuvatakse tähestikulises järjekorras. Lisaks nimele, kuvatakse iga meeskonna kohta selle suurus, vabade kohtade arv arendusüksuses, inseneride arv, projektidel mitteosalevate töötajate arv, projektidel mitte osalevate arendajate arv ning meeskonnal käimasolevate ja tulevaste arendusprojektide arv. (Joonis 2)

Tribe	Size (free spots)	Engineers	Launchpad	Available	Missions	Launchpad / Mission
accusamus	13 (2)	12	9	4	●●	
alias	7 (0)	0	0	0		
aliquid	20 (0)	3	3	0		
animi	13 (5)	11	10	5	●●	
blanditiis	30 (0)	24	11	3	●●●●●	
commodi	18 (2)	14	9	6	●●●●●	
dignissimos	16 (2)	15	12	6	●●●	
distinctio	21 (0)	19	10	5		
dolorem	33 (0)	8	8	0		

Joonis 2. "Tribe" vaade

Konkreetse osakonna nime peale klikkides, avaneb uus vaade, kus on välja toodud kõik meeskonna liikmed, kes saavad osaleda mõnes arendusprojektis. Välja ei ole toodud näiteks meeskonnas olevad personalitöötajad või andmeanalüütikud, sest neil puudub otsene seos teostatava projektiga. Tabelis kuvatud meeskonna liikmete info on välja toodud kalendri vaatenähtena, kus on näha päevade kaupa, kas töötaja viibib puhkusel, osaleb arendusprojektis, on ta „on call” ehk lahendab tarkvaraga tekkinud probleeme ja on ööpäevaringselt kättesaadav või on tal mõni muu tegevus käsil. Lisaks on võimalik kalendri vaadet muuta päevadest nädalateks. (Joonis 3)

Navigation: Ideas | Exploration | Missions | Tribes | **People** | Planning | Badges | Hackathon

Engineering (348) | Product Management (114) | Engineering Operations (71) | Business Development (12) | G&A (91) | Marketing (53) | Sales (32) | Support (128) | Customer Success (48)

Search by person, mission, tribe etc. [Q]

Level [v] | Tribe [v] | Status [v] | Guilds [v]

Showing 346 people

Name	Badges	Profile
Zetta McGlynn		Yasmin Luetngen Junior Software Engineer - blanditiis
Zella Barton		Tartu Domenico_Harber@yahoo.com
Yoshiko Lemke		MISSION/LAUNCHPAD RATIO [Progress Bar]
Yasmine Toy		MISSION HISTORY (1) Marielle_Reilly20 FINISHED Developer 19 Jul - 23 Aug 2021
Yasmin Luetngen		PIPEDRIVE HISTORY (3) Joined blanditiis Tribe 2 Aug 2021
Winston Hodkiewicz		Joined totam Tribe 7 Jun 2021
Wilson Cormier		Joined Pipedriver 7 Jun 2021
Wilson Becker		
Willow Conroy		Front End Developer et Launchpad (203)

Joonis 4. "People" vaade

3 Probleem ja projekti eesmärk

Antud peatükis kirjeldatakse, milline on lõputöös lahendatav probleem, millised eesmärgid on probleemi lahendamiseks seatud ja milline on lõputöö läbiviimise meetoodika. Antud rakenduse arendamine algas koos kaastudengite, Aleksei Kadeikin, Sergei Vassiljev, Annabel Matkur ning Lii Saluvere, ITB1706 infosüsteemide arendamise meeskonnaprojekt: tellimus (2021/2022 sügis) aine raames.

3.1 Probleemi kirjeldus ja lõputöö eesmärk

Pipedrive OÜ ettevõttes puudub hetkel võimalus kõigil üheselt planeerida inimressursse arendusprojektidesse. Iga osakond planeerib ressursse kasutades erinevaid platvorme ning seetõttu puudub ülevaade kogu ettevõtte vabadest inimressurssidest. Uut projekti alustades on vajalike inimeste leidmine aeganõudev, mistõttu pikeneb ka reaalne arendusprojekti alustamise aeg.

Pipedrive OÜ ettevõttes on loodud eraldi tööriist Missions Tool, mis koondab enda alla kõikide arendusmeeskondade ja käimasolevate arendusprojektide info. Uute projektide registreerimiseks kasutatakse tööriista Confluence[6], mis on ühendatud Missions Tool'ga. Iga osakond kasutab ressursside planeerimiseks erinevaid platvorme ning need platvormid pole kõigile kättesaadavad. See tähendab seda, et puudub ülevaade kogu ettevõtte vabadest inimressurssidest, mis teeb uute projektide registreerimise keeruliseks. Lihtsustamaks Pipedrive'i töötajatel uutesse projektidesse inimesi leida, luuakse kasutajaliides, milles ühendatakse kaks funktsiooni:

1. kuvatakse ülevaade inimestest, kellel puuduvad arendusprojektid, koos nendega seonduva infoga,
2. võimalus luua läbi Missions Tool rakenduse uus arendusprojekt.

Kasutajaliidesele luuakse järgnevad funktsioonid:

1. Projekti loomine Missions Tool rakenduse kaudu.
2. Projekti muutmine Missions Tool rakenduse kaudu.
3. Projekti liikmete lisamine Missions Tool rakenduse kaudu, saates valitud arendajale kutse.

4. Planeerimisvaate kalendri muutmine ehk projektide toimumisaegade ning kestvuse muutmine otse kalendrist.

3.2 Metoodika

Antud bakalaureusetöö teema kasvas välja läbitud aine ITB1706 infosüsteemide arendamise meeskonnaprojekti teostamise tulemusena.

Antud lõputöös planeeritavad funktsionaalsused luuakse juba olemasolevale süsteemile ning seetõttu viiakse esmalt läbi juba loodud süsteemi analüüs. Selle käigus selgitatakse välja olemasolevad funktsionaalsused ja analüüsitakse loodud koodi korduvkasutatavuse võimalusi.

Kasutajalood luuakse koostöös tootejuhtide ja disaineriga ning selleks korraldatakse töötuba, mille käigus mõeldakse välja lahendus kahele probleemile:

1. Kasutajal puudub üldine ülevaade ressurssidest, mistõttu ei osata arendusprojekti planeerimiseks esialgset sobivat aega valida.
2. Kasutaja peab arendajaid kutsuma oma arendusprojekti eraldi ja tihti peale puudub teadmine vabadest arendajatest arendusüksuses.

Töötuba on lahti kirjeldatud peatükis 4.1.1 Töötuba.

Toetudes töötoas tehtud ettepanekutele probleemi lahendada, luuakse tootejuhtide poolt esmane prototüüp, mida testitakse Pipedrive töötajate peal. Saades üldise heakskiidu loodud disainile, antakse disain üle antud lõputöö autoritele, eesrakenduse arendajatele, kes hakkavad seda ellu viima. Juhul, kui arenduse ajal leitakse paremaid disaine prototüübile, viiakse kooskõlas tootejuhtidega muudatused sisse.

Uute lahenduste teostamisel kasutatakse ettevõtte Pipedrive'i poolt võimaldatud tehnoloogiaid ja arendusmetoodika pannakse arendajate enda poolt paika.

Arendusprotsessi käigus on arendajatele toeks ettevõtte poolsed mentorid, kes aitavad arendajaid tekkinud tehniliste probleemide korral ja suunavad lahendamisel õigele teele.

Viimase etapina tuuakse välja projekti tulevikuvaated ja analüüsitakse tehtud tööd.

4 Loodava kasutajaliidese analüüs

Järgnevas peatükis kirjeldatakse lahti kuidas toimus nõuete määramine antud projekti jaoks, millised funktsionaalsed nõuded on loodud kasutajalugude põhjal ning millised on arendusprojekti mittefunktsionaalsed nõuded. Lisaks tuuakse välja kolme vaate prototüübid, kirjeldatakse ja analüüsitakse arenduskeskkonna, raamistike, programmeerimiskeele, koodihoidmiskeskonna ja arendusmetoodika valikut. Viimasena on lahti kirjeldatud projekti arhitektuur.

Ettevõtte Pipedrive ja lõputöö autorite vahel on loodud tingimus, et loodud koodi ei tohi avaldada lõputöö kirjalikus osas. Seetõttu on kõik koodinäited toodud eesti keelsete nimedega, et ära hoida kirjutatud koodi avalikustamist.

4.1 Nõuete kaardistamine

Järgnevalt kirjeldatakse veebirakenduse funktsionaalsed ja mittefunktsionaalsed nõuded ning kirjeldatakse nende loomiseks tehtud töötoa protsessi. Lisaks tuuakse välja autorite ideed ja mõtted, mis aitasid kaasa töötoas loodud lahendusteni jõudmiseks.

4.1.1 Töötuba

Nõuete kogumiseks viidi tootejuhtide Annabel Matkuri ja Lii Saluvere poolt läbi töötuba, mille eesmärk oli leida parim teostus järgnevatele lahendustele:

- arendusprojekti looja saab luua mugavalt uut arendusprojekti ja rakendus pakub talle automaatselt sobiva aja projekti tegemiseks,
- arendusprojekti looja saab kutsuda projektiks vajalikke inimesi läbi rakenduse arendusprojekti osalema.

Kõik töötoas osalejad said pakkuda välja oma ideid ja mõtteid ning järgnevalt tuuakse välja lõputöö autorite ideed ja mõtted lahenduste teostamiseks eelnevale kahele punktile.

Töötuba koosnes kolmest etapist.

Töötoa esimeses etapis tuli leida juba olemasolevaid lahendusi erinevate ressursside planeerimiseks ja selgitada miks antud näide leidjale meeldib.

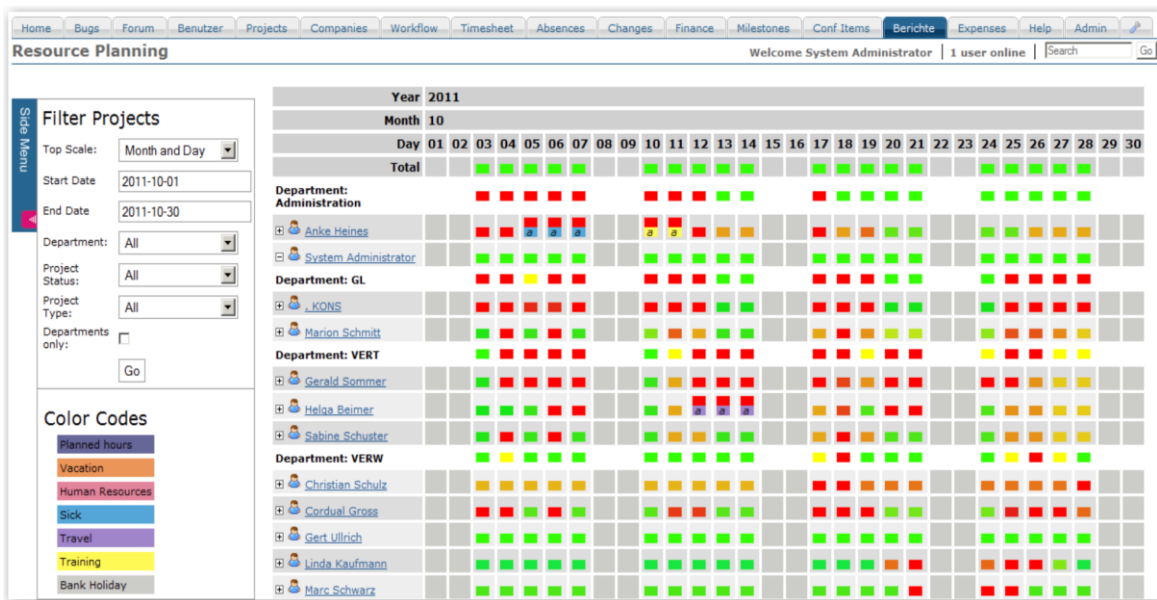
Peale esimeste ideede ja näidete otsimist tuli järgmiseks etapiks visualiseerida ühele paberile kaheksa pilti. Ülesande mõte oli joonistada välja kõik ideed, mis peas tekkisid probleemi peale mõeldes ja kuidas need on seotud probleemi lahendamisega. Peale kiiret esmaste mõtete visualiseerimist, tuli järgnevalt joonistada detailne plaan probleemi lahendamiseks, mis koosnes kolmest pildist. Iga pildi juures on kirjeldatud etapid, mis selgitavad täpsemalt pildi sisu. Ülesande lahendamisel pidi toetuma eelnevalt joonistatud kaheksale pildile.

Töötoa viimases etapis tuli kõikidel osalejatel hinnata teiste töid ja see läbi valiti välja parim lahendus probleemidele.

4.1.1.1 Mariliis'i ideed

Esimeses etapis tõi Mariliis välja kolm näidet veebilehtede kohta, kus kõigil oli kujutatud kalendrivaates ressurside kuvamist.

Joonisel 5 on välja toodud kalendrivaates töötajate tegevused. Igale tegevusele on rakendatud oma värv. Nii on näha, et näiteks inimesel, kes viibib kuu kümnendal päeval puhkusel, on kuupäev märgitud kollaseks. Näide on autori poolt välja valitud, sest lehe vasakule küljele on loodud filtreerimine, mis lihtsustaks inimeste leidmist projektidesse ka Missions Tool rakenduses. Joonisel 5 võimaldab filtreerimine ettevõtte projekte filtreerida algus- ja lõppkuupäevade, osakondade, staatuse ja projekti tüübi järgi. Sellist filtreerimise lahendust saaks kasutada vabade inimeste ja vajaminevate ametipositsioonide leidmisel. (Joonis 5)



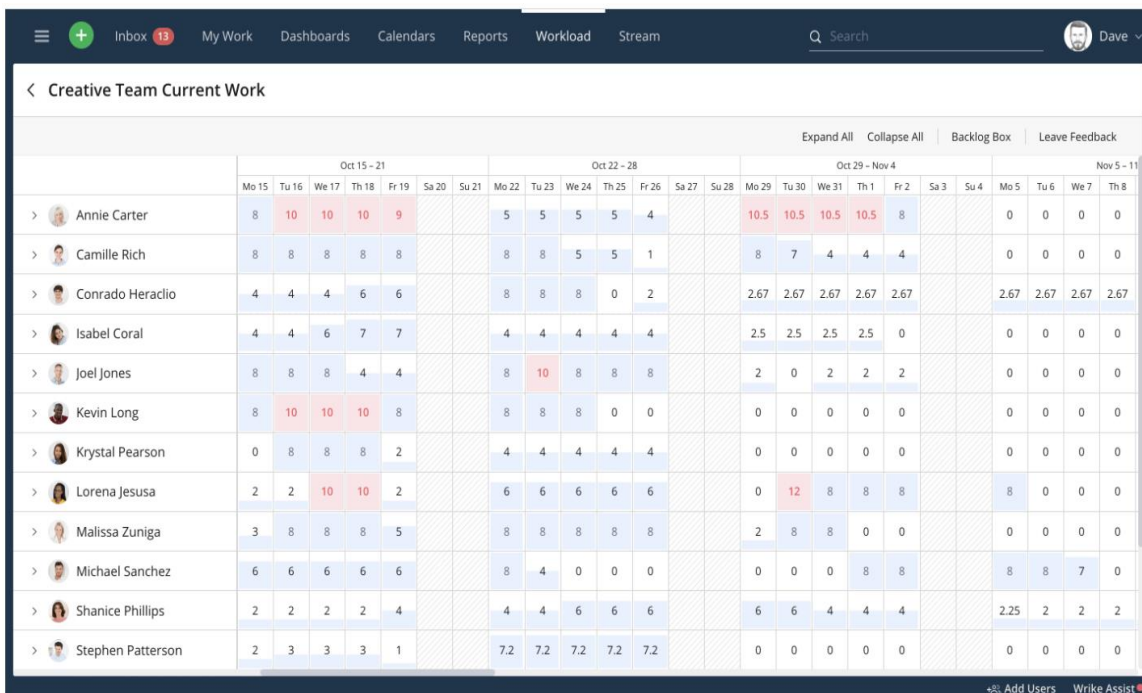
Joonis 5. Veebilehe näide filtreerimisest [7]

Joonisel 6 on taas kuvatud kalendrivaates inimeste info. Autorile jäi silma antud näite värvilahendus allikate kuvamiseks, mis annab arusaama, kas ressurs on täielikult vaba, kas talle on määratud ülesandeid, kuid samal ajal saab ta veel osaleda mõnes projektis, või on ta täielikult tööle pühendunud ega oma aega uute ülesannete jaoks. Info kuvamiseks on kasutatud kolme värvi tooni: valge, heleroosa ja tumeroosa. Valge annab märku, et inimesel puuduvad hetkel tööülesanded, heleroosa näitab, et talle saab määrata olemasolevate ülesannete juurde uusi ja tumeroosa näitab, et inimene on täiesti kinni ega võta uusi ülesandeid vastu. Näide andis autorile idee kasutada sama värvikoodeksit üldise arusaama jaoks oma tiimiliikmete koormusest projekti planeerides. (Joonis 6)



Joonis 6. Veebilehe näide ressursside kuvamisest [8]

Joonis 7 kuvab töötajate nimekirja ja kalendrivaates teostatud töötunnid päevade kaupa. Näites meeldib autorile naturaalsete värvide kasutamine ning aja kuvamine kalendrivaates on toodud välja nii nädala kui ka eraldi päevade kaupa. Autori arvates on hea, et sisse on jäetud ka nädalavahetus (laupäev ja pühapäev), sest see aitab kergesti eraldada nädalaid. Lisaks on üleval paremal toodud välja lisamise nupp, mis on taustal lihtsasti eristatav. (Joonis 7)



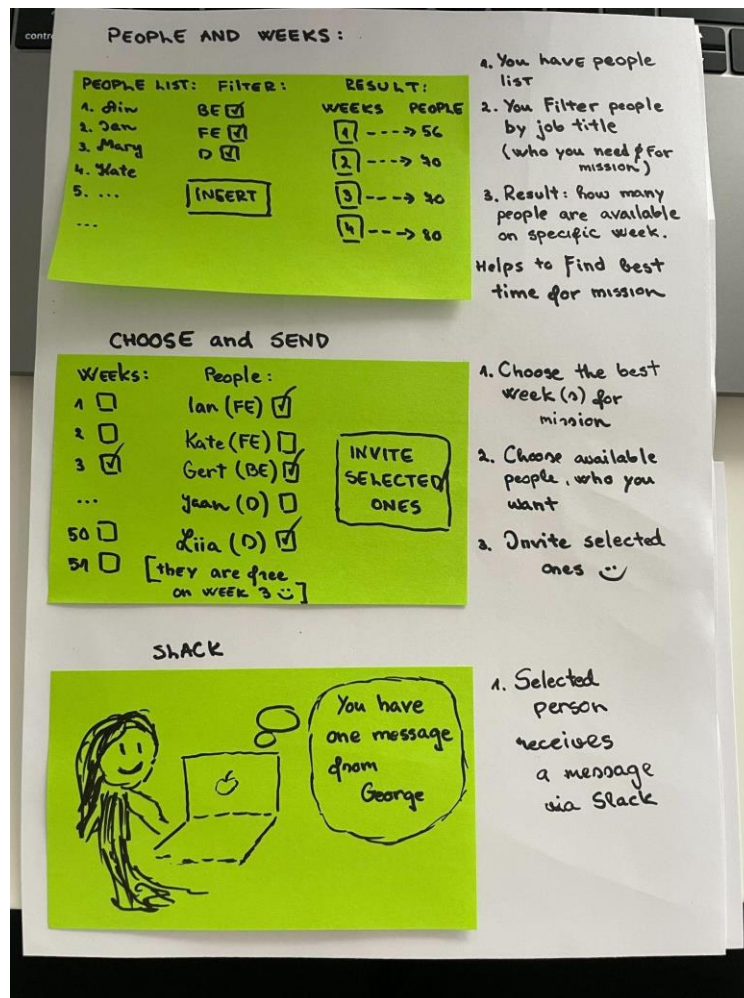
Joonis 7. Veebilehe näide värvigamast [9]

Peale esimest etappi skitseeris autor plaani (Joonis 8). Esimesel pildil on joonistatud inimeste loetelu, filtreerimine ja selle tulemus. Esmalt näeb kasutaja inimeste nimekirja, kust ta saab välja filtreerida vastavalt soovitud ametipositsioonidele vajalikud inimesed. Ametipositsioonid on valitud vastavalt sellele, keda oma projekti jaoks vajatakse. Filtreerimine teostatud, kuvatakse kasutajale nädalaid koos vabade inimeste arvuga. Idee mõtte on arendusprojekti loojale kuvada kiiresti ülevaade vabade inimeste arvust nädalate kaupa. Nii saab projekti looja aimduse, millal on sobivam oma projekt ellu viia ning millistel nädalatel on tal kindlalt võimalik leida vajalikke inimesi.

Järgmisel pildil on kuvatud taas nädalate nimekiri ja inimeste arv. Kasutajal on võimalus valida arendusprojekti kestvuse ajaks endale sobivad nädalad, mille tulemusel kuvatakse talle kõikide nende nädalate vabade inimeste nimekiri koos ametipositsiooniga. Järgmiseks saab kasutaja valida endale sobivad inimesed ja edastada läbi Slack keskkonna neile kutse projektile.

Viimasel pildil on näha inimest, keda on projektile kutsunud ning kellele on tulnud teade kutses Slack'i.

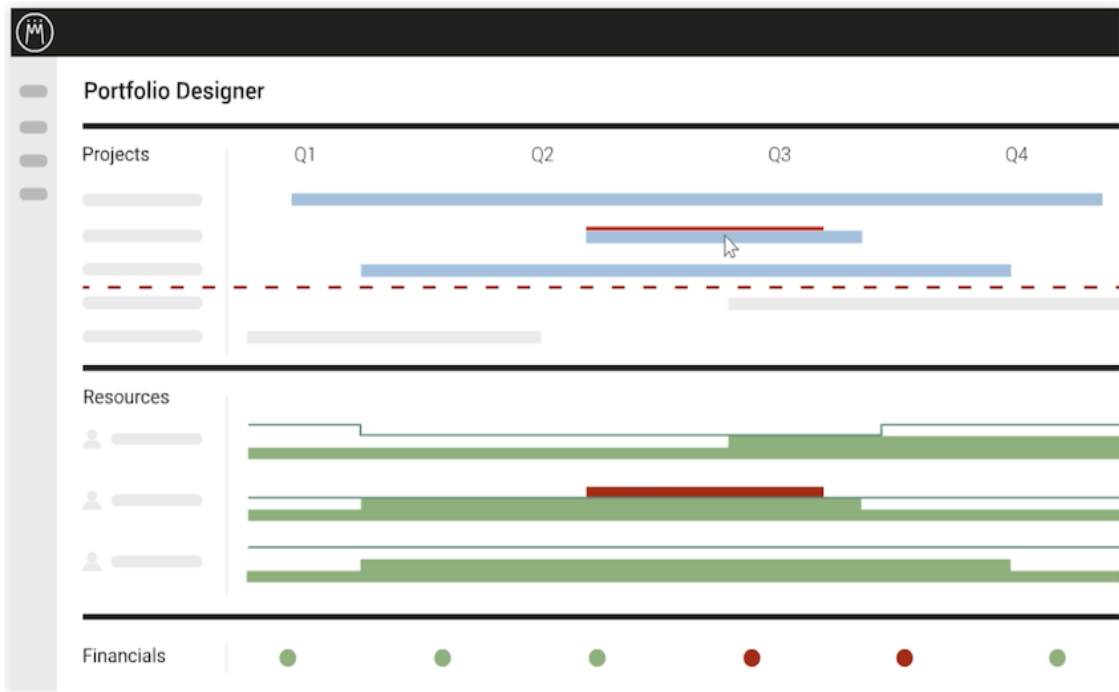
Antud ideedest sai kasutust ainult inimeste nimekirja kuvamine, kus saab sobilikud inimesed valida ja neile edastada läbi Slack'i kutse.



Joonis 8. Töötoa viimane etapp: lahenduse skitseerimine

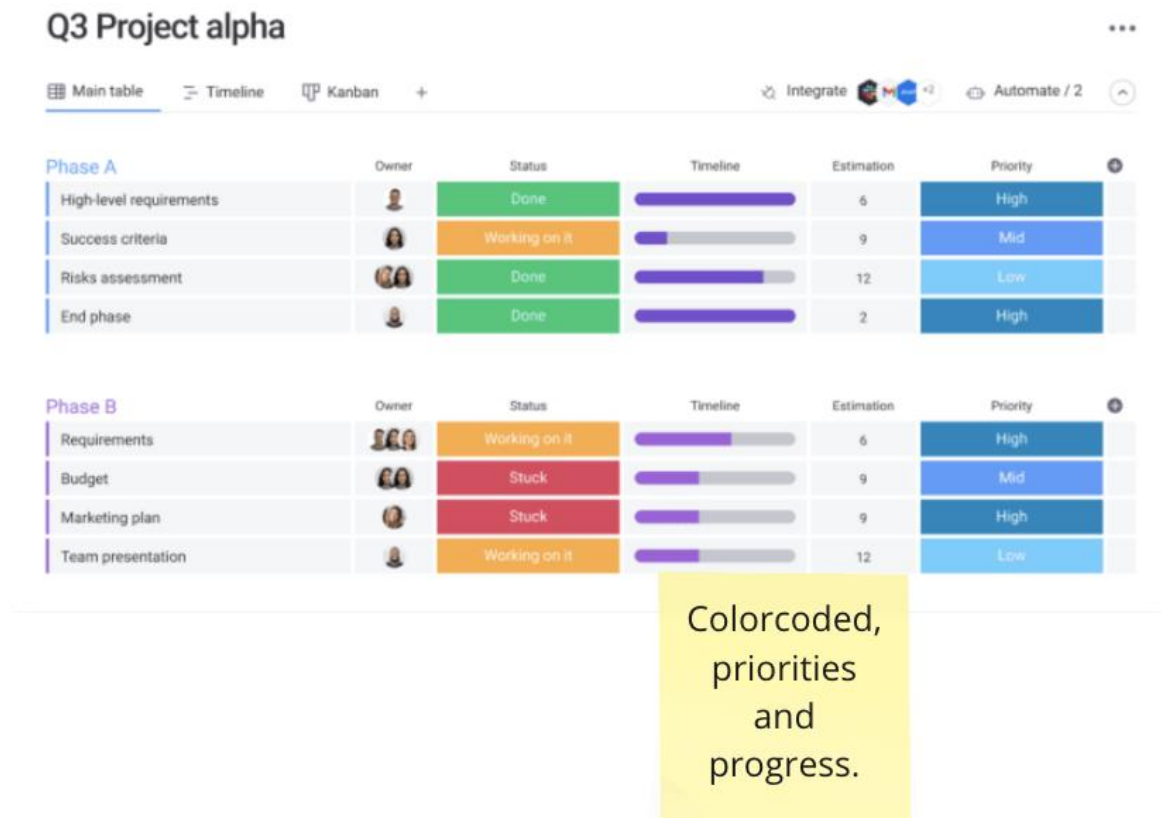
4.1.1.2 Reia ideed

Esimeses etapis tõi Reia välja kahe veebilehe pealt kolm näidet. Joonisel 9 on kujutatud kalender kus on eraldi välja toodud projektid ning inimressursid. Kalender on nädalate vaates ning eri värvidega, mis aitab kiiremini saada parema ülevaate. Ressursside osas on tulpdigrammi laadselt välja toodud kui palju või kas üldse on vabu ressursse. Näide on autori poolt välja valitud just sellepärast, et on kokku koondatud nii projektid kui ka ressursid ühele lehele ning üksteise alla. Kasuks tuli ka see, et kalendril on kasutatud erinevaid värve, et seda oleks lihtsam lugeda. (Joonis 9)



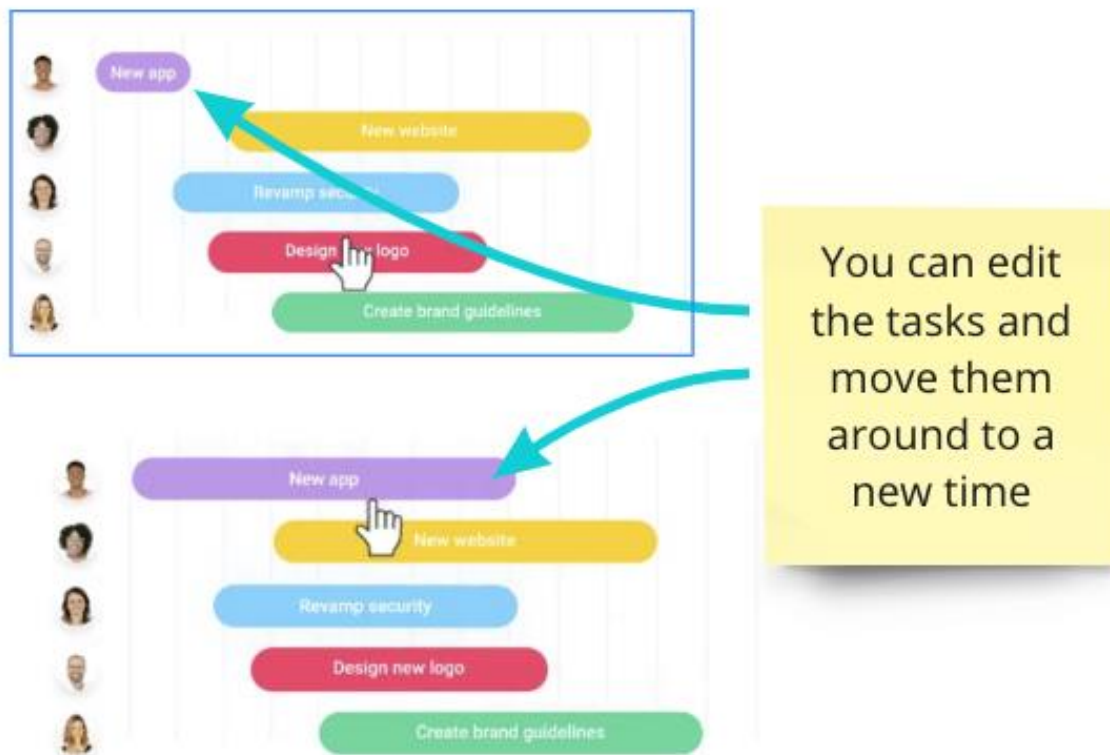
Joonis 9. Veebilehe näide projektide ja ressursside eraldatusest [10]

Joonis 10 kujutab projekti planeerimise tabelit. Seal on välja toodud erinevad faasid, nende faaside eest vastutavad isikud, staatus, ligikaudne ajakulu ning tähtsus. Autorile jäid silma erinevad värvid, mida kasutatakse staatuse ning tähtsuse määramiseks. Kui faasiga ollakse hädas, siis kasutatakse punast värvi staatuse kuvamiseks, kui faas on pooleli, siis tähistatakse seda kollasega ning kui faas on tehtud, siis rohelisega. Tähtsust ehk prioriteeti tähistatakse kolme erinevat tooni sinise värviga. Tumesinine tähistab kõige tähtsamat faasi, sinine tähistab keskmise tähtsusega faasi ning helesinine kõige väiksema tähtsusega faasi. Ligikaudne ajakulu on märgitud nädalate arvuga ja lisaks on kuvatud ka visuaalselt sinise joonena. (Joonis 10)



Joonis 10. Veebilehe näide värvide järgi kategoriseerimisest. [11]

Joonisel 11 on samuti kujutatud kalender, kus on kuvatud projektid. Jällegi kõik projektid on eri värvidega. Kõige enam jäi autorile silma aga see, et neid projekte sai nihutada edasi-tagasi ehk muuta kuupäevi. Samuti oli võimalik teha projekte pikemaks ning lühemaks, mis tundus väga kasutajasõbralik. Üks funktsioon oli sellel kalendril veel, mida pildi pealt ei ole näha, aga seal on võimalik projektide omanike ka vahetada klõpsates inimeste ikoonidele. Missions Tool'is tuleksid sellised funktsioonid kasuks, et projektide planeerimine oleks veelgi lihtsam ja mugavam. Projektide aja ja pikkuse muutmine samast kohast on autori arvates väga kasutajasõbralik lahendus. (Joonis 11)



Joonis 11. Veebilehe näide projektide ja ressursside eraldatusest [11]

Peale näidete väljatoomist skitseerisid autorid oma visiooni järgi lahenduse. Joonisel 12 on kuvatud kolm skitseeringut koos selgitustega.

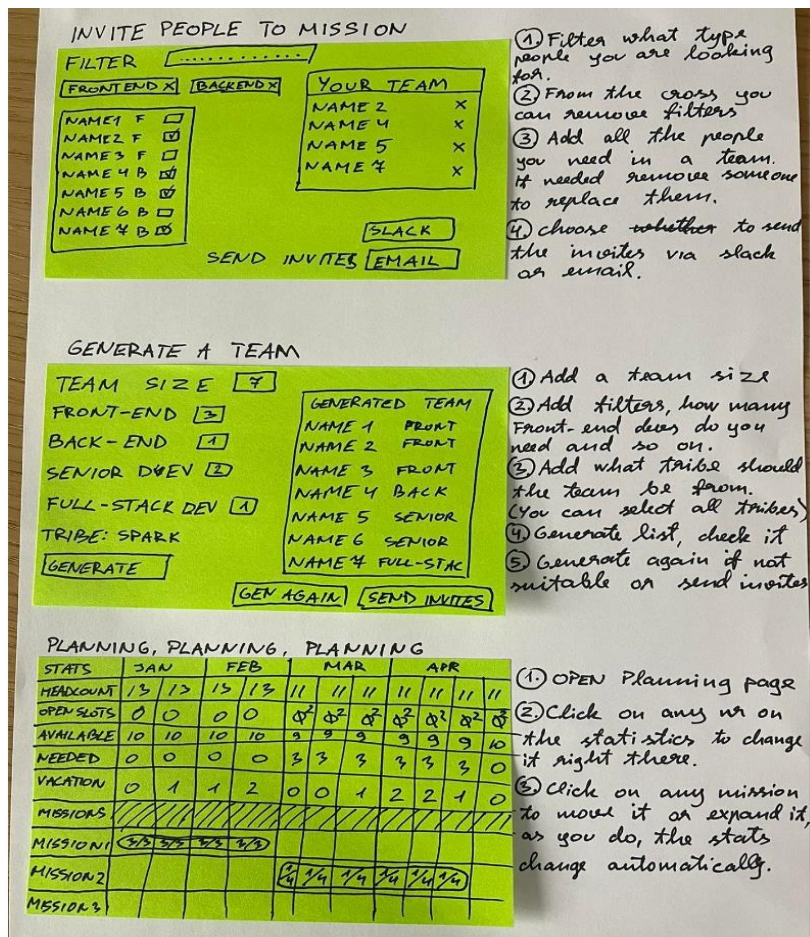
Esimesel joonisel on autor teinud projektile inimeste kutsumise vormi. Autor kujutas vormile filtreerimise võimaluse, kus saab otsida töötajaid positsiooni, tööstaaži, nime, valdkonna või asukoha järgi. Otsingulahtrisse vastava positsiooni või mõne muu filtreerimiseks vajaliku filtri kirjutamine ning enteri vajutamine jätab süsteem filtri meelde ning rakendab selle automaatselt. Filtreid saab lisada lõpmata palju ning ristikesel vajutades ka neid eemaldada. Otsingulahtri alla avaneb nimekiri filtritele vastavate töötajatega. Nimekirjas iga töötaja nime kõrval on kastike, kuhu saab panna linnukese, et valida töötaja enda meeskonda. Nimekiri valitud töötajatest ilmub kohe nimekirja kõrvale. Kui meeskond on valitud siis on kaks võimalust, kas saata töötajatele projektile kutsed läbi e-maili või Slack'i. (Joonis 12)

Teisel joonisel on vorm, mille kaudu on võimalik lasta süsteemil ise genereerida projektile meeskond. Vormile peab sisestama meeskonna suuruse ning kui palju on teatud arendajaid vaja ning millisest valdkonnast neid soovitakse. Kui andmed on sisestatud, siis

tuleb vajutada nuppu “Genereeri”, seejärel ilmub nimekiri valitud isikutest. Kui nimekiri on sobiv, saab saata neile kutsed, kui aga ei sobi, siis võib uuesti genereerida otsingu ning vajadusel muuta ka andmeid, kuni saadud meeskond on sobiv. (Joonis 12)

Kolmandal joonisel on kuvatud kalendrivaade planeerimisest. Seal on nädalate kaupa kuvatud statistika ning nende all nädalate kaupa ka projektid. Statistikaks on nädalas vabade arendajate arv, projektidesse vajalike arendajate arv ning puhkusel olevate arendajate arv. Projektidel on nädalate kaupa, projekti kestvuse aja nädalatel, kuvatud projektile vaja minevate arendajate arv ning projektiga liitunud arendajate arv. Vaates on võimalik muuta nii statistikat kui ka projekte. Statistikat saab muuta arvule klõpsates ning projektidel saab muuta nende kestvust nihutades neid pikemaks või lühemaks või hoopis uuele nädalale. (Joonis 12)

Reia lahenduste skitseerimistest läks Missions Tool’i kasutusse töötajate filtreerimine ning mõneti ka statistika vaade.



Joonis 12. Töötoa viimane etapp: Reia lahenduse skitseerimine

4.1.2 Funktsionaalsed nõuded kasutajalugudena

Järgnevalt tuuakse välja tabelitena rakenduse funktsionaalsed nõuded kasutajalugudena. Antud kasutajalood on kirjapandud tootejuhtide Annabel Matkuri ja Lii Saluvere poolt.

4.1.2.1 Arendusüksuse statistika vaatamine

Tabel 1. Arendusüksuse statistika vaatamine

Nimetus	Arendusüksuse statistika vaatamine
Eesmärk	Kasutaja saab vaadata arendusüksuse statistikat (arendusüksuses töötavate arendajate koguarvu, mitut arendajat on vaja igapäevaste ülesannetega tegelema, mitu arendajat tegelikult igapäevaste ülesannetega tegeleb, mitut arendajat on kokku projektidesse vaja, mitu arendajat on süsteemis märkinud, et osalevad konkreetses projektis, mitu vaba kohta on veel projektides, mitu arendajat teistest arendusüksustest konkreetse arendusüksuse projektides osaleb ning mitu arendajat viibib puhkusel) ning filtreerida, millist statistikat arendusüksuse kohta soovitakse nädalate lõikes näha. [12]
Tegevuse kirjeldus	Kasutaja navigeerib end veebilehel „ <i>Planning</i> ” olevate meeskondade nimede kaudu “...”planning/tribe/:id” ning seal avaneb vaade soovitud statistika kohta.

4.1.2.2 Arendusüksuse statistika ridade selgituste vaatamine

Tabel 2. Arendusüksuse statistika ridade selgituste vaatamine

Nimetus	Arendusüksuse statistika ridade selgituste vaatamine
Eesmärk	Kasutaja saab vaadata sõnalisi selgitusi statistika ridade tähenduste kohta. [12]
Tegevuse kirjeldus	Kasutaja navigeerib end veebilehel „ <i>Planning</i> ” olevate meeskondade nimede kaudu “...”planning/tribe/:id”, kust edasi navigeerib hiirega statistika real oleva info ikooni peale ning sealt ilmub ikooni kohale statistika rea seletus.

4.1.2.3 Projekti staatuse vaatamine

Tabel 3. Projekti staatuse vaatamine

Nimetus	Projekti staatuse vaatamine
Eesmärk	Kasutaja saab vaadata, milline on projekti staatus. [12]
Tegevuse kirjeldus	Kasutaja navigeerib end veebilehel „ <i>Planning</i> ” olevate meeskondade nimede kaudu “...” <i>planning/tribe/:id</i> ”, kust edasi liigub hiirega projekti kõrval olevale värvilisele ringikesel, seejärel ilmub vastava projekti staatus selle kohale.

4.1.2.4 Projektide hoiatuse sisu vaatamine

Tabel 4. Projektide hoiatuse sisu vaatamine

Nimetus	Projektide hoiatuse sisu vaatamine
Eesmärk	Kasutaja saab vaadata projektide taustal kuvatava hoiatuse sisu. Hoiatus kuvatakse, kui projekti alguseni on jäänud vähem kui kolm nädalat ning seal ei ole piisavalt arendajaid. [12]
Tegevuse kirjeldus	Kasutaja navigeerib end veebilehel „ <i>Planning</i> ” olevate meeskondade nimede kaudu “...” <i>planning/tribe/:id</i> ”, kust edasi liigub hiirega hoiatusega projekti peale ning seejärel kuvatakse selle kohale hoiatuse sisu.

4.1.2.5 Projekti detailse sisu vaatamine

Tabel 5. Projekti detailse sisu vaatamine

Nimetus	Projekti detailse sisu vaatamine
Eesmärk	Kasutaja saab vaadata detailset infot projekti kohta, projekti nime, asukohta, toimumisaega, teemat ning osalejaid. Samuti näeb kasutaja viidet, mis viib projektiga seotud <i>Confluence</i> 'i lehele. [12]
Tegevuse kirjeldus	Kasutaja navigeerib end veebilehel „ <i>Planning</i> ” olevate meeskondade nimede kaudu “...” <i>planning/tribe/:id</i> ”, kust edasi klikib projekti nime peale, seejärel avaneb projekti detailne vaade hüpikaknana.

4.1.2.6 Projekti loomine

Tabel 6. Projekti loomine

Nimetus	Projekti loomine
Eesmärk	Kasutaja saab lisada uusi projekte. Projekti lisamisel on vaja määrata projekti nimi, arendusüksuse nimetus, asukoha info, projekti kestus nädalates, staatus, arendusmeeskonna suurus, sh kui palju on meeskonda vaja eesrakenduse arendajaid ning kui palju tagarakenduse arendajaid, ning kirjeldus. Samuti saab määrata kuupäevade vahemiku, millal soovitakse projekti läbi viia. [12]
Tegevuse kirjeldus	Kasutaja navigeerib end veebilehel „ <i>Planning</i> ” olevate meeskondade nimede kaudu “...” <i>planning/tribe/:id</i> ”, kust edasi leiab roheline pluss nupu, sellele klikkides avaneb vaade “...” <i>planning/tribe/:id/create-mission</i> ” vajalike väljadega, et uus projekt luua.

4.1.2.7 Projekti inimeste kutsumine

Tabel 7. Projekti inimeste kutsumine

Nimetus	Projekti inimeste kutsumine
Eesmärk	Kasutaja saab kutsuda projekti inimesi, filtreerida inimesed saadavuse, positsiooni, taseme, arendusüksuse ja asukoha põhiseelt ning vajadusel täiendades automaatselt genereeritud kutsumise sõnumit. Samuti on võimalik otsida inimest nime, positsiooni ja taseme järgi. [12]
Tegevuse kirjeldus	Kasutaja navigeerib end veebilehel „ <i>Planning</i> ” olevate meeskondade nimede kaudu “...” <i>planning/tribe/:id</i> ”, kust liigutakse hiirega projekti peale, kuhu soovitakse arendajaid kutsuda ning siis avaldub inimeste lisamise nupp, kuhu klikkides avaneb vaade “...” <i>/missions/:id/invite-people-to-mission</i> ” vabade arendajate listiga, kust saab kasutaja valida endale sobivad arendajad.

4.1.2.8 Inimese detailse info vaatamine

Tabel 8. Inimese detailse info vaatamine

Nimetus	Inimese detailse info vaatamine
Eesmärk	Kasutaja saab inimeste projekti kutsumisel vaadata detailset infot inimese kohta, sealhulgas inimese asukohta, meiliaadressi, millistes arendusüksustes ta on töötanud, milliste projektidega on tegelenud ning kui suures mahus oma tööajast on ta tegelenud projektidega ja kui palju igapäevaste tööülesannetega. [12]
Tegevuse kirjeldus	Kasutaja navigeerib end veebilehel „ <i>Planning</i> ” olevate meeskondade nimede kaudu “...” <i>planning/tribe/:id</i> ”, kust liigutakse hiirega projekti peale, kuhu soovitakse arendajaid kutsuda ning siis avaldub inimeste lisamise nupp, kuhu klikkides avaneb vaade “...”/ <i>missions/:id/invite-people-to-mission</i> ” vabade arendajate listiga, seejärel saab kasutaja klikkida inimese nime peale, mis avab hüpikaknana vastava inimese kohta detailse info.

4.1.3 Mittefunktsionaalsed nõuded

Järgnevalt tuuakse välja veebirakenduse mittefunktsionaalsed nõuded kasutaja põhiselt.

- Kasutajana soovin, et rakenduse vaate ja andmete laadimine oleks võimalikult kiire ja ilma viivitusteta.
- Kasutajana soovin, et rakenduse funktsionaalsuste kasutamine on tehtud võimalikult arusaadavaks ja mugavaks.
- Kasutajana soovin, et rakenduse värviskeem ja kasutatavad ikoonid põhinevad eelnevalt loodud rakendusel.

4.2 Eesrakenduse disain

Järgnevates alampeatükkides tuuakse välja kolme vaate loodud prototüübid: statistika vaade, projekti loomise vaade ja inimeste projekti kutsumise vaade. Prototüübid on loodud tootejuhtide Annabel Matkuri ja Lii Saluvere poolt.

4.2.1 Statistika prototüüp

Joonisel 13 on näha prototüüpi, mis loodi aine ITB1706 raames. Prototüübil kujutatakse statistika vaadet, kus kuvatakse andmeid vaadeldava arendusüksuse ressursside kohta.

Lisaks statistikale kuvatakse antud meeskonna projektide nimekirja ja kalendri vaates on nädalate kaupa välja toodud arendajate arv, mis on vaja projekti teostamiseks, ning kui suur osa sellest on juba olemas. (Joonis 13)

Product Vision Mission Tool People Portal																	
Programs Ideas Exploration Missions Tribes Plan People Statistics Hackathon																	
Tribe 1 Edit calendar																	
Tribe 1 IN PREPARATION, IN PROGRESS, IN DRAFT																	
More	March				April				May				June				
	W12 21.03-25.03	W13 28.03-01.04	W14 04.04-08.04	W15 11.04-15.04	W16 18.04-22.04	W17 25.04-29.04	W18 02.05-06.05	W19 09.05-13.05	W20 16.05-20.05	W21 23.05-27.05	W22 30.05-03.06	W23 06.06-10.06	W24 13.06-17.06	W25 20.06-24.06	W26 27.06-01.07	W27 04.07-08.07	W28 11.07-15.07
TRIBE STATISTICS																	
Engineers headcount	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
Launchpad minimum size	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
Launchpad actual size	18	18	16	17	17	8	8	9	9	19	19	19	16	16	19	19	19
Needed for missions	2	2	0	3	3	3	3	3	3	1	1	1	1	1	1	1	1
Engineers on mission	2	2	0	2	2	2	2	2	2	1	1	1	1	1	1	1	1
Open slots on missions	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0
Outtribers on mission	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Vacation	0	0	4	0	0	9	9	8	8	0	0	0	3	3	0	0	0
Available for missions	14	14	12	13	13	4	4	5	5	15	15	15	12	12	15	15	15
MISSIONS (2)																	
Mission 1	2/2	2/2															
Mission 2				2/3	2/3	2/3	2/3	2/3	2/3								
LOANED OUT (2)																	
Other tribe mission 1										1							
Other tribe mission 2											1	1	1	1	1	1	1

Joonis 13. Statistikaandmete vaate prototüüp [12]

4.2.2 Projekti lisamise vormi prototüüp

Joonisel 14 on kujutatud uue projekti lisamiseks loodud ankeeti, mis koosneb väljadest projekti nimi, arendusüksus, asukoht, kestvus nädalates, staatus, arendustiimi suurus, eesrakenduse ja tagarakenduse arendajate arv ning kirjeldus. (Joonis 14)

Peale andmete sisestamist, kuvatakse allolevas kalendris projekti kestvuse aeg põhinedes nädala arvul ja arendajate saadavusest nädalatel. Valged kastid tähistavad vabasid nädalaid, mille peal saab rohelist ristkülikut lohistada. Hallid kastid tähistavad ebasobivat aega projekti toimumiseks ning nende peale ristkülikut lohistades, liigub see esialgsele kohale tagasi. (Joonis 14)

Create new Mission

General

Mission Name (required)

Tribe

Location

Duration in Weeks

Status (required)

Dev Team Size (required)

Frontend

Backend

Description

Dates selected: 📅 18 April 2022 - 27 May 2022 (6 weeks)

March		April				May				June			July						
W12	W13	W14	W15	W16	W17	W18	W19	W20	W21	W22	W23	W24	W25	W26	W27	W28	W29	W30	
21.03-25.03	28.03-01.04	4.04-8.04	11.04-15.04	18.04-22.04	25.04-29.04	2.05-6.05	9.05-13.05	16.05-20.05	23.05-27.05	30.05-3.06	6.06-10.06	13.06-17.06	20.06-24.06	27.06-1.07	4.07-8.07	11.07-15.07	18.07-22.07	25.07-29.07	
						18 April 2022 - 27 May 2022													

Joonis 14. Projekti lisamisvormi prototüüp [12]

Peale projekti salvestamist kuvatakse uus projekt planeerimislehe projektide nimekirjas. (Joonis 15)

Tribe 1

Edit calendar

Tribe 1 IN PREPARATION, IN PROGRESS, IN DRAFT

More	March			April							May			June			
	W12 21.03-25.03	W13 28.03-01.04	W14 04.04-08.04	W15 11.04-15.04	W16 18.04-22.04	W17 25.04-29.04	W18 02.05-06.05	W19 09.05-13.05	W20 16.05-20.05	W21 23.05-27.05	W22 30.05-03.06	W23 06.06-10.06	W24 13.06-17.06	W25 20.06-24.06	W26 27.06-01.07	W27 04.07-08.07	W28 11.07-15.07
TRIBE STATISTICS																	
Engineers headcount	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
Launchpad minimum size	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
Launchpad actual size	18	18	16	17	14	5	5	6	6	16	19	19	16	16	19	19	19
Needed for missions	2	2	0	3	6	6	6	6	6	4	1	1	1	1	1	1	1
Engineers on mission	2	2	0	2	2	2	2	2	2	1	1	1	1	1	1	1	1
Open slots on missions	0	0	0	1	4	4	4	4	4	3	0	0	0	0	0	0	0
Outtribers on mission	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Vacation	0	0	4	0	0	9	9	8	8	0	0	0	3	3	0	0	0
Available for missions	14	14	12	13	10	1	1	2	2	12	15	15	12	12	15	15	15
MISSIONS (3)																	
Mission 1	2/2	2/2															
Mission 2				2/3	2/3	2/3	2/3	2/3	2/3								
Your new mission					0/3	0/3	0/3	0/3	0/3	0/3							
LOANED OUT (2)																	
Other tribe mission 1 Tribe 1										1							
Other tribe mission 2 Tribe 2											1	1	1	1	1	1	1

Joonis 15. Planeerimisvaade peale projekti lisamist [12]

4.2.3 Projekti inimeste kutsumise prototüüp

Joonisel 16 on kujutatud vorm, mille kaudu saab arendajaid kutsuda projektile osalema. See koosneb otsinguväljast, viiest filtreerimise väljast, arendajate nimekirjast, sõnumi kastist ning katkestamise ja kutsumise nuppudest. (Joonis 16)

Filtreerimisväljadeks on arendajate positsioon, tase, arendusüksus, asukoht ning valik kas kuvatakse kõiki või ainult vabasid arendajaid.

Arendajate nimekirjas on välja toodud arendaja nimi, positsioon, tase, arendusüksus, asukoht ning valimiskastike, mille kaudu saab valida sobiva arendaja. Sõnumi kast on selle jaoks, et projekti looja saab kirjutada arendajatele lühitutvustuse projektist või muu vajaliku info.

Kui kõik soovitud arendajad on valimiskasti kaudu valitud ning sõnum on kirjutatud, saab „Send invites via Slack” nupu kaudu saata kõikidele valitud arendajatele selle sama

sõnumi Slack keskkonda. Kui poole peal otsustatakse, et praegu ei soovita projektile inimesi kutsuda, tuleb vajutada „Cancel” nuppu, mis viib tagasi statistika lehele.

Invite people to mission

Search by person, position, level etc

Only available engineers Position Level Tribe 1 Location

Name	Position	Level	Tribe	Location	Select
Person 1	Frontend	Junior	Tribe 1	Tartu	<input type="checkbox"/>
Person 2	Frontend	Senior	Tribe 1	Tallinn	<input type="checkbox"/>
Person 3	Backend	Mid	Tribe 1	Tallinn	<input type="checkbox"/>
Person 4	Full Stack	Junior	Tribe 1	Tallinn	<input type="checkbox"/>

B I [link] [list] [code] [image]

Hi! 🚀🚀🚀
I will start a new mission called "My new mission" where I would like to invite you!
More information and applying [HERE](#) or if you have any questions please contact me!
🤖

+ [link] [image] [emoji] [mention] [text]

Cancel Send invites via Slack

Joonis 16. Projekti inimeste kutsumise prototüüp [12]

4.3 Arendusmetoodika valik

Järgnevalt kirjeldatakse lahti kaks agiilset arendusmetoodikat, mida lõputöö autorid kasutasid hübriidselt, samuti tuuakse välja mõlema metoodika head ja vead.

Arendusmetoodika valikul on lõputöö tegemisel lähtunud agiilsetest metoodikatest. Agiilne tarkvaraarendus on kontseptsioon, filosoofia ja metoodika, mis kujunes välja 90. aastatel [13]. Metoodikaga lähenetakse iteratiivselt tarkvaraarendusele, kasutades lühemaid ja kergemaid arendustsükleid [13]. Agiilses arenduses kogutakse nõuded, kavandatakse teekonda lahenduseni, realiseeritakse ja testitakse loodud tulem vaheldumisi [14]. Meeskond on avatud muudatustele ja kõiki uusi kliendi soove on võimalik lihtsalt ellu viia. Arendusprotsessi käigus uute ideede kohta tehakse otsuseid kogu protsessi ajal ja tavaliselt kogutakse uusi mõtteid suheldes oma klientidega.

Agiilsuse manifestis on välja toodud printsiibid, mida tuleb jälgida. Lõputöö autorid toovad välja peamised printsiibid, mis puudutasid enim nende tehtud tööd:

- Tuleb aru saada muutuvatest olukordadest ja kohaneda nendega, isegi kui need esinevad arendustegevuse lõppjärgus [14].
- Klientidele, kelleks antud lõputöö puhul olid ettevõtte Pipedrive töötajad, edastatakse uuendusi tarkvaras võimalikult tihti [14].
- Arendusprotsessis osalejad (arendajad ja tootejuhid) töötavad projekti vältel tööpäeviti koos ja hoiavad pidevat suhtlust omavahel [14].

Lisaks agiilse arenduse headele tavadele, järgitakse häid tavasid tarkvaraarenduses, milleks on

- arendajate omavaheline pidev suhtlus arendusprotsessi ajal [14],
- pannakse rohkem rõhku töötavale tarkvarale kui dokumentatsioonile [14],
- pidev suhtlemine klientidega [14],
- reageerimine muutustele [14].

Hea praktikana kasutati tuntud koodi ülevaatamist teiste arendajate poolt, mis aitas kaasa puhtama koodi kirjutamisele ning vigade märkamisele ja likvideerimisele.

Lõputöö autorid kasutasid tarkvaraarenduses hübriidselt kaht agiilse tarkvaraarenduse raamistikku: ekstreemprogrammeerimine ja Scrum. Mõlemast raamistikust kasutati neile omaseid praktikaid, et parimat tarkvara luua.

Järgnevalt on kahes peatükis lahti seletatud ja analüüsitud ekstreemprogrammeerimise ja Scrum'i kasutamist lõputöö projekti arendusel.

4.3.1 Ekstreemprogrammeerimine

Ekstreemprogrammeerimine ehk XP on agiilne tarkvaraarenduse raamistik [15]. See sai alguse Kent Beck'i, Ward Cunningham'i ja Ron Jeffries'i poolt Chrysler Comprehensive Compensation System (C3) palgaarvestussüsteemi projekti kallal töötades. Kent Beck kirjutas ekstreemprogrammeerimisest ka raamatu „Extreme Programming Explained”, mis ilmus välja 1999. aastal [16]. XP eesmärk on toota kvaliteetsemat tarkvara ja kõrgendada arendusmeeskonna elukvaliteeti. XP-l on viis põhilist väärtust: suhtlus, lihtsus, tagasiside, julgus ning ausus [15].

XP alusel kuuluvad projekti järgmised rollid:

- Klient, kes määrab mida ning millises järjekorras on vaja teha. [17]
- Programmeerija, kes kirjutab kasutajalugude põhjal koodi ning ühikteste. [17]
- Treener, kes vastutab projekti tehnilise teostuse eest, omab ülevaadet kogu projektist ning suhtleb kliendiga. [17]
- Testija, kes aitab kliendil valida ja kirjutada funktsionaalseid teste. [17]
- Konsultant, kes palgatakse siis, kui projekt vajab tugevaid tehnilisi teadmisi [17].

XP spetsiifilisuse tõttu ei pruugi kogu metoodika iga arendamise käigus olla kasulik, aga see ei takista kasutamast arendajate konteksti arvestades vajalike tavadid. Ektreemprogrammeerimisel on 12 erinevat praktikat, millest autorid kasutasid ainult ühte, paarisprogrammeerimist.

Paarisprogrammeerimine on kahe arendaja ühiselt programmeerimine, kus arendajad töötavad kombineeritult koos ühe arvuti taga. Üks arendaja kirjutab koodi ja teine jälgib ning mõtleb juba järgmistele sammudele, iga teatud aja tagant vahetatakse rollid. Paarisprogrammeerimise idee on selles, et kaks pead on parem kui üks pea. Selle kasutamine aitas kiiremini leida lahendusi, sest pidevalt toimus koodi ülevaatamine kellegi teise poolt ning see võimaldas koheselt leida vead ning need likvideerida. Üldine esmamulje, et paarisprogrammeerimine võtab kaks korda rohkem aega, sai koheselt tõestatud vastupidiseks. Paarisprogrammeerides keskendutakse rohkem käsil olevale probleemile, mis võimaldab luua puhtamat koodi. Lisaks tekkis paarisprogrammeerimisega mõlemal arendajal ühine arusaam koodist.

Vaatamata sellele, et paarisprogrammeerimine tõi tööle kaasa lisaväärtust, pidid autorid jätkama iseseisvalt programmeerimisega. Kuna nende koodi pidid üle vaatama Pipedrive OÜ töötaja, kellel oli samal ajal käsil ka teised projektid, kulus koodi ülevaatamise peale oodatust rohkem aega. Seetõttu otsustasid autorid paarisprogrammeerimist enam mitte kasutada ning jagasid ülesanded ära. Paarisprogrammeerimist kasutati kokku kuu aega ning selle aja jooksul oli näha, et mõlemad arendajad said üheselt koodist aru ning lahendusi leiti kiiremini, sest toimus pidev arutelu. Autorid leidsid, et iseseisvalt töötades jõuavad rohkem realiseerida erinevaid funktsioone.

4.3.2 Scrum

Scrum on arendatud välja Jeff Sutherlandi poolt 1993. aastal ning see on suunatud iteratiivse arendusprotsessi juhtimisele, mitte arenduspraktikate juurutamisele. [18], [14]

Scrum on jaotatud kolme faasi: planeerimine, sprindid ja projekti lõpetamine.

Scrum raamistikust võtsid autorid kasutusele koosolekute läbiviimise püstijala koosolekute näol ja lisaks toimus arendusprotsessi paika panemine sprintide abil.

Püstijalakoosolekud toimusid iganädal esmaspäeva ja neljapäeva hommikuti ning kestsid umbes 10-15 minutit. Selle aja jooksul oli igal tiimiliikmel ülesanne vastata kolmele küsimusele:

1. „Mida ta on teinud?”
2. „Mida ta hakkab tegema?”
3. „Kas on ülesande lahendamisel mõni takistus?”

Nii, nagu on kirjeldatud artiklis „Investigating daily team meetings in agile software projects”, toovad ka autorid välja, et kui koosolekuid on tihti ja kõigile on sobiv selle läbiviimise formaat, siis on hea koostöös tiimiga otsuseid vastu võtta [19]. On uuritud, kuidas kolme minutilised püstijalakoosolekud töötavad ettevõtetes ning kas need annavad tiimile pigem positiivset või negatiivset mõju. Uurides raamatut „Agile Processes in Software Engineering and Extreme Programming”, tuuakse seal välja, kuidas meeskonna liikmete vaheline suhtlus paranes. Uurimuses on välja toodud, et koosolekust osavõtjad kuulasid tähelepanelikult üksteise mõtteid, otsuste langetamisel on kõik osalejad kaasatud, kohtumine oli efektiivne, tunda oli head koostoimimist ja emotsioonide väljendamine jäi tervislikule tasandile [20]. Töö autorid on eelnevate väidetega nõus. Arendusprotsessi juures on oluline suhelda ning anda teada tekkinud probleemidest, mille lahendamine on võtnud oodatust kauem aega. Võttes arendusprotsessis kasutusele kindlatel aegadel koosolekud, mis toimuvad rohkem kui kord nädalas, annavad võimaluse arendajale end nähtavaks teha, kuulda teiste töökaaslaste arvamust ja leida ühine lahendus probleemile. Veel enam annavad lühikoosolekud aimduse, mida teised tiimikaaslased teevad. Näiteks toimus antud lõputöö arendus koostöös tootejuhtidega, kelle ülesandeks oli eesrakenduse arendajatele koguda vajalikud nõuded ja luua rakenduse prototüüp. Ilma iganädalaste püstijalakoosolekuteta oleks puudunud pidev ülevaade nende töö protsessist

ning pidev info uurimine tootejuhtidelt oleks olnud tüütu ja aeganõudev. Lisaks andis hetk enne koosoleku algust oma mõtted fokuseerida täielikult tehtud tööle ning tuletada meelde, millega eelmisel päeval tegeleti.

Arendusprotsessi planeerimiseks kasutati Scrum metoodikast tuntud sprinte. Need on järjepidevuse loomiseks fikseeritud pikkusega iteratsioonid, mis kestavad tavaliselt kaks kuni neli nädalat [21]. Hea tava on jätta sprinti kestvus umbes kahe kuni kolme nädala peale, et säiliks pidev uuenduste tootmine tarkvarale [22]. Antud lõputöös kestsid sprintid kauem, umbes neli kuni viis nädalat, sest viie tööpäeva asemel panustavad autorid arendusele 2,5 päeva nädalas. Sprintide loomisel võeti aluseks tootejuhtide poolt loodud ülesannete suuremad plokid, milles kirjeldati lühidalt loodava funktsiooni mõte. Arendajate ülesanne oli plokk väiksemateks ülesanneteks genereerida ning seejärel panna paika sprinti kestvuse ajal tehtavate ülesannete kogum. Kõik ülesanded pandi kirja keskkonnas Jira [23].

4.4 Tehnoloogia valik

Järgnevalt tuuakse välja lõputöö arendusprotsessiks kasutatud tehnoloogiad ja põhjendus nende valikul. Igale arendusprojektile valib arendusmeeskond ise sobivad tööriistad, programmeerimiskeskonna ja -keeled. Antud lõputöös käsitletakse ainult eesrakenduse arendust ning seetõttu tuuakse välja võimalikud platvormid ja keskkonnad arenduseks ning analüüsitakse erinevate raamistike ning programmeerimiskeelte tugevusi ja nõrkusi.

4.4.1 Raamistiku valik: React vs Vue

Ettevõttes Pipedrive on kasutusel peamiselt kaks raamistikku: React.js ja Vue.js. Järgnevalt kirjeldatakse nende kahe eelised ja lõputöös kasutatava raamistiku plussid.

React.js on loodud 2011. aastal ettevõtte Facebook tarkvaraarendaja poolt. React on mõjutatud XHP-st, mis on PHP jaoks lihtne HTML-i komponentraamistik. [24]

React.js kohta on toodud välja plussidena kergus ja lihtne kasutatavus ka mobiili seadmetes, mis võimaldab antud rakendust arendada ka edasi mobiilirakenduseks. Lisaks on lihtne taaskasutada koodis loodud komponente ja raamistik on kiire, kui tuleb kuvada renderdatud tulemust suurest andmehulgast. [25]

Miinus poolena on välja toodud, et React.js võtab enda alla liigselt palju mälu, mis võib tuua arendusprotsessis kulusid juurde, mida eelnevalt pole arvestatud. See võib olla tingitud sellest, et mõnedel juhtudel tuleb kirjutada liiga palju koodi. Lisaks on toodud välja, et see on liiga vaatele orienteeritud. [25]

Vue.js raamistik on loodud Google arendaja Evan You poolt 2014. aastal [26]. Vue-d on kirjeldatud kui progressiivset raamistikku, mida kasutatakse kasutajaliideste rajamisel veebi jaoks. Üheks Vue.js eripäraks on, et see on täielikult väljatöötanud avatud lähtekoodiga kogukond, mitte suur ettevõtte. Vue.js põhineb komponentidel, mille näide on välja toodud joonisel 17. Lisaks on joonisel 18 näha Vue.js faili ülesehitust.

```
<People>
  :name="Kati"
  :age="20"
  :isAdult="true">
</People>

<script>
props: {
  name: {
    type: String,
    required: true,
  },
  age: {
    type: Number,
    required: true,
  },
  isAdult: {
    type: Boolean,
  }
}
</script>
```

Joonis 17. Vue komponentide näide

```

<template>
  <form>
  </form>
</template>

<script>
import { addMissionStatusList, locations }
from '../Missions/data';
export default {
};
</script>

<style lang="postcss"
src="./CreateMission.postcss"></style>

```

Joonis 18. Vue faili näide

Vue-l on mitmeid eeliseid teiste raamistikega võrreldes.

- Lihtne selgeks õppida, kui varasem kasutajakogemus puudub. [27]
- Suudab soovitada arendajale erinevaid lähenemisviise, aga ka aktsepteerib erinevaid arendusvõtteid (näiteks organiseerides tehnoloogiaid eraldi). [27]
- Parem võimekus kasutada uusimaid tehnoloogiaid. [27]
- Google Chrome'i Vue laiendus arendajate tööriistade seas. [27]

Vue.js miinusteks võib välja tuua järgnevaid tähelepanekuid.

- Vähene kasutus tarkvara arendajate seas. [28]
- Limiteeritud funktsioonide arv. [28]
- Väheste professionaalide arv.

Lõputöös arendatav projekt on loodud enne seda, kui autorid hakkasid uusi funktsionaalsusi arendama ning seetõttu puudus neil kaasaráäkimise võimalus arenduskeskkonna valikul. Küll aga nõustuvad autorid eelnevalt välja toodud väitega, et raamistiku on lihtne selgeks õppida. Enne Pipedriveri ettevõttes tööle asumist puudus mõlemal autoril eelnev kogemus Vue.js raamistikuga, kuid peale projekti lõppu võib väita, et oskused on märgatavalt arenenud.

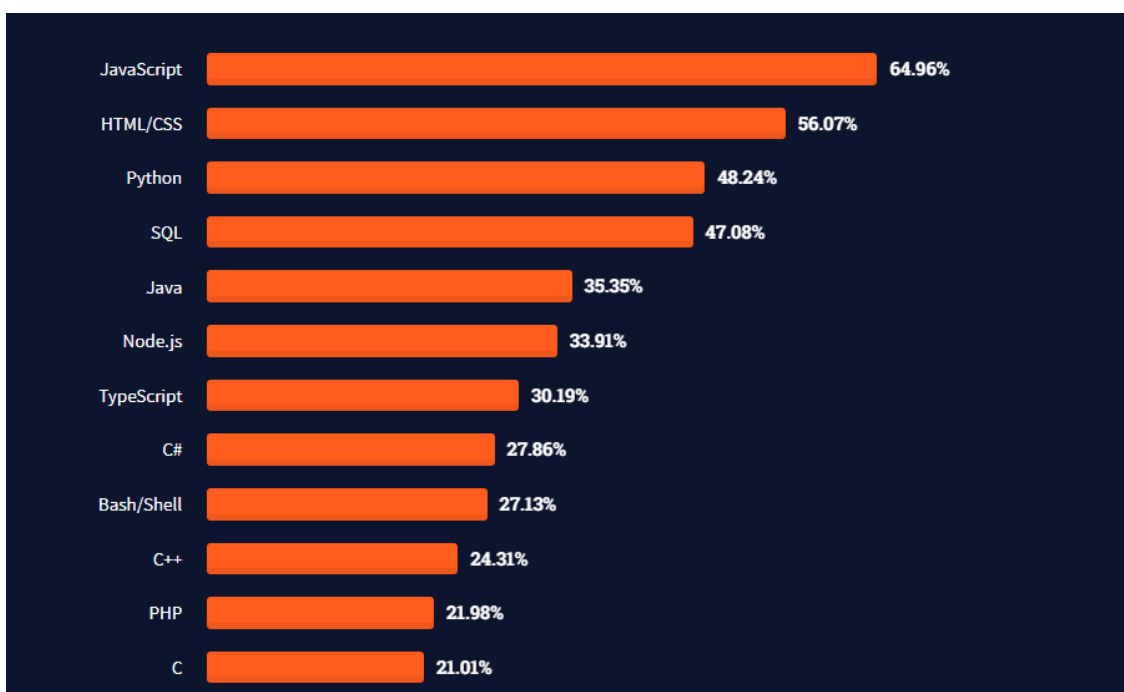
Miinusena toovad autorid välja, et kuna tegemist on üsna uue raamistikuga tarkvara arenduses, siis seetõttu on selle kohta vähem infot internetikeskkonnas, mis natukene aeglustab Vue.js õppimist.

4.4.2 Programmeerimiskeele valik

Laialdaselt on kasutusel ettevõttes programmeerimiskeeltena JavaScript, TypeScript, Python, CSS ja PHP. Huvitava tähelepanekuna võib välja tuua, et vähe kasutatakse populaarset programmeerimiskeelt Java't ning üldse ei ole arenduseks valitud C# keelt. Antud rakenduses Missions Tool on arendamiseks valitud programmeerimiskeelteks JavaScript, HTML ja disainimiseks CSS programmeerimiskeel.

JavaScript on universaalne: seda saab kasutada nii rakenduse esiliidese kui ka tagarakenduse arenduses. [29] JavaScript keele plussiks on veebilehe interaktiivsuse tagamine [30]. See võimaldab lisada kasutajaliidesesse komponente nagu näiteks hüpikaknad, navigeerimine menüüs ja andmete valideerimine [30].

Oma populaarsuselt on programmeerimiskeel JavaScript tõusnud. Toetudes StackOverflow läbiviidud uuringus, siis on näha, et 83 000 vastajast ligi 65% kasutab seda oma igapäevases töös arendajana. (Joonis 19)



Joonis 19. Programmeerimiskeelte kasutuse võrdlus [31]

Lõputöö autorid olid eelnevalt vähe kokku puutunud JavaScript programmeerimiskeelega, kuid kuna puudus kaasärääkimise võimalus projekti alguses, siis pidid autorid kiirelt selgeks õppima antud keele põhitõed interneti keskkonnas

leiduvate materjalide abil. Lisaks korraldati Pipedrive'i mentori poolt arendajatele töötuba, kus anti edasi harjutuste abil baastadmised JavaScript keele osas.

Jooniselt 19 on näha, et teisel kohal programmeerimiskeelte kasutamise võrdluses on HTML/CSS, mida autorid kasutasid veebilehte disainides. HTML on standardne märgistuskeel dokumentidele, mis on mõeldud veebis info kuvamiseks ja vaatamiseks, samuti aitab luua veebilehe struktuuri [32]. CSS on samuti märgistuskeel, mida kasutatakse koos HTML-ga. Selle abil luuakse disain veebilehtedele ning selleks kasutatakse erinevaid silte. Antud programmeerimiskeeltega olid autorid varem kokku puutunud ning nende kasutamine arenduses probleeme ei tekitanud. Mõlemad on väga lihtsalt õpitavad ja jäävad arendajale kergelt külge. Autorid toovad välja, et Pipedrive'is oldud aeg on arendanud märkimisväärselt oskusi HTML ja CSS osas ning kasvanud on ka enesekindlus neid tulevastes arendusprojektides kasutada.

4.4.3 Arenduskeskkonna valik

Arenduskeskkonnana kasutatakse ettevõttes Visual Studio Code'i, mis on loodud Microsoft poolt. Antud arenduskeskkond on pärjatud 2019. aastal Stack Overflow arendusuuringu tulemusel kõige populaarsema arenduskeskkonna tiitliga [33].

Visual Studio Code kohta on välja toodud mitmeid positiivseid külgi.

- Tasuta allalaadimine kõikidele inimestele. [33]
- Lihtsus nii disainil kui ka kasutamisel. [33]
- Tuttav arenduskeskkond arendajatele, sest on ehitatud töölaarakenduste raamistiku Electron'i kasutades. [33]
- Lihtsalt kättesaadavad ja allalaetavad laiendused. [33]

Lõputöö autorid toovad plussidena välja lisaks eelnevalt toodule hea kasutatavuse mac OS operatsioonisüsteemiga ning mugavuse uute klassifailide ja kaustade loomisel. Visual Studio Code on üles ehitatud nii, et see toetab peaaegu kõiki populaarsemaid programmeerimiskeeli (näiteks C#, JavaScript, Python, CSS) ning ühte kausta võib lisada erinevas keeles kirjutatud faile [34]. Lisaks on loodud tarkvarale mugavus pakkuda välja soovitusi koodi parandamiseks ja silumiseks, mis tuleb kasuks koodi kirjutamisel.

4.4.4 Koodihalduskeskkonna valik

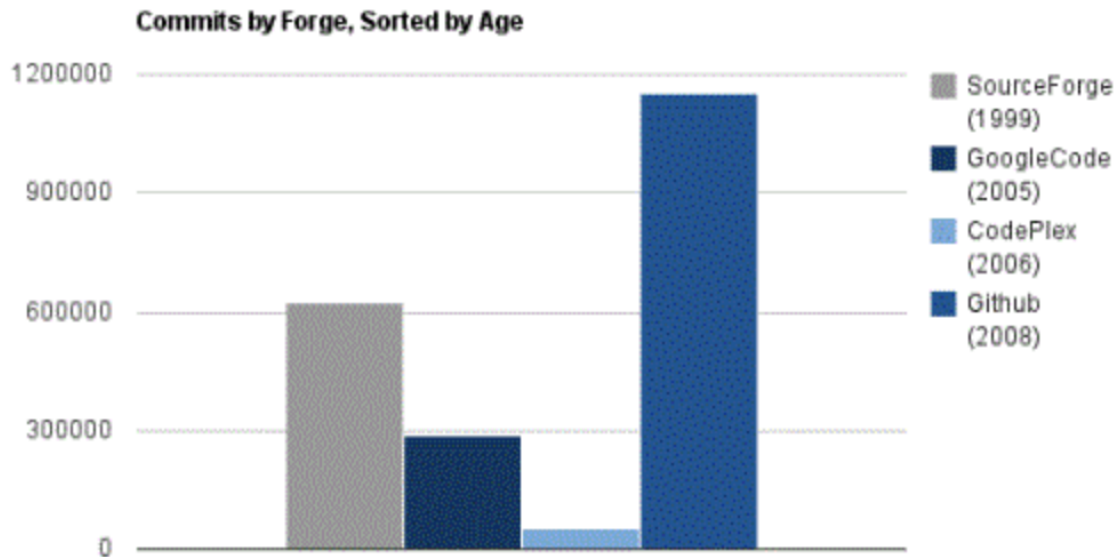
Koodihaldus keskkond oli samuti ettevõtte poolt ette antud, selleks oli GitHub. GitHub on veebipõhine platvorm, mis sisaldab giti versioonihaldus funktsioone, et neid saaks koos kasutada. See muudab giti abil koostöö tegemise lihtsamaks. GitHub hoiab koodi pilve-põhisel salvestusruumil, mis aitab mitmel arendajal korraga sama projekti arendada ning üksteise muudatusi reaalselt näha [35]. Pipedrive'is töötavad tihti ühe projekti kallal mitu arendajat, seega võimalus kahel arendajal samaaegselt arendada ühte projekti on oluline. GitHub'i peamised positiivsed omadused on järgnevalt välja toodud.

- Lihtsustab avatud lähtekoodiga projektidesse panuse andmist.
- Lihtsustab dokumentatsiooni hankimist.
- Võimalus jagada oma tööd potentsiaalsetele tööandjatele.
- *Markdown* ehk lihtteksti vormindamise süntaks võimaldab tekstiredaktorit kasutades vormindada tekste. [36]
- Koodi muudatusi on võimalik jälgida erinevates versioonides.
- Võimalik integreerida erinevate platvormidega, näiteks Google Cloud või Amazon. [36]

GitHubi puudusteks võib pidada järgnevaid tähelepanekuid.

- Turvalisus - turvarikkumised toimuvad ette teadmata paljudel saitidel.
- Hinnakujundus - mõned GitHub'i funktsioonid on SaaS-i tasulise seina taga ning soovituslik on kasutada oma sisemist giti. [37]

Pipedrive asutati 2010. aastal ja sel ajal oli GitHub kõige populaarsem koodihaldus keskkond olles kõigest kaks aastat vana (Joonis 20). See töötas Pipedrive'le, et GitHub on innovatiivne ning areneb koos nendega, ning seetõttu otsustati just GitHub'i kasutada igapäevases töös.



Joonis 20. 2012. aasta koodihaldus keskkondade kasutamise võrdlus [38]

4.5 Arhitektuur

Süsteemi arhitektuur hõlmab mitut mõõdet, millest peamised kaks on:

- Loogika arhitektuur, mis kirjeldab süsteemi selle korralduse kaudu kihtides, pakettides, põhiraamistikes, klassides, liidestest ja alamsüsteemis. [39]
- Juurutamise arhitektuur, mis kirjeldab süsteemi protsesside töötlemisüksustele jaotamise ja võrgu konfiguratsiooni kaudu. [39]

Antud lõputöö arhitektuur on kolmetasandiline (Joonis 21):

- Andmebaasi kiht, mille alla on kokku koondatud kõik vajalikud andmed Missions Tool rakenduse toimimiseks.
- Rakenduse loogika kiht, mille alla on kokku pandud kogu koodi osa, andmete küsimine ja nende töötlemine.
- Kasutajaliides, mis koosneb visuaalist.

Projekti arhitektuur on loodud ettevõtte Pipedrive arendajate poolt ja selle kaardistamine toimus lõputöö autorite poolt. Kolmetasandilise arhitektuuri ainulaadsus seisneb selles, et rakenduse loogika on eraldatud eraldiseisvaks keskmiseks kihiks. Liidese tasand on vaba rakenduse töötamisest: päringud edastatakse keskmisele kihile, mis omakorda saadab päringud edasi tagarakenduse salvestuskihile. [39]

KASUTAJALIIDES:

Create new Mission

General

Mission Name (required)	<input type="text" value="Your new mission"/>
Tribe	<input type="text" value="Tribe 1"/>
Location	<input type="text" value="Tallinn"/>
Duration in Weeks	<input type="text" value="6"/>
Status (required)	<input type="text" value="Draft"/>
Dev Team Size (required)	<input type="text" value="3"/>

RAKENDUSE LOOGIKA:

Arendusüksuste päring

Asukohtade päring

Töötajate päring

ANDMEBAAS:



Joonis 21. Kolmetasandiline arhitektuur.

Kolmetasandilise arhitektuuri kohta on välja toodud plussidena parem jõudlus, kättesaadavus ja mugavus. Arenduse teeb mugavaks see, et iga kihti saavad arendada samal ajal erinevad arendusmeeskonnad ja iga kihi programmeerimiskeel võib olla erinev. [40]

Sellise arhitektuuri peamiseks puuduseks on paindlikkuse puudumine rakenduse muutmisel. Iga kaasneva muudatusega tuleb arvestada rakendusekasutajate ootusega. [40]

Järgnevalt tuuakse välja, kuidas on ülesehitatud lõputöö arenduse loogika arhitektuur.

Loogika arhitektuur on üles ehitatud nii, et iga kindla objekti kohta käivad komponendid on koondatud ühte kausta. Näiteks kasutatakse Missions Tool rakenduses palju kalendrivaateid ja kõik komponendid, mis käivad kalendri vaadete kohta, on hoitud

kaustas „*Calendar*”. Nii on seal failid, mis on vajalikud inimeste ja projektide nimekirja kuvamiseks.

Kaustas „*assets*” hoitakse kõiki pilte ja ikoone, mida soovitakse veebirakenduses kuvada. Näiteks hoitakse seal Pipedrive’i logo või inimestele omastatavaid märke.

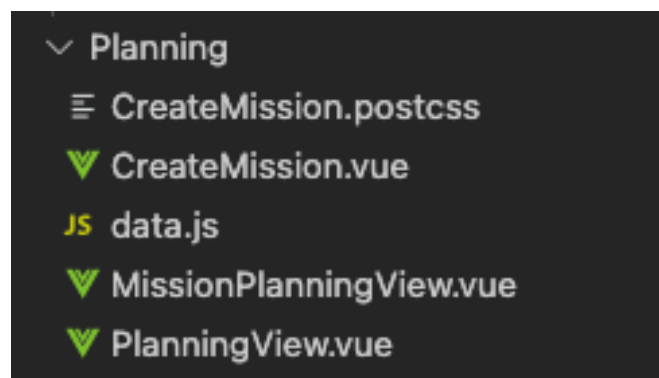
Kaustas „*store*” hoitakse kõiki päringuid, mis on tehtud API kaudu. Seal toimuvad andmete määramised ja juhul, kui andmeid pole võimalik andmebaasist kätte saada, määratakse vaikimisi väärtused.

Komponente, mida kasutatakse veebirakenduses sagedamini, hoitakse kaustas „*Common*”. Nii on sinna alla koondatud kogu disain ja kood erinevate nuppude kohta, välja on toodud rippmenüüde, tabelite ning hüpikakende disain ja kood. Kõikidele komponentidele on loodud oma disaini fail, mis hoiab tulevaste arendustegevustes koodi kirjutamist kokku ning aitab paremini koodi taaskasutada. Lisaks võimaldab selline komponentide hoidmine jälgida ühtlast joont veebirakenduse disainimisel ning hoiab ära näiteks samasuguse funktsionaalsusega nuppude erineva välja nägemise.

Sarnaselt eelnevalt kasutatavale arhitektuurile on ka lõputöö autorid kogu info koondanud projekti planeerimise ja loomise kohta kausta „*Planning*”. See võimaldab kõiki andmeid hoida ühes kohas ja lihtsustada pärimisi komponentide vahel. (Joonis 22)

Klassifaili *data.js* all hoitakse kõiki andmeid sisaldavaid konstante, mis saavad kätte vajalikud andmed statistika ja projektide täituvuse kuvamiseks. Selline lahendus aitab ära hoida püsiprogrammeerimist.

Igas kaustas on olemas disaini jaoks loodud Postgress CSS fail. Näites hoitakse disaini koodi *CreateMission.postcss* faili all, mis on loodud uue projekti loomise vormi jaoks.



Joonis 22. Failide ülesehitus kaustades.

Klassifaili *router.js* all on välja toodud kõikide erinevate vahelehtede nimetused ning nende URL-ide ja komponentidega.

App.vue on tavaliselt rakenduse juur ehk põhi ning see on määratletud Vue komponendi failivormingus.

Fail *package.json* sisaldab kogu projekti metaandmeid kui ka funktsionaalseid metaandmeid, näiteks projekti nimi ja kirjeldus ning sõltuvuste kui ka arendussõltuvuste loetelu koos nende versioonidega. Sõltuvus on teek, mida kogu projekt vajab tõhusaks toimimiseks, ning arendussõltuvus on pakett, mida arendaja arenduse ajal vajab. [41]

4.6 Kasutatud mustrid

Muster on läbiproovitud lahendus tüüpprobleemile ning on kasutatav erinevates kontekstides. Järgnevas peatükis tuuakse välja C. Larmani raamatu „Applying UML and Patterns” põhjal, kuidas projektis on rakendatud mõnda mustrit.

4.6.1 OOP - Pärilikkus

Toetudes raamatule „Applying UML and Patterns” on pärilikkus funktsioon, mille abil saab klasse spetsialiseerida üldisemateks superklassideks. [39] Ülemklasside atribuudid ja meetodid omandab alamklass automaatselt. [39] Selline lähenemine hoiab kokku koodi kirjutamist ja võimaldab ära hoida koodi dubleerimist samade funktsioonide ja nimetuste kaudu.

Missions Tool rakenduses on mitmeid lehekülgi, kus kuvatakse infot kalendrivaadetes. Välja võib tuua inimeste kalendrivaate, kus kuvatakse kuude ja nädalate kaupa, millal mõni töötaja viibib puhkusel, projektil või mõnel muul tegevusel. Kalendrivaate loomiseks on loodud neli kalendri komponenti: *CalendarEvent*, *CalendarTemplate*, *CalendarMain* ja *CalendarView*. *CalendarEvent* ehk rakenduse mõistes kalendriinfo, on komponent, mis hoiab enda all infot, mida kuvatakse veebirakenduse kalendrites. Vastav info (nt inimeste andmed, statistika andmed) määratakse ära objekti tüübina, mis joonisel on märgitud „objekt” (Joonis 23).

```

<KalendriInfo>
  <div :stiil="kalendriInfoStiil" :klass="klassiNimi">
    <component :andmed="andmed" :objekt="objekt" />
  </ KalendriInfo>

```

Joonis 23. KalendriInfo komponent.

CalendarMain ehk rakenduse mõistes põhikalender saab kalendriinfo komponendi käest kätte vajalikud andmed, mida tuleb kalendris kuvada. *For*-lausega saadakse kõik vajalikud väärtused kätte ning pannakse kalendris vastavalt järjekorda. Lisaks klassi nime all määratakse ära kujundus kalendrile. (Joonis 24)

```

<PõhiKalender>
  <Sektsioon klass="põhiKalender">
    <KalendriInfo
      :v-for="väärtus in statistikaVäärtused"
      :võti="väärtus.võti"
      :andmed="väärtus"
      :tüüp="väärtus.tüüp"
    />
  </Sektsioon>
</PõhiKalender>

```

Joonis 24. Põhikalendri komponent

CalendarTemplate ehk rakenduse mõttes *KalendriMall* on komponent, mis loob kalendri koos sinna käivate elementidega: aastad, kuud, päevad ja nädalad. Läbi selle komponendi määratakse ära, kas soovitakse kalendrit kuvada nädalate vaates või päevade vaates. Lisaks on võimalik määrata ära kalendris ka antud ajahetkel kalendrivaatamise päev, mis on tähistatud punase joonega. All olevas näites on välja toodud kõik võimalikud esitlused (Joonis 25).

```

<KalendriMall>
  <Aasta>
    <Kuu>
      <Nädal>
        <Päev>
        </Päev>
      </Nädal>
    </Kuu>
  </Aasta>
</KalendriMall>

```

Joonis 25. Kalendrimalli komponent

CalendarView ehk rakenduse mõistes kalendrivaade on komponent, kus kõik eelnevad, Kalendriinfo, Põhikalender ja Kalendrimall, pannakse üheks kokku ja luuakse täielik kalender. Juhul, kui kalendris soovitakse näiteks kuvada mõne kindla projekti andmeid, siis esmalt antakse sisendiks projektide nimekiri. (Joonis 26)

Edasi antakse kätte Kalendrimall, mis kuvab andmed vastavalt kasutaja soovile kas nädalate või päevade kaupa.

Põhikalendris saadakse kätte kogu info, mida soovitakse kalendris kuvada (Joonis 24).

```
<Kalendrivaade>
  <AndmeteNimekiri>
</AndmeteNimekiri>
  <div klass="KalendriKonteiner">
    <KalendriMall />
    <PõhiKalender />
  </div>
</Kalendrivaade>
```

Joonis 26. Kalendrivaate komponent

5 Töö tulemus

Järgnevas peatükis kirjeldatakse lahti milline on tehtud töö tulemus ja kui kaugele projektiga jõuti. Lisaks tuuakse välja võimalikud edasiarendused tulevikuks ja autorite hinnang tehtud tööle. Lõputöö raames on loodud eraldi andmefail, mis sisaldab võltsitud andmeid Pipedrive töötajate ja arendusüksuste kohta. Selle abil jääb ära ettevõtte töötajate ja arendusüksuste pärisnimede kuvamine.

Kõik peatükis esinevad koodinäited ja vaatepildid (välja arvatud joonis 39) on loodud antud lõputöö autorite poolt.

Koodinäited on eesti keeles, sest ettevõtte Pipedrive'i ja lõputöö autorite vahel on tingimus, et loodud koodi ei tohi avaldada avalikult.

5.1 Planeerimise vaates statistika kuvamine

Peale ITB1706 infosüsteemide arendamise meeskonnaprojekt: tellimus (2021/2022 sügis) aine lõppu jätkasid autorid projektiga ning täitsid meeskonnaprojekti aine raames seatud eesmärgid.

Statistika olemasolu oli vajalik eelseisvaks arenduseks. Enne uue projekti loomist oli vaja saada projekti loojal arendusüksuse kohta üldist infot, kellele uut projekti soovitakse luua. Statistika võimaldab uue projekti loomisel selle loojale anda esmast aimdust sobiva aja kohta, mil projekt läbi viia, võttes arvesse projektiks vajaminevate arendajate ning vabade arendajate arvu nädalate kaupa. Lisaks annab see projekti loojale teadmise, kui palju tuleks teistest arendusüksustest inimesi juurde leida. Statistikavaade oli vajalik ka projekti inimeste kutsumiseks. Seda seetõttu, et inimeste kutsumiseks on vaja projektide olemasolu, kuhu oleks võimalik neid kutsuda. Lisaks aitas statistika kiiremini kätte saada arendusüksuse vabade arendajate ning igapäevaste tööülesannetega tegelevate arendajate arvu. Kõik, kellel puuduvad projektid, kuvatakse projektile inimeste kutsumise vormil.

Planeerimisvaates oli eesmärk luua kalender, mis oli jaotatud kaheks osaks: arendusüksuse statistika ja arendusprojektide nimekiri koos täituvuse infoga. Kõik kuvatavad andmed saadakse automaatselt andmebaasist, mis hoiab ära käsitsi sisestamisel tekkivad vead. Andmed on kuvatavad nädalate kaupa. Lisaks on üle kalendri punane joon,

mis tähistab nädalat, milles vaadeldaval hetkel viibitakse. Statistika osas on kuvatud järgnevad väljad (Joonis 27):

- „*Engineers Headcount*” ehk arendajate arv meeskonnas.
- „*Launchpad Minimum Size*” ehk arendajate minimaalne arv, kellel puuduvad arendusprojektid.
- „*Launchpad Actual Size*” ehk arendajate arv, kellel puuduvad hetkel arendusprojektid.
- „*Need for Missions*” ehk arendajate arv, keda on vaja tulevasteks arendusprojektideks.
- „*Engineers on Mission*” ehk arendajate arv, kes on seotud mõne arendusprojektiga.
- „*Open Slots on Mission*” ehk meeskonnas käimasolevate arendusprojektide vabade kohtade arv.
- „*Outtribers on Mission*” ehk arendajate arv, kes ei kuulu vaadeldavasse meeskonda.
- „*Vacation*” ehk arendajate arv, kes viibivad puhkusel.
- „*Available for Missions*” ehk arendajate arv, kes saavad osaleda arendusprojektides.

STATISTICS	February		March					April			
	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17
Engineers Headcount	17	17	17	17	17	17	17	17	17	17	17
Launchpad Minimum Size	6	6	6	6	6	6	6	6	6	6	6
Launchpad Actual Size	13	13	13	13	13	13	13	13	13	13	13
Need for Missions	0	0	0	0	0	0	0	0	0	0	0
Engineers on Mission	4	4	4	4	4	4	4	4	4	4	4
Open Slots on Mission	0	0	0	0	0	0	0	0	0	0	0
Outtribers on Mission	0	0	0	0	0	0	0	0	0	0	0
Vacation	0	0	0	0	0	0	0	0	0	0	0
Available for Missions	6	6	6	6	6	6	6	6	6	6	6

Joonis 27. Statistika väljad ja andmed nädalate kaupa.

Igale statistika rea nimetusele on juurde lisatud kirjeldus, mis selgitab täpsemalt, mida reaväärtus tähendab (Joonis 28). Infonupp tuleb nähtavale siis, kui hiirekursorit selle peal hoida. Lisaks juhul, kui projektis osaleda saavate arendajate arv on üks või null, kuvatakse nende arv punase värvusega (Joonis 29). See annab märku, et arendusüksuses on koormus liiga suureks läinud ning uute projektide jaoks ei jätku piisavalt arendajaid.

STATISTICS	February		
	W6	W7	W8
Engineers Headcount	14	15	15

Total number of engineers in the tribe

Joonis 28. Statistika veeru kirjeldus.

STATISTICS	2022				February		
	W1	W2	W3	W4	W5	W6	W7
Engineers Headcount	17	17	16	16	16	16	16
Launchpad Minimum Size	6	6	6	6	6	6	6
Launchpad Actual Size	13	10	10	9	9	7	7
Need for Missions	2	2	2	4	4	4	4
Engineers on Mission	5	5	5	7	7	7	7
Open Slots on Mission	2	2	2	2	2	2	2
Outtribers on Mission	1	1	1	1	1	1	1
Vacation	3	2	2	0	0	0	2
Available for Missions	3	4	3	3	3	3	1

Joonis 29. Punane värvus, kui arendajate arv on üks või null.

Arendusprojektide osas on välja toodud nimekiri kõikidest osakonna projektidest. Kalendrivaates kuvatakse projekti kestvuse nädalatel arendusprojektis vajaminevate arendajate ja juba olemasolevate arendajate arvu suhe. Lisatud on funktsioon, mis kuvab arendusprojektide täituvuse arvu juures punast värvust ja hoiatusteadet, kui kaks nädalat

enne projekti algust on üks arendaja projektist puudu. Juhul, kui andmebaasis puudub info arendajate arvu kohta, kuvatakse küsimärk „?”. (Joonis 30)

 Ideas Exploration Missions Tribes People Planning Badges Hackathon 																											
blanditiis																											
STATISTICS	September				October				November				December				2022				February						
	W06	W07	W08	W09	W10	W11	W12	W13	W14	W15	W16	W17	W18	W19	W20	W21	W22	W23	W24	W25	W26	W01	W02	W03	W04	W05	W06
Engineers Headcount	21	21	21	26	26	26	25	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
Launchpad Minimum Size	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
Launchpad Actual Size	13	10	10	16	19	19	13	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
Need for Missions	7	7	7	7	7	7	9	9	9	9	5	5	5	5	5	3	3	3	3	3	3	3	3	3	0	0	0
Engineers on Mission	6	6	6	6	6	10	12	12	13	13	9	9	9	5	5	3	3	3	3	3	3	3	3	3	0	0	0
Open Slots on Mission	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Outtribers on Mission	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
Vacation	1	1	2	3	4	6	2	4	1	1	0	0	0	3	3	1	0	1	0	0	0	0	0	0	0	0	0
Available for Missions	7	7	6	10	9	3	4	1	3	3	8	8	8	9	9	13	14	13	14	14	17	17					
MISSIONS (5) +																											
Mckayla49	●	4/?	4/?	4/?	4/?	4/?	4/?	4/?	4/?	4/?	4/?	4/?	4/?	4/?													
Toney_Weber	●	2/3	2/3	2/3	2/3	2/3	2/3	2/3	2/3																		
Leta_Pacocha17	●	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4																
Vito.Murray	●					2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2	2/2											
Linwood_Kuhlman56	●					3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3				

Joonis 30. ITB1706 aine planeeritud “Planning” vaade

Järgnevalt tuuakse koodinäidetel välja, kuidas tekitati kastidena statistika ja projektide täituvuse info kuvamine.

Meetod *tegevusteStiiliMaaramine()* on jaotatud kaheks osaks: tegevused tagastatakse ühtse reana kalendrivaates või lõigatakse eraldi kastideks. (Joonis 31)


```

export function tegevusteStiiliMaaramine(tegevused) {
    tegevused.forEach((t) => {
// tegevuse kuvamine ühtse reana
t.tegevused = t.tegevused.map((m) => {
        return { ...m, stiil: kohandatudStiil(m.stiil)
    });
});
// lõigatakse kastid eraldi tükkideks
t.tegevused = kastideksLõikamine(t.tegevused);
});
return tegevused;
}

```

Joonis 31. Tegevuste stiili määramise koodinäide disaini loomiseks

Ühtse reana kuvamine määratakse meetodis *kohandatudStiil()*. Elementidele määratakse nii parameetrite vasaku kui ka laiuse pikslite arv tegevuse kestvuse põhjal. Selle tulemusel kuvatakse vastavalt tegevuse pikkus. (Joonis 32)

```

export function kohandatudStiil(stiil) {
    return {
        left: maaraVasak(stiil.vasak),
        width: maaraLaius(stiil.laius),
    };
}

```

Joonis 32. Ühtse rea kuvamise koodi näide disaini loomiseks

Eraldi kastideks lõikamine toimub meetodi *kastideksLõikamine()* abil. Kõik elemendid, mis koodi näites on nimetatud tegevusteks, tagastatakse massiivina. Nii parameeter vasak kui ka laius määratakse kasutades kalendris nädala kasti mõõtmeid. Võtme abil pannakse kastid üksteise kõrvale. (Joonis 33)

```

export function kastideksLõikamine(tegevused) {
  const massiiv = [];
  tegevused.forEach((teg) => {
    if (teg.stiil && teg.vasak >= 0 && teg.laius){
      for (let i = 0; i <teg.laius/nädala_laius; i++) {
        massiiv.push({
          ...teg,
          vöti: `${teg.vöti}-${indeks + 1}`,
          stiil: { vasak: teg.vasak + nädala_laius
            * indeks, laius: nädala_laius },
        });
      }
    }
  });
  return massiiv;
}

```

Joonis 33. Kastideks lõikamise koodinäide disaini loomiseks

5.2 Uue projekti lisamine

Navigeerides arendusüksuse leheküljelt rohelise pluss nupu kaudu uue projekti loomise lehele, avaneb vaade vormile (Joonis 34).

The screenshot shows a 'Create New Mission' form with the following fields and values:

- Mission Name (required): Uus Missioon
- Tribe: blanditis
- Location: Tallinn
- Duration in weeks: 7
- Status (required): IN PREPARATION
- Dev Team Size (required): 5
- Front End: 2
- Back End: 3
- Description: Uus Missioon loomisel

Below the form is a calendar view showing the mission duration from May to June. A green bar highlights the mission period from the end of May to the end of June. The calendar shows weeks from February to August.

Joonis 34. Loodud vorm uue projekti lisamiseks koos lisatud andmetega.

Loodud on väljad: projekti nimi, arendusüksus, asukoht, kestvus, staatus, arendusmeeskonna suurus, esi- ja tagarakenduse arendajate arv ning kirjeldus.

Automaatselt on kuvatud arendusüksuse teksti kastis nimi, kelle üksuse planeerimislehel soovitakse uut projekti luua.

Rippmenüüna on loodud arendusüksuse, asukoha ja staatuse valimine ning kõikide nende kolme andmed saadakse kätte läbi andmebaasi. Pipedrive arendusüksuste õiged nimed on asendatud valeandmetega, mis on kuvatud rippmenüüs.

Peale vormiväljade on lisatud vaatele kalender, mis genereerib automaatselt projekti soovitud kestvuse põhjal sobiva vahemiku projekti toimumiseks. Antud vahemikku on võimalik nihutada kalendris meelepärasemale ajale. Hetkel on lahendus poolik, sest ei arvestata arendajate arvu projekti toimumisaja genereerimisel.

Vormi loomisel on arvestatud sellega, et juhul, kui veebiakna suurst muudetakse, siis kõiki välju kuvatakse korrektselt.

Vältimaks olukorda, kus projekti loomisel unustatakse kirja panna kõige tähtsam info, on muudetud väljad projekti nimi ja arendusmeeskonna suurus kohustuslikuks. Ilma nende andmeteta uut projekti luua ei saa ning selle ära hoidmiseks kuvatakse hoiatussõnum „*Save and publish*” nuppu vajutades. Juhul, kui kõik kohustuslikud andmed on valitud, siis vajutades sama nuppu lisatakse uus projekt andmebaasi.

Nuppu „*Cancel*” vajutades liigutakse tagasi esialgsele planeerimislehele, kus kuvatakse arendusüksuse statistikat ja projekte.

Kogu kasutajaliidese arengul on arvestatud rakenduse Missions Tool eelnevalt kasutatud disainiga, et säilitada rakenduse ühtsus disainis. Taaskasutatud ja kohandatud on olemasolevaid komponente ja muudetud neid uuele vaatele omasemaks, lisades juurde näiteks vajalikke välju.

Rippmenüüs on kuvatud andmed otse andmebaasist ning osade andmete puhul võivad need olla vananenud. Nii oli hetkel asukohtade andmestikuga, mis hetkel sisestatakse projekti looja poolt käsitsi. Kõik sisestatud andmed koondatakse kokku andmebaasi, mille tõttu esineb seal palju kordusi ja vananenud asukohti. Selle vea parandamiseks loodi korrektsete asukohtade nimekiri andmebaasi.

Kõik sisestatavad andmed peavad olema kooskõlas andmebaasis olevate tabelite väljade tüüpidega. Hetkel on arenduses jäänud vale tüüp väljale arendajate tiimi suurus, kuid kui andmed salvestatakse andmebaasi, siis see muudetakse automaatselt numbri tüüpi, mis on ka õige.

5.3 Inimeste kutsumine

Navigeerides arendusüksuse leheküljelt projekti kõrval oleva lisamise nupu kaudu inimeste projekti kutsumise lehele. Seal kuvatakse otsinguväli, filtrid, inimeste nimekiri, sõnumivälja kast ning kaks nuppu, „Cancel” ja „Send invites via Slack”. (Joonis 35)

Invite people to mission

Search by person, mission, tribe etc.

Position Level Tribe Location Status

Showing 14 people

Name	Position	Level	Tribe	Location	Select
Amir Blick	Head of Engineering, Product Platform	Head	blanditiis	Tartu	<input type="checkbox"/>
Berta Ward	Junior Full Stack Developer	Junior	blanditiis	Tartu	<input type="checkbox"/>
Candido Emmerich	Front End Developer	Mid	blanditiis	Tartu	<input type="checkbox"/>
Carmen Runte	Back End Developer	Mid	blanditiis	Tartu	<input type="checkbox"/>
Cittalli Orn	Intern	Intern	blanditiis	Tartu	<input type="checkbox"/>






Hi! I will start a new mission called "My new mission" where I would like to invite you! More information and applying HERE or if you have any questions please contact me!

Cancel Send invites via Slack

Joonis 35. Vaikimisi inimeste kutsumise leht.







Otsinguvälja kaudu saab otsida inimesi nime, positsiooni, taseme, asukoha või arendusüksuse järgi.

Filtreid on kokku viis, arendajate saadavuse, nende positsiooni, taseme, arendusüksuse ning asukoha kohta. Kõik filtrid on loodud rippmenüuna, kus on kuvatud nimekiri kõigi võimalike väärtustega. (Joonis 36)

Position	Level	Tribe	Location	Status
Showing 14 people				
Name	Position	Tribe	Location	Select
 Amir Blick	Head of Engineering, Product	blanditiis	Tartu	<input type="checkbox"/>
 Berta Ward	Junior Full Stack Developer	blanditiis	Tartu	<input type="checkbox"/>
 Candido Emmerich	Front End Developer	blanditiis	Tartu	<input type="checkbox"/>
 Carmen Runte	Back End Developer	Mid	Tartu	<input type="checkbox"/>
 Citlalli Orn	Intern	Intern	Tartu	<input type="checkbox"/>

Joonis 36. Filtri rakendamine arendusüksuste järgi filtreerimiseks.

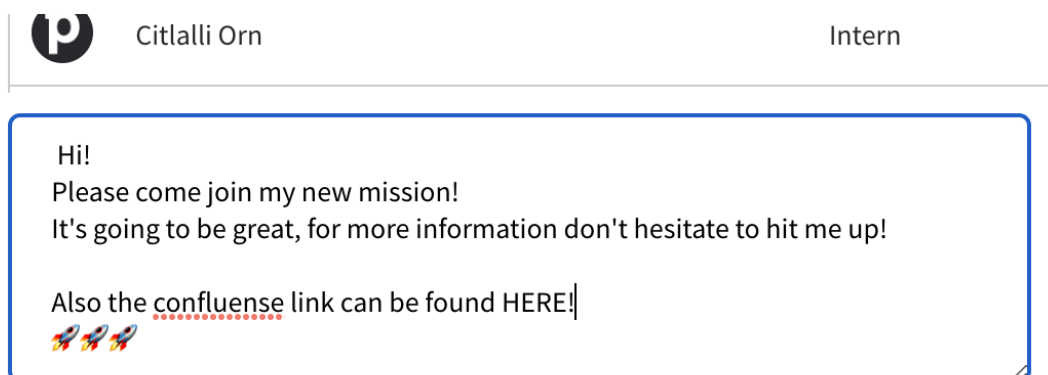
Inimeste nimekirjas on kuvatud iga inimese pilt, täisnimi, positsioon, tase, arendusüksus, asukoht ning kastike, kuhu saab lisada linnukese, kui soovitakse seda inimest projektile kutsuda. Linnukesi saab lisada ühele või ka mitmele inimesele. Vaikimisi on kuvatud vastava arendusüksuse need inimesed, kes tegelevad igapäevaste tööülesannetega ning saavad osaleda uues projektis. Visuaali parandamiseks on inimeste nimekirjas korraka nähtaval viis inimest. Juhul, kui inimesi on rohkem kui viis, siis on võimalik kerida üles-alla, et neid kõiki näha. (Joonis 37)

Name	Position	Level	Tribe	Location	Select
 Berta Ward	Junior Full Stack Developer	Junior	blanditiis	Tartu	<input checked="" type="checkbox"/>
 Candido Emmerich	Front End Developer	Mid	blanditiis	Tartu	<input type="checkbox"/>
 Carmen Runte	Back End Developer	Mid	blanditiis	Tartu	<input checked="" type="checkbox"/>
 Citlalli Orn	Intern	Intern	blanditiis	Tartu	<input checked="" type="checkbox"/>
 Cordie Stark	Back End Developer	Mid	blanditiis	Tartu	<input type="checkbox"/>
 Eduardo Stamm	Senior Back End Developer	Senior	blanditiis	Tallinn	<input type="checkbox"/>

Joonis 37. Inimeste nimekiri valitud inimestega.

Iga inimese kohta on võimalik vaadata tema detailset infot. Selleks peab vajutama soovitud inimese nime peale ning seejärel avaneb hüpikaknana kogu info inimese kohta. Seal on näha inimese nimi, amet, e-mail ning samuti inimese ajalugu ettevõttes ja osaletud projektides.

Sõnumivälja eesmärk on anda kutsutud inimestele lühike ülevaade millise projektiga neid osalema kutsutakse. Sõnumiväljal on vaikumisi kuvatud sõnumi mustand, mida on võimalik igal hetkel muuta personaalsemaks, kui selleks on soovi. (Joonis 38)



Joonis 38. Slack'i sõnumikast

Nuppu „Cancel” vajutades liigutakse tagasi esialgsele planeerimislehele, kus kuvatakse arendusüksuse statistikat ja projekte. Sellega tühistatakse ka kõik valikud ning uuendatud sõnum.

Vastava lehe arendamisel on arvestatud rakenduses Missions Tool kasutatava disainiga. Mitmed komponendid olid rakenduses juba olemas, selle vaate jaoks pidi neid kohandama, lisama või eemaldama vajalike välju. Uute komponenti lisamisel võeti arvesse rakenduses kasutatav disain, et säilitada ühtne disain üle kogu rakenduse.

Nagu ka eelnevalt mainitud võib andmebaasides olla vananenud info, mis tekitas filtritele õigete väärtuste kuvamisega probleeme. Töötajate positsioon ning tase on tavaliselt sisestatud arendusüksuse juhi poolt manuaalselt. Seega nii nagu asukohtadega, oli ka nendes palju kordusi andmebaasis. Seetõttu loodi ka võimalike positsioonide ning tasemete kohta uus nimekiri andmebaasi.

Inimeste nimekirja kuvamisel tekkis probleeme peamiselt kogenumatuse tõttu. Eelkõige moodustasid autorid mitmeid keerukaid meetodeid, et projekti kaudu saada kätte arendusüksus, kus projekt toimub ning seejärel arendusüksuse kaudu inimesed, kes sellesse kuuluvad. Lõpuks selgus, et kogu seda protsessi sai teha ühe lausega, sest igal arendusüksusel on juurde lisatud inimeste nimekiri. Projektis osaleda saavate inimeste

kuvamiseks oli vaja filtreerida arendusüksuse inimeste hulgast välja need, kelle staatuseks oli „Launchpad“.

5.4 Võimalikud edasiarendused

Järgnevalt kirjeldatakse lahti statistika, projekti loomise, inimeste kutsumise ja projekti muutmise võimalikud edasiarendused ning tuuakse välja, mis jäi arenduses veel tegemata.

5.4.1 Statistika vaate edasiarendused

Statistika vaatele on planeeritud veel nimekiri projektidest, mille on loonud teine arendusüksus, kuid seal osalevad vaadeldava üksuse arendajad. Näiteks kui Arendusüksus 1 arendaja osaleb Arendusüksus 2 loodud projektil, siis on seda projekti näha ka Arendusüksus 1 statistika vaates eraldi kalendris (Joonis 39).

More ▾	March				April					
	W12 21.03-25.03	W13 28.03-01.04	W14 04.04-08.04	W15 11.04-15.04	W16 18.04-22.04	W17 25.04-29.04	W18 02.05-06.05	W19 09.05-13.05	W20 16.05-20.05	W21 23.05-27.05
TRIBE STATISTICS										
Engineers headcount	20	20	20	20	20	20	20	20	20	20
Launchpad minimum size	4	4	4	4	4	4	4	4	4	4
Launchpad actual size	18	18	16	17	17	8	8	9	9	19
Needed for missions	2	2	0	3	3	3	3	3	3	1
Engineers on mission	2	2	0	2	2	2	2	2	2	1
Open slots on missions	0	0	0	1	1	1	1	1	1	0
Outtribers on mission	0	0	0	0	0	0	0	0	0	0
Vacation	0	0	4	0	0	9	9	8	8	0
Available for missions	14	14	12	13	13	4	4	5	5	15
MISSIONS (2)										
Mission 1	2/2	2/2								
Mission 2				2/3	2/3	2/3	2/3	2/3	2/3	
LOANED OUT (2)										
Other tribe mission 1										1
Other tribe mission 2 Tribe 1										

Joonis 39. Statistika vaade koos “Loaned out” osaga [12]

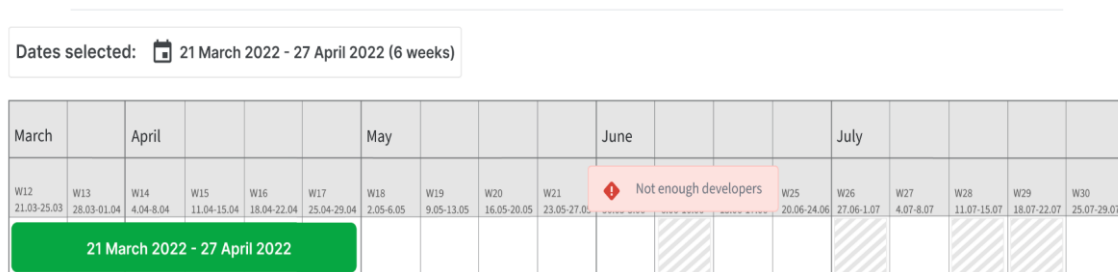
Samuti on planeeritud lisada statistika vaate kõige viimaseks veel inimeste kalendrivaade, mis on täpselt samasugune nagu eelnevalt näidatud arendusüksuse kalendrivaade (Joonis

3). Seda kalendri vaadet saab vajadusel noolele vajutades sulgeda või avada. Sellise edasiarenduse eesmärgiks oleks kogu info kokku koondamine ühele lehele ja juhul, kui kasutajatele on selline lahendus mugav, siis tulevikus eemaldada arendusüksuste inimeste kalendrivaade arendusüksuse lehel.

Statistikavaates olev projektide kalendriosa planeeriti teha muudetavaks ehk otse kalendri vaates saab muuta projektide toimumise aega. Selleks tuleb vajutada „*edit*“ nupule, mis muudab projektid aktiivseks. Seejärel saab nihutada projekti aega edasi-tagasi uuele ajale kui ka venitada projekt pikemaks või lühemaks, muutes alguse või lõpu aega.

5.4.2 Projekti loomine

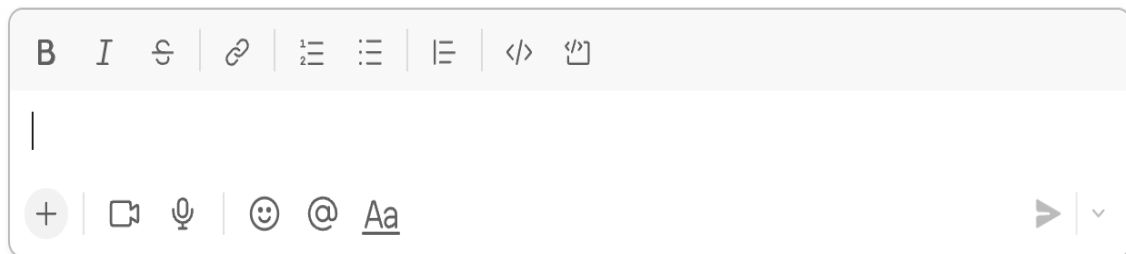
Projekti loomise vormil jäid puudujäägid kalendril, mis võimaldab loojal sobivat aega valida. Hetkel pannakse ajavahemik paika nädalate põhjal, mis sisestatakse vormil. Lisaks puudub loodud vormil halliga tähistatud ala, mis märgib ära nädalad, kus projekti tegemine on ebasobiv seoses ressursside puudumisega. (Joonis 40)



Joonis 40. Kalender projekti toimumisaja valimiseks.

5.4.3 Inimeste kutsumine projekti

Inimeste projekti kutsumise vormil on veel mitmeid edasiarendusi. Sõnumiväljale oli plaanis lisada tekstiredaktor, lisades sinna kõik võimalused, mis on ka näiteks Slack'i sõnumikastil. Sinna hulka kuuluvad teksti muutmine paksuks, kaldkirjaks, läbijooneitud kirjaks, nummerdatud listiks või hoopis koodiks. Samuti võimalus lisada emotikone, linke, faile, pilte või helisalvestisi sõnumile ning ka mainida inimesi. (Joonis 41)



Joonis 41. Slack'i korrektne sõnumikast.

5.4.4 Projekti muutmine

Lisaks projekti loomisele ja inimeste kutsumisele, oli prototüübile toetudes plaanis ka luua projekti muutmise võimalus. Antud lahenduse loomist saab alustada siis, kui on valmis uue projekti loomise lehekülge, sest selle jaoks loodud komponenti tuleb kasutada ka muutmisvõimaluse loomisel.

Muutmine toimub sarnaselt projekti loomisele ja inimeste kutsumisele läbi arendusüksuse planeerimislehe, kus vajutatakse muutmist vajava projekti peale ning viiakse vajalikud andmemuudatused sisse.

5.5 Tulu teenimisvõimalus

Lõputöö autorid leiavad, et antud rakendust saaks kasutada ka ettevõttele tulu teenimiseks seda teistele firmadele edasi müües või määrata rakenduse kasutamiseks kuutasu. Tasu määramiseks tuleb välja selgitada halduskulud. Lisaks on rakendus pidevas ühenduses tarkvarade Bamboo ja Confluence'ga ning seetõttu peavad kasutama ka teised ettevõtteid neid platvorme.

5.6 Hinnang projekti teostamisele

Lõputöö projekti alguses rakendati paariprogrammeerimist statistika kuvamise arendamiseks. Hiljem jaotati autorite vahel ülesanded ära: Mariliis asus arendama uue projekti loomise vormi ja Reia inimeste kutsumise vormi.

Lõputöö eesmärgid said suuremalt jaolt täidetud. Tehti valmis statistika, mis aitab uue projekti loomisel anda esmase ülevaate arendusüksuse ressurssidest. Üheks raskemaks osaks kujunes iga üksuse andmete kätte saamine andmebaasist ning nende korrektne

kuvamine vaates. Tuli luua andmebaasis puuduolevatele väärtustele meetodid, mis arvutatakse välja korrektsed numbrid ning hoida andmed ajaga kooskõlas. Autorid on rahul vaate disaini poolega ning loodud andmete korrektse kuvamisega. Loodi meetodid, mis kuvavad kastidena vajalikud numbrid koos õige värvusega. Statistika vaate üheks miinuseks on veebilehe laadimise kiirus, mis hetkel võtab aega viis sekundit suure andmehulga kätte saamise tõttu. Lisaks on puudu kaks kalendrit: vaadeldava arendusüksuse töötajate info ja nende projektide kalender, milles on arendajaid väljaspoolt arendusüksust.

Valmis sai loodud Mariliis Sinivee poolt põhi uue projekti lisamiseks. Mariliis jääb rahule Missions Tool rakenduse jaoks loodud komponentide korduvkasutamisega, sest sellega hoiti ära koodi dubleerimine. Olemasolevaid komponente tuli kohandada loodud vaatele; näiteks rippmenüü komponent arendusüksuse valikul. Lisaks sai algus loodud uuele funktsionaalsusele: projekti toimumisaja valimine kalendri abil. Hetkel on arendus pooleli ning info kuvamine toimub kestvuse sisendi põhjal. Lisaks esineb disainis iluvigu, näiteks topelt rida kalendril, mis tuleb tulevikus lahendada.

Valmis sai ka Reia Rõõmuse poolt tehtud projektile inimeste kutsumise vorm. Reia jääb rahule olemasolevate komponentide korduvkasutamisega. Olemasolevatest komponentidest kasutati „*People*” vaates olevat inimeste nimekirja, kust filtreeriti välja ainult arendusprojektidest vabad olevad arendajad. Projekti kutsumiseks on võimalik vastavalt soovile valida nii üks kui ka mitu inimest. Lisaks valmis ka sõnumikast, kuhu on vaikimis ette kirjutatud valmisi sõnum, mida on võimalik muuta. Sõnumikasti arendus jäi kahjuks pooleli, see peaks sisaldama veel teksti vormindamise võimalusi. Samuti esineb iluvigu disainis, veebiakna suurust muutes ei mahu kõik komponendid vaatesse ära.

Lõputöö käigus õpiti kasutama mitmeid uusi platvorme ja tehnoloogiaid ning õpiti tundma autorite jaoks uut programmeerimiskeelt, JavaScript'i, ja täiendati teadmisi raamistikust Vue.js. Lisaks saadi proovida paarisprogrammeerimist ning praktiseerida Scrum teadmisi, mida varem kasutati ainult ülikooli projekte tehes. Seekord tuli oskusi rakendada päris tehnoloogia ettevõttes ja koos seal töötavate inimestega. Varasemalt on näiteks puudunud kogemus arenduses endale ise tööülesandeid kirjutada ning puudunud arusaam, millises mahus peaks üks ülesanne olema. Kuuldes nõuandeid ja soovitusi mentoritelt, osatakse nüüd palju paremini oma tööd planeerida.

Projekti üheks keerukamaks osaks oli suure arendusprojekti ülevõtmine, selle tundma õppimine ja arenduse jätkamine. Tuli aru saada, kuidas eelnevalt loodud meetodid töötavad, neid kohandada uute funktsionaalsuste loomiseks ning tagada rakenduse töötamine ka peale muudatusi. Lisaks tuli kätte saada andmebaasist vajalikud andmed ning aru saada, millised väärtused on kindlatele objektidele omandatud. Juhul, kui vajalikke väärtusi polnud loodud (näiteks statistika loomisel vabade arendajate arv), tuli genereerida meetod, mis arvutaks soovitud väärtuse välja.

Autorid mainivad, et selline kogemus olla uue funktsiooni juurutamisel algusest lõpuni, on kogemust andev ning avardab silmaringi.

Lõputöö projekti 100% teostamiseks jäi aega väheks ning töö tegemine oleks võinud olla pikemale perioodile jaotatud. Kooli poolse juhendaja ja ettevõtte mentoriga suhtlemine sujus hästi ning projekt valmis suuremate probleemideta.

6 Testimine

Järgnevas peatükis kirjeldatakse kuidas toimus antud arendusprojekti testimine ja milliseid loodud tarkvarasid selleks kasutati.

6.1 Integratsioonitestimine

Integratsioonitestimisel eemärk on kontrollida, et kõik komponentide vahelised vastavused on omased tarkvara disainile ning komponentide vaheline toime on korrektne. Integratsioonitestimine paljastab defekte liidestest, toob välja kasutamata jäänud funktsionaalsused ja märkamata jäänud kõrval efekte protseduuride ajal. [42]

Loodud projektis oli integratsioonitestimise eesmärk kontrollida kasutatud komponentide ühilduvust üksteisega. Loodud arhitektuuris on komponendid integreeritud kõik ühe korraga.

Iga uue komponendi loomisel anti kood mentoritele ülevaatomiseks, kus mentorid katsetasid kõik uued nupud ning lahtrid läbi ning vaatasid kuidas ja kas uued funktsionaalsused olemasolevatega koos töötavad. Samuti vaadati üle kas prototüüp ja loodud komponent on omavahel vastavuses ning töötavad ühtemoodi.

6.2 SonarQube

SonarQube on universaalne tarkvara koodi analüüsimiseks. See on avatud lähtekoodiga platvorm, mis on loodud tarkvara ettevõtte SonarSource poolt koodi kvaliteedi pidevaks kontrollimiseks. Tarkvara toetab rohkem kui 25 programmeerimiskeelt ja on kasutusel rohkem kui 85 000 ettevõttes. [43]

Koodi analüüsi tulemusel luuakse detailne raport, mis toob välja koodis leiduvad vead, koodi duplikeerimise, haavatavuse ja haisva koodi. Ühendades SonarQube'i GitHub tarkvaraga kuvatakse arendajale koodis peituvad vead, mis soovitatakse lahendada koheselt.

SonarQube'il on olemas nii head kui ka vead. Aastal 2019 viidi läbi uuring Brasiilia ja Para ülikoolide õpilaste poolt, milles uuriti arendajatelt kuidas nad kasutavad koodi analüüsimisel tarkvara SonarQube'i ning milliseid positiivseid ja negatiivseid nurki nad

välja toovad. Uuringus selgus, et ligi 80% vastanutest usuvad, et tarkvara poolt välja toodud vead on relevantssed puhta koodi kirjutamiseks ja kõik analüüsist tulnud soovitused viivad korrektse arhitektuurini [43].

Pipedrive ettevõttes kasutatakse SonarQube igal arendusprojektil ning nii kasutati seda ka antud lõputöö arendustegevuses. Peale igat muudatuste ühist salvestamist kontrollitakse kas lisatud koodis on koodi korduskasutust ja kuvatakse antud arv protsentidena. Lisaks on protsendina välja toodud uue koodi kattuvuse arv testidega. Maksimaalne lubatud koodi dubleerimise protsent on 0,4%. Lisaks tuuakse välja koodis leiduvate vigade arv, koodi haavatavus ja turvalisuse kuumkoht.

Lõputöö autorid toovad välja kasutatava koodi analüüsimise tarkvara head ja vead.

Positiivse küljena märgitakse ära turvatunne, et on olemas tarkvara, mis aitab tuvastada koodis leiduvaid vigu, mida arendaja esmalt ei märka. Koodi dubleerimishoiatus annab hea tava arendajale koodi korduvkasutada.

Miinus poolena märgitakse, et tekkivate vigade lahendamine võtab kauem aega, kui esialgu võis oodata. Probleemi likvideerimiseks tuleb leida parem lahendus või viis, kuidas koodi korduvkasutada. Tihti peale tuleb ka olemas olevat koodi kohandada, mille tõttu kulub aega olemasolevate failide ümber muutmiseks.

6.3 Snyk

Snyk on platvorm, mis tagab koodi, konteinerite, sõltuvuste ja infrastruktuuri kui koodi turvalisuse. Snyk on ühendatud Pipedrive's koodihaldus tarkvara GitHub'iga ja omab veebipõhiseid liidesed, mis kontrollivad juhuslikult valitud koodi mooduleid, et avastada välise sõltuvuste haavatavaid kohti. Selle saavutamiseks kasutatakse nimekirja, kuhu on üles loendatud levinumad haavatavused ja riskid, et hoida neid tarkvara arendusest eemal.

Uurides teiste kasutajakogemust Snyk'i kasutamisel koodi turvalisuse tagamisel, on tagasiside olnud positiivne. Näitena võib tuua Dubai Manipal Akadeemia kahe tudengi läbi viidud uuringus, kus kasutati Snyk tarkvara koodi haavatavuse kohtade leidmiseks. Uuringus kasutati sarnaselt Pipedrive'ile Snyk'i läbi GitHub keskkonna, kus kontrolliti automaatselt koodi turvalisust. Erinevus seisneb selles, et uuringus on kasutatud ainult Snyk tarkvara ega pole juurde võetud tarkvarasid turvalisuse tõstmiseks koodis (nagu

näiteks on Pipedrive'l ka SonarQube). Tagasisides tõdeti, et selline koodi turvalisuse tarkvara aitab ära hoida haavatavusi nagu „Ebakindel krüpteerimine”, „Ootamatu koodikäitavus” ja „Teenuse keeldumine”. Sellised tulemused tagavad rakenduse turvalisuse ja muudavad seda vastupidavamaks. [44]

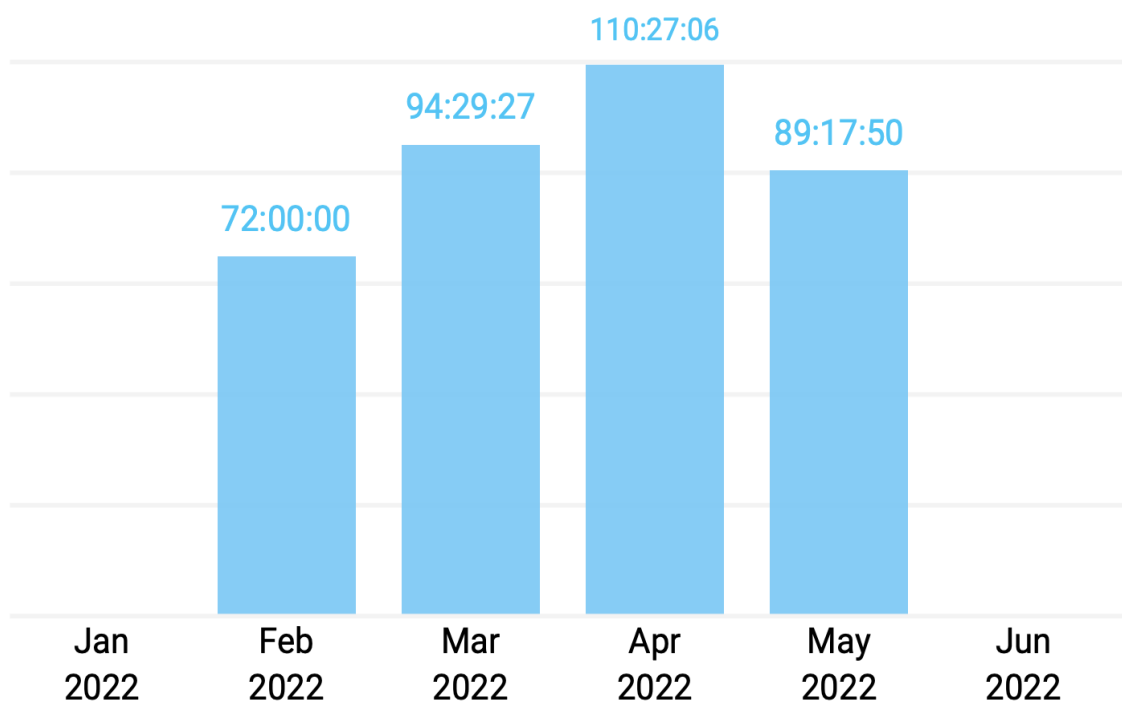
Juhul, kui esinesid Snyk koostatud aruannetes ohuteated, tuli need kiiremas korras eemaldada. Lõputöö arenduse käigus ühtegi ohuteadet ei kuvatud ning seetõttu puudub autoritel mõtteid Snyk'i kasutamisest tarkvara arendusel. Üks on kindel: inimvead on rakenduse ja tarkvara arendusel lihtsalt tekkima ning seetõttu on hea, kui taustal on olemas platvorm, mis kontrollib riske ja annab nende tekkimisest märku.

7 Teostatud tööde logi

Järgnevalt tuuakse välja autorite töötunnid. Kõik tunnid on jälgitud ja kirjapandud kasutades rakendust Toggle [45].

7.1 Mariliis Sinivee logi

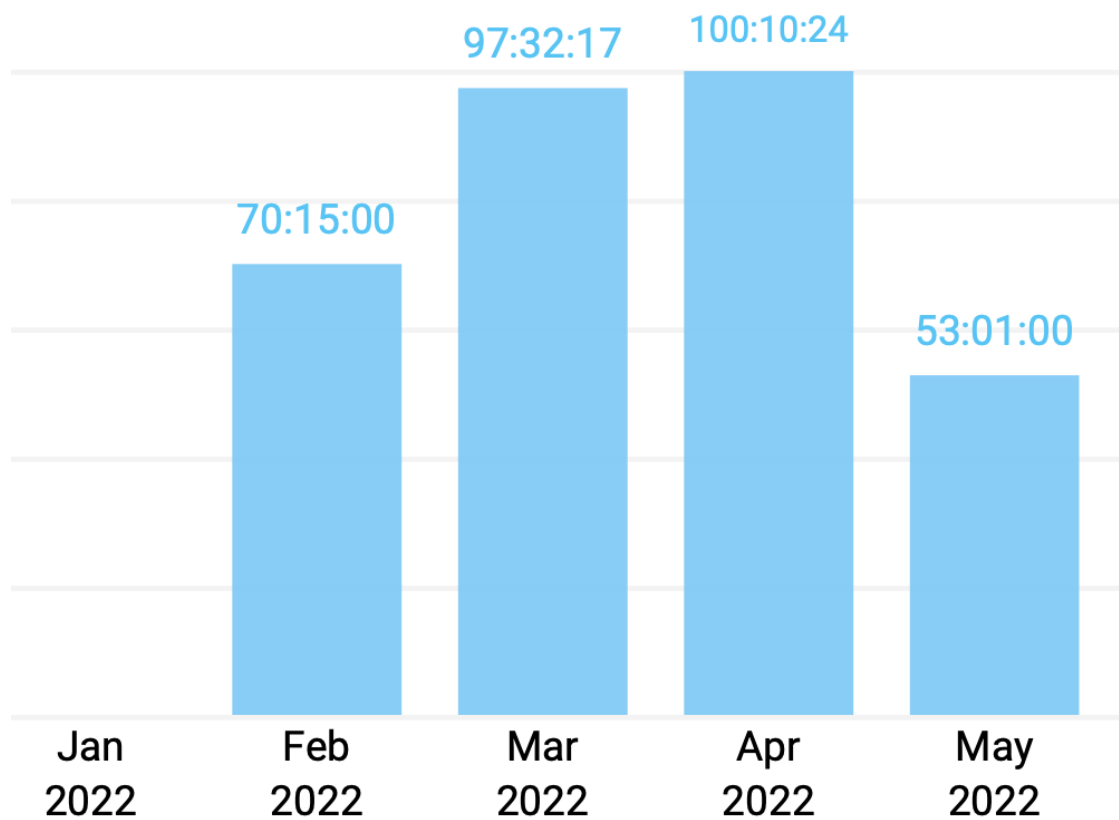
TOTAL HOURS: 366:14:23



Joonis 42. Mariliis Sinivee Toggle logi

7.2 Reia Rõõmuse logi

TOTAL HOURS: 320:58:41



Joonis 43. Reia Rõõmus Toggle logi

8 Tellija tagasiside

Järgnevalt tuuakse välja projekti tellija tagasiside tehtud tööle ja arendusprotsessile. Tagasiside andis ettevõtte Pipedrive poolne mentor.

“Tarkvara kasutajaliidese täiendamise eesmärgiks oli luua kasutajatele süsteem lihtsustamaks planeerimist eesootavate tööde jaoks. Tallinna Tehnikaiülikooli tudengid Reia Rõõmus ja Mariliis Sinivee löid vajalikud vaated, vormid ning otsingud arendatava planeerimissüsteemi jaoks. Üliõpilased näitasid süsteemi valmistamise käigus head analüüsisivõimet, oskust erinevate probleemidega toime tulla ning nendele lahendusi leida. Samuti mõeldi arenduse käigus, kuidas lõppkasutaja jaoks tehtav lahendus oleks kõige kasutajasõbralikum. Vaatamata sellele, et tudengid polnud varasemalt süvitsi kokku puutunud JavaScript ja Vue.js ning koodibaas, milles tuli muudatusi teha, oli üpriski suur, said üliõpilased uute lahenduse loomisega suurepäraselt hakkama.”

9 Kokkuvõte

Bakalaureusetöö eesmärgiks oli luua ettevõtte Pipedrive loodud veebirakendusele Missions Tool täiendus ressursside planeerimiseks. Valminud täiendus peaks võimaldama projekti loojal luua mugavalt uut projekti ning kutsuda lihtsalt ja kiirelt läbi rakenduse inimesi uude või juba olemasolevasse projekti. Siiani on puudunud võimalus rakenduses Missions Tool näha arendusüksuste kohta käivat üldist infot ning projekti loojatel on valitsenud teadmatus vabadest arendajatest, keda saaks projekti osalema kaasata. Lisaks puudub hetkel mugav süsteem inimeste kutsumiseks, nende poole peab isiklikult või läbi Slack keskkonna pöörduma.

Analüüsi osas tuuakse välja vajalikud nõuded loodavale kasutajaliidesele ja kirjeldatakse protsessi, kuidas nendeni jõuti. Lisaks tuuakse välja tootejuhtide loodud prototüüp igale vaatele, põhjalik analüüs kasutatavate tehnoloogiate kohta ja kirjeldatakse koodi arhitektuuri. Nii on välja toodud kasutatavad arendusmetoodikad, raamistik, programmeerimiskeeled, arendus- kui ka koodihalduskeskkond.

Käesoleva töö tulemusena loodi kasutajaliides uute projektide loomiseks ja nendesse inimeste kutsumiseks. Lisaks loodi statistika, mis kirjeldab andmeid iga arendusüksuse meeskonna kohta. Näiteks arendajate arvu, kellel puuduvad projektid, või puhkusel viibivate töötajate arv. Rakenduse funktsioonid, mis realselt valmis said, on välja toodud piltide abil. Lisaks on toodud välja autorite eneseanalüüs projekti teostamisel. Kuna loodud lõputöö maht oli piiratud, siis selle tulemusel jäid osad soovitud funktsioonid täitmata, näiteks projekti muutmise võimalus. Loodud rakendus on lihtsa ja selge struktuuriga ning vastab püstitatud eesmärkidele muuta mugavaks uute projektide loomine ja inimeste kaasamine.

Kasutatud allikad

- [1] Salesforce, „CRM 101: What is CRM?“, 12 Aprill 2022. [Võrgumaterjal]. Available: <https://www.salesforce.com/crm/what-is-crm/>. [Kasutatud 17 Veebruar 2022]
- [2] S. H. Perveez, „What is Git: Features, Command and Workflow in Git“, Simplilearn, 16 Veebruar 2022. [Võrgumaterjal]. Available: <https://www.simplilearn.com/tutorials/git-tutorial/what-is-git>. [Kasutatud 5 Mai 2022].
- [3] V. Vortel ja J. Laanpere, „Tarkvara areendusnõuded“, Tallinna Ülikool, HITSA, [Võrgumaterjal]. Available: <https://web.htk.tlu.ee/digitalu/testimine/back-matter/appendix/>. [Kasutatud 5 mai 2022].
- [4] Pipedrive OÜ, „Meist“, Pipedrive OÜ, [Võrgumaterjal]. Available: <https://www.pipedrive.com/et/about>. [Kasutatud 20 Märts 2022].
- [5] bambooHR, „About us“, bambooHR, [Võrgumaterjal]. Available: <https://www.bamboohr.com/about>. [Kasutatud 8 märts 2022].
- [6] Atlassian, „Why Confluence?“, Atlassian, [Võrgumaterjal]. Available: https://www.atlassian.com/software/confluence?&aceid=&adposition=&adgroup=99427413992&campaign=9616888075&creative=542596263067&device=c&keyword=confluence&matchtype=e&network=g&placement=&ds_kids=p52352124674&ds_e=GOOGLE&ds_eid=700000001542923&ds_e1=GOOG. [Kasutatud 24 märts 2022].
- [7] T. Dillion, „Project Resource Planning Quotes“, QuotesGram, [Võrgumaterjal]. Available: <https://quotesgram.com/img/project-resource-planning-quotes/10857955/>. [Kasutatud 17 Veebruar 2022].
- [8] analysistabs, „Resource Planning Template Excel“, analysistabs, [Võrgumaterjal]. Available: <https://analysistabs.com/project/plan/templates/excel/resource/>. [Kasutatud 17 Veebruar 2022].
- [9] Productivity Land, „18 Best Resource Management Software & Tools in 2022“, Productivity Land, 8 aprill 2021. [Võrgumaterjal]. Available: <https://productivityland.com/best-resource-management-software/>. [Kasutatud 17 veebruar 2022].
- [10] Meisterplan, „People-centric Portfolio Management Software“, Meisterplan, [Võrgumaterjal]. Available: https://meisterplan.com/?utm_term=meisterplan&utm_campaign=All++Brand++Meisterplan&utm_source=adwords&utm_medium=ppc&hsa_acc=8008683239&hsa_c

am=6764413588&hsa_grp=77663434737&hsa_ad=388940331242&hsa_src=g&hsa_tgt=kwd-336214702532&hsa_kw=meisterplan&hsa_. [Kasutatud 17 veebruar 2022].

- [11] monday, „Manage your team's resources on a collaborative platform,“ monday, [Võrgumaterjal]. Available: https://monday.com/lp/resource/comparison/bundle?marketing_source=adwordssearch&marketing_campaign=ww-s-resource_allocation-e-desk-monday&aw_keyword=resource%20planning%20software&aw_match_type=e&cluster=&subcluster=&gclid=CjwKCAiAgbiQBhAHEiwAuQ6BkmagYGCm. [Kasutatud 17 veebruar 2022].
- [12] A. Matkur ja L. Saluvere, „Ettevõttesise tarkvara täiendamine – kasutajaliidese ja andmebaasi kavandamine ning sobivuse hindamine,“ Tallinna Tehnikaülikool, Tallinn, 2022.
- [13] H. Hajjdiab ja A. S. Taleb, „Adopting Agile Software Development: Issues and Challenges,“ *International Journal of Managing Value and Supply Chains*, p. 11, september 2011.
- [14] Tallinna Ülikool, „Agiilne tarkvaraarendus,“ Tallinna Ülikool, 2019. [Võrgumaterjal]. Available: http://www.cs.tlu.ee/~inga/TTP/Agiilsed_meetodid_2019.pdf. [Kasutatud 18 märts 2022].
- [15] Agile Alliance, „Extreme Programming (XP),“ Agile Alliance, [Võrgumaterjal]. Available: [https://www.agilealliance.org/glossary/xp/#q=~\(infinite~false~filters~\(postType~\(~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video\)~tags~\(~'xp\)\)~searchTerm~'~sort~false~sortDirection~'asc~page~1\)](https://www.agilealliance.org/glossary/xp/#q=~(infinite~false~filters~(postType~(~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video)~tags~(~'xp))~searchTerm~'~sort~false~sortDirection~'asc~page~1)). [Kasutatud 18 märts 2022].
- [16] P. Valdmees, „Ekstreemprogrammeerimine: ülevaade, praktikad ja võrdlus teiste väledate arendusmeetoditega,“ Tallinna Ülikool, Tallinn, 2007.
- [17] T. Dudziak, „eXtreme Programming An Overview,“ *Methoden und Werkzeuge der Softwareproduktion*, 2000.
- [18] B. V. d. Carvalho ja C. H. P. Mello, „Scrum agile product development method - literature review, analysis and classification,“ University Center of Itajuba, Itajuba, 2011.
- [19] V. Stray, N. Moe ja A. Aurum, „Investigating daily team meetings in agile software projects,“ %1 *38th Euromicro Conference on Software Engineering and Advanced Applications*, 2012.

- [20] XP 2019 Workshops, „Agile Processes in Software Engineering and Extreme Programming – Workshops,“ %1 *XP: International Conference on Agile Software Development*, Kanada, 2019.
- [21] Scrum.org, „What is a Sprint in Scrum?,“ Scrum.org, [Võrgumaterjal]. Available: <https://www.scrum.org/resources/what-is-a-sprint-in-scrum>. [Kasutatud 19 märts 2022].
- [22] Agile pain relief, „Choosing a Scrum Sprint Length – Shorter Beats Longer,“ Agile Pain Relief, [Võrgumaterjal]. Available: <https://agilepainrelief.com/blog/choosing-scrum-sprint-length.html>. [Kasutatud 24 märts 2022].
- [23] Atlassian, „What is Jira used for?,“ Atlassian, [Võrgumaterjal]. Available: <https://www.atlassian.com/software/jira/guides/use-cases/what-is-jira-used-for>. [Kasutatud 9 mai 2022].
- [24] Education Ecosystem, „REACT.JS HISTORY,“ Education Ecosystem, [Võrgumaterjal]. Available: <https://educationecosystem.com/guides/programming/react-js/history>. [Kasutatud 23 aprill 2022].
- [25] N. I. Naim, „ReactJS: An Open Source JavaScript Library for Front-end Development,“ Metropolia University of Applied Sciences, Helsinki, 2017.
- [26] S. Azam, „What is Vue.js, and Why is it Cool?,“ 2020. [Võrgumaterjal]. Available: https://linuxhint.com/about_vue_js/. [Kasutatud 1 mai 2022].
- [27] E. Wohlgethan, „Supporting Web Development Decisions by Comparing Three Major JavaScript Frameworks: Angular, React and Vue.js,“ Hamburg University of Applied Science, Hamburg, 2018.
- [28] J. Dakowicz, „REACT VS VUE: WHICH ONE TO CHOOSE IN 2022?,“ Pagepro, 15 veebruar 2022. [Võrgumaterjal]. Available: <https://pagepro.co/blog/react-vs-vue-comparison/>. [Kasutatud 24 aprill 2022].
- [29] GeeksforGeeks, „Top 7 Programming Languages for Backend Web Development,“ GeeksforGeeks, 13 veebruar 2022. [Võrgumaterjal]. Available: <https://www.geeksforgeeks.org/top-7-programming-languages-for-backend-web-development/>. [Kasutatud 1 mai 2022].
- [30] K. Schroeder ja K. Sugiura, „What Is JavaScript Used for in Front-end Web Development?,“ Hileman Group, 11 juuli 2018. [Võrgumaterjal]. Available: <https://www.hilemangroup.com/Thought-Leadership/Hilelights-Blog/JavaScript-and-Front-end-Development>. [Kasutatud 1 mai 2022].

- [31] Stack Overflow, „2021 Developer Survey,“ Stack Overflow, mai 2021. [Võrgumaterjal]. Available: <https://insights.stackoverflow.com/survey/2021#technology>. [Kasutatud 1 mai 2022].
- [32] GeeksforGeeks, „Advantages and Disadvantages of HTML,“ GeeksforGeeks, 1 veebruar 2022. [Võrgumaterjal]. Available: <https://www.geeksforgeeks.org/advantages-and-disadvantages-of-html/>. [Kasutatud 1 mai 2022].
- [33] D. Cangemi, „The Reasons why you Must Use Visual Studio Code,“ Dev Genius, 29 juuli 2020. [Võrgumaterjal]. Available: <https://blog.devgenius.io/the-reasons-why-you-must-use-visual-studio-code-b522f946a849>. [Kasutatud 1 mai 2022].
- [34] P. Pedamkar, „What is Visual Studio Code?,“ Educba, [Võrgumaterjal]. Available: <https://www.educba.com/what-is-visual-studio-code/>. [Kasutatud 1 mai 2022].
- [35] Kinsta, „Git vs Github: What’s the Difference and How to Get Started with Both,“ Kinsta, aprill 2022. [Võrgumaterjal]. Available: <https://kinsta.com/knowledgebase/git-vs-github/>. [Kasutatud 1 mai 2022].
- [36] E. Novoseltseva, „Benefits Of Using Github,“ Apiumhub, 30 november 2020. [Võrgumaterjal]. Available: <https://apiumhub.com/tech-blog-barcelona/using-github/>. [Kasutatud 1 mai 2022].
- [37] J. Clancy, „The Advantages and Disadvantages of Using GitHub,“ CodeClouds, 19 november 2018. [Võrgumaterjal]. Available: <https://www.codeclouds.com/blog/advantages-disadvantages-using-github/>. [Kasutatud 1 mai 2022].
- [38] A. Vitus, „GitHub: The New Locus of Software Development,“ Scale, 10 jaanuar 2012. [Võrgumaterjal]. Available: <https://www.scalevp.com/blog/github-the-new-locus-of-software-development>. [Kasutatud 1 mai 2022].
- [39] C. Larman, Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, Pearson, 2004.
- [40] Leally, „KOLMETASANDILINE ANDMEBAASI ARHITEKTUUR. ANDMEBAAS,“ Leally, 2021. [Võrgumaterjal]. Available: <https://leally.ru/et/excel/trehurovnevaya-arhitektura-bazy-dannyh-bazy-dannyh/>. [Kasutatud 24 aprill 2022].
- [41] GeeksforGeeks, „Difference between dependencies, devDependencies and peerDependencies,“ GeeksforGeeks, 27 jaanuar 2022. [Võrgumaterjal]. Available: <https://www.geeksforgeeks.org/difference-between-dependencies-devdependencies-and-peerdependencies/>. [Kasutatud 24 aprill 2022].

- [42] A. Orso, „Integration Testing of Object-Oriented Software,“ POLITECNICO DI MILANO, Milano.
- [43] D. Marcilio, R. Bonifácio, E. Monteiro, E. Canedo, W. Luz ja G. Pinto, „Are Static Analysis Violations Really Fixed? A Closer Look at Realistic Usage of SonarQube,“ *IEEE Xplore*, 29 august 2019.
- [44] Z. Ahmed ja S. C. Francis, „Integrating Security with DevSecOps: Techniques and Challenges,“ *IEEE Xplore*, 2 juuni 2020.
- [45] Toggl, „Any Job is a Dream Job,“ Toggl, [Võrgumaterjal]. Available: <https://toggl.com/mission/>. [Kasutatud 10 mai 2022].

Lisa 1 – Enda panuse kirjeldus ja eneseanalüüs, Mariliis Sinivee

Meie arendustegevus algas ettevõttes Pipedrive sügisel, kui liitusime ainega ITB1706 infosüsteemide arendamise meeskonnaprojekt: tellimus (2021/2022 sügis). Selle aine ajal oli meie ülesandeks saada planeerimisvaade koos arendusüksuste statistikaga ning selle lahendamiseks hakkasid kõik arendajad pihta. Ülesanne osutus oodatust keerulisemaks ja aega nõudvamaks ning seetõttu jätkasime sellega ka peale meeskonnaprojekti ainet.

Statistika ülesandega töötati kõik koos ühiselt. Ülesandeks oli luua statistika kalendrisse nimekiri kõikidest vajalikest andmeväljadest ning saada kuvama vastavad väärtused. Lõime meetodid ja klassifailid, mille kaudu saadi kätte andmebaasist vajalikud andmed arendusüksuste kohta, mis on ka seletatud peatükis 5. Andmed, mis olid väärtalt lisatud andmebaasi, tuli läbi meetodite uuesti arvutada.

Lõputöö raames jaotasime omavahel ära kaks ülesannet: projekti loomise vorm ja inimeste kutsumine algavale projektile. Mina sain enda ülesandeks luua vormi, mille kaudu saab arendusprojekti looja uut projekti luua. Tuli rakendada juba olemasolevaid loodud komponente ja kohandada need mulle vajalikuks arenduseks. Lisaks tuli saada kätte iga arendusüksuse andmed andmebaasist ning õigete päringute kaudu saada õiged arvud kuvama. Kui kõik andmed olid olemas, tuli need panna ilusasti visuaali ja disainida kalender kasutajasõbralikuks.

Üheks keerulisemaks osaks kujunes mulle just olemas olevate komponentide rakendamine statistika jaoks ja projekti loomiseks. Tuli luua uus komponent ja kasutada olemasolevaid, mida kasutati ainult projekti loomise jaoks ning üles ehitada sinna kogu loogikat projekti lisamiseks. Lisaks, kuna paljud funktsioonid vajasisid loomist, siis tuli need lisada juba olemasolevale koodile ning arvestama, et iga tehtud muudatus ei lõhuks töötavat rakendust.

Pipedrive'is oldud aeg õpetas ja arendas mind märkimisväärselt. Omades null kogemust ühes ettevõttes programmeerijana töötamisest, sain näha, kuidas toimub ühe toote või funktsiooni loomine päriselt. Mul on hea meel, et sain osaleda töötoas, mis oli korraldatud minu kaastudengite poolt, ning rääkida kaasa kuidas mina lahendaksin ettevõttes esinevat probleemi. Veel enam sain hakata loodud lahendust teostama.

Omades null kogemust esiliidese arendamisel, sain ma ka selles valdkonnas palju areneda. Peale kaheksat kuud ettevõttes oldud aja, olen saanud head teadmised ja oskused JavaScript programmeerimiskeelest ning Vue.js raamistikust.

Tagasivaadates oldud ajale kahetsen, et pole enda initsiatiivist võtnud koolis juurde just eesrakenduse arendamisele keskenduvaid aineid. Töötades ettevõttes, tundsin, et selles osas jäi natuke teadmisi väheks, kuid õnneks headele mentoritele ja nende juhendamisele polnud see väga suureks probleemiks.

Lisa 2 – Enda panuse kirjeldus ja eneseanalüüs, Reia Rõõmus

Meie alustasime ettevõttes Pipedrive arendusega 2021.aasta sügisel ITB1706 infosüsteemide arendamise meeskonnaprojekt: tellimus (2021/2022 sügis) aine raames. Meie ülesandeks oli arendada Pipedrive sisest veebirakendust Missions Tool. Eesmärk oli lisada rakendusse võimalus planeerida ressursse. Ülesande keerukuse tõttu ei jõudnud me sellega ITB1706 aine raames valmis ning otsustasime jätkata projektiga lõputööna.

Lõputöö raames pidime valmis saama statistika kuvamise, mis oli meeskonnaprojekti käigus määratud eesmärk. Seda ülesannet otsustasime lahendada kõik koos. Ülesande sisuks oli eelnevalt valmis saanud statistika kalendrisse saada kuvama arendusüksuse käimasolevate projektide kohta info ning luua nimekiri vajalikest statistilistest andmetest ja kuvada nende väärtused. Lahenduseks lõime uued vajalikud failid ning meetodid, mille kaudu saadi nõutavad väärtused nädalate kaupa kalendrisse kuvama.

Kuna statistika kuvamine võttis oodatust rohkem aega, otsustasime edaspidi ülesanded ära jagada. Minu ülesandeks oli teha vorm algavatele projektidele inimeste kutsumiseks. Selleks pidin leidma uusi kui ka kohandama vanu komponente ning saama andmebaasidest kätte õiged andmed. Lõpuks tuli muuta disaini, et vorm oleks ilusam ning kasutajale arusaadav.

Kõigepealt oli vaja leida uus nupu komponent, mis ühilduks olemasolevatega ning mis oleks korrektselt kuvatud. Edaspidi oli vaja saada kuvama igapäevaste tööülesannetega tegelevate arendajate nimekiri. Pealt näha lihtne ülesanne, sest sarnane nimekiri on juba „*People*” vaates kuvatud, osutus hoopis kõige aega nõudvamaks. Selleks, et saaks ka inimeste projektile kutsumise vaates inimesi kuvada oli vaja lisada koodi juba olemasolevale koodile. Kuna projekt on suur ja sisaldab palju erinevaid seoseid, siis võib iga uus koodijupp midagi katki teha ja nii juhtus ka minul. Lõpuks osutus, et mitme erineva keeruka meetodi asemel piisas kõigest ühest.

Pipedrive'is oldud aeg õpetas mulle IT ettevõtte töökultuuri. Kuna mul on olnud varasemalt kogemus väiksemas IT ettevõttes, siis oli huvitav näha kuidas erineb töö väiksemas ja suuremas ettevõttes. Mul oli eelnevast praktikast kogemus React.js'iga seega Vue ja Javascript olid uued asjad minu jaoks. Tunnen, et olen nüüd saanud väga head teadmised eesrakenduse arenduses.

Koolis õpitu peale mõeldes, tunnen, et eesrakenduse arendamise teadmistest jäi seal natukene puudu. Õnneks sain vajalikud teadmised headelt mentoritelt praktika ja lõputöö tegemise raames.

Lisa 3 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Meie, Mariliis Sinivee ja Reia Rõõmus

1. Anname Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) meie loodud teose “Ettevõttesisese tarkvara kasutajaliidese täiendamine”, mille juhendaja on Liisa Jõgiste
 - 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.

16.05.2022