

Tartu Ülikool

Loodus- ja täppisteaduste valdkond

Tehnoloogiainstituut

Rauno Põlluäär

**Veebirakendus-põhine kasutajaliides avatud
robotplatvormi Robotont juhtimiseks ja haldamiseks**

Bakalaureusetöö (12 EAP)

Arvutitehnika eriala

Juhendajad:

Robotika kaasprofessor Karl Kruusamäe

PhD Veiko Vunder

Tartu 2021

Resümee

Veebirakendus-põhine kasutajaliides avatud robotplatvormi Robotont juhtimiseks ja haldamiseks

Robotid opereerivad mitmetes erinevates valdkondades, aidates suurendada töö efektiivsust ja vabastades inimesed ohtlikest ja kurnavatest ülesannetest. Selleks, et inimesed saaksid roboteid mugavalt kasutada, on vaja kasutajaliidest. Tartu Ülikoolis arendatud robotplatvormil Robotont puudub lõppkasutajale suunatud kasutajaliides. Käesoleva bakalaureusetöö raames valmis veebirakendus-põhine kasutajaliides, mille loomiseks kasutati Robot Web Tools tööriistu, Node.js JavaScripti käituskeskonda ja Vue.js tarkvararaamistikku. Valminud kasutajaliidese kaudu saab Robotonti juhtida ja hallata.

CERCS: T125 Automatiseerimine, robotika, juhtimistehnika, P170 Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine, P175 Informaatika, süsteemiteooria

Märksõnad: veebirakendus, kasutajaliides, Robot Web Tools, ROS, Robotont, inimese ja roboti interaktsioon

Abstract

Web application-based user interface for controlling and managing open-source robotics platform Robotont

Robots operate in many different areas, helping to increase efficiency and free people from dangerous and exhausting tasks. For humans to use robots comfortably, a user interface is needed. The robot platform Robotont that has been developed at the University of Tartu does not have a user interface for the end user. As a result of this bachelor's thesis a web application based user interface was created for Robotont. The web application was created using Robot Web Tools, Node.js JavaScript runtime environment, and Vue.js JavaScript framework. Via the interface the user can control and manage Robotont.

CERCS: T125 Automation, robotics, control engineering, P170 Computer science, numerical analysis, systems, control, P175 Informatics, systems theory

Keywords: web application, user interface, Robot Web Tools, ROS, Robotont, human-robot interaction

Sisukord

Resümee	2
Abstract	2
1 Sissejuhatus	5
2 Kirjanduse ülevaade	6
2.1 Veebirakendus	6
2.2 Veebirakendused robotitel	7
2.2.1 Robotgiid Lindsey	7
2.2.2. Tööstuslik transpordirobot MiR	9
2.2.3. Robotmanipulaator xArm	11
2.3 Veebirakenduse loomise tehnoloogiad	12
2.3.1 HTML	12
2.3.2 CSS	13
2.3.3 JavaScript	13
2.3.4 Node.js	13
2.3.5 Vue.js	13
2.4 Robotont	14
2.5 ROS (Robot Operating System)	14
2.5.1 Robot Web Tools	15
3 Robotondi veebirakendus	17
3.1 Veebirakenduse nõuded	17
3.1.1 Funktsionaalsed nõuded	17
3.1.2 Mittefunktsionaalsed nõuded	18
3.2 Veebirakenduse lahendus	18
3.2.1 Ülevaade ja arhitektuur	18
3.2.2 Kasutajaliidese ülevaade	19
3.2.3 Kasutaja vaade	21
3.2.4 Administratiivne vaade	22
3.2.5 Eessüsteem	23
3.2.6 Tagasüsteem	24
4 Arutelu	26
5 Kokkuvõte	27
Viited	28
Lihtlitsents	31

1 Sissejuhatus

Tänapäeva digitaalses maailmas on oluline osa internetil ja veebil. 2021. aasta jaanuari seisuga on aktiivseid interneti kasutajaid 4,7 miljardit, mis on 59,5% kogu maailma populatsioonist [1]. Internetis on suur osa veebibrauseri kaudu kasutatavatel programmidel ehk veebirakendustel [2]. Veebirakenduste abil on võimalik kasutada üle interneti kättesaadavaid veebiteenuseid, näiteks saata ja lugeda e-kirju, sooritada makseid internetipangas, suhelda sotsiaalmeedias ja ostelda veebipoes.

Robotid opereerivad mitmetes erinevates valdkondades, aidates suurendada töö efektiivsust ja vabastades inimesed ohtlikest ja kurnavatest ülesannetest [3]. Näiteks kasutatakse roboteid tööstuses, logistikas, põllumajanduses ja meditsiinis [4]. Kuna robotikal põhinevaid tehnoloogilisi lahendusi leidub üha rohkem, on nende mugavaks kasutamiseks vaja inimese ja roboti vahelist kasutajaliidest. Üheks mooduseks on luua robotile veebirakendus-põhine kasutajaliides.

Robotite põimimine veebirakendustega jaguneb peamiselt kahte kategooriasse [5]:

- 1) robotid, mis pakuvad veebiteenust ehk robotit on veebi kaudu võimalik juhtida või anda ülesandeid ja
- 2) robotid, mis tarvitavad veebiteenuseid ehk robotid koguvad internetist informatsiooni ja sooritavad ülesandeid, mida inimesed ei pea enam tegema.

Enamus roboteid langevad esimesse kategooriasse ja kasutavad veebi kasutajaliideste jaoks [5].

Tartu Ülikoolis arendatav avatud robotplatvorm Robotont on mobiilne robot, mida kasutatakse robotika-alaseks õppe- ja teadustööks [6]. Robotondil puudub lõppkasutajale mugav kasutajaliides, mille abil robotit hõlpsalt juhtida või muuta peamisi sätteid. Käesoleva bakalaureusetöö eesmärgiks on luua Robotondile taoline veebirakendus-põhine kasutajaliides. Bakalaureusetöös antakse ülevaade veebirakendustest, nende loomise tehnoloogiast ning kirjeldatakse Robotondile valminud veebirakenduse lahendust.

2 Kirjanduse ülevaade

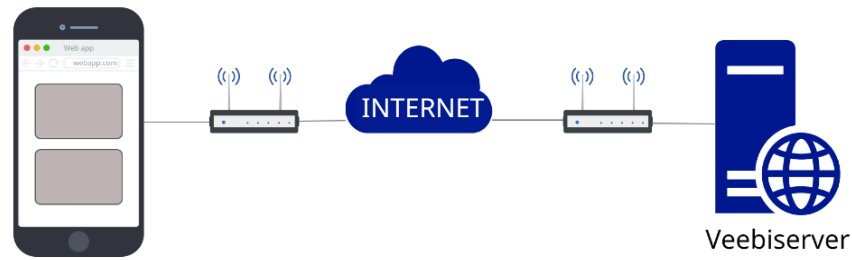
Selles peatükis selgitatakse, mis on veebirakendused, tutvustatakse lähemalt nende loomise tehnoloogiaid ja tööloogikat. Lisaks antakse ülevaade erinevate robotite veebipõhistest kasutajaliidestest ja kirjeldatakse nende funktsionaalsusi. Lõpuks tutvutakse lähemalt robotplatvormiga Robotont ja tarkvararaamistikuga ROS (Robot Operating System) ning kuidas seda liidestada veebirakendustega.

2.1 Veebirakendus

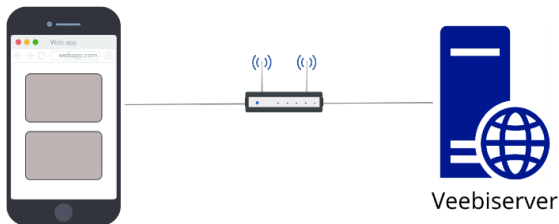
Veebirakendus on tarkvara, mida hoitakse veebiserveris, millele ligi pääsemiseks peab kasutama veebibrauserit [7]. Veebirakenduse saab jagada kahte ossa: eessüsteem (ingl *frontend*) ja tagasüsteem (ingl *backend*) [8]. Eessüsteem tegeleb kliendipoolse süsteemiga, mille peamine eesmärk on kasutajaliidese ning informatsiooni kuvamine brauseris [8]. Tagasüsteem tegeleb serveripoolse süsteemiga, kus tegeletakse rakenduse loogika ja kliendi päringutele vastamisega [8].

Võrreldes tavalise arvutiprogrammiga ei pea veebirakendust eraldi alla laadima ning paigaldama. Lisaks on veebirakendust võimalik kasutada erinevatel platvormidel nagu Windows, Linux, Mac, Android ja iOS eeldusel, et nad toetavad kaasaegset veebibrauserit [2]. Veebirakendus võimaldab luua hajussüsteemi, kus tööloogika on ühes seadmes ning kasutajaliides on teises seadmes. Rakenduse kasutamiseks on vaja luua võrguühendus veebibrauseri ja serveri vahel, kus rakendust hoitakse [2].

Veebirakendust on võimalik kasutada nii internetiühendusega kui ka lokaalselt ehk ilma internetiühenduseta. Selleks, et rakendust kasutada üle internetiühenduse peavad kasutaja seade ja veebiserver olema ühendatud internetti (joonis 2.1a). Lokaalse rakenduse korral asuvad veebirakendust hoidev server ja klient samas kohtvõrgus (joonis 2.1b) või arvutis (joonis 2.1c).



(a)



(b)



(c)

Joonis 2.1: Kasutaja seadme ja veebiserveri ühendus internetivõrgus (a), kasutaja seadme ja veebiserveri ühendus lokaalses arvutivõrgus (b) ja kasutaja seade on ka veebiserver (c).

2.2 Veebirakendused robotitel

Mitmete robotite juhtimiseks on loodud veebipõhiseid lahendusi [9, 10, 11]. Järgnevalt antakse näitlikustav ülevaade juba eksisteerivatest rakendustest ning kirjeldatakse lühidalt nende funktsionaalsusi.

2.2.1 Robotgiid Lindsey

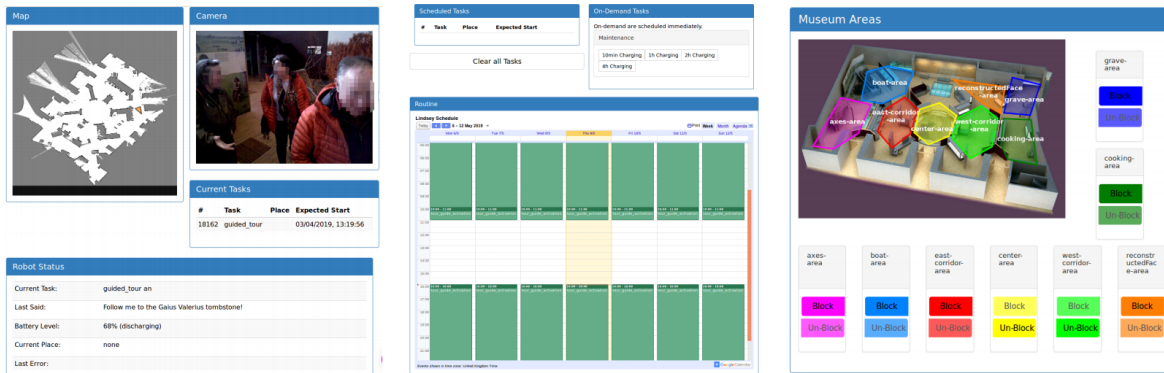
Lindsey on autonoomne robot, mis võeti kasutusele 2018. aasta oktoobris Suurbritannias Lincolni muuseumis robotgiidina [9]. Roboti eesmärk on pakkuda muuseumi külastajatele tuure ning anda informatsiooni muuseumi väljapanekute kohta (joonis 2.2).



Joonis 2.2: Kooliõpilased kasutamas giid-robot Lindseyt [9].

Lindseyle on loodud kaks veebirakendust, millest üks on loodud roboti haldamiseks (joonis 2.3) ning teine külastajatele kasutamiseks (joonis 2.4). Haldamiseks ettenähtud rakendus on jagatud kolme vaate vahel [9].

- 1) Esimese vaate kaudu saab näha roboti asukohta kaardilt, kaamerapilti ja roboti olekut (joonis 2.3a).
- 2) Teise vaate kaudu saab näha, milliseid ülesandeid ta täidab ja mis ajal neid täidetakse (joonis 2.3b).
- 3) Kolmanda vaate kaudu on võimalik määrata muuseumi alad, kuhu robotil on lubatud või keelatud minna (joonis 2.3c).



(a)

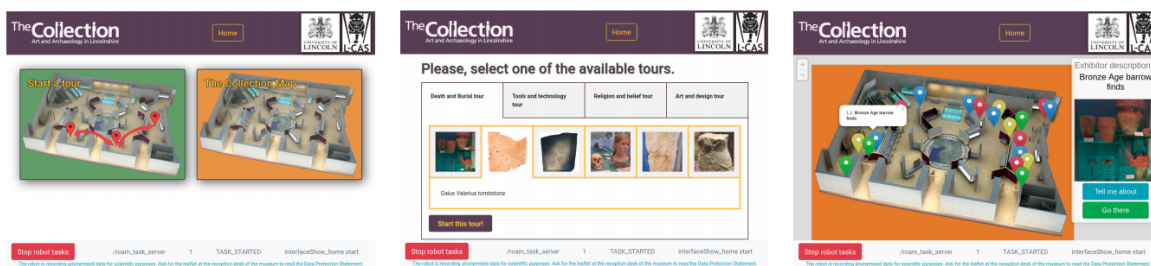
(b)

(c)

Joonis 2.3: *Giid-robot Lindsey haldamise veebiliidese vaated [9].*

Külastajad saavad neile suunatud kasutajaliidest kasutada läbi robotil oleva puutetundliku ekraani. Ka see veebirakendus on jagatud kolme erineva vaate vahel [9].

- 1) Esimene vaade on kasutajaliidese koduleht, mille kaudu saab valida, kas soovid alustada muuseumi tuuri või leida mõne eksponaadi asukoht (joonis 2.4a).
- 2) Teise vaate kaudu saab külastaja valida kõikide saadavate muuseumi tuuride vahel (joonis 2.4b).
- 3) Kolmanda vaate kaudu saab külastaja leida kindla eksponaadi asukoha muuseumis ning soovi korral ka lühikese tutvustuse selle kohta (joonis 2.4c).



(a)

(b)

(c)

Joonis 2.4: *Muuseumi külastajatele suunatud veebiliidese vaated [9].*

2.2.2. Tööstuslik transpordirobot MiR

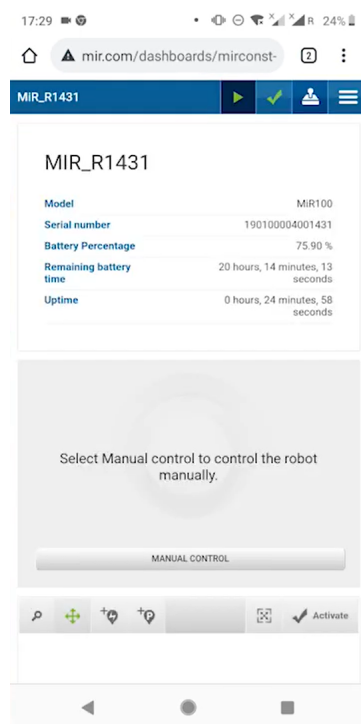
MiR on autonoomsete ja mobiilsete robotite seeria, mida kasutatakse siseruumides transpordiks ja logistikaks (joonis 2.5). Roboti eesmärk on pakkuda turvalist ja

kuluefektiivset lahendust, et tõsta produktiivsust veonduses ja vabastada töötajad asjade transportimisest [10].

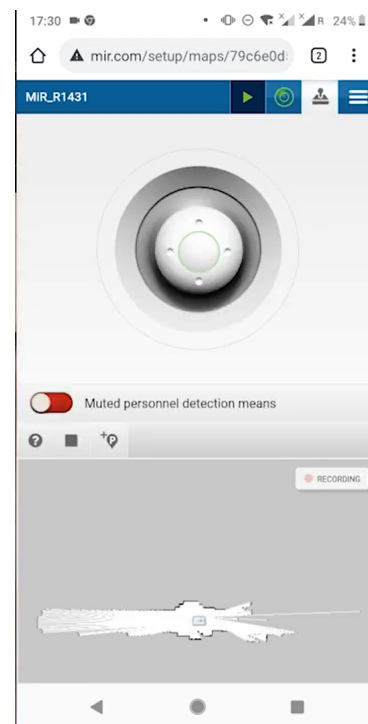


Joonis 2.5: Tööstuslik transpordirobot MiR100 [12].

MiR robotitele on loodud ka veebipõhine kasutajaliides [10]. Veebiliidese kaudu on võimalik vaadata roboti olekut (joonis 2.6a), selle asukohta kaardil, määrata koht, kuhu robot sõitma peab, luua uus kaart ruumist ja vajadusel robotit ka manuaalselt juhtida (joonis 2.6b) [12].



(a)



(b)

Joonis 2.6: Robot MiR100 veebiliidese vaated oleku kuvamiseks (a) ja kaardistamiseks ning juhtimiseks (b).

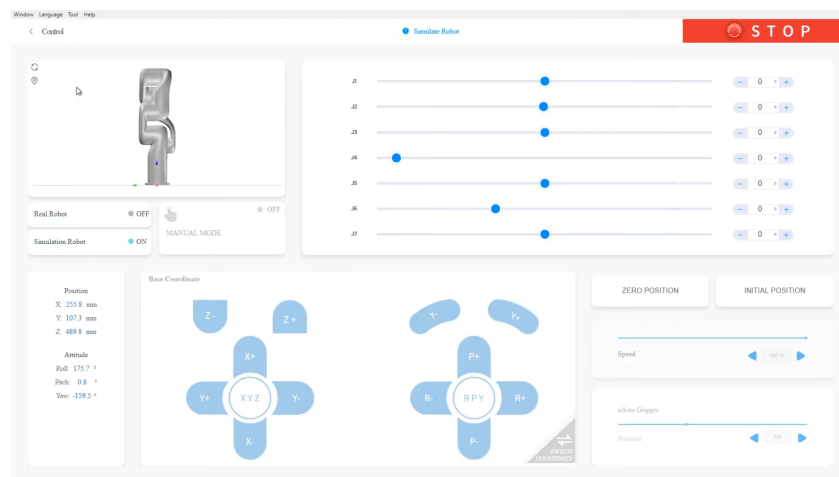
2.2.3. Robotmanipulaator xArm

xArm on robotmanipulaatorite seeria, mille eesmärgiks on pakkuda kergekaalulist ja portatiivset robotit, millega on võimalik automatiseerida erinevaid ülesandeid [11].

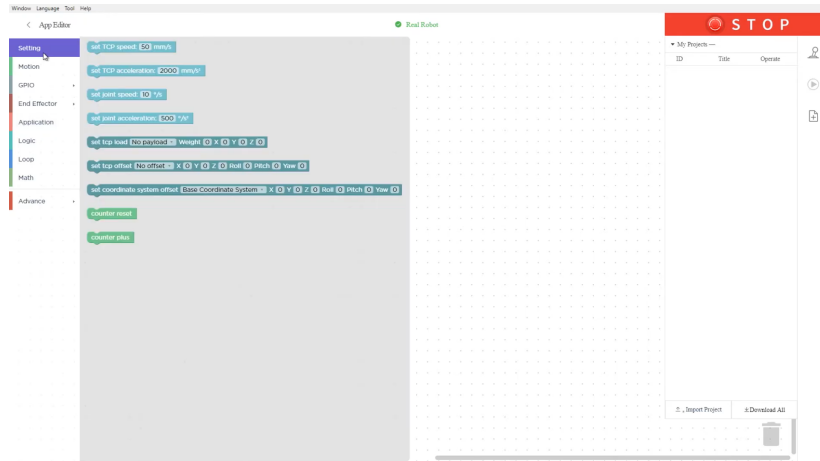


Joonis 2.7: Robotmanipulaator xArm7 [13].

Selleks, et xArm roboteid lihtsasti kasutada, on loodud veebipõhine kasutajaliides xArm Studio, mis võimaldab robotit juhtida (joonis 2.8a), seadistada ning vajadusel ka programmeerida (joonis 2.8b) [11, 13].



(a)



(b)

Joonis 2.8: *xArm kasutajaliides roboti juhtimiseks (a) ning programmeerimiseks (b).*

2.3 Veebirakenduse loomise tehnoloogiad

Veebirakenduse loomiseks on erinevaid võimalusi ja vahendeid. Üldiselt jaguneb veebirakenduse arendamine kaheks: eessüsteemi ja tagasüsteemi loomine. Eessüsteemi arendus tegeleb rakenduse kliendipoolse osaga ehk luuakse ja disainitakse kasutajaliides. Peamised vahendid, mida kasutatakse eessüsteemi arenduses on HTML, CSS ja JavaScript [8]. Tagasüsteemi arendus tegeleb serveripoolse osaga ehk luuakse veebirakenduse töö loogika ja funktsionaalsus. Tuntumad vahendid, tagasüsteemide loomiseks on Java, PHP, Python ja Node.js [14].

2.3.1 HTML

HTML ehk *HyperText Markup Language* on märgendkeel, millega ehitatakse üles veebilehe peamine struktuur. HTML fail koosneb elementidest, mida tähistatakse märgendite abil, mis ütleb veebibrauserile, kuidas märgendite vahel olevat sisu kuvada. Sellega saab lisada veebilehele teksti, pilte, videoid ning muid lihtsamaid funktsionaalsusi. Lisaks kaasatakse HTMLiga CSSi ja JavaScripti skripte, millega kujundatakse veebilehe väljanägemist ning muudetakse see interaktiivseks [15].

2.3.2 CSS

CSS ehk *Cascading Style Sheets* on veebilehtede kujundamiseks ja küljendamiseks kasutatav keel. Peamiselt kasutatakse seda HTML lehtede kujundamiseks. Näiteks saab CSSiga muuta veebilehel kasutatavaid värve, tekstistiile ja lehe üldist paigutust. Tänu CSS-ile on võimalik muuta veebilehed ka reageerivamaks ehk brauseris kuvatav sisu kohandub vastavalt veebibrauseri akna suurusele [16]. Veebilehti võib ka teha kasutades ainult HTMLi, aga sellisel juhul ei näe need eriti atraktiivsed välja.

2.3.3 JavaScript

JavaScript on objektorienteeritud programmeerimiskeel, mida käivitatakse veebibrauseris ning kasutatakse enamasti veebiarenduses. Tänu JavaScriptile on võimalik muuta veebilehed interaktiivseks. Selle abil on võimalik luua 2D ja 3D animatsioone, graafikuid ning täita muid keerulisi funktsioone. JavaScriptile on loodud ka palju teeke, mis teevad veebirakenduste loomise lihtsamaks [17].

Näiteks võimaldab Nipple.js JavaScripti teek luua virtuaalseid juhtkange, toetades ka puuetundlike ekraane [18]. Enamasti kasutatakse juhtkange arvutimängude mängimiseks või mingite objektide juhtimiseks.

2.3.4 Node.js

Node.js on avatud lähtekoodiga JavaScripti käituskeskkond, mis kasutab Google V8 JavaScripti mootorit. Selle abil on võimalik kasutada JavaScripti väljaspool veebibrauserit ning luua nii veebiservereid kui ka veebirakendusi. Node.js kasutab sündmustepõhist arhitektuuri ja andmete asünkroonset sisestamist ja väljastamist ning seetõttu on Node.js kiire ja tõhus reaalajaliste rakenduste loomiseks [19].

2.3.5 Vue.js

Vue.js on avatud lähtekoodiga JavaScripti tarkvararaamistik, millega on lihtne luua üheleheküljelisi kasutajaliideseid. Vue üheks oluliseks osaks on *komponendid*, mille abil on võimalik veebirakendus jagada väiksemateks osadeks, muutes projekti lihtsasti jälgitavaks.

Vue komponendid teevad kasutajaliideste arendamise mugavaks, kuna kogu HTML, JavaScript ning CSS koodi saab kirjutada ühte faili aidates projekti hoida kompaktsena [20].

2.4 Robotont

Robotont on Tartu Ülikoolis arendatud avatud robotplatvorm, mis on loodud õppe- ja teadustööks [6, 21]. Tegemist on mobiilse ja omniilikuva robotiga (joonis 2.9), mis on varustatud võimeka pardaarvuti ning Intel RealSense D435i sügavuskaameraga [6]. Robotonti on võimalik juhtida mängupuldi või klaviatuuriga ning sügavuskaamera abil on robotil võimalik kaardistada ümbrust kas kahe- või kolmemõõtmelisena [6]. Robot kasutab ROSil (Robot Operating System) põhinevat tarkvara, mis on leitav Robotondi GitHub-i koodihoidlast [6].



Joonis 2.9: Robotplatvorm Robotont.

2.5 ROS (Robot Operating System)

ROS on avatud lähtekoodiga tarkvararaamistik, mis sisaldab erinevaid tööriistu ning teeke, et lihtsustada robotite tarkvara arendamist [22]. ROSi eesmärk on standardiseerida robotika tarkvara arendamine ehk uue süsteemi loomisel ei pea alustama otsast peale [23].

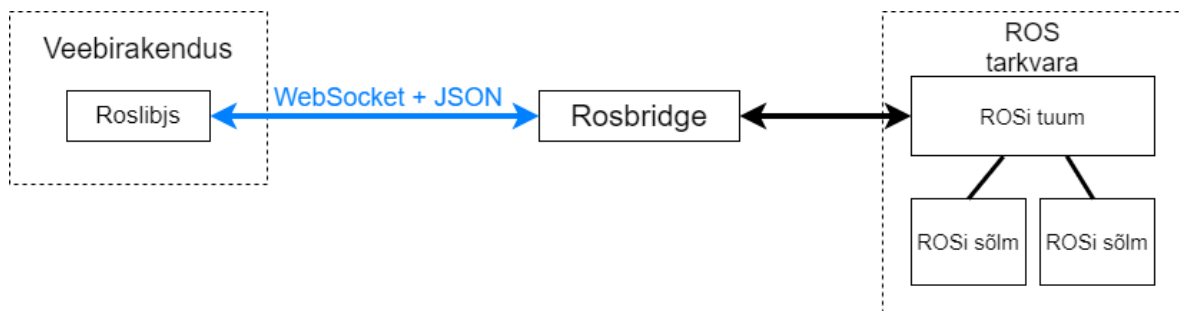
ROS põhineb protsessidel ehk sõlmedel, kus iga sõlm vastutab kindla ülesande eest. Sõlmed saavad teineteisega suhelda sõnumite kaudu, mille vahendamiseks on vaja kasutada rubriiki.

Rubriik on unikaalse nimega tarkvaraline siin sõnumite vahetamiseks. Kindla sõnumi kuulutamiseks või tellimiseks peab sõlm pöörduma vastava rubriigi poole [24].

2.5.1 Robot Web Tools

Veebirakenduste ja ROSi lihtsaks ning mugavaks liidestamiseks on loodud teekide ja tehnoloogiliste vahendite kogum nimega Robot Web Tools. Selle kogumi eesmärgiks on lihtsustada ja standardiseerida veebirakenduste loomist ROS robotitele. Vahendeid on nii veebirakenduse eessüsteemi kui ka tagasüsteemi loomiseks [25].

Veebirakenduse kaudu ei saa otse ROSiga suhelda ja selle probleemi lahenduseks on loodud tööriist Rosbridge, mis vahedab veebirakenduse ja ROSi vahelisi sõnumeid. Rosbridge on ROSi kimp, mis võimaldab JSON rakendusliidese kaudu liidestada erinevaid rakendusi või programme ROSiga [25]. Rosbridge kasutab WebSocket andmevahetusprotokolli ja tänu sellele on võimalik luua suhtluskanal ROSi ja veebirakenduse vahel (joonis 2.10). Selle ühenduse abil saab brauserist saata JSON sõnumid Rosbridge-ile, mis konverteerib saadud informatsiooni ROSi sõnumiteks ning edastab rubriikidesse. Andmevahetus saab toimida ka vastupidi [26].



Joonis 2.10: ROSi ja veebirakenduste vaheline suhtlus kasutades Rosbridge-i.

Selleks, et mugavalt luua ühendus veebirakenduse ja Rosbridge vahel, on loodud teek roslibjs. Tegemist on JavaScripti teegiga, mille eesmärgiks on lihtsustada veebirakenduste loomist, mis vajavad liidestamist ROSiga. Roslibjs kasutab WebSocket andmevahetusprotokolli, et luua ühendus Rosbridge-iga kasutades ROSiga suhtlemiseks JSON formaadis sõnumeid. Roslibjs võimaldab lihtsasti veebibrauseri kaudu kasutada ROSi

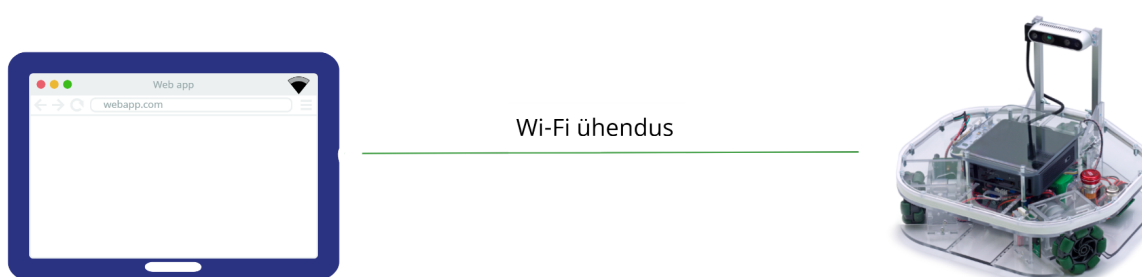
funktsionaalsusi näiteks sõnumite kuulutamine ning tellimine [27]. Selleks, et kuvada veebirakenduses 2D või 3D objekte, on loodud abistavad teegid: ros2djs ja ros3djs.

Ros3djs on JavaScripti teek, mis aitab brauseris visualiseerida ROSi kaudu saadavaid 3D objekte. Tänu ros3djs-ile on võimalik brauseris visualiseerida roboti mudeleid, sügavuspilve, punkt pilve ning muid objekte [28].

Veebirakenduses roboti poolt edastavate video ja pildi voogude kuvamiseks on loodud tööriist nimega Web video server. See on ROSi sõlm, mille abil on võimalik luua server edastamiseks ROSi rubriikide kaudu saadavaid videoid või pilte. Video ja piltide edastamiseks veebirakendusse kasutatakse HTTP andmevahetusprotokolli [29].

3 Robotondi veebirakendus

Robotplatvormi Robotonti ei saa hetkel veebi kaudu juhtida ega hallata, kuna puudub veebipõhine kasutajaliides. Kuna tegemist on mobiilse robotiga, võiks kasutajaliidese kasutus käia juhtmevabalt (joonis 3.1). Käesoleva töö eesmärgiks on luua Robotondile taoline veebirakendus. Antud peatükis antakse ülevaade veebirakendusele esitatud nõuetest, arhitektuurist ja lahendusest.



Joonis 3.1: Tahvelarvuti ja Robotondi ühendus üle Wi-Fi võrgu.

3.1 Veebirakenduse nõuded

Lähtudes Robotondi arhitektuurist, on veebirakendusele kliendi poolt esitatud nõuded järgmised:

3.1.1 Funktsionaalsed nõuded

- Rakenduse kasutajaliidesel on kaks vaadet
 - Kasutaja vaade
 - Administratiivne vaade
- Rakenduse kaudu saab robotit juhtida
- Rakenduses kuvatakse robotilt saadud kaamerapilti
- Rakenduses visualiseeritakse robotilt saadud sügavuspilv (ingl *DepthCloud*)
- Robotit on rakenduse kaudu võimalik välja lülitada
- Rakenduse kaudu on võimalik kasutada roboti parjaarvuti käsurida
- Rakenduse kaudu on võimalik Robotondi teenust käivitada, taaskäivitada või peatada

- Nii kasutaja kui ka administratiivne vaade peavad automaatselt sobituma seadme ekraani suurusele

3.1.2 Mittefunktsionaalsed nõuded

- Rakendus on ühilduv ROS distributsioonidega Melodic (Ubuntu 18.04) või Noetic (Ubuntu 20.04)

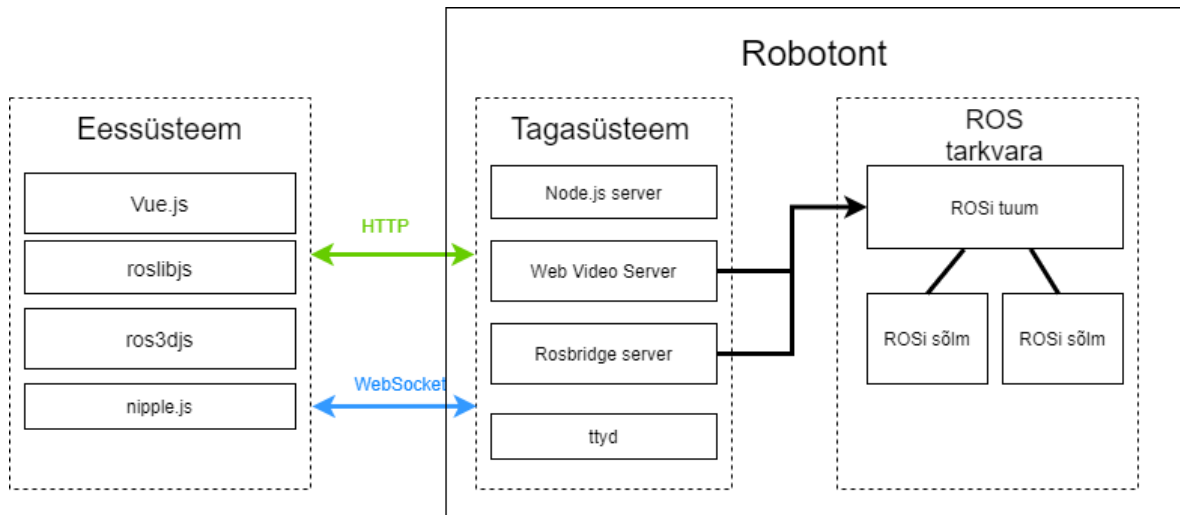
3.2 Veebirakenduse lahendus

Käesolevas alampeatükis antakse ülevaade valminud veebirakenduse arhitektuurist ja kasutajaliidesest.

3.2.1 Ülevaade ja arhitektuur

Töö raames valminud veebirakendus põhineb klient-server arhitektuuril ehk rakendust hoitakse ühes arvutis ning klient pääseb ligi, kasutades veebibrauserit oma seadmes. Antud juhul täidab serveri rolli Robotondi pardaarvuti ja kasutaja saab veebirakendust kasutada, kui ta on Robotondiga samas arvutivõrgus.

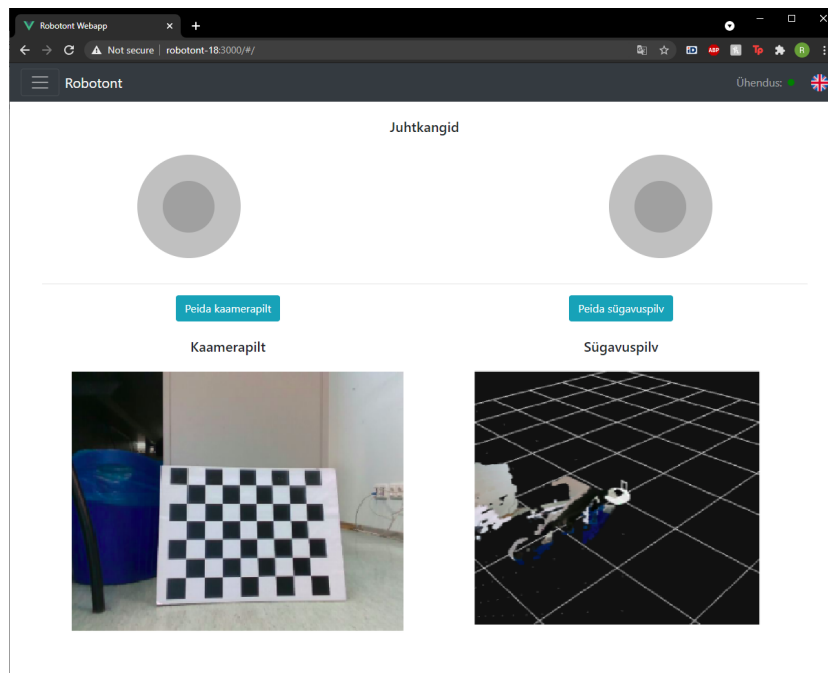
Veebirakenduse tagasüsteemi loomiseks kasutati Rosbridge'i, Node.js-i ning Web video serverit ja tööriista ttyd, millega saab arvuti käsurida kasutada veebibrauseri kaudu [30]. Eessüsteemi loomiseks kasutati Vue.js tarkvararaamistikku ning roslibjs, ros3djs ja nipple.js teeke (joonis 3.2). Käesoleva töö raames oli autoril vaba valik, milliseid tehnoloogiaid veebirakenduse loomiseks kasutada. Kuna töö valmimise ajal läbis autor Tartu Ülikooli õppeainet LTAT.05.004 Veebirakenduste loomine, siis valiti rakenduse loomiseks tehnoloogiad, mida käsitleti nimetatud aines ning mis tundusid sobivaimad antud veebirakenduse arendamiseks.



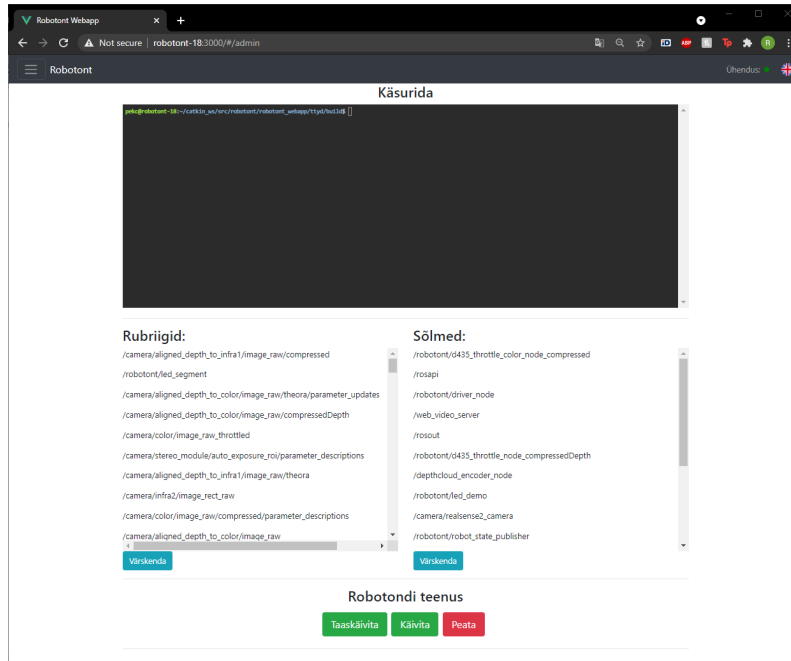
Joonis 3.2: Veebirakenduse eessüsteemi, tagasüsteemi ja Robotondi ROS tarkvara vahelised ühendused.

3.2.2 Kasutajaliidese ülevaade

Veebirakendus on jagatud kahe vaate vahel: kasutaja vaade (joonis 3.3a) ja administratiivne vaade (joonis 3.3b).



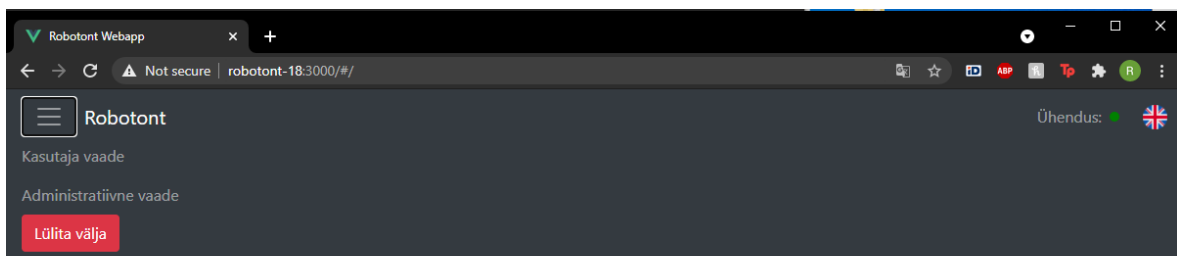
(a)



(b)

Joonis 3.3: Robotondi kasutajaliidese kasutaja vaade (a) ja administratiivne vaade (b).

Mõlema vaate ülemises osas paikneb menüüriba, mille kaudu on võimalik liikuda vaadete vahel, robot välja lülitada, jälgida Rosbridge-i ja kliendi vahelise ühenduse olekut ja vahetada kasutajaliidese keelt inglise ja eesti keele vahel (joonis 3.4).

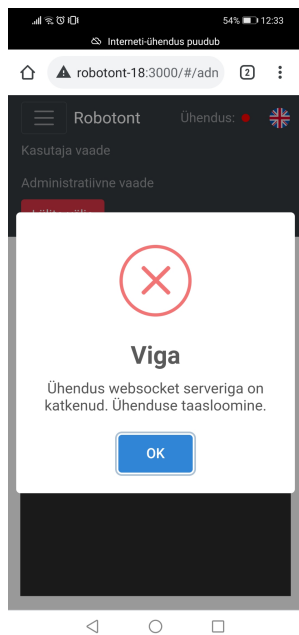


Juhtkangid

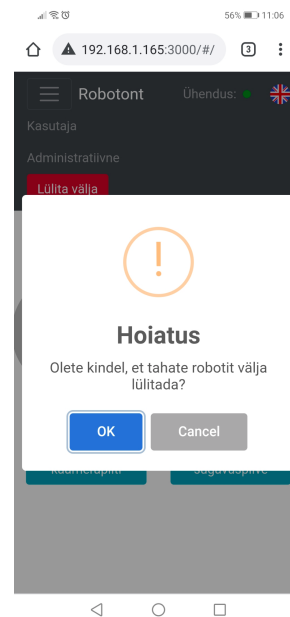
Joonis 3.4: Kasutajaliidese avatud menüüriba.

Kui ühendus peaks mingil põhjusel katkema, siis kuvatakse ekraanile kasutajat teavitav sõnum (joonis 3.5a). Kasutajat teavitatakse ka ühenduse taastumisest. Roboti väljalülitamisel

kuvatakse kasutajale sõnum, kus peab kinnitama, et tahetakse robot välja lülitada (joonis 3.5b).



(a)



(b)

Joonis 3.5: Veebirakenduse ühenduse katkemise sõnum (a) ja roboti välja lülitamise kinnitussõnum (b).

3.2.3 Kasutaja vaade

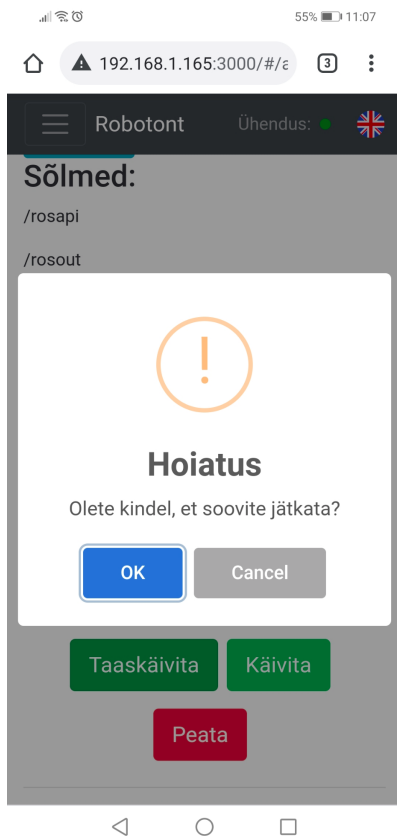
Kasutaja vaade on suunatud eelkõige tavakasutajale, kelle peamine eesmärk on robotit kaugjuhtida. Vaatesse on paigutatud roboti juhtimiseks kaks juhtkangi (joonis 3.3a). Vasakpoolse juhtkangiga saab valida, mis suunas sõita ning parempoolsega saab robotit pöörata. Nende alla on paigutatud kaks nuppu - vasakpoolse nupuga saab valida, kas näidata roboti kaamerapilti või mitte ning parempoolne nupp omab sama funktsionaalsust, kuid sügavuspilve kohta. Roboti kaameralt saadav pilt ja sügavuspilv koos Robotondi 3D mudeliga kuvatakse nuppude alla (joonis 3.3a). Vaate paigutus sõltub veebibrauseri akna suuruselt. Kui kasutada nutitelefoni, siis paigutatakse kaamerapilt ja sügavuspilv teineteise alla (joonis 3.6).



Joonis 3.6: Kasutaja vaade nutitelefone brauseris.

3.2.4 Administratiivne vaade

Administratiivne vaade on suunatud kasutajale, kellel on vaja näiteks roboti seadistusi muuta või mõnda tekkinud probleemi lahendada. Vaate esimeseks elemendiks on roboti pardaarvuti käsurida, mille alla on paigutatud ROS-ilt saadud rubriikide ja sõlmede nimekirjad ning omakorda nende alla kolm nuppu Robotondi teenuse taaskäivitamiseks, käivitamiseks või peatamiseks (joonis 3.3b). Kui neid nuppe vajutada, kuvatakse kasutajale sõnum ja küsitakse jätkamiseks luba (joonis 3.7a). Samuti võib ka selle vaate paigutus muutuda sõltuvalt sellest, kui suurel brauseri aknal veebirakendust kasutatakse (joonis 3.7b).



(a)



(b)

Joonis 3.7: Kasutajale kuvatud kinnituskast (a) ja administratiivne vaade nutitelefoni brauseris (b).

3.2.5 Eessüsteem

Veebirakenduse eessüsteemi ülesehitamiseks kasutati Vue.js tarkvararaamistikku ning roslibjs, ros3djs ja nipple.js JavaScripti teeki.

Robotondi veebirakendusele pääseb ligi, kui olla robotiga ühes arvutivõrgus ja sisestada oma seadme veebibrauserisse roboti IP aadress ja rakendusele määratud port. Kuna Robotondil töötab nimeserver, siis on võimalik IP aadress asendada roboti hostinimega, näiteks robotont-18:3000.

Selleks, et luua ühendus ROSi ja veebirakenduse vahel, kasutatakse roslibjs teeki, mis rakenduse avamisel loob automaatselt WebSocket ühenduse Rosbridge-ga. Kui ühendus on edukalt loodud, siis saab kasutada veebirakenduse funktsionaalsusi.

Roslibjs ja nipple.js teekide koostööga on võimalik Robotonti rakenduse kaudu juhtida. Nipple.js-iga on loodud kasutajaliidesele virtuaalsed juhtkangid (joonis 3.3a), mida on võimalik kasutada nii hiirega tavalisel ekraanil kui ka sõrmega puutetundlikul ekraanil. Juhtkangidelt saadud info edastatakse edasi roslibjs-i ja Rosbridge-i vahelise ühenduse kaudu robotile.

Robotondi mudeli ning kaameralt saadud sügavuspilve kuvamiseks kasutati roslibjs ja ros3djs teek (joonis 3.3a). Kuna tegemist on 3D objektidega, siis on vaja kasutada ros3djs funktsionaalsusi. Selle abil luuakse 3D kuvaja JavaScripti objekt, mille kaudu saab brauseris visualiseerida erinevaid 3D kujutisi. Sügavuspilve visualiseerimiseks kasutatakse ros3djs sügavuspilve JavaScripti objekti ning allikaks on Web video serverilt saadav sügavuspilve voog. Mudeli kuvamiseks luuakse ka vastav JavaScripti objekt ning roslibjs-i ja Node.js serveri abil tuuakse mudeli failid veebirakendusse.

Robotondi kaameralt saadud värvilist kaamerapilti visualiseeritakse, kasutades HTML-i funktsionaalsusi ja Web video serverilt saadud pildi voogu. Selleks, et pildi voogu näidata, kasutatakse HTMLi pildi kuvamise märgendeid, kus tuleb määrata pildi allikas. Allikaks on määratud URL, millega saab Web video serverilt HTTP andmevahetusprotokolli kaudu pildi.

Veebirakenduses käsurea kuvamiseks kasutatakse samuti HTMLi funktsionaalsusi. Kuna ttyd kasutab enda HTML dokumenti, et käsurida brauseris kasutada, siis selle teiste rakendustega liidestamiseks on HTMLil märgend *iframe*, mis võimaldab ühe HTML dokumendi sees kuvada teist.

Robotondi välja lülitamist ja teenuse käivitamist, taaskäivitamist ja peatamist on võimalik teha kasutajaliidesele loodud nuppude kaudu. Nupu vajutamisel tehakse HTTP päring Node.js serverile ning see täidab päringud vastavalt.

3.2.6 Tagasüsteem

Loodud veebirakenduse tagasüsteem koosneb mitmest erinevast programmist, millel on oma kindel ülesanne (joonis 3.2). Node.js-iga on loodud HTTP server pordile 3000, mille eesmärgiks on edastada kliendile eessüsteemi HTML, CSS ja JavaScripti failid, roboti mudeli failid ja käsitleda kasutajalt saadud HTTP päringuid. Hetkel saab kliendilt vastava HTTP

päringu kaudu roboti välja lülitada ja Robotondi teenust käivitada, taaskäivitada või peatada. Päringu täitmisel käivitatakse Robotondi pardaarvutil vastav käsk, näiteks roboti väljalülitamiseks antakse edasi käsk *sudo poweroff*.

Rosbridge serveri ülesandeks on olla vahendaja ROSi ja veebirakenduse vahel. Rosbridge server avab roboti pardaarvutil porti 9090 ning jääb ootama WebSocket ühendust kliendiga. Antud juhul kasutatakse Rosbridge-i, et veebirakenduse kaudu oleks robotit võimalik juhtida ja edastada informatsiooni sügavuspilve ja roboti 3D mudeli kuvamiseks. Lisaks edastatakse kliendile ka Robotondil töötavate ROSi rubriikide ja sõlmede nimekiri, et vajadusel neid mugavalt kasutajaliidese kaudu vaadata.

Web video serveriga luuakse HTTP server, mille kaudu on võimalik veebibrauserisse edastada ROSi rubriikide kaudu saadavaid pildivooge, kasutades HTTP andmevahetusprotokolli. Server avatakse portil 4000 ning seejärel jäädakse ootama kliendi päringuid. Robotondi kasutajaliidesele edastatakse kaks pildivoogu: värviline kaamerapilt ning sügavuspilv RealSense sensorilt.

Ttyd on WebSocket andmevahetusprotokollil põhinev rakendus, mille kaudu saab kasutada arvuti käsurida veebibrauserist. Tegemist on mitmeplatvormilise tööriistaga ehk on võimalik kasutada erinevate operatsioonisüsteemide käsuridasid [29]. Käsureale pääseb ligi, kui veebibrauserisse sisestada arvuti IP-aadress ning vastav port. Antud juhul käivitatakse ttyd-ga server roboti pardaarvutil portil 5000 ja on seadistatud kasutama Ubuntu käsurida.

Kogu valminud lahendus on vormistatud ROSi kimbuna ja veebirakenduse mugavaks tööle panemiseks on loodud kiirkäivitusfail. Seda faili saab käivitada, kasutades käsku *roslaunch* ja selle tulemusena alustavad tööd Node.js, Rosbridge ja Web video serverid ja tööriist ttyd.

4 Arutelu

Lõputöö käigus valminud veebirakenduse esimene versioon õnnestus, kuna kõik rakendusele esitatud nõuded täideti. Töö tulemusena valmis Robotondile veebirakendusel põhinev kasutajaliides, millega saab robotit juhtida ning vajadusel ka seadistada. Kasutajaliidest võimalik kasutada erinevatel platvormidel, mis toetavad kaasaegset veebibrauserit. Kogu lahendus on vormistatud ROSi kimbuna ning on leitav Robotondi GitHub-i koodihoidlast [31].

Veebirakenduses implementeeriti kõik nõuetes sätestatu, et Robotonti läbi veebiliidese kasutada. Kuna töö jaoks kasutati kaasaegseid veebiarenduse tehnoloogiaid ning ROSiga liidestamiseks standardseid JavaScripti teeke, on veebirakendust lihtne edasi arendada ning lisada juurde teisi funktsionaalsusi. Näiteks võiks järgmisena lisada kasutajaliidesele nupud, millega saab käivitada programme, mis demonstreerivad Robotondi võimekust. Veel võiks lisada võimaluse kasutajal vahetada kasutajaliidese värve, näiteks valida tumeda ja heleda režiimi vahel.

Veebirakendusel vajavad mõned asjad ka parandamist. Hetkel ei tööta juhtkangid nii nagu ette nähtud: roboti juhtimiseks saab kasutada vaid ühte juhtkangi korraga. Seega ei ole hetkel võimalik robotiga edasi liikudes seda ümber oma telje pöörata. Nõuetest lähtudes saab kasutajaliidese kaudu ka Robotondi teenust käivitada ja peatada. Kuna käesolevas lahenduses käivitatakse rakenduse tagasüsteem koos Robotondi teenusega, siis teenuse peatamisel lõpetab tegevuse ka rakenduse tagasüsteem ning veebirakenduse kaudu ei saa seda enam käivitada.

Enne käesoleva lõputöö valmimist polnud autor varasemalt veebiarendusega kokku puutunud. Sellest tingitult võib esineda rakenduse koodis lahendusi, mis ei ole kõige efektiivsemalt lahendatud ning vajavad paremat lähenemist. Kindlasti aitas rakenduse valmimisele kaasa õppeaine LTAT.05.004 Veebirakenduste loomine läbimine, kus tehti selgeks veebiarenduse põhitõed.

5 Kokkuvõte

Veebirakendused võimaldavad inimestel mugavalt kasutada veebiteenuseid, milleks on vaja vaid arvutit või nutiseadet. Üha rohkem kasutatakse ka robotikal põhinevaid lahendusi inimeste elu lihtsustamiseks ja abistamiseks. Selleks, et inimestel oleks robotite kasutamine lihtsam on loodud kasutajaliidesed. Üks lahendus kasutajaliidese loomiseks on kasutada veebiarenduse tehnoloogiaid ning luua selleks veebirakendus.

Käesoleva bakalaureusetöö eesmärgiks oli luua robotplatvormile Robotont veebipõhine kasutajaliides. Töö tulemusena valmis robotile veebirakendus, mille kaudu saab robotit juhtida ning seadistada.

Viited

- [1] J. Johnson, „Internet users in the world 2021 | Statista“. <https://www.statista.com/statistics/617136/digital-population-worldwide/>
- [2] „Desktop vs. Web Applications: A Deeper Look and Comparison“. <https://www.seguetech.com/desktop-vs-web-applications/> (vaadatud märts 30, 2021).
- [3] T.-M. Wang, Y. Tao, ja H. Liu, „Current researches and future development trend of intelligent robot: A review“, *Int. J. Autom. Comput.*, kd 15, nr 5, lk 525–546, 2018.
- [4] G. Mester, „Applications of mobile robots“, 2006.
- [5] M. B. Blake, S. L. Remy, Y. Wei, ja A. M. Howard, „Robots on the web“, *IEEE Robot. Autom. Mag.*, kd 18, nr 2, lk 33–43, 2011.
- [6] R. Raudmäe, „Avatud robotplatvorm Robotont“, Tartu Ülikool, Tartu, 2019.
- [7] „What is Web Application Architecture? Components, Models, and Types“. <https://hackr.io/blog/web-application-architecture-definition-models-types-and-more> (vaadatud apr 04, 2021).
- [8] „What is Web Development? - Definition from Techopedia“. <https://www.techopedia.com/definition/23889/web-development> (vaadatud märts 30, 2021).
- [9] F. Del Duchetto, P. Baxter, ja M. Hanheide, „Lindsey the tour guide robot-usage patterns in a museum long-term deployment“, *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2019, lk 1–8.
- [10] T. Visti, „Mobile Robots Increase Efficiency“, *ATZproduction Worldw.*, kd 6, nr 2, lk 48–51, 2019.
- [11] „xArm – store.ufactory.cc“. <https://www.ufactory.cc/pages/xarm> (vaadatud mai 02, 2021).
- [12] „MiR100™ | Mobile Industrial Robots“. <https://www.mobile-industrial-robots.com/en/solutions/robots/mir100/> (vaadatud apr 17, 2021).
- [13] „xArm 7 – store.ufactory.cc“. <https://www.ufactory.cc/products/xarm-7-2020> (vaadatud mai 19, 2021).
- [14] „Introduction to the server side - Learn web development | MDN“. https://developer.mozilla.org/en-US/docs/Learn/Server-side/First_steps/Introduction

(vaadatud apr 02, 2021).

- [15] D. Raggett, A. Le Hors, ja I. Jacobs, „HTML 4.01 Specification“, *W3C Recomm.*, kd 24, 1999.
- [16] J. N. Robbins, *Learning web design: A beginner's guide to HTML, CSS, JavaScript, and web graphics*. O'Reilly Media, Inc., 2012.
- [17] „What is JavaScript? - Learn web development | MDN“. https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript (vaadatud märts 29, 2021).
- [18] „yoannmoinet/nipplejs“, *GitHub*. <https://github.com/yoannmoinet/nipplejs> (vaadatud märts 31, 2021).
- [19] M. Cantelon, M. Harter, T. Holowaychuk, ja N. Rajlich, *Node.js in Action*. Manning Greenwich, 2014.
- [20] „Introduction — Vue.js“. <https://vuejs.org/v2/guide/#What-is-Vue-js> (vaadatud märts 29, 2021).
- [21] M. K. Nigol, „Õppematerjalid Robotplatvormile Robotont“, Tartu Ülikool, Tartu, 2019.
- [22] „ROS/Introduction - ROS Wiki“. <http://wiki.ros.org/ROS/Introduction> (vaadatud apr 02, 2021).
- [23] „What is ROS? - The Robotics Back-End“. <https://roboticsbackend.com/what-is-ros/> (vaadatud apr 02, 2021).
- [24] „ROS/Concepts - ROS Wiki“. <https://wiki.ros.org/ROS/Concepts> (vaadatud apr 28, 2021).
- [25] R. Toris *et al.*, „Robot web tools: Efficient messaging for cloud robotics“, *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, lk 4530–4537.
- [26] „rosbridge_server - ROS Wiki“. http://wiki.ros.org/rosbridge_server?distro=melodic (vaadatud märts 30, 2021).
- [27] „roslibjs - ROS Wiki“. <http://wiki.ros.org/roslibjs> (vaadatud märts 29, 2021).
- [28] „ros3djs - ROS Wiki“. <http://wiki.ros.org/ros3djs> (vaadatud märts 30, 2021).
- [29] „web_video_server - ROS Wiki“. http://wiki.ros.org/web_video_server (vaadatud apr 02, 2021).
- [30] „tsl0922/ttyd: Share your terminal over the web“. <https://github.com/tsl0922/ttyd>

(vaadatud apr 11, 2021).

[31] „robotont/robotont_webapp“. https://github.com/robotont/robotont_webapp (vaadatud apr 27, 2021).

Lihtlitsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Rauno Põlluäär,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose “Veebirakendus-põhine kasutajaliides avatud robotplatvormi Robotont juhtimiseks ja haldamiseks”,

mille juhendajad on Karl Kruusamäe ja Veiko Vunder,

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

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

Rauno Põlluäär

20.05.2021