

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Rain Rähni 223345IAIB

**RUUMIBRONEERIMISE TAOTLUSTE
MENETLEMINE JA RAAMATUKOGU RUUMIDE
ÜHENDAMINE ÜHTSESSE SÜSTEEMI**

Bakalaureusetöö

Juhendaja: Tarvo Treier
MSc

Tallinn 2025

Autorideklaratsioon

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

Autor: Rain Rähni

04.06.2025

Annotatsioon

Käesoleva bakalaaurusetöö eesmärgiks on võimaldada Tallinna Tehnikaülikooli raamatukogu õpiruumide broneerimine BRON [1] rakenduses ja luua funktsionaalsus taotlusprotsessi nõudvate KVO ruumide taotlemiseks. Töö käigus luuakse autoriseerimislahendus, mis võimaldab ülikoolivälistel klientidel siseneda rakendusse ja taotlussüsteemi, mis lubab taotleda broneeringut ruumidesse, mis nõuavad detailsemat menetlusprotsessi.

Autoriseerimislahendus sisaldab avaliku sisselogimislehe loomist ja erinevate kasutajate ligipääsu lubamist rakendusse. Raamatukogu kasutajaga sisenedes autentitakse ennast ESTERi [2] kasutajaga, peale mida antakse vajalikud õigused, et otsida ja broneerida kasutajarollile vastavaid ruume.

Taotlussüsteemi loomine võimaldab kasutajatel taotleda ruumidesse broneeringuid, mis loob ruumihalduri jaoks projektijuhtimise platvormi Jira [3] pileti. Ruumihaldur saab vajadusel küsida täpsustavat infot ja lisada manuseid kasutajale. Taotlus kinnitatakse ruumihalduri ja kasutaja vahelise lepingu sõlmimisega.

Lõputöö tulemusena on võimalik koondada raamatukogu ruumid BRON rakendusse ja loobuda olemasolevast raamatukogu keskkonnast. Samuti on võimalik kasutajal samas kohas ka taotleda erinevate ruumide broneeringut.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 35 leheküljel, 7 peatükki, 1 joonist, 5 tabelit.

Abstract

IMPLEMENTING ROOM BOOKING REQUEST HANDLING AND INTEGRATING LIBRARY ROOMS INTO A UNIFIED SYSTEM

The aim of this bachelor's thesis is to enable the booking of Tallinn University of Technology library study rooms in the BRON [1] application and to create functionality for requesting rooms that require an application process. The work includes creating an authorization solution that allows non-university clients to access the application and an application system that permits booking requests for rooms requiring a more detailed processing procedure.

The authorization solution includes creating a public login page and allowing different users access to the application. When logging in with a library user account, authentication occurs with ESTER [2] user, after which necessary permissions are granted to search and book rooms according to the user role.

The creation of the application system allows users to request room bookings, which creates Jira [3] ticket for the room administrator on the project management platform. The room administrator can request additional information and add attachments for the user if necessary. The application is confirmed by concluding an agreement between the room administrator and the user.

As a result of this thesis, it is possible to consolidate library rooms into the BRON application and abandon the existing library environment. It also allows users to request bookings for various rooms in the same place.

The thesis is written in Estonian and is 35 pages long, including 7 chapters, 1 figures and 5 tables.

Lühendite ja mõistete sõnastik

API	<i>Application Programming Interface</i> , rakendusliides
Autentimine	Protsess kontrollimaks teise olemi identideedi tõesust
Autoriseerimine	Protsess, mis määrab, millistel olemitel on lubatud või keelatud ligipääs teatud veebiressurssidele
BIM	<i>Building Information Modeling</i> , hoonete ja rajatiste digitaalse modelleerimise meetoodika
BRON	Tallinna Tehnikaülikooli ruumide broneerimiskonstruktsioon
CAS	<i>Central Authentication Service</i> , keskne autentimissüsteem, mis võimaldab kasutajal ühekordse sisselogimisega pääseda ligi mitmele rakendusele
DOM	<i>Document Object Model</i> , brauseri poolt hallatav puustruktuur
ESTER	Veebipõhine kataloog, mis koondab ELNET Konsortsiumiga liitunud raamatukogude, sealhulgas nende eriala- ja haruramatukogude, ning arhiivide ja muuseumide kogusid
IDP	<i>Identity provider</i> , süsteem või teenus, mis haldab kasutajate identiteete ja pakub autentimisteenuseid teistele süsteemidele
KVO	Tallinna Tehnikaülikooli Kinnisvaraosakond
Küpsis	<i>Cookie</i> , väike tekstifail, mis salvestatakse brauserisse, et identifitseerida kasutajat või koguda infot tema eelistuste kohta
Lõpp-punkt	Liides, mis asub kindlal aadressil ja võimaldab ligipääsu rakenduse teenustele
NGAC	<i>Next Generation Access Control</i> , järgmise põlvkonna ligipääsukontroll
PTW	<i>Permit to Work</i> , formaalne ja struktureeritud süsteem
RBAC	<i>Role-based access control</i> , rollipõhine ligipääsu kontroll
RPA	<i>Robotic Process Automation</i> , tarkvaraline robot, mis automatiseerib korduvaid ja reeglipõhiseid tegevusi infosüsteemides
SSO	<i>Single Sign-On</i> , autentimisviis, kus ühe sisselogimisega saab kasutada mitut teenust

Teek	<i>Library</i> , korduvkasutamiseks mõeldud kollektsioon funktsioonidest ja rutiinidest
Token	Pikk sõne, mida kasutatakse näiteks kasutaja tuvastamisel
UMA	<i>User-Managed Access</i> , kasutaja hallatav ligipääsuprotokoll
Uni-ID	<i>University identification</i> , digitaalne identiteet
URL	<i>Uniform Resource Locator</i> , internetiaadress
Webhook	Kasutaja määratletud HTTP-tagasikutse veebilehe või veebirakenduse käitumise muutmiseks
XACML	<i>Extensible Access Control Markup Language</i> , standard, mis võimaldab määratleda ja rakendada ligipääsupoliitikaid struktureeritud kujul
XML	<i>Extensible Markup Language</i> , märgistuskeel, mida kasutatakse vormingute loomiseks struktureeritud andmeid sisaldavatele tekstidele

Sisukord

1	Sissejuhatus	11
1.1	Eesmärk	12
2	Olemasolevad lahendused	14
2.1	Autoriseerimislahenduse analüüs	14
2.1.1	Lahenduse sobilikkuse analüüs	14
2.1.2	OAuth 2.0 ja RBAC	15
2.1.3	XACML	16
2.1.4	NGAC	16
2.1.5	UMA	16
2.1.6	Autoriseerimislahenduste kokkuvõte	17
2.2	Taotluse detailvaate kasutustingimuste sektsiooni lahenduste analüüs	18
2.2.1	Lahenduse sobilikkuse analüüs	18
2.2.2	Akordion märkeruuduga	19
2.2.3	Modaalaken	19
2.2.4	Eraldi lehekülg	20
2.2.5	Taotluse detailvaate kasutustingimuste sektsiooni lahenduse kokkuvõte	20
2.3	Taotluse detailvaate kommentaaride sektsiooni lahenduse analüüs	21
2.3.1	Vasak-parempoolne paigutus	21
2.3.2	Kõik kommentaarid vasakul, järjestikku üksteise all	22
2.3.3	Hierarhiline struktuur	22
2.4	Taotluse detailvaate kommentaaride sektsiooni lahenduse analüüsi kokkuvõte	23
2.5	Digitaalsete menetlussüsteemide võrdlus	23
3	Süsteemi nõuded	25
3.1	Funktsionaalsed nõuded	25
3.1.1	Raamatukogu ruumide ületoomine	25
3.1.2	Broneeringu taotluse funktsionaalsuse loomine	25
3.2	Mittefunktsionaalsed nõuded	26
4	Kasutatud tehnoloogiad	27
4.1	Rakenduse komponendid	27
4.2	Rakenduse arhitektuur	27
5	Lahenduste arendus	29

5.1	Metoodika	29
5.2	Dokumenteerimine	29
5.2.1	Kasutusvood	30
5.2.2	Järjestusdiagrammid	30
5.2.3	Andmebaasi skeemi osad	31
5.3	Ühine sisselogimisleht	31
5.4	Autoriseerimine	32
5.5	Taotlussüsteem	33
6	Tulemused	35
6.1	Valideerimine	35
6.2	Testid	37
6.3	Projektitjuhi tagasiside	38
6.4	Edasiarenduse võimalused	38
7	Kokkuvõte	40
	Kasutatud kirjandus	41
	Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks	47
	lisa 2 – Tegevusdiagrammid	48
	Lisa 3 – Järjestusdiagrammid	49
	Lisa 4 – Andmebaasi skeemid	52
	Lisa 5 – Loodud vaadete osad	53
	Lisa 6 – Testimise ülesanded	56
	Lisa 7 – Testimise küsimused	57
	Lisa 8 – Autoriseerimislahenduse testimise tulemused	60
	Lisa 9 – Taotlussüsteemi testimise tulemused	62

Jooniste loetelu

Joonis 1	<i>Rakenduse arhitektuur</i>	28
----------	--	----

Tabelite loetelu

Tabel 1	<i>Autoriseerimislahenduste võrdlus.</i>	18
Tabel 2	<i>Kasutustingimuste sektsiooni lahendused.</i>	21
Tabel 3	<i>Kommentaaride sektsiooni lahendused.</i>	23
Tabel 4	<i>Taotlussüsteemide võrdlus.</i>	24
Tabel 5	<i>Sprindid.</i>	29

1. Sissejuhatus

Tallinna Tehnikaülikooli Infotehnoloogia osakonna poolt on loodud ruumide broneerimisrakendus BRON, mis võimaldab kõigil ülikooliga seotud isikel broneerida õppetööga seotud Kinnisvaraosakonna ruume. Teiste arendajate poolt valmisid esimese versiooni raames järgnevad funktsionaalsused: ruumi otsing, otsingutulemuste kuvamine, ruumi detailvaade, vaates kuvatud ruumi info ja kalender, broneeringu tegemine ja oma broneeringute haldamine.

Praegune põhiprobleem seisneb selles, et tudengitele tuleb eraldi teavitada, et ülikoolil on kaks erinevat broneerimisrakendust. Tallinna Tehnikaülikooli Kinnisvaraosakonna (KVO) ruume saab broneerida uues rakenduses, samas kui Tallinna Tehnikaülikooli raamatukogu (edaspidi raamatukogu) õpiruumide broneerimiseks tuleb kasutada nende enda keskkonda, mille lahendus on tehniliselt aegunud. See tekitab tudengitele segadust ja ebamugavusi, kuna broneeringud asuvad erinevates süsteemides. Üliõpilastel tuleb kasutada mitut erinevat platvormi, et broneerida ülikooli õpiruume, mis muudab broneerimisprotsessi keerulisemaks ja vähem mugavaks. Raamatukogul tuleb lisaks tudengitele teenindada ka ülikooliväliseid kliente ja sellepärast ei ole raamatukogu nõus oma ruume uude rakendusse lisama, kuna ülikooli tudengitel tekib majaväliste klientide ees eelis ruumide broneerimisel.

Oodatud tulemused:

1. Ülikoolivälisel kliendil on võimalik broneerida ainult raamatukoguga seotud ruume.
2. Ülikoolisisene klient näeb raamatukogu ruume.

Samuti on probleem BRON rakenduses “unikaalsete” ruumide broneeringu taotlemise funktsionaalsuse puudumine. Unikaalsed ruumid on need, kus broneerimise õigus ei tulene automaatselt kasutaja staatusest organisatsioonis, vaid on ruumihalduri otsustada. Seetõttu nõuab nende ruumide broneerimine üksikasjalikumat menetlusprotsessi. Hetkel on rakenduses loodud õigused selliselt, et kui kasutajal on õigus ruumi näha ja selles on vaba aeg, siis saab otsekohe ruumi ära broneerida. See on tekitanud olukorra, kus osad ruumid peavad rakenduses olema kasutajate eest peidetud. Hetkel toimub “unikaalsete” ruumide broneerimine väljaspool rakendust e-posti teel ja see on tekitanud olukordi, kus mitu klienti on broneerinud sama ruumi samale ajale.

Oodatud tulemused:

1. Broneerija peab nõustuma ruumi kasutamistingimustega ja seejärel on võimalik taotleda broneeringut.
2. Taotluse peab üle vaatama üks ruumihalduritest.
3. Menetlejal on võimalik suhelda broneerijaga rakenduse siseselt.
4. Broneeringu taotlust on võimalik tagasilükata või kinnitada.
5. Kinnitamine toimub kas rakenduse väliselt sõlmides lepingu või rakenduse siseselt juriidiliselt samaväärse meetodiga.

1.1 Eesmärk

Bakalaureusetööl on kaks ärilist eesmärki: võimaldada raamatukogu ruumide üleviimine BRON rakendusse ning võimaldada ruumibroneeringute taotluste menetlemist. Raamatukogu ruumide üleviimine on jagatud väiksemateks alamülesanneteks:

1. Analüüsida erinevaid autentimis- ja autoriseerimismeetodeid ning sisselogimislehe loomise häid tavaid.
2. Leida sobilik autoriseerimismeetod TalTechi ja ESTERi *tokenite* ühildamiseks.
3. Arendada lahendus, mis võimaldab ülikoolivälistel klientidel ligipääseda BRON rakendusse.
4. Dokumenteerida tehtud töö.
5. Testida loodud lahenduse funktsionaalsust ja kasutajamugavust.

Keerukus seisneb selles, et majavälistel klientidel puudub Uni-ID, nad autentivad enast ESTER raamatukogusüsteemi kasutajate parooliga ja sellest peab piisama, et neil oleks võimalus siseneda BRON rakendusse. Raamatukogu ruumidele kehtivad täiendavad broneerimisreeglid ning raamatukogu kasutajad moodustavad eraldi kasutajagrupi, mis ei ühildu ülikooli broneerijatega.

Ruumibroneeringute taotlemise menetlusprotsessi eesmärk on jagatud väiksemateks alamülesanneteks:

1. Analüüsida lahendusi broneeringu taotluse loomiseks, broneerija ja menetleja vaheliseks suhtluseks ning juriidiliselt samaväärseid meetodeid ruumi kasutuslepingu sõlmimiseks.
2. Arendada funktsionaalsused lahenduseni jõudmiseks.
3. Dokumenteerida tehtud töö.
4. Testida funktsionaalsust ja kasutajamugavust.

Keerukus seisneb selles, et tuleb tagada ruumihalduri ja taotleja vaheline suhtlus. Lepingu

sõlmimine peaks olema võimalik rakenduse väliselt digiallkirjastades ja rakenduse siseselt mõnel teisel viisil, mis on juriidiliselt samaväärne. Mõlemale eelmainitud keerukusele lisandub lahenduse integreerimine olemasolevasse süsteemi. Lahendus eeldab mitme erineva olemasoleva süsteemiga suhtlemist ning samuti tuleb seda jätkusuutlikult dokumenteerida, et seda saaks tulevasele tootemeeskonnale üle anda.

Töö pakub märkimisväärseid eeliseid ka väljaspool ülikooli ja raamatukogu. Loodud lahendus on skaleeritav ja seda saavad kasutada ka teised asutused, kellel on probleeme mitme kasutaja ja mitmeotstarbeliste ruumide haldamisel. Näiteks konverentsikeskused, koostöötamiskeskused ja kultuuriasutused, mis vajavad kohandatud broneerimissüsteeme nii sise- kui ka välisklientidele. Kinnisvaraosakond soovib pikemas perspektiivis luua sarnast rakendust Ülemiste City Konverents [4] broneerimisrakendusele. Autentimissüsteemide, näiteks ESTERi, integreerimine ja unikaalsete ruumide taotluse funktsionaalsuse rakendamine pakub lahenduse, mis aitab ühendada erinevaid kasutajabaase.

2. Olemasolevad lahendused

Enne raamatukogu õpiruumide ületoomise lahenduse kavandamist ja arendamist on vaja teostada olemasolevate lahenduste analüüs, et määrata milline autoriseerimisviis on kõige sobilikum. Vajalik on ka sarnaste olemasolevate taotlusprotsesside võrdlus ning taotluse detailvaate lahenduse analüüs. Eesmärgiks on selgitada välja erinevate lahenduste head küljed, puudused ja sobilikkus rakendusse, mille põhjal on võimalik otsustada, millist autoriseerimismeetodit kasutada.

2.1 Autoriseerimislahenduse analüüs

Tallinna Tehnikaülikooli broneerimisrakenduses autentivad kasutajad ennast ülikooli Uni-ID kasutajakontoga. Sisselogides kasutatakse Microsoft Azure Identity Provider [5] teenust, mis genereerib OAuth 2.0 protokollil *tokeni*. See võimaldab kasutajal ligipääseda rakendusele ja ruume broneerida. ESTER raamatukogusüsteem kasutab kasutajate autentimiseks Central Authentication Service [6] lahendust.

2.1.1 Lahenduse sobilikkuse analüüs

Enne lahenduste analüüsi on vaja määrata selged kriteeriumid, mille järgi tehnoloogilisi lahendusi analüüsitakse ja hinnatakse. Väljavalitud kriteeriumid on

- Turvalisus
- Integreerimise lihtsus olemasolevasse süsteemi
- Skaleeritavus

Turvalisus on oluline, kuna autoriseerimissüsteem reguleerib juurdepääsu tundlikele andmetele ja ressurssidele. Iga turvaviga võib põhjustada konfidentsiaalsete andmete lekkimist või ressursside väärkasutust [7]. Arendatavas rakenduses töödeldakse kasutajate isikuandmeid, mistõttu peab valitud lahendus tagama andmekaitse.

Integreerimise lihtsus olemasolevasse süsteemi vähendab märkimisväärselt arendusaega ja kulusid. Sujuv integratsioon võimaldab säilitada olemasoleva süsteemiarhitektuuri põhikomponendid, minimeerides muudatuste ulatust ja keerukust. Samuti tagab see kiirema kasutuselevõtu ja lihtsustab süsteemihaldajate kohanemist uue lahendusega [8].

Skaleeritavus tagab rakenduse jätkusuutlikkuse tulevikus, kui kasutajate ja päringute hulk suurenevad. Eriti oluline on rakenduse võime tulla toime kõrgemate kasutusperioodidega (näiteks semestri alguses või eksamiperioodil). Lisaks võimaldab hästi skaleeruv lahendus tulevikus lisada teisi autoriseerimismeetodeid või kasutada sarnast autoriseerimisloogikat teistes rakendustes [8].

Järgnevalt hinnatakse eeltoodud kriteeriumide põhjal erinevaid autoriseerimislahendusi, et leida ESTERi kasutajatele sobivaim ligipääsumudel BRON rakenduses.

2.1.2 OAuth 2.0 ja RBAC

OAuth 2.0 on autoriseerimisprotokoll, mida kasutatakse kasutajate autentimiseks ja ligipääsuõiguste andmiseks rakendustele või teenustele. See võimaldab jagada juurdepääsuõigusi ilma autentimistegureid (näiteks parooli) jagamata. Teenused või rakendused saavad väljastada ajutisi turvalisi *tokeneid*, mis määravad, millistele ressurssidele ja kui kaua kasutaja võib ligipääseda [9].

RBAC on meetod, mis reguleerib ligipääsuõigusi süsteemis tuginedes kasutajatele määratud rollidele. Iga roll on seotud kindlate õiguste ja kohustustega, mis määravad, milliseid ressursse või funktsioone kasutaja saab kasutada. See lähenemine võimaldab lihtsamat ja turvalisemat õiguste haldamist, eriti suuremates süsteemides [10].

OAuth 2.0 ja RBAC kombineerimine võimaldab luua terviklikuma ligipääsuõigussüsteemi. Praktikas defineeritakse erinevad rollid ja määratakse neile konkreetsed ligipääsuõigused. Seejärel seostatakse kasutajarollid OAuth 2.0 skoopidega, mis on määratud *tokenis*.

- + Laialdaselt aktsepteeritud standard ja ühildub hästi Microsoft Azure infrastruktuuriga [5].
- + *Tokenid* võimaldavad ressursile turvalist juurdepääsu, ilma kasutaja mandaate edastamata [9].
- + Kasutajarollid võimaldavad lihtsalt hallata suuri kasutajabaase ja õiguste struktuure [10].
- + OAuth 2.0 tõendite arhitektuur on staatiline ehk iga päringu puhul ei pea kasutajat autentima [9].
- Vajab hoolikat rollidesüsteemi disaini ja tekib ligipääsutõendihalduse lisakulu [9].

2.1.3 XACML

XACML on OASIS-e (Organization for the Advancement of Structured Information Standards) [11] poolt välja töötatud standard, mis võimaldab ligipääsuõiguste poliitikate defineerimist, haldamist ja jõustamist struktureeritud XML-põhises formaadis. See võimaldab atribuutidel põhinevat juurdepääsukontrolli, mis võimaldab organisatsioonidel määratleda üksikasjalikke ja kontekstitundlikke juurdepääsureegleid [12].

- + Võimaldab väga täpset juurdepääsukontrolli ja määratleb reegleid, millal ja kuidas ressursse saab kasutada [12].
- + Kõik juurdepääsukontrolli reeglid on ühes kohas, mis teeb süsteemi paremini haldatavamaks [12].
- XACML-i rakendamine on keerukas ja aeganõudev, kuna tuleb määrata juurdepääsukontrolli reeglid ja integreerida need olemasolevasse süsteemi [12].
- Reeglite pidev hindamine võib põhjustada täiendavat jõudluse koormust süsteemis [12].
- XACML on keeruline, mis võib olla raske teistele arendajatele, kuna nõuab põhjalikku mõistmist selle rakendamisest ja haldamisest [12].

2.1.4 NGAC

NGAC on NIST-i (National Institute of Standards and Technology) [13] poolt välja töötatud autoriseerimisraamistik, mis pakub poliitikatel põhinevat juurdepääsukontrolli, kasutades graafipõhist lähenemist autoriseerimissuhete esitamiseks ja jõustamiseks [14].

- + Pakub granulaarsemat ja kontekstipõhisemat juurdepääsukontrolli [14].
- + Saab kohandada keerukate juurdepääsunõuetega ilma rollide üleküllastumiseta [14].
- Keerukam disainida ja rakendada kui RBAC [14].
- Raskem auditeerida ja visualiseerida juurdepääsuõigusi võrreldes rollipõhiste mudelitega [14].
- Võib autoriseerimisotsuste ajal põhjustada täiendavat töötlemiskoormust [14].

2.1.5 UMA

UMA on Kantara Initiative'i [15] poolt standardiseeritud OAuth 2.0 protokollil põhinev autoriseerimisprotokoll, mis võimaldab kasutajatel hallata juurdepääsu oma isiklike andmetele ja ressursidele üle mitmete teenuste ja süsteemide. UMA keskendub sellele, et kasutajad saavad ise määrata, kes saab nende andmetele juurdepääseda ja millistel

tingimustel, pakkudes suuremat kontrolli kasutajale [16].

- + Võimaldab kasutada olemasolevat OAuth 2.0 protokollit ja infrastruktuuri [16].
- + Kasutajad saavad hallata isiklike ressursside ja andmete ligipääsu [16].
- OAuth 2.0-le täiendav kihiline ülesehitus, mis nõuab eraldi autoriseerimisserverit ja ressursside omanikule mõeldud portaali [16].
- Piiratud kasutusala, kuna ei ole toetatud paljude teenuste ja teenusepakkujate poolt [16].
- Täiendavate turvakihtide arendamise ja haldamise rakendamine on kulukas ja ajamahukas [16].

2.1.6 Autoriseerimislahenduste kokkuvõte

Eelnevalt analüüsitud autoriseerimislahendused on tõhusad meetodid, mis sobivad erinevate kasutusjuhtumite jaoks. Iga lahendus võimaldab tulemuslikku juurdepääsu haldamist ja turvalist autoriseerimist, kuid nende sobivus sõltub konkreetse süsteemi vajadustest ja arhitektuurist. Kuna lõputöö raames arendatav süsteem keskendub raamatukogu ruumide ületoomisele ja kasutajate ligipääsu reguleerimisele, on oluline valida lahendus, mis vastab optimaalselt süsteemi kasutusjuhtudele ja pikaajalistele eesmärkidele.

Arendatav lahendus peab olema võimeline kasvama koos kasutajate ja andmete hulga. Lahendus peab toetama süsteemi laienemisel keerukamaid juurdepääsureegleid ning võimaldama hallata mitut erinevat kasutajarolli. Kuna süsteem kasutab SSO-lahendust, tuleb valida autoriseerimismeetod, mis integreerub sujuvalt olemasolevate autentimisteenustega ja võimaldab valideerida kasutajaid väliste süsteemide kaudu. Süsteemi üheks olulisemaks nõudeks on kasutajate õiguste täpne määramine vastavalt nende rollile. Lahendus peab toetama RBAC-i, et määratleda, milliseid ruume iga kasutaja näeb ja saab broneerida, ning tagama, et õiguste määramine oleks lihtne ja hallatav ka suurema koormuse korral. Kuna süsteem peab töötleva tundlikku infot, on oluline, et valitud lahendus tagab kõrge turvalisuse.

Pärast põhjalikku analüüsi on selge, et OAuth 2.0 koos RBAC-iga vastab kõige paremini arendatava rakenduse vajadustele. See lahendus on lihtne, skaleeritav ja hästi integreeritav olemasoleva rakendusega. Kuna süsteemi autoriseerimisreeglid on sirgjoonelised ja keskenduvad rollipõhisele ligipääsule, on OAuth 2.0 koos RBAC-iga piisavalt paindlik ja kasutajasõbralik lahendus. Lisaks võimaldab see tulevikus vajadusel süsteemi laiendada. Kuigi UMA, XACML ja NGAC pakuvad suuremat paindlikkust ja keerukamate juurdepääsureeglite haldamist, on nende rakendamine antud juhul ebavajalikult keerukas ning ei lisaks süsteemi kasutusjuhtudele märkimisväärset väärtust. Tabelis 1 toodud hinnangud

põhinevad eelnevalt esitatud plussidel, miinustel ja lõppanalüüsil. Kuigi need hinnangud on subjektiivsed, püüavad need peegeldada üldist muljet ja kaalutlust, mis analüüsi käigus kujunes.

Tabel 1. *Autoriseerimislahenduste võrdlus.*

Lahendus	Turvalisus	Integreerimise lihtsus	Skaleeritavus
OAuth 2.0 + RBAC	Kõrge	Kõrge	Kõrge
XACML	Väga kõrge	Madal	Keskmine
NGAC	Kõrge	Madal	Keskmine
UMA	Keskmine	Keskmine	Madal

2.2 Taotluse detailvaate kasutustingimuste sektsiooni lahenduste analüüs

Järgnevalt analüüsitakse kolme erinevat kasutustingimuste sektsiooni kuvamise lahendust. Eesmärgiks on selgitada välja, milline esitamisviis tagab optimaalse kasutajakogemuse. Kasutajaliides on kasutajasõbralik, kui see vähendab kasutaja kognitiivset koormust, edastab informatsiooni selgelt ja struktureeritult ning võimaldab sujuvat, katkestusteta tegevusvoogu ka mobiilsetel platvormidel [17].

2.2.1 Lahenduse sobilikkuse analüüs

Enne lahenduste analüüsi on vaja määrata selged kriteeriumid, mille järgi tehnoloogilisi lahendusi analüüsitakse ja hinnatakse. Kriteeriumid, mille järgi lahendusi analüüsitakse ja hinnatakse on

- Kognitiivne koormus
- Visuaalne müra
- Mobiilne sobivus

Kognitiivne koormus on oluline, kuna kõrge kognitiivne koormus aeglustab ülesannete täitmist, suurendab vigade hulka ja vähendab rahulolu. Lihtsamad ja selgemalt struktureeritud liidesed parandavad kasutajakogemust ja suurendavad tagasipöördumist [18].

Visuaalse müra vähesus on oluline, kuna selge ja korrastatud liides aitab suunata kasutaja tähelepanu olulistele elementidele, parandades kasutajaliidese tõhusust ja kasutajakoge-

must. Samuti vähem segavad liidesed suurendavad kasutajate rahulolu [19].

Mobiilne sobivus on oluline, kuna paljud inimesed kasutavad rakendust peamiselt oma mobiilseadmetes. See aitab parandada üldist kasutajakogemust, võimaldades kasutajatel kiiresti ja tõhusalt soovitud toiminguid teha [20].

2.2.2 Akordion märkeruuduga

Akordion on interaktiivne kasutajaliidese element, mis võimaldab kasutaja vajutusel dünaamiliselt kuvada või peita sisu [21]. Selle lahenduse korral on kasutustingimused vaikumisi peidetud ning muutuvad nähtavaks kui kasutaja akordionile vajutab. Elemendi all on märkeruut, mille abil kasutaja kinnitab, et on tingimustega tutvunud ja nõustub nendega.

- + Vähendab visuaalset segadust ja kognitiivset koormust, kuna tingimused on peidetud ja kuvatakse ainult vajadusel [21].
- + Mobiilis hästi kasutatav, kuna ei võta palju ekraaniruumi ja toetab interaktiivset sisu kuvamist väikestel ekraanidel [21].
- + Ei katkesta kasutaja tegevusvoogu, kuna jääb samale lehele ja võimaldab sujuvat kasutuskogemust [21].
- + Võimaldab visuaalselt ja funktsionaalselt kinnitada nõustumist märkeruudu abil [21].
 - Kui disain pole piisavalt selge, võib kasutaja sisu mitte avada või jätta märkamata [21].
 - Vajab head visuaalset eristust ja juhiseid, et kasutaja mõistaks, et sisu on peidetud ja avamiseks vajutatav [21].

2.2.3 Modaalaken

Modaalaken on interaktiivne kasutajaliidese element, mis ilmub põhisisu ette, blokeerides ajutiselt muu interaktsiooni ning nõuab kohest tähelepanu enne põhitegevusega jätkamist [22]. Selle lahenduse puhul kuvatakse kasutustingimused eraldiseisvas modaalas ja kasutaja peab modaalas sulgema või kasutustingimustega nõustuma, et jätkata ruumi broneeringu taotlemisega.

- + Tagab, et kasutaja märkab ja teadvustab kasutustingimuste olemasolu, kuna katkestab tegevusvoo ja tõmbab tähelepanu [22].
- + Sobib juhul, kui on oluline sundida tingimuste lugemist või nõusolekut enne jätkamist [22].

- Mobiilis raskesti kasutatav, kuna modaalid võivad olla ekraanil halvasti skaleeritud või sulgemine keeruline [22].
- Võib põhjustada takistusi erivajadustega kasutajatele, kuna katkestab fokuseeritud navigeerimise ja segab abivahendite kasutust [22].
- Katkestab kasutaja tegevusvoo ja võib tekitada frustratsiooni, kui see ei ole õigustatud [22].

2.2.4 Eraldi lehekülg

Kasutustingimusi esitatakse sageli eraldi lehel, millele pääseb ligi hüperlingi kaudu [23]. See lahendus suunab kasutaja ühelt leheküljelt teisele ning võimaldab keskenduda konkreetsele sisule, säilitades samal ajal võimaluse naasta varasema tegevuse juurde brauseri tagasinupu kaudu.

- + Võimaldab kuvada väga detailset ja struktureeritud infot, kuna eraldi lehel on rohkem ruumi sisu esitlemiseks [23].
- + Lihtne tehniliselt teostada ja hallata, kuna ei vaja keerukaid kasutajaliidese elemente [23].
- Kasutaja peab lahkuma taotluse detailvaatest ja naasma hiljem tagasi, mis võib katkestada voogu [23].
- Suurem risk, et kasutaja jätab protsessi pooleli ega naase tagasi varasemale tegevusele [23].
- Aeglasem ja kohmakam kasutajakogemus võrreldes sisseehitatud lahendustega [23].

2.2.5 Taotluse detailvaate kasutustingimuste sektsiooni lahenduse kokkuvõte

Eelnevalt analüüsitud kasutustingimuste esitusviisid on mitmekesised erinevate kasutusjuhutumite jaoks. Kuigi kõik kolm meetodit võimaldavad kasutajal tingimustega tutvuda ja anda oma nõusoleku, erinevad nad oluliselt kasutajakogemuse kvaliteedi, rakendamise keerukuse ja kontekstilise sobivuse poolest.

Ruumi broneeringu taotluse detailvaate jaoks on kõige sobivam lahendus akordion koos märkeruuduga, kuna see meetod vähendab kasutaja vaimset koormust ja visuaalset müra. Samuti on kasutustingimuste tekst kompaktne ja ei vaja eraldi lehekülge. Lahendus tagab hea kasutajakogemuse ka piiratud ekraanisuurusega seadmetel.

Modaalakna lahendus, kuigi tagab tingimustega tutvumise, sunnib kasutajat katkestama

oma tegevusvoogu, põhjustades mobiilseadmetes märkimisväärset ebamugavust ja potentsiaalset frustratsiooni. Eraldi lehekülje lahendus tekitab ebavajaliku navigatsioonilise katkestuse, mis ei ole õigustatud käesoleva rakenduse kasutustingimuste kompleksuse ja mahu juures. Tabelis 2 toodud hinnangud põhinevad eelnevalt esitatud plussidel, miinustel ja lõppanalüüsil. Kuigi need hinnangud on subjektiivsed, püüavad need peegeldada üldist muljet ja kaalutlust, mis analüüsi käigus kujunes.

Tabel 2. Kasutustingimuste sektsiooni lahendused.

Lahendus	Kognitiivne koormus	Visuaalne müra	Mobiilne sobivus
Akordion + märkeruut	Madal	Madal	Kõrge
Modaalaken	Keskmine	Kõrge	Madal
Eraldi lehekülg	Keskmine	Keskmine	Madal

2.3 Taotluse detailvaate kommentaaride sektsiooni lahenduse analüüs

BRON rakenduses on kriitilise tähtsusega tagada tõhus kommunikatsioon taotluse esitanud kasutaja ja menetleja vahel. Selleks tuleb implementeerida kommentaaride sektsioon, mis võimaldab ruumihalduril täpsustada broneeringuga seotud detaile ja vajadusel lisada manuseid. Järgnevalt analüüsitakse kolme erinevat kommentaaride sektsiooni lahendust. Analüüsi peamine eesmärk on määratleda kasutajasõbralikum lahendus. Lahenduste sobilikkust analüüsitakse ja hinnatakse sarnaselt kasutustingimuste sektsioonile kognitiivse koormuse, visuaalse müra ja mobiilse sobivuse põhjal.

2.3.1 Vasak-parempoolne paigutus

Vasak-parempoolne paigutus rakendab interaktiivset disainimustrit, kus taotleja kommentaarid kuvatakse vasakul ja ruumihalduri kommentaarid paremal pool kasutajaliidese vaadet. See lahendus järgib levinud sõnumirakenduste disainiparadigmat, mille kohaselt on erinevate vestlusosapoolte sõnumid visuaalselt selgelt eristatavad nii paigutuse kui ka värvikodeeringu abil [24].

- + Tuttav ja intuitiivne paigutus, kuna järgib laialt levinud sõnumirakenduste mustrit [24].
- + Tekib tunne, nagu vestlus toimiks reaalajas, mis lisab vestlusele personaalsust [24].

- Mobiilseadmetes võib see paigutus olla visuaalselt ebajärjekindel [24].
- Erinevate paigutuskohtade vaheldumine võib suurendada kasutaja vaimset koormust [24].
- Keeruline integreerida olemasolevasse rakendusse, kuna nõuab rohkem arendustööd [24].

2.3.2 Kõik kommentaarid vasakul, järjestikku üksteise all

Vasakpoolses järjestuses kuvatakse kõik kommentaarid ühtses vertikaalses jadas, kus iga sõnum järgneb eelnevale, ilma visuaalset asümmeetriat kasutamata. Sõltumata sõnumi lisajast on see alati vasakul [25].

- + Aitab kasutajal kiiresti mõista vestluse järjestust ja jälgida kogu suhtlust ilma visuaalsete katkestusteta [25].
- + Vähendab vajadust otsida visuaalseid vihjeid sõnumite eristamiseks [25].
- + Järjepidev ja intuitiivne kasutajakogemus ka mobiilseadmetes [25].
- + Lihtne integreerida olemasolevasse rakendusse, kuna ei nõua keerulist paigutusloogikat [25].
- Kasutaja vajab tekstilisi vihjeid osalejate eristamiseks, kuna visuaalne eristus puudub [25].

2.3.3 Hierarhiline struktuur

Hierarhiline kommentaaristruktuur rakendab mitmetasandilist vestluse organiseerimist, kus kasutajad saavad vastata konkreetsetele kommentaaridele, luues seeläbi pesastatud suhtlusvoo [26]. Selline disainilahendus on laialt levinud erinevates diskussioonifoorumites ja sotsiaalmeediaplattformidel, kus see võimaldab keerukamate vestlusteemade hargnemist.

- + Võimaldab selgelt visualiseerida kommentaaride omavahelisi seoseid ja kuuluvust, sobides keerukamate teemade jaoks [26].
- Ülemäärane ja segadust tekitav kahe osapoole suhtluseks, kuna toob kaasa liigselt kihistatud struktuuri [26].
- Vestluse jälgimine on raskendatud, eriti kui vastuseid on palju ja need on pesastatud [26].
- Suurendab kognitiivset koormust, kuna kasutaja peab pidevalt analüüsima kommentaaride struktuuri [26].

2.4 Taotluse detailvaate kommentaaride sektsiooni lahenduse analüüsi kokkuvõte

Eelnevalt analüüsitud kommentaaride sektsiooni lahendused pakuvad mitmekesiseid võimalusi ruumihalduri ja taotleja vaheliseks suhtluseks. Kuigi kõik kolm meetodit võimaldavad kasutajatel lisada kommentaare ja manuseid, erinevad nad oluliselt kasutajakogemuse loomulikuse, rakendamise keerukuse ja mobiilmugavuse poolest.

Ruumi broneeringu taotluse detailvaate jaoks on kõige kasutajasõbralikum meetod kommentaaride kuvamine vasakul, järjestikku üksteise all, kuna see lahendus vähendab kasutaja vaimset koormust ja võimaldab vestlust loomulikul viisil jälgida. Samuti on järjestikune kuva kompaktne ja kohaneb hästi erinevate ekraanisuurustega.

Vasak-parempoolne paigutus võib mobiilseadmetes põhjustada märkimisväärset ebamugavust ja tekitada katkestusi lugemisrütmis. Hierarhiline struktuur omakorda lisab tarbetut keerukust kahepoolse kommunikatsiooni jaoks, mis ei ole õigustatud käesoleva rakenduse vestluse iseloomu juures. Tabelis 3 toodud hinnangud põhinevad eelnevalt esitatud plussidel, miinustel ja lõppanalüüsil. Kuigi need hinnangud on subjektiivsed, püüavad need peegeldada üldist muljet ja kaalutlust, mis analüüsi käigus kujunes.

Tabel 3. *Kommentaaride sektsiooni lahendused.*

Lahendus	Kognitiivne koormus	Visuaalne müra	Mobiilne sobivus
Vasak-parem paigutus	Keskmine	Keskmine	Kõrge
Kõik vasakul järjestikku	Madal	Madal	Kõrge
Hierarhiline struktuur	Kõrge	Kõrge	Madal

2.5 Digitaalsete menetlussüsteemide võrdlus

Käesolevas alapeatükis võrreldakse nelja erinevat taotlussüsteemi: Jira-põhine menetlus, BIM-põhine ehituslubade süsteem, elektrooniline PTW süsteem ja RPA toega töövoogude haldus.

Jira-platvorm võimaldab igale taotlusele luua pileti ning liikuda etapp-haaval läbi eelnevalt määratletud töövoos, kus skriptid ja inimeste tehtud toimingud töötavad koos. Süsteemi paindlik konfigureerimine ja ühtne veebiliides toetavad kiiret kasutuselevõttu ning reaalajas staatuse- ja teavituste haldust [27].

3D-andmemudelitel põhinev lahendus võimaldab teha automaatseid vastavuskontrolle ehitusnormidele ning kuvada ülevaadet detailsetest disainidest ühes keskkonnas. Selline meetod vähendab käsitsi tehtavate jooniste ja dokumentide arvu ning kiirendab tehnilise ülevaate protsessi, kuigi lõplikud ametkondade kinnitused jäävad endiselt manuaalseks [28].

Digitaalne PTW süsteem asendab pabervormid veebirakenduse ja mobiiliportaaliga, kus töölubade taotlemine, riskihinnang ja kinnitus toimuvad auditeeritavalt ning juhendatult. Integreeritud GPS ja biomeetrilised moodulid aitavad tagada töötajate volitatud juurdepääsu ja vähendada inimlikke vigu [29].

RPA-robotid imiteerivad korduvaid reeglipõhiseid toiminguid ärirakendustes, mis võimaldab automatiseerida andmesisestust, aruandlust ja muid rutiinseid ülesandeid ilma infosüsteemide põhjaliku ümberkirjutamiseta. Selline lähenemine tõstab protsesside täpsust ja kiirust, vähendades samal ajal tööjõu koormust ja kulusid [30].

Tabel 4. *Taotlussüsteemide võrdlus.*

Süsteem	Automaatika	Kasutajakogemus	Kinnitusi	Peamine eelis
Jira	Hübriid	Veeb	2	Kiire juurutus
BIM	Täisautomaatne	BIM-tööriistad	4+	Automaatne vastavuskontroll
PTW	Kõrge	Veeb/mobiil	3-5	Tunniga lubade kinnitus
RPA	Keskmine	Robotliides	Muudetav	Rutiinide automatiseerimine

3. Süsteemi nõuded

Järgnevas peatükis loetletakse autoriseerimislahenduse ja taotlussüsteemi funktsionaalseid ning mittefunktsionaalseid nõudeid. Autor koostas ärinõuete analüüsile tuginedes autoriseerimislahenduse funktsionaalsed nõuded, mida autor programmikoodi tasemel lahendab. Samuti toob autor paralleele teaduskirjandusest ning näiteid ärinõuete ja tehniliste nõuete seostamiseks.

3.1 Funktsionaalsed nõuded

Funktsionaalsed nõuded kirjeldavad süsteemi konkreetseid funktsioone ja käitumist. Need määravad, mida süsteem peab suutma teha, et täita oma eesmärgi. Nõuded kirjeldavad näiteks süsteemi ärioloogikat, kasutajate autentimist ning ligipääsuõiguste haldust [31].

3.1.1 Raamatukogu ruumide ületoomine

Autoriseerimislahenduse funktsionaalsed nõuded on koostatud tuginedes Tallinna Tehnikaülikooli raamatukogu ruumide ja Kinnisvaraosakonna ruumide broneerimisreeglitele ning paralleele võib leida VCU raamatukogude ruumibroneerimise uuringust [32]:

- Rühmatööruumid on mõeldud kasutajate grupile 3-8 inimest.
- Individuaaltööruumi saab broneerida kuni 2 inimest.
- Kasutusaeg on 15min - 4 tundi päevas.
- Ühele inimesele on lubatud maksimaalselt 2 raamatukogu ruumi broneeringut päevas ja nädalas.
- Raamatukogu ruumi on võimalik broneerida maksimaalselt 14 päeva ette.
- Ülikooliga seotud isikul saab olla ainult 1 ruum broneeritud korraga Raamatukogu ja KVO osakonna ruumide peale.
- Ruume saab broneerida ainult maja lahtiolekuaegadele.

3.1.2 Broneeringu taotluse funktsionaalsuse loomine

Broneeringu taotluse funktsionaalsuse loomise funktsionaalsed nõuded on koostatud BRON rakenduse IT projektijuhi poolt:

- Nuppu „Taotle ruumi“ kuvatakse kui kasutajal puudub õigus ruumi broneerida.
- Ruumi taotlemisel on võimalik kasutajal lisada mitu ruumi numbrit.
- Kasutaja saab valida lisateenuseid ruumi broneerimiseks.
- Ruumi broneerimiseks esitatud taotlus saab staatuse "Kinnitamisel".
- Kui ruumihaldur lisab kommentaari kasutajale, saab taotlus staatuse "Lisainfo ootel".
- Kui taotluse menetlemine tühistatakse, saab taotlus staatuse "Tagasi lükatud".
- Kasutajal on võimalik vaadata taotluse detailvaadet.
- Kasutajal on võimalik tühistada esitatud taotlust.

3.2 Mittefunktsionaalsed nõuded

Mittefunktsionaalsed nõuded määratlevad süsteemi kvaliteediomadused – need kirjeldavad, kuidas süsteem peaks funktsionaalsed nõuded täitma. Nende alla kuuluvad näiteks jõudlus, skaleeritavus, töökindlus ja turvalisus [31]. Lõputöö mõlema osa mittefunktsionaalsed nõuded on eeldefineeritud Tallinna Tehnikaülikooli Infotehnoloogia osakonna poolt. Autor toob välja olulisemad nõuded:

- Andmebaasi tabelite ja veergude nimed peavad olema inglise keeles.
- Rakenduse kasutamisel tekkinud veale peab kasutajaliides vastama kasutajale tema eelistatud keeles (eesti- või inglisekeelse) kasutajasõbraliku veateatega, mis sisaldab soovituslikult ka vea koodi. Veateated peavad olema hallatavad.
- Rakenduse värviskeem ja logo kasutamine peab vastama TalTech ametlikule visuaalsele identiteedile.
- Muutujate, tüüpide ja funktsioonide nimed peavad olema sisulised ja andma aimu nende otstarbest.
- Arendus peab olema orienteeritud toodangukeskkonnas toimimiseks.
- Rakenduse äriloogika tuleb realiseerida andmebaasist eraldi sõltumatus rakenduskihis.

Järgnevalt on toodud näide, mis illustreerib seost ärivajaduse ja tehnilise nõude vahel. Ärivajadus on, et ühele inimesele on lubatud maksimaalselt kaks raamatukogu ruumi broneeringut päevas ja nädalas. Teenus peab tagama, et kasutaja broneeringuid kontrollitakse vastavalt sellele piirangule. Selle teostamiseks lisatakse tagakomponendi äriloogika kihis ruumi broneerimisel kontroll raamatukogu kasutajate broneeringute arvu üle.

4. Kasutatud tehnoloogiad

Järgnevates alapeatükkides antakse ülevaade süsteemi arhitektuurist ja süsteemi arendamisel kasutatud tehnoloogiatest. Nii esirakenduse, tagarakenduse kui ka andmebaasi tehnoloogia valik oli tehtud enne bakalaureusetöö alustamist Tallinna Tehnikaülikooli Infotehnoloogia osakonna poolt.

4.1 Rakenduse komponendid

Esirakendus on kirjutatud kasutades React raamistikku. React paistab silma paindlikkuse ja suure ökosüsteemiga, mis muudab selle sobivaks projektidele, mis nõuavad kohandatud lähenemist. Selle komponentide taaskasutatavus ja jõudlus, mida toetab virtuaalne DOM, muudavad selle atraktiivseks valikuks arendajatele, kes otsivad modulaarset ja skaleeritavat arhitektuuri [33].

Tagarakendus on kirjutatud kasutades PHP Laravel raamistikku. Laravel eristub oma arhitektuuri poolest, kus andmete haldamine, ärioloogika ja kasutajaliides on üksteisest eraldatud, võimaldades arendajatel rakendust lihtsamini hooldada ja testida. Raamistik pakub mitmeid sisseehitatud tööriistu, mis kiirendavad arendusprotsessi ja vähendavad ühekordse koodi hulka [34].

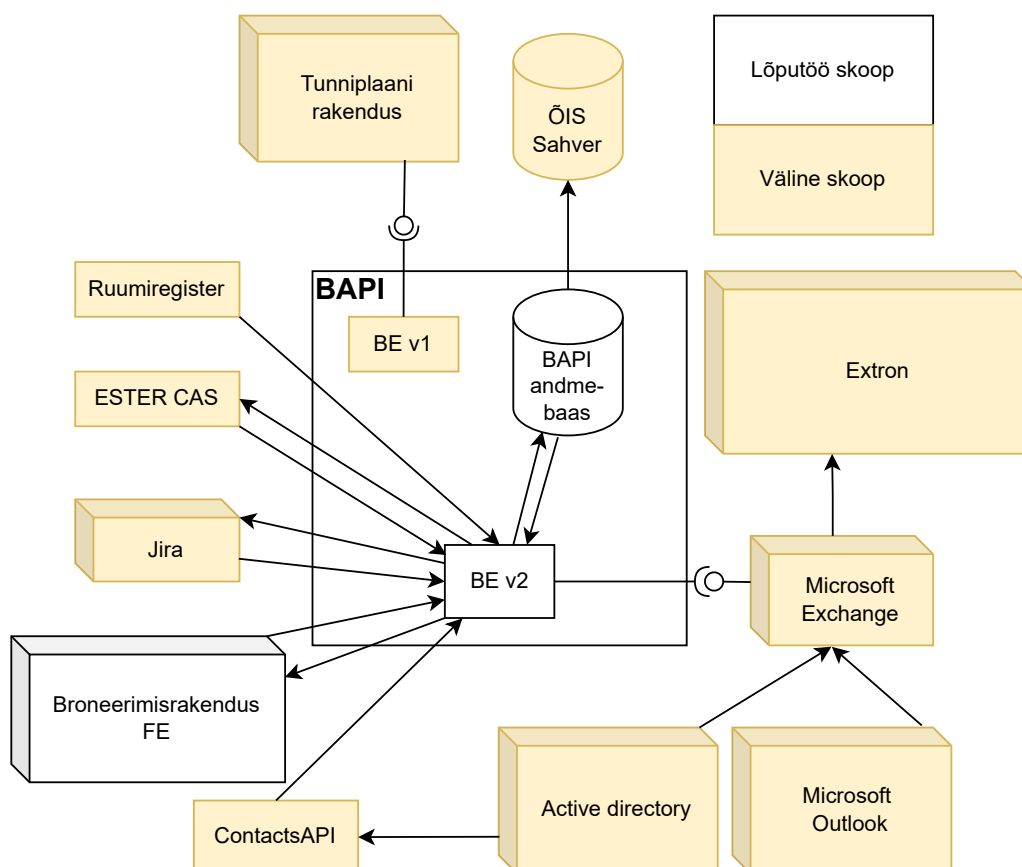
Rakendus kasutab MariaDB andmebaasi. MariaDB on avatud lähtekoodiga relatsiooniline andmebaas, mis on välja arendatud MySQL-i põhjal. MariaDB säilitab MySQL-iga kõrge ühilduvuse, kuid pakub paremat jõudlust ja täiustatud funktsionaalsust. Samuti on see usaldusväärne, skaleeritav ning vaba kommertsipiirangutest [35].

4.2 Rakenduse arhitektuur

Rakendus kasutab mikroteenuste arhitektuuri. Mikroteenuste arhitektuur põhineb mitmel lõdvalt seotud (ingl. k *loosely coupled*) individuaalsel teenusel, mis töötavad iseseisvalt ja suhtlevad teiste teenustega läbi hästi määratletud API-de, võimaldades süsteemi luua uut ärioloogikat ilma kogu rakendust korruga muutmata. Iga mikroteenus on väike, autonoomne komponent, mis vastutab ühe piiratud ärifunktsiooni eest, omab oma andmemudelit ja andmebaasi ning saab sõltumatult arendada, testida ja skaleerida [36].

Hetkeses rakenduses on järgmised komponendid (visualiseeritud joonisel 1):

1. Tunniplaani rakendus - Tunniplaani koostamisega seotud broneerimisloogika on implementeeritud rakenduse esimese versiooni tagarakenduses.
2. ÕIS Sahver - ÕIS vahebaas, mille kaudu toimub broneerimiskrakenduses loodud eksamite ja arvestuste aegade edastamine ÕISi.
3. Extron - Õpiruumidesse tehtud ruumibroneeringute välja näitamine uste kõrval asuvatel ekraanidel.
4. Microsoft Exchange - Saadab ruumibroneeringute kohta e-posti teavitusi.
5. Microsoft Outlook - Hoiustab broneeritavate ruumide ja kasutajate e-posti kontosid.
6. Active directory - Hoiustab infot kasutajate õiguste kohta (kuulumine gruppidesse, nendele on rakendustes antud konkreetsed õigused).
7. Broneerimiskrakendus FE - Kasutajaliides broneerimiskrakendusele.
8. Jira - Kasutatakse ruumi broneeringute taotluse menetlemiseks.
9. ContactsAPI - Vahendab ülikooliga seotud isikute informatsiooni Active Directoryst.
10. ESTER CAS - Kasutatakse raamatukogu kasutaja autentimiseks.
11. Ruumiregister - Hoiustab broneeritavate ruumide andmeid.



Joonis 1. Rakenduse arhitektuur.

5. Lahenduste arendus

Järgnevas peatükis käsitletakse lõputöö peamisi komponente ja lahendusteni jõudmise metoodikat. Lähemalt tutvustatakse mõlema osa tehnilist teekonda ja tehtud valikuid.

5.1 Metoodika

Agiilsetele metoodikatele kohaselt oli lahenduste analüüs ja arendamine jaotatud kahe nädala pikkusteks *sprintideks*, kuna see ajavahemik võimaldab efektiivselt planeerida, arendada ja testida uusi funktsionaalsusi [37]. *Sprindid* on näidatud tabelis 5:

Tabel 5. *Sprindid*.

<i>Sprint</i>	Töömaht	Ülesanded
1	7 piletit - 15h	Huvitatud osapoolte, ärinõuete ja kasutuslugude analüüs
2	7 piletit - 16h5min	Sisselogimislehe disain, autoriseerimislahenduse analüüs
3	4 piletit - 9h30min	Testimise analüüs, sisselogimislehe arendamine
4	4 piletit - 97h	Autoriseerimislahenduse arendamine
5	3 piletit - 14h	Broneeringu taotluse osa analüüs
6	2 piletit - 5h35min	Broneeringu taotluse osa arendusega alustamine
7	3 piletit - 35h	Ruumi taotluse loomise lahenduse arendamine
8	2 piletit - 80h30min	Taotluse haldamise ja tühistamise arendamine
9	2 piletit - 17h	Kasutajatestide läbiviimine ja taotluse osa täiendamine
-	2 piletit - 67h35min	Dokumendi kirjutamine ja projektihaldus

5.2 Dokumenteerimine

Järgnevas peatükis käsitletakse arendatud lahenduste dokumenteerimist. Dokumentatsioon soodustab teadmusjagamist, mis annab meeskonnaliikmetele ülevaate protsessidest ja lõpptulemustest [38]. Broneerimisrakenduse dokumentatsioon asub Confluence [39] keskkonnas ja autori dokumenteeritud töö lisatakse sinna. Autor dokumenteeris tehtud töö kasutades erinevaid kasutusvooge ja järjestusdiagramme.

5.2.1 Kasutusvood

Enne lahenduste arendamist kirjeldas autor peamised kasutajalood, mida loodud lahendused peavad toetama. Bakalaureusetöö jaoks on need kohandatud kasutusjuhtudeks. Tudengil, töötajal ja tudeng-töötajal on raamatukogu kontekstis samad õigused ja edaspidi on kõik kolm ühe termini "Tudeng" all. Autoriseerimislahenduse kontekstis olulised rollid on:

- Tudeng
- Ülikooliväline kasutaja
- Raamatukogu töötaja

Taotlussüsteemi kontekstis olulised rollid on:

- Tudeng
- Ruumihaldur

Kasutusjuht (ingl. k *use case*) kirjeldab süsteemi ja välise aktori (kasutaja või muu süsteemi osa) vahelist suhtlust samm-sammult, keskendudes tingimustele, käitumuslikele voogudele ja erandolukordadele [40]. Kasutajalugu (ingl. k *user story*) on lihtne, lühikese vormiga kirjeldus, mis väljendab funktsionaalse nõude ärilist väärtust lõppkasutaja perspektiivist [41].

Lisas 2 on näidatud nii autoriseerimislahenduse kui ka taotlussüsteemiga seotud peamised kasutajavood. Kõigepealt on näidatud nii ülikoolivälise kasutaja kui ka tudengi kogu voog, mis tuleb läbida, et broneerida raamatukogu ruum. Voog algab bakalaureusetöö raames loodud avalikul sisselogimislehel, kus on võimalik valida sisselogimismeetod. Täiendavat taotlusprotsessi raamatukogu individuaal- ja rühmatööruumide puhul ei rakendata.

Samuti on lisas 2 näidatud kogu voog, mida ülikoolisisene kasutaja peab läbima, et esitada taotlus ruumi broneerimiseks. Voog algab ruumi otsingu vaates, kus kasutaja valib sobiva ruumi.

5.2.2 Järjestusdiagrammid

Lisas 3 on näidatud nii autoriseerimislahenduse kui ka taotlussüsteemiga seotud järjestusdiagrammid, mis iseloomustavad tehnilist teekonda. Autentimis- ja autoriseerimisvoog kujutab ESTERi kontoga sisselogimist.

Taotluse loomise järjestusdiagramm iseloomustab taotluse loomist. Tagarakendus pärib nii taotleja kui ka ruumihalduri Jira kasutaja ID-d, et määrata pileti loomisel pileti aruandja (ingl. k *reporter*) ja vastutaja (ingl. k. *assignee*). Pileti edukal loomisel salvestatakse rakenduse andmebaasi iga valitud kuupäeva ja ruumi kohta sündmus ja taotlus.

Taotluse muutmise järjestusdiagramm iseloomustab protsessi, mille käigus lisatakse taotlusele kommentaare ja manuseid. Kommentaare on võimalik lisada tehes POST-päringu lõpp-punktile `/rest/api/3/issue/{issueIdorKey}/comment`. Tehes POST päring lõpp-punktile `/rest/api/3/issue/{issueIdorKey}/attachments` on võimalik manuseid lisada. Et manused ilmuksid ka pileti kommentaariumis, tuleb peale manuse lisamise luua uus kommentaar, mis viitab äsja üleslaaditud failile.

5.2.3 Andmebaasi skeemi osad

Autor tegi olemasolevasse andmebaasi muudatusi nii autoriseerimise kui ka taotlussüsteemi osas. Lisas 4 on kujutatud nii autoriseerimisega kui ka taotlussüsteemiga seotud andmebaasi skeemi osad.

Tabelis `oauth_access_tokens` hoitakse lühiajalisi ligipääsuõigusi, mis kinnitavad API-päringute tegija identiteeti ja õigusi. Kui ligipääsuluba aegub, saab kliendi rakendus esitada serverile uuendusloa ja taotleda selle alusel uut ligipääsuluba, vähendades korduvate sisselogimiste vajadust. Iga uuendusluba salvestatakse `oauth_refresh_tokens` tabelis, kus iga rida tähistab ühte pikema elueaga uuendusluba, mida saab vajadusel kasutada uue lühiajalise ligipääsuõiguse saamiseks.

Tabel `events` on loodud enne lõputööd ja autor seda ei muutnud. Taotlusi ja viiteid üritustele, mis on loodud taotlustega, salvestatakse `room_booking_requests` tabelis. Iga erineva kellaaja ja ruumi kohta luuakse erinev üritus ja taotlus.

5.3 Ühine sisselogimisleht

Autor valis ülikooliväliste kasutajate rakendusse sisenemise meetodiks ühise sisselogimislehe, sest see võimaldab kõigil kasutajatel autentida end samalt ja selgelt määratletud leheküljelt. Teisena kaaluti sisselogimismodaali kasutuselevõttu, mille korral rakenduse avaleht jääks tagataustale, kuid see lähenemine ei ole rakenduse loogika tõttu optimaalne, kuna eri kasutajatüüpidele kuvatakse erinevaid ruumivaateid juba enne autentimist.

Sisselogimislehe disain järgib lihtsuse ja selguse põhimõtteid, et kasutaja ei peaks sobivat

autentimismeetodit valides kahtlema, millist nuppu vajutada [42].

Sarnast lahendust kasutab näiteks StackOverflow [43], mille sisselogimislehel kuvatakse kasutajale erinevaid IDP-de sisselogimismeetodeid. See lähenemine võimaldab kasutajal valida sobivaima autentimismeetodi vastavalt tema eelistustele ja vajadustele, pakkudes lihtsamat ja kiiremat juurdepääsu süsteemile. Erinevate autentimisvõimaluste kuvamine samal lehel vähendab segadust ja aitab vältida liigset infot, tagades samas, et kasutaja ei kaotaks aega sobiva meetodi otsimisele.

Laboriuuring „Influences of Displaying Permission-related Information on Web Single Sign-On Login Decisions” kinnitab, et 200 osalejaga eksperimentides vähendas kõrvuti esitatud SSO-valikud kasutajate kognitiivset koormust võrreldes hierarhiliste menüüdega. Uuringu tulemused rõhutavad, et mitme autentimisvõimaluse korraga näitamine parandab kasutajaliidese intuiivsust ning muudab sisselogimisprotsessi kasutajatele lihtsamaks ja selgemaks [44].

5.4 Autoriseerimine

Lõputöö raames loodi autoriseerimislahendus, mis võimaldab ülikoolivälistel kasutajatel turvaliselt rakendusele ligipääseda. Peale esirakenduse päringut jätkub autentimine vastavalt sisselogimismeetodile. Raamatukogu kasutaja puhul kasutatakse autentimiseks ESTERi SSO süsteemi, mille kaudu tuvastatakse kasutaja identiteet. Pärast edukat autentimist suunatakse kasutaja rakendusse tagasi, kus tagarakendus valideerib autentimisandmed ja algatab autoriseerimisprotsessi. Autoriseerimine on ülesehitatud OAuth 2.0 standardi alusel, kasutades rollipõhist ligipääsukontrolli (RBAC). See võimaldab määrata erinevaid ligipääsuõigusi sõltuvalt kasutaja rollist, näiteks tavakasutaja või ruumihaldur.

Pärast autentimist kutsutakse välja abstraktne autoriseerimismeetod, kus genereeritakse kasutajale juurdepääsu- ja uuendusload (ingl k. *access* ja *refresh tokens*), mis võimaldavad piiratud aja jooksul API-dele turvalist ligipääsu. Autoriseerimismeetodisse antakse kaasa autentimismeetodile vastav(ad) kasutaja roll(id). Esialgselt kasutati Laravel Passport [45] teeki, kuid piiratud paindlikkuse tõttu asendati see iseseisvalt implementeeritud JWT-põhise lahendusega [46]. JWT [47] sisaldab kasutajaga seotud metaandmeid nagu kasutajanimi, e-posti aadress ja rollid.

Tokenid väljastatakse tagarakenduses ja edastatakse esirakendusele, kus need salvestatakse brauseri küpsistena. Kuna kogu autoriseerimisloogika asub tagarakenduses, on võimalik jõustada tsentraalne ligipääsukontroll ning vähendada kliendipoolse manipuleerimise riske [48].

Ligipääsuloa aegumisel saab klient esitada uuendusloa alusel uue taotluse, vältides vajadust iga kord uuesti sisselogida. Vastavad *tokenid* talletatakse andmebaasis, mis võimaldab hallata seansside kehtivust ja tuvastada lubade väärkasutust.

Sarnast autoriseerimismudelit kasutab näiteks Google Workspace for Education [49], kus OAuth 2.0 ja JWT tokenid moodustavad aluse ligipääsuõiguste haldamiseks. Kasutajate identiteet pärineb Google'i keskse identiteedihalduse süsteemist, mille kaudu määratakse neile rollid (nt õpilane, õpetaja, administraator), mille alusel antakse neile ligipääs konkreetsetele teenustele nagu Google Classroom [50] või Drive [51]. Süsteem võimaldab ka organisatsioonil kehtestada täiendavaid poliitikaid, nt piirates ligipääsu, vaid konkreetse kooli domeeniga seotud kontodele.

5.5 Taotlussüsteem

Taotlussüsteem loodi selleks, et kasutajatel oleks rakenduse siseselt võimalik taotleda broneeringuid ruumidesse, mis nõuavad täiendavat menetlusprotsessi. Süsteem võimaldab ülikoolisisesel kasutajal esitada taotluse soovitud ruumi broneerimiseks, märkides ära vajaliku kuupäeva(d), ajavahemiku, osalejate arvu ning muu vajaliku info ürituse kohta. Taotluse esitamine toimub ruumi detailvaates, mille kaudu kasutaja saab kogu protsessi jooksul ka oma taotluse hetkeseisu jälgida, lisada vajadusel kommentaare või täiendavaid faile (näiteks sündmuse kava või muud lisadokumendid), ning suhelda ruumihalduriga.

Tehnilise lahenduse keskmes on taotluste automaatne suunamine Jira piletisüsteemi, mis on Kinnisvaraosakonna poolt igapäevaselt kasutatav töökorralduse vahend. Iga esitatud taotluse korral loob süsteem uue Jira pileti, kus määratakse automaatselt nii taotluse esitanud isik kui ka vastutav ruumihaldur. Pileti kaudu toimub edasine menetlus, sealhulgas kinnitamine, küsimuste esitamine, lisamaterjalide nõudmine või taotluse tagasilükkamine. Tagarakendus suhtleb Jira süsteemiga API päringute kaudu, mille kaudu saadakse ligipääs taotlusega seotud andmetele. Vajalikud metaandmed salvestatakse rakenduse andmebaasi.

Süsteem toetab kahepoolset suhtlust taotleja ja ruumihalduri vahel. Kommentaaride ja manuste lisamine toimub samuti Jira kaudu, kuid need on kuvatud kasutajale otse rakenduses. Kui kasutaja lisab manuse, siis see saadetakse esmalt Jira süsteemi ning tehakse vastav päring, et manusele luua ka viitav kommentaar. Selline lahendus tagab, et kogu suhtlus on ühes kohas koondatud ning hiljem vajadusel jälgitav.

Oluliseks osaks süsteemis on automaatne sünkroniseerimine. Selleks, et kasutaja ei peaks igal hetkel taotluse olekut käsitsi kontrollima, on rakendus seadistatud Jira Webhookide [52] kaudu kuulama taotluse oleku muudatusi. Näiteks juhul kui ruumihaldur kinnitab

taotluse, saadetakse automaatne teade rakenduse vastavale lõpp-punktile, mille kaudu uuendatakse taotluse olekut rakenduse andmebaasis ning edastatakse vajadusel teavitust taotlejale. See lähenemine võimaldab vältida liigset koormust Jira API-le ning tagab efektiivse sündmuspõhise olekuhalduse.

Lõpuks on taotlussüsteem loodud nii, et see oleks võimalikult hästi laiendatav ja hooldatav. Kuna töövoog põhineb olemasolevatel süsteemidel (nt Jira), siis on välditud dubleerimist ja tagatud organisatsiooniline sobivus. Turvalisusele pöörati suurt tähelepanu, sealhulgas autentimis- ja autoriseerimismehhanismide ning andmekaitse osas.

Sarnast lähenemist on edukalt rakendatud ka teistes organisatsioonides. Näiteks kasutab Zendesk [53] oma klienditoe platvormi integreerimist Jira'ga, võimaldades klienditoe agentidel luua Jira pileteid otse Zendeskist. See integratsioon võimaldab arendusmeeskondadel jälgida klientide probleeme ja suhelda otse klienditoe agentidega, parandades koostööd ja kiirendades probleemide lahendamist. Lisaks võimaldab see kahepoolset suhtlust: arendajad saavad lisada kommentaare Jira's, mis kuvatakse automaatselt vastavas Zendesk'i piletil, tagades sujuva ja läbipaistva suhtluse mõlemas suunas.

6. Tulemused

Lõputöö üheks eesmärgiks oli autoriseerimislahenduse arendamine, mis võimaldab ülikoolivälistel ESTERi kasutajatel ligipääseda BRON rakendusele ja broneerida Tallinna Tehnikaülikooli raamatukogu ruume. Lisaks oli eesmärgiks ruumi broneeringu taotluse funktsionaalsuse loomine, mis võimaldab kasutajal taotleda broneeringuid rakenduses. Oodatud tulemused olid:

- Ülikoolivälisel kliendil on võimalik broneerida ainult raamatukoguga seotud ruume.
- Ülikoolisisene klient näeb raamatukogu ruume.
- Broneeriija peab nõustuma ruumi kasutamistingimustega ja seejärel on võimalik taotleda broneeringut.
- Taotluse peab üle vaatama üks ruumihalduritest.
- Menetlejal on võimalik suhelda broneerijaga rakenduse siseselt.
- Broneeringu taotlust on võimalik tagasilükata või kinnitada.
- Kinnitamine toimub kas rakenduse väliselt sõlmides lepingu või rakenduse siseselt juriidiliselt samaväärse meetodiga.

Kuna rakendus töötleb isikuandmeid ja rakendust arendatakse veel edasi, siis mõistagi peavad kõik loodud lahendused vastama turvalisuse nõuetele, olema lihtsalt hallatavad, veakindlad ning hästi laiendatavad.

Lõputöö tulemuseks arendas autor iseseisvalt autoriseerimislahenduse, mis võimaldab ülikoolivälisel kliendil siseneda rakendusse, näha raamatukogu ruume ja broneerida neid ruume. Samuti implementeeris autor taotlussüsteemi, mis võimaldab broneerijal esitada taotlust, mille vaatab üle üks ruumihalduritest. Samuti on võimalik menetlejal ja taotlejal omavahel rakenduses suhelda, ning broneeringut nii kinnitada kui ka tagasilükata.

6.1 Valideerimine

Funktsionaalsuste toimimise ja kasutajamugavuse kontrollimiseks on oluline läbi viia testimine, et veenduda lahenduste sobivuses kasutajate ootustega ning tagada süsteemi praktiline kasutatavus. Uued funktsionaalsused testis enne kasutuselevõttu rakenduse projektijuht.

Ükski üksik mõõdik ei suuda täielikult kirjeldada kogu kasutajakogemuse ulatust ning

liiga paljude mõõdikute kasutamine võib viia segaduseni ja vähendada fookust. Seetõttu keskendus autor vaid neile mõõdikutele, mis on lõputöö eesmärkide ja konteksti seisukohalt kõige olulisemad. Mõõdikute valik põhines neljal kriteeriumil:

- Rakendatav - mõõdik peab andma sisulist tagasisidet, mille põhjal saab disainiotsuseid teha ja täiustada rakendust [54].
- Asjakohane - mõõdik peab olema kooskõlas rakenduse eesmärkide ja arendusetapiga [54].
- Korduvmõõdetav - mõõdik peab andma stabiilseid tulemusi erinevates kasutajates-timise sessioonides ja keskkondades [54].
- Võrreldav - mõõdikut peab olema võimalik võrrelda varasemate tulemustega [54].

Lähtudes nendest põhimõtetest valis autor neli mõõtmismeetodit, mis olid kõige paremini kooskõlas ülaltoodud väärtustega.

Ülesande edukuse määr näitab, kui suur osa kasutajatest suudab konkreetse ülesande edukalt lõpule viia ilma vea või katkestamiseta [55]. See on üks kõige otsesemaid mõõdikuid kasutajakogemuse hindamiseks, kuna see peegeldab kasutaja suutlikkust täita eesmärki.

Ülesandele kulunud aeg mõõdab aega, mis kulub kasutajal ülesande täitmiseks. Lühem aeg võib viidata paremale kasutajamugavusele, kuid oluline on hinnata seda koos edukuse määraga - liiga lühike aeg võib tähendada ka katkestatud või valesti sooritatud ülesannet [55].

Süsteemi kasutatavuse skaala (SUS) on standardiseeritud küsimustik, mis koosneb kümnest väitest, mida hinnatakse viiepunktilise skaalaga. See annab kvantitatiivse hinnangu süsteemi üldisele kasutatavusele ja on laialdaselt kasutatud referentsväärtuste loomiseks [56].

Tajutav kasutuslihtsus hindab, kui lihtsalt kasutajad tunnetavad süsteemi kasutatavust subjektiivselt. See aitab mõista kasutajate ootusi ja tunnetuslike takistusi, mida kvantitatiivsed mõõdikud ei paljasta [57].

Autor testis autoriseerimislahendust 20 osalejaga, et tagada statistiliselt usaldusväärsed tulemused kvantitatiivsete mõõdikute põhjal [58]. Taotluse osa testis autor 5 osalejaga, kuna varasemate sarnaste taotluste esitajate hulk on piiratud ja uuringud on näidanud, et viiest kasutajast piisab ligikaudu 85% kasutatavuseprobleemide avastamiseks [59].

6.2 Testid

Raamatukogu osa testimiseks täitis iga osaleja kaks ülesannet, mis on kirjeldatud lisas 3. Peale ülesannete täitmist vastas iga osaleja küsimustikule, mis hindas nii süsteemi kasutatavuse skaalat (SUS) kui ka tajutavat kasutuslihtsust. Küsimustik on esitatud lisas 4. Iga testija tegi samad ülesanded ja vastas samale küsimustikule nii olemasoleva lahendusega kui ka lõputöö käigus arendatud lahendusega.

SUS skoori arvutamisel teisendatakse esmalt iga kasutaja vastus 1-5 skaalalt järgmiselt: paaritute küsimuste puhul lahutatakse kasutaja punktist üks ja paarisarvuliste puhul lahutatakse kasutaja punkt vastandina viiest. Saadud väärtused liidetakse kokku ning korrutatakse 2.5-ga, et viia tulem vahemikku 0 - 100 [60]. SUS skoori tõlgendamisel on järgmised tasemed:

- Halb: kuni 49
- Rahuldav: 50 - 70
- Hea: 71 - 80
- Suurepärase: 81 - 90
- Ideaalne: 91 - 100

Testimise tulemuste võrdlus on lisas 5. Tabelites on näidatud keskmised näitajad kõikide testijate vahel. Raamatukogu osa mõlemas ülesandes on uue lahenduse puhul vigade arv vähenenud vähemalt 45%. Samuti raamatukogu ruumide broneeringu muutmine on 17% kiirem. See viitab, et broneerimisrakenduses on ruumide broneerimine intuitiivsem, on vähem vigade altim ja pakub efektiivsemat lahendust.

Nii süsteemi kasutatavuse kui ka tajutava kasutuslihtsuse skoor on uue lahenduse puhul parem. Vana lahenduse SUS skoor 56.2/100, mis on kvalifitseeritav kui "rahuldav" kasutatavus. Uue lahenduse SUS skoor on 82/100, mis on kvalifitseeritav kui "hea" kasutatavus. See tähendab märkimisväärset 45.8% tõusu kasutatavuse hinnangus.

Taotluse osa testimiseks vastasid osalejad esmalt küsimustele varasema kogemuse põhjal, hinnates olemasolevat süsteemi. Seejärel täitsid osalejad küsimustiku, mis hindas nii süsteemi kasutatavuse skaalat (SUS) kui ka tajutavat kasutuslihtsust. Teises testimise voorus täitsid osalejad praktilise ülesande uue süsteemiga ja vastasid samale küsimustikule. Varasema kogemuse küsimused on esitatud lisas 3 ning süsteemi kasutatavuse skaala ja tajutava kasutuslihtsuse küsimused lisas 4. Taotlussüsteemi implementeerides on taotluse esitamine 78% kiirem, mis on oluliselt efektiivsem. Vigade arv on vähenenud 62.5%, mis viitab, et uus lahendus on intuitiivsem ja vähem vigade altim. Vana lahenduse SUS skoor

oli 52.5/100, mis on kvalifitseeritav kui "rahuldav" kasutatavus. Uue lahenduse skoor ulatus 79.4 punktini, mis on kvalifitseeritav kui "hea" kasutatavus. Tegemist on 51.2% tõusuga.

Autoriseerimislahenduse ja taotlussüsteemi kõige suuremad positiivsed muutused olid süsteemi lihtsamaks kasutamiseks, suuremas soovis süsteemi sagedamini kasutada, süsteemi funktsioonide paremas integratsioonis ja vähenenud vajaduses tehnilise abi järele. Samuti paranesid oluliselt protsessi mõistmise lihtsus, esitamiseks vajaliku info selgus ja üldine rahulolu protsessiga.

6.3 Projekitjuhi tagasiside

BRON rakenduse projektijuht andis tööle ja töö autorile järgneva tagasiside: "Tudengi ülesandeks oli bakalaureuse lõputöö raames teostada arendustöid Tallinna Tehnikaülikooli (edaspidi TalTech) ruumide broneerimise rakendusele. Lõputöö tulemusena pidid valmima kaks uut funktsionaalsust: TalTechi raamatukogu kasutajatel tekib võimalus läbi broneerimisrakenduse broneerida raamatukogu ruume ning võimaldada majasisestel kasutajatel taotleda broneeringut ruumidele, mis peavad läbima Kinnisvaraosakonna kooskõlastusringi.

Tudengi tugevusteks olid kiire kohanemisvõime, lahenduste leidmisele orienteeritus ning hea analüüsivõime, mille tulemusel sai tudeng aru, milliseid küsimusi soovib äri lahendada ja kuidas seda teha kasutusmugavalt. Tudeng on tugev tiimikaaslane, mis väljendus arendustöid tehes, kui ta julgelt pöördus teiste tiimiliikmete poole, et kooskõlastada või läbi arutada, milline parem tehniline lahenduskäik sobiks näiteks raamatukogu kasutajate autentimiseks või ruumi broneeringu taotleja ja halduri suhtlemise tagamiseks. Tööde teostamisel oli tudeng iseseisev ja rahulik, sest analüüsi käigus täpsustunud vajaduste muutumisel suutis tudeng töö teostamisel endiselt tähtaegades püsida.

Tudeng on arendajana väärtuslik tiimikaaslane, kellele saab usaldada arendustöid ning kellega koostöös saab realiseeritud parim võimalik lahendus võttes arvesse hetke oskuseid ja teadmisi kõigi projektis osalejate poolt."

6.4 Edasiarenduse võimalused

Kinnisvarasakonna soov on täiendada sisselogimisvõimaluste valikut meetodiga "Jätka külalisena", mis lubab rakendust kasutada ka ilma eelneva Uni-ID või "Raamatukogu kasutaja" autentimiseta. Külalisena sisenedes peab kasutaja täitma oma andmed ning

esitama taotluse ruumi broneerimiseks. Samuti soovitakse, et raamatukogu kasutajal oleks ligipääs kõikidele ülikooli ruumidele, kuid Kinnisvaraosakonna ruumide broneerimiseks tuleb esitada taotlus.

Eelmainitud funktsionaalsuste implementeerimist raskendab asjaolu, et Jira pileti loomiseks peab olema taotlejal Jira konto. Seda probleemi saab lahendada kasutades Jira konto puudumisel teenuskontot. Lõputöö koostamise ajal ei ole võimalik Jira API kaudu luua Jira kontot.

Lõputöö raames loodud taotluse detailvaates on võimalik kuvada faili eelvaade. Tulevikus saab veel lisada failide allkirjastamise detailvaates ilma faili allalaadimata.

Autoriseerimislahenduse puhul on üldisemad komponendid sisselogimisvaade, autentimist suunav meetod tagarakenduses ja autoriseerimismeetod, mis genereerib vastavalt sisendile ligipääsu- ja uuendusload. Spetsiifilisem on autentimismeetodi enda meetod, mis lõputöö raames pärib ESTERist informatsiooni ja valideerib andmete õigsust. Teistes süsteemides kasutamiseks peab autentimist suunavasse meetodisse lisama nii uue välimoodustaja (IDP) konfiguratsiooni kui ka selle IDP kaudu autentiva meetodi, mis tagab vastava autentimisvoo käivitumise. Autoriseerimismeetodit pole pärast selle algset seadistust vaja muuta.

Taotlussüsteemi äri loogika on üles ehitatud modulaarse lahendusena, mida saab üks-ühele integreerida ka teistesse rakendustesse. Ainsana on BRON-spetsiifilised sündmused ja taotluste haldamine andmebaasis. Kõik ülejäänud komponendid ja vood töötavad ilma täiendavate kohandusteta igas kontekstis.

7. Kokkuvõte

Lõputöö üheks eesmärgiks oli autoriseerimislahenduse loomine, mis võimaldab Tallinna Tehnikaülikooli raamatukogu kasutajatel ligipääseda ülikooli ruumide broneerimisrakendusse. Teiseks eesmärgiks oli ruumi broneeringute taotlussüsteemi loomine, mis võimaldab rakenduse siseselt taotleda broneeringut ruumidesse, mille broneerimisõigus ei tulene automaatselt kasutaja õigustest rakenduses. Taotlussüsteem võimaldab ka taotleja ja ruumihalduri vahelist suhtlust, manuste esitamist ja staatuse muutmist rakenduses.

Eesmärkide saavutamiseks analüüsiti erinevaid autoriseerimislahendusi, sisselogimislehe häid tavaid ja taotluse detailvaate kasutajaliidese disaini. Samuti analüüsiti kasutajavooge, ärinõudeid ja Jira API dokumentatsiooni. Lõputöö raames valminud lahendus koosneb avalikust sisselogimislehest, OAuth 2.0 ja RBAC autoriseerimislahendusest, taotluse esitamise detailvaatest ja taotlussüsteemi loomisest, mis sisaldab erinevaid Jiraga suhtlemismetodeid.

Testimistulemused näitasid märkimisväärset edasiminekut võrreldes varasemate lahendustega. Süsteemi kasutatavuse skaalal tõusis raamatukogu osa hinnang 56.2 punktilt 82.8 punktini ning taotluse osa hinnang 52.5 punktilt 79.4 punktini. Praktiliste ülesannete sooritamisel vähenes vigade arv mõlemas osas vähemalt 45% ning kasutajate üldine rahulolu tõusis oluliselt. Need tulemused kinnitavad, et väljatöötatud lahendus täidab edukalt püstitatud eesmärgid ning pakub kasutajatele märkimisväärselt paremat kogemust.

Kasutatud kirjandus

- [1] TalTech. *Ruumide broneerimisrakendus*. [Kasutatud: 13.04.2025]. 2025. URL: <https://booking.taltech.ee/>.
- [2] ELNET Konsortsium. *E-kataloog ESTER*. [Kasutatud: 13.04.2025]. 2025. URL: <https://www.elnet.ee/teenused/e-kataloog-ester/>.
- [3] Atlassian. *Great outcomes start with Jira*. [Kasutatud: 28.04.2025]. 2025. URL: <https://www.atlassian.com/software/jira>.
- [4] Mainor Ülemiste AS. *Ülemiste City Future Forum*. [Kasutatud: 13.04.2025]. 2025. URL: <https://konverents.ulemistecity.ee/>.
- [5] Microsoft Learn. *What is the Microsoft identity platform?* [Kasutatud: 11.01.2025]. 2025. URL: <https://learn.microsoft.com/en-us/entra/identity-platform/v2-overview>.
- [6] Okta. *Central Authentication Service (CAS) Protocol Explained*. [Kasutatud: 11.02.2025]. 2025. URL: <https://www.okta.com/identity-101/central-authentication-service/>.
- [7] Elira Mollakuqe and Vesselina Dimitrova. “Comparative analysis of identity management, access control, and authorization practices in public and private universities”. In: *Open Research Europe* (2024). URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC11294802/>.
- [8] Richard Pang et al. “Zanzibar: Google’s Consistent, Global Authorization System”. In: *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*. USENIX Association, 2019. URL: <https://www.usenix.org/conference/nsdi19/presentation/pang>.
- [9] Auth0. *What is OAuth 2.0?* [Kasutatud: 11.01.2025]. 2025. URL: <https://auth0.com/intro-to-iam/what-is-oauth-2>.
- [10] Immuta. *What is RBAC (Role-Based Access Control) - And is it Right For You?* [Kasutatud: 11.01.2025]. 2025. URL: <https://www.immuta.com/guides/data-security-101/rbac-role-based-access-control/>.
- [11] OASIS. *OASIS: Advancing Open Standards for the Information Society*. [Kasutatud: 12.01.2025]. 2025. URL: <https://www.oasis-open.org/org/>.

- [12] OASIS XACML Technical Committee. *eXtensible Access Control Markup Language (XACML) Version 3.0 Core Specification*. [Kasutatud: 05.05.2025]. 2013. URL: <https://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html>.
- [13] NIST. *About NIST*. [Kasutatud: 12.01.2025]. 2025. URL: <https://www.nist.gov/about-nist>.
- [14] Zoran Duric and Serban Gavrilă. *Extensible Access Control Markup Language (XACML) and Next Generation Access Control (NGAC)*. [Kasutatud: 20.04.2025]. 2016. URL: https://www.academia.edu/81244839/Extensible_Access_Control_Markup_Language_XACML_and_Next_Generation_Access_Control_NGAC.
- [15] Kantara Initiative. *About Kantara Initiative*. [Kasutatud: 12.01.2025]. 2025. URL: <https://kantarainitiative.org/about-kantara-initiative/>.
- [16] Kantara Initiative. *User-Managed Access (UMA) Profile of OAuth 2.0*. [Kasutatud: 12.01.2025]. 2015. URL: <https://docs.kantarainitiative.org/uma/rec-uma-core.html>.
- [17] Walter Voronovic. *What Is a User-Friendly Interface? Two Examples*. [Kasutatud: 16.03.2025]. 2015. URL: <https://waltervoronovic.com/glossary/what-is-a-user-friendly-interface-explained/>.
- [18] A. Smith and B. Müller. “Measuring Cognitive Load in User Interfaces”. In: *International Journal of Human–Computer Studies* (2025). [Kasutatud: 16.03.2025]. URL: <https://www.sciencedirect.com/science/article/pii/S1474034625002216>.
- [19] R. Rosenholtz, Y. Li, and L. Nakano. “Measuring Visual Clutter”. In: *Proceedings of CHI 2005*. [Kasutatud: 16.03.2025]. 2005. URL: <https://web.mit.edu/rruth/www/Papers/RosenholtzEtAlCHI2005Clutter.pdf>.
- [20] W. Almeida, E. Duarte, and R. Gonçalves. “Efficiency, Effectiveness, and Satisfaction of Responsive Mobile Tourism Websites: A Mobile Usability Study”. In: *Journal of Travel Research* (2015). [Kasutatud: 16.03.2025]. URL: https://www.researchgate.net/publication/284929478_Efficiency_effectiveness_and_satisfaction_of_responsive_mobile_tourism_websites_a_mobile_usability_study.
- [21] RocketSpark. *10 creative ways to use accordions on your website to improve user experience*. [Kasutatud: 16.03.2025]. 2023. URL: <https://www.rocketspark.com/blog/post/360/10-creative-ways-to-use-accordions-on-your-website-to-improve-user-experience/>.

- [22] Nielsen Norman Group. *Modal & Nonmodal Dialogs: When (& When Not) to Use Them*. [Kasutatud: 16.03.2025]. 2017. URL: <https://www.nngroup.com/articles/modal-nonmodal-dialog/>.
- [23] TermsFeed. *Where should Terms and Conditions be displayed?* [Kasutatud: 16.03.2025]. 2025. URL: <https://www.termsfeed.com/faq/where-should-terms-and-conditions-be-displayed/>.
- [24] Steve Krug. *Don't Make Me Think: A Common Sense Approach to Web Usability*. [Kasutatud: 16.03.2025]. New Riders, 2000.
- [25] Hriday Checker. *The Power of Left Alignment: A User-Centric Perspective in Design*. [Kasutatud: 16.03.2025]. 2023. URL: <https://medium.com/design-bootcamp/the-power-of-left-alignment-a-user-centric-perspective-in-design-8546ea330c71>.
- [26] Ceren Budak et al. "Threading is Sticky: How Threaded Conversations Promote Comment System User Retention". In: *Proceedings of the ACM on Human-Computer Interaction* (2017). [Kasutatud: 16.03.2025].
- [27] Christoph M. Lüders, Amel Bouraffa, and Walid Maalej. *Beyond Duplicates: Towards Understanding and Predicting Link Types in Issue Tracking Systems*. [Kasutatud: 13.05.2025]. 2022. URL: <https://arxiv.org/abs/2204.12893>.
- [28] Kereshmeh Afsari, Chuck Eastman, and Dennis Shelden. "A Review of Building Permission Process Automation Using Building Information Modelling". In: *Buildings* (2022). [Kasutatud: 13.05.2025]. URL: <https://www.mdpi.com/2075-5309/12/1/45>.
- [29] M. F. M. Zulkipli, M. R. Majid, and M. E. Ayob. *Enhancement of Permit to Work System Toward Safety Risk Assessment Supporting by GPS, Face Recognition and IoT: Research on Company XYZ*. [Kasutatud: 13.05.2025]. 2021. URL: https://www.researchgate.net/publication/355020508_Enhancement_of_Permit_to_Work_System_Toward_Safety_Risk_Assessment_Supporting_by_GPS_Face_Recognition_and_IoT_Research_on_Company_XYZ.
- [30] Leslie P. Willcocks, Mary Lacity, and Andrew Craig. "Robotic Process Automation (RPA) Adoption: A Systematic Literature Review". In: (2022). [Kasutatud: 13.05.2025]. URL: https://www.researchgate.net/publication/362035572_Robotic_Process_Automation_RPA_Adoption_A_Systematic_Literature_Review.

- [31] Ironhack. *Functional vs Non-Functional Requirements: Understanding the Core Differences and Examples*. [Kasutatud: 13.04.2025]. 2024. URL: <https://www.ironhack.com/gb/blog/functional-vs-non-functional-requirements-understanding-the-core-differences-and>.
- [32] M. T. Doherty and E. R. White. "Room Reservations at VCU Libraries: How We Coped with Rapid Growth and Overwhelming Demand for Student Study Space". In: *College & Research Libraries News* (2012). [Kasutatud: 07.05.2025]. URL: <https://crln.acrl.org/index.php/crlnews/article/view/8722/9231>.
- [33] Phani Sekhar Emmanni. "Comparative Analysis of Angular, React, and Vue.js in Single Page Application Development". In: *International Journal of Science and Research (IJSR)* (2023). [Kasutatud: 20.04.2025]. URL: https://www.researchgate.net/publication/380291017_Comparative_Analysis_of_Angular_React_and_Vuejs_in_Single_Page_Application_Development.
- [34] A. Niarman, Iswandi, and A. K. Candri. "Comparative Analysis of PHP Frameworks for Development of Academic Information System Using Load and Stress Testing". In: *International Journal Software Engineering and Computer Science (IJSECS)* (2023). [Kasutatud: 20.04.2025]. URL: <https://pdfs.semanticscholar.org/21b8/b39fa83a751f412943ebf074511380bafa70.pdf>.
- [35] Nurhidayati, Rahmadya Gernowo, and Muhammad Suryanegara. "Comparative Analysis of MariaDB's Performance Efficiency as a Suitable Replacement for MySQL". In: *International Journal of Computer Applications* (2019). [Kasutatud: 21.04.2025]. URL: https://www.researchgate.net/publication/336530695_Comparative_Analysis_of_MariaDB's_Performance_Efficiency_as_a_Suitable_Replacement_for_MySQL.
- [36] Microsoft Learn. *Microservice architecture style*. [Kasutatud: 20.04.2025]. 2025. URL: <https://learn.microsoft.com/en-us/azure/architecture/guide/architecture-styles/microservices>.
- [37] Adarsh Anand et al. "Optimal Sprint Length Determination for Agile-Based Software Development". In: *Computers, Materials & Continua* (2021). [Kasutatud: 20.04.2025]. URL: <https://www.techscience.com/cmc/v68n3/42509/html>.
- [38] Vikas Sitaram Chomal and Jatinderkumar R. Saini. "Significance of Software Documentation in Software Development Process". In: *International Journal of Computer Applications* (2014). [Kasutatud: 20.04.2025]. URL: <https://www>.

- researchgate.net/publication/281965276_Significance_of_Software_Documentation_in_Software_Development_Process.
- [39] Atlassian. *Goodbye silos, hello teamwork*. [Kasutatud: 20.04.2025]. 2025. URL: <https://www.atlassian.com/software/confluence>.
- [40] Alistair Cockburn. *Writing Effective Use Cases*. [Kasutatud: 21.04.2025]. Addison-Wesley, 2000.
- [41] Mike Cohn. *User Stories Applied: For Agile Software Development*. [Kasutatud: 21.04.2025]. Addison-Wesley, 2004.
- [42] Vitaly Friedman. *10 Principles Of Good Web Design*. [Kasutatud: 21.04.2025]. 2021. URL: <https://www.smashingmagazine.com/2008/01/10-principles-of-effective-web-design/>.
- [43] Stack Overflow. *Stack Overflow – Where Developers Learn, Share, & Build Careers*. [Kasutatud: 09.05.2025]. 2025. URL: <https://stackoverflow.com/>.
- [44] Srivathsan G. Morkonda, Sonia Chiasson, and Paul C. van Oorschot. “Influences of Displaying Permission-related Information on Web Single Sign-On Login Decisions”. In: (2023). [Kasutatud: 09.05.2025]. URL: <https://arxiv.org/abs/2308.13074>.
- [45] Taylor Otwell. *Laravel Passport - The PHP Framework For Web Artisans*. [Kasutatud: 25.04.2025]. 2025. URL: <https://laravel.com/docs/11.x/passport>.
- [46] Gabriel L. Manor. *A Guide to Bearer Tokens: JWT vs. Opaque Tokens*. [Kasutatud: 25.04.2025]. 2025. URL: <https://www.permit.io/blog/a-guide-to-bearer-tokens-jwt-vs-opaque-tokens>.
- [47] JWT.io. *Introduction to JSON Web Tokens*. [Kasutatud: 25.04.2025]. 2025. URL: <https://jwt.io/introduction>.
- [48] Judith Kahrer. *Best Practices for Storing Access Tokens in the Browser*. [Kasutatud: 25.04.2025]. 2024. URL: <https://curity.medium.com/best-practices-for-storing-access-tokens-in-the-browser-6b3d515d9814>.
- [49] Google for Education. *Google Workspace for Education Overview*. [Kasutatud: 09.05.2025]. 2025. URL: <https://edu.google.com/workspace-for-education/editions/overview/>.
- [50] Google for Education. *Google Classroom*. [Kasutatud: 09.05.2025]. 2025. URL: <https://edu.google.com/workspace-for-education/products/classroom/>.

- [51] Google Workspace. *Google Drive: almacena y comparte archivos online*. [Kasutatud: 09.05.2025]. 2025. URL: <https://workspace.google.com/products/drive/>.
- [52] Atlassian. *Jira Webhooks*. [Kasutatud: 28.04.2025]. 2025. URL: <https://developer.atlassian.com/server/jira/platform/webhooks/>.
- [53] Zendesk. *Zendesk: The Complete Customer Service Solution*. [Kasutatud: 01.05.2025]. 2025. URL: <https://www.zendesk.com/>.
- [54] Tuba Ilkay Yildiz and Ahmet Topal. “Mobile health application for monitoring diabetes patients in Turkey: usability, satisfaction, and effectiveness evaluation”. In: (2024). [Kasutatud: 01.05.2025]. URL: <https://link.springer.com/article/10.1007/s12553-024-00859-2>.
- [55] Maze. *7 Key usability metrics to unlock user insights*. [Kasutatud: 01.05.2025]. 2023. URL: <https://maze.co/blog/measure-usability-metrics/>.
- [56] Interaction Design Foundation. *System Usability Scale for Data-Driven UX*. [Kasutatud: 01.05.2025]. 2025. URL: <https://www.interaction-design.org/literature/article/system-usability-scale>.
- [57] Hubble. *Usability Metrics that Matter*. [Kasutatud: 01.05.2025]. 2025. URL: <https://www.hubble.team/blog/ux-usability-metrics>.
- [58] Nielsen Norman Group. *Quantitative Studies: How Many Users to Test?* [Kasutatud: 01.05.2025]. 2006. URL: <https://www.nngroup.com/articles/summary-quant-sample-sizes/#:~:text=Summary%3A%2040%20participants%20is%20an,you%20can%20recruit>.
- [59] Nielsen Norman Group. *Why You Only Need to Test With 5 Users*. [Kasutatud: 01.05.2025]. 2000. URL: <https://www.nngroup.com/articles/quantitative-studies-how-many-users/>.
- [60] Test Science. *Measuring Usability*. [Kasutatud: 04.05.2025]. 2023. URL: <https://testscience.org/measuring-usability/>.

Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks¹

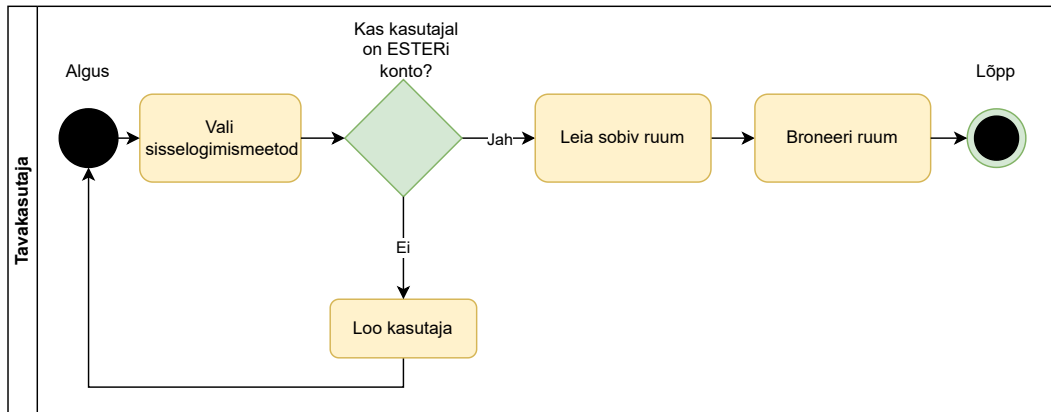
Mina, Rain Rähni

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose “RU-UMIBRONEERIMISE TAOTLUSTE MENETLEMINE JA RAAMATUKOGU RUUMIDE ÜHENDAMINE ÜHTSESSE SÜSTEEMI”, mille juhendaja on [Supervisor’s Name]
 - 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. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

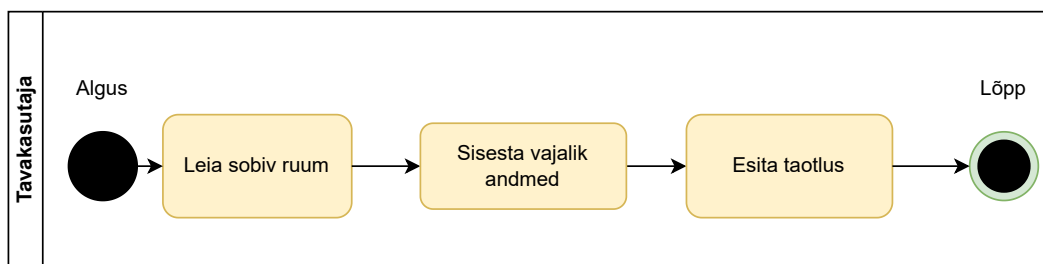
04.06.2025

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

Lisa 2 – Tegevusdiagrammid

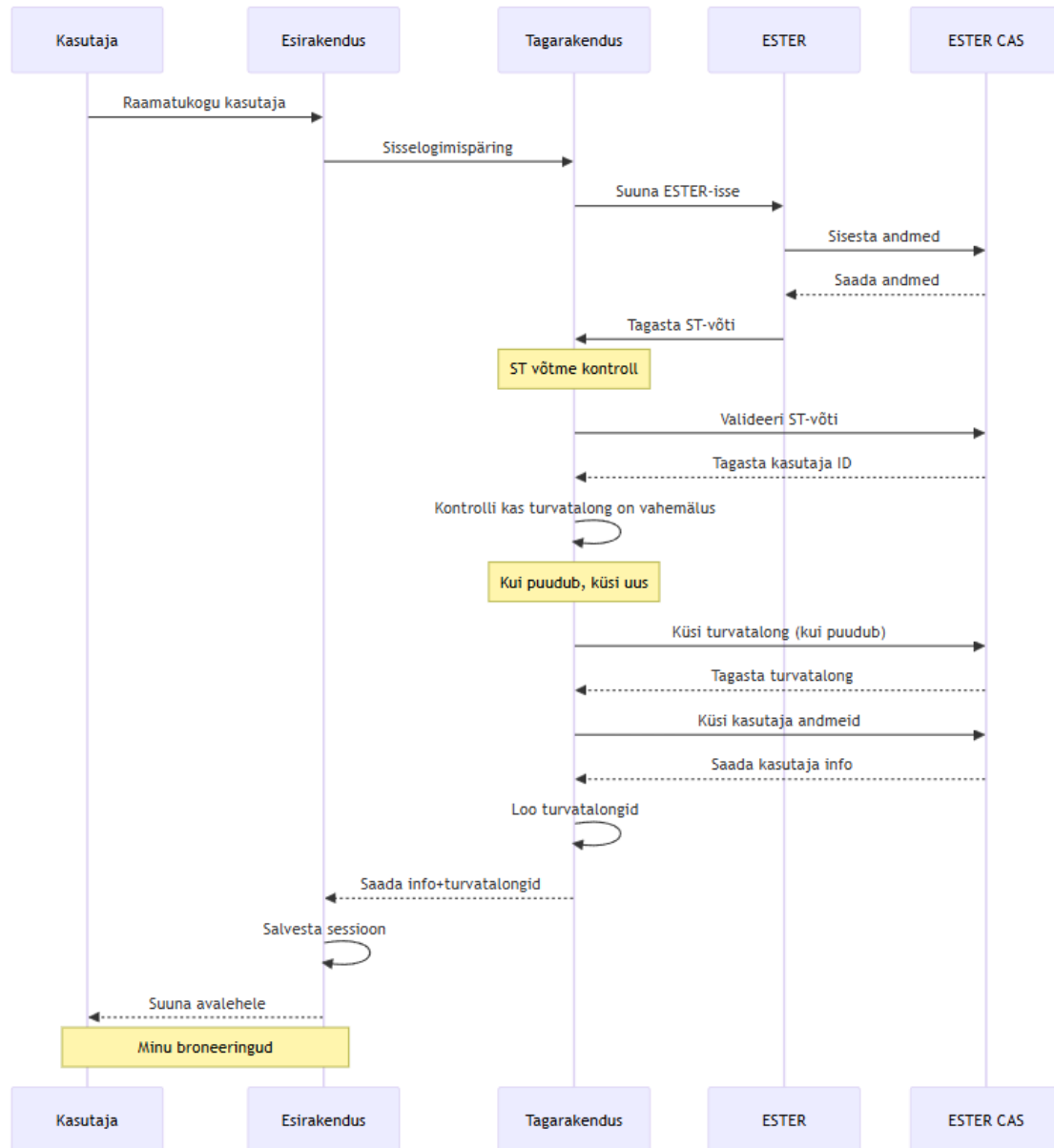


Tavakasutaja ruumi broneerimise tegevusdiagramm.

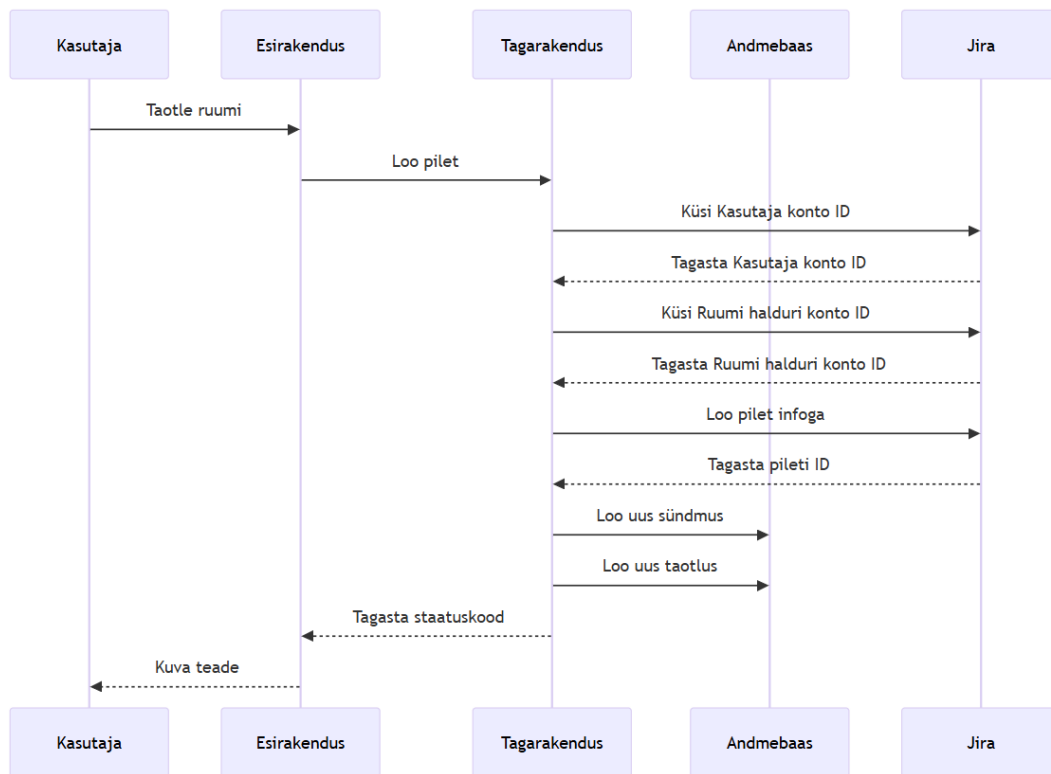


Ruumi broneeringu taotluse esitamise kasutajavoog.

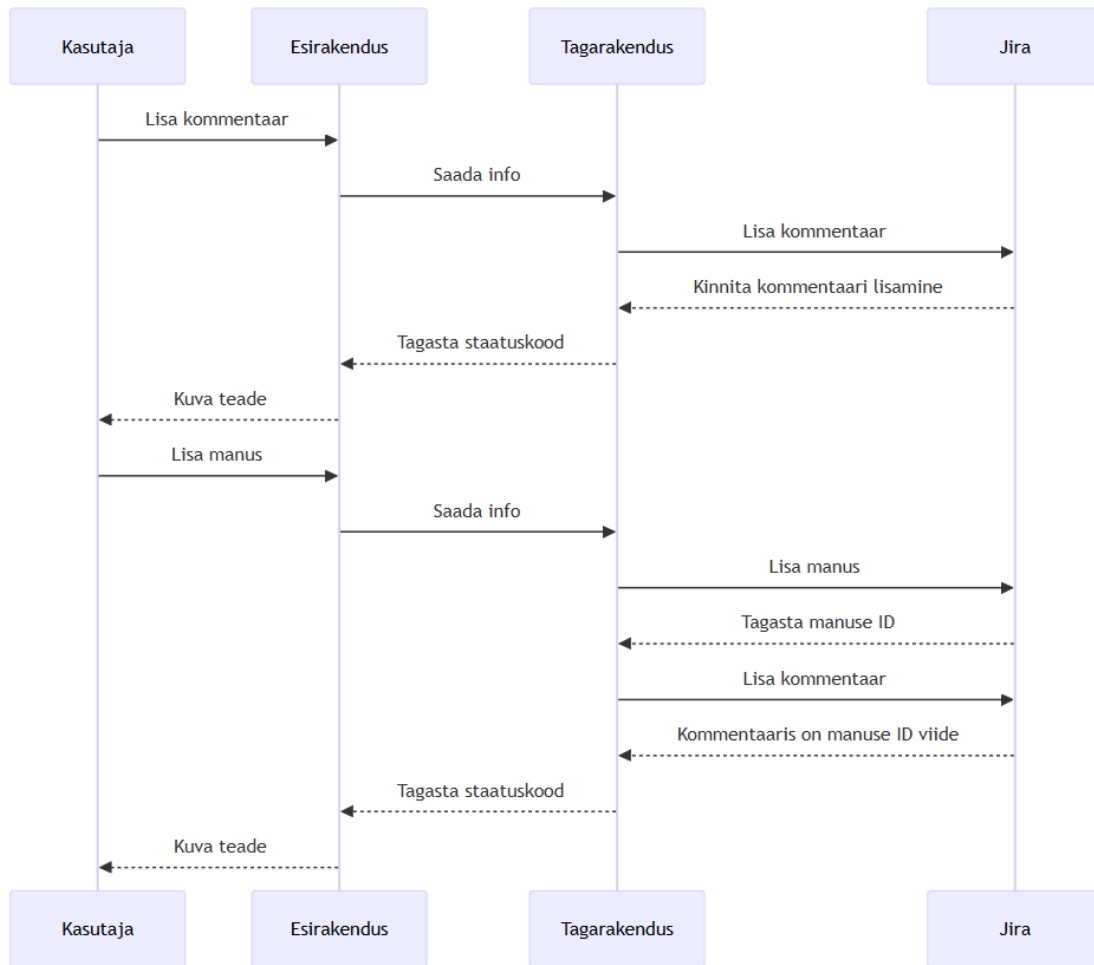
Lisa 3 – Järjestusdiagrammid



Autentimis- ja autoriseerimisvoog.

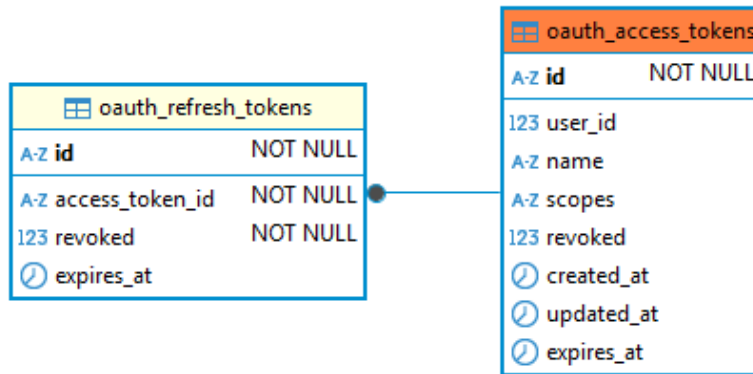


Taotluse loomise järjestusdiagramm.

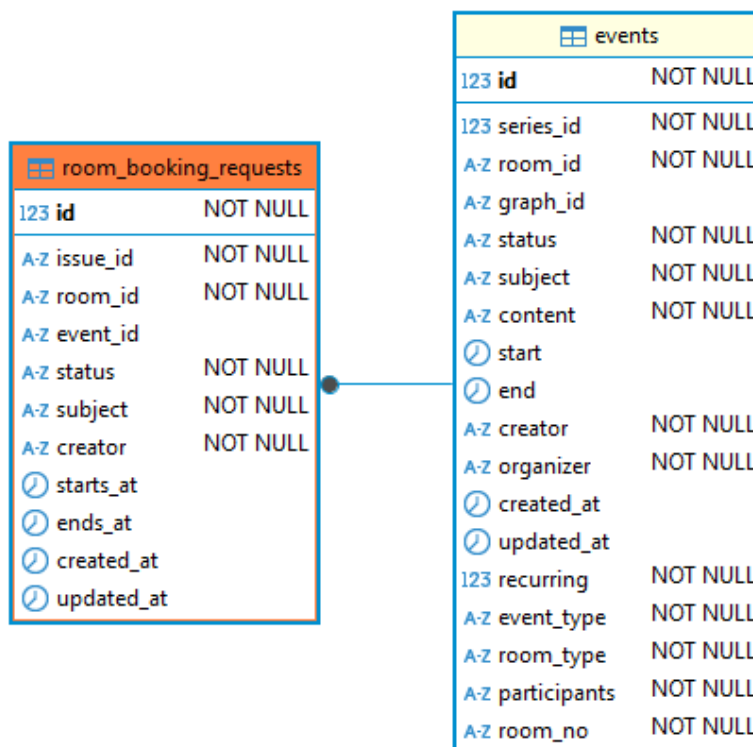


Taotluse muutmise järjestusdiagramm.

Lisa 4 – Andmebaasi skeemid



Autoriseerimise andmebaasi skeemi osa.



Ruumi broneeringu taotluste andmebaasi skeemi osa.


Lisa 5 – Loodud vaadete osad

RUUMIDE BRONEERIMINE [Tugiportaal](#) [Delta](#) [Välisveeb](#) [ÕIS](#) [Moodle](#) [SMART](#) [Õigusaktid](#) [Fitekln](#) [Research Professional](#) [Vihjellin](#) [Coursey](#) [VOK](#)

TAL TECH TALLINNA TEHNILIKULIKOOL

Ruumide broneeringud

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis cursus semper laculis. Suspendisse laoreet, dui vitae facilisis pretium, nunc neque faucibus orci, sit amet hendrerit dui dui sit amet metus. Duis ipsum nulla, ornare ac gravida quis, dignissim sed magna. Nullam commodo mi nec facilisis facilisis gravida. Maecenas euismod diam eu ipsum ullamcorper, quis pretium justo blandit. Praesent Praesent Praesent bibendum gravida quam. Sed porttitor, risus et finibus lobortis, mi eros laculis feugiat magna dolor sed nisl. Mauris scelerisque bibendum purus, non mollis augue tempus nec. Etiam ut luctus tortor.



Logi sisse

TalTech Uni-ID Raamatukogu kasutaja

Sisselogimisvaade.

Ruumi taotlemine

Ürituse toimumise aeg

Kuupäev *

13.05.2025



Algus *

13:00



-

Löpp *

14:00



[Lisa kuupäev](#)

Taotlus

Taotluse info

* - Tärniga märgitud väljade täitmine on kohustuslik

Renditavad ruumid*

U01-202



Lisateenused

Osalejate arv *

1

Telefoni number *

Ürituse pealkiri *

Lisa täpsem info



Kasutustingimused*



[← Tagasi](#)

Taotle ruumi

Ruumi broneeringu taotlemise vaate autori loodud info sisestamise osa.

Taotluse info

test - JSM-202

📅 26.04.2025 ⌚ 13:00 – 14:00

📅 27.04.2025 ⌚ 13:00 – 14:00

• Tühistatud

Osalejate arv

1

Taotluse kirjeldus



Ruum	Ruumi tüüp	Korrus
U01-202	Üldkasutatav auditoorium	2
U02-301	Üldkasutatav auditoorium	3

Hoone
Õppehoone 1
Õppehoone 2

Lisateenused

Helivõimendus, Dataprojektor

Ürituse kirjeldus

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis cursus semper iaculis. Suspendisse laoreet, dui vitae facilisis pretium, nunc neque faucibus orci, sit amet hendrerit dui dui sit amet metus.

Kasutustingimused

Korraldaja

Rain Rähni (rarahn@taltech.ee)

Kommentaariid

Kommentaari

Manused



Manuse lisamiseks lohista failid siia või laadi arvutist

< Tagasi

Saada vastus

Tühista taotlus

Ruumi broneeringu taotluse detailvaate autori loodud info kuvamise osa.

Lisa 6 – Testimise ülesanded

Raamatukogu osa ülesanne 1 - Ruumi broneerimine:

Broneeri individuaaltööruum järgmiseks esmaspäevaks kell 10.00 - 11.00.

Raamatukogu osa ülesanne 2 - Ruumi broneeringu muutmine:

Muuda tehtud broneeringu kuupäev üks päev hilisemaks.

Taotluse osa ülesanne 1 - Küsimused varasema kogemuse kohta:

Kui sageli õnnestub teil taotlus esimesel katsel korrektselt esitada? (%) Kas olete kunagi loobunud taotluse esitamisest protsessi keerukuse tõttu? Milliste sammude juures taotlemisel tekib kõige rohkem probleeme või ebaõnnestumisi? Ligikaudu mitu minutit kulub teile tavaliselt taotlusprotsessis e-posti tegemiseks?

Taotluse osa ülesanne 2 - Ruumi broneeringu taotluse esitamine:

Esita ruumi U02-102 broneeringu taotlus.

Raamatukogu osa mõlema ülesande ja taotluse osa teise ülesande käigus vastas autor järgnevatele küsimustele:

Edukuse määr: Jah/Ei (Kas õnnestus ülesanne täita)

Aeg: (Mõõdeti sekundites alates ülesande algusest kuni kinnituse saamiseni)

Vigade arv: (Loendati valede valikute või klikkide arvu)

Lisa 7 – Testimise küsimused

"Ma kasutaksin seda süsteemi sageli."

- Tugevalt ei nõustu
- Pigem ei nõustu
- Neutraalne
- Pigem nõustun
- Tugevalt nõustun

"Ma leian, et süsteem on ebavajalikult keeruline."

- Tugevalt ei nõustu
- Pigem ei nõustu
- Neutraalne
- Pigem nõustun
- Tugevalt nõustun

"Ma leian, et süsteemi on lihtne kasutada."

- Tugevalt ei nõustu
- Pigem ei nõustu
- Neutraalne
- Pigem nõustun
- Tugevalt nõustun

"Ma vajaksin tehnilise inimese abi selle süsteemi kasutamiseks."

- Tugevalt ei nõustu
- Pigem ei nõustu
- Neutraalne
- Pigem nõustun
- Tugevalt nõustun

"Selle süsteemi funktsioonid on hästi integreeritud."

- Tugevalt ei nõustu
- Pigem ei nõustu
- Neutraalne
- Pigem nõustun
- Tugevalt nõustun

"Selles süsteemis on liiga palju ebakõlasid."

- Tugevalt ei nõustu
- Pigem ei nõustu
- Neutraalne
- Pigem nõustun
- Tugevalt nõustun

"Ma kujutan ette, et enamik inimesi õpiks seda süsteemi kiiresti kasutama."

- Tugevalt ei nõustu
- Pigem ei nõustu
- Neutraalne
- Pigem nõustun
- Tugevalt nõustun

"Ma leian, et süsteemi kasutamine on kohmakas."

- Tugevalt ei nõustu
- Pigem ei nõustu
- Neutraalne
- Pigem nõustun
- Tugevalt nõustun

"Mul on selle süsteemi kasutamisel enesekindel tunne."

- Tugevalt ei nõustu
- Pigem ei nõustu
- Neutraalne
- Pigem nõustun
- Tugevalt nõustun

"Ma pidin enne süsteemi kasutamist paju õppima."

- Tugevalt ei nõustu
- Pigem ei nõustu
- Neutraalne
- Pigem nõustun
- Tugevalt nõustun

"Kui lihtne on protsessi mõista?"

- Üldse mitte lihtne
- Väga raske
- Pigem raske
- Neutraalne
- Pigem lihtne
- Väga lihtne
- Täiesti lihtne

"Kui selge on, millist infot peate broneerimisel esitama?"

- Üldse ei ole selge
- Väga ei ole selge
- Pigem ei ole selge
- Neutraalne
- Pigem selge
- Väga selge
- Täiesti selge

"Kui rahul olete üldiselt kogu protsessiga?"

- Üldse mitte rahul
- Väga rahulolematu
- Pigem rahulolematu
- Neutraalne
- Pigem rahul
- Väga rahul
- Täiesti rahul

Lisa 8 – Autoriseerimislahenduse testimise tulemused

Autoriseerimislahenduse ruumi broneerimise ülesande tulemused.

Mõõdik	Vana lahendus	Uus lahendus	% Muutus
Aeg (sekundid)	66.6	73.2	-10%
Vigade arv	1	0.6	45%
Edukuse määr	100%	100%	0%

Autoriseerimislahenduse ruumi broneeringu muutmise ülesande tulemused.

Mõõdik	Vana lahendus	Uus lahendus	% Muutus
Aeg (sekundid)	54.2	45	-17%
Vigade arv	0.8	0.1	-82.4%
Edukuse määr	100%	100%	0%

Autoriseerimislahenduse SUS väidete tulemused.

SUS väide	Vana lahendus	Uus lahendus	% Muutus
Ma kasutaksin seda süsteemi sageli.	2.9	3.9	34.5%
Ma leian, et süsteem on ebavajalikult keeruline.	3.5	2.35	-32.9%
Ma leian, et süsteemi on lihtne kasutada.	2.9	4.25	46.6%
Ma vajaksin tehnilise inimese abile selle süsteemi kasutamiseks.	1.5	1.05	-30%
Selle süsteemi funktsioonid on hästi integreeritud.	3.05	4.1	34.4%
Selles süsteemis on liiga palju ebakõlasid.	2.55	1.55	-39.2%
Ma kujutan ette, et enamik inimesi õpiks seda süsteemi kiiresti kasutama.	3.5	4.45	27.1%
Ma leian, et süsteemi kasutamine on kohmakas.	3.65	2.2	-39.7%
Mul on selle süsteemi kasutamisel enesekindel tunne.	3	4.35	45%
Ma pidin enne süsteemi kasutamist palju õppima.	1.65	1.1	-33.3%

Autoriseerimislahenduse tajutava kasutuslihtsuse tulemused.

Väide	Vana lahendus	Uus lahendus	% Muutus
Kui lihtne on protsessi mõista?	4.45	5.95	33.7%
Kui selge on, millist infot peate broneerimisel esitama?	4.85	5.75	18.6%
Kui rahul olete üldiselt kogu protsessiga?	4.5	5.95	32.2%

Lisa 9 – Taotlussüsteemi testimise tulemused

Taotluse esitamise ülesande testimise tulemused.

Mõõdik	Vana lahendus	Uus lahendus	% Muutus
Aeg (minutid)	12.5	2.8	-78%
Vigade arv	2	0.8	62.5%
Edukuse määr	100%	100%	0%

Taotlussüsteemi SUS väidete tulemused.

SUS väide	Vana lahendus	Uus lahendus	% Muutus
Ma kasutaksin seda süsteemi sageli.	2.75	4.25	54.5%
Ma leian, et süsteem on ebavajalikult keeruline.	3.5	2.75	-21.4%
Ma leian, et süsteemi on lihtne kasutada.	2.75	4	45.5%
Ma vajaksin tehnilise inimese abile selle süsteemi kasutamiseks.	1.75	1	-42.9%
Selle süsteemi funktsioonid on hästi integreeritud.	2.25	4.25	88.9%
Selles süsteemis on liiga palju ebakõlasid.	3.5	2.5	-28.6%
Ma kujutan ette, et enamik inimesi õpiks seda süsteemi kiiresti kasutama.	3.25	4.5	38.5%
Ma leian, et süsteemi kasutamine on kohmakas.	3.75	2.75	-26.7%
Mul on selle süsteemi kasutamisel enesekindel tunne.	4.25	5	17.6%
Ma pidin enne süsteemi kasutamist palju õppima.	1.75	1.25	-28.6%

Taotlussüsteemi tajutava kasutuslihtsuse tulemused.

Väide	Vana lahendus	Uus lahendus	% Muutus
Kui lihtne on protsessi mõista?	3.5	6	71.4%
Kui selge on, millist infot peate broneerimisel esitama?	3.5	5.75	64.3%
Kui rahul olete üldiselt kogu protsessiga?	3.67	5.75	56.8%