

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Romet Tony Jõenurm 213290IADB

## **Treeningkeskuse rakendus naistele**

Bakalaureusetöö

Juhendaja: Meelis Antoi

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: Romet Tony Jõenurm

20.11.2024

## Annotatsioon

Käesoleva bakalaureusetöö eesmärgiks on luua kliendi poolt loodava treeningkeskuse jaoks veebirakendus, mis on mõeldud ainult naistele, et naised saaksid muretu ja turvalises keskkonnas treenida. Bakalaureusetöö käigus arendatakse veebirakendus kliendile, mis võimaldab naistel avada treeningkeskuse uks ainult naistele mõeldud treeningkeskuses.

Veebirakendus annab võimaluse avada nupu vajutusega treeningkeskuse uks, näha mitu inimest hetkeseisuga viibib treeningkeskuses, samuti on näha, kus asuvad treeningkeskused kaardi peal, mitu korda on inimene treeningkeskuses käinud ning hallata oma treeningpakkumisi. Veebirakenduses on võimalik ka näha, mis päeval on millised grupidreeningud ning ka registreerida ennast grupidreeningutesse. Veebirakenduses on kajastatud ka kasutajate edetabel, mis aitab motivatsiooni tõsta ning saab ülevaate enda treeningu külalistest.

Veebirakendus on täis-stack. Kogu veebirakendus on kirjutatud Java keeles Vaadin Flow raamistikuga, mis võimaldab kirjutada kogu äri- ja kasutajaliidese loogika ainult Java keeles, ilma et peaks kasutama JavaScripti, HTML-i või CSS-i ning andmed salvestatakse PostgreSQL andmebaasi. Rakendust majutatakse Microsoft Azure'i kasutajakeskkonnas.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 27 leheküljel, 6 peatükki, 10 joonist, 1 tabel.

## **Abstract**

### **Fitness Center App for Women**

The aim of this bachelor's thesis is to develop a web application for a training center created by the client, specifically designed for women, providing them with a worry-free and safe environment for exercising. The web application developed during the thesis will allow women to access the training center, which is exclusively for women.

The web application enables users to open the training center door with the click of a button, see how many people are currently present in the training center, locate training centers on a map, track the number of visits, and manage their training packages. Additionally, users can view group training schedules for specific days and register for group sessions. In most cases, to increase motivation and provide an overview, a ranking of visits to the training center has also been added.

The web application is full stack. It is entirely written in Java using the Vaadin Flow framework, enabling the development of business and user interface logic solely in Java, without the need for JavaScript, HTML, or CSS. Data is stored in a PostgreSQL database, and the application is hosted on the Microsoft Azure platform.

The thesis is written in Estonian and comprises 27 pages of content, 6 chapters, 10 figures and 1 table.

## Lühendite ja mõistete sõnastik

API	<i>Application Programming Interface</i> , rakenduse liides
CSS	<i>Cascading Style Sheets</i> , programmeerimiskeel rakenduse välisilme loomiseks
HTML	<i>HyperText Markup Language</i> , skriptikeel, mida veebibrauserid kasutavad veebilehtede renderdamiseks
JPA	<i>Java Persistence API</i> , liides, mis võimaldab rakendusel säilitada ja hankida teavet salvestussüsteemist
ORM	<i>Object-relational mapping</i> , viis programmeerimiskoodi joondamiseks andmebaasi struktuuridega
QR-code	<i>Quick response code</i> , kahemõõtmeline vötkoodi tüüp
REST	<i>Representational state transfer</i> , reeglite ja juhiste kogum selle kohta, kuidas peab koostama veebi API
SMTP	Simple Mail Transfer Protocol, meili saatmise protokoll
SPA	<i>Single Page Application</i> , üheleheküljeline rakendus
URL	Uniform Resource Locator, veebilehe aadress

# Sisukord

1 Sissejuhatus .....	10
2 Olemasolevad lahendused .....	11
2.1 GYM! .....	11
2.2 24-7 Fitness .....	11
2.3 MyFitness .....	12
2.4 Olemasolevate lahenduste analüüs ja kokkuvõte .....	12
3 Arendatava rakenduse analüüs.....	15
3.1 Metoodika.....	15
3.2 Nõuete koostamine .....	15
3.2.1 Funktsionaalsed nõuded .....	15
3.2.2 Mittefunktsionaalsed nõuded.....	18
3.3 Tehnoloogiate valik .....	18
3.3.1 Esirakendus.....	18
3.3.2 Tagarakendus .....	19
3.3.3 Rakenduse kasutajakeskkond .....	19
4 Rakenduse arendus .....	20
4.1 Andmebaasi koostamine .....	20
4.2 Tagarakenduse loomine .....	21
4.3 Veebilehe disain.....	22

4.3.1 Veebilehe põhivaated .....	23
4.4 Veebilehe ülesehitus .....	29
4.5 Rakenduse kasutajakeskkonda lisamine .....	30
5 Tulemused.....	32
5.1 Nõuete täitmine .....	32
5.2 Kliendi tagasiside .....	32
5.3 Edasiarenduse võimalused.....	34
6 Kokkuvõte .....	36
Kasutatud kirjandus .....	37
Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks .....	40
Lisa 2 – Rakenduse andmebaasi skeem.....	41
Lisa 3 – Kasutajavood süsteemi sisenemisel ja profiili vaatamisel.....	42

## Jooniste loetelu

Joonis 1. Veebilehe vaade .....	23
Joonis 2. Jõusaali sissepääsu ukse avamisvaade .....	24
Joonis 3. Grupitreeningute vaade .....	25
Joonis 4. Suhtluskanali vaade .....	26
Joonis 5. Jõusaalide asukohad kaardil .....	26
Joonis 6. Treenerite vaade .....	27
Joonis 7. Edetabeli vaade.....	28
Joonis 8. Kasutaja profiili vaade.....	28
Joonis 9. Administraatori kasutajate halduse vaade .....	29
Joonis 10. Vaadin Flow, Angular ja React võrdlus [27] .....	30



## **Tabelite loetelu**

Tabel 1. Loodava prototüübi ja olemasolevate lahenduste võrdlus.....	17
--	----

# 1 Sissejuhatus

Kaasaegses ühiskonnas on naistele suunatud spordirakenduste vajadus kasvamas, kuid paljud neist ei paku piisavat turvalisust ja mugavust treeningute ja jõusaalide kasutamisel. Käesoleva lõputöö eesmärk on välja töötada kliendile rakendus, mis on mõeldud spetsiaalselt naistele ja mis lahendab mitmeid puudujääke teiste jõusaalide rakenduste suhtes ning integreerida seda tulevikus loodava naiste jõusaaliga.

Hetkeseisuga pole loodud Eestis rakendust, mis annab võimaluse käia naistele mõeldud jõusaalis ning kus oleks turvaline ja mugav keskkond. Umbkaudselt 60% naistest ei käi seetõttu jõusaalis, et ei ole loodud naistele mõeldud rakendust, mis annaks võimaluse käia ainult naistele mõeldud jõusaalis, kuna nad ei soovi käia universaalses jõusaalis, sest meeste pilgud ja kommentaarid on rikkunud nende kogemuse [1].

Käesoleva lõputöö raames luuakse rakendus, mis võimaldab kasutajatel avada elektroonilisi jõusaali sisenemiseks, pakkudes turvalist ja privaatset keskkonda ainult naistele. Lisaks sellele saab rakenduse kaudu registreeruda gruppitreeningutele, näha kaardil jõusaalide asukohti, suhelda suhtluskanalis, näha treenereid, hallata oma profiili, osaleda kasutajate edetabelis ja reaajas jälgida, mitu inimest viibib hetkel jõusaalis. Rakenduses on ka funktsioonid, mis võimaldavad kasutajatel näha oma treeningpakette, jälgida päevaplaani ning saada kliendituge.

Lõputöö keskendub rakenduse väljatöötamisele ja selle funktsionaalsusele, pakkudes naistele sujuvat ja motiveerivat treeningkeskkonda. Lõputöö teises peatükis analüüsitakse kasutusel olevaid lahendusi. Kolmandas peatükis räägitakse arendatava rakenduse analüüsist ja milliseid programme kasutatakse rakenduse koostamisel. Neljandas peatükis räägitakse rakenduse arendusest. Viiendas peatükis antakse ülevaade tulemustest, et kas kliendi poolt seatud nõuded saadi täidetud ning millised on edasiarenduse võimalused pärast bakalaureusetööd.

## **2 Olemasolevad lahendused**

Järgnevalt uuritakse olemasolevaid lahendusi, mis töötavad sarnastel põhimõtetel, aga erinevad just sellepoolest, et olemasolevad on mõeldud universaalseks kasutamiseks. Eesmärk on koos kliendiga arutledes selgitada välja, mis on praegused puudujäägid ja kitsaskohad teistel sarnastel olemasolevatel rakendustel. Analüüsi tulemused annavad selge ettekujutuse sellest, millistele aspektidele tuleb rakenduse arendamisel tähelepanu pöörata ning millised on minimaalsed nõuded.

### **2.1 GYM!**

Gym! rakendus on universaalne kõigile kasutamiseks ja selle ainuke peamine põhimõte on, et saada avada metalluksi QR-koodiga. Kasutajad saavad vaadata enda treeningpaketti ja olemasolevaid jõusaale kaardil, mis on tehtud Gym! organisatsiooni poolt [2].

Rakendus ei vasta kliendi vajadustele, kuna esiteks nagu varasemalt mainitud, siis Gym! rakendus on universaalseks kasutuseks, aga kliendi vajadus on ainult naistele suunatud. Samuti on suureks puuduseks ka grupitreeningute nägemine ja registreerimine sinna. Kliendi sooviks on ka turvaline suhtluskanal, kus naised saavad üksteisega nõu pidada ja teineteisele abi pakkuda. Lisaks on ebamugav, et treenereid ei saa vaadata rakenduse kaudu, vaid seda tuleb teha eraldi veebilehel, mis pole kasutajasõbralik. Motivatsiooni tõstmiseks pole loodud ka kasutajate edetabelit, mis on üheks kliendi soovideks.

### **2.2 24-7 Fitness**

24-7 Fitness rakendus on samuti universaalne kõigile kasutamiseks, kuigi antud rakendusel on juba rohkem funktsioone. Nendeks funktsioonideks on metalluste avamine, grupitreeningute nägemine ja registreerumine, treeningpaketi nägemine ja selle haldus [3].

Kuigi see rakendus vastab osaliselt kliendi ootustele, jäävad endiselt probleemiks, et ei ole loodud rakendusel turvalist suhtluskanalit naiste vahel, kus nad saavad nõu küsida ja nõu anda. Teiseks ei ole tehtud rakendusesiseselt kaarti, kus oleks näha selle

organisatsiooni poolt tehtud jõusaale. Samuti on treenerite leidmine keeruline, kuna selleks tuleb taas kasutada eraldi veebilehte, mis vähendab kasutusmugavust. Motivatsiooni tõstmiseks pole loodud ka kasutajate edetabelit, mis on üheks kliendi soovideks.

### **2.3 MyFitness**

MyFitness rakendus on samuti universaalne kõigile kasutamiseks, kuigi antud rakendusel pole jõusaali avamisnuppu, vaid kohapeal on inimene, kes haldab jõusaali sissepääsu. Grupitreeningutesse broneerimise funktsionaalsus on väga hea, saab vaadata treeningute ajalugu, grupitreeningute ajad saab sünkroonida ära kasutaja telefoni kalendriga ning saab näha personaaltreenerid ja nendega ühendust võtta [4].

Selline rakendus ühtib osaliselt kliendi soovidega, aga jällegi on sellel rakendusel teatud puudujäägid. Kliendi soovidega ei ühti see, et jõusaali ust ei saa avada läbi rakenduse, mis juba toob juurde selle, et peab palkama kellegi kohapeale haldama sisse- ja väljakäimist. Teiseks ei ole näha kaardi peal, kus asuvad selle organisatsiooni poolt tehtud jõusaalid, mis vähendab kasutaja kasutusmugavust. Veel saab välja tuua selle, et rakenduse suureks probleemiks jääb ikka see, et ei ole loodud sellel rakendusel turvalist suhtluskanalit naiste vahel, kus nad saavad nõu küsida ja nõu anda. Motivatsiooni tõstmiseks pole loodud ka edetabelit, mis on üheks kliendi soovideks.

Antud rakendus ühtib kliendi soovidega selle poolest, et seal rakenduses saab näha grupitreeninguid ja registreerida ennast sinna. Rakenduses endas on välja toodud treenerid ja nende kontaktid, et nendega ühendust võtta.

### **2.4 Olemasolevate lahenduste analüüs ja kokkuvõte**

Olemasolevate jõusaali rakenduste analüüs tõi esile mitmeid olulisi punkte, mis aitavad mõista, kuidas olemasolevad lahendused toimivad ning milliseid puudujääke need rakendused sisaldavad. Analüüsitud jõusaali klubide rakendused pakuvad omavahel sarnaseid funktsionaalsusi, võimaldades hallata oma treeningpaketti, avada sissepääsu ust läbi rakenduse, näha treenerid ja nende kontaktandmeid, samuti on võimalik näha, kus asub soovidele vastav jõusaal, registreerida ja näha grupitreeninguid. Kõiki rakendusi

saab kasutada mobiilirakenduses ja veebirakenduses, veebirakenduses ei saa ainult sissepääsu ust avada ja selleks on vaja laadida alla mobiilirakendus. Iga rakenduse lahendus on oma eripäradega, mis erinevad kliendi soovidest ja vajadustest.

Gym! rakendus on lihtne ja universaalne lahendus, mille peamine funktsioon on pakkuda kasutajatele võimalust avada sissepääsu ust jõusaali QR-koodi abil. Rakendus täidab põhifunktsioone hästi, aga puuduvad olulised funktsioonid nagu gruppitreeningute registreerimine, kasutajate vaheline suhtluskanal, treenerite ülevaade, kasutajate edetabeli vaade ja kõige tähtsam on just see, et see pole suunatud ainult naistele. Kliendi jaoks on oluline see, et rakenduses oleks võimalus luua naiste omavaheline suhtlus, mis aitab kaasa turvatunde loomisele ja motiveerib rohkem suhtlema samade huvidega kasutajate vahel.

24-7 Fitness rakendus on küll funktsionaalsuse poolest mitmekülgsem just sellepoolest, et pakub gruppitreeningute registreerimist ja nende nägemist, kuid siin jääb samuti puudu naistele suunatud turvalisest suhtlusvõimalusest. Turvaline suhtlusplatvorm on kliendi jaoks üks olulisemaid vajadusi, kuna see toetab kogukonna loomist ja annab naistele võimaluse oma kogemusi vabalt jagada. Samuti pole olemas rakenduses kaardi funktsiooni, kust oleks näha olemasolevaid jõusaale, sest jõusaali leidmine nõuab täiendavat otsingut, mis on ajakulukas ning samuti vähendab kasutamismugavust. Rakenduses puudub ka võimalus treenerite valimiseks, mis on samuti vajalik funktsionaalsus, kuna on keeruline käia kahe rakenduse vahel ning see on ajakulukas ja ebamugav. Rakendusel puudub ka edetabeli vaade, mis võiks tõsta motivatsiooni kasutajates.

MyFitness rakendus küll pakkus gruppitreeningute nägemist ja registreerimist, sünkronimist kasutaja kalendriga, diagrammide ja jooniste loomist ja treenerite kontaktide vaatamist, kuid see ei pakkunud sissepääsu uste avamist, mis on ülimalt vajalik kliendi soovitud rakenduses. See tähendaks, et igapäevane sissepääsu haldus nõuab täiendavat personali, mis suurendab nii halduskulu kui ka tööjõukulu. Lisaks rakenduses puudub ka kaardi nägemise funktsionaalsus, mis teeb jõusaalide leidmise keerulisemaks. Kliendi jaoks jääb probleemiks veel suhtluskanali puudumine, mis on vajalik funktsionaalsus tema soovitud rakenduses. Rakendusel puudub ka edetabeli vaade, mis võiks tõsta motivatsiooni kasutajates.

Naiste jõusaali rakenduse arendamisel on vaja pöörata tähelepanu naiste turvalisuse, privaatsuse ja kogukonna loomise aspektidele, mis olemasolevates lahendustes on alahinnatud. Olemasolevad lahendused täidavad teatud põhifunktsioone hästi, aga need ei täida kõiki kliendi soove.

Peamisteks funktsionaalseteks nõueteks on naistele mõeldud turvaline suhtluskanal, treenerite valimise lihtsus, kaardi olemasolu, kus on märgitud organisatsiooni poolt loodud jõusaalid, mis on mõeldud ainult naistele, sissepääsu automatiseerimine ja edetabeliga liitumine soovi korral, et tõsta kasutajates motivatsiooni.

Kokkuvõttes näitavad olemasolevad lahendused, et naistele suunatud jõusaalide puhul on vajalik arvestada rohkemate spetsiifiliste funktsioonidega, mis suurendavad nii kasutusmugavust kui ka turvalisust. Kliendi jaoks on oluline, et loodav rakendus pakuks terviklikumat ja mugavamat lahendust, mis ühtlasi arvestab kõikide nende funktsionaalsete puudujääkidega, mida konkurentsivõimelised lahendused ei paku. Tulevikus on oluline, et rakendus suudaks vastata kõikidele eeltoodud vajadustele ning samas pakkuda kasutajatele personaliseeritud ja hõlpsasti kasutatavat keskkonda, mis eristab seda universaalsetest jõusaali rakendustest.

## **3 Arendatava rakenduse analüüs**

Selles osas antakse ülevaade arendatava rakenduse analüüsist ning selle funktsionaalsetest ja mittefunktsionaalsetest nõuetest, samuti valitakse tehnoloogiad, mis sobivad prototüübi loomiseks ja rakenduse edasiarendamiseks.

### **3.1 Metoodika**

Bakalaureusetöö alguses kaardistatakse kliendiga probleem, mis vajab lahendamist, milleks on naistele suunatud jõusaali rakenduse loomine, mis arvestab turvalisuse, privaatsuse ja lihtsa kasutusmugavuse nõuetega. Selleks viiakse läbi olemasolevate lahenduste analüüs, tuues välja nii nende tugevad küljed kui ka puudused. Seejärel pannakse koos kliendiga paika prototüübi olulised funktsionaalsed ja mittefunktsionaalsed nõuded. Prototüübi testimine toimub valitud sihtrühmaga, et saada tagasisidet ja teha vajalikke parandusi rakenduse edasiseks arendamiseks. Kuna rakenduse täiustamine jätkub ka pärast bakalaureusetöö lõppu, võetakse arvesse ka tulevikus vajalike funktsioonide lisamise võimalused.

### **3.2 Nõuete koostamine**

Nõuete väljatöötamisel lähtutakse kliendi ja lõppkasutajate vajadustest, kes on naistele mõeldud jõusaali kasutajad. Olemasolevate lahenduste analüüsi põhjal määratakse kindlaks kõige olulisemad funktsionaalsed ja mittefunktsionaalsed nõuded, mis tagavad rakenduse lihtsa ja turvalise kasutamise ning kaitsevad privaatsust. Rakenduse põhifunktsioonid ja tegevused pannakse paika koostöös kliendiga.

#### **3.2.1 Funktsionaalsed nõuded**

Funktsionaalsed nõuded määratlevad, mida rakendus peab suutma teha, et rahuldada kasutajate vajadusi ja tagada soovitud teenused [5]. Rakendusel on kolm peamist kasutajarolli: administraator (jõusaali haldaja), treener ja tavaline kasutaja (jõusaali külastaja). Iga rolli jaoks on määratud spetsiifilised õigused ja funktsionaalsused, nagu näiteks juurdepääs kasutajate haldusele, gruppitreeningute loomisele ja jõusaali

funktsioonide haldamisele. Funktsionaalsed nõuded hõlmavad tegevusi, mis tagavad, et süsteem täidab oma põhiülesandeid ja pakub kasutajatele vajalikku teenust vastavalt nende rollidele (vt Lisa 3).

Administraatorile määratud õigused:

- Võimaldab hallata jõusaali avamise ja sulgemise süsteemi rakenduse kaudu.
- Näeb reaalajas infot jõusaali kasutajate arvu kohta.
- Saab lisada, muuta ja hallata grupitreenereid, treeninguid ja nende graafikuid.
- Saab jälgida treeningutes osalejate statistikat.
- Saab vaadata ja hallata kasutajate informatsiooni.
- Saab värskendada kasutajate jõusaali kasutust.
- Saab muuta üldist suhtluskanalit.

Treenerile määratud õigused:

- Saab lisada, muuta treeneri poolt lisatud grupitreeningut.
- Näeb ja registreerib ennast grupitreeningutele.
- Näeb kaarti ja kaardil asuvaid jõusaale.
- Saab kontakteeruda administraatoritega.
- Saab avada jõusaali uksi läbi rakenduse.
- Saab vaadata ja hallata enda treeningpaketti.
- Saab registreerida ennast ja vaadata edetabelit
- Näeb ja saab kontakteeruda treeneritega

Tavalise kasutaja õigused

- Saab avada jõusaalide uksi läbi rakenduse.
- Näeb ja registreerib ennast grupitreeningutele.
- Saab vaadata ja hallata enda treeningpaketti.
- Näeb kaarti ja kaardil asuvaid jõusaale.
- Saab esitada küsimusi ja saada abi treeneritelt ning administraatoritelt.
- Saab suhelda teiste jõusaali liikmetega.



- Saab registreerida ennast ja vaadata kasutajate edetabelit
- Näeb ja saab kontakteeruda treeneritega

Järgmiselt võrreldakse loodava prototüübi ja olemasolevate lahenduste funktsionaalsusi ja selle kohta on tehtud tabel (vt. Tabel 1).

**Tabel 1. Loodava prototüübi ja olemasolevate lahenduste võrdlus**

<b>Funktsionaalsus</b>	<b>Loodav prototüüp</b>	<b>Gym!</b>	<b>24/7 fitness</b>	<b>MyFitness</b>
Kaardi ülevaade	OK	OK	NOK	NOK
Saab avada sissepääsu ust treeningkeskusesse	OK	OK	OK	NOK
Edetabeli nägemine	OK	NOK	NOK	NOK
Grupitreeningute nägemine ja registreerimine	OK	NOK	OK	OK
Profiili haldus	OK	OK	OK	OK
Suhtluskanal	OK	NOK	NOK	NOK
Treenerite nägemine ja nende kontaktandmed	OK	NOK	NOK	OK
Mobiilirakendus	NOK	OK	OK	OK

### 3.2.2 Mittefunktsionaalsed nõuded

Mittefunktsionaalsed nõuded kirjeldavad rakenduse omadusi ja seda, kuidas süsteem peaks täitma funktsionaalseid nõudeid, tagades rakenduse töökindluse ja kasutajamugavuse [5]. Need on olulised, et süsteem toimiks sujuvalt ning pakuks kasutajatele positiivset ja mugavat kogemust. Arendamiseks seatud nõuded on järgmised:

- Kasutajaliides peab olema lihtsasti mõistetav ja mugav
- Rakendust peab saama tulevikus laiendada uute funktsionaalsustega
- Rakendus peab olema mõeldud ainult naistele kasutamiseks

### 3.3 Tehnoloogiate valik

Tehnoloogiate valikut arvestatakse just sellega, et silmas tuleb pidada kliendi nõudmisi, arendajate kogemusi, veebiarenduse parimaid praktikaid ja kuluefektiivsust [6]. Antud rakenduse arendamisel ei kehtestatud kliendi poolt täiendavaid tehnoloogilisi piiranguid, mistõttu valiku tegemisel keskendutakse autori varasematele kogemustele ning tuttavatele programmeerimiskeeltele. Valiku aluseks on usaldusväärsus, skaleeritavus ja arendaja kompetents, et tagada rakenduse edukas ja tõhus toimimine tulevikus.

#### 3.3.1 Esirakendus

Esirakenduse moodustamiseks valiti Java keel Vaadin Flow raamistikuga, kuna autoril on varasemalt olnud kõige rohkem kogemust just Java keeles. Vaadin flow on Java põhine raamistik, mis võimaldab arendajatel kirjutada veebirakendusi täielikult Java keeles, ilma et oleks vaja kirjutada JavaScripti või HTML-i koodi [7].

Vaadin Flow abil on arendajal võimalus luua komponente erinevateks rakenduse osadeks, mis muudab rakenduse struktuuri läbipaistvamaks ja koodi kergemini hallatavaks ning taaskasutatavaks. Vaadin Flow toetab ka täiendavaid disainivõimalusi, kasutades CSS-i ja teeki, mis aitavad veebirakendusele lisada funktsionaalsust ja kujunduselemente [8].

Kuna Vaadin Flow raamistik põhineb Java keelel, tagab see tugeva tüübituvastuse ja lihtsustab vigade leidmist ning koodi hooldamist. Samuti võimaldab raamistik hõlpsalt lisada uusi komponente ja funktsioone, muutes rakenduse tulevikus laiendatavaks [8].

### 3.3.2 Tagarakendus

Tagarakenduse loomiseks valiti Spring Boot raamistik, mis võimaldab arendada veebirakendusi Java programmeerimiskeeles. Spring Boot pakub Java arendajatele tööriistakomplekti, mis lihtsustab tagarakenduse seadistamist ja arendamist, aidates keskenduda äritarkvara arendusele ilma keeruliste konfiguratsioonideta ja raamistik toetab ka moodulitest koosnevat ülesehitust, mis muudab koodi struktuuri hallatavaks ja taaskasutatavaks [9].

Andmebaasi tehnoloogiaks valiti PostgreSQL. PostgreSQL on avatud lähtekoodiga andmebaasikeel, mis on laienenud SQL ehk *Structured Query Language* andmebaasikeelelt paljude uute funktsionaalsustega ja mõeldud kõrge koormusega rakendustele ning pakub tuge keerukatele päringutele [10]. Tänu PostgreSQL ja Spring Booti tugevale ühilduvusele saab andmeedastust rakenduse ja andmebaasi vahel siduda kiiresti ja tõhusalt, kasutades sobivaid andmebaasi juurdepääsuteeke [11].

### 3.3.3 Rakenduse kasutajakeskkond

Esi- ja tagarakenduse majutamiseks valiti Microsoft Azure pilveteenuse platvorm. Azure pakub mitmekesist valikut pilveteenuseid, sealhulgas arvutus-, andmebaasi-, arendus- ja haldusteenuseid, võimaldades majutada ja hallata nii esi- kui ka tagarakendusi ühes kohas. See platvorm on laialt kättesaadav ja vastab kõrgetele turvanõuetele, toetudes andmekeskustele üle maailma, mis pakuvad varundust geograafiliselt erinevates piirkondades ning automaatset tõrkekindlust, vähendades riski, et rakendus oleks mõjutatud riistvararikke või looduskatastroofi korral [12][13].

Azure'i paindlik ressurside haldus võimaldab kulusid kokku hoida sellisel moel nagu kasutatavaid ressursse saab jooksvalt muuta vastavalt sellele, kui palju inimesi rakendust kasutab. Samuti lubab Azure'i "*Pay As You Go*" hinnamudel maksta vaid tegeliku kasutuse eest ja töötundide eest, mis muudab selle lahenduse tõhusaks ja taskukohaseks [12][14].

## 4 Rakenduse arendus

Selles peatükis antakse põhjalik ülevaade rakenduse arendusprotsessist, keskendudes tagarakenduse ja esirakenduse loomisele, veebidisaini kavandamisele ning kasutajaliidese funktsioonide rakendamisele.

### 4.1 Andmebaasi koostamine

Rakenduse andmebaas koosneb 13-st tabelist (vt. Lisa 1), mis toetavad rakenduse erinevaid funktsionaalsusi. Kõik tabelid on omavahel seotud, et tagada andmete sidusus ja sujuv kasutajakogemus. Andmebaasi haldamiseks ja andmete tõhusaks liikumiseks on kasutusel PostgreSQL.

Tabelites *User*, *Role* ja *UserRole* hoitakse kasutajate infot ja rollide määratlust. *User* tabel sisaldab kasutaja identiteeti ja kontaktandmeid nagu näiteks kasutajanimi, parooli räsi ja isiklikud andmed. Lisaks sellele kasutatakse *Role* ja *UserRole* tabeleid, et määrata kasutajatele rollid, mis võimaldavad rakenduses erinevaid juurdepääsu tasemeid.

Tabel *Membership* on tabel, kus hoitakse treeningpakette ning see hoiab endas tunnuskoodi, hinda ja kehtivusperioodi. Kasutajad saavad olla seotud kindla jõusaaliga läbi tabeli *UserGym*, mis sidustab kasutaja ja jõusaali. *Gym* tabel sisaldab infot jõusaali kohta nagu näiteks asukoht ja kui palju inimesi viibib hetkel selles jõusaalis.

Tabel *Message* on tabel, kus hoitakse suhtluskanalist tulenevat teksti ning see hoiab endas sõnumit, kuupäeva. Kasutajad saavad olla seotud erinevate sõnumitega ja kuna gruppis võib olla rohkem kui üks inimene, siis on tehtud juurde ka *UserInGroup* ja see on ühendatud *Group* tabeliga, mis omakorda on seotud *Message* tabeliga, et salvestada ära, millises gruppis on millised sõnumid ja millise *User* poolt.

Grupitreeningute info ja ajakava info on näidatud tabelites *Schedule* ja *UserSchedule*. *Schedule* tabelis hoitakse treeningute infot, sealhulgas treeningu kirjeldust, algus- ja lõppkellaega ning seotud jõusaali. *UserSchedule* tabel võimaldab kasutajatel registreerida end grupitreeningutele ja näha oma isiklikku ajakava.

Kasutajate aktiivsuse ja saavutuste jälgimiseks kasutatakse *Leaderboard* ja *UserLeaderboard* tabelleid. Samuti tehti need tabelid, et kasutajate motivatsiooni tõsta. *Leaderboard* tabel sisaldab tabeli nime, punktisüsteemi, algus- ja lõppkuupäeva. *UserLeaderboard* tabelis on info selle kohta, millal kasutaja liitus konkreetse edetabeliga ning tema saavutused.

## 4.2 Tagarakenduse loomine

Lõputöö tagarakenduse arendamiseks kasutas autor Java Spring Boot raamistikku, ühendades selle Vaadin Flow raamistiku funktsionaalsustega. See lahendus eraldas selgelt kasutajaliidese ja äri loogika kihid, võimaldades lihtsat ja loogilist struktuuri, kus iga komponent täidab kindlat ülesannet. Spring Boot toetas vajalikke teenuseid nagu autentimine ja autoriseerimine, andmete loomist ning pakkus paindlikku ja kiiret arenduskeskkonda. Selline lahendus toetas sujuvat integratsiooni, aidates arendusel keskenduda ainult äri- ja kasutajaliidese loogika eraldamisele ning kiirendades arendusprotsessi [15].

Turvalisuse jaoks kasutati Spring Security teeki, mis tagab autentimis- ja autoriseerimisfunktsionaalsuse. Loodud *AuthenticationManagerProvider* klass ühendati *UserDetailsService*-iga, mis autentimisprotsessis laeb kasutajaandmed ning valideerib parooli ja õigused. *AuthenticationManagerProvider* on peamine Spring Security liides, see haldab autentimisprotsessi olulisi funktsioone [16]. *UserDetailsService*'i kasutatakse, et saada autentimisel kasutaja parool, kasutajanimi ja teised atribuudid [17]. Paroolide krüpteerimiseks valiti *BCryptPasswordEncoder*, mis tagab, et paroolid salvestatakse turvaliselt räside kujul [18]. Kasutaja sisselogimisvaade ja autentimisprotsess on realiseeritud *VaadinWebSecurity* abil, mis integreerib Vaadini kasutajaliidese Spring Security lahendusega [19]. Lisaks sisaldab rakendus kohandatud veateateid, mis teavitavad kasutajaid e-posti kinnitamise vajadusest enne sisselogimist.

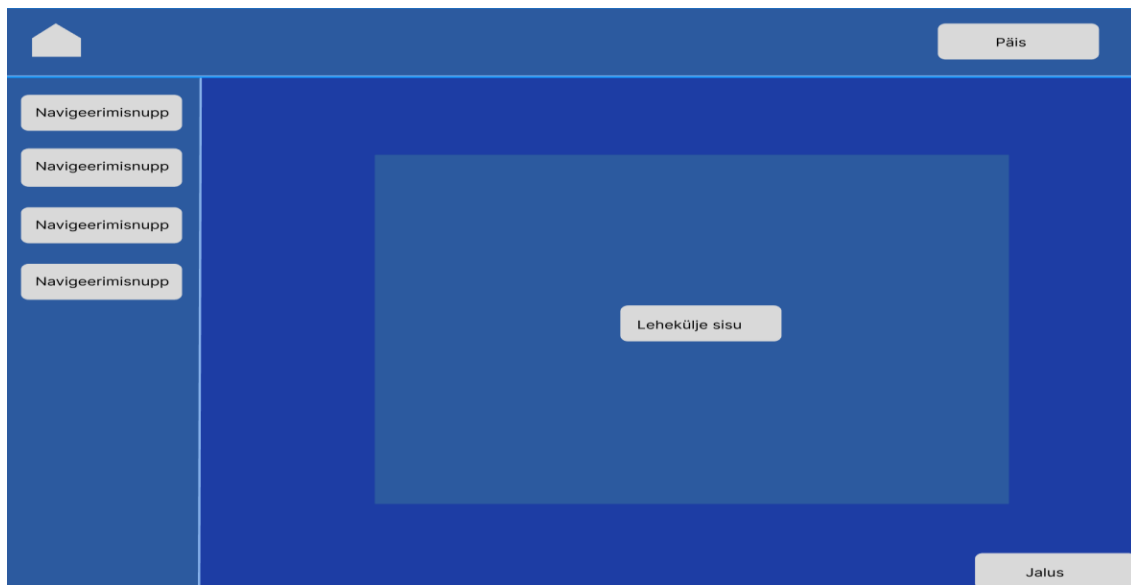
E-postiteenuse integreerimiseks kasutatakse Mailtrap'i, mis on konfigureeritud SMTP ehk *Simple Mail Transfer Protocol* protokollu kaudu ja seadistatud *spring.mail.host*, *username* ja *password* parameetrite abil. MailTrap on teenus, mida saab kasutada e-kirjade püüdmiseks, ilma et kiri päriselt kohale läheb saadetud aadressile [20]. SMTP on

internetiprotokoll, mida kasutatakse e-posti sõnumite edastamiseks arvutivõrkude vahel [21]. Kui uus kasutaja registreerib endale konto, siis registreerimisel saadetakse neile e-posti kaudu kinnituskood. Selleks kasutati *JavaMailSender* komponenti, mis saadab kasutaja e-posti aadressile kinnituskoodi, millega hiljem siis kinnitada enda e-post. Kui kasutaja kinnitab oma konto, uuendatakse andmebaasis tema profiili staatust, et võimaldada edasine ligipääs rakendusele. *JavaMailSender* on e-kirjade saatmiseks kasutatav liides [22].

Tagarakenduses kasutati JPA ehk *Java Persistence API* ja Hibernate ORM-i, mis võimaldas sujuvat andmebaasi integreerimist. JPA on objektide relatsioonilise kaardistamise raamistik, mis võimaldab Java objekte relatsiooniandmebaasi tabelisse lisada [23]. Klassidega nagu näiteks *User*, määrati andmebaasitabelite skeem ja sellega unikaalsed väljad nagu *username*, *personalCode*, email jne. Iga entiteet on omaette Java klass, mis vastab kindlale tabelile andmebaasis, pakkudes struktureeritud andmehoidu. Kuna projekti kasutajaliides on loodud Vaadin Flow abil, siis suhtleb esirakendus otse tagarakendusega. See tähendab, et kasutajate andmete kuvamine, kustutamine ja uuendamine toimub otse Spring Boot teenustes, ilma REST API-de vajaduseta. REST API ehk *representational state transfer application programming interface* on reeglite ja juhiste kogum selle kohta, kuidas peaks koostama veebi API [24].

### 4.3 Veebilehe disain

Kuna kliendil ei olnud veebilehe kujunduse osas kindlaid nõudmisi, viidi läbi olemasolevate platvormide analüüs, et määrata sobiv disain antud rakendusele. Eesmärgiks oli luua veebileht, mis oleks intuiitiivne ja kasutajasõbralik, pakkudes kasutajatele kiiret ligipääsu vajalikule informatsioonile. Veebilehe struktuur jaguneb mitmeks osaks: staatilised elemendid, sisu, mis kohandub vastavalt kasutaja rollile ja põhiosa, mis kuvab lehekülje konkreetset sisu. Staatilisteks elementideks on päis ja jalus. Jalus sisaldab kontaktandmeid ja päises on sisse- ja väljalogimisnupp, kasutaja nimi, roll ja link profiilile, kus juba saab profiili hallata (vt. Joonis 1).



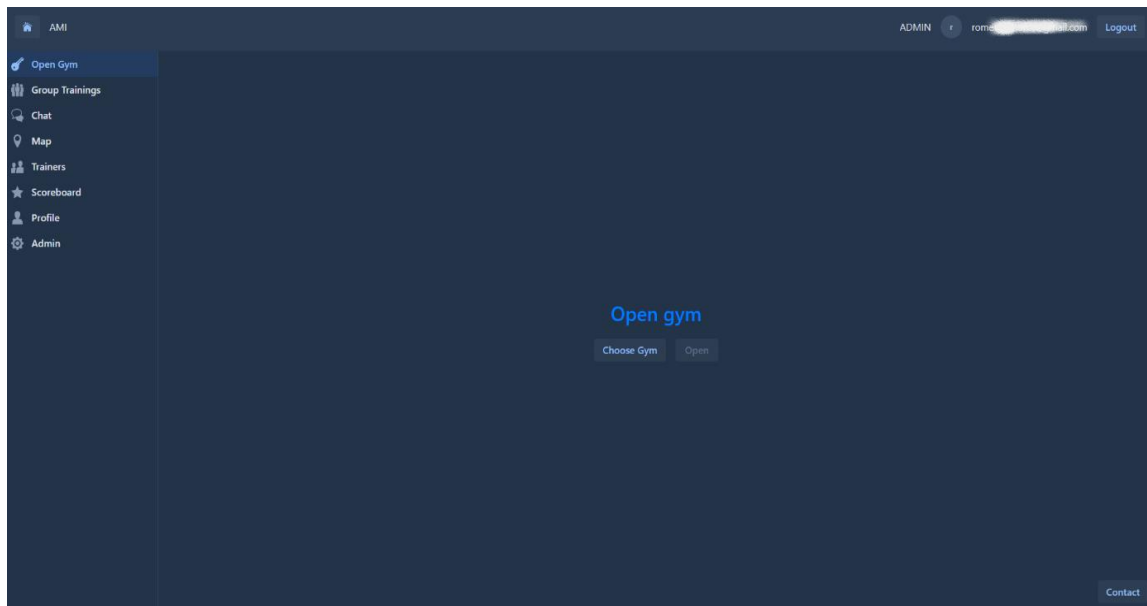
Joonis 1. Veebilehe vaade

Navigeerimisnupud muutuvad vastavalt rollile, kes on sisse loginud. See teeb kindlaks selle, et vastavad rollid saavad teha neile kindlaks tehtud tegevusi. Administraator näeb kõiki nuppe ja saab lehel teha kõiki muudatusi, treener ei näe ainult kasutajate haldusnuppu, tavaline kasutaja näeb sama palju kui treener, aga võimalused vaadetel on piiratud nagu näiteks ei saa tavaline kasutaja luua gruppitreeningut. Inimene, kes ei ole sisse logitud, näeb ainult kaardi vaadet ja treenerite vaadet. Vaated muutuvad ka vastavalt ekraani suurusele.

#### 4.3.1 Veebilehe põhivaated

Jõusaali avamise vaade on mõeldud jõusaali sissepääsu ukse avamiseks. Sinna sisse on ehitatud kontrollmehhanism, mis kontrollib, kas kasutaja on ostnud endale treeningpaketi ja kui on, siis vaadatakse, kas see veel kehtib tänase kuupäevaga. Kui nendest kontroll läbi ei lähe, siis selline vaade nagu vali jõusaal ja ava, ei ole nähtaval. Nähtaval on kiri, et treeningpakett on aegunud ja tuleks osta uus. Sellele järgneb kohe nupp, mida vajutades leiavad nad ennast treeningpakettide lehel, kus nad saavad valida juba endale sobiva paketi. Pärast paketi valimist suunatakse kasutaja maksmislehele, kus nad tasuvad ära treeningpaketi summa ja pärast seda kui kõik on õnnestunud, tuleb välja jälle vaade, mida on näha järgneval pildil (vt Joonis 2). Avamisnupp on vaikimisi hõivatud, aga pärast jõusaali valimist saab kasutada nuppu "Ava jõusaali uks". Pärast jõusaali avamist kaob eest praegune vaade ja asendub vaatega, mis ütleb, et rohkem ei saa jõusaali ust avada ja

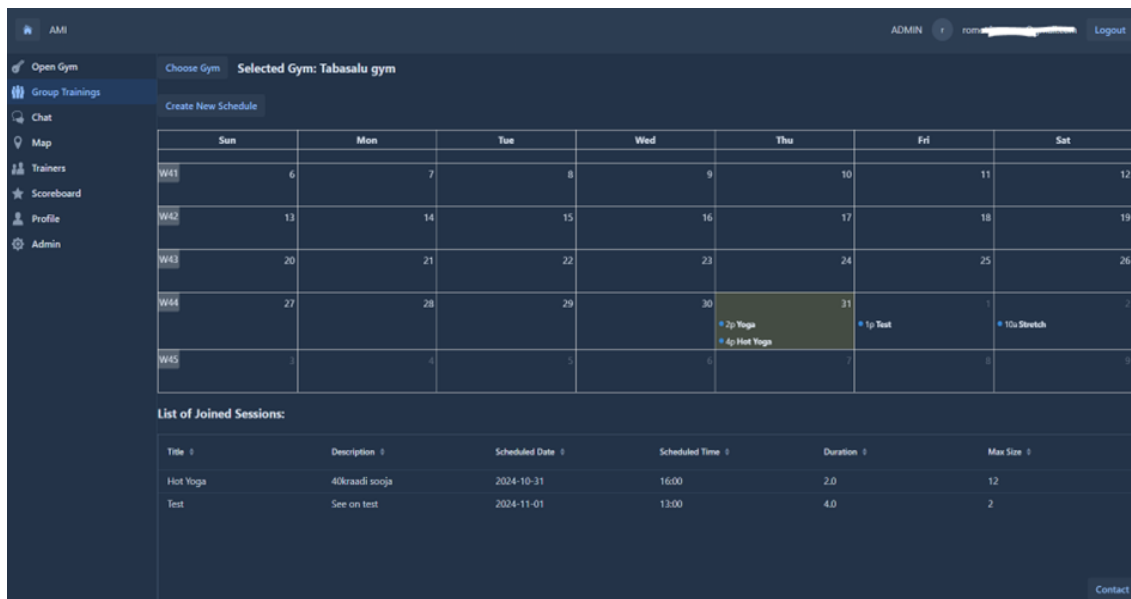
uuesti saab avada peale 00:00. Programmi on sisse ehitatud funktsioon, mis värskendab kõikide kasutajate avamisnupu ja peale 00:00 saavad nad seda uuesti kasutada.



Joonis 2. Jõusaali sissepääsu ukse avamisvaade

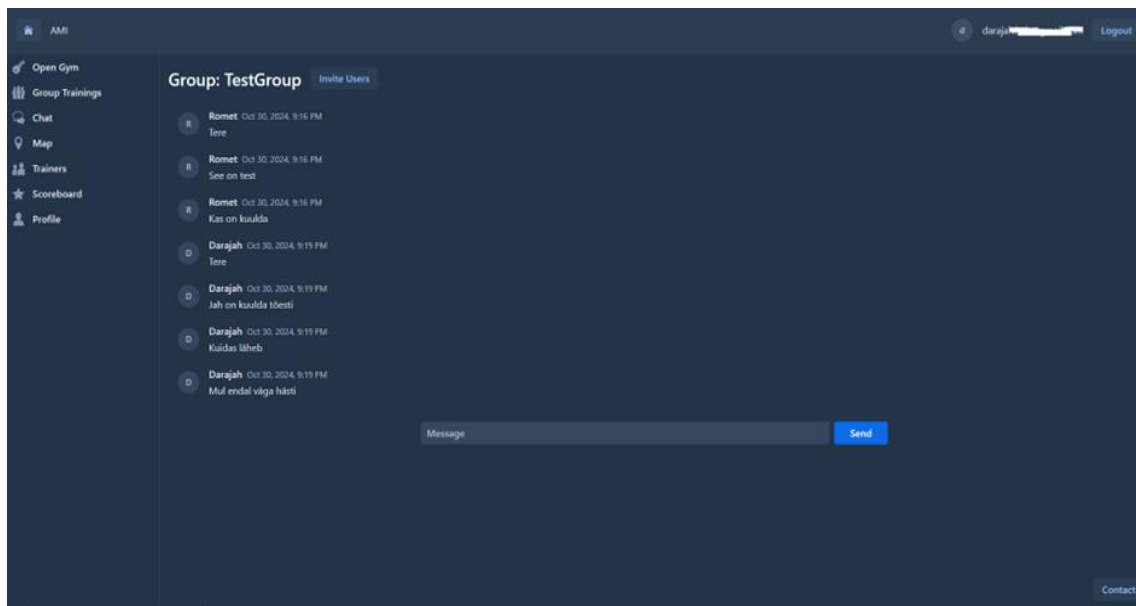
Grupitreeningute lehekülj annab ülevaate treeningutest. Kõigepealt, et näha üldse treeninguid, tuleb valida jõusaal, mille treeninguid sa soovid näha. Pärast jõusaali avamist näidatakse kalendris, mis päeval ja mis kell on treening. Treeningu peale vajutades avaneb hüpikaken, kus on näha treeningu informatsiooni ja sealt juba saab ühineda soovitud treeninguga. Kui treening on täis ehk maksimum arv inimesi on treeninguga juba liitunud, siis ühinemisnuppu ei saa vajutada. Kui roll on treener või administraator, siis saab luua uusi treeninguid ja neid kustutada, kui kalendris nende peale vajutada. Treener ja administraator saavad ise ka ühineda teiste treeningutega. Tavaline kasutaja ei näe nuppu, millega saab luua uut treeningut. Kalendri all näeb treeninguid, millega kasutaja on liitunud ja kui kasutaja tahab treeningust lahkuda, siis peab vajutama treeningu peale ja peale seda avaneb hüpikaken, kus saab lahkuda treeningust (vt Joonis 3). Treeningud, mis on juba toimunud, kustutatakse kalendrist automaatselt ja sellega kaob ka treening järjekorrast, millega kasutaja on liitunud.





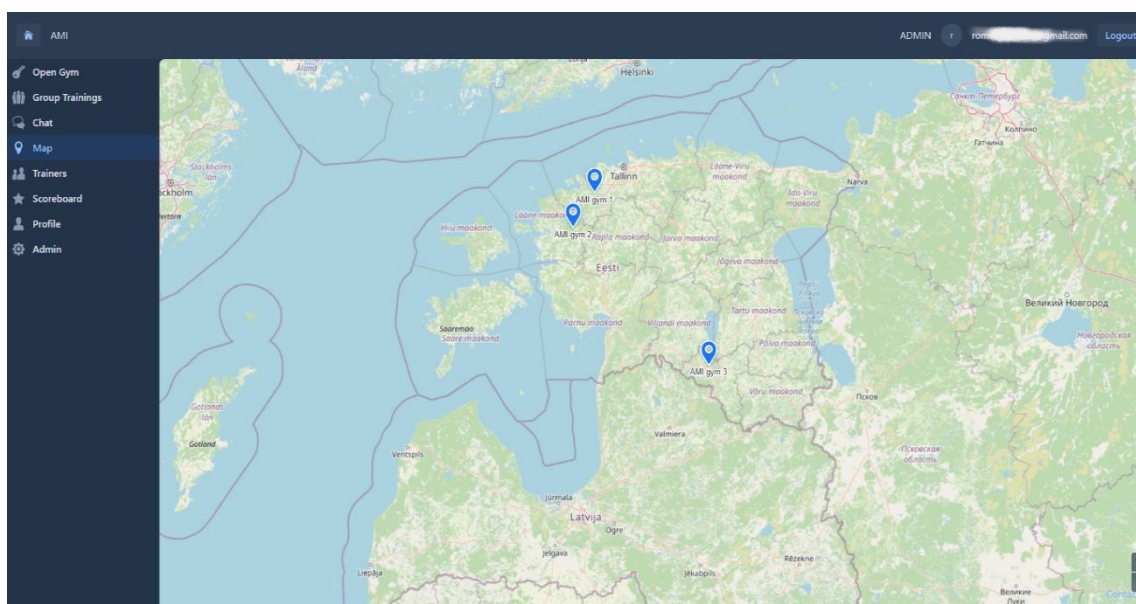
Joonis 3. Grupitreeningute vaade

Suhtluskanali vaates saab suhelda teiste kasutajatega. Kui vajutada suhtluskanali navigeerimisnupu peale, siis kõigepealt ilmuvad ette grupid, mis on kasutaja teinud või millega ta liitunud on. Olemas on juba manuaalselt süsteemi poolt loodud mõned grupid, kuhu on lisatud kõik kasutajad ning uued kasutajad lisatakse sinna automaatselt. Need grupid on teiste aitamiseks ning treenerid ja administraatorid saavad grupi liikmeid hallata. Iga kasutaja saab ka teha enda grupi ja lisada sinna liikmeid, et nendega suhelda. Kui vajutada kasutaja lisamise peale, siis avaneb hüpikaken, kus on vaja kirjutada sisse teise kasutaja e-posti aadress ja siis vajutada "Kutsu" nuppu. Teisele kasutajale seejärel ilmub gruppidesse uus grupp ja peale vajutades saab seal kirjutada (vt Joonis 4). Kui kasutaja vajutab teise kasutaja mullikese peale, siis näeb ta teise kasutaja tutvustust.



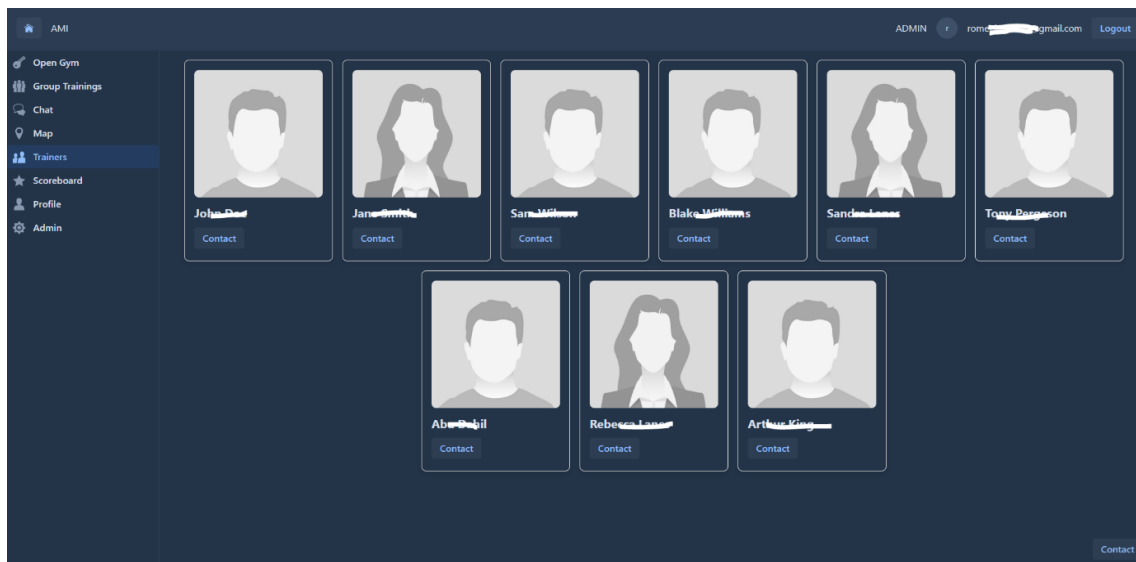
Joonis 4. Suhtluskanali vaade

Kaardivaade on lehekülj, kus saab vaadata maailma kaarti ja sealt omakorda näha, kus asuvad jõusaalid, mis on tehtud selle organisatsiooni poolt. Hetkeseisuga kaardil olevad asukohad ei vasta tõele, kuna veel ei ole jõusaale loodud ja see on ainult testimiseks ja näitamiseks, et kõik töötab nii nagu vaja (vt Joonis 5). Kaart on interaktiivne, seda saab välja suumida ja sisse suumida ja kui vajutada markeri peale, siis avaneb hüpikaken, kus näitab inimeste arvu hetkel jõusaalis ja asukoha täpset aadressi.



Joonis 5. Jõusaalide asukohad kaardil

Treenerite lehekülg on vaade, kus saab vaadata treenerid ja nendega kontakteeruda (vt Joonis 6). Vajutades treeneri kontakti nupu peale, avaneb hüpinkaken, kus on näha treeneri telefoninumbrit ja e-posti aadressi. Treenerid on manuaalselt süsteemi kaudu lisatud ja muudmoodi neid lisada ei saa.



Joonis 6. Treenerite vaade

Edetabeli lehekülg on vaade, kus saab näha võistlusel osalejaid ja näha nende kohta edetabelis (vt Joonis 7). Koheselt ei lisata kasutajat edetabelisse, vaid seda tuleb teha manuaalselt nupule vajutades. Edetabelis järjestatakse esimesed 20 kasutajat külastamise arvu järgi. Edetabeli eesmärk on luua motivatsiooni kasutajatel, kes on võistlushimulised ja esimesele kolmele positsioonile on loodud ka vastav auhind. Kui kasutaja on ühinenud edetabeliga, siis on tal ka võimalus loobuda edetabelist. Edetabeli nupu peale vajutades avaneb hüpinkaken, kus küsib uuesti, et kas kindlalt on soov lahkuda. Edetabeli sisu värskendatakse sellega, et kasutaja külaskäigu arv asendatakse nulliga iga 3 kuu tagant ja sellise süsteemiga on ka uutel tulijatel võimalik edetabelis heale kohale sattuda.

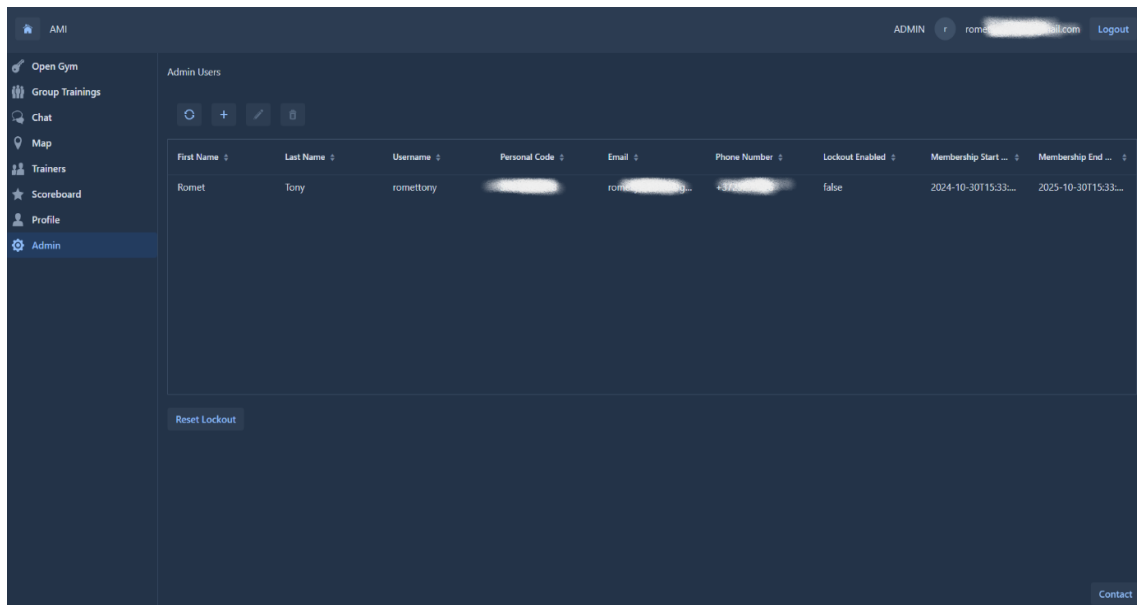
Rank	First Name	Visit Count
1	Romet	2
2	Darajah	1

Joonis 7. Edetabeli vaade

Kasutaja profiili leheküljel on vaade, kus kasutaja saab enda profiili ja sellega seonduvaid andmeid näha (vt Joonis 8). Kasutaja saab värskendada oma andmeid siis, kui soovib näha, kas treeningpaketi maksumus on läbi läinud ja kas see on hakanud kehtima. Muutmise nupu peale vajutades avaneb hüpikaken, mis näitab välja, mida kasutaja saab muuta. Kasutaja saab muuta enda telefoninumbrit, e-posti aadressi, parooli, nime ja elukoha aadressi.

Joonis 8. Kasutaja profiili vaade

Administraatori kasutajate halduse lehekülj on vaade, mida näeb ainult administraatori rolliga kasutaja. Administraator saab hallata kasutajaid. Nupuga "Värskenda kasutaja avamisnupp" saab administraator valida, millise kasutaja avamisnupu ta tahab värskendada. See tähendab, et kui tekkis äpardus jõusaali sissepääsemisega ja nüüd enam ei saa nuppu kasutada enne kuni 00:00 saabub, siis administraator saab selle koheselt lahti teha (vt Joonis 9).






Joonis 9. Administraatori kasutajate halduse vaade

## 4.4 Veebilehe ülesehitus

Veebirakendus on loodud Vaadin Flow raamistikuga, mis toetab SPA ehk *Single-Page Application* struktuuri. See tähendab, et rakenduse erinevate vaadete avamisel uuendatakse ainult konkreetse lehekülje sisu, mitte kogu veebilehte. SPA struktuuri puhul laetakse rakenduse komponendid ja stiilid ainult korra ehk siis esmakordsel veebilehe avamisel. Edasiste navigeerimiste puhul toimub üksnes konkreetsete osade vahetamine, mis tagab sujuvama kasutajakogemuse ja vähendab laadimisaega [25].

Vaadin Flow raamistik kasutab kliendipõhist vaatehaldust läbi *Route* ja *View* klasside. Need võimaldavad dünaamilist navigatsiooni, kus kasutaja liigub rakenduse sees ilma kogu lehte uuesti laadimata. SPA lihtsustab koodihaldust ja võimaldab koodi taaskasutamist, mis vähendab arenduskulusid ja võimalikke veakohti [26]. Vaadin Flow

on parem valik võrreldes Angulari ja Reactiga, kui eesmärk on lihtsustada Java arendajate tööd (vt Joonis 10). Vaadin Flow võimaldab luua täisfunktsionaalseid veebirakendusi puhtalt Java abil, ilma et oleks vaja põhjalikke teadmisi JavaScriptist, HTML-ist või CSS-ist. See tähendab, et kõik saab arendatud ühe programmeerimiskeelega, mis vähendab keerukust ja parandab töö efektiivsust. Lisaks pakub Vaadin Flow sisseehitatud turva- ja kasutajaliidese komponente, mis kiirendab arendust ja tagab kvaliteetse tulemuse [27].

			
Component model	✓	✓	✓
Reactive, declarative	✓	✓	✓
UI component library	✓	✓	
Router	✓	✓	
Forms	✓	✓	
Server communication	✓		
Backend	✓		

Joonis 10. Vaadin Flow, Angular ja React võrdlus [27]

## 4.5 Rakenduse kasutajakeskkonda lisamine

Rakenduse testimiseks kasutati Microsoft Azure platvormi, mis on taskukohane ja autorile tuttav tänu varasematele kogemustele õpingute ajal. Rakenduse juurutamiseks Azure'i keskkonda rakendati Docker'i konteinerlahendust. Docker võimaldab pakendada rakenduse koodi koos kõikide vajalike sõltuvustega ühte konteinerisse, kindlustades, et rakendus töötab ühtlaselt ja sujuvalt igas keskkonnas. Kuna kõik vajalik on konteinerisse integreeritud, saab rakenduse käivitada platvormist sõltumatult, mis lihtsustab testimist ja juurutamist erinevates infrastruktuurides [28].

Rakendus juurutati kasutajakeskkonda sujuvalt. Kulude vähendamiseks otsustas autor kasutada tasuta versiooni, mis on aeglasem, aga siiski vajadusel saab hiljem hõlpsasti üle

minna kiirematele ja võimsamatele lahendustele. Azure'i platvorm pakub vahendeid, et jälgida ressursikasutust ja saada teavitusi, kui kasutajate arv kasvab. See lähenemine aitab optimeerida rakenduse tööaega, kohandades seda vajaduspõhiselt perioodidele, millal on kõrgem kasutajate aktiivsus. Testimise käigus kasutati Azure'i automaatselt loodud URL-i, mis hoidis ajutiselt kulusid madalal. Hiljem on plaanis lisada kohandatud domeeninimi, mis on lühem ja hõlpsamini meelde jääv.

## **5 Tulemused**

Järgmises peatükis räägitakse bakalaureusetöö tulemustest. Antakse ülevaade, et kas ja kuidas said kliendi nõuded täidetud, kliendi tagasisidest ja räägitakse ka edasiarenduse võimalustest väljaspool bakalaureusetööd.

### **5.1 Nõuete täitmine**

Bakalaureusetöö alguses kliendi poolt seatud nõuded saadi täidetud ja loodi kliendi soovidele vastav prototüüp, mis loob võimaluse teha naistele eraldi jõusaal ning loodud prototüübis on võimalik teha kõike toiminguid, mida klient soovis.

Loodud prototüübiga saab avada sissepääsu ust tulevasse naiste jõusaali, näha kui palju inimesi viibib hetkel jõusaalis, saab näha treenereid, on loodud rollisüsteem, saab logida sisse ja välja, näeb kaarti, mille peal on märgitud jõusaalid, loodud suhtluskanal, kus naised saavad üksteist aidata ja üksteisega suhelda, saab näha ja registreerida grupitreeningutele, treener saab luua grupitreeninguid, saab osaleda ja näha edetabelit, mis tekitab motivatsiooni mõningates naistes, samuti saab näha ja hallata oma profiili ning administraatori rolliga kasutaja näeb ka lisaks veel kasutajate haldusvaadet, kus saab näha ja hallata teisi kasutajaid.

### **5.2 Kliendi tagasiside**

Kliendi tagasiside toimus pidevalt, nii arenduskäigu protsessis kui ka pärast valminud prototüüpi. Tagasisidet saadi iga vaate lõpetamisel ja vahepeal ka vaate arenduse ajal.

Klient andis tagasisidet näiteks sellega, et algselt ei olnud plaanis koostada suhtluskanalit bakalaureusetöö raames, vaid oli edasiarenduseks mõeldud, aga kliendi ja autori arvamus muutus ja selle tulemusena loodi ka suhtluskanal. Hetkeseisuga kujutatakse ette, et treeningpakettide eest makstakse ja ei ole realiseeritud sisseehitatud funktsiooni, mis suhtleks näiteks Smart-IDga, kuna arendatud rakendus on veel prototüüp ning on veel testimises ja kliendi soovil realiseeritakse maksusüsteem pärast bakalaureusetööd. Samuti



soovitas klient, et tulevikus võiks rakendus toetada ka täiendavaid turvalahendusi, nagu näiteks mitmetasemeline autentimine, et parandada rakenduse üldist turvalisust.

Pärast prototüübi valmimist hakati seda testima nii kliendi, administraatori kui ka tavakasutaja poolt. Sellejärgselt kõikide tagasiside on antud bakalaureusetöös kajastatud järgmiselt:

Administraatoriks oli hetkel klient ise ja administraatori sõnul oli administraatorirollile vastavad võimalused hästi koostatud, sai teha kõike, mida klient soovis, et administraator saaks teha. Kiitis, et kasutajate sissepääsu lõpuperioodi oli lihtne kasutada ja leida. Suhtluskanal oli tehtud loogiliselt ja lihtsalt, ainuke asi, et praeguse seisuga ei saa saata faile ega pilte ja selle tulemusena tuli ka uus soov, et edasiarenduses võiks saada ka saata faile ja pilte. Grupitreeningute vaade oli samuti lihtsasti arusaadav ja kõik oli kergesti leitav. Sissepääsu vaade toimis täpselt nii nagu vaja oli ja ei olnud midagi arusaamatut selle vaate juures. Profiili vaate juures mainis administraator, et seal võiks olla näha ka seda, mitu korda on kasutaja terve aja vältel jõusaalis käinud, mitte seda, et mingi aja vältel uuendatakse kasutaja andmeid ja siis sellega läheb kasutamiskorrad nulli. Administraatori sõnul oli edetabel küll lihtne ja arusaadav, aga tulevikus sooviks, et see näeks uhkem välja ja poleks nii lihtsasti tehtud. Kaardivaates oli administraatori sõnul kõik väga hea, aga arvas, et juurde võiks panna ka nupu, mis tuvastab ära kliendi praeguse asukoha.

Treener andis peamiselt ainult tagasisidet grupitreeningute vaatel. Treeneri sõnul oli see väga hästi tehtud ja lihtsasti arusaadav. Uue treeningu loomine oli väga kerge ja selle kustutamine samuti. Kohustuslikud väljad ja mittekohustuslikud väljad olid korrektselt tehtud ja rohkemat ei soovinud treener juurde lisada.

Tavakasutaja tagasisides toodi välja, et kui luua uus kasutaja ja kui kasutaja jõuab selleni, kus peab kinnitama e-posti aadressi meili teel, siis ei peaks kopeerima meili pealt koodi ja seda sisestama rakenduses, vaid oleks palju lihtsam, kui meilil vajutad nuppu ja kasutaja on kinnitatud. Tavakasutaja tõi ka ettepaneku lisada täpsem statistika selle kohta, millal jõusaalis on kõige vähem rahvast. Samuti mainis, et visuaalne disain oli väga kasutajasõbralik ja intuitiivne. Eriti kiideti rakenduse värvilahendust, mis tekitas

meeldiva ja energilise tunde. Samas mainis tavakasutaja, et teatud ikoonid võiksid olla visuaalselt eristatavamad.

### **5.3 Edasiarenduse võimalused**

Rakendus on praeguse seisuga prototüüp ja see juba ütleb ära, et rakenduse edasiarendamiseks on väga palju ruumi ja selle peatükiga tuuakse välja kõik edasiarenduse ideed, mis on kliendil tekkinud. Rakendus on koostatud ainult Java keeles ja seetõttu ei ole vaja rohkem teadmisi kui ainult Java keel, mis teeb lihtsaks lisada uusi funktsionaalsusi antud rakendusse.

Üheks oluliseks täiendusvõimaluseks on maksesüsteemi lisamine. Kliendil on soov, et treeningpakettide eest saaks maksta läbi turvalise Smart-ID või Mobiil-ID lahenduse. See tagaks mugava ja turvalise viisi maksete sooritamiseks ning tõstaks rakenduse kasutajasõbralikkust, eriti kuna tegemist on Eestis laialt levinud autentimismeetoditega.

Edasiarenduse võimaluseks on ka kasutajamugavuse suurendamine keelevaliku lisamise kaudu. Hetkel on rakendus saadaval ainult inglise keeles, kuid tulevikus võiks lisada keelevaliku, mis võimaldab kasutajatel valida neile sobiva keele, näiteks eesti, vene või inglise keele. See parandaks rakenduse kättesaadavust erinevatele sihtrühmadele. Samuti võiks erinevate vaadete stiilset poolt natukene muuta, luua rohkem modernsemad funktsionaalsused.

Veel üheks kliendi sooviks on rakenduse mobiilirakenduse kujul loomine. Kuigi praegune rakendus töötab veebipõhiselt, võib mobiilirakenduse arendamine pakkuda kasutajatele paremat ligipääsetavust ning võimalust saada reaajas teavitusi. See aitaks kaasa kiiremale ja tõhusamale info levimisele, näiteks gruppitreeningute muudatuste kohta või suhtluskanalist tulnud sõnumite teavituste kohta. Lõpuks, kuna rakendus hõlmab suhtluskanalit ja reaajas grupivestlusi, võiks edasiarenduseks olla failide ja piltide saatmise võimalus. See funktsioon suurendaks kasutajate võimalusi jagada treeningmaterjale ja muud kasulikku sisu ning muudaks suhtluskanali funktsionaalsemaks.

Andmete varundamine ja turvatestimine on samuti oluline edasiarenduse osa. Rakenduses hoitakse tundlikku kasutajainfot, seega on andmete kaitsmine esmatähtis. Andmeid saab varundada kas pilvelahendustes, näiteks Microsoft Azure'i teenuseid kasutades, või lokaalselt. Turvatestimine ja täiendavad turvameetmed, nagu mitmeastmeline autentimine ja andmete krüptimine, tagavad süsteemi töökindluse ja turvalisuse.

Progressi jälgimine ja analüüs oleks ka üks kliendi soov tulevikus. Rakendusse saaks lisada funktsionaalsuse, mis kuvab kasutaja treeningute statistikat ja edusamme graafiliselt, näiteks graafikute või diagrammidena. See võimaldaks kasutajatel jälgida oma arengut ja võrrelda seda varasemate tulemuste või teiste kasutajatega suhtluskanalis. Edetabeli täiustamine oleks samuti üks edasiarenduse osa. Edetabelit saaks edasi arendada, lisades rohkem kategooriaid ja võimalusi, näiteks kasutajad võiksid teenida märgiseid või saavutusi, mis suurendavad motivatsiooni ja aktiivsust.

Personaalne treeneritugi oleks samuti üks soovidest, kuna olemasolevatel rakendustel ei ole seda hetkel samuti olemas. Lisada võiks võimaluse broneerida isiklike treeningseansse treeneritega otse rakenduse kaudu. Treenerid võiksid pakkuda individuaalset tagasisidet ja näpunäiteid, suurendades rakenduse väärtust. Kliendi soovil võiks ka teha uue treeningplaani vaate, mis pakub kasutajatele võimalust luua ja jälgida isikupärastatud treeningkavasid, mis aitaksid neil paremini oma eesmärged saavutada. See funktsioon võiks sisaldada treeningsoovitusi, mis põhinevad kasutaja eesmärkidel ja varasematel treeningutel.

## 6 Kokkuvõte

Käesoleva bakalaureusetöö eesmärk oli luua spetsiaalne jõusaali rakendus naistele, mis parandaks olemasolevate lahenduste puudujääke ja looks turvalise ning motiveeriva treeningkeskkonna. Rakendus sisaldab erinevaid olulisi funktsionaalsusi nagu reaalsajas grupivestlused, treeningute ajakava koos registreerimisvõimalusega, kasutajatevaheline edetabel, jõusaali asukohtade kaart ning reaalsajas andmed jõusaali külastatavuse kohta. Lisaks võimaldab rakendus kasutajatel avada jõusaali ust läbi rakenduse.

Projekti läbiviimise käigus analüüsiti erinevaid tehnoloogilisi võimalusi ja kliendi soovidele vastavaid lahendusi. Arenduse peamiseks keeleks valiti Java koos Vaadin Flow raamistikuga, mis võimaldas sujuva ja kaasaegse kasutajaliidese loomise. Kasutajakeskkonnaks kasutati Microsoft Azure'i platvormi.

Valminud rakendus vastab esialgselt seatud eesmärkidele. Kuigi kõik planeeritud funktsionaalsused ei jõudnud prototüübi arendusse ajapiirangute tõttu, siis õnneks on loodud tugev platvorm edasiseks arendamiseks. Tulevikuvõimalusteks on näiteks maksusüsteemi lisamine Smart-ID ja Mobiil-ID kaudu, rakenduse mobiiliversiooni loomine, mitmekeelse kasutajaliidese lisamine, failide ja piltide saatmise funktsionaalsus suhtluskanalites ning andmete varundamise ja turvalisuse edasine täiustamine. Rakendus on loodud selliselt, et seda on lihtne edasi arendada ja kohandada vastavalt kasutajate vajadustele ja sellega pakkudes jätkusuutlikku lahendust.

## Kasutatud kirjandus

- [1] Arena District, “Use containers to Build, Share and Run your applications”, Bill Brown. [Võrgumaterjal]. Allikas: <https://www.arenaathletic.com/Blog/a-shocking-number-of-women-avoid-the-gym-for-fear-of-being-judged/>. [Kasutatud: 21.10.2024]
- [2] Apple, “Gym Eesti”, Apple. [Võrgumaterjal]. Allikas: <https://apps.apple.com/ee/app/gym-eesti/id1447661290>. [Kasutatud: 21.10.2024]
- [3] 24-7 fitness, “Mobiilirakenduse juhend”, 24-7 fitness. [Võrgumaterjal]. Allikas: <https://24-7fitness.ee/liitumine/mobiilirakenduse-juhend/>. [Kasutatud: 26.10.2024]
- [4] MyFitness, “MyFitnessi mobiilirakendus”, MyFitness. [Võrgumaterjal]. Allikas: <https://24-7fitness.ee/liitumine/mobiilirakenduse-juhend/>. [Kasutatud: 26.10.2024]
- [5] Alexsoft, “Functional and Nonfunctional Requirements: Specification and Types”, Alexsoft. [Võrgumaterjal]. Allikas: <https://www.altexsoft.com/blog/functional-and-non-functional-requirements-specification-and-types/>. [Kasutatud: 26.10.2024]
- [6] CrustLab, “How to Choose a Web Development Company? An Experts’ Guide to Choosing the Right Development Company for You”, Jakub Urban. [Võrgumaterjal]. Allikas: <https://crustlab.com/blog/selecting-your-web-development-company-factors-to-consider/>. [Kasutatud: 26.10.2024]
- [7] Vaadin, “What is flow?”, Vaadin. [Võrgumaterjal]. Allikas: <https://vaadin.com/docs/latest/flow/what-is-flow>. [Kasutatud: 18.10.2024]
- [8] Vaadin, “Build modern web apps 100% in Java”, Vaadin. [Võrgumaterjal]. Allikas: <https://vaadin.com/flow>. [Kasutatud: 27.10.2024]
- [9] VMware Tanzu, “Why Spring?”, Spring. [Võrgumaterjal]. Allikas: <https://spring.io/why-spring>. [Kasutatud: 18.10.2024]
- [10] PostgreSQL, “PostgreSQL: About”, PostgreSQL. [Võrgumaterjal]. Allikas: <https://www.postgresql.org/about/>. [Kasutatud: 18.10.2024]
- [11] Spring, “SQL Databases”, Spring. [Võrgumaterjal]. Allikas: <https://docs.spring.io/spring-boot/reference/data/sql.html>. [Kasutatud: 19.10.2024]
- [12] Microsoft, “What is Azure?”, Microsoft. [Võrgumaterjal]. Allikas: <https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-azure>. [Kasutatud: 19.10.2024]
- [13] Microsoft, “Azure infrastructure availability”, Microsoft. [Võrgumaterjal]. Allikas: <https://learn.microsoft.com/en-us/azure/security/fundamentals/infrastructure-availability>. [Kasutatud: 20.10.2024]

- [14] Microsoft, “Build in the cloud with an Azure account”, Microsoft. [Võrgumaterjal].  
Allikas: <https://azure.microsoft.com/en-us/pricing/purchase-options/azure-account?icid=payg>.  
[Kasutatud: 19.10.2024]
- [15] Vaadin, “Project Setup”, Vaadin. [Võrgumaterjal]. Allikas:  
<https://vaadin.com/docs/latest/getting-started/tutorial/project-setup>. [Kasutatud: 27.10.2024]
- [16] Hyperskill, “AuthenticationManager”, Hyperskill. [Võrgumaterjal]. Allikas:  
<https://hyperskill.org/learn/step/40291>. [Kasutatud: 20.10.2024]
- [17] Spring, “UserDetailsService”, Spring. [Võrgumaterjal]. Allikas:  
<https://docs.spring.io/spring-security/reference/servlet/authentication/passwords/user-details-service.html>. [Kasutatud: 20.10.2024]
- [18] GeeksForGeeks, “Spring Security – Implementation of BCryptPasswordEncoder”,  
GeeksForGeeks. [Võrgumaterjal]. Allikas: <https://www.geeksforgeeks.org/spring-security-implementation-of-bcryptpasswordencoder/>. [Kasutatud: 20.10.2024]
- [19] Vaadin, “Securing Spring Boot Applications”, Vaadin. [Võrgumaterjal]. Allikas:  
<https://vaadin.com/docs/latest/flow/security/enabling-security>. [Kasutatud: 20.10.2024]
- [20] MODX, “mailtrap.io”, MODX. [Võrgumaterjal]. Allikas:  
<https://docs.modx.com/3.x/en/building-sites/sending-mail/mailtrap>. [Kasutatud: 20.10.2024]
- [21] TurboSMTP, “What is an SMTP server”, TurboSMTP. [Võrgumaterjal]. Allikas:  
[https://serversmtp.com/what-is-smtp-server/?doing\\_wp\\_cron=1731415097.2872490882873535156250](https://serversmtp.com/what-is-smtp-server/?doing_wp_cron=1731415097.2872490882873535156250). [Kasutatud: 20.10.2024]
- [22] Spring, “Interface JavaMailSender”, Spring. [Võrgumaterjal]. Allikas:  
[https://docs.spring.io/spring-framework/docs/3.2.4.RELEASE\\_to\\_4.0.0.M3/Spring%20Framework%203.2.4.RELEASE/org/springframework/mail/javamail/JavaMailSender.html](https://docs.spring.io/spring-framework/docs/3.2.4.RELEASE_to_4.0.0.M3/Spring%20Framework%203.2.4.RELEASE/org/springframework/mail/javamail/JavaMailSender.html). [Kasutatud: 20.10.2024]
- [23] Turing, “Java Persistence API (JPA) For Database Access”, Turing. [Võrgumaterjal].  
Allikas: <https://www.turing.com/kb/jpa-for-database-access>. [Kasutatud: 21.10.2024]
- [24] RedHat, “What is a REST API?”, RedHat. [Võrgumaterjal]. Allikas:  
<https://www.redhat.com/en/topics/api/what-is-a-rest-api>. [Kasutatud: 21.10.2024]
- [25] bloomreach, “What Are Single Page Applications and Why Do People Like Them So  
Much?”, bloomreach. [Võrgumaterjal]. Allikas: <https://www.bloomreach.com/en/blog/what-is-a-single-page-application>. [Kasutatud: 21.10.2024]
- [26] Vaadin, “Routing & Navigation”, Vaadin. [Võrgumaterjal]. Allikas:  
<https://vaadin.com/docs/latest/flow/routing>. [Kasutatud: 21.10.2024]

[27] Medium, “Comparing Frontend Frameworks for Spring Boot: React, Angular, and Vaadin(+JHipster)”, Vaadin. [Võrgumaterjal]. Allikas: <https://medium.com/vaadin/comparing-frontend-frameworks-for-spring-boot-react-angular-and-vaadin-jhipster-c838aa53aeaa>. [Kasutatud: 21.10.2024]

[28] Docker, “Use containers to Build, Share and Run your applications”, Docker. [Võrgumaterjal]. Allikas: <https://www.docker.com/resources/what-container/>. [Kasutatud: 21.10.2024]

## **Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks**

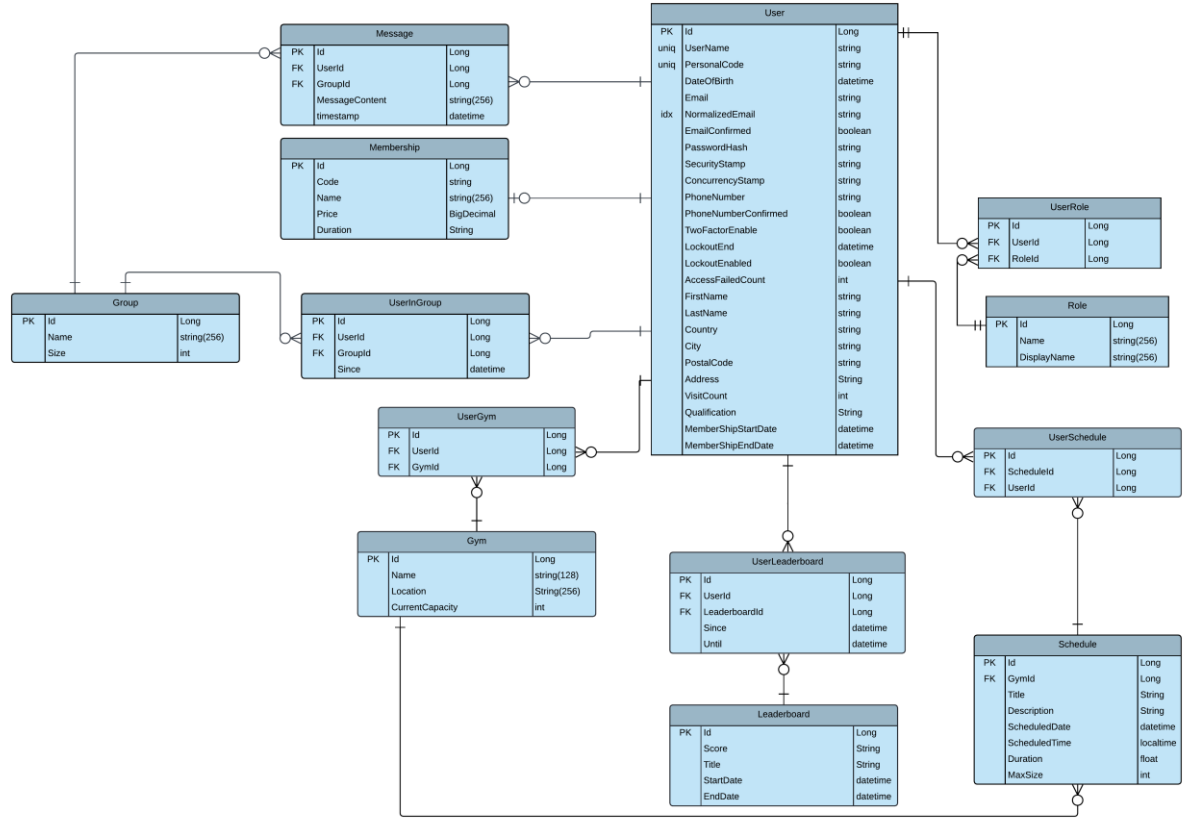
Mina, Romet Tony Jõenurm

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose “Treeningkeskuse rakendus naistele”, mille juhendaja on Meelis Antoi
  - 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.

20.11.2024



## Lisa 2 – Rakenduse andmebaasi skeem



### Lisa 3 – Kasutajavood süsteemi sisenemisel ja profiili vaatamisel

