

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Egert Andrejev 222846IABB
Sander Piik 223247IABB

Tudengiorganisatsioon Mänguklubi veebirakenduse kavandamine ja realiseerimine

Bakalaureusetöö

Juhendaja: Viljam Puusep
MSc

Tallinn 2025

Autorideklaratsioon

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

Autorid: Egert Andrejev, Sander Piik

20.05.2025

Annotatsioon

Bakalaureusetöö eesmärk oli luua Tallinna Tehnikaülikooli tudengiorganisatsioonile Mänguklubi veebirakendus, mis koondaks klubi tegevused ühtsesse keskkonda. Mänguklubil puudub keskne infosüsteem, mis võimaldaks huvilistel tutvuda organisatsiooni tegevusvaldkondadega ning saada ülevaade korraldavatest üritustest.

Käesoleva töö analüüsi osas vaadeldakse olemasolevaid tudengiorganisatsiooni lehtede lahendusi, mille kaasabil kavandatakse arendatavale rakendusele funktsionaalsed ja mittefunktsionaalsed nõuded. Bakalaureusetöö praktilises osas tegelevad autorid rakenduse elluviimisega ja testimisega. Töö tulemusena luuakse veebirakendus, mis toetab Mänguklubi igapäevast tegevust ja lihtsustab selle korraldamist.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 34 leheküljel, 5 peatükki, 11 joonist, 5 tabelit.

Abstract

Design and realization of the student organization Games Club web application

The aim of this bachelor's thesis was to create a web application for Tallinn University of Technology's student organization Games Club that gathers the club's activities into a single environment. Games Club does not have a web application that allows those of interest to familiarize themselves with the organization's areas of activity and to get an overview of the events.

The analysis part of this thesis examines existing solutions for student organization web pages, in order to design functional and non-functional requirements. In the practical part of the thesis, the authors develop and test the web application. The result of the work is a web application that supports Games Club's daily activities and simplifies its management.

The thesis is in Estonian and contains 34 pages of text, 5 chapters, 11 figures, 5 tables.

Lühendite ja mõistete sõnastik

API	<i>Application Programming Interface</i> ehk rakendusliides süsteemide vaheliseks suhtluseks
CI/CD	<i>Continuous Integration / Continuous deployment</i> ehk pidev integratsioon ja pidev juurutamine
<i>Commit</i>	Versioonihalduses tehtud muudatuste salvestuspunkt
CRUD	<i>Create, Read, Update, Delete</i> ehk andmebaasitöötamise toimingute lühendid loo, loe, uuenda, kustuta
<i>Enum</i>	<i>Enumeration</i> ehk tüübi väärtuste fikseeritud loend programmeerimises
HTTP	<i>Hypertext Transfer Protocol</i> ehk veebipõhise andmevahetuse protokoll
JWT	<i>Json Web Token</i> ehk autentimis- ja autoriseerimisvorming
REST	<i>Representational state transfer</i> ehk veebiteenuste arhitektuurstiil
UTC	<i>Universal Time Coordinated</i> ehk maailma ajavööndite aluseks olev universaalne aeg

Sisukord

1 Sissejuhatus	11
1.1 Üldine taust.....	11
1.2 Probleem	11
1.3 Eesmärk	12
1.4 Töö struktuur	12
2 Metoodika.....	13
2.1 Objekti detailne kirjeldus.....	13
2.2 Tööriistad.....	13
2.3 Arendusprotsess.....	14
2.3.1 Tööjaotus	15
2.3.2 Rollid	15
2.3.3 Strateegia	16
2.4 Disain.....	17
3 Analüüs.....	18
3.1 Olemasolevate tudengiorganisatsiooni lehtede analüüs	18
3.2 Nõuete kaardistamine	20
3.2.1 Funktsionaalsed nõuded	20
3.2.2 Mittefunktsionaalsed nõuded.....	21
3.3 Tehnoloogiate valik	22
3.4 Kasutuslood	22
3.4.1 Ürituste vaade	22
3.4.2 Inventari vaade	24
3.4.3 Tõlgete vaade.....	24
3.4.4 Keelte vaade	26
3.4.5 Žanrite vaade	27
3.4.6 Asukohtade vaade.....	27
3.4.7 Kasutajate haldamise vaade.....	27
4 Töö tulemused	29
4.1 Eesrakendus	29

4.1.1 Arhitektuur	29
4.1.2 Kasutajaliides	30
4.2 Tagarakendus.....	32
4.2.1 Arhitektuur	32
4.2.2 Kontrollerid	32
4.2.3 Teenused.....	33
4.2.4 Repositooriumid	34
4.2.5 Klassid	35
4.3 Testimine	36
5 Järeldused	39
5.1 Tehnoloogilised tööriistad	39
5.2 Lahenduse vastavus nõuetele.....	39
5.3 Tagasiside valminud arendusele	40
5.4 Hinnang projektile	40
5.5 Probleemid ja katsumused	41
5.6 Võimalused edasiarenduseks	41
5.7 Meeskonnaliikmete tegevused.....	42
5.7.1 Egert Andrejevi sisuline kokkuvõte	42
5.7.2 Sander Piiki sisuline kokkuvõte	43
Kokkuvõte	45
Kasutatud kirjandus	46
Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks	48
Lisa 2 – Egert Andrejevi eneseanalüüs	49
Lisa 3 – Sander Piiki eneseanalüüs	50
Lisa 4 – Versioonihaldussüsteemi väljavõte	51
Lisa 5 – Avaleht öövaates.....	60
Lisa 6 – „Meist“ vaheleht	61
Lisa 7 – „Üritused“ vaheleht	62
Lisa 8 – Ürituse detailvaade	63
Lisa 9 – „Inventar“ vaheleht.....	64
Lisa 10 – Mängu detailvaade.....	65
Lisa 11 – Registreerimisvorm.....	66
Lisa 12 – Sisselogimisvorm.....	67

Lisa 13 – Üritusel osalemise indikaatorid	68
Lisa 14 – Ürituse lisamise vorm	69
Lisa 15 – Ürituse redigeerimise vorm	70
Lisa 16 – Kustutamise kinnitamise modaalaken	71
Lisa 17 – Mängu lisamise vorm	72
Lisa 18 – „Tõlked“ vaheleht.....	73
Lisa 19 – Tõlke lisamise vorm	74
Lisa 20 – Tõlke redigeerimise vorm.....	75
Lisa 21 – Žanri lisamise vorm	76
Lisa 22 – Žanri detailvaade	77
Lisa 23 – Keele lisamise vorm	78
Lisa 24 – Asukoha lisamise vorm.....	79
Lisa 25 – „Kasutajad“ vaheleht	80
Lisa 26 – Järgmise ürituse indikaator	81

Jooniste loetelu

Joonis 1. Ürituse lisamine.....	23
Joonis 2. Tõlke loomine	26
Joonis 3. Mänguklubi avaleht.....	31
Joonis 4. Tagarakenduse arhitektuur	32
Joonis 5. Kontrolleri koodi näide	33
Joonis 6. Teenuse klassi näide.....	34
Joonis 7. Repositoorium klassi näide	34
Joonis 8. Klassidiagramm.....	35
Joonis 9. Ühiktestide tulemused	36
Joonis 10. Koodikaetus ühiktestidega	37
Joonis 11. Testi näide	38

Tabelite loetelu

Tabel 1. Olemasolevate tudengiorganisatsiooni veebilehtede funktsionaalsused ja omadused.....	19
Tabel 2. Funktsionaalsed nõuded	20
Tabel 3. Mittefunktsionaalsed nõuded	21
Tabel 4. Egert Andrejevi tööde sisuline kokkuvõte	42
Tabel 5. Sander Piiki tööde sisuline kokkuvõte	43

1 Sissejuhatus

Sissejuhatus annab ülevaate lõputöö lähtekohast ning põhjendab veebirakenduse loomise vajadust Tallinna Tehnikaülikooli tudengiorganisatsiooni Mänguklubi jaoks. Peatükis tutvustatakse organisatsiooni tausta ja senist toimimist, sõnastatakse probleem ning määratletakse töö eesmärk. Lisaks antakse ülevaade töö sisulisest struktuurist.

1.1 Üldine taust

Tallinna Tehnikaülikooli Mänguklubi on 2024. aasta augustis loodud tudengiorganisatsioon, mis ühendab laua- ning videomänguhuvilistest üliõpilasi. Klubi tekkis tudengite omaalgatuslikust soovist luua koht, kus mänguhuvilistel on võimalik regulaarselt kokku saada ja ühiselt vaba aega veeta.

Mänguklubi on Tallinna Tehnikaülikooli kõige uuem ametlikult registreeritud organisatsioon. Esimese tegevusaasta jooksul on liikmeid kogunenud üle saja, kelle seas on nii eesti- kui ka inglise keelt kõnelevaid tudengeid kõikidest teaduskondadest. Klubi korraldab iganädalasi mänguõhtuid ja ülenädalasi turniire, mille populaarsus on kiiresti kasvanud.

Tudengiorganisatsioonide seas on tavapärane, et igal klubil kujuneb välja oma identiteet ülikooli kogukonnas. Uute organisatsioonide puhul, nagu Mänguklubi, on oluline tegeleda liikmeskonna hoidmise ja info kättesaadavuse tagamisega juba varajases faasis, et toetada edasist organisatsiooni kasvu.

1.2 Probleem

Mänguklubil puudub veebirakendus, mis vastaks organisatsiooni sisulistele vajadustele. Klubi tegevused, nagu ürituste info jagamine ja inventari haldamine, toimuvad eraldiseisvates keskkondades, mis raskendab klubi juhatuse tööd.

Kuigi juhatus oli vajadusest ühtse süsteemi järele teadvustanud klubi loomise hetkel, ei leitud organisatsiooni sees inimest, kellel oleks olnud oskused sobivat lahendust luua.

Vajadus lihtsama töökorralduse järele ning soov parendada info liikumist klubi ja tudengite vahel, tõi esile konkreetse probleemi, mille lahendamiseks otsustati lõputöö raames arendada vajadustele vastav veebirakendus.

1.3 Eesmärk

Lõputöö eesmärk oli luua Tallinna Tehnikaülikooli tudengiorganisatsioonile Mänguklubi veebirakendus, mis koondaks klubi tegevused ühtsesse keskkonda. Loodav lahendus peab toetama kasutajasõbralikku sisu ning realiseerima kavandatud funktsionaalsed ja mittefunktsionaalsed nõuded. Süsteemi eesmärk on muuta klubi igapäevane korraldustöö lihtsamaks ja luua alus tulevikus lisanduvatele funktsionaalsustele, nagu näiteks võistluste tulemuste kajastamine ning statistika esitamine.

1.4 Töö struktuur

Töö on jaotatud kuueks osaks. Sissejuhatuses antakse ülevaade töö taustast, probleemist ning sõnastatakse eesmärk. Metoodika peatükis esitatakse kirjeldus töös kasutatud tööriistadest ja arendusprotsessidest. Analüüsi peatükis vaadeldakse olemasolevaid lahendusi ning selle abil täiendatakse töö nõudeid. Töö tulemuste peatükis selgitatakse lahti ees- ja tagarakenduse ülesehitus. Järelduste juures annavad autorid meeskondliku ja arendusliku hinnangu kogu töö valmimisele ja esitatakse testkasutajate tagasiside valminud lahendusele. Kokkuvõttes võetakse töö põhisisu ja saavutatud tulemused lühidalt kokku.

2 Metoodika

Peatüki eesmärk on anda ülevaade lõputöö käigus kasutatud arenduslikust lähenemisest ja tööprotsessi ülesehitusest. Kuna tegemist on tarkvaraarendusprojektiga, keskendub töö eeskätt rakenduse nõuete analüüsimisele, kavandamisele, realiseerimisele ja testimisele.

2.1 Objekti detailne kirjeldus

Lõputöö on praktilise suunitlusega tarkvaraarendusprojekt, mille tulemusena valmis Tallinna Tehnikaülikooli tudengiorganisatsioonile Mänguklubi veebirakendus. Enne töö algust ei olnud klubil ühtegi veebipõhist lahendust, mis toetaks organisatsiooni tegevuste haldamist. Klubi igapäevane korraldus tugines sotsiaalmeediale ja tabelitöötlusprogrammidele.

Projektiga loodi lahendus nullist, arvestades organisatsiooni vajadusi ja tulevikuvajadusi. Tegemist ei olnud varasema süsteemi laiendamise ega ümberehitamisega. Töö käigus kasutati tarkvaraarenduses levinud metoodilisi lähenemisi, mis võimaldas funktsionaalsuse järk-järgulist realiseerimist ja valideerimist.

2.2 Tööriistad

Arenduseks kasutati JetBrains Rider ning Visual Studio Code arenduskeskkondasid. Tagarakendus loodi C# programmeerimiskeeles ning realiseeriti .NET raamistikus [1]. Eesrakendus töötati välja Next.js raamistikus ning TypeScript keeles [2], [3]. Kasutajaliidese kujundamiseks kasutati Tailwind CSS-i [4].

Andmete salvestamiseks kasutati PostgreSQL andmebaasi ning selle haldamiseks pgAdmin 4 liidest [5]. Serveri ja kliendivaheline suhtlus toimus HTTP protokolliga kaudu, rakendades REST API päringuid. Autentimine ja autoriseerimine lahendati JWT abil, kus lisaks võeti kasutusele rollipõhine ligipääsukontroll.

Versioonihalduseks kasutati GitLab platvormi, kus hoiti koodi, harusid ja arendusega seotud ülesandeid. Projekti arendus viidi läbi eraldiseisvates ees- ja tagarakenduse projektides.

Tagarakenduses loodi eraldi ühiktestide alamprojekt, kus loodi testid xUnit raamistikus ja sõltuvused isoleeriti Moq abil. Arenduse käigus kasutati ka Postman tööriista, mille abil testiti manuaalselt API päringute vastuseid.

2.3 Arendusprotsess

Arendusprotsessis järgiti iteratiivse ja agiilse töökorralduse põhimõtteid. Kasutati SCRUM ja Extreme Programming metoodikaid, et saavutada võimalikult sujuv töövoog [6], [7].

Veebirakenduse loomiseks tehti esmalt GitLab keskkonda kaks eraldiseisvat projekti:

- manguklubi-api – serveripoolse loogika arendamiseks,
- manguklubi-fe – kliendipoolse kasutajaliidese loomiseks.

Iga arendustegevuse eel loodi GitLabis pilet, mis kirjeldas konkreetse funktsionaalsuse eesmärki. Iga pileti alusel loodi vastav tööharu, mille valmimisel viidi koodimuudatused läbi *merge request*'i peaharusse. See võimaldas hoida ees- ja tagarakendust sõltumatult hallatavana.

Autorid pidasid kaks korda nädalas regulaarseid lühikoosolekuid (*stand-up*), mille käigus anti ülevaade tehtud töödest, määrati järgmised sammud ning lahendati probleeme. Lisaks tehti paarisprogrammeerimist, mis võimaldas autoritel tehnilisi vigu kiiremini tuvastada ning parandada.

Arendusprotsessi käigus hoiti aktiivset kontakti Mänguklubi juhatusega, mille eesmärgiks oli koguda kliendipoolset tagasisidet valminud funktsionaalsustele. Lisaks toimusid iga kahe nädala tagant kõned ka juhendajaga, mille käigus saadi suuniseid arenduse parendamiseks. Tagasiside andis autoritele võimaluse muuta ja täiendada valminud funktsionaalsusi, et arendada kliendi soovidele vastav rakendus.

2.3.1 Tööjaotus

Meeskond koosnes kahest liikmest, kes jagasid arendustööd vastavalt oma varasematele kogemustele ja tehnilistele pädevusele. Üks meeskonnaliige keskendus serveripoolse loogika arendamisele ja teine kliendipoolse kasutajaliidese arendusele. Selline jaotus võimaldas mõlemal liikmel keskenduda arendustegevustele, milles nad tundsid end kindlalt.

Serveripoolse arenduse eest vastutav liige tegeles .NET Web API ülesehitusega, äriloogika rakendamisega, andmebaasi struktuuride loomisega ning andmeedastusega. Samuti kuulus tema vastutusalasse autentimise ja autoriseerimise lahendus ning REST API otspunktide realiseerimine.

Kliendipoolse arenduse eest vastutav liige tegeles Next.js põhise kasutajaliidese loomise, komponentide struktureerimise, kujunduse teostamise ja administraatori funktsionaalsuste rakendamisega. Samuti liidestas liige serveripoolse rakenduse otspunktid kliendipoolse rakendusega.

Tööjaotus oli ligikaudu 50/50. Mõlemad liikmed panustasid võrdselt ning tegid koostööd funktsionaalsuste ühendamisel. Aeg-ajalt teostati ka vastastikust koodiülevaatus ning vajadusel aidati teineteist keerulisemate tehniliste probleemide lahendamisel.

Arendusprotsessi käigus tehti aktiivselt versioonihalduseks *commite* GitLabi, millest on näha tööjaotuse tegelik panus. Täpsem *commitide* ajalugu on esitatud Lisas 4.

2.3.2 Rollid

Tarkvaraarendus toimus SCRUM metoodika üldpõhimõtteid järgides, mille kohaselt jaotatakse meeskonnas rollid tüüpiliselt järgmiselt: tooteomanik (*Product Owner*), protsessijuht (*Scrum Master*) ning arendajad (*Developers*) [6].

Projekti puhul ei olnud tegemist klassikalise SCRUM tiimiga, kus rollid oleksid selgelt eristatud ja täidetud eraldiseisvate osapoolte poolt. Kuna meeskond koosnes vaid kahest arendajast, tuli mõlemal võtta endale mitmeid rolle korraga. Selline töökorraldus võimaldas autoritel paindlikult ja kiirelt reageerida töö käigus esile kerkivatele vajadustele.

Projektis täitsid mõlemad autorid paralleelselt mitut rolli. Tarkvaraarenduse osa eest vastutas kumbki enda valdkonna eest. Mõlemad autorid lõid pileteid ja uue funktsionaalsuse arendamisel koostas vastava tööülesande tavaliselt see arendaja, kes ka konkreetse funktsionaalsuse ellu viis. Tooteomaniku rollis tegeldi ühiselt tööülesannete täpsustamise ning tagasiside kogumisega juhendajalt ja Mänguklubi juhatusest. Lisaks kandsid mõlemad ka protsessijuhi vastutust, mille raames hoolitseti arendusprotsessi sujuvuse eest, toetati üksteist tehniliste probleemide lahendamisel.

Töövoo juhtimise ja arendustegevuste koordineerimise osas võttis üks autoritest enda kanda projekti eestvedaja rolli. Tema vastutusalasse kuulus tööülesannete prioritseerimine GitLabis, regulaarsete koosolekute algatamine ja üldise töömahu ning tähtajalise valmimise jälgimine. See tagas, et tegevused liikusid edasi projekti ajakava ja eesmärkidega.

2.3.3 Strateegia

Arendusprotsessi strateegia kujundamisel lähtuti SCRUM metoodika üldpõhimõtetest, mille kohaselt on oluline pidev väärtuse loomine, regulaarne tagasiside ning läbipaistev töövoog [6]. Arvestades piiratud ajaraami ja töömahtu, oli strateegia eesmärk tagada, et kõik planeeritud funktsionaalsused valmiksid õigeaegselt ja kvaliteetselt.

Tööd jagati väiksemateks, selgelt piiritletud piletiteks, mis dokumenteeriti GitLab keskkonnas. Iga ülesande elluviimisele eelnes selle prioritseerimine vastavalt süsteemi loogilisele ülesehitusele ja sõltuvustele.

Koodi kirjutamisel järgiti Extreme Programming metoodikale omaseid praktikaid nagu:

- paarisprogrammeerimine, mille kaudu lahendati keerulisemaid loogikaprobleeme ühiselt ja vähendati vigade tekkimise tõenäosust [7];
- refaktoreerimine, et tagada koodi loetavus, hooldatavus ning tehniliste võlgade vältimine [7];
- arendamine väikeste hoomatavate tükkidena, et saada igale funktsionaalsusele kiiresti tagasisidet [7].

Erinevate töövaldkondade (ees- ja tagarakenduse) iseseisvuse säilitamiseks kasutati kahte eraldiseisvat GitLab projekti, mis võimaldasid liikmetel keskenduda oma vastutusalale

ilma koodibaasis konflikte põhjustamata. Samas toimus töö sünkroniseerimine funktsionaalsuse lõikes – näiteks enne uue sisulehe või haldusvaate loomist lepitati kokku, millised API otspunktid on vajalikud ning milline on nende struktuur.

Arenduse edenemise huvides toimusid regulaarsed koosolekud, kus määrati järgnevad prioriteetsed tööd, hinnati tehtut ning lahendati võimalikke takistusi.

2.4 Disain

Süsteemi arendamisel lähtuti tunnustatud tarkvaraarenduse disainipõhimõtetest, mille eesmärk oli tagada koodi puhtus ja edasine hooldatavus. Koodi ülesehituses kasutati meetodeid, mis toetavad komponentide isoleeritust, korduvkasutatavust ning lihtsustavad testimist ja hilisemat laiendamist.

Üheks olulisemaks põhimõtteks oli *Single Responsibility Principle* (SRP), mille kohaselt igal klassil ja komponendil oli ainult üks kindel vastutusala [8], [9]. Nii serveri- kui kliendipoolses arenduses jagati funktsionaalsus selgelt kihtide ja kaustade vahel, vältides liigseid sõltuvusi.

Rakenduse struktuuri kavandamisel järgiti ka *Dependency Injection* (DI) põhimõtet. Sõltuvuste süstimine võimaldas eraldada klasside loomise loogika nende kasutusloogikast, mis lihtsustas moodulitevahelist sõltuvust ning võimaldas teenuste efektiivset testimist [11].

Koodi korduvuse vältimiseks rakendati *Don't Repeat Yourself* (DRY) põhimõtet [10]. Ühised funktsioonid ja abimeetodid koondati eraldi klassidesse, kust neid sai taaskasutada erinevates rakenduse osades.

Tagarakendus ehitati üles viisil, mis toetab kihelist arhitektuuri, kus vastutusvaldkonnad on jagatud selgelt kontrolleri-, teenuse- ja repositooriumkihtide vahel. Selline lähenemine aitas tagada rakenduse parema hallatavuse ning loogilise ja testitava koodibaasi [12].

3 Analüüs

Peatükis antakse ülevaade arendatava infosüsteemi kavandamise aluseks olnud analüüsist. Esmalt vaadatakse, milliseid funktsionaalsusi teised tudengiorganisatsioonid oma veebilehtedel kasutavad, et selle põhjal otsustada, mida võiks ka Mänguklubi rakendusse lisada. Seejärel tuuakse välja süsteemile kavandatud nõuded ning viimasena on esitatud kasutuslood, mis kirjeldavad planeeritud infosüsteemi käitumist.

3.1 Olemasolevate tudengiorganisatsiooni lehtede analüüs

Olemasolevate tudengiorganisatsiooni lehtede analüüsimise eesmärk on kaardistada, millised võimalused eksisteerivad olemasolevatel tudengiorganisatsioonide veebilehtedel. Kaardistatud funktsionaalsuste põhjal saab otsustada, milliseid funktsionaalsusi peaks Mänguklubi veebirakendus samuti omama. Vaatluse alla võeti viis organisatsiooni, mis paistsid silma kas oma kasutajaliidese kvaliteedi poolest või on Tallinna Tehnikaülikoolis tuntud suure liikmeskonnaga tudengiorganisatsioonid: ITÜK, MTÜK, TalTechi Turundusklubi, Kultuuriklubi ja Lapikud.

ITÜK-i (ituk.ee) puhul jääb silma väga professionaalne kasutajaliides, kus kasutatakse tähelepanu võitmiseks erinevaid illustratsioone. Leht toetab nii eesti kui ka inglise keelt ja võimaldab läbi vormi organisatsiooniga liitumist. Lisaks pakub leht ülevaadet üritustest ja renditavatest vahenditest. Kasutajakogemuse seisukohast on leht kiire, sujuv, mobiilisõbralik ja lihtsasti navigeeritav [13].

MTÜK (mtuk.ee) on visuaalselt atraktiivne ja omab ühtset stiili kõikidel lehtedel. Leheküljel on olemas eraldi menüüpunkt „Kontakt“, läbi mille saab otse organisatsioonile kirjutada. Lisaks on olemas ka ürituste ja blogi vaade, mis annab hea ülevaate klubi tegemistest. Üliõpilaskoguga liitumiseks on olemas suunis liitumise ankeedile. Lehel navigeerimine on kiire, intuitiivne ja kohaldub nutiseadmetega [14].

Kultuuriklubi (kultuuriklubi.ee) ei oma ühtseid visuaalseid komponente, ehk lehtedel on kasutatud erinevaid teksti ja visuaalide stiile. Sarnaselt ITÜKile on veebilehel olemas tehnika rendi leht, kus saab ülevaate inventarist, mida on võimalik klubilt rentida. Lisaks

kuvatakse lehel tulevasi ja möödunud üritusi. Veebileht on kiire ja kohaldub kõigi ekraanisuurustega [15].

Lapikud (lapikud.ee) omavad kõikidel lehtedel ühtset stiili. Lehel on olemas blogi, mis ei tundu olevat aktiivselt kasutatud, kuid võimekus postituste tegemiseks on olemas. Leht eristub teistest selle poolest, et on olemas selge ülevaade klubi ajaloo kohta, kus kuvatakse kõiki liikmeid läbi aegade. Tarkvaraarenduse klubiga liitumiseks on olemas lehte integreeritud Google'i vorm. Üldiselt on leht kiire ja kohaldub kõigi vaadetega [16].

Turundusklubi (taltechturundusklubi.com) leht on kasutajaliidese poolest atraktiivne ja hoitakse ühtset stiili. Sarnaselt ITÜK-le toetab veebileht nii eesti kui ka inglise keelt. Lehel on ülevaade projektidest (üritused), milles osaletakse ja sarnaselt MTÜK-le on võimalik klubile otse kirjutada läbi „Kontakt“ vaate. Klubiga liitumiseks suunatakse külastaja Google'i vormi. Veebirakendus on kiire ja kohaldub kõigi vaadetega [17].

Kõik vaadeldud veebilehed omavad erinevaid funktsionaalsusi tulenevalt enda vajadusest. Kaardistatud olemasolevaid funktsionaalsuseid (Tabel 1) võetakse arvesse Mänguklubi nõuete analüüsis järgmises peatükis.

Tabel 1. Olemasolevate tudengiorganisatsiooni veebilehete funktsionaalsused ja omadused

Identifikaator	Funktsionaalsus / omadus
F-01	Mitmekeelsuse olemasolu (EST/ENG).
F-02	Klubi tegevusi tutvustav vaade.
F-03	Klubiga liitumise võimalus läbi vormi või juhendi.
F-04	Ülevaade tulevastest ja möödunud üritustest.
F-05	Ülevaade renditavatest vahenditest.
F-06	Ülevaade klubi ajaloost.
F-07	Ülevaade klubi (juhatuse) liikmetest.
F-08	Blogide ja posituste voog.
F-09	Klubiga kontakteerumise võimalus veebilehe kaudu.
F-10	Ühtse stiili ja kasutajaliidese komponentide kasutamine lehekülgedel.
F-11	Lihtne ja arusaadav navigeerimine veebilehel.
F-12	Leheküljed kohalduvad nii telefonis kui ka arvutis.

3.2 Nõuete kaardistamine

Arendatava infosüsteemi nõuded kaardistati koostöös Mänguklubi juhatusega. Esmaseks sisendiks olid klubi senised tööpraktikad ja nende kitsaskohad. Seejärel analüüsiti olemasolevate tudengiorganisatsioonide veebilehti, et mõista, mis funktsionaalsusi peaks ka Mänguklubi veebirakendus pakkuma. Arutelude käigus tuvastati süsteemile seatavad ootused, mis jagati funktsionaalseteks ja mittefunktsionaalseteks nõueteks.

3.2.1 Funktsionaalsed nõuded

Funktsionaalne nõue kirjeldab süsteemi käitumist, mida süsteem peab pakkuma külastaja, kasutaja (klubi liige), administraatori või super-administraatori vaates. Iga nõue tabelis 2 on tähistatud unikaalse identifikaatoriga kujul FR-XX, kus FR tähistab ingliskeelset terminit *Functional Requirement* ning XX viitab konkreetsele nõudele. Funktsionaalsete nõuete loomisel võeti arvesse Mänguklubi juhatuse soove ja teiste tudengiorganisatsioonide funktsionaalsusi.

Tabel 2. Funktsionaalsed nõuded

Identifikaator	Kirjeldus
FR-01	Külastajale peab olema nähtav klubi eesmärkide ja tegevuste tutvustus.
FR-02	Külastajale peab olema nähtav Mänguklubi eesseisvad ja toimunud üritused.
FR-03	Külastajale peab olema nähtav Mänguklubi olemasolevad mängud.
FR-04	Külastaja peab saama luua konto ning sisse logida.
FR-05	Kasutaja peab saama end üritustele registreerida ja osalemist tühistada.
FR-06	Administraator peab saama luua/muuta/kustutada üritust.
FR-07	Administraator peab saama luua/muuta/kustutada mängu.
FR-08	Administraator peab saama luua/muuta/eemaldada tõlkeid.
FR-09	Administraator peab saama luua/muuta/eemalda ürituste asukohti mängude žanreid ja keeli.
FR-10	Super-administraator peab saama kasutajale anda ja eemaldada administraatori rolli ning eemaldada kasutaja.

Funktsionaalsuste määratlemiseks eristatakse nelja rolli: külastaja, kasutaja, administraator ja super-administraator. Vastavalt nendele rollidele on tabelis 2 esitatud konkreetsetele kasutajatele suunatud nõuded.

Mänguklubi eesmärkide ja tegevuste tutvustus, ürituste vaade, mängude (inventari) vaade ning tõkelahendus on tuletatud vastavalt olemasolevate tudengiorganisatsioonide funktsionaalsustele F-02, F-04, F-05 ja F-01 (vaata Tabel 1). Teisi funktsionaalsuseid ei peetud Mänguklubi veebirakenduse arendamise raames esmatähtsaks, tulenevalt klubi juhatuse soovidest. Mänguklubiga liitumine võrdsustatakse antud arenduse kontekstis konto loomisega.

3.2.2 Mittefunktsionaalsed nõuded

Mittefunktsionaalsed nõuded määravad ära süsteemi töökindluse ja kasutajakogemuse aspektid. Need ei kirjelda konkreetseid funktsioone, vaid tingimusi, millele süsteem tervikuna peab vastama. Iga mittefunktsionaalne nõue on tähistatud identifikaatoriga kujul NFR-XX, kus NFR viitab ingliskeelsele terminile *Non-Functional Requirement* ning XX viitab konkreetsele nõudele. Mittefunktsionaalsed nõuete loomisel võeti arvesse Mänguklubi soove ja olemasolevate tudengiorganisatsiooni lehtede omadusi.

Tabel 3. Mittefunktsionaalsed nõuded

Identifikaator	Kirjeldus
NFR-01	Kasutajaliides peab toetama vähemalt eesti ja inglise keelt.
NFR-02	Kasutajaliides peab omama ühtset stiili kõikidel lehtedel.
NFR-03	Veebirakendus peab töötama nii arvutites kui mobiilseadmetes.
NFR-04	Uute funktsioonide lisamine ei tohi nõuda olemasoleva koodistruktuuri ümberkujundamist.
NFR-05	Rakendus peab olema arusaadav ja intuitiivne külastajale/kasutajale.
NFR-06	Kasutajaandmed ja sisselogimine peavad olema kaitstud ning nähtavad ainult õigustatud kasutajatele.
NFR-07	Administraator peab saama teha sisulisi muudatusi (nt tõlked, sisutekstid) ilma koodi muutmata.

Eesti ja inglise keele tugi, rakenduse toimimine arvutis ja mobiilis, ühtne stiil lehtedel ning intuitiivne kasutajaliides on tuletatud vastavalt olemasolevate tudengiorganisatsioonide omadustest F-01, F-12, F-10 ja F-11 (vaata Tabel 1). Ülejäänud loodud mittefunktsionaalsed nõuded on lisatud vastavalt Mänguklubi juhatuse soovidele.

3.3 Tehnoloogiate valik

Infosüsteemi arendamisel võeti tehnoloogiliste valikute tegemisel arvesse nii süsteemi funktsionaalseid ja mittefunktsionaalseid nõudeid kui ka arendajate varasemat kogemust eri raamistikega. Eesmärk oli kasutada usaldusväärseid tehnoloogiaid, mis teeksid arenduse võimalikult mugavaks ja skaleeritavaks.

Mõlemad töö autorid omavad varasemat kogemust .NET-i raamistikuga, seega otsustati tagarakenduse arendamiseks valida just see raamistik. Kuna ühel autoritest oli varasem praktiline kogemus Reactiga, otsustati Reactil põhineva raamistiku Next.js kasuks.

Tulenevalt kaardistatud nõuetest sisaldab arendatav rakendus andmeolemite vahelisi seoseid, näiteks kasutaja osalemine üritusel. Toetudes varasemale kogemusele otsustati andmebaasina kasutusele võtta PostgreSQL, mis toetab relatsioonilisi seoseid [18].

3.4 Kasutuslood

Lisaks nõuete analüüsimisele koostasid autorid ka kasutuslood, mis kirjeldavad, kuidas kasutaja erinevates vaadetes rakendust kasutab. Järgnevalt on lahti kirjutatud, millised sisendväljad peavad olema täidetud ning millised on soovituslikud, kui administraator loob või redigeerib andmeid. Peatükis on esitatud ka kaks joonist tegevusvoogude kohta. Arendatud rakenduse vaated ja vormid on loodud antud kasutuslugude baasil.

3.4.1 Ürituste vaade

Ürituste vaade võimaldab rakenduse külastajatel tutvuda tulevaste ja varasemate üritustega ning administraatoritel ka üritusi luua, muuta ja kustutada (Lisa 7).

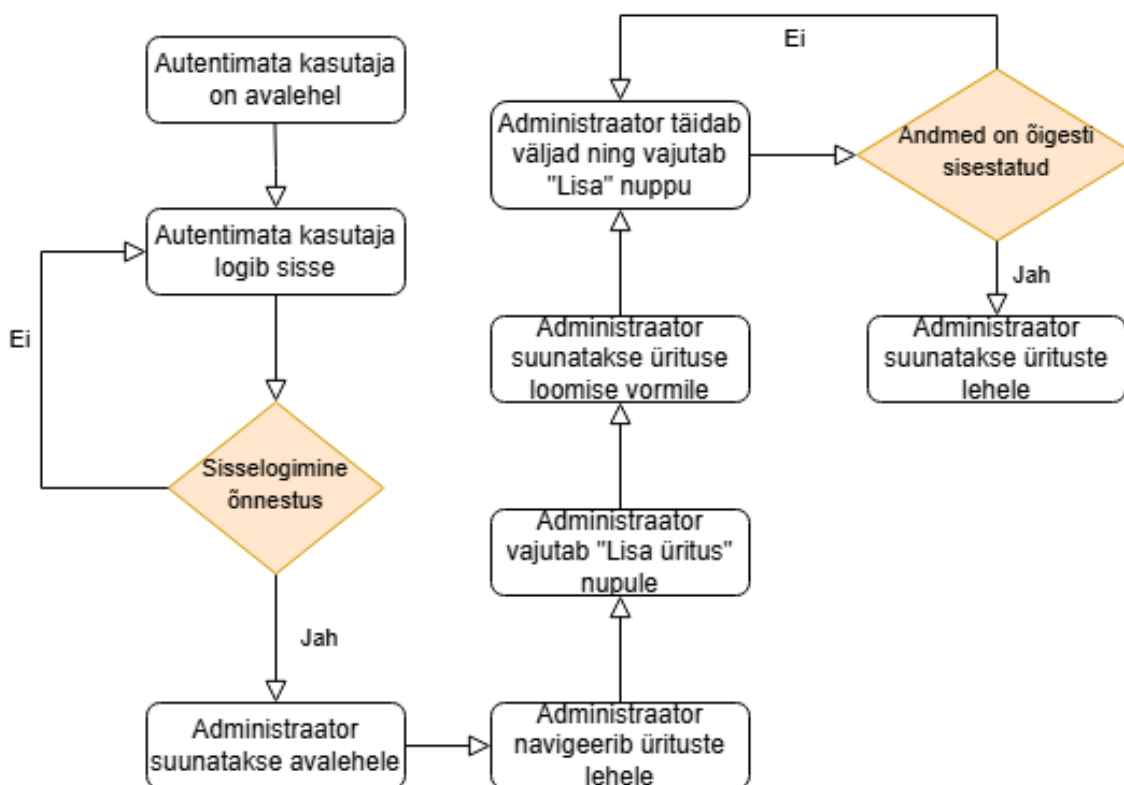
Ürituse lisamine ja redigeerimine

Administraator saab alustada uue ürituse loomist, vajutades vaates nuppu „Lisa üritus“. See avab eraldi vormivaate, kus tuleb sisestada järgmised andmed:

- Ürituse nimi (eesti ja inglise keeles) – kohustuslik
- Kirjeldus (eesti ja inglise keeles) – vabatahtlik
- Algusaeg (ei tohi olla minevikus) – kohustuslik
- Lõppaeg (peab olema hilisem kui algus) - vabatahtlik
- Asukoht – vabatahtlik, valitav olemasolevate asukohtade hulgast

- Link üritusele – vabatahtlik

Õnnestunud salvestamise korral suunatakse kasutaja tagasi ürituste vaatesse (Lisa 14). Administraatori ürituse loomise töövoogu kirjeldab ka allolev joonis (Joonis 1).



Joonis 1. Ürituse lisamine

Redigeerimisvõimalus on saadaval ainult tulevaste ürituste jaoks, ehk mineviku ürituste detaile ei saa administraator muuta (Lisa 15). Administraator pääseb redigeerimise vormivaatesse vajutades redigeerimisikoonile konkreetse ürituse peal. Vormis kehtivad samad sisendi valideerimisreeglid nagu lisamisel.

Ürituse kustutamine

Ürituse kustutamiseks tuleb vajutada vastavale ikoonile, mis avab modaalakna. Kustutamine on lubatud administraatorile (Lisa 16).

Üritusel osalemine

Autentitud kasutajal on võimalus märkida üritusel osalemine või osalus tühistada. Osalemise muutmine on võimalik ainult tulevaste ürituste korral. Toimingut saab teha nii

ürituste üldvaates kui ka konkreetse ürituse detailvaates (Lisa 13). Administraatorid näevad ka üritusel osalevate inimeste arvu, nimesid ja emaili aadresse.

3.4.2 Inventari vaade

Inventari vaade võimaldab rakenduse külastajatel tutvuda olemasolevate mängudega ning administraatori rolliga kasutajatel ka mängu luua, muuta ja kustutada (Lisa 9).

Mängu lisamine ja redigeerimine

Administraator saab uue mängu lisamiseks vajutada „Lisa mäng“ nuppu (Lisa 17). Vormil on järgmised väljad:

- Mängu nimi (eesti ja inglise keeles) - kohustuslik
- Kirjeldus (eesti ja inglise keeles) – vabatahtlik
- Kogus (täisarv, ei tohi olla negatiivne) - kohustuslik
- Minimaalne ja maksimaalne mängijate arv (täisarvud; maksimaalne ei tohi olla väiksem minimaalsest) - vabatahtlik
- Mängu kestus minutites (ei tohi olla negatiivne) - vabatahtlik
- Žanr – vabatahtlik, valitav olemasolevate žanrite hulgast
- Raskusaste – vabatahtlik, valitav olemasolevate raskusastete hulgast

Redigeerimine on võimalik kõigi mängude puhul ning sellele pääseb redigeerimisikooni kaudu.

Mängu kustutamine

Mängu kustutamiseks tuleb vajutada vastavale ikoonile, mis avab modaalakna. Kustutamine on lubatud administraatorile.

3.4.3 Tõlgete vaade

Tõlgete haldusvaade on kättesaadav ainult administraatoritele ja võimaldab hallata süsteemis kasutatavaid tõlkevõtmeid ning nende väärtuseid. Tõlked on struktureeritud tõlkevõtmete kaupa ning igale võtmele võib olla seotud mitu tõlget erinevates keeltes (Lisa 18).

Tõlgete sirvimine ja filtreerimine

Lehe ülaosas asub filtreerimisblok, kus kasutaja saab tõlkeid otsida tõlkevõtme alusel.

Tabeli igas reas kuvatakse:

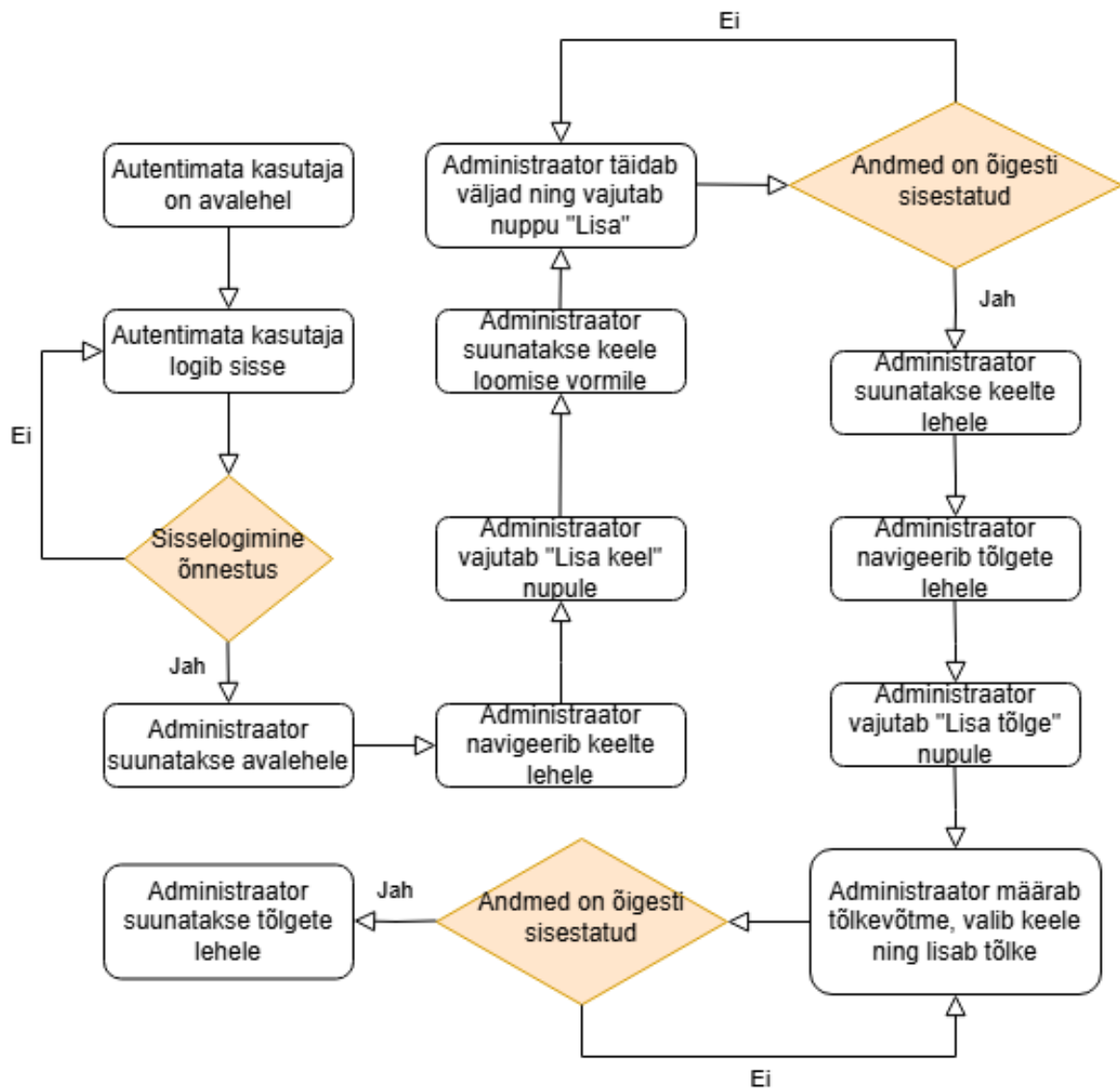
- Tõlkevõti
- Keelekoodipõhised tõlked igas keeles
- Kui mõne keele tõlge puudub, kuvatakse vastavas veerus pluss-ikoon, mille kaudu saab lisada uue tõlke.
- Redigeerimise nupp

Tõlke lisamine ja redigeerimine

Pluss-ikooni kaudu navigeeritakse tõlke lisamise vaatesse, kus on eelvalitud tõlkevõti ja keel, kuid kasutaja saab neid vajadusel muuta. Vormil tuleb täita:

- Tõlkevõti – kohustuslik
- Keel – kohustuslik
- Tõlge – kohustuslik

Olemasolevate tõlgete puhul kuvatakse tabelis iga keele jaoks muutmise ikoon. Sellega pääseb redigeerimisvaatesse, kus saab muuta tõlkesisu ning vajadusel tõlke ka kustutada (Lisa 19). Tõlke lisamise töövoog on välja toodud alloleval joonisel (Joonis 2).



Joonis 2. Tõlke loomine

Tõlke kustutamine

Kustutamine toimub standardse töövoo alusel: vajutades kustutamise ikoonile avaneb modaalaken, kus saab toimingut kinnitada või katkestada (Lisa 20).

3.4.4 Keelte vaade

Keelte lisamisel ja muutmisel peab kasutaja sisestama:

- Keele nimi – kohustuslik
- Keele kood – kohustuslik
- Aktiivne (jah/ei) – vaikimisi aktiivne

Kui keel on märgitud aktiivseks, kuvatakse see navigatsiooniriba keelevaliku rippmenüüs. Mitteaktiivseid keeli keelevalikus ei kuvata, kuid neid saab endiselt hallata administraatorivaates (Lisa 23).

3.4.5 Žanrite vaade

Žanrite haldamise vaade võimaldab administraatoritel hallata mängudele omistatavaid žanreid. Kasutajaliidese kaudu navigeerimine on piiratud – ainult autoriseeritud rollid pääsevad vaatesse, muul juhul suunatakse kasutaja tagasi avalehele. Autentimata kasutajal ning tavakasutajal puudub ligipääs.

Žanri lisamine ja redigeerimine

Administraator saab uue žanri lisamiseks vajutada „Lisa žanr“ nuppu (Lisa 21). Vormil on järgmised väljad:

- Žanri nimi (eesti ja inglise keeles) – kohustuslik

Žanri valimisel läbi otsinguriba suunatakse kasutaja žanri detailvaatesse, kus on võimalik redigeerida nime. Kehtib sama valideerimisreegel nagu lisamisel.

Žanri kustutamine

Žanri valimisel läbi otsinguriba suunatakse kasutaja žanri detailvaatesse, kus on võimalik redigeerida nimetusi ja kustutada žanr.

3.4.6 Asukohtade vaade

Asukohtade haldamise vaade on kättesaadav ainult administraatoritele (Lisa 24). Asukoha lisamisel ja muutmisel tuleb täita järgmised väljad:

- Nimi (eesti ja inglise keeles) – kohustuslik
- URL (nt Google Maps link) – vabatahtlik

3.4.7 Kasutajate haldamise vaade

Kasutajate haldusvaade on kättesaadav ainult super-administraator rolliga kasutajale. Vaate eesmärk on pakkuda mugavat liidest kasutajakontode haldamiseks, eelkõige rollide määramiseks ja kontode kustutamiseks (Lisa 25).

Rollide haldamine

Iga kasutaja kaardil kuvatakse tema täisnimi, e-posti aadress ja roll. Kui kasutajal on *User* roll, saab talle määrata administraator rolli vastava nupuga. Kui kasutaja on administraator, saab temalt administraatoriõiguse eemaldada.

Konto kustutamine

Kasutaja saab kustutada ainult siis, kui tal ei ole administraator rolli. Kustutamiseks tuleb vajutada vastavale ikoonile, mille järel avaneb kinnitamist vajav modaalaken.

4 Töö tulemused

Peatükis antakse ülevaade arendatud veebirakenduse tehnilisest teostusest. Kirjeldatakse ees- ja tagarakenduse ülesehitust ning valminud funktsionaalsust. Eraldi käsitletakse rakenduse serveripoolset loogikat ja selgitatakse, millised süsteemi komponendid realiseeriti ning kuidas erinevad kihid omavahel suhtlevad.

4.1 Eesrakendus

Töö käigus loodud eesrakenduse ülesanne on toetada Mänguklubi tegevusi. Rakendus võimaldab rollipõhist ligipääsu ning haldusfunktsionaalsusi administraatoritele. Rakendus on arendatud Next.js raamistikus ja kirjutatud TypeScripti keeles.

4.1.1 Arhitektuur

Eesrakendus on üles ehitatud Next.js App Routeri baasil, mis võimaldab failipõhist marsruutimist. Rakenduse lähtekood paikneb src/ kataloogis ning on jaotatud loogilisteks plokkideks:

- app/ – lehed ja alamlehed
- components/ – taaskasutatavad React komponendid
- stores/ – MobX
- utils/ – utiliidid ja abifunktsioonid
- constants/ – püsiväärtused
- styles/ - disainireeglid
- middleware.ts – autoriseerimise eelkontroll (URL-põhine)
- app/api/ – NextAuth.js autentimissüsteemi konfiguratsioon

Eesrakendus suhtleb backendiga Axios teegi kaudu, mis võimaldab HTTP põhist suhtlust ja JWT autentimise toe rakendamist [19].

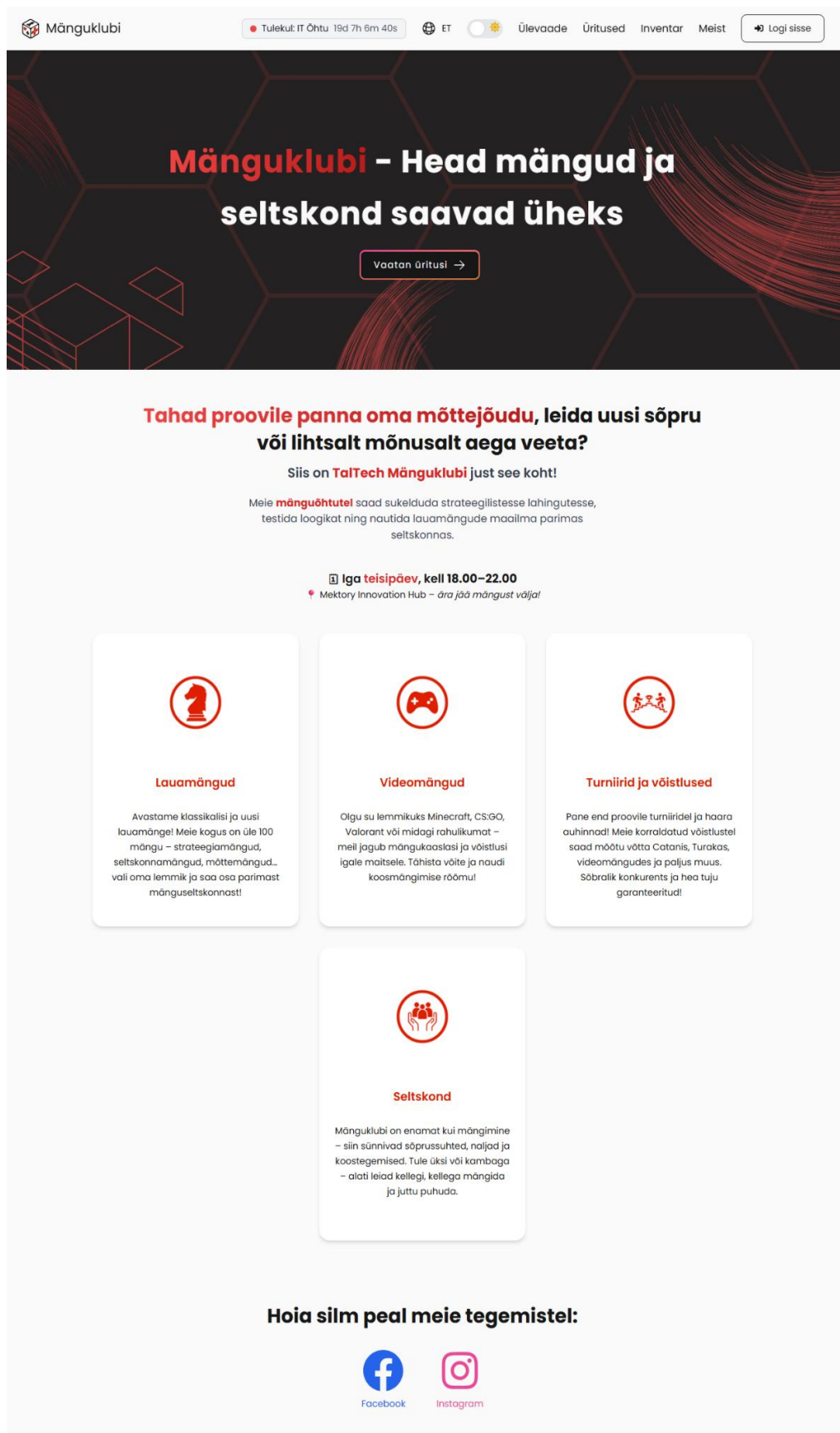
Kasutajate autentimine on lahendatud NextAuth.js abil. Vahetarkvara vastutab URL põhise marsruutimise autoriseerimise eest ning vajadusel suunab kasutaja kodulehele [20].

Rakenduse seisundihaldus toimub MobX teegi abil, mis võimaldab hallata olekuid lokaalselt ja globaalselt ilma liigse koodita. MobX-i andmehoidlaid (*store*) kasutatakse nii ürituste ja mängude halduses kui ka admini vaadete puhul [21].

Kujundus on loodud Tailwind CSS abil, mis võimaldab kiiresti ehitada modulaarseid, mobiilisõbralikke ja ligipääsetavaid kasutajaliideseid. UI-komponentidena kasutatakse ka HeadlessUI elemente ja näiteks ikooniteeki Heroicons [22], [23]. Osad ikoonid ning taustad on kujundatud graafilise disaini tööriista Canva-ga [24].

4.1.2 Kasutajaliides

Eesrakenduse jaoks said realiseeritud kõik funktsionaalsed ning mittefunktsionaalsed nõuded. Rakendus annab kasutajatele ülevaate Mänguklubi tegevusvaldkondadest (Joonis 3), eesmärkidest, üritustest ning mängudest. Samuti saavad administraatorid läbi rakenduse graafilise kasutajaliidese mugavalt sisu redigeerida. Kõik vaated, mis demonstreerivad funktsionaalsuste realiseerimist, on leitavad Lisades 5 kuni 26.



Joonis 3. Mänguklubi avaleht

4.2 Tagarakendus

Töö käigus loodud tagarakenduse ülesanne on hallata kogu süsteemi äriloogikat ning korraldada andmevahetust andmebaasiga. Rakendus on üles ehitatud .NET raamistikul ja kirjutatud C# programmeerimiskeeles. Andmebaasina kasutatakse PostgreSQL-i.

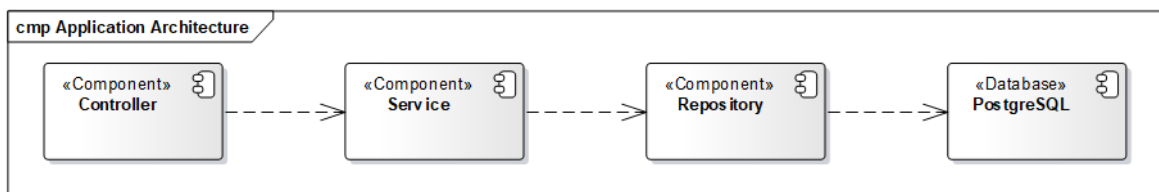
4.2.1 Arhitektuur

Tagarakendus järgib kihilise arhitektuuri põhimõtteid, mille eesmärk on jagada vastutus loogiliselt eristatud kihtidesse [12]. Selline struktuur toetab süsteemi laiendatavust, testitavust ja hooldatavust.

Arhitektuur koosneb neljast põhikomponendist:

- Kontrollerikiht (rakenduses projekti nimetus API)
- Teenusekiht (rakenduses projekti nimetus Application)
- Repositooriumkiht (rakenduses projekti nimetus Infrastructure)
- Andmebaasikiht tugineb PostgreSQL andmebaasile.

Kihid on illustreeritud joonisel 4, kus on kujutatud ka omavahelised suunatud suhtluse seosed.



Joonis 4. Tagarakenduse arhitektuur

4.2.2 Kontrollerid

Kontroller on rakenduse kiht, mis võtab vastu kasutajalt saabuvad HTTP päringud ja suunab need edasi teenusekihile töötlemiseks. Kontroller ei sisalda äriloogikat, vaid vastutab päringute vahendamise ja vastuste tagastamise eest. Joonisel 5 on esitatud ka näide kontrollerklassist ja ühest otspunktist.

```

[Route("api/[controller]")]
[ApiController]
public class EventController : ControllerBase
{
    private readonly IEventService eventService;

    public EventController(IEventService e)
    {
        eventService = e;
    }

    [AllowAnonymous]
    [HttpGet("{id}")]
    public async Task<ActionResult<Event>> GetEvent(int id)
    {
        var result = await eventService.GetByIdAsync(id);
        return result.Success ? Ok(result.Data) : NotFound(result);
    }
}

```

Joonis 5. Kontrolleri koodi näide

4.2.3 Teenused

Teenusekiht sisaldab süsteemi ärioloogikat ja vastutab andmete töötlemise eest. Teenuseklassid täidavad rakenduse põhifunktsioone, kasutades sisendina kontrolleriist saadud andmeid ning suheldes andmete salvestamiseks repositooriumkihiga. Joonisel 6 on esitatud on esitatud ka näide teenusklassist ja ühest meetodist.

```

public class EventService : IEventService
{
    private readonly IEventRepository eventRepository;
    private readonly IRepository<Location> locationRepository;

    public EventService(IEventRepository e, IRepository<Location> l)
    {
        eventRepository = e;
        locationRepository = l;
    }

    public async Task<ServiceResponse> DeleteAsync(int id)
    {
        var existingEvent = await eventRepository.GetByIdAsync(id);
        if (existingEvent == null)
            return ServiceResponse.ErrorResult(Constants.notFound);

        await eventRepository.DeleteAsync(id);

        return ServiceResponse.SuccessResult(Constants.deleted);
    }
}

```

Joonis 6. Teenuse klassi näide

4.2.4 Repositooriumid

Repositooriumkiht vastutab andmete pärimise, salvestamise, muutmise ja kustutamise eest. Repositooriumiklassid suhtlevad otse andmebaasiga ning pakuvad teenusekihile ühtset ligipääsuliidest andmete haldamiseks, ilma et rakendaksid äriloogikat. Joonisel 7 on esitatud näide repositooriumklassist ja ühest meetodist, mis teostab ürituste andmete pärimist andmebaasist.

```

public class EventRepository : IEventRepository
{
    private readonly DataContext context;

    public EventRepository(DataContext c)
    {
        context = c;
    }

    public async Task<IEnumerable<Event>> GetAllAsync(
        string? searchQuery = null)
    {
        var query = context.Events.AsQueryable();
        return await query.ToListAsync();
    }
}

```

Joonis 7. Repositoorium klassi näide

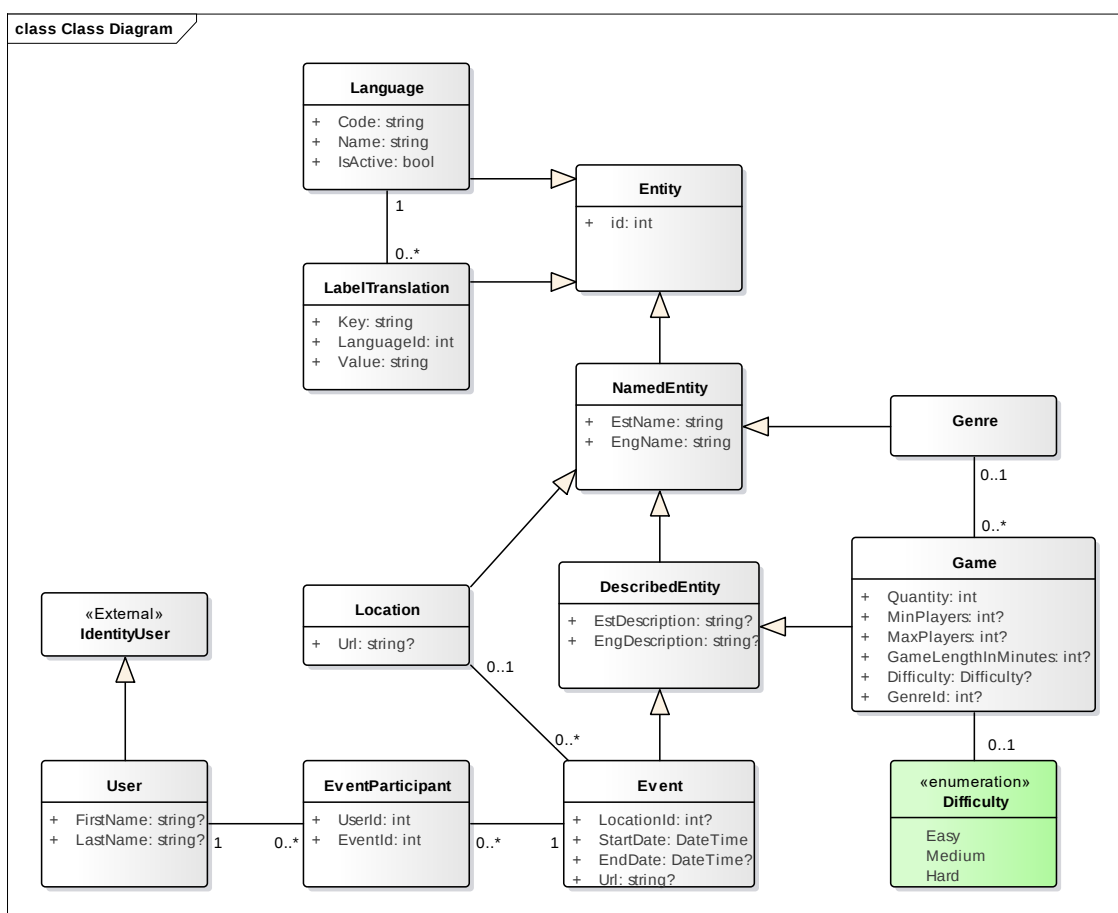
4.2.5 Klassid

Klassidiagramm kirjeldab süsteemi põhiklasside struktuuri ja seoseid (Joonis 8). Kõik objektid pärinevad baasklassist *Entity*, millel on ühine identifikaator *Id*. Sellest pärinevad *NamedEntity* ja *DescribedEntity*, mis lisavad mitmekeelsed nime ja kirjelduse väljad.

Game sisaldab infot mängu kohta ja võib olla seotud žanriga (*Genre*). Mängu keerukus määratletakse *Difficulty* enum kaudu (*Easy*, *Medium*, *Hard*).

Event sisaldab teavet ürituse kohta ning on seotud toimumispaigaga (*Location*). Kasutajad on seotud üritustega vaheklassi *EventParticipant* kaudu. Iga kasutaja põhineb *IdentityUser* klassil ning lisatud on juurde ka ees- ja perekonnanimi.

Mitmekeelsuse toetamiseks on olemas *Language* ja *LabelTranslation* klassid, mis võimaldavad kasutajaliidese elementide tõlkimist vastavalt keelele.



Joonis 8. Klassidiagramm

4.3 Testimine

Arenduse käigus loodi ühiktestid teenusekihile, sest sinna on loodud kogu rakenduse äriloogika. Töö raames otsustati testida teenuskihti, kuna autorid pidasid esmatahtsaks arendatud loogika korrapärasest toimimist. Teenuseid testiti isoleeritult, ilma ühenduseta andmebaasi või kontrolleritesse. Testide kirjutamisel kasutati xUnit raamistikku ning sõltuvuste simuleerimiseks Moq teeki.

Testimise eesmärk oli tagada, et teenused käituksid ootuspäraselt nii kehtiva kui ka vigase sisendi, puuduvate andmete ja piirjuhtumite korral. Testidega katsetati ka otsingu, lehekülgedeks jaotamise (*pagination*) ning mitmekeelsuse toimimist.

Kokku kirjutati teenusklassidele 115 testi, millest kõik läbisid edukalt ja ükski test ei nurjunud (Joonis 9).



Joonis 9. Ühiktestide tulemused

Testimise edukust saab vaadata ka koodikaetuse põhjal. Testitud said kõik teenusklassid ehk saavutati 100% koodikaetus teenuskihis. Kogu rakenduse üldine koodikattuvus jäi 66% tasemele (Joonis 10).

Symbol	Coverage (%) ▾	Uncovered/Total Stmts.
▼ Total	66%	980/2866
> Tests	100%	0/1061
▼ Application	92%	66/839
▼ Application	92%	66/839
▼ Services	100%	0/462
> AccountService	100%	0/103
> EventParticipantService	100%	0/52
> EventService	100%	0/68
> GameService	100%	0/59
> GenreService	100%	0/40
> LabelTranslationService	100%	0/62
> LanguageService	100%	0/38
> LocationService	100%	0/40
> ApiResponses	93%	2/30
> DTOs	82%	61/344
> Utils	0%	3/3
> Domain	78%	15/67
> API	0%	277/277
> Infrastructure	0%	622/622

Joonis 10. Koodikaetus ühiktestidega

Testide kirjutamisel järgiti nimestruktuuri, kus iga testinimi koosnes testitavast meetodist, olukorra kirjeldusest ja oodatud tulemusest [25]. See parendab testide loetavust ning kirjeldab ära, mida reaalselt testitakse. Joonisel 11 on esitatud üks näidis testmeetod, mis kontrollib, kas sündmuse uuendamine toimub edukalt korrektse sisendi korral.

```

public class EventServiceTests : BaseServiceTest<IEventService>
{
    private readonly Mock<IEventRepository> eventRepoMock;
    private readonly Mock<IRepository<Location>> locationRepoMock;

    private const int ValidId = 1;
    private const string UpdatedName = "Updated";

    public EventServiceTests()
    {
        eventRepoMock = Mocks.Create<IEventRepository>();
        locationRepoMock = Mocks.Create<IRepository<Location>>();
    }

    protected override IEventService GetService() =>
        new EventService(eventRepoMock.Object, locationRepoMock.Object);

    [Fact]
    public async Task
        UpdateAsync_WhenRequestIsValid_UpdatesEventSuccessfully()
    {
        // Arrange
        var existing = NewEvent();
        SetId(existing, ValidId);
        var request = new EventUpdateRequest { EstName = UpdatedName };

        eventRepoMock.Setup(r =>
            r.GetByIdAsync(ValidId)).ReturnsAsync(existing);
        eventRepoMock.Setup(r =>
            r.UpdateAsync(existing)).Returns(Task.CompletedTask);

        // Act
        var result = await Service.UpdateAsync(ValidId, request);

        // Assert
        Assert.True(result.Success);
        Assert.Equal(UpdatedName, result.Data!.EstName);
    }
}

```

Joonis 11. Testi näide

5 Järeldused

Peatükis käsitletakse kasutatud tööriistade sobivust, lahenduse vastavust püstitatud nõuetele ning kasutajatelt saadud tagasisidet. Lisaks hinnatakse arendusprotsessi kulgu, käsitletakse töö käigus ilmnunud probleeme ja esitatakse võimalused edasiseks arendamiseks. Peatüki lõpuosas tuuakse välja autorite isiklikud tegevused ja panused lõputöö valmimisse.

5.1 Tehnoloogilised tööriistad

Kasutatud tehnoloogilised tööriistad osutusid arendusprotsessi käigus sobivaks. Valitud raamistikud ja platvormid toetasid rakenduse ülesehitust ning võimaldasid realiseerida kõik kavandatud funktsionaalsused ilma suuremate takistusteta.

Tööriistade valiku õigsust kinnitab asjaolu, et süsteem valmis vastavalt nõuetele ja tähtaegselt. Arenduse jooksul ei tekkinud olukordi, mis oleksid nõudnud tehnoloogiate vahetamist või olulisi arhitektuurilisi ümberkorraldusi. Kõik valitud komponendid alates .NET-i ja Next.js-i raamistikest kuni PostgreSQL-i andmebaasini töötasid ootuspäraselt ja integreerusid omavahel hästi.

5.2 Lahenduse vastavus nõuetele

Autorid töötasid projekti alguses välja kõik esialgselt vajaminevad funktsionaalsed ja mittefunktsionaalsed nõuded koostöös Mänguklubi juhatusega. Töö tulemusena sai realiseeritud kõik kaardistatud funktsionaalsed nõuded FR-01 kuni FR-12 (Tabel 2) ning mittefunktsionaalsed nõuded NFR-01 kuni NFR-07 (Tabel 3).

Lisaks kaardistatud nõuetele arendati lisaks kasutajaliidese parendamiseks öö- ning päevarežiim, mis muudab lehekülje värviskeemi vastavalt kas tumedaks või heledaks (Lisa 5). Samuti sai lisatud järgmise ürituse indikaator navigatsiooniribal, mis võimaldab lehekülje külastajal näha järgmist või käimasolevat üritust (Lisa 26).

5.3 Tagasiside valminud arendusele

Arendustöö lõpus anti valminud rakenduse lokaalne versioon testimiseks kolmele inimesele, kes ei olnud arendusprotsessiga seotud. Nende ülesandeks oli hinnata rakenduse kasutusmugavust, navigeeritavust ja üldist funktsionaalsust.

Kõik rakenduse testijad suutsid iseseisvalt luua kasutajakonto, logida sisse ning orienteeruda lehe erinevates vaadetes. Rakenduse ülesehitust hinnati loogiliseks ja navigeerimine sujus probleemideta. Esmased eesmärgid, nagu üritustele registreerumine, olid lihtsasti teostatavad.

Kuigi kasutajaliidest peeti visuaalselt atraktiivseks, sooviti tagasiside põhjal siiski rakenduses näha rohkem illustratiivset materjali. Eriti toodi esile, et ürituste vaadet, kus pildid aitaksid kasutajal kiiremini mõista ürituse olemust ja sisu. Samuti leiti, et inventari vaade võiks sisaldada mängude visuaale, et anda paremat ülevaadet sellest, milliseid lauamängud klubis on olemas.

Lisaks toodi välja, et oleks hea kui kasutajal oleks ka oma isiklik profiilivaade, kus saaks näha isiklikke andmeid. Pakuti välja, et isiklikus vaates võiksid olla näha ka turniiride tulemused ja informatsioon osaletud üritustest.

5.4 Hinnang projektile

Veebirakenduse arendusprotsess kulges planeeritult ja ilma suuremate tagasilöökideta. Töö edenemisele aitas kaasa meeskonnaliikmete vaheline pidev suhtlus, samuti regulaarne koostöö juhendaja ja Mänguklubi juhatuse esindajaga. Arendusprotsessis suudeti paindlikult kohanduda nii funktsionaalsete kui ka mittefunktsionaalsete nõuete täpsustumisega, mis tagas, et lahendus arenes vastavalt tegelikele vajadustele.

Arendusprotsessis jaotati tööd võrdselt, arvestades meeskonnaliikmete tehnilisi tugevusi ning seniseid kogemusi. Projekti teostamise ajagraafik püsis kogu arendusperioodi jooksul kontrolli all ning planeeritud tähtajad suudeti täita.

Autorid jäid arendusprotsessiga rahule ning leiavad, et lõputöö raames püstitatud eesmärk sai täidetud. Realiseeritud funktsionaalsus vastab Mänguklubi esmastele vajadustele ning loob aluse edasiste arenduste ja laienduste jaoks. Täpsemad individuaalsed hinnangud on esitatud lõputöö Lisas 2 ja Lisas 3.

5.5 Probleemid ja katsumused

Arendusprotsessi vältel suuremaid tõrkeid ei esinenud. Kuna meeskonnaliikmed keskendusid vastavalt oma tehnilisele pädevusele, sujus tööjaotus tõrgeteta ning ülesannete täitmine toimus plaanipäraselt. Varasem kogemus valitud tehnoloogiatega võimaldas vältida tavapäraseid takistusi, mis võivad ilmned a renduskeskkonna ülesseadmisel.

Ainsaks arvestatavaks katsumuseks osutus sobiva tõlkelahenduse realiseerimine. Esialgsed lahendused ei pakkunud piisavat paindlikkust ega lihtsust kasutajaliidese tasandil. Selle tulemusena tuli tõlkelahendust kolmel korral ümber kujundada, et saavutada lahendus, mis võimaldaks kasutajaliidese poolel tõlkeid hallata võimalikult mugavalt ilma arendaja sekkumiseta. Ülejäänud arendusetapid kulgesid ootuspäraselt ning arendustöö viidi lõpuni ilma tagasilöökideta.

5.6 Võimalused edasiarenduseks

Arendatud veebirakendus võimaldab tulevikus edasiste funktsionaalsuste lisamist. Arvestades klubi tegevusvaldkonda ja testkasutajate tagasisidet, siis on mitmeid võimalusi, mida saaks potentsiaalselt juurde arendada.

Esimese loogilise sammuna saaks juurutada CI/CD lahenduse. Lahendus võimaldaks automatiseerida rakenduse testimise ja ehitamise protsessi, mis suurendaks arendustöö töökindlust.

Ühe funktsionaalsusena, mida Mänguklubi saaks edaspidi arendada, oleks turniiride statistikasüsteemi loomine. See võimaldaks igal kasutajal näha oma seniseid tulemusi, koguda turniiridel või etappidel punkte ning võrrelda oma saavutusi teiste liikmetega. Seda funktsionaalsust pakkusid välja ka testkasutajad, kes oleksid soovinud sellist informatsiooni näha ka enda profiili all.

Lisaks saaks täiendada visuaalset poolt illustratiivsete materjalidega. Testkasutajad märkisid, et ürituste ja mängude juures võiks rohkem olla pilte, et haarata rohkem tähelepanu. Piltide hoiustamiseks saaks kasutada kolmanda osapoole pilveteenuseid.

Liikmelisuse haldamise osas oleks võimalik rakendada kinnitamisprotsess, kus registreeritud kasutajad ei saaks enam automaatselt liikmestaatust, vaid nende liitumissoovi peaks eelnevalt kinnitama või tagasi lükkama administraator.

5.7 Meeskonnaliikmete tegevused

Lõputöö autorid hindavad teineteise panust veebirakenduse teostamisel võrdseks. Mõlema autori *commitide* ajalugu on esitatud Lisas 4.

5.7.1 Egert Andrejevi sisuline kokkuvõte

Järgnevalt on esitatud Egert Andrejevi tegevused nädalate kaupa (Tabel 4).

Tabel 4. Egert Andrejevi tööde sisuline kokkuvõte

Nädal	Tegevused
1	- Mänguklubi juhatusega esialgsete nõuete ja visiooni väljatöötamine - Tehnoloogiliste tööriistade valimine - Teiste tudengiorganisatsiooni veebilehtede analüüs
2	- Nõuete täpsustamine - Tööde ja vastutuslade täpne jaotamine autorite vahel - Esialgse tagarakenduse struktuuri ja andmebaasi ülesseadmine
3	- Ürituste CRUD realiseerimine - Autoriseerimise ja kasutajarollide realiseerimine
4	- Üritustel osalemise realiseerimine - Andmete valideerimise lisamine
5	- Veateadete ühtlustamine - Refaktoreerimine
6	- Rollihalduse väljatöötamine administraatoritele - Mängude (inventari) CRUD realiseerimine
7	- Pagineeritud vastuste loomine üritustele ja mängudele - Žanri CRUD realiseerimine - Asukoha CRUD realiseerimine - Otspunktide autoriseerimine
8	Kohtumine Mänguklubi juhatusega
9	- Koodiparandused - Päringute vastuste restruktureerimine

10	• Ürituste filtreerimise loomine
11	• Keelte CRUD realiseerimine • Esialgse tõlkestruktuuri realiseerimine • Eesti- ja inglisekeelne keeletugi olemisel
12	• Tõlkestruktuuri lihtsustamine ja ümber tegemine • Mängude filtreerimise loomine
13	• Ühiktestide loomine teenusklassidele • Kuupäevade normaliseerimine UTC kujule • Filtreerimise parandamine vastavalt uutele keeltele • Kasutajaliidese jaoks uute tõlkelahenduse otspunktide loomine
14	• Grupeeritud tõlgete otspunkt administraatoritele • Ürituste jaoks lisanduvate otspunktide loomine • Ühiktestide, kontrollerite ja teenusklasside viimine ühtlustatud kujule
15	• Andmebaasi tabelite käitumisviiside muutmine • Lõputöö dokumentatsiooni kirjutamine
16	• Lõputöö dokumentatsiooni kirjutamine

5.7.2 Sander Piiki sisuline kokkuvõte

Järgnevalt on esitatud Sander Piiki tegevused nädalate kaupa (Tabel 5).

Tabel 5. Sander Piiki tööde sisuline kokkuvõte

Nädal	Tegevused
1	• Mänguklubi juhatuses esialgsete nõuete ja visiooni väljatöötamine • Tehnoloogiliste tööriistade valimine • Teiste tudengiorganisatsioonide veebilehtede analüüs
2	• Tööde ja vastutusalade täpne jaotamine autorite vahel • Esialgse eesrakenduse seadistamine ning sõltuvuste installimine ja konfigureerimine
3	• Esialgsete staatiliste lehtede loomine
4	• API-ga liidestamine • Ürituste GET päringu realiseerimine • Esialgne autentimisega seotud loogika konfigureerimine • Registreerimisega seotud loogika implementeerimine
5	• Sisselogimise funktsionaalsuse implementeerimine

6	• Üritustega seonduva loogika implementeerimine • Esialgse ürituste vahelehe loomine
7	• Ürituste vahelehega seotud administraatori toimingute loogika implementeerimine
8	• Ürituste vahelehega seotud administraatori vaadete loomine
9	• Ürituste vahelehele filtreerimise ning pagineerimise lisamine
10	• Inventari vahelehe ning administraatori toimingute ja vaadete lisamine
11	• Sisendväljade valideerimisreeglite rakendamine • Kuupäeva filtri lisamine ürituste vahelehele
12	• Keelevaliku lisamine • Žanri ning raskusastme filtri lisamine mängude vahelehele • Osalejate nimekirja kuvamise loogika implementeerimine administraatorile
13	• Administraatorile mõeldud vaadete loomine: žanrid, asukohad, keeled, tõlked ning kasutajad • Kodulehe ning “Meist” vahelehele disaini parendused: mobiilivaate toe lisamine vahelehtedele, taaskasutatavate disainikomponentide loomine ning kujunduse üldine parendamine • Ürituste ning inventari vahelehtede disaini parendused: mobiilivaate toe lisamine vahelehtedele, taaskasutatavate disainikomponentide loomine ning kujunduse üldine parendamine
14	• Järgmise ürituse indikaatori lisamine • Lõputöö dokumentatsiooni kirjutamine
15	• Lõputöö dokumentatsiooni kirjutamine
16	• Lõputöö dokumentatsiooni kirjutamine

Kokkuvõte

Bakalaureusetöö eesmärk oli luua Tallinna Tehnikaülikooli tudengiorganisatsioonile Mänguklubi veebirakendus, mis koondaks klubi tegevused ühtsesse keskkonda. Rakendus võimaldaks lihtsustada Mänguklubi juhatuseliikmete tööd ning pakkuda liikmetele ja huvilistele ülevaadet organisatsiooni tegevustest.

Arenduse alguses analüüsiti teiste tudengiorganisatsioonide veebilehti, et saada ülevaade olemasolevatest lahendustest. Sellele järgnes Mänguklubi funktsionaalsete ja mittefunktsionaalsete nõuete kaardistamine. Arendusprotsess toimus agiilselt ja iteratiivselt ning autorid jaotasid töö vastutused vastavalt oma tugevustele. Rakenduse loomiseks kasutati .NET raamistikul ning C# programmeerimiskeelel põhinevat serveripoolset lahendust ning Next.js-ga loodi eesrakendus, andmete salvestamiseks kasutati PostgreSQL andmebaasi.

Lõputöö tulemusena valmis veebirakendus, mis võimaldab kasutajatel saada ülevaate organisatsiooni tegevustest, üritustest ning Mänguklubi mängudest. Ühtlasi võimaldab lahendus autentitud kasutajatel registreeruda üritustele. Samuti loodi administraatorile mõeldud vahelehed, mis võimaldavad veebirakendust hallata graafilise kasutajaliidesega.

Loodud veebirakendus täitis kaardistatud nõuded, mistõttu võib lugeda eesmärki täidetuks. Kuna võimalusi edasiarenduseks on palju, saab arendatud lahendusele tulevikus funktsionaalsusi juurde arendada.

Kasutatud kirjandus

- [1] Microsoft, „.NET Documentation,” Microsoft. [Online]. Available: <https://learn.microsoft.com/en-us/dotnet/>. Accessed: May 18, 2025.
- [2] Vercel, “Introduction,” Next.js. [Online]. Available: <https://nextjs.org/docs>. Accessed: May 18, 2025.
- [3] „TypeScript Documentation“, Microsoft. [Online]. Available: <https://www.typescriptlang.org/docs/>. Accessed: May 18, 2025.
- [4] Tailwind Labs Inc., “Get started with Tailwind CSS,” Tailwind CSS. [Online]. Available: <https://tailwindcss.com/docs>. Accessed: May 18, 2025.
- [5] The PostgreSQL Global Development Group, “PostgreSQL Documentation,” PostgreSQL. [Online]. Available: <https://www.postgresql.org/docs/>. Accessed: May 18, 2025.
- [6] K. Schwaber and J. Sutherland, The 2020 Scrum Guide™, 2020. [Online]. Available: <https://scrumguides.org/scrum-guide.html>. Accessed: May 11, 2025.
- [7] K. Beck and C. Andres, *Extreme Programming Explained: Embrace Change*, 2nd ed. Boston, MA, USA: Addison-Wesley Professional, 2004. [Online]. Available: <https://www.oreilly.com/library/view/extreme-programming-explained/0321278658/>. Accessed: May 11, 2025.
- [8] R. C. Martin, *Clean Code: A Handbook of Agile Software Craftsmanship*, Pearson Education, Harlow, 2008. [Online]. Available: <https://learning.oreilly.com/library/view/clean-code-a/9780136083238/>. Accessed: May 11, 2025.
- [9] R. C. Martin, *Agile Software Development: Principles, Patterns, and Practices*. Harlow, Essex, U.K.: Pearson, 2014. [Online]. Available: <https://dl.ebooksworld.ir/motoman/Pearson.Agile.Software.Development.Principles.Patterns.and.Practices.www.EBooksWorld.ir.pdf>. Accessed: May 18, 2025.
- [10] A. Hunt and D. Thomas, *The Pragmatic Programmer: Your Journey to Mastery*, 20th Anniversary Ed., 2nd ed. Boston, MA, USA: Addison-Wesley Professional, 2019. [Online]. Available: <https://learning.oreilly.com/library/view/the-pragmatic-programmer/9780135956977/>. Accessed: May 11, 2025.
- [11] M. Seemann and S. van Deursen, *Dependency Injection Principles, Practices, and Patterns*. Shelter Island, NY, USA: Manning Publications, 2019. [Online]. Available: <https://learning.oreilly.com/library/view/dependency-injection-in/9781617294730/>. Accessed: May 18, 2025.

- [12] M. Fowler, *Patterns of Enterprise Application Architecture*. Boston, MA, USA; Old Tappan, NJ, USA: Addison-Wesley Professional, Pearson Education [distributor], Nov. 2002. [Online]. Available: <https://learning.oreilly.com/library/view/patterns-of-enterprise/0321127420/>. Accessed: May 18, 2025.
- [13] IT-teaduskonna üliõpilaskogu. [Online]. Available: <https://ituk.ee>. Accessed: February 7, 2025.
- [14] MTÜK. [Online]. Available: <https://mtuk.ee>. Accessed: February 7, 2025.
- [15] Kultuuriklubi. [Online]. Available: <https://kultuuriklubi.ee>. Accessed: February 7, 2025.
- [16] Lapikud. [Online]. Available: <https://lapikud.ee>. Accessed: February 7, 2025.
- [17] TalTech Turundusklubi. [Online]. Available: <https://taltechturundusklubi.com>. Accessed: February 7, 2025.
- [18] MongoDB Inc., „Relational vs. Non-Relational Databases“. [Online]. Available: <https://www.mongodb.com/resources/compare/relational-vs-non-relational-databases>. Accessed: May 18, 2025.
- [19] “Getting started” Axios. [Online]. Available: <https://axios-http.com/docs/intro>. Accessed: May 18, 2025.
- [20] NextAuth.js, Vercel. [Online]. Available: <https://next-auth.js.org>. Accessed: May 18, 2025.
- [21] MobX. [Online]. Available: <https://mobx.js.org/README.html>. Accessed: May 18, 2025.
- [22] Headless UI, Tailwind Labs. [Online]. Available: <https://headlessui.com>. Accessed: May 18, 2025.
- [23] Heroicons, Tailwind Labs. [Online]. Available: <https://heroicons.com>. Accessed: May 18, 2025.
- [24] Canva, Canva. [Online]. Available: <https://www.canva.com>. Accessed: May 18, 2025.
- [25] jpreese, „Best Practices for Writing Unit Tests“ - .NET. [Online]. Available: <https://learn.microsoft.com/en-us/dotnet/core/testing/unit-testing-best-practices>. Accessed: May 18, 2025.

Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks¹

Autorid Egert Andrejev ja Sander Piik

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

20.05.2025

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

Lisa 2 – Egert Andrejevi eneseanalüüs

Lõpuprojekti tegime koos Sander Piikiga ning minu vastutusalaks oli tagarakenduse arendamine ja äriloogika implementeerimine. Projekti tegemisel oli väga hea, et vastutusalad said kohe alguses ära jaotatud. See aitas mõlemal olla oma osa eest vastutav ja tagas mõlemapoolse aktiivse panustamise projekti, sest eesrakenduse ja tagarakenduse koostöö eeldas pidevat suhtlust.

Töö üheks keerulisemaks osaks osutus ülesannete prioritseerimine ning projekti ulatuse täpne määratlemine. Kuna varasem kokkupuude mahuka rakenduse planeerimisega puudus, kuid saime lahenduse õigeaegselt valmis, pean valminud lahendust edukaks.

Kuigi ma ei vastutanud eesrakenduse arenduse eest, pean kogu tööprotsessi kõige arendavamaks osaks koostööd Sanderiga, kes vastutas kasutajaliidese loomise eest. Tema vajaduste põhjal tuli mul luua vastavad otspunktid tagarakenduses. Eriti kasulik oli jälgida, kuidas Sander kasutajaliidest üles ehitas, sest see aitas mul oluliselt paremini mõista eesrakenduse loogikat ning selle seoseid rakenduse üldise arhitektuuriga.

Tagantjärele hinnates oleks võinud suuremat tähelepanu pöörata tavakasutaja funktsionaalsuste arendamisele. Kuigi administraatori funktsioonide katmine oli esialgu prioriteetne ja vajalik, oleks hea olnud pakkuda ka tavakasutajale rohkem võimalusi süsteemiga suhestumiseks. Usun, et see oleks suurendanud rakenduse kasutusväärtust.

Kõige sujuvamalt kulges tagarakenduse struktuuri loomine. Selle aluseks oli varasem praktiline kogemus ülikoolivälistest projektidest ning ka teoreetiline baas, mis oli omandatud ülikooliõpingute käigus. Kuigi ülikoolis saadud teadmised andsid tugeva aluse, osutasid reaalseks rakendamiseks eriti väärtuslikuks just varasemad praktilised kogemused, mis aitasid töö edukalt lõpule viia.

Lisa 3 – Sander Piiki eneseanalüüs

Lõpuprojekti tegime Egert Andrejeviga ning rollid said määratud töö alguses lähtuvalt varasemale kogemusele tehnoloogiatega. Minu vastutusalaks oli eesrakenduse arendamine ning selleks vajaminevate tööriistade ja tehnoloogiate konfigureerimine.

Suurimaks väljakutseks kujunes esialgselt töömahtude planeerimine, kuna mul puudus varasem kogemus nullist arendamisega. Arenduse käigus tuli esialgseid nõudeid ning töömahtu vastavalt tegelikult vajadusele kohandada. Rakenduse funktsionaalsuste osas kujunes kõige keerulisemaks tõlkelahenduse välja mõtlemine ning realiseerimine. Seda sai ka töö käigus korduvalt kohendada, kuid õnneks aktiivne suhtlus töö kaasautoriga võimaldas probleemidele leida kiireid lahendusi.

Kõige arendavamaks osaks töö juures pean esialgse rakenduse püsti seadmist, tehnoloogiate valikut ning vajaminevate tööriistade konfigureerimist. Näiteks sain konfigureerida autentimislahendust NextAuth.js, millega varasem kokkupuude puudus, kuid mida kindlasti kavatsen ka tulevastes projektides kasutada selle mugavuse ning paindlikkuse pärast. Samuti puudus arenduskogemus Next.js-ga, kuid kuna React-i olin arendustöodes kasutanud, siis vajaminevate detailide selgeks tegemine kujunes oodatust palju lihtsamaks.

Rohkem oleks võinud keskenduda töös tavakasutaja funktsionaalsuste arendamisele ning ka autentimislahenduse parendamisele. Kuna arendustöös keskenduti suuresti ka administraatorivaadetele ja nendega seotud funktsionaalsuste realiseerimisele, jäi vähem aega lisavõimaluste arendamiseks tavakasutajale. Näiteks oleks soovinud lisada e-posti valideerimise ning parooli vahetamise funktsionaalsuse. Õnneks tegeletakse rakenduse arendamisega aktiivselt edasi ka lõputööjärgselt, seega lisafunktsionaalsusi arendatakse iteratiivselt juurde.

Omandatud teadmistest ülikoolis ning ülikoolivälised arendusprojektid andsid väga hea aluse lõpuprojekti realiseerimiseks. Eesrakenduse arendamiseks jäid ülikoolist omandatud teadmised pigem väheks, kuid õnneks ülikoolivälised projektid aitasid omandada piisava vilumuse lõpuprojekti realiseerimiseks. Samas andsid ülikoolis omandatud arendusprintsüübid kindlust ja suuna arendamisel ning aitavad kindlasti kaasa ka tulevastes projektides.

Lisa 4 – Versioonihaldussüsteemi väljavõte

Egert Andrejevi *commitid*

c08ed08	Egert Andrejev	06.05.2025	get next event status updates
1912087	Egert Andrejev	05.05.2025	changes to db relations
0534a58	Egert Andrejev	03.05.2025	reorganize service
cc49f1a	Egert Andrejev	03.05.2025	controller methods readjusting
126fe81	Egert Andrejev	03.05.2025	all tests follow Given_When_Then style
5a7a1dd	Egert Andrejev	03.05.2025	all service layer cases are tested (115)
0bb7de7	Egert Andrejev	03.05.2025	updating restrictions for event and participation
bab8a07	Egert Andrejev	03.05.2025	get next event endpoint
53d74f3	Egert Andrejev	03.05.2025	get all user search fix
6f7fe6a	Egert Andrejev	29.04.2025	grouped translations for table view
9794034	Egert Andrejev	28.04.2025	language attached
b7a7f43	Egert Andrejev	27.04.2025	get all translations by key
f8f63f9	Egert Andrejev	27.04.2025	removed unnecessary url validation feat
7febfae	Egert Andrejev	27.04.2025	update changes for location/genre
0bd9b97	Egert Andrejev	27.04.2025	remove //

95b5a88	Egert Andrejev	27.04.2025	refactor
7041750	Egert Andrejev	27.04.2025	Merge remote-tracking branch 'origin/10-endpoint-tests'
41d7e5a	Egert Andrejev	27.04.2025	test
e2b35de	Egert Andrejev	25.04.2025	refactor
b7459b7	egandr	24.04.2025	Merge branch '10-endpoint-tests' into 'main'
6b46c69	Egert Andrejev	25.04.2025	user and login responses
940f536	Egert Andrejev	24.04.2025	location url
a695304	Egert Andrejev	24.04.2025	url to events
673f8e2	Egert Andrejev	23.04.2025	game service tests finalized
792c87f	Egert Andrejev	23.04.2025	fixed eventparticipation interface
bbf2e3a	Egert Andrejev	23.04.2025	event test cases all covered now
0dc5e4f	Egert Andrejev	23.04.2025	account service tests
0e05725	Egert Andrejev	23.04.2025	account service tests
5995aa6	Egert Andrejev	23.04.2025	event participant service tests
5b95d3a	Egert Andrejev	23.04.2025	label translation tests
c5e0a1a	Egert Andrejev	23.04.2025	language service tests
0f9d574	Egert Andrejev	23.04.2025	refactor event tests
4de0403	Egert Andrejev	23.04.2025	genre service tests
cde866e	Egert Andrejev	23.04.2025	location service tests
19ef5ef	Egert Andrejev	22.04.2025	game unit tests

eb2b101	Egert Andrejev	22.04.2025	base test and event service tests
4b93eb3	Egert Andrejev	21.04.2025	refactored
6af562e	Egert Andrejev	21.04.2025	endpoint that returns labeltranslation keys + id
94f1bdc	Egert Andrejev	21.04.2025	fix
3e7400a	Egert Andrejev	21.04.2025	normalize dates to UTC
7a17217	Egert Andrejev	21.04.2025	game search by language
d78ca91	Egert Andrejev	21.04.2025	events language based search filtering
a623332	Egert Andrejev	19.04.2025	fix usings
9ee63e6	Egert Andrejev	19.04.2025	refactor
185fd24	Egert Andrejev	18.04.2025	item entity name to game
1e4930a	Egert Andrejev	18.04.2025	items can be filtered by difficulty and genre
609883c	Egert Andrejev	18.04.2025	remove UILabel class
b014920	egandr	13.04.2025	Merge branch 'feat/est-eng-support' into 'main'
f1eecf2	Egert Andrejev	13.04.2025	est eng support for entities
90d79a2	Egert Andrejev	13.04.2025	refactor
8b5475f	Egert Andrejev	12.04.2025	all CRUDs working correctly
39bb215	Egert Andrejev	10.04.2025	Merge remote-tracking branch 'origin/main' into feat/est-eng-support
8707a79	Egert Andrejev	10.04.2025	labels structure amended

3a10f94	Egert Andrejev	09.04.2025	languages CRUD
542299f	Egert Andrejev	09.04.2025	added label translation endpoints
4e24489	Egert Andrejev	09.04.2025	fix
f403e5c	Egert Andrejev	09.04.2025	events can be filtered by start and end date
910244e	Egert Andrejev	30.03.2025	update
39f93bb	Egert Andrejev	30.03.2025	location and genre added to get
d29ec6c	egandr	29.03.2025	Merge branch '12-fix-tables' into 'main'
2845556	Egert Andrejev	29.03.2025	genre done
8915dc5	Egert Andrejev	28.03.2025	location added to event
f64cb33	egandr	16.03.2025	Merge branch '11-location-genre-tabels-crud' into 'main'
9367cd7	Egert Andrejev	16.03.2025	upgraded packages
1afc2bf	Egert Andrejev	16.03.2025	authorize endpoints
2fc93fc	Egert Andrejev	16.03.2025	update create request changes
cc200c1	Egert Andrejev	16.03.2025	improved location and genre table logic
77cc96f	Egert Andrejev	16.03.2025	location added
aa683a4	Egert Andrejev	16.03.2025	genre repo change
9197d4c	Egert Andrejev	13.03.2025	get all genres finished
7793ebe	Egert Andrejev	13.03.2025	fixed item validation

77572bd	Egert Andrejev	13.03.2025	Genre table, added new service interface
91a413d	egandr	12.03.2025	Merge branch '9-three-layer-role-logic' into 'main'
4deb8df	Egert Andrejev	12.03.2025	has next or previous page logic refactored
67cc7b8	Egert Andrejev	12.03.2025	event and item are now paginated
bfc35fa	Egert Andrejev	11.03.2025	paginated response for users
efe12a5	Egert Andrejev	09.03.2025	get all users with search and role filtering
8f85929	Egert Andrejev	09.03.2025	only users with role User can be deleted
6ab7c5c	Egert Andrejev	09.03.2025	revoke admin role
8dcedd5	Egert Andrejev	09.03.2025	modified roles
7b7afe1	Egert Andrejev	09.03.2025	refactor
b299957	Egert Andrejev	09.03.2025	account can only have 1 role at once
5834130	Egert Andrejev	09.03.2025	new role implemented
98293fb	egandr	06.03.2025	Merge branch '2-item-api' into 'main'
596bc93	Egert Andrejev	04.03.2025	Item CRUD implementation
7204fef	egandr	01.03.2025	Merge branch 'feat/eventparticipation' into 'main'
5f5b2f3	Egert Andrejev	01.03.2025	endpoint for events that user has registered in

661f697	Egert Andrejev	27.02.2025	error messages as constants
beead96	Egert Andrejev	27.02.2025	refactored eventparticipant service and repository layer
83114ed	Egert Andrejev	27.02.2025	added non-generic serviceresponse class for return clarity
f6a94aa	Egert Andrejev	27.02.2025	refactored participation methods
72e89bc	Egert Andrejev	27.02.2025	Event participant data for admin
8ceb8ca	Egert Andrejev	20.02.2025	Authorization for Eventparticipant endpoints
838346a	Egert Andrejev	20.02.2025	user validation
5c6eb33	Egert Andrejev	19.02.2025	checks if event exists
a32f31e	Egert Andrejev	19.02.2025	Event participation initial setup
49d1f28	egandr	16.02.2025	Merge branch '3-user-register-login' into 'main'
a5ed5d2	Egert Andrejev	16.02.2025	Login response change
45cbbab	Egert Andrejev	14.02.2025	Role names as const
ca8148b	Egert Andrejev	14.02.2025	Auth setup works
ac14c31	Egert Andrejev	14.02.2025	Added login and JWT
71b51ca	Egert Andrejev	13.02.2025	Account registration works
fef1790	Egert Andrejev	12.02.2025	Added User entity and registration dto
b680190	egandr	12.02.2025	Merge branch '1-events-api' into 'main'
cd5aa08	Egert Andrejev	10.02.2025	Event update dto

1cc58ad	Egert Andrejev	10.02.2025	Event create dto Validation handler
3b60c58	Egert Andrejev	08.02.2025	Service response
f6eec1e	Egert Andrejev	07.02.2025	events update fix
6379d81	Egert Andrejev	07.02.2025	Events repository service and controller
6ca21df	Egert Andrejev	07.02.2025	structure and database initial setup

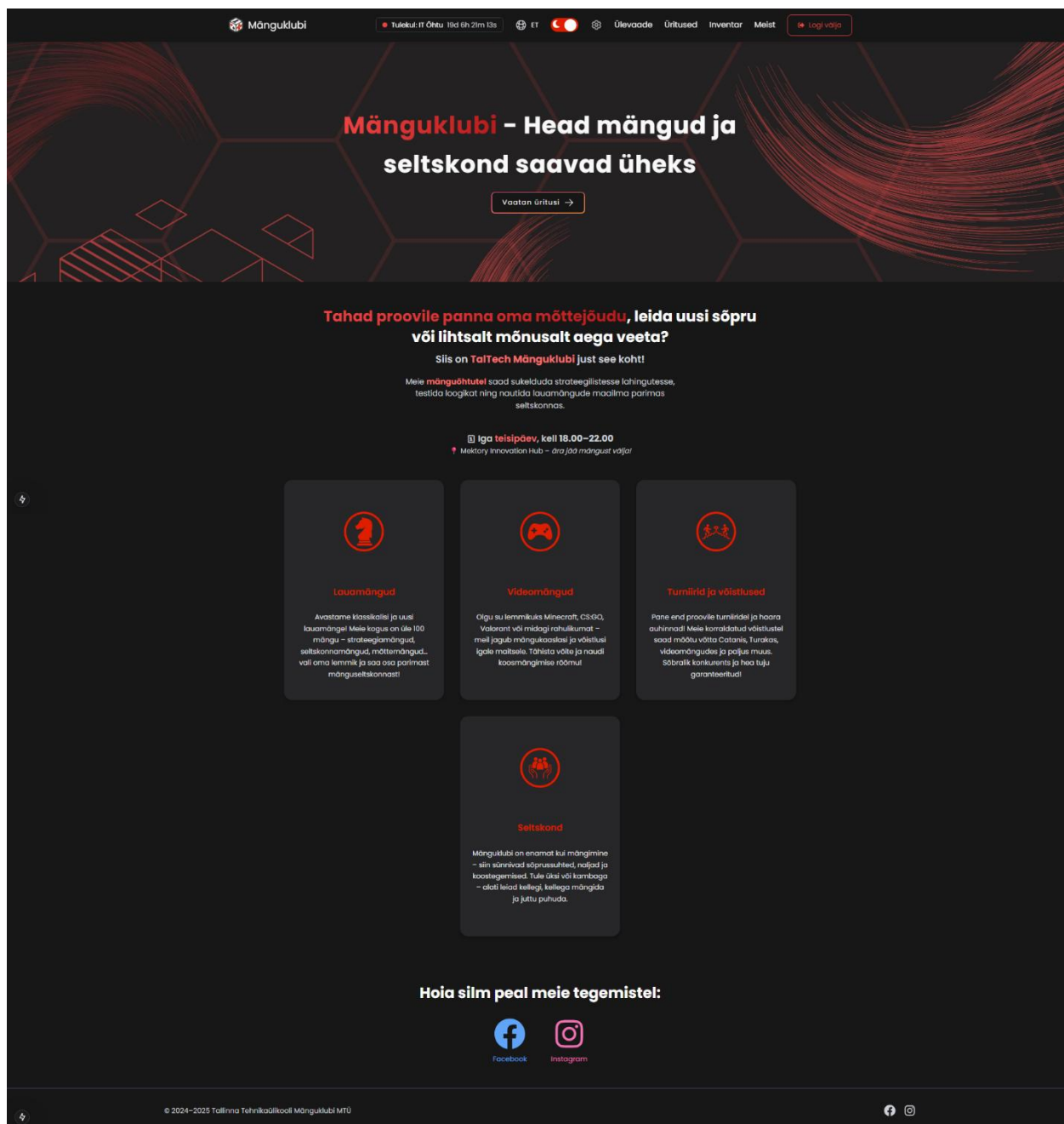
Sander Piiki *commitid*

c613c091	sapiik	17.05.2025	Merge branch 'feat/MK-00021' into 'main'
02624f47	sander piik	17.05.2025	fix: remove console logs
4a9514df	sander piik	17.05.2025	feat: refactor code
14888c6	sapiik	09.05.2025	Merge branch 'feat/MK-00020' into 'main'
0e094ec	sander piik	09.05.2025	fix: remove unnecessary file and fix readme
a6a3b5e	sapiik	09.05.2025	Merge branch 'feat/MK-00020' into 'main'
f24ff7d	sander piik	09.05.2025	feat: add next event ticker and fix readme
0f67961	sapiik	05.05.2025	Merge branch 'feat/MK-00019' into 'main'
ad4cd77	sander piik	05.05.2025	feat: add design to dynamic pages
fe841d4	sapiik	03.05.2025	Merge branch 'feat/MK-00018' into 'main'
a1337d5	sander piik	04.05.2025	feat: design dynamic pages
37a83d5	sapiik	03.05.2025	Merge branch 'feat/MK-00017' into 'main'
f4c8ba4	sander piik	03.05.2025	feat: design static pages
8508a99	sapiik	01.05.2025	Merge branch 'feat/MK-00016' into 'main'

ffb3499	sander piik	01.05.2025	feat: add admin operations
ab20f79	sapiik	23.04.2025	Merge branch 'fix/MK-00015' into 'main'
1f585c3	sander piik	23.04.2025	fix: refactor code
e9257fa	sapiik	23.04.2025	Merge branch 'feat/MK-00014' into 'main'
d94d428	sander piik	23.04.2025	feat: add participant section
2668855	sapiik	23.04.2025	Merge branch 'feat/MK-00013' into 'main'
d2b21cd	sander piik	23.04.2025	feat: add filter for games
3a8a684	sapiik	23.04.2025	Merge branch 'feat/MK-00012' into 'main'
a1322ce	sander piik	23.04.2025	feat: add language selection
be5df3d	sapiik	17.04.2025	Merge branch 'feat/MK-00011' into 'main'
5a045bd	sander piik	17.04.2025	feat: add date filter for event
1,19E+11	sapiik	14.04.2025	Merge branch 'feat/MK-00010' into 'main'
e212401	sander piik	14.04.2025	feat: add input field validations
90d6821	sapiik	12.04.2025	Merge branch 'feat/MK-00009' into 'main'
335604c	sander piik	12.04.2025	feat: create CRUD operations for inventory
3b6cf75	sapiik	02.04.2025	Merge branch 'feat/MK-00008' into 'main'
6624938	sander piik	02.04.2025	feat: added filtering and pagination
581fcfd	sapiik	25.03.2025	Merge branch 'feat/MK-00006' into 'main'
2b1712a	sander piik	25.03.2025	feat: added admin operations with initial design
bd6c986	sapiik	16.03.2025	Merge branch 'feat/MK-00005' into 'main'
1f6768e	sander piik	16.03.2025	feat: initial event subpage

b585d2a	sapiik	13.03.2025	Merge branch 'feat/MK-00004' into 'main'
ea36a98	sander piik	13.03.2025	feat: added events store
56a3afa	sapiik	09.03.2025	Merge branch 'feat/MK-00003' into 'main'
b39580b	sander piik	09.03.2025	feat: added login functionality
7dfb34e	sapiik	02.03.2025	Merge branch 'feat/MK-00002' into 'main'
71c539f	sander piik	02.03.2025	feat: created user context
1dccf84	sapiik	02.03.2025	Merge branch 'feat/MK-00001' into 'main'
cb11997	sander piik	02.03.2025	feat: api integration completed
e31ef9c	sapiik	24.02.2025	Merge branch 'feat/AboutUs' into 'main'
10dd080	sander piik	24.02.2025	feat: added about us section and changed contacts style
e466b56	sapiik	24.02.2025	Merge branch 'feat/Contact' into 'main'
c96b5a2	sander piik	16.02.2025	feat: contacts subpage
47e1ab8	sapiik	16.02.2025	Merge branch 'feat/HomePage' into 'main'
d3f707a	sander piik	16.02.2025	fix: added follow us section and improved design
434ec11	sander piik	09.02.2025	fix: configured eslint and added sections
d320981	sander piik	09.02.2025	feat: setup
c4ab2d1	sander piik	02.02.2025	Initial commit

Lisa 5 – Avaleht öövaates



Lisa 6 – „Meist“ vaheleht

Mänguklubi

Tulekut IT Ohtu 19d 7h 5m 28s

ET

Ülevaade

Üritused

Inventar

Meist

Logi sisse

Meist

Mänguklubi eesmärk on tuua kokku tudengid ja teised huvilised ning luua siiraid inimsuhteid.

Tunneme, et tänapäeval on liiga palju üksinda tegutsemist – üksinda mängimist, õppimist ja olemist. Meie loome koha, kus inimesed saavad tulla kokku ja tunda end osana millestki suuremast.

Usume, et parim viis selle eesmärgi täitmiseks on pakkuda võimalusi koos mängimiseks – olgu selleks laua-, mõtte- või videomängud.

Meie pidevalt laienev kollektsioon erinevaid meelelahutusvorme võimaldab igal huvilisel leida endale midagi sobivat. Tule ja uur, mida meil pakkuda on!

Juhatuse liikmed

Siim Tiivel
Juhatuse esimees
Mänguklubi sai alguse tundeist, et ülikoolis puudus koht, kus saaks regulaarselt kokku tulla ja lauamänge mängida. Täna on klubi kasvanud oma algsetest juurtest palju laiemaks, mistõttu pean tagama, et kõik teeksid täpselt, mis on nende roll ja ülesanded. Minu peamine fookus on Mänguklubi jätkusuutlikkuse ja järelkasvu kindlustamine.

Kermo Aart
Aseesimees
Olen olnud sellel teekonnal Siimuga kaasas esimesest päevast alates. Mõletan siiani eredalt, kuidas kõik alguse sai – esimesed suveõhtused koosolekud, kus arutasime Mänguklubi visiooni ja panime kirja põhikorra punkti. Täna olen Siimu parem käsi ja nõunik. Samuti olen võtnud südameosjaks toetada turundustegevusi ning suhelda teiste organisatsioonidega.

Sebastian Juhan Liiva
Arvutimängude liiga juht
Arvutimängude liiga ühendab sarnaste huvidega inimesi ning loob võimalusi uute sõprade leidmiseks ja mõnusaks ajaveetmiseks ka väljaspool iganädalasi mänguõhtuid. Valdanna eestvedajana jagan hea meelega oma lemmikut – Europa Universalis IV. Mind köidavad kõige enam strateegiamängud, mis pakuvad sügavust ja sisukaid väljakutseid. Minu lemmik lauamäng on Carcassonne.

Martin Mätkönen
Rahandusjuht
Liitusin klubiga, sest nägin selles suurt potentsiaali. Olen andnud oma panuse, et Mänguklubi kasvaks ja haaraks enda alla võimalikult erinevaid mängulike ning et klubi oleks piisavalt ressursside nende arendamiseks. Soovin näha, kuidas Mänguklubi aitab taaselustada Eesti e-sporti maastikku ja kujuneb mängukultuuri keskuseks kogu Baltikumis.

© 2024–2025 Tallinna Tehnikaülikooli Mänguklubi MTÜ

Lisa 7 – „Üritused“ vaheleht

Üritused

Otsi Üritust

Otsi...

dd/mm/yyyy

dd/mm/yyyy

Kõik

Otsi

Nulli filtrid

Lisa üritus

Nimi	Kuupäev	Asukoht	Tegevused
Kevadine Seminar	15.07.2025 12:00 – 15.07.2025 15:00	EBS	Otsi Üksikasjad
Mänguõhtu	01.07.2025 21:00 – 02.07.2025 00:00	Pärnu koolid2	Otsi Üksikasjad
Hackathon	10.06.2025 13:00 – 11.06.2025 21:00	Vijandi Kultuuriakade...	Otsi Üksikasjad
IT Öhtu	05.05.2025 21:00 – 06.05.2025 01:00	TallTech	Otsi Üksikasjad
Mingiiiiiiiiiiii AAASI	08.05.2025 17:37 – 08.05.2025 17:39	Tartu Ülikool	Otsi Üksikasjad

Loe rohkem

Lisa 8 – Ürituse detailvaade

Mänguõhtu

Kirjeldus

Toimub **iganädalane** mänguõhtu!

Algus

01.07.2025 15:00

Lõpp

01.07.2025 18:00

Asukoht

[Pärnu Kolledž](#)

Facebook:

<https://www.facebook.com>

Osalen

Lisa 9 – „Inventar“ vaheleht

Inventar

search-field-heading

Otsi mängu...

Raskusaste

-- Vali Raskusaste --

Žanr

Vali žanr...

otsi

Null filtrid

Lisa mäng

Nimi	Žanr	Raskusaste	Kogus	Tegevused
Pilet reisile	Strateegia	Keskmine	2	Uudinfo
Catan	Simulatsioon	Keskmine	4	Uudinfo
Plahvatavad kassid	Seiklus	Lühine	3	Uudinfo
7 imelist	Platvorm	Keskmine	2	Uudinfo
Carcassonne	Simulatsioon	Keskmine	4	Uudinfo

Loe rohkem

Lisa 10 – Mängu detailvaade

Pilet reisile	
Kogu kaarte ja hõiva raudteeliine.	
Kogus	Mängu kestus minutites
2	45 min
Minimaalne mängijate arv	Maksimaalne mängijate arv
2	5
Raskusaste	Žanr
Keskmine	Strateegia

Lisa 11 – Registreerimisvorm

Registreeru

Registreeru

* Registreerudes annad nõusoleku, et sinu nimi ja e-posti aadress on nähtavad ainult Mänguklubi administraatoritele ning neid ei jagata kolmandatele osapooltele.

Lisa 12 – Sisselogimisvorm

Logi sisse

Email

Parool

Logi sisse

Pole veel kontot? [Registreeru siin](#)

Lisa 13 – Üritusel osalemise indikaatorid

		Osalen	Lisainfo
---	---	------------------------	--------------------------

		Tühista	Lisainfo
---	---	-------------------------	--------------------------

Lisa 14 – Ürituse lisamise vorm

Lisa Üritus

EST Nimi *

ENG Nimi *

Alguskuupäev *



Algusaeg *



Lõppkuupäev



Lõppaeg



EST Kirjeldus

B / U •

ENG Kirjeldus

B / U •

Sotsiaalmeedia link

Asukoht

Vali asukoht... | ▾

Tagasi

Lisa

Lisa 15 – Ürituse redigeerimise vorm

Redigeeri üritust

EST Nimi *

ENG Nimi *

Alguskuupäev





Lõppkuupäev





EST Kirjeldus

B / U •

Toimub **iganädalane** mänguõhtu!

ENG Kirjeldus

B / U •

Weekly game night!

Sotsiaalmeedia link

Asukoht

Tegasi

Salvesta

Lisa 16 – Kustutamise kinnitamise modaalaken



Lisa 17 – Mängu lisamise vorm

Lisa mäng

EST Nimi *

ENG Nimi *

Kogus *

0

EST Kirjeldus

B / U •

ENG Kirjeldus

B / U •

Minimaalne mängijate arv

Maksimaalne mängijate arv

Mängu kestus (min)

Raskusaste

-- Vali Raskusaste --

Žanr

Vali žanr...

Tagasi

Lisa

Lisa 18 – „Tõlked“ vaheleht

Tõlked

Otsi tõlkevõtme järgi

Sisesta tõlkevõti...

Otsi

[Lisa tõlge](#)

Tõlkevõti	ET	EN	Tegevused
event.heading	Üritused	Events	
inventory.heading	Inventar	+	

Lisa 19 – Tõlke lisamise vorm

Lisa tõlge

Tõlkevõti *

Keel *

-- Vali keel --

Tõlge *

Lisa

Lisa 20 – Tõlke redigeerimise vorm

Muuda tõlget: event.heading

Eesti keel *

Üritused



English *

Events



Salvesta

Lisa 21 – Žanri lisamise vorm

Lisa žanr

Eesti nimi *

Inglise nimi *

Lisa

Lisa 22 – Žanri detailvaade

Žanri detailid

EST nimi *

ENG nimi *

KustutaSalvesta

Lisa 23 – Keele lisamise vorm

Lisa keel

Nimi *

Kood *

Aktiivne

☒

Lisa

Lisa 24 – Asukoha lisamise vorm

Lisa asukoht

EST nimi *

ENG nimi *

Asukoha link

Lisa

Lisa 25 – „Kasutajad“ vaheleht

Kasutajad

Kõik rollid ▼

Otsi 🔍

Nulli filtrid

Nimi	Email	Roll	Tegevused
Liis Teder	liis.teder@example.c...	Admin	Eemalda admin õigused
Mari Oja	mari.oja@example.c...	User	Anna admin õigused Kustuta kasutaja
Rasmus Vaher	rasmus.vaher@exam...	User	Anna admin õigused Kustuta kasutaja
Helen Rebane	helen.rebane@exam...	SuperAdmin	
Jaan Lepik	jaan.lepik@example...	Admin	Eemalda admin õigused
Karl Tamm	karl.tamm@example...	User	Anna admin õigused Kustuta kasutaja
Laura Saar	laura.saar@example...	Admin	Eemalda admin õigused
Markus Pärtel	markus.partel@exam...	Admin	Eemalda admin õigused

Lisa 26 – Järgmise ürituse indikaator

 Mänguklubi

Tulekul: IT Õhtu 17d 19h 38m 36s

 ET 

 Mänguklubi

Toimumas: IT Õhtu

 ET 