

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Kristofer Sokk 185796IAIB

Suure keelemudeli vestlusrakenduse realiseerimine

Bakalaureusetöö

Juhendaja: Ago Luberg

PhD

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: Kristofer Sokk

05.06.2025

Annotatsioon

Lõputöö eesmärk on luua avalik veebirakendus, mis võimaldab suhelda ChatGPT-laadse keeletehnoloogiamudeliga, millel on ligipääs ka Moodle'i õppekeskkonna andmetele. Teenuse põhiülesandeks on pakkuda mugavat ja turvalist kasutajaliidest, kuhu saavad sisse logida TalTechi tudengid ja töötajad ning pidada tasuta vestlusi tehisintellektiga. Süsteem toetab tudengeid Moodle'i kursuste läbimisel, võimaldades küsida kursusega seotud küsimusi, saada vastuseid ja juhiseid ning leida vajalikku teavet kiirelt ja lihtsalt. Lisaks aitab lahendus parandada õppeprotsessi efektiivsust ning toetab iseseisvat õppimist.

Töö käigus viidi läbi põhjalik nõuete analüüs ning arendati klient-server arhitektuuril põhinev veebirakendus. Klientrakendus loodi React raamistikus, serverrakendus Java Spring Boot raamistikus. Lisaks viidi läbi testimine ja valideerimine, et tagada süsteemi töökindlus ja kasutusmugavus. Valminud rakendus on kättesaadav aadressil <https://cs.taltech.ee/chatbot/>.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 19 leheküljel, 4 peatükki, 2 joonist, 3 tabelit.

Abstract

Development of Chat Application for Large Language Model

The aim of this thesis is to develop a public web application that enables users to interact with a ChatGPT-like language model, which also has access to data from the Moodle learning management system. The primary objective of the service is to provide a convenient and secure user interface, allowing students and staff of Tallinn University of Technology to log in and engage in free conversations with artificial intelligence.

The system is designed to support students in completing Moodle courses by enabling them to ask course-related questions, receive answers and guidance, and quickly find relevant information. In addition, the solution aims to improve the efficiency of the learning process and support independent study.

During the course of the thesis, a thorough requirements analysis was conducted, followed by the development of a web application based on a client-server architecture. The client application was implemented using the React framework, while the server application was developed with the Java Spring Boot framework. Comprehensive testing and validation were carried out to ensure the reliability and usability of the system.

The completed application is publicly available at <https://cs.taltech.ee/chatbot/>.

The thesis is in Estonian and contains 19 pages of text, 4 chapters, 2 figures, 3 tables.

Lühendite ja mõistete sõnastik

agiilne arendus	Agile arendusmeetod, mis keskendub iteratiivsele ja järkjärgulisele tarkvaraarendusele (<i>Agile Development</i>)
API	Rakendusliides (<i>Application Programming Interface</i>)
avalik võti	Krüptograafiline võti, mida kasutatakse andmete dekrüpteerimiseks ja autentimiseks (<i>Public Key</i>)
CI/CD	Jätkuv integratsioon ja pidev tarnimine (<i>Continuous Integration/Continuous Delivery</i>)
CSS	Stiililehtede keel, mida kasutatakse veebilehtede välimuse määramiseks (<i>Cascading Style Sheets</i>)
Discord bot	Discordi bot, mis võimaldab kasutajatel suhelda ja hallata Discordi servereid (<i>Discord Bot</i>)
Docker	Avatud lähtekoodiga konteinerite haldamise platvorm, mis võimaldab rakenduste isoleeritud keskkonnas käivitamist (<i>Docker</i>)
Docker compose	Tööriist, mis võimaldab hallata mitme konteineri rakendusi Dockeris (<i>Docker Compose</i>)
Endpoint	API lõpp-punkt, kuhu saadetakse päringud ja kust saadakse vastused (<i>Endpoint</i>)
Entra ID	Microsofti identiteedihalduslahendus, mis võimaldab autentimist ja autoriseerimist (<i>Microsoft Entra ID</i>)
GitLab	Veebipõhine versioonihaldussüsteem, mis võimaldab koodi haldamist ja koostööd (<i>GitLab</i>)
HTML	Hüperteksti märgistuskeel, mida kasutatakse veebilehtede struktuuri määramiseks (<i>Hypertext Markup Language</i>)
HTTP	Hüperteksti ülekande protokoll (<i>Hypertext Transfer Protocol</i>)
identsustõend	Kasutaja identiteedi tõend, mis sisaldab teavet kasutaja kohta, nagu e-posti aadress ja kasutajanimi (<i>ID Token</i>)
issues	Probleemid või ülesanded, mis vajavad lahendamist või täitmist (<i>Issues</i>)
Java	Objektorienteeritud programmeerimiskeel, mida kasutatakse laialdaselt tarkvaraarenduses (<i>Java</i>)
JWT	Java teek JWT töötlemiseks (<i>Java JSON Web Token</i>)
JPA	Java Persistence API, standardne Java API andmete püsivuse haldamiseks (<i>Java Persistence API</i>)

JS	JavaScript, programmeerimiskeel, mida kasutatakse veebilehtede interaktiivsuse loomiseks (<i>JavaScript</i>)
JUnit	Java testimise teek, mis võimaldab kirjutada ja käivitada automaatseid teste (<i>JUnit</i>)
JWT	JSON Web Token, standardiseeritud viis andmete edastamiseks turvaliselt veebirakendustes (<i>JSON Web Token</i>)
Kanban	Agile arendusmeetod, mis keskendub töövoos visualiseerimisele ja pidevale täiustamisele (<i>Kanban</i>)
LaTeX	Dokumentide koostamise süsteem, mida kasutatakse teaduslike ja tehniliste dokumentide kirjutamiseks (<i>LaTeX</i>)
ligipääsutunnus	Tunnus, mis annab õiguse API päringute tegemiseks (<i>Access Token</i>)
LTS	Pikaajalise toe versioon, mis saab turvauuendusi ja veaparandusi pikema aja jooksul (<i>Long Term Support</i>)
MSAL	Microsofti autentimise teek (<i>Microsoft Authentication Library</i>)
MVP	Minimaalne töötav toode. Rakenduse esimene skoop ainult töötamiseks vajalike osadega (<i>Minimum Viable Product</i>)
on-premise	Kohapealne server, mis hoiab rakendusi ja andmeid (<i>On-Premise</i>)
privaatvõti	Krüptograafiline võti, mida kasutatakse andmete allkirjastamiseks ja autentimiseks (<i>Private Key</i>)
REST	RESTful API, mis järgib REST arhitektuurstiili põhimõtteid (<i>Representational State Transfer</i>)
Session Storage	Veebilehitseja salvestusruum, mis säilitab andmeid seansi jooksul (<i>Session Storage</i>)
SPA	Ühe lehe rakendus, mis laadib kogu sisu ühel lehel ja muudab kasutajakogemuse sujuvamaks (<i>Single Page Application</i>)
süntaksi esiletõstmine	Teksti esiletõstmine, et muuta see kergemini loetavaks ja arusaadavaks (<i>Syntax Highlighting</i>)
Teek	Koodikomplekt, mis pakub funktsionaalsust ja mida saab kasutada rakenduste arendamisel (<i>Library</i>)
UniID	Universaalne identiteedihalduslahendus, mis võimaldab kasutajatel hallata oma identiteeti ja ligipääsu erinevatele teenustele (<i>UniID</i>)
uuendusluba	Luba, mis võimaldab küsida ligipääsutunnust (<i>Refresh Token</i>)
vahemälu	Andmete ajutine salvestamine, et kiirendada andmete töötlemist ja vähendada serveri koormust (<i>Cache</i>)
WebSocket	Protokoll, mis võimaldab reaalajas suhtlust kliendi ja serveri vahel (<i>WebSocket</i>)

Sisukord

1	Sissejuhatus.....	11
2	Analüüs.....	13
2.1	Probleemi kirjeldus	13
2.2	Mittefunktsionaalsed nõuded.....	14
2.3	Funktsionaalsed nõuded.....	16
2.4	Tehnilised nõuded	16
2.5	Võimekused	17
2.6	Valideerimine	18
2.6.1	Klientrakenduse ühiktestid	18
2.6.2	Serverrakenduse ühiktestid.....	18
2.6.3	Kasutajaliidese testimine	18
2.6.4	Vestluste testimine	19
2.6.5	Kasutajate tagasiside.....	19
2.7	Projekti ülesehitus.....	19
3	Tehnoloogiline ülevaade.....	20
3.1	Arhitektuur.....	20
3.2	Klientrakendus	21
3.3	Serverrakendus	22
3.4	Andmebaas	23
3.4.1	PostgreSQL	23
3.4.2	Tabelid.....	23
3.5	Autentimine ja autoriseerimine	24
3.6	Juurutamine	25
3.6.1	Docker	25
3.7	Põhiline funktsionaalsus	25
3.7.1	Vestluste pärimine	25
3.7.2	Vestluse loomine	26

3.7.3	Vestluse nime muutmine	26
3.7.4	Vestluse kustutamine	26
3.7.5	Sõnumi loomine.....	26
3.7.6	Vastussõnumi saamine	26
3.7.7	Vestluse WebSocketi ühendus	27
3.8	Kasutuslugude erijuhud ja haldus.....	27
3.8.1	Päringute protsess	27
3.8.2	Päringute vead	27
3.8.3	Rollid	27
3.9	Tulemuste valideerimine	27
4	Kokkuvõte.....	29
	Kasutatud kirjandus	30
	Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks.....	31
	Lisa 2 – Kasutajaliidese kuvatõmmised	32
	Lisa 3 – Küsitluse tulemused.....	35

Jooniste loetelu

Joonis 1. Arhitektuur.	20
Joonis 2. Chat tabel.	23

Tabelite loetelu

Tabel 1. Klientrakenduses kasutatud tehnoloogiad.....	21
Tabel 2. Serverrakenduses kasutatud tehnoloogiad.	22
Tabel 3. Chat tabeli veerud.....	23

1 Sissejuhatus

TalTechil on olemas mitmeid teenuseid, mis on loodud selleks, et pakkuda tehisintellekti (AI) mudelitele ligipääsu ning toetada ülikooli õppe- ja teadustööd. Hetkel on need teenused kättesaadavad peamiselt informaatika eriala tudengitele ja õppejõududele, kuid ülikooli strateegiline eesmärk on laiendada AI teenuste kasutusvõimalusi kõikidele tudengitele ja töötajatele. Tehisintellekti rakendused on kiiresti arenev valdkond, mis võimaldab õppijatel omandada uusi teadmisi, arendada probleemilahendusoskusi ning saada tuge õppetöös ja uurimistöös.

Tehisintellekti teenuste laiem kättesaadavus loob uusi võimalusi nii individuaalseks kui ka koostööpõhiseks õppimiseks. Näiteks saab tehisintellekti abil pakkuda personaalseid soovitusi, toetada iseseisvat õppimist ning aidata lahendada keerukamaid ülesandeid. Lisaks on loodud ühendused teiste ülikooli infosüsteemidega, nagu Moodle, mis võimaldab pakkuda tudengitele kursusepõhist infot ja tuge vastavalt nende individuaalsetele vajadustele. Selline integreeritud lähenemine aitab parandada õppeprotsessi kvaliteeti ja tõhusust.

Käesoleva lõputöö eesmärk on arendada uus veebirakendus, mis võimaldab kõikidel TalTechi tudengitel ja töötajatel turvaliselt sisse logida ning suhelda tehisintellektiga. Rakendus võimaldab pidada vestlusi, küsida kursustega seotud küsimusi ning saada kiireid ja asjakohaseid vastuseid. Eesmärgiks on luua kasutajasõbralik ja töökindel lahendus, mis toetab õppimist ning aitab kaasa ülikooli digipöördele.

Töö käigus määratletakse arendatava veebirakenduse MVP [1] (ingl *minimum viable product*) ehk minimaalne funktsionaalsus, mis võimaldab süsteemi kasutusele võtta ja testida selle sobivust reaalsetes tingimustes. Lõputöös käsitletakse rakenduse arhitektuuri, kasutatud tehnoloogiaid, arendusprotsessi ning testimist ja valideerimist. Töö tulemusena valmib veebirakendus, mis on avalikult kättesaadav ning valmis kasutamiseks TalTechi tudengite ja töötajate poolt.

Oskusterminite eesti keelde tõlkimisel on kasutatud Cybernetica AKIT tõlkeid, mis on

avalikult kättesaadavad aadressil [2].

2 Analüüs

Töö on osa suuremast juturoboti projektist, mis on pooleli. Selle projekti ülesandeks on luua LLM-põhine raamistik, mis ühendub teiste ülikooli süsteemidega ja on kasutatav võimalikult paljudele ülikooli tudengitele ja töötajatele. Selle projekti raames on juba olemas erinevad süsteemid, mis suhtlevad OpenAI teenusega, ühenduvad Moodle API külge ja koguvad andmeid erinevatest süsteemidest kokku.

Esimene süsteem on nimega Context Provider. See on suuteline suhtlema Moodle-ga ja pakkuma erinevatele vestlustele konteksti. See teab, mis on kasutaja UniID, mis kursustel kasutaja on, mis on kodused tööd, mis nende sisu on, ja veel mitmeid infokilde.

Teine süsteem on Interaction Manager. See vahendab sõnumeid ja infot erinevate süsteemide vahel. Kasutaja poolt sõnumid liiguvad OpenAI API-sse, kust tulevad vastu sõnumid vastustega. Kui OpenAI annab teada, et oleks vaja konteksti Moodle-st, siis Context Provider süsteem annab selle info. Sõnumid käivad nii edasi-tagasi mitu korda ja valmis vastused jõuavad tagasi kasutajani, tekitades nii vestluse efekti.

Kolmas süsteem on Discord bot, mis juhib vestlust Discord rakenduses mingi kasutajaga. Lisaks sõnumitele ja vastustele on võimalik ka saata käsk, mis omavad mingit muud efekti. Näiteks on võimalik käskudega muuta konteksti ja kustutada konteksti.

2.1 Probleemi kirjeldus

Hetkel ainukene viis, kuidas kasutaja saab süsteeme kasutada, on läbi Discord boti. Sellel on aga mitmeid puudujääke.

- Discord ei ole tarkvara, mida ülikool ametlikult toetab. Osadel erialadal, näiteks Informaatikas, on tavaline, et Discordis on mingi server, mida on võimalik kasutada. Ülikooliüleselt aga sellist nõuet ega võimalust pole. Selle tõttu on keeruline võimaldada sellist rakendust kõikidele tudengitele ja töötajatele.

- Discordis on keeruline luua süsteemi, kus kasutaja saaks hoida mitut eraldiseisvat vestlust, millel on erinevad kontekstid ja sõnumiajalood.
- Kuna vestlus toimub ülikooliga mitteseotud tarkvara sees, siis selle üle pole ülikoolil kontrolli. Pole võimalik teha muid erinevaid kasutusviise ja UI elemente, mis lubaksid luua parema UX-i. On ainult võimalik kirjutada sõnumeid.
- Samuti on ülikoolil soov, et oleks lahendus, mis vastaks TalTechi stiiliraamatule ehk näeks esteetiline.

2.2 Mittefunktsionaalsed nõuded

Peatükk kirjeldab rakenduse mittefunktsionaalseid nõudeid [3], mis on olulised rakenduse kvaliteedi ja kasutatavuse tagamiseks. Need nõuded ei ole seotud konkreetsete funktsioonidega, vaid määravad, kuidas rakendus peaks käituma ja millised omadused tal olema peavad.

- Jõudlus
 - Rakendus peab olema kasutatav veebibrauseris.
 - Rakendus peab olema samakiiresti initsialiseeriv nagu keskmine veebilehekülg.
 - Kõik tegevused peale sõnumivestluste laadimine toimuvad ideaalselt ühe sekundi jooksul.
- Skaleeritavus
 - Rakendus peab olema kasutatav sadade kasutajate poolt samal ajal.
- Turvalisus
 - Rakendus peab olema võimeline sisse logima kasutajat Entra ID domeeni-kontoga. Ehk kõik tudengid ja töötajad peaksid omama ligipääsu ja saavad rakendust kasutada.
 - Kõik serverrakenduse lõpp-punktid, mis on ligipääsetavad klientrakendusest, peavad olema kaitstud.
 - Kui rakendus elab samas masinas, kus elavad ka teised süsteemid nagu Context Provider ja Interaction Manager, siis nende vaheline suhtlus võib olla mitteturvatud.
- Kasutatavus
 - Rakendus peab olema tehtud TalTechi stiiliraamatu järgi.

- Rakendus on kasutatav erinevatel ekraanisuurustel, telefonist kuni monitorideni välja.
- Rakendust on võimalik kasutada eesti keeles või inglise keeles. Rakenduse enda keelt on võimalik menüüst vahetada. Vestluste keeleks osutub vestluse esimese sõnumi keel.
- Kasutaja on võimeline välja logima rakendusest.
- Rakenduse arendamisel hoitakse silmas vaegnägijate ja kurtide vajadusi ning lähtutakse nendest.

■ Hooldatavus

- Rakendus kasutab võimalikult uusi tehnoloogiaid ja nende versioone. Samuti eelistatakse tehnoloogiaid, mida ülikool on teiste rakenduse jaoks varasemalt kasutatud.
- Rakenduse kood on versiooneeritud ja kasutab versioonihaldust, et oleks tehtud muudatuste ja arenduse ajalugu ja et oleks võimalik mitmel arendajal rakendust ehitada samal ajal efektiivselt.
- Rakenduse koodi hoitakse GitLabi süsteemis ühtses kohas teiste seotud rakendustega.
- Klientrakenduse ja serverrakenduse kood on eraldi koodihoidlates (ingl *repositories*).
- Rakendusel on README dokument, kus on rakenduse kirjeldus, kuidas seda arendada, millised on kasutatud tehnoloogiad jms.
- Rakendusel on mingi hulk ühikteste.
- Rakendust on võimalik tööle panna konteineriseerituna kasutades Docker tehnoloogiat.
- Rakendusel on olemas GitLabis CI/CD torustik, mis iga muudatusega koodis valideerib koodi stiili, jooksub testid, valideerib, et koodis poleks lihtsamaid vigu. Kui need sammud kõik lähevad läbi, siis torustik on võimeline ka soovil rakenduse ehitama ja panema püsti ülikooli masinates, kus see rakendus peaks jooksuma.

2.3 Funktsionaalsed nõuded

Peatükk kirjeldab rakenduse funktsionaalseid nõudeid, mis on seotud konkreetsete funktsioonide ja omadustega, mida rakendus peab pakkuma kasutajatele. Need nõuded määravad, kuidas rakendus peaks käituma ja milliseid tegevusi kasutajad saavad teha.

- Peale Entra ID sisselogimist näitab rakendus sisselogitud kasutaja nime.
- Menüüst on võimalik keelt vahetada ja välja logida.
- Rakendusel on täpselt üks vaade, vestluste vaade.
- Vaates vasakul pool on nimekiri tehtud vestlustest koos nimedega.
- On nupp, mis lubab luua uut vestlust juurde.
- Vestlusi on võimalik ümber nimetada ja kustutada.
- Vaates paremal pool on vestlus koos eelnevalt tehtud sõnumitega.
- On tekstikast, kuhu on võimalik kirjutada uut sõnumit.
- Kui kirjutada sõnum, siis mõne sekundi pärast tuleb automaatselt tagasi üks või rohkem sõnumit vastusega.
- Sõnumite maksimumpikkus on 16000 tähemärki.
- Sõnumid on võimalised näitama matemaatilisi valemeid LaTeX-i kujul.
- Sõnumid on võimalised näitama koodijuppe eraldi värvi blokkides ja neil on ka süntaksi esiletõstmise (ingl syntax highlighting) võimekus.

2.4 Tehnilised nõuded

Peatükk kirjeldab rakenduse tehnilisi nõudeid, mis on seotud kasutatavate tehnoloogiate, raamistikute ja tööriistadega. Need nõuded määravad, milliseid tehnoloogiaid ja tööriistu tuleb kasutada rakenduse arendamiseks ja käitamiseks.

Töö raames on tehtud kaks eraldi rakendust, serverrakendus ja klientrakendus.

- Serverrakendus
 - Java 21 (LTS) [4]
 - Spring Boot 3.4
 - Sessiooni- või ligipääsutunnusepõhine turvalahendus [5]
 - Relatsiooniline andmebaas
 - Andmebaasi versioonihaldus

- REST lõpp-punktid
- WebSocket võimekus [6]
- WebSocket on turvatud, see töötab ainult sisselogitud kasutajaga
- JUnit abil üksuste testimine
- Klientrakendus
 - React 18
 - Kasutab TalTechi stiiliraamatu teeki
 - SPA [7], ehk ehitab kokku üheks JS kimbuks, mida on võimalik juurutada kuhugi HTTP serverisse
 - WebSocket võimekus

2.5 Võimekused

Peatükk kirjeldab rakenduse võimekusi, mis on seotud selle funktsionaalsuse ja omadustega, mida rakendus peab pakkuma kasutajatele. Need võimekused määravad, kuidas rakendus peaks käituma ja milliseid tegevusi kasutajad saavad teha.

- Avalikust veebist on kättesaadav lehekülg, mis vastab TalTechi stiiliraamatule.
- Lehekülg on kasutatav erinevatelt ekraanisuurustelt, sealhulgas telefonist.
- Ülikooli tudengid ja töötajad on võimelised sisse logima Entra ID domeenikontoga ehk UniID-ga.
- Näidatakse sisselogitud kasutaja nime.
- On nimekiri vestlustest, mis on loodud.
- On võimalik lisada juurde uut vestlust.
- Kui valida nimekirjast vestlus, siis avaneb kogu vestluse ajalugu ehk varasemalt tehtud sõnumid.
- Vestluses on võimalik kirjutada uus sõnum, mille maksimumpikkus on 16000 tähemärki.
- Iga uue sõnumi peale tuleb mõne sekundi pärast vastus, kas siis õige tekstiga või veateatega, kui läks midagi valesti.
- Sõnumite laadimise ajal on näha laadimise indikaatorit.
- On võimalik, et tuleb tagasi mitu erinevat sõnumit, kui tekst ei mahu ära ühte sõnumisse.

- Vestlusel on ligipääs Moodle andmetele ehk juturobot on teadlik tudengi või töötaja registreeritud kursustest ja kodutöödest ning on võimeline neid arvesse võtma vastamisel.
- Vestlus on võimeline näitama õigesti välja nimekirju, matemaatilisi valemeid ja koodijuppe eraldi blokkides.
- Vestlusi on võimalik ümber nimetada.
- Vestlusi on võimalik kustutada.

2.6 Valideerimine

Peatükk kirjeldab rakenduse valideerimise protsessi, mis on seotud selle funktsionaalsuse ja omaduste testimisega. Valideerimine on oluline, et tagada rakenduse nõuetele vastavus ja kvaliteet.

2.6.1 Klientrakenduse ühiktestid

Klientrakendusel on olemas ühiktestide jooksmise võimekus. Selle töö raames on tehtud üks ühiktest, mis testib, kas rakendus läheb püsti õigesti. Neid ühikteste jooksutatakse GitLabis CI/CD torustiku kaudu. Testide läbimine on vajalik, et rakenduse saaks ehitada kokku ja panna püsti ülikooli masinatesse.

2.6.2 Serverrakenduse ühiktestid

Serverrakendusel on olemas ühiktestide jooksmise võimekus. Selle töö raames on tehtud kaks ühiktesti, mis testivad, kas rakendus läheb püsti õigesti. Neid ühikteste jooksutatakse GitLabis CI/CD torustiku kaudu. Testide läbimine on vajalik, et rakenduse saaks ehitada kokku ja panna püsti ülikooli masinatesse.

2.6.3 Kasutajaliidese testimine

Rakenduse arendusversioon on püsti pandud ülikooli masinatesse ja on kasutatav veebib-rauseris leheküljelt ¹. Rakendust on testitud erinevatel brauseritel, sealhulgas Chrome, Firefox ja Edge. Rakendus on testitud ka erinevatel seadmetel, sealhulgas telefonidel ja arvutitel. Kasutajaliides vastab TalTechi stiiliraamatule ning on kasutatav eesti ja inglise

¹<https://cs.taltech.ee/chatbot/>

keeles. Rakendusse saab sisse logida nii tudengi kui töötajana, seega on eraldi pandud rõhku, et mõlemast grupist oleksid kasutajad, kes testivad rakendust.

2.6.4 Vestluste testimine

Vestlusi on testitud erinevate teemadega, sealhulgas matemaatika, programmeerimine ja üldine teave. Vestlused on testitud erinevate pikkustega sõnumitega, sealhulgas lühikeste ja pikkade sõnumitega. Vestlused on testitud ka erinevate keeltega, sealhulgas eesti ja inglise keeles. Vestluste käigus on testitud ka konteksti kasutamist Moodle andmete põhjal.

2.6.5 Kasutajate tagasiside

Kasutajate tagasiside on kogutud rakenduse kasutamise käigus. Seda kasutatakse tulevikus rakenduse täiustamiseks ja parendamiseks.

2.7 Projekti ülesehitus

Peatükk kirjeldab iganädalast tööd, arendusprotsessi ja tööde jagamist. Kuna see töö on osa suuremast juturoboti projektist, siis iganädalaselt toimub koosolek, kus arutatakse läbi mitme arendaja ja projektijuhi vahel, mis on tehtud, mis on pooleli ja mis on plaanis teha. See aitab hoida projekti fookuses ja tagada, et kõik arendajad töötavad ühtse eesmärgi nimel.

Projekt koosneb kahest rakendusest, nii et neile on tehtud GitLabi keskkonnas eraldi koodihoidlad. Neil on ka tehtud eraldi ülesanded (ingl *issues*). Arendus käib agiilsel meetodil ja vastab Kanban metoodikale.

Versioonihalduseks on Git. Koodihoidlad on GitLabis, kus on tehtud ka CI/CD torustik, mis valideerib koodi stiili, jooksub testid ja ehitab rakenduse kokku. CI/CD torustik on seadistatud nii, et see käivitub iga kord, kui koodis tehakse muudatusi.

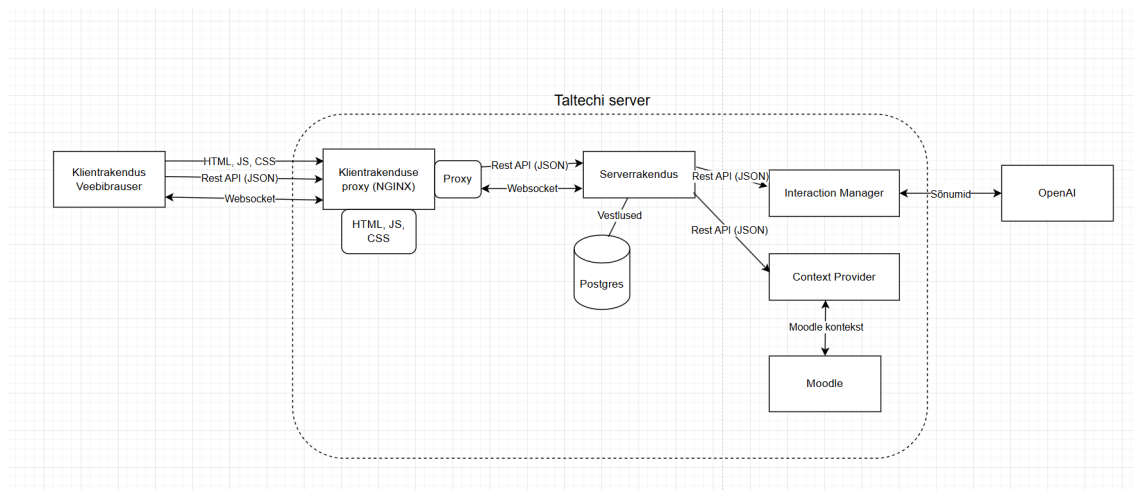
3 Tehnoloogiline ülevaade

Peatükis antakse ülevaade rakenduse arhitektuurist, kasutatud tehnoloogiatest ja nende rakendamisest. Samuti käsitletakse kasutuslugude erijuhte ja haldust ning tulemuste valideerimist.

3.1 Arhitektuur

Rakendus koosneb kahest osast: klientrakendus ja serverrakendus. Klientrakendus on veebirakendus, mis suhtleb serveriga API kaudu. Serverrakendus haldab andmebaasi, autentimist ja autoriseerimist ning pakub API-d klientrakendusele.

Serverrakendus suhtleb ka väliste teenustega, et saada andmeid ja teostada operatsioone, mis ei ole seotud andmebaasiga. Interaction Manager on rakendus, mis on võimeline suhtlema OpenAI süsteemidega ning andma kasutajale vastuseid, mis on genereeritud OpenAI mudelite poolt. Context Provider on rakendus, mis on võimeline suhtlema Moodle süsteemiga ning on võimeline hoidma vestluste konteksti, kaasaarvatud vestluste ajalugu.



Joonis 1. Arhitektuur.

3.2 Klientrakendus

Klientrakendus on veebirakendus, mis põhineb SPA (ingl *Single Page Application*) arhitektuuril. See tähendab, et kogu rakendus laaditakse ühel lehel ja kasutaja liides uuendatakse ilma lehte uuesti laadimata. Klientrakendus on kirjutatud Reacti raamistikus ja kasutab TypeScripti programmeerimiskeelt. Klientrakendus suhtleb serveriga API kaudu, kasutades HTTP päringuid. Klientrakendus on jagatud mitmeks komponendiks, mis haldavad erinevaid funktsionaalsusi, nagu kasutajaliides, autentimine, vestluste haldamine ja andmete kuvamine.

Tabel 1. Klientrakenduses kasutatud tehnoloogiad.

Tehnoloogia	Kirjeldus
React	JavaScript teek, mis võimaldab luua kasutajaliidese komponente.
TypeScript	JavaScripti ülaltoodud versioon, mis lisab staatilise tüübi kontrolli ja muudab koodi loetavamaks ning hooldatavamaks.
Bootstrap	CSS raamistik, mis võimaldab luua responsiivseid ja mobiilisõbralikke kasutajaliideseid. TalTechi stiiliraamatu teek on loodud Bootstrapi baasil, et tagada ühtne ja kaasaegne välimus.
Axios	Teek, mis võimaldab teha HTTP päringuid ja hallata vastuseid. See on kasutusel API päringute tegemiseks serverrakendusele.
Azure MSAL browser	Microsoft Authentication Library, mis võimaldab autentida kasutajaid Microsofti teenustes (nt Entra ID [8]). Kasutatakse autentimiseks ja autoriseerimiseks. Peale sisselogimist saab rakendus ligipääsutunnuse serverrakendusega suhtlemiseks.
i18next	Teek, mis võimaldab rakenduses kasutada mitut keelt. See on kasutusel kasutajaliidese tõlkimiseks ja lokaliseerimiseks.
Markdown, Remark ja Rehype	Markdown on lihtne märgistuskeel. Remark ja Rehype võimaldavad Markdowni teksti töödelda ja kuvada veebirakenduses. Kasutatakse vestluste ja ülesannete kirjelduste kuvamiseks. Hetkel toetatud valemid ja kood [9], tulevikus plaanis lisada tabelid ja pildid.
Zod	Teek, mis võimaldab valideerida andmeid ja luua skeeme. See on kasutusel API päringute andmete valideerimiseks ja veateadete kuvamiseks.
TanStack Query	(Varem React Query) Teek, mis võimaldab hallata serveri andmete päringuid ja nende protsesse. See on kasutusel API päringute haldamiseks ja andmete kuvamiseks kasutajaliideses.

ESLint	Tööriist, mis võimaldab kontrollida koodi kvaliteeti ja stiili. See on kasutusel koodi analüüsimiseks ja vigade leidmiseks [10].
Prettier	Tööriist, mis võimaldab vormindada koodi vastavalt kindlatele reeglitele. See on kasutusel koodi vormindamiseks ja loetavuse parandamiseks.
Vitest	Testimisraamistik, mis võimaldab kirjutada ja käivitada teste Reacti rakendustes. See on kasutusel komponentide ja funktsionaalsuste testimiseks.
Vite	Arendusserver ja ehitustööriist, mis võimaldab kiiret arendust ja kompileerimist. See on kasutusel rakenduse arendamiseks ja ehitamiseks.
Node.js	JavaScripti jooksukeskkond, mis võimaldab käivitada JavaScripti koodi serveris. See on kasutusel klientrakenduse arendamiseks ja ehitamiseks.

3.3 Serverrakendus

Serverrakendus on realiseeritud Java keeles ja kasutab Spring Boot raamistikku. See haldab andmebaasi, autentimist ja autoriseerimist ning pakub API-d klientrakendusele. Serverrakendus kasutab Java teeki ja raamistikke, et teostada andmebaasi operatsioone, autentimist ja autoriseerimist ning sõnumside ja sündmuste töötlemist.

Tabel 2. Serverrakenduses kasutatud tehnoloogiad.

Tehnoloogia	Kirjeldus
Java	Objektorienteeritud programmeerimiskeel, mis on tuntud oma platvormi sõltumatuse ja turvalisuse poolest. Kasutatakse serverrakenduse arendamiseks ja haldamiseks.
Spring Boot	Java raamistik, mis võimaldab luua veebirakendusi ja REST API-sid. Kasutatakse serverrakenduse arendamiseks ja haldamiseks.
Spring Data JPA	Teek, mis võimaldab lihtsat andmebaasi ligipääsu ja haldamist. Kasutatakse andmebaasi operatsioonide teostamiseks ja andmete salvestamiseks.
Spring Security	Teek, mis võimaldab rakenduses autentimist ja autoriseerimist. Kasutatakse kasutajate autentimiseks ja ligipääsu kontrollimiseks.
Spring Boot Messaging	Teek, mis võimaldab rakenduses sõnumside ja sündmuste töötlemist. Kasutatakse sõnumite saatmiseks ja vastuvõtmiseks Web-Socketi abil klientrakenduse ja serverrakenduse vahel [11].

JJWT (JSON Web Token)	Teek, mis võimaldab luua ja valideerida JSON Web Tokeneid (JWT). Kasutatakse autentimise ja autoriseerimise protsessis, et tagada turvaline ligipääs API-le.
PostgreSQL adapter	Teek, mis võimaldab Spring Boot rakendusel suhelda PostgreSQL andmebaasiga. Kasutatakse andmete salvestamiseks ja päringute teostamiseks.
Flyway	Teek, mis võimaldab hallata andmebaasi migratsioone ja versioonihaldust. Kasutatakse andmebaasi struktuuri muutmiseks ja uuendamiseks.

3.4 Andmebaas

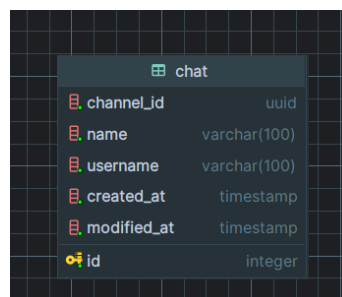
Serverrakendusel on PostgreSQL andmebaas, mis hoiab vestlusi, mis on seotud kasutajatega.

3.4.1 PostgreSQL

PostgreSQL on avatud lähtekoodiga objekti-relatsiooniline andmebaasihaldussüsteem, mis on tuntud oma jõudluse ja skaleeritavuse poolest. See on kasutusel serverrakenduse andmete salvestamiseks ja haldamiseks.

3.4.2 Tabelid

Andmebaasis on üks tabel, Chat, mis hoiab vestlusi.



chat	
channel_id	uuid
name	varchar(100)
username	varchar(100)
created_at	timestamp
modified_at	timestamp
id	integer

Joonis 2. Chat tabel.

Tabel 3. Chat tabeli veerud.

Võti	Väärtus
channel_id	unikaalne ID, mis identifitseerib vestluse kanali, mis on seotud kasutajaga.

name	vestluse nimi, mis on alguses staatiliselt määratud, kuid hiljem saab kasutaja seda muuta.
username	kasutaja email Entra ID süsteemi.
created_at	kuupäev ja kellaaeg, millal vestlus loodi.
updated_at	kuupäev ja kellaaeg, millal vestluse parameetreid viimati uuendati.

Joonisel 2 on näidatud Chat tabel. Selle tabeli veerud on kirjeldatud tabelis 3.

3.5 Autentimine ja autoriseerimine

Klientrakendus kasutab Microsofti autentimise teeki (MSAL) kasutaja autentimiseks Entra ID-s. MSAL hoiab võtmeid sessioonitalletuses (ingl *Session Storage*), mis on veebilehitseja salvestusruum, mis säilitab andmeid seansi jooksul. See tähendab, et veebilehitseja kinni pannes ja uuesti avades on kasutaja veel sisse logitud.

Sisselogimise nupule vajutades suunab klientrakendus kasutaja Entra ID sisselogimislehele, kus kasutaja saab kasutada erinevaid sisselogimismeetodeid. Klientrakendus saab sisselogimise käigus kätte identsustõendi (ID token), mis sisaldab kasutaja teavet, nagu e-posti aadress ja kasutajanimi. Kui klientrakendusel on autentimine tehtud, siis on olemas identsustõend. Identsustõendi tüüpiline aegumisaeg on 1 tund, kuid see võib varieeruda sõltuvalt Entra ID seadistustest.

Identsustõendi abil saab klientrakendus küsida Entra ID süsteemist uuendusloa (access token). Uuendusloa tüüpiline aegumisaeg on 24 tundi, eriti SPA rakenduste puhul. Teist tüüpi rakendustel võib uuendusluba kesta mitu kuud, hoides kasutajaid pikemat aega sisse logituna.

Uuendusloa abil saab klientrakendus küsida Entra ID süsteemist ajutise ligipääsutunnuse, millega saab teha päringuid serverrakenduse vastu. Ligipääsutunnus on lühiajaline ja selle aegumisaeg on tavaliselt 1 tund. Klientrakendus annab iga API päringuga kaasa ligipääsutunnuse. Samuti pannakse ligipääsutunnus WebSocketi sees olevatele sõnumitele, et serverrakendus saaks neid valideerida. Serverrakendus valideerib selle tunnuse ja lubab või keelab päringu.

Kõik kolm tüüpi tunnust on krüpteeritud Entra ID poolt privaatvõtmega (ingl *Private Key*)

ja neid saab dekrüpteerida avaliku võtmega (ingl *Public Key*). Klientrakenduses vastutab avaliku võtme küsimise ja kasutamise eest MSAL teek. Serverrakenduses kasutatakse JWT teeki, et dekrüpteerida ja valideerida identsustõendit, uuendusluba ja ligipääsutunnust. Serverrakendus küsib ise avalikust Microsofti teenusest avalikku võtit ja salvestab selle vahemällu, et vältida pidevat päringute tegemist.

Lisaks avaliku võtme dekrüpteerimisele kontrollitakse ka ligipääsutunnuse kehtivust ja kasutajaskonda (ingl *audience*).

3.6 Juurutamine

Rakendus on juurutatud TalTechi on-premise serverisse, mis hoiab mitmeid dev ja test keskkondade rakendusi. Rakendus on kasutatav taltech'i leheküljel ¹.

3.6.1 Docker

Nii klientrakendus kui ka serverrakendus on konteineriseeritavad Dockeri abil. See võimaldab rakendust hõlpsasti juurutada ja hallata erinevates keskkondades. Mõlemal rakendusel on olemas Docker compose fail, mis defineerib konteinerid ja nende seaded.

3.7 Põhiline funktsionaalsus

Siin peatükis antakse ülevaade rakenduse põhifunktsionaalsusest, mis on seotud kasutajaliidese ja API päringutega.

3.7.1 Vestluste pärimine

Kui rakendusse logitakse sisse, siis kasutaja on koheselt vestluste vaates. Klientrakendus teeb HTTP GET päringu `/v1/api/chat` lõpp-punktile, et saada kasutaja vestluste loetelu. Serverrakendus vastab JSON formaadis andmetega, mis sisaldavad vestluste ID-sid, nimesid ja loomise kuupäevi.

¹<https://cs.taltech.ee/chatbot/>

3.7.2 Vestluse loomine

Kui kasutajaliideses vajutada pluss ikooniga nupule, siis klientrakendus teeb HTTP POST päringu */v1/api/chat* lõpp-punktile, et luua uus vestlus. Serverrakendus loob uue vestluse andmebaasis ja tagastab selle ID.

3.7.3 Vestluse nime muutmine

Kui kasutaja soovib muuta vestluse nime, siis ta vajutab nime kõrval olevale pliiatsi ikoonile. Klientrakendus avab dialoogi, kus kasutaja saab sisestada uue nime. Kui kasutaja vajutab salvestamise nuppu, siis klientrakendus teeb HTTP POST päringu */v1/api/chat/modify* lõpp-punktile, kuhu antakse kaasa vestluse ID ja uus nimi. Serverrakendus muudab andmebaasis vastava vestluse nime ja tagastab uuendatud vestluse andmed.

3.7.4 Vestluse kustutamine

Kui kasutaja soovib kustutada, siis ta vajutab nime kõrval olevale pliiatsi ikoonile. Klientrakendus avab dialoogi, kus kasutaja saab vajutada kustutamise nupule. Kui kasutaja vajutab kustutamise nuppu, siis klientrakendus teeb HTTP DELETE päringu */v1/api/chat/delete* lõpp-punktile, kuhu antakse kaasa vestluse ID.

3.7.5 Sõnumi loomine

Kui kasutaja soovib saata sõnumi, siis ta kirjutab selle sisestusvälja ja vajutab sisestamise nuppu. Klientrakendus teeb HTTP POST päringu */v1/api/message* lõpp-punktile, kuhu antakse kaasa vestluse ID ja sõnumi sisu. Serverrakendus loob uue sõnumi, saadab selle Interaction Managerile, ja saadab ka läbi WebSocketi klientrakendusele tagasi.

3.7.6 Vastussõnumi saamine

Serverrakendusel on HTTP POST lõpp-punkt */v1/api/message/send*, kuhu Interaction Manager saadab sõnumi, mis on genereeritud OpenAI mudelite poolt. See lõpp-punkt on kaitstud ja klientrakendus seda välja kutsuda ei saa. Serverrakendus saada selle sõnumi läbi WebSocketi klientrakendusele, et kuvada seda kasutajaliideses.

3.7.7 Vestluse WebSocketi ühendus

Kui vestluste loetelust avada vestlus, siis klientrakendus loob WebSocketi ühenduse serverrakendusega. Tehakse tellimus (ingl *subscription*) vestluse sõnumitele, et saada reaalsajas värskendusi. Tellimuse loomisel pärib serverrakendus Context Providerilt vestluse ajalugu, ja saadab need sõnumid läbi WebSocketi klientrakendusele. Tellimused on seotud kindla vestluse ja kasutajaga, et tagada, et kasutaja saab ainult oma vestluse sõnumeid.

3.8 Kasutuslugude erijuhud ja haldus

Kasutuslood on defineeritud GitLabi keskkonna Issues vaates, kus on kirjas ka nende prioriteet ja olek.

3.8.1 Päringute protsess

Klientrakendus on teadlik kõikidest päringutest, mida tehakse ning vastavalt vajadusele lülitab välja või visuaalselt muudab kasutajaliidese elemente, et anda hea kasutajakogemus, ja eemaldada võimalus, et mingeid tegevusi tehakse kogemata mitu korda. Näiteks, mitmeid nuppe lülitatakse välja ja tegevust alustanud elemendi juures või sees on laadimisikoon, mis näitab, et tegevus on pooleli.

3.8.2 Päringute vead

Kui päringud serverrakendusele ebaõnnestuvad, siis klientrakendus kuvab veateadet, mis annab kasutajale teada, et midagi läks valesti.

3.8.3 Rollid

Rakendus on võimeline eristama kahte rolli: tudeng ja töötaja. Hetkel pole erinevusi nende kahe rolli vahel, kuid tulevikus on plaanis lisada töötajatele rohkem funktsionaalsusi, nagu näiteks vestluste haldamine ja andmete analüüs.

3.9 Tulemuste valideerimine

Tulemuste valideerimine toimub kasutajate tagasiside kaudu, mis on saadud rakenduse testimise käigus. Kasutajad on andnud tagasisidet rakenduse kasutatavuse, funktsionaalsuse

ja kasutajaliidese kohta. Rakenduse testimine on toimunud TalTechi tudengite ja töötajate seas, kes on andnud tagasisidet rakenduse kasutamise kohta.

Kasutajatele anti Google Forms keskkonnas ankeet, kus nad said hinnata rakenduse kasutatavust ja funktsionaalsust.

Ankeedis küsiti kasutajate arvamust rakenduse kasutatavuse, funktsionaalsuse ja kasutajaliidese kohta. Kasutajad said hinnata rakendust skaalal 1-10, kus 1 on väga halb ja 10 on väga hea. Kasutajad said ka anda vaba tagasisidet rakenduse kohta, kus nad said kirjutada, mis neile meeldis ja mis mitte. Ankeedi tulemuste kuvatõmmised on lisatud lõputöö lisadesse (4).

Suuremad väljatoodud probleemid:

- Moodle kursusi ei saa kätte (Hetkel pole juturobot paljudele kursustele registreeritud).
- Rakendus on telefonis kasutatav, aga ebamugav.
- Pole infot, mida suudab see juturobot teha, mida ei suuda teha tavaline ChatGPT.
- Lehe eesmärk on ebakindel, võiks olla mingi juhend.
- Osad värvid ei lähe omavahel kokku, näiteks lilla ja roosa.

4 Kokkuvõte

Käesoleva lõputöö raames arendati veebirakendus, mis võimaldab TalTechi tudengitel ja töötajatel suhelda tehisintellekti mudelitega turvalises ja mugavas keskkonnas. Rakenduse eesmärk on muuta tehisintellekti teenused kättesaadavaks kogu ülikooli kogukonnale, toetades nii õppe- kui ka teadustööd ning soodustades iseseisvat õppimist.

Loodud lahendus võimaldab kasutajatel vestelda ChatGPT-laadse keeletehnoloogiamudeliga, mis on lisaks üldteadmistele kursis ka kasutaja individuaalsete õppeainete ja TalTechi teenustega. See võimaldab pakkuda personaalsemat tuge, vastata kursusega seotud küsimustele ning aidata leida vajalikku infot kiirelt ja tõhusalt. Süsteemi arendamisel pöörati erilist tähelepanu kasutusmugavusele, turvalisusele ning integratsioonile ülikooli infosüsteemidega, nagu Moodle.

Töö käigus viidi läbi põhjalik nõuete analüüs, kavandati ja realiseeriti klient-server arhitektuuril põhinev lahendus ning viidi läbi testimine ja valideerimine, et tagada süsteemi töökindlus ja vastavus seatud eesmärkidele. Valminud veebirakendus on avalikult kättesaadav, nii arendus¹ kui lõplik² versioon ning on kasutatav kõigi TalTechi tudengite ja töötajate poolt.

¹<https://cs.taltech.ee/chatbot/dev/>

²<https://cs.taltech.ee/chatbot/>

Kasutatud kirjandus

- [1] Nidji Raj. *Minimum Viable Product (MVP) - What is it and how to start*. [Accessed: 05-06-2025]. URL: <https://www.atlassian.com/agile/product-management/minimum-viable-product>.
- [2] Cybernetica AS. *ANDMEKAITSE JA INFOTURBE PORTAAL*. [Accessed: 29-05-2025]. URL: <https://akit.cyber.ee/>.
- [3] John Dalbey. *Non-functional Requirements*. [Accessed: 05-06-2025]. URL: <https://users.csc.calpoly.edu/~jdalbey/SWE/QA/nonfunctional.html>.
- [4] Curtis Johnson. *What is Java LTS and Why Does It Matter?* [Accessed: 05-06-2025]. URL: <https://www.jrebel.com/blog/java-lts>.
- [5] Natalia Moskaleva. *User Authentication: Comparing Session-based and Token-based Implementations*. [Accessed: 05-06-2025]. URL: <https://www.criipto.com/blog/session-token-based-authentication>.
- [6] Dipali Ambaliya. *WebSocket + Spring Boot: Build a Real-time, Bidirectional Applications*. [Accessed: 05-06-2025]. URL: <https://medium.com/simform-engineering/websocket-spring-boot-build-a-real-time-bidirectional-applications-e8c95bc19cbf>.
- [7] Chiradeep BasuMallick. *What Is a Single-Page Application? Architecture, Benefits, and Challenges*. [Accessed: 05-06-2025]. URL: <https://www.spiceworks.com/tech/devops/articles/what-is-single-page-application/>.
- [8] Microsoft. *Microsoft Entra ID*. [Accessed: 05-06-2025]. URL: <https://www.microsoft.com/en-us/security/business/identity-access/microsoft-entra-id>.
- [9] Remark. *awesome remark*. [Accessed: 05-06-2025]. URL: <https://github.com/remarkjs/awesome-remark>.
- [10] George Guimarães. *What is a linter and why your team should use it?* [Accessed: 05-06-2025]. URL: <https://sourcelevel.io/blog/what-is-a-linter-and-why-your-team-should-use-it>.
- [11] Spring. *WebSockets*. [Accessed: 05-06-2025]. URL: <https://docs.spring.io/spring-framework/reference/web/websocket.html>.

Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks³

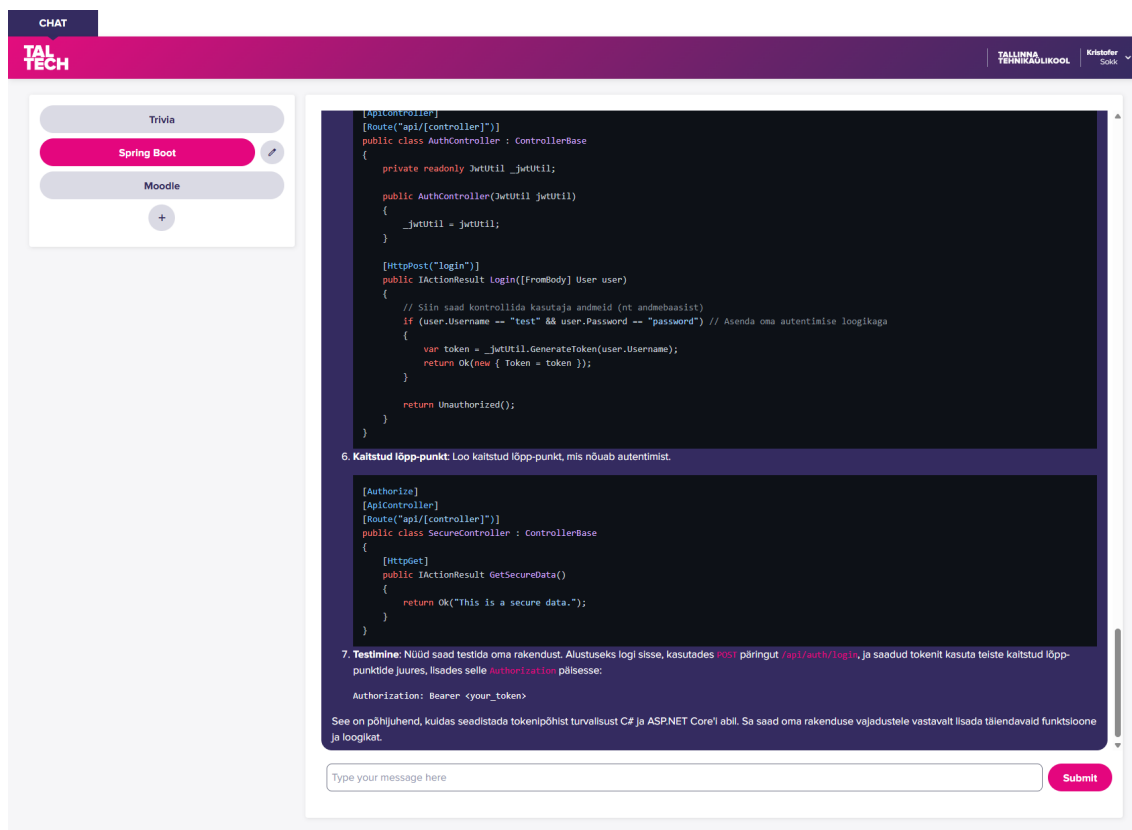
Mina, Kristofer Sokk

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose “Suure keelemudeli vestlusrakenduse realiseerimine”, mille juhendaja on Ago Luberg
 - 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 veebikeskonna 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.

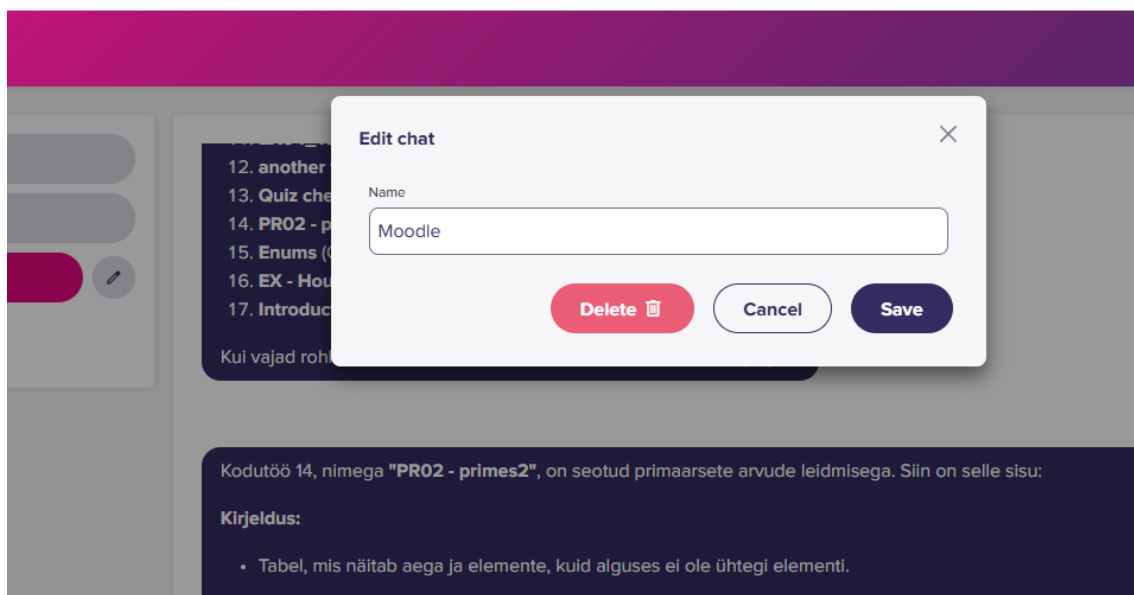
05.06.2025

³Lihtlitsents 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 tähtjaks nõusolekut lõputöö reprodutseerimiseks ja avalikustamiseks vastavalt lihtlitsentsi punktidele 1.1. ja 1.2, siis lihtlitsents nimetatud tähtaja jooksul ei kehti.

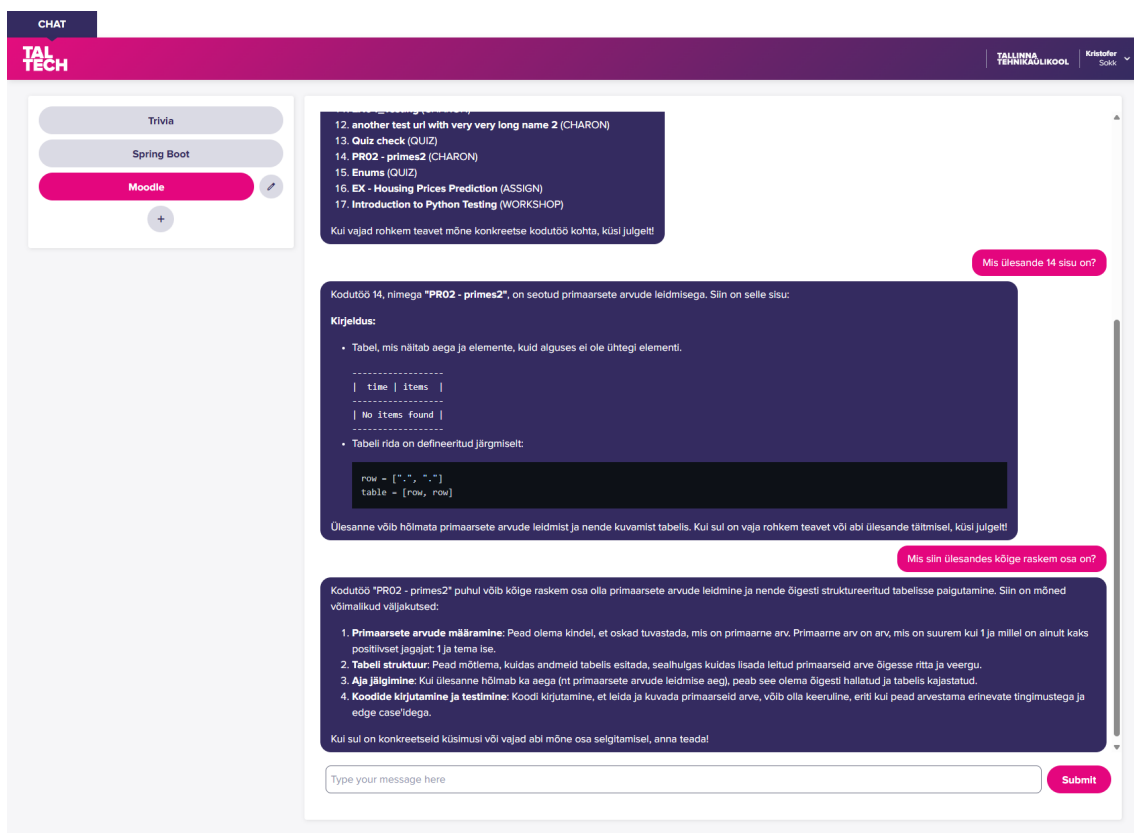
Lisa 2 – Kasutajaliidese kuvatõmmised



Koodijuppide süntaksi esiletõstmine.



Vestluse muutmise või kustutamise aken.



Moodle andmete pärimine.

CHAT
TAL TECH
TALLINNA TEHNILIKAKOOL
Krisofer Sõuk

Trivia
Spring Boot
Moodle

Now, let's follow these steps:

1. Calculate the discriminant ($D = b^2 - 4ac$):

$$D = (-4)^2 - 4 \cdot 3 \cdot 2$$
2. Substitute the values into the discriminant:

$$D = 16 - 24 = -8$$
3. Interpret the discriminant: Since $D < 0$, this means there are no real solutions to the equation; instead, there are two complex solutions.
4. Calculate the complex solutions using the quadratic formula:

$$x = \frac{-(-4) \pm \sqrt{-8}}{2 \cdot 3}$$
5. Simplify the expression:

$$x = \frac{4 \pm \sqrt{8}i}{6}$$

Since $\sqrt{8} = 2\sqrt{2}$, we can rewrite it as:

$$x = \frac{4 \pm 2\sqrt{2}i}{6} = \frac{2 \pm \sqrt{2}i}{3}$$

So, the solutions are:

$$x = \frac{2 + \sqrt{2}i}{3} \quad \text{and} \quad x = \frac{2 - \sqrt{2}i}{3}$$

If you have any questions about the steps or need further assistance, feel free to ask!

what remark plugins I should use for chatgpt messages?

It seems like you're asking about remark plugins that could enhance your experience with ChatGPT messages. However, I'm not sure what specific context you're referring to regarding "remark plugins."

If you're looking for ways to improve the interaction, here are a few general suggestions:

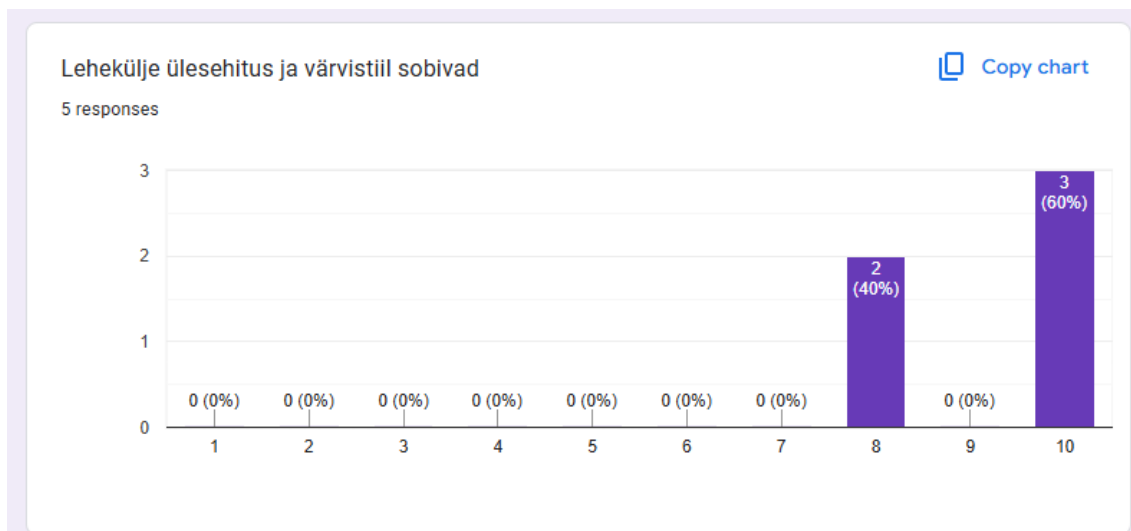
1. **Formatting Tools:** Use plugins that allow you to format your messages with markdown, which can help in emphasizing certain parts of the text (bold, italics, lists, etc.).
2. **Code Highlighting:** If you're discussing programming or technical topics, a plugin that highlights code syntax can make it easier to read.
3. **Integration Plugins:** Depending on the platform you're using, there might be plugins that allow integration with other tools (like task managers or note-taking apps) to keep track of your queries and responses.
4. **Feedback Tools:** Plugins that allow you to provide feedback on the responses can help improve the quality of interactions over time.

If you meant something different or specific regarding "remark plugins," could you please clarify?

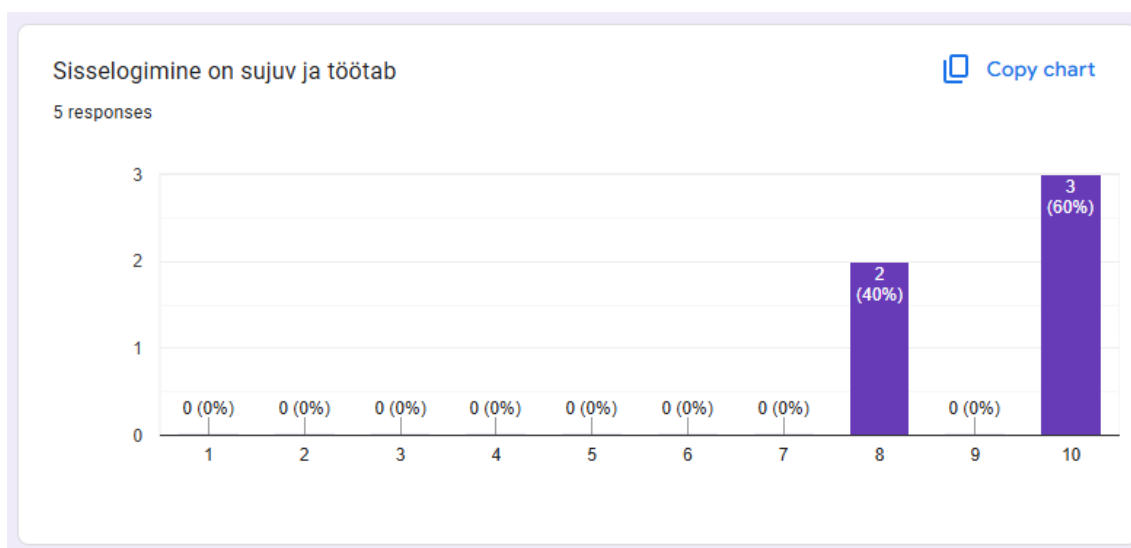
Type your message here
Submit

Matemaatika ja üldine teadmine.

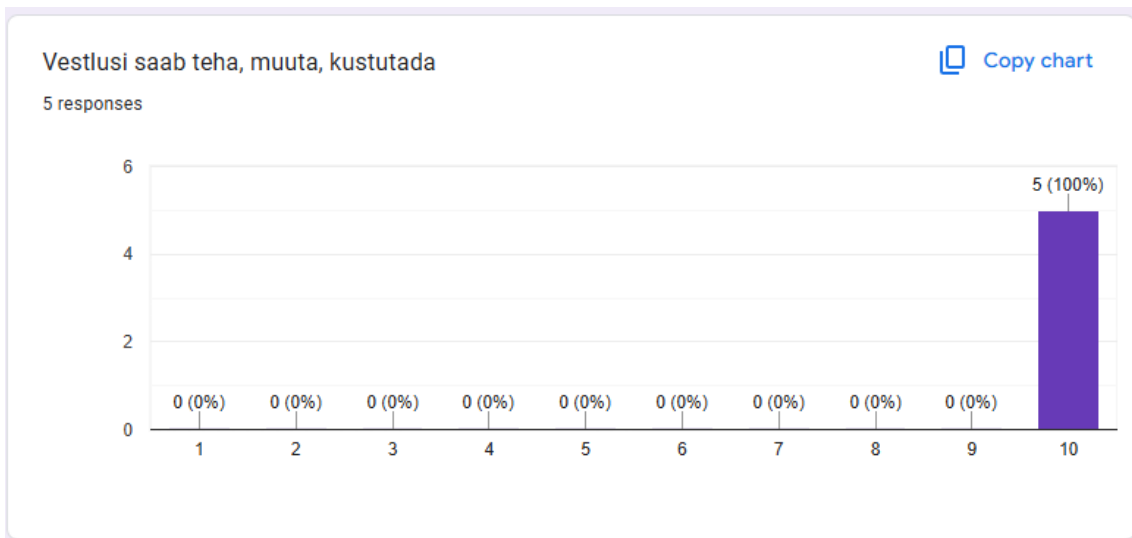
Lisa 3 – Küsitluse tulemused



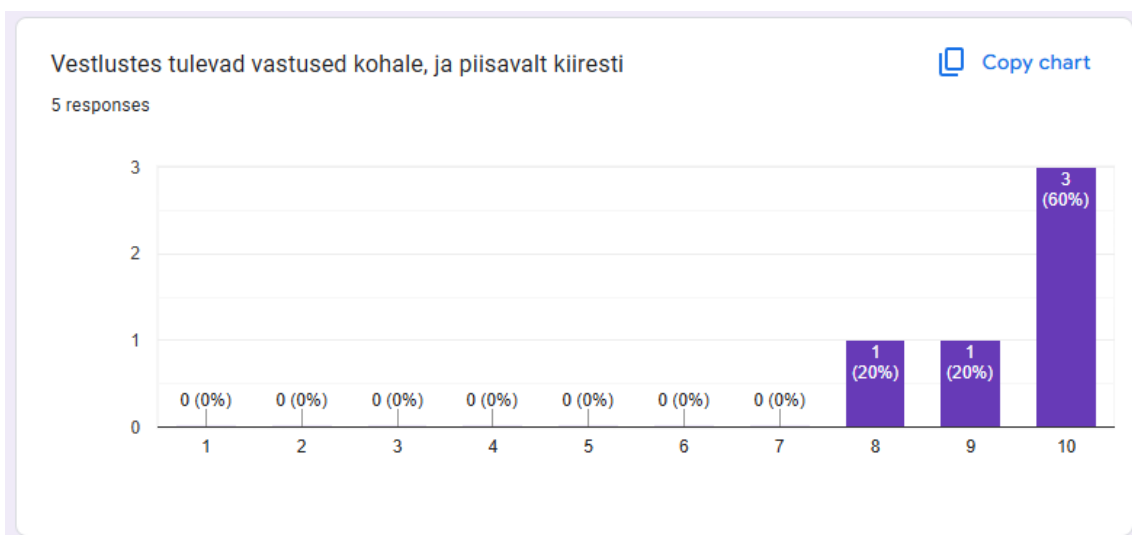
Ülesehitus ja värvistiil.



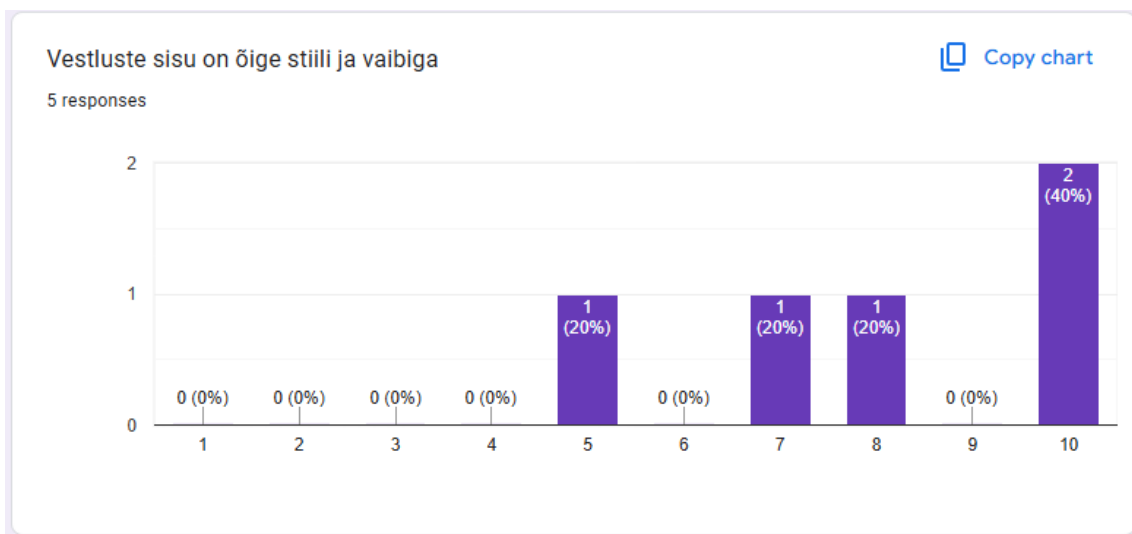
Sisselogimine.



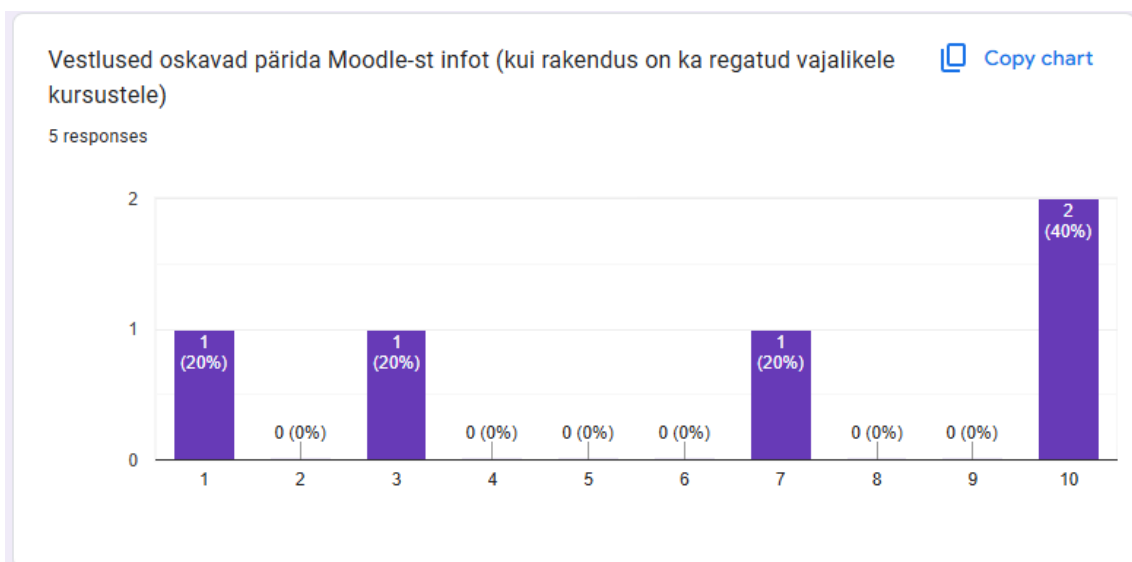
Vestluste loomine, muutmine, kustutamine.



Vestluste tulemuste kiirus.



Vestluste tulemuste stiil.

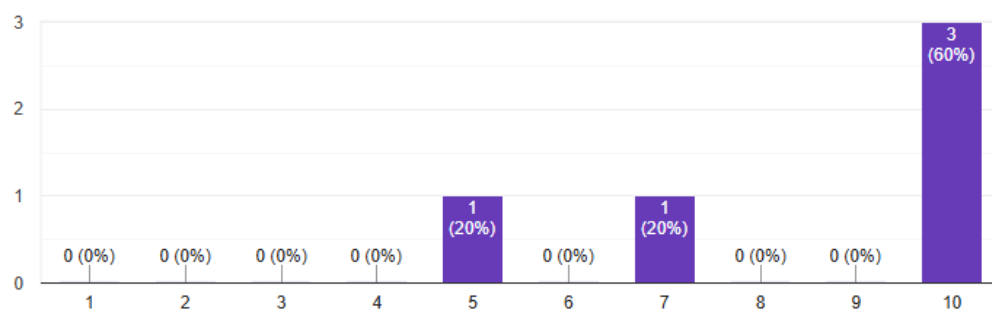


Vestluste tulemuste ühenduvus Moodle-ga.

Rakendus töötab hästi nii eesti kui inglise keeles

 Copy chart

5 responses

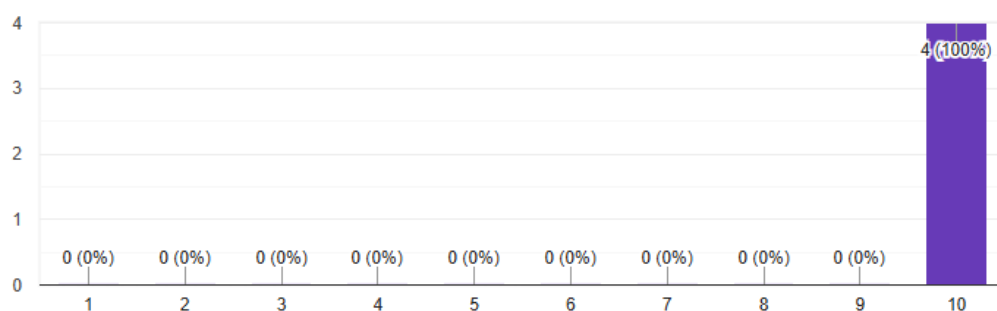


Rakendus töötab nii eesti kui ka inglise keeles.

Vestlused töötavad mitmetes keeltes

 Copy chart

4 responses

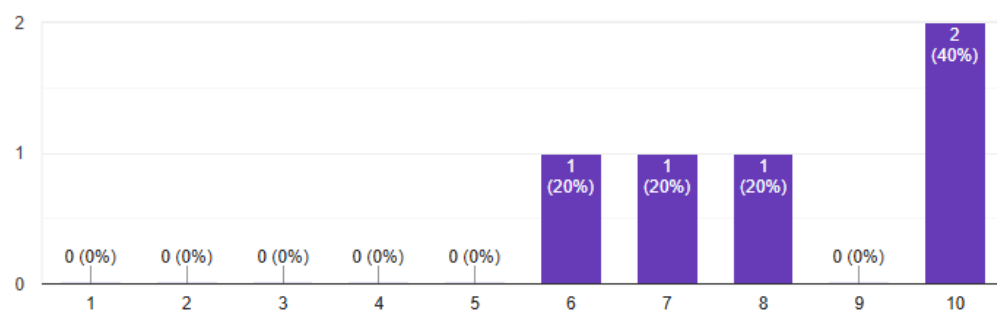


Vestlused toevad mitmeid keeli.

Telefonis on kogu UI kasutatav

 Copy chart

5 responses



Rakendus on kasutatav telefonis.

Oli midagi katki?

3 responses

UI on telefonis pigem ebamugav. Hetkel pole ka teadmist, et millistelt kursustelt tuleb päris info loodetavasti ja millistega pole lootustki.

AI ei leidnud mu infot üles, aga võib ka sellest olla, et mul on ainult lõputöö deklareeritud? Kuigi ütles, et ei leia mind Moodlest või miskit

Ta ei saa mu kursusi kätte

Mis funktsionaalsus võiks kindlasti veel olla?

2 responses

Paremini võiks olla välja toodud, et mis chatilt küsida saab

Võiks mingi selgitav juhend, miskit sellist olla? Algul jäi veidi segaseks, et mis eesmärk lehel on

Veel mingeid kommentaare?

2 responses

Vestluse värvid ei ole hästi loetavad, kui küsida, mingi konkreetse ülesande kohta. Taust on lilla ja osa teksti roosa, mida raske näha. Muidu hea leht.

Miks ma peaksin kasutama seda chatgpt asemel?

Muud probleemid/ettepanekud.