

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Ksenija Kozlova 193910IADB

Veebirakenduse loomine võõrkeele õppimiseks ristsõnamõistatuste abil

Bakalaureusetöö

Juhendaja: Meelis Antoi

Magistrikraad

Tallinn 2022

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: Ksenija Kozlova

23.02.2022

Annotatsioon

Käesoleva bakalaureusetöö eesmärgiks on luua töötav veebirakendus inglise keele õppimiseks B1 taseme ristsõnade abil, mis põhineb “Oxford English File” õpikul ning “inglise keele õppimine inglise keeles” tehnikal ja on kohandatud ka mobiilseadmetes kasutamiseks.

Arendusprotsessi käigus on plaanis luua veebirakendus, mis võimaldab kasutajatel registreeruda, oma taset valida, soovitud teemat valida ja ristsõnu lahendada. Arendus on jagatud mitmesse ossa, mis sisaldavad rakenduse *frontend* ja *backend*.

Töö rõhk on keele õppimise võimaluse loomisel mittetraditsioonilise meetodi abil. Käesoleva töö hõlmab alternatiivsete lahenduste analüüsi, loodava rakenduse nõudeid, loodava rakenduse perspektiive, kasutatud tehnoloogiaid ja osaliselt arendusprotsessi.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 39 leheküljel, 4 peatükki, 13 joonist, 7 tabelit, 1 koodinäide.

Abstract

Creating a Web Application for Learning a Foreign Language Using Crosswords

The aim of current thesis is to create a working web application for learning English using level B1 crossword puzzles, which is based on the "Oxford English File" textbook and the "Learn English in English" technique and is also adapted for use on mobile devices.

During the development process, it is planned to create a web application that allows users to register, choose their level, select the desired topic and solve crossword puzzles. The development is divided into several parts, which include the application frontend and backend.

The emphasis of the work is on creating an opportunity to learn a language using non-traditional method. This work covers the analysis of alternative solutions, the requirements of the application to be created, the perspectives of the application to be created, the technologies used and partly the development process.

The dissertation is written in Estonian and contains text on 39 pages, 4 chapters, 13 figures, 7 tables, 1 code example.

Lühendite ja mõistete sõnastik

.NET Framework	Microsofti 2002. aastal välja antud tarkvaraplatvorm
andmebaasihaldussüsteem	Tarkvarapakett, mis on loodud andmebaasis olevate andmete määratlemiseks ja haldamiseks.
Angular	Populaarne raamistik veebi- ja mobiilirakenduste loomiseks, kirjutatud TypeScriptis
Apache	Tasuta avatud lähtekoodiga serveri tarkvara
API	<i>Application Programming Interface</i> , definitsioonide ja protokollide kogum rakendustarkvara koostamiseks ja integreerimiseks
<i>backend</i>	Teenusepoolne keskkond
C#	Kaasaegne, objektorienteeritud ja tüübikindel programmeerimiskeel
C++	Universaalne objektorienteeritud programmeerimiskeel
Cambridge University Press	Cambridge-i ülikooli kirjastus
CDN	<i>Content Delivery Network</i> , geograafiliselt jaotatud serverite rühm, mis töötavad koos, et pakkuda Interneti-sisu kiiret edastamist.
<i>cookie</i>	Väikeste andmetega tekstifailid, mida kasutatakse konkreetse arvuti tuvastamiseks arvutivõrgu kasutamisel
Django	Kõrgetasemeline Pythoni veebiraamistik, mis soodustab kiiret arengut ja puhast, pragmaatilist disaini
Express.js	Node.js veebirakenduse serveri raamistik, mis on mõeldud üheleheliste, mitmeleheliste ja hübriidveebirakenduste loomiseks
Flask	Pythonis kirjutatud mikroveebiraamistik, ei vaja spetsiaalseid tööriistu ega teeke
<i>framework</i>	Struktuur, millele arendajad saavad tarkvara ehitada. See defineerib reeglid ja annab projekti loomiseks vajalikud tööriistad
<i>frontend</i>	Kasutajaliides, kliendipoolne keskkond
HTML	<i>HyperText Markup Language</i> , kood, mida kasutatakse veebilehe ja selle sisu struktureerimiseks

IDE	<i>Integrated Development Environment</i> , rakendus, mis pakubarendajatele keskkonda koodi kirjutamiseks
IELTS	<i>International English Language Testing System</i> , inglise keele test
Java	Java on platvormist sõltumatu objektorienteeritud programmeerimiskeel
Jinja	Pythoni programmeerimiskeele mallimootor
kompilaator	Programm, mis tõlgib programmeerimiskeeles kirjutatud teksti masinkoodi
Laravel	PHP veebirakenduste raamistik väljendusrikka ja elegantse süntaksiga
Macmillan	Inglise keele õpetamise ja kooli õppekava materjalide väljaandja
MPA	<i>Multi Page Application</i> , veebirakendus, mis koosneb suurest hulgast lehtedest, mida värskendatakse täielikult iga kord, kui nende andmed muutuvad
MVC	<i>Model View Controller</i> , rakendustes kasutatav arhitektuur, mis jaotab rakenduse kolmeks loogiliseks osaks
Node.js	Avatud lähtekoodiga ja platvormideülene JavaScripti käituskeskkond.
NoSQL	Lähenedamine andmebaasi haldamisele, mis mahutab väga erinevaid andmemudeleid
objektorienteeritud programmeerimiskeel	Programmeerimiskeel, mis järgib objektorienteeritud programmeerimise kontseptsioone
Oxford English File	Õpik inglise keele õppimiseks
Oxford University Press	Oxfordi Ülikooli juures tegutsev kirjastus. See on suurim ülikoolikirjastus maailmas.
PHP	Hypertext Preprocessor, avatud lähtekoodiga serveripoolne skriptikeel
Python	Interpreteeritav, objektorienteeritud kõrgetasemeline programmeerimiskeel dünaamilise semantikaga
React.js	Avatud lähtekoodiga JavaScripti teek kasutajaliideste arendamiseks
Redux	Prognoositav olekukonteiner, mis on loodud selleks, et aidata kirjutada JavaScripti rakendusi, mis käituvad kliendi-, serveri- ja algkeskkonnas ning mida on lihtne testida
REST API	Rakendusprogrammi liidese arhitektuurne stiil, mis kasutab andmetele juurdepääsuks ja nende kasutamiseks HTTP-päringuid

Ruby	Interpreteeritav kõrgetasemeline üldotstarbeline programmeerimiskeel, mis toetab mitut programmeerimisparadigmat
skript	Programm või käskude jada, mida tõlgendab või teostab mõni muu programm, mitte arvutiotsessor
Spring	Avatud lähtekoodiga rakendusraamistik, mis pakub infrastruktuuri tuge Java rakenduste arendamiseks
<i>tag</i>	Märgistuskeele element HTML, WML, AmigaGuide, XML perekonna keeled
<i>technology stack</i>	Tehnoloogiapinu
TOEFL	<i>Test of English as a Foreign Language</i> , standardiseeritud inglise keele test
Unicode	Universaalne tähemärkide esitusstandard teksti arvutitöötuses
Unixoid	Meenutab Unixi operatsioonisüsteeme
URI	<i>Uniform Resource Identifier</i> , on tähemärkide jada, mis tuvastab üheselt nime või ressursi Internetis
URL	<i>Uniform Resource Locator</i> , kordumatu identifikaator, mida kasutatakse ressursi leidmiseks Internetis
Urls.py	Vastutab projekti marsruutide ja radade kaardistamise eest
Xamarin	Ameerika tarkvaraarenduse ettevõte

Sisukord

1 Sissejuhatus	12
1.1 Metoodika.....	13
2 Ülevaade probleemist	14
2.1 Eksisteerivad lahendused.....	14
2.1.1 Meetodid.....	15
2.1.2 Riigikoolid, keeltekoolid ja järeleaitajad.....	16
2.1.3 Raamatud ja õpikud.....	17
2.1.4 Veebi- ja mobiilirakendused.....	17
2.2 Uue lahenduse skoop	18
2.2.1 Koostatud plokk skeemid	19
2.2.2 Programmeerimiskeelte võrdlemine	24
3 Loodava veebirakenduse analüüs	27
3.1 Nõuete määramine	27
3.1.1 Funktsionaalsed nõuded	27
3.1.2 Mittefunktsionaalsed nõuded.....	27
3.1.3 Rakenduse arendamine tulevikus	28
3.2 Tehnoloogia valik	28
3.2.1 Programeerimiskeele valik	28
3.2.2 Teenusepoolsed raamistikud.....	29
3.2.3 Kliendipoolne tehnoloogia (frontend)	33
3.3 Andmebaasi valik	34
3.4 Veebirakenduse disain	36
3.5 Analüüsi kokkuvõte	37
4 Veebirakenduse arendus.....	38
4.1 Veebilehe komponentide planeerimine	39
4.2 Bootstrap'i kasutus.....	42
4.3 Andmebaas	42
4.4 Valminud rakenduse kasutamine ning edasiarendused	43
5 Kokkuvõte	45

Kasutatud kirjandus	46
Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks	49
Lisa 2 – Kasutajalood	50
Lisa 3 – Olemi-suhte diagramm	51

Jooniste loetelu

Joonis 1. Kasutaja registreerimise plokkiskeem	19
Joonis 2. Keele raskusastme valimine	20
Joonis 3. Teema valimine	21
Joonis 4. Vale või õige vastuse sisestamine ristsõnades	22
Joonis 5. Õpilase konto kustutamine	22
Joonis 6. Raskusastme lisamine administraatori abiga.....	23
Joonis 7. Teemade lisamine administraatori abiga.....	23
Joonis 8. Ristsõna lisamine administraatori abiga.....	24
Joonis 9. Loodud projekti struktuur.....	39
Joonis 10. „Levels” lehe komponentide diagramm	40
Joonis 11. „List of words” lehe komponentide diagramm.....	41
Joonis 12. „Crossword” lehe komponentide diagramm	41
Joonis 13. Lihtsustatud olemi-suhte diagramm	43

Tabelite loetelu

Tabel 1. PHP programmeerimiskeele eelised ja puudused.....	24
Tabel 2. C# programmeerimiskeele eelised ja puudused	25
Tabel 3. Python programmeerimiskeele eelised ja puudused.....	25
Tabel 4. Java programmeerimiskeele eelised ja puudused.....	26
Tabel 5. Programmeerimiskeelte võrdlus	29
Tabel 6. Populaarseimad veebiraamistikud	31
Tabel 7. Django ja Flask võrdlus.....	32

1 Sissejuhatus

Võõrkeelte õppimine on maailmas alati aktuaalne olnud. Inglise keelt peetakse ärimaailmas ametlikuks keeleks, samuti kasutatakse seda rahvusvahelistes suhetes. Vaatamata kõikidele boonustele, mis inglise keele oskusega kaasnevad, on inimesi, kes ei ole aastate jooksul olnud suutelised keelt selgeks õppima. Isegi kui esineb suur tahe seda teha. Käesoleva töö autor kuulub samuti eelnimetatud inimeste hulka.

Mida kiiremini arenesid infotehnoloogiad, seda aktiivsemalt kasutati neid kõikvõimalikes valdkondades. Võõrkeelte õppimise valdkond ei olnud selles osas erand. Maailmas on suur hulk erinevaid keeleõppe võimalusi: alates keeltekoolidest ning lõpetades mobiilirakendustega. Kõik need on loodud eesmärgiga aidata inimestel uut keelt selgeks õppida. Samas on läbi viidud uuring, mille tulemuste põhjal selgus, et uue keele omandamisel on üks tõhusamaid lähenemisi õppida uut keelt samas keeles. Antud lähenemine võimaldab ajul alustada tööd ja mõtlemist uues keeles. Tänu sellele on lihtsam jätta meelde uusi sõnu, nende tähendusi ning mõista keele struktuuri. Hetkel õpib käesoleva lõputöö autor keeltekoolis, mis töötab “inglise keele õppimine inglise keeles” meetodi alusel. Ühtlasi kaasab meetod eraõpetajaid, kasutab usaldusväärseid infoallikaid ning õpikuid.

Eelnimetatu baasil on käesoleva lõputöö eesmärk luua rakendus, mis põhineb “Oxford English File” õpikul ja “inglise keele õppimine inglise keeles” tehnikal. Lahendus on kohandatud ka mobiilseadmetes kasutamiseks. Rakendus on mõeldud inimestele, kelle inglise keele tase on B1 ning kes on väsinud aastatepikkusest ingliskeelsete sõnade õppimisest. Rakendus näeb ette lihtsat harjutamist, millele tuleb kulutada iga päev 15 minutit. Keeleõppe puhul on peamine stabiilne ja järjepidev lähenemine. Nii on võimalik kasutajatel tavapärasest keeleõppesüsteemist loobuda. Töötav lahendus pakub kasutajatele emakeelsete tõlgete asemel õpikupõhiseid lihtsate ingliskeelsete selgitustega sõnu. Lisaks sellele, et sõnad on varustatud selgitustega, näeb harjutus ette ka ristsõna nende sõnade läbitöötamiseks. Antud tehnika treenib kasutaja mälu, õigekirja, sõnavara ning võimaldab sõnadest õigesti aru saada ning nende sisu tõlgendada.

Käesoleva lõputöö raames on autori eesmärk analüüsida ja võrrelda erinevaid olemasolevaid lahendusi, tuua välja toimiva lahenduse loomiseks vajalikke funktsionaalseid nõudeid ning leida eesmärgi saavutamiseks parimaid vahendeid.

1.1 Metoodika

Lõputöö käigus selgitatakse esmalt olemasolevat probleemi, olemasolevaid lahendusi ja nende puudusi. Seejärel pakutakse välja probleemile sobiv IT-lahendus ning selgitatakse selle lahenduse eripära. Lisaks kaalutakse ka võimalikke viise selle lahenduse väljatöötamiseks tulevikus.

Lahenduse analüüsimisel arutatakse kahe kasutajarolli üle, mille põhjal esitatakse kasutaja lugusid. Samuti tuuakse välja plokk skeeme ning pakutakse funktsionaalseid ja mittefunktsionaalseid nõudeid. Töö raames analüüsitakse töövahendeid ning põhjendatakse lõplikku vahendite valikut eesmärgi saavutamiseks.

2 Ülevaade probleemist

Tänapäeval on nii karjäärivõimaluste kui ka isikliku arengu seisukohalt võõrkeelte oskus väga oluline [1].

Käesolevas lõputöös on rõhk inglise keelele, kuna see keel on maailmas kõige populaarsem. Inglise keel on ametlikult tunnustatud ärikeeelena ning selles keeles toimuvad peaaegu kõik rahvusvahelised foorumid, konverentsid, koosolekud ja läbirääkimised [1].

Ristsõnu lahendamises on inimestel lõbus ning ühtlasi tõuseb selle käigus harituse tase. Inglisekeelse ristsõna lahendamine pakub meelelahutust ning harituse taseme arendamist [2].

Traditsioonilistest keeleõppe meetoditest on võimalik aru saada, kui meenutada, kuidas toimus keeleõpe koolis. Võõrkeele õppimine toimub emakeeles ja kasutades meetodit, mis põhineb grammatika prioritseerimisel keele tuumana ja tõlget kui keele valdamise peamist viisi. Sellist meetodit kasutatakse kõige sagedamini maailmas. Samas on see põhjus, miks keeli on nii raske õppida. Eriti, kui sellise süsteemi järgi õpetada täiskasvanud.

Saladus, kuidas lihtsa vaevaga õppida keelt, mõista õigesti sõnu ja uue keele struktuuri, peitub selles, et uut keelt tuleb õppida nii, nagu õpitakse oma emakeelt. Üks tõhusamaid lähenemise viise on võõrkeele õppimine samas keeles. Esialgu õpitakse uusi sõnu piltide ja assotsiatsioonide kaudu, ning hiljem kasutatakse neid lihtsaid sõnu uute keerukamate sõnade seletamiseks.

Kuna keeleõppe võtmepunktid on sõnavara, sõnade õige mõistmine, kvaliteetsed infoallikad, järjepidevus ja stabiilsus, siis antud lõputöö ja töötav IT-lahendus keskendubki sellele.

2.1 Eksisteerivad lahendused

Tänapäeva maailmas valitseb IT. See on juba seotud kõikvõimalike inimese elu valdkondadega. Keelte õpimiseks eksisteerivad pole kõigest riigikoolide programmid, järeleaitajad, keeltekoolid, vaid ka erinevad meetodid, mis on hõlpsasti Internetist

leitavad. Samuti raamatud ja õpikud, mis on saadaval nii Internetis kui ka raamatukogudes, erinevad veebisaidid, programmid ja rakendused.

Kõik eelnimetatud on loodud ühise eesmärgiga aidata inimestel õppida uut keelt.

2.1.1 Meetodid

Keeleõppe meetodeid on tohutult palju. Järgnevalt tutvustatakse tänapäeval kõige levinumaid võõrkeelte õppimise meetodeid. “Loodusliku” ja “kommunikatiivse” meetodi alusel koolitatakse lõputöö autorit keeltekoolis.

“Grammatika ja tõlke” meetod põhineb grammatika prioritseerimisel keele tuumana ja tõlget kui keele valdamise peamist viisi. Antud tehnika, mis põhineb teaduse poolt ammu ümber lükatud uskumustel, on SRÜ-s kõige levinum. Kui õppisite koolis sõnu loenditena, tõlkisite tekste, tegite sulgude avamise harjutusi ja peaaegu ei rääkinud võõrkeeles, olete selle tehnikaga tuttav. See meetod ilmus Euroopa koolidesse ladina ja vanakreeka, juba sel ajal surnud keelte õppimise vahendina. Õpilastelt nõuti vaid pühakirjade mõistmist, mitte oskust vabalt suhelda ning eelnimetatud meetod tuli selle ülesandega hästi toime. Antud meetod sobis ka NSV Liidus, kus suletud piiride ja range tsensuuri tingimustes võis võõrkeel olla kasulik vaid teadusliku dokumentatsiooni lugemiseks ja tõlkimiseks, mitte aga välismaalastega suhtlemiseks. Kui 20. sajandi lõpus vajasime tõhusat tehnoloogiat just nimelt suhtlemise õpetamiseks, jätkati grammatika-tõlke meetodi kasutamist lihtsalt harjumusest. Paraku lähevad selle meetodiga püstitatud eesmärgid vastuollu eesmärkidega, mis on 21. sajandi tudengil [3].

“Looduslik” meetod on loomulik meetod, mis on tegelikult “Grammatika ja tõlke” meetodi vastand. See on õppimine kordamise ja korrigeerimise kaudu. Õpilaste emakeelt ei kasutata tundides üldse. Siin pööratakse tähelepanu keelebarjääri ületamisele. Kursusel osaleja peab keelt õppima nagu väike laps – esmalt jälgima ja siis kordama. Grammatikareeglitel pole tähendust ja neid ei õpetata. Selle asemel peavad õpilased ise ingliskeelse kõne kontekstist aru saama. Fookus on vestlusel. Suurt tähtsust omistatakse kuulajate kiitmisele, kui nad õigesti räägivad. Kui õpilased teevad vea, parandab õpetaja neid pehmelt. Praktikas esitab õpetaja tunni ajal küsimusi, millele õpilased vastavad. Teine võimalus on lugeda ette teksti ja siis teevad seda ka õpilased [4].

Meetodi “Callan” lõi 1960. aastatel Robin Callan. Õppimine põhineb peamiselt vestlusel, kuigi tegemist ei ole täielikult “loodusliku” meetodiga. Õpilastele esitatakse juhuslikke küsimusi ning nende ülesandeks on vastata võimalikult kiiresti ilma mõtlemata. Õpilased ei pea moodustama oma lauseid, vaid peavad harjutama ja kasutama õpitud mustreid. Tunnid toimuvad ainult inglise keeles. Materjali korratakse mitu korda ja vead parandab õpetaja kohe. Õpilased ei tee tunnis mingeid märkmeid [4].

“Keelekümbluse” meetod tähendab täielikku süvenemist keelde ja kultuuri. Keelekursuse raames on seda raske saavutada. Pigem on see meetod edukas reisides keelelaagritesse või välismaa tundides, kus seda keelt õpetatakse (näiteks inglise keele kursus Suurbritannias) [4].

“Kommunikatiivne” meetod on selline, mida tänapäeva õpilased kogevad filmide vaatamise, mängude mängimise ning kergeste vestluste pidamisel võõrkeeles. Tegelikult on suhtlustehnika puhul tegemist hästi läbimõeldud süsteemiga keeleoskuse arendamiseks läbi eluliste olukordade. Õpilased ei õpi grammatikat grammatika pärast ega loe ette lõputuid tekste. Selle asemel teevad nad läbi harjutusi, mis imiteerivad maksimaalselt neid olukordi, kus nad kasutaksid võõrkeelt päriselus. Samal ajal areneb mitte ainult kõnekeel (see on levinud eksiarvamus), vaid ka kirjutamine, lugemine ja kuulamine. Lõppude lõpuks sisaldub kõik ülaltoodu mõistes “suhtlus”. Viimaste aastakümnete jooksul on tehnikat täiustatud ning selle tõhusust on teaduslikult tõestanud enam kui ühe põlvkonna uuringud. Sellest sai alus peaaegu kõikide lääne kirjastajate, nagu Oxford University Press, Macmillan, Cambridge University Press, inglise keele kursustele. Samuti koostati selle põhjal ka tuntud rahvusvaheliste inglise keele tasemeeksamite, nagu IELTS ja TOEFL, vorm [3].

2.1.2 Riigikoolid, keeltekoolid ja järeleaitajad

Riiklikud koolid võtavad reeglina kasutusele “Grammatika ja tõlke” meetodi. Üksikutel õpetajatel on võimalus lisada õppeprotsessi ka teiste meetodite osi.

Keeltekoolidel nagu International House Tallinn, mis kuuluvad suurtesse rahvusvahelistesse organisatsioonidesse, on võimalus erinevaid meetodeid katsetada. Neil on võimalik meetodeid omavahel isegi ühendada, jälgides tulemust ja analüüsides oma lähenemist edasiste muudatuste sisseviimiseks. Järeleaitajad saavad pakkuda igale

õpilasele täiesti individuaalset lähenemist, lähtudes sellest, kuidas õpilane infot paremini omandab ning mida ta üldiselt peab õppima ja mõistma.

2.1.3 Raamatud ja õpikud

Kui keeleõppe õpikuid kasutatakse riigikoolides ja keeltekoolides, siis raamatuid kasutatakse peamiselt iseseisvaks õppimiseks.

Parimad õpikud inglise keele õppimiseks on [5]:

- English File Third Edition – Väljaandja: Oxford University Press (UK);
- New Total English – Väljaandja: Pearson Longman (UK);
- Speakout – Väljaandja: Pearson Longman (UK);
- Upstream – Väljaandja: Express Publishing (UK).

Parimad raamatud inglise keele iseseisvaks õppimiseks on:

- English Grammar in Use – Autor: Raymond Murphy & Martin Hewings [6];
- 4000 Essential English Words – Autor: Paul Nation [6];
- Face2Face – Autor: Chris Redston, Gillie Cunningham [6];
- How to Learn Any Language Fast and Never Forget it – Autor: Gabriel Wyner [7].

2.1.4 Veebi- ja mobiilirakendused

Iseseisvalt õppides pööratakse reeglina esialgu mitte raamatute ja õpikute, vaid veebi- ja mobiilirakenduste poole, kuna just viimased on enamasti tasuta ja alati käepärast. Selleks aga, et päriselt iseseisvalt ning selliste vahendite abil keelt selgeks õppida, on tarvis distsipliini.

Parimad leheküljed keele iseseisvaks õppimiseks kasutavad erinevaid teoreetilisi materjale, harjutusi ja kõnelejaid:

- FluentU – lehekülg keelte õppimiseks video abil [8];
- Conversation Exchange – sellel leheküljel saab suhelda kõnelejatega nende emakeeles [8];
- Test-english – sisaldab palju tasuta harjutusteste ja materjale, mis aitavad inglise keele oskust parandada [9].

IOS-i ja Androidi jaoks saadaolevad parimad tasuta mobiilirakendused võõrkeelte iseseisvaks õppimiseks sisaldavad erinevaid teooriatunde, harjutusi, emakeeles kõnelejaid ja mängu;

- Duolingo – sisaldab ülesannete rühmitamist raskusastmete järgi, treenib lugemisostust, kuulmis- ja visuaalset mälu [10];
- Memrise – teenus pakub põnevat õppimist kümnes keeles, suheldes veebis otse keelt emakeelena kõnelevate inimestega [10];
- Mondly – siin on kogu õppeprotsess jagatud lühikesteks ja hõlpsasti jälgitavateks tundideks [10];
- Drops – haridusbaasis on tuhat seitsesada sõna, mis on jagatud üheksakümmend üheksaks hariduslike illustratsioonide teemaks koos häälnäitlemise ja kiirete minimängudega [10];
- MosaLingua – see on helirakendus. Kordate ja tõlgite kõneleja järel sõnu, mida rakendus kontrollib [8].

2.2 Uue lahenduse skoop

Mõningate olemasolevate lahenduste suurimaks puuduseks on see, et nad põhinevad traditsioonilisel keeleõppemeetodil. Koolisüsteemide muutmine on väga suur ettevõtmine, mis toob endaga kaasa palju väljakutseid. Siiski võivad keeltekoolid, eraõpetajad, veebisaidid ja rakendused lubada endale katsetamist erinevate teabe esitamise meetoditega. Sellele vaatamata eelistavad inimesed ikkagi kasutada harjumuspärast lähenemist isegi siis, kui see on ebaefektiivne.

Sellega seoses pakub käesoleva lõputöö autor luua uut veebirakendust, mis on kohandatud ka mobiilseadmetes kasutamiseks. Rakendus on mõeldud assotsiatiivse teabetajuga kasutajatele. Antud taju areneb igal inimesel lapsepõlves, kuid mitte kõik ei kasuta seda edasi täiskasvanueas. Reeglina on just sellistel inimestel keerulisem omandada teavet traditsiooniliste meetoditega. Seetõttu võivad nad aastaid üritada õppida keeli, kuid edutult.

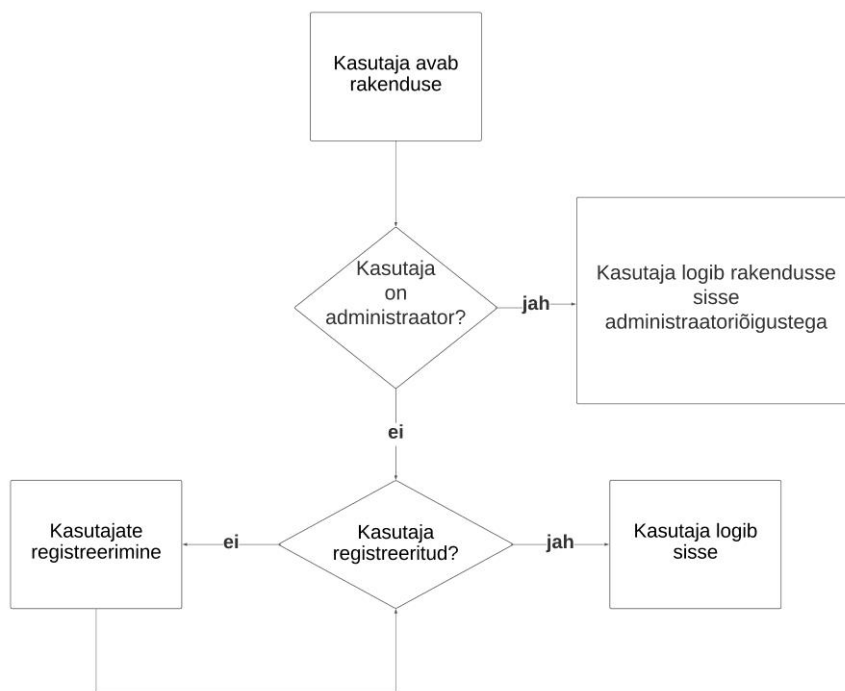
Eelnimetatud keskkond võimaldab ristsõnu lahendades õppida inglise keelt B1-tasemel. Samas jätab lahendus piisavalt ruumi ka edasiseks arenguks: võimalus õppida teisi keeli ning läbida muid harjutusi, mis samuti kasutaja assotsiatiivsele teabetajule mõjuvad.

Autor planeerib paluda kasutada rakendust inimestel, kes õpivad B1-tasemel inglise keelt, et saada potentsiaalsetelt kasutajatelt tagasisidet rakenduse kohta. Katses osalejad võiksid pärast keskkonna kasutamist oma muljeid jagada. Antud tagasiside põhjal on võimalik rakenduses vajalikke muudatusi sisse viia, samuti planeerida edasisi uuendusi ja lahenduse arendusi, mis on välja toodud peatükis “Rakenduse arendamine tulevikus”.

2.2.1 Koostatud plokkskeemid

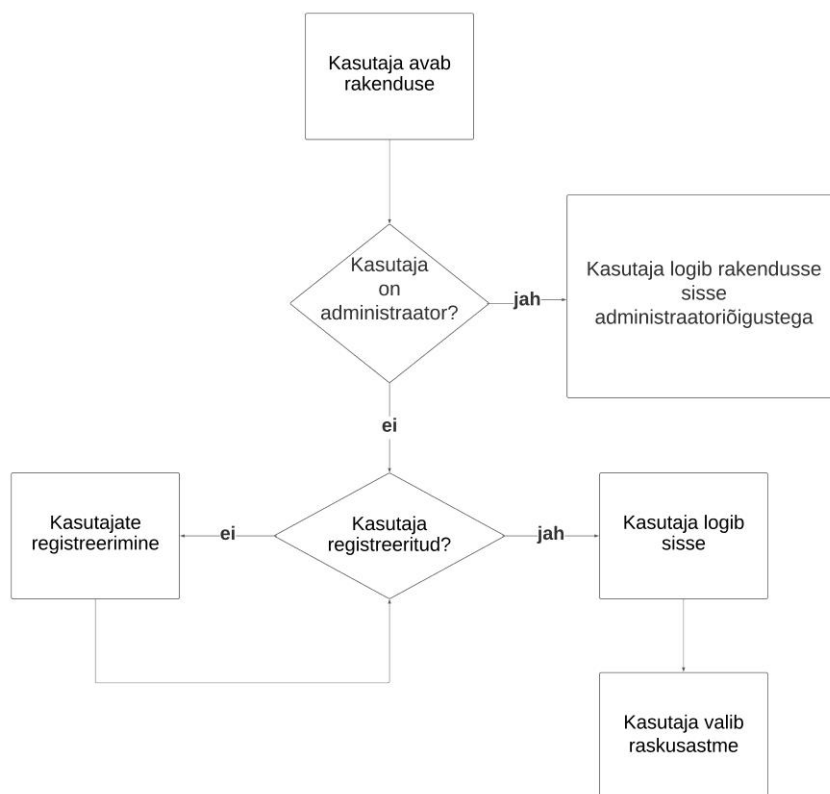
Kõigepealt koostati plokkskeeme ja kasutajalugusid iga kasutaja põhitegevuste kohta õpilase ja administraatori rollis. Kasutajalood on esitatud Lisas 2. Plokkskeemid annavad selgema ülevaate programmi loogikast ja funktsioonide järjestusest. Kasutajalood annavad selgema ülevaate kasutajate vajadustest.

Joonisel 1 on välja toodud kasutaja registreerimine õpilaseks. Esiteks kontrollitakse kasutaja rolli. Kui kasutaja ei ole administraator, kontrollitakse, kas kasutaja on juba registreeritud. Kui kasutaja ei ole registreeritud, siis peab kasutaja enne rakendusse sisenemist ennast registreerima.



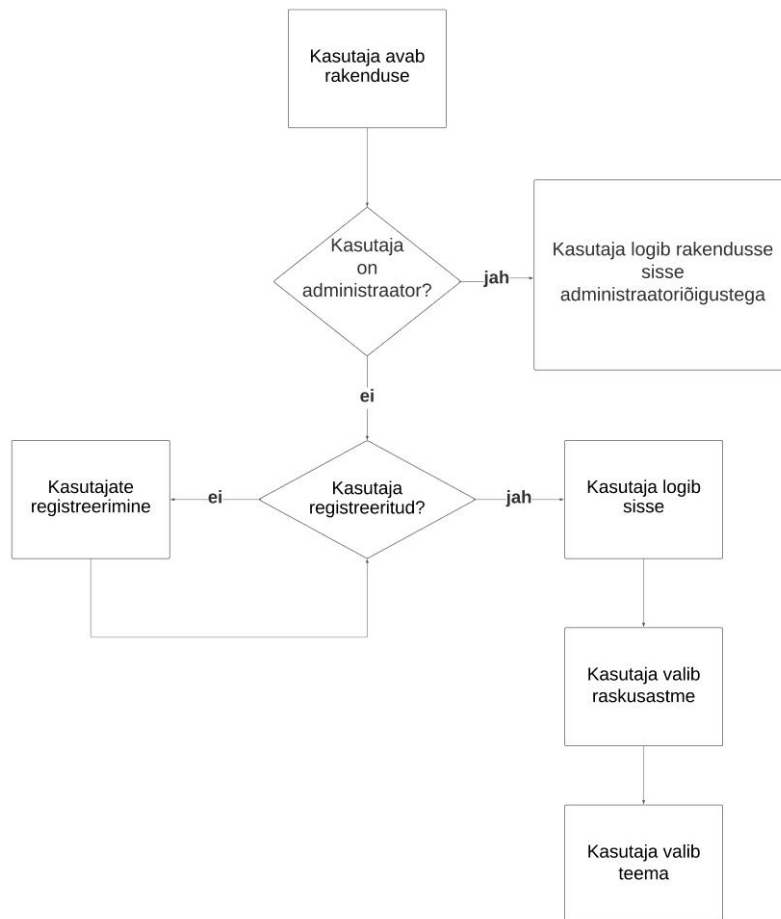
Joonis 1. Kasutaja registreerimise plokkskeem

Joonisel 2 on välja toodud keele raskusastme valimine. Kui kasutaja on registreeritud õpilaseks, avab ta ülesannete raskusastmete loendi, kus saab valida keele õppimiseks sobiva taseme.



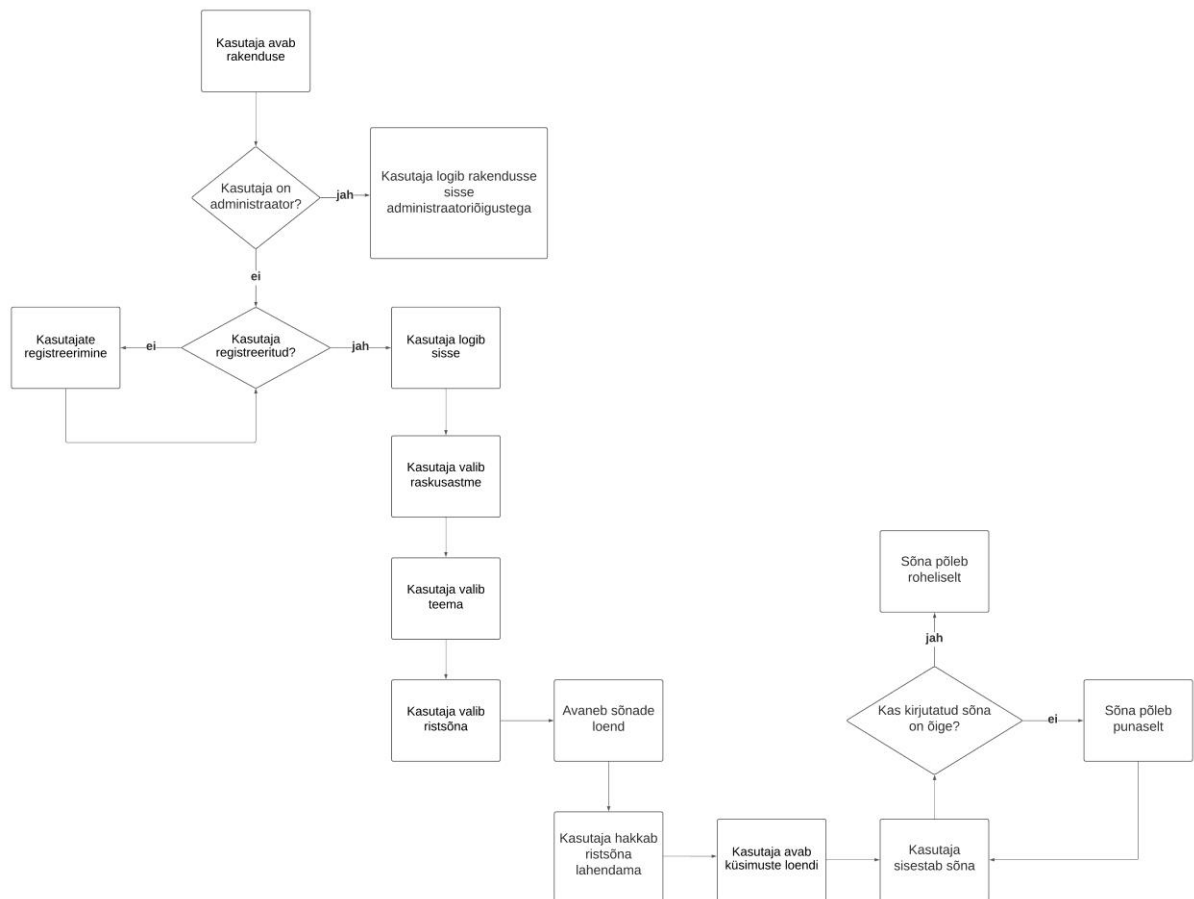
Joonis 2. Keele raskusastme valimine

Joonisel 3 on välja toodud teema valimine. Kui kasutaja on registreeritud õpilasena ja on sobiva keele raskusastme valinud, saab ta valida sobiva teema, mille raames ta edaspidi keelt õppima hakkab.



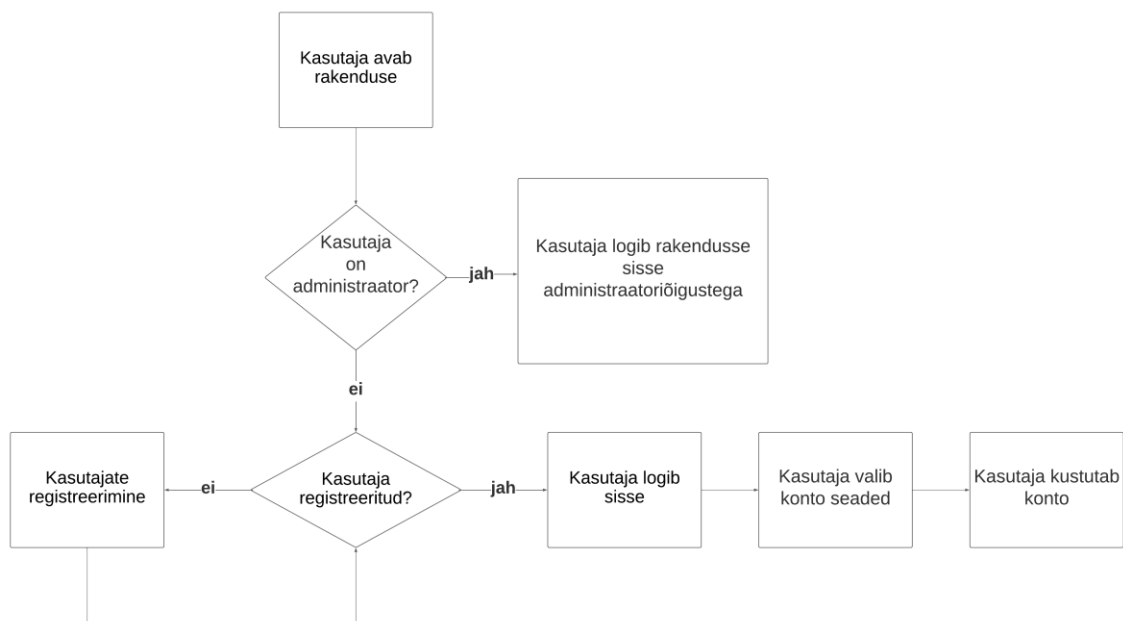
Joonis 3. Teema valimine

Joonisel 4 on välja toodud vale ja õige vastuse sisestamise protsess ristsõnas. Pärast teema valimist satub kasutaja erinevate ristsõnade loendiga lehele. Klõpsates ühele neist avaneb sõnade loend koos ingliskeelsete selgitustega. Selgitused sisaldavad kõige lihtsamaid sõnu, mida kasutaja juba teab, arvestades tema valitud raskusastet. Pärast sõnade loendi uurimist vajutab kasutaja nupule ja alustab ristsõna lahendamist. Olles avanud ristsõnaga lehe, näeb ta ruudustikku tähtede sisestamiseks ja nuppu küsimustega lehele minekuks. Varem esitatud küsimused ja selgitused on erinevad, kuid küsimused koosnevad samuti lihtsatest sõnadest, mida kasutaja teab. Õige sõna sisestamisel muutub taustavärv roheliseks ja vale sõna sisestamisel punaseks.



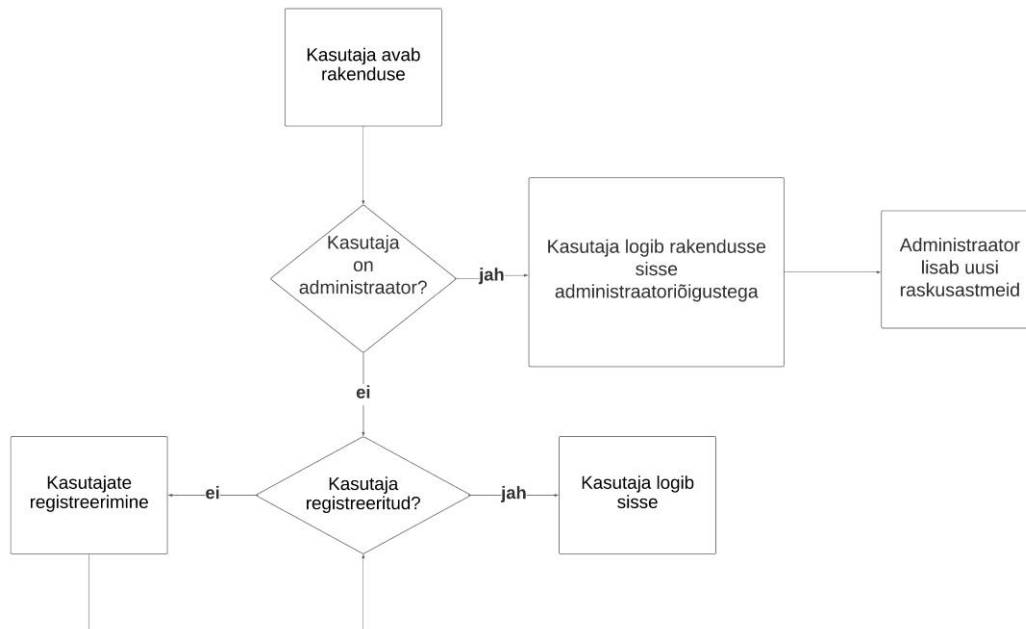
Joonis 4. Vale või õige vastuse sisestamine ristsõnades

Joonisel 5 on välja toodud õpilase konto kustutamine. Kui kasutaja soovib kunagi oma kontot kustutada, leiab ta selle funktsiooni konto seadetest.



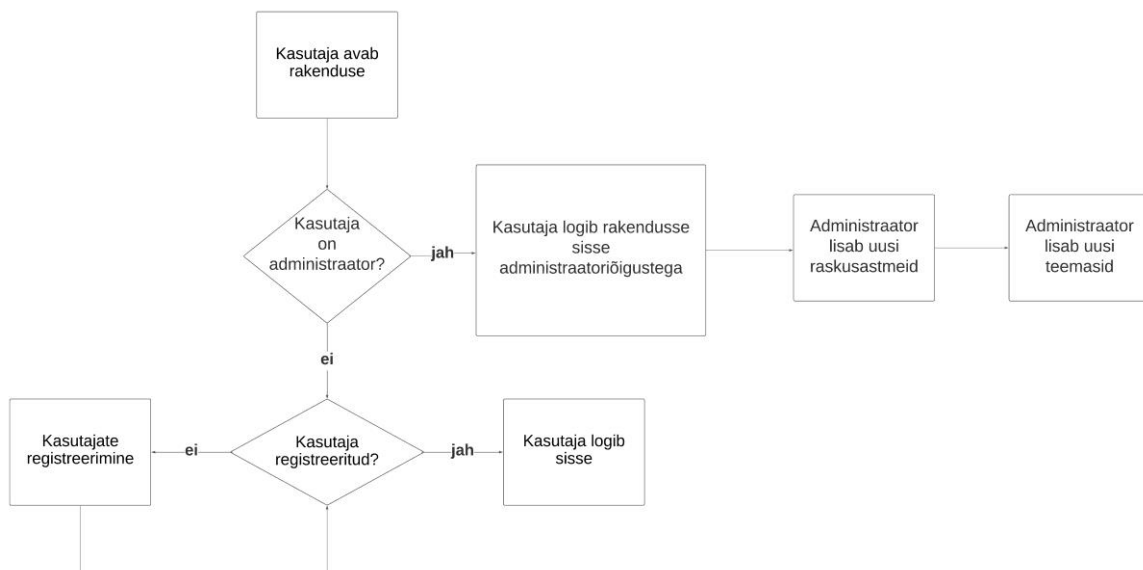
Joonis 5. Õpilase konto kustutamine

Joonisel 6 on välja toodud raskusastme lisamine administraatori abiga. Omades administraatori õigusi, saab kasutaja teha programmis muudatusi ja uuendusi, et selle funktsionaalsust laiendada. Tulevikus võivad lisanduda kõikvõimalikud keeleõppe tasemed.



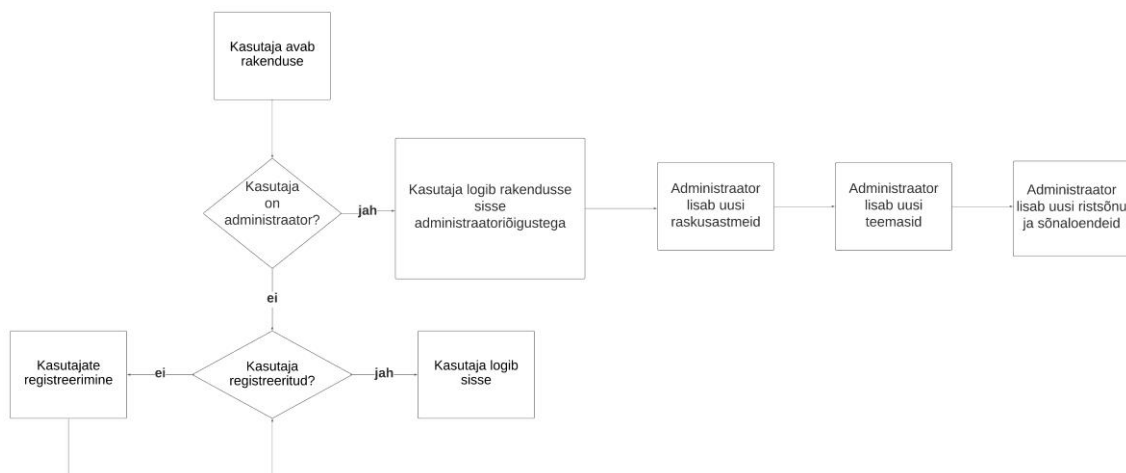
Joonis 6. Raskusastme lisamine administraatori abiga

Joonisel 7 on välja toodud teemade lisamine administraatori abiga. Samuti saab administraator lisada uusi teemasid, mille kohta edaspidi koostatakse ja lisatakse uusi ristsõnu.



Joonis 7. Teemade lisamine administraatori abiga

Joonisel 8 on välja toodud ristsõna lisamine administraatori abiga. Pärast teema loomist on võimalik sellele lisada ristsõnu. Esialgu on plaanis teha üks ristsõna ühe teema kohta, kuid edaspidi on võimalik iga teema kohta erinevaid ristsõnu lisada.



Joonis 8. Ristsõna lisamine administraatori abiga

2.2.2 Programmeerimiskeelte võrdlemine

PHP on laialt levinud avatud lähtekoodiga üldotstarbeline programmeerimiskeel. PHP on loodud spetsiaalselt veebirakenduste loomiseks ja selle koodi saab manustada otse HTML-i [11].

Tabel 1. PHP programmeerimiskeele eelised ja puudused

Eelised [12]	Puudused [12]
Teatud tasemel oskuste korral on võimalik mallimootori abil luua mitte ainult veebirakenduste skripte, vaid ka täisväärtuslikke programme	Kitsas profiil
On lahendusi, mis võimaldavad luua PHP baasil mobiilirakendusi	Vastuolud koodis. Kood on täis erinevate keelte jääke
Platvormideülesus. PHP-d saab käivitada mis tahes operatsioonisüsteemis, sealhulgas Unixoidides	
Tasuta levitamine	

C# on üks kiiremini kasvavaid, populaarsemaid ja samal ajal mugavamaid programmeerimiskeeli. See on põhikeele C modifikatsioon Microsoftilt, mille eesmärk on luua võimalikult universaalne tööriist tarkvara arendamiseks suure hulga seadmete ja operatsioonisüsteemide jaoks [13].

Tabel 2. C# programmeerimiskeele eelised ja puudused

Eelised [12]	Puudused [12]
Microsofti tugi.	Orienteerunud peamiselt .NET-ile
Paljud konstruktsioonid on loodud koodi kirjutamise ja mõistmise hõlbustamiseks ega mängi kompileerimisel rolli.	Tasuta ainult väikeettevõtetele, üliõpilastele ja vabakutselistele programmeerijatele.
Algajatele on see üks perspektiivikamaid keeli õppimiseks.	
Tänu Xamarini ostule on C#-s nüüd võimalik kirjutada Androidi ja iOS-i jaoks.	
Palju vabu töökohti C # programmeerija ametikohale mis tahes piirkonnas.	

Python on interpreteeritav objektorienteeritud kõrgetasemeline programmeerimiskeel, millel on dünaamiline semantika. Pythoni interpretaator ja ulatuslik standardteek on kõikidel suurematel platvormidel tasuta toor- või binaarses vormingus saadaval ning neid saab vabalt levitada [14].

Python asutaja põhieesmärk oli luua programmeerimiskeel, mis on lihtne ja arusaadav laiale hulgale inimestele. Python on mitmekülgne tänu oma mahukale standardteegile (funktsioonide komplektile, mis on saadaval ilma täiendava seadistamiseta), mistõttu kasutatakse seda väga erinevates valdkondades [15].

Tabel 3. Python programmeerimiskeele eelised ja puudused

Eelised [12]	Puudused [12]
Pythoni süntaks on loogiline, lühike ja selge, mis teeb selle algajate jaoks lihtsaks	Madal programmi täitmise kiirus võrreldes teiste keeltega
Interpretaatorid on olemas mobiilseadmete ja ebapopulaarsete süsteemide jaoks	Puudub võimalus Pythoni programmi teisendamiseks exe-ks
Kasutatakse veebirakenduste, mängude arendamiseks, mugav automatiseerimiseks, matemaatilisteks arvutusteks ja masinõppeks	Unicode ja vene tähemärgid
Suur nõudlus tööturul	
Põhjalik ja juurdepääsetav dokumentatsioon	

Java on kõrgetasemeline programmeerimiskeel, mille on välja töötanud Sun Microsystems. Java süntaks sarnaneb C++ omal, kuid Java on rangelt objektorienteeritud programmeerimiskeel [16].

Seda peetakse üheks populaarseimaks programmeerimiskeeleks ja seetõttu on see hästi dokumenteeritud.

Tabel 4. Java programmeerimiskeele eelised ja puudused

Eelised [12]	Puudused [12]
Kõrgetasemeline keel lihtsa süntaksi ja sujuva õppimiskõveraga.	Tasuline kaubanduslikel eesmärkidel kasutamiseks.
Standard ettevõtete arvutisüsteemidele.	Madal jõudlus.
Sõltumatus platvormist.	Paljusõnaline ja keeruline kood.
Keel hajutatud programmeerimiseks ja mugavaks kaugkoostööks.	
Mitmelõimelisus.	

3 Loodava veebirakenduse analüüs

Autor määrab skoobi alusel nõuded loodavale veebirakendusele ning pakub ka funktsionaalsuse laiendamist koos rakenduse edasiarendamisega.

3.1 Nõuete määramine

Funktsionaalsed ja mittefunktsionaalsed nõuded põhinevad plokkskeemidel ja kasutajalugudel, kus on põhirollideks õpilane ja adminisraator.

3.1.1 Funktsionaalsed nõuded

Rakenduse jaoks tuletati järgmised funktsionaalsed nõuded:

- süsteem peaks kasutajanime, parooli ja e-maili salvestama andmebaasi, kui andmebaasis sellist nime, parooli või e-maili veel ei ole;
- kasutaja saab ristsõnu lahendada ainult registreerituna;
- süsteem peaks salvestama kasutaja saavutusi ainult siis, kui ta on registreeritud;
- kasutajanime ja parooli pikkus peaks olema vähemalt 6 tähemärki;
- süsteem peaks õiged ja valed vastused ristsõnas ära tundma;
- süsteem peaks võimaldama administraatoril lisada uusi tasemeid, teemasid ja ristsõnu;
- enne ristsõna lahendamist esitatakse kasutajale kõigepealt sõnade loetelu koos selgitustega.

3.1.2 Mittefunktsionaalsed nõuded

Rakenduse jaoks tuletati järgmised mittefunktsionaalsed nõuded:

- materjal võetakse Oxfordi õpikust English File;
- rakendatakse metoodikat “uue keele õppimine uues keeles”;
- veebilahendus HTML5 on kohandatud kasutamiseks ka mobiilseadmetes;
- kui kasutajanimi, parool või e-maili aadress on juba kasutusel, siis kuvatakse veateade;
- kui parool või kasutajanimi on lühem kui 6 tähemärki, kuvatakse veateade;
- kujunduses tuleks kasutada selliseid värvikombinatsioone, mis mõjutavad mälu;
- õige vastuse sisestamisel tõstetakse sõna roheliselt esile;

- vale vastuse sisestamisel tõstetakse sõna punaselt esile;
- raskusastmed on kasvavas järjekorras;
- ristsõnade teemad on paigutatud samas järjekorras nagu õpikus;
- teemad on tähistatud kleebiste ja tekstiga;

3.1.3 Rakenduse arendamine tulevikus

Rakendust saab edasi arendada, lisades teiste õpikute põhjal täiendavaid raskusastmeid, teemasid, ülesandeid ja lisades sellele tehnikale tugineva ka teiste keelte õppimisse. On võimalik rakenduse funktsionaalsust veelgi laiendada:

- kontode kustutamine administraatori poolt;
- kasutajatele teadete saatmine;
- näpunäiteid ristsõnade lahendamiseks;
- rakenduse värvipaleti muutmine;
- konto linkimine meiliga;
- *offline* mobiiliversiooni arendamine.

Et pöörata suuremat tähelepanu tavakasutaja kogemust mõjutava funktsionaalsuse arendamisele, on planeeritavad funktsionaalsused jäetud väljapoole antud diplomitöö skoopi. Siia hulka kuuluvad administraatori jaoks mõeldud funktsionaalsused:

- administratorina soovin lisada raskusastmeid, et meelitada rohkem kasutajaid;
- administratorina soovin lisada teemasid, et võimaldada inimestel oma sõnavara eri suundades laiendada;
- administratorina soovin lisada uusi ristsõnu ja sõnade selgitusi, et inimesed saaksid uusi sõnu õppida.

3.2 Tehnoloogia valik

Kõigepealt peaks töö autor otsustama rakenduse arendamiseks kasutatava programmeerimiskeele ja valima sobivad arendustööriistad.

3.2.1 Programeerimiskeele valik

Arvestades programmeerimiskeelte eeliseid ja puudusi, on ka mõistlik kaaluda keeli, millega antud lõputöö autoril on kõige rohkem kogemust. Tabeli abil on näha, et see on Java ja Python.

Tabel 5. Programmeerimiskeelte võrdlus

Keel	Autori kogemus	Dokumentatsioon ja kogukond	Õppimiskeeruks [17]
PHP	Rahuldav	Ulatuslik	Keskmine
C#	Väike	Keskmine	Kõrge
Python	Hea	Ulatuslik	Madal
Java	Hea	Keskmine	Keskmine

Python oli algselt mõeldud lihtsalt arusaadava ja õpitava programmeerimiskeelena ning tänu sellele eelistavad seda algajad. Python on universaalne tänu oma mahukale standardteegile (funktsioonide komplektile, mis on saadaval ilma täiendava seadistamiseta), mistõttu kasutatakse seda väga erinevates valdkondades, sealhulgas veebiarenduses.

Lõputöö eesmärgi elluviimisel imponeerib autorile pigem Python, kuna ülikoolis veebiarendust selles keeles ei õpitud. Edaspidi plaanib autor selles konkreetset keeles oma oskusi aktiivselt edasi arendada, kuna viimastel aastatel on arendajate seas tendents aktiivsemalt Pythonile üle minna. Sel juhul on lõputöö hea algus selle keele tõsisemaks õppimiseks ning võib tulevikus ka portfooliosse lisanduda.

Pythoni üks peamisi eeliseid peitub tema enda filosoofias – koodi loetavuses. Võrreldes teiste programmeerimiskeeltega on Python oluliselt loetavam [18].

Pythoni kasutuse laienemine on aidanud kaasa ka kogukonna kiirele kasvule. Veebis on Pythoni kohta tohutult palju materjali, antud teemal on olemas suurepärased kursused. Teemaatilistes foorumites või Python Software Foundationis on lihtne leida vastuseid kõigile tekkivatele küsimustele [18].

3.2.2 Teenusepoolsed raamistikud

Stack on tehnoloogiate kogum, mille alusel arendatakse veebisaiti või rakendust. See sisaldab programmeerimiskeeli, *framework* (tarkvara arenduskeskkonda), andmebaasihaldussüsteeme, kompilaatorit (tõlgib programmeerimiskeeles kirjutatud teksti masinkoodi) jne. Konkreetse pinu valik sõltub projekti arhitektuurist, veebilehe keerukusest ja funktsionaalsusest. Samuti süsteemi nõuetest, mis määravad, millist operatsioonisüsteemi ja andmebaasihaldussüsteemi valida, millist veebiserverit ja programmeerimiskeelt kasutada [19].

Kõige populaarsemad veebiarendajad kasutatavad *technology stacks* [20]:

- LAMP – Linux, Apache, MySQL, PHP. LAMP-i peetakse kõige populaarsemaks pinuks ning seda hinnatakse selle skaleeritavuse, turvalisuse ja rohkete seadistamisvõimaluste poolest;
- Python-Django põhineb Pythoni programmeerimiskeelel. Samuti kasutab see Apache veebiserverit, MySQL andmebaasi ning Django infrastruktuuri, mis on samuti Pythonis kirjutatud. Arendajatele meeldib see pinu, kuna seda saab kiiresti arendada, säilitades kõrget kvaliteeti;
- MEAN – MongoDB, Express.js, AngularJS, Node.js. Siin sisaldab pinu struktuur nii sisemisi kui ka väliseid arendustööriistu, mistõttu nimetatakse seda täispinuks. MEAN pinu on tuntud oma suure jõudluse ja paindlikkuse poolest. Ühtlasi ka selle õppimise lihtsuse poolest, kuna see kasutab kõigis komponentides sama keelt;
- MERN – MongoDB, Express, React/Redux, Node.js. Arvestades ReactJSi populaarsust *frontend'is* ja NodeJS-i *backend'is*, on mõistlik nimetada seda kombinatsiooni enimkasutatavaks JavaScripti pinuks, mis on mõeldud suure jõudlusega üheleheliste rakenduste loomiseks. Kui võrrelda MEAN-i MERN-iga, taandub valik Reactile või Angularile, ning siin on oluline arvestada projekti nõudeid ja eesmärke, samuti arendajate teadmisi JavaScripti kasutamisest.

Tuntumad teenusepoolsed raamistikud:

- RoR – populaarne raamistik, mida on lihtne õppida ja kasutada, skaleeritav ja kiire, mis kasutab ära disainimustreid DRY ning MVC. Vajab rohkem võimsust serveri poolelt [21];
- Spring – avatud lähtekoodiga rakendusraamistik, põhineb Javal. Spring keskendub ettevõtte rakenduste “torustikule”, et meeskonnad saaksid keskenduda rakenduse tasemel ärioloogikale ilma tarbetute sidemeteta konkreetsete juurutuskeskkondadega [22];
- Django – kiire ja skaleeritav raamistik, mis on väga hästi dokumenteeritud ning millega kaasneb isiklik server [21];
- .NET – suletud lähtekoodiga raamistik .NET pinu hulka kuuluv, mis omab põhjalikku dokumentatsiooni, head infrastruktuuri tuge ning võimaldab erinevate veebirakenduste kiiret arendamist [21];

- Node.js – võimaldab optimeerida koodi suure jõudlusega, keerukates ja andmemahukates reaalajarakendustes. See on kiire, lihtne ja väljendusrikas. Sobib kõige paremini rakendustele, mis hõlmavad reaalajas voogesitust ja koostööööriistu [21];
- Express.js – nõuab väga vähe pingutust ning kujutab endast minimaalset paindlikku raamistikku, mis kasutab oma ressursse tõhusalt. Sobib vaid lihtsamate veebirakenduste loomiseks [21];
- Flask – aktiivse kasutajabaasiga dokumenteeritud raamistik, mis on hea valik, kui rakendusele saadaolevad ressursid on piiratud [21];
- Laravel – avatud lähtekoodiga serveriplatvorm, mida kasutatakse laia valiku kohandatud veebirakenduste loomiseks. See on täielikult serveripoolne raamistik, mis haldab andmeid MVC konstruktsiooniga, mis jagab rakenduse sisemise arhitektuuri loogilisteks osadeks [23].

Arendajate seas populaarseimad veebiraamistikud [20]:

Tabel 6. Populaarseimad veebiraamistikud

Keel	Raamistik
JavaScript	Node.js
Ruby	RoR
Python	Django
PHP	Laravel
C#	.NET

Võttes arvesse autori poolt valitud programmeerimiskeelt, on koostatud Django ja Flask võrdlev tabel, kuna mõlemad raamistikud sobivad Pythoni baasil arendamiseks [24]:

Tabel 7. Django ja Flask võrdlus

Django ja Flaski võrdlusalus	Django	Flask
Struktuur	Django on tasuta ja avatud lähtekoodiga Pythonil põhinev keskkond, mis vastab sellele, kuidas MVC läheneb arhitektuurimustrile.	Flask on Pythonil põhinev mikroraamistik, millel pole spetsiaalseid tööriistu ega väliseid teeke. Samuti pole sellel andmebaasikihti ega vormi kontrollimise tingimusi ja see kasutab laiendusi.
Omadused	<p>Komponendid:</p> <ul style="list-style-type: none"> ▪ Relatsiooniline andmebaasi mudel ▪ Veebimallide süsteem ▪ Regulaaravaldistel põhinev URL-i haldur ▪ Pärimine ▪ Vahemällu salvestamine ▪ Vahevara klasside tugi ▪ Rahvusvahelistumine ▪ Ühiku testimise moodul <p>Seotud rakendused:</p> <ul style="list-style-type: none"> ▪ Autentimissüsteem ▪ Administraatori liides ▪ Google-i veebilehe kaardid ▪ Serveri asukoht 	<ul style="list-style-type: none"> ▪ Arendusserver ▪ Arenduse silumine ▪ Moodulitestimise sisseehitatud tugi ▪ Jinja2 mallid ▪ RESTful päringute saatmine ▪ Turvaliste küpsiste tugi ▪ Ulatuslik dokumentatsioon ▪ Unikoodi baasil ▪ Laiendused soovitud funktsioonide tarbeks ▪ Lihtne mooduli tüüpi ülesehitus ▪ Paindlikkus ▪ HTTP päringute tagamine ▪ API-liidestel on meeldiv vorm
Neid raamistikke kasutavad veebilehed	Mozilla, Instagram, The Washington Times, BitBucket, Disqus, Nextdoor.	Pinterest, LinkedIn ja Flask
Projektiplaan	Tüüpiline projekti struktuur	Meelevaldne struktuur
Paindlikkus	Ei välista seadistamise paindlikkust	Ollakse arvamusel, et kõikvõimalikud Flask koodi organiseerimise permutatsioonid on võrdsed juba Flaskis olevate rakenduste arvuga.

Django ja Flaski võrdlusalus	Django	Flask
Marsruutimise mehhanism	Urls.py-d kasutatakse ühenduse atribuutide määramiseks ja päringut töödeldakse esimese sobiva regulaaravaldiste loendi esitamisega.	URI-d enamasti ei paigalda vaate kujundaja ning tsentraliseeritud konfiguratsioon on võimalik. Enne mallide URI-ga vastavusse viimist sorteeritakse viimast vaikejärjestuses.
Eelised	<ul style="list-style-type: none"> ▪ Brauseri API ▪ Perioodilised ja regulaarsed väljaanded ▪ Jäik rakenduse struktuur ▪ Funktsionaalne administraatori paneel ▪ Suur kogukond ▪ Teiste rakenduste toetamine ▪ Kirjeldav ja konstruktiivne dokumentatsioon 	<ul style="list-style-type: none"> ▪ Kiirus ▪ NoSQL tugi ▪ Minimaalne raskusaste ▪ Täielik minimalism ▪ ORM-i pole, lihtsal moel seotud laiendustega ▪ Siluja on brauserisse sisse ehitatud ▪ Lühike ja lihtne kood teiste Pythoni veebilehe ülesehituste hulgas

Pärast raamistiku valimise juhiste uurimist otsustas selle lõputöö autor proovida töötada Django raamistikuga.

3.2.3 Kliendipoolne tehnoloogia (frontend)

Frontend on veebirakenduste avalik osa, millega saab kasutaja otse interaktsiooni astuda. Esiosa sisaldab kliendi funktsionaalsete ülesannete ja kasutajaliidese kajastamist ning kasutajapäringute töötlemist [25].

Järgnevalt on välja toodud mõningad populaarsemad raamistikud ja paketid [21]:

- Bootstrap – CSS-i raamistik, mida valida, kui arendate mobiilseadmete rakenduse jaoks. See on kohandatav ja säästab veebisaidi arendamise aega, kuna seda on lihtne kasutada ja see pakub kasulikke komponente;
- Angular – universaalne JS-põhine raamistik; sobib hästi projektidele, kus koodide lihtne integreerimine ja kiire veebirakenduse arendamine on hädavajalikud. Uue Angular 5-ga on tänu sisseehitatud koodi optimeerijale tööaega vähendada lihtne. Parim üheleheliste veebirakenduste ja platvormideüleste mobiilirakenduste,

samuti sihtlehtede ja levinud veebisaitide arendamiseks. Kasutab: Google, PayPal, Upwork;

- Vue.js – JavaScripti raamistik, mis integreerub hõlpsalt JS-i teekidega. See on hea valik dünaamiliste, suuremahuliste ja üheleheliste projektide jaoks. Kasutab: Alibaba, WizzAir, Grammarly;
- React – JS-i teek kasutajaliideste (UI) loomiseks. Sellel on aktiivne kogukond ja arvukalt valmiskomponente, mis säästavad arendusaega. See sobib suurepäraselt igasuguste veebirakenduste või platvormide loomiseks, mis nõuavad väga tundlikku kasutajaliidest. Kasutavad: Facebook, Reddit, Netflix;
- JQuery – JS-i teek, mida kasutatakse koodi optimeerimiseks, nt kasutades meetodi kirjeldamiseks tosina koodirea asemel ühte koodi.

Need lahendused on täiesti tasuta, enamik neist on hästi dokumenteeritud ja sobivad kiireks arendamiseks. Käesoleva lõputöö autoril on osaline kogemus Bootstrapi, Reacti ja Vue.js-ga. Kuna Bootstrapi on lihtne kasutada ning selle raamistiku elemendid sobivad omavahel hästi kokku, tänu millele saab tagada mugavat vaatamiskogemust mobiilseadmetes, valiti just see käesoleva projekti *frontend* loomiseks.

3.3 Andmebaasi valik

Andmebaase laiemas mõttes on iseloomult mitmesuguseid ja need on jagatud mitmetesse kategooriatesse: puustruktuuriga, relatsioonilised, objekt-orienteeritud, dokument-orienteeritud ja võti-väärtus andmebaasid [26].

Relatsioonilises andmebaasis on andmed organiseeritud teatud põhimõtete järgi ja seotud terviklikkuse reeglitega. On andmete puhul tegemist spetsiifiliste väljadega tabelites, tabelite hulk kokku moodustabki relatsioonilise andmebaasi [27].

Järgnevalt on välja toodud mõned tuntumad relatsioonilised andmebaasid:

- SQLite – on manustatud andmebaas avatud lähtekoodiga, mis töötab rakenduse osana ja on täielikult välja töötatud C-keeles. SQLite kirjutab andmed otse faili ja saab üsna hõlpsalt liikuda ja sobib ideaalselt väikese andmehulga jaoks. SQLite kasutatakse tavaliselt mobiilirakenduste jaoks [28];
- MySQL – on kõige populaarsem avatud lähtekoodiga andmebaas, mis nõuab kliendi kaudu võrgu kaudu suhtlemiseks andmebaasiserverit. MySQL saab

hakkama mitme samaaegse ühendusega ja seal saab luua mitu kasutajat, kellel on erinevad õigused ja rollid. See andmebaas pakub ja toetab paljusid autentimismeetodeid, et kaitsta loata juurdepääsu andmebaasile. MySQL-i kasutatakse tavaliselt veebirakenduste ja töölaarakenduste jaoks, mis nõuab palju arvutusi ja sagedasi operatsioone [28];

- PostgreSQL – on kõige võimalusterohkem ja suurfirmade vajadusi silmas pidav avatud lähtekoodiga andmebaasitarkvara. Kuna PostgreSQL katab ka vajadused mida SQLite ning MySQL rahuldavad ning paljugi enam, on see kõige sobivam tarkvara, mille peale üks relatsiooniliste andmebaaside koolitus üles ehitada [26].

Dokument-orienditud andmebaaside puhul on tegemist andmed salvestatakse JSON või XML formaadis. Selle asemel, et andmeid ridadesse ja veergudesse salvestada, kasutab see andmete salvestamiseks ja hankimiseks dokumente [29].

Järgnevalt on välja toodud mõned tuntumad dokument-orienditud andmebaasid [30]:

- MongoDB – kergelt skaleeritav, väga paindlik ja sellel on palju käsitsi seadistamist vajavaid osi. See salvestab andmed binaarselt kodeeritud JavaScript Object Notation või BSON-dokumentide kogudesse. MongoDB saab hoolduseks toetada ainult CLI tööriistu. Ja mõnikord on vaja selleks kasutada ka kolmanda osapoolse kasutajaliidese tööriistu;
- CouchDB – on avatud lähtekoodiga dokumendile orienteeritud NoSQL andmebaas. Couchbase-il on integreeritud administraatorikonsool ja see konfigureerib kõik ühes kohas ning kõiki hooldustoiminguid saab teha selliste tööriistade abil nagu veebiliides, käsurea liides (CLI) või REST API.

Objekt-orienditud andmebaas on andmebaasisüsteem, mis suudab töötada keerukate andmeobjektidega, st objektidega, mis peegeldavad objekt-orienditud programmeerimiskeeltes kasutatavaid objekte. Objekt-orienditud programmeerimises on kõik objektid ja paljud objektid on üsna keerulised, neil on erinevad omadused ja meetodid [31].

Objekt-orienditud andmebaasi struktuur on graafiliselt kujutatud puuna, mille sõlmedeks on objektid. Objekti atribuute kirjeldab mõni standardtüüp või kasutaja loodud tüüp [32].

Järgnevalt on välja toodud mõned tuntumad objekt-orienteeritud andmebaasid [33]:

- ObjectDB – ObjectDB on võimas objektorienteeritud andmebaasihaldussüsteem. See on kompaktne, töökindel, lihtsalt kasutatav ja ülikiire. ObjectDB pakub kõiki standardseid andmebaasihaldusteenuseid, kuid viisil, mis muudab arenduse lihtsamaks ja rakendustele kiiremaks;
- WakandaDB – pakub sisseehitatud REST API-t, et pääseda juurde serveripoolses JavaScriptis määratletud andmeklassidele. WakandaDB on Wakanda server, mis sisaldab spetsiaalset, kuid mitte kohustuslikku Ajaxi raamistikku ja spetsiaalset IDE.

Võti-väärtus andmebaasid on kõige paindlikumad NoSQL andmebaasid, kus salvestatavad andmed pole mitte vaid dokumentide kujul, vaid võivad olla ka numbrid, sõnad, pildid, videod ja palju muud. Tavaliselt pakub see suurepärasest jõudlust ja seda saab optimeerida vastavalt organisatsiooni vajadustele. Võti-väärtuste hoidlatel pole päringukeelt, kuid need pakuvad võimalust võti-väärtus paaride lisamiseks ja eemaldamiseks. Väärtusi ei saa küsida ega otsida. Küsida saab ainult võtit [34].

Üks tuntumaid võti-väärtus andmebaase on Redis, mille tugevuseks on selle kiirus nagu teistelgi NoSQL andmebaasidel.

3.4 Veebirakenduse disain

Veebilehtede välimus on kasutajate jaoks ülimalt tähtis. Kasutajad tahavad kiiresti ja lihtsalt leida vajalikku teavet. Samuti eeldavad nad, et Interneti-ressurs on ka mobiilseadmete jaoks kohandatud. Allpool on välja toodud mõningad uuringute põhjal tehtud järeldused kujunduse mõjust kasutajate reaktsioonile [35]:

- 48% inimestest peavad veebilehe kujundust esmaseks faktoriks, mis määrab otsust, kas informatsioon on usaldusväärne või mitte;
- 2/3 inimestest eelistavad lugeda pigem kauni kujundusega artiklit kui lihtteksti;
- 38% inimestest lõpetab saidi sirvimise, kui sisu ei tundu nende jaoks atraktiivne;
- veebiressursi omanikul on 10 sekundit aega, et edastada olulist informatsiooni platvormi kohta ja jätta selle kohta õige mulje;
- 3/4 külastajatest vajutavad tagasi-nupule, kui nad satuvad veebilehele, millel on liiga palju reklaamplokke, pilte ja kirjud fondid.

Keeltekoolis, kus õpib käesoleva lõputöö autor, soovitatakse täiendavaks keeleõppeks kasutada veebilehte Test-english. Kodulehel on meeldiv kujundus, mis mõjub kasutaja meeltele. Sellel põhjusel on kujunduse loomisel võtnud käesoleva lõputöö autor aluseks eelnimetatud kodulehte.

3.5 Analüüsi kokkuvõte

Analüüsis käsitles autor erinevaid kasutajatüüpe ning nende funktsionaalseid ja mittefunktsionaalseid nõudeid. Analüüsis on samuti esitatud rakenduse funktsionaalse komponendi võimalikke uuendusi ning uusi funktsionaalseid nõudeid selleks, et vajaduse korral rakendust edasi arendada.

Progromeerimiskeelte hulgast langes valik Pythonile, kuna käesoleva töö autoril on sellega kogemusi. Pythonil on põhjalik tugi ulatusliku dokumentatsiooni, arenduskeskkondade ja raamistike näol. Samuti on see arendamisel arusaadav ning lihtne. Algselt valiti esiosa loomiseks Bootstrap, kuna seda on lihtne kasutada ning selle elemendid on üksteisega kooskõlas. See omakorda aitab säästa aega, mis kulub rakenduse kujundamise peale.

Kood on kirjutatud PyCharmis ja andmebaasilahendus on avatud lähtekoodiga SQLite, sest see on standartne andmebaas, mida kasutatakse Django raamistikuga. Antud lahenduse kujunduse inspiratsiooniallikaks on koduleht Test-english, mida soovitasid inglise keele õpetajad.

4 Veebirakenduse arendus

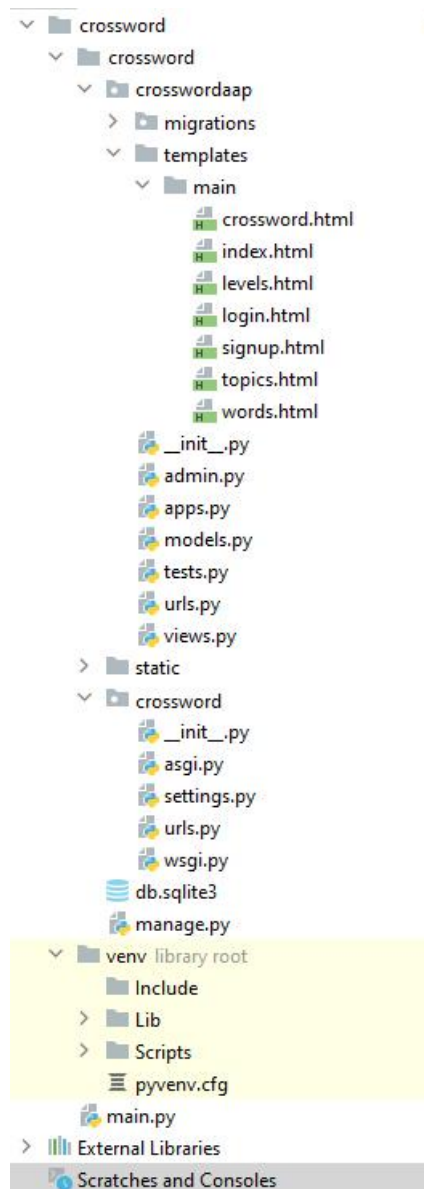
Pärast projekti loomist PyCharmis, installeeriti raamistikku Django *pip install django* käskluse abil. Edasi luuakse selles projektis uus project Django baasil *django-admin startproject crossword* käskluse abil, mis sisaldab faili *manage.py* ning põhirakendust dubleeriva projektinimega. Järgmisena loodi selles Django projektis uus rakendus *python manage.py startapp crosswordapp* käskluse abil, mis on samal tasemel nagu fail *manage.py* ja põhirakendus.

HTML failid – rakenduse *crosswordapp* lehemallid asuvad kaustas *templates\main*. Kui kasutaja läheb rakenduse lehele, siis *templates* kaustas käivitatakse ette faili *urls.py*, mis omakorda kutsub välja *views.py* ette. Fail *views.py* sisaldab meetodeid lehte näitamise jaoks. Faili *models.py* kasutatakse edaspidi mudelite jaoks ja faili *tests.py* testide jaoks.

Fail *settings.py* sisaldab kõiki globaalseid projekti sätteid. Kui loodi uus rakendus tuleb see registreerida selles failis. Samuti sisaldab see fail näiteks andmebaasi nime ja sätteid, millega töö toimub, ning CSS-i ja Javascripti failide tee.

Django ülesehitust ning loogikat peetakse mugavaks, kuna ühe rakenduse kõik funktsionaalsus on ühes kaustas. Seega, kui on vaja luua täpselt samasugust rakendust, siis piisab selle kausta dubleerimisest.

Joonisel 9 on esitatud ekraanipilt loodud projekti struktuurist.



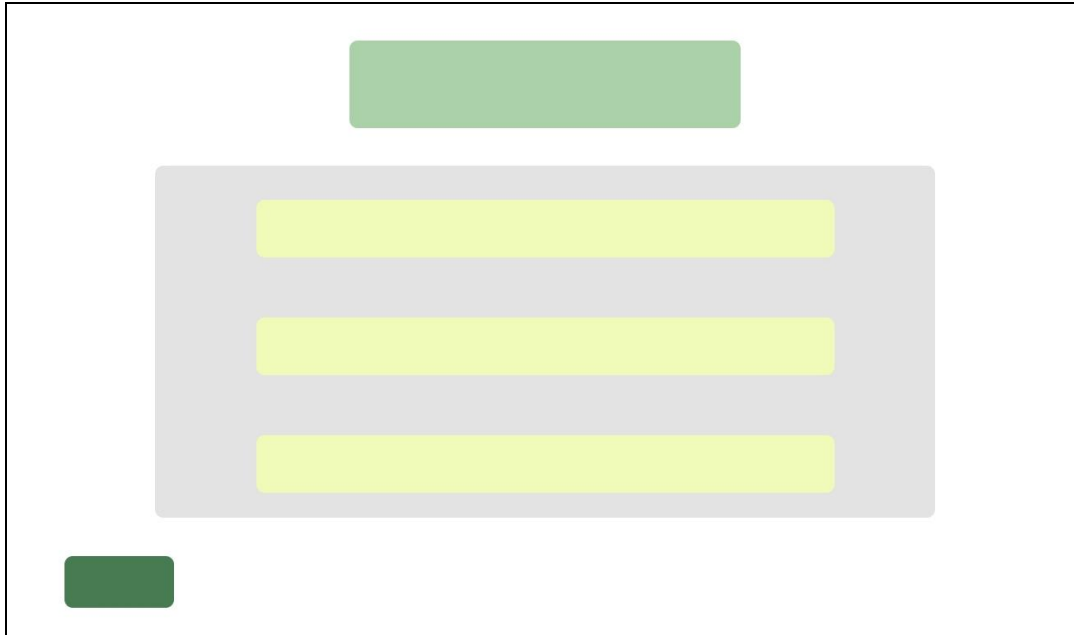
Joonis 9. Loodud projekti struktuur

4.1 Veebilehe komponentide planeerimine

Paljud uuringud näitavad, et mida lihtsam on kasutajal rakendust kasutada, seda rohkem see kasutajale meeldib. Arvestades seda, eelistas käesoleva töö autor minimalismi lehe paigutuses ja elementide arvus.

Joonisel 10 on kujutatud “*Levels*” lehe kuvamiseks vajalikud komponendid. Kasutaja pääseb sellele lehele pärast sisselogimist või registreerumist, et valida inglise keele raskusaste õpimise jaoks. Valitud raskusastmest sõltub loend teemadega, mida kasutaja näeb järgmisel veebilehel.

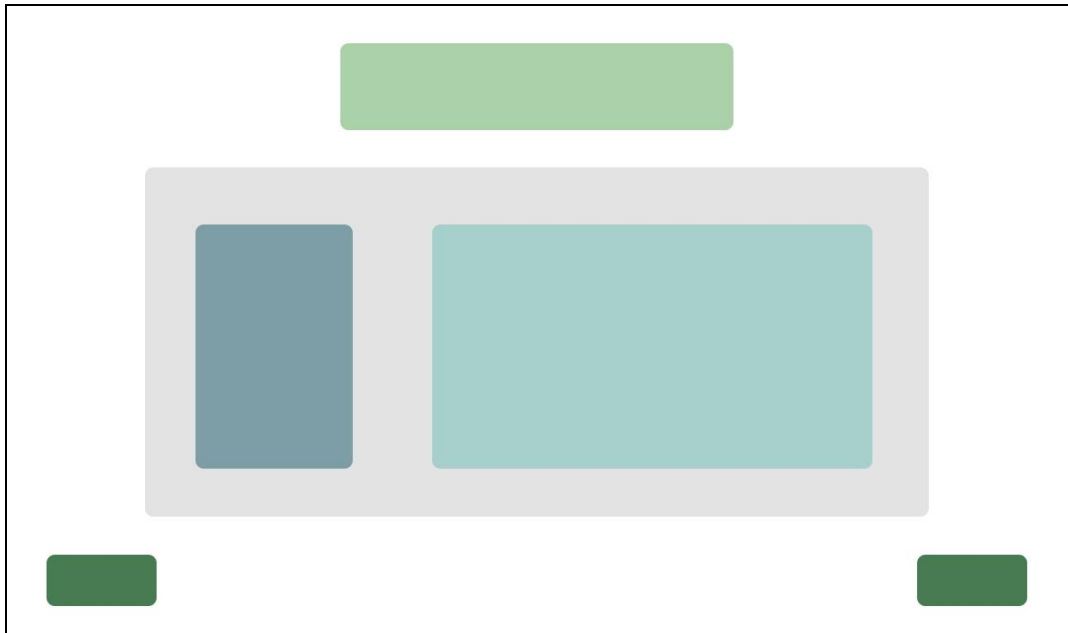
Heleroheline kast on lehe nimi, mis annab kasutajale teada, mida näeb lehel ja mida on vaja valida. Kollased kastid on taseme nimedega kaardid, millest igauks suunab kasutaja selle raskusastme teemade juurde. Tumeroheline komponent on nupp, mis võimaldab kasutajal tagasi minna eelmisele veebilehele.



Joonis 10. „Levels” lehe komponentide diagramm

Joonisel 10 on kujutatud „*List of words*” lehe kuvamiseks vajalikud komponendid. Pärast teema valimist satub kasutaja sõnade loendiga lehele. Sellel lehel on selgitud sõnad, mida ta peaks ristsõnade lahendamise jaoks kasutama.

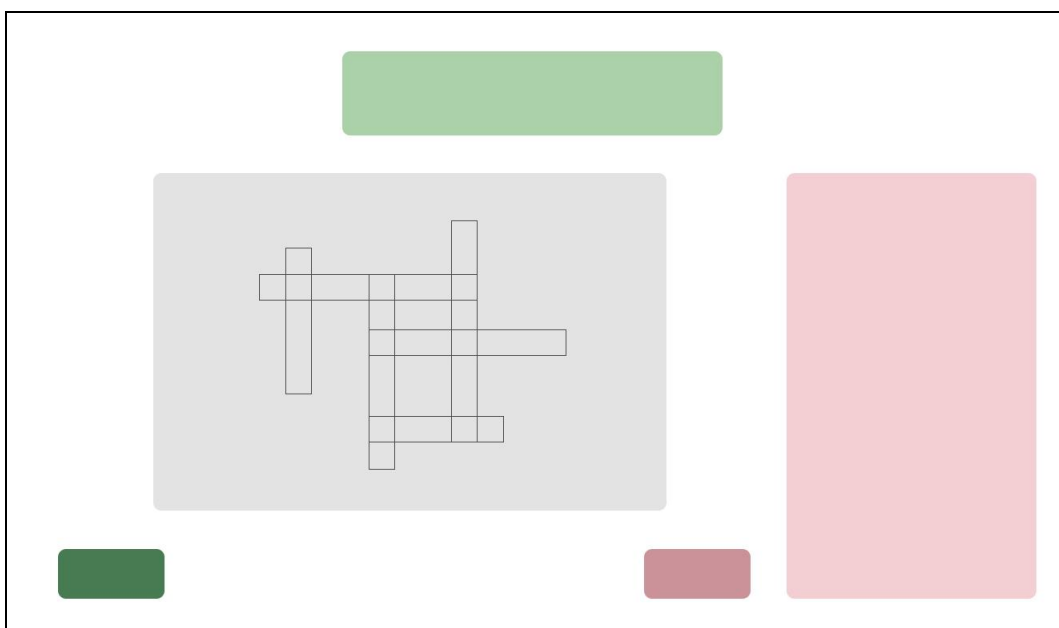
Helesinine kast on loend sõnadest, mida kasutaja kasutab ristsõnas ja tumesinine kast on sõnade selgitustega plokk. Tumeroheline komponent vasakul on nupp, mis võimaldab kasutajal tagasi minna eelmisele veebilehele ja tumeroheline komponent paremal on nupp, mis võimaldab kasutajal minna järgmisele veebilehele.



Joonis 11. „List of words” lehe komponentide diagramm

Joonisel 12 on kujutatud „Crossword” lehe kuvamiseks vajalikud komponendid. Pärast sõnade loendi uurimist, vajutab kasutaja nupule ning läheb sellele lehele, et oma teadmisi kontrollida ristsõna lahendamise abil.

Hall kast on komponent, millel ristsõna kuvatakse. Roosa kast on küsimuste loend ristsõna lahendamise jaoks. Tumeroosa komponent on nupp sisestatud sõna kontrollimiseks.



Joonis 12. „Crossword” lehe komponentide diagramm

Paljud kasutajad eelistavad neutraalset värviskeemi, mille üks värv on valitud aktsendiks ja teised on neutraalsed. Arvestades seda on valitud roheline aktsentide jaoks, sest see tekitab turvatunnet, usaldust ja loob positiivse mängumeeleolu. Abivärvide rolliks valiti valge ja must nagu kõige neutraalsemad värvid.

4.2 Bootstrapi kasutus

Eialgu oli plaanis Bootstrap3 *framework* projekti installeerida, kuid tekkis probleem, et Django ei näe *static* kausta, mis sisaldab CSS faile. Sellel põhjusel käesoleva töö autor soovis Bootstrap kasutada jsDelivr CDN abil. See annab võimaluse Bootstrapi malle importida HTML faili linkide abil. Samuti lisati mõne elemendi puhul stiili omadused HTML *tag*-idele eraldi.

Bootstrap CDN on maailma populaarseima CSS-i raamistiku levitamine, kasutades sisu edastamise võrku. See on geograafiliselt jaotatud serverite rühm, mis töötavad koos, et pakkuda Interneti-sisu kiiret edastamist [36].

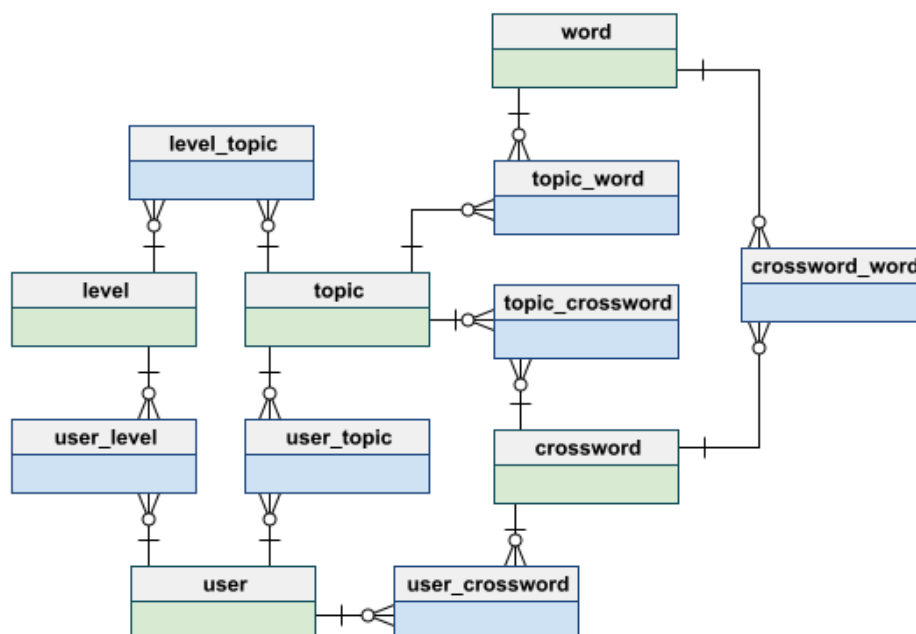
```
<!-- CSS only -->
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
  rel="stylesheet" integrity="sha384-
1BmE4kWBq78iYhF1dvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
crossorigin="anonymous">

<!-- JavaScript Bundle with Popper -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/d
```

Koodinäide 1. CDN kasutamise näide Bootstrapi importimiseks

4.3 Andmebaas

Lähtuvalt plokk skeemidest ning kasutajalugudest tuli arenduse protsessi käigus hea ettekujutus loodavast andmebaasi struktuurist ning olemitest. Olemi-suhte diagramm on loodud tööriista Vertabelo abil ja on esitatud Lisas 3.



Joonis 13. Lihtsustatud olemi-suhte diagramm

Loodava rakendus andmebaas on vajalik andmete salvestamiseks kasutajate, keeletasemete, teemade ning koostatud või lahendatud ristsõnade kohta.

SQLite andmebaas on loodud *python manage.py migrate* käskluse abil. Olemi-suhte diagrammi baasil on andmebaasi ehitamise jaoks automaatselt kirjutatud SQL kood Vertabelo tööriistas. Pärast *org.sqlite.JDBC* draiveri installimist PyCharmi nõuandel, loodi andmebaasis olevate elementidega tabelid.

4.4 Valminud rakenduse kasutamine ning edasiarendused

Tulevikus, kui rakendus on viimistletud autori kavandatud tulemuseni, saaks käesoleva töö autor rakendust kasutada inglisekeelsete sõnade õppimiseks esmase või sekundaarse allikana. Täiendava arendusega saavad rakendust kasutada ka teised inimesed. See sobiks neile, kelle keeletase on A2-B1, kuna sõnade selgitamiseks kasutatakse väga lihtsat inglise keelt.

See rakendus on üsna usaldusväärne teabeallikas, kuna põhineb ühel maailma populaarseimal inglise keele õppimise õpikul. Arvestades seda, on rakendust võimalik kasutada A2-B1 tasemega õpilaste õpetamiseks International House Tallinn koolis, kus käesoleva töö autor õpib.

Kui arendada rakendust edasi ja näiteks lisada teisi keeletasemeid, siis on võimalik rakendust kasutada kõikides gruppides International House Tallinn koolis. Samuti, kui lisada võimalus teisi keeli õppida sama metoodikaga, saab teha koostööd teiste keeltekoolidega. Et võimalikult paljud inimesed seda metoodikat prooviksid.

Rakenduse kasutajate jaoks on võimalik lisada ka funktsioone nagu näpunäiteid ristsõnade lahendamiseks, värviteemade muutmine võimalus ja *offline* mobiiliversiooni. *Offline* versiooni on vaja selleks, et kasutajad saaksid vabal ajal ristsõnu lahendada ilma võrku ühendamata. Kõik see muudaks rakenduse paljudele tulevastele õpilastele atraktiivsemaks. Nii neile, kes õpivad keelt iseseisvalt või soovivad sellega alustada, ja ka neile, kes õpivad keeli keeltekoolides, riigikoolides või ülikoolides.

Rakenduse funktsionaalsuse laiendamine võib sisaldada ka administraatori jaoks lisavõimalusi. Näiteks kasutajatele teadete saatmine või kontode kustutamine administraatori poolt ja vastavad põhifunktsioonid, näiteks tasemete või teemade lisamine otse rakenduses administraatori kontolt.

5 Kokkuvõte

Töö autor koostas alguses plokk skeemid ja kasutajalood ning nende baasil funktsionaalsed ja mittefunktsionaalsed nõuded veebirakenduse analüüsi osas. Seejärel uuris autor eksisteerivaid lahendusi, programmeerimiskeeli, raamistikke ja andmebaase; vahendite valik toimus koostatud võrdlustabelite abil. Arenduseks sai valitud Python programmeerimiskeelena, Django raamistikuna, Bootstrap teenusepoolse raamistikuna ja SQLite andmebaasina. Arendus toimub PyCharmi keskkonnas.

Arenduse käigus loodi kõigepealt veebilehtede komponentide diagrammid, mis vastavad kasutajate peamistele eelistustele ja töö autor kasutas soovitud raamistikku jsDelivr CDN abil, mis võimaldas saada diagrammidele vastava *frontend*. Plokk skeemide ja kasutajalugude põhjal koostati olemi-suhte diagramm, et vajalikku andmebaasi paremini ette kujutada. Diagrammid on loodud Vertabeloga. Samuti on Vertabelos tehtud SQL kood konstrueeritud diagrammi põhjal. Seda koodi kasutati projekti andmebaasis atribuutidega tabelite koostamiseks.

Loodav rakendus on MPA tüüpi, see kujutab endast mitut HTML lehte, mida kuvatakse erinevate URL-ide puhul. Rakendus sisaldab õppematerjali "Oxford English File" õpikust ja see on esitatud tehnikat "inglise keele õppimine inglise keeles" kasutades. Rakendus on kohandatud mobiilseadmetes kasutamiseks ja kasutaja saab registreeruda või sisse logida õpilasena, valida keeleõppeks taseme B1 ja soovitud teema ning uurida sõnade loendit koos ingliskeelsete selgitustega.

Kui autor arendusega soovitud tulemuse saavutab, siis saab ta rakendust kasutada ingliskeelsete sõnade õppimiseks esmase või sekundaarse allikana ja edasiarendusena saaksid rakendust kasutada ka teised inimesed. Sellel juhul on võimalik teha koostööd International House Tallinn kooliga. Rakenduse funktsionaalsuse laiendamine võib sisaldada õpilase ja administraatori jaoks lisavõimalusi ning teiste keelte õppimine võimalust koostööks teiste keeltekoolidega.

On plaanis saada pärast arenduse lõpetamist potentsiaalsetelt kasutajatelt tagasisidet rakenduse kohta. Kui rakendust katsetavad keeleõppijad, siis arvestatakse saadud tagasisidet kindlasti edasises arenduses.

Kasutatud kirjandus

- [1] Simplex, „Самые востребованные иностранные языки - Что учить?“, Mai 2014. [Online]. Available: <https://simplex.ua/articles/samie-vostrebovannie-inostrannie-yaziki>.
- [2] Englhouse, „Kuidas teha ristsõna inglise keeles“, [Online]. Available: <https://englhouse.ru/et/uchashhimsya/kak-sdelat-krossvord-po-anglijskomu-yazyku.html>.
- [3] J. T. Ksenia Ljubskaja, „Методики изучения иностранных языков“, Märts 2021. [Online]. Available: <https://education.forbes.ru/authors/metodiki>.
- [4] SpeakUp, „5 самых популярных и эффективных методов изучения английского языка“, November 2020. [Online]. Available: <https://speak-up.com.ua/ru/news/5-samih-populyarnih-metodov>.
- [5] Englex, „4 лучших учебника по английскому языку“, Juuni 2015. [Online]. Available: <https://englex.ru/4-best-textbooks-detailed-review/>.
- [6] A. Skorohodova, „Топ лучших учебников по английскому языку для самостоятельного изучения“, September 2018. [Online]. Available: <https://eng911.ru/beginners/dlya-samostoyatelnogo-izucheniya.html>.
- [7] FluentU, „The 10 Best Language Learning Books“, [Online]. Available: <https://www.fluentu.com/blog/best-language-learning-books/>.
- [8] A. Boretskiy, „13 лучших сайтов для бесплатного изучения иностранных языков“, Märts 2020. [Online]. Available: <https://vc.ru/u/388199-andrey-boretskiy/111394-13-luchshih-saytov-dlya-besplatnogo-izucheniya-inostrannyh-yazykov>.
- [9] Test-english. [Online]. Available: <https://test-english.com/>.
- [10] VseKursy, „15 лучших бесплатных приложений для изучения иностранных языков“, Aprill 2020. [Online]. Available: <https://vse-kursy.com/read/738-prilozheniya-dlya-izucheniya-inostrannyh-yazykov.html#>.
- [11] PHP, „What is PHP?“, [Online]. Available: <https://www.php.net/manual/ru/intro-what-is.php>.
- [12] ITVDN, „Плюсы и минусы языков программирования“, Märts 2020. [Online]. Available: https://teletype.in/@itvdn_it/zDfuiuHU.
- [13] Timeweb, „Язык программирования C#“, Aprill 2021. [Online]. Available: <https://timeweb.com/ru/community/articles/chto-takoe-csharp>.
- [14] Python, „What is Python?“, [Online]. Available: <https://www.python.org/doc/essays/blurb/>.

- [15] S. Šaškov, „Почему Python — один из самых популярных языков нашего времени,“ Märts 2020. [Online]. Available: <https://rb.ru/young/IT-python/>.
- [16] TechTerms, „Java,“ April 2012. [Online]. Available: <https://techterms.com/definition/java>.
- [17] J. Kartavets, „Какой язык программирования мне выбрать?,“ Märts 2021. [Online]. Available: <https://blog.skillfactory.ru/kakoj-yazyk-programirovaniya-mne-vybrat/>.
- [18] D. Žarikov, „Почему Python — отличный выбор для web-разработки?,“ [Online]. Available: <https://wezom.academy/pochemu-python--otlichnyj-vybor-dlja-web-razrabotki/>.
- [19] A. Sokolova, „Что такое стек технологий,“ Oktober 2021. [Online]. Available: <https://secretmag.ru/enciklopediya/chto-takoe-stek-tekhnologii-obyasnyajem-prostymi-slovami.htm>.
- [20] P. В., „Какой технологический стек выбрать для разработки сайта?,“ Juuni 2019. [Online]. Available: <https://ru.linkedin.com/pulse/%D0%BA%D0%B0%D0%BA%D0%BE%D0%B9-%D1%82%D0%B5%D1%85%D0%BD%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9-%D1%81%D1%82%D0%B5%D0%BA-%D0%B2%D1%8B%D0%B1%D1%80%D0%B0%D1%82%D1%8C-%D0%B4%D0%BB%D1%8F-%D1%80%D0%B0%D0%>.
- [21] V. Struk, „How to Choose a Technology Stack for Your Web Application,“ [Online]. Available: <https://relevant.software/blog/how-to-choose-a-technology-stack-for-your-web-application/>.
- [22] T. Contributor, „Spring Framework,“ [Online]. Available: <https://www.techtarget.com/searcharchitecture/definition/Spring-Framework>.
- [23] A. Burets, „What Is Laravel Used for – Most Common Applications,“ Juuni 2021. [Online]. Available: <https://scand.com/company/blog/what-is-laravel-used-for/>.
- [24] EducationWiki, „Django vs Flask,“ [Online]. Available: <https://ru.education-wiki.com/9156325-django-vs-flask>.
- [25] DanIt, „Разработка со стороны Frontend,“ Jaanuar 2020. [Online]. Available: <https://dan-it.com.ua/blog/razrabotka-so-storony-front-end-chto-jeto-takoe-i-chem-otlichaetsja-ot-back-end/>.
- [26] L. Võsandi, „Andmebaaside võrdlus,“ [Online]. Available: <https://lauri.xn--vsandipxa.com/rdbms/comparison.html>.
- [27] V. Kehtna, „Andmebaaside põhimõisted,“ [Online]. Available: <https://vainol.kehtna.edu.ee/andmebaasid/abpohimoist.htm>.
- [28] EducationWiki, „MySQL vs SQLite,“ [Online]. Available: <https://et.education-wiki.com/4341906-mysql-vs-sqlite>.

- [29] D. G. Gonzalez, „Andmebaasid,“ [Online]. Available: <https://www.linuxadictos.com/et/andmebaasid-v%C3%A4ike-sissejuhatus-teemasse.html>.
- [30] C. T. Labs, „The Essential Difference Between Couchbase & MongoDB,“ Juuni 2018. [Online]. Available: https://medium.com/@Cubet_Techno_Labs/the-essential-difference-between-couchbase-mongodb-5b4d7d192cff.
- [31] MongoDB, „What Is an Object-Oriented Database?,“ [Online]. Available: [https://www.mongodb.com/databases/what-is-an-object-oriented-database#:~:text=An%20object%20oriented%20database%20\(OOD,having%20different%20properties%20and%20methods](https://www.mongodb.com/databases/what-is-an-object-oriented-database#:~:text=An%20object%20oriented%20database%20(OOD,having%20different%20properties%20and%20methods).
- [32] Leally, „Objektorienteeritud andmebaasid,“ [Online]. Available: <https://leally.ru/et/polezno-znat/obektno-orientirovannye-bazy-dannyh-dostizheniya-i-problemy/>.
- [33] CsharpCorner, „What Are Object-Oriented Databases And Their Advantages,“ September 2019. [Online]. Available: <https://www.c-sharpcorner.com/article/what-are-object-oriented-databases-and-their-advantages2/#:~:text=Common%20examples%20are%20Smalltalk%20is,%2C%20calculations%2C%20and%20faster%20results>.
- [34] Redis, „What is a Key-Value Database?,“ [Online]. Available: <https://redis.com/nosql/key-value-databases/>.
- [35] ITFactory, „Как дизайн сайта влияет на поведение пользователей,“ Oktoober 2019. [Online]. Available: <https://vc.ru/marketing/88898-kak-dizayn-sayta-vliyaet-na-povedenie-polzovateley>.
- [36] Themesberg, „What is Bootstrap CDN,“ 2020. [Online]. Available: <https://themesberg.com/knowledge-center/bootstrap/cdn>

Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks¹

Mina, Ksenija Kozlova

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „Veebirakenduse loomine võõrkeele õppimiseks ristsõnamõistatuste abil”, mille juhendaja on Antoi Meelis
 - 1.1. reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
 - 1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

11.04.2022

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 – Kasutajalood

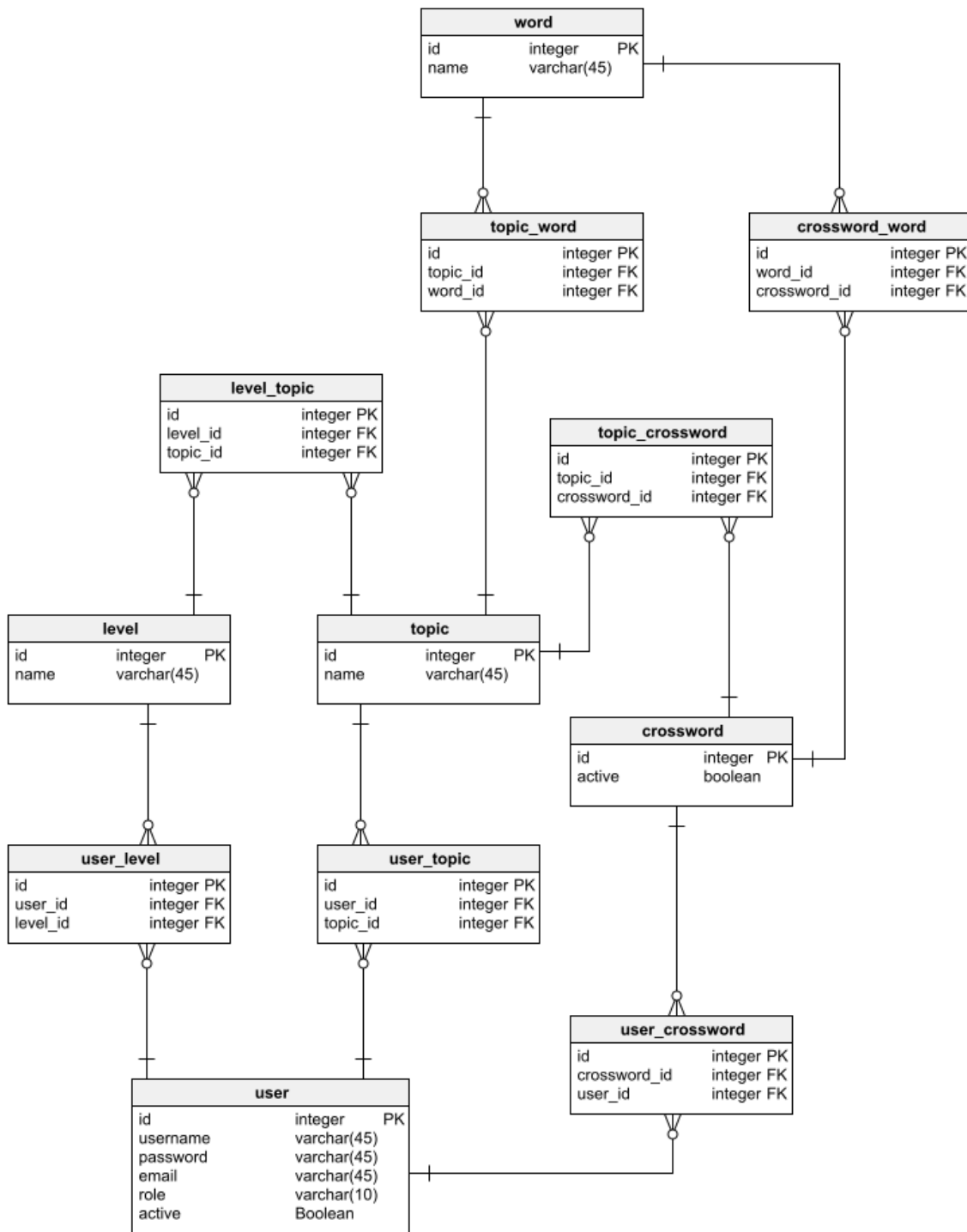
<i><Kasutajatiüübina></i>	<i>Soovin <täita mõnda ülesannet></i>	<i>et saaksin <saavutada mingi eesmärgi></i>
Õpilasena	Soovin registreeruda,	et minu ristsõna tulemused salvestataks
Õpilasena	Soovin, et saaksin valida ristsõna teema,	et saaksin õppida sõnu sellel teemal, mis mind huvitab
Õpilasena	Soovin, et mul oleks palju erinevaid teemasid,	et saaksin oma sõnavara igas suunas laiendada
Õpilasena	Soovin valida raskusastet,	et saaksin aru ingliskeelsetest sõnade selgitustest
Õpilasena	Soovin näha, et sisestasin õige vastuse,	et mõista, kas ma mäletasin sõna õigesti
Õpilasena	Soovin näha, et sisestasin vale vastuse,	et teaksin, et seda tuleb muuta
Õpilasena	Soovin oma konto kustutada,	et minu andmed andmebaasist kustutataks

Tabel 8. Õpilase kasutajalood

<i><Kasutajatiüübina></i>	<i>Soovin <täita mõnda ülesannet></i>	<i>et saaksin <saavutada mingi eesmärgi></i>
Administratorina	Soovin lisada raskusastmeid,	et meelitada rohkem kasutajaid
Administratorina	Soovin lisada teemasid,	et võimalda inimestel oma sõnavara eri suundades laiendada
Administratorina	Soovin lisada uusi ristsõnu ja sõnade selgitusi,	et inimesed saaksid uusi sõnu õppida

Tabel 9. Administraatori kasutajalood

Lisa 3 – Olemi-suhte diagramm



Joonis 14. Olemi-suhte diagramm andmebaasi jaoks