

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Marti Mölder 134570IAPB

**TEADMISTE KONTROLLIKS
VALMISTUMISE PROGRAMMID
ÕPILASTELE - ANALÜÜS JA PROTOTÜÜBI
KOOSTAMINE**

Bakalaureusetöö

Juhendaja: Inna Švartsman
Magister

Tallinn 2020

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: Marti Mölder

08.01.2020

Annotatsioon

Lõputöö eesmärgiks on analüüsida õpilastele suunatud tasuta pakutavaid õpiabi programme, teostada analüüsi tulemusi kasutades teadmiste omandamise kontrolli programmi prototüüp, mis võtaks üle enamuse analüüsi käigus selgunud positiivsetest omadustest ning pakkuda kasutajale autori lahenduse puuduva osa korvamiseks.

Analüüsiks valis autor 4 populaarsemat tasuta pakutavat õpiabi programmi. Analüüsi käigus selgus, et kõik valitud programmid kasutavad variatsiooni *Spaced repetition* tehnikast. Autor leidis, et antud tehnika elemente on võimalik kasutada ka prototüübis, eemaldades *Spaced repetition* tehnikast aja faktori.

Analüüsi tulemusena selgus, et kõigis valikus olnud õpiabi programmides, kas puudus täielikult või oli väga puudulikult lahendatud õppimise protsessi tagasiside andmine kasutajale. Kuna programmid kasutavad *Spaced repetition* õppimistehnikat, siis vastavalt tehnika omapärale puudus efektiivne võimalus õppida pikkade vastustega küsimusi.

Peale analüüsi teostamist töötas autor välja teadmiste omandamise kontrollimise programmi prototüübi, milles demonstreerib võimalikke lahendusi õpilastele, kes soovivad kontrollida oma teadmisi testides, mis sisaldavad pikkade vastustega küsimusi ja saada adekvaatset tagasisidet õppimise protsessi edukuse kohta.

Samuti töötas autor välja õpiabi programmi prototüübis kasutamiseks mõeldud algoritmi, kasutades selleks programmide analüüsi tulemusi, tehes selleks oma edasiarenduse *Spaced repetition* algoritmile, et paremini kohalduda kasutajate ootustega.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 26 leheküljel, 7 peatükki, 14 joonist.

Abstract

Knowledge Check Preparation Programs for Students - Analysis and Prototype Creation

The thesis aims to analyze the students directed learning support programs offered free of charge and to use the results of the analysis to implement a prototype of a knowledge acquisition control program that would take over most of the positive features revealed during the analysis and provide the user with a solution for missing part.

For analysis, the author selected the 4 most popular free learning support programs. The analysis revealed that all of the selected programs use a variation of the technique *Spaced repetition*. The author found that elements of this technique can also be used in a prototype by removing the time factor from the Spaced repetition technique.

After conducting the analysis, the author developed a prototype of the Knowledge Acquisition Testing Program, which demonstrates possible solutions for students who wish to test their knowledge in tests that include long-answer questions and receive adequate feedback on the success of the learning process.

The author also developed an algorithm to use in the prototype of a learning aid program, using the results of program analysis and further developing the *Spaced repetition* algorithm to better meet user expectations.

The thesis is in Estonian and contains 26 pages of text, 7 chapters, 14 figures.

Sisukord

1 Sissejuhatus	8
2 Teadmiste kontrolli õppimise protsessi ülevaade	9
2.1 Teadmiste kontrolli kirjeldus	9
2.2 Konspekterimine	9
2.3 Teadmiste omistamine	10
2.4 Teadmiste omistamise kontrollimine	10
3 Olemasolevate õpiabi programmide kirjeldus	12
3.1 Anki	12
3.2 Quizlet	14
3.3 Tinycards	16
3.4 Goconqr	17
4 Olemasolevate õpiabi programmide plussid ja miinused	19
4.1 Anki plussid	19
4.2 Anki miinused	20
4.3 Quizlet plussid	21
4.4 Quizlet miinused	21
4.5 Tinycards plussid	21
4.6 Tinycards miinused	22
4.7 Goconqri plussid	22
4.8 Goconqri miinused	22
5 Õpiabi programmide analüüsi kokkuvõte	24
6 Teadmiste omistamise kontrolli prototüüp	25
6.1 Prototüübi küsimuste lisamine	27
6.2 Prototüübi eksami koostamine	28
6.3 Prototüübi eksami sooritamine	29
6.4 Prototüübi tagasiside kuvamine	31
6.5 Prototüübi esileht	31
6.6 Prototüübi võrdlus olemasolevate õpiabi programmidega	32

7 Kokkuvõte	34
Kasutatud kirjandus	35

Jooniste loetelu

Joonis 1. Teadmiste kontrolliks õppimise skeem	9
Joonis 2. „Test“ tüüpi õppimise seaded.....	15
Joonis 3. „Sobitamine“ tüüpi õppimise viis.	15
Joonis 4. „Gravitatsioon“ tüüpi õppimise viis.....	16
Joonis 5. Tinycards näide küsimuse küsimisest.	17
Joonis 6. Kasutaja tegevused prototüübis.....	25
Joonis 7. Kasutatud oleva algoritmi joonis.....	26
Joonis 8. Küsimuste lisamise vaade.	27
Joonis 9. Eksami lisamise vaade.	28
Joonis 10. Eksami sooritamise vaade.	29
Joonis 11. Loodud eksam.	29
Joonis 12. Küsimuste kontrollimise vaade.	30
Joonis 13. Tagasiside kuvamise vaade.	31
Joonis 14. Prototüübi esilehe vaade.....	32

1 Sissejuhatus

Teadmiste kontrolliks õppimine on tehnoloogia arenedes märgatavalt muutunud. Pidevalt lisandub juurde uusi õpiabi programme, mis püüavad ühel või teisel moel aidata õpilasel muuta õppimise protsessi mugavamaks ja efektiivsemaks.

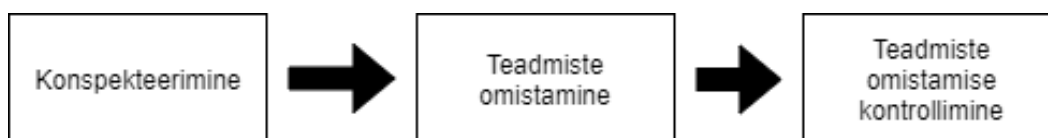
Antud bakalaaurusetöös analüüsib autor olemasolevaid populaarsemaid õpilastele suunatud tasuta õpiabi programme. Analüüsitavate programmide eelduseks on, et need peavad olema suutelised pakkuma õpilasele võimaluse korrata teadmiste kontrolliks ja peavad olema saadaval tasuta.

Autor, tuginedes isiklikule kogemusele teadmiste kontrolliks õppimise protsessist, mis koosneb konspekterimisest, teadmiste omistamisest ja teadmiste omistamise kontrollimisest leiab, et enamus teadmiste omistamise kontrollimise programme on pigem suunatud õpetajatele ja õppejõududele.

Lähtuvalt analüüsi tulemustest koostab autor õpiabi programmi prototüübi, mis võimaldab õpilasel ise koostada endale sobiliku teadmiste omistamise kontrolli.

2 Teadmiste kontrolli õppimise protsessi ülevaade

Teadmiste kontrolliks õppimiseks on mitmeid erinevad viise ja meetodikaid. Selguse huvides kasutatakse antud bakalaureusetöös alljärgneval skeemil põhinevat õppimist.



Joonis 1. Teadmiste kontrolliks õppimise skeem

2.1 Teadmiste kontrolli kirjeldus

Teadmiste kontrollile on väga palju erinevaid tüüpe. Olgu see siis näiteks eksam, tavaline tunnikontroll või kontrolltöö. Selguse säilitamise huvides määratleb autor ära, et tegemist on eksamiga. Õppejõud on andnud eksamiks kordamisküsimused. Kordamisküsimusi on kokku 10 tükki ja eksamisse tuleb neist 3. Ehk eksam koosneb 3 küsimusest, mis on suvaliselt valitud 10 kordamisküsimuse hulgast.

Küsimused on antud kujul küsimus-vastus. Küsimus on keskmiselt 10 sõna pikk. Vastuste pikkus on 50 ja 60 sõna vahel. Ehk küsimuste vastused on suhteliselt pikad.

2.2 Konspekteerimine

Konspekteerimise alla kuulub teadmiste omandamine erinevatest materjalidest. Näiteks raamatud, loengud, õpetajat konspekteerides, internetist otsides. Antud protsessi osa eesmärk on koguda võimalikult palju teemakohast materjali, mille alusel on võimalik hiljem õppida.

Antud protsessi konspekteerimise osa lihtsustavad programmid nagu „Evernote“ ja „OneNote“. Nendes programmides on võimalik koguda info kokku ja sorteerida need sobilikesse teemadesse nii, et hiljem oleks võimalikult efektiivne ja mugav õppida.

Raamatutest info kogumine on samuti tehtud märgatavamalt kergemaks, kuna suurem osa raamatuid on saadaval e-raamatute näol andes võimaluse kergelt kopeerida ja kokku koguda õppimiseks vajalik informatsioon.

Seega konspekterimise osa on võimalik teha selliseks, et kogu informatsioon on kätte saadava arvuti ekraanilt, mis võimaldab kergesti lugeda ja otsida teadmiste kontrolliga seonduvat infot. Arvutist info hoiustamise eeliseks on ka see, teadmiste omistamisel on võimalus kergelt taaskasutada olemasolevat infot, ilma et õppija peaks ise kogu informatsiooni uuesti ümber kirjutama.

2.3 Teadmiste omistamine

Teadmiste omistamise alla kuuluvad õpiabi programmid, mis aitavad õppijal omistada eelnevalt konspekteritud materjale. Osa eesmärk on teha selgeks võimalikult palju materjali, et teadmiste kontroll võimalikult edukalt läbida. Antud osas omab võtmetähtsust aja efektiivne kasutus.

Aja kasutus on oluline, kuna õpilase ajaressurss on piiratud ei ole mõtet seda raisata küsimuste peale, mida õpilane juba eelnevalt teab paremini, jättes seega vähem aega küsimustele, mida antud õpilane ei tea.

Antud protsessi osa lihtsustavad õpiabi programmid, millest 4 populaarsemat tasuta saadaval olevat kirjeldatakse järgmistes peatükkides. Need programmid aitavad õpilasel omandada materjali võimalikult efektiivselt, samas erinedes üksteisest ülesehituse poolest. Mis pakub õpilasele võimaluse valida vastavalt vajadusele erinevate programmide vahel.

2.4 Teadmiste omistamise kontrollimine

Peale teadmiste omistamist on oluline ka kontrollida, kui hästi on õpiabi programme kasutades minid teemad omistatud. Kuna õpiabi programmide eesmärk, nagu nimigi ütleb, on teadmiste omistamise lihtsustamine, siis antud programmides ei ole võimalust teostada kogu teadmiste kontrolli ja hinnata materjali omandatust. See on ka arusaadav, sest antud programmid on kitsalt suunitletud õppimise osale, mida nad täidavad väga edukalt.

Võib juhtuda, et kui õpilasel läheb hästi on õpetaja postitanud näiteks moodle-sse testi, mida on kordamise eesmärgil võimalik korduvalt teostada. Kuna see on pigem hrv nähtus kui reegel, siis on eluliselt vajalik võimalus koostada teadmiste kontrolli testi õpilase enda poolt.

Interneti avarustest otsides ja programmidega lähemalt tutvudes, võib kiiresti avastada, et teadmiste kontrolli koostamise programmid on suunitletud pigem õpetajatele või õppejõududele. Õpilasele endale teadmiste kontrolli koostamise võimalust ei pakuta. Seega on antud bakalaureuse töö eesmärk analüüsida olemasolevaid õpiabi programme ja koostada prototüüp teadmiste omandamise kontrollimise programmist.

3 Olemasolevate õpiabi programmide kirjeldus

Antud peatükis kirjeldan nelja populaarsema õpilastele suunatud õpiabi programmi funktsionaalsust. Valimisse sattumise eelduseks oli eksami küsimuse ja vastuse esmase sisestamise võimalus programmi. Populaarsuse määramise kriteeriumiteks oli asukoht Google otsingu tulemustes ja kasutajate tagasiside, mille leidsin foorumitest ning kättesaadavus. Vaatlesin ainult programme, mis olid tudengitele kättesaadavad tasuta.

3.1 Anki

Anki on ainuke programm, mis on kättesaadav nii töölaua variandina kui ka APP'ina. Anki on programm, mis kasutab eksamiteks õppimise lihtsustamiseks pöördkaartide abi [1]. Pöördkaart on kaart kus ühel pool on kirjutatud küsimus ja teisele poole selle vastus. Anki's on kasutuses „*Active Recall*“ [2] printsiip, mis väidab, et efektiivseks õppimiseks peab aju pidevalt stimuleerima. Programm kasutab *Spaced repetition* [3] tehnikat, mis kordab küsimusi järjest suureneva ajaintervalli järel, mida kutsutakse *SuperMemo SM-2* algoritmiks. Ehk kui kasutajalt küsitakse küsimus ja kasutaja hindas oma vastuse korrektseks siis suureneb ajaintervall eksponentsiaalselt näiteks 15 minutilt 2 tunnini ja sealt edasi juba näiteks 2 päeva peale.

Anki's kasutatavad pöördkaardid sarnanevad väga tavalistele paberist analoogidele, kus ühel pool on küsimus ja teisel pool on vastus. Programmis näeb see välja näiteks selline: Kaardil olevale küsimusele „Kas koer on imetaja?“, mõtled vastusele, näiteks „Jah“, klikid näita vastust nupule ja ekraanil kuvatakse vastus : „Jah“. Seejärel on võimalus Anki sisestada hinnang oma vastusele. Vastavalt antud tagasisidele määrab programm, millise intervalliga antud küsimust uuesti näidata.

Pöördkaardid on võimalik jagada erinevatesse pakkidesse [4]. Kaardipakkidele on võimalik määrata erinevaid seadeid, näiteks ajaintervalli, mille tagant pöördkaarte uuesti näidatakse. Pakkide loomisel soovitatakse kasutada eksamist võimalikult laia küsimuste valikut. Programmi seadistustes on hoiatus, et kui omada liiga palju kaardi pakke, siis aeglustab see programmi tööd märkimisväärselt. Seega selle asemel, et teha palju

väikseid teemakooslusi, oleks märgatavalt mõttekam kasutada ühes pakis võimalikult laiad teemade valikut, et programmi jõudlus oleks parim.

Küsimuste vastuse hindamisel võimaldab Anki valida kolme võimaliku variandi vahel:

- „Uuesti“, annab küsimuse ajaintervallile vaikimisi väärtuse.
- „Hästi“ annab küsimusele järgmise, pikema ajaintervalli.
- „Lihtne“, annab kaardile tüübi „üle vaadatav“, muutes kaardi esinemise sageduse märgatavalt harvemaks, kui eelnevad kahel staatusel. Tänu eksponentsiaalsele algoritmile muutub kaardi esinemise sagedus väga kiiresti peaaegu olematuks.

Vaikimis on kaardid enne õppimise alustamist jagatud kolme kategooriasse:

- „Uus“, sisaldab kaarte, mida pole veel õpitud.
- „Õppimises“, sisaldab kaarte, mida on nähtud hiljuti esimest korda ja mida hetkel õpitakse.
- „Üle vaadata“, on kaardid, mis olid eelnevalt õpitud ja nüüd vajavad üle kordamist, et ära ei ununeks.

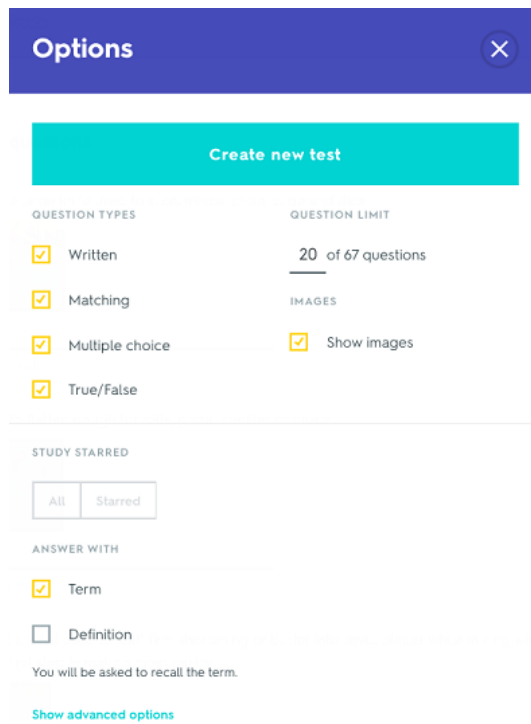
Õppimise tulemuslikkuse kohta on võimalik näha erinevaid graafe.

- „Prognoos“ graafil on võimalik näha umbkaudset kordamiste arvu määratud päeval.
- „Ülevaadete arv“ graaf näitab, mitu korda on erinevaid kaarte korratud.
- „Läbivaatamise aeg“ graaf näitab aega, mis on kaartide õppimise peale kulunud.
- „Intervallid“ graaf näitab kui palju küsimusi on kindla intervalliga.
- „Vastuse nupud“ graaf näitab, mitu korda on erinevaid vastuste nuppe vajutatud kindlal ajavahemikul.

3.2 Quizlet

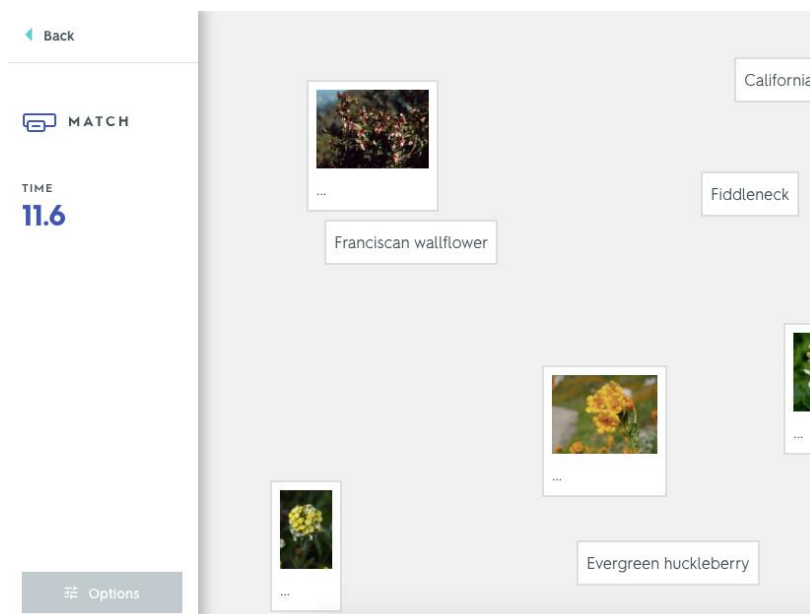
Quizlet [5] on *online* programm, mis kasutab õppimise hõlbustamiseks pöördkaartide abi. Programm võimaldab koostada õppimiseks küsimuste komplekte. Antud komplekte saab õppida kasutades erinevaid õppimise viise ja mängu.

- „Õpi“, antud õppeviisil alustatakse kergemat tüüpi küsimustega nagu pöördkaardid ja valik vastustega küsimused. Küsimusi õigesti vastates muutuvad küsimuste tüübid raskemaks, näiteks sagedamini hakkab ilmuma küsimusi, kus on vaja vastata kirjalikult. Kogu protsess toimub seeriates. Seeria lõppeb, kui igale küsimusele on vastatud kaks korda õigesti.
- „Mälukaart“, õppimine toimub kasutades pöördkaarte. Lisa võimalusena saab kasutada hääle süntesaatorit, mis võimaldab küsimusi ette lugeda. Õppimise seeria lõppeb siis kui on üle vaadatud kõik küsimus-vastus pöördkaardid.
- „Kirjuta“, õppimine toimub küsimustele kirjalikult vastates. Programm hindab küsimuste vastamist ja annab teada, millist materjali peaks veel õppima. Vastuste hindamine toimub automaatselt, aga on võimalik valepositiivsed vastused õigeks kinnitada. Ehk kui vastus on õige, aga tehti näiteks kirjaviga siis on võimalik märkida vastus õigeks, kuigi algselt programm luges selle valeks.
- „Test“, võimaldab luua eksami testi nagu see toimuks päris eksamil ja näha tulemusi, mis võidakse saavutada, kui eksamit peaks sooritama kohe. Testi koostamisel on võimalik valida paljude erinevate seadete vahel, et optimeerida õppimise käiku.



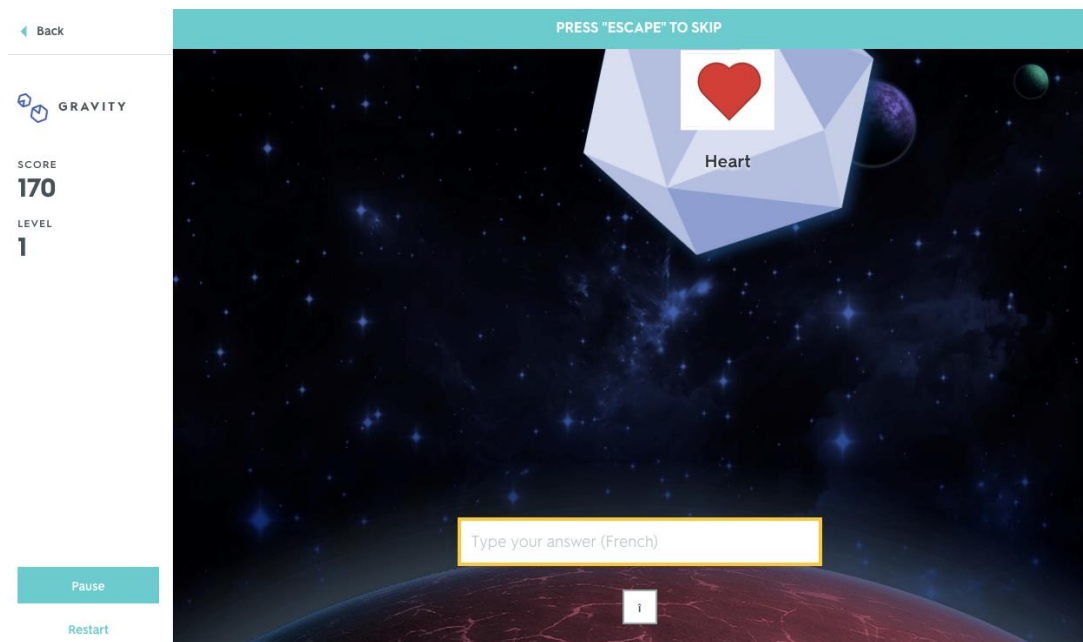
Joonis 2. „Test“ tüüpi õppimise seaded.

- „Sobitamine“ on mänguline õppimine viis ajalise piiranguga. Mängu eesmärk on vastata nii kiiresti, kui võimalik erinevaid termineid ja definitsioone lohistades definitsioone terminite või piltide peale.



Joonis 3. „Sobitamine“ tüüpi õppimise viis.

- „Gravitatsioon“ on mänguline õppimise võimalus, mille eesmärk on vastata küsimusele kirjalikult kiiremini kui asteroid puutub kokku planeedi pinnaga.



Joonis 4. „Gravitatsioon“ tüüpi õppimise viis.

3.3 Tinycards

Tinycards on *online* programm, mis kasutab *Spaced repetition* tehnikat ja pöördkaarte, et õpetada materjale mänguliselt [6]. Eelnevalt on Duolingo, kes on antud programmi loonud firma, tegelenud keeleõppe programmide arendamisega. Antud programmis kasutatakse Duolingo keeleõppe algoritmi, et kohanduda iga õppija arenguga ja vastavalt õppimise progressiga esitada küsimusi.

Tinycards programmis on sarnaselt Anki'le võimalik lisada küsimuste kaarte ja neid pakkidesse või teemadesse jagada. Kaartidele on võimalik lisada küsimus teksti või pildi kujul ning selle vastus.

Testi alustades näitab programm esmalt neid kaarte, mida ei ole kasutajale veel kunagi näidatud. Seejärel hakkab programm varieerima erinevaid küsimuse esitamise viise. Näiteks on tudeng eelnevalt sisestanud küsimuse kõige tavalisema pöörd kaardi kujul, kus ühel pool on pilt ja teisel selle vastus. Õppimise edenedes varieerib programm vastamise viisi, kus lihtsa vastuse vaatamise asemel nõutakse vastuse sisestamist kirjalikult.



Joonis 5. Tinycards näide küsimuse küsimisest.

Programmi üleval osal on edenemisriba, mis näitab kasutaja progressi kaardipaki vastamisel. Iga õigesti vastatud küsimusi viib järjest lähemale lõpule, iga valesti vastatud küsimus aga viib progressimeetri sammast natukene tagasi.

3.4 Goconqr

Goconqr on multifunktsionaalne programm pakudes palju erinevaid õppimise viise aidates optimeerida õppimist.

Pakutakse võimalust koostada pöördkaarte ja õppida *Spaced repetition* tehnikat kasutades. Samamoodi nagu eelpool kirjeldatud programmidki võimaldab Goconqr luua kaarte ja kaardipakke, mille põhjal on kasutajal võimalik optimeerida õppimise protsessi. Lisaks annab Goconqr kasutajale võimaluse peita kaarte, mida juba teatakse, eesmärgiga õppida süvendatult teemasid, mis ei ole veel selged.

Samuti võimaldab Goconqr koostada teste. Testidesse on võimalik lisada erinevat tüüpi küsimusi. Küsimuste juurde on võimalik lisada selgitus küsimuse kohta, mida õppija näeb testi tulemusi vaadates, andes õpilasele lisavõimaluse korrata küsimuse vastuse põhitõdesid. Testid koostatakse käsitsi. Peale testi sooritamist kuvatakse kasutajale testi sooritamiseks kulunud aeg, palju oli valesid vastuseid ning mitu punkti saadi. Iga küsimuse vastus on väärt 1 punkt. Seega kui vastati 3 küsimust 5-st valesti saab kasutaja 2 punkti.

Kasutajal on võimalik üle vaadata oma küsimused ja vastused ning näha üldist statistikat, näiteks paljudele küsimustele on eelnevalt õigesti vastatud.

4 Olemasolevate õpiabi programmide plussid ja miinused

Antud peatükis tuuakse välja õpiabi programmide plussid ja miinused. Nende välja selgitamiseks kasutasin isiklikku kogemust ja foorumitest loetud kasutajate tagasisidet.

4.1 Anki plussid

Pikaajaliselt õppides on Anki väga efektiivne, kuna kasutab laialt levinud *Spaced repetition* algoritmi. Anki-t kasutades teeb programm enamuse tööst ise ära määrates, millal kasutaja peaks õppima ja milliseid küsimusi kordama.

Lisaks on Anki's suur valik erinevaid muudetavaid seadeid õpiprotsessi personaliseerimiseks võimaldades kasutajal muuta õppimise võimalikult efektiivseks. Näiteks on võimalik muuta, mitu kaarti päevas õpitakse või kui tihti konkreetset materjali korratakse.

Positiivselt tasub ära märkida ka Anki kasutaja kordamise testide loomise algoritmi, mis fokuseerib algselt neile kaartidele, mida kasutaja ei ole üldse korranud. Õppimise edenedes ja lähtuvalt vastuste õigsusest suurendab algoritm küsimise esinemise intervalli. See muudab õppimise märgatavamalt efektiivsemaks, sest kasutajale kuvatakse tihedamini neid küsimusi, millega ei lähe hästi samas periooditi korrates küsimusi, mida on vahepeal korrektselt vastatud, muutes niimoodi küsimustiku vähem etteaimatavaks ja aidates samas korrata juba omandatud materjali.

Anki üks suuremaid plusse on programmi põhjalikkus ja kasutajasõbralikkus. Antud programm võimaldab luua kergelt küsimusi ja vastuseid, andes kasutajale võimaluse jagada oma materjali või siis kasutada teiste küsimusi-vastuseid.

Anki võimaldab kasutajal jälgida oma õppimise progressi graafiliselt. Arengut jälgides on kasutajal võimalik panna rohkem rõhku mingile kindlale küsimustiku osale ja planeerida õpinguid. Kuna Anki kasutab intervall õppimise algoritmi siis teadmine, millal peaks materjale uuesti kordama annab kasutajale hea ajaplaneerimise võimaluse. Üks huvitavamaid graafe, mida saab Anki's kasutada on „Läbivaatamise aeg“, mis näitab kaua kulus kasutajal aega küsimustele vastamiseks, võimaldades muuta küsimustiku seadeid nii, et aeg mille jooksul vastata saab oleks võimalikult lähedane graafil olevale. Lubades

kasutajal proovida kuidas ta saaks hakkama stressi olukorras. Ehk siis kuidas eksami situatsioonis stress kasutaja tulemust mõjutab.

4.2 Anki miinused

Anki üks peamine miinus on küsimuse-vastuse pikkuse piirang. Ehk kui küsimuse vastus on rohkem kui 3 lauset pikk peaks selle küsimuse jagama väiksemateks osadeks, kuigi lühikesi kaarte õpitakse märgatavamalt kiiremini ja kergemini kui suuremahulisi küsimusi siis ei ole seda paljudel juhtudel võimalik kasutada, sest tõuseb oht kahjustada küsimuse terviklikkust. Kui küsimus on võimalik ümber teha väiksemateks osadeks, siis probleemi ei teki. Anki kasutamise loogika on ülesse ehitatud lühi vastuseid võimaldavetele küsimustele. Pikemaid küsimusi on võimalik Anki abil õppida, aga see nõuab märgatavamalt suuremat töö mahtu, sest küsimuse tuleb ümber sõnastada mitmeks lühikeseks ilma, et kahjustuks teema terviklikkus. Üks kasutaja on selle probleemi hästi kokku võtnud alljärgnevalt: [7] „I'm not sure if this is a good analogy, but I like to think of the process of learning as akin to building a house out of reinforced concrete: You start by laying down a framework of steel. These are the connections and associations that nurture conceptual knowledge. Then you proceed to fill that framework with the liquid concrete it needs to actually be useful, the facts. It takes some time for these two components to bond and set in, but once they do, you'll have something that will be as strong as anything can be”.

Ankil on küll olemas erinevad diagrammid andmaks kasutajale tagasisidet testide sooritamise kohta üldiselt, kuid puudub täielikult või on puudulik tagasiside konkreetsete küsimuste arengu jälgimiseks.

Samuti võib miinusena välja tuua, et õigesti vastatud küsimuste esinemissagedus on väga harv.

Sobib kasutamiseks juhul kui kasutajal on eksamiks õppimiseks väga palju aega.

4.3 Quizlet plussid

Quizleti plussideks on laialt levinud *Spaced repetition* algoritmi kasutamine. Nagu juba eelnevalt mainitud võimaldab selline õppimise viis lühikeste küsimuste korral omandada materjali efektiivselt.

Quizleti üheks suurimaks eelisteks teiste analoogsete programmide ees on võimalus kasutada palju erinevaid õppimisvorme, mis aitab kasutajal vältida rutiini tekkimist ning hoiab õppimise protsessi pidevalt uuena. Ehk iga päev ei pea sama tüüpi küsimustikku kasutama, vaid on võimalik neid varieerida. Üks paremaid õppimise viise Quizleti's on „Õppe režiim“, kus algselt küsitakse küsimusi kergemates formaatides ja mida aeg edasi ja rohkem küsimusi õigesti vastatakse seda raskemaks vastamise formaadid muutuvad.

4.4 Quizlet miinused

Üks suuremaid miinuseid Quizlet'i puhul on diagrammide puudumine. Kasutajal ei ole võimalik kusagilt jälgida oma üldist progressi. Tagasiside puudumine raskendab arusaamist, milline materjal on omandatud ja milline mitte. Quizlet küll järjestab oma algoritmiga küsimusi igas testis eraldi aga arengut ei ole graafiliselt võimalik jälgida, mis jätab kasutaja teadmatusesse, kuidas on testid siamaani sooritatud.

Quizletis ei ole võimalik sisestada pikki küsimusi või küsimustele pikalt vastata. Võimalik on küll sisestada pöörd kaartile pika vastusega küsimus, aga siis kaotab antud programmi mõtte, sest pöörd kaartide puhul on just tähtis hoida info võimalikult lühike.

Samuti ei ole Quizletis võimalik kirjalikult pikkade vastustega küsimusi õppida viisil, et kasutaja saaks harjutada nende läbi kirjutamist. Paljud inimesed omandavad materjali paremini just pidevalt läbi kirjutades, kahjuks seda varianti ei pakuta.

4.5 Tinycards plussid

Tinycard's pakub *Spaced repetition* tüüpi õppimise algoritmi. Nagu juba eelpool sai mainitud, siis on seda tüüpi õppimise algoritm väga laialt levinud õppimise viis. Selle abil õppides on võimalik omandada eksami materjal põhjalikult.

Tinycardsiga õppimine pakub pidevad vaheldust varieerides küsimuse vormi. Kord küsitakse näiteks pöörkaardiga küsimusele vastust, kuid õppimise edenedes pead vastuse hoopis kirjutama. Selline lähenemine võimaldab kasutajal materjali paremini kinnistada.

4.6 Tinycards miinused

Puudub igasugune graafiline tagasiside. Kasutajal on raske aru saada, millised küsimused vajavad rohkem tähelepanu kas, üldse ei vastata õigesti või olid kõik vastused korrektsed. Kasutajal peaks kindlasti olema võimalus saada tagasisidet, milliste küsimustega peaks rohkem tegelema.

Programm ei toeta pikkade vastustega küsimusi. Kuna tegemist on peamiselt pöörd kaartide vastamisega siis eeldatakse et, kasutaja koostab lühikeste vastustega kaarte. Samuti puudub võimalus küsimustikke eksamilaadselt kirjalikult läbi harjutada.

4.7 Goconqri plussid

Kasutatakse *Spaced repetition* tüüpi õppimis algoritmi, mis nagu eelnevalt mainitud on väga efektiivne materjalide põhjalikuks omandamiseks.

Testide koostamisel on võimalik lisada küsimuse alla selgitus või viide materjalile. See annab kasutajale võimaluse saada eriti raskete küsimuste puhul lisateavet materjali paremaks kinnistumiseks.

Peale testi sooritamist on kasutajal võimalik küsimuste vastused üle vaadata ühtse teksti aruandena. Kus kasutajale kuvatakse küsimus ja tema poolt sisestatud vastus. Andes kasutajale lihtsa ja kiire tagasiside teostatud testi kohta. Võimaldades kasutajal välja selekteerida küsimused, millega peaks veel tegelema või mille vastus oli vale.

4.8 Goconqri miinused

Programmis puudub graafiline võimalus näha erinevaid statistikaid vastatud küsimuste kohta. Puudub aruanne või graaf, mis näitaks, kuidas mingi konkreetse küsimusega kasutajal hetkel läinud on. Seega on kasutajal raske aru saada tema õppimise progressist, mida ta on siiaani saavutatud, milliste küsimustega peab rohkem tegelema.

Pikkade küsimuste või pikkade vastuste puhul on raske antud programmi kasutada, kuna programm põhineb *Spaced repetition* algoritmil, mis eeldab, et vastused on lühikesed. Samuti ei paku programm võimalust pikkade vastustega küsimusi kirjalikult läbi lahendada.

5 Õpiabi programmide analüüsi kokkuvõte

Kokkuvõtteks võime öelda, et *Spaced repetition* algoritmi kasutamine pöördkaartidega õpiabi programmides on laialt levinud ja võimaldab õpingumaterjali põhjalikult läbi töötada. Viidete lisamine küsimustele annab kasutajale lisavõimaluse vastuse sisu paremini mõista. Erinevate õppetüüpide varieerimine aga pakub kasutajale vaheldust ning väldib rutiini langemist sundides kasutajat aktiivsemalt õpiprotsessis osalema.

Miinustena võib välja tuua eelkõige kasutajale tagasiside andmise puudulikkuse. Enamustel programmidel puuduvad diagrammid või visuaalsed abid, kirjeldamaks küsimuste vastamise progressi. Pidev tagasiside saamine õpiprotsessi käigust on positiivse lõpptulemuse jaoks väga oluline. Olemasolevates programmides pakutakse ainult üldist tagasisidet, palju küsimusi õigesti vastati kuid puudub täielikult konkreetsete küsimuste tagasiside.

Samuti ei paku mõned programmid võimalust pikkade vastustega küsimusi kirjalikult läbi lahendada. Kuid paljude kasutajate arvates aitab just küsimuste kirjalik läbivastamine paremini omandada eksami materjali, võimaldades paremini kontrollida vastuse õigsust kõikides aspektides.

Väga erinev oli ka võimalus seadistustega mõjutada küsimuste esinemissagedust. Näiteks Anki algoritm suurendas mõnede paremini vastatud küsimuste esinemisintervalli peaaegu nädala peale. Mis on pingeliseksamigraafiku korral, kus eksamid on iga 3 või 4 päeva tagant, vastuvõetamatu.

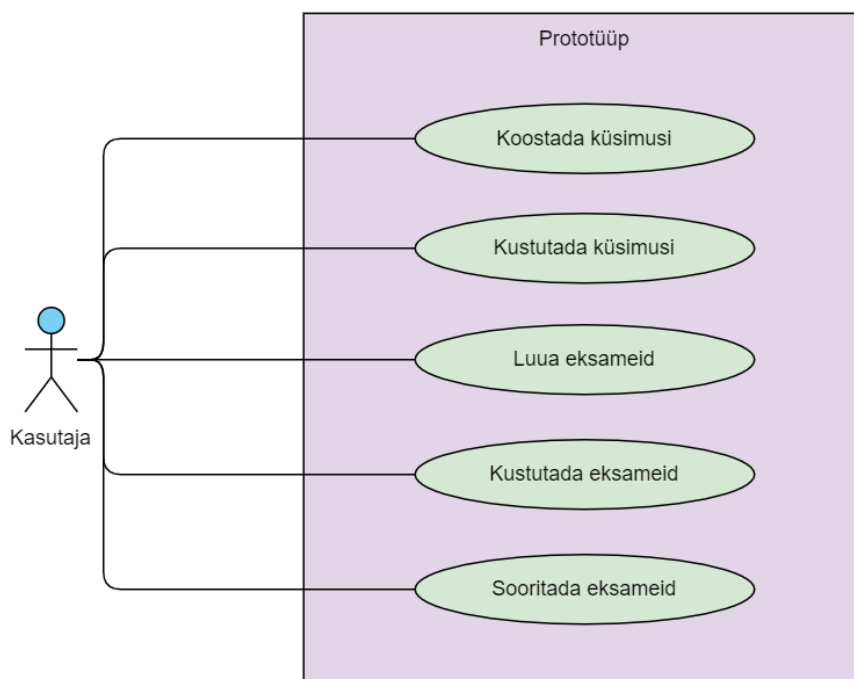
Ja lõpuks puudub programmidel võimalus küsimustikke, dünaamiliste komplektidena, eksamilaadselt kirjalikult läbi harjutada.

6 Teadmiste omistamise kontrolli prototüüp

Eelnevalt tehtud analüüsi põhjal pakun välja eksamiks õppimise programmi uut tüüpi laiendatud lahenduse, mis kasutab *Spaced repetition* algoritmi edasiarendust. Kus küsimuste järjestamiseks kasutatakse eelnevate vastuste statistikat, prioritiseerides kõrgemalt neid küsimusi, mille vastused on olnud kesised ning valides nende vahele küsimusi millede õigete vastuste skoor on olnud maksimumi lähedane. Antud algoritmi modifikatsioon kindlustab, et tagasihoidlikuma skooriga küsimusi küsitakse tihedamalt kuni positiivse maksimum skoori saavutamiseni.

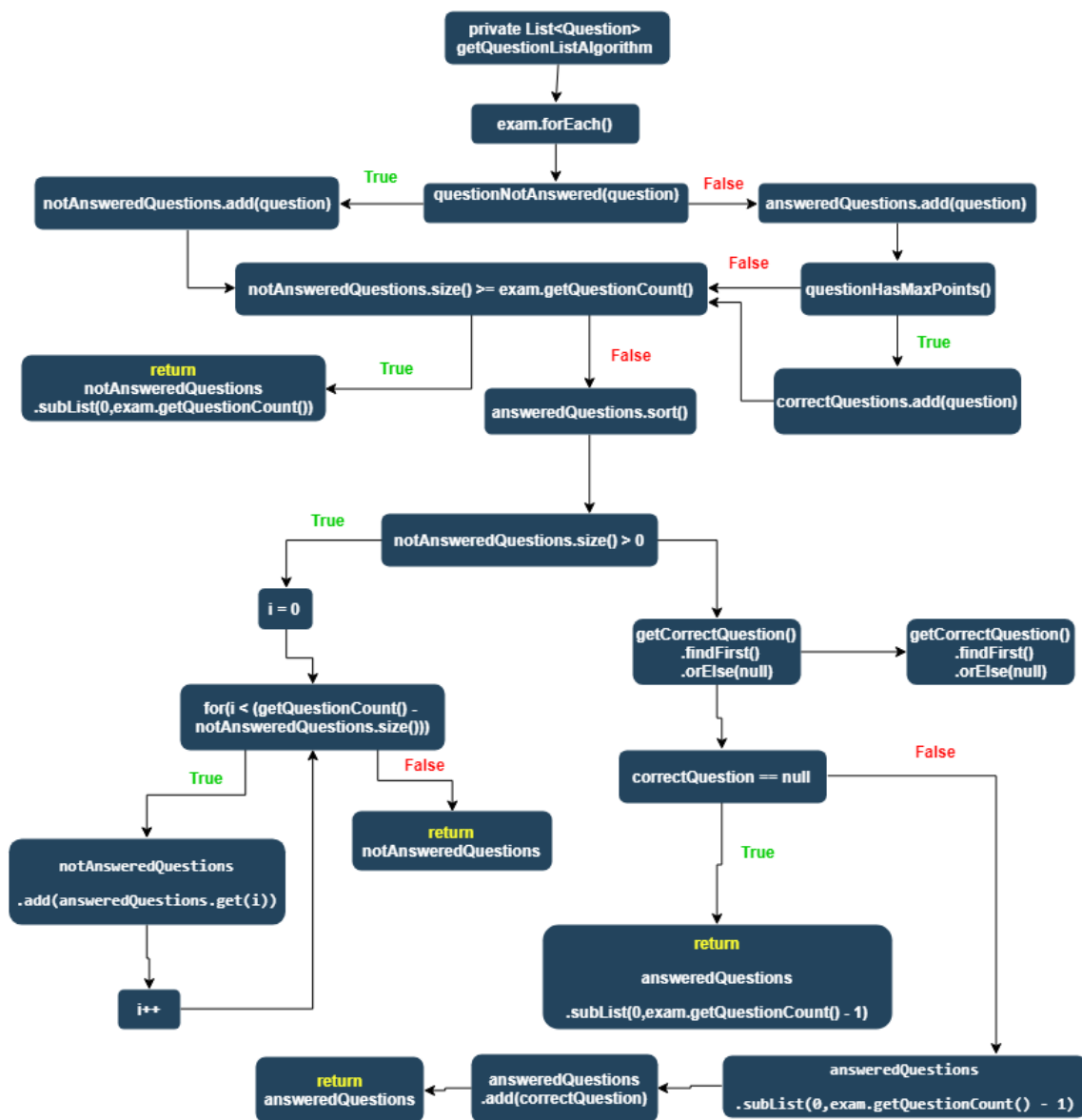
Dünaamilised küsimustikud koosnevad 4 küsimusest, kus esimesed kolm valitakse kõige tagasihoidlikuma skoori saanud küsimuste hulgast juhuslikkuse alusel ning neljas valitakse peaaegu maksimum skoori saavutanute hulgast. Selline küsimustiku ülesehitus võimaldab kasutada õppimise aega kõige efektiivsemalt, keskendudes rohkem nendele küsimustele, mida kasutaja teab halvemini unustamata samal ajal paremini vastatud küsimuste kontrolli funktsiooni.

Prototüübi kasutaja üldised tegevused on välja toodud joonisel 6.



Joonis 6. Kasutaja tegevused prototüübis.

Kuidas programm täpsemalt küsimustikku protsessib näete jooniselt 7.



Joonis 7. Kasutatud oleva algoritmi joonis.

Arenduses on välja pakutud lahendus, kuidas on võimalik korrata ka pikkade vastustega küsimusi, võimaldades samal ajal kuvada jätkuvalt tagasisidet.

Samuti ei ole ära unustatud ka kasutajatele tagasiside kuvamist. Kuna programm salvestab küsimuste vastuste statistika siis võimaldab see anda kasutajale detailset tagasisidet tema õppimise progressi kohta mitte ainult üldises plaanis vaid kuni üksiku küsimuse detailsuseni välja.

Võtsin eelpool analüüsitud programmide positiivsed omadused parendasin neid, lüasin detailse aruandluse ja tulemuseks sain lahenduse, mis peaks rahuldama ka kõige nõudlikuma õppija vajadusi.

Prototüübi tegemiseks kasutan Gradle, Java-t ning JavaFx teeki. Prototüübi *frontendiks* on kasutatud FXML ja *backendi* jaoks Javat. Projekt on kaetud osaliselt JUnit testidega.

6.1 Prototüübi küsimuste lisamine

Küsimusi on võimalik lisada olles vaates „Add Questions“ (Joonis 8).

The screenshot displays the 'Add Questions' interface. At the top, there are five navigation buttons: 'Main Menu', 'Add Questions', 'Create Exam', 'Take Exam', and 'Graphs'. Below these are two text input fields: 'Question' and 'Answer'. A green 'Add question' button is centered below the inputs. The main content area shows a list of added questions with a 'Delete' button.

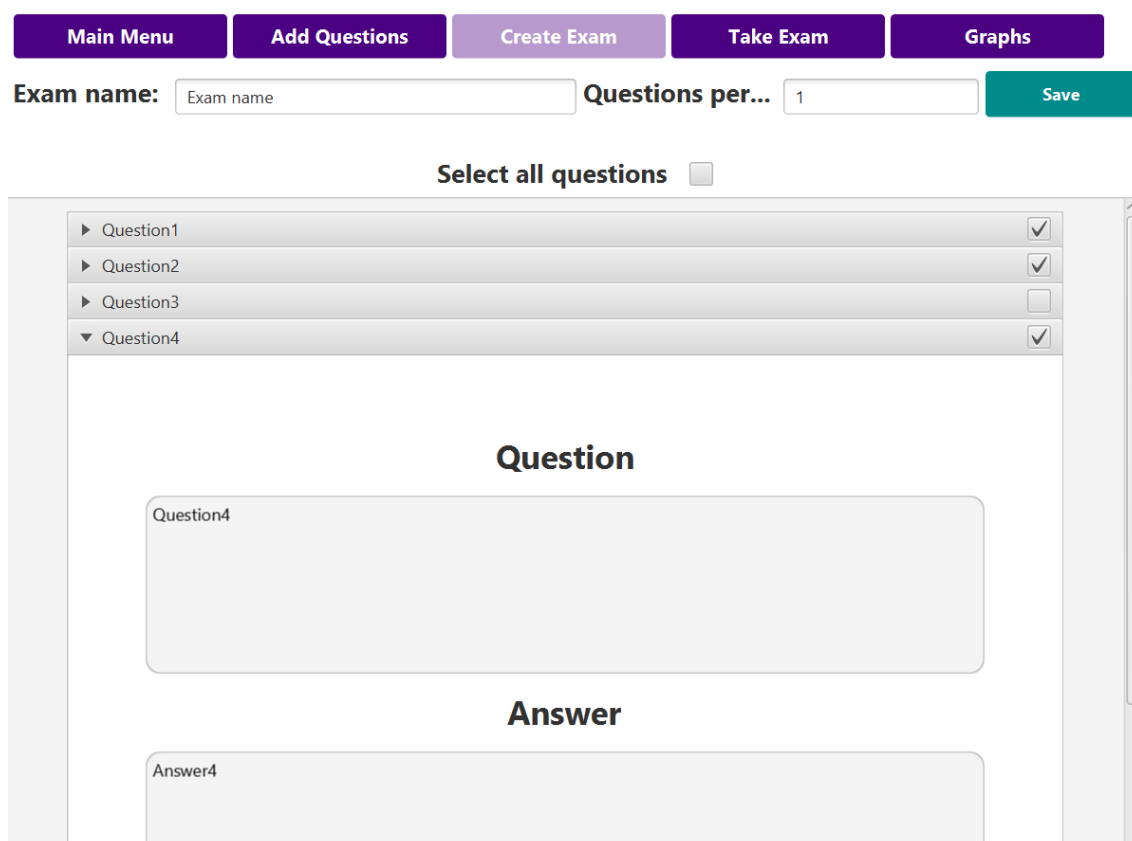
Joonis 8. Küsimuste lisamise vaade.

Küsimuste lisamise vaates kuvatakse kasutajale rakenduse all osas olemasolevad küsimused, mida on võimalik vajadusel kustutada. Küsimuse lisamiseks tuleb kirjutada küsimus märksõna „Question“ all asuvasse teksti lahtrisse. Vastuse lisamiseks tuleb kirjutada vastust märksõna „Answer“ all asuvasse teksti lahtrisse. Peale küsimuse ja vastuse sisestamist vajuta nuppu „Add question“, mille järel kuvatakse sisestatud küsimus-vastus komplekt rakenduse all olevasse loendisse.

Küsimuse kustutamiseks tuleb vajutada nuppu „Delete“, millele vajutades küsimus kustutatakse nimistust. Eksamitelt, millel antud küsimus esineb, küsimust ei kustutata.

6.2 Prototüübi eksami koostamine

Eksamit on võimalik koostada olles vaates „Create Exam“ (Joonis 9).



Joonis 9. Eksami lisamise vaade.

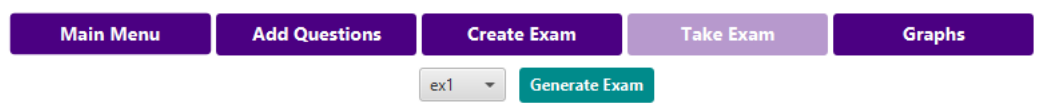
Eksami lisamise vaates kuvatakse rakenduse allosas olemasolevad küsimused, küsimuste päises paremal pool asuvad märkeruudud. Küsimuse peale klikkides avaneb antud küsimuse kogu informatsioon ehk nii küsimus kui ka vastus.

Kõikide küsimuste valimiseks tuleb vajutada märkeruutu „Select all questions“, mille peale kõik märkeruudud saavad märgitud. Uuesti sama märkeruutu vajutades, eemaldatakse märkeruutude valikud.

Eksami lisamiseks tuleb täita väljad „Exam name“ ja „Question count“ ja valida eksamisse kuuluvad küsimused märkides selleks küsimuse päise paremal pool asuvaid märkeruute. Kui eelnev on täidetud tuleb vajutada „Save“ nuppu, mille järel antud eksam salvestatakse.

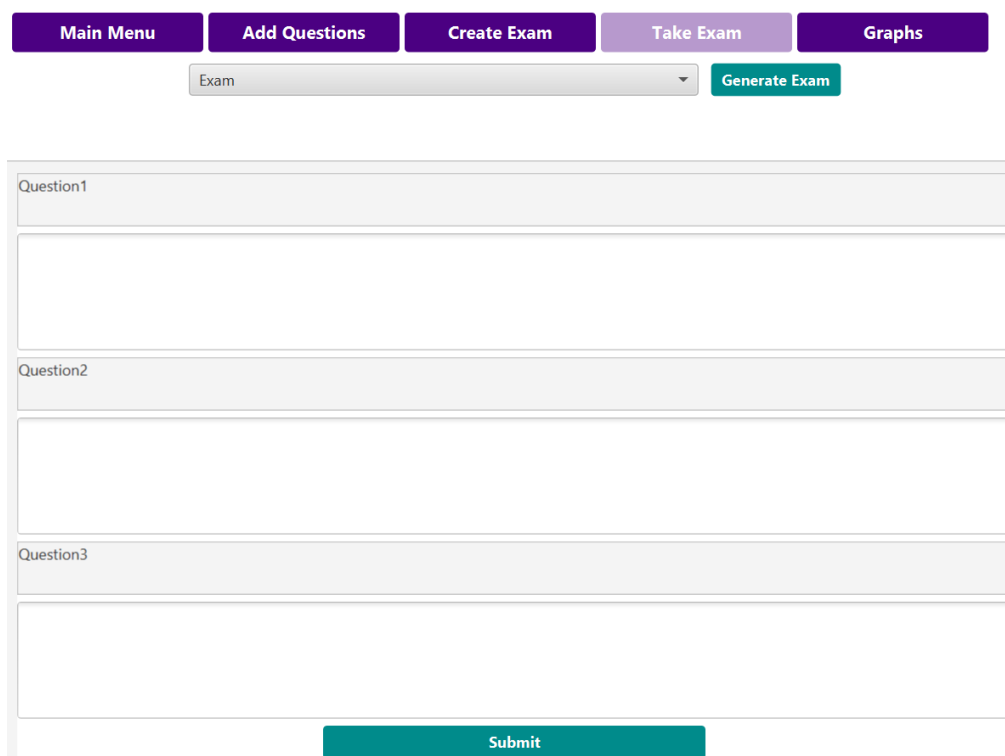
6.3 Prototüübi eksami sooritamine

Eksami on võimalik sooritada „Take Exam“ vaates (Joonis 10).



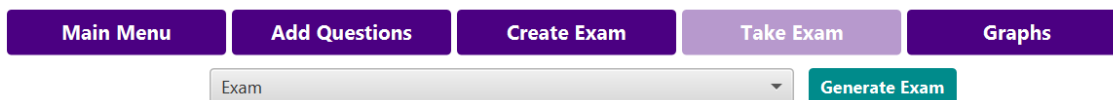
Joonis 10. Eksami sooritamise vaade.

Eksami soorituse loomiseks tuleb rippmenüüst valida eksam, mida soovitakse sooritada ja vajuta nuppu „Generate Exam“. Peale nupule vajutamist, kuvatakse kasutajale eksamiküsimused (Joonis 11).



Joonis 11. Loodud eksam.

Peale küsimustele vastamist tuleb vajutada nuppu „Submit“, mille järel vaade muutub küsimuste kontrollimise vaateks (Joonis 12).



Joonis 12. Küsimuste kontrollimise vaade.

Küsimuste kontrollimiseks kuvatakse vasakul pool tulbas kasutaja poolt sisestatud küsimuse vastus ja paremal pool õige vastus, mis sisestati küsimuse koostamisel. Mille järel peab kasutaja andma oma vastusele hinnangu.

Hinnangu andmiseks on kasutajal võimalik valida paremal pool iga vastuse taga asuvast neljast erinevast hinnangust üks.

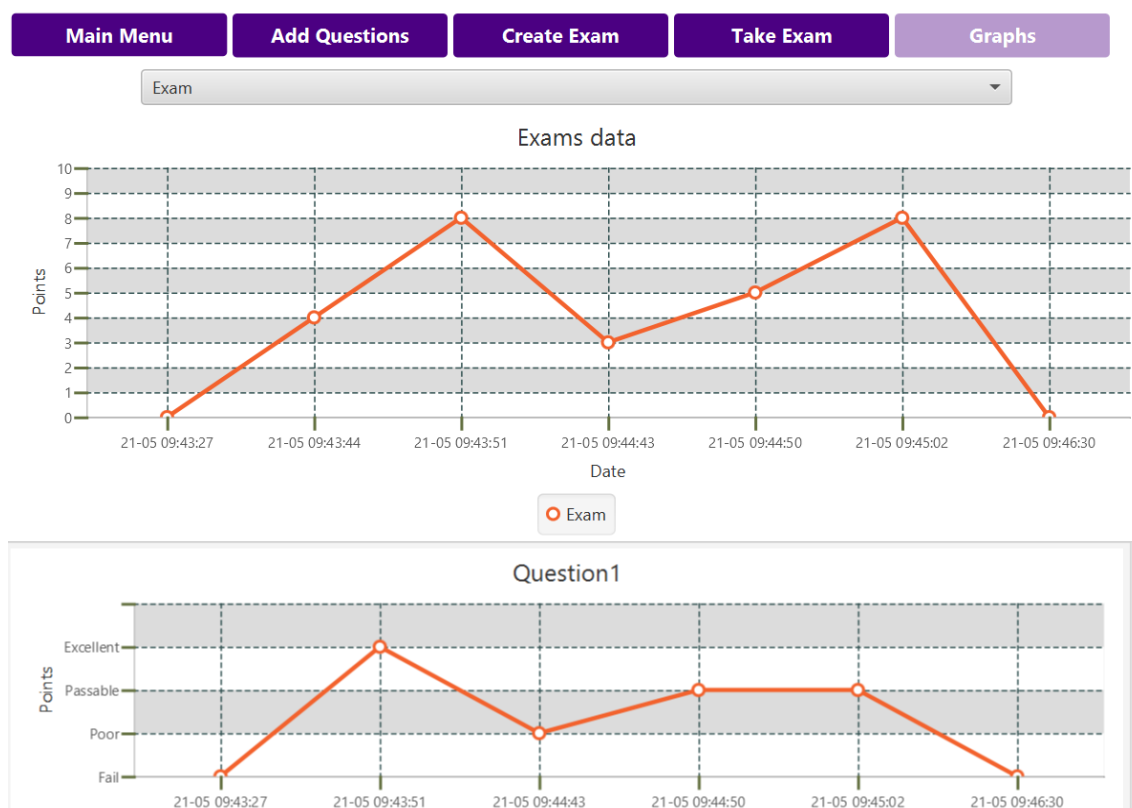
Hinnangute tähendused on järgnevad:

- „Excellent“ - kasutaja poolt sisestatud vastus oli täiesti korrektne.
- „Passable“ - kasutaja poolt sisestatud vastus on vähemalt 75% ulatuses õige.
- “Poor“ - kasutaja poolt sisestatud vastus oli 50% ulatuses õige.
- “Fail“ - küsimuse vastus oli vähem kui 50% ulatuses õige.

Igal vastusel saab olla ainult üks hinnang. Ja kui kõik küsimused on üle kontrollitud, siis tuleb vajutada „Check“ nuppu, mille järel vastuste tulemused salvestatakse .

6.4 Prototüübi tagasiside kuvamine

Tagasisidet on võimalik kasutajal näha olles vaates „Graphs“ (Joonis 13).



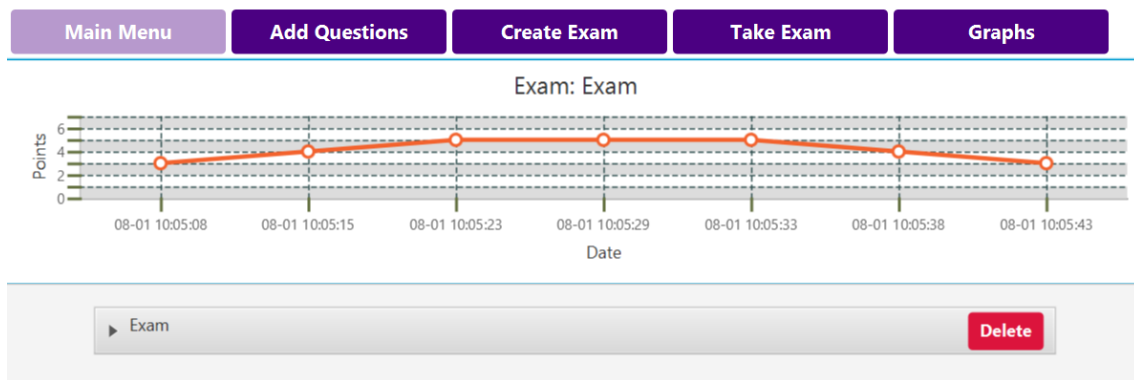
Joonis 13. Tagasiside kuvamise vaade.

Antud vaates on võimalik näha kõikide sooritatud eksamite tulemusi. Vaate üleval osas on võimalik rippmenüüst valida, millist eksami tagasisidet soovitakse näha.

Vaates on kaks joon diagrammi. Ülemine diagramm kirjeldab eksami üldist tulemust ja alumine diagramm kirjeldab tulemust üksikute küsimuste kaupa. Alumist diagrammi on võimalik alla kerida, et näha teiste küsimuste diagramme. Punkti skaala on sama, mis hinnangu andmisel ehk 0 punkti tähendab vastamise ebaõnnestumist ja 3 punkti tähendab, et vastatud on suurepäraselt.

6.5 Protüübi esileht

Tagasisidet on võimalik kasutajal näha olles vaates „Main Menu“ (Joonis 14).



Joonis 14. Prototüübi esilehe vaade.

Esilehe vaates kuvatakse kasutajale erinevate eksami soorituste üldised tulemused ekraani üleval osas. Alumises osas on võimalik kustutada eksamit vajutades nupule „Delete“. Peale nupule vajutamist kustub antud eksam ära ja graafik eemaldatakse ka üleval olevadest graafidest. Eksami peale vajutades, kuvatakse kasutajale kõik küsimused, mis antud eksamiga on seotud.

6.6 Prototüübi võrdlus olemasolevate õpiabi programmidega

Suurim erinevus prototüübi ja analüüsitud õpiabi programmide vahel on kasutajale detailse tagasiside pakkumine, mis annab kasutajale võimaluse jälgida küsimuse progressi õppimise protsessi käigus, andes kasutajale hea ülevaate teadmiste tasemest. Analüüsitud programmides puudus võimalus näha küsimuse põhise visuaalset tagasisidet, mistõttu oli kasutajal raske aru saada, kuidas õppimise protsess edeneb, kas areng toimub või mitte. Prototüüp pakkus antud puudusele omapoolse lahenduse detailse küsimuste põhise ülevaadete näol.

Lisaks ei olnud analüüsitud programmides võimalust vastata pikkade vastustega küsimusi kirjalikult, mille tõttu pakkus autor prototüübis välja oma nägemuse pikkade vastustega küsimustega tegelemiseks. Prototüübis on kasutatud eelneva analüüsi tulemusena saadud kogemust hindamise skaaladest. Skaala on jagatud 4 kategooriasse, eelneva kogemuse põhjal leidis autor, et mõttekas on kasutada lineaarset skaalat jaotustega iga 25% järel. Seega jagati hinnangud : 25, 50 , 75, ja 100%. Selline skaala jaotus on kasutajale hästi hoomatav ja lihtsustab märgatavalt hindamise protsessi. Näiteks, kas tekst vastab 25% ulatuses õige vastuse omale.

Analüüsitud programmides oli enamasti kasutuses *Spaced repetition* tehnika, autor kasutas selles leiduvad loogikat ja arendas seda edasi, et halvemini vastatud küsimusi

korrata tihedamini ja paremini vastatud küsimusi harvemini, aga hoides neid siiski kordamise eesmärgil perioodiliselt küsimuste voos.

Prototüübi algoritm lahendati ilma ajalise intervalli kasutamata, andes kasutajale võimaluse teha teste läbi korduvalt, järjest, et õppimise protsess oleks võimalikult kiire.

Analüüsitud programmide eelis prototüübi ees on nende märgatavalt suurem valik lisafunktsioone ja erinevaid küsimustele vastamise viise.

Spaced repetition ja prototüübi algoritmi on raske võrrelda, sest üks kasutab õppimiseks ajalise intervalli, teises aga on võimalik õppida pidevalt. Lisaks keskendub prototüüp võimalusele pikkade küsimuste-vastustega õppimisele, samas analüüsitud programmid on efektiivsemad lühikeste küsimuste korral. Kuna programmid on liiga erinevad siis on raske teostada testi, mis võrdleks programmide loogikat ja oleks kohaldatav mõlema programmi jaoks. Kuna on olemas väga palju erinevaid lähenemisi eksamitele ja küsimuste pikkusele siis antud prototüübi eesmärk ei ole asendada täielikult olemasolevaid programme vaid prototüüp on pigem eraldiseisev üksus ja näitab võimalust kuidas täita tühi koht õppimise protsessis.

7 Kokkuvõte

Käsitletava bakalaaurusetöö eesmärk oli analüüsida olemasolevaid õpilastele suunatud tasuta saadaolevaid õpiabi programme, mis abistaks õpilasi eksamiteks õppimisel, luues analüüsi tulemustele toetudes prototüübi, kus autor pakub välja lahenduse teadmiste omistamise kontrolliks. Eesmärk täideti, autor töötas välja prototüübi enda poolse visiooniga võimalikest lahendustest.

Olemasolevate õpiabi programmide analüüsimine andis autorile hea ülevaate nende programmide plussidest ja miinustest. Selle tulemusena valmis programmi prototüüp, kus autor pakkus välja lahenduse teadmiste omistamise kontrollile. Autor leidis kolm puudust, mis oli kajastatud kasutajate tagasisidest. Millest esimene oli puudulik detailne küsimuste põhine tagasiside õppimise protsessile ja teine puudutas pikkade küsimustevastustega eksamiküsimusi, mille tükeldamine väiksemateks küsimusteks ei oleks mõistlik või läheks selle tulemusena kaduma küsimuse algne mõte. Kolmandaks selgunud puuduseks oli „*Space repetition*“ tehnika kasutamise omapära küsimuste kuvamisel kuid, mis osutub suureks puuduseks juhul kui kasutajal on õppimiseks vähe aega. Probleemi lahendamiseks pakkus autor välja oma algoritmi, mis võimaldab sooritada teste pidevalt nii, et halvemini läinud küsimusi korratakse tihedamini ja paremini läinud küsimusi harvemini kuid siiski perioodiliselt, kordamise eesmärgil.

Kokkuvõtteks leiab autor, et olemasolevad õpiabi programmid on väga efektiivsed ja suure kasutusega. Samas leiab, et programme mis pakuvad teadmiste omandamise kontrollimist, ei suutnud leida. Prototüübina valminud programm rahuldab algelised õpilase soovid ja on sobilik kasutada ka juhtudel kui on lihtsal soov testida, millised oleks teadmiste kontrolli sooritades tulemused.

Bakalaaurusetöoga valminud prototüüpi on võimalik alla laadida <https://gitlab.cs.ttu.ee/Marti.Molder/iapb>

Kasutatud kirjandus

- [1] „About - AnkiWeb,“ 19 08 2018. [Võrgumaterjal]. Available: <https://apps.ankiweb.net>. [Kasutatud 08 11 2019].
- [2] „Active recall,“ 12 12 2015. [Võrgumaterjal]. Available: https://en.wikipedia.org/wiki/Active_recall. [Kasutatud 12 11 2019].
- [3] „Spaced repetition,“ 12 12 2015. [Võrgumaterjal]. Available: https://en.wikipedia.org/wiki/Spaced_repetition. [Kasutatud 5 10 2019].
- [4] „Decks,“ 19 08 2018. [Võrgumaterjal]. Available: <https://apps.ankiweb.net/docs/manual.html#decks22>. [Kasutatud 5 10 2019].
- [5] „What is Quizlet and how can I use it?,“ Quizlet Inc, 17 07 2017. [Võrgumaterjal]. Available: <https://help.quizlet.com/hc/en-us/categories/360001601132-Studying>. [Kasutatud 2 10 2019].
- [6] F. Lardinois, „Duolingo wants to reinvent flashcards with Tinycards,“ 2016. [Võrgumaterjal]. Available: https://techcrunch.com/2016/07/19/duolingo-wants-to-reinvent-flashcards-with-tinycards/?guccounter=1&guce_referrer_us=aHR0cHM6Ly93d3cuZ29vZ2xlLmVlLw&guce_referrer_cs=-ujpqVCJjkOLgcHne3ZpxQ. [Kasutatud 10 5 2019].
- [7] Glutanimate, „Does Anki really help me learn something?,“ 16 01 2018. [Võrgumaterjal]. Available: https://www.reddit.com/r/Anki/comments/7qr1qh/does_anki_really_help_me_learn_something/dsrc4z4/. [Kasutatud 1 9 2019].
- [8] „About - AnkiWeb,“ 19 08 2018. [Võrgumaterjal]. Available: <https://apps.ankiweb.net/docs/manual.html#introduction>. [Kasutatud 10 05 2019].