

Tallinna Ülikool
Digitehnoloogiaste instituut
Informaatika õppekava

Kõnetuvastuse rakendamine keeleõppe mängu näitel

Bakalaureusetöö

Autor: Carl Maask

Juhendaja: Martin Sillaots

Autor: „ „ 2017

Juhendaja: „ „ 2017

Instituudi direktor: „ „ 2017

Tallinn 2017

Autorideklaratsioon

Deklareerin, et käesolev bakalaureusetöö on minu töö tulemus ja seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

.....

(kuupäev)

.....

(autor)

**Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks
tegemiseks**

Mina _____ (sünnikuupäev: _____)

(autori nimi)

1. annan Tallinna Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose

(lõputöö pealkiri)

mille juhendaja on _____,

(juhendaja nimi)

säilitamiseks ja üldsusele kättesaadavaks tegemiseks Tallinna Ülikooli Akadeemilise Raamatukogu repositooriumis.

2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.
3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tallinnas/Haapsalus/Rakveres/Helsingis, _____

(digitaalne) allkiri ja kuupäev

Sisukord

Sissejuhatus	5
1. Õppemängud	6
2. Kõnetuvastus	8
2.1. Kõnetuvastuse ajalugu	8
2.2. Kõnetuvastuse kasutamine.....	10
2.3. Kõnetuvastus Eestis	10
2.4. Kõnetuvastuse tehnoloogiad	12
2.4.1. Google Cloud Speech	12
2.4.2. CMU Sphinx.....	14
2.4.3. Windows Speech Recognition.....	15
2.4.4. Wit.ai	16
2.4.5. IBM Speech to Text.....	17
2.4.6. Speechmatics	18
2.5. Kokkuvõtte tehnoloogiatest	18
3. Arendusmetoodika	21
4. Arendus	23
4.1. Mängu kontseptsioon	23
4.2. Mängu reeglid	24
4.3. Prototüübi disain	24
4.4. Mängu arendus.....	26
4.4.1. Kõnetuvastuse rakendamine	26
4.4.2. Mängumehaanika arendamine	27
4.4.3. Vaadete arendamine	28
5. Tulemused	30
5.1. Kõnetuvastuse kasutamine.....	30
5.2. Valminud prototüüp	31
5.3. Edasised sammud.....	31
Kokkuvõtte	34
Summary of „The Usage of Speech Recognition Based on a Language Learning Game“	35
Kasutatud kirjandus	36

Sissejuhatus

Tänapäeval on saanud õppimine läbi mängude arvestatavaks osaks iga noore inimese õppeprotsessist. Õppemänge on olemas väga mitut tüüpi ning nende tüübid sõltuvad tavaliselt kas mängu sisust või eesmärgist. Olemas on näiteks: loovmänge, mälumänge, rollimänge, matemaatilisi mänge jne. Üheks nendest tüüpidest on keeleõppe mängud.

Aastate jooksul on mängustamine õppes saanud järjest populaarsemaks ning õppemängude edasine areng toimub pidevalt. Seminaritöö „Digitaalsed keeleõppemängud“ keskendus konkreetsemalt keeleõppe mängudele. Uuringu käigus selgus, et kuigi paljud õppemängud kasutavad tänapäevast tehnoloogiat, on õppemängudes rakendatavatest tehnoloogiast puudu just keeleoskuse arendamine. Inimesi õpetatakse lugema, kirjutama ning mõistma, mida mõni sõna, väljend või lause tähendab, kuid igal õppuril on vaja keelt ka suuliselt praktiseerida, et osata end paremini väljendada ning keelt kinnistada.

Käesoleva teema valik tulenes autori isiklikust huvist keeleõppemängude vastu ning seminaritöö tulemusest. Kõnetuvastuse ehk inimkõne süstemaatilise sisendiks töötlemise jaoks on olemas erinevaid lahendusi, mida saaks kasutada, et õppuritele paremini keelt edasi anda. Näiteks saaks õppur kirjutamise asemel vastata küsimustele suuliselt või nähes mingisugust pilti saaks ta pildile vastava võõrkeelse sõna öelda. Võimalusi kõnetuvastust rakendada on palju.

Töö eesmärgiks on rakendada kõnetuvastustehnoloogiat keeleõppe mängus. Eesmärgi saavutamiseks on autor püstitanud järgmised uurimisküsimused:

- 1) Millised on kõnetuvastustehnoloogiate kasutusvõimalused?
- 2) Millised kõnetuvastustehnoloogiad on tänapäeval kasutusel?
- 3) Kuidas rakendada olemasolevat kõnetuvastustehnoloogiat keeleõppe mängus?

Eesmärgi saavutamiseks tehakse alustuseks lühitutvustus õppemängudesse. Teises peatükis antakse ülevaade kõnetuvastusest ning kõnetuvastuse tehnoloogiast. Teostatakse lühike taustauuring erinevatest tehnoloogiast, et valida keeleõppe mängu prototüübi loomise jaoks parimad võimalikud vahendid. Kolmandas peatükis antakse ülevaade arendusmeetoditest ning planeeritakse prototüübi arendusprotsess. Neljandas peatükis arendatakse loodava mängu prototüüp. Viimases ehk viimases peatükis antakse ülevaade töö tulemustest ja puudustest ning pakutakse ideid edasisteks arendusteks.

1. Õppemängud

Tõsimängud on mängud, mille peamine eesmärk erineb meele lahutamisest (Djaouti, Alvarez & Jessel, 2008). Kui tavalised mängud keskenduvad meelelahutamisele, siis autori silmis on hea tõsimängu üheks omaduseks, et mängul on sees ka meelelahutuslikud elemendid. Selle all peab autor silmas seda, et kuigi tõsimängu eesmärk erineb tavalise mängu eesmärgist, peaks ta olema integreeritud meelelahutusliku eesmärgiga.

Tõsimängud jagunevad peamiselt nende kasutamise valdkondade ja eesmärkide järgi. Olemas on mängude tervise jaoks, reklaammänge, mängude tööna jne. Õppemängud omakorda on üks tõsimängude alamliikidest. Õppemängude näol on tegemist mängudega, mis kasutavad ära mängulist ülesehitust selle jaoks et muuta kasutaja jaoks igavana näiv tegevus huvitavamaks. Näiteks luuakse kaasahaarava tegevusega mäng, mille taustal toimub probleemide lahendamine (Buckley, 2015). Selle asemel, et kasutajale lihtsalt matemaatilisi ülesandeid ette anda, antakse talle mänguline ülesanne. Selleks võib olla näiteks välja arvutamine, kui palju nende mänguline tegelaskuju peab turult soetatud esemete eest maksma ning seda kõike selleks, et tegelaskuju ei saaks müüjalt petta. Lisaks sellele liigitakse ülesande lahendamise korral mängus saavutatava eesmärgi suunas edasi.

Keeleõppemängud on üks tüüp õppemängudest, mille eesmärgiks on kasutaja keeleoskuse arendamine. Sellist tüüpi tõsimängudega üritatakse kasutajatele mänguliste elementide abil õpetada nii uute sõnade tähendust kui ka konteksti, milles neid kasutatakse.

Eelnevalt on autor koostanud seminaritöö teemal „Digitaalsed keeleõppe mängud“. Seminaritöö raames andis autor ülevaate keeleõppes, tõsimängudest ning erinevatest keeleõppe mängudest. Keeleõppe peatükis tõi autor välja keeleõppes arendatavad oskused. Nendeks olid: lugemine, kirjutamine, kuulamine ja rääkimine. Seminaritöö käigus selgus, et suurem osa autori poolt leitud keeleõppe mängudest arendavad vaid kirjalikku väljendusoskust ning vähesel määral arendatakse ka kuulamise oskust. Tegemist on väga vajalike oskustega eneseväljendamise jaoks, kuid paraku jääb mängudes puudu suulise keeleoskuse arendamisest.

Lähtudes tänapäevaste tehnoloogiate võimalustest leidis autor, et seminaritöös välja toodud puudusest tulenevalt võiks rakendada kõnetuvastust õppimise eesmärgil. Sellest lähtuvalt on püstitatud käesoleva töö eesmärk - rakendada kõnetuvastuse tehnoloogiat keeleõppemängus. Selle jaoks, et sellist mängu looma hakata on esmajoonel tarvis uurida

milliseid kõnetuvastuse tehnoloogiaid üldse eksisteerib ning uurida kuidas nad töötavad. Seejärel on tarvis välja selgitada, kuidas neid tehnoloogiaid saab töö eesmärgi saavutamise jaoks rakendada.

2. Kõnetuvastus

Kõnetuvastuse all mõistetakse tehnoloogiat, mille abil leitakse sisendist, tavaliselt kas inimkõnest või helifailist, tulenevalt kõige paremini sobivamad sõnad, fraasid ning laused. Antud tehnoloogia võimaldab inimestel luua mitmesuguseid erinevaid programme ning nende funktsionaalsusi, kus kõnetuvastuse tulemusest tulenevalt antakse kasutajale kas mingisugune väljund või jätkatakse edasist töötlemist kõnetuvastusest tulenevate andmetega (Foneetika ja kõnetehnoloogia laboratoorium, kuupäev puudub). Väljund oleneb täielikult sellest, mis otstarbel rakendust kasutatakse. Näiteks, kui eesmärgiks on kasutajale kuvada tema öeldu ekraanil, siis see on sisendist tulenev väljund.

2.1. Kõnetuvastuse ajalugu

Esimene teadaolev ning dokumenteeritud kõnetuvastuse süsteem loodi aastal 1952. Siis sai *Bell Laboratories*'i poolt loodud süsteem aru numbritest. Täpsemalt sai loodud süsteem aru numbrite jadadest, mille vahele olid paigutatud pausid. Toonase tehnoloogiataseme kohta oli tegemist kalli süsteemiga, mistõttu ka loodut laiemalt rakendama ei hakatud. Seda sündmust peetakse ühtlasi ka kõnetuvastuse süsteemide alguseks ning juba kümme aastat hiljem, aastal 1962 esitles IBM oma masinat „*Shoebox*“, mis toona sai aru juba kuueteistkümnest inglisekeelsest sõnast. „*Shoebox*“ suutis aru saada numbritest 0st 9ni, kui öeldi number ning sellele lisati näiteks käsklus „pluss“, „miinus“ või „kokku“, suutis ta instrueerida arvutusmasinat arvutama ning printima vastuseid lihtsamatele aritmeetilistele probleemidele (IBM, kuupäev puudub).

1970ndatel loodi esimene kõnetuvastusega seotud kommertsfirma „*Threshold Technology*“, mis samal aastakümnel andis välja ka oma „*VIP-100*“ süsteemi, mis leidis esmast kasutust televisioonifirmade tootekvaliteedi kontrollis ning *FedEx*'is. Selle süsteemi edust tuleneval investeeriti läbi Ameerika kaitseministeeriumi agentuuri *DARPA* (*Defense Advanced Research Projects Agency*) kõnetuvastuse arengusse üpris kopsakaid summasid nende programmi „*SUR*“ (*Speech Understanding Research*). Programmile sai seatud toonase tehnoloogia kohta üpris suur eesmärk – tuhandet sõna valdav kõnetuvastuse süsteem. Aastal 1976 programm lõpetati, sest nende eesmärki täita ei suudetud. Ometigi said *DARPA*-st innustust paljud teised ettevõtjad (Juang & Rabiner, 2004).

Nagu eelnevalt mainitud, siis kõnetuvastuse algusaegadel oli antud temaatika suunatud pigem teadlastele. Üldisemaks eesmärgiks oli luua masin, mis saaks aru inimese kõnest ning oskaks sellele vastavalt reageerida ning isegi vastata. 1980ndatel aastatel jõudis kõnetuvastus sellisele tasemele, et erinevate tehnoloogiate abil osati tänu *Hidden Markov Model*'i kõnetuvastusele sõnavara ohtralt laiendada. Tegemist on mudeliga, milles asendati varem kasutusel olnud kõnetuvastuse süsteem tõenäosusega, et tegemist on kõnega. Seda ideed rakendati seetõttu, et iga inimene räägib erinevalt. Erinevalt rääkimisega kaasneb ka erinev akustiline mudel ning vastavalt sellele on raske kõnetuvastust kõigile rakendada. Kui mudel 80ndate keskpaigas avalikustati, muutus ta tolle ajastu põhiliseks mudeliks kõnest aru saamisel (Juang & Rabiner, 2004).

80ndate lõpus hakati kõnetuvastust laiemalt rakendama ka kommertseesmärkidel ning vastava suunitlusega tööstustes. Kodudesse jõudis kõnetuvastus esmalt läbi *Worlds of Wonder*'i loodud nuku „Julie“. Tegemist oli nukuga, mida lapsed said oma häälele reageerima õpetada (Pinola, 2011). Kuigi 80ndate lõpuks paistis, et kõnetuvastus oli saavutanud olulised edusammud, seisid nad siiski ühe väga suure probleemi ees. Nimelt oli probleem, et pärast igat sõna pidi inimene pausi tegema, et masin jõuaks sellest aru saada. Arvestades seda, et tavaline inimene räägib keskmiselt umbes 100 kuni 150 sõna minutis, tähendas see seda, et iga sõna vahele pausi tegemine aeglustas oluliselt masinaga suhtlemist. Selle probleemi lahenduse leidmise suunas oli ka edasine suunitlus kõnetarkvara arendamisel.

Läbivalt aastate jooksul arenes kõnetuvastus vahelduva eduga edasi ning tänaseks on inimesteni jõudnud kõnetuvastus läbi erinevate rakenduste ja tehnika ning pea iga inimese jaoks on tuttavad *Google Speech*'i ning Apple'i toode *Siri*. Tänapäevasel nutitefonil on võimalik kasutada kõnetuvastust, et oma telefoni kasutamist lihtsustada. Kõnetuvastuse areng on laienenud isegi sõidukite maailma. Näiteks on olemas autosid, mille mõningaid toiminguid saab häälkäsklustega juhtida ning Ameerika Ühendriikide kõige uuema hävitaja F-35 kokpit on hääljuhitav (Gallagher, 2014). Piloot saab samal ajal, kui ta lendab anda käsklusi, mis tänu kõnetuvastusele täidetakse. Pideva tehnoloogia arengu abil on võimalik laiendada kõnetuvastuse võimaluste kasutamist läbi erinevate olemasolevate tehnoloogiate.

2.2. Kõnetuvastuse kasutamine

Väga pikka aega oli kõnetuvastus inimeste jaoks pigem ulmeline teema. Inimesed kujutasid kõnetuvastust ette kui midagi müstilist, mille kohta peamiselt filmidest näha ja raamatutest lugeda sai. Kõnetuvastuse arenguga on kaasnenud kättesaadavus ning tema rakendamise võimalused sellise tasemeni, et kui varasemalt oli kõnetuvastus peamiselt kasutusel inimeste poolt, kes selle vastu konkreetset huvi või selle järgi vajadust tundsid, siis tänaseks päevaks on ta pea kõigi jaoks kättesaadav.

Kõnetuvastuse rakendamise valdkondi on väga mitmeid. Kui eelnevalt sai mainitud, et tänapäevast kõnetuvastust rakendatakse nii kommertseesmärgil kui ka igapäevaselt, siis lisaks sellele on üheks väga oluliseks valdkonnaks meditsiin. Lisaks sellele, et inimesed kel on vaja abi arvutite või seadmete kasutamisel, on läbiviidud uuringud selgitanud, et läbi uudsete kõnetuvastuse meetodite on kõne ning emotsioonide salvestamise abil võimalik tuvastada Alzheimeri tõve või Parkinsoni tõve varajasi sümptomeid (Xiaobo, Singhal, Fang, Krishna, Patel & Hsiao, kuupäev puudub).

Eelnevalt toodi näide Ühendriikide uue hävitaja F-35 kõnetuvastuse võimalusest, kuid militaarvaldkonnas on sama riigi poolt välja töötatud ka teisi tehnoloogiaid. Üheks selliseks tehnoloogiaks on näiteks erinevad tõlkevahendid. Kuna sõjajoonides on sageli puudu tõlkidest, kes aitaksid kohalikus keeles suhelda, otsustati nende kaitseministeeriumi agentuuri *DARPA* eestvedamisel arendada tarkvara reaajas kõne tõlkimiseks (Phys.org, 2010). Sellise tarkvara arendamine vähendab nii inimressursi kulu kui ka ohuolukordi, mis tekivad võõrkeelsete inimeste omavahelisest arusaamatusest.

Need võimalused on kõik kasutatavad tänu kõnetuvastuse arengule ning rakendamisele erinevates elu valdkondades. Lisaks olemasolevatele võimalustele tekib edasise tehnoloogia ja ühiskonna arenguga võimalusi järjest juurde.

2.3. Kõnetuvastus Eestis

Lisaks eeltoodud välismaistele näidetele ja teistes riikides, peamiselt Ameerika Ühendriikides, toimunud kõnetehnoloogia arengule, on ka Eestis teema leidnud teatud käsitlust. Peamiseks eestikeelse kõnetuvastuse arendajaks võib lugeda Tallinna Tehnikaülikoolis olevas Küberneetika Instituudis leiduv Foneetika ja kõnetehnoloogia

laboratoorium. Nende tegevusvaldkonnad sisaldavad nii eesti keele foneetilisi uuringuid, kui ka kõnetehnoloogilisi uuringuid ja arendustöid (Foneetika ja kõnetehnoloogia laboratoorium, kuupäev puudub). TTÜ Küberneetika Instituut on ka Eesti Keeletehnoloogia Riikliku Programmi kõnetuvastuse projektide läbiviija.

Projektide „Kõnetuvastus“ ja „Kõnetuvastus 2“ eesmärkideks on arendada eestikeelset kõnetuvastust ning parandada selle kvaliteeti, seda nii reaajas kui ka reaajatu kõnetuvastuse korral. Lisaks sellele soovitakse teha kõnetuvastustehnoloogia koos lähtekoodiga kättesaadavaks (Eesti Keeletehnoloogia Riiklik programm, kuupäev puudub).

Läbi antud projektide arengu on kättesaadavaks tehtud mitmeid erinevaid rakendusi ning funktsionaalsusi. Heaks näiteks on vabatahtlikult projektis osaleva Kaarel Kaljuranna poolt välja antud rakendused „Kõnele“ ja „Diktofon“. Tegemist on Android operatsioonisüsteemile suunatud rakendustega, mis pakuvad kasutajale võimalust kasutada oma telefoni mikrofoni selle jaoks, et talle käsklusi anda ja heli salvestada. Rakendus „Diktofon“ suudab salvestada ning tulemuse seejärel tekstiks ümber tõlgendada. Rakendus „Kõnele“ on Foneetika ja kõnetehnoloogia laboratooriumis arendamisel. Lisaks Android'i rakendustele on nad välja andnud „Veebipõhine kõnetuvastus“ nimelise veebis paikneva rakenduse. Tegemist on helifailide tõlgendamiseks loodud rakendusega, see tähendab, et kasutaja poolt lisatud helifail tõlgendatakse ning seejärel saadetakse kasutaja e-mailile või kuvatakse koheselt ekraanile.

Lisaks eelnevalt välja toodud rakendustele on nende veebilehel kättesaadavad kaks reaajalise kõnetuvastuse varianti. Üheks neist on kõnetuvastaja lähtekood, mis kujutab endast üles seatavat serverit, mida seejärel saab kõnetuvastuseks kasutada. Lähtekoodile lisaks on lehe kirjelduses välja toodud nii inglisekeelne kui ka eestikeelne demo. Mõlema rakenduse katsetamisel, mõne lause abil, ilmnesid küll vead, kuid rakendus töötas. Teiseks reaajaliseks kõnetuvastuse variandiks on *CMUSphinx*'il põhinev kõnetuvastuse server, mida käesoleva töö tehnoloogiapeatükis eraldi käsitletakse.

Uurides Eesti Keeletehnoloogia Riikliku Programmi ning nende kahte kõnetuvastuse projekti, selgub, et ka Eestis tegeletakse kõnetuvastuse arendamisega ning nende suund näib autori silmis õige. Kindlasti oleks huvitav näha väga väheste vigadega eesti keelt tuvastavat kõnetuvastuse tehnoloogiat.

2.4. Kõnetuvastuse tehnoloogiad

Antud peatükis tutvustatakse kõnetuvastuse tehnoloogiaid, mida bakalaureusetöö raames võrreldakse ning bakalaureusetöö eesmärgi saavutamiseks valitakse nende hulgas sobivaim. Antakse ülevaade vabalt kättesaadavate kõnetuvastuse tehnoloogiate olemusest. Samuti proovib autor saada tööle nende poolt pakutav demo või demo puudumisel, tehnoloogia.

Töö eesmärgi saavutamiseks on tarvis leida kõnetuvastuse tehnoloogia, mis pakub eelkõige reaajas talle õpetatud sõnastiku abil kõne tuvastamist või saab võimalikult palju aru ka ilma talle sõnade õpetamiseta. Sellele lisaks annavad tehnoloogiale lisaväärtust kasutamise lihtsus ning dokumenteeritus. Töö käigus kasutada olnud riistvara piirangutest tulenevalt on eelistatuks kas veebibrauseris, Windowsis või Android operatsioonisüsteemil toimivad lahendused.

2.4.1. Google Cloud Speech

Google Cloud Speech on Google'i *Cloud Platform*'i teenus, millega on võimalik nii heli kui ka kõne tuvastada. *Cloud Platform* on Google'i platvorm mille abil saab rakendada nende poolt pakutavaid erinevaid teenuseid. Teenuste hulka kuuluvad nii andmetöötluse mootorid, Google'i *Cloud* andmebaasid, nende poolt pakutavad andmete hoidlad kui ka erinevad turvasüsteemid. *Google Cloud* platvormi teenused on omavahel ühilduvad ning nende teekides on olemas omavahelised seosed, et erinevaid rakenduse osasid omavahel integreerida ja kasutada (Google Cloud Platform, kuupäev puudub).

Cloud Speech'i näol on tegemist sama tehnoloogiaga, mida kasutatakse Android operatsioonisüsteemides kõne abil nii otsimiseks, kui ka näiteks telefoni kasutamise lihtsustamiseks. *Cloud Speech* saab aru rohkem kui 80st keelest, mis võimaldab teda rakendada globaalselt. Samaaegselt kui informatsiooni töödeldakse, toimub ka masinõppimine, läbi mille parandatakse tulemusi ning mõistetakse järjest paremini inimkõnet. Google'i masinõppimine on üks *Cloud* platvormi teenuseid, mille abil töödeldakse informatsiooni ning sarnaselt närvivõrgu struktuurile luuakse infokildude vahel sidemeid. Google'i kõnetuvastus töötab väga paljudel erinevatel platvormidel. Üheks huvitavaks lisaomaduseks on, et *Cloud Speech* oskab töötada mürarikas keskkonnas, kus

kasutajal on vaja mingisugust informatsiooni edasi anda (Google Cloud Platform, kuupäev puudub).

Tavaline inimene on antud kõnetuvastusega kokku puutunud kas läbi Android operatsioonisüsteemis oleva *Google*'i otsingu või operatsioonisüsteemile antavate käskluste kaudu. *Cloud Speech*'ile on võimalik saata andmeid nii asünkroonselt kui sünkroonselt. Sünkroonse variandi puhul antakse kasutajale esimesel võimalusel kohest tagasisidet, samal ajal kui asünkroonse puhul peab kasutaja ise kõne salvestamise peatama. Asünkroonse kõnetuvastuse puhul töödeldakse heliklipp pärast salvestust ning seejärel antakse kasutajale vastav info tagasi. Sama süsteemi on võimalik kasutada, et rakendusse saadetakse helifail mille *Cloud Speech* oma võimekusele vastavalt ära tõlgendab ning seejärel saadud andmetega edasi tegeleb (Google Cloud Platform, kuupäev puudub).

Google Cloud Speech API ehk siis arenduse jaoks vajalikud teegid on jätkuvalt veel *beta* versioonis, mis tähendab, et *Cloud Speech* on pideval arenemisel samaaegselt kui teda juba rakendatakse erinevates rakendustes ning kasutatakse üle kogu maailma. Selle jaoks, et *Cloud Speech*'i arendada või teda oma rakenduses kasutada, on vaja hakata *Google Cloud Platform*'i kliendiks. Kasutaja saab rakendust proovida tasuta prooviajal, mille raames antakse kasutajale 300 dollari väärtuses krediiti, mida saab kasutada järgneva 12 kuu jooksul. Prooviaja lõppedes on teenuste edasised hinnad sõltuvad kasutajate hulgast ning kasutatavatest teenustest. Antud võimalust rakendati käesolevas uurimuses. Pärast *Google*'i platvormi *Google*'i infosüsteemi ülesseadmist on võimalik luua endale projekt ning sealt saadud võtme abil on juba võimalik käivitada nende poolt pakutav demo rakendus, mille ülesandeks on reaajas lugeda kõiki sõnu, mida kasutaja ütleb. Loetud sõnad kuvatakse kasutajale ning selle põhjal on võimalik näha, kuidas pakutav teenus loetut tõlgendab.

Google'i kõnetuvastuse süsteem kasutab sama tehnoloogiat ning pakub samu teenuseid, mida nende Android operatsioonisüsteemides oleva kõnetuvastuse puhul rakendatakse. See tähendab, et on võimalik kuulata kasutajapoolset sisendit nii reaajas kui ka audiofailide abil. Kuna tegemist on *Google Cloud* platvormi teenusega, ühildub ta nende sõnul ülejäänud pakutavate teenuste ja teekidega (Google Cloud Platform, kuupäev puudub).

2.4.2. CMU Sphinx

Carnegie Mellon University Sphinx, ühtlasi tuntud kui lihtsalt *CMU Sphinx* või *Sphinx* on kogum kõnetuvastuste arendamiseks kasutatavaid teeke ja tööriistu, mida on võimalik rakendada kõne kasutamiseks. *CMU Sphinx* on vabalt kättesaadav ning tema teeke ning näidiskoodi võib vabalt kasutada nii uurimis- kui kommertseesmärkidel. Tegemist on Carnegie Mellon'i ülikoolis tegutseva grupiga, kelle põhirahastus pärineb Ameerika Ühendriikide kaitseministeeriumi alluvusse kuuluvalt agentuurilt nimega *DARPA (The Defense Advanced Research Projects Agency)*, mille ülesandeks on murranguliste tehnoloogiate väljatöötamine rahvusliku julgeoleku nimel (DARPA, kuupäev puudub). Grupi eesmärgiks on stimuleerida kõnet kasutavate tööriistade rakenduste loomist ning antud suuna edasist arendamist (Lenzo, kuupäev puudub).

Sphinx'i on tänaseks välja antud neli samanimelist versiooni, *Sphinx* 1-4. Nendele lisaks on veel eraldi välja antud *PocketSphinx*. *Sphinx*'i erinevate versioonide vahe on nende funktsionaalsus ning kohandumine tehnoloogia arenguga. *Sphinx* 1 ja 2 on tänaseks vananenud ning on ka nende enda kodulehel soovitatud vaid inimestele, kes teavad täpselt mida nad teevad. Esimene versioon oli kõige algelisem. Teise versiooni kõige olulisemaks omaduseks on tema kiirus, ühtlasi on ta ka *PocketSphinx*'i eelkäijaks. *Sphinx* 3 on suure sõnavaraga kõnetuvastuse süsteem. Tegemist on vanema C-keelel põhineva tarkvaraga, mida *CMU* jätkuvalt toetab. Ühtlasi on kolmas väljalase baaskontrollijaks teistele versioonidele. See tähendab, et kolmandat versiooni *Sphinx*'ist kasutatakse, et võrrelda kui täpselt nad kõnet tuvastada oskavad. Neljas versioon on üsna sarnane kolmandale väljalaskele, lihtsalt uuendatud tehnoloogiliselt ning ühtlasi vahetatud *Java* keelele (CMUSphinx, kuupäev puudub).

PocketSphinx on *CMU* sõnul nende kõige kiirem süsteem. Antud versiooni puhul on tagasi pöördutud C-keele juurde ning sellele versioonile pakutakse erinevate seadmete (nt: iPhone, Windows jne) tuge. Seda versiooni soovitatakse kasutada *live* rakenduste puhul. Lähtuvalt sellest sai järgmises katses seda proovitud. *CMU* kodulehelt on võimalik alla laadida ning proovida nende enda poolt loodud Android'ile suunatud *demo*. Demo on võimalik kohe pärast alla laadimist käivitada ja kasutada. Antud demo näol on tegemist üsnagi lihtsalt ülesehitatud programmiga.

Demo käivitamise jaoks on vajalik alla laadida *Android Studio* ning kasutada, kas virtuaalset masinat või Android operatsioonisüsteemiga telefoni. Demo katsetamise jaoks kasutas autor oma isiklikku telefoni, millele ta läbi *Android Studio* rakenduse alla laadis. Demonstratsiooni käivitamisel on vaja juurdepääsu telefoni mikrofonile. Seejärel kuvatakse esimene vaade, kus telefonile öeldes „*oh mighty computer*“ liigutakse järgneva kolme vaate juurde. Seal on võimalik kolme erineva häälkäskluse abil katsetada erinevaid võimalusi. Esimeseks variandiks on, et saab rakendusele öelda numbreid ning need tõlgendatakse ümber tekstiks. Öeldud numbrid kuvatakse kasutaja ekraanile. Teiseks variandiks on „*forecast*“, millega tuntakse ära erinevaid ilmanähtusi vastavalt sellele, mida kasutaja ütleb. Kolmanda variandi puhul tuntakse ära erinevaid häälikuid ning kuvatakse neid ekraanile.

PocketSphinx'i dokumentatsioonist on võimalik välja lugeda, et tema põhilisteks funktsionaalsusteks on kasutajapoolse sisendi lugemine nii reaajas kui ka audiofailidest. Selle jaoks, et teda kasutama asuda, on vaja loodava projekti failide hulka lisada nende teek. Läbi selle teegi saab rakendada erinevaid meetodeid, mis kõnetuvastuse funktsionaalsust tagavad. Oluline on silmas pidada, et kõne tuvastuse jaoks kasutatakse hetkel koodis defineeritud keelemudelit ning rakenduse looja poolt defineeritud sõnu. Hetkel kuuluvad nende poolt pakutavate keelemudelite hulka saksa, prantsuse, hollandi, inglise, hispaania, mandariini, india, kasahstani ning vene keel (CMUSphinx, kuupäev puudub).

2.4.3. Windows Speech Recognition

Windows Speech Recognition on Microsoft'i poolt arendatud kõne tuvastuse komponent. Microsofti panus kõnetuvastusse sai alguse juba aastal 1993, kui nad palkasid endale kolm spetsialisti *Carnegie Mellon University*'st, kus parajasti arendati *Sphinx 2*-te. Microsofti eesmärgiks oli luua kõnetehnoloogia, mis oleks täpne ja arendajatele kättesaadav. Esimene laiem väljalase Microsofti kõnetarkvarast kaasnes Vista operatsioonisüsteemiga. Windows Vista sisaldab endas kõnetuvastuse tarkvara, mille abil saab operatsioonisüsteemi läbi häälkäskluste juhtida ning ühtlasi ka hääle abil trükkida (Robert Brown, 2006). Tänapäevaks on *Windows Speech Recognition* olnud jätkuvalt arengus ning operatsioonisüsteemidest on alates Windows 7-st saadik võimalik kasutada kõnetuvastust.

Microsoft on panustanud oma otsingumootorisse *Bing* kõnetuvastuse võimalustega ning pakuvad ka erinevaid võimalusi kõnega otsingut juhtida. Üheks oluliseks rakenduseks, kus Microsoft oma kõnetuvastustarkvara on rakendanud, on digitaalne assistent *Cortana*. *Cortana* on leidnud kasutust nende mobiilses operatsioonisüsteemis ning ühtlasi nende uusimas Windowsi operatsioonisüsteemis. *Cortana* on võimalik suhelda sarnaselt nagu paljudele tuttava iOS'i rakenduse *Siri*'ga.

Kui Microsofti eesmärgiks oli luua arendajatele kättesaadav kõnetuvastuse tehnoloogia, siis *Windows.speech* teegiga on nende eesmärgi suunas liigutud. Tegemist on üsnagi vabalt kättesaadava teegiga, mis on kaasatud *Unity* enda dokumentatsiooni ja teekidesse. Selle jaoks, et *Windows Speech Recognition*'it *Unity*'s loodud rakendusena arvutis kasutada on tarvis rakendada uusimat Windowsi operatsioonisüsteemi Windows 10. Sinna sisse ehitatud funktsionaalsus *Cortana* võimaldab kasutada rakendust ka internetiühenduse puudumise korral, kui rakendus on eelnevalt alla laetud. Ühilduvus on ka Windowsi telefonide operatsioonisüsteemiga. Microsofti enda poolt teegile näidiskrakendust loodud ei ole. Selle jaoks, et teegiga eksperimenteerida, on tarvis luua iseseisvalt rakendus või proovida mõnda internetis leiduvat näidist. Antud teegi näol on tegemist lihtsasti kasutatavate funktsionaalsustega, mida koostöös *Unity* funktsionaalsuste ja koodinäidetega on kerge kasutusele võtta. Kõnetuvastuse teegis välja toodud meetoditega saab arendaja seadistada teda ära tundma grammatikat, erinevaid fraase ning ka lihtsalt ette määratud sõnu.

2.4.4. Wit.ai

Wit.ai on *start-up*'ina alguse saanud kõnetuvastuse tarkvara, mille loomise eesmärgiks oli pakkuda väiksematele firmadele ning iseseisvatele arendajatele teatud võimekust, mis ei tuleks nii kalli hinnaga (Metz, 2014). *Wit* on peamiselt käsklustele ning dialoogidele suunatud tarkvara. Selle suuna võtsid nad seepärast, et olla olemas nutiseadmete kasvaval turul. Tänapäevaks on *Wit.ai* kasutusel üle 65 000 arendaja poolt (*Wit.Ai*, kuupäev puudub).

Kuigi tegemist on teiste suuremate kõnetuvastuse teenust pakkuvate organisatsioonide kõrval üsnagi pisikese ettevõttega, pakub nende tarkvara kõik vajaliku oma püstitatud eesmärgi täitmiseks. Ettevõtte kodulehel oleva demoga on võimalik veebilehitsejas anda

visuaalselt kujutatud maja kontrollivale „robotile“ käsklusi. Kui arendaja soovib *Wit*'i kasutada või selle kasutamist lähemalt uurida, piisab kui ta oma *GitHub*'i kontoga sisse logib ning seejärel on tal täielik juurdepääs teekidele ning lihtsamini loodavatele „*quickstart*“ rakendustele. Nende lihtsamate rakenduste peamiseks funktsionaalsuseks on käskluste õpetamine ning dialoogide loomine, st et kasutaja öeldule oskaks kas rakendus või masin vastata. Kuigi *Wit*'i rakendamine on üsna lihtne, vähemalt lihtrakenduste puhul, on tema peamiseks piiranguks *HTTP (Hypertext Transfer Protocol)* päringute kiirus. Selle jaoks, et väljaspool *Wit.ai* kodulehte nende kõnetuvastust kasutada on vaja teostada *HTTP*-päringuid nende serverisse, kust kasutaja poolt sisestatud käsklusele reageeritakse. Nimelt töötab *Wit.ai* kõnetuvastus nende poolt pakutava teenuse abil, mis suudab käsklustest välja lugeda tahte ning sellega kaasnevad argumendid.

2.4.5. IBM Speech to Text

IBM'i *Speech to Text* on osa Watson Developer Cloud'ist, mis pakuvad erinevaid teeki keele, kõne ja nägemise rakendamiseks arendajate rakendustes. *Speech to Text*'i näol on tegemist teenusega, mis pakub teeki rakendustele kõne tõlgendamise võimaluste lisamist. Selle jaoks, et inimese häält täpselt tõlgendada kasutab nende teenus IBM'i poolt kogutud informatsiooni grammatikast ja keele struktuurist koos teadmistega kõne signaali kompositsioonist. Nende poolt kogutud informatsioon põhineb erinevatel kõrge kvaliteediga audiofailidel, millest on omakorda koostatud närvivõrgu-laadne struktuur. Pakutav teenus tagastab ja pidevalt uuendab tõlgendust, mida rohkem kõnet kuulatakse (IBM, kuupäev puudub). Põhimõtteliselt on tegemist masinõppega keeletuvastuse tehnoloogias.

IBM'i kõneteenuse proovimiseks on loodud demo. Demo ise on üsna lihtsakoeline, võimalik on lisada audiofaile või salvestada oma mikrofonist heli. IBM'i kõneteenuse rakendamiseks on tarvis kasutada platvormi põhise teeki. Erinevate teekide võrdlusel selgub, et nende erinevused on küllaltki suured. Näiteks kui *Unity*'l põhinevas teegis pole veel võimaldatud mikrofonist kõne salvestamist siis *Java*-põhises teegis on seda juba võimaldatud. Kõige suuremaks probleemiks IBM'i kõneteenuse kasutamisel on, et teenuse tasuta kasutamine on piiratud tuhande tunni peale kuus. Koos rakenduse loomise, testimise ning juba mõnekümne kasutajaga saaks kuulimist täis ning sellega seoses edasine töö oleks piiratud.

2.4.6. Speechmatics

Speechmatics on kõnetuvastuse teenust pakkuv väikefirma, mis esmapilgul sarnaneb vägagi *Wit.ai*'le. Nende eesmärgiks on luua tehnoloogia, mida saab rakendada erinevatele kõne rakendustele. Seetõttu on nad teinud oma tehnoloogia kasutamise lihtsaks ning vabalt kättesaadavaks (*Speechmatics*, kuupäev puudub). Täna pakuvad nad kõnetuvastust kahekümmes erinevas keeles.

Nende veebilehte külastades jääb silma pakkumine, proovida reaajas demo, mis antud juhul on vaid *beta* versioonis. Hetkel kättesaadava kõnetuvastuse põhiliseks omaduseks on audiofailide üleslaadimine ning nende tõlgendamine. Kahjuks ei kattu nende poolt pakutav teenus eelnevalt seatud eeldusega leida reaajas kõne tuvastamise jaoks sobilik tehnoloogia.

2.5. Kokkuvõte tehnoloogiast

Eelnevalt käsitleti populaarsemaid ja kättesaadavamaid kõnetuvastuse tehnoloogiaid. Nende ühiseks teguriks on, et nad pakuvad võimalusi luua interaktiivseid lahendusi, kus inimesel on võimalik masinaga suhelda. Välja toodud tehnoloogiate vahel on teatud sarnasused ning nad täidavad sama eesmärki – võimaldada inimesel arvuti või masinaga suhelda. Kuldset keskteed on antud tehnoloogiate vahel raske leida, teatud asjad töötavad paremini arvutis ja teatud asjad mobiilsel platvormil.

Järgnevas tabelis 1 on välja toodud eelnevalt käsitletud tehnoloogiad. Tabelis on eraldi välja toodud nendega paremini ühilduvad platvormid, kõnetuvastuse põhimõtte ning lahenduse poolt seatud piirangud nii arendajatele kui ka kasutajatele. Koos piirangutega on välja toodud tehnoloogiapoolsed puudused.

Tabel 1. Kõnetuvastuse tehnoloogiate võrdlus

Tehnoloogia	Platvormid	Kõnetuvastuse põhimõte	Piirangud ja puudused
Google Cloud Speech	Ühildub kõigi platvormidega mis suudavad saata <i>REST</i> päringuid Google'i serverisse.	Kasutatakse Google Cloud platvormi kõnetuvastuse süsteemi kõne tõlgendamiseks.	Tasuline teenus, võimalik kasutada vaid aastase prooviajaga.
CMUSphinx	Ühildub nende sõnul kõigi platvormidega.	Rakendatakse kas nende poolt pakutavat või iseseisvalt loodud keelemudelit.	Raskesti loetav dokumentatsioon.
Windows Speech Recognition	Windows 10, Windows Phone	Rakendatakse Windowsi sisse ehitatud kõnetuvastust koos arendaja poolt koostatud andmetega.	Piiratud platvormide kasutus, sõltub operatsioonisüsteemis olevast kõnetuvastusest.
Wit.Ai	Ühildub kõigi platvormidega, mis suudavad saata <i>HTTP</i> päringuid Wit.ai serverisse.	Kasutatakse Wit.Ai süsteemi selleks, et tuvastada saadetud lausest tahe, tekst ning lisaks kaasa antud argumendid.	Serverisse saadetavate päringute vastuste aeglus. Põhineb käskluste ja dialoogide loomisel.
IBM Speech-To-Text	Ühildub kõigi platvormidega, mis suudavad <i>REST</i> päringuid saata.	Kasutatakse nende poolset teenust sõnade ja grammatika tõlgendamiseks.	Tasuline teenus, tasuta võimaldatakse vaid tuhat tundi kuus.
Speechmatics	Ettevõtte poolt ei ole täpsustatud	Kasutatakse nende poolset teenust sõnade tõlgendamiseks audiofailist või <i>beta</i> versioonis reaalaajast.	Reaalaajas on kõnetuvastus alles beta versioonis, võimalik vaid faile üles laadida.

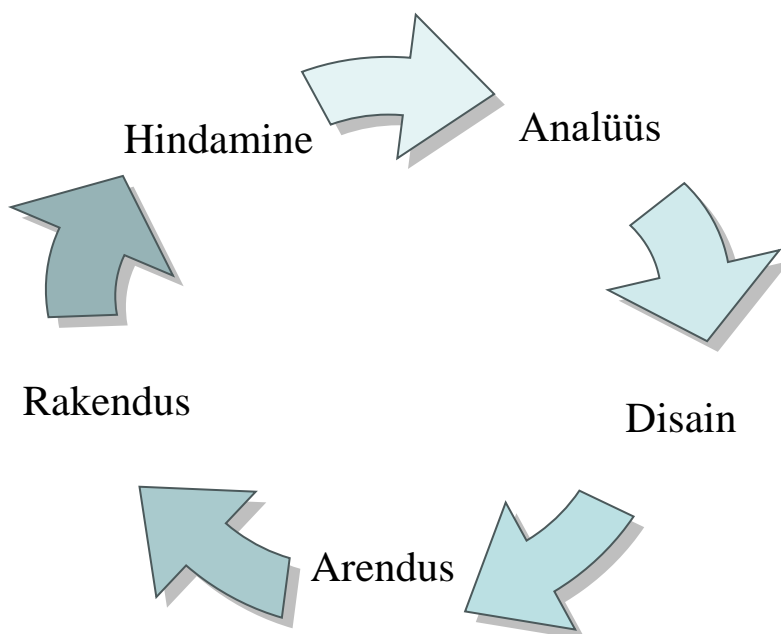
Eeltoodud tabeli põhjal jäid autorile parimate tehnoloogiatena silma Windowsi poolt pakutav teenus Windows 10 operatsioonisüsteemile ja telefonidele ning Google'i poolt pakutav teenus, mis ühildub väga paljude erinevate platvormidega. Windowsil põhinev kõnetuvastus paistis teiste hulgast välja oma rakendamise lihtsuse tõttu. Nimelt on teda

võimalik väga kiiresti kasutusele võtta. Lisaks sellele on ta sisse ehitatud *Unity* mängumootori teekidesse, mis teeb arendajale tema kasutamise veelgi mugavamaks. Lisaks eeltoodutele jäi oma kasutusvõimaluste ja platvormide mitmekesisuse poolest silma ka *CMUSphinx*'i toode *PocketSphinx*.

3. Arendusmetoodika

Antud bakalaureusetöö näol on tegemist arendusuuringuga. See tähendab, et lisaks teoreetilisele poolele antakse käesolevas töös edasi arenduse kulg ning selle tulemused. Selles peatükis on lähemalt välja toodud kasutusel olev metoodika ning autoripoolsed ideed loodava rakenduse jaoks.

Töö arenduse kavandamise jaoks on kasutatud ADDIE mudelit (vt joonis 1). Tegemist on peamiselt õppematerjalide ning instruksioonide koostamise jaoks kasutusel oleva protsessimudeliga. Kuna bakalaureusetöö eesmärgiks on luua õppeotstarbelise mängu prototüüp, siis protsessimudel on eesmärgi saavutamise jaoks väga sobiv.



Joonis 1. ADDIE mudel

Analüüsi pool on teostatud läbi seminaritöö ning ühtlasi ka eelnevas peatükis. Seminaritöö raames koostatud uurimusega analüüsiti olemasolevaid õppemänge ning nende võimalusi. Uuringu käigus selgus, et väga vähesed õppemängud pakuvad kõnetuvastuse jaoks võimekust ning sellest tulenes ka sobiva õppemängu vajadus. Käesoleva töö tehnoloogiapeatükis on analüüsitud erinevaid kõnetuvastuse tehnoloogiaid ja nende funktsionaalsusi.

Mängu disain sai alguse kontseptuaalse disaini loomisest. Kõigepealt pandi paika mängu kontseptuaalne disain. Selle käigus kujundas autor oma visiooni mängust. Seejärel loodi

kontseptuaalse disaini põhjal paberprototüübid loodavast mängust ning tema vaadetest. Disainifaasi lõpuks lõi töö autor esmased vaated kasutatavasse platvormi.

Arendusprotsessi käigus loodi esmalt arenduspiletid, lähtudes eelnevalt koostatud analüüsist ja disainist. Piletid on vastavalt funktsionaalsusele või eeldatavale funktsionaalsusele koostatud lühikirjeldused, millel on oma staatus. Staatused jagunevad tüüpiliselt kolmeks: vajab tegemist, töös ja valmis. Läbi sellise süsteemi on võimalik lihtsamini projekti hallata ning hoida pidevat ülevaadet projekti või rakenduse valmimisest. Arendusprotsessi läbiviimiseks on kasutatud kose arendusmudelit. Arendusprotsessis sai ühtlasi langetatud kõnetuvastuse tehnoloogia valik. Tehnoloogiapeatükis väljatoodud kolmikust sai autori valikuks *Windows Speech Recognition*. Kuigi tehnoloogia poolt kehtestatud piirangud takistavad teistele platvormidele laienemist, leidis autor, et selle kõnetuvastuse süsteemi rakendamine on talle jõukohane ning vastab tema poolt seatud tehnoloogilistele piirangutele. Läbi tehnoloogiapeatükis koostatud uuringu selgus, et *Windows Speech Recognition* on kaasatud ka *Unity* platvormi põhiteekidesse. Sellest tulenevalt otsustas autor kasutada *Unity* platvormi mängu arendamiseks. *Unity* näol on tegemist *Unity Technologies* poolt pakutava platvormiga ilusate ja kaasahaaravate 2D, 3D ja virtuaalreaalsuse mängude ja rakenduste loomiseks (*Unity*, kuupäev puudub).

Platvormi valikust tulenevalt valiti ka arenduse keel, milleks sai *C#*. Mängu loomise käigus kasutas autor põhisisu ja disaini jaoks *Unity* enda teeke. Kuna loodud mäng kasutab väga vähe andmeid, rakendas autor andmebaasi asemel failide nimesid. Pildifailidele on pandud sobivad nimed ning neid kasutab mängumootor loogika kuvamiseks. Versioonihalduse jaoks kasutas autor privaatset *GitHub*'i repositooriumit. *GitHub*'i versioonihaldus annab hea ülevaate läbi viidud muudatustest ning laseb arendajal probleemide korral liikuda tagasi eelneva versiooni juurde.

Bakalaureusetöö eesmärgist tulenevalt sai loodud kõnetuvastust rakendava keeleõppemängu funktsionaalne prototüüp. Prototüüpi testiti autori poolt, edastati autoripoolsed ideed ning edasise arenduse mõtted.

4. Arendus

Põhinedes metoodika peatükis välja toodud informatsioonile ning ühtlasi seal kavandatud mudelist lähtudes, kujunes välja antud mängu prototüübi arendus. Käesolevas peatükis antakse ülevaade:

- 1) loodava keeleõppemängu kontseptsioonist;
- 2) mängu reeglitest;
- 3) prototüübi disainist;
- 4) arendusest ja sellega kaasnevatest tegevustest;

4.1. Mängu kontseptsioon

Mängu arenduse disain sai alguse autoripoolsest mängu kontseptsiooni koostamisest. Selle raames lõi autor visiooni loodavale mängule.

- 1) **Nimi:** kuna plaanis on luua esmane prototüüp ei lähtunud mängu nime panemisel eesmärgist saavutada unikaalne nimi. Mängu nimi on justkui sissejuhatavaks küsimuseks mängule. Mängu nimeks sai „Kes on pildil?“.
- 2) **Idee:** mängu ideeks sai autori visiooni kohaselt luua lihtsasti kasutatav pildidel põhinev mäng. Mängu sisuks on kasutajale kuvatav pilt, millele vastavalt kasutaja kõnetuvastuse abil üritab õiget sõna leida.
- 3) **Õppemärk:** mäng on üles ehitatud eesti keeles ning tema õppe eesmärgiks on õpetada noortele mängijatele inglisekeelsete sõnade hääldust selliselt, et kõne tuvastav süsteem suudaks aru saada, millega on tegemist. Ühtlasi on kasutajatel võimalik läbi mängu korrata oma sõnavara.
- 4) **Meelelahutuslik aspekt:** kasutajatel on võimalik rakendada kõnetuvastust mängule vastamiseks ning koguda õigesti vastatud sõnade eest punkte.
- 5) **Mängu väljakutse:** mängu väljakutseks on leida vastavalt pildile õige sõna ning seda hääldada selliselt, et süsteem saaks sisestatavast sõnast aru.
- 6) **Mängija roll:** mängija on ühtlasi õppuri rollis. Vastavalt tema sisendile antakse talle punkte ning ta saab mängus edasi liikuda.
- 7) **Žanr:** mängu žanriks on õppemäng.
- 8) **Kasutajaliides:** autori visiooni kohaselt tuleks kasutajaliideses tagada lihtsus ja kasutusmugavus. Selle all mõtleb autor seda, et kasutajal on lihtne erinevate

vaadete vahel navigeerida ning keerukamates kohtades oleks lisatud kirjeldus edasisest tegevusest.

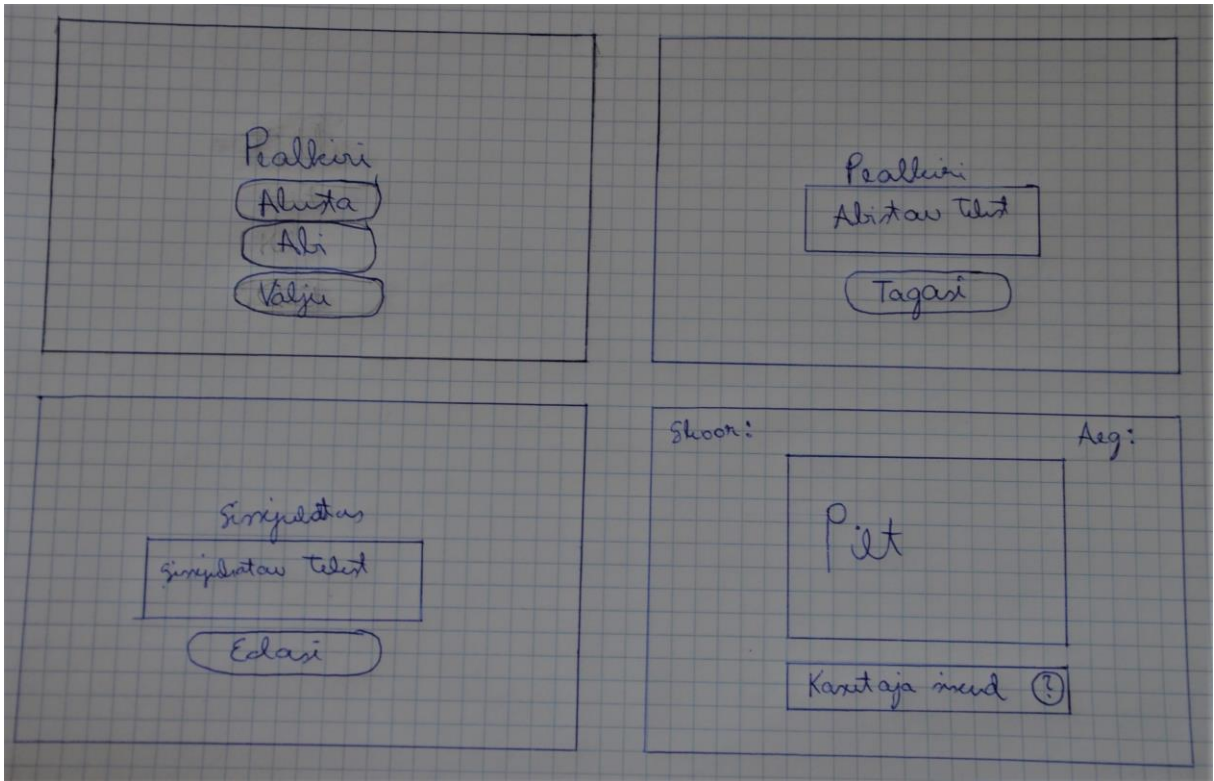
4.2. Mängu reeglid

Mängu reeglite kokkupanekul lähtuti eelkõige kontseptsiooni mudeliga paika pandud õppeesmärgist. Kuna keeleõpe saab tavaliselt alguse grammatikast, on antud mängu loomisel võetud suund grammatika ning häälduse õpetamisele. Mängijale näidatakse pilti ning mängija peab ütlema pildil oleva objekti nime. Mäng kontrollib seda, kas vastus oli õige või mitte.

- 1) Kasutaja, kes alustab mängu kasutamist, saab alustada mängimist kohe pärast lühikest sissejuhatust.
- 2) Mäng koosneb kahest peamisest osast, milleks on ekraanile kuvatav pilt ning kasutaja heliline sisend.
- 3) Vastavalt kasutaja sisendile antakse talle õigete vastuste korral punkte.
- 4) Ekraanil on ka võimalus valida järgmine sõna, mille korral kasutaja punkti ei saa. Selle järel liigutakse järgmise sõna juurde.
- 5) Mäng lõpeb kui kasutaja on kõikidele piltidele vastanud (vastuseks loetakse ka seda, kui ta järgmise pildi juurde edasi liigub).

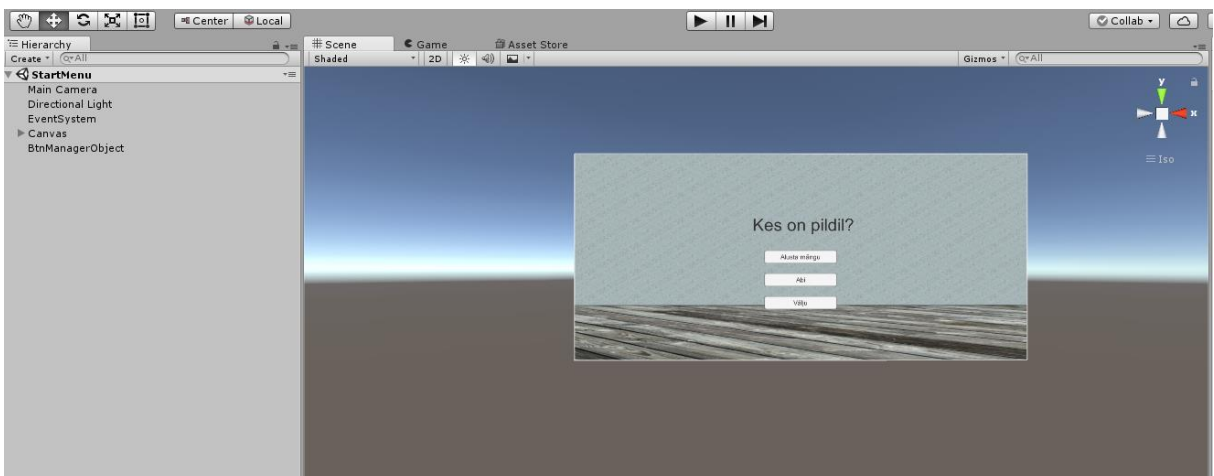
4.3. Prototüübi disain

Mängu enda disainimisega tehti algust visuaalsete prototüüpide loomise abil. Selle jaoks kasutati erinevaid visualiseerimise vahendeid, peamiselt paberprototüüpe (vt joonis 2). Paberprototüüpide heaks omaduseks on võimekus neid kiiresti luua ning nendega visualiseerida loodavat rakendust. Kuna mängu loogika ja ülesehitus on üsna lihtne, said ka prototüübid sarnase välimuse. Kasutajale antakse edasi võimalikult palju informatsiooni, et teha mängu mängimine kergesti arusaadavaks.



Joonis 2. Paberprototüübid

Paberprototüüpide põhjal koostati esimesed digitaalsed prototüübid, ehk *Unity* vaatenurgast juba esimesed arendused. Loodi vastavad stseenid, mille põhjal mängu edasine areng toimuma hakkab. Antud stseenid said koheselt ka sisu teksti, nuppude ja piltide näol (vt joonis 3).



Joonis 3. Kuvatõmmis rakenduses loodud stseenist

4.4. Mängu arendus

Mängu esimesed arendused tehti juba disainifaasis, paika pandi esmased stseenid. Sellega oli projekt juba *Unity*'s loodud ning üles seatud. Mängu arenduse jooksul jälgiti pidevalt olemasolevate piletite olukorda ning nende autoripoolset olulisuse hinnangut. Esmajoones tegutseti olulisemate ülesannetega, näiteks kõnetuvastuse rakendamine ning seejärel, kui funktsionaalsused olid olemas, asuti tegelema vaadete disainimisega.

4.4.1. Kõnetuvastuse rakendamine

Arendamise alustamiseks tuli esmalt uurida *Unity* dokumentatsiooni, et saada paremat ülevaadet *Windows Speech*'i võimekusest ning olemasolevast funktsionaalsusest ja meetoditest. Autor leidis, et mängu kõige olulisem osa – kõnetuvastus tuleb esmalt kasutusele võtta. Kõnetuvastuse jaoks on *Unity* teekides olemas eraldi *Windows Speech* nimeline teek. Käesolevas teegis on kõnetuvastust võimaldatud mitmel erineval meetodil. Võimalik on ära tunda grammatikat, erinevaid fraase ning ka lihtsalt ette määratud sõnu, mida kuulatakse. Samuti on võimalik eraldi failide abil määrata fraasidele semantilist tähendust. See tähendab, et seotakse erinevaid teadmisi ära tuntavate fraasidega. Antud mängu jaoks leidis autor parimaks lahenduseks ette määratud sõnad, kuna töötatakse peamiselt teadaolevaga ning see kiirendab ka mängu protsessi. See omakorda tekitab eelnevalt paika pandud mängu loogikasse juba esimese pisikese erinevuse. Nimelt oli autori esimeseks ideeks, et kasutaja näeb pilti ning saab seejärel mikrofoni öeldes arvutile teada anda, mida tema arvates antud visuaal kujutab. Kuna antud teegis puudub võimekus ära tunda täiesti võõraid sõnu (st sõnu, mis pole veel rakenduse sõnastikku arendaja poolt lisatud), tuleb vastavalt võimalustele rakendada teistsugust lähenemist. Selle all peab autor silmas, et teegi eripärast tulenevalt pakutakse välja funktsiooni, mis reageerib vaid õige sõna korral.

Antud kõnetuvastuse teegi kasutusele võtmine on üsna lihtne. Tarvilikud asjad on võimalik *Unity* dokumentatsioonist välja lugeda ning koheselt kasutusele võtta. Ühtlasi on dokumentatsiooni paigutatud ka rakendatavad näited, mis antud juhul lihtsustasid autori tööd.

Selle jaoks, et esmast mänguloogikat luua, lõi autor jada erinevatest ingliskeelsetest sõnadest. Kui kasutaja, kes mängu avab, ütleb sobiva sõna ning seejärel liigutakse järgmise sõna juurde. Samas kui kasutaja ütleb sõna, mis on kõnetuvastusele võõras, ei juhtu midagi, seda just eeltoodud teegi puuduse tõttu. Sellele probleemile nägi autor kahte lahendust. Üheks võimalikuks variandiks on õpetada rakendusele võimalikult palju sõnu, kuigi siis jääb täiesti võõraste sõnade probleem veel püsima. Teiseks võimalikuks lahenduseks on rakendada ajalisi piiranguid, mille lõppedes liigutakse järgmise sõna juurde. Kõnetuvastuse rakendamise jaoks antud probleemi lahendamine kohekselt vajalik polnud, seega jäi see edasiseks lahendamiseks.

Sellelega oli kõnetuvastuse pool kasutusele võetud. Selle jaoks, et mängul juba algusest peale rohkem visuaalset sisu oleks, sai sisse toodud ka vastavalt sõnadest koosnevale jadale pildimaterjal. Sellest tulenevalt otsustas autor, et töötab edasi pildimaterjali nimedega. See tähendab, et piltidele on antud sobivad nimed kasutajatele ära arvamiseks. Sõnade ära arvamine toimub juhuslikus järjekorras ning sõnade arv jadas oleneb ressursside kaustas olevast piltide hulgast. Antud lisaga muutus visuaalne pilt rohkem arusaadavamaks.

4.4.2. Mängumehaanika arendamine

Eelnevalt paika pandud stseenide ja esmase loogika põhjal sai hakata arendama edasist mehaanikat. Raskuspunktiks osutus mängu põhiline sisu ehk mängu enda funktsionaalsus. Eelmises peatükis püstitatud probleem – ajalise piirangu seadmine osutus oluliseks osaks mängu loogikas. Selle rakendamine kasvatab nii mängu raskust kui põnevust ning muudab mängu voolu sujuvamaks. Probleemi lahendamise jaoks sai leitud lihtne lahendus – kasutajal on aega kolmkümmend sekundit, et leida õige sõna. Kui mängija seda sõna selle aja jooksul öelda ei suuda, siis ta punkti ei saa ning liigutakse järgmise pildi juurde (vt joonis 4). Mänguloogikale lisandus siin veel üks mõte – kui juba on aega rakendatud, siis võiks seda kasutada ka punktide arvestamisel. Nimelt, mida kiiremini kasutaja vastab, seda rohkem punkte ta iga pildi eest saab.



Joonis 4. Ekraanitõmmis aja lõppemise korral loodud funktsionaalsusest

Autor lisas disainifaasis lisatud piltidele ja sõnadele veel sõnu, et anda kasutajale paremat ülevaadet tulemustest ning muuta tulemus kasutajale paremini arusaadavamaks. Kokku sai lisatud kümme pilti, mis kõik pärinevad *Pixabay* lehelt. Tegemist on veebilehega, millel on kõik pildid ja videod on kasutamiseks üles laetud *Creative Commons CC0* litsentsiga (*Pixabay*, kuupäev puudub). See tähendab, et materjali võib vabalt viitamiseta ning autoriõigusi tasumata kasutada.

4.4.3. Vaadete arendamine

Mängu mehaanika arendamise järel leidis autor, et oluline on tagada kasutajale liikuvus nii mängu sisse kui ka mängust välja. Üheks heaks omaduseks mängude juures on võimalus neid pausile panna ning jätkata siis, kui on kasutaja jaoks sobiv hetk. Samamoodi võib kasutaja otsustada mängu sulgeda. Antud probleemi lahendamise jaoks lõi autor mängu põhivaate sisse nupu „Menüü“, mille abil on võimalik panna mängu lihtsalt pausile, lõpetada antud mängu mängimine või mäng sulgeda (vt joonis 5). Ühtlasi vajab sarnast funktsionaalsust loodud lõpumenüü. Selle jaoks lisas autor loodud menüüsse vajalikud nupud ning funktsionaalsused.



Joonis 5. Ekraanitõmmis loodud menüüst

Loogika juures paika pandud mõte, et juhul kui kasutaja ei tunne antud sõna võiks tal olla võimalus liikuda järgmise juurde vajas samuti lahendust. Selle jaoks sai lisatud eraldi nupp, millele vajutades liigutakse järgmise sõna juurde. Esimest korda nupule vajutades antakse kasutajale vihje sõna pikkuse kohta. Teistkordsel vajutamisel ei anta kasutajale punkte, vaid käitatakse sarnaselt nagu vale vastuse korral.

Pärast mängu põhiliste funktsionaalsuste ja nende vaadetega tegelemist pöördus autor tagasi mängu algusesse. Disainipeatükis loodud peamenüü sai endale esmased funktsionaalsused, peamiselt nuppude näol. Loodi sissejuhatav tekst enne mängu algust ning eraldi stseen lisainfo jaoks. Sissejuhatavas tekstis antakse kasutajale teada, millega on tegemist ning mida järgnevalt oodata. Sissejuhatava stseeni järel teostatakse kontroll, kas kasutajal on olemas kõik eeldused selle jaoks, et antud mäng tema arvutis toimiks. Peamiseks eelduseks on kõnetuvastuse toimimine, mille kontroll teostatakse läbi *Windows Speech*'i teegi. Kui kasutajal puuduvad eeldused mängu toimimise jaoks, antakse talle sellest teada ning pakutakse lisainformatsiooniga linki probleemi parandamiseks. Seda just seetõttu, et Windowsi teek töötab vaid koostöös nende enda operatsioonisüsteemil põhineva kõnetuvastusega.

5. Tulemused

Mängu prototüübi arendusega jõudis autor nii kaugele, et täitis kõik loodud esimesed funktsionaalsuste piletid. Piletite üle arvestuse pidamiseks kasutas autor oma privaatset *GitHub*'i keskkonda. Piletid sisaldasid lühikirjeldust teostatavast tegevusest. Pileti sulgemisel kirjutas autor juurde, mis tegevused selle raames tehtud said. Huviliste jaoks on suletud pileteid võimalik vaadelda samast kohast kuhu ka prototüübi lähtekood on üles laetud: <https://github.com/carlmaask/TLUSpeechReco17/issues>.

Arenduse käigus lõi autor töötava prototüübi, mida on võimalik sobiva operatsioonisüsteemi ning eelduste täitmisega kasutada. Arendusega jõuti järgmiste tulemusteni.

5.1. Kõnetuvastuse kasutamine

Pärast autori koostatud uuringut erinevatesse kõnetuvastuse tehnoloogiatesse selgus, et erinevaid kättesaadavaid ning hästi kasutatavaid kõnetuvastuse elemente on võimalik leida. Kõnetuvastuse tehnoloogiate erisused ning puudused on välja toodud tehnoloogiate peatüki lõpus.

Antud bakalaureusetöö arenduse käigus rakendas autor Microsofti poolt pakutavat teeki *Windows Speech Recognition*. Nagu eelnevalt mainitud, siis tegemist on üsnagi lihtsasti rakendatava teegiga, just läbi *Unity* platvormi. Microsofti teegiga kaasnevad puudused, mis on antud bakalaureusetöös välja toodud. Kõige suuremaks puuduseks on see, et ta toetab hetkel vaid Windows 10 platvormi, sest Windows 10 platvormile on sisse ehitatud Microsoft'i toode *Cortana*. Töö käigus selgunud olulisteks lisapuudusteks olid teegis võõraste sõnade ära tundmise puudumine ning kõnetuvastuse toe tuvastamine. Nimelt puudub teegis meetod selliste sõnade, mida arendaja pole sisestanud, ära tundmine. See omakorda tähendab, et tuleb rakendada alternatiivseid meetodeid, näiteks käesolevas töös kasutatud ajalist piirangut sõnade arvamiseks. Toe tuvastamise probleem seisneb selles, et kui kasutaja arvutis pole Windows 10 operatsioonisüsteemi, ei suuda rakendus hetkel tuvastada kas kõnetuvastus on võimaldatud või mitte.

Kokkuvõtlikult võib öelda, et antud teegi kasutamine kõnetuvastuse jaoks on küll lihtne, aga oma puudustega. Samas on tegemist vabalt kättesaadava ja tasuta teegiga, seega

leiduvad tal nii positiivsed kui ka negatiivsed küljed. Antud bakalaureusetöö eesmärgi saavutamise jaoks oli teek väga sobiv ning tema abil prototüüp loodi.

5.2. Valminud prototüüp

Bakalaureusetöö raames valminud prototüüp on esmane versioon sellest, kuidas üks lihtne kõnetuvastust rakendav mäng autori arvates välja võiks näha. Olemas on esmased funktsionaalsused, mängu algus ning ka mängu lõpp. Ühtlasi on prototüüp lihtsasti disainitud ning tagatud on kasutaja liikuvusvõimekus. Kasutajale on antud abistavat informatsiooni mängu toimimise kohta ning lisatud on juhised kasutajale, kellel pole kõnetuvastust vastavalt tööle seatud.

Prototüübiga said täidetud kõik prototüübile seatud mänguloogika eesmärgid. Olemasolevate funktsionaalsustega on võimalik olemasolevaid sõnu harjutada ning nende hääldest kinnistada. Täidetud said arenduse jaoks seatud eesmärgid. Autor leiab, et mõningaid asju oleks veel võimalik täiendada ning need on välja toodud järgnevas peatükis.

Valminud prototüübi lähtekoodiga on võimalik huvilistel tutvuda *GitHub*’i avalikus repositooriumis veebiaadressil: <https://github.com/carlmaask/TLUSpeechReco17>. Prototüübi lähtekoodi on võimalik avada *Unity* abil ning sealt on võimalik ka mängu jooksutada. Ühtlasi on avalikku repositooriumisse laetud ülesse ka autori poolt ehitatud prototüübi versioon, mida Windows 10 operatsioonisüsteemil olevad inimesed jooksutada saavad. Prototüübi jooksutamise jaoks on tarvis sama operatsioonisüsteemi ning ühtlasi on tarvis ka võimaldada kõnetuvastus oma arvutis. Lisaks sellele on autori poolt koostatud ka lühike video rakenduse ja tema kõnetuvastuse toimimisest. Video on kättesaadav järgmiselt aadressilt: <https://www.youtube.com/watch?v=GORVPrAzPoU>.

5.3. Edasised sammud

Edasisteks sammudeks mängu täieliku arenemise suunas, oleks vaja tagada mängu ühilduvus teiste platvormidega, täiendada mängu mehaanikat ja loogikat ning luua lisafunktsionaalsusi. Kuna olemasolev prototüüp suudab töötada hetkel vaid Windows 10 platvormil, leiab autor, et edasiste uuringute jaoks võiksid huvilised katsetada samalaadse

lahenduse loomist mõne teise kõnetuvastuse tehnoloogiaga. Lisaks sellele võib kindel olla, et koos tarkvara edasise arenguga areneb ka antud kasutusel olev tehnoloogia.

Mänguloogika poolelt leiab autor, et edasisteks sammudeks võiks olla mängu keerukuse kasvatamine. Selle all on mõeldud nii õpivõimaluste kasvatamist, kui ka raskusastmete lisamist. Nimelt on võimalik täpsustada, kui täpselt sõnu kuuldaks ka teegipõhiselt. Selle all peab autor silmas seda, et *Windows.Speech* teegis on võimalus panna kõnetuvastust kuulama sõna vastavalt arendaja poolt kõnetuvastust käivitades seatud enesekindluse tasemele. Lisaks sellele võiks mängu edasiste arenduste korral lisada veel rohkem pilte ning sellega kaasnevalt ka sõnu. Õpivõimaluste kasvatamise all mõtleb autor erinevate sõnade lisamist ning erinevate kategooriate lisamist. Üheks lisaarenduseks võiks olla kasutaja tulemuse salvestamine ning edetabelisse kuvamine, mis antud prototüübi raames jäi realiseerimata.

Üheks huvitavaks lisamõtteks on see, et kui kasutaja kasutab paigutatud abinuppu, öeldakse talle sõna arvuti poolt. Kahjuks antud teek ei võimalda *speech-to-text* funktsionaalsust, mis tähendab, et selle teegiga sellist võimalust ei ole.

Lisaks nendele edasiarendustele leiab autor, et kindlasti võiks mängust arendada ka muukeelseid versioone kui vaid antud versioon, mis on mõeldud eesti keelt kõnelevatele noortele inglise keelsete sõnade harjutamiseks. See jääb hetkel küll antud teegi raames vaid ideetasandile, sest *Windows Speech Recognition* töötab hetkel vaid inglise keeles. Ühtlasi läbi tehnoloogiapeatükis teostatud uuringu jäi autorile silma, et väljatoodud teekidest polnud ühelgi neist ametlikku eesti keele tuge.

Bakalaureusetöö raames tekkinud põhiliseks küsimuseks jääb jätkuvalt erinevate kõnetuvastuse tehnoloogiate puudused ning kasutusvõimalused. Autori arvates võib antud teema olla eriti kasulik tulevastele õppemängude arendajatele, sest kõne näol on tegemist siiski ühe keeleõppe alustalaga. Selleks, et keelt osata, tuleb osata seda rääkida.

Nagu eelnevalt sai välja toodud võiks üheks lisamõtteks olla uurimus teistesse kõnetuvastuse mootoritesse ja teekidesse, mis võivad pakkuda veel rohkem ning paremaid funktsionaalsusi kui autori poolt valitud teek.

Üheks ettepanekuks võiks olla ka idee rakendada kõnetuvastust selliselt, et luua rakendus mis saab aru ja oskab tegutseda lausete kaupa. Bakalaureusetöö raames koostatud uurimuse käigus selliseid raamistike autor leida ei suutnud. Üheks huvitavaks näiteks oli *Wit.Ai*, mis

suudab käskluse sisust aru saada ning eraldada eesmärk ehk tegevus, tekst ning lisaargumendid.

Kokkuvõte

Antud bakalaureusetöö eesmärgiks oli rakendada kõnetuvastustehnoloogiat keeleõppe mängus. Eesmärgi saavutamiseks püstitatud uurimisküsimustele vastuste leidmise jaoks uuris autor esmalt kõnetuvastustehnoloogiate kasutusvõimalusi, seejärel lõi õppemängu prototüübi ning testis loodud prototüüpi.

Selle jaoks et uurida kõnetuvastustehnoloogiate kasutusvõimalusi teostas autor analüüsi erinevatesse vabalt kättesaadavatesse tehnoloogiatesse. Lisaks analüüsile katsetas autor võimaluse korral kõnetuvastuse rakendamist ning andis edasi sellega seondunud informatsiooni. Seejärel koostas autor analüüsitud tehnoloogiatest lühikokkuvõtte.

Õppemängu prototüübi loomisel selgitab autor esmalt kasutatud arendusmetoodikat ning platvormide ja tehnoloogiate valikut. Arenduse protsessis toob autor esmajärgus välja nii kontseptuaalse disaini loomise kui ka kasutajaliidese loomise protsessi alguse. Seejärel kirjeldab autor arenduse protsessi detailsemalt, tuues välja kõnetuvastuse rakendamise ning mängu prototüübi loomise protsessi.

Tulemustes annab autor kõigepealt lühiülevaate loodud prototüübi tulemustest. Seejärel kirjeldab autor kõnetuvastuse rakendamist ning annab edasi täpsemat informatsiooni prototüübi lõpptulemusest. Lõpetuseks annab autor ideid edasisteks prototüübi arendusteks ning toob välja tekkinud lisaküsimused.

Bakalaureusetööst võiks kasu olla arendajatel ja tudengitel, kes soovivad tutvuda kõnetuvastuse maailmaga või luua kõnetuvastuse tehnoloogiaid rakendavaid mänge või rakendusi.

Summary of „The Usage of Speech Recognition Based on a Language Learning Game“

The aim of this Bachelor's thesis was to implement speech recognition technology in a language learning game. To achieve this goal the author set three research questions:

- What are the usage possibilities of speech recognition software?
- What speech recognition technologies are in use today?
- How to implement an already existing speech recognition technology in a language learning game?

In the first chapter, the author provides an insight into learning games and speech learning games. The first part of the second chapter focuses on voice recognition. In the second part of the chapter, the author provides an analysis of freely available voice recognition technologies and ends the chapter with a summary of the findings.

The third chapter describes the development methods and tools used for the development of the prototype. The fourth chapter gives an insight into the development process of the prototype. In the last chapter, the author displays the results of the developed prototype and provides ideas for further developments and research.

This research paper could be useful for both developers and students who are interested in the world of speech recognition or have intentions to create games or applications that implement speech recognition.

Kasutatud kirjandus

Brown, R. (jaanuar, 2006). *Talking Windows: Exploring New Speech Recognition And Synthesis APIs In Windows Vista*. Loetud 8. aprill 2017 aadressil:

<https://web.archive.org/web/20080307054756/http://msdn2.microsoft.com/en-us/magazine/cc163663.aspx>

Buckley, C. (detsember, 2012). *Design And Implementation of a Genre Hybrid Video Game that Integrates the Curriculum of an Introductory programming course* (magistritöö). Loetud 23. aprill 2017 aadressil:

<https://www.slideshare.net/CoryBuckley/educational-game-design-thesis>

CMUSphinx (kuupäev puudub). *Versions Of Decoders*.

Loetud 23. aprill 2017 aadressil: <http://cmusphinx.sourceforge.net/wiki/versions>

DARPA (kuupäev puudub). *Defense Advanced Research Projects Agency*.

Loetud 17. aprill 2017 aadressil: <http://www.darpa.mil/>

Djaouti, D., Alvarez, J., Jessel, J.P. (2008). *Classifying Serious Games : the G / P / S model*. Loetud 28. märts 2017 aadressil:

http://www.ludoscience.com/files/ressources/classifying_serious_games.pdf

Eesti Keeletehnoloogia Riiklik Programm (kuupäev puudub). *Kõnetuvastus*.

Loetud 1. mai 2017 aadressil: <https://www.keeletehnoloogia.ee/et/ekt-projektid/konetuvastus>

Eesti Keeletehnoloogia Riiklik Programm (kuupäev puudub). *Kõnetuvastus 2*.

Loetud 1. mai 2017 aadressil: <https://www.keeletehnoloogia.ee/et/ekt-projektid/konetuvastus-2>

Foneetika ja kõnetehnoloogia laboratoorium (kuupäev puudub). *Kõnetuvastus*.

Loetud 1. mai 2017 aadressil:

<https://phon.ioc.ee/dokuwiki/doku.php?id=konetuvastus.et>

Gallagher, S. (25. juuli, 2014). „*Magic Helmet“ for F-35 ready for delivery*.

Loetud 23. aprill 2017 aadressil: <https://arstechnica.com/information-technology/2014/07/magic-helmet-for-f-35-ready-for-delivery/>

Google (kuupäev puudub). *Google Cloud Platform*.

Loetud 23. aprill 2017 aadressil: <https://cloud.google.com/>

IBM (kuupäev puudub). *IBM Shoebox*.

Loetud 8. aprill 2017 aadressil:

https://www-03.ibm.com/ibm/history/exhibits/specialprod1/specialprod1_7.html

IBM (kuupäev puudub). *About Speech to Text*.

Loetud 23. aprill 2017 aadressil:

<https://www.ibm.com/watson/developercloud/doc/speech-to-btext/index.html>

Juang, B. H., Rabiner, R. L. (10. august 2004). *Automatic Speech Recognition – A brief history of the Technology Development*.

Loetud 8. aprill 2017 aadressil:

http://www.ece.ucsb.edu/Faculty/Rabiner/ece259/Reprints/354_LALI-ASRHistory-final-10-8.pdf

Knight, W. (29. mai 2012). *Where Speech Recognition Is Going*.

Loetud 8. aprill 2017 aadressil:

<https://www.technologyreview.com/s/427793/where-speech-recognition-is-going/>

Lenzo, K. (kuupäev puudub). *CMU Sphinx*.

Loetud 23. aprill 2017 aadressil:

<http://www.speech.cs.cmu.edu/sphinx/doc/Sphinx.html>

Metz, R. (24. oktoober 2014). *Voice recognition for the Internet of Things*.

Loetud 17. aprill 2017 aadressil:

<https://www.technologyreview.com/s/531936/voice-recognition-for-the-internet-of-things/>

Microsoft (kuupäev puudub). *Microsoft Speech Platform*. Loetud 24. aprill 2017 aadressil:

<https://msdn.microsoft.com/en-us/library/jj127860.aspx>

Moiropoulos, M. (22. veebruar 2017). *Speech Recognition in Unity*.

Loetud 24. aprill 2017 aadressil: <http://lightbuzz.com/speech-recognition-unity/>

Niglas, K. (31. märts 2011). *Kombineeritud ning alternatiivsed uuringudisainid sotsiaalteaduslikus uurimistöös*. Loetud 17. aprill 2017 aadressil:

<http://www.tlu.ee/~katrin/generalia/Generalia.pdf>

Otsason, V. (21. november 2003). *Väle tarkvaraarendus. Väledad metoodikad*.

Loetud 17. aprill 2017 aadressil:

<http://kodu.ut.ee/~kiho/TVTKonspekt/VOtsasonAgile/VOtsasonAgile.pdf>

Phys.org (30. juuli 2010). *Breaking the language barrier: NIST tests language translation devices for US troops*.

Loetud 19. aprill 2017 aadressil: <https://phys.org/news/2010-07-language-barrier-nist-devices-troops.html>

Pieraccini, R. (kuupäev puudub). *From AUDREY to Siri. Is Speech recognition a solved problem?*

Loetud 17. aprill 2017 aadressil:

<http://www.icsi.berkeley.edu/pubs/speech/audreytosiri12.pdf>

Pinola, M. (2. november 2011). Loetud 8. aprill 2017 aadressil:

http://www.pcworld.com/article/243060/speech_recognition_through_the_decades_how_we_ended_up_with_siri.html

Pixabay (kuupäev puudub). *Pixabay*. Loetud 1. mai 2017 aadressil: <https://pixabay.com/>

Robarts, S. (17. oktoober 2014). *Speech recognition software: top six on the market*.

Loetud 8. aprill 2017 aadressil: <http://www.techradar.com/news/software/business-software/speech-recognition-software-top-six-on-the-market-1259815>

Speechmatics. (kuupäev puudub). *SpeechMatics*.

Loetud 23. aprill 2017 aadressil: <https://www.speechmatics.com/>

Unity. (kuupäev puudub). *About us*. Loetud 29. aprill 2017 aadressil:

<https://unity3d.com/public-relations>

Wit.Ai. (kuupäev puudub). *Wit.Ai*. Loetud 23. aprill 2017 aadressil: <https://wit.ai/>

Xiaobo, B., & Singhal, K., & Fang S. S., & Krishna S. S., & Patel V., & Hsiao W. (kuupäev puudub). *Application of Speech Recognition Technology in Speech-Related Disabilities: An Analysis and Forecast*. Loetud 8. aprill 2017 aadressil: <https://medium.com/@TEAM.BULLS/application-of-speech-recognition-technology-in-speech-related-disabilities-an-analysis-and-5b102da57f86>