

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond
Arvutiteaduse instituut

ITV40LT

Martin Kitsemägi 134658IAPB

**TEKSTITUVASTUSEGA
MOBIILIRAKENDUS EESTI LOTO
PILETITE KONTROLLIKS**

Bakalaureusetöö

Juhendaja: Jaagup Irve

Tehnikateaduste
magister
Tarkvarainsener

Tallinn 2016

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: Martin Kitsemägi

23.05.2016

Annotatsioon

Käesoleva bakalaureusetöö eesmärgiks on teha Android platvormile mobiilirakendus, mis võimaldab rakenduse kasutajal kontrollida füüsilisi Eesti Loto pileteid: Bingo Loto, Viking Lotto, Eurojackpot. Piletite kontrollimiseks peab kasutaja tegema oma lotopiletist pildi. Peale pileti pildistamist tuvastatakse pildilt vajalikud andmed, mida seejärel võrreldatakse vastava loosimise andmetega.

Bakalaureusetöö peamiseks probleemiks on pildilt andmete tuvastamine. Andmete tuvastamise efektiivsuse tõstmiseks on vajalik testida optilise kirjamärkide tuvastamise teeke ning erinevaid pildi töötlemiseks mõeldud algoritme. Samuti on vaja välja töötada pileti andmete filtreerimiseks, loosimise andmete kättesaamiseks ja nende andmete võrdlemiseks sobivad algoritmid.

Bakalaureusetöö eesmärgiks on viia pildi optiline kirjamärkide tuvastamine, sellele eelnev pildi töötlemine ning andmete filtreerimise, kogumise algoritmid võimalikult heale tasemele. Töö oodatavaks tulemuseks on pildilt vajaliku informatsiooni täielik kättesaamine ja selle põhjal pileti korrektne kontrollimine.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 35. leheküljel, 6 peatükki, 10 joonist, 0 tabelit.

Abstract

Mobile OCR solution for Estonian Lottery tickets

The purpose of this thesis is to create an Android mobile application, which allows the user to check physical Estonian Lottery tickets: Bingo Loto, Viking Lotto, Eurojackpot. In order to check tickets, the user must first take a picture of the according ticket. After taking a picture, necessary data is identified and retrieved from the ticket, which thereafter is compared to according draw data.

The main problem of this thesis is identifying necessary data from the picture. To increase data identification efficiency, various optical character recognition libraries and image processing algorithms need to be tested. It is also required to develop algorithms for data filtering, collecting draw data and algorithms to compare those data sets.

The aim of this thesis is to make the optical character recognition, image processing and data filtering, collecting algorithms as good as possible. The expected result is to completely retrieve necessary information from the picture and then correctly checking the ticket based on the retrieved data.

In order to realize the thesis, Android optical character recognition libraries are used to extract data from an image. For creating and processing an image, internal Android applications and algorithms are used. The Estonian Lottery homepage is used to collect specific draw data.

The thesis is written in Estonian and contains 35 pages of text, 6 chapters, 10 figures, 0 tables.

Lühendite ja mõistete sõnastik

ADT	<i>Android development tools</i> Android-i arendamiseks vajalikud töövahendid Eclipse keskkonnas.
Android NDK	<i>Android native development kit</i> Töövahend C/C++ koodi arendamiseks Android-is.
API	<i>Application programming interface</i> Tarkvara ja rakenduse ehitamisel kasutatavad rutiinid ning protokollid.
HTML	<i>HyperText Markup Language</i>
Lõim	Arvutiprogrammi osa, mille abil on võimalik jooksutada mitut protsessi paralleelselt.
OCR	<i>Optical character recognition</i> Tähemärkide teisendamine digitaalsele kujul.
Parsimine	Sõnade hulga analüüsimine, läbikäimine.
Regulaaravaldis	Tähemärkide kombinatsiooni mustripõhine otsimine sõnest.
SDK	<i>Software development kit</i> Sisaldab vastava platvormi arendamiseks vajalikke töövahendeid.
Sõne	Programmeerimiskeeltes teksti kujul olev märgijada.
Teek	Meetodid, komponendid, klassid, mida on võimalik programmides kasutada.
URL	<i>Uniform Resource Locator</i> Veebilehekülje internetiaadress.
XML	<i>Extensible Markup Language</i> Struktureeritud info märgistuskeel.

Sisukord

1	Sissejuhatus	9
1.1	Eesmärgid	9
1.2	Metoodika	10
1.3	Ülesehitus	10
2	Teooria	11
2.1	Optiline kirjamärkide tuvastus	11
2.1.1	Meetodid andmete tuvastamiseks	11
2.1.2	Pildi töötlemine	12
2.2	Material Design	13
2.2.1	Elementide asetus	14
2.2.2	Varjud	15
2.2.3	Värvid	16
3	Arendamisel kasutatud tehnoloogiad	17
3.1	Tesseract	17
3.1.1	Tess-two	17
3.2	Eclipse ja Android ADT	18
4	Rakenduse arendamine	19
4.1	Eeltöö	19
4.2	Pildi tegemine	20
4.3	Pildilt andmete kättesaamine	21
4.4	Pildilt saadud andmete filtreerimine ja valideerimine	22
4.4.1	Loosimise numbri filtreerimine	23
4.4.2	Mängunumbrite filtreerimine	24
4.4.3	Mängunumbrite valideerimine	25
4.5	Loosimise andmete kättesaamine	25
4.6	Pileti kontrollimine	27
4.7	Kasutajaliides	28

5	<u>Hinnang rakendusele.....</u>	<u>31</u>
6	<u>Kokkuvõte.....</u>	<u>33</u>
	<u>Kasutatud kirjandus.....</u>	<u>34</u>

Jooniste loetelu

Joonis 1. Eurojackpoti pilet enne ja pärast lävendamist.....	13
Joonis 2. Material Design-i mobiilirakenduse struktuur.....	15
Joonis 3. Kasutajaliidese elementide eristamine varjude abil.....	15
Joonis 4. Näide Material Design-i värvivalikust.....	16
Joonis 5. Rakenduse töö tegevusdiagramm.....	20
Joonis 6. Tess-two tuvastamise objekti loomine eraldi lõimes.....	22
Joonis 7. Pileti loosimise numbri filtreerimine.....	23
Joonis 8. Eesti Loto koduleheküljelt Bingo Loto loositud numbrite ning võidutasandite võidusummade kättesaamine HTML parsimise teel.....	27
Joonis 9. Rakenduse ekraanivorm, millel kuvatakse kasutajale pileti tuvastamisel saadud andmed.....	29
Joonis 10. Rakenduse ekraanivorm, millel kuvatakse kasutajale vastava loosimise loositud numbrid ning tema pileti kontrollimisel saadud tulemused.....	30

1 Sissejuhatus

Eesti Loto loosimistel osalemiseks on igal lotomängijal vaja osta lotopilet. Lotopileteid saab soetada interneti, sõnumi vahendusel või kaubanduskeskuste lototerminalidest. Internetist ja sõnumi vahendusel ostetud piletite kontrollimine toimub Eesti Loto vahendusel automaatselt, lototerminalidest ostetud piletite kontrollimine on aga raskendatud.

Müügipunktidest ostetud pileteid on hetkel võimalik kontrollida lototerminali abil või Eesti Loto koduleheküljel [1] oleva info põhjal. Käsitsi pileti kontrollimine võib osutuda tülikaks, eriti kui kontrollida on vaja mitu piletit. Samuti võib käsitsi kontrollides sisse tulla mõni viga. Vea tõttu on võimalik võidust ilma jääda, sest tihtipeale kontrollitud piletit eraldi enam lototerminali kontrollima ei minda.

Probleemi lahendamiseks tekkis idee luua mobiilirakendus, mis võimaldaks lotomängijal mugavalt kontrollida lotopileteid. Internetist vastava loosimise andmete otsimine ning piletinumbrite kontrollimine on aeganõudev tegevus. Rakenduse abil saaks kontrollimist automatiseerida ja vältida veaohtrikke olukordi.

1.1 Eesmärgid

Rakenduse eesmärgiks on pakkuda lotomängijatele alternatiivset meetodit lotopileti kontrollimiseks järgneva funktsionaalsusega:

1. Bingo Loto [2], Viking Lotto [3] ja Eurojackpoti [4] piletitest pildi tegemine, pildi alusel korrektne pileti loosimise numbri ning mängunumbrite tuvastamine.
2. Pileti kontrollimine vastava loosimise numbrite põhjal ning kasutajale tulemuste kuvamine.

1.2 Metoodika

Rakenduse realiseerimiseks kasutati Eclipse [5] keskkonda koos Android SDK [6] ja ADT [7] töövahenditega. Rakendus on kirjutatud Java [8] programmeerimiskeeles. Pildilt andmete tuvastamiseks kasutati avatud lähtekoodiga tess-two [9] töövahendit.

Rakenduse kujundamisel lähtuti Material Design [10] põhimõtetest. Kasutajaliidese elementide paigutus ning struktuur on vastavuses Material Design-i mobiilirakenduse nõuetega. Kasutajaliides on kirjutatud XML [11] formaadis.

1.3 Ülesehitus

Bakalareusetöö teine peatükk käsitleb optilise kirjamärkide tuvastamise ja Material Design-i teooriat. Kolmandas peatükis antakse ülevaade rakenduse arendamisel kasutatud tehnoloogiast. Neljandas peatükis antakse ülevaade rakenduse arendamisest ning kasutajaliidese disainist. Bakalareusetöö viiendas peatükis hinnatakse rakenduse lõpptulemust. Välja tuuakse rakenduse positiivsed, negatiivsed küljed ning lahendused, mille alusel rakendust paremaks muuta.

2 Teooria

Järgnev peatükk käsitleb optilise kirjamärkide tuvastamise [12] ning Material Design-i teooriat. Optilise kirjamärkide tuvastamise teoorias keskendutakse erinevatele meetoditele, mis on vajalikud pildilt andmete tuvastamiseks. Material Design-i teoorias tuuakse välja disainikeele põhitunnused ning nende rakendamine.

2.1 Optiline kirjamärkide tuvastus

Optiline kirjamärkide tuvastus, lühendina OCR, on meetod pildil oleva teksti muutmiseks masinloetavale kujule. OCR süsteemid suudavad eristada erinevaid kasutusel olevaid fonte ning tuvastada neid suure täpsusega. Üksikud OCR süsteemid on võimelised tuvastama ka käsitsi kirjutatud teksti.

Optiline kirjamärkide tuvastamine on tänapäeval leidnud kasutust mitmetes valdkondades. OCR-i kasutatakse tähtsate dokumentide, raamatute või tšekkide digiteerimisel. Ametivõimud kasutavad OCR-i näiteks passide kontrollimisel ning autode numbrimärkide tuvastamisel. Samuti on OCR kasutust leidnud pimedate inimeste abistamiseks mõeldud tarkvaras.

Andmete tuvastamise täpsuse tõstmiseks on välja töötatud erinevaid lahendusi. Tavaliselt loetakse pildil olevat teksti üks tähemärk korraga ning püütakse seda võrrelda OCR süsteemis olevate andmetega, kuid on olemas lahendusi, mis võrdlevad korraga terveid sõnu. Tuvastamise täpsuse tõstmiseks eelneb tuvastamisele pildi eeltöötlus, mille abil viiakse pilt must-valgele kujule, keeratakse pilti ning eemaldatakse pildilt häireid.

2.1.1 Meetodid andmete tuvastamiseks

Peamised meetodid andmete tuvastamiseks OCR-i abil:

- Maatriks sobitamine (*Pattern recognition*), mis võrdleb pildil olevaid tähemärke piksel–piksli haaval salvestatud tähemärkidega. Antud meetodi kasutamine eeldab, et võrreldav tähemärk on korralikult eraldatud ülejäänud pildist ning et salvestatud tähemärk kasutab sama fonti ja on samas moods. Maatriks sobitamine toimib kõige paremini trükitud teksti tuvastamiseks.
- Tunnuste eraldamine (*Feature detection*), mis kogub iga tähemärgi kohta kindlaid omadusi. Omaduste põhjal proovitakse kindlaks teha, mis tähemärgiga tegemist on. Vastavat meetodit kasutatakse käsitsi kirjutatud teksti tuvastamiseks.
- Kahe osaline tuvastus, mis kõigepealt leiab pildilt võimalikud tähemärgid ning proovib seejärel neid tähemärke tuvastada (*Adaptive recognition*). Vastav meetod võimaldab madala kvaliteediga või tundmatu fondiga pildilt suurema tõenäosusega kätte saada korrektseid andmeid.

2.1.2 Pildi töötlemine

Tuvastamise täpsuse tõstmiseks viiakse OCR süsteemides läbi erinevaid pilditöötlus operatsioone:

- Pildi pööramine, kus pilti pööratakse vajaduse korral, et pildil olevat teksti horisontaalselt või vertikaalselt joondada.
- Pildi lävendamine (*Binarization*), kus kõigepealt pildil olevad pikslid muudetakse vastavaks halltooniks [13] ning seejärel mustaks või valgeks [14]. Mustaks muudetakse pikslid, mille värvitooni väärtus on suurem (tumedam) antud lävendväärtusest ning valgeks pikslid, mille värvitooni väärtus on antud lävendväärtusest väiksem (heledam). Antud operatsioon aitab pildil olevat teksti paremini eristada taustast. Järgnev joonis illustreerib Eurojackpoti pileti pilti enne ja pärast pildi lävendamist.



Joonis 1. Eurojackpoti pilet enne ja pärast lävendamist.

- Pildi silumine, kus pildil olevate pikslite värvitooni väärtused viiakse paremini vastavusse vastavate pikslite naaberpikslite värvitooni väärtustega. Selline tegevus muudab pilti udusemaks, kuid aitab siluda pildil olevate tähemärkide ääri, tõstes sellega OCR tuvastamise täpsust.
- Tähemärkide eraldamine, kus kõigepealt leitakse üles pildil olevat sõnad ning tähemärgid. Kui kaks erinevat tähemärki asuvad üksteisele liiga lähedal või on üksteisega ühendatud, siis eraldatakse vastavad tähemärgid üksteisest. Antud operatsiooni kasutatakse OCR süsteemides, kus andmete tuvastamiseks võrreldakse ühte tähemärki korraga.

2.2 Material Design

Material Design on Google poolt loodud kasutajaliideste arendamise jaoks mõeldud disainikeel, mis baseerub klassikalistel hea disaini põhimõtetel. Disainikeel on loodud kasutamiseks erinevatel platvormidel ning erineva suurusega seadmetel. Material Design-i kasutatakse peaaegu kõigis Google mobiilirakendustes ja veebikeskkondades.

Material Design aitab värvide, varjude ja animatsioonide toel esile tõsta kasutajaliidesses olevaid elemente. Elemente ekraanil kujutatakse kahemõõtmeliste objektidena, kuid varjude abil on neid näiliselt võimalik ülespoole tõsta, luues sellega eristumise efekti.

Värvide ja animatsioonide abil suunatakse kasutaja tähelepanu kindlatele elementidele, mis omavad tähtsust.

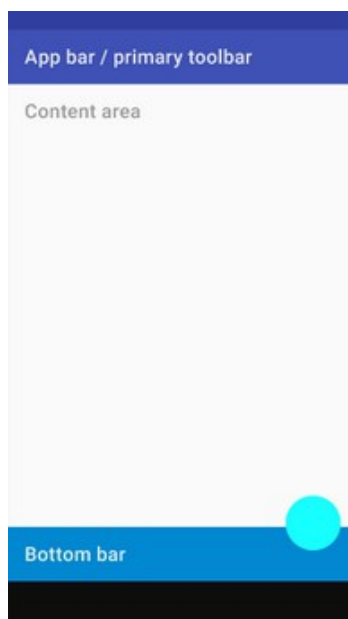
Kasutajaliidese disainimiseks on paigas kindlad reeglid, mida on soovitatav järgida. Reeglid puudutavad kõiki kasutajaliidese aspekte (nupud, värvid, kirja pilt, kirja suurus, ikoonid jpm.). Reeglite eiramine võib mõjutada kasutajakogemust.

2.2.1 Elementide asetus

Elementide eristamiseks varjude abil, on elementide paigutamisel vaja järgida kindlat struktuuri. Mobiilirakenduste struktuur on järgnev:

- Ekraani üleval servas asetseb rakenduse peamine tööristariba, kus kasutajale kuvatakse antud ekraanivormi pealkiri ning rakenduse kasutamisel vajalikud nupud, näiteks otsingu vormi avamise nupp või rakenduse sätete nupp.
- Tööristariba all asetseb rakenduse keha ehk sisu, kus kuvatakse kasutajale rakendusega seotud informatsiooni ning kus toimub rakendusega seotud põhitegevus.
- Ekraani alumises servas asetsevad antud ekraanivormiga seotud nupud (näiteks uue tegevuse alustamine, andmete kinnitamine).

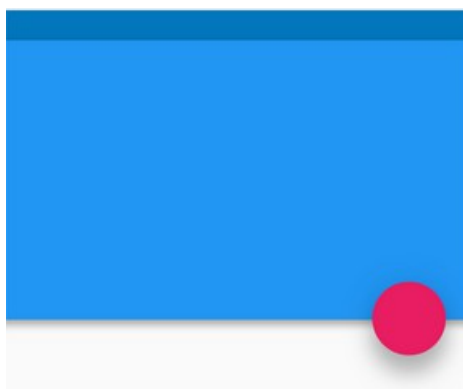
Järgnev joonis illustreerib Material Design–i mobiilirakenduse põhistruktuuri.



Joonis 2. Material Design-i mobiilirakenduse struktuur.

2.2.2 Varjud

Varjud pakuvad kasutajale visuaalset informatsiooni objekti sügavuse ning liikumissuuna kohta. Varjude abil on kasutajaliidese komponente võimalik üksteisest eristada. Material Design-i reeglite järgi, on igal elemendi tüübil erineva kõrgusega vari. Erinevat tüüpi elemendid ei tohi asetseda üksteise sees, vaid peavad kasutajaliidese seisukohalt olema eristatavad. Järgnev joonis illustreerib elementide eristamist varjude abil.



Joonis 3. Kasutajaliidese elementide eristamine varjude abil.

2.2.3 Värvivid

Kasutajaliidese elementide värvide valik, on veel üks moodus, mille alusel kasutajaliidese elemente üksteisest eristada või esile tuua. Material Design–is lähtutakse värvivalikul tavaliselt ühest põhitoonist. Erinevad elemendid saavad oma värvi põhitooni tumedamatest või heledamatest toonidest. Põhitooni valikust sõltub samuti rakenduses kasutatav tekstivärv, kus tumedatel taustadel eelistatakse heledamat ning heledatel taustadel tumedamat tekstivärvi. Mobiilirakenduse struktuurist lähtudes, on rakenduse tööriistariba põhitooniga ühte värvi. Kõik, mis tööriistaribast ülespoole jääb on põhitoonist tumedama värvusega ning kõik elemendid, mis jäävad tööriistaribast allapoole on põhitoonist heledama värvusega. Järgnev joonis illustreerib kasutajaliidese disainimisel üht võimalikku värvivalikut.

Primary text	#000000	87%
Secondary text	#000000	54%
Disabled / Hint text	#000000	38%
Primary color	#3E50B4	100%
Accent color	#FF3F80	100%

Joonis 4. Näide Material Design–i värvivalikust.

3 Arendamisel kasutatud tehnoloogiad

Järgnevas peatükis antakse ülevaade rakenduse arendamisel kasutatud tehnoloogiatest. Välja tuuakse erinevate tehnoloogiate positiivsed küljed ning põhjused, miks just neid tehnoloogiaid rakenduse arendamisel kasutati.

3.1 Tesseract

Tesseract [15] on avatud lähtekoodiga OCR töövahend, mille arendamist on rahastanud Google alatest 2006. aastast. Tesseract on kirjutatud C ja C++ programmeerimiskeeltes, kuid olemas on ka teistes programmeerimiskeeltes kirjutatud Tesseract-i alamharusi. Näiteks Tess4J [16], mis võimaldab Tesseract-i kasutada Java-s.

Viimase paari aasta jooksul on Tesseract-i erinevate keelte tuvastamise täpsus suurenenud. Hetkel võimaldab Tesseract tuvastava enamus maailmas kasutusel olevaid keeli. Igäühel on võimalik Tesseract-i nõ. treenida, luues vastava sõnastiku, mis sisaldab vastava keele või kirjamärkide tähemärkide andmestikku. Sõnastikus olevaid andmeid kasutatakse OCR tuvastamisel leitud tähemärkide võrdlemisel.

Tesseract-i loetakse hetkel üheks parimaks OCR töövahendiks, mida on võimalik igäühel tasuta kasutada. Võrreldes Tesseract-i teiste tasuliste OCR töövahenditega, tuleb Tesseract-is kasutajal pildi tuvastamise täpsuse suurendamiseks teha palju eeltööd (pildi keeramine, häirete vähendamine). Pildi töötlemis operatsioonidest, toimub Tesseract-i siseselt eelnevalt välja toodud pildi lävendamine.

3.1.1 Tess-two

Rakenduse arendamisel kasutatakse Tesseract-i alamharu tess-two, mis võimaldab Tesseract-i kasutada Android platvormil. Tess-two on palju eeldefineeritud erinevate keelte sõnastikke, mida saab arendamisel kasutada. Tess-two kasutamiseks, tuleb vastav lähtekood alla laadida ning seejärel eraldi Android NDK [17] töövahendiga

kompileerida. Kompileeritud lähtekoodi on edaspidi võimalik kasutada eraldi teegina igas Android-i projektis.

3.2 Eclipse ja Android ADT

Kogu rakenduse arendamine toimub Eclipse keskkonnas Java programmeerimiskeeles. Eclipse keskkonnas Android-i arendamiseks, on eelnevalt vaja alla laadida Android SDK töövahendid, mis sisaldavad erinevate Android API [18] versioonide arendamiseks vajalikke teeke. Samuti peab alla laadima Android ADT töövahendid, mis võimaldavad Eclipse keskkonnas luua Android-i projekte, neid testida, kujundada ning jooksutada erinevatel seadmetel. Android ADT kasutamisel, tuleb viidata Android SDK teekidele.

Android ADT-sse on sisseehitatud Android süntaks ning kasutajaliides rakenduse kasutajaliidese arendamise lihtsustamiseks. Kõik rakenduse kasutajaliidesega seotud failid on XML formaadis, kus defineeritakse erinevaid ekraanivormidel olevaid elemente. Elementide paigutamise jaoks kirjutatakse automaatselt XML koodi, kuid iga elemendi kujunduse ja funktsionaalsuse peab kasutajaliidese arendaja ise kirjutama. Sisseehitatud süntaks kontrollib, mis API versioonile parajasti rakendust arendatakse ning annab vea või halvasti kirjutatud koodi kohta tagasisidet (mis on halvasti, mis ei kõlba antud API versiooni arendamiseks).

4 Rakenduse arendamine

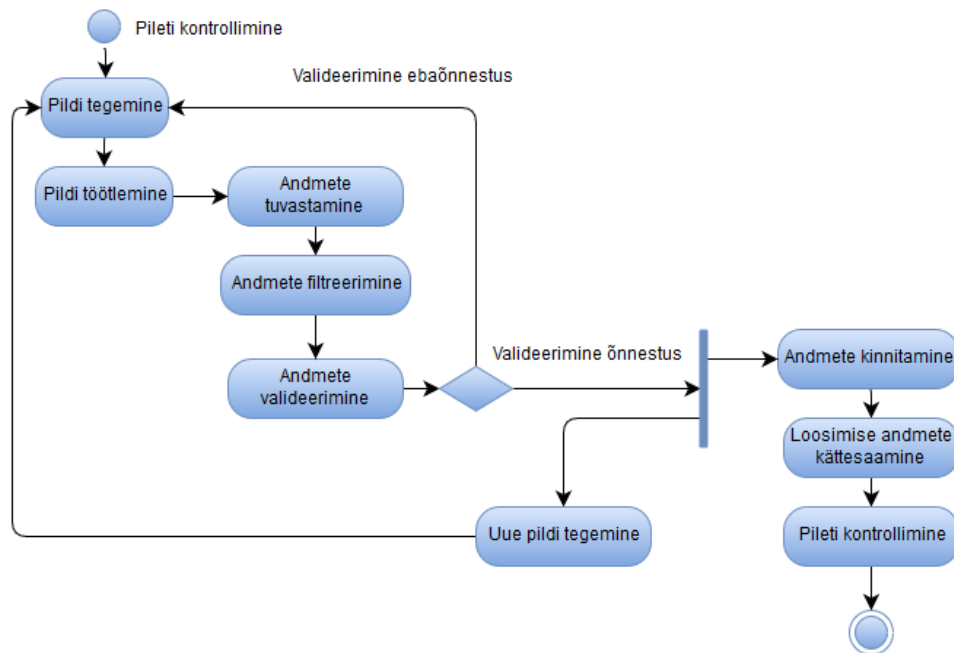
Järgnevas peatükis antakse ülevaade rakenduse arendamise protsessist ning selle erinevate komponentide tööpõhimõtetest. Komponentide kirjeldamisel tuuakse välja konkreetsed näited ning illustreerivad joonised.

4.1 Eeltöö

Enne Android-i platvormile rakenduse ehitamise alustamist, valmisid Java-s rakenduse tööks vajalikud komponendid. Java-s pandi peamiselt rõhku filtreerimis ning vastava loosimise andmete kättesaamiseks mõeldud algoritmide väljatöötamisele. Osaliselt testiti OCR teeke ning pilditöötlus algoritme. Põhi algoritmide väljatöötamine, andis ligikaudse ettekujutuse kogu rakenduse struktuurist. Samuti andis tähtsamate komponentide väljatöötamine kindlustunde, et vähemalt osa rakenduse komponentidest töötab. Kindlustunne oli vajalik, sest varem ei oldud Android platvormil midagi loodud.

Hiljem Android-ile üle minnes, tuli eelnevat kirjutatud koodi osaliselt ümber kirjutada. Valminud komponendid tuli ühildada rakenduse kasutajaliidesega ning erinevate operatsioonide tööks vajalike algoritmidega. Üldjoontes oli üleminek sujuv, sest Android-i arendus toimub samuti Java-s. Nüüd kulus suurem osa ajast väljaselgitamisele, kuidas Android-il arendada kasutajaliidest, kuvada veateateid ja ekraanivorme ning kuidas erinevaid telefonisiseseid operatsioone kasutada rakenduse töös.

Järgnev tegevusdiagramm illustreerib rakenduse pileti kontrollimise töö käiku. Pileti kontrollimise alustamiseks, tuleb vastavast piletist pilti teha. Peale pildi tegemist töödeldakse pilti ning alustatakse tess-two abil andmete tuvastamist. Kui pildilt ei õnnestunud tuvastada andmeid või tuvastatud andmed on valed, on pileti kontrollimiseks vaja teha uus pilt. Andmete kinnitamisel, otsitakse Eesti Loto kodulehelt vastava loosimise andmed, mille alusel toimub pileti kontrollimine.



Joonis 5. Rakenduse töö tegevusdiagramm.

4.2 Pildi tegemine

Peale rakenduse käivitamist, tuleb kasutajal pileti kontrollimise alustamiseks teha oma lotopiletist mobiiltelefoniga pilt. Avaekraanil antakse kasutajale teada, et pildile peab kindlasti jääma lotopileti loosimise number ning kõik vastava loosimisega seotud mängunumbrid.

Kaamera ja telefoni mälu kasutamiseks, tuleb rakendusele anda vajalikud õigused. Selleks tuleb AndroidManifest.xml faili lisada kaamera ning mälu kasutamise tingimused, mille alusel lubatakse antud rakendusel vastavaid operatsioone kasutada. Manifest fail sisaldab kõiki rakendusega seotud õigusi, vaateid ning sätteid.

Pildi tegemiseks, küsitakse kaamera operatsiooni käest luba (*Intent*) kaamera kasutamiseks. Kui kaamera kasutamine on võimalik, avatakse telefonisisene kaamera rakendus, millega kasutajal on võimalik pilti teha. Peale pildi tegemist, salvestatakse mallu pildi asukoht ning pildi orientatsioon. Pilt konverteeritakse ümber baidijadaks

(*Bitmap*) kasutades ARGB_8888 kodeeringut, kus iga piksel kasutab salvestamisel nelja baiti.

Enne pildilt andmete lugemist, vähendatakse pildi suurust ning muudetakse pildi orientatsiooni. Kuna telefoniga tehtud pildid on üldjuhul kõrge resolutsiooniga, tuleb OCR protsessi kiiremaks muutmisel vähendada tehtud pildi mõõtmeid. Kui mõõtmeid mitte vähendada võtab OCR tuvastus väga kaua aega (tess–two puhul ~30 sekundit), mis on kasutajale ebameeldiv. Mõõtmete vähendamisel on oluline, et pildil olevate tähemärkide kõrgus ei langeks alla kahekümne piksli, mida on vaja OCR poolt edukaks tuvastamiseks. Samuti tuleb kontrollida pildi orientatsiooni. Kui telefoni on hoitud pildi tegemisel näiteks horisontaalses asendis, tuleb pilt peale salvestamist eraldi vertikaalsesse asendisse keerata. Orientatsioon on vajalik, et pildilt loetakse hiljem andmeid õiget pidi.

4.3 Pildilt andmete kättesaamine

Nagu eelnevalt mainitud, kasutatakse pildilt andmete kättesaamiseks tess–two töövahendit. Tess–two töölepanekul, tuleb määrata andmetuvastus keel (traineddata), mille alusel andmeid tuvastatakse. Andmetuvastuskeel on fail, mis sisaldab vastava keele erinevaid tähemärke, mida kasutatakse pildilt saadud tähemärkide võrdlemisel. Andmetuvastuskeele fail tuleb eelnevalt telefoni salvestada ning selle kasutamiseks tuleb faili asukohale viidata. Andmetuvastuse täpsuse testimisel, testiti inglise ja eesti keele andmetuvastus keeli, millest täpsemaid tulemusi andis inglise keele kasutamine (eng).

Andmetuvastus operatsiooni ressursikulukuse tõttu, tuleb seda kasutajaliidese kokkujooksmise vältimiseks teha eraldi lõimes. Pildilt saadud andmed tagastatakse sõnena UTF–8 [19] formaadis põhilõimele, mis omakorda annab need edasi filtreerimisklassile. Andmetuvastuse käigus kuvatakse kasutajale laadimise animatsiooni.

Pildilt andmete kättesaamise kiirus ja täpsus sõltub pildi kvaliteedist, pildi valgustatuse tasemest ning pildil oleva info hulgast. Häguselt või halvasti valgustatud pildilt, ei ole

võimalik täpselt andmeid tuvastada. Kui pildi tegemisel jääb pildile kõrvalisi objekte, võtab pildi andmete läbikäimine samuti kauem aega (ajakulu kuni kümme sekundit). Pildilt andmete kättesaamine on kõige kiirem ja täpsem, kui pildil on täies mahus ainult kontrollitav lotopilet (ajakulu paar sekundit).

Järgnev joonis illustreerib tess–two objekti loomist eraldi lõimes.

```
@Override
public void run() {
    TessBaseAPI baseApi = new TessBaseAPI();

    baseApi.init(LANG_LOC, LANG);
    baseApi.setImage(bitmap);

    result = baseApi.getUTF8Text();
    baseApi.end(); //Closes tesseract

    Alerts.dismissProgressDialog(); //Dismiss progress dialog

    main.passOcrResult(result); //Pass result to main
}
```

Joonis 6. Tess–two tuvastamise objekti loomine eraldi lõimes.

4.4 Pildilt saadud andmete filtreerimine ja valideerimine

Peale pildilt andmete tuvastamist, filtreeritakse andmete seast välja pileti loosimise number ja pileti mängunumbrid, mida on vaja pileti edasiseks kontrollimiseks.

Loosimise numbrit kasutatakse vastava loosimise loositud numbrite ning võidusummade kättesaamiseks. Mängunumbreid kasutatakse vastava loosimise loositud numbrite võrdlemisel, mille abil selgitatakse välja mängija võidu tüüp ning võidetud summa. Andmete filtreerimisel kasutatakse regulaaravaldisi [20], tähemärkide asendamist (*replace*) ja sõnade tükeldamist (*split*).

Mängunumbrite filtreeritud informatsioon läbib omakorda valideerimise etapi, kus tehakse kindlaks, kas saadud andmed vastavad antud Eesti Loto loterii mängureeglitele. Kui valideerimise või filtreerimise käigus esineb mõni viga, antakse kasutajale sellest veateatega teada ning kasutaja peab tegema lotopiletist uue pildi. Õnnestunud filtreerimise ja valideerimise korral, suunatakse kasutaja järgmisele ekraanivormile, kus

kasutajal on võimalik kinnitada filtreerimise andmed ja piletit kontrollida või teha piletit uus pilt.

4.4.1 Loosimise numbril filtreerimine

Loosimise numbril filtreerimisel kehtib Bingo Loto, Viking Lotto ja Eurojackpoti piletitel sama struktuur. Loosimise numbril sisaldav rida, leitakse piletil tuvastamisel saadud sõnest üles järgneva regulaaravaldisega:

- `.*num\\w{3}\\s+\\d+.*` – mis tagastab true, kui rida mida kontrollitakse sisaldab kuuetaähelist kombinatsiooni algusega 'num' ning millele järgneb vähemalt üks tühik ja number.
- `.*ber\\s+\\d+.*` – mis tagastab true, kui rida mida kontrollitakse sisaldab kolmetähelist kombinatsiooni 'ber', millele järgneb vähemalt üks tühik ja number.

Kahe erineva regulaaravaldise mustri kontrollimine tagab suurema tõenäosuse, et sõnest leitakse üles rida sõnaga 'number', juhul kui mõni täht tuvastatakse valesti. Kui vastav rida on leitud, loetakse antud loosimise numbrilks esimest numbrilte kombinatsiooni, mis järgneb sõnale 'number'. Järgnev joonis illustreerib piletil loosimise numbril filtreerimist.

```
if (line.matches(".*num\\w{3}\\s+\\d+.*") || line.matches(".*ber\\s+\\d+.*")) {  
    if (line.matches(".*num\\w{3}\\s+\\d+.*"))  
        line = line.substring(line.indexOf("num") + 6, line.length());  
    else if (line.matches(".*ber\\s+\\d+.*"))  
        line = line.substring(line.indexOf("ber") + 3, line.length());  
  
    String[] numbers = line.replaceAll("[^\\d]", " ").split("\\s");  
    for (String number: numbers) {  
        if (ValidationMethods.isInteger(number)) {  
            gameNumber = Integer.parseInt(number);  
            return gameNumber;  
        }  
    }  
}
```

Joonis 7. Piletil loosimise numbril filtreerimine.

4.4.2 Mängunumbrite filtreerimine

Mängunumbrite filtreerimisel rakendatakse Viking Lotto ja Eurojackpoti mängunumbrite kättesaamisel järgnevat loogikat:

- Mängunumbreid sisaldavate ridade leidmiseks, kasutatakse regulaaravaldist: `.*[A–K]:s+\d+.*` – mis leiab üles kõik võimalikud mänguväljad, millega loosimisel osaleda saab. Iga erinev mänguväli on märgitud suure tähega, millele järgneb koolon ja mängunumbrid. Ühe piletiga on korraga võimalik mängida kümnel erineval mänguväljal.
- Kui pildil puudub mänguvälja algust märkiv suur täht või see tuvastati valesti, proovitakse mänguvälju tuvastada järgneva regulaaravaldise abil: `.*\d+\s+PP.*` – mis võtab arvesse fakti, et iga mänguvälja numbrite lõppu märgistab kaks P–d.
- Sobiva rea korral, eraldatakse sellelt kõik numbrid. Kui eraldatud numbrite hulk võrdub kuuega, on tegemist Viking Lotto loosimisega ning kõik ülejäänud sobivad read peavad samuti sisaldama kuute erinevat numbrit. Eurojackpoti puhul peab iga rida sisaldama seitset erinevat numbrit. Järgnevate ridade numbreid ei salvestata, kui antud rida ei vasta eelnevatele nõuetele.

Bingo Loto mängunumbrite kättesaamisel rakendatakse järgnevat loogikat:

- Mängunumbreid sisaldavate ridade leidmisel eeldatakse, et mängunumbrid on ümbritsetud nõ. kastidega, mida tess–two tuvastab eraldi tähemärkidena. Ridade üles leidmiseks, kasutatakse regulaaravaldist: `.*[I|I\|]+\s*\d+\s*[I|I\|]+.*` – mis leiab ülesse read, kus mõni number on kasti poolt ümbritsetud.
- Sobiva rea korral eraldatakse sellelt kõik numbrid ning kontrollitakse, et vastava rea ning iga järgneva sobiva rea numbrite hulk oleks võrdne viiega. Järgnevate ridade numbreid ei salvestata, kui antud rida ei vasta eelnevale nõudele.

Kui kogu leitud numbrite hulga ristsumma jääk on kuuega jagamise korral null, on tegu Viking Lotto loosimisega. Kui jääk on viiega jagamise korral null, on tegu Bingo Loto

loosimise numbritega ning kui jääk on seitsmega jagamise korral null, on tegu Eurojackpoti loosimise numbritega.

4.4.3 Mängunumbrite valideerimine

Filtreerimise tulemusel saadud mänguväljade numbrid kontrollitakse ükshaaval üle, et veenduda numbrite õigsuses. Kui vastava mänguvälja numbrite hulgast leitakse mõni number, mis ei ole vastavuses antud loterii reeglitega, jäetakse see numbrite hulk vahele. Tühja numbrite hulga tagastamisel, on valideerimine ebaõnnestunud ning antud andmeid ei saa kasutada pileti edasiseks kontrollimiseks. Viking Loto puhul peab igal mänguväljal olema kuus numbrit vahemikus 1–48. Eurojackpoti reeglite kohaselt on igal mänguväljal 5 põhinumbrit vahemikus 1–50 ja 2 lisanumbrit vahemikus 1–10, kus lisanumbrid on kaks viimast mänguväljal asetsevat numbrit. Bingo loto puhul on ühel mänguväljal kokku 25 numbrit, millest igal real asetseb viis. Esimene number vastavas reas peab jääma vahemikku 1–15, teine vahemikku 15–30, kolmas number vahemikku 30–45, neljas vahemikku 45–60 ning viies number peab jääma vahemikku 60–75.

4.5 Loosimise andmete kättesaamine

Pileti filtreerimise ja valideerimise õnnestumisel, otsitakse Eesti Loto kodulehelt üles vastava loosimise andmed, millest eraldatakse loosimise numbrid ning erinevate võidutasandite võidusummad. Andmete kättesaamiseks on telefonil vaja internetiühendust, ilma milleta ei ole võimalik URL informatsiooni parsimist võimalik alustada. Samuti tuleb AndroidManifest.xml faili lisada rakendusel interneti kasutamist lubavad tingimused. Internetiühenduse puudumisel kuvatakse kasutajale veateade.

Enne kui loosimise andmeid saab eraldada, on vaja välja selgitada, mitmendal leheküljel vastava loosimise tulemus asub. Eesti Loto koduleheküljel on iga mängutüübi tulemuste jaoks eraldi URL parameetrid, mille alusel kuvatakse loosimise tulemus. Kuna tulemuste leheküljel näeb korraga ainult viimase seitsmeteistkümne loosimise tulemusi, tuleb kindlaks teha, mitmendalt leheküljelt saab kätte kontrollitava pileti loosimise andmed. Selleks leitakse HTML parsimisel vastava loosimise tüübi viimase loosimise number, mille alusel arvutatakse välja leheküljenumber. Loosimise tulemusi

järjestatakse kodulehel kuupäevade alusel, kus viimase loosimise number asub alati esimesel leheküljel. Leheküljenumbri arvutamiseks leitakse viimase loosimise ja pileti loosimise numbriga vahe, jagatakse see seitsmeteistkümnega ning jagatise tulemusest võetakse täisarv (*Integer* tüüpi), millele liidetakse üks.

Peale loosimise leheküljenumbri väljaselgitamist, saab alustada vastava loosimise andmete eraldamisega. Eraldamiseks vajalik Bingo Loto loosimise HTML informatsioon leitakse näiteks URL-ilt: <https://www.eestiloto.ee/osi/draws.do?pageNumber=1&gameCode=13&gameName=13&sortProperty=DRAWDATE&drawNumber=1059>, kus pageNumber on loosimise leheküljenumber, gameCode ja gameName loosimise tüübi identifikaatorid ning drawNumber loosimise number. Vastavalt on Viking Lotto ja Eurojackpoti identifikaatorid 11 ning 15.

Andmete eraldamisel rakendatakse järgnevat loogikat:

- HTML parsimisel jäetakse vahele tühjad read või read, mis ei sisalda vastava loosimise tüübi numbrite algust defineerivat sõnet. Bingo Loto puhul peab sõne olema " **Võiduklass I – nurgad:** |", Viking Lotto ja Eurojackpoti puhul " **Põhinumbrid:** |".
- Kui leitakse üles numbrite algust sisaldav rida, loetakse mängunumbriks iga järgnevat rida, mille väärtus on integer tüüpi. Igalt järgnevalt realt, mis sisaldab vastava mängutüübi võidusummale viitavat sõnet, näiteks " Bingo jackpot |", eraldatakse vastava võidutasandi võidusumma.
- HTML parsimine lõpetatakse, kui jõutakse loosimise leheküljel lisainfoni. Lisainfot algust määrab sõne " **Lisainfo:** |".

Kui koduleheküljelt saadud loosimise andmed on puudulikud, ei ole võimalik nende alusel piletit kontrollida. Puudulikud andmed saadakse juhul, kui otsitava loosimise tulemus ei ole veel Eesti Loto koduleheküljel üleval. Loosimise mängunumbrite ja võidusummade edukal kättesaamisel, alustatakse pileti kontrollimist.

Järgnev joonis illustreerib koodiosa, mis HTML parsimise teel, saab kätte Bingo Loto loosimise mängunumbrid ning võidutasandite võidusummad. Massiiv 'winParsers'

sisaldab sõnesi, mis viitavad võimalikele võidutasanditele. Kui võidusummat sisaldav rida leitakse üles, eraldatakse sellest vastava võidutasandi võidusumma.

```
if (line.equals("<td><strong>Lisainfo:</strong></td>")) break;
if (ValidationMethods.isInteger(line)) {
    gameNumbers.add(Integer.valueOf(line));
} else {
    for (int i = 0; i < winParsers.length; i++) {
        if (line.equals(winParsers[i])) {
            br.readLine();
            gameWinning[i] = ValidationMethods.extractWinnings(br.readLine());
            break;
        }
    }
}
```

Joonis 8. Eesti Loto koduleheküljelt Bingo Loto loositud numbrite ning võidutasandite võidusummade kättesaamine HTML parsimise teel.

4.6 Pileti kontrollimine

Pileti kontrollimisel kehtivad Bingo Loto, Viking Lotto ja Eurojackpoti mängutüüpidele erinevad mängureeglid, mida piletilt tuvastatud ja vastava loosimise loositud mängunumbrite võrdlemisel, tuleb arvesse võtta. Kui pileti kontrollimine on lõppenud, suunatakse kasutaja järgmisele ekraanivormile, kus kuvatakse vastava loosimise andmed ning iga mänguvälja numbrite kontrollimisel saadud tulemused.

Viking Lotto ja Eurojackpoti puhul, tuleb kontrollida, millised loositud põhinumbrid ning lisanumbrid kattuvad vastava tuvastatud mänguvälja numbritega. Nagu eelnevalt mainitud, sisaldab Viking Lotto piletil iga mänguvälja kuute erinevat numbrit ning iga Eurojackpoti mänguvälja viite erinevat põhinumbrit ja kahte erinevat lisanumbrit. Viking Lottos lähevad mänguväljal olevad numbrid arvesse nii põhi, kui ka lisanumbrina. Viking Lotto kontrollimisel määratakse kindlaks, milline on suurima võidusummaga võidetud mängutasand (näiteks 5+1, 5+0 jne.). Juhul kui Viking Lotto pilet on võitnud jackpoti, kontrollitakse ka loositud õnnenumbrit. Eurojackpoti pileti kontrollimisel rakendatakse sama loogikat, kuid lisanumbreid ja põhinumbreid kontrollitakse eraldi.

Bingo Loto kontrollimisel, eraldatakse mänguvälja numbrite hulgast nurkademängu ja diagonaalide mängu võitmiseks vajalikud numbrid. Iga Bingo Loto mänguvälja sisaldab

25 erinevat numbrit, millest nurkademängu võitmiseks vajalikud numbrid on positsioonidel 1, 5, 21 ja 25 ning diagonaalide mängu võitmiseks vajalikud numbrid positsioonidel 1, 5, 7, 9, 13, 17, 19, 21, 25. Nurkademängu võitmiseks peavad esimese 33. loositud palli hulgast kattuma kõik nurgad. Diagonaalmängu võitmiseks peavad esimese 38. loositud palli hulgast kattuma kõik diagonaalid. Võita on veel võimalik täismängus, kus loosimise numbrite hulgast peavad kattuma kõik mänguväljal olevad numbrid. Kui kõik numbrid kattuvad enne 42. numbrit, on lotomängija võitnud jackpoti. Lõpliku võiduna arvestatakse suurima võidusummaga võidetud mängutasandit.

4.7 Kasutajaliides

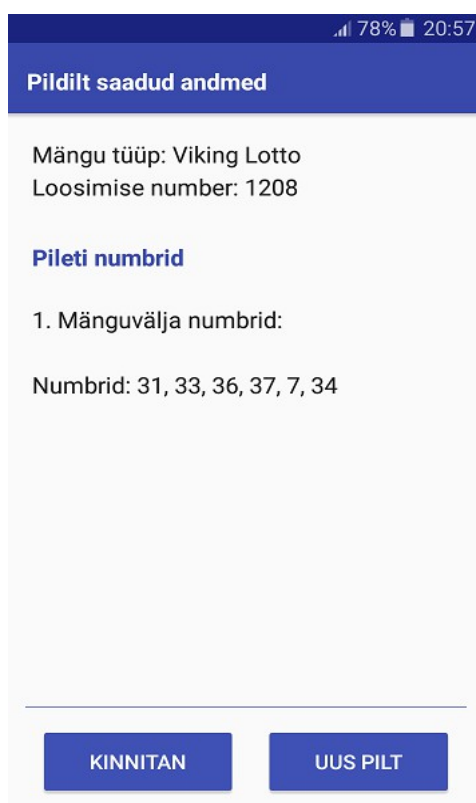
Kasutajaliidese kujundamisel lähtuti Material Design–i mobiilirakenduse struktuurist. Ekraani ülalosas asetseb nõ. tööriistariba, millel kuvatakse kasutajale antud ekraanivormi pealkirja. Pealkiri selgitab vastava ekraanivormi põhitegevust. Ekraani keskel asetseb vastava ekraanivormiga seotud informatsioon, mida vastavalt mahule on võimalik üles–alla rullida. Ekraanivormil oleva informatsiooni liigendamiseks, kasutatakse põhitekstist erinevat värvi olevaid pealkirju ning erineva taustavärviga vorme. Ekraani alumises osas on ekraanivormiga seotud nupud, läbi mille käib kõik antud ekraanivormiga seotud tegevus.

Üks oluline aspekt kasutajaliidese arendamisel on kasutajale veateadete ja informatsiooni kuvamise viis. Kogu rakenduse tekstilise informatsiooni (nuppude peal olev tekst, põhitekst, pealkirjad, veateated jpm.) valikul lähtuti asjaolust, et see oleks kasutajale selgelt mõistetav ning konkreetne. Kõikide ressursinõudlike operatsioonide ajal kuvatakse kasutajale vastav laadimise animatsioon. Veateadete puhul antakse kasutajale teada, mis valesti läks ning mida vea vältimiseks tulevikus teha. Veateadet saab ainult sulgeda veateate all nurgas oleva nupu abil, mis tagab suurema tõenäosuse, et rakenduse kasutaja veateadet loeb.

Rakenduse avaekraanil kuvatakse kasutajale õpetlik informatsioon, mis selgitab kasutajale, milliseid pileteid on võimalik kontrollida ning kuidas seda teha. Avaekraanil asetseb nupp 'TEE PILT', mille vajutamisel avaneb telefoni kaamera operatsioon. Peale

pildi tegemist, antakse laadimise animatsiooni abil kasutajale teada, et toimub antud pildi töötlemine. Kui pildi töötlemine ebaõnnestus, kuvatakse vastav veateade.

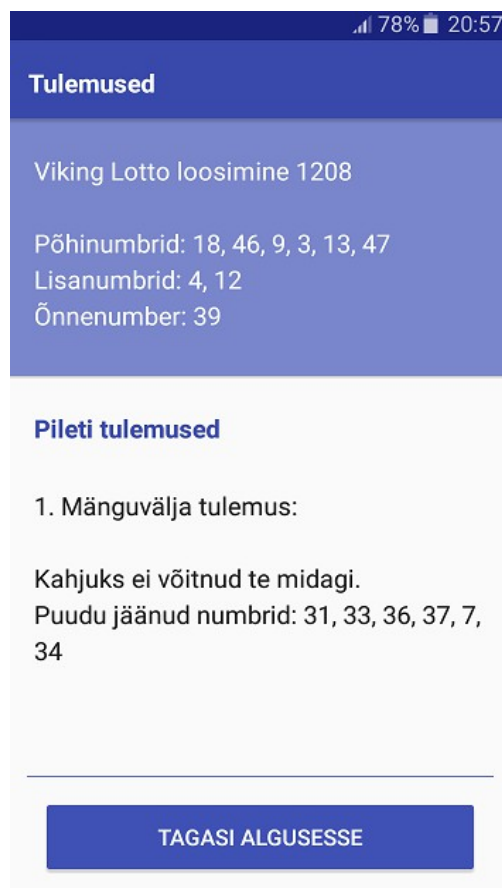
Teisel ekraanivormil kuvatakse kasutajale pildi tuvastamisel saadud andmed. Ekraanivormi ülal asetseb vastava pileti loosimise number ja mängu tüüp, millele järgnevad kõik piletil olevad numbrid mänguväljade kaupa. Allosas asetseb kaks nuppu 'KINNITAN' ning 'UUS PILT', mille vajutamisel kasutaja kas kinnitab kuvatud andmed ning kontrollib piletit või alustab uue pildi tegemist. Kui andmete kinnitamisel ei ole telefonil internetiühendust, kuvatakse kasutajale vastav veateade. Järgnev joonis illustreerib rakenduse teist ekraanivormi.



Joonis 9. Rakenduse ekraanivorm, millel kuvatakse kasutajale pileti tuvastamisel saadud andmed.

Viimasel ekraanivormil kuvatakse kasutajale vastava loosimise loositud pallid ning kasutaja pileti kontrollimisel saadud tulemused mänguväljade kaupa. Juhul kui kasutaja on mõnel mänguväljal võitnud, kuvatakse kasutajale võidu animatsioon. Ekraanivormi

all asetseb nupp 'TAGASI ALGUSESSE', mille vajutamisel suunatakse kasutaja tagasi avaekraanile. Järgnev joonis illustreerib rakenduse viimast ekraanivormi.



Joonis 10. Rakenduse ekraanivorm, millel kuvatakse kasutajale vastava loosimise loositud numbrid ning tema pileti kontrollimisel saadud tulemused.

5 Hinnang rakendusele

Töö käigus valminud rakendus, vastab algselt püstitatud eesmärkidele. Bingo Loto, Viking Lotto ning Eurojackpoti tüüpi mängupiletitelt suudetakse tuvastada vastavaid andmeid ning neid korrektselt kontrollida. Järgnevalt tuuakse välja hinnang rakenduse tähtsamate aspektide kohta:

- Pildi tegemine – pildi tegemine ning sellelt andmete tuvastamine töötab. Rakenduse testimisel jõuti järeldusele, et pildilt andmete tuvastamine on kõige efektiivsem, kui pilt on terav, hästi valgustatud ning kui pildilt on ainult lotopilet. Kui hämarates ruumides kasutada pildistamisel kaamera välgu funktsionaalsust, paranes tuvastamine oluliselt. Tuvastamise peamiseks puuduseks on tuvastamise kiirus, mis võib varieeruda mõnest sekundist mõnekümne sekundini, olenevalt sellest kuidas piletist pilti tehti. Kasutaja seisukohalt võib pikem ootamine tülakaks osutada.
- Pileti kontrollimine – pileti kontrollimine töötab korrektselt ning kiiresti. Rakenduse testimisel ei leitud antud aspekti kohta erilisi puudusi. Kui piletilt õnnestust kätte saada vajalikud andmed, suudeti pilet alati korrektselt ära kontrollida. Kontrollimine võtab aega ligikaudu pool sekundit ning andmemahu kasutatavus ühe kontrollimise kohta on paarkümmend kilobaiti. Üheks kontrollimise puuduseks on selle sõltuvus Eesti Loto lehekülje struktuurist. Kui Eesti Loto peaks oma kodulehte uuendama, tuleb parandada HTML parsimise algoritme.

Rakendust anti samuti kasutada kõrvalistele isikutele, kes polnud rakendusega varem kokku puutunud. Üldmulje rakenduse esmakordsel kasutamisel oli positiivne.

Kasutajate arvates on taolise rakenduse kasutamine palju mugavam ja kiirem viis füüsiliste lotopiletite kontrollimiseks. Kasutajad oleksid ka nõus antud rakendust tulevikus kasutama, eeldusel et rakendus iga kord ja erinevate piletite korral töötab.

Rakenduse ühe puudusena toodi välja pileti kontrollimisel mänguhasardi puudumine, mida tavaliselt pileti märkimisel kogetakse.

Rakenduse testimisel esinenud puuduste parandamiseks, tuleks eelkõige parandada pildilt andmete tuvastamise kiirust, mida võib saavutada näiteks Tesseract-i treenimisel või paremate pilditöötlemis algoritmide rakendamisel. Samuti oleks plaanis muuta kasutajaliidest kasutajasõbralikumaks ning lisada rakendusele uut funktsionaalsust. Üheks uueks funktsionaalsuseks võiks olla näiteks nõ. lotopiletite piltide salv, mis võimaldab kasutajal pildistada korraga kõiki kontrollitavaid lotopileteid ning seejärel neid järjest kontrollida.

6 Kokkuvõte

Bakalaureusetöö peamiseks eesmärgiks oli tekstituvastusega mobiilirakenduse loomine, mis võimaldaks kasutajal kontrollida füüsilisi Eesti Loto pileteid. Piletite kontrollimise jaoks on vajalik korrektne andmete tuvastamine piletist tehtud pildilt ning Eesti Loto koduleheküljelt vastava loosimise andmete kättesaamine.

Bakalaureusetöö tulemusena valminud mobiilirakendus, täidab kõiki eelnevalt püstitatud eesmärgid. Mobiilirakendus suudab pildilt välja lugeda vastava pileti tüübi loosimise numbrid ning erinevate mänguväljade numbrid. Samuti saab rakendus hakkama Eesti Loto leheküljelt kontrollitava pileti loosimise andmete kättesaamisega ning hiljem vastava pileti kontrollimisega. Rõhku pandi ka kasutajaliidese arendamisele, mis baseerub Material Design-i printsiipidel.

Kasutatud kirjandus

- [1] Eesti Loto kodulehekülg [WWW] <https://www.eestiloto.ee/> (14.03.2016)
- [2] Bingo Loto [WWW] <https://www.eestiloto.ee/osi/game/bingo/manual?org.apache.struts.action.TOKEN=null> (14.03.2016)
- [3] Viking Lotto [WWW] <https://www.eestiloto.ee/osi/game/viking/manual?org.apache.struts.action.TOKEN=null> (14.03.2016)
- [4] Eurojackpot [WWW] <https://www.eestiloto.ee/osi/game/eurojackpot/manual?org.apache.struts.action.TOKEN=null> (14.03.2016)
- [5] Eclipse kodulehekülg [WWW] <http://www.eclipse.org/> (19.05.2016)
- [6] Android SDK [WWW] <http://developer.android.com/tools/help/sdk-manager.html> (12.04.2016)
- [7] Android ADT [WWW] <http://developer.android.com/tools/help/adt.html> (12.04.2016)
- [8] Java [WWW] [https://en.wikipedia.org/wiki/Java_\(programming_language\)](https://en.wikipedia.org/wiki/Java_(programming_language)) (14.03.2016)
- [9] Tesseract Android–ile tess-two [WWW] <https://github.com/rmtheis/tess-two> (03.04.2016)
- [10] Material Design [WWW] <https://www.google.com/design/spec/material-design/introduction.html#introduction-principles> (04.05.2016)
- [11] XML Android–i kujundamisel [WWW] <https://developer.android.com/guide/topics/ui/declaring-layout.html> (04.05.2016)
- [12] Optiline kirjamärkide tuvastus [WWW] https://en.wikipedia.org/wiki/Optical_character_recognition (03.04.2016)
- [13] Halltoonides pilt [WWW] <https://en.wikipedia.org/wiki/Grayscale> (03.04.2016)
- [14] Binaarne pilt [WWW] https://en.wikipedia.org/wiki/Binary_image (03.04.2016)
- [15] Tesseract–i kirjeldus [WWW] [https://en.wikipedia.org/wiki/Tesseract_\(software\)](https://en.wikipedia.org/wiki/Tesseract_(software)) (03.04.2016)
- [16] Tesseract Java–le Tess4j [WWW] <http://tess4j.sourceforge.net/> (03.04.2016)
- [17] Android NDK [WWW] <http://developer.android.com/tools/sdk/ndk/index.html> (12.04.2016)
- [18] Android 5.0 API [WWW] <https://developer.android.com/about/versions/android-5.0.html> (12.04.2016)

- [19] UTF-8 [WWW] http://www.w3schools.com/charsets/ref_html_utf8.asp
(21.05.2016)
- [20] Regulaaravaldiste definitsioon [WWW]
https://en.wikipedia.org/wiki/Regular_expression (15.05.2016)
- [21] Stackoverflow [WWW] <http://stackoverflow.com/> (14.03.2016)
- [22] Online regulaaravaldiste testimine [WWW] <https://regex101.com/> (15.05.2016)
- [23] OpenCv [WWW] <http://opencv.org/> (03.04.2016)