

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond  
Arvutiteaduse instituut  
Võrgutarkvara õppetool

ITI40LT  
Rivo Heinsalu 120445

**RESTORANIDE TELLIMUSTE ESITAMISE  
MOODUL KASUTADES NFC  
TEHNOLOOGIAT**

Bakalaureusetöö

Juhendaja: Roger Kerse  
Mobiilirakenduste  
lektor

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: Rivo Heinsalu

09.05.2016

## **Annotatsioon**

Käesoleva bakalaureusetöö eesmärgiks on luua jätkuva iseteeninduse teenus toitlustusasutustele kahe omavahel suhtleva Android hübriidrakendusena. Kliendirakendus, mis võimaldaks kliendil oma asukohta tuvastada NFC tehnoloogia abil ning esitada tellimusi ja klienditeeninduse rakendus, mis võimaldaks kliendi tellimusi vastu võtta ja täita.

Autor käsitleb töös iseteenindusega seotud probleeme toitlustusasutustes. Lisaks uurib autor juba olemasolevaid iseteeninduse lahendusi Eestis, erinevaid tehnoloogiaid kliendi asukoha tuvastamiseks siseruumis, erinevaid rakenduse loomisvõimalusi Android platvormile ja erinevaid tehnoloogiaid iseteeninduse rakenduse loomiseks.

Töö tulemusena valmis jätkuva iseteeninduse teenus kahe Android hübriidrakendusena. Teenuse loomisel arvestas autor olemasolevate lahenduste puudujääkidega ning parimate omadustega.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 38 leheküljel, 4 peatükki, 12 joonist, 2 tabelit.

## **Abstract**

### **Ordering module for restaurants using nfc technologies**

The purpose of this thesis is to create a continuous self-service for catering establishments by creating two mutually communicating Android hybrid applications. Client application that allows the client to identify their location on NFC technology and allows to place orders and customer service application that help client orders to be adopted and implemented.

The author deals with the problems associated with the work of self-catering facilities. In addition, the author examines the existing solutions of self-service in Estonia, a variety of technologies to detect the location of the customer's internal space, a variety of applications that can be created on Android platform and a variety of technologies to create a self-service application.

The result is a continuous self-service between two Android hybrid applications. While creating the service, the existing shortcomings of solutions and the best properties were taken to account.

The thesis is in Estonian and contains 38 pages of text, 4 chapters, 12 figures, 2 tables.

## Lühendite ja mõistete sõnastik

API	<p><i>Application Programming Interface</i></p> <p>Rakendusliides on reeglistik olemasoleva valmisprogrammiga suhtlemiseks. [1]</p>
Jätkuv iseteenindus	<p>Iseteeninduse vorm, mille puhul saab tellija toote kätte enne maksmist ning tasub tarbimise lõppedes kõigi tellitud toodete eest korraga.</p>
NFC	<p><i>Near field communication</i></p> <p>Kontaktivaba lähiväljatehnoloogia on andmeside protokoll, mis võimaldab 2-4 cm kaugusel asuvate seadmete omavahelise andmevahetuse üle raadioside. [2]</p>
NFC Tag	<p>Märgised, mis koosnevad väikestest mikrokiipidest ja on võimelised talletama väikesemahulist infot, mida on võimalik lugeda ja kirjutada NFC tehnoloogia abil. [3]</p>
Ostukorvi iseteenindus	<p>Iseteeninduse vorm, mille puhul saab tellija toote kätte pärast tellimuse eest tasumist.</p>
QR kood	<p><i>Quick Response Code</i></p> <p>Ruutkood on kodeeritud kahemõõtmeline maatrikskood, mis võimaldab skaneerida infot kasutaja seadmesse, kus see dekodeeritakse. [4]</p>
SDK	<p><i>Software development kit</i></p> <p>Tarkvaraarendust lihtsustavate tööriistade kogum. [5]</p>

## Sisukord

Sissejuhatus .....	10
1 Toitlustusasutuse iseteeninduse loomise võimalused.....	11
1.1 Taust ja probleem .....	11
1.2 Olemasolevad lahendused .....	12
1.2.1 McDonalds Eesti .....	12
1.2.2 Mustamäe Apollo Kino .....	12
2 Teenuse loomine.....	14
2.1 NFC tehnoloogia kasutamise võimalused .....	14
2.1.1 NFC tehnoloogia kasutamise võimalused jätkuva iseteeninduse loomisel ...	15
2.1.2 Alternatiivid NFC tehnoloogiale .....	15
2.2 Android.....	15
2.2.1 Android rakenduse loomise võimalused .....	16
2.3 Teenusele esitatud nõuded.....	17
2.4 Tehnoloogiate ning raamistike valik .....	19
2.4.1 Ionic.....	19
2.4.2 Firebase.....	20
2.4.3 Phonegap NFC.....	21
3 Valminud teenus .....	22
3.1 Kasutatud vahendid .....	22
3.2 Teenindaja rakendus .....	23
3.2.1 Restorani detailinfo administreerimine .....	23
3.2.2 Restorani laudade administreerimine .....	24
3.2.3 Restorani menüü administreerimine.....	25
3.2.4 Restoranide laudade vaade .....	26
3.2.5 Aktiivse laua tellimuste vaade.....	27
3.3 Kliendirakendus.....	28
3.3.1 Restorani menüü kuvamine .....	28
3.3.2 Tellimuse esitamine.....	29
3.3.3 Tellimuste vaatamine.....	30
3.4 Valminud teenuse vastavus esitatud nõuetele .....	31

3.4.1 Teenindaja rakenduse vastavus nõuetele.....	31
3.4.2 Kliendi rakenduse vastavus nõuetele.....	33
3.5 Eelised võrreldes olemasolevate lahendustega.....	34
3.6 Puudujäägid ja edasiarendamise võimalused .....	34
4 Kokkuvõte .....	35
Summary.....	36
Kasutatud kirjandus .....	37

## Jooniste loetelu

Joonis 1. Androidi versioonide jaotus seadmetel [14].....	16
Joonis 2. Git Bashi käsurealt Ionic hübriidrakenduse käivitamine käsuga "Ionic serve" .....	19
Joonis 3. Firebase reaalaraja andmebaasiga ühenduse loomine .....	20
Joonis 4. NFC märgiselt info lugemine .....	21
Joonis 5. Restorani detailinfo administreerimise ekraanivaade.....	23
Joonis 6. Restorani laudade administreerimise ekraanivaade .....	24
Joonis 7. Restorani menüü administreerimise ekraanivaade .....	25
Joonis 8. Restorani laudade ekraanivaade .....	26
Joonis 9. Tellimuste ekraanivaade .....	27
Joonis 10. Restorani menüü ekraanivaade.....	28
Joonis 11. Tellimuse esitamise ekraanivaade .....	29
Joonis 12. Tellimuste ekraanivaade .....	30



## **Tabelite loetelu**

Tabel 1. Teenindaja rakenduse vastavus nõuetele.....	31
Tabel 2. Kliendi rakenduse vastavus nõuetele.....	33

## Sissejuhatus

Iseteeninduse võimaldamine toitlustusasutustes on ettevõtetele suureks väljaminekuks, mida suudavad katta vaid suuremad ketid. Suurtele frantsiisidele on tihti omane ka „võta kaasa“ tüüpi teenuse pakkumine ning neile sobivaks iseteeninduse vormiks on odavam ostukorvi lahendus. Ostukorvi tüüpi iseteenindus ei sobi aga oma olemuselt tavalistesse toitlustusasutustesse. Tänapäevased võimalused jätkuva iseteeninduse pakumiseks nõuavad seda tüüpi ettevõtelt suuri väljaminekuid arenduseks ning igale lauale statsionaarsete seadmete paigaldamiseks.

Bakalaureusetöö eesmärgiks on luua jätkuva iseteeninduse kontseptsioon kahe omavahel suhtleva mobiilirakendusena. Esimene rakendus on toitlustusasutusele tellimuste vastuvõtmiseks ning teine rakendus mõeldud kliendile tellimuste esitamiseks. Selleks, et tuvastada kliendi asukoht toitlustusasutuses ja võimaldada jätkuv iseteenindus, võtab autor kasutusele NFC tehnoloogia. Loodav teenus peaks tegema iseteeninduse kättesaadavaks restoranidele, kellel puudub võimekus teha omapoolsed arendusi ja suuri kulutusi seadmetele.

Töö teoreetilises osas analüüsib autor juba olemasolevaid iseteeninduse rakendusi Eesti toitlustusasutustes.

Bakalaureusetöö praktilises osas uurib autor NFC tehnoloogia kasutusvõimalusi ning alternatiive NFC kasutamisele. Lisaks analüüsib autor erinevaid Android rakenduste loomise viise ning vahendeid.

Viimases peatükis „Valminud rakendus“ analüüsib autor loodud mobiilirakendusi ja nende vastavust esitatud nõuetele. Lisaks analüüsib autor loodud rakenduste puudujääke ja edasiarenduse võimalusi.

# 1 Toitlustusasutuse iseteeninduse loomise võimalused

Antud peatükis analüüsib autor iseteeninduse loomisega seotud probleeme toitlustusasutustes ning uurib populaarseid iseteeninduse lahendusi Eestis.

## 1.1 Taust ja probleem

Iseteeninduse süsteemid toimivad enamasti ostukorvi põhimõttel – tellimust asutakse täitma pärast tellimuse eest tasumist. Selline loogika ei sobi aga kokku toitlustusasutustega, kus tellimine ja kauba kätte saamine toimub terve külastusaja vältel ja tasumine tellimuste eest külastuse lõpus.

Jätkuva iseteeninduse loomisel on suurimaks probleemiks kliendi asukohta tuvastamine. Kuhu tellimus toimetada ning kes selle eest tasub. Probleemi üheks lahenduseks on kõigi laudade varustamine ettevõtte enda poolt määratud seadmete ja rakendustega, kuid selliseid investeeringuid suudavad teha vaid suuremad ja võimekamad ettevõtted. Väikeettevõtjate jaoks on aga oluline püsida suurettevõtjatega konkurentsisis ning otsida samuti uudseid lahendusi teeninduse kiiruse ja mugavuse tõstmiseks.

Käesoleva bakalaureusetöö raames valmib jätkuva iseteeninduse teenus toitlustusasutustele kahe omavahel suhtleva Android hübriidrakendusena. Kliendirakendus, mis võimaldaks kliendil oma asukohta tuvastada NFC tehnoloogia abil ning esitada tellimusi ja klienditeeninduse rakendus, mis võimaldaks kliendi tellimusi vastu võtta ja täita. Hübriidrakendus valmib Android platvormil, sest Android on kõige levinum mobiilne platvorm. [6]

## **1.2 Olemasolevad lahendused**

Järgnevatel peatükkides toob autor välja mõned populaarsemad iseteeninduse lahendused Eestis. Nendega tutvumiseks ja analüüsimiseks külastas autor kahte asutust Eestis. Esimesena Rocca al Mare McDonalds kiirtoidurestorani ja teisena Apollo kino Mustamäe Keskuses. Apollo kino iseteeninduse tarbimiseks on vajalik „Staaritooli“ kinopileti soetamine.

### **1.2.1 McDonalds Eesti**

McDonalds on iseteeninduseks loonud puutetundliku ekraani ja makseaparaadiga varustatud kioskid. Tellimuse alustamiseks peab klient valima, kas tellib kaasa või sööb koha peal. Lisavõimalusena on samal vaatel võimalik vahetada kuvatava info keelt. Seejärel kuvatakse kogu menüü ning kliendil on võimalik lisada soovitud tooted ostukorvi. Tellimuse kinnitamiseks valib klient „Lõpeta tellimus“, mis suunab kliendi makse lehele. Tasuda on võimalik vaid pangakaardiga sama kioski küljes olevas seadmes. Kui tellimuse eest on tasutud, väljastab kiosk kliendile järjekorranumbri. Kui kliendi tellimus on valmis, kuvatakse tema järjekorranumber teenindusleti kohal oleval ekraanil ja klient võib oma tellimusele järele minna.

Rakenduse eelisteks on võimalus valida keelt ja tasuda tellimuse eest ning mugav menüü sirvimine.

Rakenduse puuduseks on autori hinnangul ostukorvi meetod.

### **1.2.2 Mustamäe Apollo Kino**

Mustamäe Apollo Kino saalide viimasel real asuvaid istmeid nimetatakse „Staaritoolideks“. Istmed asetsevad paaris ning keskmise käetoega sisse on ehitatud telefoni suurune puutetundlik ekraan, mille kaudu on kinokülastajatele võimaldatud söökide ja jookide tellimine. Tellimise alustamiseks valib kasutaja teda huvitava kategooria, näiteks „jook“. Seejärel avaneb kasutajale vastava kategooria menüü. Tellimuse esitamiseks peab kasutaja välja valitud toote lisama ostukorvi ning kui kõik soovitud tooted on valitud, siis navigeerima ostukorvile. Tellimuse kinnitamiseks valib kasutaja „Vormista tellimus“ ning kasutajat teavitatakse tellimuse vastuvõtmisest. Seejärel toimetab kino klienditeenindaja tellimused kinokülastajani. Lisaks tellitud

toodetele antakse külastajale ka vahearve, mis tuleb kliendil tasuda filmiseansi lõppedes kino kassas.

Rakenduse eeliseks on autori hinnangul jätkuv iseteenindus. Pärast esimese tellimuse saatmist on kasutajal võimalik tellida juurde nii mitu korda, kui soovitakse ja tasuda kõigi tellimuste eest filmiseansi lõppedes korraga.

Rakenduse puudusteks hindab autor maksevõimaluse ja mitmekeelsuse puudumist.

## 2 Teenuse loomine

Antud peatükis kirjeldab autor NFC tehnoloogiat ning selle kasutamise võimalusi ja Android platvormi ning sellele rakenduste loomise võimalusi. Lisaks kirjeldab autor teenusele esitatavad nõuded ning analüüsib erinevaid tehnoloogiaid, mida autor jätkuva iseteeninduse teenuse loomisel kasutab.

### 2.1 NFC tehnoloogia kasutamise võimalused

Near Field Communication (NFC) on kontaktivaba lähiväljatehnoloogia, mis võimaldab väikesemahuliste andmete vahetust kahe NFC toega seadme või seadme ja NFC märgise vahel. [7]

NFC toega Android seadmed võimaldavad tehnoloogiat kasutada kolmel viisil:

- Lugeja/kirjutaja režiim, mis võimaldab seadmel lugeda ja/või kirjutada andmeid NFC märgistele.
- P2P režiim, mis võimaldab omavahelise andmevahetuse kahe seadme vahel.
- Kaardi jäljendamise režiim, mis lubab seadmel käituda kui NFC kaart ja lubada enda andmete lugemist teistel NFC toega seadmetel. [7]

Lugeja/kirjutaja režiimi kasutamine on tavakasutajate seas kõige levinum. NFC märgisele on võimalik Google Playst saadavate rakenduste (näiteks „NFC tools“) abil kirjutada toiminguid, mida telefon peaks sooritama märgise lugemisel. [8] Näiteks on võimalik kirjutada NFC märgisele käsk lülitada sisse seadme sinihammas ja paigutada see märgis autosse. Nüüd iga kord kui seade seda kleebist puudutab, aktiveerub seadmes sinihamba kasutamine.

P2P režiim on näiteks kasutusel Android Beamis. Android Beam on Android platvormi rakendus, mille abil on võimalik saata teistele seadmetele kontakte, pilte, linke ja palju muud. [9]

Kaardi jäljendamise režiim on populaarsust kogumas pankade ja erinevate kliendikaartide tootjate seas. Näiteks võimaldab rakendus PayTailor aktiveerida poes oma seadmega mobiilimakse ja kinnitada makse isiklikus seadmes PIN-koodiga. [10]

### **2.1.1 NFC tehnoloogia kasutamise võimalused jätkuva iseteeninduse loomisel**

Autori loodavas kliendirakenduses on NFC tehnoloogia abil võimalik tuvastada kliendi asukoht restoranis. Selleks on vajalik kirjutada NFC märgis, mis sisaldaks endas unikaalset restorani koodi ja unikaalset laua koodi. Asutuse koodi järgi on võimalik kuvada õige restorani menüü ning teha tellimusi ja laua koodi abil on klienditeenindajal võimalik toimetada tellimused õigesse kohta.

### **2.1.2 Alternatiivid NFC tehnoloogiale**

NFC tehnoloogiat toetavad väga paljud uuemad telefonid, kuid on ka seadmeid, mis seda ei toeta või mille API-d ei ole vabalt kasutatavad. [11] Näiteks iPhone 6 on esimene iOS operatsioonisüsteemiga telefon, millel on küll NFC tugi, kuid mis lubab kasutada NFC API-sid vaid Apple enda makseteks. Seetõttu tuleb otsida ka alternatiive selliste seadmete omanikele. [12]

Üheks võimalikuks alternatiiviks on QR koodi lugemine. Ka QR koodile on võimalik genereerida unikaalsed restorani ja laua koodid, mille abil on võimalik kasutada jätkuvat iseteenindust. QR koodi puuduseks on koodi lugemiseks vajaliku lisatarkvara lisamine rakendusse ja lisategevusena kaameraga koodi lugemine.

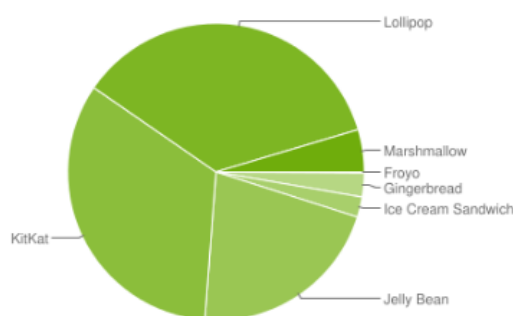
## **2.2 Android**

Android on Google poolt arendatav vabavaraline Linuxil põhinev operatsioonisüsteem, mida arendatakse eelkõige puutetundlikele seadmetele nagu mobiilid ja tahvelarvutid, kuid ka telekatele, nutikelladele ja paljudele teistele mobiilsetele seadmetele. [13]

Androidi hinnanguline turuosa on 82,8% ja see teeb Androidist enimlevinud mobiilse platvormi [6]. See on autori tingimuseks platvormi valikul – katta võimalikult suur turuosa seadmetest.

Androidi versioonid on pidevalt uuenevad ja kuna vanemad versioonid ei toeta uuematele versioonidele arendatud rakendusi, siis tuleb valida ka vanim versioon, millele on mõistlik rakendust toetada. Teenuse loomise ajal oli kõige uuemaks versiooniks 6.0 nimega „Marshmallow“ API numbriga 23. Erinevate Androidi versioonidega seadmete turuosad on välja toodud joonisel (vt Joonis 1).

Version	Codename	API	Distribution
2.2	Froyo	8	0.1%
2.3.3 - 2.3.7	Gingerbread	10	2.6%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	2.2%
4.1.x	Jelly Bean	16	7.8%
4.2.x		17	10.5%
4.3		18	3.0%
4.4	KitKat	19	33.4%
5.0	Lollipop	21	16.4%
5.1		22	19.4%
6.0	Marshmallow	23	4.6%



### Joonis 1. Androidi versioonide jaotus seadmetel [14]

Autori jaoks on oluline katta valdav turuosa Android seadmetest. Seetõttu on autor valinud viimaseks toetatavaks Android versiooniks 4.4 nimega „Kitkat“ ja API numbriga 19. Nii valides katab autor peaaegu 75% Android operatsioonisüsteemiga seadmetest.

#### 2.2.1 Android rakenduse loomise võimalused

Android rakenduse loomiseks on kolm erinevat võimalust.

- Põlirakendus
- Mobiilne veebirakendus
- Hübriidrakendus

Põlirakendus on platvormi spetsiifikat arvestades arendatud rakendus. Android platvormile põlirakenduse kirjutamine eeldab Java programmeerimiskeele valdamist. Põlirakenduse arendamise puhul on võimalik ära kasutada kõiki seadme funktsioone. Näiteks kasutada kaamerat, küsida kontakte või seadme asukohta ja palju muud. [15] Lisaks sellele on võimalik Android põlirakendusi üles ja alla laadida Google Play rakenduste keskkonnas. [16] Põhiliseks puuduseks on platvormi spetsiifika. Näiteks selleks, et sama rakendus saada tööle iOS operatsioonisüsteemil, tuleb alustada rakenduse kirjutamist iOSi spetsiifikat arvestades. Arendades põlirakendusi, tuleb ühe



ja sama rakenduse loomiseks mitmele platvormile, teha sama töö algusest lõpuni uuesti. Põlirakenduse loomine ei ole autori arvates sobiv valik, nii tuleviku vaates kui ka Java programmeerimiskeele vähese oskuse tõttu.

Mobiilne veebirakendus on veebistandardi nõuetele vastavalt kirjutatud rakendus, mis on avatav veebilehitsejaga etteantud veebiaadressil ning mille kirjutamiseks on vajalik vähemalt HTML, CSS ja Javascript tehnoloogiate tundmine. Veebirakenduse puhul ei ole võimalik ära kasutada seadme tehnilisi võimalusi. Näiteks kasutada kaamerat või küsida koordinaate. [15] Veebirakenduse loomine ei ole autori loodavate rakenduste puhul võimalik, sest jätkuva iseteeninduse loomiseks on vajalik telefoni NFC API kasutamine.

Hübriidrakendus on põlirakenduse konteinerisse installeeritud veebirakendus, mis on suuteline töötama täisekraanil kasutades mootorina ära seadme veebilehitsejat. Erinevalt tavalisest veebirakendusest on hübriidrakenduse puhul võimalik kasutada seadme funktsionaalsusi. Näiteks on võimalik kasutada kaamerat ja lugeda NFC APIst saadud infot. [17] Hübriidrakenduse puuduseks on sõltuvus seadme veebilehitsejast ja mõnes brauseris võib esineda vigu või mittesoovitud käitumisi nagu mõne teeki sisseehitatud funktsionaalsuse mittetöötamine.

Kuna töö autor on veebiarenduse taustaga ning omab peamiselt kokkupuudet Javascripti põhiste tehnoloogiatega, siis on autori valikuks Android rakenduse loomine hübriidrakendusena. Lisaks eelnevale on hübriidrakenduse puhul teistele platvormidele arendamine oluliselt kiirem kui põlirakenduse loomine ja tihti piirdub vaid väikeste modifikatsioonidega juba olemasolevas koodibaasis.

## **2.3 Teenusele esitatud nõuded**

Klienditeenindaja rakendusele esitatud nõudmised on järgmised:

- Teenuse kasutamiseks peab registreeruma kasutajaks.
- Registreerumine peab sisaldama järgmist infot:
  - Restorani nimi.
  - E-posti aadress.
  - Parool.
- Sisselogimiseks peab kasutaja sisestama e-posti ja parooli.

- Kasutaja peab saama muuta restorani detailinfo.
- Restorani detailinfo peab sisaldama järgmist infot:
  - Restorani nimi.
  - Restorani aadress.
- Kasutaja peab nägema olemasolevaid laudu.
- Kasutaja peab saama lisada uusi laudu.
- Laua info peab sisaldama järgmist infot.
  - Laua kuvamise järjekorranumber.
- Kasutaja peab saama muuta laudade järjekorda.
- Kasutaja peab saama laudu kustutada.
- Kasutaja peab nägema menüü kategooriaid.
- Kasutaja peab saama lisada menüü kategooriaid.
- Menüü kategooria peab sisaldama järgmist infot:
  - Kategooria nimetus.
  - Kategooria kuvamise järjekorranumber.
- Kasutaja peab saama muuta menüü kategooriaid.
- Kasutaja peab saama kustutada menüü kategooriaid.
- Kasutaja peab nägema kategooria menüüd.
- Kasutaja peab saama lisada kategooria menüüd.
- Kategooria menüü peab sisaldama järgmist infot:
  - Toote nimetus.
  - Toote info.
  - Hind.
  - Toote kuvamise järjekorranumber.
- Kasutaja peab saama muuta kategooria menüüd.
- Kasutaja peab saama kustutada kategooria menüüd.
- Kasutaja peab nägema aktiivseid laudu.
- Kasutaja peab nägema laua tellimusi.
- Kasutaja peab saama märkida aktiivse sessiooni lõppenuks.

Kliendi rakendusele esitatud nõudmised on järgmised:

- Kasutajal peab olema võimalik käivitada rakendus NFC märgise lugemisel.
- Kasutaja peab nägema restorani menüüd.

- Kasutaja peab saama esitada tellimusi.
- Kasutaja peab nägema kõiki laua tellimusi.
- Kasutaja peab nägema enda esitatud tellimusi.
- Kasutaja peab saama tellimuste eest maksta.

## 2.4 Tehnoloogiate ning raamistike valik

Antud peatükis annab autor ülevaate, milliseid tehnoloogiaid ning raamistikke autor kasutab jätkuva klienditeeninduse Android hübriidrakenduse loomisel.

### 2.4.1 Ionic

Ionic on Drifty Co. poolt 2013. aastal loodud avatud lähtekoodiga tarkvaraarenduse raamistik hübriidrakenduste loomiseks. Ionic on ehitatud AngularJS ja Apache Cordova põhjale ja pakub erinevaid tööriistu ja teenuseid hübriidrakenduste arendamiseks. [18]

Bakalaureusetöoga valmivas teenuses kasutab autor Ionicut hübriidrakenduse raamistikuna, sest autori sooviks on alustada kohehelt funktsionaalsuse kirjeldamisega ilma, et peaks ehitama ülesse isikliku projekti ehitusprotsessi. Ionicut kasutades on võimalik käsurealt käivitada projekti ehitusprotsess ja suunata rakendus veebilehitseja aknasse (vt Joonis 2).

```
Rivo@RI00 /C/Projects/nfcdiner <master>
$ ionic serve

Running 'serve:before' gulp task before serve
[11:58:22] Starting 'clean'...
[11:58:22] Finished 'clean' after 59 ms
[11:58:22] Starting 'watch'...
[11:58:22] Starting 'sass'...
[11:58:22] Starting 'html'...
[11:58:22] Starting 'fonts'...
[11:58:22] Starting 'scripts'...
[11:58:22] Finished 'scripts' after 220 ms
[11:58:22] Finished 'html' after 241 ms
[11:58:23] Finished 'fonts' after 347 ms
[11:58:25] Finished 'sass' after 3.16 s
9.3 MB bytes written (32.03 seconds)
[11:58:58] Finished 'watch' after 36 s
[11:58:58] Starting 'serve:before'...
[11:58:58] Finished 'serve:before' after 22 us

Running live reload server: http://localhost:35729
Watching: www/**/*, !www/lib/**/*
? Running dev server: http://localhost:8100
Ionic server commands, enter:
  restart or r to restart the client app from the root
  goto or g and a url to have the app navigate to the given url
  consolelogs or c to enable/disable console log output
  serverlogs or s to enable/disable server log output
  quit or q to shutdown the server and exit

ionic $
```

Joonis 2. Git Bashi käsurealt Ionic hübriidrakenduse käivitamine käsuga "Ionic serve"

Lisaks eelnevale sai autori jaoks raamistiku valimisel oluliseks eelnev kokkupuude Ionic 1 raamistikuga. Autori sooviks on valmivas töös kasutada hetkel arendatavat Ionic 2 beetaversiooni, et arendada oma teadmisi.

Autori hinnangul kiirendas Ionicu kasutamine oluliselt arendusprotsessi kiirust ja mugavust. Puudusena toob autor välja beetaversioonis tekkivad vead nagu teekide mitteühildumine, kuid nende tekkimisega oli autor juba varasemalt arvestanud.

## 2.4.2 Firebase

Firebase on pilve- ja taustateenuste pakkuja, kelle peamiseks tooteks on reaalaaja andmebaaside pakkumine arendajatele. Teenus lubab arendajatel kasutada nendepoolseid API-sid, mille abil on võimalik luua reaalaaja andmeid kuvavaid rakendusi. Kasutaja andmeid hoitakse Firebase andmebaasis JSON formaadis ja sünkroniseeritakse kõigi rakenduse kasutajatega reaalajas läbi Firebase poolse turvalise ühenduse kihi. [19,20] Firebase baasteenuse kasutamine on tasuta. Baasteenusega liitunud rakendusele võimaldatakse kuni 100 üheaegset ühendust tasuta. [21]

Antud töös kasutab autor Firebase andmebaasi reaalaaja andmete kuvamise võimekuse tõttu. Firebase API abil on lihtne luua kanal reaalajas andmete lugemiseks (vt Joonis 3).

```
getTableSessionOrders(sessionKey) {  
  return this.sessionsRef.child(this.restaurantService._restaurantDetails.restaurantKey).child('sessions')  
    .child(sessionKey).child('orders').child('orderItems').on('value', snap => {  
    if (!!snap.val()) {  
      this._orderedItems = _.orderBy(snap.val(), ['startedAt'], ['desc']);  
    } else {  
      this._orderedItems = {};  
    }  
    this.events.publish('orders:gotOrderedItems');  
  });  
}
```

### Joonis 3. Firebase reaalaaja andmebaasiga ühenduse loomine

Joonisel välja toodud koodis loob autor Firebase andmebaasiga ühenduse ja lisab kuulaja soovitud JSON tasemele „on('value')“ API-ga. Iga järgnev muudatus, mis kuulataval tasemel toimub, jõuab koheselt ka kasutaja rakendusesse.

Teine võimalus reaalajas andmete kuvamiseks oleks autori hinnangul WebSocketite abil kliendi ja serveri vahelise sessiooni loomine. [22] See aga nõuaks autorilt serveri kirjeldamist ja oluliselt töömahu suurenemist.

Autori hinnangul on Firebase andmebaas väga kiire ja hästi dokumenteeritud, seetõttu sujus arendus ja uuenduste sisseviimine väga kiiresti.

### 2.4.3 Phoneyap NFC

Phoneyap NFC on kolmanda osapoole loodud teek NFC tehnoloogia kasutamiseks hübriidrakendustes. Antud teeki kasutades on võimalik seadmes kasutada kõiki peatükis 2.1 „NFC tehnoloogia kasutamise võimalused“ välja toodud toimingute režiime. [23]

Käesoleva bakalaureusetöö aluseks on NFC võimekuse kasutamine ja selle abil kasutaja asukoha tuvastamine toitlustusasutuses. Phoneyap NFC on autorile ainus teadaolev töötav teek NFC tehnoloogia kasutamiseks hübriidrakendustes. Autori loodud NFC märgise lugemise tuvastamiseks rakenduses, kutsus autor välja teegis kirjeldatud funktsiooni „addMimeTypeListener“ (vt Joonis 4).

```
addNfcListener() {
  nfc.addMimeTypeListener("application/nfcdiner", (tagData) => {
    return this.nfcService.parseNfcData(tagData);
  }, () => {
    // console.log("NFC waiting for tag");
  }, err => {
    if (err == 'NFC_DISABLED') {
      this.nfcEnabled = false;
      this.events.publish('device:nfcDisabled');
    }
  });
}
```

**Joonis 4. NFC märgiselt info lugemine**

Joonisel kirjeldatud „application/nfcdiner“ NFC sõnumi tüüp on autori loodud tüüp tuvastamiseks ainult rakendusele omaseid NFC märgiseid.

Autori hinnangul on puuduseks alternatiivide puudumine ja teegi kuulumine ühele isikule, kes ei pruugi probleemide tekkimisel olla võimeline teeki piisava kiirusega uuendama.

### 3 Valminud teenus

Autori sooviks oli lahendada jätkuva iseteeninduse suurim väljakutse kliendi asukoha tuvastamisel toitlustusasutuses. Lisaks soovis autor ühtlasi pakkuda odavamalt teenust ettevõtjatele, kes soovivad ajaga kaasas käia, kuid kellel pole vahendeid isikliku iseteeninduse loomiseks ja anda kasutajatele võimalus kasutada isiklike seadmeid. Antud peatükis toob autor välja teenuse loomiseks kasutatud vahendid, tähtsama funktsionaalsuse, valminud teenuse vastavuse nõuetele ja edasiarenduse võimalused.

Käesoleva bakalaureusetöö käigus valminud rakendus ja kood ei kuulu avaldamisele ning ei ole töö esitamise hetkel allalaetav Google rakenduste poest [16].

#### 3.1 Kasutatud vahendid

Autor kasutas jätkuva iseteeninduse loomiseks järgnevaid vahendeid:

- JetBrains Webstorm 2016.1 – arenduskeskkond [24]
- Git Bash – käsurea tööriist, mis on vajalik kolmandate osapoolte teekide installimiseks ja Ionicu käskude käivitamiseks [25]
- Android SDK Tools – Androidi tööriistad, mis on vajalikud Apache Cordova toimimiseks [26]

Loodud rakenduse testimiseks kasutas autor järgnevaid vahendeid:

- Sony Xperia Z1 compact, Android versiooniga 5.1.1 Lollipop
- Galaxy Tab 2 10.1, Android versiooniga 5.1.1 Lollipop
- Google Chrome versiooniga 50.0.2661.87 m [27]
- NFC märgis – programmeeritav NFC märgis [28]

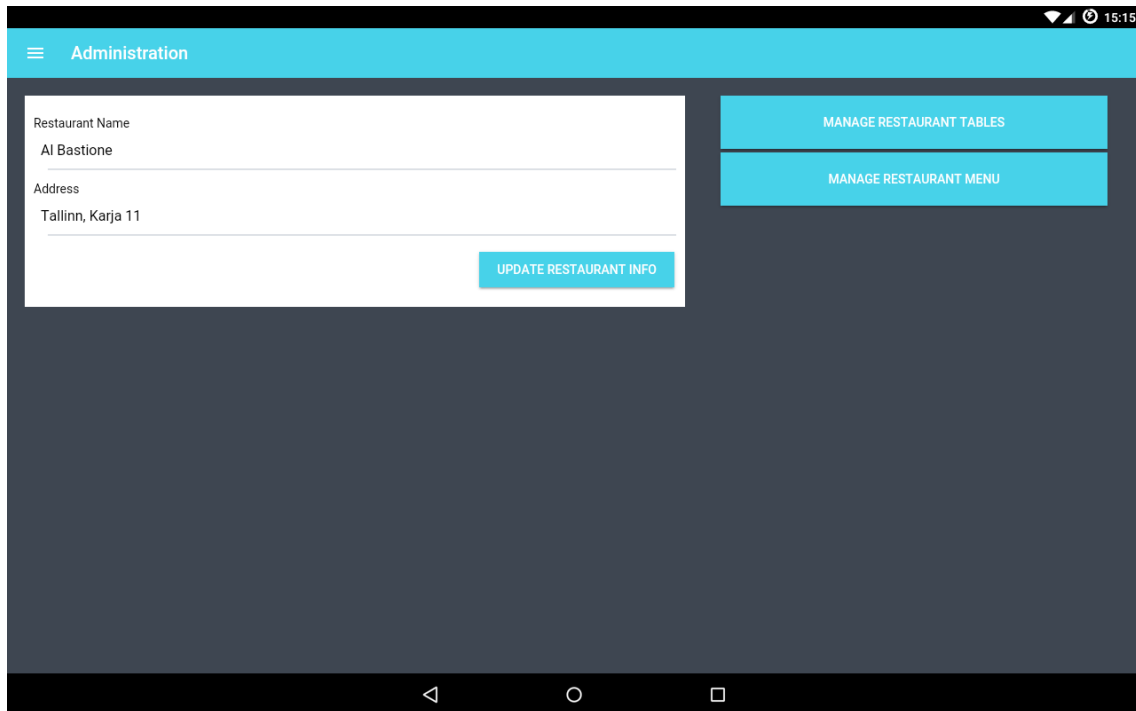
Loodud rakenduse programmeerimiseks kasutas autor põhiliselt järgmisi programmeerimiskeeli:

- Javascript ES6 standard
- HTML5
- CSS3

## 3.2 Teenindaja rakendus

### 3.2.1 Restorani detailinfo administreerimine

Administreerimise pealehel on kasutajal võimalik muuta restorani detailinfot ja navigeerida laudade ja menüü administreerimise lehele (vt Joonis 5).



Joonis 5. Restorani detailinfo administreerimise ekraanivaade

### 3.2.2 Restorani laudade administreerimine

Laudade administreerimise lehel kuvatakse kasutajale list olemasolevatest laudadest. Lisaks varem sisestatud laudadele on antud vaates võimalik lisada uusi laudu ning muuta ja kustutada juba olemasolevaid laudu (vt Joonis 6).

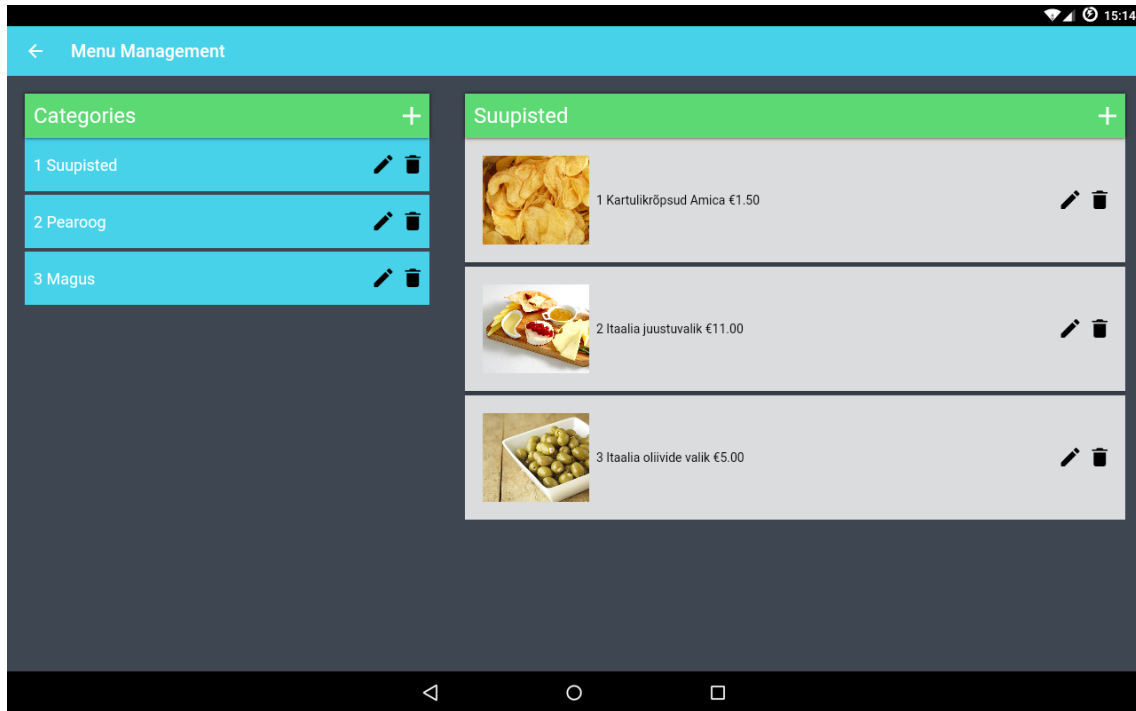


Joonis 6. Restorani laudade administreerimise ekraanivaade



### 3.2.3 Restorani menüü administreerimine

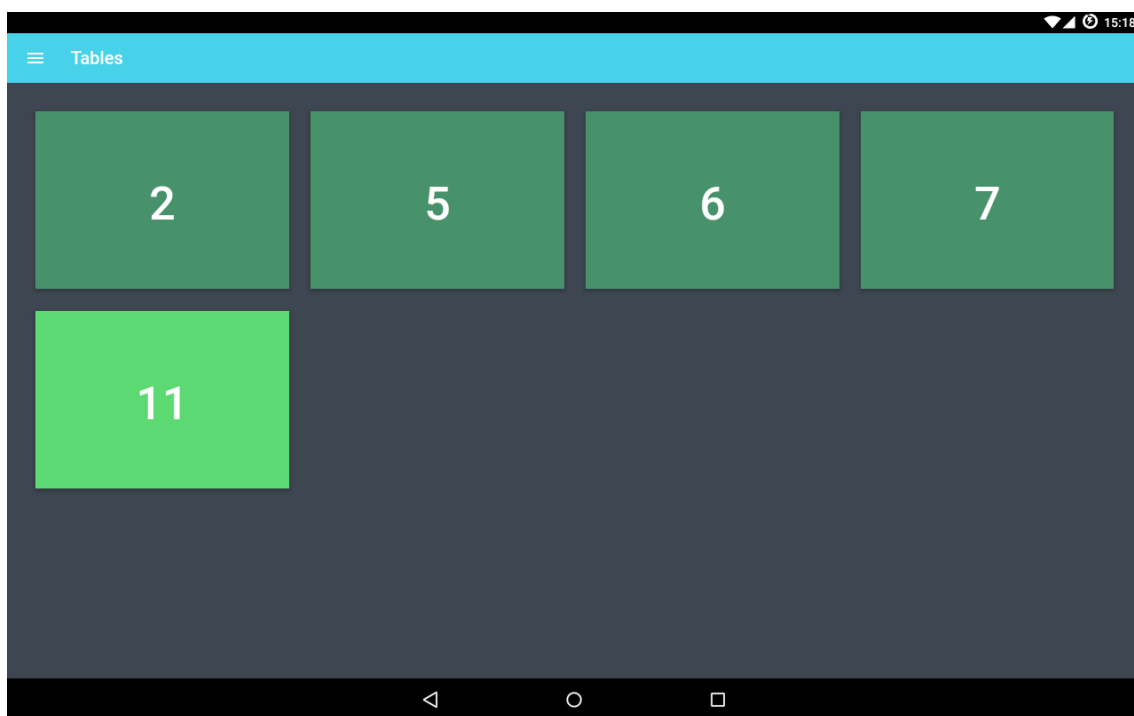
Uusi kategooriaid ja kategooria menüüd on võimalik lisada, muuta ja kustutada menüü administreerimise leheküljel. Valitud kategooria menüüd kuvatakse kategooria listi kõrval. Kategooria menüü päises kuvatakse hetkel valitud kategooria nimi (vt Joonis 7).



Joonis 7. Restorani menüü administreerimise ekraanivaade

### 3.2.4 Restoranide laudade vaade

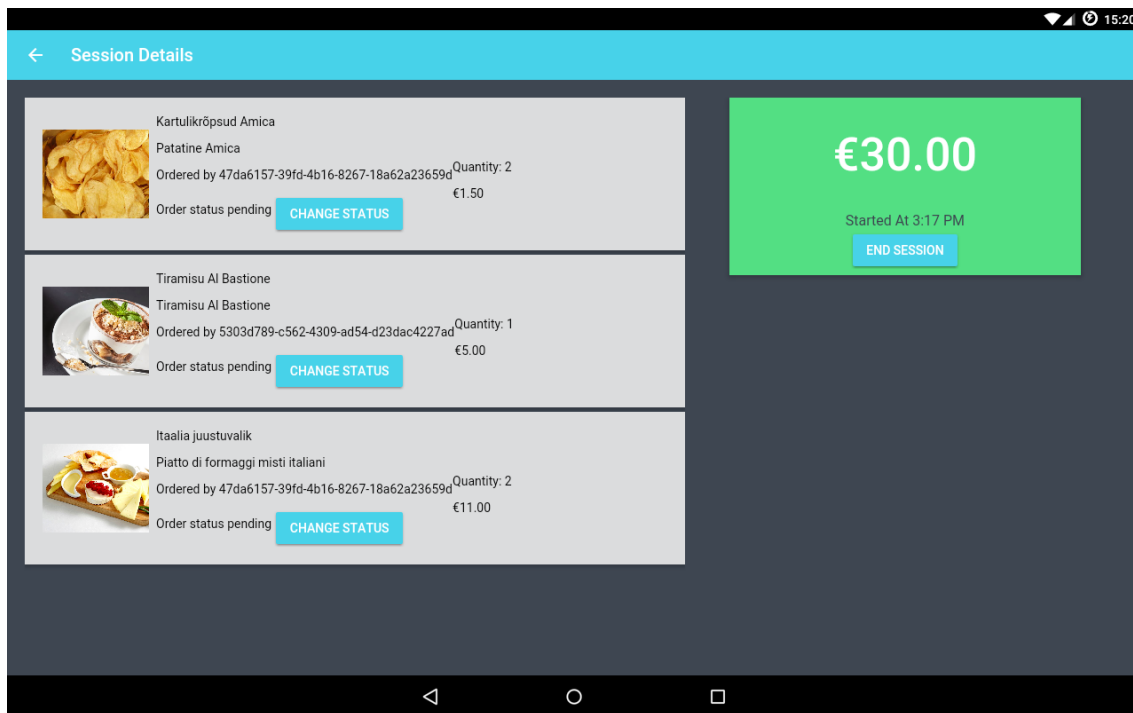
Laudade vaates kuvatakse kasutajale kõik laudade administratsiooni vaates loodud laudad. Lisaks laudade listi kuvamisele eristatakse hetkel aktiivseid laudu erksama tooniga (vt Joonis 8).



Joonis 8. Restorani laudade ekraanivaade

### 3.2.5 Aktiivse laua tellimuste vaade

Vajutades laudade vaates aktiivsele lauale (vt Joonis 8), kuvatakse kasutajale valitud laua tellimused. Lisaks tellimuste nägemisele on kasutajal võimalik antud vaates lõpetada aktiivne sessioon (vt Joonis 9). Aktiivse sessiooni soovib kasutaja lõpetada kui klient on oma tellimuste eest tasunud.

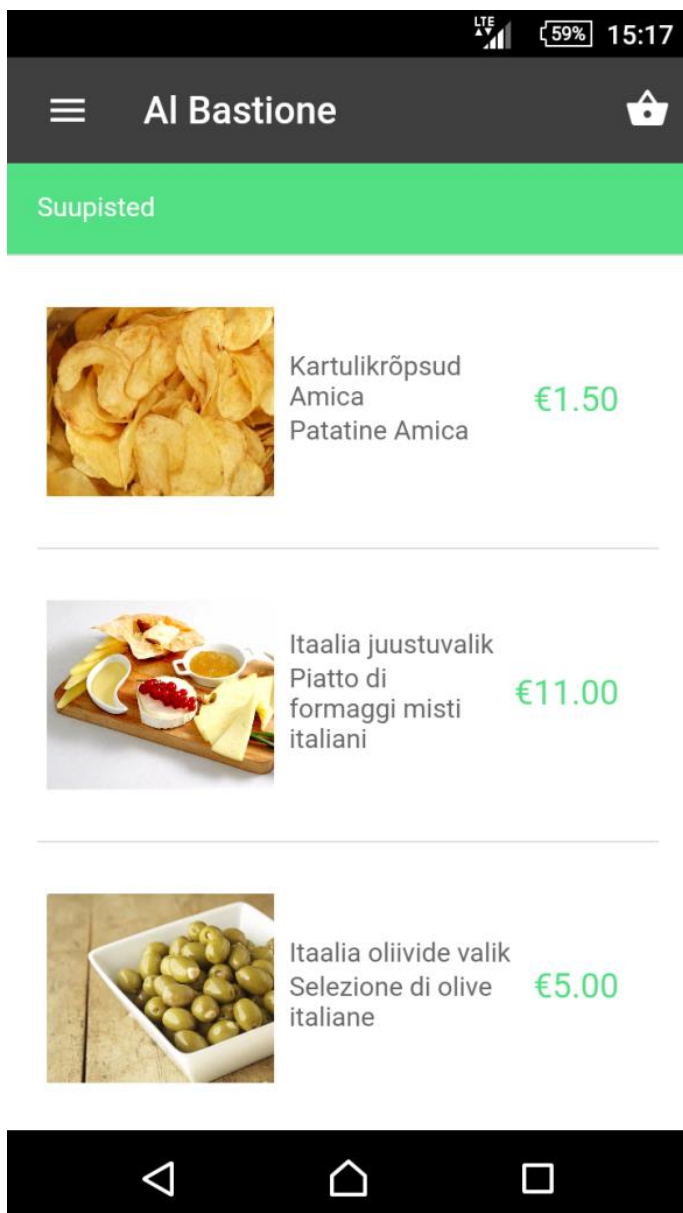


Joonis 9. Tellimuste ekraanivaade

## 3.3 Kliendirakendus

### 3.3.1 Restorani menüü kuvamine

Vastavalt NFC märgiselt saadud infole kuvatakse vastava restorani menüü. Menüü detailvaate avamiseks ja tellimiseks peab kasutaja vajutama soovitud tootel, mis avab uue akna. Juhul kui valitud lauas on juba aktiivne sessioon, kuvatakse üleval paremas nurgas ostukorv, mis võimaldab navigeerida tellimuste lehele (vt Joonis 10).



Joonis 10. Restorani menüü ekraanivaade

### 3.3.2 Tellimuse esitamine

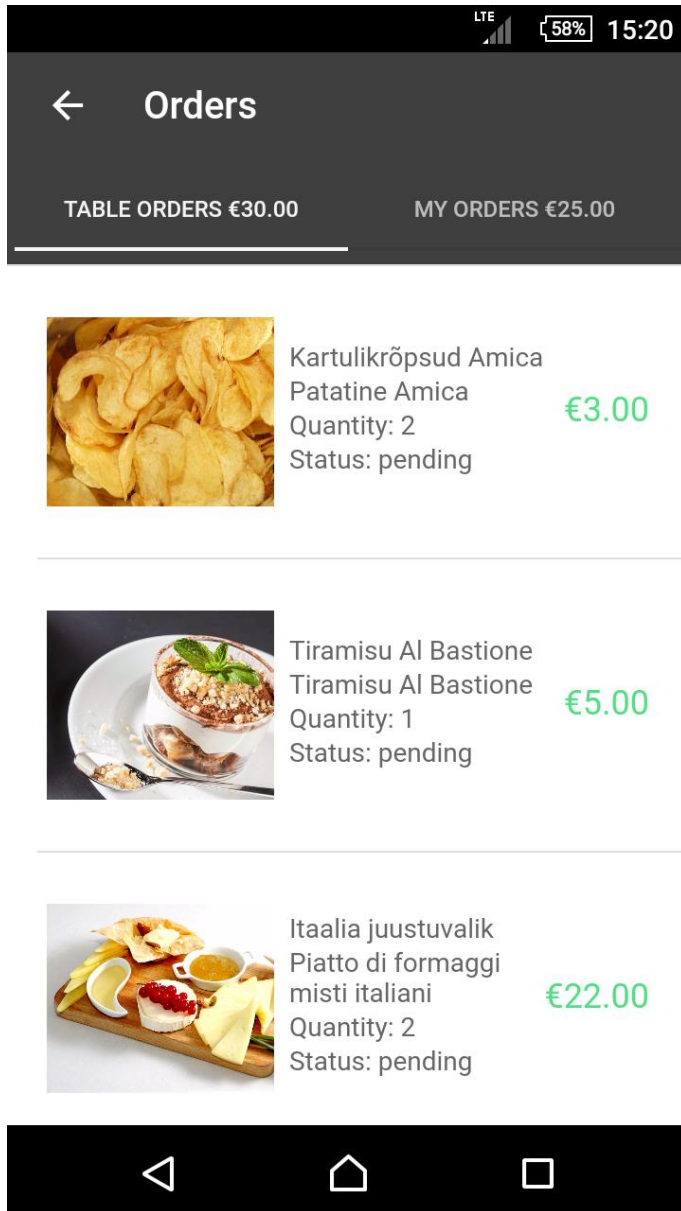
Tellimuse esitamisel on kasutajale võimaldatud ka mitme ühiku korraga tellimine. Tellimuse koguse suurendamiseks tuleb vajutada märgile „+“ või vastupidi „-“, seejärel arvutab süsteem ka uue hinna. Esitades tellimuse (Order), suletakse detailvaade automaatselt. Juhul kui tegu on kasutaja esimese tellimusega, kuvatakse talle ka kinnitusaken. Kui kasutaja ei soovi tellimust esitada, on tal võimalik detailvaade sulgeda ristist, mis asub vaatel üleval paremas nurgas (vt Joonis 11).



Joonis 11. Tellimuse esitamise ekraanivaade

### 3.3.3 Tellimuste vaatamine

Kasutajal on võimalik näha nii kogu laua tellimusi kui ka ainult enda tehtud tellimusi. Lisaks tellimuste kuvamisele, arvutab rakendus ka tellimuste koguhinnad (vt Joonis 12).



Joonis 12. Tellimuste ekraanivaade

### 3.4 Valminud teenuse vastavus esitatud nõuetele

#### 3.4.1 Teenindaja rakenduse vastavus nõuetele

Osad teenusele esitatud nõuded ei saanud täidetud. Tabelis on kirjeldatud esitatud nõuded ning vajadusel tulemuse täpsustus (vt Tabel 1).

**Tabel 1. Teenindaja rakenduse vastavus nõuetele**

<b>Nõue</b>	<b>Tulemus</b>
Teenuse kasutamiseks peab registreeruma kasutajaks.	Nõue mittetäidetud. Kasutajaid on autoril võimalik registreerida manuaalselt Firebase andmebaasis.
Registreerumine peab sisaldama restorani nime, e-posti aadressi ja parooli.	Nõue mittetäidetud (vt eelmist).
Sisselogimiseks peab kasutaja sisestama e-posti ja parooli.	Täidetud.
Kasutaja peab saama muuta restorani detailinfo.	Täidetud.
Restorani detailinfo peab sisaldama restorani nime ja aadressi.	Täidetud.
Kasutaja peab nägema olemasolevaid laudu.	Täidetud.
Kasutaja peab saama lisada uusi laudu.	Täidetud.
Laua info peab sisaldama laua kuvamise järjekorranumbrit.	Täidetud.
Kasutaja peab saama muuta laudade järjekorda.	Täidetud.
Kasutaja peab saama laudu kustutada.	Täidetud. Laua kustutamisel küsitakse

	kinnitust.
Kasutaja peab nägema menüü kategooriaid.	Täidetud.
Kasutaja peab saama lisada menüü kategooriaid.	Täidetud.
Menüü kategoria peab sisaldama nimetust ja kategoria kuvamise järjekorranumbrit.	Täidetud
Kasutaja peab saama muuta menüü kategooriaid.	Täidetud.
Kasutaja peab saama kustutada menüü kategooriaid.	Täidetud. Kategooria kustutamisel küsitakse kasutajalt kinnitust. Kinnituse korral kustutatakse ka kogu kategoria menüü.
Kasutaja peab nägema kategoria menüüd.	Täidetud.
Kasutaja peab saama lisada kategoria menüüd.	Täidetud.
Kategooria menüü peab sisaldama nimetust, toote infot, hinda ja menüü kuvamise järjekorranumbrit.	Täidetud.
Kasutaja peab saama muuta kategoria menüüd.	Täidetud.
Kasutaja peab saama kustutada kategoria menüüd.	Täidetud. Menüü kustutamisel küsitakse kinnitust.
Kasutaja peab nägema aktiivseid laudu.	Täidetud. Aktiivsed laudad kuvatakse



	kasutajale erksamates toonides.
Kasutaja peab nägema laua tellimusi.	Täidetud.
Kasutaja peab saama märkida aktiivse sessiooni lõppenuks.	Täidetud.

### 3.4.2 Kliendi rakenduse vastavus nõuetele

Osad teenusele esitatud nõuded ei saanud täidetud. Tabelis on kirjeldatud esitatud nõuded ning vajadusel tulemuse täpsustus (vt Tabel 2).

**Tabel 2. Kliendi rakenduse vastavus nõuetele**

<b>Nõue</b>	<b>Tulemus</b>
Kasutajal peab olema võimalik käivitada rakendus NFC märgise lugemisel.	Täidetud. Rakendust on võimalik käivitada ilma rakendust eelnevalt avamata lugedes seadmega otse NFC märgist.
Kasutaja peab nägema restorani menüüd.	Täidetud.
Kasutaja peab saama esitada tellimusi.	Täidetud. Sessiooni esimese tellimuse esitamisel küsitakse kasutajalt kinnitust.
Kasutaja peab nägema kõiki laua tellimusi.	Täidetud.
Kasutaja peab nägema enda esitatud tellimusi.	Täidetud.
Kasutaja peab saama tellimuste eest maksta.	Nõue mittetäidetud. Nõude turvaliseks täitmiseks on vajalik serverirakenduse kirjutamine.

### **3.5 Eelised võrreldes olemasolevate lahendustega**

Võrreldes olemasolevate lahendustega on suurimaks eeliseks jätkuv iseteenindus, kus kasutajal on võimalik esitada ja saada kätte tellimusi terve teenuse kasutamise aja vältel tarbides teenust isikliku seadmega. See on suureks eeliseks ka ettevõtjatele, kes peab teenuse kasutamiseks soetama vaid ühe seadme, mis on vajalik tellimuste täitmiseks. Tänu sellele on ka väikeettevõtjatel võimalik kasutajatele pakkuda kiiret ja mugavat iseteenindust ilma, et peaksid kliendi asukoha tuvastamiseks paigaldama laudadele statsionaarsed seadmed.

### **3.6 Puudujäägid ja edasiarendamise võimalused**

- Maksmine – tellimuste esitamine ja lõpetamine peaksid toimuma ühes süsteemis. Turvaliste maksete loomiseks on vajalik serverirakenduse loomine.
- Mitmekeelsus – rakendus on ehitatud ingliskeelsena. Lisaks rakenduse enda keelele, peaks olema võimalik sisestada ja kuvada mitmekeelset menüüd.
- Rahvusvaheliskus – rakendus kuvab kõik hinnad automaatselt eurodes ja kuupäevad Eesti formaadis. Tulevikus võiks saada restorani omanik ise määrata, mis rahaühikutes tema restoranis arveldatakse ning millises formaadis kuupäevi näidatakse.
- Dünaamiline kujundus – valminud rakenduses kuvatakse kõikide restoranide menüüd ühtse stiiliga. Lisateenusena võiks restoranile võimaldada oma kujunduse tellimise.
- Ainult ühe platvormi ja tehnoloogia põhine – bakalaureusetööga valminud rakenduse platvormiks valis autor Androidi ja kliendi asukoha määramiseks NFC tehnoloogia. Et võimaldada rohkematele inimestele rakenduse kättesaadavus, tuleks rakendus kirjutada ka iOS platvormile ja võimaldada info edastamine rakendusele ka QR koodiga, et katta ilma NFC tehnoloogiata seadmed [4,12].
- Äriplaani ja hinnakirja koostamine – valminud rakendusega on plaanis klientidelt teenida. Lisaks oma arendusele tuleb tasa teenida ka Firebase teenuse kuutasud.

## 4 Kokkuvõte

Käesoleva lõputöö eesmärgiks oli luua jätkuva iseteeninduse kontseptsioon kahe omavahel suhtleva mobiilirakendusena. Kliendi asukoha tuvastamiseks ja tellimuste esitamise võimaldamiseks võttis autor kasutusele NFC tehnoloogia.

Töö käigus uuris autor olemasolevaid iseteeninduse lahendusi Eestis, NFC tehnoloogia kasutusvõimalusi ja alternatiive seadmetele, mis ei toeta NFC tehnoloogiat. Lisaks uuris autor erinevaid Android rakenduste loomise viise ning vahendeid.

Autori poolt seatud eesmärk sai täidetud. Bakalaureusetöö tulemusena valmisid kaks omavahel suhtlevat rakendust. Esimene rakendus on toitlustusasutusele tellimuste vastuvõtmiseks ning teine rakendus mõeldud kliendile tellimuste esitamiseks. Selleks, et tuvastada kliendi asukoht toitlustusasutuses ja võimaldada jätkuv iseteenindus, võttis autor kasutusele NFC tehnoloogia. Loodud teenuse suurimateks eelisteks on toitlustusasutustele omapoolse arenduse ning suure hulga seadmete soetamisvajaduse puudumine ning kasutajatele isikliku seadme kasutamise võimaldamine.

Autori hinnangul võiks valminud teenus ja selle edasiarendus olla tugevaks ja paindlikuks konkurendiks tänaste teenusepakkujate turul.

## Summary

The purpose of this final thesis was to create a continuous self-service concept of two mutually communicating mobile apps. The author used the NFC technology to identify the customers location and allow the filing of orders.

During the work the author examined the existing self-service solutions in NFC technology applications and alternatives for devices that do not support NFC technology. In addition, the author studied the different ways of creating Android applications and tools.

The purpose, set by the author was accomplished. The thesis resulted in two interacting application. The first application is a catering facility for receiving orders, and the other one is for the implementation of customer orders. In order to identify the customer location in the establishment and allow for continued self-service, author used the NFC technology. The biggest advantage of the service that was created is that the catering establishments won't need to develop their own app and also they don't need to buy a large number of devices.

The author estimates that the service and its full development could make it a strong and flexible competitor on the service providers market.

## Kasutatud kirjandus

- [1] Vikipeedia, vaba entsüklopeedia. [Online]. <https://et.wikipedia.org/wiki/Rakendusliides>
- [2] Wikipedia, the free encyclopedia. [Online]. [https://en.wikipedia.org/wiki/Near\\_field\\_communication](https://en.wikipedia.org/wiki/Near_field_communication)
- [3] Kimtag. [Online]. [http://kimtag.com/s/nfc\\_tags](http://kimtag.com/s/nfc_tags)
- [4] Vikipeedia, vaba entsüklopeedia. [Online]. <https://et.wikipedia.org/wiki/QR-kood>
- [5] Wikipedia, the free encyclopedia. [Online]. [https://en.wikipedia.org/wiki/Software\\_development\\_kit](https://en.wikipedia.org/wiki/Software_development_kit)
- [6] IDC. [Online]. <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>
- [7] Android Developers. [Online]. <http://developer.android.com/guide/topics/connectivity/nfc/index.html>
- [8] Google Play. [Online]. <https://play.google.com/store/apps/details?id=com.wakdev.wdnfc&hl=et>
- [9] Wikipedia, the free encyclopedia. [Online]. [https://en.wikipedia.org/wiki/Android\\_Beam](https://en.wikipedia.org/wiki/Android_Beam)
- [10] Paytailor. [Online]. <http://www.paytailor.com/et/>
- [11] NFC world. [Online]. <http://www.nfcworld.com/nfc-phones-list/>
- [12] Cult of Mac. [Online]. <http://www.cultofmac.com/296093/apple-confirms-iphone-6-nfc-apple-pay/>
- [13] Wikipedia, the free encyclopedia. [Online]. [https://en.wikipedia.org/wiki/Android\\_\(operating\\_system\)](https://en.wikipedia.org/wiki/Android_(operating_system))
- [14] Android Developers. [Online]. <http://developer.android.com/about/dashboards/index.html>
- [15] Salesforce Developers. [Online]. [https://developer.salesforce.com/page/Native,\\_HTML5,\\_or\\_Hybrid:\\_Understanding\\_Your\\_Mobile\\_Application\\_Development\\_Options](https://developer.salesforce.com/page/Native,_HTML5,_or_Hybrid:_Understanding_Your_Mobile_Application_Development_Options)
- [16] Google Play. [Online]. <https://play.google.com/store>
- [17] Telerik Developer Network. [Online]. <http://developer.telerik.com/featured/what-is-a-hybrid-mobile-app/>
- [18] Wikipedia, the free encyclopedia. [Online]. [https://en.wikipedia.org/wiki/Ionic\\_\(mobile\\_app\\_framework\)](https://en.wikipedia.org/wiki/Ionic_(mobile_app_framework))
- [19] Wikipedia, the free encyclopedia. [Online]. <https://en.wikipedia.org/wiki/Firebase>
- [20] Firebase. [Online]. <https://www.firebase.com/features.html#features-database>
- [21] Firebase. [Online]. <https://www.firebase.com/pricing.html>
- [22] Mozilla Developer Network. [Online]. [https://developer.mozilla.org/en-US/docs/Web/API/WebSockets\\_API](https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API)
- [23] Github. [Online]. <https://github.com/chariotsolutions/phonegap-nfc>
- [24] JetBrains. [Online]. <https://www.jetbrains.com/webstorm/?fromMenu>

- [25] Git Bash. [Online]. <https://git-scm.com/downloads>
- [26] Cordova. [Online].  
<https://cordova.apache.org/docs/en/6.x/guide/platforms/android/>
- [27] Google Chrome. [Online].  
<https://www.google.com/chrome/browser/desktop/index.html>
- [28] Ebay. [Online].  
[http://www.ebay.co.uk/itm/291492961175?\\_trksid=p2057872.m2749.l2649&ssPageName=STRK%3AMEBIDX%3AIT](http://www.ebay.co.uk/itm/291492961175?_trksid=p2057872.m2749.l2649&ssPageName=STRK%3AMEBIDX%3AIT)