

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Miha Šumer

MOBILNA APLIKACIJA NFC V  
SKLADIŠČNIH PROCESIH

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE  
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

Ljubljana, 2014



UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Miha Šumer

MOBILNA APLIKACIJA NFC V  
SKLADIŠČNIH PROCESIH

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE  
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc.dr. Mira Trebar

Ljubljana, 2014



Rezultati diplomskega dela so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.





Št. naloge: 00489 / 2013  
Datum: 11.4.2013

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **MIHA ŠUMER**

Naslov: **MOBILNA APLIKACIJA NFC V SKLADIŠČNIH PROCESIH  
NFC MOBILE APPLICATION IN WAREHOUSE PROCESSES**

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija prve stopnje


Tematika naloge:

Tehnologija komunikacije kratkega dosega (NFC - Near Field Communication) predstavlja pomemben napredek na področju razvoja pametnih telefonov. Koristno bi jih lahko uporabili v informatizaciji in optimizaciji poslovnih procesov tako, da bi v manjših organizacijah nadomestili papirno dokumentacijo z elektronsko obdelavo podatkov. Kandidat naj v diplomskem delu analizira problem informatizacije skladišča za proizvode, ki so označeni z nalepkami NFC. Preuči naj možnosti uporabe v povezavi s pametnimi telefoni in komunikacijo s strežnikom. Na osnovi pridobljenih znanj naj izdela mobilno aplikacijo, ki bo delovala na operacijskem sistemu Android in bo omogočala enostavno uporabo v delovnem okolju. Analiza rezultatov naj bo izvedena na osnovi testiranja realnih podatkov, ki se nahajajo na strežniku v laboratoriju LRA.

Mentor:

  
doc. dr. Mira Trebar

Dekan:

  
prof. dr. Nikolaj Zimic





## IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Miha Šumer, z vpisno številko **63080433**,

sem avtor diplomskega dela z naslovom:

### **Mobilna aplikacija NFC v skladiščnih procesih.**

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom doc.dr. Mire Trebar,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki "Dela FRI".

V Ljubljani, dne 10. februar 2014

Podpis avtorja:



## **ZAHVALA**

Zahvaljujem se vsem, ki smo mi v kakršnkoli obliki pomagali in pripomogli k uspešno izdelani diplomski nalogi. Posebna zahvala gre doc. dr. Miri Trebar, ki mi je bila ves čas na razpolago in mi pomagala pri problemih, ki so se pojavili pri delu. Prav tako bi se rad zahvalil as. Roku Češnovarju v LRA za pomoč pri delu s strežnikom.

Na koncu bi se rad zahvalil še staršem, ki so mi omogočili študij in me ves čas podpirali, preganjali in spodbujali.



# KAZALO VSEBINE

1	UVOD.....	1
2	TEHNOLOGIJE IN PROGRAMSKA ORODJA .....	3
2.1	NFC (Near Field Comunication) .....	3
2.1.1	AKTIVNE NAPRAVE NFC.....	4
2.1.2	ZNAČKE NFC .....	5
2.1.3	VARNOST .....	7
2.2	ANDROID .....	8
2.2.1	VERZIJE SISTEMOV ANDROID.....	8
2.2.2	VELIKOST ZASLONA IN RESOLUCIJA.....	9
2.2.3	SHRANJEVANJE PODATKOV .....	10
2.3	PROGRAMSKO OKOLJE IN DRUGO .....	10
2.3.1	ECLIPSE V PROGRAMSKEM JEZIKU JAVA.....	10
2.3.2	NETBEANS IN STREŽNIK.....	11
3	PROCES SKLADIŠČENJA.....	13
3.1	INFORMATIZACIJA PROCESA SKLADIŠČENJA.....	14
3.2	ARHITEKTURA SISTEMA .....	17
4	NAČRTOVANJE IN IZDELAVA APLIKACIJE .....	19
4.1	PROCES SKLADIŠČENJA .....	19
4.2	OBLIKOVANJE IN ZASNOVA APLIKACIJE.....	21
4.2.1	OSNOVNI MENI.....	21
4.2.2	INFORMACIJE.....	22
4.2.3	NAROČILNICE .....	22
4.2.4	SPREJEM IZDELKOV .....	23
4.2.5	SKLADIŠČE .....	25
4.2.6	IZDAJNICE.....	25
4.2.7	IZDAJA IZDELKOV .....	26
4.3	ZAJEM PODATKOV - STREŽNIK.....	27

5	UPORABA APLIKACIJE SKLADIŠČENJA .....	31
5.1	OSNOVNI MENI IN INFORMACIJE.....	31
5.2	NAROČILNICE IN SPREJEM .....	34
5.2.1	NAROČILNICE .....	34
5.2.2	SPREJEM .....	36
5.3	SKLADIŠČE .....	41
5.4	IZDAJNICE TER IZDAJA .....	41
6	SKLEPNE UGOTOVITVE.....	43
7	LITERATURA .....	45

## **SEZNAM KRATIC**

API (Application Programming Interface)

CDDL (Common Development and Distribution License)

ECMA (European Computer Manufacturers Association)

EE (Enterprise Edition)

EPC (Electronic Product Code)

GPL (GNU General Public License)

HTTP (Hypertext Transfer Protocol)

IBM (International Business Machines Corporation)

IEC (International Electrotechnical Commission)

IDE (Integrated Development Environment)

ISO (International Organization for Standardization)

Java EE (Enterprise Edition)

JMS (Java Message Service)

JPA (Java Persistence API)

JSON (JavaScript Object Notation)

LLCP (Logical Link Control Protocol)

NDEF (NFC data exchange format)

NFC (Near Field Communication)

NFCIP-1 (NFC Interface and Protocol-1)

QR (Quick Response)

REST (Representational State Transfer)

RF (Radio-Frequency)

RFID (Radio-frequency Identification)

RMI (Remote Method Invocation)

URI (Uniform Resource Identifier)

URL (Uniform Resource Locator)

XML (Extensible Markup Language)



## **POVZETEK**

V diplomski nalogi je predstavljena aplikacija, ki omogoča avtomatsko izvedbo procesa skladiščenja. Izdelana je z uporabo tehnologije NFC (Near Field Communication) za mobilne naprave z operacijskim sistemom Android in omogoča preprosto identifikacijo blaga. V skladiščih se večinoma še vedno uporablja papirnato dokumentacijo, ki jo je mogoče delno ali v celoti že zamenjati z elektronsko, medtem ko se za označevanje in identifikacijo uporablja črna koda ali koda QR (Quick Response).

Pri izdelavi diplomske naloge so za izvedbo procesa skladiščenja uporabljene naprave za komunikacijo na kratek doseg, to so značke NFC, ki označujejo izdelke in pametni telefoni z vgrajenim modulom NFC. Papirnati dokumenti so nadomeščeni z elektronskimi v formatu XML (Extensible Markup Language) in se nahajajo v datotekah na strežniku. Mobilna aplikacija pridobiva podatke z uporabo spletnih storitev RESTful in omogoča identifikacijo izdelkov, jih razvršča in skrbi za pregled zalog. Pomembne značilnosti njene uporabe so v zmanjšanju potrebnega časa za izvajanje posameznih postopkov, poenostavitvi nadzora v skladišču in predvsem v uporabi vsesplošno razširjenih mobilnih telefonov tudi za praktično delo v poslovnih procesih. Testiranje in analiza uporabnosti rešitve sta bila izvedena na osnovi realnih podatkov, ki se nahajajo na strežniku v Laboratoriju za računalniško arhitekturo.

### **Ključne besede:**

Skladiščenje, označevanje, identifikacija, mobilna aplikacija, Android, NFC (Near Field Communication)



## **ABSTRACT**

This thesis presents an application that enables automatic execution of the storage process. It is made for NFC (Near Field Communication) technology use for mobile devices with Android operating system and allows simple product identification. Warehouses usually still use paper documentation that can be partly or totally replaced by the electronic version, while for labelling and identification they use barcode or QR (Quick Response).

In our thesis, we used short range communication devices, NFC tags that label products and smart phones with the implemented NFC module for the execution of the storage process. Paper documentation was replaced by electronic XML (Extensible Markup Language) format and was located in files on a server. The mobile application gets the information using the RESTful web services and enables product identification, sorts the products out and provides stock overview. Some important characteristics of its usage are the reduction of the required time for implementation of individual procedure, simplifying warehouse supervision and most of all the wide use of mobile phones for practical work in business processes. Testing and analysis of usability solutions were carried out based on real data located on the server in the Laboratory for Computer Architecture.

### **Keywords:**

Storage, labelling, identification, mobile application, Android, NFC (Near Field Communication)



## 1 UVOD

Z razvojem sodobnih tehnologij in množico splošno razširjenih mobilnih naprav obstajajo različne možnosti za hitrejšo in enostavnejše delo v številnih proizvodnih procesih. Po napovedih in pričakovanjih številnih podjetij naj bi se v prihajajočih letih znatno povečala uporaba in razvoj tehnologije NFC, ne samo v poslovnem svetu, temveč tudi v vsakdanji rabi. Prav tako je pričakovati vključitev te tehnologije v vse pametne telefone.

Naš cilj je bil izdelati sistem, ki bi z uporabo splošno razširjenih NFC naprav omogočal v procesu skladiščenja izdelkov hitrejšo in enostavnejše delo. Hkrati pa bi predstavljal relativno cenejšo izvedbo in zmanjšal potrebo po papirni dokumentaciji in s tem pripomogel tudi k reševanju okoljevarstvenih problemov z zmanjševanjem uporabe lesa.

V diplomskem delu smo za izdelavo mobilne aplikacije izbrali tehnologijo NFC, ki jo je mogoče uporabljati na mobilnih napravah za poenostavljeno izvedbo postopkov v skladiščnih procesih. Motivacijo za delo smo dobili pri pregledovanju nalog in problemov manjših skladišč, ki se za razliko od velikih, katera so tehnološko že dobro opremljena, še vedno soočajo s problemom papirne dokumentacije. Razvili in testirali smo aplikacijo, ki za svoje delovanje potrebuje napravo Android, vsebuje tehnologijo NFC in omogoča lažje delo v procesih skladiščenja, obenem pa se lahko zaradi shranjevanja podatkov v podatkovne baze izognemo prej omenjeni papirni dokumentaciji. Upoštevali smo postopke, kot so prevzem izdelka, njegovo shrambo ter izdajo iz skladišča.

Razvoj in testiranje sta bila izvedena s pomočjo pametnega telefona Samsung Galaxy S4 s sistemom Android 4.3, ki skenira NFC značko nalepljeno na pošiljki blaga. Iz nje prebere vse potrebne podatke za nadaljnjo računalniško obdelavo. Za preverjanje sprejetih in izdanih izdelkov smo imeli na strežniku podatke v datotekah xml. Na pametni telefon so se podatki prenesli z uporabo RESTful spletne storitve. Zaključeno aplikacijo smo testirali še na dveh drugih napravah, enem pametnem telefonu in tabličnem računalniku.



## 2 TEHNOLOGIJE IN PROGRAMSKA ORODJA

Za razvoj mobilne aplikacije je potrebno izbrati in analizirati tehnologije in ustrezna programska orodja, ki omogočajo uporabo sodobnih naprav v poslovnih procesih. Predstavljena je tehnologija NFC (Near Field Communication), ki omogoča brezžični prenos podatkov na kratke razdalje v povezavi z operacijskim sistemom Android. Izdelava aplikacije in urejanje podatkov vključujeta tudi izbiro ustreznega programskega okolja in druge opreme.

### 2.1 NFC (Near Field Communication)

Začetki tehnologije NFC (Near Field Communication) segajo v leto 2002, ko sta podjetji Philips in Sony skupaj predstavila brezkontaktno povezavo, ki jo je v decembru istega leta sprejelo evropsko združenje proizvajalcev računalnikov ECMA (European Computer Manufacturers Association). Mednarodna organizacija za standardizacijo ISO (International Organization for Standardization) in mednarodna elektrotehnična komisija IEC (International Electrotechnical Commission) sta sprejeli tehnologijo NFC decembra 2003 [1].

NFC deluje kot brezžična tehnologija na kratke razdalje, običajno do 10cm. Pravzaprav je eno izmed področij radio frekvenčne identifikacije RFID (Radio-Frequency Identification) s to razliko, da morata biti tukaj napravi relativno blizu skupaj. Delujejo na osnovi magnetne indukcije. Bralnik oddaja majhen električni tok, ki ustvari magnetno polje med napravama. To polje nato druga naprava s pomočjo tuljave sprejema, ga spremeni nazaj v električne impulze ter nato v podatke, kot so identifikacijska številka, informacija o statusu ali katerekoli druge vrste informacij [7].

NFC deluje na radijski frekvenci 13.56 MHz in uporablja manj kot 15 mA moči za komunikacijo med napravama, na razdalji 10 cm ali manj. Značke največkrat hranijo med 96 in 512 biti podatkov in prenašajo podatke s hitrostjo 106 Kb/S, 212 Kb/S, 424 Kb/S ali 848 Kb/S, kar je dovolj za prenos majhnih delov informacij skoraj v trenutku. To je zelo pomembno pri aplikacijah z velikim številom prenosov podatkov [7].

Podatki na znački NFC so za sprejem in oddajo med napravami hranijo v obliki NDEF (NFC data exchange format) sporočil. Uporablja se več tehnologij dostopa, kot so NfcA, NfcB, NfcF, NfcV, IsoDep ter Ndef in NdefFormatable [10]. Za branje sporočila iz značke NFC se največ uporablja Ndef, ki analizira prebrano značko in ustrezno kategorizira podatke. Na osnovi te kategorizacije se zažene aplikacijo, ki je odgovorna za dobljene podatke. Aplikacija, s katero se poskenira značko, lahko nato deklarira namenski filter in zahteva obdelovanje podatkov kot:

- »action ndef discovered« - najpomembnejši in prevzame najvišjo prioriteto,
- »action tech discovered« - z malo manjšo in
- »action tag discovered« - z najmanjšo prioriteto [11].

Naprave so lahko pasivne ali aktivne. Pasivne naprave uporabljajo energijo bralnika, da lahko kodirajo odgovor, medtem, ko imajo aktivne naprave svoj vir napajanja ter odgovorijo bralniku z uporabo svojega elektromagnetnega polja [7].

### 2.1.1 AKTIVNE NAPRAVE NFC

Aktivna naprava (Slika 2.1), lahko bere in pošilja informacijo. NFC naprava, kot je pametni telefon, lahko prejme podatke iz NFC značke, izmenjuje te podatke z drugimi napravami, lahko pa tudi spremeni podatke na NFC znački, če ima za to ustrezna dovoljenja [13].



Slika 2.1: Delovanje NFC naprav [13].

Obstajajo trije načini delovanja tehnologije NFC:

- bralnik/pisalnik (ang. reader/writer),
- vsak z vsakim (ang. peer to peer) in
- emulacija kartice (ang. card emulation).

**Način bralnik/pisalnik** je komunikacija med napravo s tehnologijo NFC in značko NFC z namenom branja ali pisanja na značko. Aktivna naprava, ki z aplikacijo zažene proces vsposatvi povezavo in prebere podatke iz značke NFC. Pri pisanju, naprava deluje tako, da zapiše podatke na značko. V primeru, da so na znački že zapisani podatki pred procesom pisanja, se ti izbrišejo in nanjo zapišejo novi. Možno je tudi uporabiti postopek, ki podatke posodobi in jih ne izbriše [1].

**Način vsak z vsakim** omogoča dvema napravama, ki vsebujeta tehnologijo NFC, izmenjavo informacij, kot so zapis kontaktov, tekstovno sporočilo ali kakršno koli drugo obliko podatkov. Obstajata dva načina izmenjave podatkov:

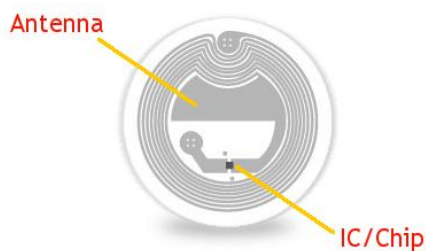
1. NFCIP-1 (NFC Interface and Protocol-1) je narejen po vzorcu pobudnik-cilj, v katerem sta obe napravi definirani pred začetkom komunikacije.
2. LLCP (Logical Link Control Protocol), ki se uporablja v povezavi z NFCIP-1. V tem načinu sta obe napravi, ki sodelujeta v komunikaciji, identični. V primeru vsak z vsakim sta obe napravi v aktivnem stanju in delujeta tako, da takrat ko ena oddaja, druga posluša in začne oddajati podatke, ko prva zaključi [1].

**Način emulacije kartice** omogoča, da se naprava, ki vsebuje tehnologijo NFC lahko obnaša kot brezkontaktna pametna kartica na katero se lahko shrani več aplikacij. Najbolj poznani primeri uporabe brezkontaktnih pametnih kartic so kreditne kartice, kartice zvestobe itd. Trenutno podprti komunikacijski vmesniki za način emulacije kartic so ISO/IEC 14443 tipa A in tipa B ter FeliCa (ang. Sony's FeliCa contactless IC card technology). Ta način omogoča tudi uporabo v plačilnih sistemih in aplikacijah vezanih na preverjanje vstopnic [1].

### 2.1.2 ZNAČKE NFC

Za prepoznavo tehnologij, ki vsebujejo NFC, poznamo štiri tipe značk ter aktivno in pasivno vlogo delovanja. Tip značke je odvisen od hitrosti in združljivosti med NFC značko in NFC bralnikom. Vloga pa določa, kako se aktivna ali pasivna naprava odziva med NFC komunikacijo. Največkrat se uporablja značka NFC z vgrajenim naslovom URL (Uniform Resource Locator), ki zavzame majhno količino pomnilnika in je dokaj poceni, saj se pogosto zgodi, da se take značke uporabi na raznih reklamnih plakatih in jih nato zavrže. Prav tako je možno shranjevanje večjih količin podatkov, vendar je tudi dražje [11].

Značke so najpogosteje pasivne naprave (Slika 2.2), kot je nalepka ali kartica in vsebuje informacijo, katero lahko druge naprave preberejo, sama naprava pa ni sposobna branja.



Slika 2.2: NFC značke - pasivna naprava [17].

Poznamo štiri različne tipe značk:

- Tip 1 - ima zaščito proti koliziji podatkov (ang. data collision) in se jo lahko nastavi na branje in zapisovanje ali le branje. Le branje preprečuje, da bi uporabnik, informacijo na znački, ko je ta enkrat vpisana, spremenil ali prepisal. Značka ima 96 bitov spomina, kar je dovolj za URL ali drugo identifikacijsko oznako z majhno količino podatkov. Spomin značke se po potrebi lahko tudi poveča. Ta tip značke je najprimernejši za večino potreb in je tudi najpogostejši, saj je cenovno ugoden.
- Tip 2 - ima vse kar ima značka tipa 1. Od nje se razlikuje le v strukturi in v spominskem prostoru, ki se začne pri 48 bitih, kar je pol toliko kot pri tipu 1. Povečanje spomina in hitrost komunikacije sta enaki tako pri tipu 1 kot pri tipu 2.
- Tip 3 - se od predhodnih dveh razlikuje po večji kapaciteti spomina in hitrejši komunikaciji. Ta tip je del sistema FeliCa. Večja velikost dovoljuje bolj kompleksne zapise od URL, vendar je tudi cena višja.
- Tip 4 - lahko vsebuje NFC-A ali NFC-B komunikacijo in ima, tako kot vsi ostali tipi, zaščito pred kolizijo podatkov. Znački se nastavi način branja/pisanja že med izdelavo in ga uporabnik, za razliko od ostalih tipov, ne more spremeniti. Značka ima 32 Kbitov in deluje hitreje od vseh ostalih [12].

Slika 2.3 prikazuje več različnih vrst značk, kot so nalepka, kartica, obesek, zapestnica in drugo [4].



Slika 2.3: Različni tipi ter vrste značk [4].

Uporabljata se dva standarda, ki sta povezana z načinom bralnik/pisalnik ISO/IEC 14443 in ISO/IEC 15693. Oba podpirata pasivni način komunikacije s tem, da ISO/IEC 14443 zahteva neposredno bližino značke in deluje s hitrostjo 106 Kbps, ISO/IEC 15693 pa omogoča večjo okoliško oddaljenost značke in deluje s hitrostjo do 26Kbps [1].

### 2.1.3 VARNOST

Za interakcijo med napravami NFC, kot je v primeru uporabe telefona, ni potrebno posebno posredovanje, zato se upravičeno pojavi vprašanje varnosti. A tudi zato je v določeni meri poskrbljeno. Povezava je mogoča le, ko ima naprava aktiviran NFC način oddajanja in sprejemanja ter mora biti v času željene interakcije aktivna, kar pomeni, da ekran ne sme biti zaklenjen. Tako ni mogoče, da bi se naprimer neželjena oseba, v avtobusu polnem ljudi, skrito povezala z uporabnikovo napravo, tako da bi enostavno stala v njegovi neposredni bližini in bila s tem v območju dosega njegove naprave medtem, ko bi ga imel v žepu.

Za zagotovitev varnosti NFC pogosto vzpostavi varen kanal in uporabi kodiranje medtem, ko pošilja občutljive informacije, kot je na primer številka kreditne kartice ali podobni pomembni podatki. Seveda pa, kot so uporabniki mobilnih naprav že navajeni, je potrebno, da za varnost podatkov poskrbi tudi uporabnik sam. Priporočljiva je uporaba protivirusnih programov ter gesla za vstop v mobilno napravo v primeru izgube ali kraje [12].

## 2.2 ANDROID

Android je operacijski sistem, ki se je v zadnjih letih močno razširil. Uporabljajo ga v več kot 190 državah po celem svetu na več sto milijonih naprav, kot so telefoni in tablični računalniki. Dnevno se z operacijskim sistemom Android sreča preko milijon novih uporabnikov, ki začnejo brskati za različnimi aplikacijami, igrami ter drugimi uporabnimi vsebinami [6].

Operacijski sistem je zelo dobro izdelan, a se kljub temu razvijalci nenehno trudijo še izboljšati njegove funkcionalnosti. Razvita je svetovna platforma, ki omogoča kreiranje novih aplikacij in iger, katere lahko uporabnik naloži na odprto trgovino in jih posreduje drugim uporabnikom. Prav Androidova odprtost je najbolj vplivala na priljubljenost in hitro rast števila uporabnikov. Kot zanimivost lahko povemo, da uporabniki Androida mesečno naložijo več kot 1.5 bilijona aplikacij in iger.

Android ponuja vse za razvoj aplikacij odlične kakovosti in izgleda, ki se prav tako prilagodijo sposobnostim uporabnikove naprave. Uporabniški vmesnik se avtomatsko prilagodi tako, da aplikacija izgleda najbolje, istočasno pa omogoča dober nadzor nad njo [6].

»Google Play« je spletna trgovina, ki omogoča prodajo in posredovanje aplikacij Android. Ker je »Google Play« odprta trgovina, lahko razvijalec, ponudnik aplikacije določa način prodaje izdelka. Lahko ga trži po celem svetu ali se osredotoči na določen segment, določeno napravo ali morda na naprave s specifičnimi sposobnostmi.

Aplikaciji lahko razvijalec postavi ceno po svoji izbiri ter z njo upravlja kadarkoli, seveda pa lahko aplikacijo naloži tudi kot brezplačno. V primeru, odločitve razvijanja aplikacij, je dobro vedeti, da sistem Android avtomatsko skrbi za razvijalčevo razpoznavnost. Aplikacije, ki dosegajo višje zanimanje, jim sistem določi višjo prioriteto in tako uvrsti višje na seznam priljubljenih ter jih bolje oceni.

### 2.2.1 VERZIJE SISTEMOV ANDROID

V razvoju operacijskega sistema Android obstaja veliko verzij, kjer vsaka predstavlja pomemben razvojni napredek. Tabela 2.1 ponazarja deleže posamezne verzije vseh registriranih naprav v aplikaciji Google Play in prikazuje katero najnižjo API (Application Programming Interface) podpira. Prikazan je rezultat raziskovanja in ne vključuje verzij nižjih od Android 2.2.

Version	Codename	API	Distribution
<a href="#">2.2</a>	Froyo	8	1.7%
<a href="#">2.3.3 -2.3.7</a>	Gingerbread	10	26.3%
<a href="#">3.2</a>	Honeycomb	13	0.1%
<a href="#">4.0.3 - 4.0.4</a>	Ice Cream Sandwich	15	19.8%
<a href="#">4.1.x</a>	Jelly Bean	16	37.3%
<a href="#">4.2.x</a>		17	12.5%
<a href="#">4.3</a>		18	2.3%

Tabela 2.1: Delež uporabe različnih verzij sistema Android [6].

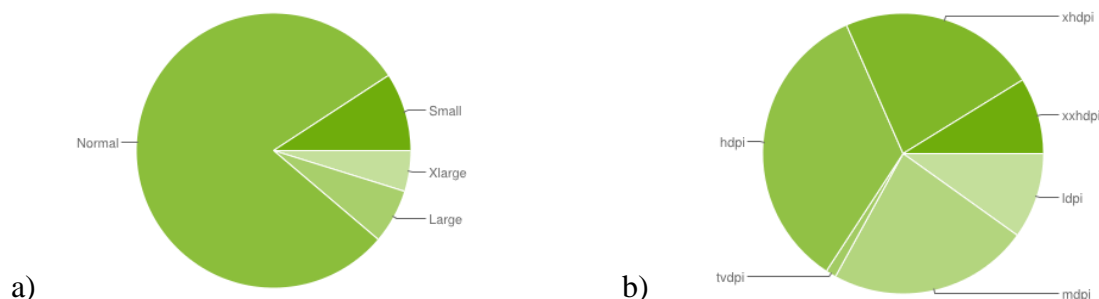
## 2.2.2 VELIKOST ZASLONA IN RESOLUCIJA

Android sistem je zasnovan tako, da pozna štiri različne velikosti (small, normal, large in xlarge) ter šest resolucij (ldpi, mdpi, tvdpi, hdpi, xhdpi, xxhdpi), kjer se najmanj uporabljata tvdpi in xxhdpi. Velikost »small« predstavlja zaslone z vsaj 426dp x 320dp, »normal« z vsaj 470dp x 320dp, »large« z vsaj 640dp x 480dp in »xlarge« z vsaj 960dp x 720dp. V Tabeli 2.2 je prikazan delež uporabe različnih kombinacij Android naprav za verzije od 2.2 glede na velikost zaslona in resolucijo.

	ldpi	mdpi	tvdpi	hdpi	xhdpi	xxhdpi	Total
<b>Small</b>	9.2%						<b>9.2%</b>
<b>Normal</b>	0.1%	15.1%		33.4%	22.2%	8.8%	<b>79.6%</b>
<b>Large</b>	0.6%	3.6%	1.2%	0.5%	0.5%		<b>6.4%</b>
<b>Xlarge</b>		4.4%		0.3%	0.1%		<b>4.8%</b>
<b>Total</b>	<b>9.9%</b>	<b>23.1%</b>	<b>1.2%</b>	<b>34.2%</b>	<b>22.8%</b>	<b>8.8%</b>	

Tabela 2.2: Delež uporabe velikosti v razmerju z resolucijo zaslona [6].

Slika 2.4(a) ponazarja deleže uporabe različnih velikosti zaslona med uporabniki sistema Android, Slika 2.4(b) pa ponazarja deleže uporabe različnih resolucij zaslona med uporabniki sistema Android [6].



Slika 2.4: Deleži uporabe različnih velikosti (a) in resolucij (b) zaslona [6].

### 2.2.3 SHRANJEVANJE PODATKOV

Android vsebuje razrede za upravljanje s podatkovno bazo SQL (ang. SQLite databse) [6]. Aplikacija uporablja te razrede, za dostop do privatnih podatkovnih baz in shranjevanja podatkov vanje. Podatkovni bazi se dodeli:

- ime baze,
- ime tabele, ki jo baza hrani in
- verzijo.

Ime baze enolično določa podatkovno bazo. Tabela v njej je dvodimenzionalna in vsebuje attribute z vrednostmi. Tabel je lahko v eni podatkovni bazi tudi več. V primeru zahteve po preoblikovanju baze je v aplikaciji potrebno spremeniti njeno verzijo, na podlagi katere se potem začne preoblikovanje. Podatkovne baze se oblikuje in ureja s pomočjo ukazov SQL.

## 2.3 PROGRAMSKO OKOLJE IN DRUGO

### 2.3.1 ECLIPSE V PROGRAMSKEM JEZIKU JAVA

Za razvoj mobilne aplikacije smo uporabili programsko okolje Eclipse, verzije Kepler Service Release 1 [3]. Izdelano je v okviru skupnosti za posameznike in organizacije, ki želijo sodelovati v komercialno prijazni, odprto-kodni programski opremi. Projekti te skupnosti so osredotočeni na gradnjo odprte platforme za razvoj, ki je sestavljena iz delovnih okvirjev (ang. frameworks), orodij in trajanj izvajanja (ang. runtimes) za gradnjo, vpeljavo ter upravljanje programske opreme nad celotnim procesom.

Eclipse Foundation je neprofitna korporacija, ki nudi pomoč uporabnikom in gosti odprtokodne projekte, imenovane '*Eclipse Projects*' ter podpira tako odprto-kodno skupnost kot tudi ekosistem dopolnjujočih izdelkov ter storitev [3]. Ustanovljena je bila januarja 2004 kot neodvisna organizacija, katera bi trgovcem omogočila vzpostaviti nevtralno, odprto ter pregledno skupnost v zvezi z Eclipse in bi jo ta korporacija upravljala. Ponuja štiri storitve za Eclipse skupnost:

1. IT (Information Technology) infrastrukturo
2. IP (Internet Protocol) management
3. Razvojne procese
4. Razvoj ekosistema

Zaposleno osebje je povezano z vsakim izmed teh področji in s pomočjo Eclipse skupnosti poskuša zadovoljiti in izpolnjevati potrebe interesne skupine.

*Eclipse Project* je prvotno ustanovil IBM (International Business Machines Corporation), novembra 2001 in bil podprt s strani konzorcija za programsko opremo. Sedaj pa pokrivajo:

- trajanja izvajanja (ang. runtimes),
- statične ter dinamične jezike,
- »thick-client« (debel-klient), »thin-client« (tanek-klient) ter delovne okvirje s strani delovanja strežnika,
- modeliranje in poslovno poročanje,
- najboljši Java IDE (integrated development environment).

### 2.3.2 NETBEANS IN STREŽNIK

Za delovanje naše aplikacije je bilo potrebno narediti storitev (ang. service), ki bo skrbela za pravilno prejemanje naročilnic in izdajnic. Odločili smo se za programiranje v okolju NetBeans in uporabo RESTful Web services ter strežnika GlassFish, ker se naročilnice in izdajnice nahajajo na strežniku, na katerem je nameščena omenjena programska oprema.

#### **NetBeans**

Začetki ustanove NetBeansa segajo v leto 1996, kot študentski projekt (prvotno imenovan Xelfi) na Češkoslovaškem oz. današnji Češki Republiki. Cilj je bil narediti Delphi podoben Java IDE napisan v Javi. Projekt je pritegnil dovolj pozornosti, da so se študenti, ki so delali na tem projektu odločili, da ga bodo poskusili tržiti kot komercialni produkt.

Spomladi, leta 1999, so izdali NetBeans DeveloperX2, ki je podpiral Swing. Uspešnost izboljšav v JDK 1.3, izdanih jeseni 1999 je naredilo NetBeans izvedljivo izbiro za razvojno

orodje. Do poletja 1999 je ekipi uspelo preoblikovati DeveloperX2 v bolj modularen NetBeans, ki še danes oblikuje osnovo programske opreme [5].

NetBeans IDE nudi podporo za kar nekaj programskih jezikov (PHP, JavaFX, C/C++, JavaScript, itd.) in delovnih okvirjev. Je odprto-kodni projekt namenjen zagotavljanju programske opreme za razvoj produktov (NetBeans IDE in NetBeans platforma), ki obravnavajo potrebe razvijalcev, uporabnikov in podjetij, ki se zanašajo na NetBeans kot osnovo njihovih produktov. Prav tako želijo omogočati hitro, učinkovito in enostavno delovanje razvoja produktov z izkoriščanjem prednosti platforme Java in drugih industrijskih standardov.

NetBeans IDE in NetBeans platforma sta brezplačna za komercialno in nekomercialno rabo. Izvorna koda za oboje je na voljo vsakomur in jo lahko uporabi ali preoblikuje po potrebi v skladu s pogoji uporabe v opredeljeni licenci [5].

### **RESTful Web services**

REST (Representational State Transfer) opredeljuje niz arhitekturnih načel, s katerimi se definira spletne storitve. Te se osredotočajo na sistemske vire, vključno s tem, kako so viri stanj naslovljeni in preneseni v okviru aplikacijskega protokola HTTP (Hypertext Transfer Protocol). Načela oblikovanja REST Web service so:

- izrecna uporaba HTTP metod,
- brez uporabe stanj,
- URI struktura in
- prenos XML, JSON (JavaScript Object Notation).

Za kreiranje vira na strežniku se uporablja metoda POST, za pridobitev vira se uporablja metoda GET, za posodobitev ali spremembo stanja se uporablja PUT in za izbris se uporablja DELETE [16].

### **GlassFish**

GlassFish je projekt odprto-kodnega aplikacijskega strežnika, ki ga je razvil Sun Microsystems za Java EE (Enterprise Edition) platformo in jo sedaj sponzorira Oracle Corporation. Podprta različica se imenuje Oracle Glassfish Server. To je brezplačna programska oprema, ki se lahko uporablja na osnovi dveh licenc programske opreme: CDDL (Common Development and Distribution License) in GPL (GNU General Public License) [9]. GlassFish je referenčno izvajanje Java EE in kot podpira Enterprise JavaBeans, JPA (Java Persistence API), JavaServer Faces, JMS (Java Message Service), RMI (Remote Method Invocation), JavaServer Pages, itd. To omogoča razvijalcem ustvarjanje poslovnih aplikacij, ki so prilagodljive in enostavno prenosljive.

### 3 PROCES SKLADIŠČENJA

Skladišče je sestavni del vsake oskrbovalne verige proizvodov in predstavlja pomembno področje v gospodarjenju vsakega podjetja [2]. Njegova glavna vloga je vzdrževanje zalog, ki vključujejo nadzor izdelkov v procesih skladiščenja. Najpomembnejši so prevzem (sprejem), hranjenje in oddaja (izdaja) blaga. Ne glede na velikost skladišča, uporabo avtomatizacije in informacijskih tehnologij je osnovni postopek oz. potek procesov enak. Celoten proces predstavlja skupek vseh dogodkov, ki so potrebni za zahtevano manipulacijo blaga od prevzema pa vse do izdaje le tega. Cilj je hranjenje blaga, brez kakršnegakoli spreminjanja njegovih lastnosti, z namenom, da se izniči časovne razlike med potrošnjo ter proizvodnjo. Postopki skladiščenja so:

- **Sprejem blaga** – predstavlja začetni proces. Ko proizvajalec prejme naročilo od nekega podjetja, ga potrdi in začne s pripravo blaga. Ob zaključku priprave, se blago primerno označi in izroči prevozniku. Izdelke, ki so pripeljeni v skladišče s transportnim vozilom, je potrebno ob prihodu identificirati oz. preveriti pripadajoče dokumente. Prevzem se vrši na podlagi naročilnic, ki vsebujejo seznam vseh izdelkov in vse druge potrebne informacije za posamezen izdelek. Preveri se število izdelkov in ugotovlja se ali so sprejeti izdelki pravi. Uporabljata se dve vrsti prevzema, in sicer grobi prevzem in podrobni prevzem. Pri grobem se preverja številčno stanje prispelega blaga ali se s tehtanjem primerja dejansko stanje s tistim, ki je zapisano v naročilnicah. Pri podrobnem prevzemu se poleg količinskega ujemanja preverja tudi kakovost.
- **Hranjenje blaga** – je zelo pomemben del skladiščnega poslovanja. Po sprejemu je potrebno podatke o sprejetem blagu shraniti v lastne evidence, kar omogoča dober nadzor dejanskega stanja zalog. Dober nadzor in urejenost skladišča pripomoreta k hitrejšemu in pravilnejšemu delovanju ter močno zmanjšata možnost napak ali blagovne izgube. Razporejanje je odvisno od velikosti, vrste in karakteristike blaga. Pravilna razporeditev po skupinah, podskupinah in posameznih elementih še izboljša preglednost skladišča. Izogibati se je potrebno tudi prekomerni zasedenosti skladišča, saj se s tem oteži dostop do posameznih izdelkov.
- **Izdaja blaga** – predstavlja končni proces. Gre za obraten proces kot pri sprejemu blaga. Dobro delovanje skladišča se tako obrestuje pri izdaji, saj je omogočena popolna kontrola nad skladiščenim blagom. Izdelki, ki čakajo na izdajo se preverijo v izdajnici in jih je potrebno naložiti na transportno vozilo, ki jih nato dostavi naročniku.

Potrebno je paziti, da je transportno vozilo pravilno in enakomerno naloženo, zato je pomembno, da se pri nakladanju blaga upošteva informacije o posameznem izdelku.

Večja podjetja imajo običajno kompleksna skladišča z velikimi zalogami, ki so tudi finančno bolj likvidna in so večinoma že avtomatizirana ter imajo celovit informacijski sistem v katerem je vključen tudi proces skladiščenja. Za identifikacijo izdelkov uporabljajo različne metode označevanja, kot so črna koda, QR (Quick Response) in nalepke RFID.

Številna manjša, najpogosteje družinska podjetja pa večinoma še vedno uporabljajo papirnato dokumentacijo s katero skrbijo za beleženje zalog, naročil in tudi prodajo. Ta problem je znatno vplival na odločitev, da smo se osredotočili na razvoj aplikacije, ki bi ustrezala tudi manjšim skladiščem in jim omogočila informatizacijo procesov z uporabo sodobnih tehnologij. Ker ima danes vsak pametni telefon možnost internetne povezave (preko brezžičnih povezav ali mobilnega omrežja), lahko kadarkoli beremo in shranjujemo vse podatke o poteku skladiščenja posameznega izdelka, ki ga hranimo v podatkovni bazi. Tako bi z uporabo standardne opreme, kot je pametni telefon, ki vsebuje tehnologijo NFC, poskrbeli za zamenjavo papirne dokumentacije ter enostavnejšo identifikacijo z uporabo značk NFC. Ker vse poteka avtomatsko, je proces hiter in na nek način podoben uporabi črtne kode, kjer je potrebno pri preverjanju skenirati posamezne izdelke. Poleg tega se lahko izognemo zamudnemu pregledu papirne dokumentacije in napakam, ki se pri pregledih pojavljajo.

### 3.1 INFORMATIZACIJA PROCESA SKLADIŠČENJA

Zelo pomemben del procesa skladiščenja je nadzor nad izdelki, za kar je potrebna dobra identifikacija. Najobičajnejša je uporaba različnih vrst označevanja izdelkov (črna koda, QR), ki vključujejo potrebne podatke o samem izdelku, kot je rok trajanja in drugo. V povezavi z razvojem mobilne aplikacije za uporabo pametnih telefonov, ki imajo vgrajen modul NFC, pa je najprimerneje izbrati tudi uporabo značk NFC.

Značka NFC, ki je prirejena za posamezen izdelek, vsebuje sporočilo NDEF z zapisom URI (Uniform Resource Identifier) (»U«), ki je tipa 'NFC Forum Well-Known', s podatkovnim poljem (ang. payload) oblike urn:epc:id:sgtin:»epc«. Oznaka sgtin določa uporabo splošnega načina označevanja izdelkov, ki ga določa standard GS1. Oznaka »epc« pa je zapis, ki se nanaša na kodo EPC (Electronic Product Code). Ta predstavlja identifikator posameznega izdelka in se nahaja v seznamu izdelkov v naročilnici.

Za izvedbo in testiranje naše aplikacije smo uporabili značke ICODE SLI (SL2ICS20) z velikostjo 106 bitov spomina in ustrezajo standardu ISO/IEC 15693-2 in ISO/IEC 15693-3 ter povezavo z Android, v obliki `android.nfc.tech.Ndef` in `android.nfc.tech.NfcV` [14].

Za delo v skladišču se uporabljata naročilnica in izdajnica. To sta seznama posameznih izdelkov, ki jih je potrebno sprejeti oz. izdati za določen dan. Zapisani sta v obliki datoteke XML, ki vsebuje informacije o strankah in o izdelkih. Shranjeni sta na strežniku do katerega dostopa mobilna aplikacija.

Za testiranje smo uporabili podatke, ki so bili definirani v pilotni izvedbi sistema sledljivosti RFID-F2F [8]. Za vsako stranko so podatki označeni kot transakcija (Slika 3.1). Informacije o transakciji in podatki za posamezen izdelek se shranijo v lokalno podatkovno bazo naročilnic oz. izdajnic na napravi. Izpis podatkovnih baz, ki jih uporabljamo v naši aplikaciji oz. njihove tabele, se v obliki datotek tekstovnega formata nahajajo na napravi, v mapi F2F (naročilnice, izdajnice, skladišče). Aplikacija operira s temi podatki in jih izpisuje, kjer je potrebno. Ker se podatki shranijo v lokalno bazo, tako ne potrebujemo več internetne povezave, kar omogoča pravilno delovanje aplikacije tudi v primeru, da se ta povezava prekine. Če je skenirani izdelek pri sprejemu ali izdaji pravilen, kar pomeni, da se EPC izdelka ujema z enim izmed izdelkov v naročilnici oz. izdajnici aplikacija iz baze prebere njegove informacije in prikaže ime, velikost, težo in način priprave.

Datoteka XML vsebuje seznam strank, ki imajo naročeno določeno število izdelkov. Vsaka stranka je predstavljena kot transakcija, ki vključuje oznako dokumenta (`bizTransaction`), ime (`shortName`), informacije o izdelku (`lot`), datum priprave (`date`) in atribut ali je bilo naročilo pripravljeno (`processed`). Vsaka transakcija nato vsebuje enega ali več zabojev oz. izdelkov. Vsak izdelek, ki je določen z uporabo zaboja (`box`), je opemljen s podatki o enolični serijski številki (`boxID`), kategorijo, ki označuje tip izdelka (`category`), tip naročila (`orderType`), količino (`quantity`), številko priprave (`preparationNr`), način priprave (`preparation`), število produktov - v našem primeru rib (`fishNr`), EPC kodo, EPC razred za zapis v bazi (`EPCClass`), informacije o črtni kodi (`barcode`), težo (`weight`), datum (`date`), dodaten komentar in informacijo o posredovanju (`comment`) in potrditev izdelka (`processed`).

```

<transaction>
  <bizTransaction>http://wmrfid.org/farm2fork/383004776/document/00001/1002160</bizTransaction>
  <shortName>Stranka</shortName>
  <lot>11071226102810</lot>
  <date>2012-07-11T11:23:46+02:00</date>
  <processed>true</processed>
  <box>
    <boxID>1005140</boxID>
    <category>2-3</category>
    <orderType>R</orderType>
    <quantity>20</quantity>
    <preparationNr>3</preparationNr>
    <preparation>N</preparation>
    <fishNr>20</fishNr>
    <EPC>300D6D4306800780000F5654</EPC>
    <EPCClass>http://wmrfid.org/farm2fork/383004776/product/0030</EPCClass>
    <barcode>(01)99999999999999(21)99</barcode>
    <weight>5.2</weight>
    <date>2012-07-11T11:22:20+02:00</date>
    <comment>
    <comment>
    <processed>true</processed>
  </box>
  ..... // tu je lahko tudi več zabojev
</transaction>

```

Slika 3.1: Podatki o stranki in izdelkih – zapis XML.

Mobilna aplikacija vsebuje pet različnih podatkovnih baz, ki se zapišejo v obliki datotek na napravi:

- baza naročilnice,
- baza izdelkov z napako ob sprejemu,
- baza skladišča,
- baza izdajnice in
- baza izdelkov z napako ob izdaji.

Baza naročilnic, izdajnic in skladišča hranijo vse sprejete podatke (Slika 3.1) in še dodatne podatke, potrebne za delovanje aplikacije:

- podatek o sprejemu - pove ali je bil izdelek iz naročilnice že sprejet oz. izdan,
- komentar o izdelku - kadar si želi uporabnik shraniti dodatne informacije,
- datum sprejema - pove kdaj je bil izdelek sprejet oz. izdan (datum) in
- čas sprejema - pove kdaj je bil izdelek sprejet oz. izdan (čas).

V bazi za izdelke z napako se hrani le EPC izdelka ter njegov opis, saj EPC izdelka ni bil najden v naročilnici ali izdajnici in tako aplikacija ne more dostopati do ostalih informacij o tem izdelku.

**Baza naročilnice** - vsebuje izdelke, ki morajo biti sprejeti z vsemi njihovimi informacijami. Ob sprejetju posameznega izdelka se baza modificira, tako, da se sprejeti izdelek primerno označi. Tako imamo nadzor nad vsemi že sprejetimi izdelki in tistimi, ki jih še moramo sprejeti, kar se prikaže v naši aplikaciji.

**Baza izdelkov z napako ob sprejemu** - vsebuje izdelke (EPC kode), ki so v aktivnosti sprejema obarvani rdeče, kar prikazuje napako izdelka. Razlog za napako je lahko v tem, da naročilnica ne obstaja ali je ni bilo mogoče sprejeti, ker uporabniku ni uspelo vzpostaviti internetne povezave. Zaradi tega ni mogoče ugotoviti ali je trenutni izdelek ustrezen. Ta podatkovna baza omogoča shranjevanje izdelkov in jih prikazuje dokler jih uporabnik ne izbriše (ugotovi, da je trenutni izdelek nepravilen in ga ne smemo sprejeti) ali uspe sprejeti ustrezno naročilnico, čemur sledi uspešen sprejem izdelka.

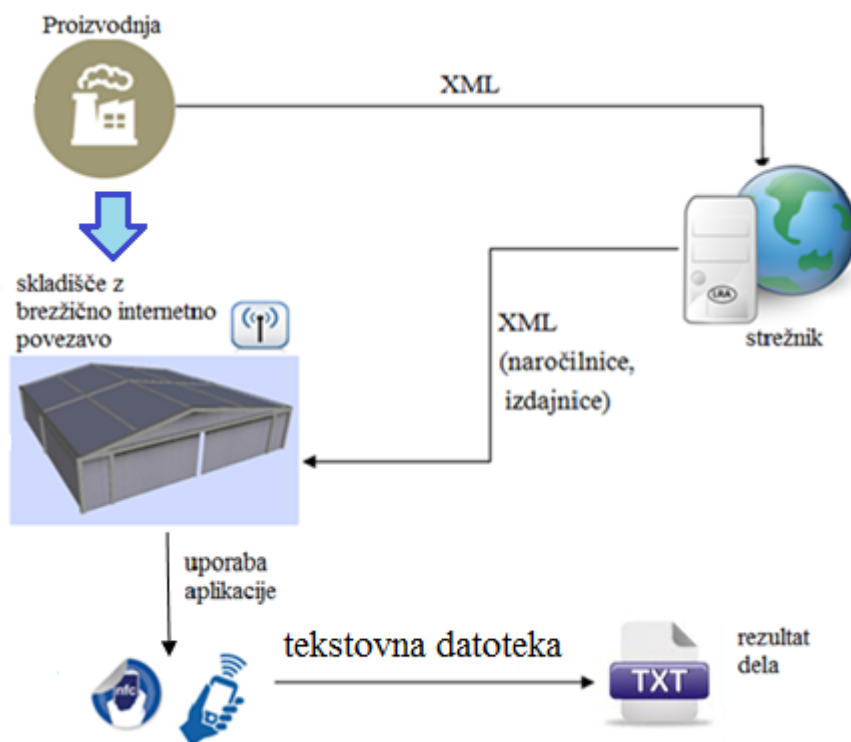
**Baza skladišča** – vsebuje izdelke, ki so trenutno hranjeni v skladišču. V to podatkovno bazo se dodaja izdelke in njihove informacije ob sprejemu in se jih iz nje briše ob izdaji. V skladišču je poleg imena naročnika izdelka vidna tudi teža, velikost in način priprave.

**Baza izdajnice in baza izdelkov z napako ob izdaji** - sta podobni bazi naročilnice in bazi izdelkov z napako ob sprejemu. Razlikuje se le v tem, da delujeta za nasprotna procesa delovanja, to je za izdajanje.

## 3.2 ARHITEKTURA SISTEMA

V procesu skladiščenja je za uspešno obdelavo potrebno zagotoviti komunikacijo s strežnikom. Slika 3.2 predstavlja arhitekturo sistema z uporabo brezžičnega internetnega omrežja v skladišču. V tem primeru bo zagotovljeno sprotno preverjanje podatkov o naročilnicah in izdajnicah na strežniku.

V proizvodnji se generira datoteka XML s podatki o izdelkih, ki bodo poslani v skladišče. Prav tako se vsak izdelek označi z nalepko NFC, ki vsebuje kodo EPC. Za testiranje so podatki na voljo na strežniku v LRA, Fakultete za računalništvo in informatiko. V skladišču nato uporabnik z mobilno aplikacijo avtomatsko sproži klic storitve s poznanim URL naslovom, ki preko brezžične ali mobilne povezave dostopa do podatkov o naročilnicah in izdajnicah. Ko izdelki prispejo do skladišča, se opravi sprejem, to je skeniranje značk NFC in preverjanje z naročilnico. V primeru izdaje so izdelki, ki so označeni z značkami NFC, že v skladišču. Podatki o sprejetih in izdanih izdelkih se zapišejo v tekstovne datoteke o sprejetih oz. izdanih izdelkih in skladišču.



Slika 3.2: Arhitektura sistema.

V primeru, da skladišče nima brezžičnega omrežja, je potrebno internetno povezavo vzpostaviti preko mobilnega omrežja naprave. Brezžično omrežje je za razliko od mobilnega brezplačno oz. stroške prenosa plača ponudnik omrežja. Preko mobilnega omrežja pa operater zaračuna prenos podatkov uporabniku naprave.

## 4 NAČRTOVANJE IN IZDELAVA APLIKACIJE

Mobilna aplikacija za informatizacijo skladišča je namenjena uporabi v manjših skladiščih, kjer bi v obstoječih procesih lahko nadomestili predvsem ročno pregledovanje in preverjanje proizvodov na osnovi papirnate dokumentacije v obliki naročilnic oz. izdajnic. Za predstavljeno rešitev smo uporabili pametne naprave, ki imajo vključeno tehnologijo NFC in delujejo v operacijskem sistemu Android. Naprave so zelo razširjene in omogočajo identifikacijo izdelkov, če so le ti označeni z nalepkami NFC že v času priprave ali pakiranja predno se pojavijo v logističnih procesih. Prepoznavnost omenjenih naprav je vedno večja in število njihovih uporabnikov se hitro povečuje. Aplikacija podpira naprave Android, ki imajo verzije od Android 4.0 (ICE\_CREAM\_SANDWITCH), ker je samo delovanje povezano z zahtevo po uporabi programskega vmesnika API 14.

### 4.1 PROCES SKLADIŠČENJA

Ob zagonu aplikacije se izvede aktivnost »Osnovni meni«, ki ponuja izbiro procesov skladiščenja in ima še enega ali dva nivoja, odvisno od posamezne izbire. Osnovni meni vsebuje pet možnosti (aktivnosti):

- sprejem naročilnice - izbiro in pregled naročilnice,
- sprejemanje izdelkov in zapis v podatkovno bazo,
- skladiščenje - pregled izdelkov v skladišču,
- sprejem izdajnice - izbiro in pregled izdajnice in
- izdajanje izdelkov in zapis v podatkovno bazo.

*Naročilnice in izdajnice* so datoteke tipa XML, s kodiranjem UTF-8, ki vsebujejo seznam transakcij izdelkov za določen datum, ki jih je potrebno sprejeti oz. izdati. Poleg informacije o atributu EPC, brez katerega sprejem ali izdaja nista mogoča, transakcija obsega še vrsto drugih informacij o posameznem izdelku.

Aplikacija, za sprejem in pregled naročilnic oz. izdajnic prikazuje zgolj ime posamezne transakcije (ime naročnika) in atribut EPC za vsak izdelek v transakciji. To je dovolj, da uporabnik točno ve kateri izdelek je potrebno sprejeti ali izdati. Po sprejemu oz. izdaji izdelkov se ob sprejetih izdelkih prikaže oznaka uspešnega sprejema (zelena kljukica), s čimer izboljšamo pregled nad že sprejetimi izdelki in tistimi, ki še čakajo na sprejem. V primeru, da bi uporabnik že skenirani izdelek pomotoma skeniral ponovno, aplikacija prikaže obestilo, da je izdelek že skeniran.

Če internetne povezave ni mogoče vzpostaviti bo lahko uporabnik nemoteno opravljal delo naprej, čeprav v načinu brez interneta. Aplikacija je zasnovana tako, da zaradi omenjene

težave, res ni mogoče preverjanje skeniranih izdelkov z izdelki v naročilnici oz. izdajnici in jih ne moremo sprejeti v skladišče, vendar se vsi izdelki avtomatično preverijo kasneje, ko se povezava vzpostavi. Do takrat se vsi podatki hranijo v lokalni bazi in so vidni v aktivnosti sprejema, kot je opisano v poglavju 5.4.2.

*Sprejemanje in izdajanje* sta aktivnosti, kjer je logika delovanja enaka. Razlikujeta se po tem, da se pri sprejemu izdelke dodaja (sprejem) v skladišče, pri izdaji pa odstranjuje (izdaja) iz skladišča. Izdelek, ki ga uporabnik želi sprejeti je potrebno skenirati. To se naredi tako, da se z mobilno napravo približa znački NFC na izdelku in prejme njen zapis o izdelku v obliki kode EPC, ki se jo nato primerja s kodami EPC v naročilnici ali izdajnici, odvisno od akcije, ki se jo izvaja.

Vsak novo skenirani izdelek se doda v seznam že skeniranih izdelkov sprejema ali izdaje. Če je izdelek uspešno najden v seznamu naročilnic ali izdajnic, se prikažejo dodatne informacije za posamezen izdelek, ki ga podrobneje opisujejo. To so ime naročnika (shortName), kategorija, ki označuje tip izdelka (category), način priprave (preparation) in teža (weight). Prav tako pa je z dotikom na izdelek možno videti tudi njegov EPC. V primeru, da izdelka ni mogoče najti na seznamu izdelkov naročilnice oz. izdajnice se izpiše zgolj EPC izdelka z rdečo barvo, kar ponazarja napako. Za kakšno napako gre se lahko ugotovi z dotikom na izbrani izdelek in prikaže se nova aktivnost, ki prikazuje opis izdelka. Po potrebi se lahko izbriše z dolgim dotikom na izdelek in potrditvijo izbriša ali z menijsko možnostjo za izbris izdelkov.

V zgornjem desnem kotu je viden števec, ki ponazarja koliko izdelkov je še potrebno sprejeti in koliko jih je uporabnik že uspešno sprejel ( $x/y$ , kjer je  $x$  število uspešno prejetih izdelkov,  $y$  pa število vseh izdelkov, ki morajo biti sprejeti). Za vsak skenirani izdelek, ki je uspešno najden na seznamu naročilnic ali izdajnic se »x« poveča za ena. Ko so željeni izdelki sprejeti ali izdani, aplikacija ob izhodu iz aktivnosti sprejema, izdelke shrani v bazo skladišče oz. jih izbriše iz njega, v primeru izdaje.

*Skladiščenje* omogoča pregled zalog. Vsi sprejeti izdelki so vidni v skladišču vse dokler niso izdani. Skladišče torej predstavlja trenutno stanje hranjenih izdelkov, ki čakajo na izdajo. Izdelki so razporejeni po abecednem vrstnem redu glede na ime naročnika in prikazujejo informacije enako kot pri sprejemu, ko je izdelek uspešno najden v naročilnici. Podatki o zalogi se hranijo v podatkovni bazi oblikovani posebej za skladišče. Z dotikom na izdelek, se odpre opis izdelka, kjer se lahko shrani komentar, opis ali razlago, ki jo uporabnik potrebuje. Ta informacija je na voljo vse dokler se izdelka ne izbriše ali izda.

## 4.2 OBLIKOVANJE IN ZASNOVA APLIKACIJE

Aplikacija je zasnovana na uporabi tehnologije NFC in dostopnosti do strežnika. To pomeni, da je potreben dostop do internetnega omrežja, ki je tako rekoč povsod in ves čas na voljo, prenos podatkov pa je tudi vse cenejši. Proces skladiščenja je razdeljen na pet delov:

- naročilnice,
- sprejem,
- skladišče,
- izdaja ter
- izdajnice

in glede na te aktivnosti je oblikovan tudi osnovni meni, kateremu je dodana še aktivnost za informacije izdelkov (Slika 4.1).

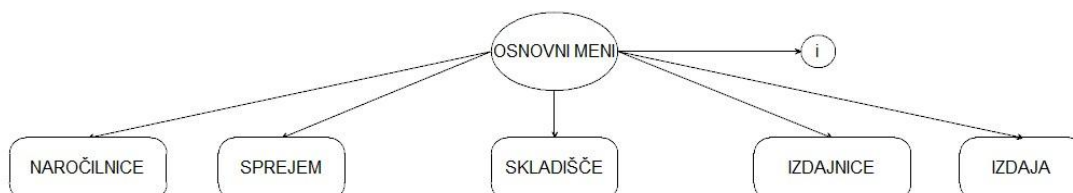
Aplikacija omogoča delovanje v pokončnem načinu (ang. portrait), kar je povezano z velikostjo naprav, saj je ta možnost dosti bolj uporabna za delo v poslovnih procesih. V ležečem načinu (ang. landscape) bi bil celoten grafični vmesnik dokaj nepregleden, kot tudi ne bi koristil pri izpisu daljših seznamov.

Dinamično razporejanje elementov omogoča, da aplikacija izgleda zelo podobno v različnih resolucijah zaslona. Uporablja se lahko na majhnem zaslonu (primer: 480x800) ali zelo velikem (primer: 2560x1600). Aplikacija bo prilagodila velikost elementov tako, da bo zagotovila pravilno razporeditev. Za testiranje aplikacije smo uporabili dva različna pametna telefona in tablični računalnik [15]:

- Samsung Galaxy S4 (resolucija: 1080x1920),
- Nexus S (resolucija 480x800)
- Tablični računalnik, Nexus 10 (resolucija: 2560x1600)

### 4.2.1 OSNOVNI MENI

Osnovni meni predstavlja vstopno stran aplikacije, ki omogoča izbiro aktivnosti. Poleg petih osnovnih aktivnosti je narejena še dodatna aktivnost za informacije o aplikaciji (Slika 4.1).



Slika 4.1: Model osnovnega menija.

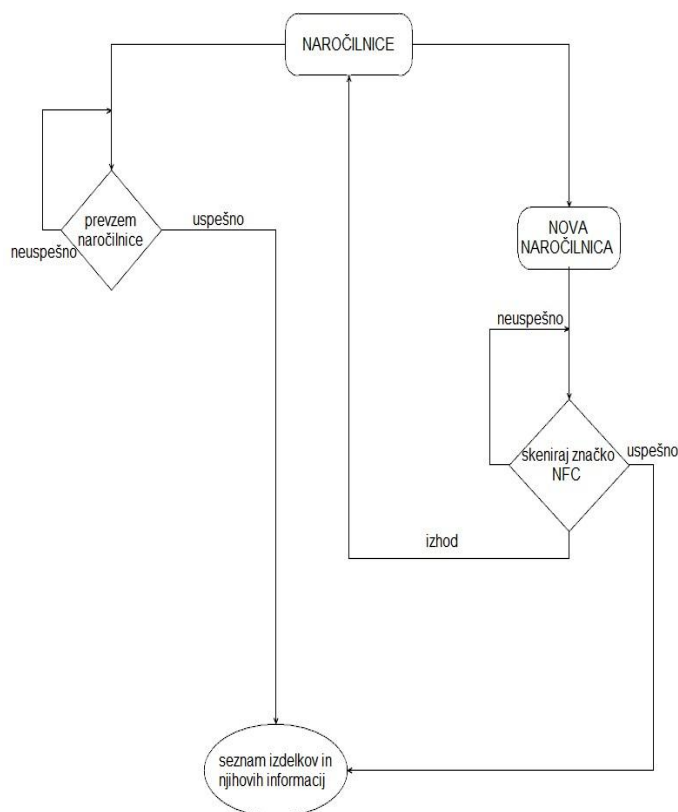
## 4.2.2 INFORMACIJE

Gumb za informacije se nahaja na desni strani naslovne vrstice osnovnega menija. Z dotikom nanj se odpre aktivnost, ki uporabniku nudi:

- osnovne podatke o avtorju aplikacije,
- kontaktne informacije z možnostjo pošiljanja elektronskega sporočila o problemu, ki se lahko pojavi med uporabo aplikacije,
- pomoč za uporabo ter opis aplikacije in
- odpiranje aplikacije za skeniranje QR kode, ki jo lahko najdemo na izdelku. Na njej je zapisan spletni naslov za dostop do spletne strani, ki vsebuje podrobnejši opis izdelka ter dodatne informacije.

## 4.2.3 NAROČILNICE

Aplikacija se po izbiri možnosti Naročilnice poveže s strežnikom in z uporabo spletne storitve (ang. RESTful service) z imenom *F2Flogistika*, ki se nahaja na strežniku, pregleda datoteke naročilnic ter na podlagi datuma izbere ustrezno. Za boljši pregled se celoten seznam naročilnic kot tudi naročilnica, ki se bo uporabljala, preneseta na telefon v obliki datoteke XML (Slika 4.2). Te datoteke se nahajajo v mapi *F2F/narocilnice*. Če je naročilnica uspešno prenešena, se v aplikaciji prikažejo vsi izdelki naročilnice grupirani glede na ime naročnika po abecednem redu. V primeru, da naročilnica ni bila uspešno prenešena (ni možno vzpostaviti internetne povezave) se zažene nova nit, ki v ponavljajoči zanki v intervalu dveh sekund (2s) ponovno poskuša prejeti naročilnico, dokler ni prenos uspešno opravljen. Dodana je možnost uporabe nove naročilnice, ki jo želi uporabnik sprejeti. Seveda mora biti ta naročilnica shranjena na strežniku. V pomoč se ob zagonu aktivnosti »Nova naročilnica« prikaže seznam vseh naročilnic, ki jih je mogoče prenesti. Za prenos željene naročilnice je potrebno skenirati značko NFC, ki ima ustrezen zapis (Narocilnica, dd/MM/yyyy). Po potrditvi nove naročilnice se le ta doda v seznam naročilnic.



Slika 4.2: Uvoz in ogled naročilnic.

#### 4.2.4 SPREJEM IZDELKOV

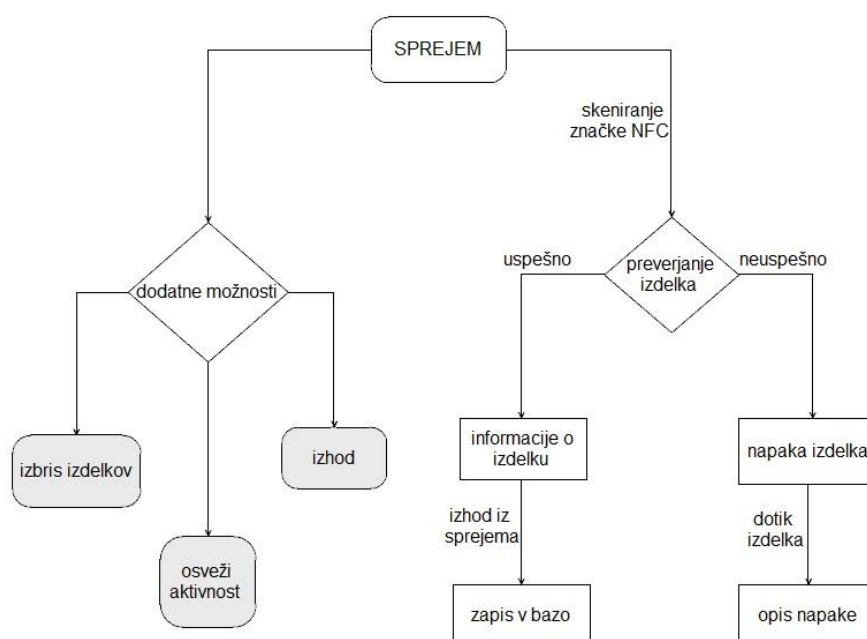
Izdelke, ki čakajo na sprejem, je potrebno identificirati. To se naredi tako, da se skenira značko NFC na izdelku, za katerega se zažene preverjanje v naročilnici. Postopek je prikazan na Slika 4.3 in vključuje dve možnosti:

- ob uspešni identifikaciji se izdelek označi z zeleno kljukico, poleg tega pa se prikažejo še dodatne informacije, ki ga opisujejo. To so ime naročnika, velikost, način priprave in teža. Če uporabnik želi zapisati kakšne opombe za posamezen izdelek, to naredi tako, da se dotakne izbranega izdelka. S tem se zažene nova aktivnost, kjer lahko uporabnik vpiše ustrezno besedilo. Ko je skeniranje vseh željenih izdelkov zaključeno, se s tipko »nazaj« na napravi zapre trenutna aktivnost in se ponovno prikaže osnovni meni. S tem se sprejeti izdelki zapišejo tako v bazo sprejetih izdelkov kot tudi v bazo skladišča, kamor se shranijo.
- ob neuspešni identifikaciji se izdelek označi z rdečo barvo. To pomeni, da gre za napako. Z dotikom na obarvan izdelek, se prikaže opis napake (izdelka ni v naročilnici, naročilnica ne obstaja). Če želi uporabnik k izdelku dodati še kakšen

komentar ali opombo, to zapiše v polje, kjer je opisana napaka, potrdi vnos in se enostavno vrne v aktivnost sprejema. Zapisano besedilo se za posamezen izdelek shrani v bazo nesprejetih izdelkov. Ti ostanejo vidni v aktivnosti sprejema dokler se jih ne izbriše ali sprejme (primer: prejetje ustrezne naročilnice za ta izdelek).

Ob izhodu iz aplikacije se podatki shranijo v:

- F2F/Sprejeto-Izdano - zapišeta se tekstovni datoteki, ki vsebujeta informacije o izdelkih, ki so bili sprejeti oz. izdani.
- F2F/NEsprejeto-Neizdano - zapišeta se tekstovni datoteki, ki vsebujeta informacije o izdelkih, ki še čakajo na sprejem oz. izdajo.



Slika 4.3: Postopek sprejema izdelkov.

Z uporabo osnovnega gumba »meni« na telefonu se ponudijo dodatne možnosti:

- izbris izdelkov,
- osvežitev aktivnosti in
- izhod.

Izbris izdelkov je uporaben za izdelke z napako. Ta možnost se pojavi šele, ko se skenira vsaj en napačen izdelek. Uspešno preverjene izdelke ni možno izbrisati. Ob izbiri te možnosti se pri vsakem izdelku pojavi prazen kvadratik (ang. Checkbox). Če želi uporabnik izdelek izbrisati, pritisne na ta kvadratik in, ko je končal z izbiro potrdi izbris z gumbom »potrdi«. Posamezen izdelek je mogoče izbrisati tudi tako, da se uporabnik dotakne izdelka in dotik pridrži ter potrdi opozorilo za izbris, ki se pojavi.

Osvežitev aktivnosti se uporabi, ko pri prejemanju naročilnice ni mogoče vzpostaviti povezave in posredno tudi ni mogoče prejeti naročilnice, a se uporabnik vseeno odloči nadaljevati s sprejemom. Vsi skenirani izdelki bodo obarvani rdeče oz. bodo označeni kot napaka, saj zanje ni naročilnice in jih tako ni mogoče preveriti. Vendar kot omenjeno v poglavju »Naročilnice« se ob neuspešnem prejemu naročilnice zažene zanka, ki znova in znova poskuša sprejeti naročilnico. Ko ji to uspe (vzpostavljena je internetna povezava) se v obliki kratkega sporočila izpiše, da je bila prebrana nova naročilnica. Uporabnik bo po prejemu informacije lahko zagnal preverjanje vseh skeniranih izdelkov s pritiskom na gumb Osveži in nadaljeval z delom. Pravilno preverjeni izdelki se označijo kot uspešno identificirani.

Izhod – zapre aktivnost sprejema in ponovno prikaže osnovni meni.

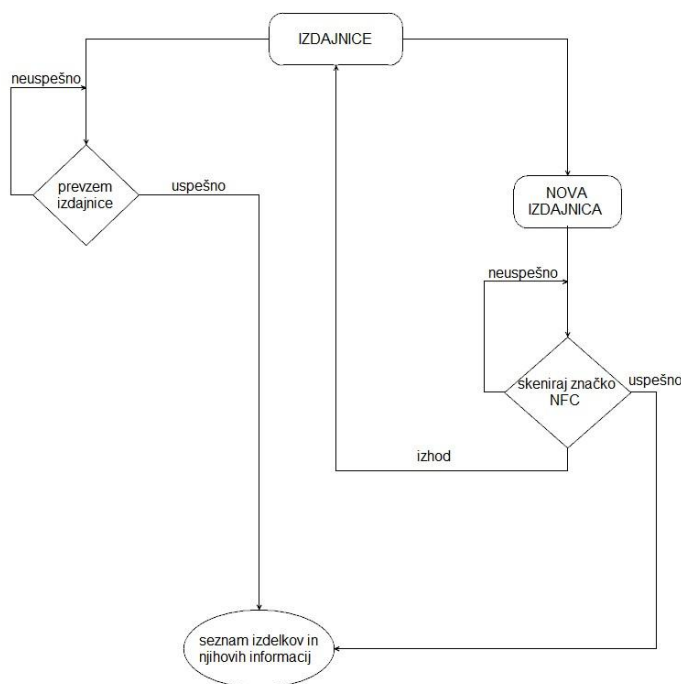
#### 4.2.5 SKLADIŠČE

V tem meniju se prikaže seznam izdelkov, ki so bili sprejeti in se hranijo v skladišču. Informacije o teh izdelkih se nahajajo v bazi skladišča (F2F/Skladisce). Izdelki so razvrščeni po abecednem vrstnem redu strank in prikazujejo osnovne informativne podatke o izdelkih.

#### 4.2.6 IZDAJNICE

Za izdajnice velja podobno kot za naročilnice. V ozadju se izvede sprejem ustrezne izdajnice iz strežnika. Ob neuspešnem prejetju (ni možno vzpostaviti internetne povezave) se zažene nova nit, ki v ponavljajoči zanki znova in znova poskuša uspešno prejeti izdajnico. Ko ji to uspe, izdajnica postane vidna v obliki seznama. Seznam izdajnic in trenutno izdajnico je mogoče najti v mapi F2F/izdajnice (Slika 4.4).

Znotraj aktivnosti izdajnic je ponujena možnost Nova izdajnica, ki se jo vključi takrat kadar uporabnik želi sprejeti izdelke iz druge izdajnice v sistemu (podrobnejši opis je v poglavju 4.2.3).



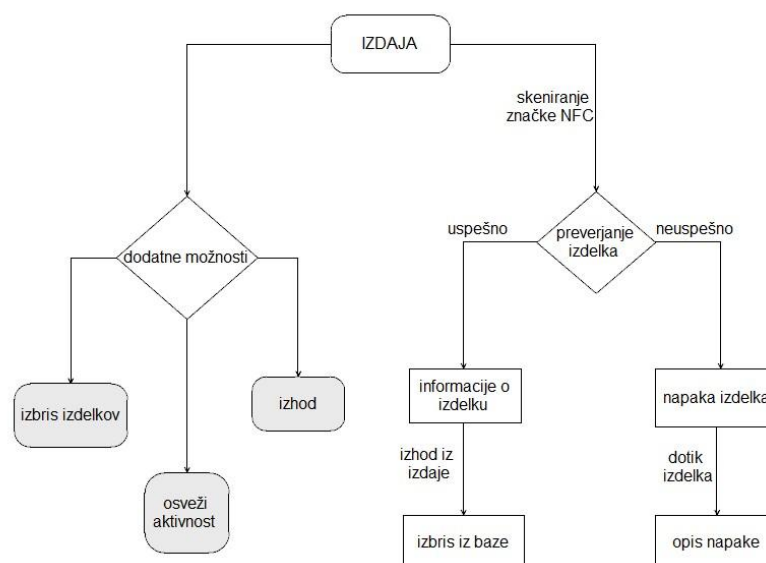
Slika 4.4 Uvoz in ogled izdajnic.

#### 4.2.7 IZDAJA IZDELKOV

Izdaja je podobna sprejemu. Skeniran izdelek se preveri v trenutno prenešeni izdajnici (Slika 4.5).

- Uspešno sprejet izdelek se označi z zeleno kljukico in prikaže še nekatere druge informacije.
- Neuspešno sprejet izdelek se označi z rdečo barvo. Tako se ponazori, da gre za napako. Po kratkem pritisku na tako obarvan izdelek, se prikaže opis napake, kjer je zapisano v čem je problem (izdelka ni v izdajnici, izdajnica ne obstaja).

Ko se zaključi z izdajo in zapusti aktivnost izdaje, se uspešno izdani izdelki zapišejo v podatkovno bazo izdajnic in izbrisejo iz baze skladišča. Neuspešni izdelki ostanejo v aktivnosti izdaje, dokler se jih ne izbriše ali uspešno izda (podrobnejši opis je v poglavju 4.2.4).



Slika 4.5: Aktivnost izdaje.

### 4.3 ZAJEM PODATKOV - STREŽNIK

Datoteke XML, ki vsebujejo podatke za proces skladiščenja so shranjene v mapi (orders) in se nahajajo na strežniku v Laboratoriju za računalniško arhitekturo (UL, FRI). Pridobljeni so bili v pilotni izvedbi sistema sledljivosti v okviru projekta RFID-F2F [8] in bodo uporabljeni za testiranje naše aplikacije.

RESTful storitev je dodatna aplikacija, ki se vseskozi izvaja in se nahaja na strežniku. Zasnovana je tako, da omogoča pregled datotek na strežniku. Uporabniku ponuja dve možnosti:

- pregled zapisa XML z naslovi URL, ki služijo za pregled vsebine posameznih datotek, znotraj mape orders in
- pregled zapisa XML, z vsebino določene datoteke.

Za pregled **zapisa XML z naslovi URL** - storitev dostopa do mape orders, kjer so hranjene datoteke naročilnic oz. izdajnic. Storitev kreira zapis XML, ki vsebuje naslove URL, s katerimi je možen pregled podatkov za vsako izmed datotek. V tej mapi niso le datoteke vezane na našo aplikacijo, ampak tudi nekatere druge, v povezavi s projektom RFID-F2F (Slika 4.6). Zapis XML, torej prikazuje seznam vseh datotek v mapi orders in je do njega možno dostopati tudi preko internetnih brsklanikov, pod naslovom: <http://lra-f2f.fri.uni-lj.si:8080/orders/resources/order-files>.

Datoteke, ki ustrezajo aplikaciji (naročilnice in izdajnice), imajo v zapisu poleg začetnega http naslova (<http://lra-f2f.fri.uni-lj.si:8080/orders/resources/order-files/>) strežnika še ime, ki ga sestavlja »datum«-PR-weight.xml. Podatek »datum« je v obliki yyMMdd formata, kjer yy predstavlja zadnji dve številki leta (npr. 2014→14), MM predstavlja mesec (npr. januar→01, november→10) in dd predstavlja dan (npr. 01, 25, 30). Aplikacija zapis XML shrani v datoteko, imenovano F2F\narocilnice\seznam-narocilnic.xml oz. F2F\izdajnice\seznam-izdajnic.xml.

```
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/120608-PR-weight.xml
</file>
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/120518-145120-DEATH-event-O.xml
</file>
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/120518-150609-FEEDING-event-O.xml
</file>
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/130508-PR-weight.xml
</file>
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/120622-144516-FEEDING-event-O.xml
</file>
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/120623-090132-FISHNET-event-O.xml
</file>
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/120622-181434-DEATH-event-O.xml
</file>
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/120613-PR-weight.xml
</file>
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/120103-OR.xml
</file>
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/120126-POSTPR-weight.xml
</file>
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/120622-175231-FEEDING-event-O.xml
</file>
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/120711-123631-PR30-event-OQTO.xml
</file>
▼<file>
  http://lra-f2f.fri.uni-lj.si:8080/orders/resources/crder-files/120627-PR-weight.xml
</file>
```

Slika 4.6: Zapis XML z naslovi URL nekaterih datotek v mapi orders.

Za pregled **zapisa XML z vsebino določene datoteke** - storitev dostopa do potrebne datoteke znotraj mape orders in prikaže njeno vsebino, do katere je prav tako možen dostop preko internetnih brskalnikov. Slika 4.7 prikazuje podatke za stranko, z enim izdelkom, ki se nahaja v datoteki <http://lra-f2f.fri.uni-lj.si:8080/orders/resources/order-files/120711-PR-weight.xml>. Naslov URL, ki ga je potrebno uporabiti za pregled posamezne datoteke, se nahaja v seznamu datotek z naslovi URL (Slika 4.6). Aplikacija za skladiščenje prebere zapis XML tiste datoteke (naročilnica oz.izdajnica), ki jo trenutno potrebuje in ga zapiše v svojo datoteko XML, ki jo shrani na napravo. Ta datoteko se nahaja v mapi F2F/narocilnice oz. F2F/izdajnice.

```
▼<transactions>
  ▼<transaction>
    ▼<bizTransaction>
      http://wmrfid.org/farm2fork/383004776/document/00001/1002189
    </bizTransaction>
    <shortName>CASA DEL SAL</shortName>
    <lot>11071226102810</lot>
    <date>2012-07-11T11:21:03+02:00</date>
    <processed>true</processed>
    ▼<box>
      <boxID>1005229</boxID>
      <category>3-4</category>
      <orderType>Z</orderType>
      <quantity>1</quantity>
      <preparationNr>1</preparationNr>
      <preparation>OČ</preparation>
      <fishNr>14</fishNr>
      <EPC>300D6D4306800A40000F56AD</EPC>
      <EPCClass>http://wmrfid.org/farm2fork/383004776/product/0041</EPCClass>
      <barcode>(01)9999999999999999(21)99</barcode>
      <weight>4.4</weight>
      <date>2012-07-11T11:21:03+02:00</date>
      <comment></comment>
      <processed>true</processed>
    </box>
  </transaction>
</transactions>
```

Slika 4.7: Zapis XML z vsebino datoteke 120711-PR-weight.xml.



## 5 UPORABA APLIKACIJE SKLADIŠČENJA

V tem poglavju je predstavljena konkretna uporaba programa s prikazom in razlago posameznih procesov skladiščenja. Predstavljene so tudi ostale funkcionalnosti, ki jih ponuja aplikacija. Za testiranje smo uporabili datoteko XML z oznako Narocilnica, 11/07/2012, ki vsebuje podatke za 32 strank in 72 izdelkov. Za preverjanje sprejema in izdaje smo uporabili 10 nalepk NFC, v katere smo zapisali EPC kode izdelkov iz uporabljenega seznama.



### 5.1 OSNOVNI MENI IN INFORMACIJE

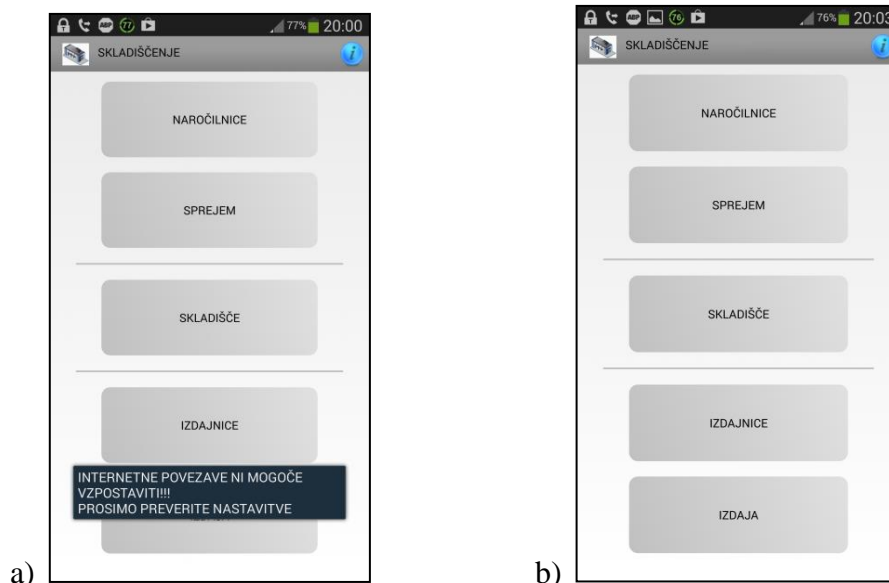
Uporabnik mora imeti nameščeno aplikacijo Skladiščenje na svoji mobilni napravi. Zagon aplikacije je možen na dva načina:

1. Z administratorsko kartico NFC, kjer se preverja ali je uporabniku dovoljena uporaba aplikacije. Za kreiranje administratorske kartice je potrebno z našim programom za urejanje značk v polje za vpis EPCjev zapisati tekst »XADMINISTRX«.
2. S klikom na ikono, ki zažene osnovni meni. Tukaj predpostavimo, da je uporabniku dovoljena uporaba aplikacije in ne zaženemo nobenega preverjanja pravic.

V nadaljevanju je prikazano delovanje za zagon aplikacije z ikono.

Po dotiku ikone se prikaže osnovni meni. Če ni mogoče vzpostaviti internetne povezave, se pojavi opozorilo (Slika 5.1(a)). Slika 5.1 prikazuje osnovni meni z vsemi funkcionalnostmi in naslovno vrstico, ki vključuje:

- sliko (programska ikona) - ,
- ime aplikacije – SKLADIŠČENJE,
- gumb za prikaz informacij ter pomoč uporabnikom - .



Slika 5.1: Osnovni meni brez internetne povezave (a); internetna povezava (b).

Z izbiro gumba za informacije se prikaže novo okno, v katerem so štirje zavihki:

- zavihek z osnovnimi informacijami (Slika 5.2(a)),
- zavihek za kontakt s proizvajalcem ali skrbnikom sistema, kjer lahko pošljemo željeno elektronsko sporočilo s problemom ali vprašanjem, (Slika 5.2(b)),
- opis in pomoč, kjer je na kratko opisan program (Slika 5.3(a)) in
- skeniranje QR kode – alternativna možnost branja dodatnih informacij o izdelku (Slika 5.3(b), Slika 5.4(a), Slika 5.4(b)), ki prebere naslov URL in ga izpiše v naši aplikaciji. S klikom nanj se nam v primeru, da imamo vzpostavljeno povezavo odpre spletna stran z dodatnimi informacijami o izdelku.



Slika 5.2: Osnovne informacije aplikacije (a); kontakt s proizvajalcem (b).



Slika 5.3: Pomoč in opis uporabe aplikacije (a) in začetni meni za skeniranje črtne kode (b).

V primeru, da Android naprava nima ustreznega programa za skeniranje črtne kode, se nam pojavi obvestilo ter navodila postopka za namestitev ustreznega programa in uspešno nadaljevanje skeniranja QR kode.



Slika 5.4: Proces skeniranja QR kode (a) in prikaz po njem (b).

Izhod iz informacij in vrnitev v osnovni meni se izvede s tipko »nazaj« (ang. back button).

## 5.2 NAROČILNICE IN SPREJEM

Naročilnico, ki jo je potrebno prejeti, je vezana na datum. To pomeni, da je predvideno, da je za vsak dan svoja naročilnica. Za testiranje smo izbrali eno izmed naročilnic in jo definirali kot današnjo (120711-PR-weight.xml), ker na strežniku nimamo na voljo vsak dan novih naročilnic.

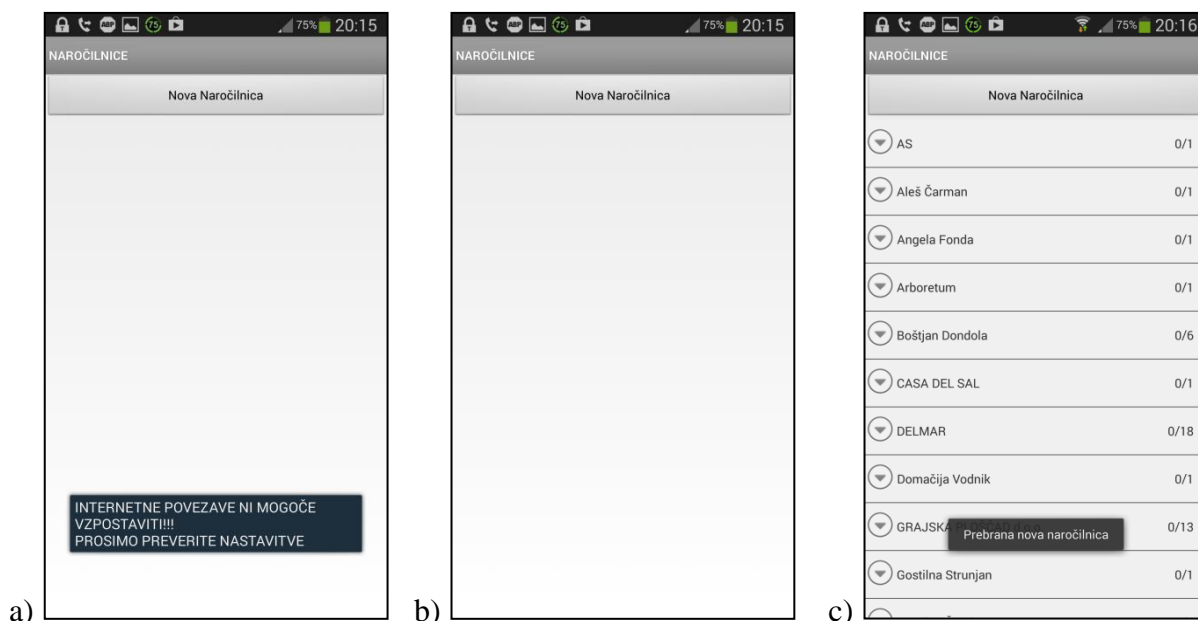
### 5.2.1 NAROČILNICE

Ob pritisku na gumb »Naročilnice« se prične proces sprejema blaga. Pojavi se nova aktivnost, ki vsebuje gumb »Nova Naročilnica«. Ali bo mogoče ustrezno naročilnico sprejeti je odvisno od tega ali naročilnica na strežniku obstaja in ali je vzpostavljena internetna povezava. Če naročilnica ne obstaja ni mogoče narediti ničesar, razen, da se obvesti proizvajalca o problemu neobstoječe naročilnice. V kolikor predpostavimo, da do te napake ne sme priti, ostane le še problem internetne povezave.

- *Internetne povezave ni* (Slika 5.5(a)) - pojavi se obvestilo, ki opozarja, naj uporabnik preveri ali ima vključene vse potrebne nastavitve za pravilen prenos podatkov. Čeprav ni internetne povezave je možno nadaljevati z delom, saj se bodo skenirani izdelki preverili takoj, ko bo uporabniku uspelo vzpostaviti povezavo. Do takrat, bodo vsi izdelki označeni

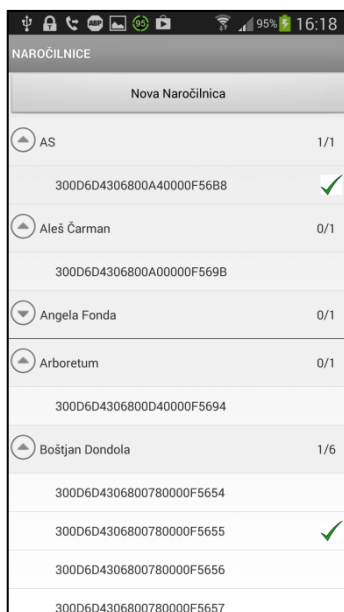
z rdečo barvo, ki ponazarja napako (ni naročilnice, kar pomeni, da izdelka ne moremo preveriti) (Slika 5.8(a)).

- *Internetna povezava je* (Slika 5.5(b)) - ob pojavu aktivnosti se prikaže tudi seznam izdelkov, ki je urejen po abecednem vrstnem redu naročnikov. Na prvi pogled, torej vidimo le seznam vseh naročnikov (Slika 5.5(c)), vendar se nam s pritiskom na poljubnega naročnika v spustnem meniju prikaže tudi EPC koda vsakega izmed izdelkov, ki so v naročilnici (Slika 5.6).



Slika 5.5: Aktivnost naročilnic brez internetne povezave (a), z internetno povezavo (b) in po prejemu naročilnice (c).

Po sprejemu izdelkov v aktivnosti sprejema je na seznamu v aktivnosti naročilnic, na desni strani pri naročnikih, vidno koliko izdelkov je uporabnik že sprejel in koliko je vseh, ki so bili naročeni. V spustnem meniju (Slika 5.6) se na desni strani kode EPC pojavi zelena kljukica, ki označuje, da je bil izdelek že sprejet.



Slika 5.6: Prikaz uspešno sprejete izdelka, vidnega v aktivnosti naročilnic.

Gumb »Nova Naročilnica« služi v primeru, da uporabnik želi sprejeti naročilnico, ki ne ustreza današnjemu datumu. V tem primeru mora imeti značko NFC, ki vsebuje ustrezno ime naročilnice. V primeru, da želi sprejeti naročilnico za dan 10.05.2012, mora značka NFC vsebovati tekstovni zapis: »Narocilnica, 10/05/2012«, ki jo nato aplikacija poišče v sistemu in doda v seznam naročil.

## 5.2.2 SPREJEM

Če pri sprejemu naročilnic ni bilo težav z internetno povezavo in je bila naročilnica uspešno sprejeta, se lahko v aktivnosti sprejema, na desni strani zgoraj, opazi števec, ki kaže koliko je vseh izdelkov, ki so v naročilnici in koliko je že sprejetih (Slika 5.8).

V primeru, da naročilnica ni bila uspešno sprejeta, (zaradi težav z internetno povezavo ali naročilnice ni bilo mogoče najti) kaže števec na desni strani »0/0«, kar pomeni, da ni nobenega izdelka, ki čaka na sprejem (oz. ni naročilnice). Kljub problemu je možno izvesti skeniranje značke NFC. Na seznam sprejema se doda EPC koda skeniranega izdelka, rdeče barve, ki opozarja na pojav napake (Slika 5.8(a)).



Slika 5.7: Aktivnost sprejema pri uspešnem prejemu naročilnice.

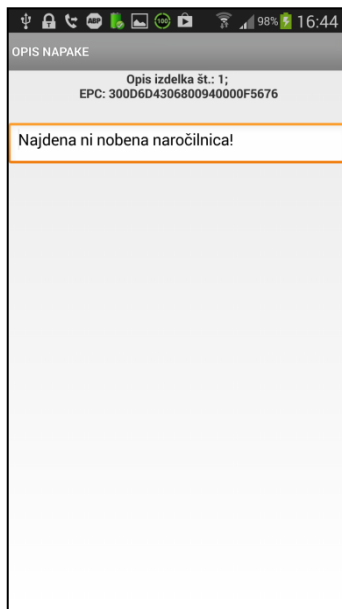
Takoj, ko se internetna povezava vzpostavi, se, v primeru ustreznega preverjanja posameznega izdelka z naročilnico, namesto zapisa EPC pojavi ime naročnika, velikost, teža in način priprave izdelka, na desni strani pa zelena kljukica (Slika 5.8(b)).



Slika 5.8: Aktivnost sprejema brez internetne povezave (a) in po uspešni vzpostavitvi internetne povezave in uspešnem preverjanju izdelkov (b).

S kratkim dotikom na kodo EPC izdelka se v primeru napake oz. dotikom na uspešno preverjen izdelek odpre aktivnost opisa izdelka. Program je zasnovan tako, da je omogočeno

hranjenje in spreminjanje opisa izdelka (komentarji, opombe, ipd.) skozi celoten proces skladiščenja za določen izdelek. Polje z opisom se imenuje »EditText« in omogoča vnos oz. popravek besedila ali vstavljanje komentarja, ki bi ga lahko kasneje koristili (Slika 5.9).



Slika 5.9: Opis napake po kratkem dotiku na izbrani izdelek.

Ko je uporabnik sprejel vse željene izdelke, se s tipko »nazaj« vrne v osnovni meni. V tem trenutku se prejšnja aktivnost, v našem primeru aktivnost »Sprejem«, zaključi in s tem shrani vse podatke oz. izdelke v bazo. V primeru, da uporabnika zmoti, na primer telefonski klic, se podatki vseeno shranijo oz. aplikacija ostane odprta in lahko z njo nadaljuje po končanem klicu.

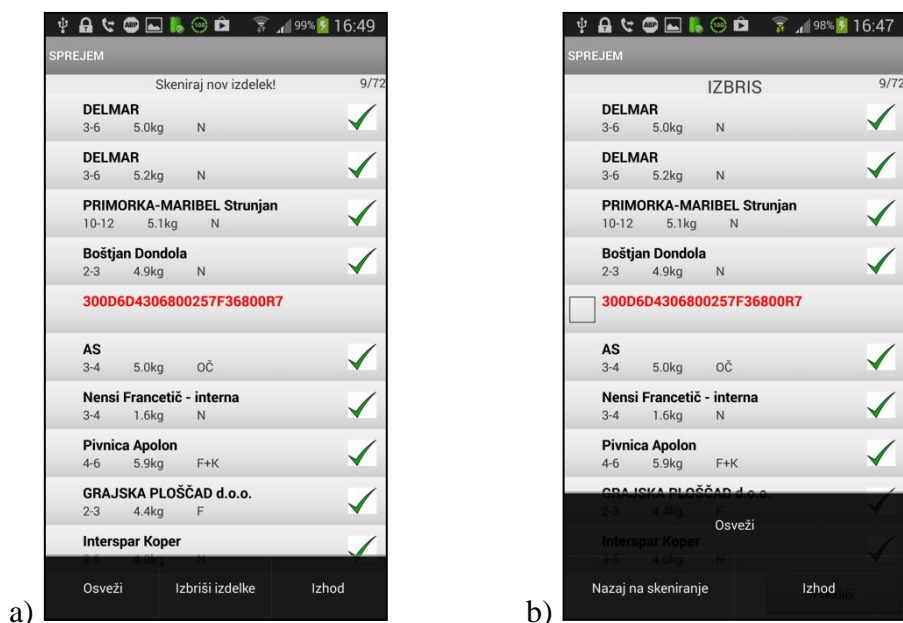
Nekoliko daljši dotik izdelka nam omogoča njegov izbris, ki je možen le v primeru izdelka z napako (rdeče barve). Za varnost in nezaželjeno brisanje poskrbi tako imenovan dialog, z vprašanjem o gotovosti namere. Brisanje izdelka z napako pride prav v primeru, ko uporabnik ugotovi, da je po pomoti skeniral izdelek, ki ga sicer ne bi smeli (Slika 5.10).



Slika 5.10: Dialog za brisanje izbranega izdelka po dolgem kliku nanj.

V aktivnosti sprejema kot tudi izdaje je mogoče s tipko za meni na napravi odpreti še tri dodatne možnosti (Slika 5.11(a)):

- Prva možnost je gumb Izhod, katerega naloga je vrnitev v prejšnjo aktivnost, enako kot pri pritisku na tipko »nazaj« na napravi.
- Druga in bolj zanimiva možnost je gumb Izbriši izdelke (Slika 5.11(b)). Čeprav smo malo prej govorili o izbrisu posameznega izdelka z dolgim dotikom nanj, je neugodno, če je do napake prišlo pri večjem številu izdelkov. To bi pomenilo, da bi moral uporabnik z dolgim dotikom izbrisati vsak izdelek posebej. V izogib tej neprijetnosti je razvita možnost označevanja. Na levi strani pred imenom se pojavi majhen prazen kvadrček na katerega je potrebno pritisniti, če želimo izdelek označiti za izbris. Ko uporabnik z označevanjem zaključi, mora za uspešen izbris pritisniti na gumb »Izbriši«, ki se pojavi takoj, ko je izbran vsaj en izdelek ali na gumb »Prekliči«, če si premisli in izdelkov ne želi izbrisati.



Slika 5.11: Gumb za meni ter ponujena izbira (a) in izbris izdelkov (b).

- Tretja možnost je gumb Osveži. Z njim se lahko preveri izdelke z napako, ne da bi bilo potrebno zapustiti aktivnost sprejema. Na ta način, pravilno preverjeni izdelki uporabniku ne bodo več vidni in se bodo shranili v bazo skladišča. Za nepravilne izdelke se bo zagnalo ponovno preverjanje z naročilnico in se bodo ponovno prikazali v aktivnosti sprejema. Uspešno preverjeni izdelki bodo v uspešno preverjenem načinu in ponovno neuspešno preverjeni v nespremenjenem načinu (z napako).

V primeru, da internetne povezave ni mogoče vzpostaviti (naročilnica ni bila sprejeta), a se uporabnik odloči nadaljevati z delom bodo vsi skenirani izdelki še vedno vsebovali napako, saj preverjanje ne bo mogoče. V kolikor se med procesom skeniranja izdelkov uspe vzpostaviti internetno povezavo, aplikacija to sporoči v obliki kratkega sporočila. S pritiskom na gumb Osveži se sproži preverjanje vseh skeniranih izdelkov. Pomembno je, da se uporabnik zaveda, da v takem primeru z enim pritiskom na osveževanje zažene preverjanje in tako ugotovi kateri izdelki so pravilni in šele nato lahko s ponovnim pritiskom na isti gumb ali z gumbom Izhod oz. tipko nazaj na napravi dejansko sprejme uspešno preverjene izdelke.

### 5.3 SKLADIŠČE

Z izbiro gumba »Skladišče« se odpre informativna aktivnost. To je prikaz izdelkov, ki se trenutno hranijo, po abecednem redu naročnika. V skladišče se izdelki shranijo po uspešnem preverjanju v aktivnosti sprejema in se izbrišejo po uspešnem preverjanju izdelkov z izdajnico. Kljub temu, da podatkovna baza skladišča, iz katere se črpa podatke o izdelkih, ki se hranijo v skladišču, vsebuje vse podatke iz naročilnice, so prikazani le podatki o imenu naročnika, velikosti, teži in načinu priprave posameznega izdelka (Slika 5.12).



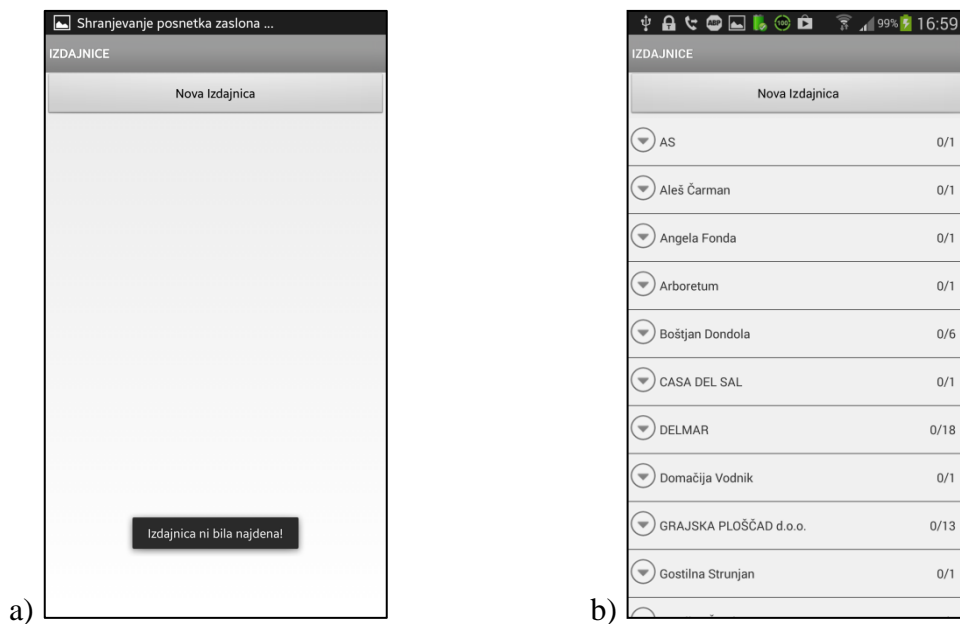
SKLADIŠČE		
V skladišču imamo:		
<b>AS</b>		
3-4	5.0kg	OČ
<b>Boštjan Dondola</b>		
2-3	4.9kg	N
<b>DELMAR</b>		
3-6	5.0kg	N
<b>DELMAR</b>		
3-6	5.2kg	N
<b>GRAJSKA PLOŠČAD d.o.o.</b>		
2-3	4.4kg	F
<b>Interspar Koper</b>		
3-6	4.0kg	N
<b>PRIMORKA-MARIBEL Strunjan</b>		
10-12	5.1kg	N
<b>Pivnica Apolon</b>		
4-6	5.9kg	F+K

Slika 5.12: Prikaz aktivnosti skladišča ter njegove vsebine.

Prav tako kot pri sprejemu je tudi pri skladiščenju možen pregled in spreminjanje opisa izdelkov, z enostavnim kratkim dotikom na izbrani izdelek.

### 5.4 IZDAJNICE TER IZDAJA

Aktivnosti izdajnic in izdaje sta zelo podobni aktivnostima naročilnic in sprejema. V izdajnicah prav tako poteka preverjanje internetne povezave in se generira nova nit v primeru, da le te ni mogoče vzpostaviti. Kot pri sprejemu, je poskrbljeno za avtomatsko kasnejše preverjanje ob vzpostavitvi internetne povezave, prav tako kot so identične vse ostale funkcionalnosti. Edina razlika je, da se pri sprejemu izdelke shranjuje, pri izdaji pa se jih briše iz skladišča. V primeru testiranja aplikacije smo za izdajnico uporabili isto datoteko XML kot pri naročilnici (Slika 5.13).



Slika 5.13: Aktivnost izdajnic z neuspešno (a) in uspešno sprejeto izdajnico (b).

## 6 SKLEPNE UGOTOVITVE

V diplomski nalogi smo predstavili rešitev, ki nam omogoča preprosto in hitro uporabo ter nadzor izdelkov v procesih skladiščenja. Osredotočili smo se predvsem na procese sprejemanja, hranjenja ter izdaje blaga. Razvili smo aplikacijo, za uporabo v sistemu Android. Za identifikacijo blaga smo uporabili tehnologijo NFC, ki je učinkovita, hitra in preprosta za uporabo, hkrati pa kljub napovedim za močno porast uporabe v prihodnosti, danes še ni tako razširjena.

Cilj našega dela je bil prikazati možnost uporabe tehnologije NFC, seveda pa smo zajeli tudi nekaj procesov ter funkcij potrebnih za delovanje celotnega postopka skladiščenja. Naročilo služi za preverjanje ali sprejemanje pravih izdelkov. Sledi hranjenje ustreznih izdelkov v skladišču (podatki se shranijo v ustrezno podatkovno bazo) in jih ob prejemu izdajnice, pred dejansko izdajo zopet preverimo. Za identifikacijo izdelkov skrbi tehnologija NFC. Med celotnim procesom preverjamo internetno povezavo, za dostop do podatkovnih baz. Prikazujejo se naročilnice ter izdajnice, ki so bile prevzete, omogočen je vpogled v seznam izdelkov v skladišču ter ob sprejemanju in izdajanju se prikazuje uspešno in neuspešno skenirane izdelke. Prav tako, je ponujena možnost izbrisa izdelka oz. izdelkov.

Mobilna aplikacija vsebuje osnovne funkcije, ki so dovolj enostavne, da tudi podjetja lahko dobijo vpogled v delovanje in potek vseh procesov. S tem smo izpolnili zahteve in cilje, ki smo si jih zastavili v načrtovanju projekta. Možnosti za razširitev aplikacije in njenih funkcij je še veliko, pri vseh procesih skladiščenja:

- različni načini zagona s pravicami uporabnikov,
- prikaz izdelkov (razvrščanje, iskanje),
- možnost različnih načinov sprejema in izdaje,
- vodenje skladišča (za bolj kompleksno hranjenje),
- prikazi zgodovine hranjenja, sprejemanja in izdaje,
- izvoz podatkov iz podatkovnih baz (glede na potrebe uporabnika).

Na osnovi testiranja smo prišli do naslednjih ugotovitev:

1. V prihodnosti bo imelo vedno večje število pametnih telefonov vgrajeno tehnologijo NFC zato bo njena uporaba vse bolj primerna in zaželjena.

2. Uporaba tehnologije NFC in sistema Android se iz programerskega vidika, rahlo razlikuje od programiranja, ki smo ga navajeni za npr. operacijski sistem Windows. Vendar, ko bolje spoznaš logiko delovanja, omogoča preprosto in enostavno uporabo. Kljub nepoznavanju dela s sistemom Android kot tudi tehnologijo NFC nam je uspelo narediti aplikacijo, ki nudi dovolj funkcij, je enostavna za uporabo in vsebuje lep uporabniški vmesnik, da lahko uporabnik dobi dober občutek o tem, kako deluje.
3. Tehnologija NFC ustreza zahtevam za identifikacijo izdelkov. Povezava je varna, saj deluje s telefonom le na razdalji do kakšnih 5 cm (odvisno od naprave in značke), prav tako je potrebno predhodno aktivirati funkcijo povezave NFC ter imeti aktiven zaslon.
4. Za zahtevnejše uporabnike so možne izboljšave in dodatki, kot so dostop za različne uporabnike, spremembe pravic in dodaten nadzor nad delom. Možno je enostavno preoblikovanje aplikacije za spreminjanje prikaza podatkov o izdelkih, inicializacija razvrščanja in iskanja po seznamu in druge funkcionalnosti, ki bi jih uporabnik potreboval pri vseh postopkih skladiščenja.

## 7 LITERATURA

[1] Vedat Coskun, Kerem Ok, Busra Ozdenizci, »NFC Application Development for Android« Wiley, 2012.

[2] Gregor Rak, »Logistika notranjega transporta in skladiščenja«, Zavod IRC, Ljubljana, 2011, pogl. 5.

[3] (2014) About the Eclipse Foundation. Dostopno na:  
<http://www.eclipse.org/org/>

[4] (2013) All About Windows Phone. Dostopno na:  
[http://allaboutwindowsphone.com/features/item/18498\\_Enhancing\\_your\\_life\\_and\\_busine.php](http://allaboutwindowsphone.com/features/item/18498_Enhancing_your_life_and_busine.php)

[5] (2013) An Introduction to NetBeans. Dostopno na:  
<https://netbeans.org/about/>

[6] (2014) Android, the world's most popular mobile platform. Dostopno na:  
<http://developer.android.com/about/index.html>

[7] (2014) apc: Inside NFC: how near field communication works. Dostopno na:  
<http://apcmag.com/inside-nfc-how-near-field-communication-works.htm>

[8] (2013) F2F Farm to Fork. Dostopno na:  
<http://www.rfid-f2f.eu/>

[9] (2013) GlassFish Server. Dostopno na:  
<https://glassfish.java.net/>

[10] (2011, modificirano 2014) HTC. Dostopno na:  
<http://www.htc.com/us/support/htc-one-sv-boost/howto/416375.html>

[11] (2014) Near Field Communication. Dostopno na:  
<http://developer.android.com/guide/topics/connectivity/nfc/index.html>

[12] (2014) Near Field Communication. Dostopno na:  
<http://www.nearfieldcommunication.org/>

[13] (2013) Near Field Communication Today and Tomorrow. Dostopno na:  
<http://united.softserveinc.com/blogs/mobility/južne-2012/near-field-communication/>

[14] (2002) Data Sheet. Dostopno na:  
[http://www.nxp.com/documents/data\\_sheet/SL064230.pdf](http://www.nxp.com/documents/data_sheet/SL064230.pdf)

[15] (2014) PDADB.net. Dostopno na:  
<http://pdadb.net/>

[16] (2014) RESTful Web services. Dostopno na:  
<http://www.ibm.com/developerworks/webservices/library/ws-restful/>

[17] (2013) Ubitap - What is NFC Tag. Dostopno na:  
<http://www.ubitap.com/whatisnfc>