



Univerza v Mariboru

Fakulteta za elektrotehniko,
računalništvo in informatiko

Uroš Brumen

**MOBILNA APLIKACIJA ZA POMOČ
STAREJŠIM PRI KLICANJU IN PISANJU
SMS-OV**

Diplomsko delo

Maribor, september 2015

MOBILNA APLIKACIJA ZA POMOČ STAREJŠIM PRI KLICANJU IN PISANJU SMS-OV

Diplomsko delo

Študent: Uroš Brumen
Študijski program: Univerzitetni študijski program
Računalništvo in informacijske tehnologije
Mentor: red. prof. dr. Peter Kokol, univ. dipl. inž. el.
Lektorica: Nina Skube, prof. slov.



Univerza v Mariboru

Fakulteta za elektrotehniko,
računalništvo in informatiko

Smetanova ulica 17
2000 Maribor, Slovenija

FERI

Številka: E1003495

Datum in kraj: 25. 08. 2015, Maribor

Na osnovi 330. člena Statuta Univerze v Mariboru (Ur. l. RS, št. 46/2012)
izdajam

SKLEP O DIPLOMSKEM DELU

1. **Urošu Brumnu**, študentu univerzitetnega študijskega programa RAČUNALNIŠTVO IN INFORMACIJSKE TEHNOLOGIJE, se dovoljuje izdelati diplomsko delo.
2. **MENTOR:** red. prof. dr. Peter Kokol
3. **Naslov diplomskega dela:**
MOBILNA APLIKACIJA ZA POMOČ STAREJŠIM PRI KLIČANJU IN PISANJU SMS-OV
4. **Naslov diplomskega dela v angleškem jeziku:**
MOBILE APPLICATION TO HELP OLDER FOR CALLS AND WRITING SMS
5. Diplomsko delo je potrebno izdelati skladno z "Navodili za izdelavo diplomskega dela" in ga oddati v treh izvodih (dva trdo vezana izvoda in en v spiralo vezan izvod) ter en izvod elektronske verzije do 30. 09. 2015 v referatu za študentske zadeve.

Pravni pouk: Zoper ta sklep je možna pritožba na senat članice v roku 3 delovnih dni.

Dekan:

red. prof. dr. Borut Žalik



B. Žalik

Obvestiti:

- kandidata,
- mentorja,
- odložiti v arhiv.

ZAHVALA

Za mentorstvo in strokovno pomoč pri izdelavi diplomske naloge se zahvaljujem mentorju, red. prof. dr. Petru Kokolu.

Še posebej se zahvaljujem svoji družini, torej staršem, ki so mi omogočili študij.

Mobilna aplikacija za pomoč starejšim pri klicanju in pisanju SMS-ov

Ključne besede: mobilna aplikacija, Android, Java

UDK: 654.15:621.395.721.5(043.2)

Povzetek

V tem diplomskem delu smo se ukvarjali z izdelavo mobilne aplikacije za pomoč starejšim pri klicanju in pisanju sporočil SMS. Pregledali smo obstoječe stanje na tem področju in si zadali izdelati aplikacijo, ki bi pomagali starejšim lažje razumeti dokaj kompleksen operacijski sistem Android, prav tako pa bi poenostavila funkcije na najosnovnejšo raven in povečala ikone ter tako izboljšala preglednost osebam z omejitvami, kot je slabovidnost in podobno. Aplikacijo bomo razvili v prosto dostopnem razvojnem okolju Android Studio in jo kasneje tudi preizkusili v praksi ter dobili odgovor o ustreznosti aplikacije in zadovoljstvu uporabnikov.

Mobile application to help older for calls and writing SMS

Key words: mobile application, Android, Java

UDK: 654.15:621.395.721.5(043.2)

Abstract

In this thesis, we addressed the development of a mobile application older people to place calls and write SMS messages. We reviewed the existing situation in this field and decided to develop an application that would assist older people in better understanding of fairly complex Android operating system, simplify features for them to the most basic level, increase icons and thus improve the manageability for those with a poor eyesight and other handicaps. The application will be developed in a freely accessible development environment Android Studio and later tested in practice, in order to obtain a feedback on the suitability of application and user satisfaction.

Vsebina

1	UVOD	1
1.1	Opis problema	1
1.2	Metode raziskovanja	1
1.3	Namen in cilji diplomske naloge.....	2
2	PREGLED MOBILNIH APLIKACIJ ZA POMOČ STAREJŠIM	3
2.1	Wiser laucher.....	3
2.2	Simple Senior Phone.....	4
2.3	Necta Launcher.....	4
2.4	Big Launcher	5
2.5	Large Launcher Senior Phone	6
2.6	Primerjava aplikacij, ki smo jih opisali, z našo aplikacijo.....	7
3	Opis uporabljenih tehnologij in orodij.....	8
3.1	Pametni telefon	8
3.2	Android	9
3.3	Android Studio.....	9
3.3.1	Izdelava uporabniških vmesnikov.....	10
3.3.2	Implementacija kode aplikacije	11
3.4	Gimp	12
3.5	Java	13
3.6	XML.....	13
4	Mobilna aplikacija za pomoč starejšim pri klicanju in pisanju SMS-jev.....	14
4.1	Razvoj osnove mobilne aplikacije.....	14
4.1.1	Osnovni uporabniški vmesnik.....	16
4.1.2	Uporabniški vmesnik nastavitve.....	19
4.2	Razvoj uporabniškega vmesnika za pomoč starejšim pri klicanju.....	20
4.3	Razvoj uporabniškega vmesnika za pomoč starejšim pri pisanju SMS-jev.....	22
5	Algoritem za pomoč starejšim pri klicanju in pisanju SMS-jev	24
5.1	Pomoč pri klicanju	24
5.2	Pomoč pri pisanju SMS-jev	27
6	Diskusija in zaključek	30
7	Viri in literatura	31

Kazalo slik

Slika 1: Wiser launcher	3
Slika 2: Simple Senior Phone	4
Slika 3: Necta Launcher	5
Slika 4: Big Launcher.....	5
Slika 5: Large Launcher Senior Phone.....	6
Slika 6: Navadni in pametni telefon	8
Slika 7: Zaščitni znak Android	9
Slika 8: Maskota Wilber.....	12
Slika 9: Orodje Gimp.....	12
Slika 10: Zaščitni znak Java	13
Slika 11: Prikaz linearne navpične in vodoravne postavitev	14
Slika 12: Osnovni uporabniški vmesnik.....	16
Slika 13: Ikone aplikacije, izdelane z Gimpom.....	18
Slika 14: Potrditveno okno pred klicanjem na 112.....	19
Slika 15: Uporabniški vmesnik nastavitve	19
Slika 16: Uporabniški vmesnik za izbiro klicatelja	20
Slika 17: Uporabniški vmesnik moje osebe	21
Slika 18: Izbira osebe	21
Slika 19: Uporabniški vmesnik za prikaz SMS-jev	22
Slika 20: Uporabniški vmesnik nov SMS	23
Slika 21: Uporabniški vmesnik izbira črke	23

Kazalo tabele

Tabela 1: Dobre in slabe lastnosti v primerjavi z našo aplikacijo	7
---	---

1 UVOD

1.1 Opis problema

Večina obstoječih aplikacij za pomoč starejšim pri klicanju in pisanju SMS-jev starejšim osebam ne pomaga dovolj. Ponavadi uporabniški vmesniki mobilnih aplikacij vsebujejo premajhno pisavo, ki jo starejši s težavo preberejo. Problem imajo tudi s pisanjem SMS-jev, saj so tipke na tipkovnicah majhne in SMS s težavo napišejo. To pomeni, da aplikacija ni dovolj prijazna do starejšega uporabnika, da bi jo znal uporabljati. V tem diplomskem smo se torej ukvarjali z razvijanjem aplikacije za pomoč starejšim pri klicanju in pisanju SMS-jev. V okviru diplomske naloge smo razvili mobilno aplikacijo, ki je razvita na platformi Android. Aplikacija je primerna tudi za kognitivne in zdravstvene (negibljivi prsti, slab vid, sluh) slabosti starejših.

1.2 Metode raziskovanja

Za razvoj aplikacije smo uporabili prosto dostopno razvojno okolje Andoid Studio, ki uporablja programski jezik Java in opisni jezik XML. Pri razvoju aplikacije za pomoč starejšim pri klicanju in pisanju SMS-jev smo uporabili na svetovnem spletu dosegljive strokovne vire, kot so članki in knjige. Za namene testiranja razvite mobilne aplikacije bomo vključili starejše osebe različnih starostnih skupin.

1.3 Namen in cilji diplomske naloge

Namen diplomskega dela je razviti mobilno aplikacijo Android, ki bo starejšim osebam v pomoč pri klicanju in pisanju SMS-jev. Največ poudarka je na uporabniškem vmesniku, ki bo preprost za uporabo, pri čemer so pomembne velika pisava, da bo dobro viden, in velike ikone za lažje klike. Tako bo aplikacija primerna tudi za kognitivne in zdravstvene (negibni prsti, slab vid, sluh) slabosti starejših.

Cilji diplomske naloge so naslednji:

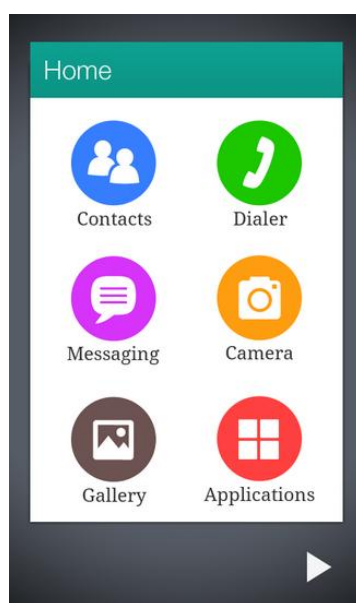
- Razviti mobilno aplikacijo na platformi Android.
- Preprost osnovni uporabniški vmesnik.
- Preprost uporabniški vmesnik za klicanje.
- Preprost uporabniški vmesnik za pisanje in branje SMS-jev.

2 PREGLED MOBILNIH APLIKACIJ ZA POMOČ STAREJŠIM

Pametni telefoni z operacijskim sistemom Android nudijo veliko funkcij in aplikacij, ki so zelo uporabne ter lahko dostopne in dokaj ugodne. Vendar pa je uporabniški vmesnik pri operacijskem sistemu Android velikokrat neprijazen do uporabnika in zapleten za uporabo. Takšen problem nastopi predvsem pri starejših osebah. Zato se v zadnjem času že razvijajo aplikacije za pomoč starejšim osebam. Zbrali smo nekaj aplikacij, ki so med najbolj razširjenimi na tem področju.

2.1 Wiser launcher

Wiser launcher je ena izmed najbolj preprostih aplikacij, ki sem jih testiral. Aplikacija v celoti spremeni uporabniški vmesnik in tako ostane samo 6 pomembnih funkcij na začetnem zaslonu z velikimi ikonami. Vsi pomembni kontakti in aplikacije se lahko dodajo med priljubljene na začetni zaslon za hiter dostop. Kontakti in aplikacije, dodani med priljubljene, so enostavno dostopni z dotikom zaslona. Aplikacija prav tako poenostavi prikaz informacij, saj so predstavljene samo osnovne stvari, kar zagotavlja boljšo preglednost.



Slika 1: Wiser launcher

2.2 Simple Senior Phone

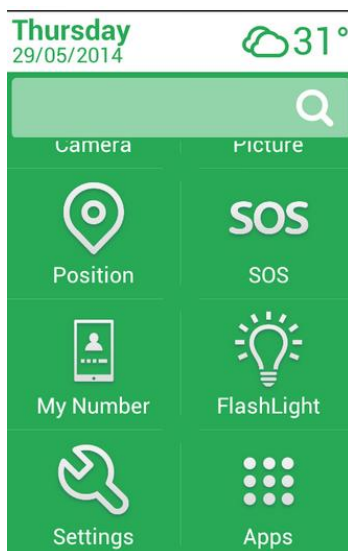
Simple Senior Phone prav tako precej poenostavi napravo Android. Privzete funkcije, kot so domači zaslon, velike ikone, pomembni kontakti in podobno, so avtomatsko generirane po instalaciji aplikacije na napravo Android. Aplikacija ne uporablja privzetega prikaza za informacije, ampak ima svoje prikaze, ki so poenostavljeni z večjo pisavo in različnimi barvami za lažje ter hitrejše branje. Ta aplikacija precej poenostavi uporabo predvsem za osebe s slabim vidom. Prav tako ima možnost blokiranja uporabnikov, ki so nezaželeni.



Slika 2: Simple Senior Phone

2.3 Necta Launcher

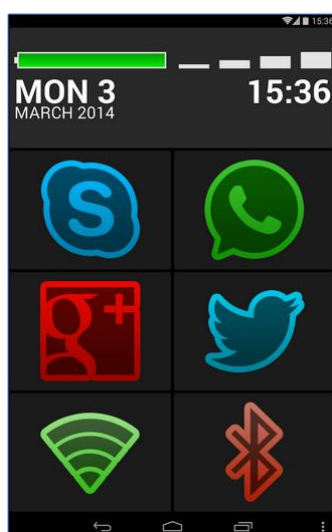
Necta Launcher je aplikacija predvsem za starejše osebe. Vse funkcije naredi preprostejše in večje za lažjo uporabo. Da uporaba vseh funkcij ne bi bila preveč zapletena, so na voljo le najbolj pomembne, kot so klicanje, sporočila in podobno. Na domačem zaslonu so ikone za te funkcije, ki so še posebej velike za lažjo preglednost. Tako lahko osebe s slabim vidom z lahkoto uporabljajo napravo. Pomembna funkcija je tudi, da lahko oseba pošlje svojo lokacijo kontaktu, ki ga ima shranjenega.



Slika 3: Necta Launcher

2.4 Big Launcher

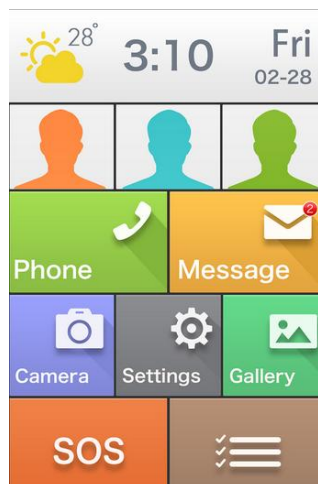
Big Launcher je še ena aplikacija, ki poenostavi operacijski sistem Android. Podobno kot ostale aplikacije poveča in poenostavi vse funkcije za lažjo preglednost. Uporabniški vmesnik prikaže vse pomembne funkcije, kot so klicanje, sporočila, klic v sili in podobno, na začetnem zaslonu. Vse ikone so prenovljene in povečane zaradi lažje preglednosti. Aplikacija je na voljo tako v brezplačni verziji kot v premium verziji. Brezplačna verzija ima nekaj omejitev glede funkcij in nastavitvev uporabniškega vmesnika.



Slika 4: Big Launcher

2.5 Large Launcher Senior Phone

V primerjavi s privzetim uporabniškim vmesnikom sistema Android, ki je dokaj zapleten, Large Launcher poenostavi funkcije za lažjo uporabo, predvsem za starejše ljudi. S preprostimi gibi po ekranu se lahko aplikacija zažene in uporablja. Na voljo je veliko opcij, od klicanja, pošiljanja sporočil in lokacije, prav tako pa je kot pri ostalih podobnih aplikacijah za starejše vse povečano ter bolj pregledno. V primerjavi z ostalimi aplikacijami se pri tej vidi zelo lepo oblikovan in prijazen uporabniški vmesnik, prav tako pa se avtomatsko nastavijo vse pomembne funkcije, ki so potrebne za osnovno uporabo.



Slika 5: Large Launcher Senior Phone

2.6 Primerjava aplikacij, ki smo jih opisali, z našo aplikacijo

Vse aplikacije, ki smo jih opisali, poenostavijo uporabo pametnega telefona starejšim uporabnikom, saj imajo velike ikone in veliko pisavo. Večina aplikacij za pomoč starejšim nima velike tipkovnice, torej tipk, da bi lahko starejše osebe preprosteje klikale po tipkovnici in s tem vnesle besedilo. Zato smo izdelali večjo tipkovnico z večjimi tipkami, da bi starejšim uporabnikom poenostavili tudi vnos besedila. Spodnja tabela prikazuje prednosti in slabosti aplikacije za pomoč starejšim. S plusom smo označili dobre lastnosti, z minusom pa slabe. Največja slabost vseh aplikacij, ki smo jih opisali, so majhne tipke na tipkovnici, ostalo imajo dobro prilagojeno za starejše uporabnike.

Aplikacija	Preprost uporabniški vmesnik	Velika pisava	Velika tipkovnica	Velike ikone
Wiser Launcher	+	+	-	+
Simple Senior Phone	+	+	-	+
Necta Launcher	+	+	-	+
Big launcher	+	+	-	+
Large Launcher Senior Phone	+	+	-	+
Naša razvita aplikacija	+	+	+	+

Tabela 1: Dobre in slabe lastnosti v primerjavi z našo aplikacijo

3 Opis uporabljenih tehnologij in orodij

Pri izbiri orodij in uporabljenih tehnologij sta bila ključna za izbiro prosta dostopnost in brezplačnost orodij. Za platformo smo izbrali Android, za razvoj mobilne aplikacije za pametne telefone pa uporabili Android Studio. Aplikacija je razvita v programskem jeziku Java in označevalnem jeziku XML. Za izgradnjo ikon aplikacije pa smo uporabili orodje Gimp.

3.1 Pametni telefon

Iz običajnih mobilnih naprav, ki so bile namenjene za klicanje, so nastale večpredstavnostne komunikacijske naprave, s katerimi lahko pošiljamo MMS-je, kličemo uporabnika z videoklicem, se povezujemo z internetom in predvajamo multimedijske vsebine. Na spletu pa je veliko razvitih aplikacij za različne namene, skoraj za vsakega uporabnika se najde zanimiva aplikacija. Običajni in pametni telefon se razlikujeta po tem, da imajo pametni telefoni zaslon na dotik skoraj po vsej napravi, imajo večji pomnilnik in zelo kakovostne fotoaparate. Zaradi hude konkurence izdelovalcev mobilnih naprav razvoj tehnologij pri mobilnih napravah strmo narašča.



Slika 6: Navadni in pametni telefon

3.2 Android

Android je operacijski sistem, ki ga razvijata Google in združenje podjetij Open Handset Alliance. Razvit je na osnovi jedra Linux, ki je sodobno večopravilno jedro. Večino jedra Linuxa je napisanega v programskem jeziku C. Največja prednost je odprtokodnost sistema, to pomeni, da so posodobitve in aplikacije, ki so na portalu Google Play, brezplačne ter prosto dostopne. Najbolj uporabljen je brezplačen navigacijski sistem. Operacijski sistem Android nam olajša uporabo Googlovih storitev, kot so Google Chat, Google Calender, Gmail in Google Reader, ki so med seboj sinhronizirane z našimi podatki.

Za mobilno aplikacijo na platformi Android smo se odločili zaradi odprtokodnosti sistema in zaradi razvijalnih okolij, ki so brezplačne za uporabo ter prosto dostopne.



Slika 7: Zaščitni znak Android

3.3 Android Studio

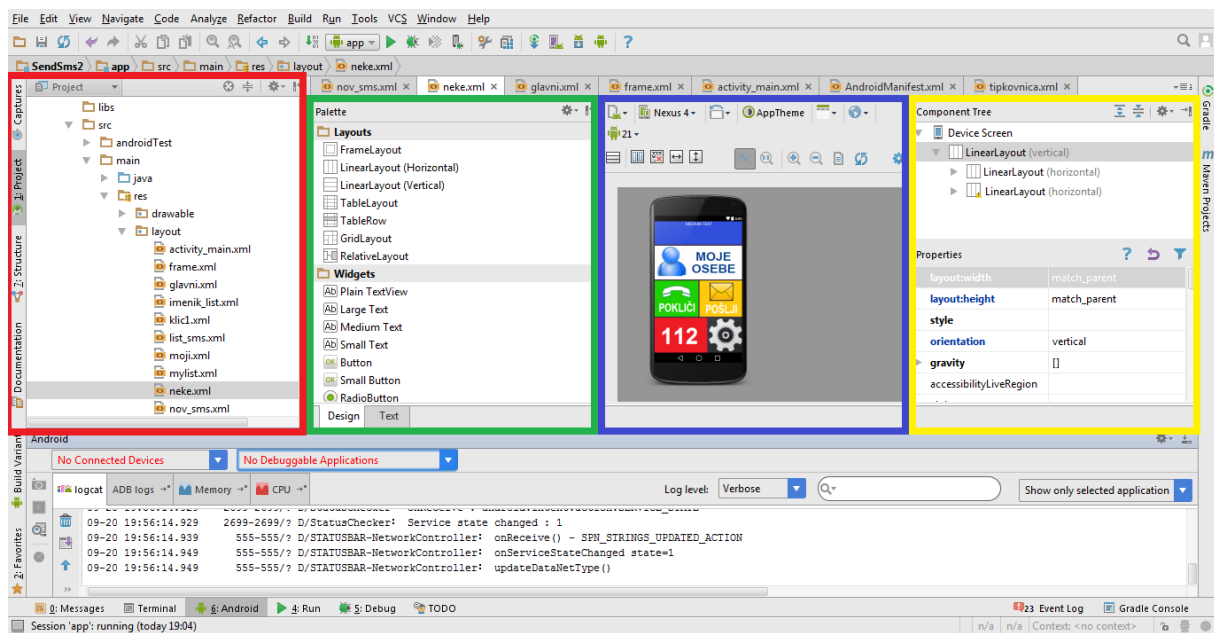
Android Studio je prostodostopno razvojno okolje, ki uporablja programski jezik Java. Prva različica je bila razvita leta 2013, prva stabilna verzija pa eno leto pozneje – leta 2014. Orodje je razvito za operacijske sisteme Windows, Linux in Mac OS X. V Android Studio je vključeno vse, kar potrebujemo za razvoj aplikacije Android.

Za razvojno orodje Android Studio smo se odločili, ker je v enem orodju združeno vse za razvoj aplikacije na platformi Android, ker je prosto dostopno, ker ima zelo dober uporabniški vmesnik za razvoj aplikacij in ker je testiranje aplikacije na simulatorjih, ki jih ponuja razvojno orodje, ali na dejanski mobilni napravi, preprosto.

3.3.1 Izdelava uporabniških vmesnikov

Videz uporabniškega vmesnika Android Studio smo razdelili na 4 pomembnejše dele pri izdelavi uporabniških vmesnikov aplikacije:

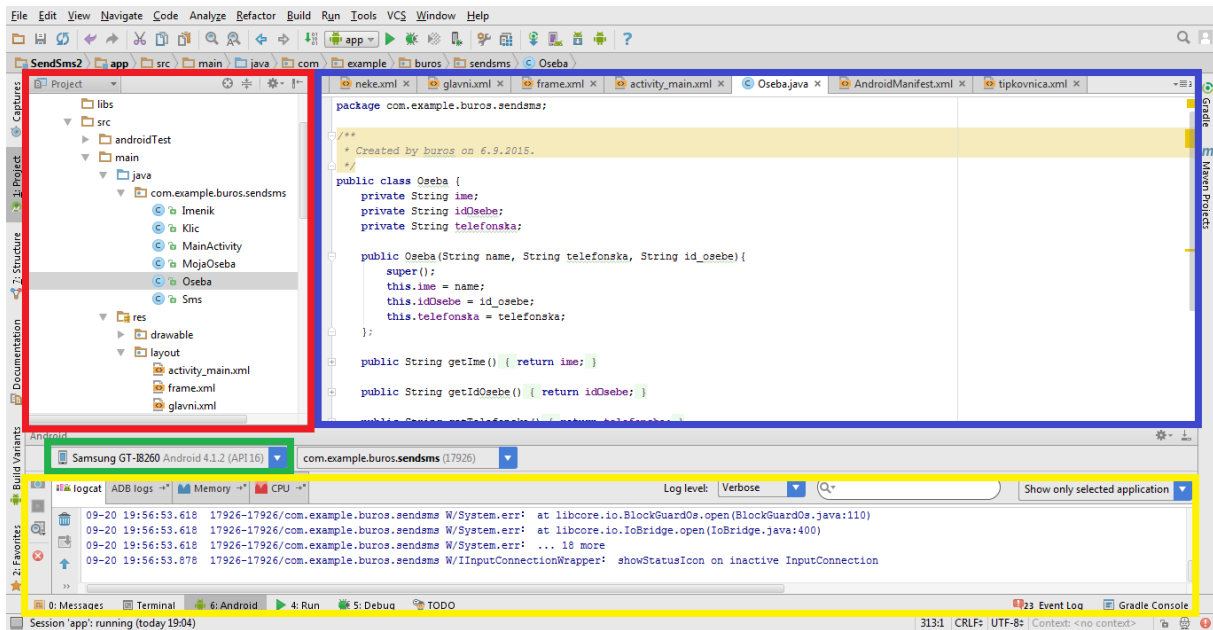
- rdeči okvir predstavlja prikaz projekta po mapah, videz uporabniških vmesnikov so XML-datoteke, ki so v mapi postavitev;
- zeleni okvir nam omogoča izbiro postavitve gradnikov uporabniških vmesnikov in same gradnike uporabniških vmesnikov, ki jih preprosto povlečemo v modri okvir, kjer se nam gradnik prikaže na napravi;
- modri ovir nam prikazuje videz uporabniškega vmesnika, ki ga izdelujemo;
- rumeni ovir nam prikazuje drevesno strukturo gradnikov glede na to, kako so dodani, in lastnosti vsakega gradnika, ko ga imamo izbranega.



3.3.2 Implementacija kode aplikacije

Videz uporabniškega vmesnika Android Studio smo razdelili na 4 pomembnejše dele pri implementaciji kode aplikacije:

- rdeči okvir predstavlja prikaz projekta po mapah, implementacija kode je v mapi Java;
- modri ovir nam prikazuje implementacijsko kodo aplikacije;
- zeleni okvir nam prikazuje, ali je katera mobilna naprava na voljo za zagon in testiranje aplikacije;
- rumeni ovir nam lahko prikazuje komunikacijo med mobilno napravo in vmesnikom Android Studio, napake v času prevajanja, uporabo pomnilnika ali uporabo procesorja.



3.4 Gimp

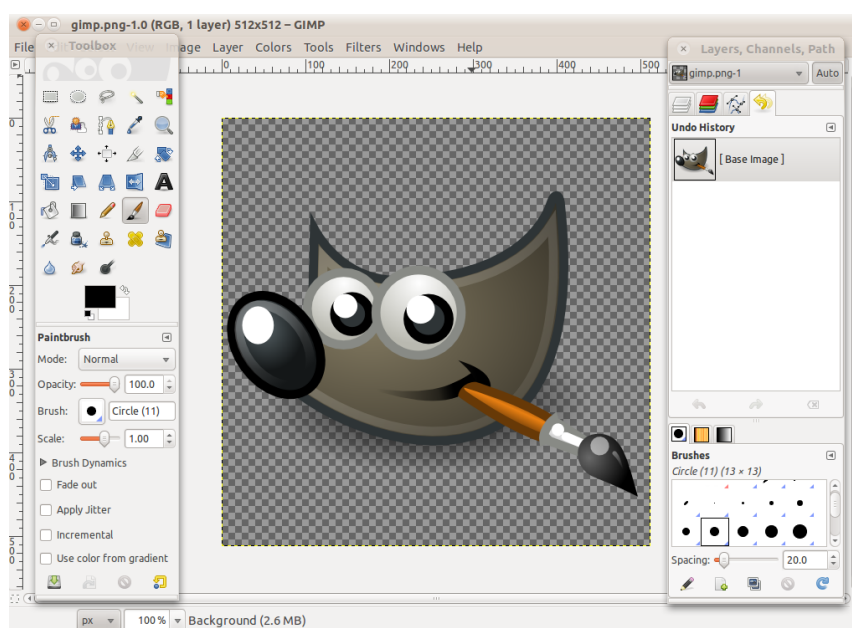
Gimp (General Image Manipulation Program) je računalniški program za urejanje rasterske in vektorske grafike. Pri vektorski grafiki se pri transformaciji ločljivost slike ne pokvari, pri rastrski pa se. Je prosto dostopen in odprtokoden, razvila sta ga študenta Spencer Kimball in Peter Mattis. Prva javno dostopna verzija je izšla leta 1996. Ko je leta 1997 Gimp postal del projekta GNU, se je pomen kratic iz »General Image Manipulation Program« spremenil v »GNU Image Manipulation Program«.

Deluje na različnih platformah, MS Windows, Linux, Solaris, FreeBSD, OS X. Uporabili smo ga za izdelovanje ikon, ki so uporabljene v uporabniških vmesnikih naše mobilne aplikacije za pomoč starejšim. Maskota Wilber je uradna maskota za skupino Gimp, ustvaril jo je Tuomas Kousmanen leta 1997.



Slika 8: Maskota Wilber

Za uporabo programa Gimp smo se odločili zaradi brezplačnosti uporabe in proste dostopnosti. Samo orodje nam omogoča veliko funkcionalnosti za izdelavo ikon.



Slika 9: Orodje Gimp

3.5 Java

Java je objektno orientiran programski jezik, saj je vse razen osnovnih podatkov tipov, kot so boolean, int, short, float, double, char in byte objekt. Java je preprosta za učenje in razvijanje, saj ima preprosto sintakso. Platforma Java je sestavljena iz osnovne knjižnice razredov, strukture vmesne kode, definicije vmesne kode in specifikacije navideznega stroja. Podpora večopravnosti je vgrajena v razredu Thread, saj lahko hkrati nalaga ikone, prikaže slike, predvaja zvok in prenaša zbirke. Java je dinamična, sej se razredi naložijo v pomnilnik takrat, ko jih potrebujemo. Ker v Javi ni kazalcev, ne moremo dostopati do pomnilniških lokacij, zaradi česar je bolj varna kot programski jezik C\C++.

Različice Jave so:

- J2SE – je standardna različica, uporablja se za računalnike;
- J2ME – je različica, ki se uporablja za mini naprave, kot so mobilni telefoni, tablice;
- J2EE – je poslovna različica.



Slika 10: Zaščitni znak Java

3.6 XML

XML je kratica za razširljiv označevalni jezik. Velika prednost je, da je razumljiv in lahko berljiv. Omogoča opisovanje strukturiranih besedil in je zelo podoben HTML-ju. Vsaka značka se začne z znakom < in konča z znakom />. Spodaj je prikaz uporabe XML-ja.

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <heading>Sporocilo</heading>
  <body>To je sporocilo!</body>
</note>
```

4 Mobilna aplikacija za pomoč starejšim pri klicanju in pisanju SMS-jev

4.1 Razvoj osnove mobilne aplikacije

Razvoj mobilne aplikacije se začne pri izbiri platforme, za katero bo aplikacija narejena, v našem primeru je to platforma Android. Za razvoj mobilne aplikacije Android smo uporabili razvojno orodje Android Studio. Za ustvarjanje ikon pa smo uporabili Gimp. Ko so orodja izbrana s katerimo bomo razvili mobilno aplikacijo. Najprej smo se osredotočili na videz začetnega uporabniškega vmesnika, ki mora biti enostaven in razumljiv starejšim osebam za uporabo. Pri postavitvi gradnikov uporabniškega vmesnika smo se odločili za linearno postavitev gradnikov v vodoravni in navpični legi. Glavna postavitev gradnikov uporabniškega vmesnika je v vodoravni legi, vodoravna postavitev pa je nato razdeljena še na navpične postavitve gradnikov. Spodaj je osnovni prikaz razdeljevanja uporabniškega vmesnika na linearne navpične in vodoravne postavitve gradnikov.



Slika 11: Prikaz linearne navpične in vodoravne postavitve

V datoteki AndroidManifest.xml so zapisani bistveni podatki o naši aplikaciji Android, kot so ime aplikacije, ime paketa in različica aplikacije. Naša aplikacija potrebuje dostop do podatkov za klicanje, branje in pisanje SMS-jev, branje podatkov iz imenika, mora dodati uporabniške pravice, da lahko dostopamo do podatkov.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.buros.sendsms" >

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

    </application>

    <uses-permission android:name="android.permission.SEND_SMS"></uses-permission>
    <uses-permission android:name="android.permission.READ_SMS"></uses-permission>
    <uses-permission android:name="android.permission.CALL_PHONE"></uses-
        permission>

    <uses-permission android:name="android.permission.READ_CONTACTS"></uses-
        permission>
    <uses-permission android:name="android.permission.READ_CALL_LOG"></uses-
        permission>

</manifest>
```

4.1.1 Osnovni uporabniški vmesnik

Opis začetnega uporabniškega vmesnika od zgoraj navzdol: na vrhu je izpis dneva ure in datuma, nato sledita gumba moje osebe in pošlji. S klikom na gumb moje osebe se prikaže uporabniški vmesnik, ki prikaže uporabnikove najljubše osebe, ki jih izbere pri prvi uporabi in jih nato lahko kliče ali jim pošlje SMS kar s pritiskom na gumb. Gumb pokliči sproži, da se nam prikaže uporabniški vmesnik z osebo iz imenika. Gumb pošlji sproži, da se nam prikaže uporabniški vmesnik, ki nam prikaže sporočilo od zadnjega proti prvemu prejetemu ali poslanemu. Gumb 112 sproži, da se opre okno, ki uporabnika vpraša, ali želi resnično poklicati na 112 – s potrditvijo se pokliče 112.



Slika 12: Osnovni uporabniški vmesnik

Razvoj smo začeli z osnovnim uporabniškim vmesnikom, ki bo razumljiv za starejše osebe. Pri postavitvi gradnikov uporabniškega vmesnika smo se odločili za linearno postavitev

gradnikov v vodoravni in navpični legi. Glavna postavitev gradnikov uporabniškega vmesnika je v vodoravni legi. Vsa ta postavitev je opisana v XML-jeziku.

Spodaj je osnovna postavitev uporabniškega vmesnika, ki ima navpično linearno postavitev.

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">

</LinearLayout>
```

Začetnemu uporabniškemu vmesniku dodamo vodoravno linearno postavitev, ki ji nastavimo širino in višino, ozadje ter postavitev gradnikov v tej postavitvi. V to postavitev dodamo gradnik za prikaz besedila, ki nam bo prikazal dan, uro in datum. Gradniku za prikaz besedila nastavimo, da se bo besedilo prikazalo na sredini tega gradnika.

```
<LinearLayout
    android:layout_width="fill_parent"
    android:layout_height="100dp"
    android:orientation="horizontal"
    android:background="#314fdb"
    android:layout_gravity="center">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:text="Medium Text"
        android:gravity="center"
        android:id="@+id/textViewDateTime"
        android:textColor="@android:color/white"
        android:textAllCaps="true" />

</LinearLayout>
```

Za tem dodamo navpično linearno postavitev gradnikov, v to postavitev pa dodamo vodoravne linearne postavitve za gumba. Spodaj je primer, kjer smo dodali dva gumba, ki sta v vodoravni linearni postavitvi, v navpično linearno postavitev.

```
<LinearLayout
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
    android:layout_weight="12">

    <LinearLayout
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:orientation="horizontal"
        android:layout_weight="4" >
        <ImageButton
            android:id="@+id/moja_osebe"
            android:layout_width="wrap_content"
            android:layout_height="fill_parent"
            android:layout_weight="1"
            android:background="@drawable/moje_osebe"
            android:layout_marginBottom="5dp" />

        <ImageButton
            android:id="@+id/nastavitve"
            android:layout_width="wrap_content"
            android:layout_height="fill_parent"
            android:layout_weight="1"
            android:background="@drawable/settings"
            android:layout_marginBottom="5dp"
            android:layout_marginLeft="5dp" />

    </LinearLayout>

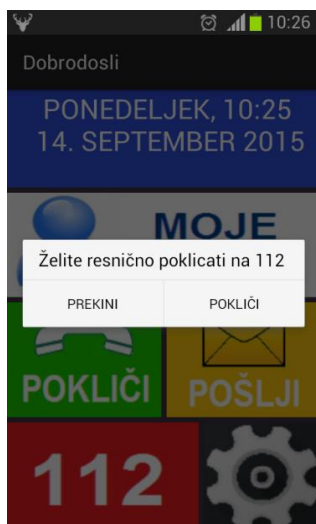
</LinearLayout>
```

Tudi vse ostale gumbе v osnovnem uporabniškem vmesniku smo dodali na enak način, kot je prikazan zgoraj – da smo torej vodoravne linearne postavitve dveh gumbov vstavljali v navpično linearno postavitev. Gumbom za ozadje nastavimo ikone, ki smo jih izdelali s programom Gimp.



Slika 13: Ikone aplikacije, izdelane z Gimpom

S klikom na gumb 112 se nam prikaže potrditveno okno. Če želimo res poklicati 112, s tem preprečimo klicanje po pomoti.



Slika 14: Potrditveno okno pred klicanjem na 112

4.1.2 Uporabniški vmesnik nastavitve

Uporabniški vmesnik nastavitve nam ponuja izbiro dveh tem, v temi sta opredeljeni barva ozadja in barva prikazanega besedila. Ena tema temelji na temni podlagi z belo pisavo, druga pa na svetli podlagi s črno pisavo. Izbiramo pa lahko tudi med tremi velikostmi pisave.



Slika 15: Uporabniški vmesnik nastavitve

4.2 Razvoj uporabniškega vmesnika za pomoč starejšim pri klicanju

Uporabniški vmesnik za pomoč pri klicanju temelji na veliki pisavi imen in priimkov oseb ter na sliki osebe iz imenika. Sestavljen je iz gumba nazaj, s katerim se vrnemo na začetni vmesnik, in seznama oseb iz imenika, za vsako osebo prikazemo sliko ter ime in priimek, s klikom na osebo pa jo pokličemo.



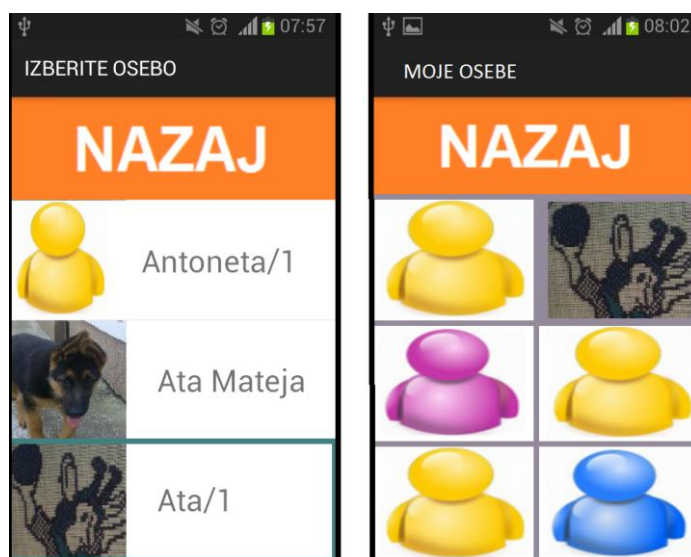
Slika 16: Uporabniški vmesnik za izbiro klicatelja

Pri pomoči pri klicanju smo naredili še en uporabniški vmesnik, kjer imamo prikazanih 6 slik oseb iz imenika, ki jih s klikom na sliko preprosto pokličemo. Uporabniški vmesnik moje osebe nam ponuja, da lahko šest priljubljenih oseb, ki si jih zapomnimo, hitro pokličemo s klikom na sliko osebe. Sestavljen je iz gumba nazaj, kjer se po kliku vrnemo v osnovni uporabniški vmesnik, in šestih gumbov, ki imajo za ozadje slike oseb iz imenika. Če še nimamo zapomnjene osebe, se nam ob kliku prikaže seznam oseb s slikami. Ko izberemo eno od oseb, se nam na gumbu prikaže slika osebe, ki smo jo izbrali, ob ponovnem kliku pa že lahko kličemo to osebo.



Slika 17: Uporabniški vmesnik moje osebe

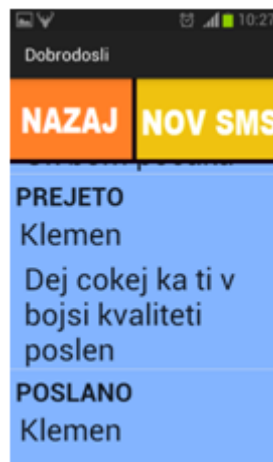
Uporabniški vmesnik moje osebe nam pri prvi uporabi ne kaže slik oseb iz imenika, moramo jih izbrati. S klikom na gumb brez slike naše osebe se prikaže uporabniški vmesnik za izbiro osebe, kjer s klikom na osebo izberemo, in se nam slika prikaže na vmesniku moje osebe.



Slika 18: Izbira osebe

4.3 Razvoj uporabniškega vmesnika za pomoč starejšim pri pisanju SMS-jev

Pri uporabniškem vmesniku za pisanje SMS-jev smo se osredotočili na veliko tipkovnico za pisanje SMS-ja in na veliko pisavo pri branju ter pri prikazovanju SMS-ja med pisanjem. Uporabniški vmesnik je sestavljen iz gumba nazaj, s klikom se vrnemo na osnovni zaslon. Naslednji je gumb nov sms, s klikom se nam prikaže vmesnik za ustvarjanje novega SMS-ja. Pod tema gumboma pa je prikaz SMS-jev – od zadnjega prejetega do prvega prejetega ali poslanega.



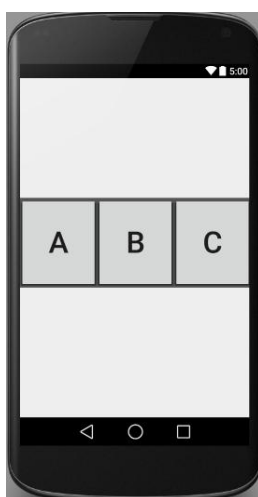
Slika 19: Uporabniški vmesnik za prikaz SMS-jev

Vmesnik za ustvarjanje novega SMS-ja je sestavljen iz gumba nazaj in gumba za izbiro prejemnika SMS-ja. S klikom se nam prikaže seznam oseb, s klikom na osebo se pošiljatelj shrani. Naslednji je tekstovni gradnik za vnos besedila SMS-ja. Tu smo se osredotočili na izdelavo lastne tipkovnice. Ker imajo starejši slabše gibalne sposobnosti in težje premikajo prste ter klikajo po zaslonu na dotik, predstavlja vnos besedila SMS-ja zanje velik problem, saj je tipkovnica majhna in težko pritisnejo na pravo tipko na tipkovnici. Zato smo izdelali tipkovnico, ki ima manj tipk, zato so tipke večje velikosti. Na tipki smo združili po 3 ali 4 črke, s pritiskom na tipko se nam prikažejo vse črke, ki so bile na tipki posamezno.



Slika 20: Uporabniški vmesnik nov SMS

Tipkovnica je razdeljena na 12 gumbov, 8 gumbov ima po tri črke po vrstnem redu abecede, eden pa ima 4 črke. Naslednji gumb ima ločila, kot so pika, vejica, klicaj in vprašaj. V zadnji vrstici so gumbi: pošlji, s katerim se pošlje SMS, briši, s katerim pobrišemo zadnjo vneseno črko, in presledek, s katerim v besedilo vnesemo presledek. S klikom na gumb, kjer so črke, se nam prikaže novo okno, kjer lahko izberemo črko. Če pritisnemo na tipko ABC, se nam prikažejo trije dodatni gumbi nad zaslon za vnos SMS-ja, to so gumbi A, B in C. S klikom se črka vnese v besedilo in prikaže se prejšnji zaslon.



Slika 21: Uporabniški vmesnik izbira črke

5 Algoritem za pomoč starejšim pri klicanju in pisanju SMS-jev

5.1 Pomoč pri klicanju

Algoritem za pomoč pri klicanju najprej pridobi seznam oseb iz imenika telefona.

Za branje telefonskega imenika iz mobilne naprave je treba nastaviti uporabniško dovoljenje.

```
<uses-permission android:name="android.permission.READ_CONTACTS"></uses-permission>
```

Ustvarili smo razred Oseba, ki ima privatne parametre ime, idSlike in telefonsko, to pomeni, da do teh parametrov tega razreda lahko dostopajo samo instance tega objekta.

```
public class Oseba {
    private String ime;
    private String idOsebe;
    private String telefonska;

    public Oseba(String name, String telefonska, String id_osebe){
        super();
        this.ime = name;
        this.idOsebe = id_osebe;
        this.telefonska = telefonska;
    };

    public String getIme() {
        return ime;
    }

    public String getIdOsebe() {
        return idOsebe;
    }

    public String getTelefonska() {
        return telefonska;
    }
}
```

Razred Imenik ima parameter list seznam_oseb in konstruktorja, ki dodata osebo v list seznam_oseb, ter metodo getSeznamOsebIzImenika, ki vrne seznam_oseb.


```

public class Imenik {

    private List<Oseba> seznam_oseb;

    public Imenik(Oseba oseba){
        super();
        this.seznam_oseb.add(oseba);
    };

    public Imenik(String name, String telefonska, String id_osebe){
        super();
        Oseba nova_oseba = new Oseba(name,telefonska,id_osebe);
        this.seznam_oseb.add(nova_oseba);
    };

    public List<Oseba> getSeznamOsebIzImenika(){ return this.seznam_oseb; }

}

```

Nato smo v razred Imenik dodali metodo preberiOsebeIzImenika, ki prebere osebe iz imenika in jih dodaja v list seznam_oseb. List seznam oseb nato sortiramo po imenu oseb. Ta metoda se pokliče v inicializaciji aplikacije.

```

public void preberiOsebeIzImenika(Cursor phones){

    while (phones.moveToNext())
    {
        String idOsebe =
phones.getString(phones.getColumnIndex(ContactsContract.CommonDataKinds.Phone.CONTACT_ID));
        String name =
phones.getString(phones.getColumnIndex(ContactsContract.CommonDataKinds.Phone.DISPLAY_NAME));
        String phoneNumber =
phones.getString(phones.getColumnIndex(ContactsContract.CommonDataKinds.Phone.NUMBER));

        seznam_oseb.add(new Oseba(name,phoneNumber,idOsebe));
    }

phones.close();

// sortiranje po imenu
Collections.sort(seznam_oseb, new Comparator() {

    public int compare(Object o1, Object o2) {
        Oseba p1 = (Oseba) o1;
        Oseba p2 = (Oseba) o2;
        return p1.getIme().compareToIgnoreCase(p2.getIme());
    }
});
}

```

Seznam oseb se nato prikaže v listi, kjer od vsake osebe prikažemo sliko, zraven pa še ime osebe, s klikom na osebo pa jo pokličemo.

Za klicanje smo morali dodati uporabniško dovoljenje za klicanje.

```
<uses-permission android:name="android.permission.CALL_PHONE"></uses-permission>
```

Dodali smo še uporabniško dovoljenje za branje zgodovine klicev.

```
<uses-permission android:name="android.permission.READ_CALL_LOG"></uses-permission>
```

5.2 Pomoč pri pisanju SMS-jev

Najprej iz mobilne naprave preberemo zgodovino SMS-jev, za kar potrebujemo uporabniško dovoljenje za branje. Ker pa bo aplikacija ponujala tudi ustvarjanje novih SMS-jev, je treba dati še uporabniško dovoljenje za pisanje SMS-jev. Uporabniški dovoljenji dodamo v datoteko manifest.

```
<uses-permission android:name="android.permission.SEND_SMS"></uses-permission>
<uses-permission android:name="android.permission.READ_SMS"></uses-permission>
```

Ustvarimo razred Sms, ki ima parametre, telefonsko, tipSporocila, datum in teloSporocila.

```
public class Sms {
    // private String
    private String number;
    private String messageType;
    private Date date;
    private String body;

    Sms(String number,String messageType, Date date, String body){
        this.number = number;
        this.messageType = messageType;
        this.date = date;
        this.body = body;
    }

    public String getNumber() {
        return number;
    }

    public Date getDate() {
        return date;
    }

    public String getMessageType() {
        return messageType;
    }

    public String getBody() {
        return body;
    }
}
```

Metoda, s katero preberemo iz mobilne naprave zgodovino SMS-jev, jih shranimo v seznam po vrsti od zadnjega prejetega do prvega prejetega.

```
private void getSmsData(int dolzina){
    seznam_sms_ov = new ArrayList<Sms>();

    // preberemo zgodovino sms-jev
    Uri uri = Uri.parse("content://sms/");
```

```

Cursor cursor = getContentResolver().query(uri, null, null, null, null);

/* pomikamo se po seznamu sms-jev in za vsakega posebej pridobimo kašnega tipa je
sms - ali je prejet, poslan ali neposlan, datum, besedilo sporočila in telefonsko,
s katere je bil sms poslan ali prejet
*/
if (cursor.moveToFirst()) {
    int count = cursor.getCount();
    if(dolzina<count)
    {
        count = dolzina;
    }
    for (int i = 0; i < count; i++) {
        String body = "";
        if(cursor.getString(cursor.getColumnIndexOrThrow("body")) != null) {
            body = cursor.getString(cursor.getColumnIndexOrThrow("body"))
                .toString();
        }
        String number = "";
        if(cursor.getString(cursor.getColumnIndexOrThrow("address")) != null)
        {
            number = cursor.getString(cursor.getColumnIndexOrThrow("address"))
                .toString();
        }
        String date = "";
        if(cursor.getString(cursor.getColumnIndexOrThrow("date")) != null)
        {
            date = cursor.getString(cursor.getColumnIndexOrThrow("date"))
                .toString();
        }
        Date smsDayTime = new Date(Long.valueOf(date));
        String type = "";
        if(cursor.getString(cursor.getColumnIndexOrThrow("type")) != null)
        {
            type = cursor.getString(cursor.getColumnIndexOrThrow("type"))
                .toString();
        }

        // tip sms
        String typeOfSMS = null;
        switch (Integer.parseInt(type)) {
            case 1:
                typeOfSMS = "PREJETO";
                break;

            case 2:
                typeOfSMS = "POSLANO";
                break;

            case 3:
                typeOfSMS = "NEPOSLANO";
                break;
        }

        if(number.length()>9){
            number = number.substring(4);
        }else {
            if(number.length()>0) {
                number = number.substring(1);
            }
        }
    }
}

```

```
    }
  }
  String ime="";

  /* poiščemo telefonsko številko sms-ja v seznamu moj_imenik in prebere iz
     seznama ime te osebe ter jo shranimo, da vemo, katera oseba je poslala
     oziroma prejela sms
  */
  for(int j = 0; j < moj_imenik.size()-1; j++) {
    String tmpNumber = moj_imenik.get(j).getTelefonska();
    if(tmpNumber.length()>9){
      tmpNumber = tmpNumber.substring(4);
    }else {
      if(tmpNumber.length()>0) {
        tmpNumber = tmpNumber.substring(1);
      }
    }

    if (tmpNumber.equals(number))
    {
      ime = moj_imenik.get(j).getIme();
    }
  }
  seznam_sms_ov.add(new Sms(ime,typeOfSMS,smsDayTime,body));

  cursor.moveToNext();

}
}
cursor.close();
}
```

6 Diskusija in zaključek

V prihodnosti želim še izboljšati aplikacijo, in sicer na ta način, da bo uporabniški vmesnik še bolj prijazen do uporabnika, dopolniti nameravam ikone aplikacije, saj sta videz aplikacije in samih ikon zelo pomembna, če želimo uporabnika pritegniti, da bo uporabljal aplikacijo. V nastavitvah bom dodal še možnost izbire angleškega jezika.

S prihodom pametnih telefonov se je na področju mobilnih naprav pojavilo veliko število aplikacij, ki nam olajšajo ali namesto nas beležijo pomembne stvari. Kljub temu, da imajo pametni telefoni ekran skoraj čez ves zaslon, pa je za starejše uporabnike še vedno pisava v aplikacijah in na tipkovnicah premajhna. Pri tipkovnicah je največji problem majhnost tipk, saj starejše osebe težko pritisnejo določeno tipko.

S pomočjo razvite aplikacije lahko starejše osebe lažje najdejo osebo, ki jo želijo poklicati, in lažje napišejo ter preberejo SMS.

7 Viri in literatura

- <https://www.android.com/>
 - Android, zadnji dostop 23. 9. 2015
- [https://sl.wikipedia.org/wiki/Android_\(operacijski_sistem\)](https://sl.wikipedia.org/wiki/Android_(operacijski_sistem))
 - Android, zadnji dostop 19. 9. 2015
- <https://developer.android.com/sdk/index.html>
 - Android Studio, zadnji dostop 22. 9. 2015
- https://en.wikipedia.org/wiki/Android_Studio
 - Android Studio, zadnji dostop 18. 9. 2015
- <http://www.gimp.org/>
 - Gimp, zadnji dostop 17. 9. 2015
- <https://sl.wikipedia.org/wiki/GIMP>
 - Gimp, zadnji dostop 17. 9. 2015
- <https://sl.wikipedia.org/wiki/XML>
 - XML, zadnji dostop 17. 9. 2015
- <http://stackoverflow.com/>
 - Programerske rešitve, zadnji dostop 22. 9. 2015
- C. Delessio, L. Darcey, S. Conder, Sams teach yourself Android application development in 24 hours, 2014, ISBN 978-0-672-33444-3



Univerza v Mariboru

Fakulteta za elektrotehniko,
računalništvo in informatiko
Smetanova ulica 17
2000 Maribor, Slovenija



IZJAVA O AVTORSTVU

Spodaj podpisani/-a

UROŠ BRUMEN

z vpisno številko

E1003495

sem avtor/-ica diplomskega dela z naslovom:

MOBILNA APLIKACIJA

ZA POMOČ STAREJŠIM PRI KLICANJU IN PISANJU
SMS-OV

(naslov diplomskega dela)

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal/-a samostojno pod mentorstvom (naziv, ime in priimek)

RED. PROF. DR. PETER KOKOL

in somentorstvom (naziv, ime in priimek)

- 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 DKUM.

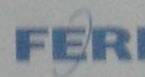
V Mariboru, dne 29.9.2015

Podpis avtorja/-ice:

Brumen



Univerza v Mariboru
Fakulteta za elektrotehniko,
računalništvo in informatiko
Smetanova ulica 17
2000 Maribor, Slovenija



IZJAVA O USTREZNOSTI ZAKLJUČNEGA DELA

Podpisani mentor :

RED. PROF. DR. PETER KOKOL

(ime in priimek mentorja)

in somentor (eden ali več, če obstajata):

(ime in priimek somentorja)

Izjavljam (-va), da je študent

Ime in priimek: UROŠ BRUMEN

Vpisna številka: E1003495

Na programu: RAČUNALNIŠTVA IN INFORMACIJSKE TEHNOLOGIJE, UNI

izdelal zaključno delo z naslovom:

MOBILNA APLIKACIJA ZA POMOČ STAREJŠIM PRI KLICANJU IN PISANJU SMS-OV

(naslov zaključnega dela v slovenskem in angleškem jeziku)

MOBILE APPLICATION TO HELP OLDER FOR CALLS AND WRITING SMS

v skladu z odobreno temo zaključnega dela, Navodilih o pripravi zaključnih del in mojimi (najinimi oziroma našimi) navodili.

Preveril (-a, -i) in pregledal (-a, -i) sem (sva, smo) poročilo o plagiatorstvu.

Datum in kraj:
28. 9. 15

Podpis mentorja:

Datum in kraj:

Podpis somentorja (če obstaja):

Priloga:

- Poročilo o preverjanju podobnosti z drugimi deli.«



Univerza v Mariboru

Fakulteta za elektrotehniko,
računalništvo in informatiko

Smetanova ulica 17
2000 Maribor, Slovenija



**IZJAVA O ISTOVETNOSTI TISKANE IN ELEKTRONSKE VERZIJE ZAKLJUČNEGA
DELA IN OBJAVI OSEBNIH PODATKOV DIPLOMANTOV**

Ime in priimek avtorja-ice: UROŠ BRUMEN

Vpisna številka: E1003495

Študijski program: FERI-E UNI RAČUNALNIŠTVO IN INFORMACIJSKE TEHNOLOGIJE

Naslov zaključnega dela: MOBILNA APLIKACIJA ZA POMOČ STAREJŠIM
PRI KLICANJU IN PISANJU SMS-OV

Mentor: RED. PROF. DR. PETER KOKOL

Somentor: _____

Podpisani-a UROŠ BRUMEN izjavljam, da sem za potrebe arhiviranja oddal elektronsko verzijo zaključnega dela v Digitalno knjižnico Univerze v Mariboru. Zaključno delo sem izdelal-a sam-a ob pomoči mentorja. V skladu s 1. odstavkom 21. člena Zakona o avtorskih in sorodnih pravicah dovoljujem, da se zgoraj navedeno zaključno delo objavi na portalu Digitalne knjižnice Univerze v Mariboru.

Tiskana verzija zaključnega dela je istovetna z elektronsko verzijo elektronski verziji, ki sem jo oddal za objavo v Digitalno knjižnico Univerze v Mariboru.

Zaključno delo zaradi zagotavljanja konkurenčne prednosti, varstva industrijske lastnine ali tajnosti podatkov naročnika: _____ ne sme biti javno dostopno do _____ (datum odloga javne objave ne sme biti daljši kot 3 leta od zagovora dela).

Podpisani izjavljam, da dovoljujem objavo osebnih podatkov, vezanih na zaključek študija (ime, priimek, leto in kraj rojstva, datum zaključka študija, naslov zaključnega dela), na spletnih straneh in v publikacijah UM.

Datum in kraj: 29.9.2015, MARIBOR Podpis avtorja-ice: Brumen

Podpis mentorja: _____
(samo v primeru, če delo ne sme biti javno dostopno)

Podpis odgovorne osebe naročnika in žig:
(samo v primeru, če delo ne sme biti javno dostopno) _____