Mobilna aplikacija "Moje dijete"

Angebrandt, Gabriela

Undergraduate thesis / Završni rad

2016

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:124722

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

zir.nsk.hr

Image not found or type unknown

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
ELEKTROTEHNIČKI FAKULTET

Sveučilišni studij računarstva

MOBILNA APLIKACIJA „MOJE DIJETE“

Završni rad

Gabriela Angebrandt

Osijek, 2016.

Sadržaj:

1. UVOD 1

1. 1. Zadatak završnog rada 2

2. ANDROID PLATFORMA 3

2. 1. Povijesni razvoj 3

2. 2. Arhitektura 3

2. 3. Verzije Androida 5

2. 4. Značajke Androida 5

3. IZRADA APLIKACIJE POMOĆU RAZVOJNE OKOLINE „ANDROID STUDIO“ 7

4. RAZVOJ APLIKACIJE 10

4. 1. Osnovna struktura aplikacije 10

4. 2. Glavni dio 11

4. 3. Baza podataka 21

4. 4. Podsjetnik 22

5. ZAKLJUČAK 25

LITERATURA 27

SAŽETAK 28

ABSTRACT 29

ŽIVOTOPIS 30

1. UVOD

Korištenje mobilnih telefona postalo je svakodnevna rutina ljudi modernoga doba. Kako bi korištenje telefona bilo što zabavnije i kako bi se postigla što veća korisnost mobitela, razvijeni su pametni mobilni telefoni. Postoji više podloga za pametne mobilne telefone, najvažnije su iOS u vlasništvu tvrtke Apple Inc., Windows Phone od tvrtke Microsoft te Android čiji je vlasnik tvrtka Google Inc.. Za ovaj rad odabrana je podloga Android operacijskog sustava. Poznat je po svojoj jednostavnosti i učikovitosti te je time postao veliki konkurent ostalim operacijskim sustavima za mobilne. Kako bi se olakšao život čovječanstvu, razvijaju se aplikacije koje pomažu ljudima ili zabavljaju iste u svakodnevici pa će tako u ovom završnom radu biti izrađena aplikacija „Moje dijete“.

Aplikacija „Moje dijete“ namjenjena je za majke, pa čak i očeve, koji imaju želju i volju voditi dnevnik aktivnosti i razvoja svoje djece. Između ostaloga, postojati će mogućnost pohranjivanja različitih fotografija vezanih za sam proces odrastanja djeteta. Kako roditelji ne bi zaboravili na važne datume (kao što su datum rutinskog cijepljenja, odlazak zubaru i slično), postojati će podsjetnik koji će podsjećati roditelja o obavezi.

Zadatak ovog završnog rada je najprije dobro proučiti arhitekturu operacijskog sustava Android te se upoznati sa njegovim značajkama. Na početku glavnog dijela završnog rada ukratko će biti objašnjena sama arhitektura i biti će navedene verzije u razvoju ovog operacijskog sustava. Također, biti će spomenuto i nešto iz njegove povijesti. Nakon uvodnog teorijskog dijela započeti će objašnjavanje praktične obrade zadatka, odnosno izrade aplikacije i pojašnjenje glavnih komponenata Android aplikacije. Aplikacija mora biti pisana u Java programskom jeziku te će se razvoj aplikacije vršiti u programu Android Studio. Najprije se mora kreirati projekt u programu. Stvaranjem novih dijelova aplikacije, tj. aktivnosti biti će omogućeno lakša snalažljivost u aplikaciji i korištenje iste. Dodavanjem fragmenata olakšati će se pristup pojedinim naredbama aplikacije. Dizajnirati će se izgled same aplikacije, odabrati će se prigodne boje za pojedini dio aktivnosti te će biti uvršten i kalendar s mogućnošću zadavanja podsjetnika. Bit će omogućen postupak aktiviranja mobilne kamere ili izbora iz galerije fotografija kako bi se pohranjivale fotografije vezane za dijete. Kada aplikacija bude završena, roditelji će moći spremiti datume kada su djeca cijepljena, kada su dobili vodene kozice, kada im je ispao prvi zubić, koja im je bila prva progovorena riječ i još štošta. Naposlijetku, biti će ispisan popis literature korišten u ovom završnom radu.

Strukturu završnog rada čine uvod, glavni dio rada zaključak, popis literature, sažetak na hrvatskom i engleskom jeziku, životopis te prilozi. Uz završni rad prilaže se i elektronička verzija završnog rada na CD-u.

1. 1. Zadatak završnog rada

U okviru ovog završnog rada potrebno je upoznati se s značajkama i arhitekturom Android podloge te ju ukratko opisati. Izraditi aplikaciju za Android operacijski sustav koja će pomagati roditeljima u bilježenju važnih podataka i situacija koje se događaju u razvoju njihovog djeteta uz mogućnost snimanja i pohranjivanja fotografija te zadavanja podsjetnika. Aplikacija treba imati mogućnost definiranja više osoba za koju se podaci unose i pohranu tih podataka u datoteku.

2. ANDROID PLATFORMA

2. 1. Povijesni razvoj

Android je operacijski sustav koji je u današnje vrijeme najčešće korišten kao operacijski sustav za mobilne telefone i tablet računala. Prilagođen je kako bi se koristio i na Smartbook i Netbook računalima te i na modernim ručnim satovima. 2003. godine osnovana je tvrtka Android Inc. kojoj je glavni cilj bio razviti programsku potporu za najnoviji oblik telefona – pametne mobilne telefone. Nedugo nakon njihovog osnutka, tvrtku kupuje Google te 2007. godine osniva grupu OHA (engl. Open Handset Alliance) koja je imala zadatak napraviti javni standard za pametne telefone. U OHA udružuje se preko 30 tvrtki različitih područja djelatnosti vezanih za mobilne telefone te njihovim radom stvorena je nova platforma za pametne mobilne telefone – Android.

2. 2. Arhitektura

Baza Androida je operacijski sustav Linux jezgra 2.6 te je, kao i sam Linux, Androidov kod javan. Za pisanje programa za Android platformu koristi se programski jezik Java. Budući da je kreiran za jednostavan rad i pokretanje, ima ograničenje na resursima vezanim za kapacitet memorije i moć procesora te se u tom pogledu razlikuje od platformi koje su kreirane za osobna računala. No, postoje i neke razlike u odnosu na Linux. Naime, Android ne sadrži X-Window sustav te neke od GNU biblioteka koje su nepotrebne za ovu jednostavnost. Također, u Linux-u je sloj aplikacija puno bliže sloju jezgre dok je kod Androida jezgra duboko skrivena u radnom okruženju.

Osnovni dijelovi njegove arhitekture su:

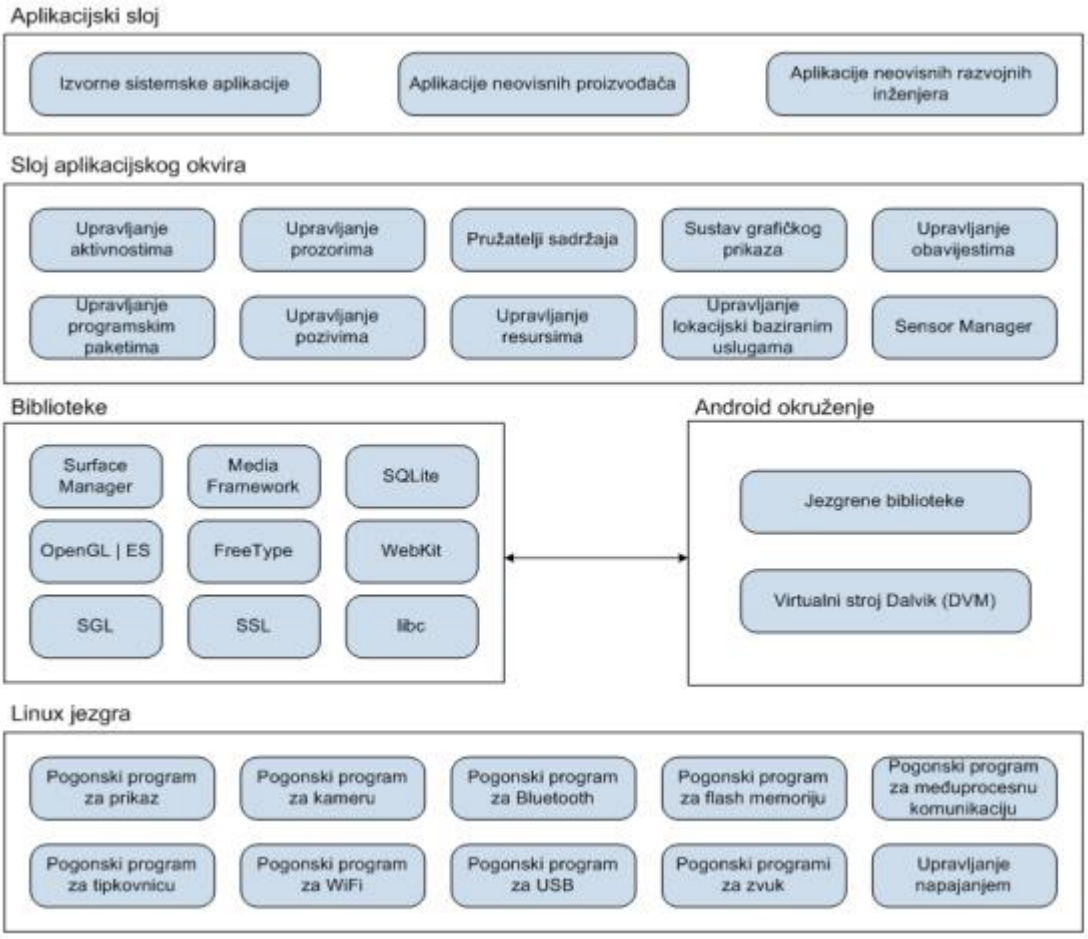
a) Linux kernel – zadatak joj je brinuti se o upravljanju procesima, memorijom te ostalim sklopovskim dijelovima.
b) Sustavske biblioteke – biblioteke su napisane u programskim jezicima C i C++ kako bi bili zadaci što brže izvršeni i prilagođeni različitim uređajima. Posebna točka je tzv. „Dalvik virtulani sustav“ (engl. Virtual Machine - V.M.) koji omogućava izvršavanje više zadataka virtualnih strojeva istrovremeno da se

iskoristi sav potencijal Linuxa. To omogućuje da se svaka aplikacija izvodi u posebnom virualnom okruženju.

c) Aplikacijske biblioteke – biblioteke koje se koriste za razvijanje i izvođenje određenih aplikacija

d) Aplikacije – najviši sloj arhitekture. Izvođenje aplikacija na mobilnom telefonu razlikuje se od izvođenja na osobnim računalima. Umjesto paralelnog izvođenja i jednakog prioriteta, na Android platformi aplikacije izvršavaju se proiritetno zauzimanjem cijele površine ekrana.

Na slici 2.1. prikazani su slojevi arhitekture Androida.



Sl. 2.1. *Građa Androida.*

Izvor: [6]

Tu postoji i Java izvorno sučelje (engl. JNI – Java Native Interface) između izvornog koda izvođenog izvan JVM i Java koda za JVM te prevodioc JIT (engl. Just-in-time) koji prevodi Dalvik kod u bolji i učinkovitiji strojni kod, prema [1].

2. 3. Verzije Androida

Android brzo evoluira pa su se tako do sada razvile sljedeće verzije:

- 2008. /2009.g. – Android 1.0 (Apple Pie) , Android 1.1 (Banana Bread), Android 1.5 (CupCake), Android 1.6 (Donut), Android 2.0 (Eclair)
- 2010.g. - Android 2.2 (Froyo), Android 2.3 (Gingerbread)
- 2011.g. – Android 3.0 (Honeycomb), Android 4.0 (Ice-Cream Sandwich)
- 2012./2013.g. – Android 4.2 (Jelly Bean), Android 4.4 (KitKat)
- 2014.g. – Android 5.0,5.1 (Lollipop)
- 2015.g. - Android 6.0 (Marshmallow)

Prvi Android pametni mobilni telefon bio je G1 HTC – Dream, verzija Android 3.0 (Honeycomb) je prva inačica za tablet uređaje (posebna zbog prilagoljivosti na velike zaslone) dok je verzija Android 5.0 (Lollipop) redizajniranija od svih inačica u području sučelja. Najzastupljenija inačica Androida, prema [8], je Lollipop. Budući da je u vrijeme početka pisanja ovog rada najzastupljenija inačica bila KitKat, ona je korištena kao platforma sustava.

2. 4. Značajke Androida

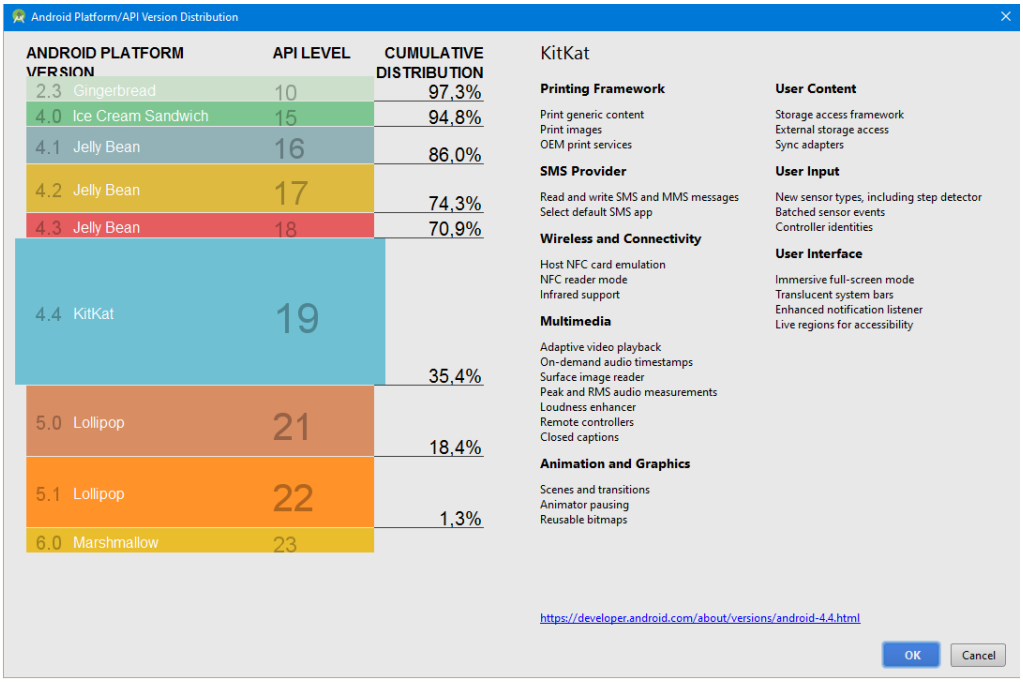
Prema[5], Android ima mnogo značajki. Neke važnije od njih su:

1. prilagođenost za rad na uređajima s 2D i 3D grafičkom knjižicom
2. za pohranjivanje podataka koristi se *SQLite* sustav za upravljanje bazom podataka koji se ponaša kako dio aplikacije koja prolazi proces pristupanja bazi poadataka
3. jednostavno upravljanje i povezivanje s drugim uređajima primjenjivanjem IDEN, CDMA, GMS/EDGE, LTE, Bluetooth, WiFi i drugih tehnologija.
4. prijenos podataka pomoću Bluetooth tehnologije između dva uređaja
5. komunikacija porukama izvršava se SMS i MMS servisima i to na 57 različitih jezika
6. korištenje Dalvik virtualnog stroja
7. podržavanje mnogih audio (.mp3, .mp4, .mid, ...) i slikovnih (.jgp, .gif, ...) formata
8. podrške za dodatne hardvere, *streaming* i veoma popularnu glasovnu pretragu

Veliki naglasak stoji na platformi koja je odabrana za razvoj aplikacije jer ako korisnik pokuša pokrenuti aplikaciju na nekoj drugoj platformi (npr. iOS), aplikacije se neće moći pokrenuti zbog toga što je iOS potpuno drugačije formiran sustav u odnosu na Android. iOS koristi drugačiji programski jezik, virtualne strojeve, sustave za bazu podataka itd. Isto vrijedi i za Windows Phone, potpuno je drugačiji od Androida. Zato će aplikaciju „Moje dijete“ moći korisiti samo korisnici Android pametnih telefona.

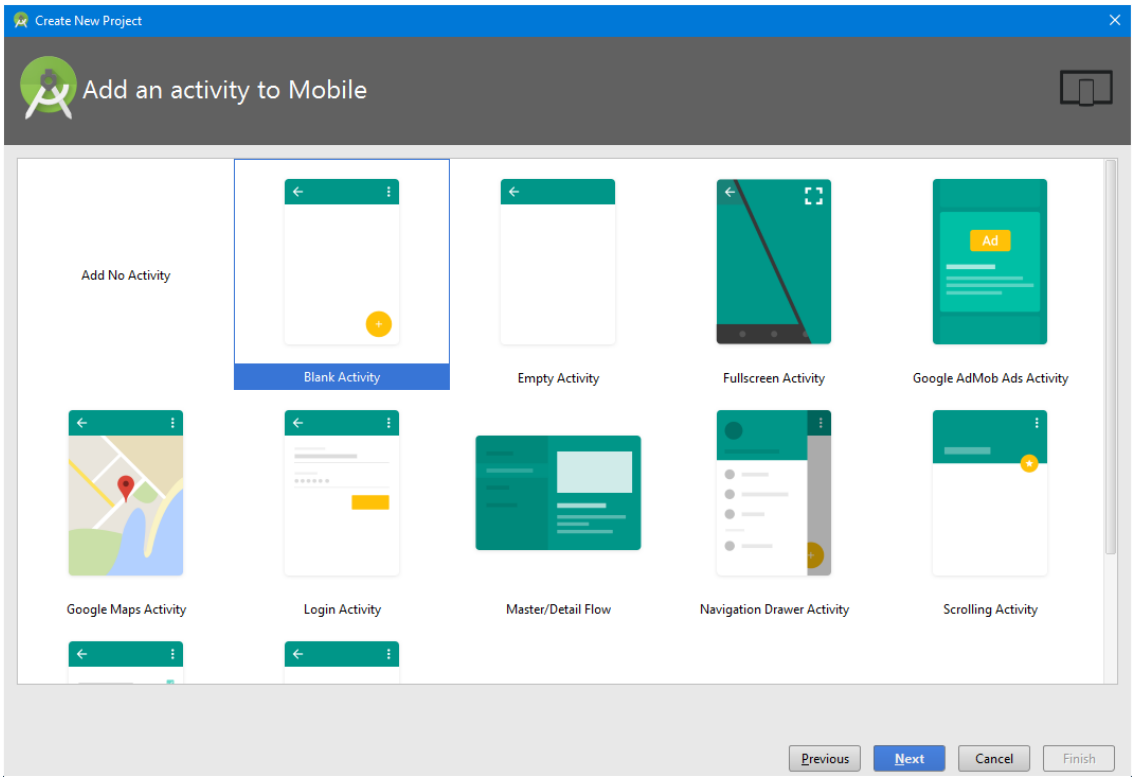
3. IZRADA APLIKACIJE POMOĆU RAZVOJNE OKOLINE „ANDROID STUDIO“

Za izradu mobilne aplikacije korišten je program „Android Studio“. Za pisanje programskog koda u programskom jeziku Java potrebno je uz Android Studio instalirati pakete JDK (engl. Java Development Kit) i JRE (engl. Java Runtime Environment) inačice 8. Zatim je kreiran projekt pod nazivom „Moje dijete“. Za platformu aplikacije odabrana je verzija Android 4.4 (KitKat) što se može vidjeti na slici 3.1.. Kriterij za odabir verzije platforme bio je postotak korisnika koji mogu pokrenuti aplikaciju na svom mobilnom telefonu koji je iznosio 70,9 %. Točnije, ovaj postotak govori da 70,9 % mobilnih telefona u svijetu ima platformu Android 4.4 ili neku od ranije razvijenih verzija Androida te im to omogućava mogućnost pokretanja ove aplikacije.



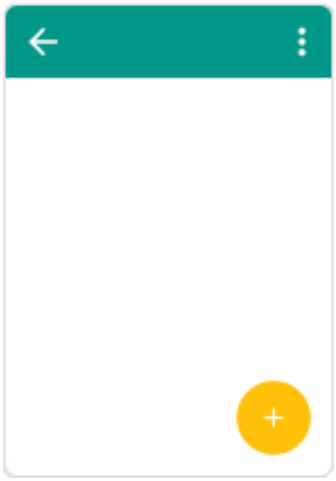
Sl. 3.1. *Odabir verzije*

Sljedeći korak razvoja aplikacije je odabir izgleda početne aktivnosti aplikacije. Početna aktivnost je ono što korisnik prvo vidi kada pokrene aplikaciju.



Sl. 3.2. *Prozor za odabir.*

Pored svih mogućih prijedloga, odabrana je prazna aktivnost (engl. *Blank Activity*) što prikazuje slike 3.2. i 3.3.

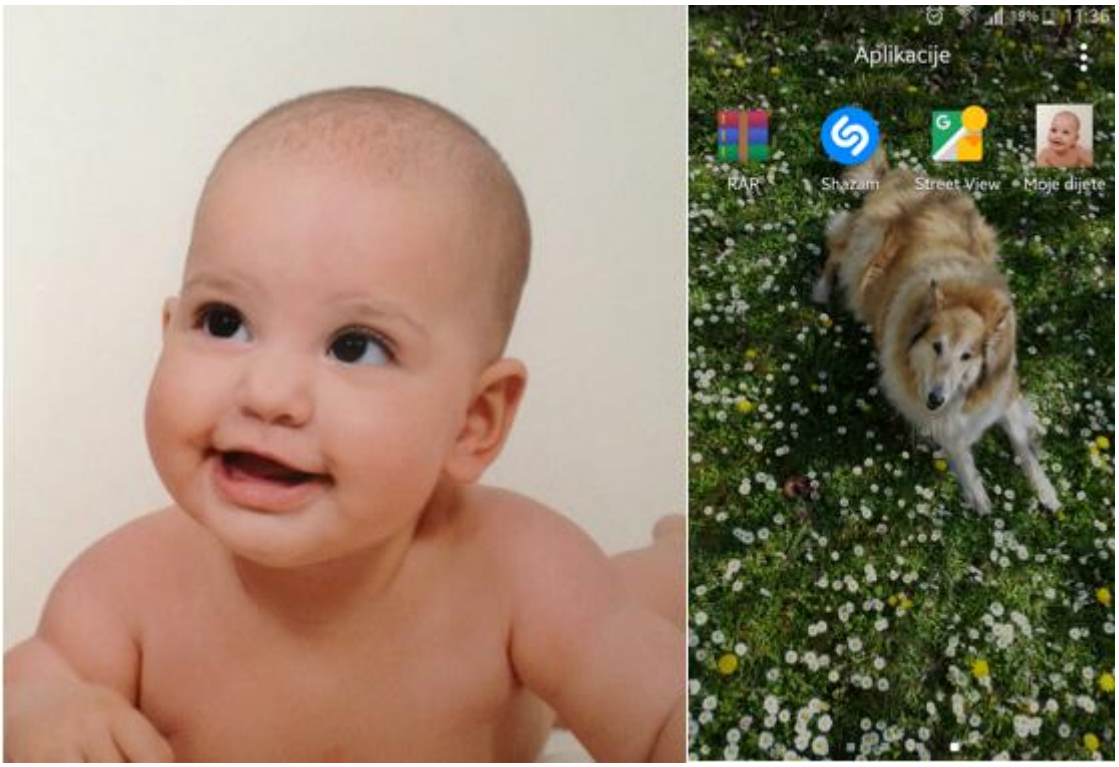


Sl. 3.3. *Prikaz početne aktivnosti(engl. Blank activity).*

Kod kreiranja aktivnosti važno je napisati naslov aktivnosti, u ovom slučaju „Moje dijete“. Uobičajene aplikacije današnjice imaju nekoliko aktivnosti koje su međusobno vrlo usko

povezane. Odabiranjem određenog sadržaja jedne aktivnosti pokreće se nova aktivnost. Tako postoji i naredba (tipka) za povratak. Ovaj proces odvija se preko stoga po principu „Last In First Out“ i na taj način prepoznaje se aktivnost koja je bila pokrenuta u prošlom koraku te se ovom naredbom može prekinuti trenutna i započeti prošla aktivnost. Tipka za povratak stvorena je linijama koda u AndroidManifest.xml mapi. Ona je definirana tako što smo odredili koja je roditeljska aktivnost. Linija s instrukcijom zapisana je u tijelu određene aktivnosti u Android Manifestu. Nakon što Android shvati koja je aktivnosti roditeljska, on sam stvara tipku koja se vraća do određene aktivnosti.

U ovoj datoteci nalaze se naslovi za svaku novootvorenu aktivnost i pojedinačne teme za drugačije izglede ekrana. Također, tu se nalazi i naredba koja postavlja sliku djeteta (slika 3.5.) za ikonu aplikacije.



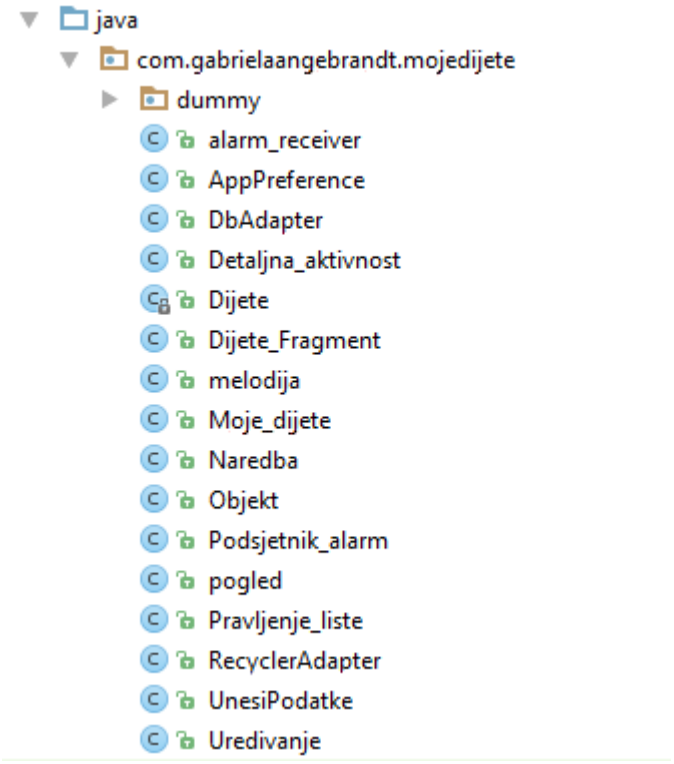
Sl. 3.5. *Ikona aplikacije,*

Izvor: izradio autor

4. RAZVOJ APLIKACIJE

4. 1. Osnovna struktura aplikacije

Aplikacija se sastoji od nekoliko sastavnih dijelova koji međusobno komuniciraju. Aktivnosti (engl. *Activites*) su zasloni s korisničkim sučeljem. Svaka aktivnost u Android studiu se prikazuje kao Java klasa u koju se zapisuju različite funkcije. Uobičajeno je da aplikacija ima nekoliko aktivnosti, a samo jedna od njih je glavna, odnosno početna aktivnost koju korisnik prvu vidi kada pokrene aplikaciju. Aktivnost imaju mogućnost da iz jedne aplikacije pokrenu drugu aplikaciju (npr. u ovoj aplikaciji pokreće se aplikacija kamere na mobilnom telefonu). Na slici 3.1. prikazane su sve korištene aktivnosti ovog rada.



Sl. 4. 1. *Aktivnosti ove aplikacije*

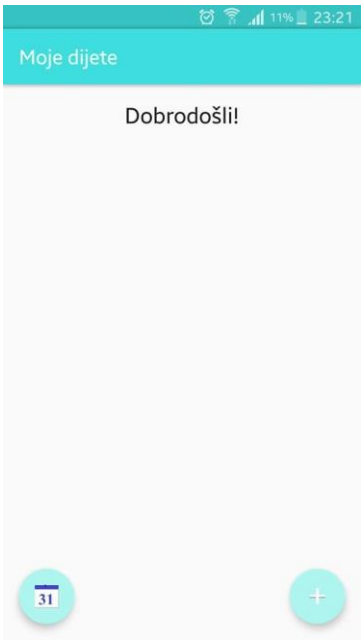
Sljedeća komponenta aplikacije su usluge (engl. *Service*). Usluge koje se izvode u pozadini aplikacije ne prikazuju se korisniku i služe za izvršavanje udaljenih procesa (npr. u ovoj aplikaciji postoji usluga za reproduciranje zvuka alarma). Usluge se pokreću u aktivnostima i moraju se deklarirati u Android Manifest mapi kao i aktivnosti.

Treća komponenta aplikacije su davatelji sadržaja (engl. *Content providers*). Davatelj sadržaja upravlja podacima aplikacije koji su pohranjeni na bilo kojoj lokaciji kojoj se može pristupiti. Preko davatelja sadržaja aplikacije vrše se radnje nad podacima uz određena dopuštenja. U slučaju korištenja baze podataka unutar samo jedne aplikacije, nije potrebno koristiti davatelje sadržaja.

Zadnja komponenta aplikacije su namjere (engl. *Intent*). Zadatak namjere je da zatraži izvršenje operacije od neke druge komponente aplikacije. Postoje dvije vrste namjera: implicitna i eksplicitna namjera. Eksplicitne namjere se koriste kada želimo pokrenitu neku od komponenata unutar vlastite aplikacije, dok se implicitne namjere koriste za definiranje radnje koju prepoznaje komponenta iz neke strane aplikacije te ju ona izvršava. U ovoj aplikaciji korisilo se puno namjera, najčešće za otvaranje sadržaja drugih aktivnosti (engl. *startActivity()*), zatim za start usluge (engl. *startService()*) te za emitiranje (engl. *sendBroadcast()*). Teorija se temeljena prema [4,9].

4. 2. Glavni dio

Razvijanje aplikacije temljeli se na stvaranju klasa i njihovih pripadajućih *layouta*. Klase ili tzv. aktivnosti međusobno se povezuju i na taj način ostvarujemo otvaranje novog izgleda ekrana pritiskom na određenu tipku. Prva klasa koja je napravljena naziva se Moje dijete. Ova početna aktivnost ima dva akcijska gumba prema slici 5.1..



Sl.4.2. *Početna aktivnost „Moje dijete“*

Jedan od njih vodi u aktivnost namjenjenu za postavljanje alarma koji podsjeća roditelje na obaveze oko djeteta. Drugi akcijski gumb vodi na aktivnost koja se zove „Unesi podatke“. Ova aktivnost služi za unošenje osnovnih podataka vezanih za dijete, a to su: ime, prezime i odabir fotografije. Nakon što korisnik unese ove podatke i spremi ih, otvara se prozor glavne aktivnosti koji sada prikazuje prikaz fotografije i pored nje ime i prezime djeteta. Ova dva podatka strukturirana su kao poseban *layout* te taj strukturirani objekt na neki način oponaša gumb s fotografijom. Pritiskom na ovaj objekt dolazi do otvaranja nove aktivnosti.

Akcijski gumb mora imati svoje identifikacijsko ime (npr. fab, fab1...) preko kojeg se može pozvati iz određenog *layouta*. Zatim se poziva funkcija koja definira što će se dogoditi u slučaju da korisnik klikne na gumb. U ovom slučaju, poziva se dohvaćanje konteksta aktivnosti „Unesi podatke“ i , u sljedećem gumbu, dohvaćanje konteksta od „podsjetnik_alarm“(slika. Svaki akcijski gumb ima posebnu sliku koje su izrađene u programu Photoshop.

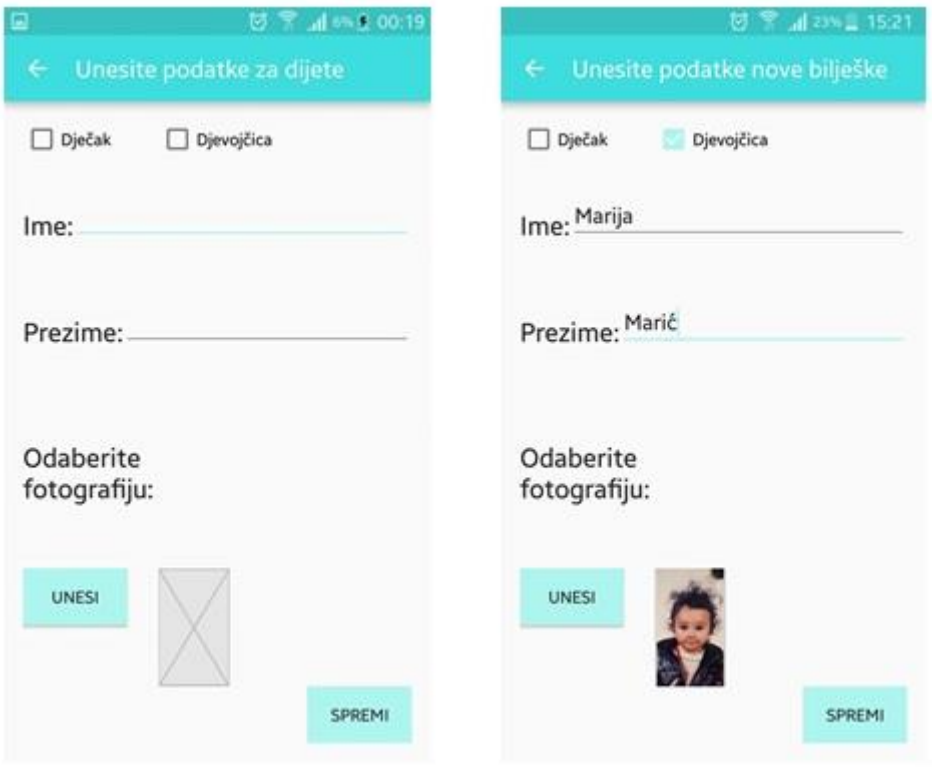


Sl.4.3. *Slike za akcijske gumbove.*

U početnoj aktivnosti također je definiran i tektualni okvir u kojem se korisniku izražava dobrodošlica u aplikaciju. Naslov početne aktivnosti isti je kao i naziv aplikacije pa je tako u Android Manifest mapi definiran naslov „Moje dijete“ u tijelu ove aktivnosti. U *layoutu* vezanom za početnu aktivnost definiran je izgled i smještaj alatne trake te smještaj akcijskih gumbova u prozoru aktivnosti. Također, odabrane su i slike za svaki akcijskih gumb tako što su pozvane iz mape *drawable* koja je automatski definirana od strane Androida. Slike smo prethodno izradili i dodali u mapu.

Kako bi roditelji mogli unositi podatke za svoju djecu, prvo je morala biti definirana klasa za unošenje osnovnih podataka za dijete (slika 4.4.). Zadatak ove klase je da pohrani unešene podatke u bazu podataka te da, nakon pohranjivanja, pozove stvaranje objekta u početnoj aktivnosti koji se ponaša kao gumb s fotografijom (iako je on zapravo stvoren kao poseban *layout*).

Najprije su u *layout* mapi relativnog rasporeda, koja je povezana s ovom aktivnošću, definirani objekti (njihov izgled i smještaj u prozoru). Tako tu postoje 2 polja, tzv. kvačice koje definiraju spol djeteta. Korisnik može označiti ili kvačicu koja govori da je dijete djevojčica ili da je dijete dječak. Zatim unosi ime i prezime djeteta u zasebna tektualna polja. Nakon toga, korisnik može unijeti fotografiju djeteta. Postoje dva gumba za odabir fotografije. Prvi gumb se zove „Unesi“ i on služi za unošenje fotografije na način da kada korisnik pritisne gumb, poziva se pokretanje ugrađene kamere u mobilnom telefonu. Tada korisnik uslika dijete i potvrđuje korištenje uslikane fotografije. Uslikana fotografija ispisuje se u prostor za pregled fotografije koji se nalazi pored gumba za unošenje iste. Sve dok se korisnik nalazi u ovom prozoru te dok još podaci nisu spremljeni, on može više puta prepravljati podatke. Nakon što se podaci spreme pritiskom na gumb „Spremi“, podaci su trajno pohranjeni u bazi podataka te ih više nije moguće mijenjati te nije moguće odabrati već uslikanu fotografiju iz galerije. Dodatnim radnjama može se omogućiti i ovaj dio, ali neće biti izveden u ovom završnom radu. Sve fotografije koje će biti uslikane pozivanjem kamere iz aplikacije spremaju se u i u galeriju Android telefona.



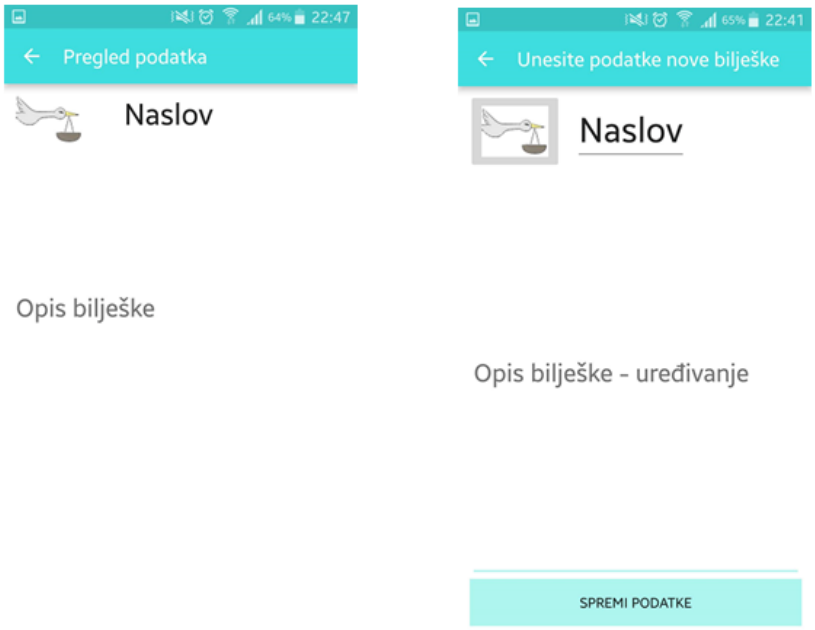
Sl. 4.4. *Aktivnost „Unesi podatke“*

Prilikom definiranja objekata korišteni su sljedeći parametri: visina, širina, identifikacija i poravnanje (s obzirom na rubove prozora). Za neke objekte, kao što su gumbi, definiran je i tekst koji se pojavljuje na objektu te boja pozadine gumba. Akcije gumbova su razrađene u aktivnosti

pa tako gumb „Spremi“ sprema podatke u bazu podataka te poziva otvaranje početne aktivnosti u kojoj poziva ispis *layouta* u obliku objekta s imenom i unesenom fotografijom djeteta. Objekt s imenom, prezimenom i fotografijom definiran je u „objekt.xml“ te se stvara pomoću tzv. *RecyclerView*-a i ima mogućnost izvšavanja klika na njega. *RecyclerView* se povezuje s bazom podataka putem *RecyclerAdaptera*. Gumb „Unesi“ poziva pokretanje aplikacije kamere.

Aktivnost „Dijete“ stvorena je s namjerom da sadrži detaljne podatke vezane za rođenje, razvoj, prehranu, izgled i zdravlje djeteta. Podaci se strukturiraju u popis podataka te za svaki podatak treba biti omogućeno pregledavanje i uređivanje istog. Zato se koristi fragment koji se implementira u ovu aktivnost. Fragmentom upravlja *FragmentManager* tako da započinje i izvršava različite transkacije u aktivnosti „AppPreference“ (prema [4]). U *layoutu* aktivnosti relativnog rasporeda definirano je jedno tekstualno polje koje ima zadatak da obavjesti korisnika na koji način da pritisne bilješku s popisa kako bi ju uređivao. Za ovo tektualno polje definirana je visina i širina polja, tekst koji se nalazi u polju, veličina teksta, boja teksta, identifikacija te poravnanje u prozoru. Zatim je napravljena implementacija fragmenta pod nazivom „Dijete_Fragment“ tako što je definiran kao objekt fragmenta ispod tekstualnog polja. Za fragment su definirani ovi parametri: visina, širina, identifikacija i naziv fragmenta.

Fragment se koristi kako bi se aktivnost podijelila na manje dijelove, komponente, koji se mogu dinamički dodavati ili uklanjati (pomoću *Fragment Manager* i *Fragment Transaction*). Razlika između fragmenta i aktivnosti je u tome što se fragment ne mora definirati u Android Manifest mapi. Fragment omogućava da se svakim njegovim dijelom može manipulirati. Vrsta ovog fragmenta je lista. Lista je stvorena u aktivnosti „Pravljenje_liste“ i to na način da su definirane pozicije redaka u listi te izgled svakog retka. Svaki redak sadrži sliku bilješke, naslov i jedan dio opisa. Ako je opis predugačak, na kraj se stavljaju tri točkice koje označavaju nastavak. Izgled retka definiran je *layoutom* „opis_jednog_retka“. Kako bi se svaka bilješka mogla pregledati normalnim pritiskom na istu, napravljena je zasebna aktivnost koja daje predložak izgleda prozora za pregled bilješke. U *layoutu aktivnosti* „pogled“ koji je relativnog rasporeda stvoren je objekt prikaza slike koji je smješten u lijevi kut prozora, zatim s desne strane do njega nalazi se tekstualno polje koje ispisuje naslov bilješke (npr. Prva izgovorena riječ), a ispod slike i naslova nalazi se tekstualno polje za opis bilješke (npr. Prva izgovorena riječ je bila : mama). Parametri su jednaki kao i u prošlim definiranjima (visina, širina, identifikacija, poravnanje...). Opširno tekstualno polje opisa može se pregledavati korištenjem kotačića na računalnom mišu. *Layout* „pogled“ prikazan je na slici 4.5. lijevo.



Sl.4.5. *Prozor pregleda i uređivanja bilješke*

Na prilično sličan način stvorena je i aktivnost i *layout* za uređivanje podataka(slika 4.5. desno). *Layout* je također relativnog rasporeda te su objekti smješteni jednako kao i kod *layouta* za pregled podatka. Jedina razlika je da su ovdje korištena tekstulna polja za uređivanje, tj. upisivanje (dodavanje ili izmjena) vlastitog teksta. Kako se bilješke razlikuju (bilješke vezane za porod, razvoj, prehranu, izgled i zdravlje) u aktivnosti „Uređivanje“ definirana je funkcija koja stvara dijaloški okvir koji se pokreće prilikom klika na sliku u prozoru uređivanja. Ovaj dijaloški okvir prikazuje kategorije bilješki te omogućuje odabir jedne od njih. Nakon što korisnik odabere određenu kategoriju, ikona bilješke mijenja se u zadanu sliku određene kategorije na način da funkcija dohvaća sliku iz mape *drawable*. Ovo je ostvareno korištenjem *switch case* naredbe te je izgled prozora prikazan na slici 4.6. Ikone korištene za kategorije nalaze se na slici 4.7.



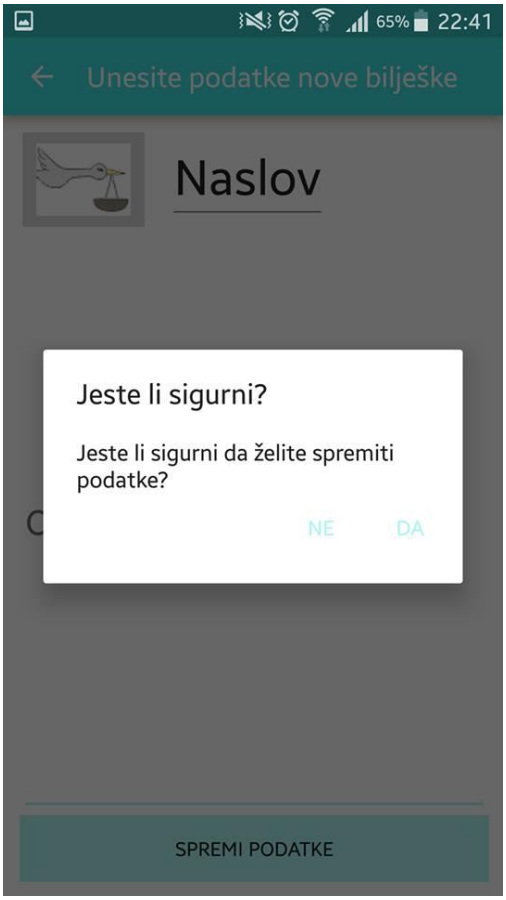
Sl.4.6. *Odabir kategorije*



Sl.4.7. *Ikone za kategorije*

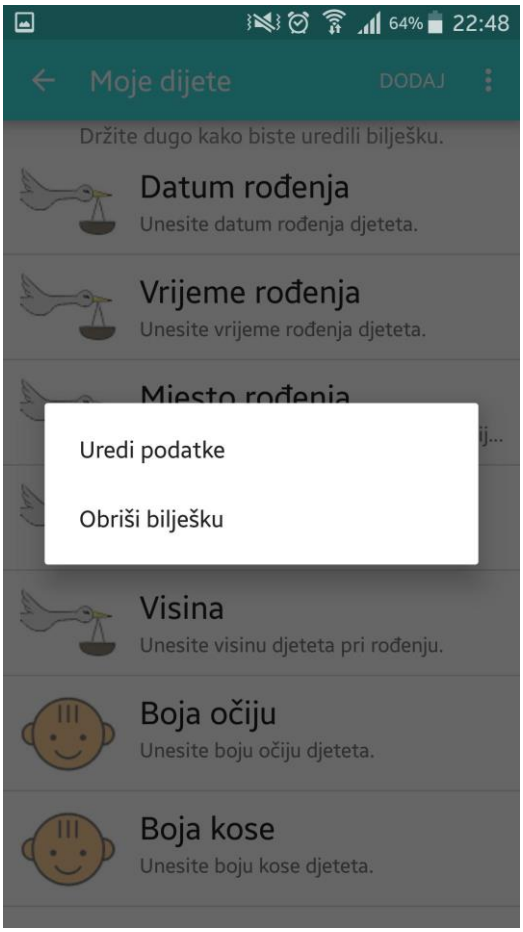
Na kraju ostaje samo spremiti bilješku pritiskom na gumb „Spremi“. Kako korisnik ne bi uneo krive podatke, otvara se novi dijaloški okvir (slika 4.8.) koji pita korisnika je li siguran da želi spremiti podatke. Ovaj dijaloški okvir također je definiran u aktivnosti „Uređivanje“ tako da su postavljena dva gumba, pozitivni i negativni. Ako korisnik pritisne pozitivni gumb u dijaloškom okviru, poziva se adapter baze podataka (o kojemu će nešto više biti rečeno kasnije) te se na taj način uređena bilješka automatski sprema u bazu podataka te otvara prethodna aktivnost s fragmentom, odnosno popisom bilješki. Važno je napomenuti da je upisana i naredba za automatsko osvježavanje popisa prilikom svakog ponovnog otvaranja aktivnosti s

fragmentom. Ako korisnik pritisne negativni gumb dijaloškog okvira, dijaloški okvir se zatvara i ništa više se ne događa, korisniku je i dalje prikazan prozor uređivanja bilješke koju je netom prije uređivao.



Sl. 4.8. *Dijaloški okvir za spremanje*

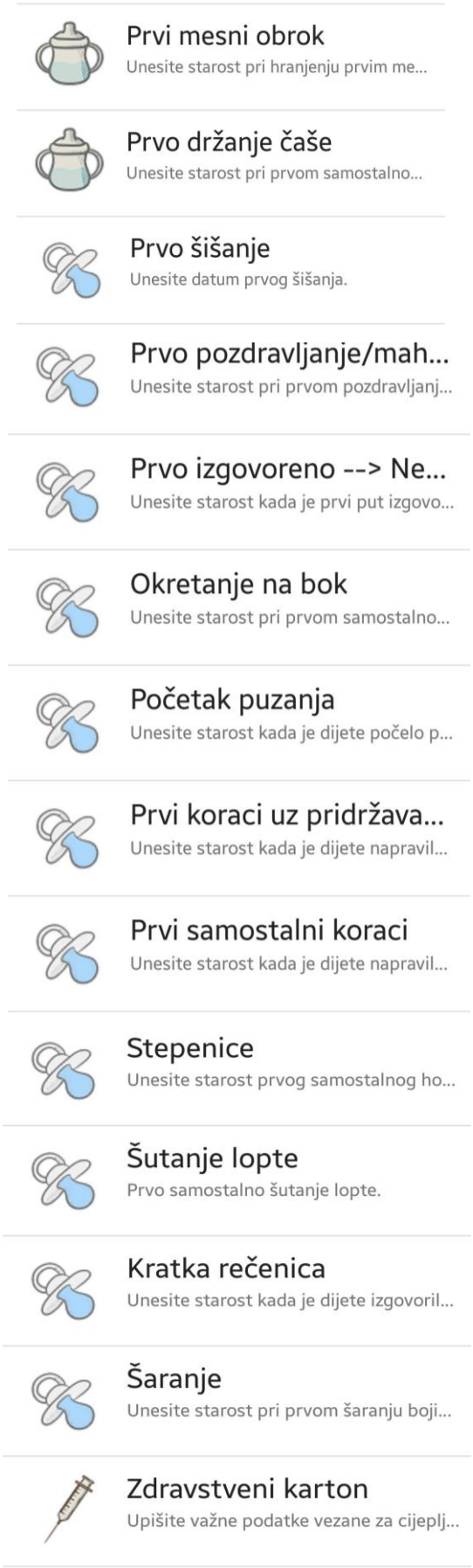
Kada korisnik drži dugi pritisak na bilješku izbacuje se još jedan dijaloški okvir. Ovaj dijaloški okvir sadrži dvije naredbe, naredbu za brisanje bilješke i za uređivanje. Ako korisnik pritisne naredbu za uređivanje, otvara se prozor aktivnosti za uređivanje. U slučaju odabiranja naredbe brisanja, bilješka se briše. Bilješka se briše na način da program prepoznaje poziciju retka popisa u kojemu se nalazi bilješka te briše redak na toj poziciji. Budući da se popis automatski osvježava, odmah je prikazano kako bilješka više ne postoji na popisu. Ovaj dijaloški okvir zove se *long_press_menu* jer se otvara na dugo držanje (slika 4.9.).



Sl. 4.9. *Dijaloški okvir s naredbama brisanja i uređivanja*

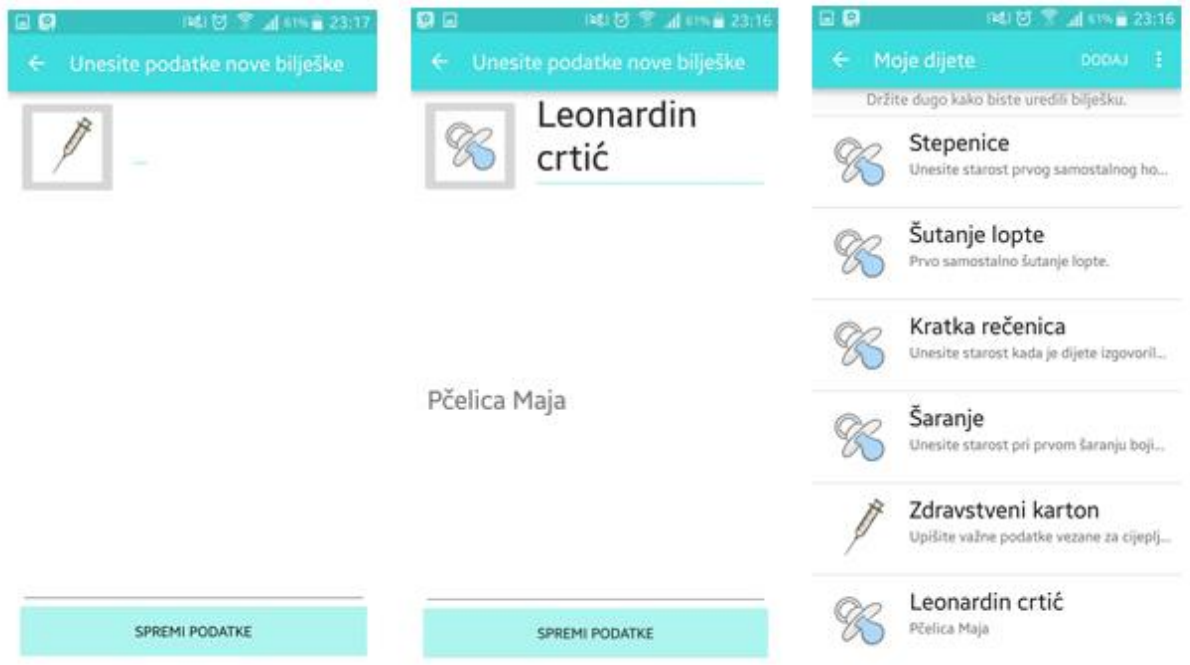
Budući da postoji naredba za brisanje bilješke, postoji i naredba za dodavanje nove bilješke. Ona se nalazi na alatnoj traci u aktivnosti „Dijete“ pored izbornika za postavke. Pritiskom na ovaj gumb, otvara se fragment za kreiranje nove bilješke prema već poznatom predlošku. U ovom prozoru korisnik može sam dodati naslov bilješke, opis i odabrati jednu od ponuđenih kategorija u koju on misli da bilješka pripada. Nakon unošenja podataka i odabiranja kategorije, preostaje samo spremiti podatke u bazu podataka, tj. potvrditi dijaloški okvir za spremanje podataka. Zatim se prikazuje popis bilješki na čijem se kraju nalazi novonastala bilješka. Da korisnik ne bi morao unositi puno bilješki i kako bi uštedio vrijeme, ispisan je popis fiksnih bilješki koje samo treba urediti/izmjeniti opis. Niz slika 4.10. prikazuje popis fiksnih bilješki koje se dobiju samom instalacijom aplikacije.


65% 22:42
Moje dijete
DODAJ
Držite dugo kako biste uredili bilješku.
Datum rođenja
Unesite datum rođenja djeteta.
Vrijeme rođenja
Unesite vrijeme rođenja djeteta.
Mjesto rođenja
Unesite naziv naselja gdje je rođeno dij...
Težina
Upišite kilažu djeteta pri rođenju.
Visina
Unesite visinu djeteta pri rođenju.
Boja očiju
Unesite boju očiju djeteta.
Boja kose
Unesite boju kose djeteta.
Datum prvog kupanja
Unesite datum prvog kupanja.
Prvi smješak
Unesite starost pri prvom osmijehu
Trajanje perioda hranjen...
Unesite do kada se dijete hranilo na pr...
Prvo držanje bočice
Unesite starost pri prvom samostalno...
Moja prva kašica
Unesite okus prve isprobane kašice.
Omiljeno voće
Unesite djetetovo omiljeno voće.
Omiljeno povrće
Unesite djetetovo omiljeno povrće.



Sl. 4.10. *Popis fiksnih bilješki*

U daljenjem razvijanju aplikacije bilo bi potrebno izvršiti testiranje na uzorku od nekoliko korisnika na temelju toga sviđa li im se više da se bilješke stavljaju na početak ili na kraj liste bilješki. Na slici 4.11. prikazan je postupak stvaranja nove bilješke.



Sl. 4.11. *Postupak kreiranja nove bilješke*

4. 3. Baza podataka

Da bi korisnik mogao pohranjivati podatke stvorena je baza podataka. Korišten je sustav za pohranjivanje u bazu podataka pod nazivom SQLite. Baza podataka se popunjava na način da se u bazi stvore tablice koje sadrže podatke koji se unose. Za ovaj rad potrebne su dvije tablice: jedna tablica pohranjuje podatke o detaljnim bilješkama za dijete, a druga pohranjuje osnovne podatke za dijete. U aktivnosti UnesiPodatke nalaze se tekstualna polja koja korisnik ispunjava s obzirom na stavku koja se traži. Klikom na gumb „Spremi“, ćelije u prvoj tablici se ispunjavaju. Za manipulaciju tablicom potreban je tzv. *Database Adapter* pa je tako stvorena aktivnost pod nazivom „DbAdapter“. U njoj su prvo definirani nazivi vrsta podataka kao što su naslov, opis, kategorija, ime itd. Zatim su vrste podataka povezane s tablicom. Definirana je baza podataka pod nazivom „sqlDb“ te su ispisane metode za manipulaciju tablicom. Metode su: stvaranje te otvaranje i zatvaranje tablice, metoda za stvaranje novih grupa podataka, metoda za brisanje grupa podataka i izmjenu istih te metode za stvaranje 2 niza podataka. U određenim metodama bio je potreban i pokazivač čija je svrha da pokazuje na određeni broj retka koji korisnik želi da

se ispiše. Potrebno je povezati DbAdapter s aktivnosti u kojoj se spremaju, odnosno ispisuju podaci. U aktivnosti se povezuju tablica i konkretni uneseni podaci u tekstualna polja. Prilikom ispisivanja podataka poziva se mjesto u tablici, odnosno ćelija pomoću pokazivača. Svaki put kada se hoće koristiti tablica podataka, najprije se mora pozvati metoda otvaranja baze podataka. Pa tako i kada je završen rad nad bazom, potrebno je pozvati metodu za zatvaranje baze podataka. Podaci koji se ispisuju iz baze podataka posloženi su u listu podataka i zato je korišten linearni raspored. Budući da se korisnik koristi s dvije tablice podataka, napravljena su dva *layouta* - objekt i Naredba. Svaki od njih prikazuje izgled jednog retka u listi podataka te imaju istoimene aktivnosti u kojima su definirani konsturktori za dohvaćanje i postavljanje podataka. Baza podataka kreirana je prema literaturi [7].

PSEUDOKOD ZA BAZU PODATAKA:

1. Stvoriti aktivnost DbAdapter
2. Inicijalizirati stavke i tablicu/e
3. Definiranje sustava za pohranjivanje u bazu podataka
4. Definiranje metoda za stvaranje, otvaranje i zatvaranje tablice
5. Definiranje metoda za kreiranje objekta te definiranje vrijednosti, deklaracija liste, postavljanje fiksnih bilješki i deklaracija pokazivača s if - petljom
6. Stvaranje aktivnosti Objekt i Naredba te implementacija konstruktora za postavljanje i dohvaćanje
7. Povezivanje DbAdaptera s određenom aktivnosti

4. 4. Podsjetnik

U zadatku završnog rada zadano je napraviti podsjetnik koji će pomagati roditeljima da ne zaborave na različite obaveze oko djeteta kao što su cijepljenje, davanje lijekova i slično. Za izvođenje ovog dijela završnog rada potrebno je bilo kreirati aktivnost „Podsjetnik_alarm“. Na U njoj je definiran *Alarm Manager* koji poziva uslugu za alarm. slici 5.x. prikazana je ikona podsjetnika. Sadržaj zaslona sastoji se od sata, 2 gumba i 2 tektualna polja. Sat služi za odabiranje vremena u koje se alarm treba oglasiti. Sat je već gotovi dizajnirani objekt u Android studiu. Zatim potrebna su 2 gumba, jedan za postavljanje alarma a jedan za gašenje alarma. I naposlijetku, 2 tekstualna polja, jedno polje služi za upisivanje vrste obaveze dok drugo služi za

ispisivanje je li alarm postavlje i ako je, na koliko sati je postavljen. Svi ovi objekti kreirani su u .xml mapi povezanom s ovom aktivnošću. U aktivnosti su, osim inicijalizacije svih objekata na zaslonu, ispisane funkcije za postavljanje alarma. Važno je spomeniti da su vrijednosti sata, koje bi trebale biti u obliku cijelih brojeva (engl. *int(integer))* konvertirane u oblik *String* radi lakšeg manipuliranja i ispisivanja. Specifičan je i problem kod ispisivanja vrijednosti alarma u tekstualnom polju jer ako se ispisuje minuta između 1 i 9, ispis će biti u obliku npr. 10:5. Kako bi sat izgledao točnije, odnosno kako bi ispis bio 10:05, dodana je if petlja koja dodaje nulu ispred jedinične minute prilikom ispisivanja kada je jedinična minuta u pitanju. Kako bi se alarm upalio u točno vrijeme, postavljena je i jedna namjera koja odgađa alarm sve do onog vremena koje je unešeno. U suprotnom slučaju od postavljanja, definirana je i funkcija gašenja alarma povezana s gumbom gašenja koja zaustavlja pokretanje alarma u definirano vrijeme i gasi zvuk alarma ako se alarm već upalio.

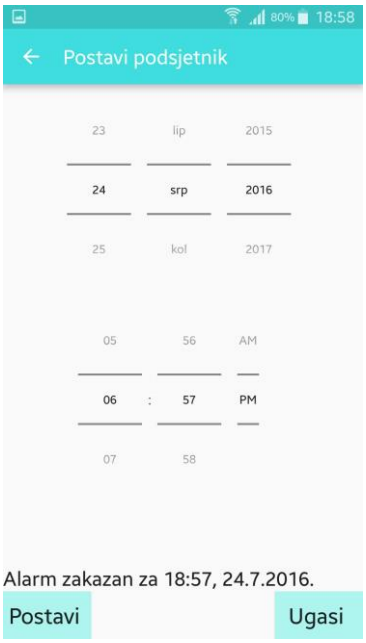


Sl. 4.12. *Ikona podsjetnika*

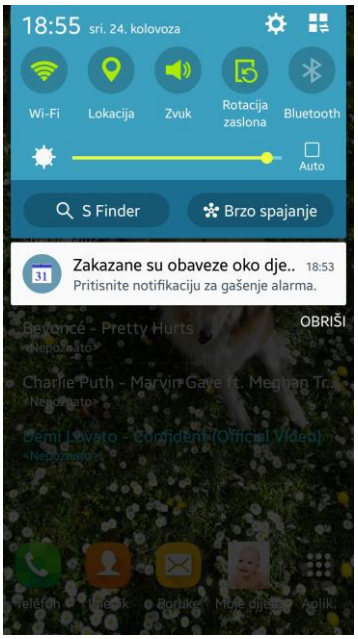
Zvuk alarma reproducira se pomoću usluge za reproduciranje. Kako bi se ona pokrenula, stvorena je aktivnost „alarm_receiver“ koja očitava je li se alarm upalio te, ako jest, ona pokreće uslugu za reproduciranje zvuka. Za reproduciranje zvuka i njegove značajke korištena je aktivnost „ringtonePlayingService“. U njoj se nalazi *if-else* petlja koja ima 4 slučaja:

a) ako melodija ne svira i alarm je upaljen – melodija počinje svirati
b) ako melodija već svira, a korisnik ugasi alarm – melodija se ugasi
c) ako melodija ne svira i korisnik ugasi alarm – ništa se ne događa
d) ako melodija svira i alarm je upaljen – ništa se ne događa.

Zadnja dva slučaja *if-else* petlje služe kako ne bi došlo do greške u aplikaciji ili izlaska iz aplikacije. Za melodiju ovog alarma odabrana je pjesma *Let it go* iz animiranog crtanog filma pod nazivom *Frozen*. Također, definirana je i sintaksa za menađera za notifikacije koji pokreće proces stvaranja notifikacije u padajućem izborniku mobilnog telefona ako se alarm upali. Osim notifikacija, definiran je i slučaj ako korisnik izađe iz aplikacije za vrijeme dok je alarm upaljen te taj izlazak iz aplikacije automatski gasi alarm. Na slici 5.13. i 5.14. prikazan je dizajn podsjetnika te notifikacija u padajućem izborniku. Za izradu alarma korištena je literatura [3].



Sl. 4.13. *Dizajn podsjetnika*



Sl. 4.14. *Prikaz notifikacije*

5. ZAKLJUČAK

Cilj ovog završnog rada bio je razviti aplikaciju pod nazivom „Moje dijete“. Ova aplikacija ima zadatak da bilježi sve važne i željene podatke o djeci koje upisuju roditelji ili staratelji djeteta (korisnici aplikacije). Korištenjem razvojnog okruženja pod nazivom „Android studio“ izrađena je navedena aplikacija bez nekih nerješivih problema. U prvim redovima glavnog dijela rada opisana je sama Android platforma, kako je došlo do ideje u Androidu, od čega se sastoji arhitektura platforme te dosad objavljene verzije evolucije iste. Nakon opisa teoretskog dijela i definiranja glavnih komponenata Android aplikacije, započet je razvoj same aplikacije. Aplikacija se razvijena pomoću klasa te istoimenih *layouta*. U klasama su definirane naredbe koje se izvršavaju pritiskom na određeni gumb te različiti konsturktori koji su potrebni za izvršavanje naredbi. Svaka klasa predstavlja jednu aktivnost, odnosno jedan izgled ekrana. U *layoutima* je definiran dizajn svake aktivnosti te je napisan kod koji opisuje svaku stavku koja se nalazi u aktivnosti kao što su tekstualna polja, tipke, okviri za označavanje, okviri za fotografiju i slično. Zadatak tipki bio je da pohranjuju podatke i vode u neku drugu aktivnost. Svi podaci koji se unose spremaju se u bazu podataka te se u nju također može pristupiti kako bi se željeni podaci pregledali. Bazu podataka također se može i uređivati prema želji korisnika. U prvim koracima izrade profila za dijete unose se osnovni podaci (spol djeteta, ime, prezime, fotografija). Mana kod pohranjivanja ovih podataka je da se oni više ne mogu mijenjati pa tako korisnik više nikad ne može promjeniti sliku djeteta. U naknadnom razvijanju ove aplikacije biti će i taj dio omogućen. Nakon spremanja ovih podataka, na početnom zaslonu stvara se okvir sa slikom, imenom i prezimenom djeteta. Pritiskom na neki od ovih okvira otvara se aktivnost sa detaljnijim podacima. Za svako dijete mogu se spremati različiti podaci kao što su boja očiju, okus prve kašice, koja je prva izgovorena riječ itd. Svi ovi podaci već su ponuđeni prilikom instaliranja aplikacije. U budućnosti, podaci svakog djeteta mogu se i nadopunjavati, npr. korisnik u dogledno vrijeme može stvoriti novi podatak na listi poput datuma kad je dijete krenulo u školu. Isti podatak može uređivati i pregledavati te se taj podataka ponaša kao i svi ostali s liste. Isto tako, kada bi se ova aplikacija razvijala još više, može se omogućiti korisniku da sprema više slika, a ne samo početnu (profilnu) sliku djeteta, recimo, spremanje slike ultrazvuka, spremanje prvih slika iz rodilišta itd. Te slike dobivene su pozivanjem ugrađene kamere u mobilnom telefonu te su one pohranjene u galeriji slika. Jedna od pogodnosti ove aplikacije je da je moguće postaviti podsjetnik koji će podsjetiti korisnika na neke obaveze, kao što je davanje lijeka djetetu. Korisnik može podesiti vrijeme ili interval javljanja podsjetnika.

Svrha ove aplikacije je da se korisniku olaška pamćenje važnih podataka koje njegovo dijete prolazi prvi puta u životu. Na ovaj način korisnik ima dugotrajnu pohranu podataka na svom mobilnom telefonu. Moguć je unos više osoba za koje se spremaju podaci. Aplikacija je prošla testna pokretanja te se na uzorku od desetero ljudi pokazala zadovoljavajuće dobrom zamjenom za razne dnevnike koji nisu u elektronskom obliku i koje korisnici najčešće negdje zametnu. U budućnosti, ova aplikacija će se dodatno razvijati kako bi se njena kvaliteta poboljšala.

LITERATURA

[1] -, -, Mobilni uređaji - Android operativni sustav , (http://www.informatika.buzdo.com/pojmovi/mobile-3.htm), pristup ostvaren 01.06.2016.

[2] V., Deželić, Lollipop s 36,1 posto najzastupljenija verzija Androida, Marshmallow na 2,3 posto, (http://www.ictbusiness.info/vijesti/lollipop-s-36-1-posto-najzastupljenija-verzija-androida-marshmallow-na-2-3-posto), pristup ostvaren 26.07.2016.

[3] , -, Android alarm clock tutorial (https://www.youtube.com/watch?v=xbBlzOblD10&list=PL4uut9QecF3DLAacEoTctzeqTyvgzqYwA), pristup ostvaren 13.8.2016.

[4] FER, Razvoj usluga i aplikacija za operacijski sustav Android, (https://www.fer.unizg.hr/_download/repository/RUAZOSA-2016_05-gui.pdf), pristup ostvaren 27.06.2016.

[5] Wikipedia, Android (operacijski sustav) , (https://hr.wikipedia.org/wiki/Android_(operacijski_sustav)), pristup ostvaren 14.09.2016

[6] I.,Validzic, Završni rad, (https://www.fer.unizg.hr/_download/repository/Zavrsni_rad_-_Ivan_Validzic.pdf), pristup ostvaren 14.09.2016

[7] -,-, Android App Development Tutorial, (https://www.youtube.com/watch?v=lApbZO7J7ZI&list=PLHs_NFdr_LaHmEh7hV-wPyS-gKnAVPzBU), pristup ostvaren 14.09.2016.

[8] Wikipedia, Dosadašnje inačice sustava Androida, (https://hr.wikipedia.org/wiki/Dosada%C5%A1nje_ina%C4%8Dice_sustava_Androida), pristup ostvaren 26.07.2016.

[9] -, -, Programiranje Android aplikacija, (http://webcache.googleusercontent.com/search?q=cache:http://www.android-aplikacije.rs/programiranje-android-aplikacija&gws_rd=cr&ei=go9yV9fmKIeUaLvJmmA), pristup ostvaren 01.06.2016.

SAŽETAK

U ovom završnom radu opisana je Android platforma i zadatak je bio upoznati se s karakteristikama njene arhitekture. Zatim je izrađena aplikacija „Moje dijete“ u programskom jeziku Java koja služi kao zamjena za dnevnike koji sadrže podatke o rastu i razvoju djece tzv. radosnice. U aplikaciju se mogu unositi podaci o više osoba. Aplikacija ima mogućnost popunjavanja i izmjene postojećih podataka te dodavanje novih podataka. Korisnik može zabilježiti i fotografiju svoga djeteta. U programskom kodu definirane su instrukcije za svaku tipku, tekstualno polje, okvir za označavanje i okvir za fotografiju. Spomenute su i raznolike funkcije za povezivanje aktivnosti koje predstavljaju pregled svakog ekrana. Svi podaci spremaju se u bazu podataka koja je stvorena također u programskom kodu. Također, definiran je i podsjetnik koji ima zadatak da podsjeti korisnika o važnim događajima vezanim za dijete. Aplikacija je uspješno razvijena te je namjenjena za dugoročno korištenje.

Ključne riječi: Android platforma, Moje dijete, podaci, instrukcije, funkcije, baza podataka, podsjetnik

ABSTRACT

This final paper describes Android platform and the assigment which was to find out what are the characteritic of Androids architecture. In addition, the application called My child has been developed in Java programming language. The application is used as a replacement for journals which are gathering data about childs growth and evolution. You can store data for more persons. The application has ability of filling out and chaning already existing data and adding new data. User can also enter photo of baby. In program code are written instructions for each button, text box, check box and image box. Also, there are mentioned functions for connecting activities which are representing every display view. All data is stored into database which was created and defined in code. There is also a reminder of important events in baby's life. The application is successfully developed and it is made for long term using.

Key words: Android platform, My child, data, instructions, functions, database, reminder

ŽIVOTOPIS

Gabriela Angebrandt rođena je 20. ožujka 1995. godine u Đakovu. Pohađala je osnovnu školu Ivana Gorana Kovačića u Đakovu. Nako završenog osnovnoškolskog školovanja, upisala je gimnaziju „A.G.Matoš“ u Đakovu gdje je maturirala 2013. Iste godine upisala je Fakultet elektrotehnike, računarstva i informacijskih tehnologija u Osijeku na kojem se školuje i danas. U 2015. godini bila je potpredsjednica *Computer Society-a* IEEE studentskog ogranka FERIT-a. Gabriela poznaje programske alate Microsoft Office-a te programske jezike C, C++ i Java. Gabriela ima iskustva u korištenju programa Visual studia, MATLAB-a, Photoshop-a, AutoCad-a i SQL-a.

Vlastoručni potpis:

_____

Angebrandt Gabriela