Android aplikacija za provođenje sportskih treninga

Čičak, Ivan

Master's thesis / Diplomski rad

2017

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:200:888662

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-07-27**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

ELEKTROTEHNIČKI FAKULTET

Sveučilišni studij

Android aplikacija za provođenje sportskih treninga

Diplomski rad

Ivan Čičak

Osijek, 2017.

Sadržaj

1. UVOD 1
1.1. Zadatak diplomskog rada 1
2. KORIŠTENE TEHNOLOGIJE 2
2.1. Android 2
2.1.1. Android slojevi 2
2.1.2. Verzije Androida 3
2.2. Android Studio 4
3. ANDROID APLIKACIJA 5
3.1. Razvoj aplikacije 5
3.2. Glavno sučelje 6
3.2.1. Kartica "Training" 9
3.3. Prikaz vježbi za odabrani dan 11
3.4. Aktivnost za izvršavanje vježbe 12
4. ZAKLJUČAK 16
LITERATURA 17
SAŽETAK 18
ABSTRACT 19
ŽIVOTOPIS 20

1. UVOD

Cilj ovog rada je napraviti Android aplikaciju za provođenje sportskih treninga.Realizacija samog zadatka je napravljena na Android platformi -operativnom sustavu koji se zasniva na Linux jezgri. Android je najkorišteniji operacijski sustav za prijenosne uređaje, otvorenog je koda i ima veliku podršku programerske zajednice što često znači brz i kvalitetan razvoj aplikacijskih rješenja.Razvoj aplikacije je realiziran u programskom alatu Android Studio, službenom IDE (engl.*Integrated Development Environment*) za Android, dok je aplikacija pisana u Java programskom jeziku.

Rad je podijeljen u više poglavlja. Prvo poglavljesadrži uvod i kratak opis zadatka ovog rada. U drugom poglavlju su opisane korištene tehnologije za izradu rada. U trećem je poglavlju opisan programski dio – razvoj i rad aplikacije, nakon čega slijedi zaključak.

1.1. Zadatak diplomskog rada

Zadatak ovog rada je napraviti Android aplikaciju za provođenje sportskih treninga. Naglasak je stavljen na izradu aplikacije koja u cilju ima što manju interakciju korisnika s mobilnim uređajem za vrijeme treninga.

2. KORIŠTENE TEHNOLOGIJE

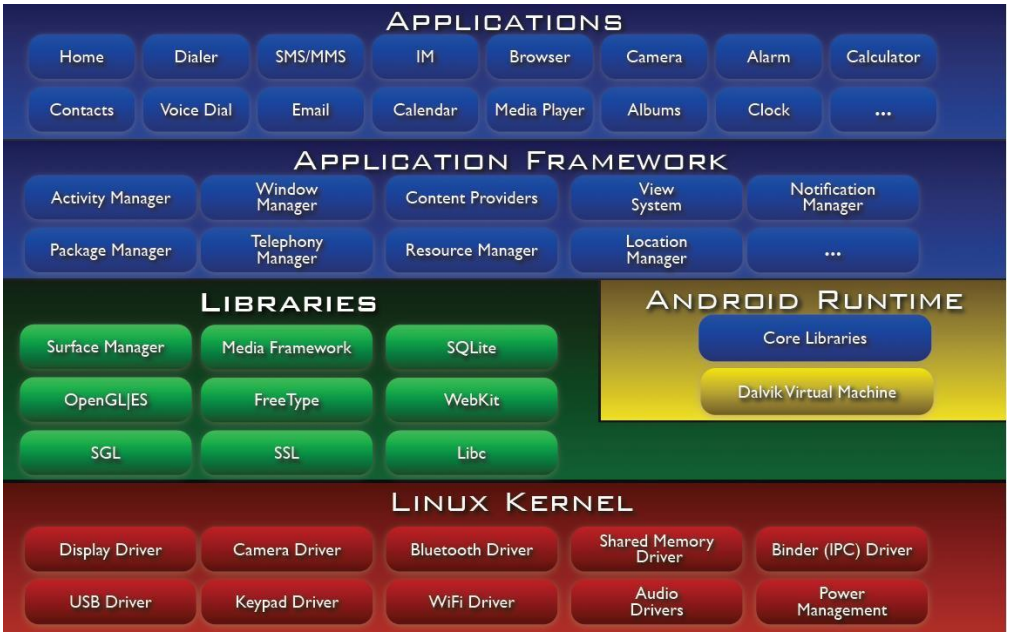
2.1. Android

Android je operacijski sustav koji se zasniva na Linux jezgri,arazvijen je od strane Google-a.[1]. Primarno je dizajniran za prijenosne uređaje. U praksi se može naći na velikom broju mobilnih telefona, tableta, čitača elektronskih knjiga i drugih prijenosnih uređaja.

2.1.1. Android slojevi

Android se sastoji od 4 osnovna sloja:

1. Linux jezgra (engl. *Linuxkernel*)
2. Biblioteke i Android podrška u realnom vremenu (engl. *Androidruntime*)
3. Radno okruženje (engl. *Applicationframework*)
4. Aplikacijski sloj



Slika 2.1. Arhitektura android platforme

Izvor: http://www.techdhamaka.com/2016/05/theming-android-part-5.html (10.8.2017.)

Linux jezgra služi za komunikaciju s fizičkom arhitekturom uređaja (engl. *hardware*). Sadrži upravljačke programe (engl. *driver*) od kojih su najvažniji programi za upravljanje memorijom i

procesima, programi za međuprocesnu komunikaciju i programi za upravljanje napajanjem (engl. *power managment*). [2]

Sav kod koji osigurava glavne funkcionalnosti Android operacijskom sustavu se nalazi u bibliotekama. Neke od biblioteka koje Android pruža su SQLite, WebKit, FreeTypem, OpenGL ES, OpenSL ES, SSL, libc, itd.

Android podrška u realnom vremenu (engl. *Android runtime*) sadrži osnovne biblioteke (engl. *Core libraries*) koje omogućuju pisanje Android aplikacija upotrebom Java programa. Također, sadrži DVK(engl.*Dalvik virtual machine*), komponentu koja je napravljena za Android i uređaje sa limitiranim resursima, a koja pokreće aplikacije kao zasebne procese i optimizira ih uz minimalni utrošak memorije. [3]

Radno okruženje (engl. *application framework*) je sloj koji pruža aplikacijama različite usluge (engl. *services*) za bolji rad.Također je napisan u programskom jeziku Java. Neke od funkcija su:

- upravljanje aktivnostima (engl. *Activity Manager*) - upravlja životnim ciklusom aplikacije
- upravljanje prozorima (engl. *Window Manager*) - služi za upravljanje aplikacijskim prozorima
- upravljanje resursima (engl. *Resource Manager*) - služi za upravljanje raspoloživim resursima: memorijom, prostorom na disku i sl.
- upravljanje pozivima (engl. *Telephone Manager*)- sadrži API-je za izradu aplikacija koje upravljaju pozivima

Aplikacijski sloj je zadnji sloj u arhitekturi Androida. Sastoji se od aplikacija samog uređaja te je vidljiv krajnjem korisniku.Primjeri aplikacija su: internet preglednik, preglednik elektronske pošte, kalendar, sat i dr.

2.1.2.Verzije Androida

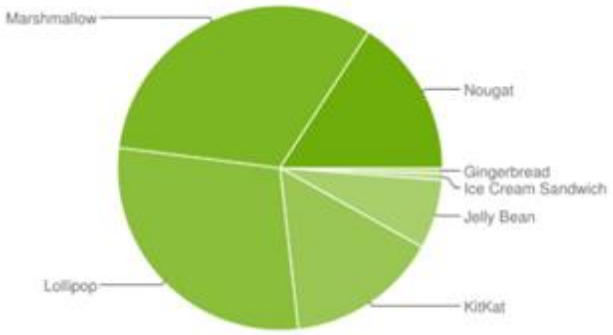
Prva verzija Androida je izdana 2007. godine te je od tada izašlo mnogo nadogradnji Android operacijskog sustava. 2009.[4] godine je izašla 1.5 verzija i to je prva verzija koja je dobila svoje kodno ime, *Cupcake*. Od tada se svakoj nadogradnji Android sustava daje novo kodno ime. Do sada su sve verzije imale ime prema nekoj slastici i imena su davana po abecednom redu.

Slika 2.2.prikazuje udio uređaja sa instaliranimrazličitim verzijama Androida u rujnu 2017. godine. Tablica prikazuje verzije Androida, kodna imena, API i zastupljenost verzija na tržištu.

Na grafu se može vidjeti da je verzija *Nougat* (7.0 i 7.1) manje zastupljena od verzije *Marshmallow*(6.0). To će se s vremenom promijeniti pošto sve veći broj korisnika prelazi na novije verzije uređaja i novije verzije Androida. [4]

Najnovija verzija Androida, *Oreo* (8.0), koja je izašla na tržište u kolovozu 2017. godine[5][6], zasad je dostupna na malom broju uređaja te malom broju korisnika. Taj broj će se postupno povećavati.

Verzija	Naziv	API	Zastupljenost
2.3.3 - 2.3.7	Gingerbread	10	0.6%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	0.6%
4.1.x	Jelly Bean	16	2.4%
4.2.x		17	3.5%
4.3		18	1.0%
4.4	KitKat	19	15.1%
5.0	Lollipop	21	7.1%
5.1		22	21.7%
6.0	Marshmallow	23	32.2%
7.0	Nougat	24	14.2%
7.1		25	1.6%



Slika 2.2. Zastupljenost verzija Androida

Izvor: https://developer.android.com/about/dashboards/index.html (15.9.2017.)

2.2. Android Studio

Android Studio je službenorazvojno okružje (engl. *IDE – Integrated Development Enviroment*) za pisanje aplikacija na Android platformi. Izgrađen je na IntelliJ platformi koja je otvorenog koda. [7]

Prije rada u Android Studio-u potrebno je instalirati alat za razvoj aplikacija SDK(engl. *Software development kit*) koji omogućuje razvoj android aplikacija pomoću Jave.

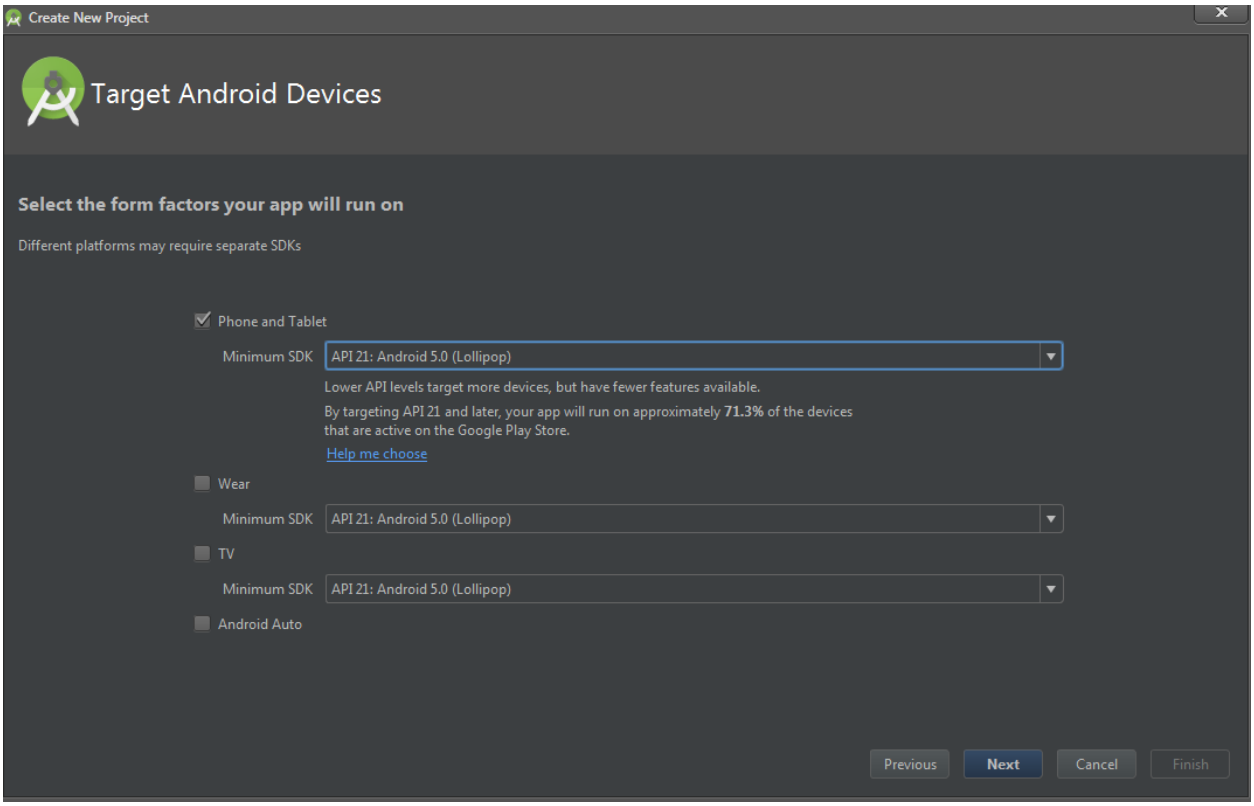
U ovom radu je korištena Android Studio verziju 2.3. i jdk veziju 1.8_91.

3. ANDROID APLIKACIJA

Aplikacija za ovaj rad je u potpunosti izrađena u razvojnom okruženju Android Studio. Napravljena je s ciljem lakšeg provođenja sportskih treninga. Pokretanjem aplikacije korisnik bira trening i dobiva informacije o vježbama koje će raditi taj dan. Sljedećim klikom se pokreće vježba. Veliki naglasak je stavljan na to da korisnik ne treba imati interakcije s mobilnim uređajem za vrijeme vježbanja - na ekranu je vidljiva animirana slika vježbe (GIF), vrijeme ili broj ponavljanja koje treba napraviti te se zadnje tri sekunde prije završetka vježbe čuje zvučni signal.

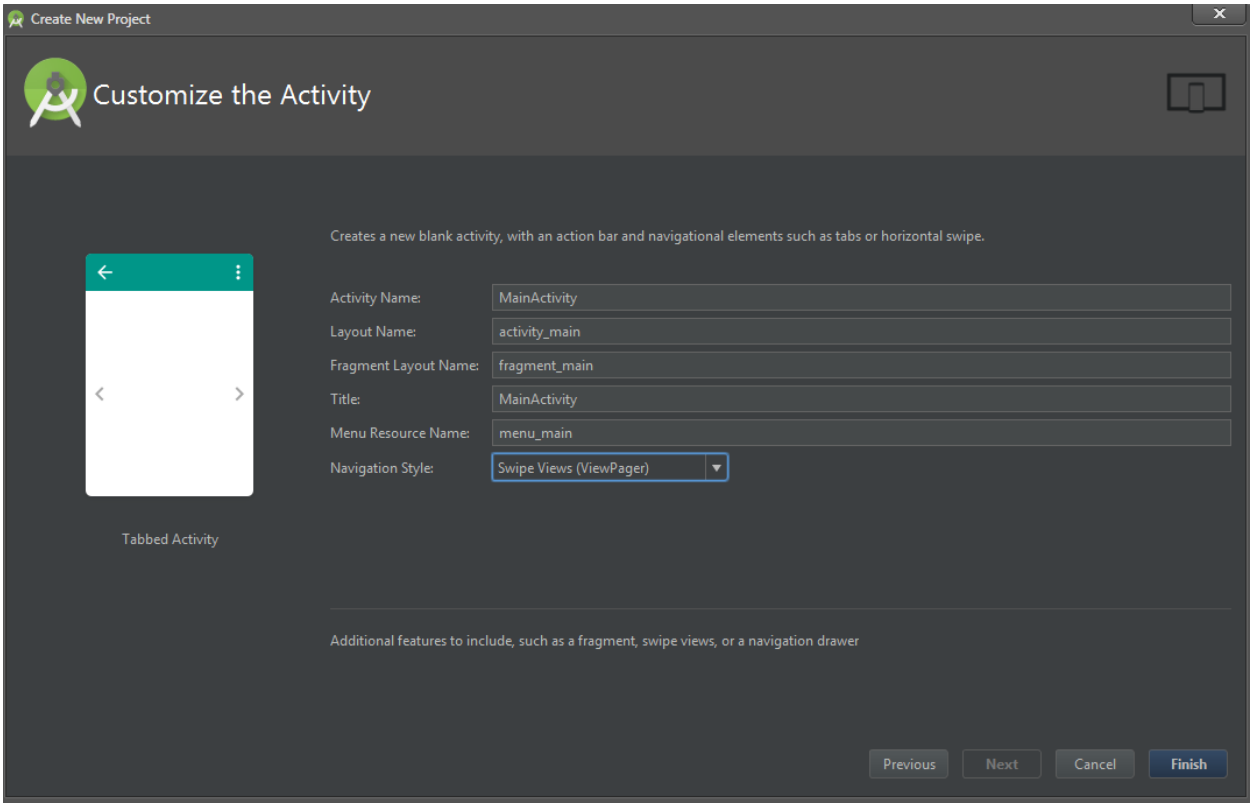
3.1. Razvoj aplikacije

Nakon pokretanja Android Studia i biranja imena projekta, dolazi se do ekrana na kojem se nalazi izbor minimalne verzije Androida na kojem će aplikacija raditi. Izabrana je android 5.0 (*Lollipop)*. Za vrijeme pisanja ovog rada, oko 80% uređaja ima verziju 5.0 ili noviju i može pokrenuti ovu aplikaciju.



Slika 3.1. Android 5.0 je najniža verzija na kojem će raditi aplikacija

Na sljedećem ekranu (Sl. 3.2) nudi se opcija odabira*Activity-ja*. Napravljen je izbor *Tabbed Activity-ja* koji olakšava pravljenje kartica (engl. *tabs*) koje su korištene u aplikaciji. Također je omogućeno kretanje kroz tabove sa povlačenjem prsta (engl. *swipe*).



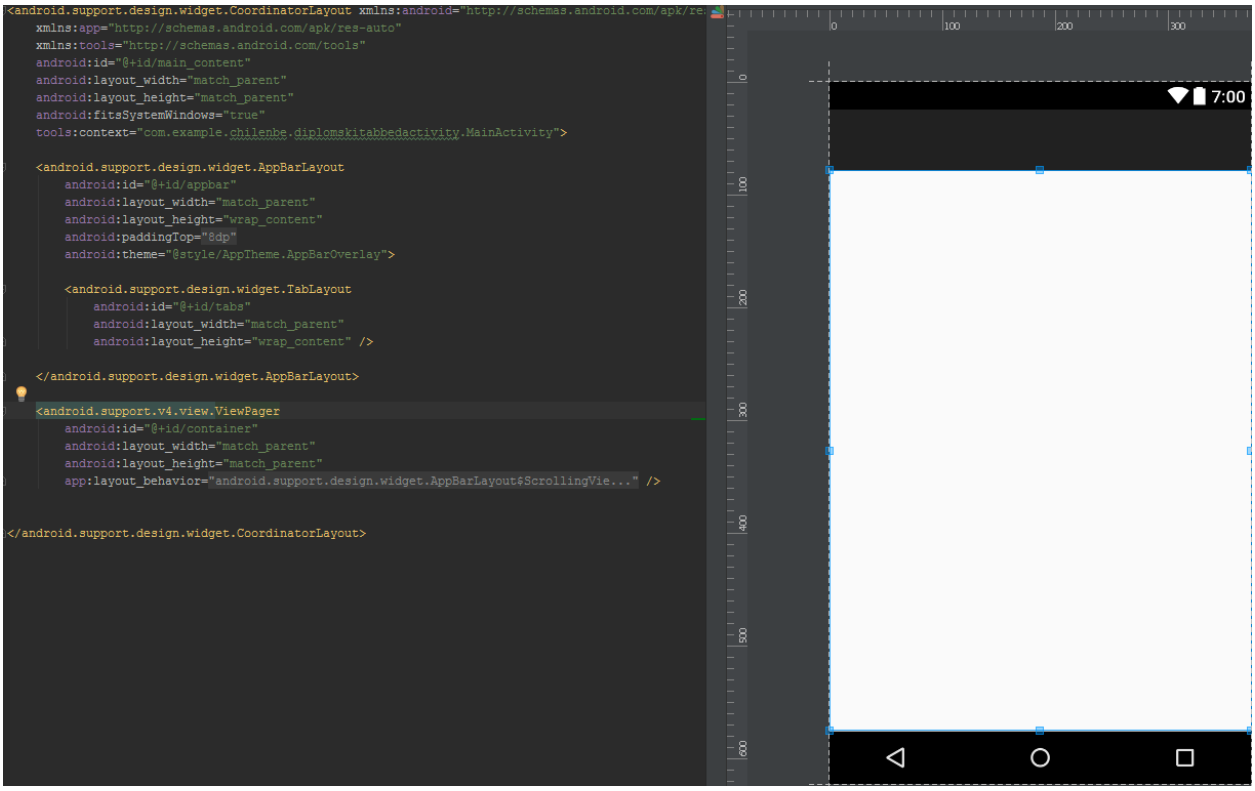
Slika 3.2. Izbor *Activity-ja*

Pritiskom na tipku završetak (engl. *Finish*) završava kreiranje projekta.

3.2. Glavno sučelje

MainActivity je prva aktivnost nakon kreiranja projekta. Kroz čarobnjak je podešeno da MainActivity bude aktivnost (engl. *activity*) koja sadrži kartice (engl. *tabs*) i ima mogućnost promjene kartica sa pokretom povlačenja prsta (engl. *swipe*).

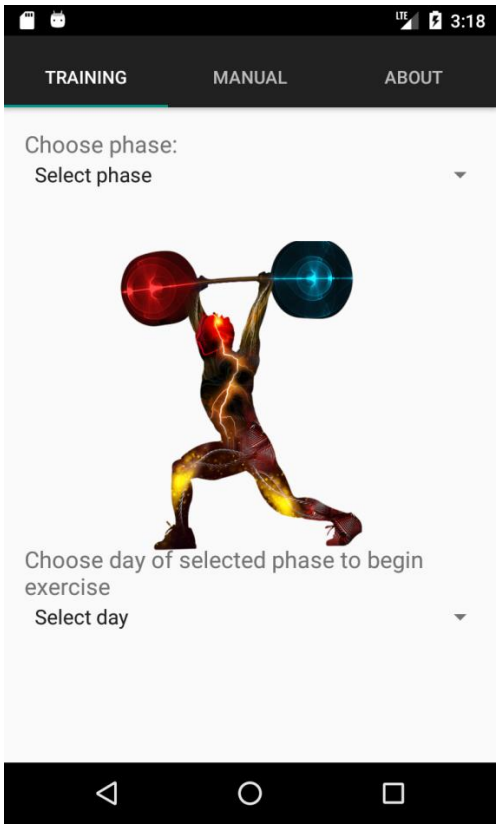
Na slici 3.3. je prikazan XML kod i dizajn sučelja. Bitno je primijetiti ViewPager na XML dijelu slike koji označava bijeli prostor omeđen plavim točkama na dizajn dijelu slike. To je prostor koji će se popunjavati fragmentima ovisno o tome koju karticu odaberemo. Na slici 3.4. vidimo kako je taj prostor popunjen sadržajem prve kartice.



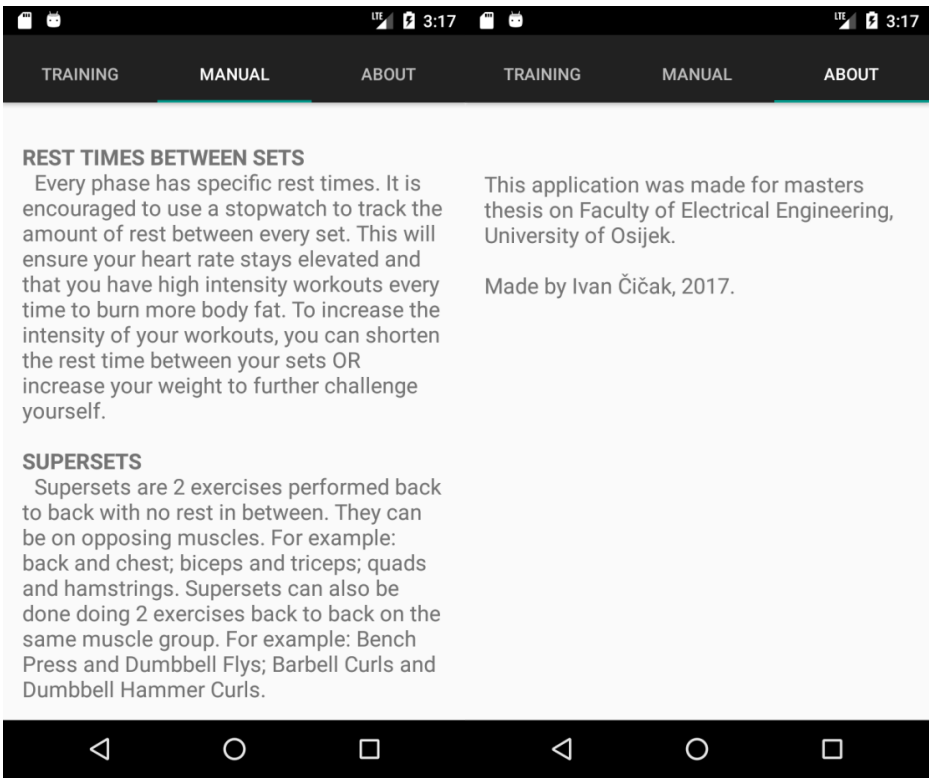
Slika 3.3. XML kod i dizajn slike

U MainActivity-ju je podešeno da će prikazivati tri kartice i da su njihovi nazivi:

- "Traning" - kartica na kojoj korisnik može izabrati na kojoj je fazi treninga, kao i u kojem danu. To se može napraviti kroz dostupne padajuće izbornike. Slika 3.4 predstavlja "Traning" karticu. Nakon što odabere fazu, korisnik na ekranu može vidjeti pojedinosti vezane za odabranu fazu treninga (Sl. 3.9.).
- "Manual" - kartica na kojoj se korisnik može informirati o izrazima koje će sretati za vrijeme korištenja aplikacije (Sl. 3.5.).
- "About" - kartica na kojoj se mogu naći informacije o aplikaciji (Sl. 3.5.).



Slika 3.4. Kartica "Traning"



Slika 3.5. Kartice "Manual" i "About"

3.2.1. Kartica "Training"

U fragmentu pomoću kojega je popunjena kartica "Training" korištena su dva padajuća izbornika (engl. *Spinner*). Za implementaciju padajućih izbornika potrebno prvo dodati XML kod (Sl. 3.6.), a zatim definirati ponašanja preko .java datoteke (Sl. 3.7.).

```
<Spinner
    android:id="@+id/spinner"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:entries="@array/Phases"
    android:layout_below="@+id/textView4"
    android:layout_alignParentStart="true" />
```

Slika 3.6. Xml kod za padajući izbornik

```
// Create an ArrayAdapter using the string array and a default selectPhaseSpinner layout
        ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(getActivity(),
                R.array.Phases, android.R.layout.simple_spinner_item);
// Specify the layout to use when the list of choices appears
        adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
// Apply the adapter to the selectPhaseSpinner
        selectPhaseSpinner.setAdapter(adapter);
        selectPhaseSpinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
            @Override
            public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {...}
            @Override
            public void onNothingSelected(AdapterView<?> parent) {
            }
        });
```

Slika 3.7. Definiranje padajućeg izbornika u .java datoteci

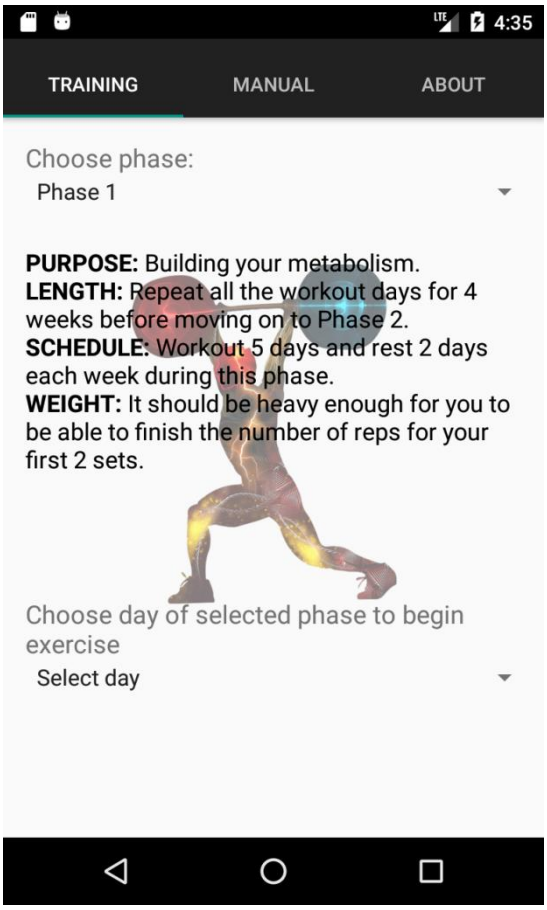
Iz slike 3.7 je vidljivo da je padajući izbornik popunjen vrijednostima strukture (engl. *array*) "Phases". Metoda "setOnItemSelectedListener" omogućava da se definira ponašanje padajućeg izbornika ako se odabere neki od ponuđenih izbora.

Na slici 3.8. vidimo naredbe koje se nalaze u "onItemSelected" (sl 3.7) ako se odabere prva opcija (faza 1). U tom slučaju se popunjavaju opcije drugog padajućeg izbornika sa strukturom "Days_phase1".

```
ArrayAdapter<CharSequence> adapter2 = ArrayAdapter.createFromResource(getActivity(),
        R.array.Days_phase1, android.R.layout.simple_spinner_item);
adapter2.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
selectDaySpinner.setAdapter(adapter2);
imageView.setAlpha(0.5f);
```

Slika 3.8. Naredbe koje se izvrše ako se odabere prva opcija u padajućem izborniku

Zadnja naredba na slici 3.8 služi tome da se poveća prozirnost pozadinske slike za 50% s ciljem da bi se više isticao tekst koji se pojavi iznad slike. Usporedbom slika 3.4 i 3.9. je vidljiva razlika u prozirnosti.



Slika 3.9. Ekran nakon odabira prve opcije u prvom padajućem izborniku

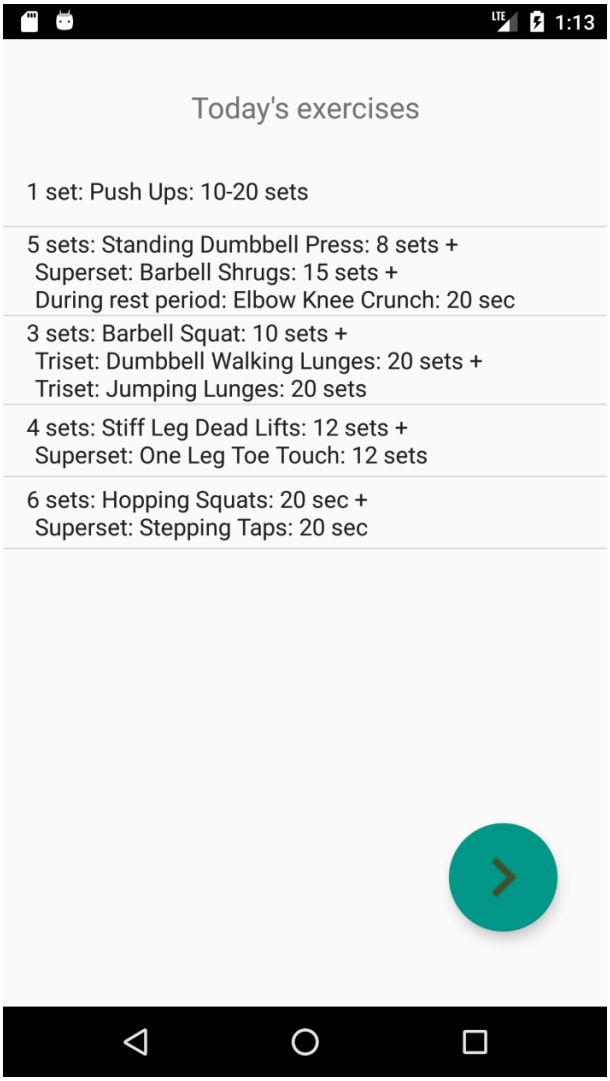
Na slici 3.10. su naredbe koje se pokrenu klikom na drugu opciju u padajućem izborniku. Sprema se informacija o tome koji trening je korisnik odabrao koja je bitna za buduće korištenje programa. Također, nakon što korisnik odabere opciju u drugom padajućem izborniku, pokreće se namjera (engl. *intent*) koja pokreće novu aktivnost - "TrainingList" o kojoj je napisano poglavlje 3.3.

```
DoingExerciseActivity.day = "day1x2";
startActivity(new Intent(getActivity(), TrainingList.class));
break;
```

Slika 3.10. Naredbe koje se pokreću nakon odabira opcija u drugom izborniku kartice "Training"

3.3. Prikaz vježbi za odabrani dan

Nakon što je korisnik odabrao opcije na oba padajuća izbornika u kartici "Training", otvara se nova aktivnost, "TrainingList" pomoću koje su prikazane vježbe za odabrani dan (Sl. 3.11.).



Slika 3.11. Prikaz vježbi za odabrani dan

Prikaz vježbi je realiziran pomoću pregledne liste (engl. *ListView*). Na slici 3.12. je prikazana implementacija *ListView-a* u .java datoteci.. Pomoću *switch* naredbe i na temelju vrijednosti koje je korisnik odabrao u padajućim izbornicima "Training" kartice, "listViewArray" struktura se puni različitim vrijednostima, koje *ListView* prikazuje.

```
// Define a new Adapter
// First parameter - Context
// Second parameter - Layout for the row
// Third parameter - ID of the TextView to which the data is written
// Forth - the Array of data
ArrayAdapter<String> adapter = new ArrayAdapter<>(this,
        android.R.layout.simple_list_item_1, android.R.id.text1, listViewArray);

listView.setAdapter(adapter); // Assign adapter to ListView
```

3.12. *ListView* implementacija u .java datoteci

Nakon informiranja o vježbama koje su u planu vježbanja taj dan, korisnik treba stisnuti tipku (engl. *Button*) na dnu ekrana. Pritiskom na tipku se pokreće namjera (engl. *Intent*) koji pokreće "DoingExerciseActivity" aktivnost (Sl. 3.13.) koja je opisana u poglavlju 3.4.

```
FloatingActionButton fab = (FloatingActionButton)findViewById(R.id.floatingActionButton);
fab.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        startActivity(new Intent(TrainingList.this, DoingExerciseActivity.class));
    }
});
```

Slika 3.13. Naredbe koje se pokreću pritiskom na tipku u "TrainingList" aktivnosti

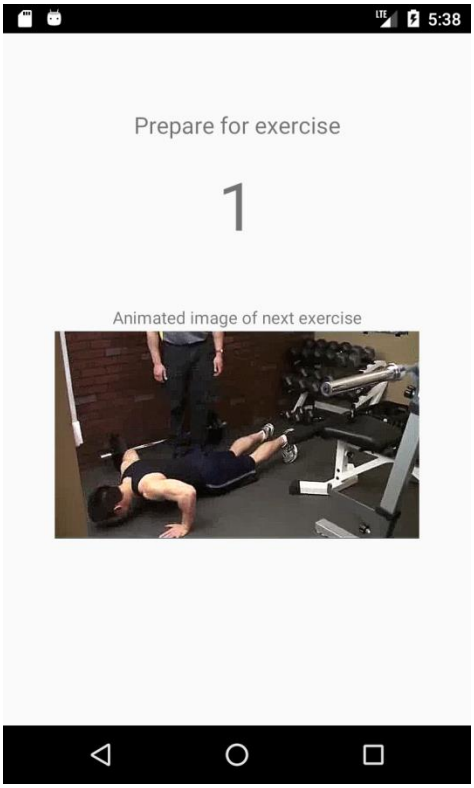
3.4. Aktivnost za izvršavanje vježbe

Tijekom aktivnosti "DoingExerciseActivity" korisnik bi trebao izvršavati vježbe. Aktivnost je napravljena s ciljem da korisnik ima što manju interakciju s ekranom za vrijeme vježbe. Predviđeno je da se mobilni uređaj ostavi na vidljivo mjesto i da se prati instrukcije prikazane na ekranu.

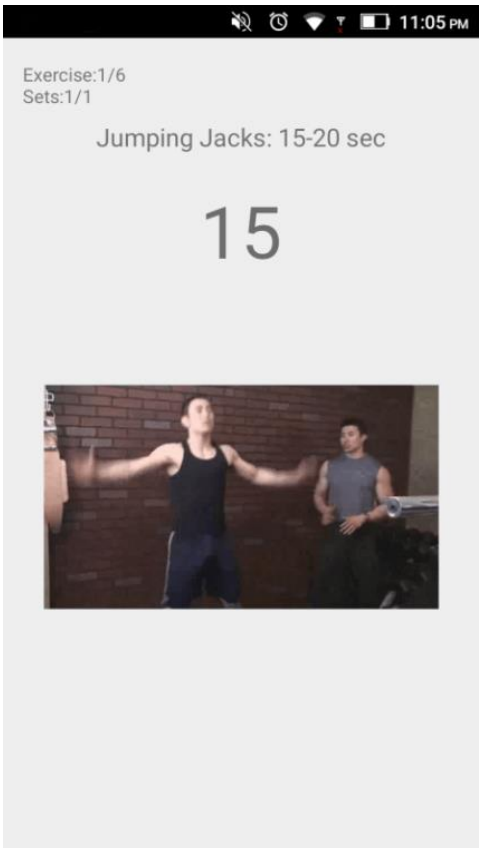
To se realiziralo tako što:

- korisnik ima 5 sekundi prije prve vježbe da se pripremi za vježbu, za vrijeme kojih mu je prikazana sljedeća vježba (Sl. 3.14.)
- korisnik u svakom trenutku ima napisano ime vježbe i vidi animiranu sliku (engl. *gifs - Graphics Interchange Format*) vježbe koju bi trebao raditi
- korisniku je na ekranu prikazan brojač vremena koji prikazuje koliko dugo još treba raditi vježbu (Sl. 3.15.)
- za vrijeme pauze, na ekranu se prikazuje *gif* sljedeće vježbe i odbrojava preostalo vrijeme pauze
- korisnik čuje zvučnu obavijest 3 sekunde pred kraj vježbe ili pauze
- u svakom trenutku korisnik u lijevom gore kutu vidi na kojoj je vježbi, na kojem je setu te vježbe i koliko ukupno ima vježbi taj dan (Sl. 3.15.)

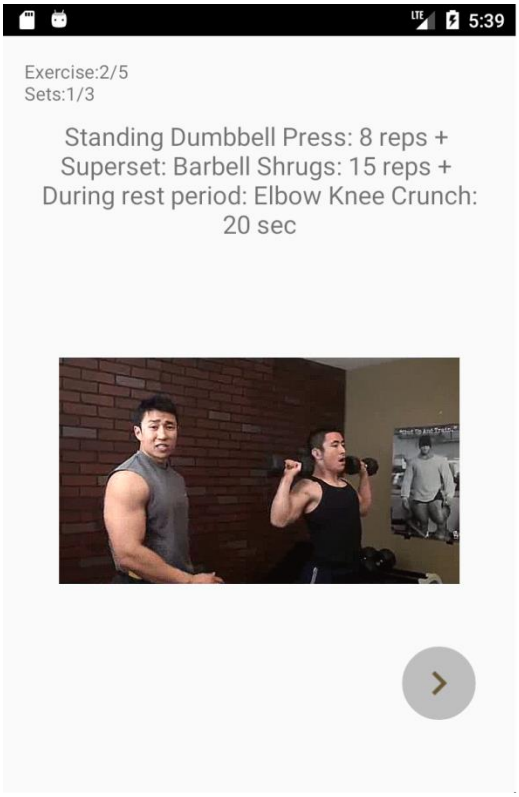
Jedino vrijeme kada korisnik mora dirati ekran za vrijeme "DoingExerciseActivity" aktivnosti se za vježbu treba napraviti određeni broj ponavljanja. U tom slučaju se pojavi tipka koju korisnik treba pritisnuti kad završi ponavljanja vježbe da bi nastavio trening.



Slika 3.14. Priprema za prvu vježbu



Slika 3.15. Vježba kod koje se treba raditi određeni vremenski period



Slika 3.16. Vježba kod koje treba odraditi određeni broj ponavljanja

Kod vježbi koje se trebaju raditi zadano vrijeme na ekranu je prikazano preostalo vrijeme koje se treba raditi vježba. Na slici 3.17. je prikazano kako je implementirano odbrojavanje vremena pomoću*CountDownTimer*- a.

```
cdt = new CountDownTimer((timers[counter] + 1) * 1000, 1000) {
    @Override
    public void onTick(long millisUntilFinished) {
        textViewTimer.setText("" + millisUntilFinished / 1000);
        if (millisUntilFinished / 1000 == 3) {
            mp.start();
        }
    }
    @Override
    public void onFinish() {...}
};
cdt.start();
if (counter < timers.length) {
    counter++;
}
```

Slika 3.17. *CountDownTimer*

Na zadnje tri sekunde odbrojavanja vremena pali se zvučna signalizacija. To je realizirano pomoću *MediaPlayer* klase. Implementacija je vidljiva na slici 3.17., u *onTick* metodi.

4. ZAKLJUČAK

Cilj ovog rada je izraditi Android aplikaciju za provođenje sportskih treninga. Na početku rada su opisane tehnologije koje se koriste pri programskoj realizaciji. U glavnom dijelu rada je objašnjeno kako je aplikacija napravljena.

Pokretanjem aplikacije korisnik bira trening i dobiva informacije o vježbama koje će raditi taj dan. Sljedećim klikom se pokreće vježba. Veliki naglasak je stavljan na to da korisnik ne treba imati interakcije s mobilnim uređajem za vrijeme vježbanja - na ekranu je vidljiva animirana slika vježbe (GIF), vrijeme ili broj ponavljanja koje treba napraviti te se zadnje tri sekunde prije završetka vježbe čuje zvučni signal.

LITERATURA

[1] Android operacijski sustav, internet stranica, rujan 2017.

https://en.wikipedia.org/wiki/Android_(operating_system)

[2] Android arhitektura, internet stranica, rujan 2017.

https://developer.android.com/guide/platform/index.html#linux-kernel

[3] Android arhitektura, internet stranica, rujan 2017.

https://www.edureka.co/blog/beginners-guide-android-architecture/

[4] Povijest Androida, internet stranica, rujan 2017.

https://www.android.com/history/

[5] Zatupljenost Android verzija na tržištu, internet stranica, rujan 2017.

https://developer.android.com/about/dashboards/index.html

[6] Android Oreo, internet stranica, rujan 2017.

http://www.express.co.uk/life-style/science-technology/844284/Android-8-0-Oreo-Release-Date-Cant-Download-Install

[7] Android Studio, internet stranica, rujan 2017.

https://developer.android.com/studio/index.html

SAŽETAK

Cilj ovog rada je izraditi aplikaciju za provođenje sportskih treninga. Korištene tehnologije su Android, Java, XML i razvojno okruženje Android Studio. Aplikacija se može koristit na svim operacijskim sustavima Android, verzijama 5.0 *Lollipop*ili novijim. Naglasak kod izrade aplikacije je stavljen na to da se aplikacija, nakon što se pokrene vježba, mora što manje dirati. tj. da korisnik može pratiti trening bez interakcije sa mobilnim uređajem i da se može fokusirati na vježbanje.

Ključne riječi: Android studio, Java, XML, Android, Lollipop

ABSTRACT

Title: Android application for conducting sport trainings

The aim of this work was to make Android application for conducting sport exercises. Used technologies were Android, Java, XML and integrated development environment Android Studio. This application can be used on all Android operating systems that have installed version 5.0 Lollipop or newer version. The emphasis in making the application was making required interaction with mobile phone while exercising as minimal as possible so the customer can focus on exercising.

Key words: Android studio, Java, XML, Android, Lollipop

ŽIVOTOPIS

Ivan Čičak je rođen 16.11.1988. u Osijeku. Osnovnu školu „Antuna Mihanovića“ završava 2003. godine u Osijeku nakon čega upisuje Isusovačku klasičnu gimnaziju s pravom javnosti u Osijeku. Preddiplomski studij računarstva na Elektro-tehničkom fakultetu u Osijeku upisuje 2007. koji završava 2013. te odmah nakon toga upisuje diplomski studij procesnog računarstva na istom fakultetu.