Android aplikacija za konverziju i praćenje valuta i kriptovaluta

Džoić, Mario

Undergraduate thesis / Završni rad

2018

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:200:285861

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-05-13**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURAJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA

Stručni studij

Android aplikacija za konverziju i praćenje valuta i kriptovaluta

Završni rad

Mario Džoić

Osijek, 2018.

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac Z1S: Obrazac za imenovanje Povjerenstva za obranu završnog rada na preddiplomskom stručnom studiju

Osijek, 25.09.2018.

Odboru za završne i diplomske ispite

Imenovanje Povjerenstva za obranu završnog rada na preddiplomskom stručnom studiju

Ime i prezime studenta:	Mario Džoić
Studij, smjer:	Preddiplomski stručni studij Elektrotehnika, smjer Informatika
Mat. br. studenta, godina upisa:	AI4505, 20.09.2017.
OIB studenta:	16177109834
Mentor:	Izv. prof. dr. sc. Krešimir Nenadić
Sumentor:	
Sumentor iz tvrtke:	
Predsjednik Povjerenstva:	Doc.dr.sc. Ivica Lukić
Član Povjerenstva:	Doc.dr.sc. Mirko Köhler
Naslov završnog rada:	Android aplikacija za konverziju i praćenje valuta i kriptovaluta
Znanstvena grana rada:	**Informacijski sustavi (zn. polje računarstvo)**
Zadatak završnog rada	Objasniti mehanizam promjene vrijednosti valuta i kriptovaluta. Izraditi Android aplikaciju pomoću koje korisnik može pratiti trenutno stanje valuta i kriptovaluta. U aplikaciju ugraditi funkcionalnost pretvorbe valuta i kriptovaluta u željenu valutu. Testirati rad aplikacije i opisati postupak testiranja.
Prijedlog ocjene pismenog dijela ispita (završnog rada):	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 2 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 2 bod/boda Jasnoća pismenog izražavanja: 2 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene mentora:	25.09.2018.

Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

IZJAVA O ORIGINALNOSTI RADA

Osijek, 04.10.2018.

Ime i prezime studenta:	Mario Džoić
Studij:	Preddiplomski stručni studij Elektrotehnika, smjer Informatika
Mat. br. studenta, godina upisa:	AI4505, 20.09.2017.
Ephorus podudaranje [%]:	11%

Ovom izjavom izjavljujem da je rad pod nazivom**: Android aplikacija za konverziju i praćenje valuta i kriptovaluta**

izrađen pod vodstvom mentora Izv. prof. dr. sc. Krešimir Nenadić

i sumentora

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.

Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

SADRŽAJ

1. UVOD 1
1.1 Zadatak završnog rada 1
2. VALUTE I KRIPTOVALUTE 2
2.1 Definicija novca 2
2.2 Funkcije novca 2
2.3 Novčane jedinice (valute) 2
2.4 Kriptovalute 2
2.5 Blokchain 3
3. KORIŠTENE TEHNOLOGIJE 4
3.1 Android 4
3.2 Android studio 5
3.3 Java 6
3.4 JSON 7
3.5 XML 8
3.6 API 9
4. RAZVOJ ANDROID APLIKACIJE ZA KONVERZIJU I PRAĆENJE VALUTA I KRIPTOVALUTA 10
4.1 Korisničko sučelje 10
4.2 Android Manifest 10
4.3 Izdrada aplikacije 12
4.4 Povezivanje API-a 17
5. ZAKLJUČAK 21
LITERATURA 22
SAŽETAK 23
ABSTRACT 24
ŽIVOTOPIS 25

1. UVOD

Tema ovog završnog rada je izrada Android aplikacije za konverziju i praćenje valuta i kriptovaluta. S obzirom na sve veću popularnost i vrijednost kriptovaluta, korisnici istih, ali i oni koji žele to postati imaju potrebu pratiti stanje u bilo kojem trenutku te vršiti konverziju. Kako danas velika većina ljudi posjeduje Android uređaje, putem ove aplikacije mogu to učiniti. Također, korisnik može vidjeti stanje valute kroz neko vrijeme. U drugom poglavlju su objašnjene valute i kriptovalute te mehanizam promjene njihovih vrijednosti. U trećem poglavlju su obrađene korištene tehnologije i alati za izradu aplikacije. U četvrtom poglavlju je objašnjena izrada aplikacije i prikazani su rezultati rada. Aplikacija je izrađena koristeći programski jezik Java u razvojnom okruženju Android Studio.

1.1 Zadatak završnog rada

Zadatak ovog završnog rada je izrada Android aplikacije za konverziju i praćenje valuta i kriptovaluta. Potrebno je objasniti mehanizam promjene valuta i kriptovaluta.

2. VALUTE I KRIPTOVALUTE

U ovom poglavlju su ukratko opisani pojmovi novac, odnosno valute i njegove funkcije te kriptovalute i *blockchain*.

2.1 Definicija novca

Novac se može definirati kao stvar kojom se može plaćati i razmjenjivati roba. Novac uglavnom služi kao sredstvo razmjene [1].

2.2 Funkcije novca

Novac služi za razmjenu ljudskih roba te osigurava lakšu, bržu i sigurniju razmjenu roba. Postoje dvije vrste razmjena, a to su domaća i međunarodna. Kod domaće razmjene, domaća valuta služi kao sredstvo razmjene. Nasuprot tome, u međunarodnoj razmjeni, kao svjetski novac, koriste se najjače svjetske valute kao što su euro i dolar. Na međunarodnom tržištu, odnos između pojedinih valuta ovisi o stabilnosti istih te o politici središnjih banaka. Za stabilnost valuta važna je stabilnost privrede te dobra monetarna politika. Namjernim povećanjem neke valute u opticaju, može se rušiti tečaj valute na međunarodnom tržištu. Obično to rade države koje imaju visoki vanjski dug. Na stabilnost valute, uz podršku središnjih banaka, mogu utjecati međunarodni špekulanti valutama na način da kupuju određene valute. Na taj način tečaj kupljenih valuta raste, što u konačnici dovodi do ostvarivanja zarade prodajom [1].

2.3 Novčane jedinice (valute)

Svaka država ima vlastitu valutu. Tečaj valute države, u kojoj je velika proizvodnja i koristi se u drugim državama, raste. Suprotno tome, tečaj valute države u kojoj je uvoz veliki, a izvoz mali, vrijednost te valute će padati. Hrvatska kuna često jača u ljetnim mjesecima zbog turizma. Veliki broj turista dolazi u Hrvatsku te mijenja strane valute u hrvatsku kunu i time dolazi do jačanja tečaja kune u odnosu na euro [1].

2.4 Kriptovalute

Kriptovalute se mogu promatrati kao vrste digitalnog novca osmišljenog kao sredstvo razmjene koje koristi kriptografiju kako bi se osigurale financijske transakcije. Sadrži sve osobine prave valute. Prva i najpoznatija kriptovaluta je bitcoin. Za razliku od centralnih bankarskih

sustava, kriptovalute koriste decentraliziranu kontrolu koristeći tehnologiju *blockchain*. Kriptovalute se mogu promatrati kao sustavi koji:

1. ne sadrže centralnu točku
2. sadrže broj kriptovalutnih jedinica i njihove vlasnike
3. definiraju može li se nova kriptovalutna jedinica stvoriti
4. vlasništvo kriptovalutne jedinice se može dokazati kriptografski
5. omogućavaju transakcije

Nakon bitcoin-a nastale su mnoge druge alternativne kriptovalute (eng. *Altcoins*). Kriptovalute se mogu promatrati kao digitalni tokeni. Digitalni tokeni se prenose na način da se broj tokena na jednoj adresi smanji, a na drugoj adresi se broj tokena poveća. U mreži kriptovaluta, „rudarenje“ služi kao potvrda transakcija te za taj trud rudari su nagrađeni kriptovalutnom jedinicom. Novčanik kriptovaluta sadrži javni i privatni ključ koji se koriste za slanje i primanje kriptovaluta [2], [3].

2.5 Blokchain

Blockchain (hrv. Lanac-blokova) je lanac podatkovnih zapisa u blokovima povezanih kriptografijom koja osigurava privatnost i sigurnost. Svaki blok u nizu ovisi o prethodnom bloku. Navedeni blokovi sadrže podatke o transakcijama. *Blockchain* je dizajniran tako da je otporan na promjenu podataka i u svakom trenutku se mogu provjeriti transakcije između dvije strane. Obično je upravljan *peer-to-peer* mrežom. To je mreža u kojoj dvije točke razmjenjuju informacije na način da obje strane sadrže ista prava. Promjena nekog bloka u lancu zahtjeva promjenu svih ostalih blokova. *Blockchain* sustav se smatra sigurnim sustavom otpornim na kvarove te se može definirati kao decentralizirani sustav koji prati sve transakcije između korisnika. Danas se *blockchain* uglavnom koristi za kriptovalute u svrhu praćenja transakcija [2], [3].

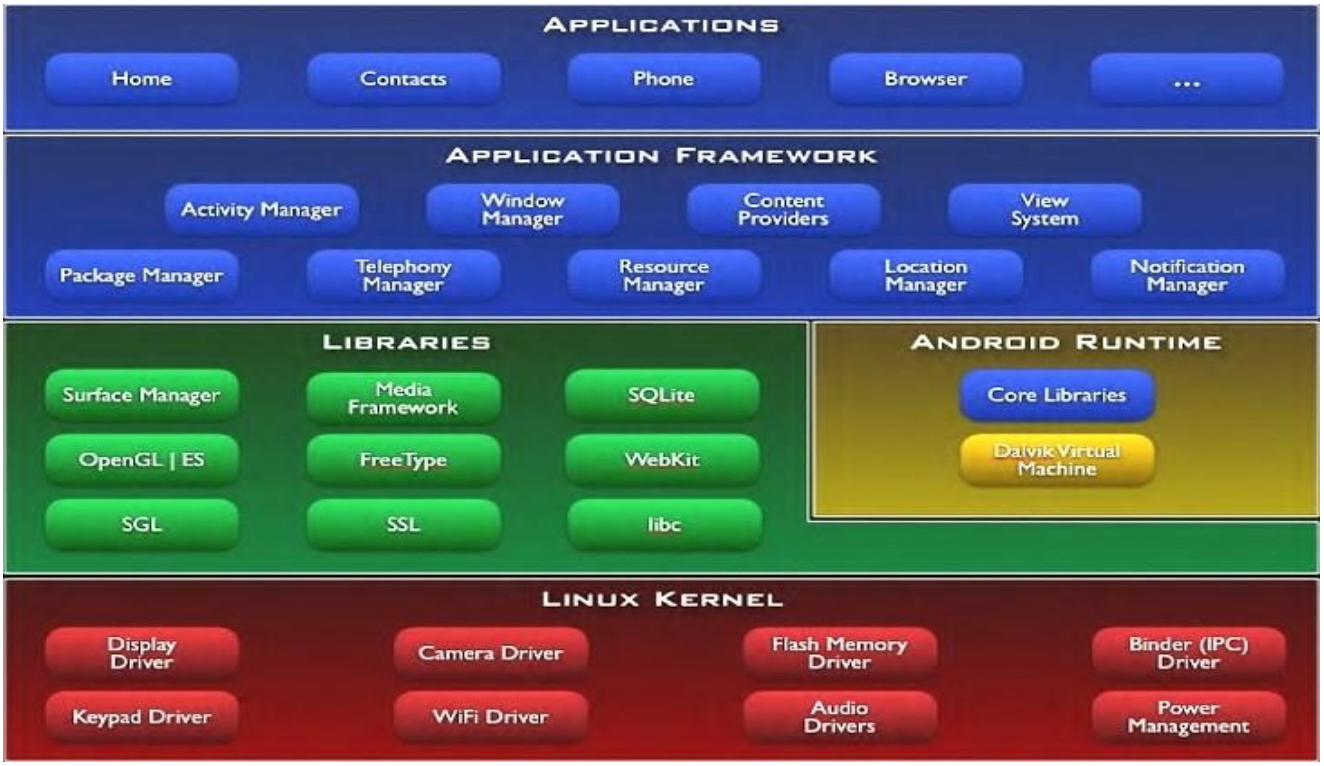
3. KORIŠTENE TEHNOLOGIJE

U ovom poglavlju će se opisati tehnologije korištene pri izradi Android aplikacije.

3.1 Android

Android je operacijski sustav koji se temelji na jezgri Linux, a koristi se u mobilnim uređajima. Android Inc je tvrtka koja je razvila Android operacijski sustav, a 2005. godine Google preuzima navedenu tvrtku te nastavlja s razvojem Andorid-a. Prvi komercijalni Android uređaji su se pojavili 2008. godine. Trenutna verzija Androida je 9.0 *Pie*. Android operacijski sustav je otvoreni projekt poznat pod pojmom *Android Open Source Project* (AOSP). Kao projekt otvorenog izvornog koda, cilj Androida je izbjegavanje situacije u kojoj jedna strana ograničava druge i nadzire inovacije.

Jezgra Android sustava je bazirana na jezgri Linuxa. Jezgra je također ovisna o Android uređaju. Memorija na Android uređajima je podijeljena na nekoliko vrsta, kao što su memorija za korisničke aplikacije te memorija za operacijski sustav. Korisnici Android uređaja nemaju pristup korijenu operacijskog sustava niti nekim osjetljivim dijelovima. Iskorištavanjem sigurnosnih propusta Android-a, može se pristupiti navedenim dijelovima. Android operacijski sustav ne sadrži GNU C biblioteku koja osigurava jako bitne biblioteke za GNU/Linux sustave. Kao zamjenu koristi Bionic biblioteku. Na vrhu se nalaze korisničke aplikacije koje se služe raznim bibliotekama, ispod korisničkih aplikacija se nalaze biblioteke sustava, rutine sistemskih programa te Dalvik virtualni stroj koji koriste Linux jezgru, prema slici 3.1.



Sl. 3.1. Arhitektura Android operativnog sustava [5].

Sustavima na sklopovskom nivou upravlja jezgra operacijskog sustava, koja se nalazi i djeluje kao sloj između hardvera i operacijskog sustava. Biblioteke sustava su prilagođene svakom uređaju zasebno te su radi brzine izvođenja napisane C i C++ programskim jezicima. Do verzije 5.0 koristi se *Dalvik* virtualni sustav, koji maksimalno iskorištava Linux jezgru na način da koristi više virtualnih strojeva. Na taj se način svaka aplikacija nalazi unutar zasebnog virtualnog okruženja. Od verzije 5.0 se koristi ART (eng. *Android Runtime*) novi sustav koji prevodi cijeli *bytecode* aplikacije u strojni kod prilikom instaliranja aplikacije. Prednost korištenja Bionic biblioteke u odnosu na GNU C biblioteku je bolja optimizacija za slabije procesore te brže pokretanje [4], [5].

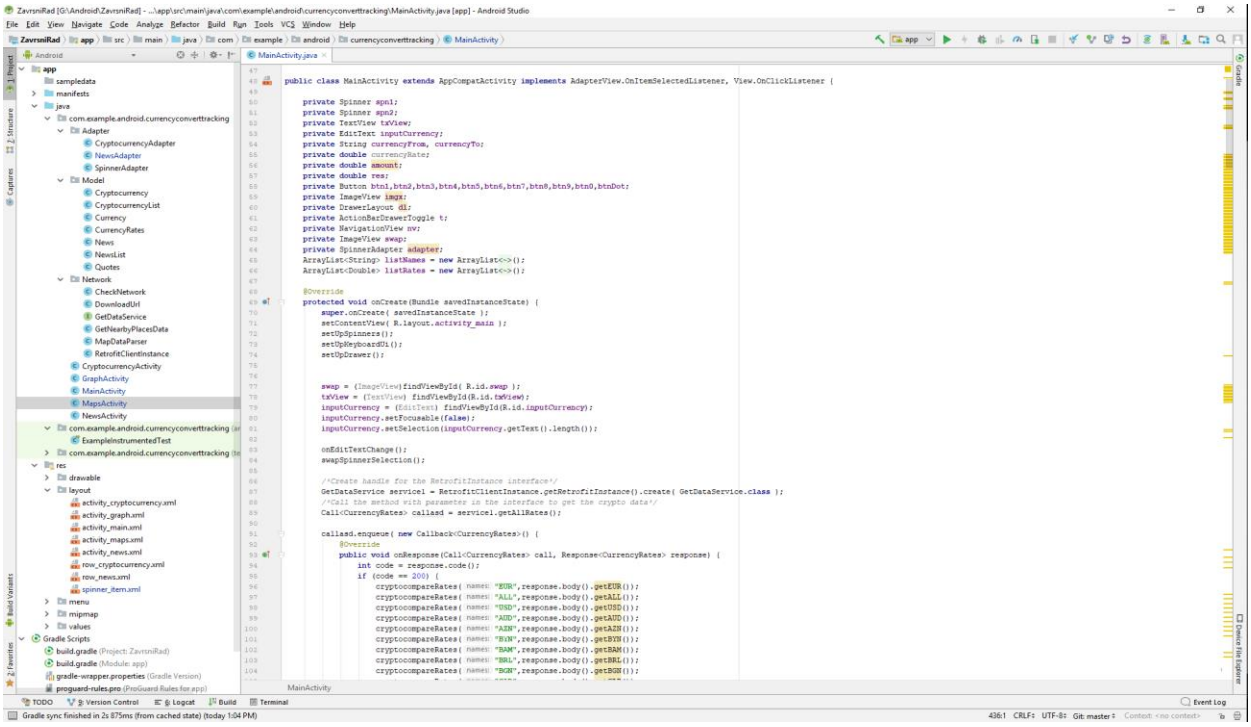
3.2 Android Studio

Android Studio je softver korišten za razvoj aplikacija na Android platformi. U kolovozu 2018. godine izašla je zadnja verzija 3.1.4. Android Studio se može koristiti na operacijskim sustavima Windows, Linux i Mac OS X te podržava svojstva kao što su:

- *gradle-based build* sustav
- brze virtualne uređaje s velikim brojem značajki

- jedinstveno okruženje u kojem se mogu razvijati aplikacije za sve Android uređaje
- brzo pokretanje aplikacije s izmjenama bez stvaranja novog APK-a
- primjere kodova
- dodatne alate za testove
- lint alate za ubrzavanje rada, korisnosti te kompaktibilnosti
- C++ i NDK podrška

Svrha Android Studi-a je ubrzavanje razvoja te povećanje kvalitete izrade aplikacija. Sučelje Android Studio razvojnog okruženja može se vidjeti prema slici 3.1. [6], [7].



Sl. 3.2. Sučelje Android Studio razvojnog okruženja

3.3 Java

Java je objektno orijentirani programski jezik. Jednom napisan i preveden Java kod bi se mogao pokrenuti na svim operacijskim sustavima koji podržavaju Javu bez ponovnog prevođenja. Java programi se obično prevode u *bytecode* koji može pokrenuti JVM (eng. *Java virtual machine*). Namjera je bila stvoriti programski jezik koji bi zadovoljio sljedeća svojstva:

- mora biti jednostavan i objektno orijentiran

- robustan i siguran
- mora biti neovisan o arhitekturi
- izvršavati se s visokim performansama

Programi pisani u Javi mogu se koristiti na svim operacijskim sustavima koji sadrže JVM (eng. *Java Virtual Machine*), što čini veliku prednost u odnosu na ostale programske jezike. Ti programi su nešto sporiji i zahtijevaju više memorije nego programi pisani u programskom jeziku C++ iako, svakom novom verzijom Jave raste i brzina programa. Klase u programskom jeziku Java su organizirane i odvojene u pakete (eng. *Packages*). Svaki paket sadrži sučelja, klase i iznimke. Java programski jezik je jedan od glavnih jezika za razvoj Android aplikacija. Zadnja verzija je Java 10 koja je izašla u ožujku 2018. godine [9].

3.4 JSON

JSON (engl. *JavaScript Object Notation*) je format koji koristi tekst čitljiv čovjeku za prijenos podataka. Neovisan je format dobiven iz JavaScripta. Koristi se za razmjenu podataka između servera i aplikacije. U nastavku je prikaz JSON formata (Programski kod 3.1). JSON objekt se nalazi unutar vitičastih zagrada te može sadržavati podatke tipa *String*, broj i boolean [10].

```
{
    "data": [
        {
            "id": 233,
            "name": "SolarCoin",
            "symbol": "SLR",
            "website_slug": "solarcoin",
            "rank": 562,
            "circulating_supply": 47168341.0,
            "total_supply": 98034518136.0,
            "max_supply": null,
            "quotes": {
                "USD": {
                    "price": 0.0739734134,
                    "volume_24h": 29697.6846314157,
                    "market_cap": 3489203.0,
                    "percent_change_1h": -0.33,
                    "percent_change_24h": -1.29,
                    "percent_change_7d": -26.32
                }
            },
            "last_updated": 1537288335
        }
    ],
    "metadata": {
        "timestamp": 1537287802,
        "num_cryptocurrencies": 1969,
        "error": null
    }
}
```

Programski kod 3.1. Primjer Json formata

3.5 XML

XML (eng. *eXtensible Markup Language*) je jezik za označavanje podataka, koji sadrži skup određenih pravila prevođenja dokumenta u format prepoznatljiv čovjeku i računalu. Glavni cilj je jednostavnost i mogućnost korištenja diljem interneta. Kako bi se zadovoljio glavni cilj, sadržaj treba uokviriti oznakama koje imaju lako shvatljivo značenje ili ga opisuju. Sadržaj XML dokumenta omeđen je početnom i završnom oznakom (eng. *tag*). U nastavku se nalazi prikaz XML koda (Programski kod 3.2) [11].

```xml
<?xml version="1.0" encoding="utf-8"?>
    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="5dp"
        android:id="@+id/imageView"
        android:layout_weight="1" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="5dp"
        android:textColor="@color/white"
       android:textAppearance="@style/Base.TextAppearance.AppCompat.Large"
        android:id="@+id/textView"
        android:layout_weight="1"/>
```

Programski kod 3.2. Primjer XML koda

3.6 API

API (eng. *application programming interface*) je skup pravila, komunikacijskih protokola i alata za izradu programa. Generalno se može definirati kao metoda komunikacije između komponenata. Dobar API olakšava programeru izradu programa. API se može koristiti za web sustave, operacijske sustave, baze podataka, računalno sklopovlje ili programske biblioteke.

4. RAZVOJ ANDROID APLIKACIJE ZA KONVERZIJU I PRAĆENJE VALUTA I KRIPTOVALUTA

U ovom poglavlju će se detaljno opisati razvoj aplikacije uz primjere i objašnjenja. Također, u ovom poglavlju će biti prikazane slike aplikacije. Kako je ranije navedeno, aplikacija je izrađena u programu Android Studio, koristeći programski jezik Javu. Korišteni su cryptocompare api, coinmarketcap api, google map i google places api za dohvaćanje bitnih podataka za rad aplikacije.

4.1 Korisničko sučelje

U ovom potpoglavlju su opisane aktivnosti koje sadrži aplikacija, točnije 5 aktivnosti. U tablici 4.1. je naveden popis istih.

Tab. 4.1. Popis aktivnosti unutar aplikacije

Vrsta	Naziv	Opis
Aktivnost	*MainActivity*	Sučelje za konverziju valuta i kriptovaluta, sadrži 2 padajuća izbronika te sadrži okvir za unos količine i okvir za ispis rezultata
Aktivnost	*GraphActivity*	Sučelje za prikaz grafa ovisnosti dvije valute ili kriptovalute, sadrži 2 padajuća izbornika te sadrži opciju željenog vremena u koje će se graf iscrtati
Aktivnost	*NewsActivity*	Sučelje koje sadrži vijesti vezane za kriptovalute
Aktivnost	*MapsActivity*	Sučelje koje prikazuje trenutnu lokaciju korisnika te lokaciju obližnjih bankomata
Aktivnost	*CryptocurrencyActivity*	Sučelje koje prikazuje 100 najkorištenijih kriptovaluta te njihovo stanje

4.2 Android Manifest

Svaka aplikacija mora sadržavati AndroidManifest.xml datoteku koja se nalazi u korijenskom (eng. *Root*) direktoriju. Aplikacija se ne može pokrenuti bez podataka koji se nalaze u Manifest datoteci. Dozvole su jedna od informacija koje su jako bitne za rad same aplikacije. Bez njih aplikacija ne bi mogla ispravno funkcionirati. Uz dozovle, Manifest datoteka sadrži ime paketa aplikacije, komponente aplikacije te hradverske i programske značajke potrebne za instalaciju aplikacije.

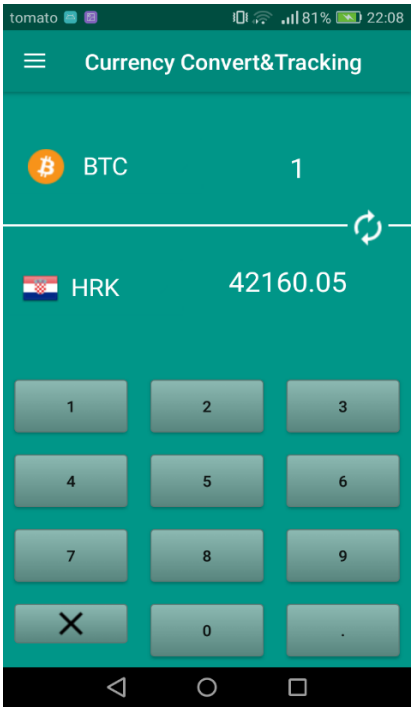
```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.android.currencyconverttracking">

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="com.google.android.providers.gsf.permisson.READ_GSERVICES" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/icon"
        android:label="Currency Convert&Tracking"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".GraphActivity" />
        <activity android:name=".NewsActivity" />

        <meta-data
            android:name="com.google.android.geo.API_KEY"
            android:value="AIzaSyCCoM4g_Kv1mxoMvjl6xrOyCwv88IUBXRs" />

        <activity
            android:name=".MapsActivity"
            android:label="Map" />
        <activity android:name=".CryptocurrencyActivity" />
    </application>
</manifest>
```

Sl. 4.1. Manifest datoteka

Prema slici 4.1. vidljivo je da su potrebne dozvole za pristup internetu, dohvaćanje približne i precizne lokacije te dozvola za pristup informacijama o mreži. Sadrži google api key koji je potreban za neke od funkcija aplikacije.

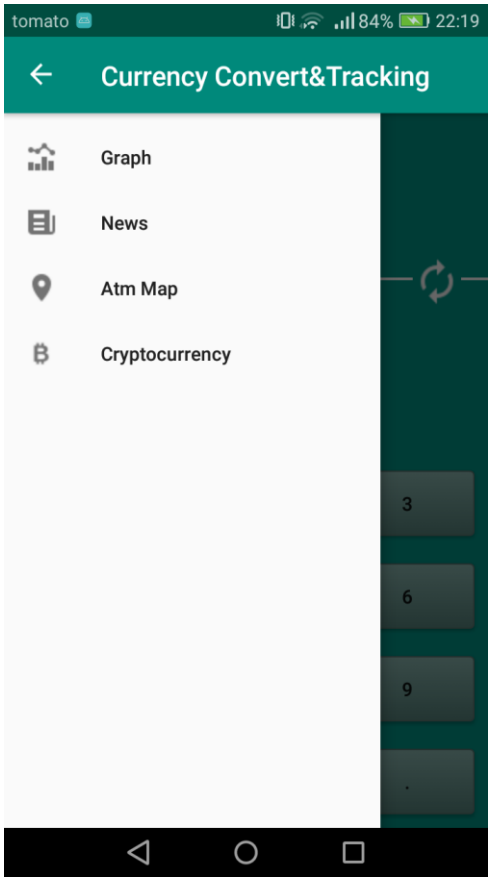
4.3 Izdrada aplikacije

Prilikom pokretanja aplikacije korisniku se prikaže zaslon prema slici 4.2.



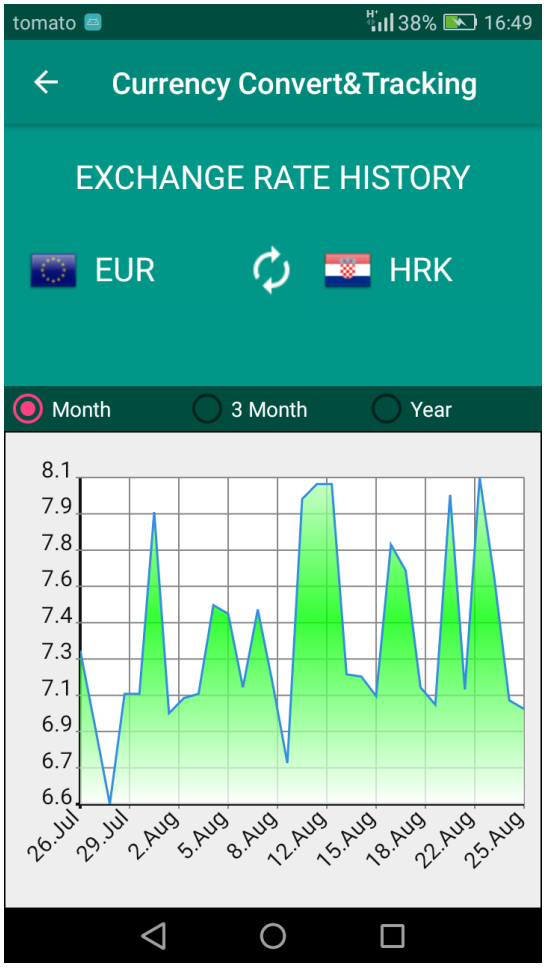
Sl. 4.2. Početni zaslon

Korisnik ima dva padajuća izbornika te odabire valutu ili kriptovalutu iz koje želi vršiti konverziju pa zatim odabire valutu ili kriptovalutu u koju želi vršiti konverziju. Korisnik može kliknuti na gumb kako bi zamijenio mjesta odabranim opcijama. Tipkovnica je napravljena od gumba koji su smješteni u relative layout. Relative layout je grupa elementa koji su posloženi u odnosu jedan na drugoga. Za dobivanje ovakvog izgleda početnog zaslona korišten je još linear layout. U linear layout-u su elementi prikazani u jednom smjeru koji može biti okomit ili vodoravan. Unošenjem željenog iznosa u *editText*, aplikacija automatski izračunava vrijednost konverzije te prikazuje rezultat u *textView*-u. Ukoliko korisnik nema pristup internetu, neće moći izvršiti konverziju. Prilikom pokretanja ove aktivnosti, korištenjem biblioteke retrofit, koja nam olakšava dohvaćanje i korištenje JSON podataka, te korištenjem cryptocompare api-a dobije se tečajna vrijednost svih valuta i kriptovaluta koje aplikacija podržava na osnovu dolara. Za navigiranjem aplikacijom korištena je navigacijska traka koja se može otvoriti klikom na tzv. hamburger ikonu ili povlačenjem prsta s lijeva na desno kao što je vidljivo prema slici 4.3.



Sl. 4.3. Navigacijska traka

U navigacijskom prikazu se mogu odabrati neke od navedenih opcija. Svaka od navedenih opcija otvara novu aktivnost. Navigacijska traka se zatvara klikom na gumb strelicu ili povlačenjem prsta s desna na lijevo. Odabirom *Graph* opcije u navigacijskom prozoru otvara se nova aktivnost prema slici 4.4.



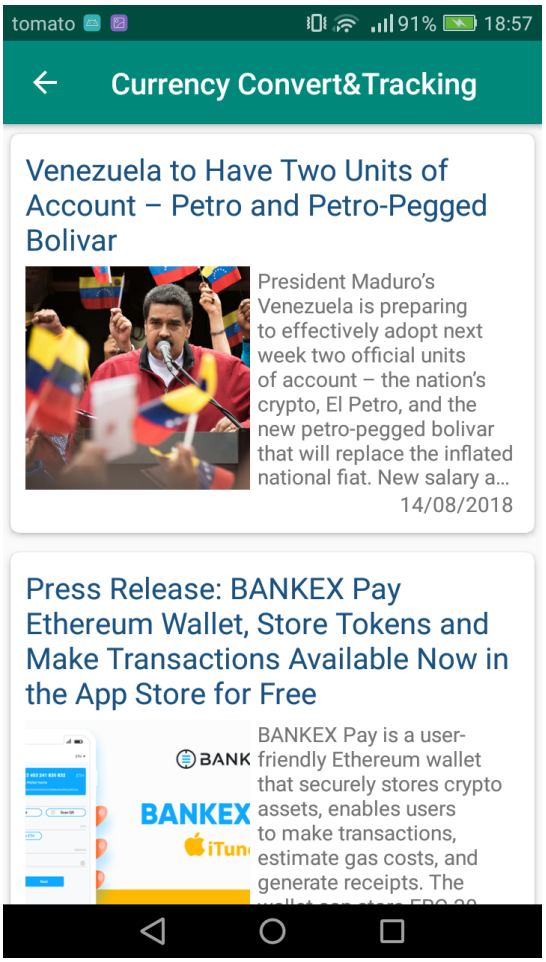
Sl. 4.4. Zaslon nakon otvaranja *Graph* aktivnosti

Funkcinalost ove aktivnosti je iscrtavanje grafa na osnovu odabranih valuta ili kriptovaluta te na osnovu odabranog vremena. Za iscrtavanje grafa korištena je biblioteka AndroidPlot. AndroidPlot je besplatna biblioteka koja se koristi za iscrtavanje dinamičkih i statičkih grafova. Za dobivanje podataka u određenom vremenu korišten je cryptocompare api, a za odabir vremena korišten je *radioButton*. *RadioButton* omogućava istovremeno označavanje samo jednog gumba. Datum preko api-a se dobije u *unix* formatu te se mora formatirati kako bi bio čitljiv čovjeku kao što je prikazano programskim kodom 4.1.

```
        private SimpleDateFormat sdf = new SimpleDateFormat( "d.MMM");
```

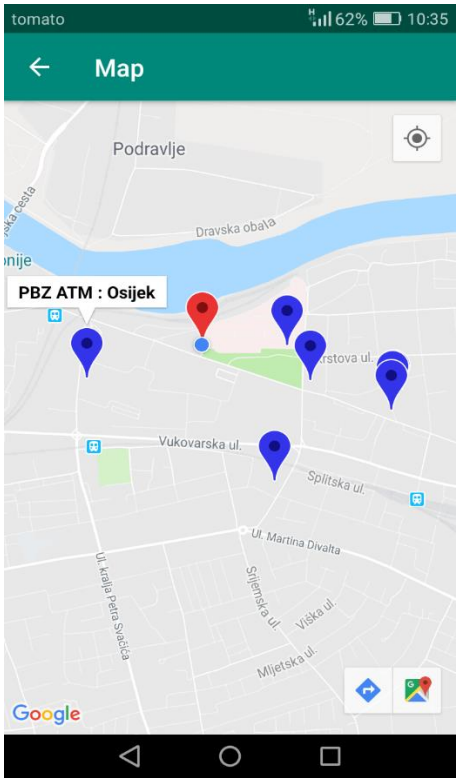
Programski kod 4.1. Formatiranje datuma

Odabirom *News-a* u navigacijskom prozoru otvara se nova aktivnost prema slici 4.5.



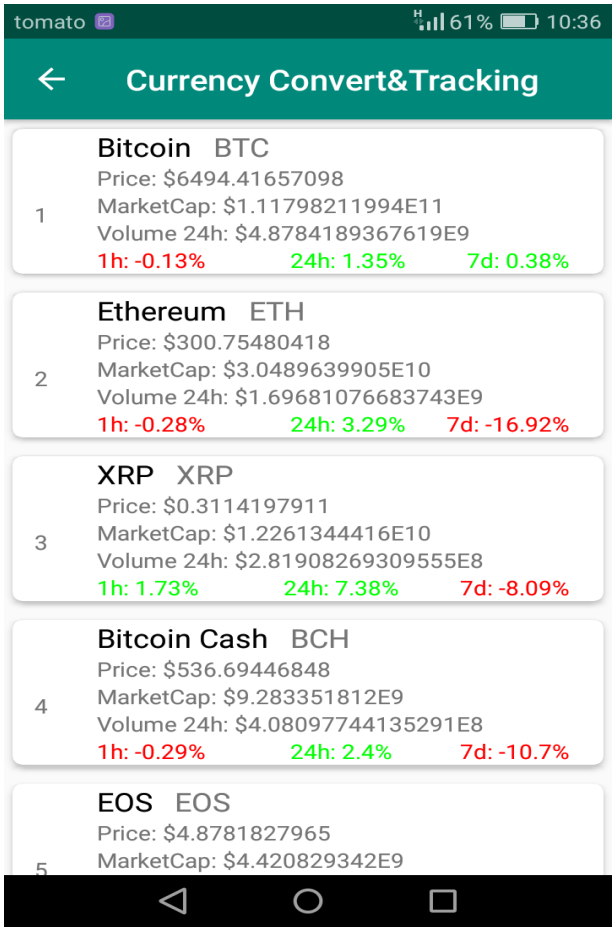
Sl. 4.5. Zaslon nakon otvaranja *News* aktivnosti

Na ovoj aktivnosti korisnik može gledati vijesti za kriptovalute te klikom na naslov, putem interneta, otvara se cijeli članak u web pregledniku. Vijesti su prikazane u *RecyclerView-u*. Odabirom *Atm map* u navigacijskom prozoru otvara se nova aktivnost prema slici 4.6.



Sl. 4.6. Zaslon nakon otvaranja *Atm Map* aktivnosti

Ova aktivnost prikazuje trenutnu lokaciju korisnika te lokacije obližnjih bankomata. Za prikaz ovih lokacija koriste se google map i google places API. Odabirom *Cryptocurrency* opcije u navigacijskom prozoru, otvara se nova aktivnost prema slici 4.7.



Sl. 4.7. Zaslon nakon otvaranja *Cryptocurrency* aktivnosti

U ovoj aktivnosti korisnik može pratiti stanje 100 najkorištenijih kritpovaluta. Za dohvaćanje ovih podataka korišten je coinmarketcap api.

4.4 Povezivanje API-a

U aplikaciji je korišteno nekoliko API-a. U ovom potpoglavlju će biti objašnjeno korištenje coinmarketcap API-a. Za povezivanje i korištenje API-a korištena je biblioteka Retrofit. Za korištenje coinmarketcap API-a potreban je ključ koji se dobije registracijom na coinmarketcapi stranici. Prilikom pokretanja *Cryptocurrency* aktivnosti, asinkrono šalje zahtjev, koji vraća *callback*. Postoje dva *cllback*-a *onResponse*() ako je poziv uspješan i *onFailure*() ako poziv nije uspješan. Kako je zahtjev asinkron, Retrofit obavlja radnje u pozadinskoj jezgri. U nastavku su prikazani primjeri svake klase (Programski kod 4.1., Programski kod 4.2., Programski kod 4.3., Programski kod 4.4., Programski kod 4.5.) Kako bi se zahtjev poslao za mrežnim podatcima, treba kreirati instancu koristeći *Retrofit.Builder* kao što je prikazano programskim kodom 4.1.

Programski kod 4.2. pokazuje sučelje koje definira HTTP metodu koja se poziva te sadrži API put. Navedena metoda vraća podatke koji se nalaze u *Call* sučelju. Klasa koja služi za pohranu podataka u *ArrayList* kako bi se lakše koristile prikazana je programskim kodom 4.3. U aktivnosti *Cryptocurreny* kreira se instanca Retrofita te se dohvaćaju potrebni podatci (Programski kod 4.4). Za prikaz kriptovalute, kako je viđeno ranije, koristi se funkcija prikazana programskim kodom 4.5.

```
public class RetrofitClientInstance {

    private static Retrofit retrofit = null;
    private static Retrofit retrofit2 = null;


    private static final String BASE_URL = "https://min-
api.cryptocompare.com";
    private static final String BASE_URL_CRYPTO =
"https://api.coinmarketcap.com";


    public static Retrofit getRetrofitInstance() {
        if (retrofit == null) {
            retrofit = new retrofit2.Retrofit.Builder()
                    .baseUrl(BASE_URL)
                    .addConverterFactory( GsonConverterFactory.create())
                    .build();
        }
        return retrofit;
    }

    public static Retrofit getCryptoData() {
        if (retrofit2 == null) {
            retrofit2 = new retrofit2.Retrofit.Builder()
                    .baseUrl(BASE_URL_CRYPTO)
                    .addConverterFactory( GsonConverterFactory.create())
                    .build();
        }
        return retrofit2;
    }

}
```

Programski kod 4.1 *RetrofitClientInstance* klasa

```
public interface GetDataService {
    @GET("/data/v2/news/")
    Call<NewsList> getAllNews();

    @GET("/v2/ticker/?&sort=rank&structure=array")
    Call<CryptocurrencyList> getAllCrypto();


@GET("/data/price?fsym=USD&tsyms=USD,EUR,ALL,EUR,AUD,AZN,BYN,BAM,BRL,BGN,CAD," +

"CNY,HRK,CZK,DKK,GEL,HKD,HUF,ISK,INR,JPY,MKD,MXN,MDL,NZD,NOK,PLN,RON," +

"RUB,RSD,SGD,ZAR,KRW,SEK,CHF,TRY,UAH,GBP,BTC,ETH,XRP,BCH,EOS,XLM,LTC," +

"ADA,MIOTA,USDT,TRX,NEO,XMR,DASH,ETC,XEM,XTZ,BNB,VEN,OMG,ZEC,QTUM,ZRX," +
            "BCD,BCN,DCR,ZIL,ICX,BTS,ISK")
    Call<CurrencyRates> getAllRates();
}
```

Programski kod 4.2 *GetDataService* sučelje

```
public class CryptocurrencyList {

    @SerializedName("data")
    private ArrayList<Cryptocurrency> cryptocurrencyList;

    public ArrayList<Cryptocurrency> getCryptocurrencyArrayList() {
        return cryptocurrencyList;
    }

    public void setCryptocurrencyList(ArrayList<Cryptocurrency>
cryptocurrencyArrayList) {
        this.cryptocurrencyList = cryptocurrencyArrayList;
    }
}
```

Programski kod 4.3 *CryptocurrencyList*

```
GetDataService service =
RetrofitClientInstance.getCryptoData().create(GetDataService.class);
Call<CryptocurrencyList> call = service.getAllCrypto();
call.enqueue(new Callback<CryptocurrencyList>() {
    @Override
    public void onResponse(Call<CryptocurrencyList> call,
Response<CryptocurrencyList> response) {
        int code = response.code();
        if (code == 200) {
        generateCryptoList(response.body().getCryptocurrencyArrayList());
    }}

    @Override
    public void onFailure(Call<CryptocurrencyList> call, Throwable t) {
        Toast.makeText(CryptocurrencyActivity.this, "Something went
wrong...Please try later!", Toast.LENGTH_SHORT).show();
    }
});
```

Programski kod 4.4. Slanje zahtjeva u *Cryptocurrency* aktivnosti

```
recyclerView = (RecyclerView) findViewById(R.id.recycler_view_cryptocurrency);
adapter = new CryptocurrencyAdapter(this,cryptoDataList);
RecyclerView.LayoutManager layoutManager = new
LinearLayoutManager(CryptocurrencyActivity.this);
recyclerView.setLayoutManager(layoutManager);
recyclerView.setAdapter(adapter);
}
```

Programski kod 4.5 Funkcija *getCryptocurrencyArrayList*

5. ZAKLJUČAK

U završnom radu je napravljena aplikacija za konverziju i praćenje valuta i kriptovaluta . U teorijskom dijelu je opisan novac i kriptovalute. Također, opisane su tehnologije korištene za izradu aplikacije te je objašnjen postupak izrade i korištenja aplikacije. Za izradu ove aplikacije bilo je potrebno pronaći api koji omogućuje dohvaćanje tečaja za valute i kriptovalute. Također, bilo je potrebno naučiti koristiti JSON podatke. Korištenjem ove aplikacije, korisnik može vršiti konverziju valuta i kriptovaluta, može pratiti vijesti za kriptovalute te pogledati obližnje bankomate. S obzirom na to da aplikacija radi samo ako ima pristup internetu, dobra nadogradnja bi bila napraviti da aplikacija radi bez pristupa internetu, što bi zahtijevalo izradu baze podataka. Također bi se mogao dodati veći broj podržanih valuta i kriptovaluta.

LITERATURA

[1] Wikipedija, Novac, https://hr.wikipedia.org/wiki/Novac, rujan 2018

[2] Bug, https://www.bug.hr/tehnologije/sto-je-u-stvari-blockchain-i-kako-radi-3011, rujan 2018

[3] Bitfalls, Što su to kriptovalute, https://bitfalls.com/hr/2017/08/20/cryptocurrency/, rujan 2018

[4] Mobilni uređaji, Android operativni sustav, https://informatika.buzdo.com/pojmovi/mobile-3.htm, rujan 2018

[5] Wikipedija, Android (Operating system), https://en.wikipedia.org/wiki/Android_(operating_system), rujan 2018

[6] Wikipedija, Android Studio, https://en.wikipedia.org/wiki/Android_Studio, rujan 2018

[7] Developers, Android Studio, https://developer.android.com/studio/intro/, rujan 2018

[8] Wikipedija, Android software development, https://developer.android.com/docs/, rujan 2018

[9] Wikipedija, Java (Programming Languange), https://en.wikipedia.org/wiki/Java_(programming_language), rujan 2018

[10] Wikipedija, JSON, https://en.wikipedia.org/wiki/JSON, rujan 2018

[11] Wikipedija, XML, https://en.wikipedia.org/wiki/XML, rujan 2018

SAŽETAK

Tema završnog rada je izrada Android aplikacije za konverziju te praćenje valuta i kriptovaluta. Glavna funkcionalnost ove aplikacije je vršenje konverzije između valuta i kriptovaluta. Za korištenje ove aplikacije potreban je stalni pristup internetu. Uz glavnu funkcionalnost, neke od dodatnih su praćenje vijesti i prikaz obližnjih bankomata. Bilo je potrebno pronaći odgovarajući api kako bi se dobili potrebni podatci za rad aplikacije. Također je bilo potrebno proučiti na koji način dohvatiti JSON podatke te kako ih koristiti. U teorijskom dijelu je kratko opisan novac i kritpovalute. Za razvoj ove aplikacije korišten je Java programski jezik te Retrofit biblioteka za dohvaćanje i korištenje JSON podataka. AndroidPlot biblioteka je korištena za iscrtavanje grafa. Korišteni su coinmarketcap api, cryptocompare api te google map i google places api koji pružaju potrebne podatke za izradu aplikacije.

Ključne riječi: Android, aplikacija, konverzija, kriptovalute, valute

ABSTRACT

Android application for conversion and tracking of currency and cryptocurrency

The goal of this final work was development of Android application for conversion of currency and cryptovalue tracking. The main function of this application is to make a conversion between currency and cryptovalue. To use this application, permanent internet access is required. With the main functionality of the application, some of the additional functions are tracking news and displaying nearby ATMs. It was necessary to bring the appropriate APIs to get the necessary data for the application work. It was also necessary to read how to retrieve JSON data and how to use them. In the theoretical part, the money and the criticisms are described briefly. To develop this application, the Java programming language and Retrofit library were used to retrieve and create JSON data. The AndroidPlot library was used to plot the graph. Used coinmarketcap api, cryptocompare api and google maps and google places APIs that provide the necessary data for creating the application.

Keywords: Android, apps, conversion, crypts, currencies

ŽIVOTOPIS

Mario Džoić je rođen 30. lipnja 1995. godine u Vinkovcima, Hrvatska. Pohađao je osnovnu školu Mate Lovrak u Županji. Nakon završenog osnovnoškolskog školovanja, upisao je Elektrotehničku školu također u Županji. Godine 2015. upisao je stručni studij Informatike na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku. U Osijeku 2018. godine pohađao je backend developer praksu u trajanju od 8 tjedana u Atos Convergence Creators d.o.o.

Mario Džoić

_____