



Kripto konvertor

(Projektni zadatak)

PREDMET: CS330 Razvoj mobilnih aplikacija

Profesor:

Prof. dr Vladimir Milićević

Student:

Miloš Savić

Br. indeksa: 2377

Beograd, 2018.

## **Sadržaj:**

<b>1. Uvod .....</b>	<b>4</b>
<b>2. Korisnički zahtevi - funkcionalni zahtevi.....</b>	<b>4</b>
<b>3. Korisnički zahtevi - nefunkcionalni zahtevi.....</b>	<b>4</b>
<b>4. Slučaj korišćenja .....</b>	<b>5</b>
<b>5. Hijerarhija projekta .....</b>	<b>5</b>
<b>6. Java klase .....</b>	<b>7</b>
6.1. Glavna klasa aktivnosti.....	7
6.2. Delovi koda .....	12
<b>7. Prikaz aplikacije.....</b>	<b>15</b>

## PROJEKTNI ZADATAK

<b>NAZIV PROJEKTA</b>	Kripto konvertor
<b>RAZVOJNI TIM / PROGRAMER STUDENT</b>	Miloš Savić
<b>PREDMETNI NASTAVNIK</b>	Vladimir Milićević
<b>ASISTENT</b>	Nikola Dimitrijević
<b>ČLANOVI PROJEKTOG TIMA</b>	Miloš Savić
<b>POLJE PROJEKTA</b>	Mobilne aplikacije; Računarske nauke
<b>MESTO IZVOĐENJA PROJEKTA</b>	Prostorije studenta
<b>FIRMA ZA KOJU SE PROJEKAT RADI</b>	Projekat je samostalna ideja
<b>CILJ PROJEKTA</b>	Razvoj funkcionalne android aplikacije
<b>IZLAZ KOJI SE DOBIJA PROJEKTOM</b>	Funkcionalno i testirano softversko rešenje za konverziju kriptovaluta
<b>AKTIVNOSTI TOKOM REALIZACIJE PROJEKTA</b>	Ideja - rezultat naručenog posla ili dogovora projektnog tima;  Izrada projektne dokumentacije; Projektovanje rešenja;  Kodiranje rešenja;  Testiranje i implementacija.
<b>TEHNOLOGIJE I ALATI</b>	Java JDK 8, Android SDK, Android Studio
<b>HARDVERSKI RESURSI</b>	Lični računar - za izradu projekta; Lični mobilni telefon - za testiranje
<b>VREMENSKI PLAN REALIZACIJE PROJEKTA</b>	18.07. - 23.07.2018.
<b>MESTO I DATUM IZRADE PROJEKTOG ZADATKA</b>	23.07.2018.

## 1. Uvod

Aplikacija "Kripto konvertor" kao što joj samo ime kaže predstavlja softver za konvertovanje kriptovaluta u obične valute. Aplikacija u pozadini preuzima JSON podatke o trenutnom kursu koje kasnije računa sa unetim iznosom. Takođe ima ugrađenu mogućnost otvaranja veb stranice koja prikazuje kurseve svih kriptovaluta.

## 2. Korisnički zahtevi - funkcionalni zahtevi

Funkcionalni zahtevi aplikacije predstavljaju spisak zahteva koji se odnose na funkcionalnosti i mogućnosti aplikacije. Oni su definisani i obrazloženi sledećim elementima:

- **Konvertovanje u više valuta** - korisnik može da konvertuje sve najkorišćenije kriptovalute u mnogo različitih običnih valuta koje su ponuđene na listi.
- **Dinamička konverzija** - prilikom unosa iznosa konverzija se vrši automatski bez potrebe da se dodano klikne dugme za računanje.
- **Prikaz veb stranice** - korisnik može da otvori veb stranicu na kojoj se nalazi trenutni kurs.

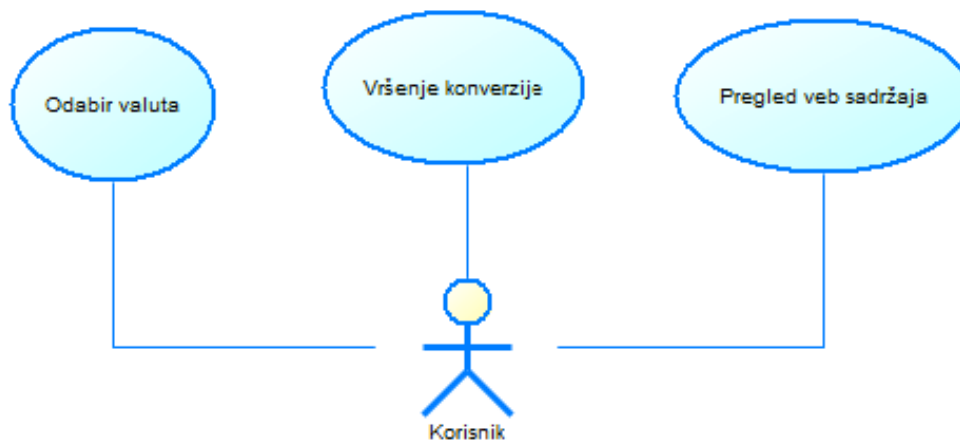
## 3. Korisnički zahtevi - nefunkcionalni zahtevi

Nefunkcionalni zahtevi predstavljaju zahteve koji se tiču načina implementacije aplikacije.

- **Brz odziv** - aplikacija ima odlične performanse.
- **Notifikacije** - aplikacija će obavestiti korisnika ukoliko nije povezan na internet ili ako je protok loš.

## 4. Slučaj korišćenja

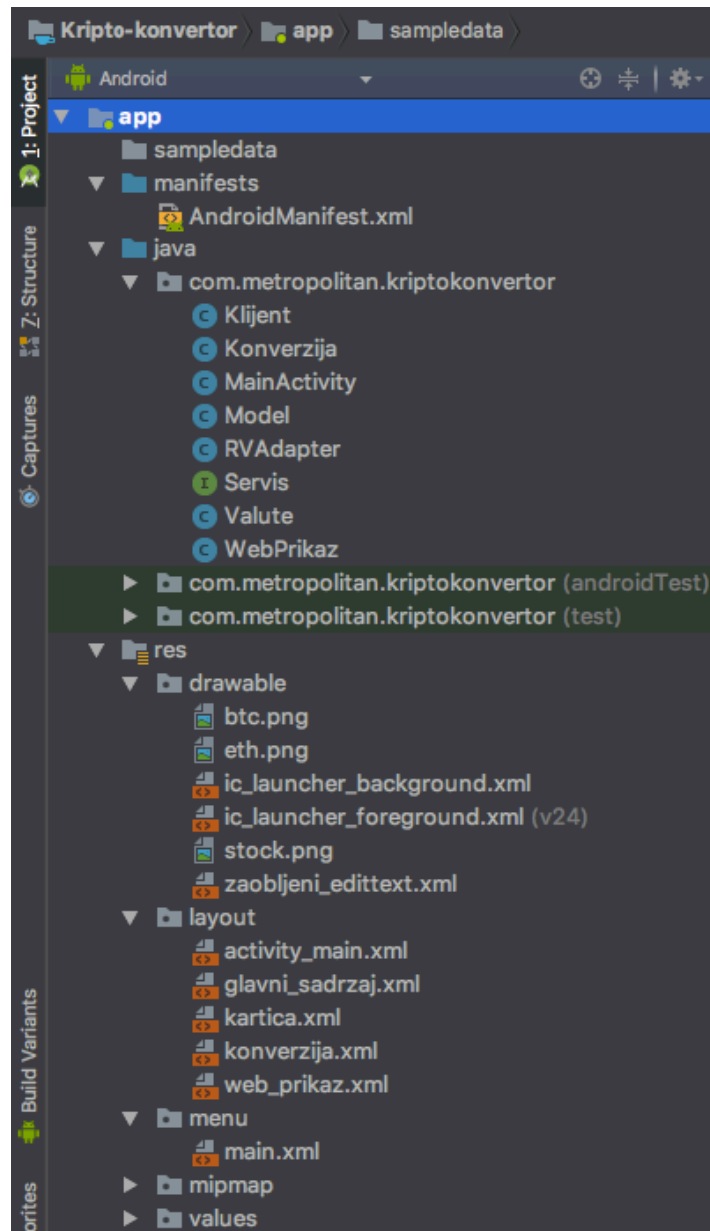
Slučajevi korišćenja prikazuju šta korisnik može da radi u aplikaciji. (Slika 1.)



Slika 1. Dijagram slučaja korišćenja

## 5. Hijerarhija projekta

Na sledećoj slici se može videti hijerarhija projekta. (Slika 2.)



Slika 2. Hijerarhija projekta

## 6. Java klase

### 6.1. Glavna klasa aktivnosti

```
package com.metropolitan.kriptokonvertor;

import android.app.AlertDialog;
import android.app.ProgressDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.support.annotation.Nullable;
import android.support.design.widget.FloatingActionButton;
import android.support.design.widget.Snackbar;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;

import com.google.gson.JsonObject;

import java.util.ArrayList;
import java.util.List;

import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class MainActivity extends AppCompatActivity {
    private ProgressDialog pgDijalog;
```

```

List<Model> kursevi = new ArrayList<>();
List<Model> iniKurs = new ArrayList<>();
private RecyclerView rv;
private RVAdapter rvAdapter;
private FloatingActionButton fab;
boolean provereno[] = Valute.proveriValute();

@Override
protected void onCreate(@Nullable Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    rv = findViewById(R.id.recycler);
    fab = findViewById(R.id.fab);
    pocetniPrikaz();
}

//Pravi listu valuta za dodavanje kartica
private void dodajKarticu(){
    String[] lista = Valute.kod;
    AlertDialog ad = new AlertDialog.Builder(this)
        .setTitle(R.string.dialogMessage)
        .setMultiChoiceItems(lista, provereno, new
DialogInterface.OnMultiChoiceClickListener() {
            @Override
            public void onClick(DialogInterface dialogInterface, int i, boolean b) {
                if (b){
                    provereno[i] = true;
                    iniKurs.add(kursevi.get(i));
                    rvAdapter.notifyItemInserted(rvAdapter.getItemCount());
                }else {
                    provereno[i] = false;
                    int index = iniKurs.indexOf(kursevi.get(i));
                    iniKurs.remove(index);
                    rvAdapter.notifyItemRemoved(index);
                }
            }
        })
        .create();
    ad.show();
}

```



```

        }
    }

    }).setNeutralButton("OK", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int i) {
            dialogInterface.dismiss();
        }
    })
    .create();
ad.show();
}

private void pocetniPrikaz(){
    pgDijalog = new ProgressDialog(this);
    pgDijalog.setMessage("Preuzimanje podataka");
    pgDijalog.setCancelable(false);
    pgDijalog.show();
    rv.setLayoutManager(new LinearLayoutManager(this));
    rv.smoothScrollToPosition(0);
    //Proverava da li je konekcija dostupna
    if (isKonekcijaDostupna(this)){
        fab.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                dodajKarticu();
            }
        });
        getResponse();
    }else {
        pgDijalog.dismiss();
        //Notifikacija ukoliko internet nije dostupan
        setSnackBar("Uključite internet konekciju");
    }
}
}

```

```

private void getResponse(){
    Klijent kl = new Klijent();
    Servis servis = kl.getKlijent().create(Servis.class);
    Call<JsonObject> call = servis.getKurs();
    call.enqueue(new Callback<JsonObject>() {
        @Override
        public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
            fab.setVisibility(View.VISIBLE);
            pgDijalog.dismiss();
            JsonObject jsonObject = response.body();
            //Preuzima kurs na osnovu model klase u kojoj se nalaze kodovi valuta
            for (int i=0;i<22;i++){
                kursevi.add(new Model(Valute.drzava[i],Valute.kod[i],
jsonObject.get("BTC").getAsJsonObject().get(Valute.kod[i]).getAsDouble(),
jsonObject.get("ETH").getAsJsonObject().get(Valute.kod[i]).getAsDouble()));
            }
            iniKurs.add(kursevi.get(0));
            rvAdapter = new RVAdapter(iniKurs);
            rv.setAdapter(rvAdapter);
        }

        @Override
        public void onFailure(Call<JsonObject> call, Throwable t) {
            pgDijalog.dismiss();
            setSnackBar("Internet konekcija je loša");
        }
    });
}

private void setSnackBar(String message){
    Snackbar snackbar = Snackbar.make(findViewById(R.id.LinearLayout), message,
Snackbar.LENGTH_INDEFINITE)
        .setAction("Osveži", new View.OnClickListener() {

```

```

        @Override
        public void onClick(View v) {
            pocetniPrikaz();
        }
    }); snackbar.show();

}

//Proverava da li je konekcija dostupna
public static boolean isKonekcijaDostupna(Context context) {
    ConnectivityManager cm = (ConnectivityManager)
context.getSystemService(Context.CONNECTIVITY_SERVICE);
    NetworkInfo activeNetworkInfo = cm.getActiveNetworkInfo();
    return activeNetworkInfo != null && activeNetworkInfo.isConnectedOrConnecting();
}

//Dodaje ActionBar
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}

//Poziva veb prikaz
@Override
public boolean onOptionsItemSelected(MenuItem item) {

    switch (item.getItemId()) {

        case R.id.stock:
            Intent intent = new Intent(this, WebPrikaz.class);
            startActivity(intent);
            return true;

        default:

```

```

        return super.onOptionsItemSelected(item);
    }

}

}

```

## 6.2. Delovi koda

Za preuzimanje JSON podataka koristi se retrofit, okhttp3 i gson.

```

public class Klijent {
    private static final String glavni_url = "https://min-api.cryptocompare.com/";
    Retrofit retrofit = null;
    public Retrofit getKlijent(){
        if (retrofit==null){
            HttpLoggingInterceptor interceptor = new HttpLoggingInterceptor();
            interceptor.setLevel(HttpLoggingInterceptor.Level.BODY);
            OkHttpClient klijent = new
OkHttpClient.Builder().addInterceptor(interceptor).build();
            retrofit = new Retrofit.Builder()
                .baseUrl(glavni_url)
                .addConverterFactory(GsonConverterFactory.create())
                .client(klijent)
                .build();
        }
        return retrofit;
    }
}

```

Klasa Valute sadrži nazive i skraćenice svih dostupnih valuta u aplikaciji.

```
public class Valute {  
  
    public static String[] drzava = {"Srbija", "Bosna i Hercegovina", "Nigerija", "Sjedinjene  
Američke Države", "Ujedinjeno Kraljevstvo",  
    "Gana", "Nemačka", "Kongo", "Kanada", "Avganistan", "Albanija", "Kolumbija", "Egipat", "Danska", "Južna  
Koreja", "Južna Afrika", "Saudijska Arabija", "Japan", "Argentina", "Brazil", "Kina", "Hong Kong"};  
  
    public static String[] kod =  
    {"RSD", "BAM", "NGN", "USD", "GBP", "GHS", "EUR", "XAF", "CAD", "AFN", "ALL", "COP", "EGP", "DKK", "KRW", "ZA  
R", "SAR", "JPY", "ARS", "BRL", "CNY", "HKD"};
```

U klasi RVAdapter je kod koji omogućava poziv konverzije klikom na karticu.

```
//Klik na karticu ; poziv konverzije  
  
    v.setOnClickListener(new View.OnClickListener() {  
  
        @Override  
  
        public void onClick(View view) {  
  
            int pozicija = getAdapterPosition();  
  
            if (pozicija!=RecyclerView.NO_POSITION){  
  
                Intent intent = new Intent(view.getContext(), Konverzija.class);  
  
                intent.putExtra("btc", nazivi.get(pozicija).getBtc())  
  
                    .putExtra("drzava", nazivi.get(pozicija).getDrzava())  
  
                    .putExtra("kod", nazivi.get(pozicija).getKod())  
  
                    .putExtra("eth", nazivi.get(pozicija).getEth());  
  
                view.getContext().startActivity(intent);  
  
            }  
  
        }  
  
    }  
  
}
```

Sledi deo klase za konverziju:

```
double btcUOsnova(double btcVrednost) {  
  
    return getIntent().getExtras().getDouble("btc") * btcVrednost;  
  
}  
  
  
double ethUOsnova(double ethVrednost) {
```

```
        return getIntent().getExtras().getDouble("eth") * ethVrednost;
    }

    double osnovaUBtc(double osnovnaValutaVrednost) {
        return osnovnaValutaVrednost / getIntent().getExtras().getDouble("btc");
    }

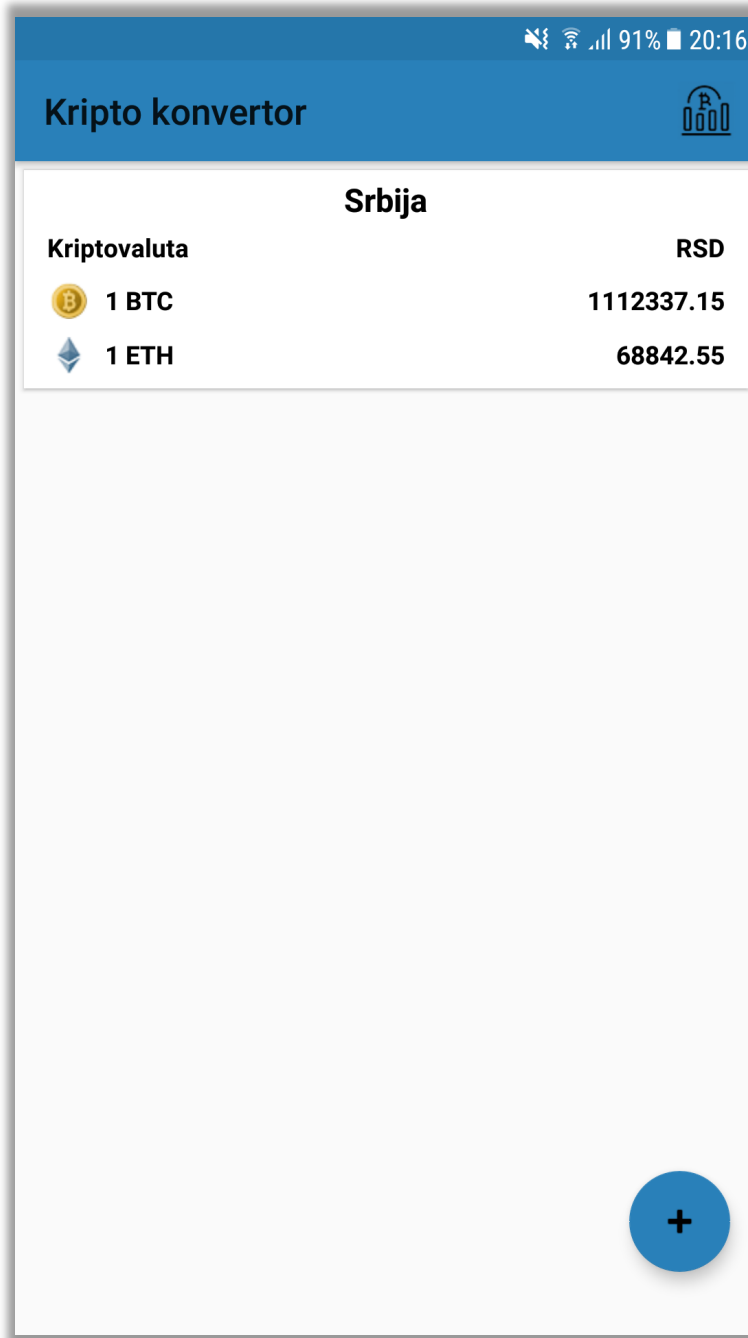
    double osnovaUEth(double osnovnaValutaVrednost) {
        return osnovnaValutaVrednost / getIntent().getExtras().getDouble("eth");
    }

    double ethUBtc(double ethVrednost) {
        return ethVrednost * getIntent().getExtras().getDouble("eth") /
        getIntent().getExtras().getDouble("btc");
    }

    double btcUEth(double btcVrednost) {
        return btcVrednost * getIntent().getExtras().getDouble("btc") /
        getIntent().getExtras().getDouble("eth");
    }
}
```

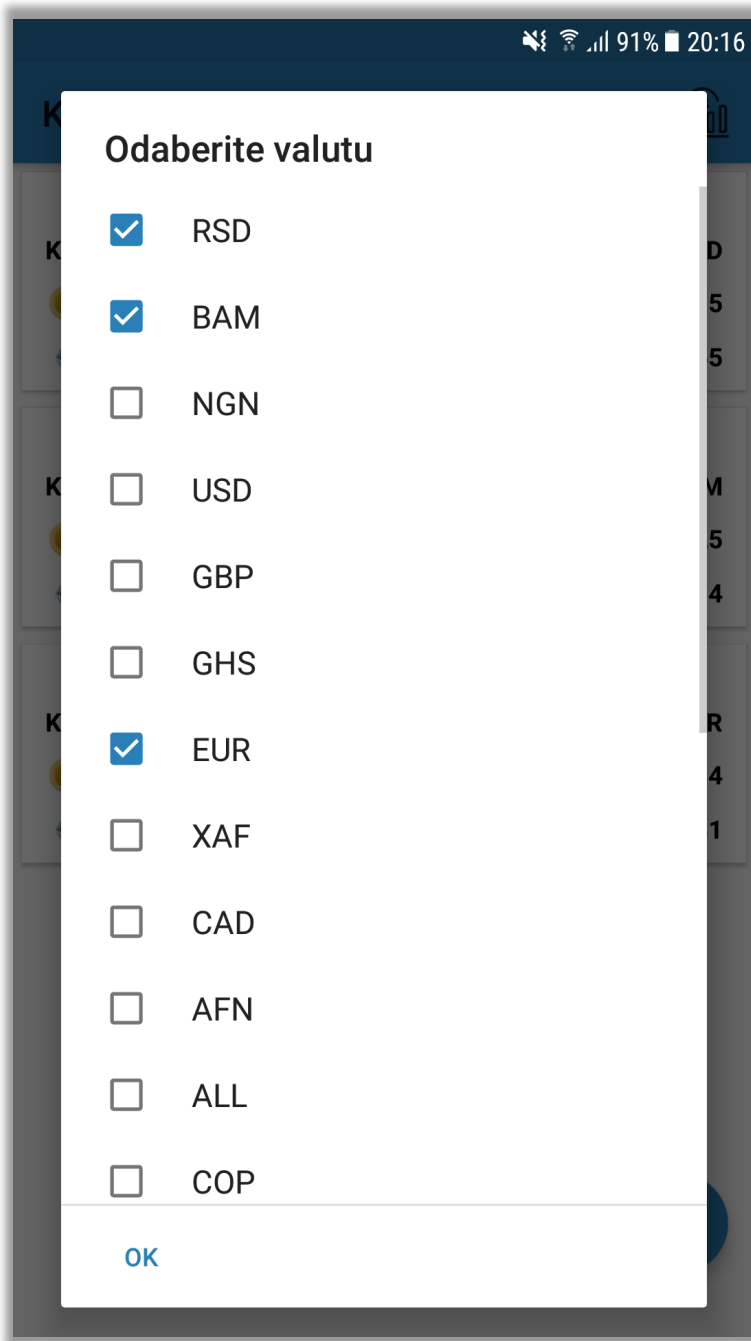
## 7. Prikaz aplikacije

Početna stranica učitava prvu karticu za konverziju. (Slika 3.)



Slika 3. Početna strana

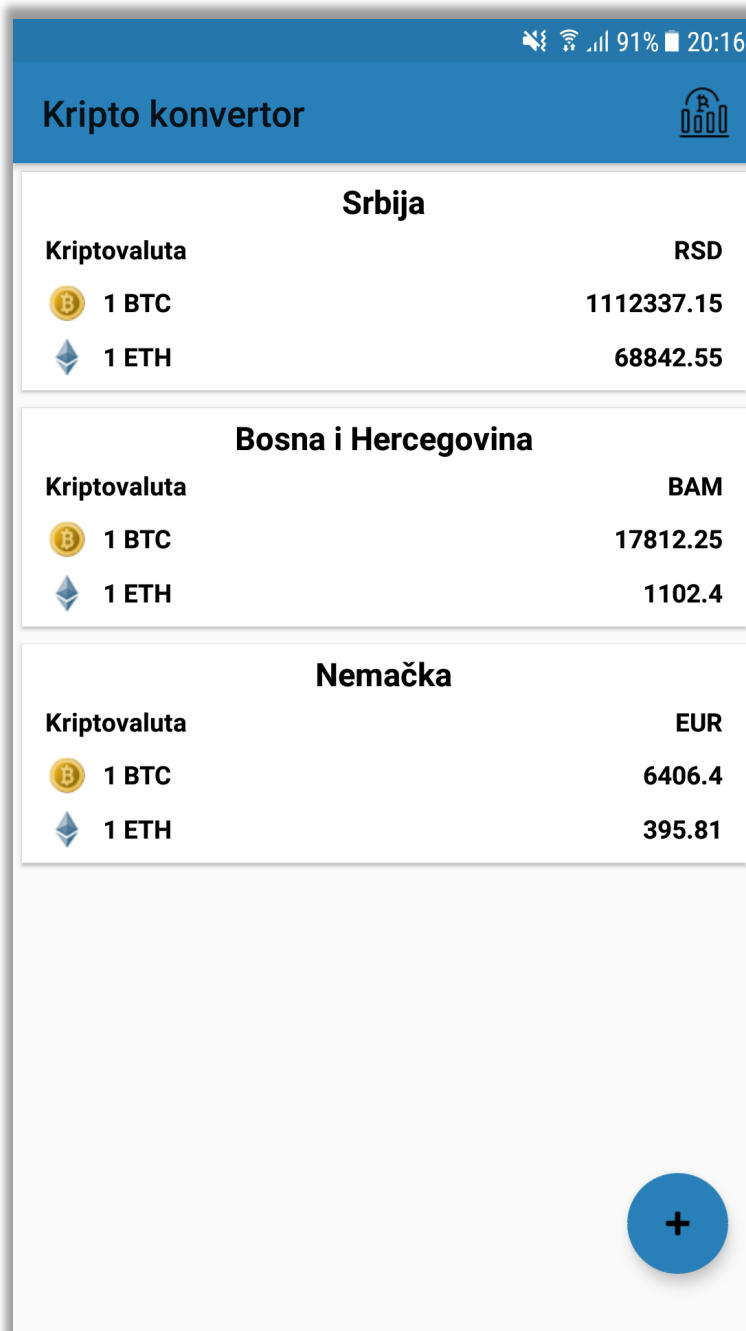
Klikom na dugme plus prikazuje se lista sa svim dostupnim valutama za konverziju, čekiranjem željene valute kartica se dodaje na početni ekran. (Slika 4.)



Slika 4. Lista valuta



Na sledećoj slici se vide dodate kartice. (Slika 5.)



Slika 5. Dodate kartice

Klikom na neku od kartica otvara se nova stranica za konverziju odabrane valute. (Slika 6.)

The screenshot shows a mobile application interface for currency conversion. At the top, there is a blue header bar with a back arrow and the title "Konverzija". Below the header, the location "Srbija" is displayed. The main content area is divided into two sections. The first section, titled "Kriptovaluta" and "RSD", shows conversion rates for 1 BTC (1112337.15 RSD) and 1 ETH (68842.55 RSD). The second section contains three input fields: "RSD" with the value "15000", "BTC" with the value "0.0135", and "ETH" with the value "0.2179". At the bottom of the screen is a numeric keypad with buttons for digits 1-9, 0, a backspace button, a "Next" button, and a settings gear icon.

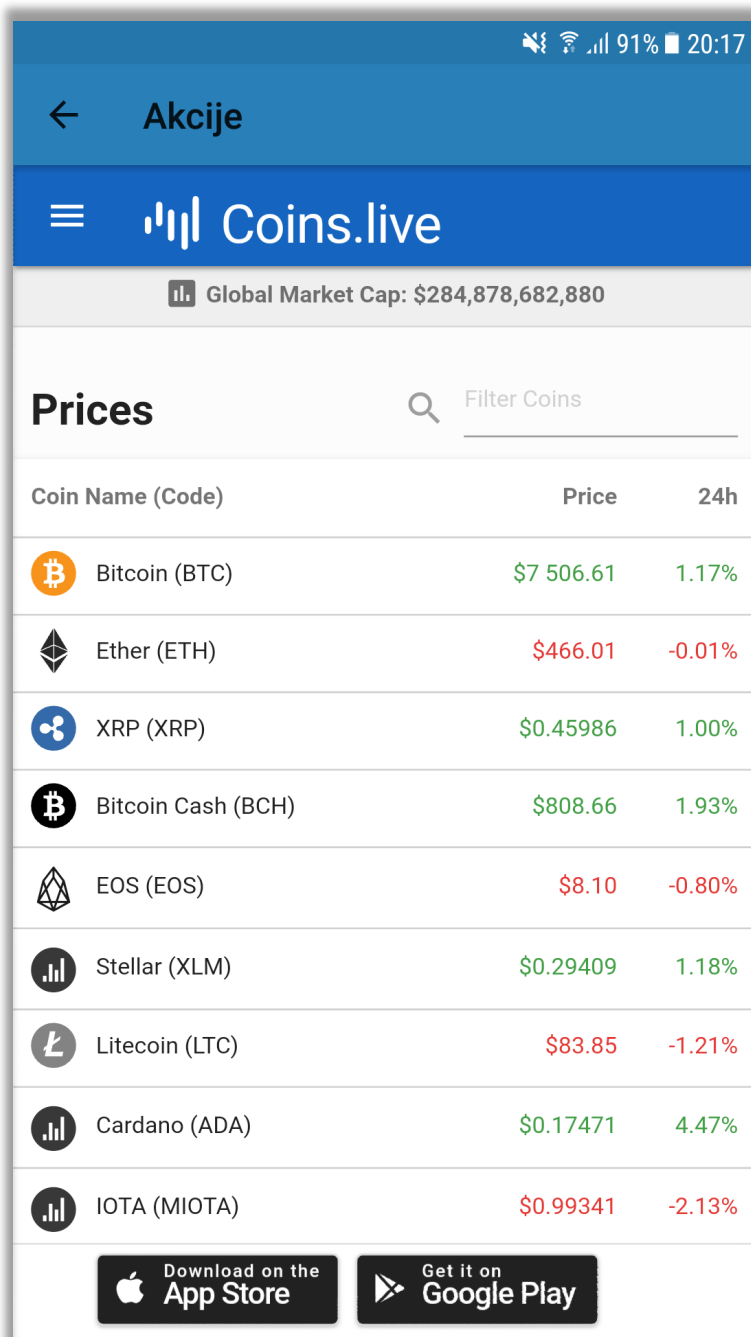
Kriptovaluta	RSD
1 BTC	1112337.15
1 ETH	68842.55










RSD	15000
BTC	0.0135
ETH	0.2179

Slika 6. Stranica za konverziju

Klikom na dugme Bitcoina u ActionBar-u otvara se WebView sa stranicom koja prikazuje trenutne kurseve. (Slika 6.)



The screenshot shows a mobile application interface for 'Coins.live'. At the top, there's a blue header with a back arrow and the word 'Akcije'. Below it, a blue bar contains the 'Coins.live' logo. A grey bar below that displays 'Global Market Cap: \$284,878,682,880'. The main section is titled 'Prices' and includes a search bar labeled 'Filter Coins'. Below this is a table listing various cryptocurrencies with their icons, names, current prices, and 24-hour percentage changes. At the bottom, there are two buttons: 'Download on the App Store' and 'Get it on Google Play'.

Coin Name (Code)	Price	24h
 Bitcoin (BTC)	\$7 506.61	1.17%
 Ether (ETH)	\$466.01	-0.01%
 XRP (XRP)	\$0.45986	1.00%
 Bitcoin Cash (BCH)	\$808.66	1.93%
 EOS (EOS)	\$8.10	-0.80%
 Stellar (XLM)	\$0.29409	1.18%
 Litecoin (LTC)	\$83.85	-1.21%
 Cardano (ADA)	\$0.17471	4.47%
 IOTA (MIOTA)	\$0.99341	-2.13%

Slika 7. WebView