

# **Sistem za konvertovanje**

## **Predmet: Klijent Server Sistemi**

**Profesor:**

**dr Mirko Kosanović**

**Miloš Kosanović**

**Student:**

**Strahinja Milošević**

**PEp 31/17**

**Septembar 2020.**

## SADRŽAJ

|  |       |
|--|-------|
| 1. Uvod.....   | - 3 - |
| 2. Instalacija i podešavanje projekta .....          | - 3 - |
| 2.1. Instaliranje modula .....                       | - 3 - |
| 3. Arhitektura aplikacije .....                      | - 4 - |
| 3.1 Serverski deo.....                               | - 4 - |
| 3.2 Klijentski deo.....                              | - 4 - |
| 3.3 Baza podataka .....                              | - 5 - |
| 3.4 Komunikacija.....                                | - 5 - |
| 4. Rad aplikacije .....                              | - 5 - |
| 4.1 Opis implementacije .....                        | - 8 - |
| 4.2 Opis funkcionalnosti – korisničko uputstvo ..... | - 9 - |
| 5. Literatura.....                                   | - 9 - |

## 1. Uvod

U ovom projektu obrađena je izrada veb aplikacije koja će služiti kao sistem za konverziju. Tehnologije koje su korišćene na klijentskoj strani su HTML5, CSS sa Bootstrap framework-om, JavaScript, fontovi iz FontAwesome grupacije, dok je na serverskoj strani korišćen NodeJS sa Express framework-om. U okviru moje aplikacije koristim i EJS templejtski jezik koji je sadržan u okviru Express View Engin-a. Celokupan projekat odrađen je u IDEA programskom okruženju za razvoj softwera. Pretraživač koji koristim za testiranje projekta je Google Chrome.

Aplikacija se sastoji iz početne strane gde možemo da odaberemo tip konvertora koji želimo. U navigacionom meniju imamo konvertor valute, konvertor dužine i konvertor snage. Svaki od konvertora se sastoji od dva polja, s tim što je jedno polje predviđeno za unos vrednosti, a drugo za prikazivanje rezultata, tek kada kliknemo na dugme predviđeno za izvršenje. Svaki od konvertora može da vrši i obrnutu konverziju, tako što ćemo ići na dugme predviđeno za to. Aplikacija je potpuno primenljiva i upotrebljiva za bilo koju veličinu ekrana.

## 2. Instalacija i podešavanje projekta

Da bismo pokrenuli samu aplikaciju potrebni su nam pored samog NODEJS-a i WAMP server koga koristimo kao deo koji nam omogućava korišćenje baze podataka odnosno MYSQL servis kao i grafički dizajn samog okruženja za kreiranje baze podataka – PHPMyAdmin. Zatim moramo instalirati IDEA programsko okruženje za razvoj softwera u kome ćemo uz pomoć brojnih NPM modula kreirati našu aplikaciju.

### 2.1. Instaliranje modula

U ovom projektu nalazi se **package.json** fajl, to je fajl koji se inicijalizuje sa projektom, naravno na zahtev programera, i ukoliko popunimo sve informacije ispravno biće kreiran fajl. U koliko je sve to ispravno kreirano, da bi smo na nekom drugom računaru pokrenuli i instalirali module, potrebno je ukucati sledeću naredbu **npm install** ova naredba prvo pretražuje **package.json** fajl i u njemu traži i instalira sve dependence (tj. Module ili biblioteke) koji su potrebni za ovaj projekat. Kada koristimo razvojno okruženje, u ovom slučaju IDEA, on će ove foldere kreirati automatski prilikom kreiranja projekta. Tu će se nalaziti svi osnovni moduli koji dolaze u folderu node-moduls. Za sve druge module koji će nam kasnije biti potrebni uvozićemo ih preko terminala komandama `npm install [ime-samog-modula]` i tome slično.

### 3. Arhitektura aplikacije

Ova aplikacija sledeću arhitekturu. Na vrhu je sam folder sa imenom aplikacije koji se kreira prilikom kreiranja projekta. U njemu se automatski smestaju folder .idea u kome se nalaze fajlovi koji su potrebni samom programskom okruženju za rad, zatim folder bin u kome se nalazi java script fajl www, zatim imamo folder node-modules u kome su smešteni svi neophodni node moduli za rad naše aplikacije, zatim imamo public folder u kome su smešteni redom folderi images, javascript, stylesheet i drugi folderi koje možemo dodati ručno. Oni nam služe za smeštanje slika koje koristimo na klijentskom delu, za smeštanje nekog java scripta koji se može pozivati kroz aplikaciju, ovde je takođe smešten i CSS u okviru foldera stylesheet koji nam služi za realizaciju izgleda. Sledeći bitan folder je folder routes u kome su smešteni serverski delovi naše aplikacije odnosno fajlovi sa ekstenzijom .js. Ovde pišemo pozadinu naše aplikacije. Sledeći folder je folder views koji nam služi za smeštanje svih HTML stranica koji će se koristiti kroz našu aplikaciju. Ovi fajlovi imaju ekstenziju .EJS zbog toga što ovo nisu samo obični HTML fajlovi, već oni sadrže templejski jezik EJS za ubacivanje nekih podataka na klijentskoj strani odnosno na strani samog HTML-a. Ovi podaci mogu biti recimo rezultati upita baze podataka. Ove podatke dobijamo od serverskog dela, odnosno od onih stranica koje se nalaze u routes folderu i imaju ekstenziju .js. Sledeći fajl je app.js fajl koji nam služi kao neka konfiguracija same aplikacije. Ovde dodajemo neke putanje, odnosno rute i koristimo module. Ovde takođe povezujemo .js fajlove iz foldera routes sa putanjama koje koristimo kao url. Sledeći bitan folder je package-json folder u kome se smešta JSON dokument koji opisuje samu aplikaciju i sadrži spisak modula od kojih je serverski deo aplikacije zavistan.

#### 3.1 Serverski deo

Na serverskom delu imamo NODE JS tehnologiju u kombinaciji sa expressom trenutno najpopularnijim frameworkom za Node JS. Node js je izuzetno brz programski jezik i može da radi sa velikom grupom podataka. Još jedna bitna stavka ovog programskog jezika je ta što podržava neblokirajuću arhitekturu. Kao takav jezik izuzetno je pogodan za izradu real time aplikacija poput čatovanja, lutrije i drugih tome sličnih aplikacija.

#### 3.2 Klijentski deo

Na klijentskoj strani imamo skup tehnologija kao što su HTML kojim pravimo strukturu same stranice uz koji koristimo EJS templejski jezik za prikazivanje podataka sa servera, imamo CSS za stilizovanje elemenata koji se prikazuju, imamo CSS-ovu biblioteku BOOTSTRAP koju

takođe koristimo u velikoj meri. Java Script jezik kojim radimo određene funkcionalnosti na stranici, a za fontove koristimo FontAwesome grupaciju.

### **3.3 Baza podataka**

Baza je sastavni deo svakog ozbiljnijeg projekta i kao takva igra izuzetno važnu ulogu. Treba dobro osmisлити koncept baze podataka kao i vrstu iste pre izrade samog projekta. U ovom projektu korišćena je MySQL relacionalna baza podataka. Kao podrška bazi korišćen je WAMPserver. Zbog svoje moći i dobro konstruisanoj arhitekturi ovo je izuzetno korišćena tehnologija za izradu baza podataka.

### **3.4 Komunikacija**

U ovom delu svaka ruta biće ukratko objašnjena.

| <b>RUTE KOJE SE KORISTE</b> | <b>KRATAK OPIS SVAKE OD RUTA</b>  |
|-----------------------------|---|
| GET /                       | Ovom rutom se inicira početna strana  |
| GET /valute                 | Ovom rutom se inicira strana za konvertor valute  |
| POST /valute                | Ovom rutom se prosleđuju parametri za računanje i vraća rezultat na strani konvertora za valutu |
| GET /obrisivalutu/:id       | Ovom rutom u bazi pronalazimo vrednost sa datim id-jem i njegove podatke brisemo                |
| POST /valuteunos            | Ovom rutom upisujemo izračunate vrednosti u bazu podataka u odgovarajuću tabelu                 |
| GET /valuteobrnuti          | Ovom ruto otvaramo stranicu koja je predviđena za konvertor vuleteobrnuti                       |
| POST /valuteobrnuti         | Ovom rutom se prosleđuju parametri i ona vraća rezultat   |
| POST /valuteunosobrnuti     | Ovom rutom upisujemo dobijene vrednosti u bazu  |
| GET /obrsiobrnutuvalu/:id   | Ovom rutom pronalazimo odgovarajući id i brisemo njegove podatke                                |
| GET/duzina                  | Ovom rutom iniciramo konvertor dužine   |
| POST/duzinaunos             | Ovom rutom u bazu upisujemo izračunate vrednosti  |
| GET/obrisiduzinu/:id        | Ovom rutom u bazi pronalazimo vrednost sa datim   |

|                           |  |
|---------------------------|--|
|                           | id-jem i njegove podatke brisemo   |
| GET/duzinaobrnuti         | Ovom rutom se inicira strana za duzinaobrnuti konvertor  |
| POST/duzinaobrnuti        | Ovom rutom se prosleđuju parametri za računanje  |
| POST/duzinaunosobrnuti    | Ovom rutom upisujemo dobijene vrednosti u bazu podataka  |
| GET/obrisiunetuduzinu/:id | Ovom rutom pronalazimo odgovarajući id i brišemo njegove podatke                               |
| GET/snaga                 | Ovom rutom se inicira strana za konvertor snage  |
| POST/snaga                | Ovom rutom se prosleđuju parametri za računanje i vraća rezultat na strani konvertora za snagu |
| POST/snagaunos            | Ovom rutom u bazu podataka upisujemo odgovarajuće vrednosti                                    |
| GET/obrisisnagu/:id       | Ovom rutom u bazi pronalazimo vrednost sa datim id-jem i njegove podatke brisemo               |
| GET/snagaobrnuti          | Ovom rutom se inicira strana za snagaobrnuti konvertor   |
| POST/snagaobrnuti         | Ovom rutom se prosleđuju parametri za računanje  |
| POST/snagaunosobrnuti     | Ovom rutom upisujemo izračunate vrednosti u bazu podataka u odgovarajuću tabelu                |

|                          |  |
|--------------------------|--|
| GET/obrisiunetusnagu/:id | Ovom rutom pronalazimo odgovarajući id i brisemo njegove podatke |
|--------------------------|--|

## 4. Rad aplikacije

### 4.1 Opis implementacije

```
var express = require('express');  
var router = express.Router();  
var mysql=require('mysql');  
var bodyParser = require('body-parser');  
  
router.use(bodyParser.json());  
router.use(bodyParser.urlencoded({ extended: true }));
```

Na početku svakog projekta kao što smo već napomenuli moramo uključiti sledeće module. Neki od modula su već instalirani u node-modules dok druge trebamo instalirati.

Ukratko rečeno, Express je trenutno najpopularniji framework za Nodejs. Isto kao i Nodejs, koristi se na strani web server-a i široko se primenjuje zbog svoje brzine i minimalizma u kodu.

Pomoću naredbe **require()**, vrši se uključivanje modula u trenutni fajl. Kao parametar se prosleđuje putanja do fajla ili ime modula.

Rutiranje omogućavamo pomoću naredbe `express.Router()`. Da bismo mogli kroz url da pokrećemo odgovarajuće stranice i prikazujemo odgovarajuće elemente i izvršavamo događaje.

Pomoću sledeće komande `require('mysql')` uvozimo modul koji nam služi za rad sa bazom podataka. U ovom slučaju koristimo MySQL bazu podataka. Konekcija na bazu se izvršava sledećim kodom:



```
const con = mysql.createConnection( config: {  
  host: 'localhost',  
  user: 'root',  
  port: 3308,  
  database: 'konvertor'  
});
```

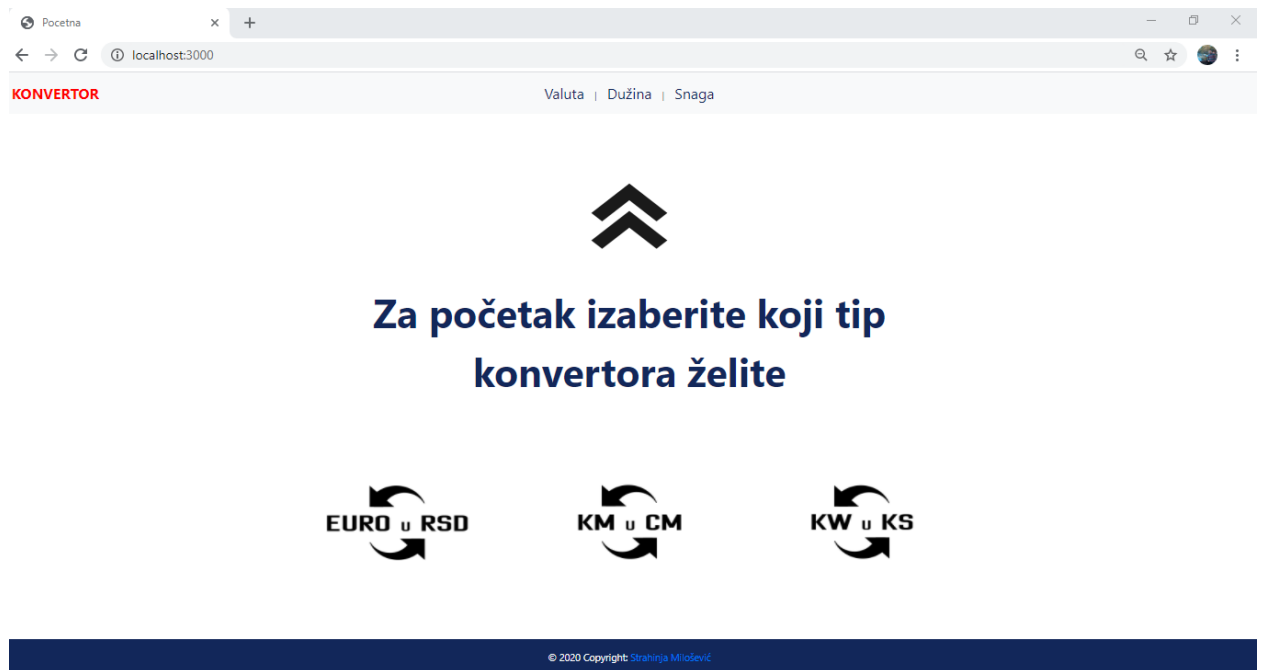
Primer upita sa servera za bazu podataka izgleda ovako:

```
let sql = "SELECT * FROM valute";  
//SALJEM UPIT NA IZVRSENJE  
con.query(sql, function (err, result) {  
  console.log(result);  
  var uneta="";  
  var dobijena="";  
  console.log(uneta, dobijena);  
  res.render( {view: 'KonverterTip1.ejs', options: {title:"Valute", dobijena, uneta, data: result}});  
});
```

Pomoću `require('body-parser')` vršimo parsiranje POST zahteva i dodajemo prosledene vrednosti u `body` atribut request objekta .

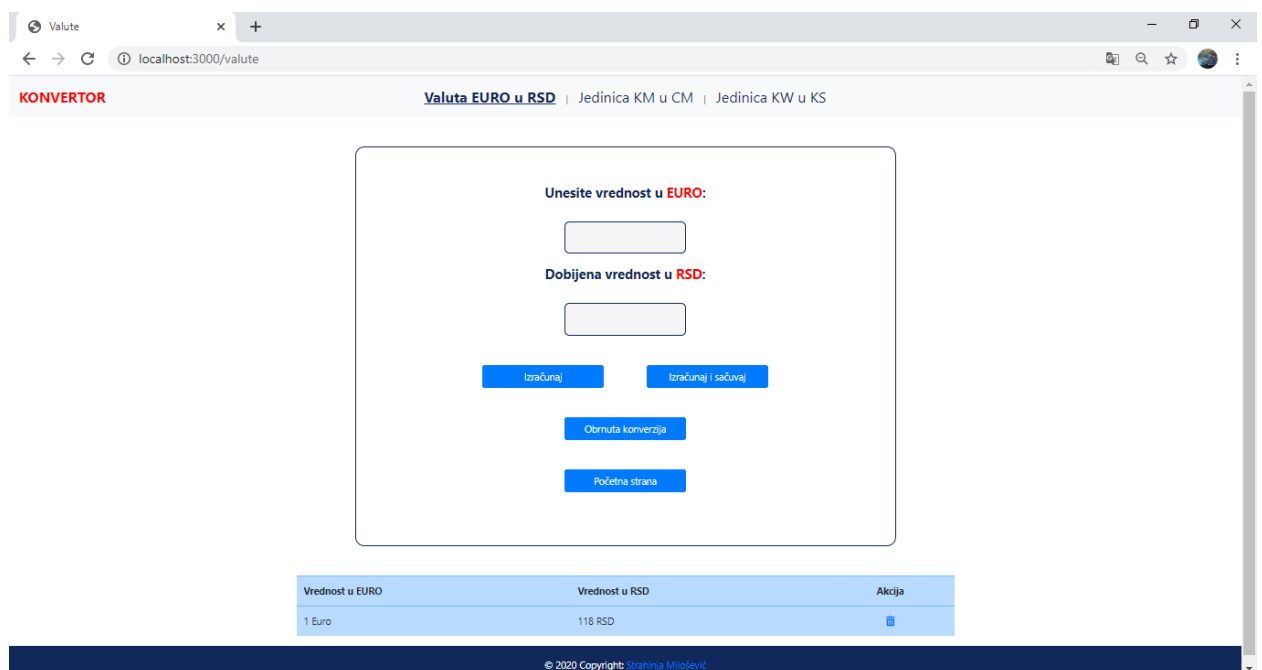
## 4.2 Opis funkcionalnosti – korisničko uputstvo

Ovaj projekat se pokreće preko IDEA radnog okruženja time što u terminalu ukucamo komandu **nodemon start** (Ukratko: Nodemon je korisna alatka koja nam služi kao pomoć pri kodiranju same aplikacije. Ona za nas prati promene u radu i čim se desi komanda save odmah restartuje server. Time se sve novonastale promene prikazu na ekran.). Kada smo pokrenuli server, potrebno je da pokrenemo i wamp aplikaciju koja će nam omogućiti rad sa bazom povezanom sa datim projektom. Kada je sve to startovano pokrećemo Google Chrome i kucamo u url-u `localhost:3000`. Ova komanda će nas automatski odvesti na prvu stranicu projekta a to je (slika ispod).



Nakon odabira konvertora, u zavisnosti od odabira, prikazuju se sledeće stranice.

Konvertor Valute



Konvertor Dužine

The screenshot shows a web browser window with the URL `localhost:3000/duzina`. The page header includes the logo **KONVERTOR** and navigation links: `Valuta EURO u RSD`, `Jedinica KM u CM` (active), and `Jedinica KW u KS`. The main content area contains a form with two input fields: **Unesite vrednost u KM:** and **Dobijena vrednost u CM:**. Below the inputs are four buttons: `Izračunaj`, `Izračunaj i sačuvaj`, `Obrnuta konverzija`, and `Početna strana`. At the bottom, a table displays the conversion result:

| Vrednost u KM | Vrednost u CM | Akcija |
|---------------|---------------|--------|
| 1 KM          | 100000 CM     |        |

The footer shows the copyright notice: `© 2020 Copyright: Strahinja Milošević`.

## Konvertor Snage

The screenshot shows a web browser window with the URL `localhost:3000/snaga`. The page header includes the logo **KONVERTOR** and navigation links: `Valuta EURO u RSD`, `Jedinica KM u CM`, and `Jedinica KW u KS` (active). The main content area contains a form with two input fields: **Unesite vrednost u KW:** and **Dobijena vrednost u KS:**. Below the inputs are four buttons: `Izračunaj`, `Izračunaj i sačuvaj`, `Obrnuta konverzija`, and `Početna strana`. At the bottom, a table displays the conversion result:

| Vrednost u KW | Vrednost u KS | Akcija |
|---------------|---------------|--------|
| 1 KW          | 1,34 KS       |        |

The footer shows the copyright notice: `© 2020 Copyright: Strahinja Milošević`.

Kao što možemo videti postoji polje u kojem unesemo željenu vrednost i preko dugmeta 'Izračunaj' dobijemo sračunatu vrednost u polje 'Dobijena vrednost'.

Valute x +

localhost:3000/valute

**KONVERTOR** Valuta **EURO u RSD** | Jedinica KM u CM | Jedinica KW u KS

Unesite vrednost u **EURO**:

20

Dobijena vrednost u **RSD**:

2360

Izračunaj Izračunaj i sačuvaj

Obrnuta konverzija

Početna strana

| Vrednost u EURO | Vrednost u RSD | Akcija |
|-----------------|----------------|--------|
| 1 Euro          | 118 RSD        |        |

U koliko želimo da izračunamo i sačuvamo vrednost u bazu, onda kliknemo na dugme 'Izračunaj i sačuvaj'.

| Vrednost u EURO | Vrednost u RSD | Akcija |
|-----------------|----------------|--------|
| 1 Euro          | 118 RSD        |        |
| 20 Euro         | 2360 RSD       |        |

Ako želimo da izvršimo konverziju obrnuto, recimo RSD da pretvorimo u EVRO onda kliknemo na dugme 'Obrnuta konverzija' i to izgleda ovako.

Valute x +

localhost:3000/valuteobrnuti

**KONVERTOR** Valuta **RSD u EURO** | Jedinica CM u KM | Jedinica KS u KW

Unesite vrednost u **RSD**:

40569

Dobijena vrednost u **EURO**:

344.03

Izračunaj Izračunaj i sačuvaj

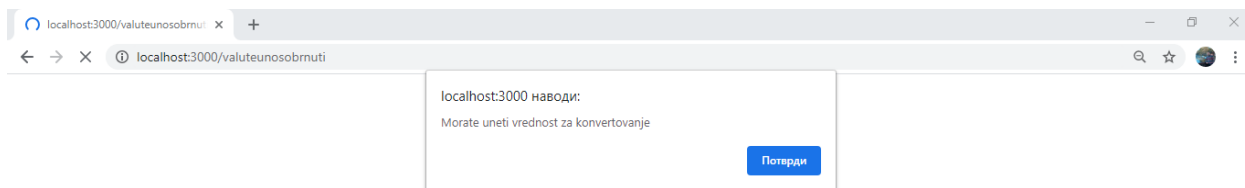
Obrnuta konverzija

Početna strana



| Vrednost u RSD | Vrednost u EURO | Akcija |
|----------------|-----------------|--------|
| 118 RSD        | 1.00 EURO       |        |

© 2020 Copyright: Strahinja Milošević

Ako smo zaboravili da unesemo vrednost, a pritom idemo na dugme 'Izračunaj i sačuvaj' izbacice poruku sa sledećim tekstom.



Možemo i da obrišemo vrednost koju smo dobili u bazi, a to ćemo ostvariti tako što ćemo ići na dugme koje se nalazi ispod Akcije.

| Vrednost u EURO | Vrednost u RSD | Akcija  |
|-----------------|----------------|---|
| 1 Euro          | 118 RSD        |  |
| 200 Euro        | 23600 RSD      |  |
| 4 Euro          | 472 RSD        |  |

## 5. Literatura

Praktikum Klijent server sistemi, M.Kosanović, interna skripta laboratorijske vežbe  
[https://vtsnis.edu.rs/wp-content/plugins/vts-predmeti/uploads/2019\\_Praktikum\\_V1.pdf](https://vtsnis.edu.rs/wp-content/plugins/vts-predmeti/uploads/2019_Praktikum_V1.pdf)  
<http://expressjs.com/>  
<https://nodejs.org/api/>  
<https://www.w3schools.com/nodejs/>  
<https://docs.npmjs.com/>  
[https://www.w3schools.com/nodejs/nodejs\\_mysql.asp](https://www.w3schools.com/nodejs/nodejs_mysql.asp)