

**SEMINARSKI RAD**  
**FAKULTETSKA BIBLIOTEKA**

**Sadržaj:**

- 1. Uvod – opis rada fakultetske biblioteke**
- 2. Specifikacija funkcija softvera**
- 3. Faze razvoja softvera**
  - 3.1. Projektne ideje**
- 4. Dizajn softvera**
  - 4.1. Sloj podataka**
  - 4.2. Srednji aplikacioni sloj**
  - 4.3. Prezentacioni sloj (korisnički interfejs)**
- 5. Prikaz softvera**
- 6. Realizacija i implementacija**
  - 6.1. Prikaz karakterističnih elemenata realizacije**
  - 6.2. Rad sa Github sistemom**
  - 6.3. Testiranje softvera**
- 7. Korišteni alati i softveri**
- 8. Literatura**

## 1. UVOD – OPIS RADA FAKULTETSKE BIBLIOTEKE

Fakultetska biblioteka izdaje bibliotečke jedinice na korišćenje članovima biblioteke. Osnovna delatnost biblioteke je izdavanje i evidentiranje zaduživanja i vraćanja bibliotečkih jedinica, obrada korisnika biblioteke – ućlanjivanje, ispis, praćenje zaduživanja.

Bibliotečka jedinica može biti: završni rad (diplomski, specijalistički, magistarski, master, doktorat), knjiga, udžbenik, zbirka zadataka, časopis, zbornik radova, monografija, CD, DVD, USB kartica. Završni rad stiže u biblioteku iz studentske službe nakon predaje od strane studenta. Knjige, časopisi i udžbenici se nabavljaju periodično. Svaki nastavnik dostavlja spisak knjiga koji se prosleđuje dekanatu koji radi poručivanje, plaćanje i preuzimanje knjiga. Sve bibliotečke jedinice se evidentiranje i dobijaju karton i inventarni broj, te se odlažu se na odgovarajuću policu.

Članovi biblioteke su zaposleni na fakultetu – nastavnici i nenastavno osoblje, kao i studenti svih nivoa studija. Zaposleni fakulteta nakon što budu primljeni na posao, treba da se jave bibliotekaru kako bi im se otvorio karton člana. Zaposleni imaju pravo da koriste knjige iz bibliotečkog fonda u skladu sa pravilnikom o radu biblioteke, što znači da vreme koje im je na raspolaganju povodom iznajmljivanja knjige je duže nego studentima. Nakon prestanka radnog odnosa, zaposleni mora da vrati sve knjige i tada prestaje da bude korisnik biblioteke. Sudenti nakon upisa dobijaju indeks. Sa indeksom student dolazi u biblioteku i javlja se bibliotekaru kako bi mu se otvorio karton člana. Studenti imaju pravo da koriste knjige iz bibliotečkog fonda u skladu sa pravilnikom o radu biblioteke. Ukoliko student ne vrati knjigu na vreme, dobija opomenu. Sve knjige mora vratiti prilikom diplomiranja, kada mu se izdaje potvrda o razduženosti.

Opšti postupak zaduživanja i razduživanja za sve članove: Korisnik na osnovu evidencija o fondu ili uvida u stanje na policama, bira određene bibliotečke jedinice. Potpisuje se na kartonima tih jedinica, sa datumom uzimanja i ti kartoni se ulažu u karton člana. Po povratku bibliotečke jedinice, upisuje se datum vraćanja i karton se vraća u bibliotečku jedinicu i ona se odlaže na police. Ukoliko korisnik vrati oštećenu bibliotečku jedinicu, potrebno je u dogovoru sa dekanatom da nadoknadi štetu.

Bibliotečka jedinica se može rashodovati ili otpisati u slučaju da je izgubljena. Broj zaduženih knjiga ne može biti veći od 2 za nenastavno osoblje, 3 za studente odnosno 5 za nastavnike. Korisnik ne može da zadužuje nove naslove, ako prethodne nije vratio. Period izdavanja je 3 meseca za studente i 6 meseci za zaposlene.

## **2. SPECIFIKACIJA FUNKCIJA SOFTVERA**

### **FAKULTETSKA BIBLIOTEKA - SPECIFIKACIJA FUNKCIONALNOSTI SOFTVERA**

#### ***Zahtev 1. Evidentiranje novih bibliotečkih jedinica***

*Naziv* Evidentiranje novih bibliotečkih jedinica

*Oznaka* ZAHTEV01

*Opis* Unos: naziva, autora, izdavač, godina izdavanja, vrsta jedinice, jezika, ISBN, inventarni broj, stanje, opis, rashodovana, otpisana, datum otpisa.

Prikaz poruke o uspešnosti izvršenog unosa.

*ID* 1.

*Prioritet implementacije* 1

*Status* Skica

*Implemetacioni rizik* Nizak

#### ***Zahtev 2. Pretraga bibliotečkog fonda***

*Naziv* Pretraga bibliotečkog fonda

*Oznaka* ZAHTEV02

*Opis* Pretraga po: autoru, nazivu, godini, jeziku, vrsta jedinice.

Prikaz pronadjenih jedinica - tabelarno.

Mogućnost sortiranja pronađenih podataka.

Prikaz poruka u slučaju da kriterijum pretrage nije zadovoljen.

*ID* 2.

*Prioritet implementacije* 2

*Status* Skica

*Implemetacioni rizik* Nizak

#### ***Zahtev 3. Izmena bibliotečkih jedinica***

*Naziv* Izmena bibliotečkih jedinica

*Oznaka* ZAHTEV03

*Opis* Pokretanje izmene sa stranice za pretragu.

Prikaz podataka o jedinici koja se menja.

Mogućnost promene prikazanih podataka.

Mogućnost odustanka od izmene.

Prikaz poruka o uspešnosti izvršene izmene.

*ID* 3.

*Prioritet implementacije* 1

*Status* Skica

*Implemetacioni rizik* Srednji

**Zahtev 4. Brisanje bibliotečkih jedinica**

Naziv Brisanje bibliotečkih jedinica

Oznaka ZAHTEV04

Opis Koja bibliotečke jedinice brisati?

Samo one (nove) koje nisu bili iznajmljivane nijednom!

Pokretanje brisanja je sa stranice za pretragu.

Prikaz podataka o jedinici koja se brise.

Potvrda brisanja.

Brisanje po inventarnom broju!!!

Prikaz poruka o uspešnosti izvršenog brisanja.

ID 4.

Prioritet implementacije 1

Status Skica

Implemetacioni rizik Srednji

**Zahtev 5. Izdavanje bibliotečkih jedinica**

Naziv Izdavanje bibliotečkih jedinica

Oznaka ZAHTEV05

Opis Relevantni podaci: Datum iznajmljivanja, inventarni broj, broj člana, period iznajmljivanja.

Mogućnost pretrage fonda, članova.

Zabrana izdavanja jedinica članovima koji imaju opomene.

Prikaz poruka o uspešnosti izvršenog izdavanja bibliotečkih jedinica.

ID 5.

Prioritet implementacije 2

Status Skica

Implemetacioni rizik Nizak

**Zahtev 6. Vraćanje bibliotečkih jedinica**

Naziv Vraćanje bibliotečkih jedinica

Oznaka ZAHTEV06

Opis Pronalazenje zaduženja.

Relevantni podaci: Datum vraćanja, inventarni broj, broj člana, ocena stanja jedinice.

Omogućiti vraćanje samo jednog dela zaduženja.

Prikaz poruke o uspešnosti izvršenog vraćanja.

ID 6.

Prioritet implementacije 2

Status Skica

Implemetacioni rizik Nizak

**Zahtev 7. Prijava korisnika**

Naziv Prijava korisnika

Oznaka ZAHTEV07

Opis Prijava/unos korisnickog imena (ili email adrese) i sifre korisnika.

Prikaz poruka o uspešnosti prijave.

ID 7.

Prioritet implementacije 3

Status Skica

Implemetacioni rizik Srednji

**Zahtev 8. Upis članova**

Naziv Upis članova

Oznaka ZAHTEV08

Opis Bibliotekar evidentira člana (student, nastavnik, asistent, nenastavno osoblje).

Podaci: broj člana, ime, prezime, mail, telefon mobilni, broj indeksa(student), sifra zaposlenog(ostali).

Prikaz poruka o uspešnosti upisa člana.

ID 8.

Prioritet implementacije 1

Status Skica

Implemetacioni rizik Srednji

**Zahtev 9. Ispis članova**

Naziv Ispis članova

Oznaka ZAHTEV09

Opis Pretraga članova i opomena.

Provera zaduzenosti člana.

Podaci: Datum ispisa, broj člana.

Prikaz poruke o uspešnosti ispisa člana.

ID 9.

Prioritet implementacije 3

Status Skica

Implemetacioni rizik Visok

**Zahtev 10. Izdavanje opomena**

Naziv Izdavanje opomena

Oznaka ZAHTEV10

Opis Automatsko generisanje poruka (SMS) koje se salju na mobilni telefon članu sledeceg dana nakon sto istekne period zaduzenja (studenti 2 meseca, zaposleni 3 meseca).

Izvestaj bibliotekaru u formi liste opomena koje su poslate nakon prijave.

ID 10.

Prioritet implementacije 3

Status Skica

Implemetacioni rizik Low

**Zahtev 11. Pretraga zaduženja članova**

Naziv Pretraga zaduzenja članova

Oznaka ZAHTEV11

Opis Pretraga po: broju indeksa, prezimenu, imenu.

Prikaz pronadjenih jedinica - tabelarno.

Mogućnost sortiranja pronađenih podataka.

Prikaz poruka u slučaju da kriterijum pretrage nije zadovoljen.

ID 11.

Prioritet implementacije 2

Status Skica

Implemetacioni rizik Nizak

**Zahtev 12. Odjava korisnika**

Naziv Odjava korisnika

Oznaka ZAHTEV12

Opis Prikaz poruke o uspešnosti odjave korisnika.

ID 12.

Prioritet implementacije 3

Status Skica

Implemetacioni rizik Srednji

**Zahtev 13. Izdavanje potvrde o razduženosti**

Naziv Izdavanje potvrde o razduženosti

Oznaka ZAHTEV13

Opis Pretraga clanova po broju indeksa, prezimenu, imenu.

Prikaz pronadjenih jedinica - tabelarno (sa podacima o zaduzenjima ako ih ima).

Kreiranje potvrde za izabranog studenta.

Stampanje potvrde.

ID 13.

Prioritet implementacije 3

Status Skica

Implemetacioni rizik Srednji

### 3. FAZE RAZVOJA SOFTVERA

**Početak projekta: 1.4.2022.**

**Specifikacija funkcija softvera** Izvođači: Student1, Student2, Student3

Rok: 2 dana

**Projektni plan** Izvođači: Student1, Student2, Student3, Student4

Rok: 2 dan

**Projektne ideje** Izvođači: Student1, Student2, Student3, Student4

Rok: 1 dan

**Početak faze: 8.4.2022.**

**Dizajn sloja podataka** Izvođači: Student1

Rok: 3 dana

**Dizajn KI (UI)** Izvođači: Student2

Rok: 5 dana

**Dizajn srednjeg aplikacionog sloja** Izvođači: Student3

Rok: 2 dana

**Početak faze: 20.4.2022.**

**Implementacija sloja podataka** Izvođači: Student1

Rok: 1 dan

**Implementacija KI** Izvođači: Student3

Rok: 10 dana

**Implementacija srednjeg sloja** Izvođači: Student2

Rok: 5 dana

**Početak faze: 10.5.2022.**

**Testiranje pojedinih delova** Izvođači: Student4

Rok: 3 dana

**Integracija softvera** Izvođači: Student3, Student4

Rok: 8 dana

**Testiranje celog softvera** Izvođači: Student4

Rok: 3 dana

**Dokumentovanje** Izvođači: Student1, Student2, Student3, Student4

Rok: 1 dan

**Izbor licence** Izvođači: Student4

Rok: 1 dan

**Objavljivanje** Predaja rada umesto objavljivanja

**Očekivani termin: 25.5.2022.**

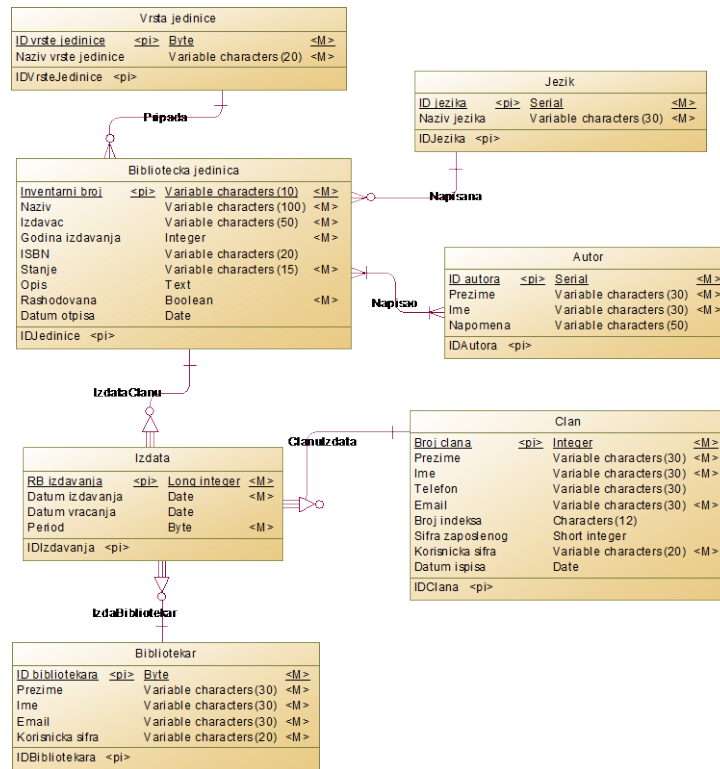
### 3.1. Projektne ideje

- **Osnovna funkcija softvera** – Podrška radu fakultetske biblioteke.
- **Funkcije softvera:** unos, izmena i brisanje (ispis) korisnika (članova), kao i bibliotečkih jedinica, izdavanje i vraćanje jedinica, otpis i rashod jedinica, formiranje opomena za nevraćene jedinice, pregled i pretraga fonda, članstva i zaduženja, generisanje izveštaja za potvrde o zaduženosti, prijava i odjava korisnika.
- **Korisnici softvera:**
  1. Bibliotekari
  2. Studenti
  3. Zaposleni
- **Vrsta (tip) softvera:** web aplikacija.
- **Softverska arhitektura:** troslojna arhitektura softvera:
  - Sloj podataka: Relaciona baza podataka, MySQL (dizajn: CASE alat).
  - Srednji aplikacioni sloj (backend): logika aplikacije, klase za povezivanje sa slojem podataka i KI (UI) PHP programski jezik, alat: Visual Studio Code.
  - Prezentacioni sloj (KI, UI, frontend): HTML, CSS, JS, PHP, Bootstrap radni okvir (framework), alat: Visual Studio Code, softver za obradu slika (Gimp, Paint).
- **Potrebni serveri:** Apache web server, MySQL server baze podataka (preko XAMP paketa servera).
- **Dizajn:** Case alata Power Designer.
- **Operativni sistem:** Microsoft Windows 7/11.
- **Dokumentovanje:** Microsoft Word 365, Apache Open Office.
- **Testiranje:** CASE alat, Cypress.
- **CVS system (repozitorijum):** GitHub.

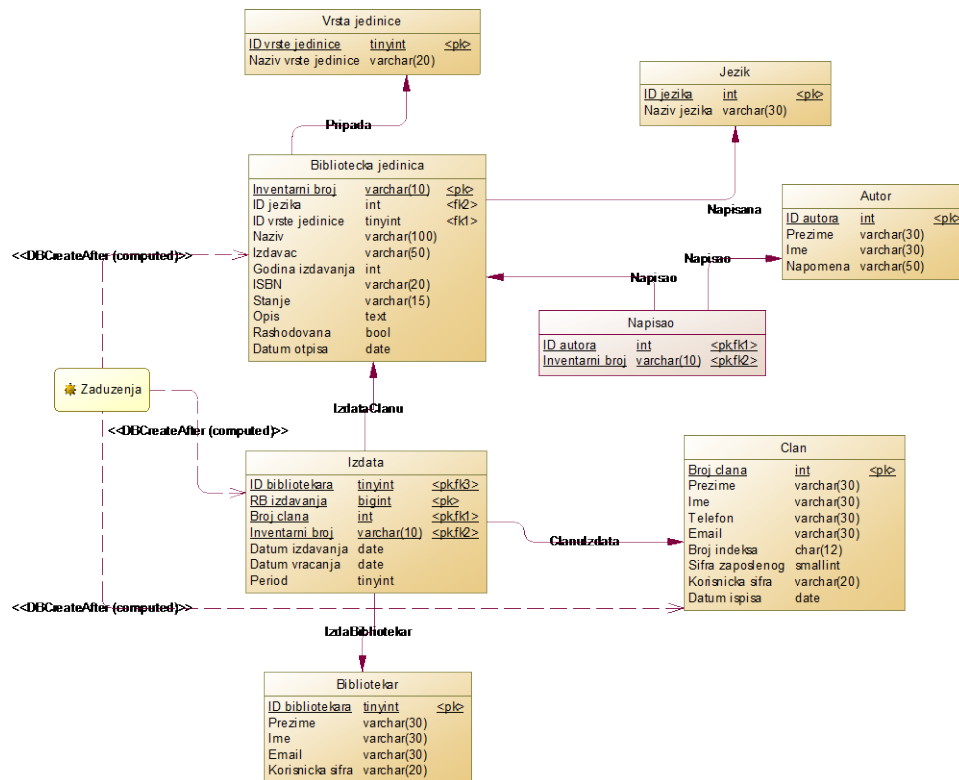


## 4. Dizajn softvera

### 4.1. Sloj podataka - Konceptualni model podataka (EER model)

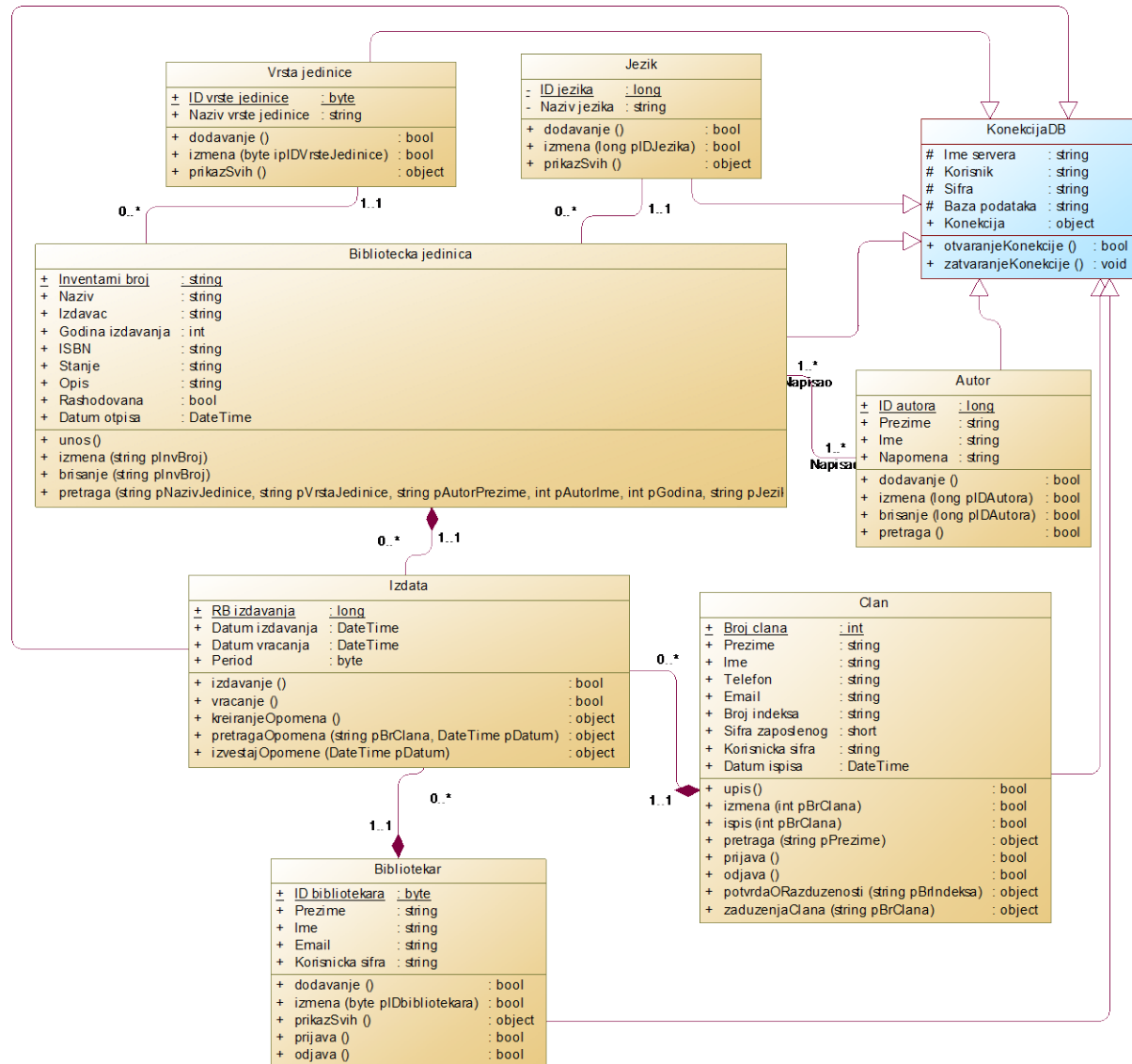


### Fizički model podataka (Relacioni model)



## 4.2. Srednji aplikacioni sloj

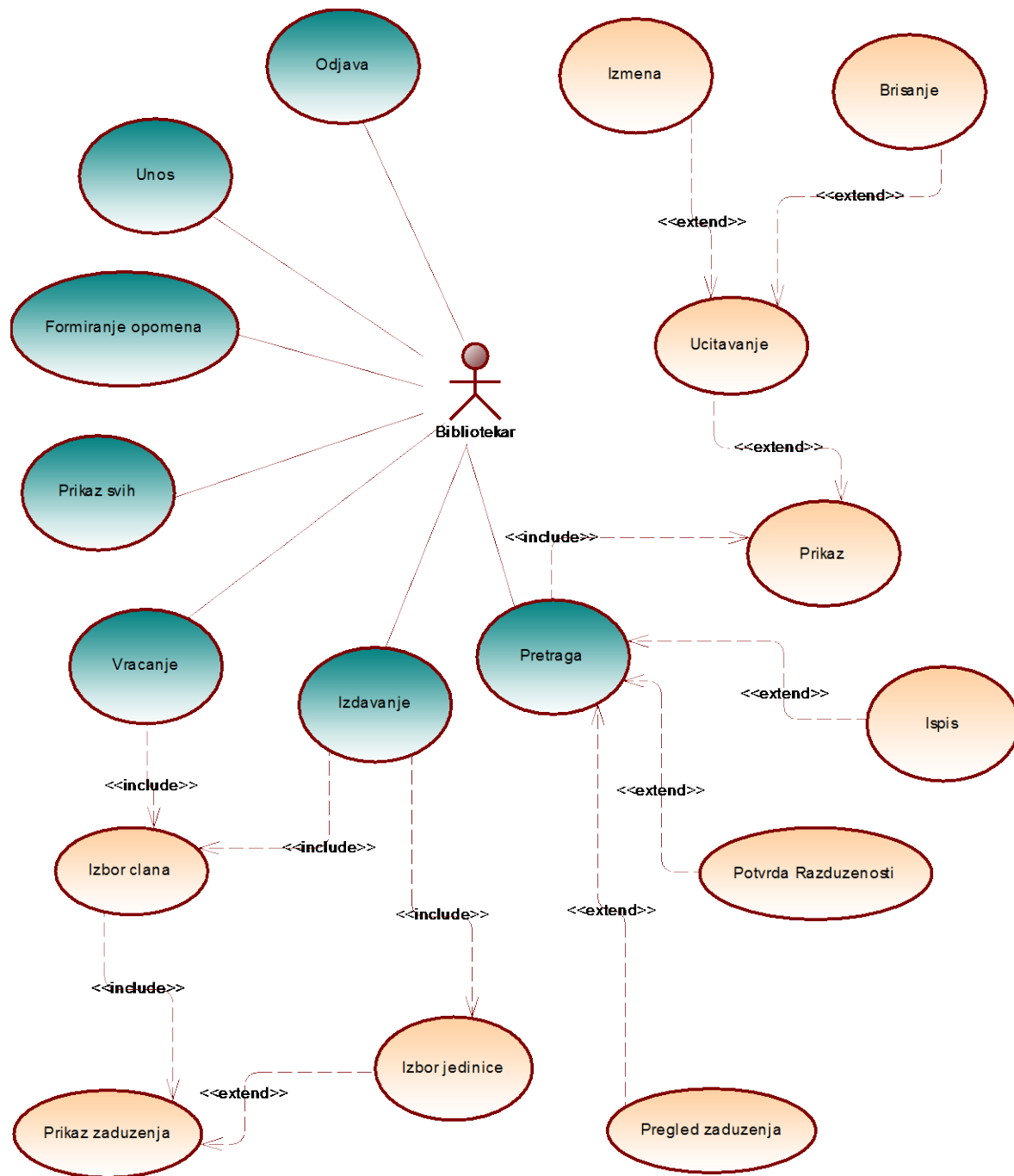
### Objektno orijentisani model - Dijagram klasa



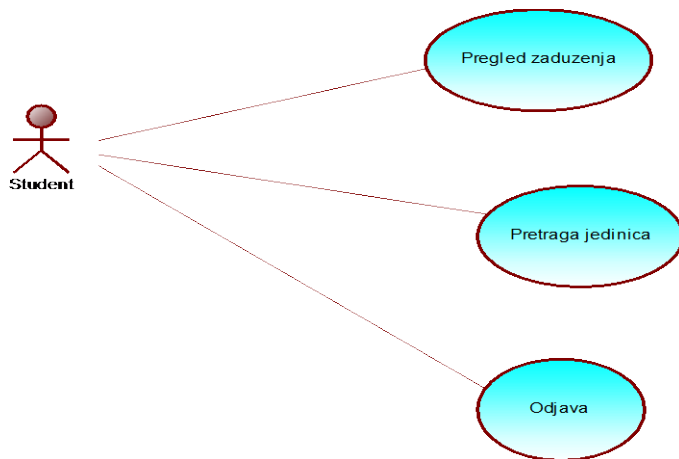
#### 4.3. Dizajn korisničkog interfejsa

##### Objektno orijentisani model - dijagram slučaja korišćenja

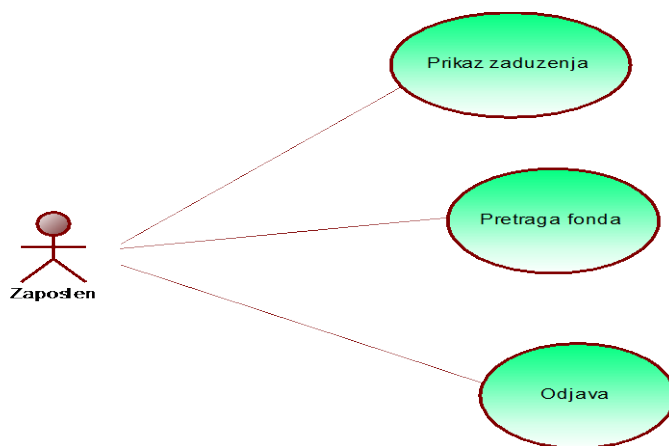
Softverske funkcije za korisnika BIBLIOTEKAR:



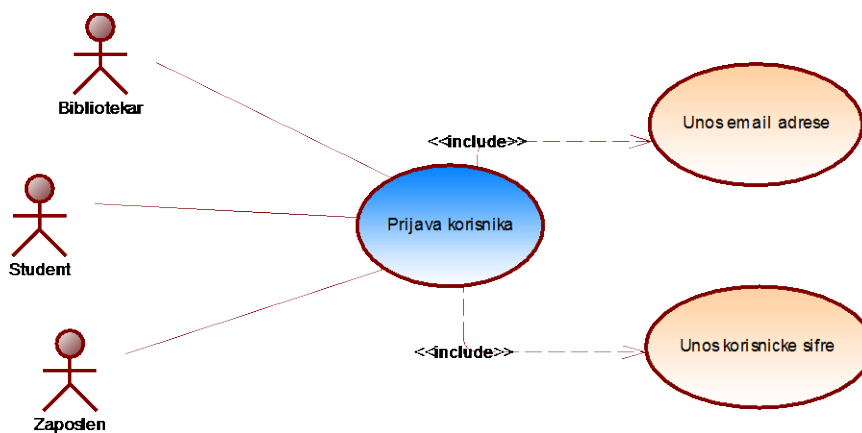
Softverske funkcije za korisnika STUDENT:



Softverske funkcije za korisnika ZAPOSLENI:

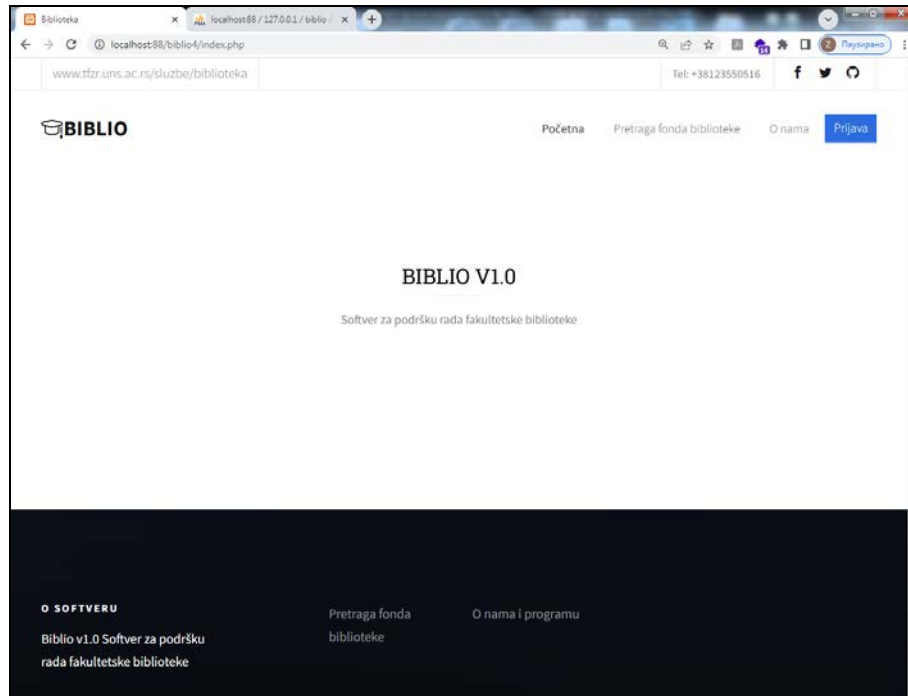


Detaljniji prikaz softverske funkcije PRIJAVA KORISNIKA, nakon čijeg izvršavanja korisnici mogu pristupati prethodno prikazanim funkcijama:



## 5. PRIKAZ SOFTVERA

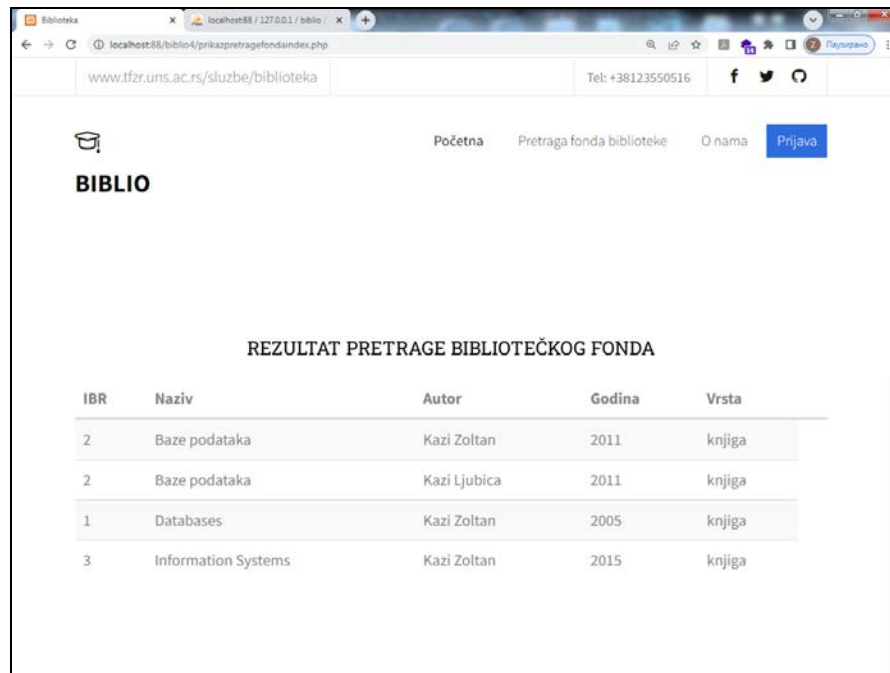
**Pokretanje aplikacije:** Nakon instaliranja XAMP softverskog paketa, aplikacija se kopira u folder C:\Xamp\htdocs\Biblioteka. U PHPMyAdmin softveru se kreira baza podataka sa nazivom: biblioteka. Zatim se izvrši SQL DDL skript za kreiranje objekata baze podataka. Pokretanje iz web browser-a, u URL polje potrebno je upisati: <http://localhost/biblioteka> ili <http://localhost/biblioteka/index.php>.



Početna stranica softvera omogućuje bilo kom korisniku da pokrene stranicu za pretragu fonda biblioteke, da uradi prijavu korisnika ili da vidi informacije o softveru i autorima. Stranica za pretragu fonda sadrži 6 polja za unos kriterijuma pretrage prema specifikaciji funkcija.

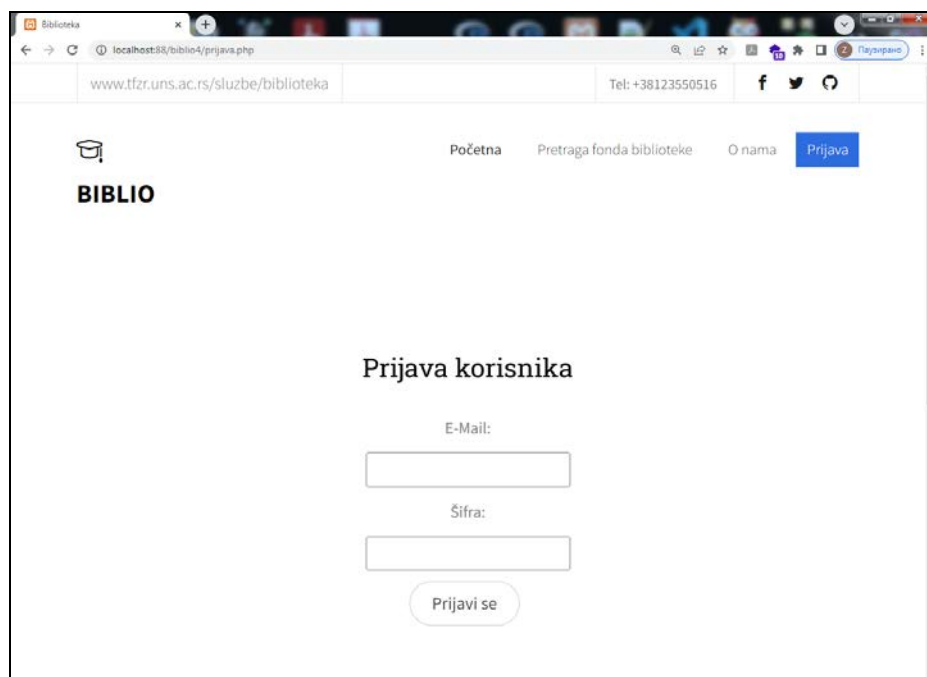
The screenshot shows the search page of the BIBLIO V1.0 software. The page has a white background with a dark blue header and footer. The header contains the BIBLIO logo, navigation links for 'Početna', 'Pretraga fonda biblioteke', 'O nama', and a 'Prijava' button. The main content area features the title 'PRETRAGA BIBLIOTEČKOG FONDA' and a search form with six input fields: 'po nazivu:', 'po jeziku:', 'po prezimenu autora:', 'po imenu autora:', 'po vrsti jedinice:', and 'po godini izdavanja:'. Below the input fields are two buttons: 'Pronađi' and 'Nova pretraga'.

Rezultat pretrage bibliotečkog fonda je stranica sa tabelarnim prikazom sa osnovnim podacima o jedinici (inventarni broj, naziv, autor, godina, vrsta jedinice).



IBR	Naziv	Autor	Godina	Vrsta
2	Baze podataka	Kazi Zoltan	2011	knjiga
2	Baze podataka	Kazi Ljubica	2011	knjiga
1	Databases	Kazi Zoltan	2005	knjiga
3	Information Systems	Kazi Zoltan	2015	knjiga

Stranica za prijavu korisnika: bibliotekara i članova, koji mogu biti zaposleni na fakultetu i studenti, se sastoji od rubrika za unos email adrese korisnika i šifre.



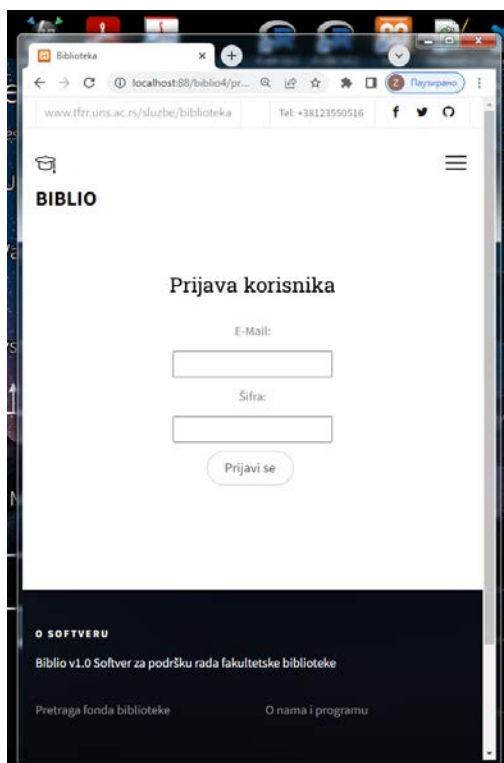
Prijava korisnika

E-Mail:

Šifra:

Prijavi se

Sve stranice softvera imaju osobinu da su "Responsive", sadržaj se automatski skalira u zavisnosti od rezolucije ekrana i veličine prozora u kom se prikazuje, što ih čini pogodnim za prikaz na različitim uređajima.



BIBLIO

### Prijava korisnika

E-Mail:

Šifra:

Prijavi se

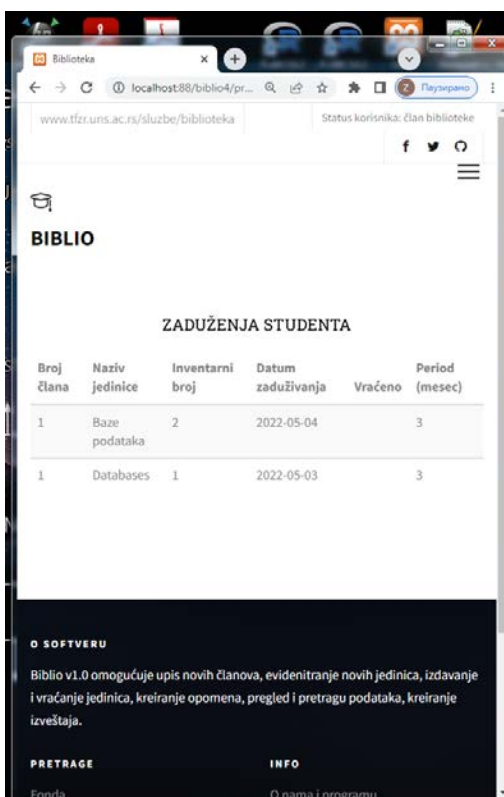
O SOFTVERU

Biblio v1.0 Softver za podršku rada fakultetske biblioteke

Pretraga fonda biblioteke

O nama i programu

Prikaz stranice sa izlistanim zaduženjima studenta koji je prijavljen:



BIBLIO

### ZADUŽENJA STUDENTA

Broj člana	Naziv jedinice	Inventarni broj	Datum zaduživanja	Vraćeno	Period (mesec)
1	Baze podataka	2	2022-05-04		3
1	Databases	1	2022-05-03		3

O SOFTVERU

Biblio v1.0 omogućuje upis novih članova, evidenitranje novih jedinica, izdavanje i vraćanje jedinica, kreiranje opomena, pregled i pretragu podataka, kreiranje izveštaja.

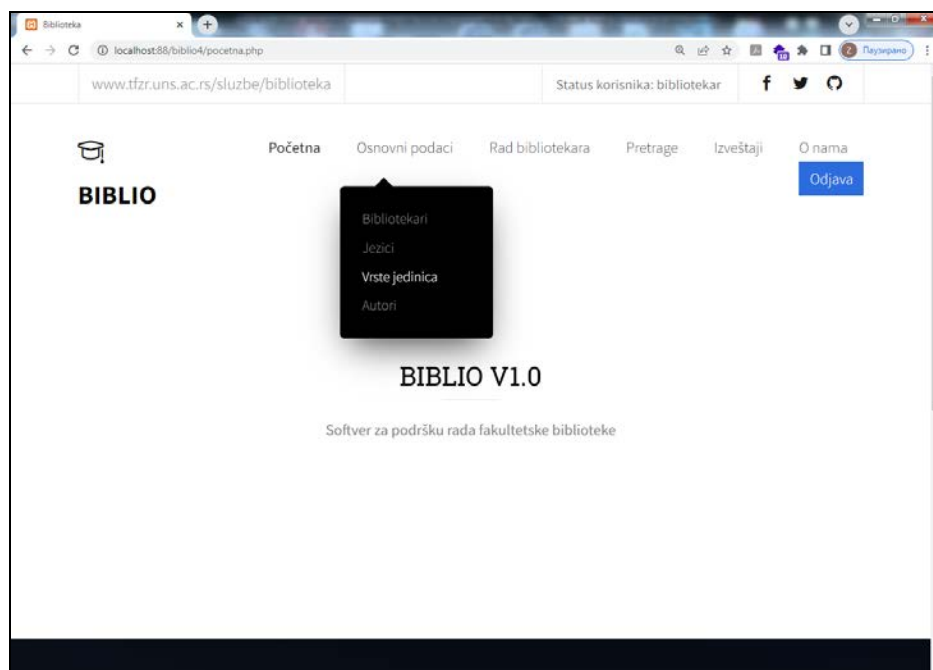
PRETRAGA

INFO

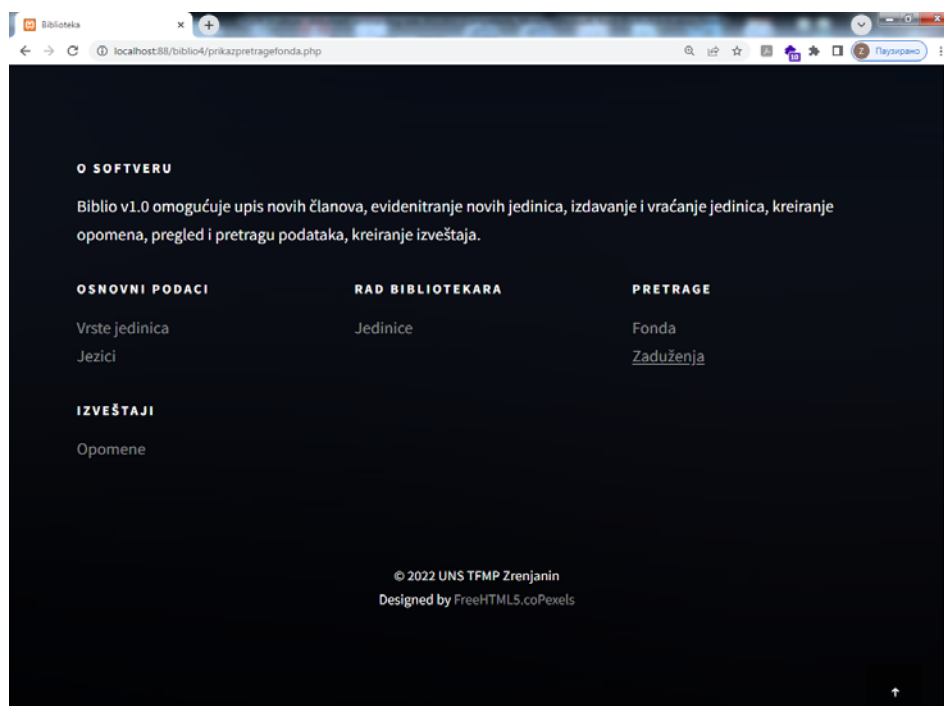
Fonda

O nama i programu

Web stranice BIBLIO web aplikacije imaju padajuće menije u gornjem delu stranice, u zaglavlju (header)...



... I sistem hyperlinkova u podnožju (footer) kojima se pristupa funkcijama softvera.





Prikaz stranice za unos osnovnih podataka, sa formom i poljima za unos vrednosti od strane korisnika. Taster "Unos nove" učitava novu stranicu na kojoj se može videti rezultat uspešnog ili neuspešnog unosa podataka. Ispod forme se prikazuje, u tabelarnom obliku, spisak svih unetih osnovnih podataka u bazu. U tabeli se nalaze tasteri koji omogućuju izmenu podataka.

**BIBLIO**

Početna Osnovni podaci Rad bibliotekara Pretrage Izveštaji O nama

**VRSTE BIBLIOTEČKIH JEDINICA**

ID vrste jedinice\*

Naziv vrste jedinice\*

Unos nove Poništi

ID	Naziv	Akcija
1	knjiga	Izmena
2	zbirka	Izmena

Prikaz stranice za izmenu podataka:

**BIBLIO**

Početna Osnovni podaci Rad bibliotekara Pretrage Izveštaji O nama

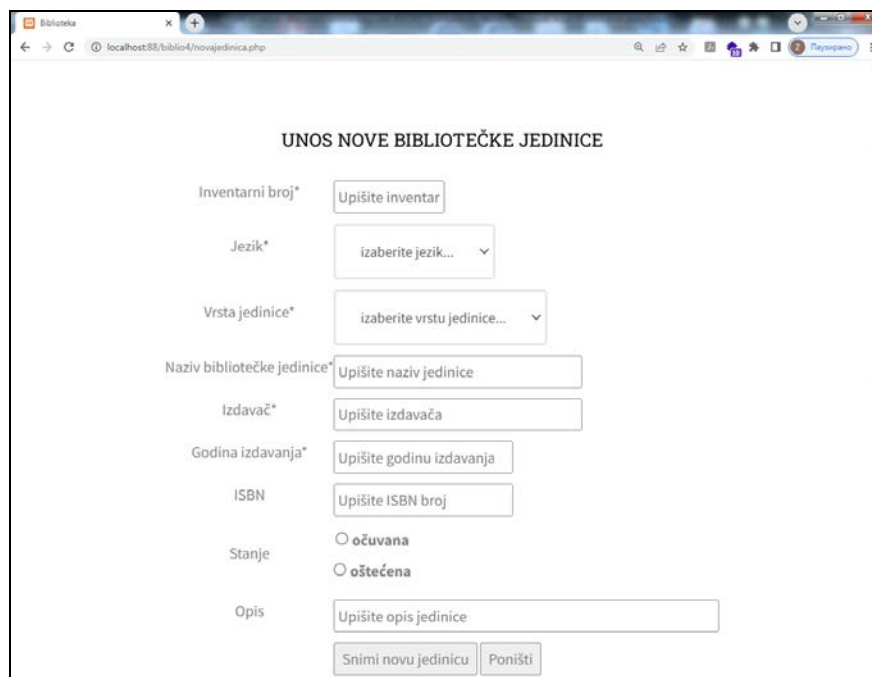
**VRSTE BIBLIOTEČKIH JEDINICA - IZMENA**

ID vrste jedinice\*

Naziv vrste jedinice\*

Snimi izmene Odustani

Stranica za unos nove bibliotečke jedinice kroz formu za poljima za unos vrednosti (inventarni broj, naziv, izdavač, godina izdavanja, ISBN, opis), listama za izbor osnovnih podataka za jezik i vrstu publikacije, kao i opcijama za izbor stanja (očuvna ili oštećena).

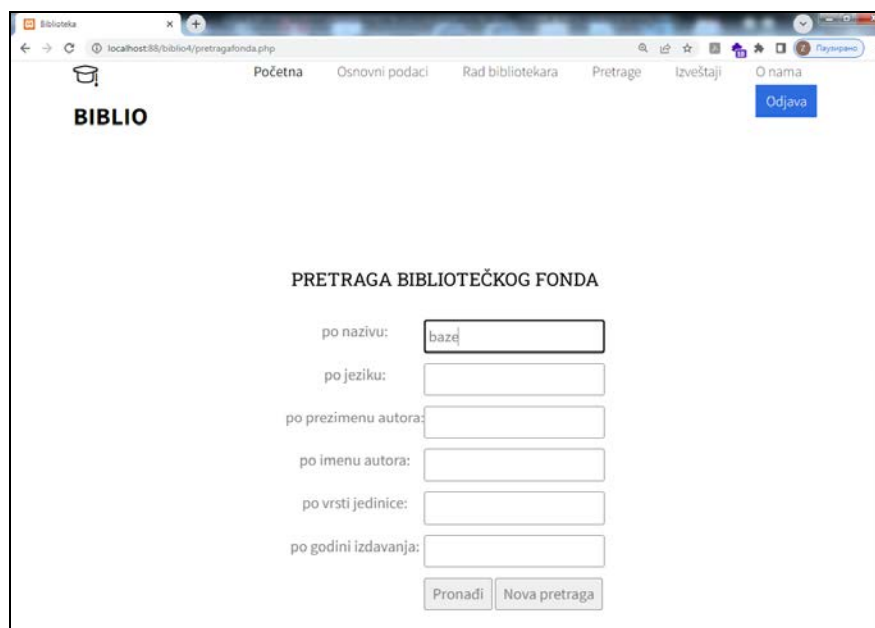


The screenshot shows a web browser window with the address bar displaying 'localhost:88/biblio4/novajedinica.php'. The page title is 'Biblioteka'. The main heading is 'UNOS NOVE BIBLIOTEČKE JEDINICE'. The form contains the following fields and controls:

- Inventarni broj\***: Text input with placeholder 'Upišite inventar'.
- Jezik\***: Dropdown menu with placeholder 'izaberite jezik...'.
- Vrsta jedinice\***: Dropdown menu with placeholder 'izaberite vrstu jedinice...'.
- Naziv bibliotečke jedinice\***: Text input with placeholder 'Upišite naziv jedinice'.
- Izdavač\***: Text input with placeholder 'Upišite izdavača'.
- Godina izdavanja\***: Text input with placeholder 'Upišite godinu izdavanja'.
- ISBN**: Text input with placeholder 'Upišite ISBN broj'.
- Stanje**: Radio buttons for 'očuvana' and 'oštećena'.
- Opis**: Text input with placeholder 'Upišite opis jedinice'.

At the bottom of the form are two buttons: 'Snimi novu jedinicu' and 'Poništi'.

Stranica za pretragu fonda sadrži polja za unos kriterijuma pretrage po nazivu dela, jeziku na kom je napisano, po prezimeni ili imenu autora, po vrsti jednice, po godini osnivanja. Taster "Nova pretraga" poništava uneti kriterijum pretrage, dok taster "Pronađi" započinje pretragu baze i učitavanje stranice za prikaz rezultata pretrage.

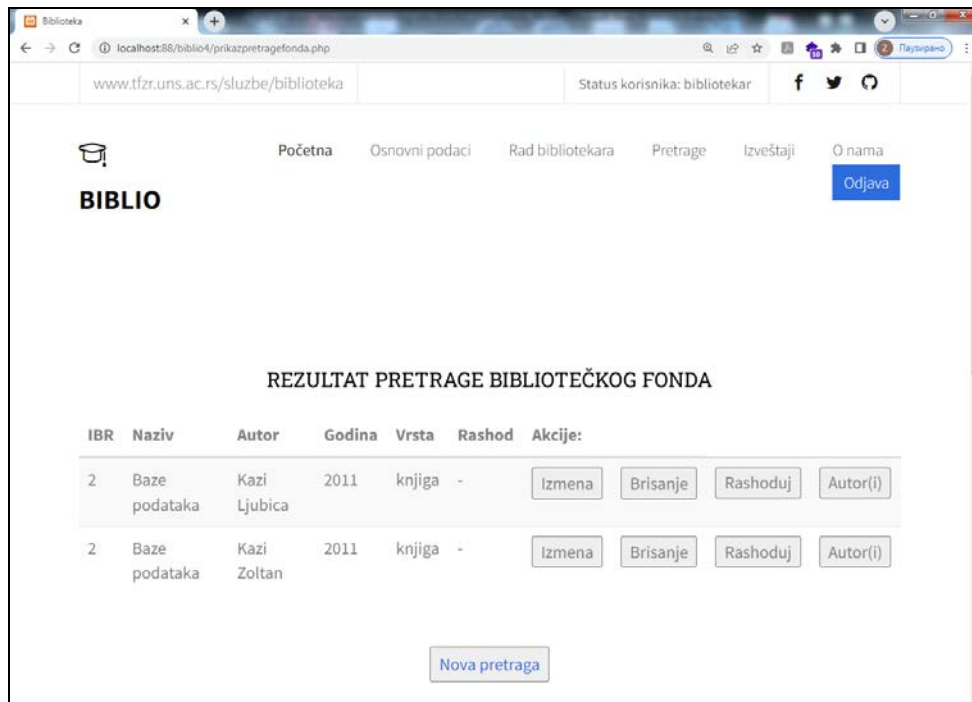


The screenshot shows a web browser window with the address bar displaying 'localhost:88/biblio4/pretragafonda.php'. The page title is 'Biblioteka'. The main heading is 'PRETRAGA BIBLIOTEČKOG FONDA'. The form contains the following fields and controls:

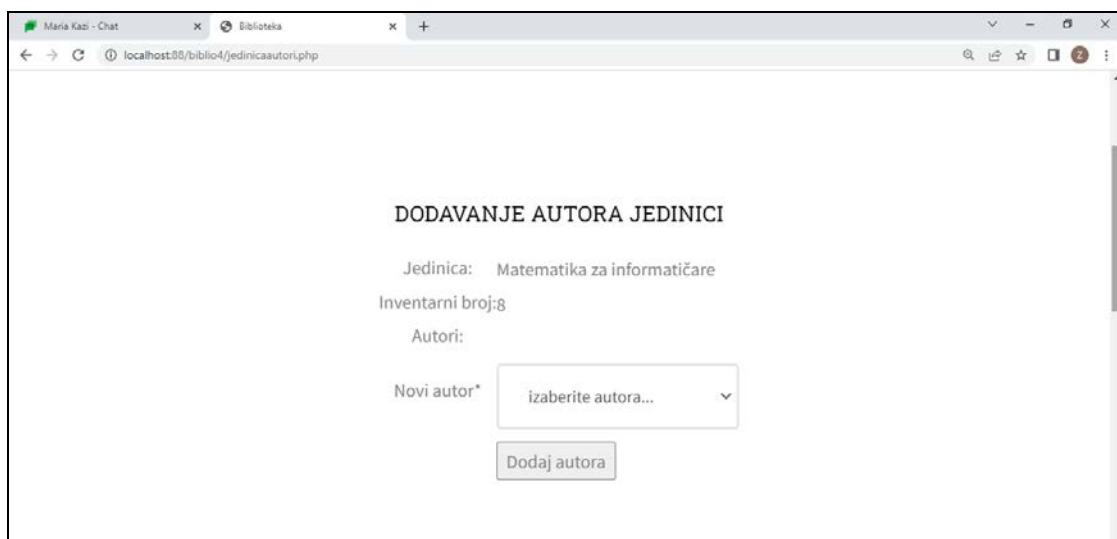
- po nazivu:** Text input with placeholder 'baze'.
- po jeziku:** Text input.
- po prezimenu autora:** Text input.
- po imenu autora:** Text input.
- po vrsti jedinice:** Text input.
- po godini izdavanja:** Text input.

At the bottom of the form are two buttons: 'Pronađi' and 'Nova pretraga'.

Stranica za prikaz rezultata pretrage podataka u tabelarnoj formi. Sadrži podatke o delima: inventarni broj, naziv, autora, godinu izdavanja, vrstu jedinice i podatak o tome da li je rashodovana ili ne. U svakom redu su dodati tasteri za izmenu, brisanje i rashodovanje dela/jedinice, kao i dodavanje novih autora.



Stranica za dodavanje novih autora se sastoji od forme koja prikazuje baziv jedinice, inventarni broj, već unete autore (ukoliko postoje) i listu svih autora, iz koje se bira novi autor. Potvrda je preko tastera "Dodaj autora". Kada se doda novi autor prikazuje se ista stranica sa mogućnošću da se dodaju i sledeći koautori.



Stranica za potvrdu brisanja bibliotečke jedinice ima samo dva tastera “Da” za potvrdu i “Ne” za odustajanje i povratak na stranicu sa prikazom rezultata pretrage podataka.



Stranica za izdavanje bibliotečke jedinice ima formu za poljima za unos vrednosti (rb izdavanja), listama za izbor publikacije, člana, bibliotekara, kao i opcijama za izbor dužine izdavanja (3 ili 6 meseci).

The screenshot shows a web browser window with the URL `localhost:88/biblio4/izdavanje.php`. The page is titled 'IZDAVANJE BIBLIOTEČKE JEDINICE'. The form contains the following fields and options:

- 'RB izdavanja\*': A text input field containing the value '9'.
- 'Jedinica\*': A dropdown menu with the selected option 'Baze podataka odabrana poglavlja'.
- 'Član\*': A dropdown menu with the selected option 'Djuric Djura-4'.
- 'Bibliotekar\*': A dropdown menu with the selected option 'Dobrosavljev Olivera-1'.
- 'Period': Radio button options for '3 meseca' (selected) and '6 meseci'.
- At the bottom, there are two buttons: 'izdaj jedinicu' and 'Poništi'.

Stranica za tabelarni prikaz zaduženja svih studenata sortirana po datumu zaduživanja tako da na vrhu tabele budu najnovija zaduživanja. Jedinice koje nisu razdužene mogu se, preko tastera “Razduživanje” vratiti u biblioteku. Kod već razduženih jedinica taster ne postoji.

Broj clana	Član	Naziv jedinice	Inventarni broj	Datum zaduživanja	Vraćeno	Period (mesec)	Vrsta	Akcija
2	Lucic Ljiljana	Information Systems	3	2022-05-05		3	knjiga	Razduživanje
1	Arbanovski Strahinja	Baze podataka	2	2022-05-04		3	knjiga	Razduživanje
1	Arbanovski Strahinja	Databases	1	2022-05-03		3	knjiga	Razduživanje

Izveštaj – spisak opomena je dokument u elektronskom obliku, koji prikazuje podatke za sve studente kojima su izdate opomene. Preko menija za opcije stranice se može štampati stranica, preko “Print,tj. Odštampaj” u papirnoj formi ili...

RB	Prezime	Ime	Broj indeksa	Biblioteka jedinica	Opomena
1	Arbanovski	Strahinja		Databases	Opomena izdata 1.4.2022.
2	Arbanovski	Strahinja		Baze podataka	Opomena izdata 1.4.2022.

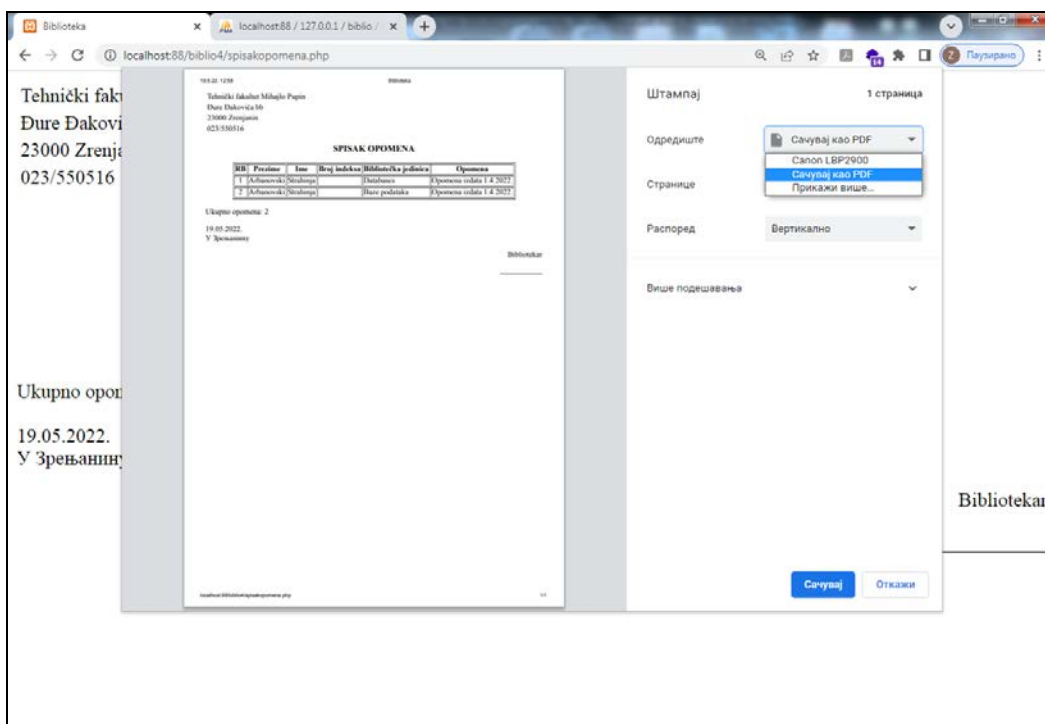
Ukupno opomena: 2

19.05.2022.  
U Zrenjaninu

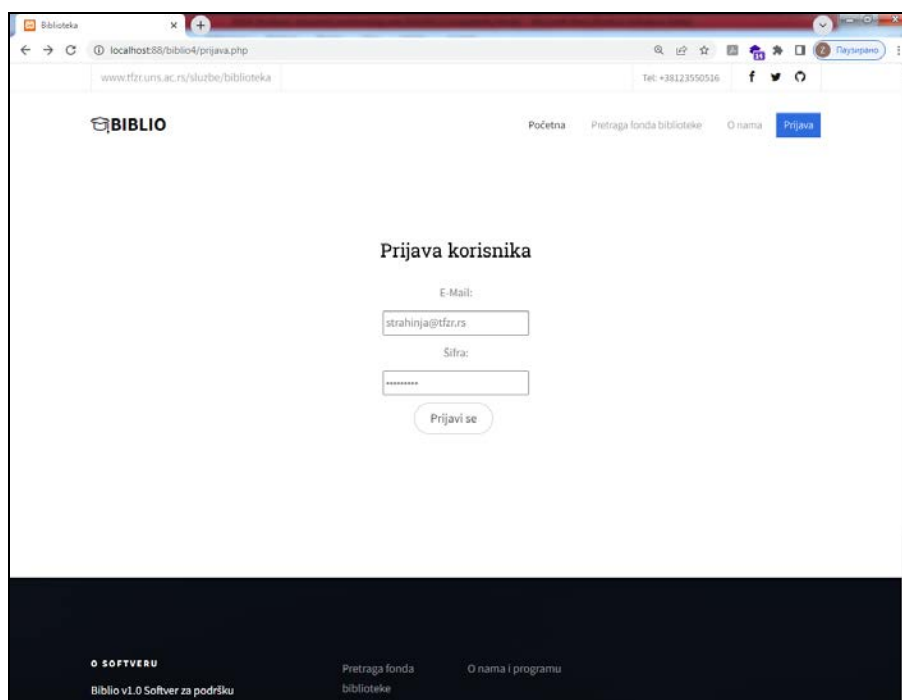
**Bibliotekar**

- Nazad (Alt+Стрелица налево)
- Погледај (Alt+Стрелица надесно)
- Учитај поново (Ctrl+R)
- Сачувај као... (Ctrl+S)
- Одштампай... (Ctrl+P)
- Пребаци...
- Тражите слику уз Google Lens
- Направи QR код за ову страницу
- Преведи на English
- Прикажи извор странице (Ctrl+U)
- Прегледај

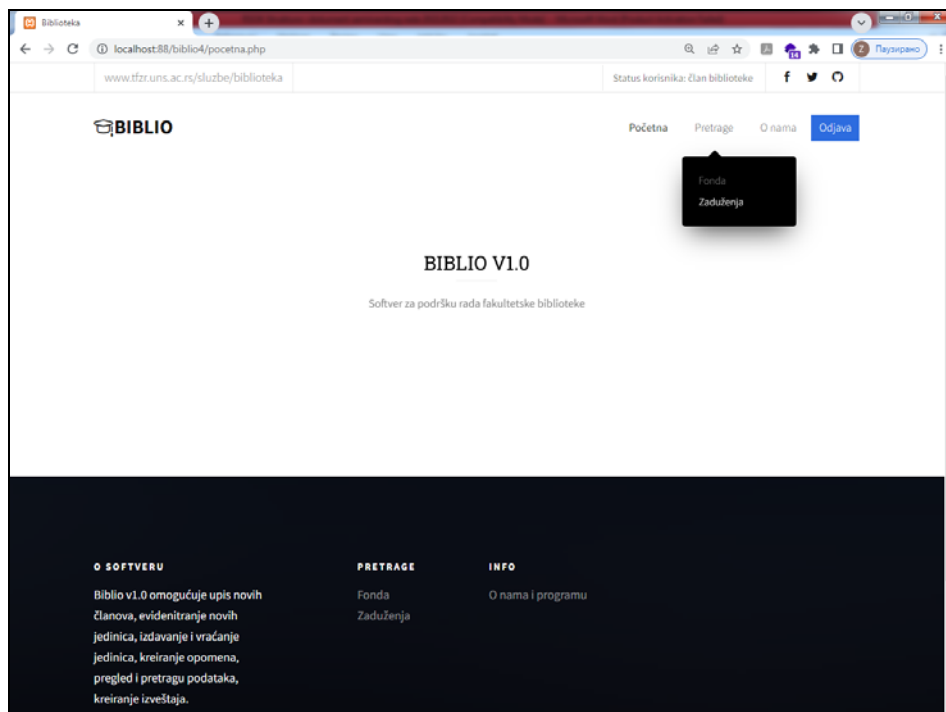
... snimiti u PDF formatu kao elektronska verzija dokumenta koji se može koristiti na fakultetu.



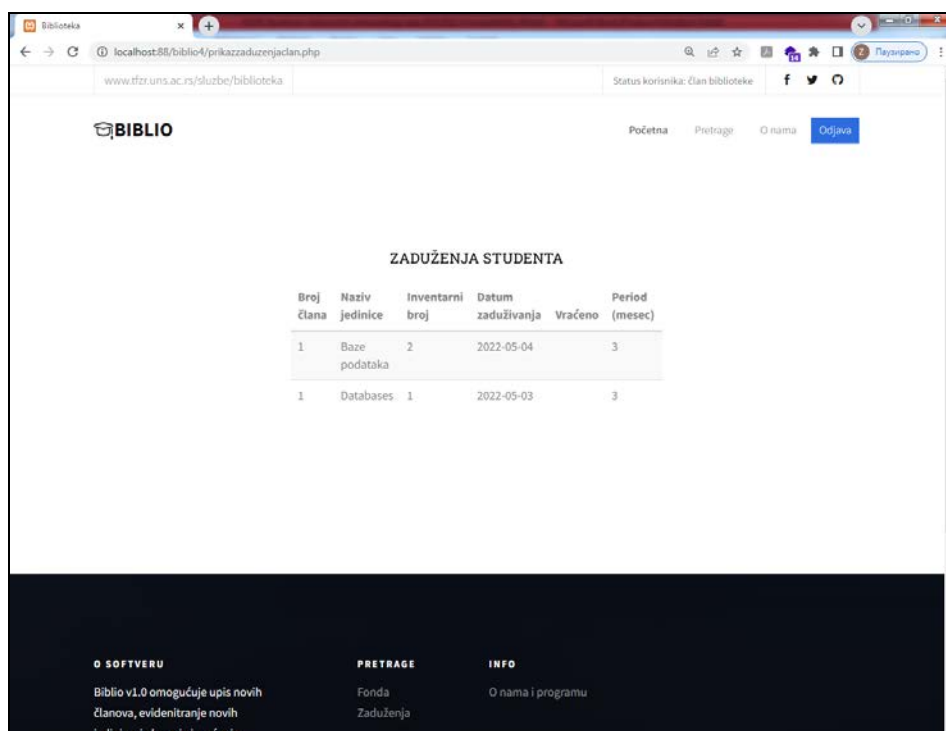
Stranica za prijavu korisnika zahteva unos email adrese koju je student dao bibliotekaru prilikom upisa i unosa šifre/lozinke koja se čuva u bazi podataka. Taster “Prijavi se” otvara novu stranicu za proveru da li postoji taj korisnik sa tom šifrom. Ukoliko postoji učitava se i prikazuje početna stranica za svaki profil korisnika. U protivnom korisnik se preusmerava na ponovnu prijavu.



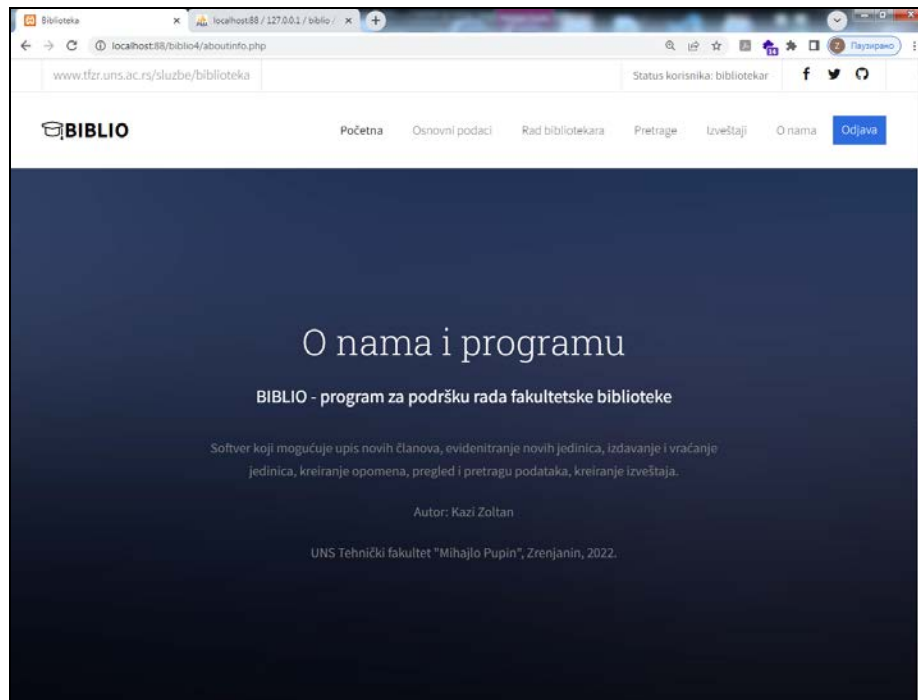
Kada se prijave članovi biblioteke, imaju na raspolaganju samo dve opcije: pretragu fonda i prikaz svojih zaduženja.



Prikaz stranice sa zaduženjima prijavljenog člana biblioteke:



Prikaz stranice sa informacijama o programu, autorima, sa kratkim opisom namene web aplikacije:





## 6. REALIZACIJA I IMPLEMENTACIJA

### 6.1. PRIKAZ KARAKTERISTIČNIH ELEMENATA REALIZACIJE

SQL DDL skript za kreiranje relacione baze podataka:

```
/*=====*/
/* DBMS name:      MySQL 5.0                                */
/* Created on:      17.03.2022. 14:50:02                     */
/*=====*/

drop procedure if exists Zaduzenja;
drop table if exists Autor;
drop table if exists `Biblioteka jedinica`;
drop table if exists Bibliotekar;
drop table if exists Clan;
drop table if exists Izdata;
drop table if exists Jezik;
drop table if exists Napisao;
drop table if exists `Vrsta jedinice`;

/*=====*/
/* Table: Autor                                             */
/*=====*/
create table Autor
(
    `ID autora`          int not null auto_increment,
    Prezime              varchar(30) not null,
    Ime                  varchar(30) not null,
    Napomena             varchar(50),
    primary key (`ID autora`)
);

/*=====*/
/* Table: `Biblioteka jedinica`                             */
/*=====*/
create table `Biblioteka jedinica`
(
    `Inventarni broj`    varchar(10) not null,
    `ID jezika`          int not null,
    `ID vrste jedinice`  tinyint not null,
    Naziv                varchar(100) not null,
    Izdavac              varchar(50) not null,
    `Godina izdavanja`   int not null,
    ISBN                varchar(20),
    Stanje               varchar(15) not null,
    Opis                text,
    Rashodovana          bool not null,
    `Datum otpisa`       date,
    primary key (`Inventarni broj`)
);

/*=====*/
/* Table: Bibliotekar                                       */
/*=====*/
create table Bibliotekar
(
    `ID bibliotekara`    tinyint not null,
    Prezime              varchar(30) not null,
    Ime                  varchar(30) not null,
    Email                varchar(30) not null,
    `Korisnicka sifra`   varchar(20) not null,
    primary key (`ID bibliotekara`)
);

/*=====*/
```

```

/* Table: Clan */
/*=====*/
create table Clan
(
    `Broj clana`          int not null,
    Prezime               varchar(30) not null,
    Ime                   varchar(30) not null,
    Telefon               varchar(30),
    Email                 varchar(30) not null,
    `Broj indeksa`        char(12),
    `Sifra zaposlenog`     smallint,
    `Korisnicka sifra`     varchar(20) not null,
    `Datum ispisa`         date,
    primary key (`Broj clana`)
);

/*=====*/
/* Table: Izdata */
/*=====*/
create table Izdata
(
    `RB izdavanja`        bigint not null,
    `Broj clana`          int not null,
    `Inventarni broj`      varchar(10) not null,
    `ID bibliotekara`      tinyint not null,
    `Datum izdavanja`      date not null,
    `Datum vracanja`       date,
    Period                 tinyint not null default 1,
    primary key (`Broj clana`, `Inventarni broj`, `ID bibliotekara`, `RB izdavanja`)
);

/*=====*/
/* Table: Jezik */
/*=====*/
create table Jezik
(
    `ID jezika`           int not null auto_increment,
    `Naziv jezika`         varchar(30) not null,
    primary key (`ID jezika`)
);

/*=====*/
/* Table: Napisao */
/*=====*/
create table Napisao
(
    `ID autora`           int not null,
    `Inventarni broj`      varchar(10) not null,
    primary key (`ID autora`, `Inventarni broj`)
);

/*=====*/
/* Table: `Vrsta jedinice` */
/*=====*/
create table `Vrsta jedinice`
(
    `ID vrste jedinice`    tinyint not null,
    `Naziv vrste jedinice` varchar(20) not null,
    primary key (`ID vrste jedinice`)
);

alter table `Bibliotecka jedinica` add constraint FK_Napisana foreign key (`ID jezika`)
references Jezik (`ID jezika`) on delete restrict on update cascade;

alter table `Bibliotecka jedinica` add constraint FK_Pripada foreign key (`ID vrste
jedinice`)

```

```

        references `Vrsta jedinice` (`ID vrste jedinice`) on delete restrict on update
        cascade;

alter table Izdata add constraint FK_ClanuIzdata foreign key (`Broj clana`)
        references Clan (`Broj clana`) on delete restrict on update cascade;

alter table Izdata add constraint FK_IzdaBibliotekar foreign key (`ID bibliotekara`)
        references Bibliotekar (`ID bibliotekara`) on delete restrict on update cascade;

alter table Izdata add constraint FK_IzdataClanu foreign key (`Inventarni broj`)
        references `Bibliotecka jedinica` (`Inventarni broj`) on delete restrict on update
        cascade;

alter table Napisao add constraint FK_Napisao foreign key (`ID autora`)
        references Autor (`ID autora`) on delete restrict on update cascade;

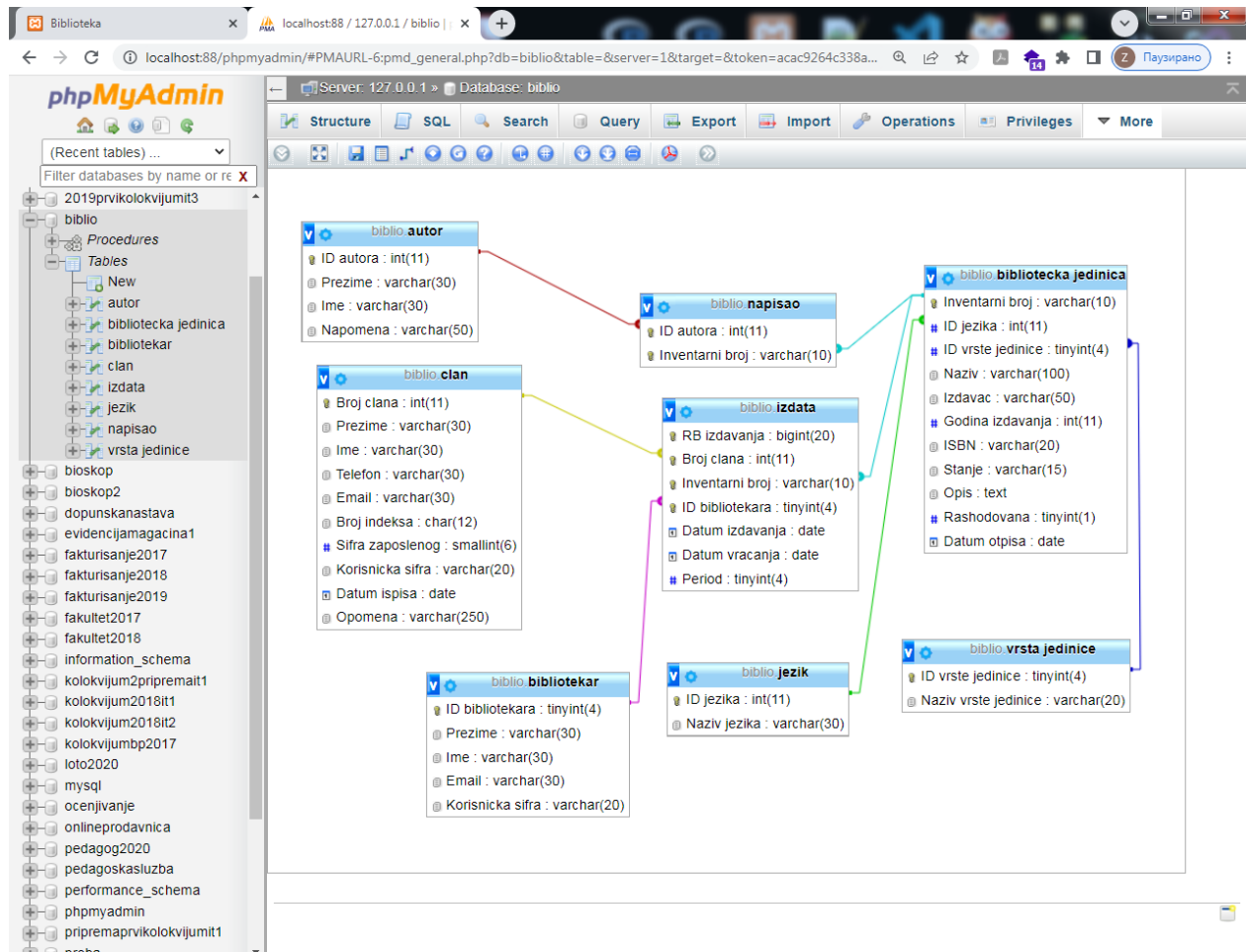
alter table Napisao add constraint FK_Napisao2 foreign key (`Inventarni broj`)
        references `Bibliotecka jedinica` (`Inventarni broj`) on delete restrict on update
        cascade;

/*=====*/
/* Naknadne izmene */
/*=====*/
ALTER TABLE `clan` ADD `Opomena` nvarchar(250) null;

delimiter //
create procedure Zaduzenja (IN pautorime varchar(30), IN pautorprezime varchar(30))
begin
select `Bibliotecka jedinica`.`Inventarni broj`,
`Bibliotecka jedinica`.Naziv,
`Bibliotecka jedinica`.Izdavac,
Clan.`Broj clana`,
Clan.Prezime,
Clan.Ime,
Izdata.`Datum izdavanja`,
Izdata.`Datum vracanja`
from Izdata, Clan, `Bibliotecka jedinica`
where Clan.`Broj clana` = Izdata.`Broj clana` and
`Bibliotecka jedinica`.`Inventarni broj` = Izdata.`Inventarni broj` and
Clan.Ime=pautorime and Clan.Prezime=pautorprezime;
end //
delimiter;

```

**Kreirana relacionalna baza podataka** (uz pomoć SQL DDL skripta), u okviru PHPMyAdmin web aplikacije za administraciju baze podataka:



## Biblioteka klasa (PHP programski jezik)

**Konekcija do baze podataka:**

```
<?php
class KonekcijaDB
{
    protected $ImeServera="localhost";
    protected $Korisnik="root";
    protected $Sifra="";
    protected $BazaPodataka="biblio";
    public $Konekcija;

    public function OtvaranjeKonekcije()
    {
        //Otvaranje konekcije do BP
        $this->Konekcija="";
        $this->Konekcija=mysqli_connect($this->ImeServera,$this->Korisnik,$this->Sifra,$this->BazaPodataka);
        if (!$this->Konekcija)
        {
            echo('Nije uspostavljena konekcija do baze podataka!');
        }
        return $this->Konekcija;
    }
}
```

```

public function ZatvaranjeKonekcije()
{
    //Zatvaranje konekcije do BP uz
    //proveru da li je otvorena pre zatvaranja
    $suspeh="false";
    if($this->Konekcija)
    {
        mysqli_close($this->Konekcija);
        $suspeh="true";
    }
    return $suspeh;
}
}??>

```

#### Vrsta jedinice:

```

<?php
class VrstaJedinice extends KonekcijaDB
{
    public $IdVrsteJedinice;
    public $NazivVrsteJedinice;

    public function Dodavanje()
    {
        //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
        $result = "";
        $suspehzbkp = "";
        $suspeh = "false";
        //otvaranje konekcije do BP
        $konekcija=$this->OtvaranjeKonekcije();
        //formiranje SQL DML upita za unos podataka
        $upit = "INSERT INTO `vrsta jedinice`(`ID vrste jedinice`, `Naziv vrste jedinice`) VALUES ('$this->IdVrsteJedinice', '$this-
>NazivVrsteJedinice')";
        //izvršavanje SQL upita
        $result = mysqli_query($konekcija, $upit);
        /*provera rezultata izvršavanja SQL upita
        i ispis informacije o gresci*/
        if(!$result)
        {
            mysqli_error($konekcija);
        }
        else
        {
            $suspeh="true";
        }
        //zatvaranje konekcije do BP
        $suspehzbkp=$this->ZatvaranjeKonekcije();
        return $suspeh;
    }

    public function Izmena($pIDVrsteJedinice)
    {
        //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
        $result = "";
        $suspehzbkp = "";
        $suspeh = "false";
        //otvaranje konekcije do BP
        $konekcija=$this->OtvaranjeKonekcije();
        //formiranje SQL DML upita za izmenu podataka
        $upit = "UPDATE `vrsta jedinice` SET `ID vrste jedinice`='<strong>$this->IdVrsteJedinice', `Naziv vrste jedinice`='<strong>$this->NazivVrsteJedinice'
WHERE `ID vrste jedinice`='<strong>$pIDVrsteJedinice'";
        //echo $upit;
        //izvršavanje SQL upita
    }
}

```

```

$result = mysqli_query($konekcija, $upit);
/*provera rezultata izvršavanja SQL upita
i ispis informacije o gresci*/
if(!$result)
{
    mysqli_error($konekcija);
}
else
{
    $suspeh="true";
}
//zatvaranje konekcije do BP
$suspehzkbp=$this->ZatvaranjeKonekcije();
return $suspeh;
}

public function PrikazSvih()
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $suspehzkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL upita za izdvajanje podataka
    $upit = "SELECT * FROM `vrsta jedinice` ORDER BY `Naziv vrste jedinice` ASC;";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        $suspeh="true";
    }
    //zatvaranje konekcije do BP
    $suspehzkbp=$this->ZatvaranjeKonekcije();
    return $result;
}
}??>

```

#### Klasa Jezik:

```

<?php
class Jezik extends KonekcijaDB
{
    private $IdJezika;
    private $NazivJezika;

    public function getIDJezika()
    {
        return $this->IdJezika;
    }
    public function setIDJezika($p)
    {
        $this->IdJezika=$p;
    }
    public function getNazivJezika()
    {
        return $this->NazivJezika;
    }
}

```

```

}
public function setNazivJezika($p)
{
    $this->NazivJezika=$p;
}

public function Dodavanje()
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i ostalih promenljivih
    $result = "";
    $suspehzkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvoranjeKonekcije();
    //formiranje SQL DML upita za unos podataka
    $upit = "INSERT INTO `jezik`(`ID jezika`, `Naziv jezika`) VALUES ('$this->IdJezika', '$this->NazivJezika');";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        {
            $suspeh="true";
        }
    }
    //zatvaranje konekcije do BP
    $suspehzkbp=$this->ZatvaranjeKonekcije();
    return $suspeh;
}

public function Izmena($pIdJezika)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $suspehzkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvoranjeKonekcije();
    //formiranje SQL DML upita za izmenu podataka
    $upit = "UPDATE `jezik` SET `ID jezika`='$this->IdJezika', `Naziv jezika`='$this->NazivJezika' WHERE `ID jezika`='$pIdJezika';";
    //echo $upit;
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        {
            $suspeh="true";
        }
    }
    //zatvaranje konekcije do BP
    $suspehzkbp=$this->ZatvaranjeKonekcije();
    return $suspeh;
}

public function PrikazSvih()

```

```

{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehzkbp = "";
    $uspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvoranjeKonekcije();
    //formiranje SQL upita za izdvajanje podataka
    $upit = "SELECT * FROM `jezik` ORDER BY `Naziv jezika` ASC;";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        $uspeh="true";
    }
    //zatvaranje konekcije do BP
    $uspehzkbp=$this->ZatvaranjeKonekcije();
    return $result;
}
}??>

```

#### Klasa Izdata:

```

<?php
class Izdata extends KonekcijaDB
{
    public $Rblzdavanja;
    public $DatumIzdavanja;
    public $DatumVracanja;
    public $Period;
    public $Clan;
    public $Jedinica;
    public $Bibliotekar;
    public $InventarniBroj;

    public function Izdavanje()
    {
        //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i ostalih promenljivih
        $resultClanIspisan = "";
        $resultOpomena = "";
        $resultRashodovana = "";
        $resultIzdataJedinica = "";
        $result = "";
        $uspehzkbp = "";
        $uspeh = "false";
        //otvaranje konekcije do BP
        $konekcija=$this->OtvoranjeKonekcije();
        //Prilikom izdavanja bibliotecke jedinice
        //clanu treba proveriti:
        //1.da li je clan ispisan
        //2.da li clan ima opomenu
        //3.da li je jedinica rashodovana
        //4.da li je jedinica trenutno izdata
        //U slucaju da je ispunjen bar jedan
        //od ova 4 uslova jedinica se ne sme izdati clanu!
        //Provera da li je clan ispisan
    }
}

```



```

$upit = "SELECT `Broj clana`, `Prezime`, `Ime`, `Telefon`, `Email`, `Broj indeksa`, `Sifra zaposlenog`, `Korisnicka sifra`, `Datum ispisa`, `Opomena` FROM `clan` WHERE `Broj clana`='{$this->Clan}' and `Datum ispisa` IS NOT NULL;";
$resultClanIspisan = mysqli_query($konekcija, $upit);
//Provera da li je clan ima opomenu
$upit = "SELECT `Broj clana`, `Prezime`, `Ime`, `Telefon`, `Email`, `Broj indeksa`, `Sifra zaposlenog`, `Korisnicka sifra`, `Datum ispisa`, `Opomena` FROM `clan` WHERE `Broj clana`='{$this->Clan}' and `Opomena` IS NOT NULL;";
$resultOpomena = mysqli_query($konekcija, $upit);
//Provera da li je jedinica rashodovana ili otpisana
$upit = "SELECT * FROM `biblioteka jedinica` WHERE (`Datum otpisa` IS NOT NULL OR Rashodovana=1) AND `Inventarni broj`='{$this->InventarniBroj}';";
$resultRashodovana = mysqli_query($konekcija, $upit);
//Provera da li je jedinica trenutno izdata
$upit = "SELECT * FROM `izdata` WHERE `Datum vracanja` IS NULL AND `Inventarni broj`='{$this->InventarniBroj}';";
$resultIzdataJedinica = mysqli_query($konekcija, $upit);
//Izdavanje jedinice clanu
if (!$resultClanIspisan && !$resultOpomena && !$resultRashodovana && !$resultIzdataJedinica)
{
    $upit = "INSERT INTO `izdata`(`RB izdavanja`, `Broj clana`, `Inventarni broj`, `ID bibliotekara`, `Datum izdavanja`, `Datum vracanja`, `Period`) VALUES ('{$this->Rblzdavanja', '{$this->Clan}', '{$this->InventarniBroj}', '{$this->Bibliotekar}', '{$this->Datumizdavanja', '{$this->DatumVracanja', '{$this->Period}');";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        $suspeh="true";
    }
}
else
{
    $suspeh="false";
}
//zatvaranje konekcije do BP
$uspehZkbp=$this->ZatvaranjeKonekcije();
return $suspeh;
}

```

```

public function Vracanje()
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehZkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL DML upita za izmenu podataka
    $upit = "UPDATE `izdata` SET `Datum vracanja`='{$this->DatumVracanja}' WHERE `RB izdavanja`='{$this->Rblzdavanja}';";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {

```

```

        $uspeh="true";
    }
    //zatvaranje konekcije do BP
    $uspehzkbp=$this->ZatvaranjeKonekcije();
    return $uspeh;
}

public function KreiranjeOpomena($pBrClana)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehzkbp = "";
    $datum = "";
    $uspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvoranjeKonekcije();
    //formiranje SQL DML upita za izmenu podataka
    //ocitati sistemski datum kao deo teksta opomene
    $datum=date('d.m.Y');
    $upit = "UPDATE `clan` SET `Opomena`='Opomena izdata ".$datum."' WHERE `Broj clana`='".$pBrClana.'";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        {
            $uspeh="true";
            //slanje poruke na email
            //...
        }
    }
    //zatvaranje konekcije do BP
    $uspehzkbp=$this->ZatvaranjeKonekcije();

    return $uspeh;
}

public function PretragaOpomena($pBrClana, $pDatum)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehzkbp = "";
    $uspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvoranjeKonekcije();
    //formiranje SQL upita za izdvajanje podataka
    if (!$pBrClana=="")
    {
        $upit = "SELECT * FROM `clan` WHERE `Broj clana`='".$pBrClana.'";
    }
    //datum opomene se nalazi u delu teksta opomene
    //i ne postoji kao poseban atribut i kolona tabele BP
    if (!$pDatum=="")
    {
        $upit = "SELECT * FROM `clan` WHERE `Opomena` LIKE '%pDatum%';";
    }
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita

```

```

i ispis informacije o gresci*/
if(!$result)
{
    mysqli_error($konekcija);
}
else
{
    $uspeh="true";
}
//zatvaranje konekcije do BP
$uspehzkbp=$this->ZatvaranjeKonekcije();
return $result;
}

public function IzvestajOpomene($pDatum)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehzkbp = "";
    $uspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL upita za izdvajanje podataka
    //datum opomene se nalazi u delu teksta opomene
    //i ne postoji kao poseban atribut i kolona tabele BP
    $upit = "SELECT `bibliotecka jedinica`.*, `izdata`.*, `clan`.*";
    $upit = $upit." FROM (`bibliotecka jedinica`";
    $upit = $upit." INNER JOIN `izdata` ON `izdata`.`Inventarni broj` = `bibliotecka jedinica`.`Inventarni broj`";
    $upit = $upit." INNER JOIN `clan` ON `izdata`.`Broj clana` = `clan`.`Broj clana` WHERE `Opomena` LIKE '%pDatum%'";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        $uspeh="true";
    }
    //zatvaranje konekcije do BP
    $uspehzkbp=$this->ZatvaranjeKonekcije();
    return $result;
}
}??>

```

#### Klasa Clan:

```

<?php
class Clan extends KonekcijaDB
{
    public $BrojClana;
    public $Prezime;
    public $Ime;
    public $Telefon;
    public $Email;
    public $BrojIndeksa;
    public $SifraZaposlenog;
    public $KorisnickaSifra;
    public $DatumIspisa;
    public $Opomena;
}

```

```

public function Upis()
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i ostalih promenljivih
    $result = "";
    $suspehzkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL DML upita za unos podataka
    $upit = "INSERT INTO `clan`(`Broj clana`, `Prezime`, `Ime`, `Telefon`, `Email`, `Broj indeksa`, `Sifra zaposlenog`, `Korisnicka sifra`,
`Datum ispisa`, `opomena`) VALUES ('$this->BrojClana','$this->Prezime','$this->Ime','$this->Telefon','$this->Email','$this-
>BrojIndeksa','$this->SifraZaposlenog','$this->KorisnickaSifra','$this->DatumIspisa','$this->Opomena');"
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        $suspeh="true";
    }
    //zatvaranje konekcije do BP
    $suspehzkbp=$this->ZatvaranjeKonekcije();
    return $suspeh;
}

public function Izmena($pBrClana)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $suspehzkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL DML upita za izmenu podataka
    $upit = "UPDATE `clan` SET `Broj clana`='$this->BrojClana',`Prezime`='$this->Prezime',`Ime`='$this->Ime',`Telefon`='$this-
>Telefon',`Email`='$this->Email',`Broj indeksa`='$this->BrojIndeksa',`Sifra zaposlenog`='$this->SifraZaposlenog',`Korisnicka
sifra`='$this->KorisnickaSifra',`Datum ispisa`='$this->DatumIspisa',`opomena`='$this->Opomena' WHERE `Broj clana`='$pBrClana'";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        $suspeh="true";
    }
    //zatvaranje konekcije do BP
    $suspehzkbp=$this->ZatvaranjeKonekcije();
    return $suspeh;
}

public function Ispis($pBrClana)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";

```

```

$uspehzbkp = "";
$uspeh = "false";
//otvaranje konekcije do BP
$koneksiya=$this->OtvoranjeKonekcije();
//formiranje SQL DML upita za izmenu podataka zbog ispisa
$upit = "UPDATE `clan` SET `Datum ispisa`='<div data-bbox='\"$this->DatumIspisa' WHERE `Broj clana`='<div data-bbox='\"$pBrClana';";
//izvršavanje SQL upita
$result = mysqli_query($koneksiya, $upit);
/*provera rezultata izvršavanja SQL upita
i ispis informacije o gresci*/
if(!$result)
{
    mysqli_error($koneksiya);
}
else
{
    {
        $uspeh="true";
    }
}
//zatvaranje konekcije do BP
$uspehzbkp=$this->ZatvaranjeKonekcije();
return $uspeh;
}

```

```

public function Pretraga($pPrezime)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehzbkp = "";
    $uspeh = "false";
    //otvaranje konekcije do BP
    $koneksiya=$this->OtvoranjeKonekcije();
    //formiranje SQL upita za izdvajanje podataka
    $upit = "SELECT * FROM `clan` WHERE `Prezime` LIKE '%pPrezime%' ORDER BY `Prezime`, `Ime`";
    //izvršavanje SQL upita
    $result = mysqli_query($koneksiya, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        {
            mysqli_error($koneksiya);
        }
    }
    else
    {
        {
            $uspeh="true";
        }
    }
    //zatvaranje konekcije do BP
    $uspehzbkp=$this->ZatvaranjeKonekcije();
    return $result;
}

```

```

public function Prijava()
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehzbkp = "";
    $uspeh = "false";
    //otvaranje konekcije do BP
    $koneksiya=$this->OtvoranjeKonekcije();
    //formiranje SQL upita za izdvajanje podataka
    $upit = "SELECT * FROM `clan` WHERE `clan`.`Email`='<div data-bbox='\"$this->Email' AND `clan`.`Korisnicka sifra`='<div data-bbox='\"$this->KorisnickaSifra';";
    //izvršavanje SQL upita
    $result = mysqli_query($koneksiya, $upit);

```

```

/*provera rezultata izvršavanja SQL upita
i ispis informacije o gresci*/
if(!$result)
{
    mysqli_error($konekcija);
}
else
{
    $suspeh="true";
}
//zatvaranje konekcije do BP
$uspehzkbp=$this->ZatvaranjeKonekcije();
return $result;
}

public function Odjava()
{
    //Odjava korisnika
    $odjavljen="true";
    return $odjavljen;
}

public function PotvrdaORazduzenosti($pBrIndeksa)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehzkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL upita za izdvajanje podataka
    $upit = "SELECT `clan`.*, `bibliotecka jedinica`.*, `izdata`.* FROM (`clan` INNER JOIN `izdata` ON `izdata`.`Broj clana` = `clan`.`Broj clana`) INNER JOIN `bibliotecka jedinica` ON `izdata`.`Inventarni broj`=`bibliotecka jedinica`.`Inventarni broj` WHERE `clan`.`Broj indeksa`='`$pBrIndeksa`'";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        $suspeh="true";
    }
    //zatvaranje konekcije do BP
    $uspehzkbp=$this->ZatvaranjeKonekcije();
    return $result;
}

public function ZaduzenjaClana($pBrClana)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehzkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL upita za izdvajanje podataka

```

```

        $upit = "SELECT `clan`.*, `bibliotecka jedinica`.*, `izdata`.* FROM (`clan` INNER JOIN `izdata` ON `izdata`.`Broj clana` = `clan`.`Broj clana`) INNER JOIN `bibliotecka jedinica` ON `izdata`.`Inventarni broj`=`bibliotecka jedinica`.`Inventarni broj` WHERE `clan`.`Broj clana`='{$pBrClana}";
        //izvršavanje SQL upita
        $result = mysqli_query($konekcija, $upit);
        /*provera rezultata izvršavanja SQL upita
        i ispis informacije o gresci*/
        if(!$result)
        {
            mysqli_error($konekcija);
        }
        else
        {
            $suspeh="true";
        }
        //zatvaranje konekcije do BP
        $suspehzkbp=$this->ZatvaranjeKonekcije();
        return $result;
    }
}??>

```

#### Klasa Bibliotekar:

```

<?php
class Bibliotekar extends KonekcijaDB
{
    public $IdBibliotekara;
    public $Prezime;
    public $Ime;
    public $Email;
    public $KorisnickaSifra;

    public function Dodavanje()
    {
        //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i ostalih promenljivih
        $result = "";
        $suspehzkbp = "";
        $suspeh = "false";
        //otvaranje konekcije do BP
        $konekcija=$this->OtvaranjeKonekcije();
        //formiranje SQL DML upita za unos podataka
        $upit = "INSERT INTO `bibliotekar` (`ID bibliotekara`, `Prezime`, `Ime`, `Email`, `Korisnicka sifra`) VALUES ('$this->IdBibliotekara','$this->Prezime','$this->Ime','$this->Email','$this->KorisnickaSifra')";
        //izvršavanje SQL upita
        $result = mysqli_query($konekcija, $upit);
        /*provera rezultata izvršavanja SQL upita
        i ispis informacije o gresci*/
        if(!$result)
        {
            mysqli_error($konekcija);
        }
        else
        {
            $suspeh="true";
        }
        //zatvaranje konekcije do BP
        $suspehzkbp=$this->ZatvaranjeKonekcije();
        return $suspeh;
    }

    public function Izmena($pIdBibliotekara)
    {
        //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    }
}

```

```

$result = "";
$uspehzbkp = "";
$uspeh = "false";
//otvaranje konekcije do BP
$konekcija=$this->OtvaranjeKonekcije();
//formiranje SQL DML upita za izmenu podataka
$upit = "UPDATE `bibliotekar` SET `ID bibliotekara`='$this->IdBibliotekara', `Prezime`='$this->Prezime', `Ime`='$this->Ime', `Email`='$this->Email', `Korisnicka sifra`='$this->KorisnickaSifra' WHERE `ID bibliotekara`='pIDbibliotekara'";
//izvršavanje SQL upita
$result = mysqli_query($konekcija, $upit);
/*provera rezultata izvršavanja SQL upita
i ispis informacije o gresci*/
if(!$result)
{
    mysqli_error($konekcija);
}
else
{
    $uspeh="true";
}
//zatvaranje konekcije do BP
$uspehzbkp=$this->ZatvaranjeKonekcije();
return $uspeh;
}

```

```

public function PrikazSvih()
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehzbkp = "";
    $uspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL upita za izdvajanje podataka
    $upit = "SELECT * FROM `bibliotekar` ORDER BY Prezime ASC;";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        $uspeh="true";
    }
    //zatvaranje konekcije do BP
    $uspehzbkp=$this->ZatvaranjeKonekcije();
    return $result;
}

```

```

public function Prijava()
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehzbkp = "";
    $uspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL upita za izdvajanje podataka

```



```

        $upit = "SELECT `ID bibliotekara`, `Prezime`, `Ime`, `Email`, `Korisnicka sifra` FROM `bibliotekar` WHERE
`bibliotekar`.`Email`='$this->Email' AND `bibliotekar`.`Korisnicka sifra`='$this->KorisnickaSifra';";
        //izvršavanje SQL upita
        $result = mysqli_query($konekcija, $upit);
        /*provera rezultata izvršavanja SQL upita
        i ispis informacije o gresci*/
        if(!$result)
        {
            mysqli_error($konekcija);
        }
        else
        {
            $suspeh="true";
        }
        //zatvaranje konekcije do BP
        $suspehzkbp=$this->ZatvaranjeKonekcije();
        return $result;
    }

    public function Odjava()
    {
        //Odjava korisnika
        $odjavljen="true";
        return $odjavljen;
    }
}??>

```

#### **Klasa Biblioteka jedinica:**

```

<?php
class BibliotekaJedinica extends KonekcijaDB
{
    public $InventarniBroj;
    public $Naziv;
    public $Izdavac;
    public $Godinalzdanja;
    public $Isbn;
    public $Stanje;
    public $Opis;
    public $Rashodovana;
    public $DatumOtpisa;
    public $Jezik;
    public $VrstaJedinice;

    public function Unos()
    {
        //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i ostalih promenljivih
        $result = "";
        $suspehzkbp = "";
        $suspeh = "false";
        //otvaranje konekcije do BP
        $konekcija=$this->OtvaranjeKonekcije();
        //formiranje SQL DML upita za unos podataka
        $upit = " INSERT INTO `biblioteka jedinica`(`Inventarni broj`, `ID jezika`, `ID vrste jedinice`, `Naziv`, `Izdavac`, `Godina izdanja`,
`ISBN`, `Stanje`, `Opis`, `Rashodovana`, `Datum otpisa`) VALUES ('$this->InventarniBroj','$this->Jezik','$this->VrstaJedinice','$this-
>Naziv','$this->Izdavac','$this->Godinalzdanja','$this->Isbn','$this->Stanje','$this->Opis','$this->Rashodovana','$this-
>DatumOtpisa');";
        //izvršavanje SQL upita
        $result = mysqli_query($konekcija, $upit);
        /*provera rezultata izvršavanja SQL upita
        i ispis informacije o gresci*/
        if(!$result)
        {

```

```

        mysqli_error($konekcija);
    }
    else
    {
        $suspeh="true";
    }
    //zatvaranje konekcije do BP
    $suspehzkbp=$this->ZatvaranjeKonekcije();
    return $suspeh;
}

```

```

public function Izmena($pInvBroj)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $suspehzkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL DML upita za izmenu podataka
    $upit = "UPDATE `bibliotecka jedinica` SET `Inventarni broj`='".$this->InventarniBroj', `ID jezika`='".$this->Jezik', `ID vrste
jedinice`='".$this->VrstaJedinice', `Naziv`='".$this->Naziv', `Izdavac`='".$this->Izdavac', `Godina izdavanja`='".$this->GodinaIzdavanja', `ISBN`='".$this->Isbn', `Stanje`='".$this->Stanje', `Opis`='".$this->Opis', `Rashodovana`='".$this->Rashodovana', `Datum
otpisa`='".$this->DatumOtpisa' WHERE `Inventarni broj`='".$pInvBroj'";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        $suspeh="true";
    }
    //zatvaranje konekcije do BP
    $suspehzkbp=$this->ZatvaranjeKonekcije();
    return $suspeh;
}

```

```

public function Brisanje($pInvBroj)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $suspehzkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL DML upita za brisanje podataka
    $upit = "DELETE FROM `bibliotecka jedinica` WHERE `Inventarni broj`='".$pInvBroj'";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        $suspeh="true";
    }
}

```

```

    }
    //zatvaranje konekcije do BP
    $suspehzkbp=$this->ZatvaranjeKonekcije();
    return $suspeh;
}

public function Pretraga($pNazivJedinice, $pVrstaJedinice, $pAutorPrezime, $pAutorIme, $pGodina, $pJezik)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $suspehzkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL upita za izdavanje podataka u zavisnosti od kriterijuma pretrage
    //Sosnovniupit="SELECT Prezime, Ime, `bibliotecka jedinica`.`Inventarni broj` AS InvBr, `Jezik`.`Naziv jezika`, `Jezik`.`ID jezika`,
    `bibliotecka jedinica`.`ID vrste jedinice`, `Vrsta jedinice`.`Naziv vrste jedinice`, `Naziv`, `Izdavac`, `Godina izdavanja`, `ISBN`, `Stanje`,
    `Opis`, `Rashodovana`, `Datum otpisa` FROM `Bibliotecka jedinica`, `Vrsta jedinice`, `Autor`, `Jezik`, `Napisao` WHERE `Autor`.`ID
    autora`=`Napisao`.`ID autora` AND `bibliotecka jedinica`.`Inventarni broj`=`Napisao`.`Inventarni broj` AND `bibliotecka jedinica`.`ID
    jezika`=`Jezik`.`ID jezika` AND `bibliotecka jedinica`.`ID vrste jedinice`=`Vrsta jedinice`.`ID vrste jedinice`;
    $osnovniupit="SELECT `bibliotecka jedinica`.`Inventarni broj` AS InvBr, `bibliotecka jedinica`.`Rashodovana`, `bibliotecka
    jedinica`.`ID vrste jedinice`, `bibliotecka jedinica`.`ID jezika`, `bibliotecka jedinica`.`Naziv`, `vrsta jedinice`.`Naziv vrste jedinice`,
    `Jezik`.`Naziv jezika`, `autor`.`Prezime`, `autor`.`Ime`, `napisao`.`ID autora`, `bibliotecka jedinica`.`Godina izdavanja`;
    $osnovniupit=$osnovniupit."FROM `bibliotecka jedinica`;
    $osnovniupit=$osnovniupit."LEFT JOIN `vrsta jedinice` ON `bibliotecka jedinica`.`ID vrste jedinice` = `vrsta jedinice`.`ID vrste
    jedinice`;
    $osnovniupit=$osnovniupit."LEFT JOIN `jezik` ON `bibliotecka jedinica`.`ID jezika` = `jezik`.`ID jezika`;
    $osnovniupit=$osnovniupit."LEFT JOIN `napisao` ON `bibliotecka jedinica`.`Inventarni broj` = `napisao`.`Inventarni broj`;
    $osnovniupit=$osnovniupit."LEFT JOIN `autor` ON `napisao`.`ID autora` = `autor`.`ID autora`;

    if (!$pNazivJedinice=="")
    {
        $upit = $osnovniupit." WHERE `Naziv` LIKE '%$pNazivJedinice%' ORDER BY `Naziv` ASC;";
    }
    if (!$pVrstaJedinice=="")
    {
        $upit = $osnovniupit." WHERE `Vrsta jedinice`.`Naziv vrste jedinice`='$pVrstaJedinice' ORDER BY `Naziv` ASC;";
    }
    if (!$pAutorPrezime=="")
    {
        $upit = $osnovniupit." WHERE `Autor`.`Prezime` LIKE '%$pAutorPrezime' ORDER BY `Naziv` ASC;";
    }
    if (!$pAutorIme=="")
    {
        $upit = $osnovniupit." WHERE `Autor`.`Ime` LIKE '%$pAutorIme' ORDER BY `Naziv` ASC;";
    }
    if (!$pGodina=="")
    {
        $upit = $osnovniupit." WHERE `Godina izdavanja`='$pGodina' ORDER BY `Naziv` ASC;";
    }
    if (!$pJezik=="")
    {
        $upit = $osnovniupit." WHERE `Jezik`.`Naziv jezika`='$pJezik' ORDER BY `Naziv` ASC;";
    }
    //Ništa nije izabrano za kriterijum pretrage
    if ($pNazivJedinice==" && $pJezik==" && $pGodina==" && $pAutorIme==" && $pVrstaJedinice==" && $pAutorPrezime=="")
    {
        $upit = $osnovniupit." ORDER BY `Naziv` ASC;";
    }
    //echo $upit;
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);

```

```

/*provera rezultata izvršavanja SQL upita
i ispis informacije o gresci*/
if(!$result)
{
    mysqli_error($konekcija);
}
//zatvaranje konekcije do BP
$uspehzkbp=$this->ZatvaranjeKonekcije();
return $result;
}

public function Rashoduj()
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehzkbp = "";
    $uspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL DML upita za izmenu podataka
    $upit = "UPDATE `bibliotecka jedinica` SET `Rashodovana`='{$this->Rashodovana}', `Datum otpisa`='{$this->DatumOtpisa}' WHERE
`Inventarni broj`='{$this->InventarniBroj}'";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        $uspeh="true";
    }
    //zatvaranje konekcije do BP
    $uspehzkbp=$this->ZatvaranjeKonekcije();
    return $uspeh;
}

public function Ucitaj($pInvBroj)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $uspehzkbp = "";
    $uspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    $upit="SELECT * FROM `bibliotecka jedinica` WHERE `Inventarni broj`='{$pInvBroj}'";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    //zatvaranje konekcije do BP
    $uspehzkbp=$this->ZatvaranjeKonekcije();
    return $result;
}
}

```

**Klasa Autor:**

```

<?php
class Autor extends KonekcijaDB
{
    public $IdAutora;
    public $Prezime;
    public $Ime;
    public $Napomena;

    public function Dodavanje()
    {
        //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i ostalih promenljivih
        $result = "";
        $suspehzkbp = "";
        $suspeh = "false";
        //otvaranje konekcije do BP
        $konekcija=$this->OtvaranjeKonekcije();
        //formiranje SQL DML upita za unos podataka
        $upit = "INSERT INTO `autor`(`ID autora`, `Prezime`, `Ime`, `Napomena`) VALUES ('$this->IdAutora','$this->Prezime','$this-
>Ime','$this->Napomena');";
        //izvršavanje SQL upita
        $result = mysqli_query($konekcija, $upit);
        /*provera rezultata izvršavanja SQL upita
        i ispis informacije o gresci*/
        if(!$result)
        {
            mysqli_error($konekcija);
        }
        else
        {
            $suspeh="true";
        }
        //zatvaranje konekcije do BP
        $suspehzkbp=$this->ZatvaranjeKonekcije();
        return $suspeh;
    }

    public function Izmena($pIDAutora)
    {
        //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
        $result = "";
        $suspehzkbp = "";
        $suspeh = "false";
        //otvaranje konekcije do BP
        $konekcija=$this->OtvaranjeKonekcije();
        //formiranje SQL DML upita za izmenu podataka
        $upit = "UPDATE `autor` SET `ID autora`='$this->IdAutora',`Prezime`='$this->Prezime',`Ime`='$this->Ime',`Napomena`='$this-
>Napomena' WHERE `ID autora`='$pIDAutora';";
        //izvršavanje SQL upita
        $result = mysqli_query($konekcija, $upit);
        /*provera rezultata izvršavanja SQL upita
        i ispis informacije o gresci*/
        if(!$result)
        {
            mysqli_error($konekcija);
        }
        else
        {
            $suspeh="true";
        }
        //zatvaranje konekcije do BP
        $suspehzkbp=$this->ZatvaranjeKonekcije();
        return $suspeh;
    }
}

```

```

}

public function Brisanje($pIDAutora)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $suspehzkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL DML upita za brisanje podataka
    $upit = "DELETE FROM `autor` WHERE `ID autora`='pIDAutora'";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        {
            $suspeh="true";
        }
    }
    //zatvaranje konekcije do BP
    $suspehzkbp=$this->ZatvaranjeKonekcije();
    return $suspeh;
}

public function Pretraga()
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela) i promenljive uspeh
    $result = "";
    $suspehzkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL upita za izdvajanje podataka
    $upit = "SELECT DISTINCT `ID autora`, `Prezime`, `Ime`, `Napomena` FROM `autor` WHERE `Prezime`='{$this->Prezime}' OR
`Prezime` LIKE '%{$this->Prezime}%' ORDER BY `Prezime`, `Ime`";
    //izvršavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvršavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
    }
    else
    {
        {
            $suspeh="true";
        }
    }
    //zatvaranje konekcije do BP
    $suspehzkbp=$this->ZatvaranjeKonekcije();
    return $result;
}
}??>

```

## Implementacija Prezentionog sloja softvera/korisničkog interfejsa

**Početna stranica – index.php** se sastoji od html oznaka u koje je umetnuta php naredba za uključivanje datoteka sa sadržajem zaglavlja strane (header), menija u gornjem i donjem delu stranice, kao i html oznaka za donji deo stranice (footer). Ova stranica poseduje odeljke, set karaktera je UTF-8, a korsiti stil iz css datoteke. Sadržaj je formiran Bootstrap radnim okvirom, korišćenjem klasa za odeljke u centralnom delu stranice.

```
<!DOCTYPE HTML>
<html>
  <?php include "head.html"; ?>
  <body>
    <div class="fh5co-loader"></div>
    <!-- Meni -->
    <?php include "meniupindex.html"; ?>
    <!-- Sredina stranice -->
    <div id="fh5co-course-categories">
      <div class="container">
        <div class="row animate-box">
          <div class="col-md-6 col-md-offset-3 text-center fh5co-heading">
            <h2>BIBLIO v1.0</h2>
            <p>Softver za podršku rada fakultetske biblioteke</p>
          </div>
        </div>
      </div>
    </div>
    <!-- Meni -->
    <?php include "menidownindex.html"; ?>
    <!-- Footer stranice -->
    <?php include "footer.html"; ?>
  </body>
</html>
```

**Meni – meniupbibliotekar.html** je urađen pomoću HTML oznaka i Bootstrap klasa kao neuređena lista preko: <ul> i </ul>, kao i elementima liste, koji predstavljaju stavke menija pomoću oznaka <li> i </li>. Svaka stavka menija je realizovana preko hiperlinka, tj. oznake <a href="strana.php"> povezana sa odgovarajućom web stranicom koja se učitava kada korisnik izabere stavku menija.

```
<div id="page">
  <nav class="fh5co-nav" role="navigation">
    <div class="top">
      <div class="container">
        <div class="row">
          <div class="col-xs-12 text-right">
            <p class="site">www.tfzr.uns.ac.rs/sluzbe/biblioteka</p>
            <p class="num">Status korisnika: bibliotekar</p>
            <ul class="fh5co-social">
              <li><a href="#"><i class="icon-facebook2"></i></a></li>
              <li><a href="#"><i class="icon-twitter2"></i></a></li>
              <li><a href="#"><i class="icon-github"></i></a></li>
```

```

        </ul>
    </div>
</div>
</div>
<div class="top-menu">
    <div class="container">
        <div class="row">
            <div class="col-xs-2">
                <div id="fh5co-logo"><a href="pocetna.php"><i class="icon-
study"></i>Biblio<span></span></a></div>
            </div>
            <div class="col-xs-10 text-right menu-1">
                <ul>
                    <li class="active"><a href="pocetna.php">Početna</a></li>
                    <li class="has-dropdown">
                        <a href="#">Osnovni podaci</a>
                        <ul class="dropdown">
                            <li><a href="#">Bibliotekari</a></li>
                            <li><a href="jezici.php">Jezici</a></li>
                            <li><a href="vrstejedinica.php">Vrste jedinica</a></li>
                            <li><a href="#">Autori</a></li>
                        </ul>
                    </li>
                    <li class="has-dropdown">
                        <a href="#">Rad bibliotekara</a>
                        <ul class="dropdown">
                            <li><a href="novajedinica.php">Unos nove jedinice</a></li>
                            <li><a href="#">Upis člana</a></li>
                            <li><a href="#">Ispis člana</a></li>
                            <li><a href="izdavanje.php">Izdavanje jedinica</a></li>
                            <li><a href="#">Vraćanje jedinica</a></li>
                            <li><a href="#">Formiranje opomena</a></li>
                        </ul>
                    </li>
                    <li class="has-dropdown">
                        <a href="#">Pretrage</a>
                        <ul class="dropdown">
                            <li><a href="pretragaфона.php">Fonda</a></li>
                            <li><a href="#">Autora</a></li>
                            <li><a href="#">Članova</a></li>
                            <li><a href="prikazzaduzenja.php">Zaduženja</a></li>
                            <li><a href="#">Opomena</a></li>
                        </ul>
                    </li>
                    <li class="has-dropdown">
                        <a href="#">Izveštaji</a>

```



```

        <ul class="dropdown">
            <li><a href="spisakopomena.php">Spisak opomena</a></li>
            <li><a href="#">Potvrda o razduženosti</a></li>
        </ul>
    </li>
    <li><a href="aboutinfo.php">O nama</a></li>
    <li class="btn-cta"><a href="odjava.php"><span>Odjava</span></a></li>
</ul>
</div>
</div>
</div>
</div>
</nav>

```

**Stranica, tj. forma za unos podataka – novajedinica.php.** Forma za unos podataka – realizovana je preko FORM HTML odeljka u okviru kojeg se nalazi tabela sa nekoliko redova <tr> i poljima <td> za unos podataka pomoću <input type> oznaka. U Action atributu forme je navedena PHP stranica na kojoj će se izvršiti snimanje podataka u bazu. Metod prenosa podataka je POST.

```

<h4>UNOS NOVE BIBLIOTEČKE JEDINICE</h4>
    <!-- Forma za unos podataka -->
    <form action="novajedinicaunos.php" method="GET">
        <table align="center">
            <tr style="float:center; clear:left; margin-top:10px;"><td>Inventarni broj*</td>
                <td style="float:left; clear:right; margin-top:10px;"><input type="text"
name="InventarniBroj" autofocus onKeyPress="if(this.value.length==10) return false;" size=10
placeholder="Upišite inventarni broj" required tabindex=1></td>
            </tr>
            <tr style="float:center; clear:left; margin-top:10px;"><td>Jezik*</td>
                <td style="float:left; clear:right; margin-top:10px;">
                    <select class="form-control" id="Jezik" name="Jezik" tabindex=2 required>
                        <option value="">izaberite jezik...</option>
                    <?php
                        //Pocetne vrednosti promenljivih
                        $result="";
                        $brredova=0;
                        //Kreiranje objekta Jezik
                        require 'klase/KonekcijaDB.php';
                        require 'klase/Jezik.php';
                        $objJezik = new Jezik();
                        //Izvršavanje operacije za prikaz svih jezika
                        $result=$objJezik->PrikazSvih();
                        $brredova = mysqli_num_rows($result);
                        if ($brredova>0)
                        {
                            $i=0;
                            $red=0;
                            while($red=mysqli_fetch_array($result))
                            {

```

```

        $i++;
        $niz1[$i]=$red['Naziv jezika'];
        $niz2[$i]=$red['ID jezika'];
        echo "<option value='$niz2[$i]','$niz1[$i]</option>";
    }
}
$objJezik = null;
?>
</select>
</td>
</tr>
<tr style="float:center; clear:left; margin-top:10px;"><td>Vrsta jedinice*</td>
<td style="float:left; clear:right; margin-top:10px;">
<select class="form-control" id="VrstaJedinice" name="VrstaJedinice"
tabindex=3 required>
<option value="">izaberite vrstu jedinice...</option>
<?php
    //Pocetne vrednosti promenljivih
    $result="";
    $brredova=0;
    //Kreiranje objekta Jedinica
    require 'klase/VrstaJedinice.php';
    $objJedinica = new VrstaJedinice();
    //Izvršavanje operacije za prikaz svih vrsta jedinica
    $result=$objJedinica->PrikazSvih();
    $brredova = mysqli_num_rows($result);
    if ($brredova>0)
    {
        $i=0;
        $red=0;
        while($red=mysqli_fetch_array($result))
        {
            $i++;
            $niz1[$i]=$red['Naziv vrste jedinice'];
            $niz2[$i]=$red['ID vrste jedinice'];
            echo "<option value='$niz2[$i]','$niz1[$i]</option>";
        }
    }
    //Uništavanje objekta
    $objJedinica = null;
?>
</select>
</td>
</tr>
<tr style="float:center; clear:left; margin-top:10px;"><td>Naziv bibliotečke
jedinice*</td>

```

```

        <td style="float:left; clear:right; margin-top:10px;"><input type="text"
name="NazivJedinice" onKeyPress="if(this.value.length==100) return false;" size=30 required
placeholder="Upišite naziv jedinice" tabindex=4></td>
    </tr>
    <tr style="float:center; clear:left; margin-top:10px;"><td>Izdavač*</td>
        <td style="float:left; clear:right; margin-top:10px;"><input type="text"
name="Izdavac" onKeyPress="if(this.value.length==50) return false;" size=30 required placeholder="Upišite
izdavača" tabindex=5></td>
    </tr>
    <tr style="float:center; clear:left; margin-top:10px;"><td>Godina izdavanja*</td>
        <td style="float:left; clear:right; margin-top:10px;"><input type="number"
name="GodinaIzdavanja" onKeyPress="if(this.value.length==4) return false;" size=10 required
placeholder="Upišite godinu izdavanja" tabindex=6></td>
    </tr>
    <tr style="float:center; clear:left; margin-top:10px;"><td>ISBN</td>
        <td style="float:left; clear:right; margin-top:10px;"><input type="text"
name="Isbn" onKeyPress="if(this.value.length==20) return false;" size=20 placeholder="Upišite ISBN broj"
tabindex=7></td>
    </tr>
    <tr style="float:center; clear:left; margin-top:10px;"><td>Stanje</td>
        <td style="float:left; clear:right; margin-top:10px;">
            <input type="radio" id="ocuvana" name="Stanje" value="ocuvana" tabindex=8>
<label for="ocuvana">očuvana</label> <br/>
            <input type="radio" id="ostecena" name="Stanje" value="ostecena" tabindex=9>
<label for="ostecena">oštećena</label>
        </td>
    </tr>
    <tr style="float:center; clear:left; margin-top:10px;"><td>Opis</td>
        <td style="float:left; clear:right; margin-top:10px;"><input type="text"
name="Opis" onKeyPress="if(this.value.length==1000) return false;" size=50 placeholder="Upišite opis
jedinice" tabindex=10></td>
    </tr>
    <tr style="float:center; clear:right; margin-top:20px;">
        <td></td>
        <td style="float:left; clear:right; margin-top:10px;">
            <input type="submit" name="snimi" value="Snimi novu jedinicu"
tabindex=11>
            <button type="reset" name="ponisti" tabindex=11>Poništi</button>
        </td>
    </tr>
</table>
</form>

```

**Snimanje unetih podataka u bazu – novajedinicaunos.php.** Unos podataka u bazu je prikazan na primeru stranice za unos bibliotekskih jedinica. Preuzimaju se vrednosti promenljivih, metodom POST sa forme za unos podataka, kreira se objekat, poziva se metoda za unos podataka u MySQL bazu. U zavisnosti od uspešnosti izvršavanja metode, prikazuje se poruka korisniku pomoću echo PHP naredbe. Na kraju stranice se uništava objekat.

<h4>UNOS NOVE BIBLIOTEČKE JEDINICE</h4>

```
<?php
    //Preuzimanje podataka iz forme za unos podataka sa stranice za unos nove jedinice
    $InventarniBroj=$_GET['InventarniBroj'];
    $Jezik=$_GET['Jezik'];
    $VrstaJedinice=$_GET['VrstaJedinice'];
    $NazivJedinice=$_GET['NazivJedinice'];
    $Izdavac=$_GET['Izdavac'];
    $GodinaIzdavanja=$_GET['GodinaIzdavanja'];
    $Isbn=$_GET['Isbn'];
    $Stanje=$_GET['Stanje'];
    $Opis=$_GET['Opis'];
    //Vrednosti koje se ne unose na prethodnoj veb stranici
    $Rashodovana="0";
    //$DatumOtpisa=NULL;
    //Pocetna vrednost resultset-a
    $rezultat="";

    //Kreiranje objekta BibliotekaJedinica
    require 'klase/KonekcijaDB.php';
    require 'klase/BibliotekaJedinica.php';
    $objJedinica = new BibliotekaJedinica();
    //Dodela vrednosti atributima objekta
    $objJedinica->InventarniBroj=$InventarniBroj;
    $objJedinica->Naziv=$NazivJedinice;
    $objJedinica->Izdavac=$Izdavac;
    $objJedinica->GodinaIzdavanja=$GodinaIzdavanja;
    $objJedinica->Isbn=$Isbn;
    $objJedinica->Stanje=$Stanje;
    $objJedinica->Opis=$Opis;
    $objJedinica->Rashodovana=$Rashodovana;
    //$objJedinica->DatumOtpisa=$DatumOtpisa;
    $objJedinica->Jezik=$Jezik;
    $objJedinica->VrstaJedinice=$VrstaJedinice;
    //Izvršavanje operacije za unos podataka u BP
    $rezultat=$objJedinica->Unos();

    //Provera uspesnosti i prikaz poruka
    if ($rezultat=="true")
    {
        echo "Biblioteka jedinica ".$NazivJedinice." je uspešno upisana u bazu
podataka!";
    }
    else
    {
        echo "Biblioteka jedinica ".$NazivJedinice." nije upisana u bazu podataka! Došlo
je do greške... proverite podatke unete na prethodnoj stranici.";
        echo "<br/>";
    }
}
```

```

    }

    //Uništavanje objekta Jezik
    $objJedinica=null;

    //Link za povratak na novi unos
    ?>

```

**Stranica za izmenu podataka – jeziciizmena.php.** Ova stranica je povezana sa stranicom za pretragu ili prikaz podataka, na kojoj se pritiska taster za izmenu u redu tabele gde se želi izvršiti izmena. Pored tastera je forma sa tasterom i promenljivama koje se prenose sledećoj stranici sa formom za izmenu. Iz POST niza se uzimaju vrednosti za izmenu, dodeljuju se odgovarajućim promenljivama, kreira se objekat, poziva se metoda za izmenu podataka u MySQL bazi. U zavisnosti od uspešnosti izvršavanja metode, prikazuje se poruka korisniku pomoću echo PHP naredbe. Na kraju se uništava objekat.

```

<h4>JEZICI - IZMENA</h4>

    <?php
        //Preuzimanje podataka iz forme za unos podataka sa prethodne php strane
        $ID=$_POST['id'];
        $Naziv=$_POST['naziv'];

    ?>

    <!-- Forma za unos/izmenu podataka -->
    <form action="jeziciizmenasnimi.php" method="POST">
        <table align="center">
            <tr style="float:center; clear:left; margin-top:10px;"><td>ID Jezika*</td>
                <td style="float:left; clear:right; margin-top:10px;"><input type="number"
name="IdJezika" value="<?php echo $ID;?" maxlenth=3 size=10 required tabindex=1></td>
            </tr>
            <tr style="float:center; clear:left; margin-top:10px;"><td>Naziv Jezika*</td>
                <td style="float:left; clear:right; margin-top:10px;"><input type="text"
name="NazivJezika" value="<?php echo $Naziv;?" maxlenth=30 size=20 required tabindex=2></td>
            </tr>
            <tr style="float:center; clear:right; margin-top:20px;">
                <td></td>
                <td style="float:left; clear:right; margin-top:10px;">
                    <input type='hidden' name='stariid' value="<?php echo $ID;?">
                    <input type="submit" name="snimi" value="Snimi izmene" tabindex=3>
                    <button type="reset" name="ponisti" tabindex=4>Odustani</button>
                </td>
            </tr>
        </table>
    </form>

```

**Stranica za brisanje podataka – jeziciizmenasnimi.php.** Takođe, povezano sa stranicom za pretragu podataka, na kojoj se pritiska taster za brisanje u redu tabele gde se želi obrisati podatak. Pored tastera je forma sa tasterom i promenljivama koje se prenose sledećoj stranici za brisanje POST metodom. Iz POST niza se uzimaju vrednosti za izmenu, dodeljuju se odgovarajućim promenljivama, kreira se objekat, poziva se metoda za izmenu podataka u MySQL bazi. U zavisnosti od uspešnosti izvršavanja metode, prikazuje se poruka korisniku pomoću echo PHP naredbe. Na kraju se uništava objekat.

```

<h4>JEZICI - IZMENA</h4>

```

```

<?php
    //Preuzimanje podataka iz forme za izmenu podataka sa prethodne php strane
    $ID=$_POST['IdJezika'];
    $Naziv=$_POST['NazivJezika'];
    $StariID=$_POST['stariid'];
    $rezultat="";

    //Kreiranje objekta Jezik
    require 'klase/KonekcijaDB.php';
    require 'klase/Jezik.php';
    $objJezik = new Jezik();
    //Izvršavanje operacije za unos podataka u BP
    $objJezik->setIDJezika($ID);
    $objJezik->setNazivJezika($Naziv);
    //Izvršavanje operacije za unos podataka u BP
    $rezultat=$objJezik->Izmena($StariID);
    //Provera uspesnosti i prikaz poruka
    if ($rezultat)
    {
        echo "Jezik ".$Naziv." je uspešno promenjen u bazi podataka!";
    }
    else
    {
        echo "Jezik ".$Naziv." nije promenjen u bazi podataka! Došlo je do greške...
        proverite podatke promenjene na prethodnoj stranici.";
        echo "<br/>";
    }

    //Uništavanje objekta Vrsta bibliotecke jedinice
    $objVrstaJedinice=null;

    //Link za povratak na unos i prikaz
?>

```

**Pretraga fonda, kriterijum pretrage – pretragafonda.php.** Implementira tako što se prvo kreira forma za unos kriterijuma pretrage. Metodom POST se uneti podaci prosleđuju drugoj stranici za prikaz, na kojoj se preuzima kriterijum pretrage iz POST metode, referencira se klasa u kojoj se nalaze metode za rad sa bazom podataka, tj. navodi relativna putanja do fajla u kom se nalazi klasa, kreira se objekat, poziva se metoda za pretragu podataka u MySQL bazi. Kreira se result-set sa podacima iz baze, prebrojava koliko ima redova u result-setu (funkcijom `mysqli_num_rows`). Ukoliko je broj redova veći od jedan, formira se tabela koja prikazuje učitane podatke iz baze podataka. U tabeli se nalaze forma, promenljiva sa vrednostima i taster za pokretanje brisanja, izmene podataka, kao i eventualne dodatne funkcije. U slučaju da nije unet kriterijum pretrage, prikazuju se svi uneti podaci iz baze. Na kraju se uništava objekat.

```

<h4>PRETRAGA BIBLIOTEČKOG FONDA</h4>

<!-- Forma za unos podataka -->
<form action="prikazpretragefonda.php" method="POST">
    <table align="center">
        <tr style="float:center; clear:left; margin-top:10px;"><td>po nazivu:</td>
            <td style="float:left; clear:right; margin-top:10px;"><input type="text"
name="Naziv" maxlength=20 size=20 autofocus tabindex=1></td>

```

```

        </tr>
        <tr style="float:center; clear:left; margin-top:10px;"><td>po jeziku:</td>
            <td style="float:left; clear:right; margin-top:10px;"><input type="text"
name="Jezik" maxlenth=20 size=20 tabindex=2></td>
        </tr>
        <tr style="float:center; clear:left; margin-top:10px;"><td>po prezimenu
autora:</td>
            <td style="float:left; clear:right; margin-top:10px;"><input type="text"
name="Prezime" maxlenth=20 size=20 tabindex=3></td>
        </tr>
        <tr style="float:center; clear:left; margin-top:10px;"><td>po imenu autora:</td>
            <td style="float:left; clear:right; margin-top:10px;"><input type="text"
name="Ime" maxlenth=20 size=20 tabindex=4></td>
        </tr>
        <tr style="float:center; clear:left; margin-top:10px;"><td>po vrsti
jedinice:</td>
            <td style="float:left; clear:right; margin-top:10px;"><input type="text"
name="VrstaJedinice" maxlenth=20 size=20 tabindex=5></td>
        </tr>
        <tr style="float:center; clear:left; margin-top:10px;"><td>po godini
izdavanja:</td>
            <td style="float:left; clear:right; margin-top:10px;"><input type="text"
name="GodinaIzdavanja" maxlenth=20 size=20 tabindex=6></td>
        </tr>
        <tr style="float:center; clear:right; margin-top:20px;">
            <td></td>
            <td style="float:left; clear:right; margin-top:10px;">
                <input type="submit" name="snimi" value="Pronađi" tabindex=6>
                <button type="reset" name="novkriterijum" tabindex=7>Nova
pretraga</button>
            </td>
        </tr>
    </table>
</form>

```

#### Pretraga fonda, prikaz – prikazpretragefonda.php:

```

<h4>REZULTAT PRETRAGE BIBLIOTEČKOG FONDA</h4>
<?php
//Dodela vrednosti parametrima
$pNazivJedinice=$_POST['Naziv'];
$pVrstaJedinice=$_POST['VrstaJedinice'];
$pAutorPrezime=$_POST['Prezime'];
$pAutorIme=$_POST['Ime'];
$pGodina=$_POST['GodinaIzdavanja'];
$pJezik=$_POST['Jezik'];
//Kreiranje objekta Jedinica
require 'klase/KonekcijaDB.php';
require 'klase/BiblioteckaJedinica.php';
$objJedinica = new BiblioteckaJedinica();

```

```

        $prikaz="";
        $brojredova=0;
        //Poziv metode za prikaz svih jezika u BP
        $prikaz=$objJedinica->Pretraga($pNazivJedinice,
$pVrstaJedinice, $pAutorPrezime, $pAutorIme, $pGodina, $pJezik);
        if ($prikaz)
        {
            $brojredova=mysqli_num_rows($prikaz);
        }
        $red="";
        if ($brojredova>0)
        {
            //Tabelarni prikaz pronadjenih jedinica
            echo "<table class='table table-striped'>
<thead>
<tr>
<th scope='col'>IBR</th>
<th scope='col'>Naziv</th>

<th scope='col'>Autor</th>
<th scope='col'>Godina</th>
<th scope='col'>Vrsta</th>
<th scope='col'>Rashod</th>

<th scope='col'>Akcije:</th>
<th scope='col'></th>
</tr>
</thead><tbody>";

            $red="";
            while($red = mysqli_fetch_array($prikaz))
            {
                $autor=$red['Prezime']." ".$red['Ime'];
                $naziv=$red['Naziv'];
                $idautora=$red['IdAutora'];
                $autorprikaz="";
                if ($red['Rashodovana']==="1")
                {
                    $rashod="Da";
                }
                else
                {
                    $rashod="-";
                }
                echo "<tr>
                    <td align=left>".$red['InvBr'].</td>
                    <td align=left>".$red['Naziv'].</td>
                    <td align=left>".$autor.</td>

```



```

        izdavanja' ]. "</td>

        <td align=left>". $red[ 'Naziv vrste

jedinice' ]. "</td>

        <td align=left>". $rashod. "</td>";

        $invbr=$red[ 'InvBr' ];

        if ($rashod=="-")
        {
            echo "<td align=left> <form
action='jedinicaizmeni.php' method='POST'>";
            echo "<input type='hidden' name='invbr'
value='$invbr'>";

            echo "<input type='submit' name='izmena'
value='Izmena'>";

            echo "</form> </td>";

            echo "<td align=left> <form
action='jedinicapotvrdbrisanja.php' method='POST'>";
            echo "<input type='hidden' name='invbr'
value='$invbr'>";

            echo "<input type='submit' name='brisanje'
value='Brisanje'>";

            echo "</form> </td>";

            echo "<td align=left> <form
action='jedinicarashoduj.php' method='POST'>";
            echo "<input type='hidden' name='invbr'
value='$invbr'>";

            echo "<input type='submit' name='rashoduj'
value='Rashoduj'>";

            echo "</form> </td>";

            echo "<td align=left> <form
action='jedinicaautori.php' method='POST'>";
            echo "<input type='hidden' name='invbr'
value='$invbr'>";

            echo "<input type='hidden' name='idautora'
value='$idautora'>";

            echo "<input type='hidden' name='autorprikaz'
value='$autorprikaz'>";

            echo "<input type='hidden' name='autor'
value='$autor'>";

            echo "<input type='hidden' name='naziv'
value='$naziv'>";

            echo "<input type='submit' name='autori'
value='Autor(i)'>";

```

```

        echo "</form> </td> </tr>";
    }
}
echo "</tbody>
</table>
<br/>";
}
else
{
    echo "Za izabrani kriterijum pretrage nemate nijednu
bibliotečku jedinicu u fondu!";
}
//Taster za povratak na pretragu
echo "<a href='pretragaфона.php'><button
type='button'>Nova pretraga</button></a>";
//Uništavanje objekta Jezik
$objJedinica= null;
?>

```

**Prikaz podataka - zaduzenja.php.** referencira se klasa u kojoj se nalaze metode za rad sa bazom podataka, tj. navodi relativna putanja do fajla u kom se nalazi klasa, kreira se objekat, poziva se metoda za prikaz podataka iz MySQL baze. Kreira se result-set sa podacima iz baze, prebrojava koliko ima redova u result-setu (funkcijom `mysqli_num_rows`). Ukoliko je broj redova veći od jedan, formira se tabela koja prikazuje učitane podatke iz baze podataka. U tabeli se nalaze forma, promenljive sa vrednostima i taster za pokretanje dodatne funkcije. Na kraju se uništava objekat.

```

<h4>ZADUŽENJA STUDENATA</h4>
<?php
//Kreiranje objekta Jedinica
require 'klase/KonekcijaDB.php';
require 'klase/Izdata.php';
$objZaduzenja = new Izdata();
$prikaz="";
$brojredova=0;
//Poziv metode za prikaz svih jezika u BP
$prikaz=$objZaduzenja->PrikazSvihZaduzenja();
if ($prikaz)
{
    $brojredova=mysqli_num_rows($prikaz);
}
$red="";
if ($brojredova>0)
{
    //Tabelarni prikaz pronadjenih zaduzenja
    echo "<table class='table table-striped'>
    <thead>
    <tr>
    <th scope='col'>Broj clana</th>
    <th scope='col'>Član</th>

```

```

<th scope='col'>Naziv jedinice</th>
<th scope='col'>Inventarni broj</th>

<th scope='col'>Datum zaduživanja</th>
<th scope='col'>Vraćeno</th>
<th scope='col'>Period (mesec)</th>

<th scope='col'>Vrsta</th>
<th scope='col'>Akcija</th>
</tr>
</thead><tbody>";

$red="";
while($red = mysqli_fetch_array($prikaz))
{
    $clan=$red['Prezime']." ".$red['Ime'];
    $invbr=$red['InvBr'];
    $naziv=$red['Naziv'];
    $datizd=$red['Datum izdavanja'];
    $datvrac=$red['Datum vraćanja'];
    $rbizd=$red['RB izdavanja'];
    echo "<tr>
    <td align=left>".$red['BrClana'].</td>
    <td align=left>".$clan.</td>
    <td align=left>".$red['Naziv'].</td>
    <td align=left>".$red['InvBr'].</td>
    <td align=left>".$red['Datum izdavanja'].</td>
    <td align=left>".$red['Datum vraćanja'].</td>
    <td align=left>".$red['Period'].</td>
    <td align=left>".$red['Naziv vrste
jedinice'].</td>";

    if ($datvrac=="")
    {
        echo "<td align=left> <form
action='razduzivanje.php' method='POST'>";
        echo "<input type='hidden' name='invbr'
value='$invbr'>";
        echo "<input type='hidden' name='rbizd'
value='$rbizd'>";
        echo "<input type='hidden' name='clan'
value='$clan'>";
        echo "<input type='hidden' name='naziv'
value='$naziv'>";
        echo "<input type='hidden' name='datizd'
value='$datizd'>";
        echo "<input type='submit' name='razduzi'
value='Razduživanje'>";
    }
}

```

```

        echo "</form> </td> </tr>";
    }
}
echo "</tbody>
</table>";
}
//Uništavanje objekta Zaduzenja
$objZaduzenja = null;
?>

```

**Izveštaj – spisakopomena.php.** Stranica koja ne uključuje Bootstrap radni okvir. Ovo je “Printer friendly” stranica sa belom pozadinom na kojoj je formiran dokument koji se može odštampati. Prvo se proverava se status otvorene sesije za korisnika preko niza promenljivih. Ukoliko je status korisnika bez vrednosti, pokušan je pristup stranici bez prijave, pa se tok aplikacije preusmerava na početnu stranu, tj. index.php. U slučaju da promenljive sesije imaju vrednosti, formira se heder/zaglavljje dokumenta pomoću tabele koja sadrži podatke u školi. Pomoću <H> oznake se formira naslov, ispod kojeg se prikazuje preuzeti broj dosijea. Zatim se otvara konekcija do baze podataka, kreira se objekat dosije, poziva se metoda za učitavanje dosijea prema broju, kreira result-set sa podacima za prikaz, prebrojava koliko ima redova u result setu. Ako je ovaj broj veći od jedan, kreira se telo izveštaja tako što se funkcijom `mysqli_fetch_array` kreira niz sa vrednostima, kojima se pristupa preko naziva kolona iz baze. Kreira se HTML tabela koja prikazuje tražene podatke iz dosijea i na kraju se uništava kreirani objekat.

```

<?php
    session_start();
    $status=$_SESSION["status"];
    $username=$_SESSION["username"];
    $prezime=$_SESSION["prezime"];
    $ime=$_SESSION["ime"];
    $idkorisnika=$_SESSION["idclana"];
    if ($status=="") //provera statusa zbog sprečavanja direktnog pristupa stranici
    {
        header ('Location:index.php');
    }
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Biblioteka</title>
</head>
<body>
<?php
    echo "<table border=0>
    <tr><td>Tehnički fakultet Mihajlo Pupin</td></tr>
    <tr><td>Đure Đakovića bb</td></tr>
    <tr><td>23000 Zrenjanin</td></tr>
    <tr><td>023/550516</td></tr>
    </table>
    <h3 align='center'>SPISAK OPOMENA</h3>";

```

```

//Kreiranje objekta Opomene
require 'klase/KonekcijaDB.php';
require 'klase/Izdata.php';
$objOpomene = new Izdata();
$upit="";
$rezultat="";
$brojredova=0;
//Poziv metode za učitavanje svih opomena
$rezultat=$objOpomene->IzvestajOpomene();
$brojredova = mysqli_num_rows($rezultat);
if ($brojredova==0) //prazna tabela
{
    echo "U biblioteci ne postoji nijedna izdata opomena!";
    echo "<br/>";
}
else
{
    echo "
    <table align=center border=1>
    <th>RB</th>
    <th>Prezime</th>
    <th>Ime</th>
    <th>Broj indeksa</th>
    <th>Bibliotečka jedinica</th>
    <th>Opomena</th>";
    $red=0;
    $rb=1;
    while($red = mysqli_fetch_array($rezultat))
    {
        echo "<tr>";
        echo "<td align=center>" . $rb . "</td>";
        echo "<td>" . $red['Prezime'] . "</td>";
        echo "<td>" . $red['Ime'] . "</td>";
        echo "<td>" . $red['Broj indeksa'] . "</td>";
        echo "<td>" . $red['Naziv'] . "</td>";
        echo "<td>" . $red['Opomena'] . "</td>";
        echo "</tr>";
        $rb++;
    }// end while
    echo "</table>";

    echo "<br/>
    Ukupno opomena: $brojredova<br/><br/>";
    echo date('d.m.Y').".";
    echo "<br/>";
    echo "У Зрењанину";
    echo "<p align='right'>Bibliotekar</p>";
    echo "<p align='right'>_____</p>";

```

```
?>
</body>
</html>
```

**Prijava korisnika – prijava.php** stranica sadrži, pored već pomenutih delova na početnoj stranici i formu za unos korisničkog imena i šifre, preko polja za unos podataka (input type tipa email i password). POST metodom se uneti podaci prosleđuju stranici za proveru imena i šifre.

```
<h3>Prijava korisnika</h3>
    <form action="proverakorisnika.php" method="POST">
        <table align="center">
            <tr align="center" style="float:left; clear:left;
margin-top:10px;"><td align="center">E-Mail:</td>
                <td align="center" style="float:left; clear:right;
margin-top:10px;"><input type="email" name="email"
onKeyPress="if(this.value.length==20) return false;" size=22 required
tabindex=1></td>
            </tr>
            <tr align="center" style="float:left; clear:left;
margin-top:10px;"><td align="center">Šifra:</td>
                <td align="center" style="float:left; clear:right;
margin-top:10px;"><input type="password" name="sifra"
onKeyPress="if(this.value.length==20) return false;" size=22 required
tabindex=2></td>
            </tr>
            <tr style="float:left; clear:left; margin-top:10px;
margin-left:40px">
                <td align="center">
                    <input type="submit" align="center" class="btn
btn-default" name="prijava" value="Prijava se" tabindex=3>
                </td>
            </tr>
        </table>
    </form>
```

**Provera korisnika – proverakorisnika.php.** Stranica bez korisničkog interefejsa. Preuzimaju se vrednosti za šifru i email POST metodom, kreira se objekat, poziva se metoda za proveru člana. Kreira se result-set sa podacima iz baze, prebrojava koliko ima redova u result-setu (funkcijom mysqli\_num\_rows). Ukoliko je broj redova nula, sledi kreiranje objekta i proveru šifre i emaila za bibliotekara. Kreira se novi result-set, koji ako je prazan korisnik ne postoji u bazi i/ili šifra nije dobro upisana. Sledi preusmeravanje korisnika na početnu stranicu aplikacije. Ukoliko su pak ime i šifra pronađeni u bazi, kreira se sesija za korisnika sa nekoliko promenljivih. Na kraju stranice se uništava objekat i preusmerava aplikacija na stranicu sa menijem. U slučaju neispravno unete šifre i/ili korisničkog imena, sesija se uništava. Na kraju se uništava objekat.

```
<?php
    session_start(); //Pocetak, tj. aktiviranje/kreiranje sesije

    //Preuzimanje vrednosti iz rubrika za unos sa iste ove stranice
    $loginusername=$_POST['email'];
    $loginpassword=$_POST['sifra'];
```

```

//Pocetne vrednosti promenljivih za proveru korisnika
$logsucc=0;
$rezultatprijava="";
$status="clan";
$idclana="";
$brredova=0;

//Kreiranje objekata korisnik i bibliotekar
require 'klase/KonekcijaDB.php';
require 'klase/Clan.php';
require 'klase/Bibliotekar.php';
$objClan = new Clan();
$objBibliotekar = new Bibliotekar();
$objClan->KorisnickaSifra=$loginpassword;
$objClan->Email=$loginusername;

//Poziv metode za proveru sifre i korisnickog imena
$rezultatprijava=$objClan->Prijava();
$brredova=mysqli_num_rows($rezultatprijava);

if ($brredova==0) //Ne postoji clan, sledi provera da li postoji
bibliotekar
{
    $objBibliotekar->KorisnickaSifra=$loginpassword;
    $objBibliotekar->Email=$loginusername;
    //Poziv metode za proveru sifre i korisnickog imena
    $rezultatprijava=$objBibliotekar->Prijava();
    $brredova=mysqli_num_rows($rezultatprijava);
    $status="bibliotekar";
}
if ($brredova==1) //postoji korisnik
{
    $red=0;
    $logsucc=1;
    while($red = mysqli_fetch_array($rezultatprijava))
    {
        //dodela vrednosti promenljivama u sesiji
        $username=$red['Email'];
        $ime=$red['Ime'];
        $prezime=$red['Prezime'];
        if ($status=="bibliotekar")
        {
            $idclana=$red['ID bibliotekara'];
        }
        else
        {
            $idclana=$red['Broj clana'];
        }
    }
}

```

```

        $_SESSION["ime"] = $ime;
        $_SESSION["prezime"] = $ime;
        $_SESSION["username"] = $username;
        $_SESSION["status"] = $status;
        $_SESSION["idclana"] = $idclana;
    }
}
//Unistavanje objekata
$objClan=null;
$objBibliotekar=null;
if ($logsucc==1)
{
    //redirektovanje aplikacije na pocetnu stranicu sa menijem
    header('Location:pocetna.php');
}
else //brredova je 0, sto znaci da sifra i email nisu pronadjeni u bazi
{
    //unistavanje sesije
    session_destroy();
    session_unset();
    //redirektovanje stranice, povratak na prijavu
    header('Location:prijava.php');
}
?>

```

**Prva stranica nakon prijave – pocetna.php.** Ova stranica, kao i sve ostale kojima pristupaju korisnici, ima na samom početku blok PHP naredbi za rad sa otvorenom sesijom, čitanje vrednosti promenljivih u sesiji i proveru da li je status korisnika blanko ili ne. U slučaju da je status prazan, vrši se redirektovanje aplikacije na početnu stranicu. U slučaju regularne prijave prikazuje se zaglavlje strane, gornji meni u zavisnosti od profila korisnika, sredina stranice sa odeljcima i naslovim, donji meni i footer web strane.

```

<?php
    session_start();
    $status=$_SESSION["status"];
    $username=$_SESSION["username"];
    $prezime=$_SESSION["prezime"];
    $ime=$_SESSION["ime"];
    $idkorisnika=$_SESSION["idclana"];
    if ($status=="") //provera statusa zbog sprecavanja direktnog pristupa stranici
    {
        header ('Location:index.php');
    }
?>
<!DOCTYPE HTML>
<html>
    <?php include "head.html"; ?>

    <body>
    <div class="fh5co-loader"></div>

```



```

<!-- Meni -->

<?php
    //prikaz odgovarajuceg sadrzaja menija
    if ($status=="bibliotekar")
    {
        include "meniupbibliotekar.html";
    }
    if ($status=="clan")
    {
        include "meniupclan.html";
    }
?>

<!-- Sredina stranice -->

    <div id="fh5co-course-categories">
        <div class="container">
            <div class="row animate-box">
                <div class="col-md-6 col-md-offset-3 text-center fh5co-heading">
                    <h2>BIBLIO v1.0</h2>
                    <p>Softver za podršku rada fakultetske biblioteke</p>
                </div>
            </div>
        </div>
    </div>

<!-- Meni -->

<?php
    //prikaz odgovarajuceg sadrzaja menija
    if ($status=="bibliotekar")
    {
        include "menidownbibliotekar.html";
    }
    if ($status=="clan")
    {
        include "menidownclan.html";
    }
?>

<!-- Footer stranice -->

    <?php include "footer.html"; ?>

</body>
</html>

```

**Odjava korisnika – odjava.php.** Ova PHP stranica nema HTML elemente. Sadrži blok PHP naredbi za rad sa otvorenom sesijom, čitanje vrednosti promenljivih u sesiji, proveru da li je status korisnika blanko ili ne. U slučaju da je status prazan, vrši se redirektovanje aplikacije na početnu stranicu za slučaj da je neko hteo da direktno pokrene odjavu. Ako je stranicu pokrenuo prijavljeni korisnik, uništava se sesija, brišu se vrednosti promenljivih iz memorije i vrši se redirektovanje aplikacije na početnu stranicu, čime je korisnik odjavljen.

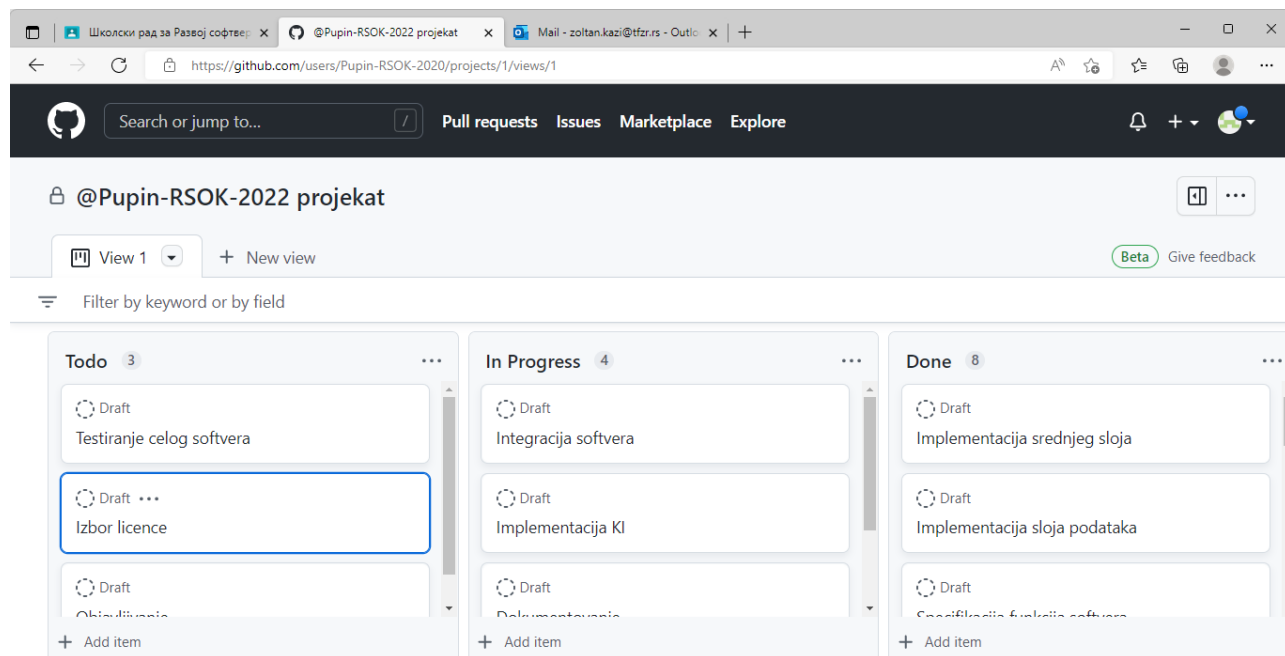
```
<?php
    session_start();
    $status=$_SESSION["status"];
    $username=$_SESSION["username"];
    $prezime=$_SESSION["prezime"];
    $ime=$_SESSION["ime"];
    $idkorisnika=$_SESSION["idclana"];
    if ($status=="") //provera statusa zbog sprecavanja direktnog pristupa stranici
    {
        header ('Location:index.php');
    }
    session_destroy();
    session_unset();
    header('Location:index.php');
?>
```

## 6.2. RAD SA GITHUB SISTEMOM

Kreirani repozitorijum je: Github.com/Vezbe2022

Učesnici projekta: 1. Kazi Zoltan (creator repozitorijuma), 2. Studenti 3. Godine smjera Softversko inženjerstvo

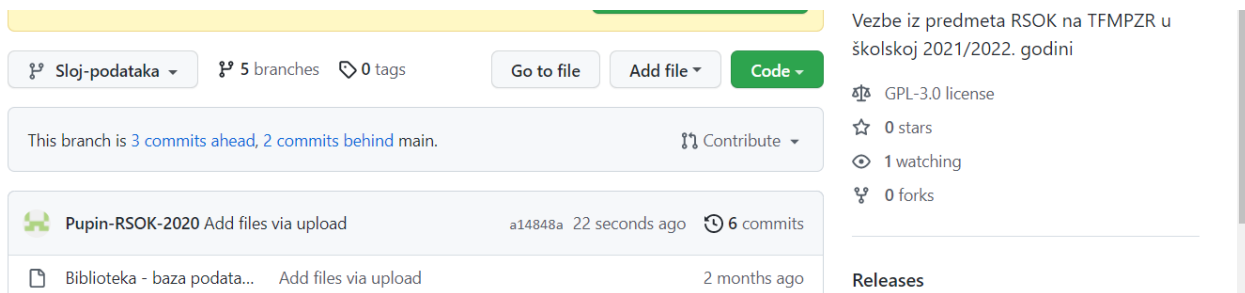
Plan projekta sa fazama razvoja: 1. Specifikacija zahteva 2. Idejno rešenje 3. Dizajn aplikacije: Baze podataka, srednjeg aplikacionog sloja i baze podataka 4. Implementacija slojeva 5. Testiranje i integracija 6. Dokumentovanje, izbor licence.



Kreriare grane: 1. Prezantacioni sloj, 2. Srednji aplikacioni sloj, 3. Sloj podataka.


### Broj COMMIT operacija


Sloj podataka (6):




## Srednji aplikacioni sloj (12):

This branch is [9 commits ahead](#), [2 commits behind](#) main. [Contribute](#)

 **Pupin-RSOK-2020** Add files via upload 79318fa 22 seconds ago 12 commits

 klase Add files via upload 22 seconds ago

 Biblioteka - dizajn srednj... Add files via upload last month

### Releases


No releases published [Create a new release](#)


### Packages

## Prezentacioni sloj (9):

[Sloj-KI](#) 5 branches 0 tags [Go to file](#) [Add file](#) [Code](#)

This branch is [6 commits ahead](#), [2 commits behind](#) main. [Contribute](#)

 **Pupin-RSOK-2020** Add files via upload b37d15c 25 seconds ago 9 commits

 css Add files via upload 2 hours ago

### Releases

No releases published [Create a new release](#)

0 stars  
1 watching  
0 forks

Postavljanje datoteka se postiže jednostavnim prevlačenjem foldera i fajlova sa lokalnog računara u region prikazan na sledećoj slici:


Школски рад за Развој софтвера Upload files - Pupin-RSOK-2020/ Mail - zoltan.kazi@tfz.rs - Outlook


https://github.com/Pupin-RSOK-2020/Vezbe2022/upload/Sloj-KI

**Pupin-RSOK-2020 / Vezbe2022** Private [Unwatch](#) 1 [Fork](#) 0 [Star](#) 0

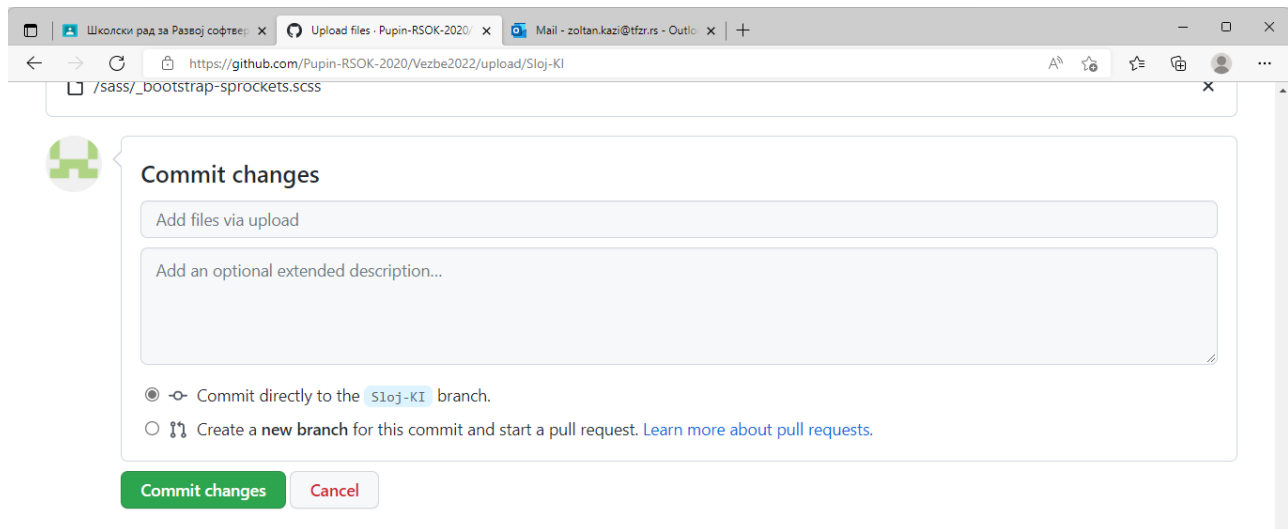
[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) 1 [Security](#) [Insights](#) [Settings](#)

[Vezbe2022](#) /

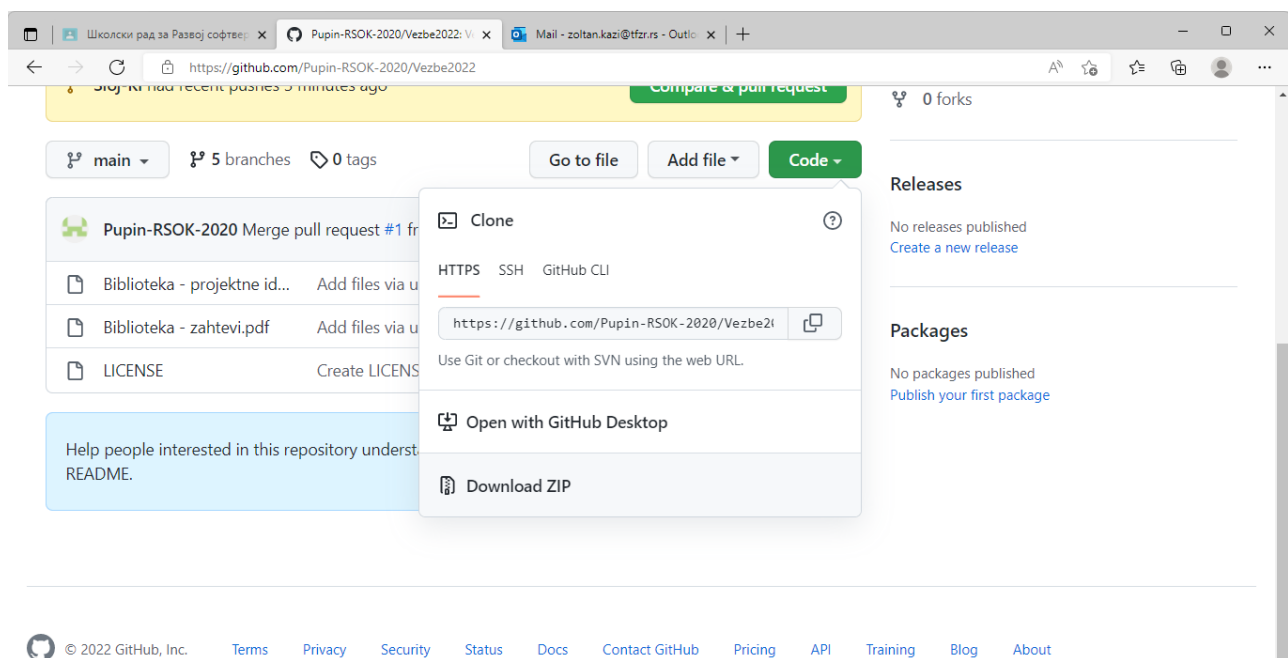
  
**Drag additional files here to add them to your repository**  
Or [choose your files](#)

 /sass/bootstrap/mixins/\_size.scss  
Uploading 26 of 75 files

Nakon kopiranja datoteka, vrši se potvrda promena, opcijom “Commit changes”, nakon čega sledi “Upload” istih.



Preuzimanje datoteka iz repozitorijuma je moguće sa osnovne ili bilo koje druge grane u projektu je preko tastera “Code” – “Download ZIP”:



Spajanje grana je izvršeno nakon završetka postavljanja svih datoteka na granama opcijom "Compare & pull request".

https://github.com/Pupin-RSOK-2020/Vezbe2022

Unwatch 1 Fork 0 Star 0

Code Issues Pull requests Actions Projects 1 Security Insights Settings

Sloj-podataka had recent pushes about 1 hour ago [Compare & pull request](#)

Sloj-aplikacione-logike had recent pushes about 1 hour ago [Compare & pull request](#)

Sloj-KI had recent pushes about 1 hour ago [Compare & pull request](#)

main 5 branches 0 tags Go to file Add file Code

Pupin-RSOK-2020 Merge pull request #1 from Pupin-RS... ff7ca85 on 24 Mar 5 commits

Biblioteka - projektne id... Add files via upload 3 months ago

About

Vezbe iz predmeta RSOK na TFMPZR u školskoj 2021/2022. godini

GPL-3.0 license

0 stars

1 watching

0 forks

Releases

No releases published [Create a new release](#)

Prikaz rezultata spajanja grana:

https://github.com/Pupin-RSOK-2020/Vezbe2022/pull/3

### Sloj ki #3

Merged Pupin-RSOK-2020 merged 6 commits into main from Sloj-KI 22 seconds ago

Conversation 0 Commits 6 Checks 0 Files changed 183 +30,844 -0

Pupin-RSOK-2020 commented 37 seconds ago

No description provided.

Pupin-RSOK-2020 added 6 commits last month

- Add files via upload Verified da38a3e
- Add files via upload Verified bb7f74d
- Add files via upload Verified d22a1b3
- Add files via upload Verified 876d7a0
- Add files via upload Verified b38cca0

Reviewers

No reviews

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Licenca – Postavljena je GPL licenca na osnovu šablona koji se već nalaze na Github.com, fajl je: LICENCE.TXT

Школки рад за Развој софтвера

Vezbe2022/LICENSE at main · Pupin-RSOK-2020

Mail - zoltan.kazi@tfz.rs - Outlook

https://github.com/Pupin-RSOK-2020/Vezbe2022/blob/main/LICENSE

Pupin-RSOK-2020 / Vezbe2022 Private

Unwatch 1 Fork 0 Star 0

<> Code

Issues

Pull requests

Actions

Projects 1

Security

Insights


Settings

main

Vezbe2022 / LICENSE

Go to file


...

 Pupin-RSOK-2020/Vezbe2022 is licensed under the **GNU General Public License v3.0**

Permissions of this strong copyleft license are conditioned on making available complete source code of licensed works and modifications, which include larger works using a licensed work, under the same license. Copyright and license notices must be preserved. Contributors provide an express grant of patent rights.

This is not legal advice. [Learn more about repository licenses.](#)

Permissions	Limitations	Conditions
✓ Commercial use	✗ Liability	① License and copyright notice
✓ Modification	✗ Warranty	① State changes
✓ Distribution		① Disclose source
✓ Patent use		① Same license
✓ Private use		

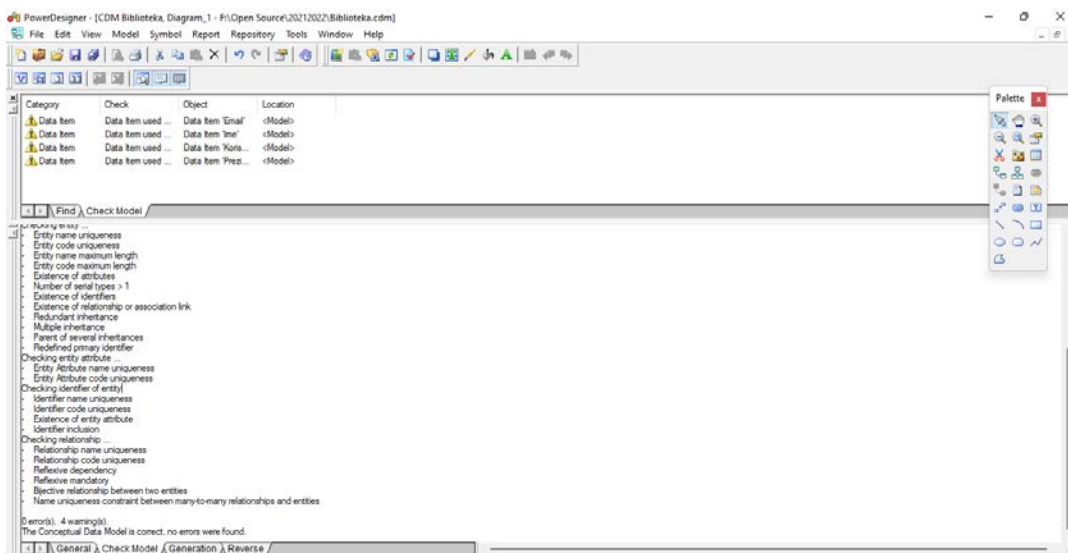
 Pupin-RSOK-2020 Create LICENSE

Latest commit 2ebd210 on 24 Mar [History](#)

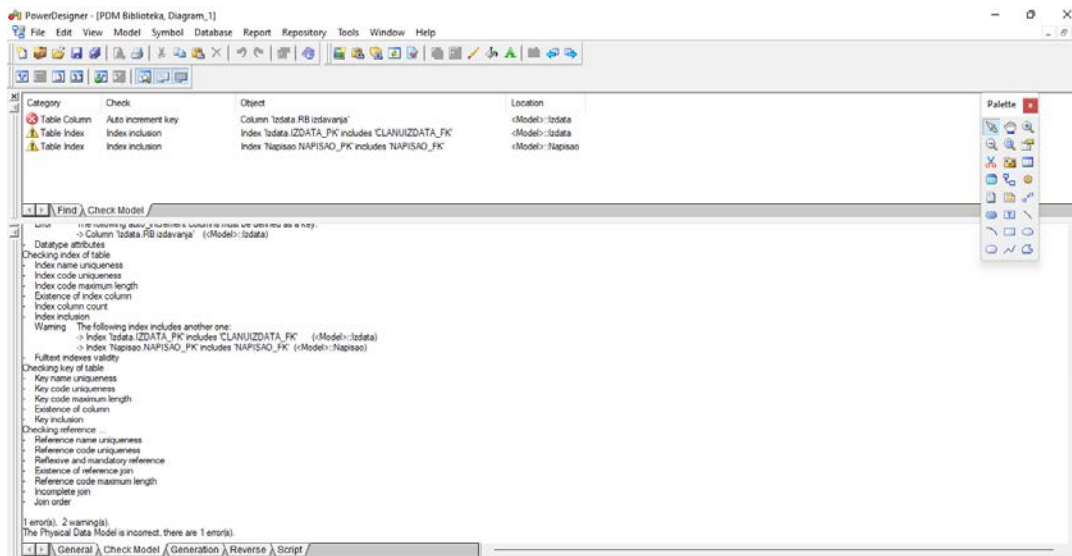
1 contributor

### 6.3. TESTIRANJE SOFTVERA

**Testiranje sloja podataka**, tj. konceptualnog i fizičkog modela podataka, automatski u okviru CASE alata, opcijom Check Model:



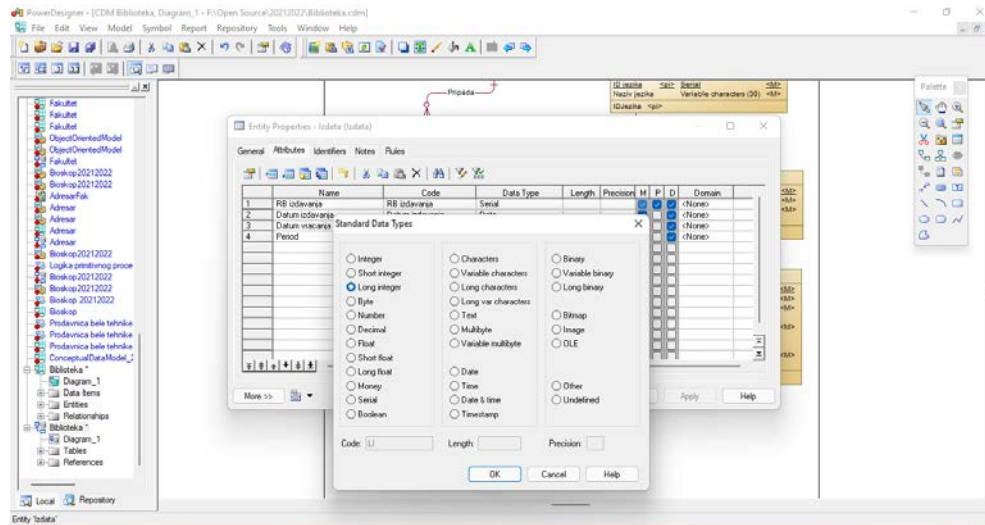
Rezultat testiranja: korektan CDM model.



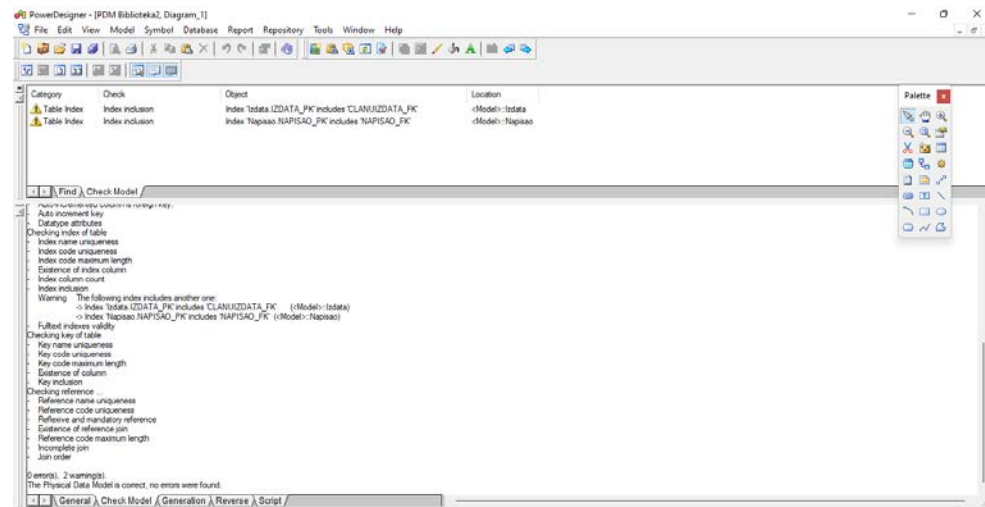
Rezultat testiranja: PDM model sadrži grešku – kolona RB IZDAVANJA u tabeli IZDATA ne može biti sa osobinom SERIAL (auto\_increment).

Otklanjanje greške – u CDM modelu je izvršena promena tipa podatka u entitetu IZDATA, za RB IZDAVANJA je izabran LONG INTEGER (bigint) tip podatka, te je ponovo je generisan PDM model.

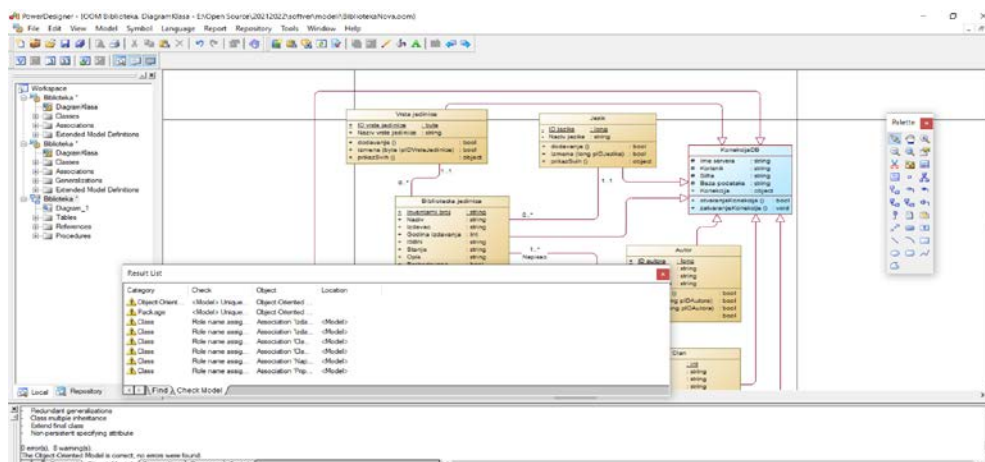




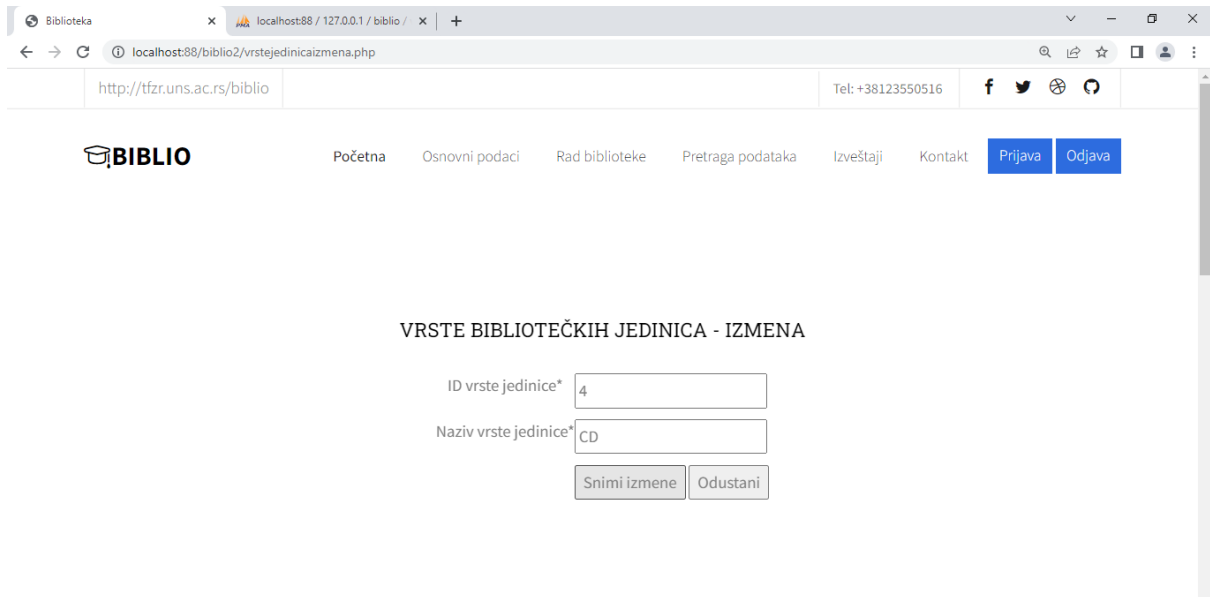
Rezultat testiranja: korektan PDM model, bez sintaksnih grešaka koje se odnose na kreiranje dijagrama.



Testiranje dizajna srednjeg aplikacionog sloja, automatski u okviru CASE alata, u OOM, opcijom Check Model:



## Funkcionalno testiranje softvera



Browser tabs: Biblioteka, localhost:88 / 127.0.0.1 / biblio / x | +

Address bar: localhost:88/biblio2/vrstejedinicaizmena.php

Header: http://tfzr.uns.ac.rs/biblio Tel: +38123550516 f t g h

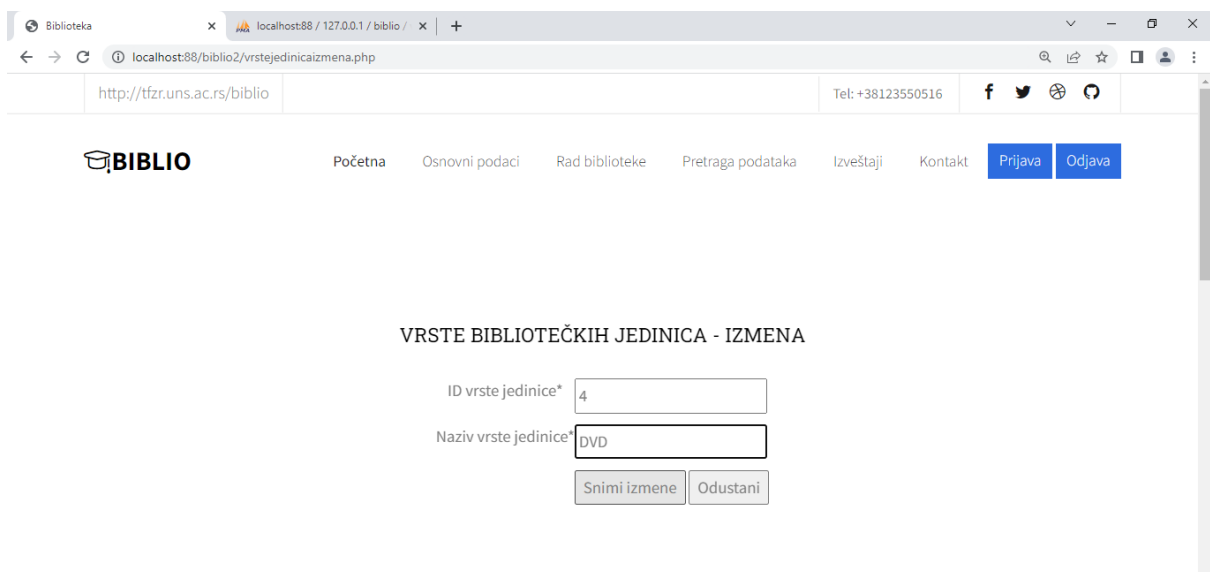
Navigation: BIBLIO Početna Osnovni podaci Rad biblioteke Pretraga podataka Izveštaji Kontakt **Prijava** **Odjava**

### VRSTE BIBLIOTEČKIH JEDINICA - IZMENA

ID vrste jedinice\*

Naziv vrste jedinice\*

Testiranje softverske funkcije za izmenu vrste bibliotečke jedinice:



Browser tabs: Biblioteka, localhost:88 / 127.0.0.1 / biblio / x | +

Address bar: localhost:88/biblio2/vrstejedinicaizmena.php

Header: http://tfzr.uns.ac.rs/biblio Tel: +38123550516 f t g h

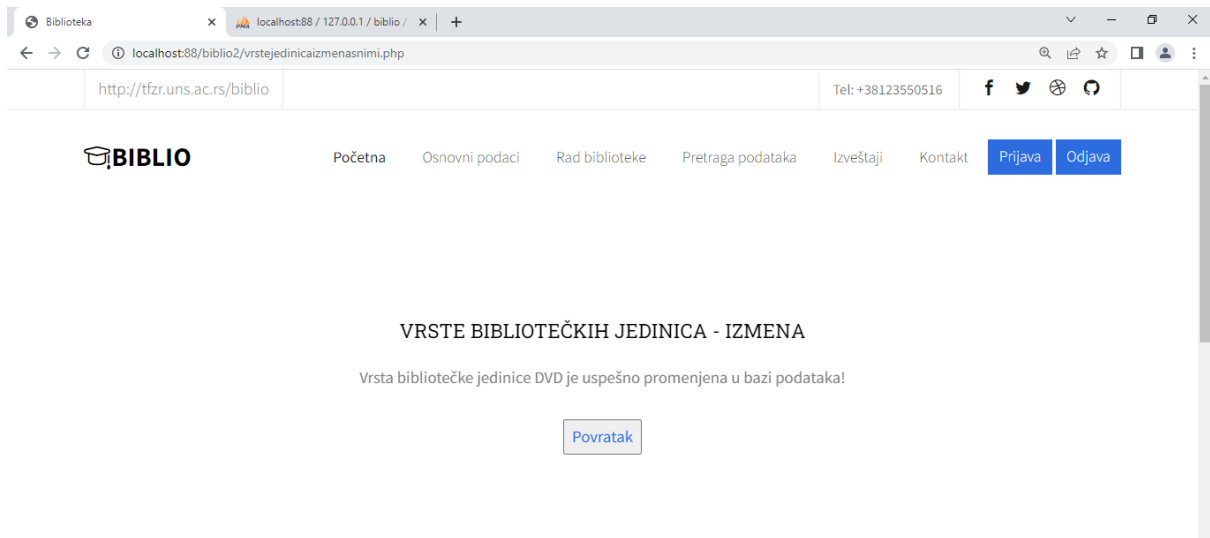
Navigation: BIBLIO Početna Osnovni podaci Rad biblioteke Pretraga podataka Izveštaji Kontakt **Prijava** **Odjava**

### VRSTE BIBLIOTEČKIH JEDINICA - IZMENA

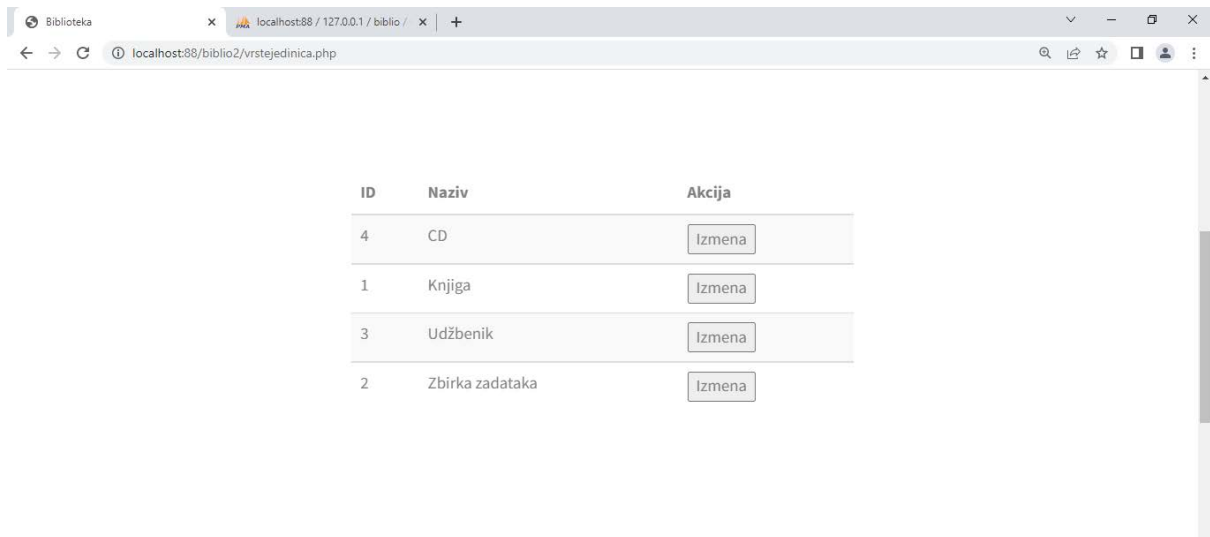
ID vrste jedinice\*

Naziv vrste jedinice\*

Rezultat izvršavanja funkcije – prikazano USPEŠNO, ali je u stvari došlo do neuspešne izmene podataka,...



... što se može videti na sledećoj slici:

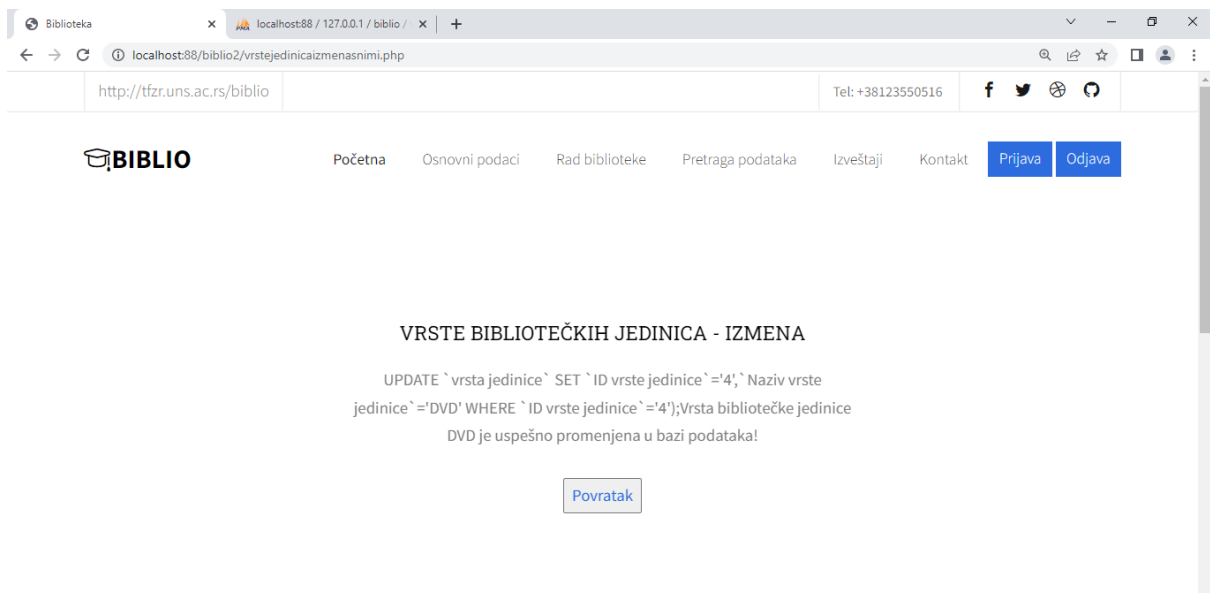


Dodavanje PHP echo naredbe za ispis SQL DML koda za izmenu podataka u web stranici VrsteJedinica.php:

The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left lists files under 'BIBLIO2' and 'OUTLINE'. The main editor area displays the file 'VrstaJedinice.php' with the following PHP code:

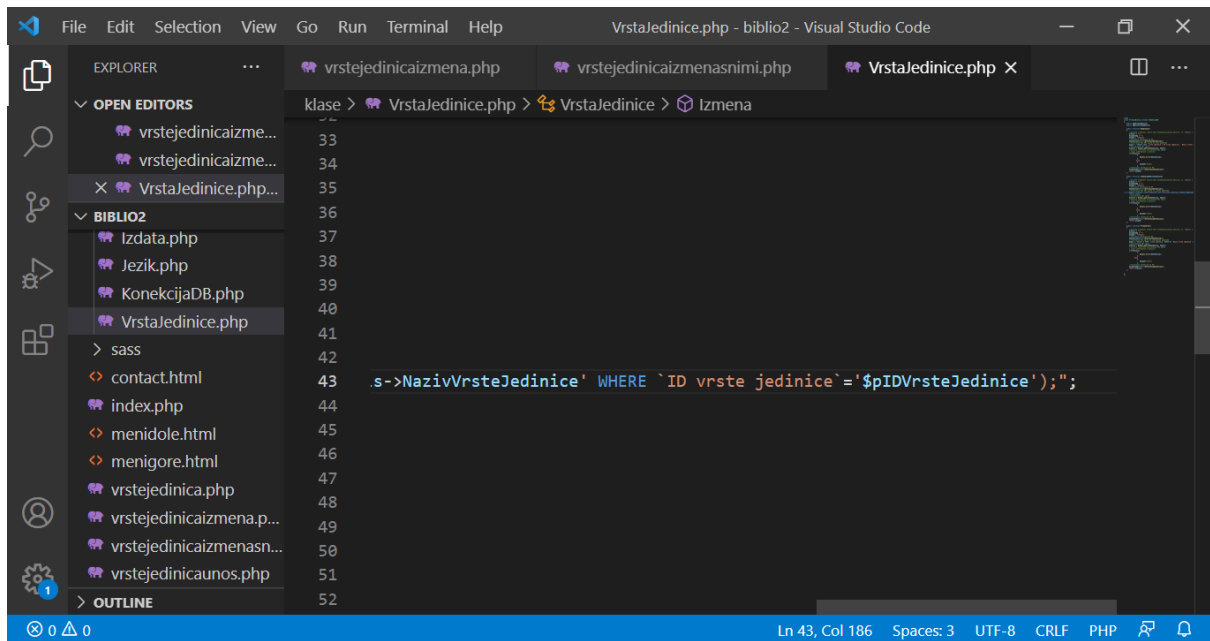
```
public function Izmena($pIDvrsteJedinice)
{
    //pocetne vrednosti result-seta (dvodimenzionalna matrica, tj. tabela)
    $result = "";
    $suspehkbp = "";
    $suspeh = "false";
    //otvaranje konekcije do BP
    $konekcija=$this->OtvaranjeKonekcije();
    //formiranje SQL DML upita za izmenu podataka
    $upit = "UPDATE `vrsta_jedinice` SET `ID vrste_jedinice`='$this->IdVrs...';
    echo $upit;
    //izvrsavanje SQL upita
    $result = mysqli_query($konekcija, $upit);
    /*provera rezultata izvrsavanja SQL upita
    i ispis informacije o gresci*/
    if(!$result)
    {
        mysqli_error($konekcija);
```

Ponovno učitavanje stranice VrsteJedinica.php I prikaz SQL upita:



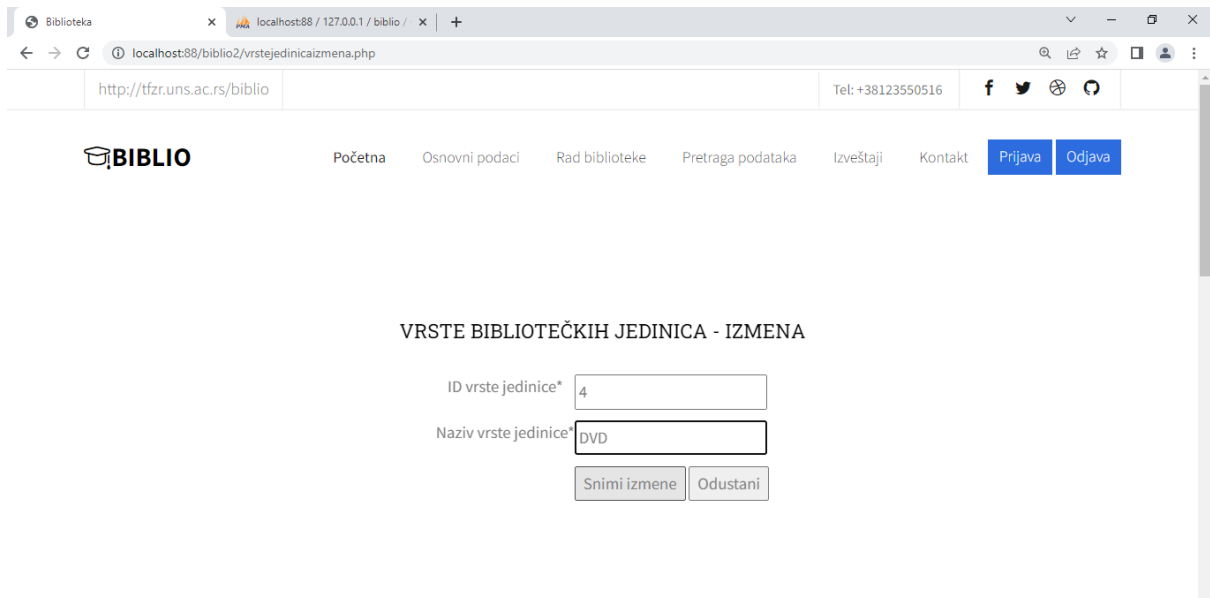
SQL DML upit prikazan na web stranici nije sintaksno dobro napisan i mora se korigovati. Greška je u simbolu „)” na samom kraju upita, koji nije sintaksno prihvatljiv.

Ispravka greške u u SQL Insert DML komandi:



```
File Edit Selection View Go Run Terminal Help
VrstaJedinice.php - biblio2 - Visual Studio Code
EXPLORER
OPEN EDITORS
vrstejedinicaizmena.php
vrstejedinicaizmenasnimi.php
VrstaJedinice.php x
BIBLIO2
Izdata.php
Jezik.php
KonekcijaDB.php
VrstaJedinice.php
sass
contact.html
index.php
menidole.html
menigore.html
vrstejedinica.php
vrstejedinicaizmena.p...
vrstejedinicaizmenasni...
vrstejedinicaunos.php
OUTLINE
VrstaJedinice.php
klase > VrstaJedinice.php > VrstaJedinice > Izmena
33
34
35
36
37
38
39
40
41
42
43 s->NazivVrsteJedinice' WHERE `ID vrste jedinice`='$pIDVrsteJedinice');";
44
45
46
47
48
49
50
51
52
Ln 43, Col 186 Spaces: 3 UTF-8 CRLF PHP
```

Ponovno testiranje softverske funkcije za izmenu vrste biblioteke jedinice, sa unosom istih podataka:



Biblioteka

localhost:88 / 127.0.0.1 / biblio /

http://tfzr.uns.ac.rs/biblio

Tel: +38123550516

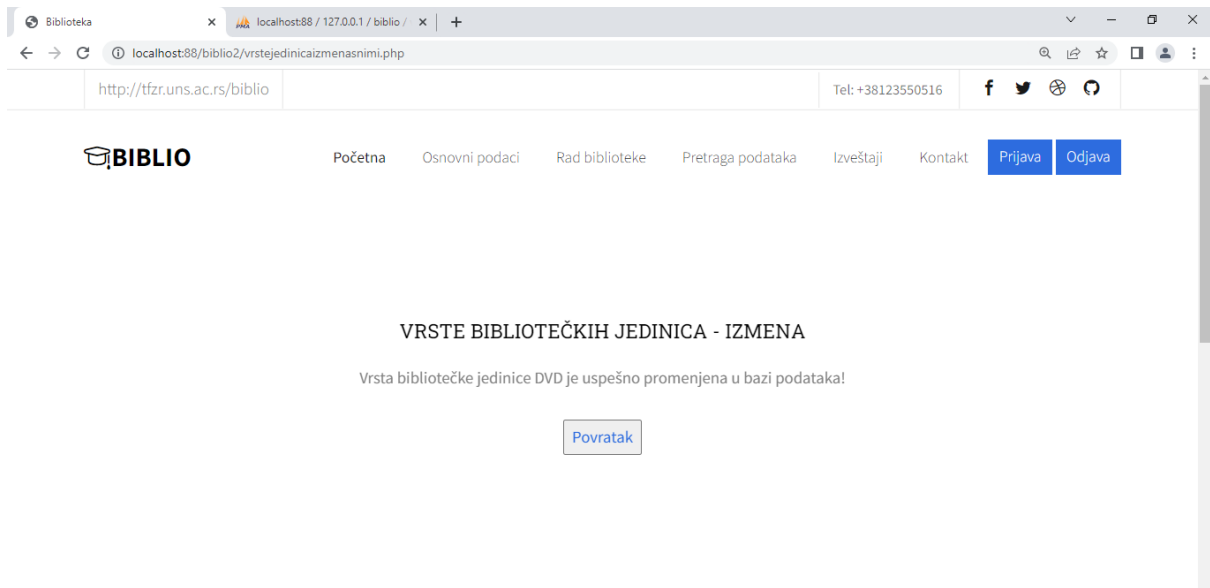
Početna Osnovni podaci Rad biblioteke Pretraga podataka Izveštaji Kontakt Prijava Odjava

### VRSTE BIBLIOTEČKIH JEDINICA - IZMENA

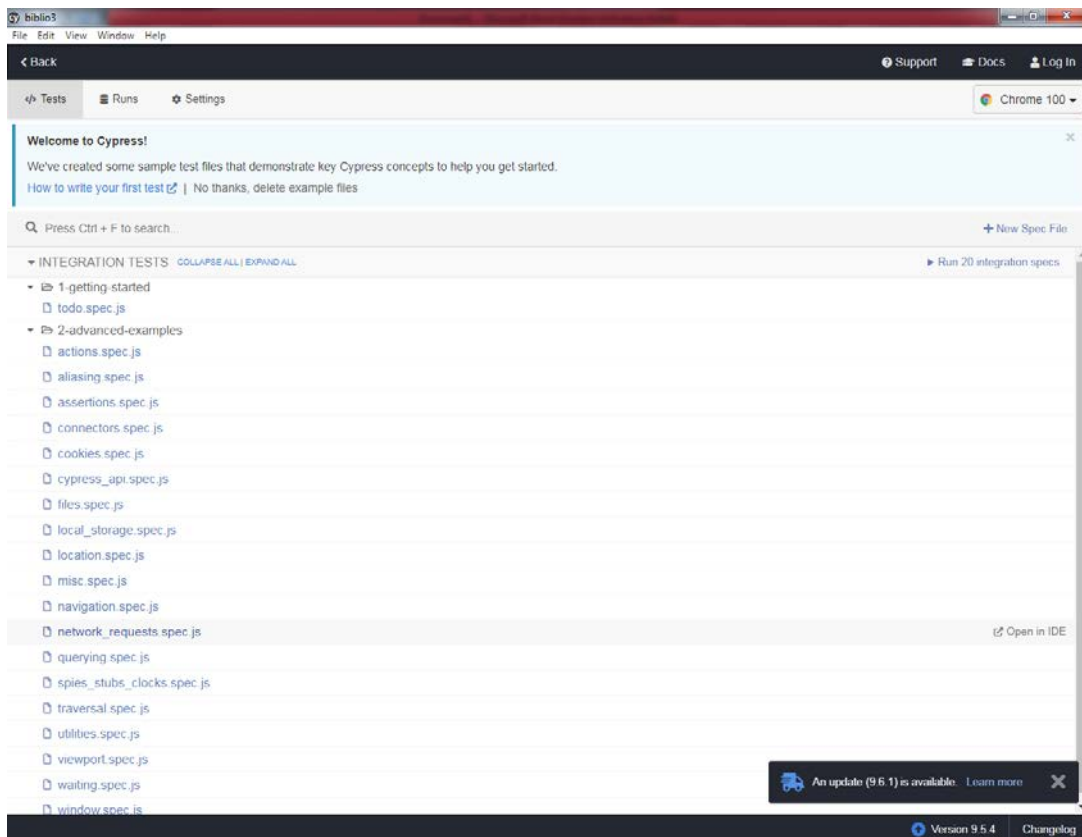
ID vrste jedinice\*

Naziv vrste jedinice\*

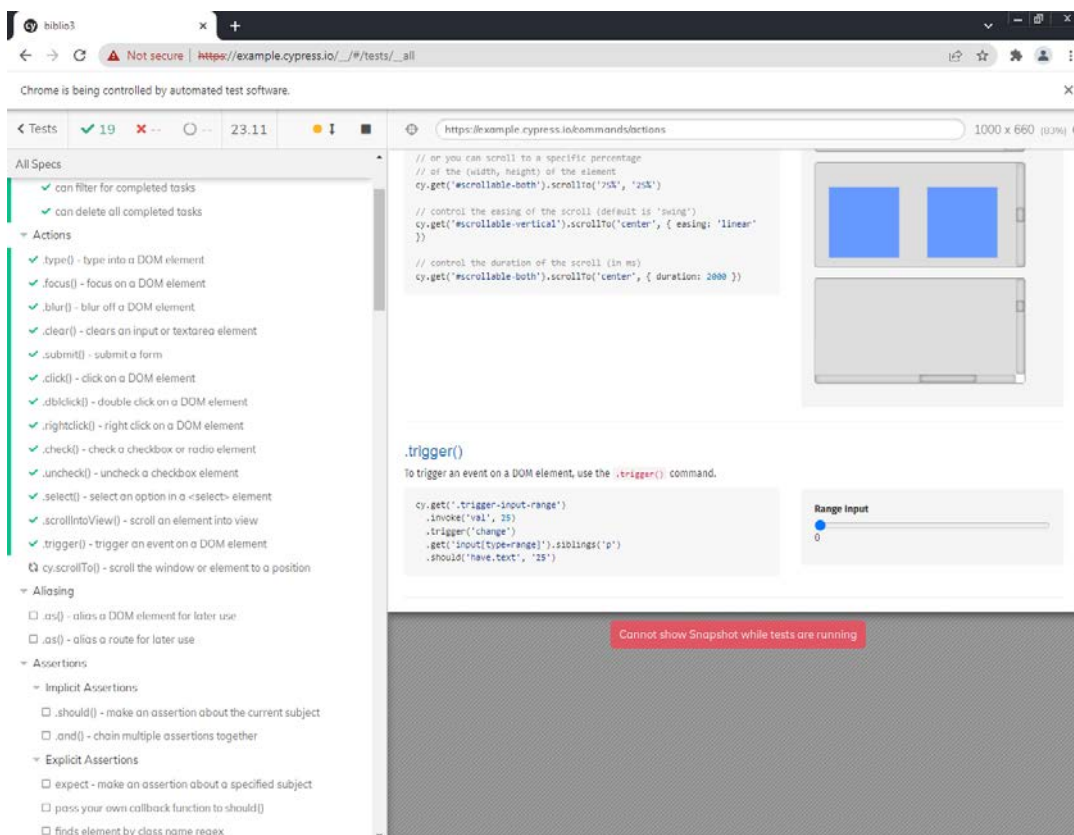
Prikaz stranice nakon uspešnog izvršavanja softverske funkcije za izmenu vrste bibliotečke jedinice nakon izvršene izmene u PHP naredbi I SQL kriptu.



Testiranje softvera Cypress alatom:



Prikaz rezultata izvršenog testiranja:



## 7. KORIŠTENI ALATI I SOFTVERI

- SAP/Sybase Power Designer 15.0,
- MySQL sistem za rukovanje bazama podataka,
- PHPMyAdmin web aplikaciju za administraciju baze podataka,
- Apache web server,
- PHP programski jezik za srednji aplikacioni sloj (Editor: Visual Studio Code),
- PHP/HTML/CSS/JS za korisnički interfejs (KI), uz korišćenje Bootstrap radnog okvira.

## 8. LITERATURA

1. Github repozitorijum, 2020., <https://github.com/>
2. Web HTML template, <https://freehtml5.co/>
3. W. Jason Gilmore Beginning PHP and MySQL: From Novice to Professional, Fourth Edition Copyright © 2010 by W. Jason Gilmore Apress, USA.
4. Fabio Cimo, Bootstrap Programming Cookbook, Web Code Geeks.