Aplikacija za praćenje fitness treninga

Wittendorfer, Edi

Undergraduate thesis / Završni rad

2017

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:135605

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2025-03-21**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

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

Stručni studij

Java aplikacija za praćenje fitness treninga

Završni rad

Edi Wittendorfer

Osijek, 2017.

SADRŽAJ

1. UVOD 1
1.1 Zadatak završnog rada 1
2. KORIŠTENE TEHNOLOGIJE 2
3. STRUKTURA APLIKACIJE 3
3.1 Funkcije sustava 5
3.2 Grafičko sučelje sustava (GUI) 13
3.3 Baza podataka 16
4. RAD S APLIKACIJOM 18
5. ZAKLJUČAK 19
LITERATURA 20
SAŽETAK 21
ABSTRACT 22
ŽIVOTOPIS 23
PRILOG 24

1. UVOD

U ovom radu biti će objašnjen način rada Java aplikacije za praćenje fitness treninga te njezin proces izrade. Aplikacija će zatražiti podatke o članu teretane, te podatke sprema u bazu podataka i iz zapisanih podataka aplikacija predlaže vježbe tom članu za određeni dan.

Aplikacija će sadržavati sljedeće funkcionalnosti:

- Prijava trenera ili trenerice teretane.
- Unos podataka o novom članu teretane.
- Pohrana podataka u bazu podataka.
- Prijedlog vježbi za članove ovisno o njihovim podacima i interesima.

U nastavku rada biti će opisano koje smo alate i tehnologije koristili za izradu aplikacije. Opisano će biti i detaljan način izrade grafičkog sučelja, baze po te ostale funkcije aplikacije

1.1 Zadatak završnog rada

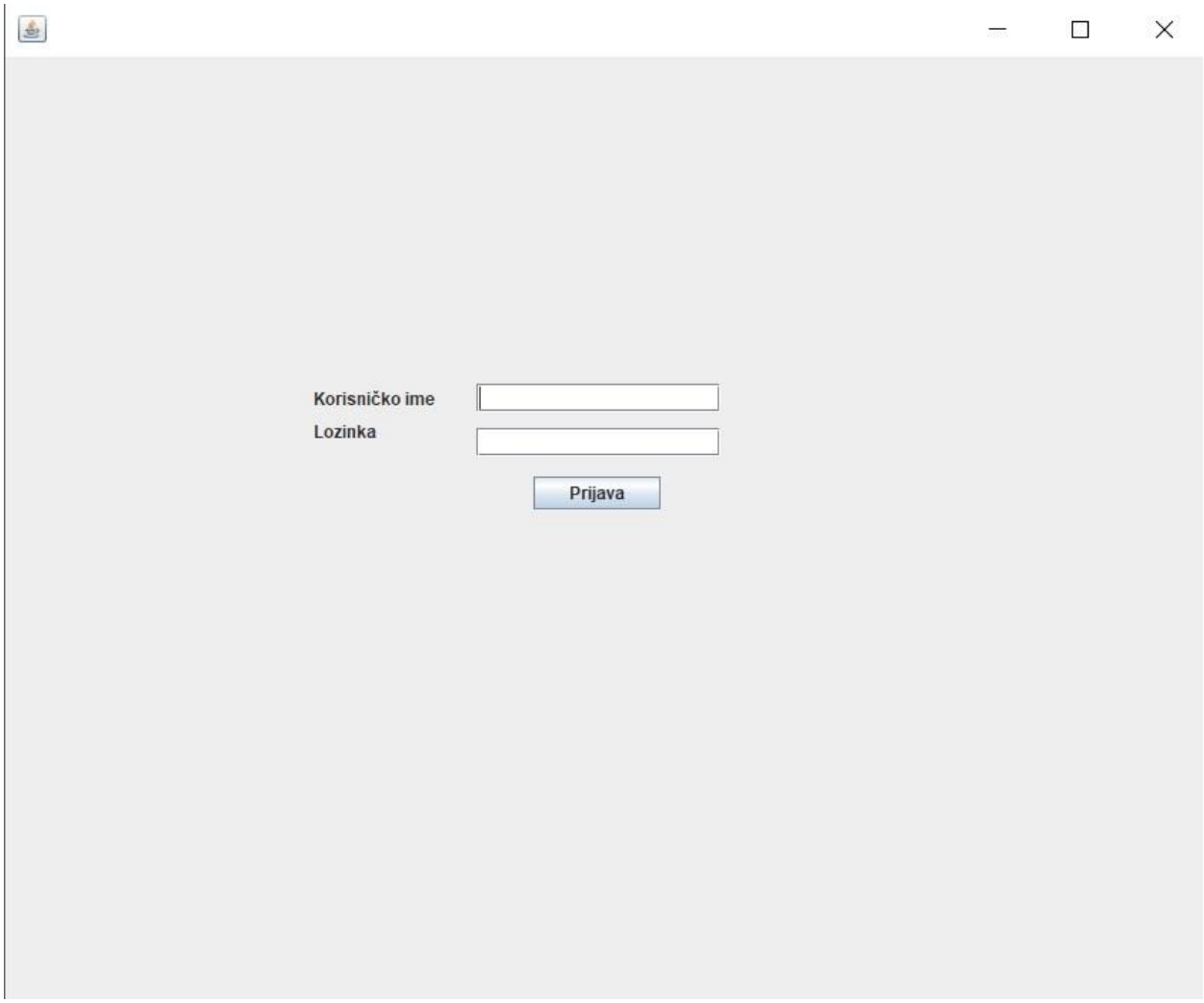
Zadatak rada je napraviti aplikaciju koja će olakšati posao trenerima i članovima kluba oko izrade plana treninga. Aplikacija treba preporučiti potrebne vježbe i nadzirati napredovanje članova.

2. KORIŠTENE TEHNOLOGIJE

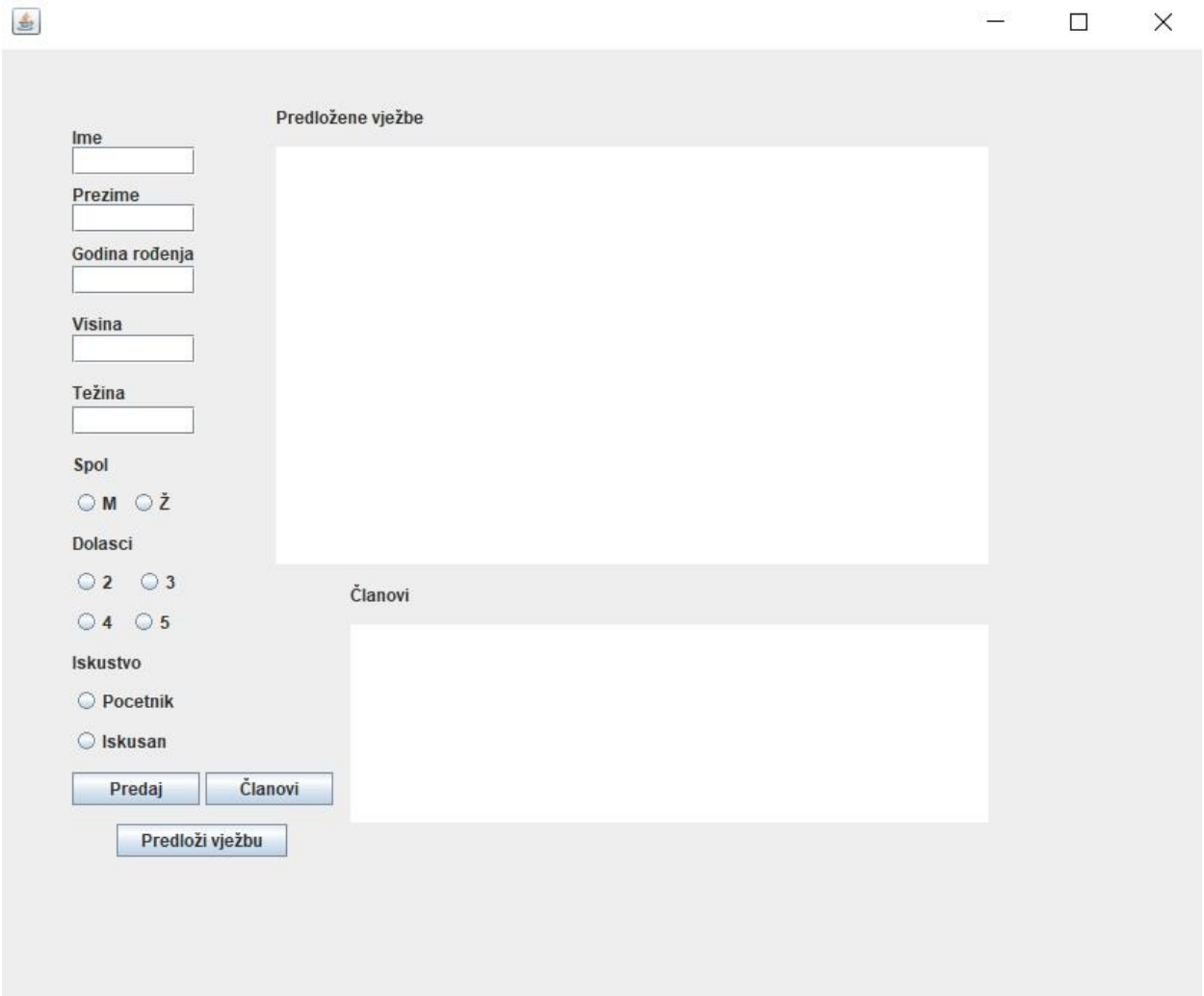
Tehnologije koje smo koristili za ove fitness aplikacije su Eclipse HTML, JAVA i XAMPP. Prilikom izrade aplikacije korištena je JAVA u Eclipse Oxygen. JAVA je objektno orijentirani programski jezik. Prednost JAVA-e je ta da se može pokrenuti na svim sustavima koji pordžavaju JAVA-u.Aplikacija je pisana u Eclipse Oxygen. Ova verzija podržava veliku raznolikost pluginova i biblioteka koje dodatno olakšavaju izradu aplikacija. MySQL je korišten za izradu, a XAMPP za razvoj baze podataka. MySQL se često koristi za upravljanje bazama podataka jer je otvorenog tipa. MySQL je čest izbor baze za projekte otvorenog koda, te se distribuira kao sastavni dio serverskih Linux distribucija, no također postoje inačice i za ostale operacijske sustave poput Mac OS-a, Windowse itd. XAMPP je vrlo jednostavna multi-platforma s kojom možemo instalirati Apache, MySQL, PHP, phpMyAdmin i još mnoge druge aplikacije korisne za razvoj/testiranje dinamičkih web stranica izravno na vašem računalu. HTML (HyperText Markup Language) je korišten za uređivanje uz alat JAVA-u Swing. HTML jezikom oblikuje se sadržaj i stvaraju se hiperveze hipertext dokumenta. HTML je jednostavan za uporabu i lako se uči, što je jedan od razloga njegove opće prihvaćenosti i popularnosti. Povezice unutar HTML dokumenata povezuju dokumente u uređenu hijerarhijsku strukturu i time određuju način na koji posjetitelj doživljava sadržaj stranica. Swing je razvijen kako bi osigurao sofisticiraniji skup GUI komponenti. Swing pruža prirodni izgled i osjećaj koji imitira izgled i dojam nekoliko platformi.

3. STRUKTURA APLIKACIJE

Ovo poglavlje objasniti će detaljno aplikaciju i opisati će kako svaka funkcija radi i koja je njena primjena.



Sl. 3.1. Prijava



Sl. 3.2. Izgled aplikacije

3.1 Funkcije sustava

Kako bih mogli pristupiti samoj aplikaciji moramo se prijaviti unosom korisničkog imena i lozinke te pritiskom na gumb „Prijava“. Unošenjem krivog korisničkog imena ili lozinke prikazati će se error poruka. Prilikom unošenja točnog korisničkog imena i lozinke dobivamo pristup aplikaciji. Pomoću tipkovnice popunjavamo informacije o osobi, njegovo/njezino ime, prezime, godinu rođenja, visinu i težinu. Pomoću radio gumba biramo spol osobe, koliko će često dolaziti tjedno na trening te njegovo/njezino iskustvo u fitnessu. Prilikom pritiska na gumb „Predaj“ unešene informacije šalju se i zapisuju u SQL bazu. Gumb „Članovi“ ispisati će sve članove koji se trenutno nalaze u bazi podataka u prozor „Članovi“. Gumb „Predloži vježbu“ ima dvije funkcije, osoba koja je novi član u teretani prilikom unošenja podataka može pritisnuti gumb i na osnovu svojih podataka dobije prijedlog vježbi u prozoru „Predložene vježbe“. Postoji i drugi slučaj gdje je osoba već član teretane te se unosom imena i prezimena dobiju predložene vježbe u istom prozoru.

```
btnPrijava.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent e) {
String loganje,paswordiranje;
loganje=textField.getText().toString();
paswordiranje=textField_1.getText.toString();

if(loganje.equals(login)&&paswordiranje.equals(pw)){
lblKorisnikoIme.setVisible(false);
lblLozinka.setVisible(false);
textField.setVisible(false);
textField_1.setVisible(false);
btnPrijava.setVisible(false);
                                    …
                                    }
else JOptionPane.showMessageDialog(frame,"Krivo korisničko ime ili lozinka. Pokušajte
opet.","Pogreška", JOptionPane.WARNING_MESSAGE);
}
});
```

Isječak koda 3.1. Funkcija za prijavu korisnika.

U ovom isječku koda vidimo kako funkcija provjerava je li unešeno korisničko ime i lozinka isto onome koje je zadano u aplikaciji. U slučaju da se korisničko ime i lozinka podudaraju zadanim dobivamo pristup samoj aplikaciji, u slučaju da uneseni podaci ne odgovaraju zadanima dobivamo error poruku: `"Krivo korisničko ime ili lozinka. Pokušajte opet."`.

```
btnPredaj.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent e) {
String imeb=textFieldIme.getText().toString();
String prezb=textFieldPrezime.getText().toString();
String godRodb=textFieldGodine.getText().toString();
float visinab=Float.parseFloat(textFieldVisina.getText().toString());
float tezinab=0;
tezinab=Float.parseFloat(textFieldTezina.getText().toString());
String spolb="";
int dolascib=0;
String iskustvob="";
if(radioButton.isSelected()) dolascib=Integer.parseInt(radioButton.getText());
if(radioButton_1.isSelected()) dolascib=Integer.parseInt(radioButton_1.getText());
if(radioButton_2.isSelected()) dolascib=Integer.parseInt(radioButton_2.getText());
if(radioButton_3.isSelected()) dolascib=Integer.parseInt(radioButton_3.getText());


if(rdbtnM.isSelected()) spolb=rdbtnM.getText().toString();
if(rdbtnZ.isSelected()) spolb=rdbtnZ.getText().toString();

if(rdbtnIskusni.isSelected()) iskustvob=rdbtnIskusni.getText().toString();
if(rdbtnPocetnik.isSelected()) iskustvob=rdbtnPocetnik.getText().toString();


try {
con = DriverManager.getConnection(host, uName, uPass);
stmt = con.createStatement();
String sql = "INSERT INTO
clanovi(Ime,Prezime,DatumRodenja,Visina,Tezina,Spol,Iskustvo,Dolasci) " + "VALUES ('"
+ imeb + "', '"+ prezb + "','" + godRodb + "','" + visinab + "','" + tezinab+ "','"
+spolb + "','" +iskustvob + "','" + dolascib+ "')";
stmt.executeUpdate(sql);

} catch (SQLException err) {
System.out.println(err.getMessage());
}
}
});
```

Isječak koda 3.2. Funkcija za predaju informacija u bazu

Funkcija sprema sve unešene podatke korisnika te pritiskom na gumb „Predaj“ šalje podatke u bazu podataka.

```
btnlanovi.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent e) {
try {
java.sql.PreparedStatement preparedStatement = null;
ResultSet rs;
con = DriverManager.getConnection(host, uName,Pass);
String sql = "SELECT * FROM clanovi;";
stmt = con.createStatement();
preparedStatement = con.prepareStatement(sql);
rs = preparedStatement.executeQuery();
table.setModel(DbUtils.resultSetToTableModel(rs));


} catch (SQLException err) {
System.out.println(err.getMessage());
}
}
});
```

Isječak koda 3.3. Funkcija za prikaz članova iz baze

U slučaju da nas zanimaju neki od podataka korisnika možemo te informacije dobiti pritiskom na tipku „Članovi“ te će se baza podataka ispisati unutar aplikacije.

```
btnNewButton.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent e) {
if(radioButton.isSelected()) dolazak=radioButton.getText();
if(radioButton_1.isSelected()) dolazak=radioButton_1.getText();
if(radioButton_2.isSelected()) dolazak=radioButton_2.getText();
if(radioButton_3.isSelected()) dolazak=radioButton_3.getText();
if(!textFieldGodine.getText().isEmpty()){
try {
switch(dolazak){
case "2": editorPaneVjezbe.setText("VJEZBE ZA 2 DANA break;
case "3": editorPaneVjezbe.setText("VJEZBE ZA 3 DANA");break;
case "4": editorPaneVjezbe.setText("VJEZBE ZA 4 DANA");break;
case "5": editorPaneVjezbe.setText("VJEZBE ZA 5 DANA");break;
default: break;

}
}catch(Exception d){};}
```

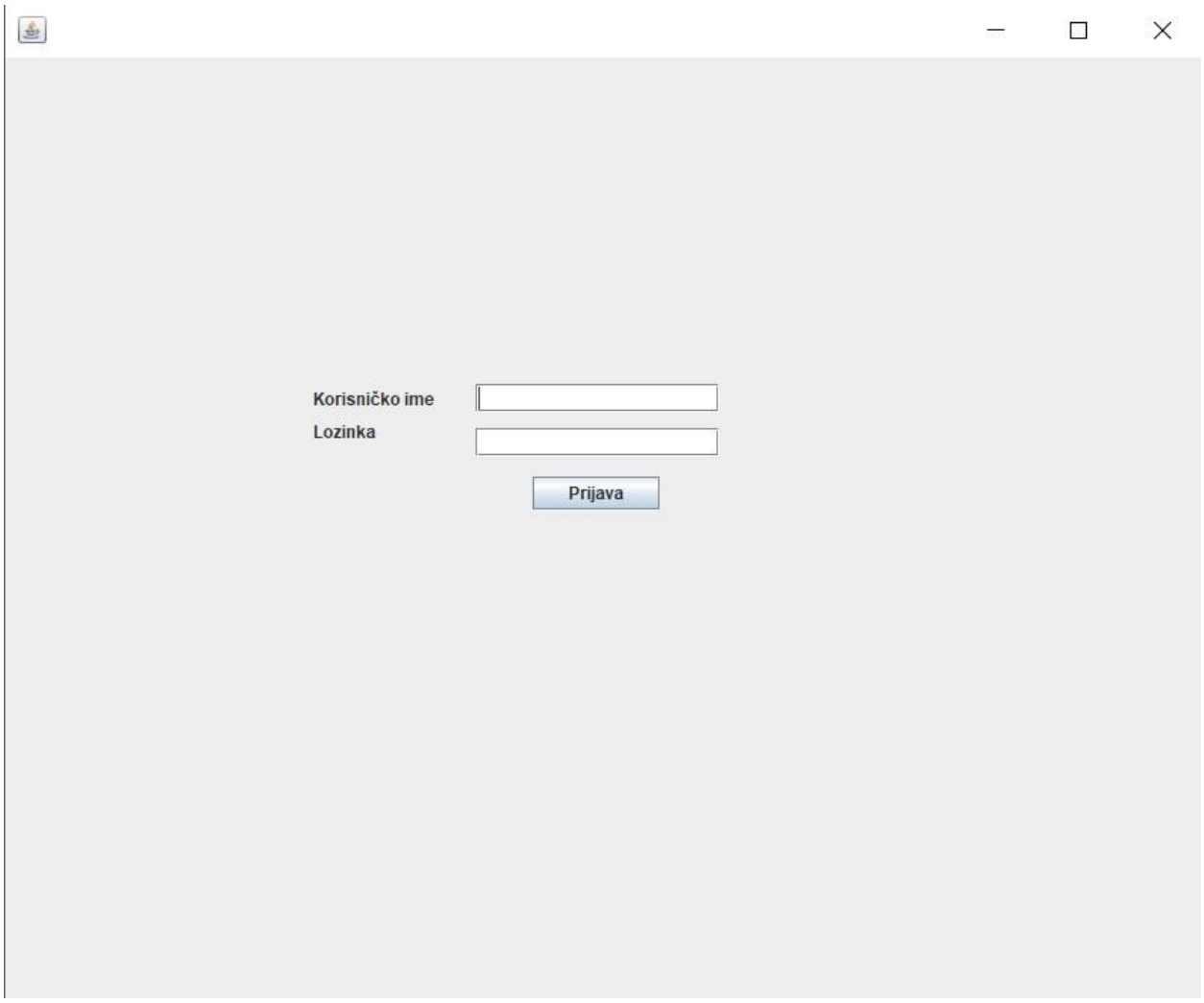
Isječak koda 3.4. Funkcija za prikaz predloženih vježbi za novo unešenog člana

Ova funkcija nam omogućuje prijedlog vježbi za novog člana teretane nakon što je upisao svoje podatke unutar aplikcije. Prilikom pritiska na gumb „Predloži vježbu“ na osnovu unešenih podataka aplikacija predlaže listu vježbi u prozoru „Predložene vježbe“.

```
else{
int dolasci=-1;
try{
String imeb="",prezimeb="";
imeb=textFieldIme.getText().toString();
prezimeb=textFieldPrezime.getText().toString();
      java.sql.PreparedStatement preparedStatement = null;
ResultSet rs;
con = DriverManager.getConnection(host, uName,Pass);
String sql ="SELECT Dolasci FROM clanovi Where Ime='" + imeb + "' AND
Prezime='"+prezimeb+"';" ;

stmt = con.createStatement();
preparedStatement = con.prepareStatement(sql);
rs = preparedStatement.executeQuery();
while(rs.next()) {
dolasci=rs.getInt("Dolasci");
System.out.println(dolasci);
}
switch(dolasci){
case 2: editorPaneVjezbe.setText("VJEZBE ZA 3 DANA break;
case 3: editorPaneVjezbe.setText("VJEZBE ZA 3 DANA");break;
case 4: editorPaneVjezbe.setText("VJEZBE ZA 4 DANA");break;
case 5: editorPaneVjezbe.setText("VJEZBE ZA 5 DANA");break;
default: break;}
}
catch(Exception c){};
}}
});
```

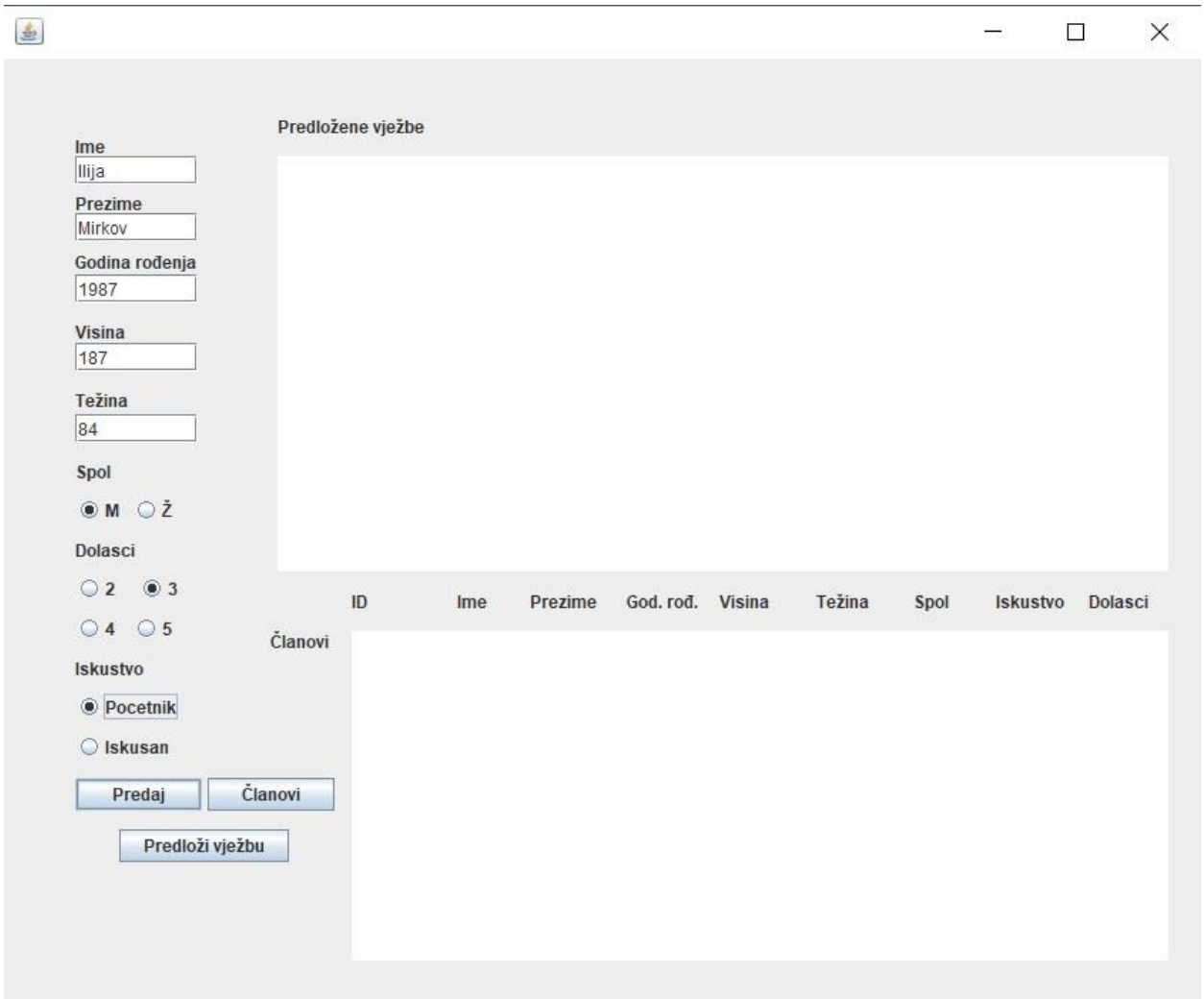
Isječak koda 3.5. Funkcija za prikaz predloženih vježbi za postojećeg člana

U slučaju da je osoba već član teretane, unosimo ime i prezime te osobe i na taj način ju pronalazimo u bazi podataka. Kada u bazi pronađemo osobu s odgovarajućim imenom i prezimenom očitavamo njegove/njezine podatke pomoću tipke „Predloži vježbu“ i na osnovu toga dobijemo listu vježbi u prozoru „Predložene vježbe“.



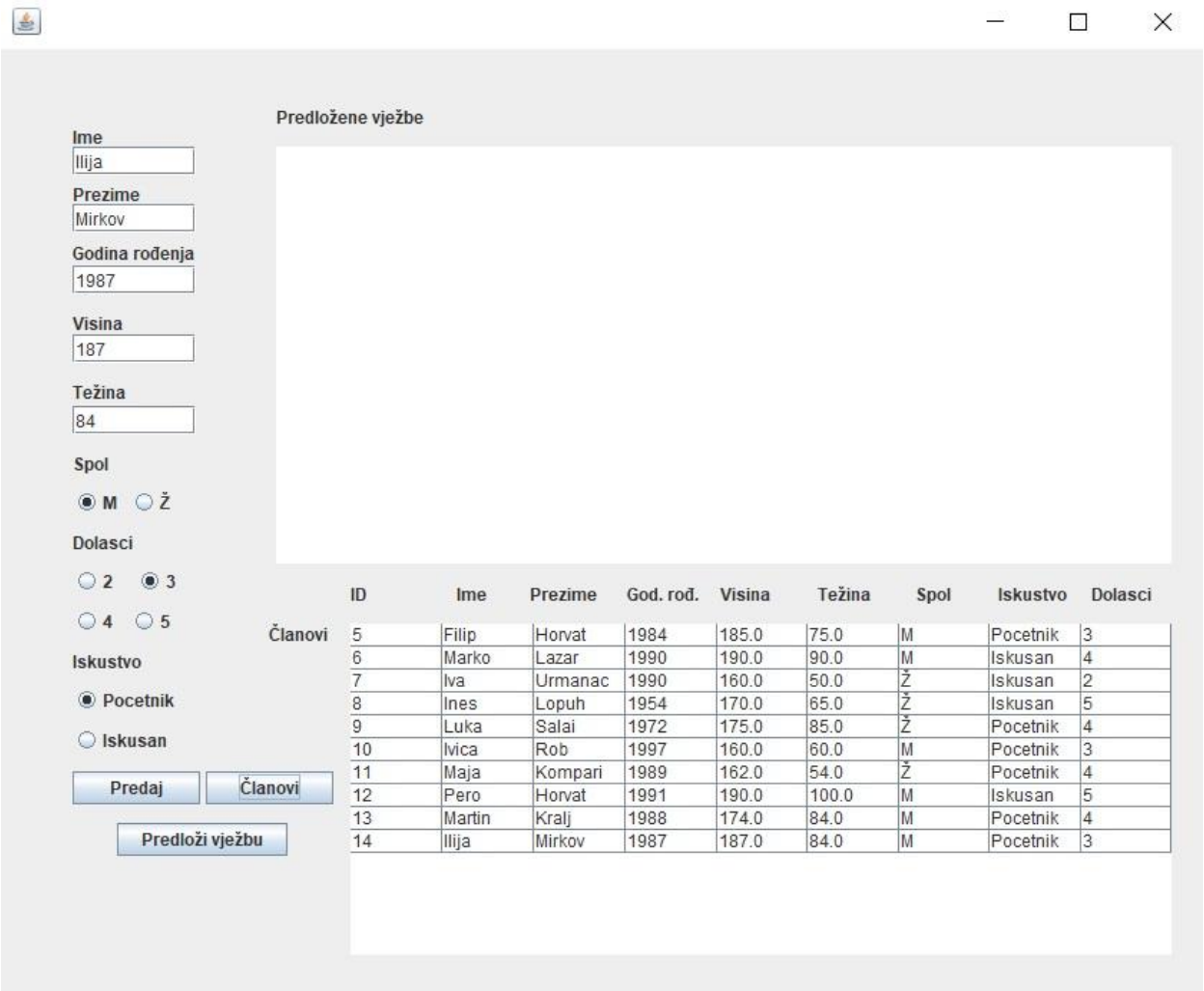
Sl. 3.3. Prijava trenera

Na slici 3. možemo vidjeti prijavu koju trener mora ispuniti kako bi pristupio podacima članova teretane i predloženim vježbama.



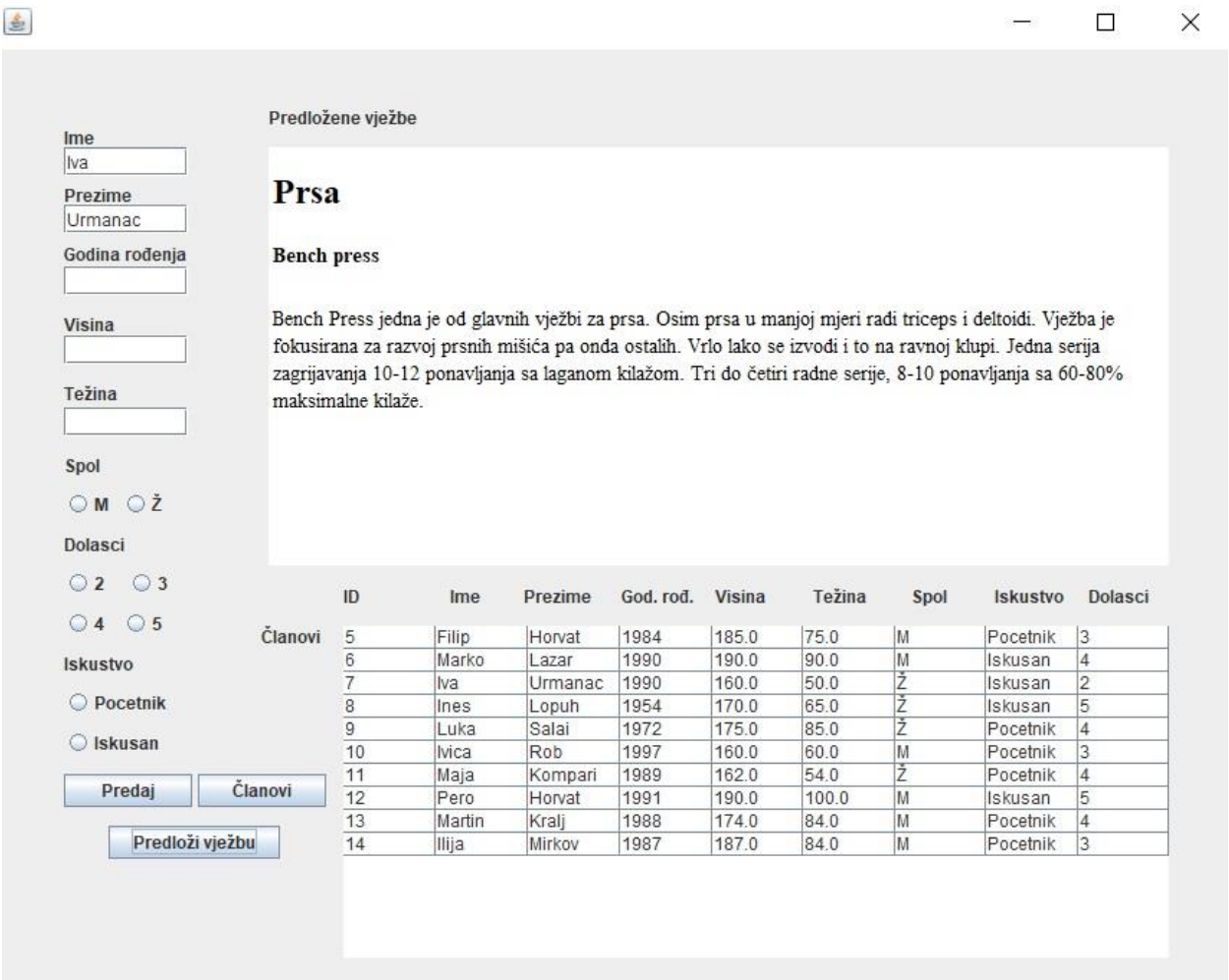
Sl. 3.4.Unos novog člana

Trener popunjava informacije o novom članu teretane s njegovim/njezinim pripadajućim informacija. Uz informacije popunjava koliko će puta osoba dolaziti tjedno u teretanu te dali je ta osoba početnik ili iskusna u fitnessu.



Sl. 3.5. Unos osobe u bazu i ispis baze

Na slici 5. možemo vidjeti kako se osoba unosi u bazu prilikom tipke „Predaj“ te ispis svih članova teretane pritiskom tipke „Članovi“.

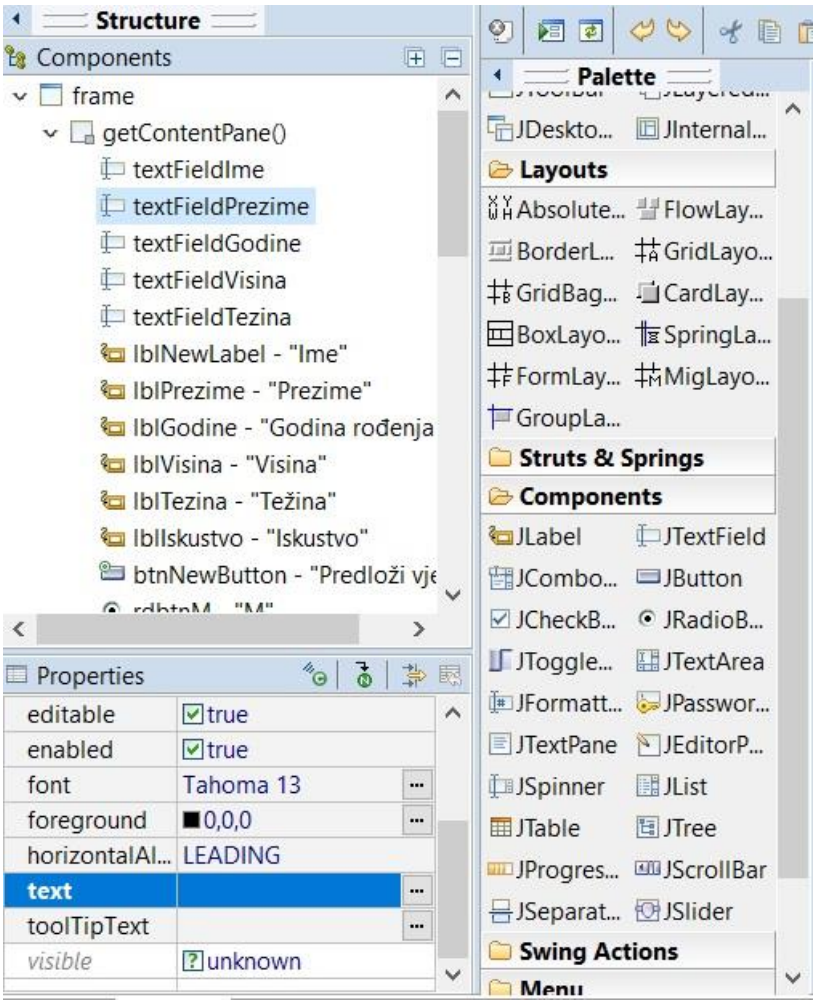


Sl. 3.6. Prijedlog vježbe

Upisom imena i prezimena osobe u aplikaciju te pritiskom na tipku predloži vježbu, aplikacija očitava podatke člana iz baze podataka te na osnovu toga predlaže vježbe za tog člana.

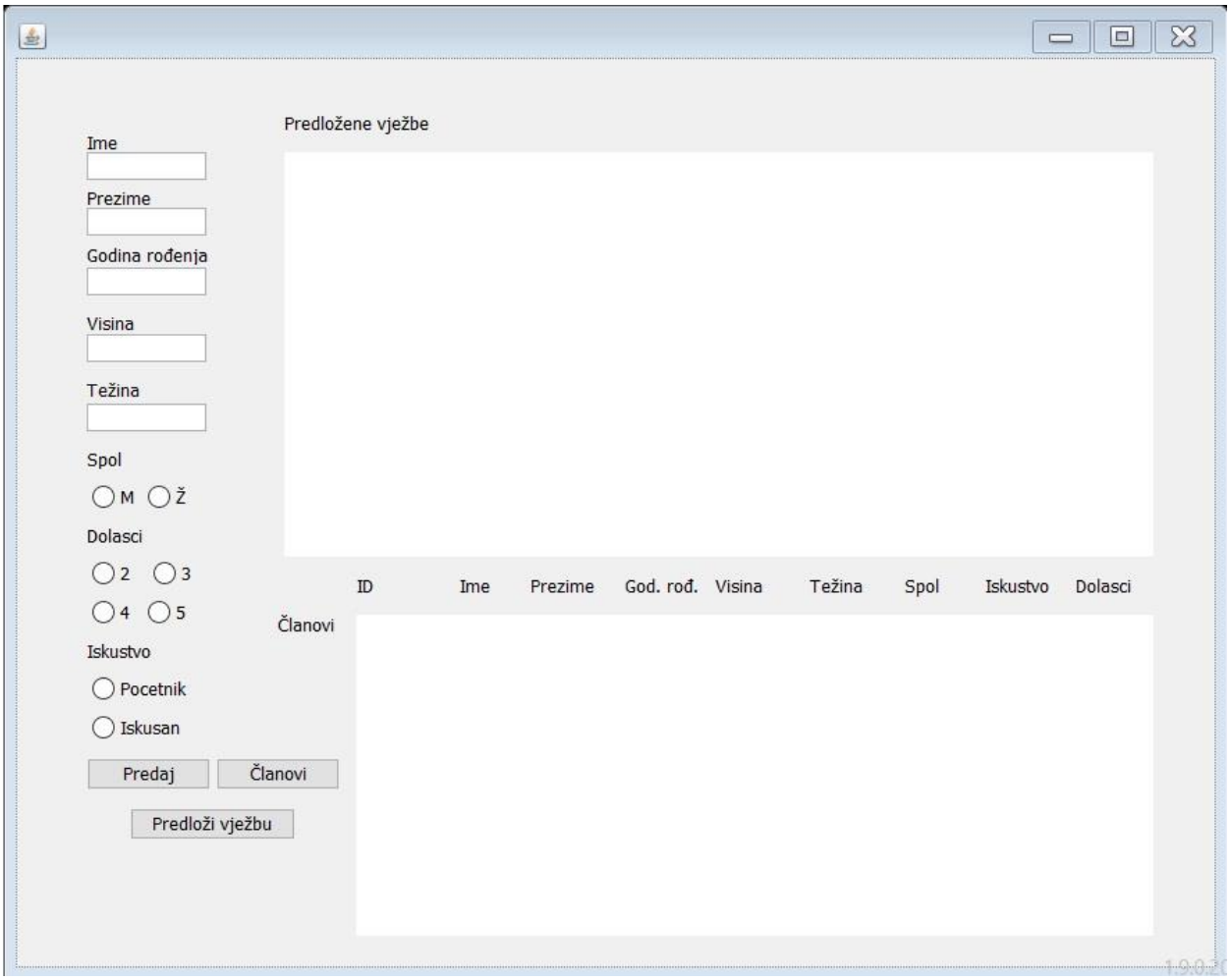
3.2 Grafičko sučelje sustava (GUI)

Pomoću Eclipse Oxygen-a možemo stvoriti grafičko sučelje sustava (GUI) pomoću njega pravimo dizajn naše aplikacije. Da bi nam izrada dizajna bila što jednostavnija koristimo JAVA Swing alat. Uz JAVA Swing koristimo i dodatak WindowBuilder on nam omogućuje kreiranje grafičkog sučelja na način da iz strukture GUI-a povlačimo elemente potrebne za našu aplikaciju.



Sl. 3.7. Elementi u dodatku WindowBuilder

Na slici 7. možemo vidjeti koje smo elemente koristili pri izradnji aplikacije i njenom dizajnu. Ispod elemenata možemo mijenjati njihove postavke od imena komponenti, njihovih valuta itd.



Sl. 3.8. Izgled aplikacije u dodatku WindowBuilder

Na slici 8. prikazani su elementi složeni u WindowBuilder-u. Podaci se ispisuju unutar JEditorPane i JTable. Elementi sadžavaju i scollPane koji omogućuje prigaz svih ispisanih informacija jer se u protivnom dio informacija ne bih prikazalo. JEditorPane ispisuje podatke koji se mogu oblikovat sa HTML-om odnosno možemo tekst dizajnirati poput web stranice. Pomoću JTable ispisati ćemo elemente baze podataka kao tablicu. U aplikaciji još možemo vidjeti elemente:3 Button-a, 21 Label-a za tekst, sedam textField za unos podataka člana. Postoji i dio aplikacije za prijavu koji se ne vidi radi ostalih elemenata u WindowBuilder-u.

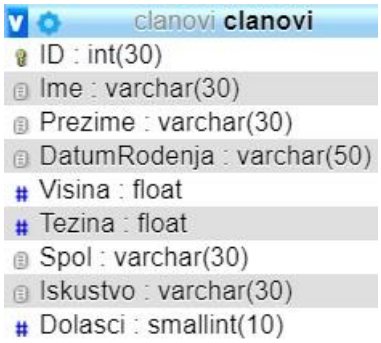
```
lblKorisnikoIme.setVisible(false);
lblLozinka.setVisible(false);
textField.setVisible(false);
textField_1.setVisible(false);
btnPrijava.setVisible(false);

textFieldIme.setVisible(true);
textFieldPrezime.setVisible(true);
textFieldGodine.setVisible(true);
textFieldVisina.setVisible(true);
textFieldTezina.setVisible(true);
                                    ...
```

Isječak koda 3.6. Mjenjanje vidljivosti prijave i ostatka aplikacije.

Prilikom pritiska na gumb “Prijava” dio aplikacije koji izvršava prijavu postane ne vidljiv, dok dio koji nam služi za upis korisnika u bazu i predlaganje vježbi postane vidljiv.

3.3 Baza podataka

Pomoću MySQL-a napisana je baza podataka, koju možemo uređivati pomoću phpMyAdmin alata. Alat nam omogućuje kreiranje i biranje baze i njezinih elemenata. Bazu podataka koristit ćemo samo za upis članova u teretanu.



Sl. 3.9. Dijagram baze podataka.

+ Options

		ID	Ime	Prezime	DatumRodenja	Visina	Tezina	Spol	Iskustvo	Dolasci
Edit Copy Delete		5	Filip	Horvat	1984	185	75	M	Pocetnik	3
Edit Copy Delete		6	Marko	Lazar	1990	190	90	M	Iskusan	4
Edit Copy Delete		7	Iva	Urmanac	1990	160	50	Ž	Iskusan	2
Edit Copy Delete		8	Ines	Lopuh	1954	170	65	Ž	Iskusan	5
Edit Copy Delete		9	Luka	Salai	1972	175	85	Ž	Pocetnik	4
Edit Copy Delete		10	Ivica	Rob	1997	160	60	M	Pocetnik	3
Edit Copy Delete		11	Maja	Kompari	1989	162	54	Ž	Pocetnik	4
Edit Copy Delete		12	Pero	Horvat	1991	190	100	M	Iskusan	5
Edit Copy Delete		13	Martin	Kralj	1988	174	84	M	Pocetnik	4
Edit Copy Delete		14	Ilija	Mirkov	1987	187	84	M	Pocetnik	3

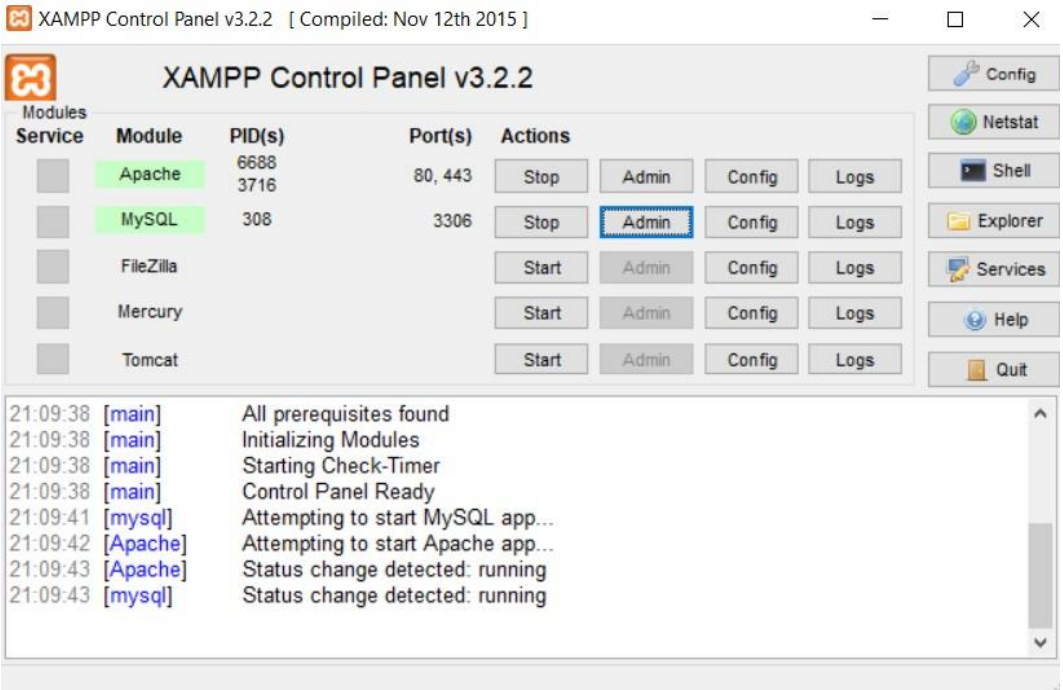
Sl. 3.10. Izgled baze podataka.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(30)			No	None		AUTO_INCREMENT	Change Drop Primary Unique Index Spatial Fulltext More
2	Ime	varchar(30)	utf8_croatian_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext More
3	Prezime	varchar(30)	utf8_croatian_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext More
4	DatumRodenja	varchar(50)	utf8_croatian_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext More
5	Visina	float			No	None			Change Drop Primary Unique Index Spatial Fulltext More
6	Tezina	float			No	None			Change Drop Primary Unique Index Spatial Fulltext More
7	Spol	varchar(30)	utf8_croatian_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext More
8	Iskustvo	varchar(30)	utf8_croatian_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext More
9	Dolasci	smallint(10)			No	None			Change Drop Primary Unique Index Spatial Fulltext More

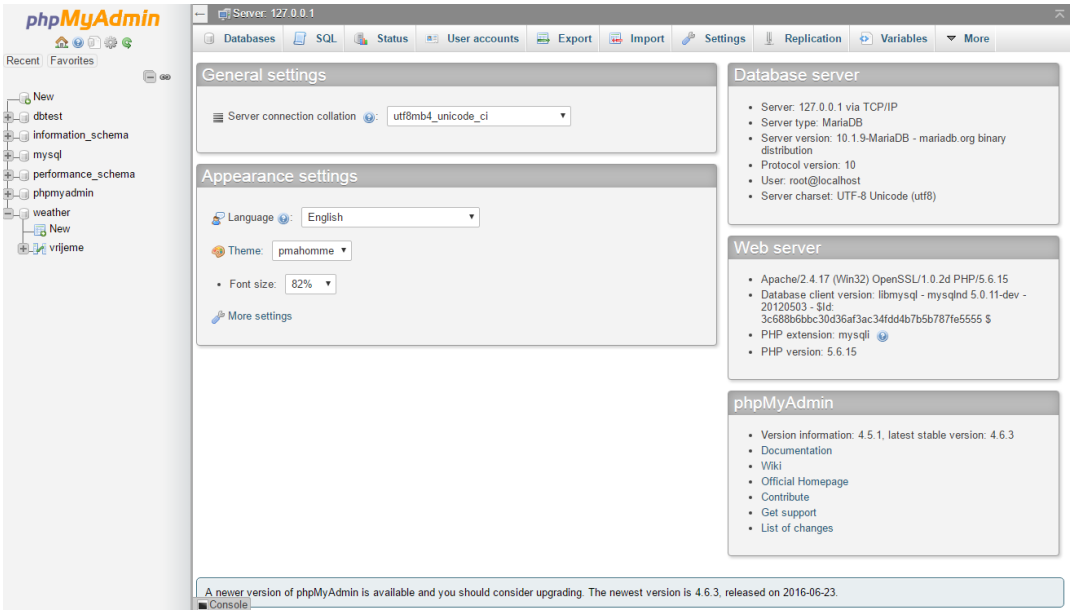
Sl. 3.11. Kreiranje baze podataka.

Ovdje možemo mijenjati, brisati i dodavati elemente u tablici.

XAMPP nam omogućuje kreiranje poslužitelja koji nam omogućuje simulaciju baze podataka. Pomoću njega testirati ćemo našu bazu prilikom izrade aplikacije, pogodan je jer se baza može izmjenjivati dok aplikacija radi.



Sl. 3.12. XAMPP



Sl. 3.13 phpMyAdmin

Kako bih smo radili s bazama koristimo PhpMyAdmin On nam služi za izradu baza i za međusobno povezivanje ostalih.

4. RAD S APLIKACIJOM

Aplikaciju koristi trener teretane koji ima nadzor nad podacima svojih članova i omogućuje mu prijedlog vježbi članova teretane i onih koji će to postati. Prilikom otvaranja aplikacije trener se mora prijaviti sa korisničkim imenom i lozinkom kako bi pristupio aplikaciji. Na ovaj način se štite podaci članova teretane od ne ovlaštenih korisnika. Korištenje aplikacije je veoma jednostavno. Nakon prijave trenera, trener ima mogućnost unosa novog člana ili prijedlog vježbe postojećim. Putem tipkovnice trener popunjava ime, prezime, godinu rođenja, težinu, visinu, spol, tjedni dolazak osobe te njegovo/njezino iskustvo u fitnessu. Nakon popunjavanja trener ima mogućnost spremanja podataka u bazu podataka ili predložak vježbe za tu osobu ovisno o unesenim podacima. U slučaju da je osoba već član teretane, trener ima mogućnost preporuke vježbi za tu osobu putem unosa njegovog/njezinog imena i prezimena u aplikaciju. Trener također ima nadzor nad svim podacima koje mu je član predložio za vrijeme upisa u aplikaciju putem gumba „Članovi“. Sve informacije se ispisuju unutar same aplikacije za to predviđena mjesta.

5. ZAKLJUČAK

Aplikacija je u punoj funkciji i može se koristit u teretanama. Prilikom testiranja nisu pronađeni „bug-ovi“ koji bi mogli narušavati funkcionalnost aplikacije. Mana je ta da aplikacija treba pristup internetu kako bi imala pristup članovima teretane, te računalo mora imati instaliran XAMPP kako bih mogao pristupiti bazi podataka. Prednost aplikacije je sigurnost svih članova jer aplikacija zahtjeva prijavu ovlaštene osobe. Aplikacija se uvijek može unaprijediti sa boljim i privlačnijim grafičkim sučeljem i boljom selekcijom vježbi za članove.

LITERATURA

[1] ORACLE ; http://www.oracle.com ; 3.5.2017.

[2] STACK OVERFLOW ; http://www.stackoverflow.com ; 20.4.2017.

[3] YOUTUBE/THENEWBOSTON ; https://www.youtube.com/user/thenewboston ;18.7.2017.

[4] W3SCHOOLS ; https://www.w3schools.com ; 16.8.2017.

SAŽETAK

Java aplikacija za praćenje fitness treninga je vrlo fleksibilna što se tiče operativnih sustava. Gotovo svaki uređaj danas podržava JAVA-u te samim time aplikacija može biti pokrenuta sa bilo kojeg uređaja. Aplikacija služi za praćenje fitness treninga člana teretane ili osobe koja to želi postati putem unosa podataka u aplikaciju. Podatci o osobi se spremaju unutar baze podataka kojoj pristup ima samo trener teretane. Veoma se lako koristi i nije zahtjevna od strane performansi računala.

Ključne riječi: API, Baza podataka, Eclipse, phpMyAdmin, XAMPP.

ABSTRACT

Java fitness tracking application

The Java application for tracking fitness training is very flexible as far as operating systems are concerned. Almost every device supports JAVA today, and the application can now be launched from any device. The application is used to track fitness training for a member of a gym or a person who wants to become a member of the application. Personal information is stored in a database that accesses only a coach of the gym. It is very easy to use and not demanding by computer performance.

Keywords: API, Database, Eclipse, phpMyAdmin, XAMPP.

ŽIVOTOPIS

Edi Wittendorfer rođen je 7. veljače 1995 u Osijeku, Hrvatska. Adresa stanovanja je Bana Jelačića 32, 31300 Beli Manastir. Osnovnu školu je završio u Osnovnoj školi Beli Manastir 2009. godine nakon toga pohađa i završava Prvu srednju školu Beli Manastir te stječe zvanje tehničar za računalstvo 2013. godine. Akademske godine 2013./2014. upisuje stručni studij informatike na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku.

Potpis:_____

PRILOG

Na CD-u priloženom uz završni rad nalazi se .doc i verzija završnog rada te kod programa.