



SVEUČILIŠTE U MOSTARU  
FAKULTET STROJARSTVA RAČUNARSTVA I ELEKTROTEHNIKE

## **Programiranje u Javi**

*Projektni zadatak: **Gym System***

**Zvonimir Vištica, Robert Čačija, Josip Kordić**

Akadska godina 2020/2021.

### PROJEKTNI ZADATAK:

#### Opis problema:

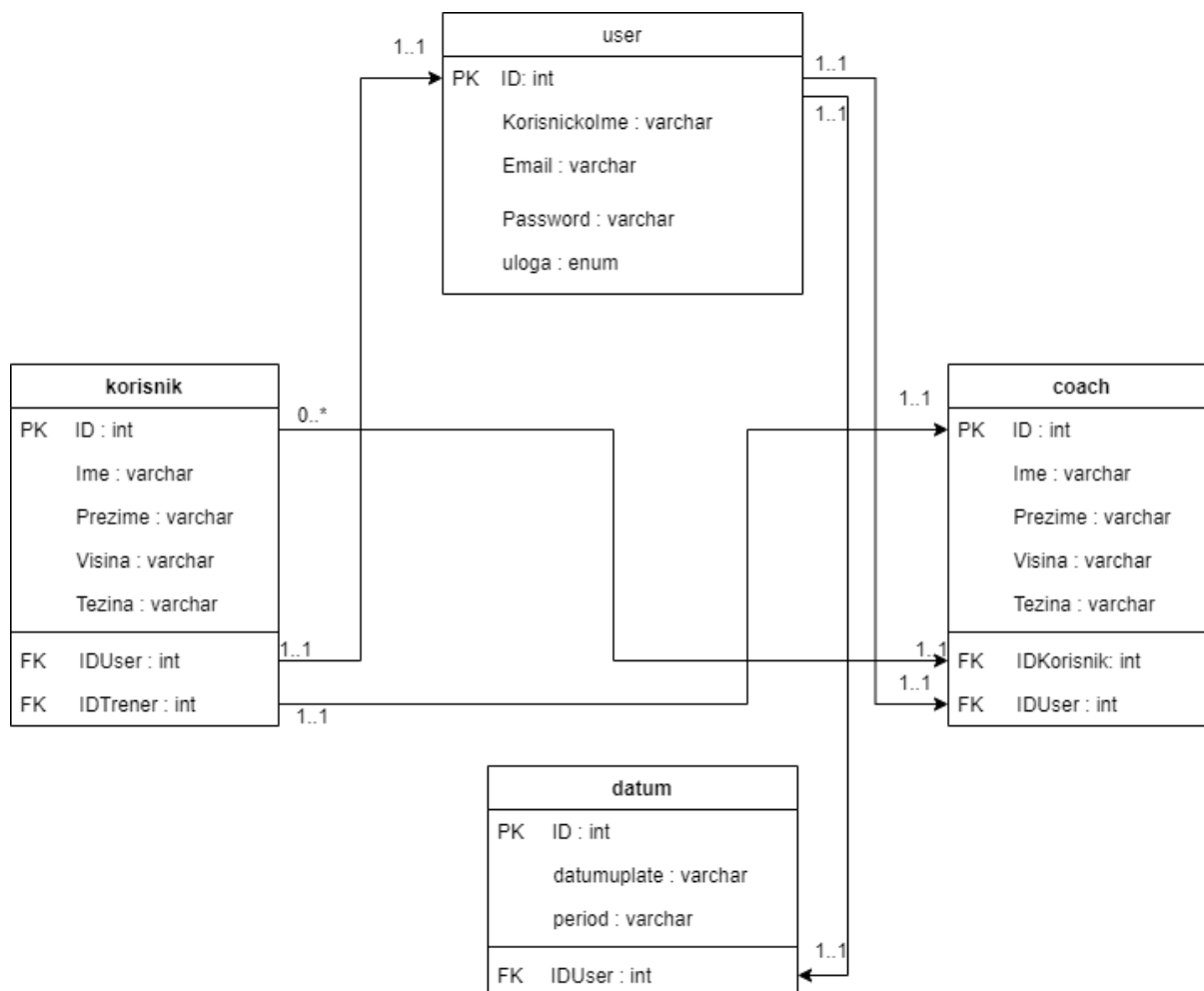
Naša zadaća je napraviti sustav kojim bi se riješio problem sustava u teretanama, odnosno načina na koji su pojedini dijelovi teretane što uključuje administrativni dio, trenere i same članove povezani u cjelinu. Prvi od problema kojem pristupamo je administrativni. Administrator je osoba koja sve treba objediniti te ona kao najviši stupanj korisnika treba imati pravo uređivanja i postavljanja članarina te informacija o samim članovima, brisanja korisnika i stvaranja novih te uvođenja novih komponenti prema potrebama same teretane. Taj problem će se riješiti međusobnim povezivanje samih tablica koji sadrže informacije o korisnicima, trenerima i samim administratorima. Kroz prikupljene informacije omogućujemo kontroliranje i manipulaciju podataka potrebnu kako bi sustav neometano funkcionirao. Nakon što se taj problem riješi na red dolaze treneri, osobe koje će biti povezane sa svojom grupom članova koje treniraju. Oni trebaju imati podatke bitne samo za njih bez ikakvog uvida i mogućnosti manipulacije nad podacima koji ne bi trebali mijenjati, niti imati pristupa. Ti podaci podrazumijevaju ID, ime, prezime, datum uplate. Za trenere bih tako trebali omogućiti da mogu utjecati samo na vrijeme koje je dogovoreno s članom njegove grupe, odnosno trebali bi također dobiti i potvrdu od strane člana. Ukoliko ne može doći do dogovora oko termina koji oboje odgovara, sustav traži novi datum u kojem bi obje strane mogle pristupiti treningu. Treneri bi isto tako trebali imati i uvid u napredak njihovih klijenata, to jest način na koji bi pratili pomak u fizičkoj spremi od početka do kraja suradnje koju određuje sam klijent. Nakon završetka međusobne suradnje sam administrator uklanja poveznicu između trenera i klijenta. Nadalje, prelazimo na korisnika koji bi kao član teretane trebao uredno plaćati članstvo u teretani. Na dan plaćanja, administrator bi podatke o plaćanju i periodu za koji je plaćena članarina spremio u bazu pod Identifikacijski broj (ID) tog korisnika. Trebamo naći način da povežemo korisnika s određenim trenerom, prema njegovim potrebama. Mi tu mogućnosti prepuštamo samom korisniku. Na izbor ima kondicijske trenere, fitness i crossfit trenere koje odabire s obzirom na željene ciljeve. Kao još jedna od mogućnosti za korisnika, trebalo bi im se omogućiti da sami unesu svoje podatke, odnosno ime, prezime, visinu, težinu koju bi trener mogao vidjeti. Trener računa BMI korisnika te s obzirom na to preporučuje najbolje vježbe koje bi im odgovarale. Kroz realizaciju svih ovih sustava i baza trebali bi omogućiti održavanje jednog sustava koji bi bez ikakvih problema mogao pratiti i olakšati rješavanje postojećih zahtjeva na koje se nailazi kada je u pitanju vođenje teretane.

### 1. Modeliranje podataka

Prvi dio zadatka se radi u .

- Poglavlje 1.1. (Relacijski model podataka) - potrebno je staviti sliku relacijskog modela kreiranog u MS Accessu.
- Poglavlje 1.2. (Opis tablica u relacijskom modelu) - potrebno je navesti i opisati sve tablice (i attribute unutar tablica) iz dobivenog modela te na kraju priložiti sliku
- Poglavlje 1.3. (Aplikacija u Javi) - sadrži screenshotove kreiranih formi Javi.

#### 1.1. Relacijski model podataka



#### 1.2. Opis tablica u relacijskom modelu

##### user

Naziv atributa	Integritet (PK/FK, NULL / NOT NULL)	Kratki opis atributa
ID	PK	Identifikacijski broj
Korisnickolme	NOT NULL	Ime u sustavu
Email	NOT NULL	Kontakt
Password	NOT NULL	Lozinka za prijavu u sustav

Uloga	NOT NULL	Administrator, trener ili korisnik
-------	----------	------------------------------------

### coach

Naziv atributa	Integritet (PK/FK, NULL / NOT NULL)	Kratki opis atributa
ID	PK	Identifikacijski broj
Ime	NOT NULL	Ime trenera
Prezime	NOT NULL	Prezime trenera
Vrsta	NOT NULL	Kondicijski, fitness ili crossfit trener
Visina	NOT NULL	Visina trenera
Tezina	NOT NULL	Težina trenera
IDKorisnika	FK	Povezivanje s korisnikom
IDUser	FK	Povezivanje s glavnom tablicom user

### korisnik

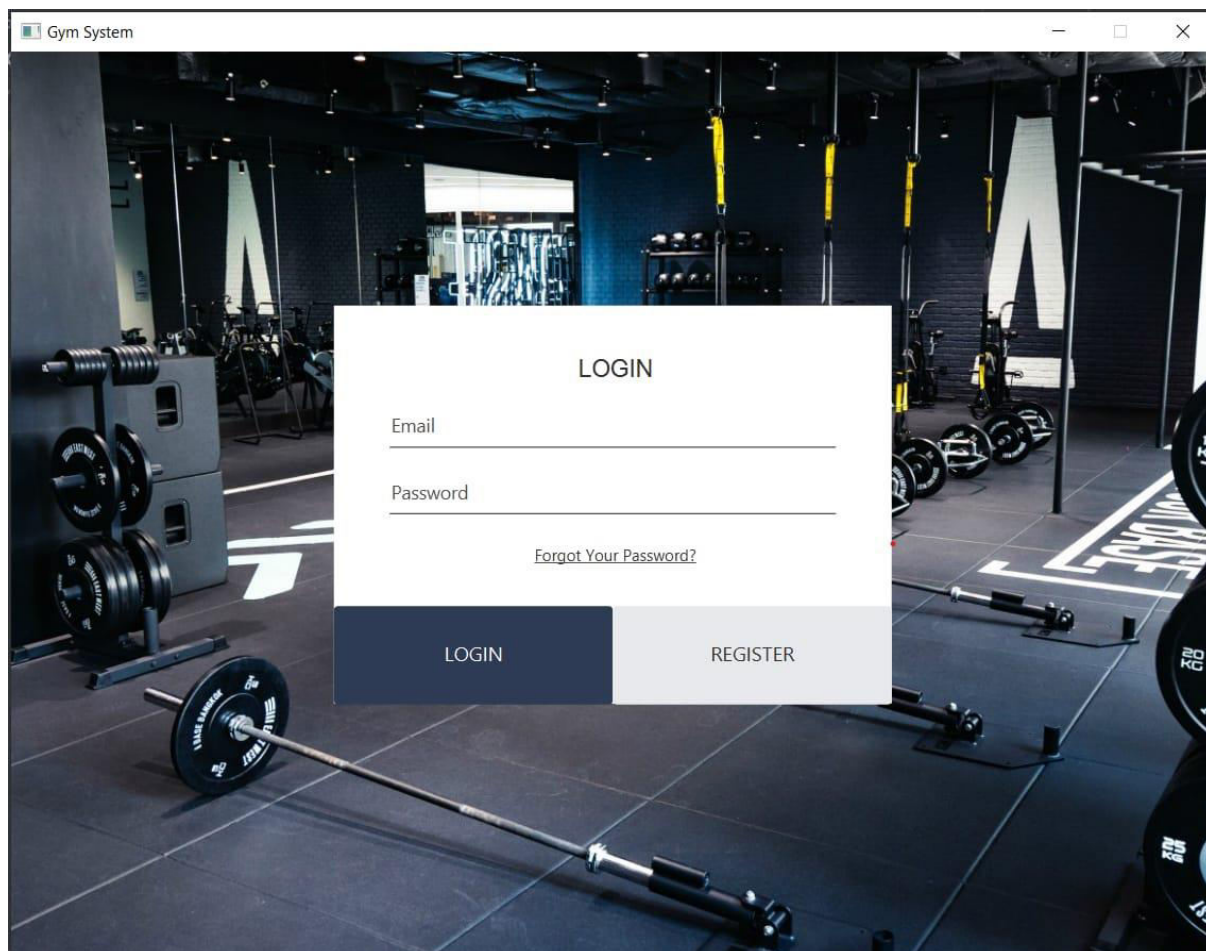
Naziv atributa	Integritet (PK/FK, NULL / NOT NULL)	Kratki opis atributa
ID	PK	Identifikacijski broj
Ime	NOT NULL	Ime trenera
Prezime	NOT NULL	Prezime trenera
Vrsta	NOT NULL	Kondicijski, fitness ili crossfit trener
Visina	NOT NULL	Visina trenera
Tezina	NOT NULL	Težina trenera
IDTrener	FK	Povezivanje s trenerom
IDUser	FK	Povezivanje s glavnom tablicom user

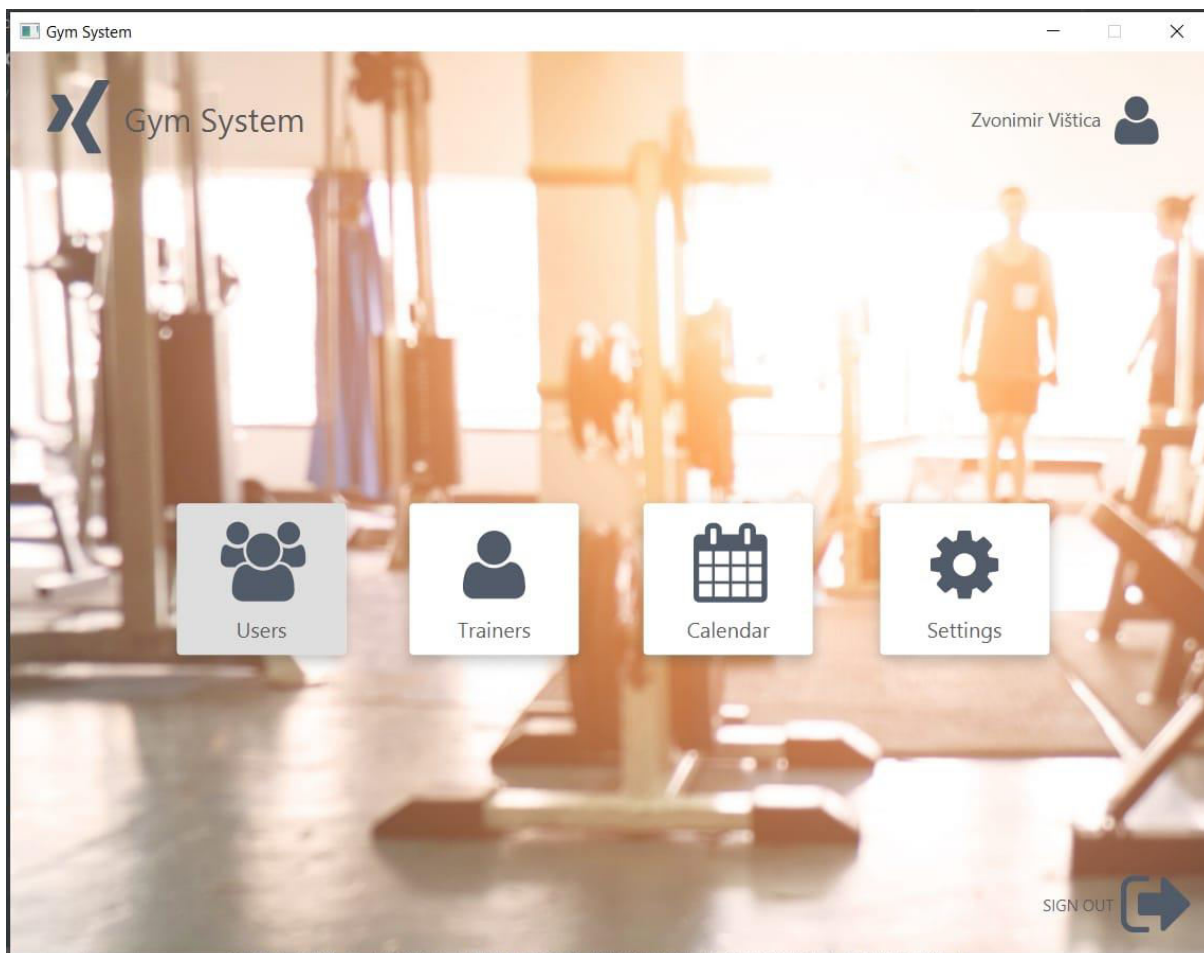
### datum

Naziv atributa	Integritet (PK/FK, NULL / NOT NULL)	Kratki opis atributa
ID	PK	Identifikacijski broj
datumuplate	NOT NULL	Pamti dan uplate
period	NOT NULL	Koliko je uplaćeno
IDUser	NOT NULL	Povezivanje s glavnom bazom

## 1.3. Aplikacija u Javi

Login





## 2. Model programske podrške

Drugi dio zadatka se odnosi na modeliranje programske podrške, detaljno ćete raspisati sve klase i na kraju priložiti dijagram klase koji opisuje izgled programske podrške:

- Napraviti class dijagram
- Opisati metode koje se nalaze unutar svake klase

U navedenim klasama potrebno je opisati :

- 1) Paketi – kratko opisati paket vaše programske podrške
- 2) Klase – kratko opisati klase koje koristite (ukratko za što će se klasa koristiti
  - a. Atribute – kratko pisati attribute koje se nalaze unutar klase;
  - b. Metode – kratko opisati metode koje se koriste za što služe i opišite njihov povratni tip;

### 2.1. Paketi

Opis paketa

Paket sadrži AlertMessage kojim postavljamo poruku u prozor. Iza toga mapu Controller gdje se nalaze klase za stvaranje objekata i njihovo upravljanje. Slijedi CSS file, uređivanje izgleda i DBConnection, povezivanje s bazom. Nakon toga imamo FXML\_files koji sadrži sve view-ove odnosno prikaze za JavaFX. Na kraju u paketu se nalazi Images u kojem se nalaze slike i Main klasa kojom se pokreće program.

### 2.2. Klase

Dijagram klasa

...

#### 2.2.1. AllTrainersControllers

Opis klase

Klasa koja služi prikazivanju liste korisnika koji su treneri

##### 2.2.1.1. Atributi

Table – tablica za prikaz podataka

id – stupac za prikaz ID-a trenera

ime - stupac za prikaz imena trenera

prezime - stupac za prikaz prezimena trenera

vrsta - stupac za prikaz vrste trenera

##### 2.2.1.2. Metode

Void Back – metoda kojom se vraćamo na prethodni view, odnosno Home page

#### 2.2.2 HomePageController

Opis klase

Klasa koja predstavlja funkcionalnost Home Pagea, odnosno glavne stranice koju vidi administrator

##### 2.2.2.1. Atributi klase 2



userDataLbl – predstavlja prikaz Usernamea

### **2.2.2.2. Metode klase 2**

---

Void Logout – odjavljivanje korisnika te povratak na Login/Sign up view

Void moveToUsers - prebacivanje na view gdje se nalazi tablica sa svim userima

Void moveToTreners - prebacivanje na view gdje se nalazi tablica sa svim trenerima

Void moveToUplate - prebacivanje na view gdje se nalazi tablica s datumima uplata

### **2.2.3. Korisnik**

---

Opis klase

Klasa kojom stvaramo objekt korisnika, sadrži konstruktor korisnik

#### **2.2.3.1. Atributi**

---

Sifra – sifra korisnika

ime - stupac za prikaz imena

prezime - stupac za prikaz prezimena

visina – stupac za prikaz visine

tezina - stupac za prikaz težine

sifraUser – ID usera

sifraTrener – ID trenera

#### **2.2.3.2. Metode**

---

getSifra() – vraća šifru

setSifra() – postavlja šifru

getIme() – vraća ime

setIme() – postavlja ime

getPrezime() – vraća prezime

setPrezime() – postavlja prezime

getVisina() – vraća visinu

setVisina() – postavlja visinu

getTezina() – vraća težinu

setTezina() – postavlja težinu

getSifraUser() – vraća ID usera

getSifraTrener() – vraća ID trenera

### **2.2.4 KorisnikController**

---

Opis klase

Klasa koja predstavlja funkcionalnost Home Pagea, odnosno glavne stranice koju vidi korisnik

#### **2.2.4.1. Atributi klase 2**

---

userDataLbl – predstavlja prikaz Usernamea

#### **2.2.4.2. Metode klase 2**

---

Void Logout – odjavljivanje korisnika te povratak na Login/Sign up view

Void moveToSettings - prebacivanje na view gdje se nalazi podaci o korisniku

Void moveToTreners - prebacivanje na view gdje se nalazi tablica sa svim trenerima

.

.

.