

**SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
VARAŽDIN**

Nikola Bekavac

**APLIKACIJA ZA UPRAVLJANJE
KORISNIČKIM RAČUNIMA**

PROJEKT

Varaždin, 2021.

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
V A R A Ž D I N

Nikola Bekavac

Matični broj:

Studij: Informacijski sustavi

APLIKACIJA ZA UPRAVLJANJE KORISNIČKIM RAČUNIMA

PROJEKT

:

Doc. dr. sc. Bogdan Okreša Đurić

Varaždin, siječanj 2021.

Izjava o izvornosti

Izjavljujem da je moj projekt izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

prihvatanjem odredbi u sustavu FOI-radovi

Sažetak

Projekt na temu aplikacija za upravljanje korisničkim računima. U ovom projektu ćemo reći nešto više o poopćenim i objektno-relacijskim bazama podataka, te na koji način se one koriste. Način korištenja ćemo prikazati na jednostavnoj aplikaciji za posudbu filmova. Vidit ćemo kako se daju uloge korisnicima ovisno o njihovim pravima i mogućnostima pristupa aplikaciji.

Ključne riječi: baze podataka, poopćena, objektno-relacijska, korisnički račun, uloge

Sadržaj

1. Uvod	1
2. Teorijski uvod	2
2.1. Poopćene baze podataka	2
2.2. Objektno-relacijske baze podataka	2
2.3. PHP	2
3. Opis aplikacijske domene	3
3.1. Neregistrirani korisnik	3
3.2. Registrirani korisnik	3
3.3. Zaposlenik	3
3.4. Administrator	3
4. Model baze podataka	4
5. Implementacija	5
5.1. Složeni tipovi podataka	5
5.1.1. Status	5
5.1.2. Role info	5
5.1.3. Address info	5
5.1.4. Gender info	6
5.2. Tablice	6
5.2.1. Tablica author	6
5.2.2. Tablica users	6
5.2.3. Tablica book	7
5.2.4. Tablica role	7
5.2.5. Tablica user-book	7
5.3. Kreiranje veza između tablica	8

1. Uvod

Projekt koji sam odabrao je na temu aplikacija za upravljanje korisničkim računima. Projekt samo odabrao iz razloga što mi se činio veoma zanimljiv jer sam znao da ću morati voditi računa o različitim vrstama korisnika, te koji podaci se za tu vrstu korisnika mogu prikazivati. Također, koncept poopćenih i objektno-relacijskih baza podataka mi se također na prvu svidio i htio sam naučit nešto više o njemu. Kao sustav za upravljanje bazom podataka sam koristio PostgreSQL, a sama aplikacija je izrađena u PHP-u.

2. Teorijski uvod

U ovom dijelu rada ću nešto više reći o teorijskim konceptima koji će se koristiti kao što su poopćene baze podataka, objektno-relacijske baze podataka i PHP.

2.1. Poopćene baze podataka

Poopćene-relacijske baze podataka omogućuju korištenje kompleksnijih objekata: (Schatten, 2008)

- BLOB
- Polja(1D, 2D)
- Pobrojene vrijednost(ENUM)
- Složeni tipovi

BLOB-ovi se koriste za pohranu raznih binarnih podataka poput različitih dokumenata, slika, videa... (Schatten, 2008).

Polja(1D, 2D) su polja koja mogu biti jednodimenzionalna i dvodimenzionalna. Tip podataka kod polja može biti bilo koji jednostavni tip podatka, kao naprimjer string, int, date...

Pobrojene vrijednosti(ENUM) su unaprijed definirane vrijednosti koji se ne mogu promijeniti.

Složeni tipovi su tipovi podataka koji se sastoje od više jednostavnih tipova. Složene tipove definiramo ovisno o potrebi.

2.2. Objektno-relacijske baze podataka

Objektno-relacijske baze podataka, kao što im i samo ime kaže, nastaju spajanjem objektnih i relacijskih baza podataka. Razlog zbog kojeg nastaju je u tome što relacijske baze podataka nisu mogle zadovoljiti potrebe za pohranom podataka. Proširenjem relacijskog modela uvode se koncepti poput: apstraktnih tipova podataka, enkapsulacije, nasljeđivanja tipova i tablica, ugniježđenih relacija i sl. (ZPR-FER, 2015).

2.3. PHP

Programski jezik koji je korišten za izradu aplikacije je PHP. PHP(Hypertext Preprocessor) je programski jezik namijenjen prvenstveno programiranju web stranica. Ističe se širokom podrškom za razne baze podataka i internet protokola što je upravo i glavni razlog za odabir ovog jezika.

3. Opis aplikacijske domene

Aplikacija služi za posudbu knjiga. Kako bi uopće započeli s izradom aplikacije, bilo je potrebno instalirati Apache server , PostgreSQL za bazu podataka i PHP za izradu aplikacije, a oni su dostupni u paketu koji se zove "Bitnami WAPP Stack". Ideja je da postoje četiri vrste korisnika, a to su : neregistrirani korisnik, registrirani korisnik, zaposlenik i administrator.

3.1. Neregistrirani korisnik

Neregistriranom korisniku prilikom pristupa aplikaciji se otvara početna stranica sa mogućnostima registracija i pregleda svih knjiga. U slučaju odabira registracije, neregistrirani korisnik se može registrirati i time postaje registrirani korisnik.

3.2. Registrirani korisnik

Registriranom korisniku se prilikom pristupa aplikaciji otvara ista početna stranica kao i neregistriranom korisniku. Registrirani korisnik ima također mogućnost pregleda svih knjiga i mogućnost prijave u sustav. Nakon što se korisnik prijave u sustav, nudi mu se mogućnost pregleda svih knjiga, pregleda svojih rezervacija i dodavanje novih rezervacija.

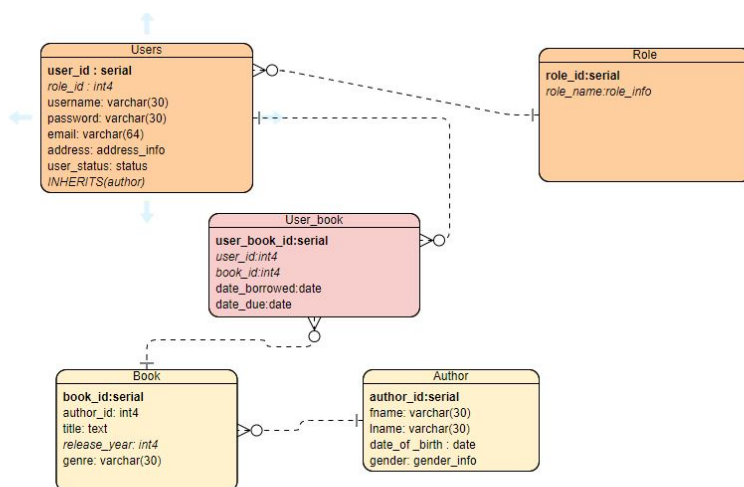
3.3. Zaposlenik

Zaposleniku se prilikom pristupa aplikaciji otvara ista početna stranica kao i svim korisnicima prije i ima sve mogućnosti na njoj kao i korisnici prije. Nakon prijave u aplikaciju, otvaraju mu se mogućnosti pregleda svih knjiga, pregleda svih rezervacija i pregleda svih korisnika.

3.4. Administrator

Administratoru se također prilikom pristupa aplikaciji otvara ista početna stranica. Nakon uspješne prijave u aplikaciju, otvaraju mu se mogućnosti pregleda svih knjiga, pregleda svih rezervacija, pregled svih korisnika, dodavanje nove knjige i pregled svih zaposlenika.

4. Model baze podataka



Slika 1: Model baze podataka

Kao što se može vidjeti na slici, model se sastoji od 5 tablica. Model baze podataka je rađen u alatu Visual Paradigm.

Tablica author se sastoji od atributa: author-id, fname, lname, date-of-birth, gender.

Tablica book se sastoji od atributa: book-id, author-id, title, release-year, genre.

Tablica role se sastoji od atributa: role-id, role-name.

Tablica users sa sastoji od atributa: user-id, role-id, username, password, email, address, user-status. Bitno je za tablicu users reći da nasljeđuje attribute od tablice authors.

Tablica user-book se sastoji od atributa: user-book-id, user-id, book-id, date-borrowed, date-due.

5. Implementacija

U ovom dijelu ćemo pobliže opisati kako smo kreirali složene tipove podataka, tablice, nasljeđivanje i veze između tablica.

5.1. Složeni tipovi podataka

Kako smo i prije naveli, jedan od koncepata koji nam omogućuju poopćene baze podataka su složeni tipovi. U ovom poglavlju ćemo vidjeti implementaciju tih složenih tipova.

5.1.1. Status

Status je složeni tip koji služi za provjeriti je li korisnik trenutno aktivan. On je implementiran kao ENUM, odnosno može poprimiti vrijednosti 'na vezi', 'nije na vezi' i 'blokirano'.

Kod za kreiranje statusa:

```
CREATE TYPE status AS ENUM('blokirano', 'nije na vezi', 'na vezi');
```

5.1.2. Role info

Role info je složeni tip podatka koji je kao i status ENUM. Može poprimiti vrijednosti 'U' što je skraćeno od user, 'A' što je skraćeno od administrator i 'E' što je skraćeno od employee.

Kod za kreiranje Role info:

```
CREATE TYPE role-info AS ENUM('U', 'A', 'E');
```

5.1.3. Address info

Address info je složeni tip podatka koji služi za pohranu adrese korisnika. On se sastoji od više jednostavnih tipova podataka: city name VARCHAR, street name VARCHAR, street number INT4, postal-number INT4

Kod za kreiranje Address info:

```
CREATE TYPE address-info AS(  
  city-name VARCHAR(50),  
  street-name VARCHAR(50),  
  street-number INT4,  
  postal-number INT4 );
```

5.1.4. Gender info

Gender info je složeni tip podatka koji je kao i status ENUM. Ovaj složeni podatak se odnosi na spol korisnika. Može poprimiti vrijednosti 'M' što znači da je riječ o muškarcu i F što znači da je riječ o ženi.

Kod za kreiranja Role info:

```
CREATE TYPE gender-info AS ENUM( 'M' , 'F');
```

5.2. Tablice

U ovom dijelu ćemo pobliže opisati i prikazati implementaciju za tablice. Baza podataka se sastoji od 5 tablica.

5.2.1. Tablica author

Tablica autor je tablica u kojoj se nalaze podaci za autora knjige. Tablica sa sastoji od 5 atributa. Kod za kreiranja tablice author:

```
CREATE TABLE "author" (  
  "author-id" SERIAL NOT NULL,  
  "fname" VARCHAR(30) NOT NULL,  
  "lname" VARCHAR(30) NOT NULL,  
  "date-of-birth" DATE NOT NULL,  
  "gender" gender-info NOT NULL,  
  PRIMARY KEY("author-id"));
```

5.2.2. Tablica users

Tablica autor je tablica u kojoj se nalaze podaci za sve korisnike aplikacije. Tablica sa sastoji od ukupno 12 atributa. Od tih 12 atributa, 5 atributa se nasljeđuje iz tablice author, čime je implementirano nasljeđivanje. Kod za kreiranja tablice users:

```
CREATE TABLE "users"(  
  "user-id" SERIAL NOT NULL,  
  "role-id" INT4 NOT NULL,  
  "username" VARCHAR(30) NOT NULL,  
  "password" VARCHAR(30) NOT NULL,  
  "email" VARCHAR(64) NOT NULL,  
  "address" address-info NOT NULL,  
  "user-status" status NOT NULL,  
  PRIMARY KEY("user-id"))
```

INHERITS(author);

5.2.3. Tablica book

Tablica book je tablica u kojoj se nalaze podaci za sve knjige dostupne u aplikaciji. Tablica sa sastoji od ukupno 5 atributa. Kod za kreiranja tablice book:

```
CREATE TABLE "book"(  
  "book-id" SERIAL NOT NULL,  
  "author-id" int4 NOT NULL,  
  "title" TEXT NOT NULL,  
  "release-year" INT4 NOT NULL,  
  "genre" VARCHAR(30) NOT NULL,  
  PRIMARY KEY("book-id"));
```

5.2.4. Tablica role

Tablica role je tablica u kojoj se nalaze podaci za uloge dostupne u aplikaciji. Tablica sa sastoji od ukupno 2 atributa. Kod za kreiranja tablice book:

```
CREATE TABLE "role"  
("role-id" SERIAL NOT NULL,  
"role-name" role-info NOT NULL,  
PRIMARY KEY("role-id"));
```

5.2.5. Tablica user-book

Tablica user-book je vezna tablica između korisnika i knjiga. Ona služi za vidjeti koji korisnik je posudio koju knjigu. Tablica sa sastoji od 5 atributa. Kod za kreiranja tablice book:

```
CREATE TABLE "user-book"  
("user-book-id" SERIAL NOT NULL,  
"user-id" INT4 NOT NULL,  
"book-id" INT4 NOT NULL,  
"date-borrowed" DATE,  
"date-due" DATE,  
PRIMARY KEY("user-book-id"));
```

5.3. Kreiranje veza između tablica