-Model podataka i perzistencija-

Stefan Sentić, 15359 Marko Vučković, 15473

Sadržaj

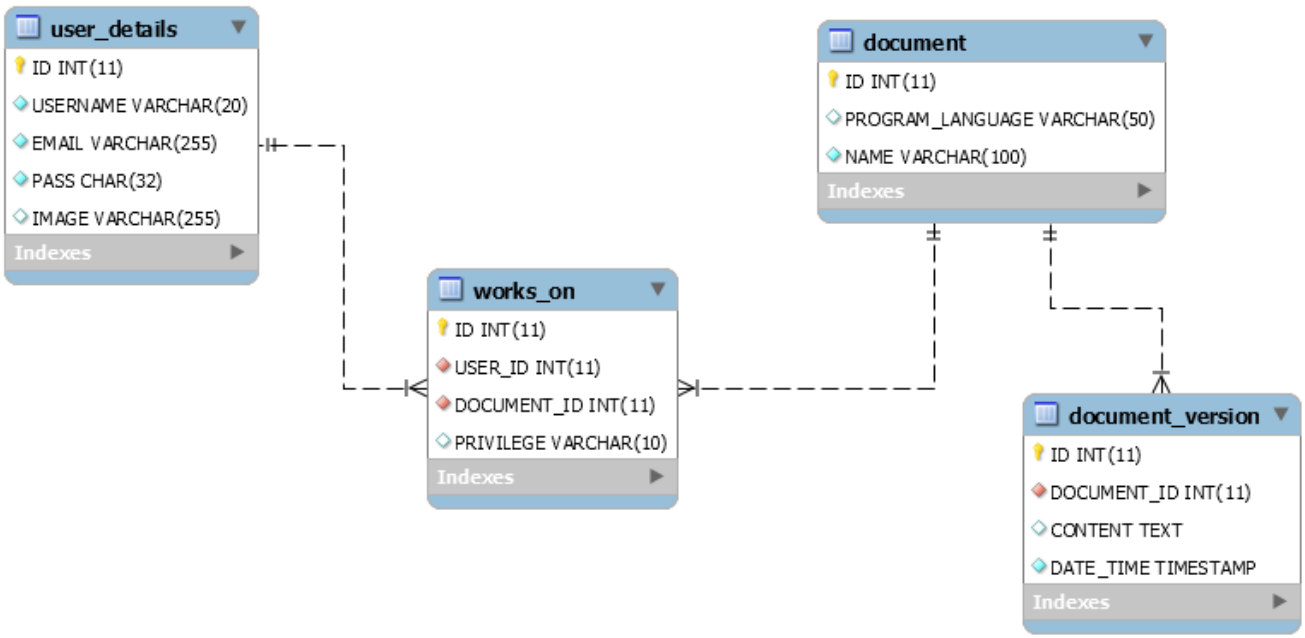
Opis dokumenta

Svrha pisanja ovog dokumenta jeste opis modela podataka i modela perzistencije koji se koristi u aplikaciji *Comet*. Za model perzistencije korišćena je *MySQL* baza podataka. Korišćen je i objektno-relacioni maper *Hibernate* da bi se programeru omogućilo da lakše komunicira s bazom.

Šema baze podataka

Aplikacija Comet sadrži sistem za autentifikaciju korisnika. Nakon logovanja ili registrovanja na sistem, korisnik može pristupiti nekom od postojećih dokumenata ili kreirati novi dokument. U bazi podataka se pamte sve verzije dokumenata na kojima se radi.

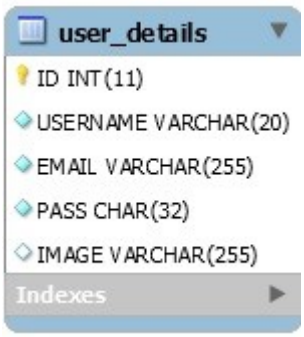
Na slici 1.1 dat je prikaz šeme baze podataka.



Slika 1.1

Tabela User

Na slici 1.2 prikazana je tabela u kojoj se pamte podaci o korisniku.



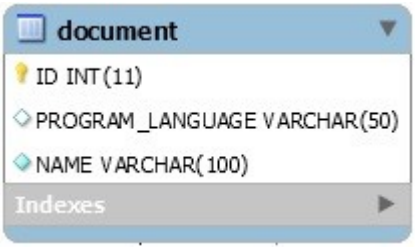
Slika 1.2

Za korisnika se u bazi pamti sledeće:

- *Id,* tipa int, *autoincrement*
- *Username*, kao podatak tipa *varchar,* koji mora biti unikatan.
- *Password,* koji se pamti u enkriptovanom obliku kao *varchar*
- *Email,* takođe tipa *varchar* i mora biti unikatan.
- *Image,* tipa *varchar,* koji predstavlja URL do slike korisnika.

Tabela Document

Na slici 1.3 prikazana je tabela u kojoj se pamte podaci o dokumentima.

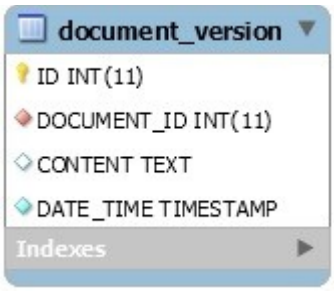


Slika 1.3

- *Id,* tipa integer, *autoincrement*
- *Program_Language*, *varchar* tipa, koji pamti programski jezik u kome je dokument pisan.
- *Name* tipa *varchar* gde se pamti ime dokumenta.

Tabela DocumentVersion

Na slici 1.4 prikazana je tabela u kojoj se pamte podaci o verzijama dokumenata.



Slika 1.4

- *Id ,* celobrojni podatak, autoincrement
- *Document_id,* tipa integer koji predstavlja strani ključ na tabelu *Document.* Pokazuje na dokument čija čija je verzija.
- *Content,* tipa *text* a ne klasičan *varchar,* jer se očekuje pamćenje veće količinie podataka.
- *Date_time*, tipa *timestamp,* koji predstavlja vreme čuvanja dokumenta.

Tabela WorksOn

Na slici 1.5 prikazana je tabela koja povezuje korisnike sa dokumentima na kojima rade, pri čemu se pamti privilegija korisnika nad dokumentom.

- *Id*, tipa *integer autoincrement*
- *User_id*, strani kluč koji referencira tabelu *User*
- *Document_id*, referencira tabelu *Document*
- *Privilege*, tipa *varchar* opisuje privilegiju korisnika nad dokumentom.

Slika 1.5

Data Mapper pattern

Pattern koji je primenjen u Data layer-u je Data Mapper. U paketu aps.dto nalaze se POJO klase u kojima je anotacijama naznačeno preslikavanje svakog atributa klase u odgovarajuću kolonu tabele, a takođe anotacijama definisano i spajanje tabela. U paketu aps.dao nalaze se klase sa funkcijama koje enkapsuliraju rad sa bazom podataka.

Na slici 1.6 dat je prikaz *User* POJO klase sa anotacijama (kao primer).

```
@Entity
@Table (name="USER_DETAILS")
public class User {

    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name="ID")
    private int id;

    @Column(name="USERNAME")
    private String username;

    @Column(name="PASS")
    private String password;

    @Column(name="EMAIL")
    private String email;

    @Column(name="IMAGE")
    private String image;

    @OneToMany(mappedBy="user", cascade = CascadeType.ALL)
    private Collection<WorksOn> documents;


    public User() {
        this.documents = new ArrayList<WorksOn>();
    }
    public User(String username,String password,String email,String image) {
        this.documents = new ArrayList<WorksOn>();
        this.username = username;
        this.password = password;
        this.email = email;
        this.image = image;
    }

    public int getId() {
        return id;
    }
```

Slika 1.6

Na slici 1.7 prikazana je klasa UserDao koja sadrži sve funkcije kojima se pristupa podacima u bazi podataka (kao primer).

```
public class UserDao {

    public Info addUser(User u) {

    public void deleteUser(int id) {

    public void updateUser(User user) {

    public User getUserById(int id) {

    public ArrayList<Document> getAllDocuments (int userId){

    public ArrayList<WorksOn> getAllWorksOn(int userId){

    public User login (String username, String password) {
}
```

Slika 1.7