

Baze de Date
Proiect Inchiriere Apartamente

Canevschii Daniel

Contents

1	Scopul si obiectivele proiectului	2
2	Lista cu intrebari ale caror raspunsuri implica interogarea tabelor	2
3	Lista cu intrebari ale carori raspunsuri implica adaugarea, actualizarea si stergerea datelor	2
4	Analiza cererilor de informatii	3
4.1	Care sunt apartamentele disponibile spre inchiriere?	3
4.2	Care sunt apartamentele (disponibile) din locatia X?	3
4.3	Ce apartamente sunt disponibile intre data X - Y?	4
4.4	Lista de apartamente cu pret intre X si Y (mai mare ca X, mai mic ca X)?	4
4.5	Lista de proprietati detinute de un anumit proprietar?	4
5	Modelarea conceptuala(diagrama Entitate - Relatie)	4
5.1	Entitati	4
5.2	Legaturi dintre entitati	4
5.3	Determinarea atributelor	4
6	Alegerea SGBD-ului	4
7	Proiectarea bazei de date (schema logica / schema fizica	5
8	Implementarea BD - scripturile SQL (creare, inserare, actualizare, stergere)	5

1 Scopul si obiectivele proiectului

Scopul proiectului este de a dezvolta o aplicatie care sa faciliteze stocare, extragere si gestionarea informatiei necesare in procesul de detinere sau inchiriere a apartamentelor.

- Stocarea informatiei despre utilizatorii aplicatiei, cu rolurile corespunzatoare.
- Stocarea informatiei despre proprietatile detinute de proprietari, precum adresa, numarul de camere, imagini, etc.
- Inregistrarea contractelor de inchiriere cu datele necesare(stare contract, parti contractate, valabilitate, suma)
- Pastrarea tuturor tranzactiilor efectuate in cadrul aplicatiei.

2 Lista cu intrebari ale caror raspunsuri implica interogarea tabelor

1. Care sunt apartamentele disponibile spre inchiriere?
2. Care sunt apartamentele (disponibile) din locatia X?
3. Ce apartamente sunt disponibile intre data X - Y?
4. Lista de apartamente cu pret intre X si Y (mai mare ca X, mai mic ca X)?
5. Lista de proprietati detinute de un anumit proprietar?

3 Lista cu intrebari ale carori raspunsuri implica adaugarea, actualizarea si stergerea datelor

1. Cum se adauga un nou user in baza de date?
2. Cum se adauga un nou apartament in baza de date?
3. Cum se depune o cerere de inchiriere a unui apartament?
4. Cum se aproba un contract de inchiriere?
5. Cum se actualizeaza informatie despre un apartament?
6. Cum se modifica informatia unui utilizator?

7. Cum se sterge un utilizator din baza de date?
8. Cum se efectueaza o plata(tranzactie)?
9. Cum se modifica parola unui utilizator?
10. Cum se modifica rolul utilizatorului(proprietar/utilizator)?

4 Analiza cererilor de informatii

4.1 Care sunt apartamentele disponibile spre inchiriere?

1. Se obtine lista apartamentelor ale caror *booking_id* in **Booking** e *Ended*.

```
1 SELECT apartment_id FROM Apartments
2 WHERE apartment_id NOT IN (
3     SELECT apartment_id FROM Booking
4     WHERE booking_status in ("Pending", "Confirmed")
5 );
```

4.2 Care sunt apartamentele (disponibile) din locatia X?

1. Se foloseste lista apartamentelor obtinute la punctul anterior, pentru care se verifica daca locatia *X* e aceeaasi cu locatia din **Locations** pentru acelasi *location_id*.

```
1 SELECT apartment_id FROM Apartments
2 WHERE apartment_id NOT IN (
3     SELECT apartment_id FROM Booking
4     WHERE booking_status in ("Pending", "Confirmed")
5 )
6 AND location_id in (
7     SELECT location_id FROM Locations
8     WHERE coutry = coutry_req
9     AND city = city_req
10 );
```

4.3 Ce apartamente sunt disponibile intre data X - Y?

4.4 Lista de apartamente cu pret intre X si Y (mai mare ca X, mai mic ca X)?

4.5 Lista de proprietati detinute de un anumit proprietar?

5 Modelarea conceptuala(diagrama Entitate - Relatie)

5.1 Entitati

- Users
- Properties
- Locations
- Bookings
- Payments

5.2 Legaturi dintre entitati

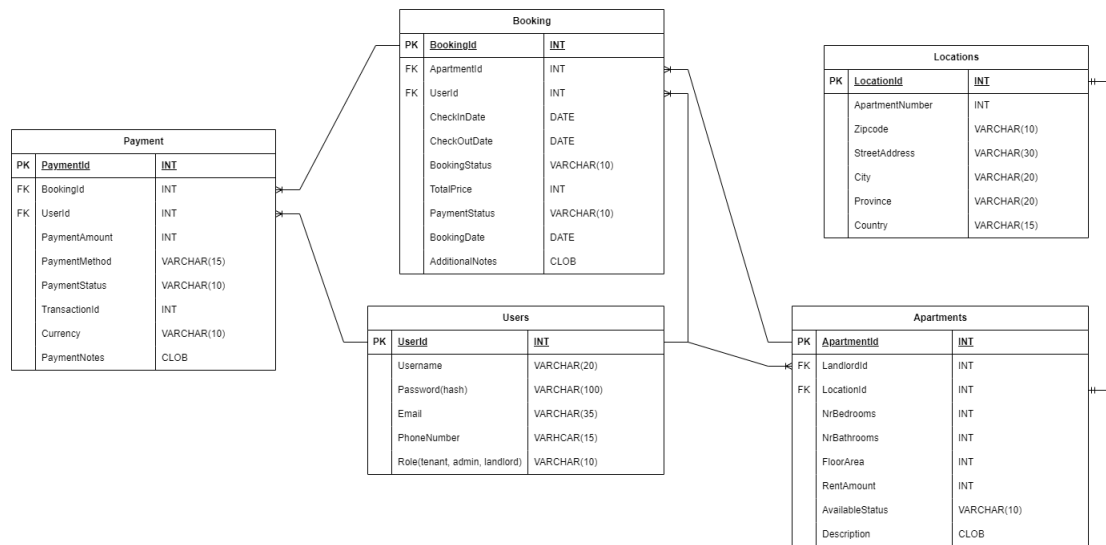
- Users.user_id [una - multe] Apartments.landlord_id
- Users.user_id [una - multe] Booking.user_id
- Users.user_id [una - multe] Payment.user_id
- Apartments.apartment_id [una - multe] Booking.apartment_id
- Booking.booking_id [una - multe] Payment.booking_id
- Locations.location_id [una - una] Apartments.location_id

5.3 Determinarea atributelor

6 Alegerea SGBD-ului

Oracle Database

7 Proiectarea bazei de date (schema logica / schema fizica)



8 Implementarea BD - scripturile SQL (creare, inserare, actualizare, stergere)

```

1 CREATE TABLE Users (
2     user_id INT PRIMARY KEY,
3     username VARCHAR(20) UNIQUE,
4     password VARCHAR(100),
5     email VARCHAR(35) UNIQUE,
6     phone_number VARCHAR(15),
7     role VARCHAR(10)
8 );
9
10 CREATE TABLE Locations (
11     location_id INT PRIMARY KEY,
12     apartment_number INT,
13     zipcode VARCHAR(10),
14     street_address VARCHAR(30),
15     city VARCHAR(20),
16     province VARCHAR(20),
17     country VARCHAR(15)
18 );
19
20 CREATE TABLE Apartments (
21     apartment_id INT PRIMARY KEY,
22     landlord_id INT,
23     location_id INT,

```

```

24     nr_bedrooms INT,
25     nr_bathrooms INT,
26     floor_area INT,
27     rent_amount INT,
28     available_status VARCHAR(10),
29     description CLOB
30     FOREIGN KEY(landlord_id) REFERENCES Users(user_id),
31     FOREIGN KEY(location_id) REFERENCES Locations(location_id)
32 );
33
34 CREATE TABLE Booking (
35     booking_id INT PRIMARY KEY,
36     apartment_id INT,
37     user_id INT,
38     check_in_date DATE,
39     check_out_date DATE,
40     booking_status VARCHAR(10),
41     total_price INT,
42     payment_status VARCHAR(10),
43     booking_date DATE,
44     additional_notes CLOB,
45     FOREIGN KEY(apartment_id) REFERENCES Apartments(apartment_id),
46     FOREIGN KEY(user_id) REFERENCES Users(user_id)
47 );
48
49 CREATE TABLE Payment (
50     payment_id INT PRIMARY_KEY,
51     booking_id INT,
52     user_id INT,
53     payment_amount INT,
54     payment_method VARCHAR(15),
55     payment_status VARCHAR(10),
56     transaction_id INT UNIQUE,
57     currency VARCHAR(10),
58     payment_notes CLOB,
59     FOREIGN KEY(booking_id) REFERENCES Booking(booking_id),
60     FOREIGN KEY(user_id) REFERENCES Users(user_id)
61 );

```