**CSI2132 Database I**

**eHotel-Course Project:**

**e-Hotels Database Application**

**Deliverable 2 Report**

Roy, Rui 300176548

Yucheng, Chen 300194614

University of Ottawa

Faculty of Engineering

March 31, 2024

**a. Technologies Used**

**Database Management System (DBMS)**

For our e-Hotels project, we utilized **PostgreSQL** as our primary Database Management System. PostgreSQL was chosen for free to use and its robust feature set, including support for complex queries, foreign keys, triggers, and its strong compliance with SQL standards.

**Programming Languages**

Our project primarily involved direct writing and execution of SQL statements for database operations and management, without employing PHP or other server-side programming languages for backend logic.

**b. Installation Guide**

**Prerequisites**

* PostgreSQL (Version X.X)
* Appropriate SQL client tool (e.g., pgAdmin or the command-line tool psql)
* Web Browser

**Installation Steps**

1. **Database Setup:**

Install PostgreSQL and create a new database named eHotel.

Execute the provided DDL scripts in the eHotel database to create the necessary tables and relationships.

1. **Application Deployment:**

Given our project's reliance on SQL operations, there isn't a traditional application deployment step. However, ensure that you can access and manipulate the database through your SQL client tool.

1. **Running the Application:**

Open your SQL client tool and connect to your eHotel database.

You are now ready to perform SQL queries and other database operations.

**c. Database Definition Language (DDL) Scripts**

Below is a list of DDL scripts used to create the database schema for the e-Hotels application. These scripts define the structure of the database tables, constraints, and relationships.

-- Hotel Chain

CREATE TABLE HotelChain (

brand\_id INT PRIMARY KEY,

brand\_name VARCHAR(255),

office\_address VARCHAR(255),

contact\_email VARCHAR(255),

phone\_no VARCHAR(20)

hotel\_counts INT

);

-- Hotel

CREATE TABLE HotelDepartments (

dp\_id INT PRIMARY KEY,

brand\_id INT,

star\_level INT,

dp\_phone VARCHAR(20),

dp\_address VARCHAR(255),

dp\_email VARCHAR(255),

manager\_id INT,

room\_counts INT,

FOREIGN KEY (brand\_id) REFERENCES HotelChain(brand\_id),

FOREIGN KEY (manager\_id) REFERENCES Manager(employee\_id)

);

-- Employees

CREATE TABLE Employees (

employee\_id INT PRIMARY KEY,

ssn\_sin VARCHAR(20),

full\_name VARCHAR(255),

address VARCHAR(255),

FOREIGN KEY (role\_id) REFERENCES Role(role\_id)

);

-- Role

CREATE TABLE Role (

role\_id INT PRIMARY KEY,

role\_name VARCHAR(255)

);-- EmployeeRole (Junction Table for many-to-many relationship

between Employees and Roles)

CREATE TABLE EmployeeRole (

employee\_id INT,

role\_id INT,

PRIMARY KEY (employee\_id),

FOREIGN KEY (employee\_id) REFERENCES Employees(employee\_id),

FOREIGN KEY (role\_id) REFERENCES Role(role\_id)

);

-- Customer

CREATE TABLE Customer (

customer\_id INT PRIMARY KEY,

full\_name VARCHAR(255),

address VARCHAR(255),

id\_proof\_type VARCHAR(50),

id\_proof\_no VARCHAR(50),

regis\_date DATE

);

-- Rooms

CREATE TABLE Rooms (

room\_no INT PRIMARY KEY,

dp\_id INT,

price DECIMAL(10, 2),

capacity INT,

has\_sea\_view BOOLEAN,

has\_mountain\_view BOOLEAN,

can\_extend\_bed BOOLEAN,

problems\_description TEXT,

FOREIGN KEY (dp\_id) REFERENCES Hotel(dp\_id)

);

-- Rooms

CREATE TABLE Amenities (

rm\_no INT PRIMARY KEY,

has\_tv BOOLEAN,

has\_air

\_conditioning BOOLEAN,

has\_fridge BOOLEAN,

FOREIGN KEY (rm\_no) REFERENCES Rooms(rm

\_

no)

);

-- Bookings

CREATE TABLE Booking (

booking\_id INT PRIMARY KEY,customer\_id INT,

room\_id INT,

employee\_id INT,

number\_of\_guests INT,

booking\_start\_date DATE,

booking\_end\_date DATE,

FOREIGN KEY (customer\_id) REFERENCES Customer(customer\_id),

FOREIGN KEY (room\_id) REFERENCES Rooms(rm\_id),

FOREIGN KEY (employee\_id) REFERENCES Employees(employee\_id)

);

-- Rentings

CREATE TABLE Renting (

renting\_id INT PRIMARY KEY,

customer\_id INT,

room\_id INT,

employee\_id INT,

check\_in\_date DATE,

check\_out\_date DATE,

number\_of\_guests INT,

payment\_status VARCHAR(100),

FOREIGN KEY (customer\_id) REFERENCES Customer(customer\_id),

FOREIGN KEY (room\_id) REFERENCES Rooms(room\_id),

FOREIGN KEY (employee\_id) REFERENCES Employees(employee\_id)

);

-- Archive (Could be for either Bookings or Rentings)

CREATE TABLE Archive (

archive\_id INT PRIMARY KEY,

original\_action\_id INT,

action\_type VARCHAR(50), -- 'Booking' or 'Renting'

customer\_id INT,

room\_id INT,

employee\_id INT,

start\_date DATE,

end\_date DATE,

number\_of\_guests INT,

problem\_reported VARCHAR(255),

special\_notes VARCHAR(255),

archive\_date DATE

);

Please refer to the attached eHotel.sql file for the complete list of DDLs.

A table with numbers and time

Description automatically generated