Rooms. Java entity

package com.example.pgweb.entity;

import jakarta.persistence.\*;

@Entity

@Table(name = "rooms")

public class Room {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String name;

private String type;

private int availability;

private double cost;

// Getters and setters

public Long getId() {

return id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getType() {

return type;

}

public void setType(String type) {

this.type = type;

}

public int getAvailability() {

return availability;

}

public void setAvailability(int availability) {

this.availability = availability;

}

public double getCost() {

return cost;

}

public void setCost(double cost) {

this.cost = cost;

}

}

Bookings. Java entity

package com.example.pgweb.entity;

import jakarta.persistence.\*;

import java.time.LocalDate;

@Entity

@Table(name = "bookings")

public class Booking {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

@ManyToOne

@JoinColumn(name = "user\_id")

private User user;

@ManyToOne

@JoinColumn(name = "room\_id")

private Room room;

private LocalDate checkInDate;

private LocalDate checkOutDate;

// Getters and setters

public Long getId() {

return id;

}

public void setId(Long id) {

this.id = id;

}

public User getUser() {

return user;

}

public void setUser(User user) {

this.user = user;

}

public Room getRoom() {

return room;

}

public void setRoom(Room room) {

this.room = room;

}

public LocalDate getCheckInDate() {

return checkInDate;

}

public void setCheckInDate(LocalDate checkInDate) {

this.checkInDate = checkInDate;

}

public LocalDate getCheckOutDate() {

return checkOutDate;

}

public void setCheckOutDate(LocalDate checkOutDate) {

this.checkOutDate = checkOutDate;

}

}

Room repository

package com.example.pgweb.repository;

import com.example.pgweb.entity.Room;

import org.springframework.data.jpa.repository.JpaRepository;

public interface RoomRepository extends JpaRepository<Room, Long> {

}

Booking repository

package com.example.pgweb.repository;

import com.example.pgweb.entity.Booking;

import org.springframework.data.jpa.repository.JpaRepository;

public interface BookingRepository extends JpaRepository<Booking, Long> {

}

User. Service

package com.example.pgweb.service;

import com.example.pgweb.entity.Booking;

import com.example.pgweb.entity.Room;

import com.example.pgweb.entity.User;

import com.example.pgweb.repository.BookingRepository;

import com.example.pgweb.repository.RoomRepository;

import com.example.pgweb.repository.UserRepository;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import java.util.List;

@Service

public class UserService {

@Autowired

private UserRepository userRepository;

@Autowired

private RoomRepository roomRepository;

@Autowired

private BookingRepository bookingRepository;

public User saveUser(User user) {

return userRepository.save(user);

}

public List<User> getAllUsers() {

return userRepository.findAll();

}

public List<Room> getAllRooms() {

return roomRepository.findAll();

}

public Booking saveBooking(Booking booking) {

Room room = booking.getRoom();

room.setAvailability(room.getAvailability() - 1);

roomRepository.save(room);

return bookingRepository.save(booking);

}

public List<Booking> getAllBookings() {

return bookingRepository.findAll();

}

}

Controller

package com.example.pgweb.controller;

import com.example.pgweb.entity.Booking;

import com.example.pgweb.entity.Room;

import com.example.pgweb.entity.User;

import com.example.pgweb.service.UserService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.PostMapping;

@Controller

public class UserController {

@Autowired

private UserService userService;

@GetMapping("/")

public String showRegistrationForm(Model model) {

model.addAttribute("user", new User());

return "registration";

}

@PostMapping("/register")

public String registerUser(User user) {

userService.saveUser(user);

return "redirect:/rooms";

}

@GetMapping("/rooms")

public String showRooms(Model model) {

model.addAttribute("rooms", userService.getAllRooms());

model.addAttribute("booking", new Booking());

return "rooms";

}

@PostMapping("/book")

public String bookRoom(Booking booking) {

userService.saveBooking(booking);

return "redirect:/rooms";

}

@GetMapping("/admin")

public String showAdminPage(Model model) {

model.addAttribute("users", userService.getAllUsers());

model.addAttribute("bookings", userService.getAllBookings());

return "admin";

}

}

Admin html

<!DOCTYPE html>

<html xmlns:th="http://www.thymeleaf.org">

<head>

<meta charset="UTF-8">

<title>Admin Page</title>

<link rel="stylesheet" th:href="@{/css/style.css}">

</head>

<body>

<h1>Admin Page</h1>

<h2>Registered Users</h2>

<table>

<tr>

<th>ID</th>

<th>Name</th>

<th>Email</th>

<th>Phone Number</th>

<th>Aadhaar Number</th>

<th>City</th>

</tr>

<tr th:each="user : ${users}">

<td th:text="${user.id}"></td>

<td th:text="${user.name}"></td>

<td th:text="${user.email}"></td>

<td th:text="${user.phoneNumber}"></td>

<td th:text="${user.aadhaarNumber}"></td>

<td th:text="${user.city}"></td>

</tr>

</table>

<h2>Bookings</h2>

<table>

<tr>

<th>Booking ID</th>

<th>User</th>

<th>Room</th>

<th>Check-in Date</th>

<th>Check-out Date</th>

</tr>

<tr th:each="booking : ${bookings}">

<td th:text="${booking.id}"></td>

<td th:text="${booking.user.name}"></td>

<td th:text="${booking.room.name}"></td>

<td th:text="${booking.checkInDate}"></td>

<td th:text="${booking.checkOutDate}"></td>

</tr>

</table>

</body>

</html>

Rooms. Html

<!DOCTYPE html>

<html xmlns:th="http://www.thymeleaf.org">

<head>

<meta charset="UTF-8">

<title>Vacant Rooms and Booking</title>

<link rel="stylesheet" th:href="@{/css/style.css}">

</head>

<body>

<h1>Vacant Rooms and Pricing</h1>

<table>

<tr>

<th>Room Name</th>

<th>Room Type</th>

<th>Availability</th>

<th>Cost per Person</th>

<th>Action</th>

</tr>

<tr th:each="room : ${rooms}">

<td th:text="${room.name}"></td>

<td th:text="${room.type}"></td>

<td th:text="${room.availability}"></td>

<td th:text="${room.cost}"></td>

<td>

<form th:action="@{/book}" method="post">

<input type="hidden" name="roomId" th:value="${room.id}">

<input type="submit" value="Book" th:disabled="${room.availability == 0}">

</form>

</td>

</tr>

</table>

<h2>Booking Form</h2>

<form th:action="@{/book}" method="post" th:object="${booking}">

<label for="roomId">Select Room:</label>

<select id="roomId" name="roomId" required>

<option value="">Select a room</option>

<option th:each="room : ${rooms}" th:value="${room.id}" th:text="${room.name + ' - ' + room.type}" th:disabled="${room.availability == 0}"></option>

</select><br>

<label for="checkInDate">Check-in Date:</label>

<input type="date" id="checkInDate" name="checkInDate" required><br>

<label for="checkOutDate">Check-out Date:</label>

<input type="date" id="checkOutDate" name="checkOutDate" required><br>

<input type="submit" value="Book Room">

</form>

</body>

</html>