**Cabs Module**

CabBooking

package com.Springbootpace.SpringBoot.Entity;

import jakarta.persistence.\*;

*@Entity*

*@Table*(name = "cab\_bookings")

public class CabBooking {

*@Id*

*@GeneratedValue*(strategy = *GenerationType*.***IDENTITY***)

private Long id;

private Long userId;

private String name;

private String cabType;

private String pickup;

private String dropLocation;

private int travellers;

private double totalPrice;

private String travelDate;

// Getters and Setters

public Long getId() { return id; }

public void setId(Long id) { this.id = id; }

public Long getUserId() { return userId; }

public void setUserId(Long userId) { this.userId = userId; }

public String getName() { return name; }

public void setName(String name) { this.name = name; }

public String getCabType() { return cabType; }

public void setCabType(String cabType) { this.cabType = cabType; }

public String getPickup() { return pickup; }

public void setPickup(String pickup) { this.pickup = pickup; }

public String getDropLocation() { return dropLocation; }

public void setDropLocation(String dropLocation) { this.dropLocation = dropLocation; }

public int getTravellers() { return travellers; }

public void setTravellers(int travellers) { this.travellers = travellers; }

public double getTotalPrice() { return totalPrice; }

public void setTotalPrice(double totalPrice) { this.totalPrice = totalPrice; }

public String getTravelDate() {

return travelDate;

}

public void setTravelDate(String travelDate) {

this.travelDate = travelDate;

}

}

CabBookingRepository

package com.Springbootpace.SpringBoot.Repositary;

import com.Springbootpace.SpringBoot.Entity.CabBooking;

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

import org.springframework.stereotype.Repository;

import java.util.List;

@Repository

public interface CabBookingRepository extends JpaRepository<CabBooking, Long> {

List<CabBooking> findByUserId(Long userId);

}

CabBookingService

package com.Springbootpace.SpringBoot.Service;

import com.Springbootpace.SpringBoot.Entity.CabBooking;

import java.util.List;

public interface CabBookingService {

CabBooking saveBooking(CabBooking booking);

List<CabBooking> getAllBookings();

List<CabBooking> getBookingsByUserId(Long userId);

void deleteBooking(Long id);

}

CabBookingServiceImpl

package com.Springbootpace.SpringBoot.Service;

import com.Springbootpace.SpringBoot.Entity.CabBooking;

import com.Springbootpace.SpringBoot.Repositary.CabBookingRepository;

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

import org.springframework.http.HttpStatus;

import org.springframework.stereotype.Service;

import org.springframework.web.server.ResponseStatusException;

import java.util.List;

@Service

public class CabBookingServiceImpl implements CabBookingService {

@Autowired

private CabBookingRepository cabBookingRepository;

@Override

public CabBooking saveBooking(CabBooking booking) {

return cabBookingRepository.save(booking);

}

@Override

public List<CabBooking> getAllBookings() {

return cabBookingRepository.findAll();

}

@Override

public List<CabBooking> getBookingsByUserId(Long userId) {

return cabBookingRepository.findByUserId(userId);

}

@Override

public void deleteBooking(Long id) {

if (!cabBookingRepository.existsById(id)) {

throw new ResponseStatusException(HttpStatus.NOT\_FOUND, "Cab booking not found");

}

cabBookingRepository.deleteById(id);

}

}

CabBookingController

package com.Springbootpace.SpringBoot.Controller;

import com.Springbootpace.SpringBoot.Entity.CabBooking;

import com.Springbootpace.SpringBoot.Service.CabBookingService;

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

import org.springframework.http.ResponseEntity;

import org.springframework.web.bind.annotation.\*;

import java.util.List;

@RestController

@RequestMapping("/api/cabs")

@CrossOrigin(origins = "\*")

public class CabBookingController {

@Autowired

private CabBookingService cabBookingService;

@PostMapping("/book")

public CabBooking bookCab(@RequestBody CabBooking booking) {

return cabBookingService.saveBooking(booking);

}

@GetMapping("/user/{userId}")

public List<CabBooking> getBookingsByUser(@PathVariable Long userId) {

return cabBookingService.getBookingsByUserId(userId);

}

@GetMapping("/all")

public List<CabBooking> getAllBookings() {

return cabBookingService.getAllBookings();

}

@DeleteMapping("/{id}")

public ResponseEntity<String> deleteBooking(@PathVariable Long id) {

cabBookingService.deleteBooking(id);

return ResponseEntity.ok("Cab booking deleted successfully");

}

}