**Capstone Project - Java Full Stack**

**Online Bus Ticket Booking Application**

**Setting up Backend:  
 Creating SpringBoot Application Using Spring Initializer**

**Entities:**

**Bus Entity:**

**@Tag(**

**name = "table for Bus Details"**

**)**

**@Entity**

**@AllArgsConstructor**

**@NoArgsConstructor**

**@Data**

**public class Buses**

**{**

**@Id**

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

**private Long id;**

**@Column(nullable = false)**

**private String busName;**

**@Column(nullable = false)**

**private String busNumber;**

**@Column(nullable = false)**

**private String startingPoint;**

**private String startTime;**

**@Column(nullable = false)**

**private String destinationPoint;**

**private String destinationTime;**

**@Column(nullable = false)**

**private double price;**

**@ManyToOne**

**@JoinColumn(name = "bus\_provider\_id")**

**private BusProvider busProvider;**

**}**

**Admin Entity:**

**@Tag(**

**name = "it store the bus operator details"**

**)**

**@Entity**

**@AllArgsConstructor**

**@NoArgsConstructor**

**@Data**

**public class BusProvider {**

**@Id**

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

**private Long id;**

**@Column(nullable = false)**

**private String name;**

**@Column(nullable = false,unique = true)**

**private String emailId;**

**@Column(nullable = false)**

**private String password;**

**@Column(nullable = false)**

**private Long phoneNumber;**

**@OneToMany(mappedBy = "busProvider",cascade =CascadeType.*ALL*)**

**private List<Buses> buses;**

**}**

**Passenger Entity:**

**@Tag(**

**name = "it store the customer data"**

**)**

**@Entity**

**@AllArgsConstructor**

**@NoArgsConstructor**

**@Data**

**public class Passenger {**

**@Id**

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

**private Long id;**

**@Column(nullable = false)**

**private String name;**

**@Column(nullable = false,unique = true)**

**private String emailId;**

**@Column(nullable = false)**

**private String password;**

**@Column(nullable = false)**

**private Long phoneNumber;**

**@OneToMany(mappedBy = "passenger", cascade = CascadeType.*ALL*)**

**List<TicketBooking> ticketBookings;**

**}**

**Ticket Entity:**

**@Tag(**

**name = "it store the all tickets"**

**)**

**@Entity**

**@AllArgsConstructor**

**@NoArgsConstructor**

**@Data**

**public class TicketBooking {**

**@Id**

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

**private Long id;**

**private LocalDate bookedDate;**

**@ManyToOne**

**@JoinColumn(name = "passenger\_id")**

**private Passenger passenger;**

**@ManyToOne**

**@JoinColumn(name = "bus\_provider\_id")**

**private BusProvider busProvider;**

**@ManyToOne**

**@JoinColumn(name = "bus\_id")**

**private Buses buses;**

**}**

**Controllers:**

**AuthController:**

**@Tag(**

**name = "loginpage area"**

**)**

**@Controller**

**public class AuthController**

**{**

**// @Operation(**

**// summary = "it open the login page"**

**// )**

**@GetMapping("/login")**

**public String login(@RequestParam(value = "error", required = false) String error,**

**@RequestParam(value = "success", required = false) String success,**

**Model model) {**

**if (error != null) {**

**model.addAttribute("errorMsg", "Invalid email or password.");**

**}**

**if (success != null) {**

**model.addAttribute("successMsg", "Account was created successfully!!!");**

**}**

**return "loginPages";**

**}**

**@GetMapping("/register")**

**public String register()**

**{**

**return "registerbuttons";**

**}**

**}**

**Admin Controller:**

**@Tag(name = "Bus Provider",**

**description = "Endpoints for bus provider registration, dashboard, and bus management")**

**@Controller**

**@RequestMapping("/busProvider")**

**@AllArgsConstructor**

**public class BusProviderController {**

**private BusProviderService busProviderService;**

**private PassengerService passengerService;**

**private PasswordEncoder passwordEncoder;**

**private BusesService busesService;**

**private TicketBookingService ticketBookingService;**

**@Operation(**

**summary = "Show registration form for bus providers",**

**responses = @ApiResponse(responseCode = "200", description = "Registration form page")**

**)**

**@GetMapping("/registration")**

**public String Registration(Model model) {**

**BusProviderDto busProviderDto = new BusProviderDto();**

**model.addAttribute("busProvider", busProviderDto);**

**return "Bus/busOperatorRegistration";**

**}**

**@Operation(**

**summary = "Register a new bus provider"**

**)**

**@PostMapping("/saveRegisration")**

**public String saveRegistration(@ModelAttribute("busProvider") BusProviderDto busProviderDto, BindingResult bindingResult, Model model) {**

**if (busProviderService.findBusProviderByEmailId(busProviderDto.getEmailId()) != null) {**

**bindingResult.rejectValue("emailId", "email.exist", "This email is already registered. Please use another email.");**

**} else if (passengerService.findByEmailid(busProviderDto.getEmailId()) != null) {**

**bindingResult.rejectValue("emailId", "email.exist", "This email is already registered in customer. Please use another email.");**

**}**

**if (bindingResult.hasErrors()) {**

**return "Bus/busOperatorRegistration";**

**}**

**busProviderDto.setPassword(passwordEncoder.encode(busProviderDto.getPassword()));**

**busProviderService.saveBusProvider(busProviderDto);**

**return "redirect:/login?success";**

**}**

**@Operation(**

**summary = "Show bus provider dashboard",**

**responses = @ApiResponse(responseCode = "200", description = "Dashboard page")**

**)**

**@GetMapping("/dashboard")**

**public String dashboard(Authentication authentication, Model model) {**

**String email = authentication.getName();**

**BusProviderDto busProviderDto = busProviderService.findBusProviderByEmailId(email);**

**model.addAttribute("busprovder", busProviderDto);**

**List<BusesDto> busesDtos = busesService.findBusesByBusProviderEmailId(email);**

**model.addAttribute("buses", busesDtos);**

**return "Bus/busproviderDashboard";**

**}**

**@Operation(**

**summary = "Show add bus form",**

**responses = @ApiResponse(responseCode = "200", description = "Add bus form page")**

**)**

**@GetMapping("/addbus")**

**public String addBus(Authentication authentication, Model model) {**

**String email = authentication.getName();**

**BusProviderDto busProviderDto = busProviderService.findBusProviderByEmailId(email);**

**model.addAttribute("busprovider", busProviderDto);**

**BusesDto busesDto = new BusesDto();**

**busesDto.setBusProvider\_id(busProviderDto.getId());**

**model.addAttribute("busdto", busesDto);**

**return "Bus/addBus";**

**}**

**@PostMapping("/Savethebuses")**

**public String SaveTHeBuses(@ModelAttribute("busdto") BusesDto busesDto, Authentication authentication) {**

**busesService.saveBueses(busesDto);**

**return "redirect:/busProvider/dashboard";**

**}**

**@Operation(**

**summary = "Delete a bus by ID"**

**)**

**@GetMapping("/deleteTheBus/{id}")**

**public String deleteTheBus(@PathVariable("id") Long id, RedirectAttributes redirectAttributes) {**

**try {**

**busesService.deleteBusById(id);**

**redirectAttributes.addFlashAttribute("successMessage", "Bus deleted successfully.");**

**} catch (IllegalStateException e) {**

**redirectAttributes.addFlashAttribute("errorMessage", e.getMessage());**

**}**

**return "redirect:/busProvider/dashboard";**

**}**

**@GetMapping("/upadteTheBusData/{id}")**

**public String upadteTheBusData(@PathVariable("id") Long id, Model model) {**

**if (id == null) {**

**// Handle error, redirect, or show message**

**return "redirect:/busProvider/dashboard?error=invalid\_id";**

**}**

**BusesDto busesDto = busesService.findyById(id);**

**model.addAttribute("oldbus", busesDto);**

**return "Bus/update\_bus";**

**}**

**@Operation(**

**summary = "Save updated bus details"**

**)**

**@PostMapping("/savetheupdatebus")**

**public String savetheupdatebus(@ModelAttribute("oldbus") BusesDto busesDto) {**

**busesService.updateTheBus(busesDto);**

**return "redirect:/busProvider/dashboard";**

**}**

**@Operation(**

**summary = "Show date entry form for viewing tickets of a bus")**

**@GetMapping("/viewBusTickets/{busId}")**

**public String showDateEntryForm(@PathVariable Long busId, Model model) {**

**model.addAttribute("busId", busId);**

**return "Bus/dateEntryForm";**

**}**

**@Operation(**

**summary = "Show tickets for a bus on a specific date")**

**@PostMapping("/viewBusTickets")**

**public String showTicketsForDate(@RequestParam Long busId,**

**@RequestParam @DateTimeFormat(iso = DateTimeFormat.ISO.*DATE*) LocalDate bookedDate,**

**Model model) {**

**Buses bus = busesService.findById(busId);**

**model.addAttribute("bus", bus);**

**List<TicketBooking> tickets = ticketBookingService.findByBusIdAndBookedDate(busId, bookedDate);**

**model.addAttribute("tickets", tickets);**

**model.addAttribute("bookedDate", bookedDate);**

**return "Bus/busTicketsByDate";**

**}**

**}**

**Passenger Controller:**

**@Tag(name = "Passenger",**

**description = "Endpoints for passenger registration, dashboard, bus search, booking, and ticket management")**

**@Controller**

**@AllArgsConstructor**

**@RequestMapping("/passenger")**

**public class PassengerController {**

**private PassengerService passengerService;**

**private BusProviderService busProviderService;**

**private BusesService busesService;**

**private TicketBookingService ticketBookingService;**

**private PasswordEncoder passwordEncoder;**

**@GetMapping("/register")**

**public String register(Model model) {**

**PassengerDto passenger = new PassengerDto();**

**model.addAttribute("passenger", passenger);**

**return "Passenger/passengerRegistration";**

**}**

**@Operation(**

**summary = "Register a new passenger"**

**)**

**@PostMapping("/savePassenger")**

**public String savePassenger(@ModelAttribute("passenger") PassengerDto passengerDto, BindingResult bindingResult, Model model) {**

**if (passengerService.findByEmailid(passengerDto.getEmailId()) != null) {**

**bindingResult.rejectValue("emailId", "email.exist", "This email is already registered. Please use another email.");**

**} else if (busProviderService.findBusProviderByEmailId(passengerDto.getEmailId()) != null) {**

**bindingResult.rejectValue("emailId", "email.exist", "This email is already registered in Bus PProvider ares. Please use another email.");**

**}**

**if (bindingResult.hasErrors()) {**

**return "Passenger/passengerRegistration";**

**}**

**passengerDto.setPassword(passwordEncoder.encode(passengerDto.getPassword()));**

**passengerService.savepassnger(passengerDto);**

**return "redirect:/login?success";**

**}**

**@Operation(**

**summary = "Show passenger dashboard"**

**)**

**@GetMapping("/dashboard")**

**public String Dashboard(Authentication authentication, Model model) {**

**String email = authentication.getName();**

**PassengerDto passengerDto = passengerService.findByEmailid(email);**

**model.addAttribute("passenger", passengerDto);**

**return "Passenger/passengerdashboard";**

**}**

**@Operation(**

**summary = "Search buses by route")**

**@GetMapping("/search")**

**public String serachbus(@RequestParam("form") String form,**

**@RequestParam("to") String to,**

**Authentication authentication,**

**Model model**

**) {**

**String email = authentication.getName();**

**PassengerDto passengerDto = passengerService.findByEmailid(email);**

**model.addAttribute("passenger", passengerDto);**

**List<BusesDto> busesDtos = busesService.findByRoute(form, to);**

**if (busesDtos.isEmpty()) {**

**model.addAttribute("errorMessage", "No Available Bus for This Destination.");**

**} else {**

**model.addAttribute("buses", busesDtos);**

**}**

**return "Passenger/afterBusSearch";**

**}**

**@GetMapping("/getBookingDetails/{id}")**

**public String getBookingDetails(@PathVariable("id") Long id, Model model) {**

**//Fetch the bus details**

**BusesDto busesDto = busesService.findyById(id);**

**model.addAttribute("bus", busesDto);**

**//Fetch the bus provider details**

**BusProviderDto busProviderDto = busProviderService.findById(busesDto.getBusProvider\_id());**

**model.addAttribute("provider", busProviderDto);**

**//prepare booking Dto**

**TicketBookingDto ticketBookingDto = new TicketBookingDto();**

**ticketBookingDto.setBusesId(busesDto.getId());**

**ticketBookingDto.setBusProviderId(busesDto.getBusProvider\_id());**

**model.addAttribute("ticket", ticketBookingDto);**

**return "ticket/ticketBookingPage";**

**}**

**// Handle booking form submission**

**@PostMapping("/saveTheBooking")**

**public String saveTheBooking(@ModelAttribute("ticket") TicketBookingDto ticketBookingDto,**

**Authentication authentication, Model model) {**

**//Fetch the passenger**

**String emailId = authentication.getName();**

**PassengerDto passengerDto = passengerService.findByEmailid(emailId);**

**ticketBookingDto.setPassengerId(passengerDto.getId());**

**//fec**

**ticketBookingDto.setBusProviderId(ticketBookingDto.getBusProviderId());**

**ticketBookingDto.setBusesId(ticketBookingDto.getBusesId());**

**// Save booking**

**ticketBookingService.saveBooking(ticketBookingDto);**

**return "redirect:/passenger/dashboard";**

**}**

**@Operation(**

**summary = "View passenger ticket history")**

**@GetMapping("/ticketHistory")**

**public String tickerHistory(Authentication authentication, Model model) {**

**String email = authentication.getName();**

**// Fetch the passenger using the login details**

**PassengerDto passengerDto = passengerService.findByEmailid(email);**

**model.addAttribute("passenger", passengerDto);**

**//Fetch all tickets were match by passenger id**

**// List<TicketBookingDto> allTickets=ticketBookingService.getTicketsByPassengerId(passengerDto.getId());**

**// model.addAttribute("tickets",allTickets);**

**List<TicketBooking> tickets = ticketBookingService.getTicketsByPassengerId(passengerDto.getId());**

**model.addAttribute("tickets", tickets);**

**return "ticket/ticketHistory";**

**}**

**@Operation(**

**summary = "View most recently booked ticket")**

**@GetMapping("/bookedticket")**

**public String bookedTicket(Authentication authentication, Model model) {**

**String email = authentication.getName();**

**// Fetch the passenger using the login details**

**PassengerDto passengerDto = passengerService.findByEmailid(email);**

**List<TicketBooking> ticketBookingList = ticketBookingService.getTicketsByPassengerId(passengerDto.getId());**

**// TicketBooking ticket=ticketBookingList.get(ticketBookingList.size()-1);**

**// if (!ticketBookingList.isEmpty()) {**

**// TicketBooking ticket = ticketBookingList.get(ticketBookingList.size() - 1);**

**// model.addAttribute("ticket", ticket);**

**// } else {**

**model.addAttribute("tickets", ticketBookingList); // or handle with a custom message**

**return "ticket/bookedticket";**

**}**

**@GetMapping("/delete/{id}")**

**public String deleteTicker(@PathVariable("id") Long id) {**

**ticketBookingService.deleteTicket(id);**

**return "redirect:/passenger/dashboard";**

**}**

**}**

**Repository:**

**Bus Repository:**

**public interface BusesRepository extends JpaRepository<Buses,Long>**

**{**

**List<Buses> findByBusProviderId(Long id);**

**List<Buses> findByStartingPointIgnoreCaseAndDestinationPointIgnoreCase(String startingPoint, String destinationPoint);**

**}**

**Admin Repository:**

**public interface BusProviderRepository extends JpaRepository<BusProvider,Long>**

**{**

**BusProvider findByEmailId(String emailId);**

**}**

**Passenger Repository:**

**public interface PassengerRepository extends JpaRepository<Passenger,Long> {**

**Passenger findByEmailId(String emailId);**

**}**

**Booking Repository:**

**public interface TicketBookingRepository extends JpaRepository<TicketBooking,Long> {**

**List<TicketBooking> findByPassengerId(Long passengerId);**

**List<TicketBooking> findByBusesIdAndBookedDate(Long busId, LocalDate bookedDate);**

**List<TicketBooking> findByBusesId(Long busId);**

**}**

**Services**

**Bus Service:**

**@Tag(**

**name = "it make curd operation",**

**description = "it make the full operation in curd like save the bus , deleted bus "**

**)**

**@Service**

**public class BusesServiceImpl implements BusesService {**

**@Autowired**

**private BusesRepository busesRepository;**

**@Autowired**

**private BusProviderRepository busProviderRepository;**

**@Autowired**

**TicketBookingRepository ticketBookingRepository;**

**@Autowired**

**private ModelMapper modelMapper;**

**@Operation**

**(summary = "it save the new bus")**

**@Override**

**public BusesDto saveBueses(BusesDto dto) {**

**Buses bus = modelMapper.map(dto, Buses.class);**

**// Fetch and set BusProvider**

**BusProvider provider = busProviderRepository.findById(dto.getBusProvider\_id())**

**.orElseThrow(() -> new RuntimeException("BusProvider not found"));**

**bus.setBusProvider(provider);**

**Buses savedBus = busesRepository.save(bus);**

**return modelMapper.map(savedBus, BusesDto.class);**

**}**

**@Operation(**

**summary = "if find all buses using the bus operator email"**

**)**

**@Override**

**public List<BusesDto> findBusesByBusProviderEmailId(String emailId) {**

**// Fetch the BusProvider entity using email I**

**BusProvider busProvider = busProviderRepository.findByEmailId(emailId);**

**if (busProvider == null) {**

**return List.*of*();**

**}**

**// Fetch all buses associated with the busProvider ID**

**List<Buses> buses = busesRepository.findByBusProviderId(busProvider.getId());**

**// Convert each Buses entity into BusesDto**

**return buses.stream()**

**.map(bus -> modelMapper.map(bus, BusesDto.class))**

**.toList();**

**}**

**@Override**

**public void deleteBusById(Long id) {**

**List<TicketBooking> bookings = ticketBookingRepository.findByBusesId(id);**

**if (!bookings.isEmpty()) {**

**throw new IllegalStateException("Cannot delete bus: tickets already booked.");**

**}**

**busesRepository.deleteById(id);**

**}**

**@Operation**

**(summary = "it find the bus by using it id")**

**@Override**

**public BusesDto findyById(Long id) {**

**Buses buses = busesRepository.findById(id)**

**.orElseThrow(() -> new RuntimeException("No buses"));**

**BusesDto busesDto = new BusesDto();**

**busesDto.setId(buses.getId());**

**busesDto.setBusName(buses.getBusName());**

**busesDto.setBusNumber(buses.getBusNumber());**

**busesDto.setStartingPoint(buses.getStartingPoint());**

**busesDto.setStartTime(buses.getStartTime());**

**busesDto.setDestinationPoint(buses.getDestinationPoint());**

**busesDto.setDestinationTime(buses.getDestinationTime());**

**busesDto.setPrice(buses.getPrice());**

**// Set busProvider\_id manually from the entity’s relation**

**if (buses.getBusProvider() != null) {**

**busesDto.setBusProvider\_id(buses.getBusProvider().getId());**

**}**

**return busesDto;**

**}**

**@Operation(**

**summary = "it update the existing bus data"**

**)**

**@Override**

**public BusesDto updateTheBus(BusesDto busesDto) {**

**Buses existingbus = busesRepository.findById(busesDto.getId())**

**.orElseThrow(() -> new RuntimeException("no buses"));**

**existingbus.setBusName(busesDto.getBusName());**

**existingbus.setBusNumber(busesDto.getBusNumber());**

**existingbus.setStartingPoint(busesDto.getStartingPoint());**

**existingbus.setStartTime(busesDto.getStartTime());**

**existingbus.setDestinationPoint(busesDto.getDestinationPoint());**

**existingbus.setDestinationTime(busesDto.getDestinationTime());**

**existingbus.setPrice(busesDto.getPrice());**

**Buses updatedBus = busesRepository.save(existingbus);**

**return modelMapper.map(updatedBus, BusesDto.class);**

**}**

**@Operation**

**(summary = "it find the bus route")**

**@Override**

**public List<BusesDto> findByRoute(String form, String to) {**

**List<Buses> buses = busesRepository.findByStartingPointIgnoreCaseAndDestinationPointIgnoreCase(form, to);**

**return buses.stream()**

**.map(a -> modelMapper.map(a, BusesDto.class))**

**.toList();**

**}**

**@Override**

**public Buses findById(Long id) {**

**return busesRepository.findById(id).orElseThrow(() -> new RuntimeException("No bus"));**

**}**

**}**

**Admin Service:**

**@Tag(**

**name = "it work for only the bus provider"**

**)**

**@Service**

**@AllArgsConstructor**

**public class BusProviderServiceImpl implements BusProviderService {**

**private BusProviderRepository busProviderRepository;**

**private ModelMapper modelMapper;**

**@Operation(**

**summary = "it save the new bus operator"**

**)**

**@Override**

**public BusProviderDto saveBusProvider(BusProviderDto busProviderDto) {**

**BusProvider busProvider=modelMapper.map(busProviderDto,BusProvider.class);**

**BusProvider savedBusprovider=busProviderRepository.save(busProvider);**

**return modelMapper.map(savedBusprovider,BusProviderDto.class);**

**}**

**@Operation(**

**summary = "it find the bus operator with their emailId"**

**)**

**@Override**

**public BusProviderDto findBusProviderByEmailId(String emailId) {**

**BusProvider busProvider=busProviderRepository.findByEmailId(emailId);**

**if(busProvider==null)**

**{**

**return null;**

**}**

**return modelMapper.map(busProvider,BusProviderDto.class);**

**}**

**@Operation(**

**summary = "it get bus operator by using id"**

**)**

**@Override**

**public BusProviderDto findById(Long id) {**

**BusProvider busProvider=busProviderRepository.findById(id)**

**.orElseThrow(()->new RuntimeException("No busProvider not found"));**

**return modelMapper.map(busProvider,BusProviderDto.class);**

**}**

**}**

**Passenger Service:**

**@Tag(**

**name = "it save and get data fro the passenger"**

**)**

**@Service**

**@AllArgsConstructor**

**public class PassengerServiceImp implements PassengerService {**

**private PassengerRepository passengerRepository;**

**private ModelMapper modelMapper;**

**@Operation(summary = "it save new passenger")**

**@Override**

**public PassengerDto savepassnger(PassengerDto passengerDto) {**

**Passenger passenger=modelMapper.map(passengerDto,Passenger.class);**

**Passenger savedPassenger= passengerRepository.save(passenger);**

**return modelMapper.map(savedPassenger,PassengerDto.class);**

**}**

**@Operation(summary = "it get the passenger by using email")**

**@Override**

**public PassengerDto findByEmailid(String email) {**

**Passenger passenger=passengerRepository.findByEmailId(email);**

**if(passenger==null)**

**{**

**return null;**

**}**

**return modelMapper.map(passenger,PassengerDto.class);**

**}**

**}**

**Ticket Service:**

**@Tag(**

**name = "this layer work was to make the tickets"**

**)**

**@Service**

**@AllArgsConstructor**

**public class TicketBookingServiceImpl implements TicketBookingService {**

**private final TicketBookingRepository ticketBookingRepository;**

**private final PassengerRepository passengerRepository;**

**private final BusProviderRepository busProviderRepository;**

**private final BusesRepository busesRepository;**

**private final ModelMapper modelMapper;**

**@Operation**

**(summary = "it save the ticket if any user book it")**

**@Override**

**public TicketBookingDto saveBooking(TicketBookingDto ticketBookingDto) {**

**TicketBooking booking=new TicketBooking();**

**booking.setBookedDate(ticketBookingDto.getBookedDate());**

**//set**

**booking.setPassenger(passengerRepository.findById(ticketBookingDto.getPassengerId())**

**.orElseThrow());**

**booking.setBusProvider(busProviderRepository.findById(ticketBookingDto.getBusProviderId()).orElseThrow());**

**booking.setBuses(busesRepository.findById(ticketBookingDto.getBusesId()).orElseThrow());**

**ticketBookingRepository.save(booking);**

**return modelMapper.map(booking,TicketBookingDto.class);**

**}**

**@Operation(summary = "it get all tickets from using the passenger id")**

**@Override**

**public List<TicketBooking> getTicketsByPassengerId(Long passengerId) {**

**List<TicketBooking> ticketBookings=ticketBookingRepository.findByPassengerId(passengerId);**

**// return ticketBookings.stream().map(a->modelMapper.map(a,TicketBookingDto.class)).toList();**

**return ticketBookings;**

**}**

**@Operation(summary = "deleted the ticket by it id")**

**@Override**

**public void deleteTicket(Long id) {**

**ticketBookingRepository.deleteById(id);**

**}**

**@Operation(summary = "it get ticket for bus operator by seraching based on day")**

**@Override**

**public List<TicketBooking> findByBusIdAndBookedDate(Long busId, LocalDate bookedDate) {**

**return ticketBookingRepository.findByBusesIdAndBookedDate(busId, bookedDate);**

**}**

**}**

**Setting Up Frontend**

**Using HTML,js,Bootstrap**

**Admin Pages**

**Admin/User Registration Page:**

**For Admin**

**<body>**

**<nav class="navbar mb-4 bg-primary fixed-top ">**

**<div class="container">**

**<span class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i class="fa-solid fa-bus"></i></span> Online Bus Reservation System</span>**

**</div>**

**</nav>**

**<div class="container col-12 mt-5">**

**<div class="container row d-flex mb-3 justify-content-center">**

**<div class="mt-4">**

**<h2 class="d-flex justify-content-center">Admin Registration</h2>**

**<div class="container shadow-lg ">**

**<div class="container">**

**<div class="container p-3">**

**<form class="" th:object="${busProvider}" method="post"**

**th:action="@{/busProvider/saveRegisration}">**

**<div class="mb-2">**

**<label id="name" class="form-label">Name</label>**

**<input type="text" th:field="\*{name}" class="form-control" placeholder="Enter your Name"**

**required>**

**</div>**

**<div class="mb-2">**

**<label id="emailId" class="form-label">Email Id</label>**

**<input th:field="\*{emailId}" type="email" class="form-control"**

**placeholder="Enter your Email id" required>**

**<p th:errors="\*{emailId}" class="text-danger"**

**th:if="${#fields.hasErrors('emailId')}"></p>**

**</div>**

**<div class="mb-2">**

**<label id="password" class="form-label">Password</label>**

**<input th:field="\*{password}" type="password" class="form-control"**

**placeholder="Enter your password" required>**

**</div>**

**<div class="mb-4">**

**<label id="phoneNumber" class="form-label">Mobile Number</label>**

**<input th:field="\*{phoneNumber}" type="text" maxlength="11" class="form-control"**

**placeholder="Enter your Mobile Number" required>**

**</div>**

**<button type="submit" class="btn btn-primary">Register</button>**

**<p class="e d-flex justify-content-center">**

**Already have an account? <a th:href="@{/login}">Login!</a>**

**</p>**

**</form>**

**</div>**

**</div>**

**</div>**

**</div>**

**</div>**

**</body>**

**For passengers**

**<body>**

**<nav class=" navbar mb-6 bg-primary fixed-top ">**

**<div class="container">**

**<h2 class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i class="fa-solid fa-bus"></i></span> Online Bus Registration System</h2>**

**</div>**

**</nav>**

**<div class="container col-12 mt-5">**

**<div class="container row d-flex justify-content-center">**

**<h2 class="d-flex justify-content-center mt-3">Passenger Registration</h2>**

**<div class="container shadow-lg ">**

**<div class="container">**

**<div class="container p-3">**

**<form class="" th:object="${passenger}" method="post" th:action="@{/passenger/savePassenger}">**

**<div class="mb-2">**

**<label id="Firstnaem" class="form-label text-dark" style="text-bold">Name</label>**

**<input type="text" th:field="\*{name}" class="form-control" placeholder="Enter your Name"**

**required>**

**</div>**

**<div class="mb-2">**

**<label id="emailId" class="form-label">Email Id</label>**

**<input th:field="\*{emailId}" type="email" class="form-control"**

**placeholder="Enter your Email id"**

**required>**

**<p th:errors="\*{emailId}" class="text-danger" th:if="${#fields.hasErrors('emailId')}"></p>**

**</div>**

**<div class="mb-2">**

**<label id="password" class="form-label">Password</label>**

**<input th:field="\*{password}" type="password" class="form-control"**

**placeholder="Enter your password"**

**required>**

**</div>**

**<div class="mb-4">**

**<label id="phonenumber" class="form-label">Mobile Number</label>**

**<input th:field="\*{phoneNumber}" type="text" maxlength="11" class="form-control"**

**placeholder="Enter your Mobile Number" required>**

**</div>**

**<button type="submit" class="btn btn-primary text-center">Register</button>**

**<p class="loginpageregence d-flex justify-content-center">**

**Already have an account? <a th:href="@{/login}">Login!</a>**

**</p>**

**</form>**

**</div>**

**</div>**

**</div>**

**</div>**

**</div>**

**</body>**

**Admin/User Login:**

**<body>**

**<nav class="navbar bg-primary mb-5" >**

**<div class="container-fluid">**

**<h2 class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i class="fa-solid fa-bus"></i></span> Online Bus Registration System</h2>**

**</div>**

**</nav>**

**<!-- Login Form -->**

**<div class="container col-12 login-container">**

**<h2 class="text-center mb-4">Welcome Back</h2>**

**<div th:if="${errorMsg}" class="alert alert-danger" th:text="${errorMsg}"></div>**

**<div role="alert" id="autoCloseAlert">**

**<div th:if="${param.success}">**

**<div class="alert alert-info">Account was Created Successfully!!!<i**

**class="fa-regular fa-face-smile-wink"></i></div>**

**</div>**

**</div>**

**<div class="container col-8 shadow-lg">**

**<div class="row">**

**<form class="loginform form shadow" th:action="@{/login}" method="post">**

**<div class="container m-3 p-3">**

**<!-- Email -->**

**<div class="mb-3 ">**

**<label for="email" class="form-label">Email ID</label>**

**<i class="fa-regular fa-envelope"></i>**

**<input type="email" class="form-control" id="email" name="username"**

**placeholder="Enter your Email ID"**

**required/>**

**</div>**

**<!-- Password -->**

**<div class="mb-3">**

**<label for="password" class="form-label">Password</label>**

**<i class="fa-solid fa-lock"></i>**

**<input type="password" class="form-control" id="password" name="password"**

**placeholder="Enter your Password"**

**required/>**

**</div>**

**<div class="mb-3">**

**<!-- Submit Button -->**

**<button type="submit" class="btn btn-primary">Login</button>**

**</div>**

**<!-- Register Link -->**

**<p class="text-center mt-3 mb-0">**

**Don't have an Account? <a th:href="@{/register}">Register!</a>**

**</p>**

**</div>**

**</form>**

**</div>**

**</div>**

**</div>**

**<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>**

**<script>**

**// Auto-close alert after 2 seconds**

**setTimeout(function() {**

**var alertNode = document.getElementById('autoCloseAlert');**

**if (alertNode) {**

**var Alert = bootstrap.Alert.getOrCreateInstance(alertNode);**

**Alert.close();**

**}**

**}, 2000);**

**</script>**

**</body>**

**Bus Related Pages**

**DashBoard:**

**<body>**

**<nav class=" navbar mb-4 bg-primary fixed-top ">**

**<div class="container">**

**<h2 class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i**

**class="fa-solid fa-bus"></i></span> Online Bus Registration System</h2>**

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

**<button type="submit" class="btn btn-danger">Logout</button>**

**</form>**

**</div>**

**</nav>**

**<div class="container col-12 mt-5">**

**<div class=" row mt-5 d-flex justify-content-center">**

**<div>**

**<div class="mt-5">**

**<div th:if="${errorMessage}" class="alert alert-danger alert-dismissible fade show" id="errorAlert"**

**role="alert">**

**<p th:text="${errorMessage}"></p>**

**</div>**

**<div th:if="${successMessage}" class="alert alert-success alert-dismissible fade show" id="SuccessAlert"**

**role="alert">**

**<p th:text="${successMessage}"></p>**

**</div>**

**</div>**

**<div class=" mt-3 d-flex justify-content-between">**

**<h2 th:text="${busprovder.name}" class="text-uppercase"></h2>**

**<a th:href="@{/busProvider/addbus}" class=" btn btn-success">Add Bus</a>**

**</div>**

**<div class=" mt-3 shadow-lg ">**

**<div class="">**

**<h2 class=" d-flex justify-content-center">Available Buses</h2>**

**<div class=" ">**

**<div class="table-responsive p-3">**

**<table class="table table-bordered ">**

**<thead class="bg-primary">**

**<tr class="bg-primary">**

**<th>Bus Name</th>**

**<th>Bus Number</th>**

**<th>Departure Place</th>**

**<th>Departure Time</th>**

**<th>Destination Place</th>**

**<th>Destination Time</th>**

**<th>Price</th>**

**<th>Action</th>**

**</tr>**

**</thead>**

**<tbody>**

**<tr th:each="bus:${buses}">**

**<td th:text="${bus.busName}"></td>**

**<td th:text="${bus.busNumber}"></td>**

**<td th:text="${bus.startingPoint}"></td>**

**<td th:text="${bus.startTime}"></td>**

**<td th:text="${bus.destinationPoint}"></td>**

**<td th:text="${bus.destinationTime}"></td>**

**<td th:text="${bus.price}"></td>**

**<td>**

**<a th:href="@{'/busProvider/viewBusTickets/'+${bus.id}}"**

**class="ViewButton btn btn-info">View</a>**

**<a th:href="@{'/busProvider/upadteTheBusData/' + ${bus.id} }"**

**class="updatebutton btn btn-warning">Update</a>**

**<a th:href="@{'/busProvider/deleteTheBus/' + ${bus.id} }"**

**class="Deletebutton btn btn-danger">Delete</a>**

**</td>**

**</tr>**

**</tbody>**

**</table>**

**</div>**

**</div>**

**</div>**

**</div>**

**</div>**

**</div>**

**</div>**

**</body>**

**Add Bus:**

**<body>**

**<nav class=" navbar mb-5 bg-primary fixed-top ">**

**<div class="container">**

**<h2 class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i**

**class="fa-solid fa-bus"></i></span> Online Bus Registration System</h2>**

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

**<button type="submit" class="btn btn-danger">Logout</button>**

**</form>**

**</div>**

**</nav>**

**<div class="container-fluid col-8 mt-4 p-4 shadow-lg ">**

**<h2 class="d-flex justify-content-center name mt-4">Add Bus Details</h2>**

**<div class="formbussadder">**

**<form th:object="${busdto}" th:action="@{/busProvider/Savethebuses}" method="post">**

**<!-- <input type="hidden" th:field="\*{id}" th:value="${busprovider.id}">-->**

**<input type="hidden" th:field="\*{busProvider\_id}" th:value="${busprovider.id}">**

**<div class="mb-3">**

**<label for="BusName" class="form-label">Bus Name</label>**

**<input th:field="\*{busName}" type="text" class="form-control" name="BusName" id="BusName"**

**placeholder="Enter Bus Name">**

**</div>**

**<div class="mb-3">**

**<label for="BusNumber" class="form-label">Bus Number</label>**

**<input th:field="\*{busNumber}" type="text" class="form-control" name="BusNumber" id="BusNumber"**

**placeholder="Enter Bus Number">**

**</div>**

**<div class="mb-3">**

**<label for="startingPoint" class="form-label">Departure Place</label>**

**<input th:field="\*{startingPoint}" type="text" class="form-control" name="startingPoint"**

**id="startingPoint" placeholder="Enter Departure Place">**

**</div>**

**<div class="mb-3">**

**<label for="startTime" class="form-label">Departure Time</label>**

**<input th:field="\*{startTime}" type="text" class="form-control" name="startTime" id="startTime"**

**placeholder="Enter Departure Time">**

**</div>**

**<div class="mb-3">**

**<label for="destinationPoint" class="form-label">Destination Place</label>**

**<input th:field="\*{destinationPoint}" type="text" class="form-control" name="destinationPoint"**

**id="destinationPoint" placeholder="Enter Destination Place">**

**</div>**

**<div class="mb-3">**

**<label for="destinationTime" class="form-label">Destination Time</label>**

**<input th:field="\*{destinationTime}" type="text" class="form-control" name="destinationTime"**

**id="destinationTime" placeholder="Enter Destination Time">**

**</div>**

**<div class="mb-3">**

**<label for="price" class="form-label">Price</label>**

**<input th:field="\*{price}" type="text" class="form-control" name="price" id="price"**

**placeholder="Enter price">**

**</div>**

**<div class="d-flex justify-content-between mb-3">**

**<button type="submit" class=" btn btn-primary btn-sm">SUBMIT</button>**

**<a th:href="@{/busProvider/dashboard}" class=" btn btn-warning">Back</a>**

**</div>**

**</form>**

**</div>**

**</div>**

**</body>**

**Update Bus:**

**<body>**

**<header class=" my-5">**

**<nav class="logo navbar navbar-dark mb-4 bg-primary fixed-top ">**

**<div class="container">**

**<span class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i class="fa-solid fa-bus"></i></span> Online Bus Reservation System</span>**

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

**<button type="submit" class="btn btn-danger">Logout</button>**

**</form>**

**</div>**

**</nav>**

**</header>**

**<div class="container-fluid col-8 mt-4 p-4 shadow-lg ">**

**<h2 class="d-flex justify-content-center name mt-4">Update Bus Details</h2>**

**<div class="formbussadder">**

**<form th:object="${oldbus}" th:action="@{/busProvider/savetheupdatebus}" method="post" >**

**<input type="hidden" th:field="\*{id}">**

**<div class="mb-3">**

**<label for="BusName" class="form-label">Bus Name</label>**

**<input th:field="\*{busName}" type="text" class="form-control" name="BusName" id="BusName" placeholder="Enter Bus Name">**

**</div>**

**<div class="mb-3">**

**<label for="BusNumber" class="form-label">Bus Number</label>**

**<input th:field="\*{busNumber}" type="text" class="form-control" name="BusNumber" id="BusNumber" placeholder="Enter Bus Number">**

**</div>**

**<div class="mb-3">**

**<label for="startingPoint" class="form-label">Departure Place</label>**

**<input th:field="\*{startingPoint}" type="text" class="form-control" name="startingPoint" id="startingPoint" placeholder="Enter Departure Place">**

**</div>**

**<div class="mb-3">**

**<label for="startTime" class="form-label">Departure Time</label>**

**<input th:field="\*{startTime}" type="text" class="form-control" name="startTime" id="startTime" placeholder="Enter Departure Time">**

**</div>**

**<div class="mb-3">**

**<label for="destinationPoint" class="form-label">Destination Place</label>**

**<input th:field="\*{destinationPoint}" type="text" class="form-control" name="destinationPoint" id="destinationPoint" placeholder="Enter Destination Place">**

**</div>**

**<div class="mb-3">**

**<label for="destinationTime" class="form-label">Destination Time</label>**

**<input th:field="\*{destinationTime}" type="text" class="form-control" name="destinationTime" id="destinationTime" placeholder="Enter Destination Time">**

**</div>**

**<div class="mb-3">**

**<label for="price" class="form-label">Price</label>**

**<input th:field="\*{price}" type="text" class="form-control" name="price" id="price" placeholder="Enter the trip price">**

**</div>**

**<div class="d-flex justify-content-between mb-3">**

**<button type="submit" class="btn btn-primary btn-sm" >SUBMIT</button>**

**<a th:href="@{/busProvider/dashboard}" class=" mb-4 btn btn-warning" >Back</a>**

**</div>**

**</form>**

**</div>**

**</div>**

**</div>**

**</body>**

**Bus Searching pages**

**<body>**

**<header>**

**<nav class="navbar navbar-dark bg-primary fixed-top">**

**<div class="container d-flex justify-content-between">**

**<h2 class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i**

**class="fa-solid fa-bus"></i></span> Online Bus Registration System</h2>**

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

**<button type="submit" class="btn btn-light fw-bold">Logout</button>**

**</form>**

**</div>**

**</nav>**

**</header>**

**<div class="container-fluid mt-5 p-4">**

**<div class="mt-5">**

**<h3 th:text="'Tickets for ' + ${bus.busName} + ' on ' + ${bookedDate}"></h3>**

**</div>**

**<div class="table-responsive">**

**<table class="table table-bordered table-hover">**

**<thead>**

**<tr>**

**<th>Ticket ID</th>**

**<th>Passenger Name</th>**

**<th>Booked Date</th>**

**<!-- Add more columns as needed -->**

**</tr>**

**</thead>**

**<tbody>**

**<tr th:each="ticket : ${tickets}">**

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

**<td th:text="${ticket.passenger.name}"></td>**

**<td th:text="${ticket.bookedDate}"></td>**

**</tr>**

**<tr th:if="${#lists.isEmpty(tickets)}">**

**<td colspan="10" class="text-center">NoOne Booked .</td>**

**</tr>**

**</tbody>**

**</table>**

**</div>**

**</div>**

**</body>**

**<body>**

**<nav class=" navbar mb-4 bg-primary fixed-top">**

**<div class="container d-flex justify-content-between">**

**<h2 class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i**

**class="fa-solid fa-bus"></i></span> Online Bus Registration System</h2>**

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

**<button type="submit" class="btn btn-danger">Logout</button>**

**</form>**

**</div>**

**</nav>**

**<div class="container mt-5">**

**<div class=" row d-flex justify-content-center">**

**<div class="container col-4 mt-3 p-3 ">**

**<div class=" p-3 shadow-lg">**

**<h3 class="d-flex justify-content-center mt-4">Select Booking Date</h3>**

**<form th:action="@{/busProvider/viewBusTickets}" method="post">**

**<input type="hidden" name="busId" th:value="${busId}"/>**

**<div class="mb-3">**

**<label for="bookedDate" class="form-label">Booking Date:</label>**

**<input type="date" name="bookedDate" id="bookedDate" class="form-control" required/>**

**</div>**

**<button type="submit" class="btn btn-primary btn-sm w-100 mb-3 p-3">View Tickets</button>**

**</form>**

**</div>**

**</div>**

**</div>**

**</div>**

**</body>**

**Passenger Related Pages**

**DashBoard**

**<body>**

**<nav class=" navbar navbar-dark mb-4 bg-primary fixed-top ">**

**<div class="container">**

**<h2 class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i**

**class="fa-solid fa-bus"></i></span> Online Bus Registration System</h2>**

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

**<button type="submit" class="btn btn-danger">Logout</button>**

**</form>**

**</div>**

**</nav>**

**<div class="container col-8 mt-5 p-5">**

**<div class="container row d-flex justify-content-center">**

**<div class="username mb-3 p-2">**

**<h2 th:text="${passenger.name}" class="text-uppercase"></h2>**

**<div class="ticketclass d-flex justify-content-between">**

**<a th:href="@{/passenger/bookedticket}" class="btn btn-dark">Ticket</a>**

**<a th:href="@{/passenger/ticketHistory}" class="btn btn-info">History</a>**

**</div>**

**</div>**

**<div class="">**

**<form th:action="@{/passenger/search}" method="get">**

**<div class="shadow">**

**<div class="container p-4">**

**<div class="searchfrom">**

**<label for="form">Form</label>**

**<input type="text" class="form-control" name="form" id="form">**

**</div>**

**<div class="searchfrom">**

**<label for="to">TO</label>**

**<input type="text" class="form-control mb-2" name="to" id="to">**

**</div>**

**<div class="mb-4">**

**<button type="submit" class=" mb-3 btn btn-secondary d-flex justify-content-center">Search**

**</button>**

**</div>**

**</div>**

**</div>**

**</form>**

**</div>**

**</div>**

**</div>**

**</body>**

**Bus Booking:**

**<body>**

**<nav class="logo navbar navbar-dark mb-4 bg-primary fixed-top ">**

**<div class="container">**

**<h2 class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i**

**class="fa-solid fa-bus"></i></span> Online Bus Registration System</h2>**

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

**<button type="submit" class="btn btn-info">Logout</button>**

**</form>**

**</div>**

**</nav>**

**<div class="container col-8 mt-5 p-5">**

**<div class="container row d-flex justify-content-center">**

**<div class="username mb-3 p-2">**

**<h4 class=" mt-4 d-flex justify-content-center">Bus Details</h4>**

**<div class=" table-responsible">**

**<table class="table table-bordered table-hover text-center align-middle">**

**<thead class="bg-primary tablehead">**

**<tr class="bg-primary">**

**<th>Bus Name</th>**

**<th>Bus Number</th>**

**<th>Departure Place</th>**

**<th>Departure Time</th>**

**<th>Destination Place</th>**

**<th>Time</th>**

**<th>Price</th>**

**<th>Action</th>**

**</tr>**

**</thead>**

**<tbody>**

**<tr th:each="bus:${buses}">**

**<td th:text="${bus.busName}"></td>**

**<td th:text="${bus.busNumber}"></td>**

**<td th:text="${bus.startingPoint}"></td>**

**<td th:text="${bus.startTime}"></td>**

**<td th:text="${bus.destinationPoint}"></td>**

**<td th:text="${bus.destinationTime}"></td>**

**<td th:text="${bus.price}"></td>**

**<td>**

**<a th:href="@{'/passenger/getBookingDetails/'+ ${bus.id} }"**

**class="bookbutton btn btn-success">Book</a>**

**</td>**

**</tr>**

**</tbody>**

**</table>**

**</div>**

**<div th:if="${errorMessage}" class="alert alert-warning text-center" role="alert">**

**<p th:text="${errorMessage}"></p>**

**</div>**

**</div>**

**</div>**

**</div>**

**</body>**

**Ticket Related Pages**

**Booking page:**

**<body>**

**<nav class="logo navbar navbar-dark mb-4 bg-primary fixed-top ">**

**<div class="container">**

**<h2 class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i**

**class="fa-solid fa-bus"></i></span> Online Bus Registration System</h2>**

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

**<button type="submit" class="btn btn-danger">Logout</button>**

**</form>**

**</div>**

**</nav>**

**<div class="container col-8 mt-5 p-5">**

**<div class="container row d-flex justify-content-center">**

**<div class="username mb-3 p-2">**

**<div class="registerform shadow">**

**<h2 class="d-flex justify-content-center">Book Bus Ticket</h2>**

**<div class="container row d-flex justify-content-center">**

**<h3>Bus Details</h3>**

**<p><b>Bus Name: </b> <span th:text="${bus.busName}"></span></p>**

**<p><b>Bus Number: </b> <span th:text="${bus.busNumber}"></span></p>**

**<p><b>Departure Place: </b> <span th:text="${bus.startingPoint}"></span></p>**

**<p><b>Departure Time:</b> <span th:text="${bus.startTime}"></span></p>**

**<p><b>Destination Place: </b> <span th:text="${bus.destinationPoint}"></span></p>**

**<p><b>Arrival Time:</b> <span th:text="${bus.destinationTime}"></span></p>**

**<p><b>Price:</b> <span th:text="${bus.price}"></span></p>**

**</div>**

**<div class="container row d-flex justify-content-center">**

**<form th:action="@{/passenger/saveTheBooking}" th:object="${ticket}" method="post">**

**<input type="hidden" th:field="\*{busesId}"/>**

**<input type="hidden" th:field="\*{busProviderId}"/>**

**<label for="BookedDate" class="fs-4"><b>Travel Date:</b></label>**

**<input type="date" th:field="\*{BookedDate}" id="BookedDate" required/>**

**<br>**

**<button type="submit" class=" btn btn-success mb-3">Book Ticket</button>**

**</form>**

**</div>**

**</div>**

**</div>**

**</div>**

**</div>**

**</body>**

**Booked Ticket:**

**<body>**

**<nav class=" navbar navbar-dark mb-4 bg-primary fixed-top ">**

**<div class="container">**

**<h2 class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i**

**class="fa-solid fa-bus"></i></span> Online Bus Registration System</h2>**

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

**<button type="submit" class="btn btn-danger">Logout</button>**

**</form>**

**</div>**

**</nav>**

**<div class=" container col-10 mt-5 p-5">**

**<div class="container row d-flex justify-content-center">**

**<h4 class="d-flex justify-content-center">Booked Ticket</h4>**

**<div class="table-responsive">**

**<table class="table table-bordered table-hover text-center align-middle">**

**<thead class="bg-primary tablehead">**

**<tr class="bg-primary">**

**<th>Bus Company</th>**

**<th>Bus Name</th>**

**<th>Bus Number</th>**

**<th>Date</th>**

**<th>Departure Place</th>**

**<th>Departure Time</th>**

**<th>Destination Place</th>**

**<th>Destination Time</th>**

**<th>Price</th>**

**<th>Action</th>**

**</tr>**

**</thead>**

**<tbody>**

**<!-- <tr th:each="tic : ${tickets}">-->**

**<tr th:each="ticket : ${tickets}">**

**<td th:text="${ticket.busProvider.name}"></td>**

**<td th:text="${ticket.buses.busName} "></td>**

**<td th:text="${ticket.buses.busNumber} "></td>**

**<td th:text="${ticket.BookedDate}"></td>**

**<td th:text="${ticket.buses.startingPoint} "></td>**

**<td th:text="${ticket.buses.startTime}"></td>**

**<td th:text="${ticket.buses.destinationPoint} "></td>**

**<td th:text="${ticket.buses.destinationTime}"></td>**

**<td th:text="${ticket.buses.price}"></td>**

**<td>**

**<a th:href="@{/passenger/delete/{id}(id=${ticket.id})}" class="Deletebutton btn btn-dark">Delete</a>**

**</td>**

**</tr>**

**<tr th:if="${ticket == null}">**

**<td colspan="10" class="text-center">No ticket found.</td>**

**</tr>**

**</tbody>**

**</table>**

**</div>**

**</div>**

**</div>**

**</body>**

**Ticket History:**

**<body>**

**<nav class="navbar navbar-dark mb-4 bg-primary fixed-top ">**

**<div class="container">**

**<h2 class="text-white mb-0 navbar-brand"><span style="font-size:30px;color:yellow;"><i**

**class="fa-solid fa-bus"></i></span> Online Bus Registration System</h2>**

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

**<button type="submit" class="btn btn-danger">Logout</button>**

**</form>**

**</div>**

**</nav>**

**<div class=" container col-10 mt-5 p-5 ">**

**<div class="container row d-flex justify-content-center">**

**<h4 class="d-flex justify-content-center mb-4">History</h4>**

**<div class="table-responsive">**

**<table class="table table-bordered table-hover text-center align-middle">**

**<thead class="bg-primary tablehead">**

**<tr class="bg-primary">**

**<th>Bus Company</th>**

**<th>Bus Name</th>**

**<th>Bus Number</th>**

**<th>Date</th>**

**<th>Departure Place</th>**

**<th>Departure Time</th>**

**<th>Destination Place</th>**

**<th>Destination Time</th>**

**<th>Price</th>**

**</tr>**

**</thead>**

**<tbody>**

**<tr th:each="tic : ${tickets}">**

**<td th:text="${tic.busProvider.name}"></td>**

**<td th:text="${tic.buses.busName} "></td>**

**<td th:text="${tic.buses.busNumber} "></td>**

**<td th:text="${tic.BookedDate}"></td>**

**<td th:text="${tic.buses.startingPoint} "></td>**

**<td th:text="${tic.buses.startTime}"></td>**

**<td th:text="${tic.buses.destinationPoint} "></td>**

**<td th:text="${tic.buses.destinationTime}"></td>**

**<td th:text="${tic.buses.price}"></td>**

**</tr>**

**</tbody>**

**</table>**

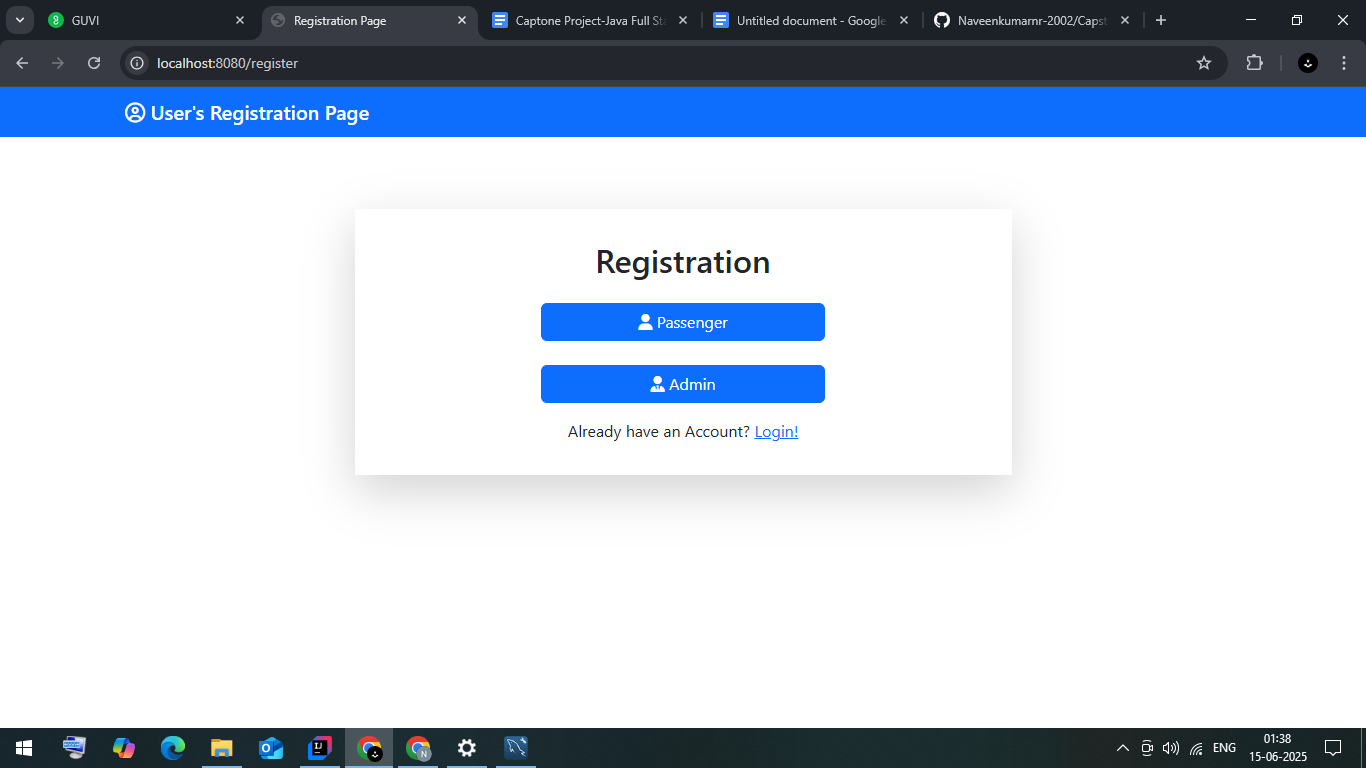
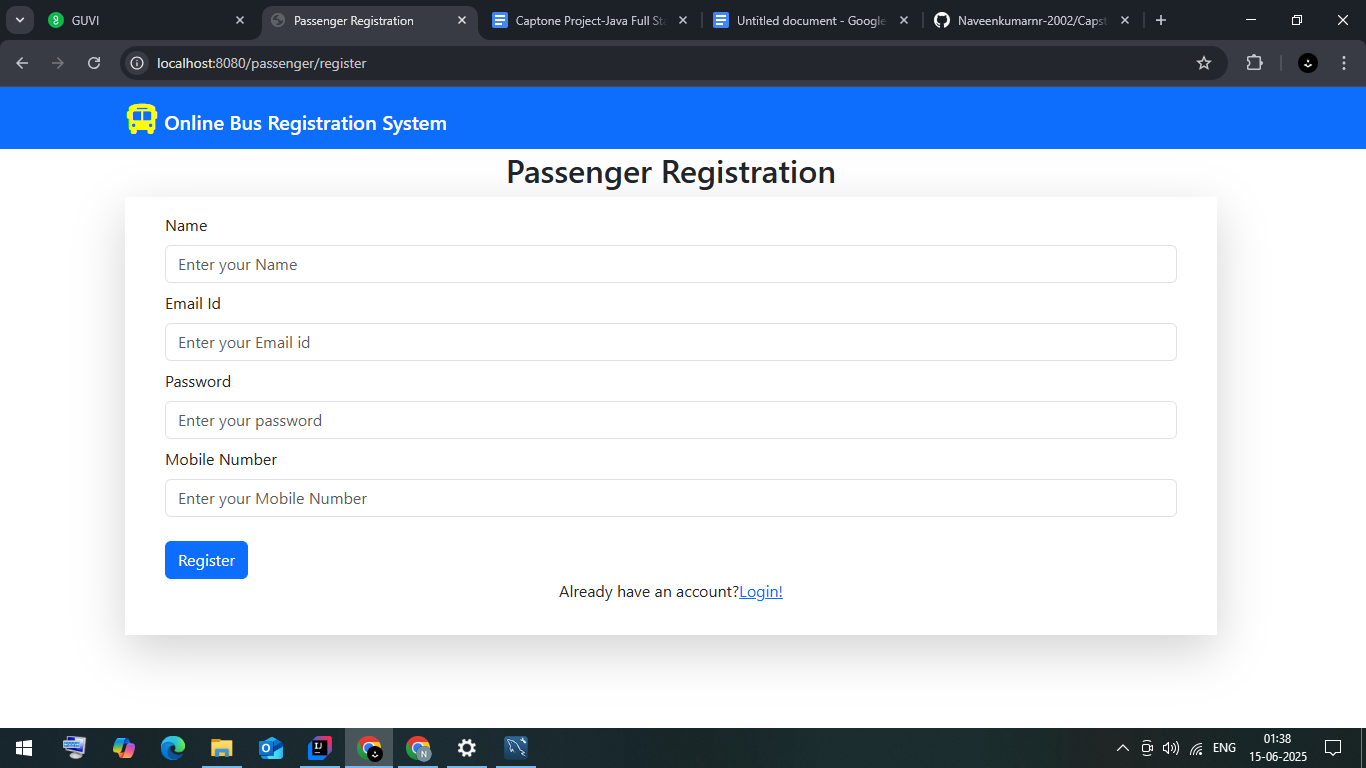
**</div>**

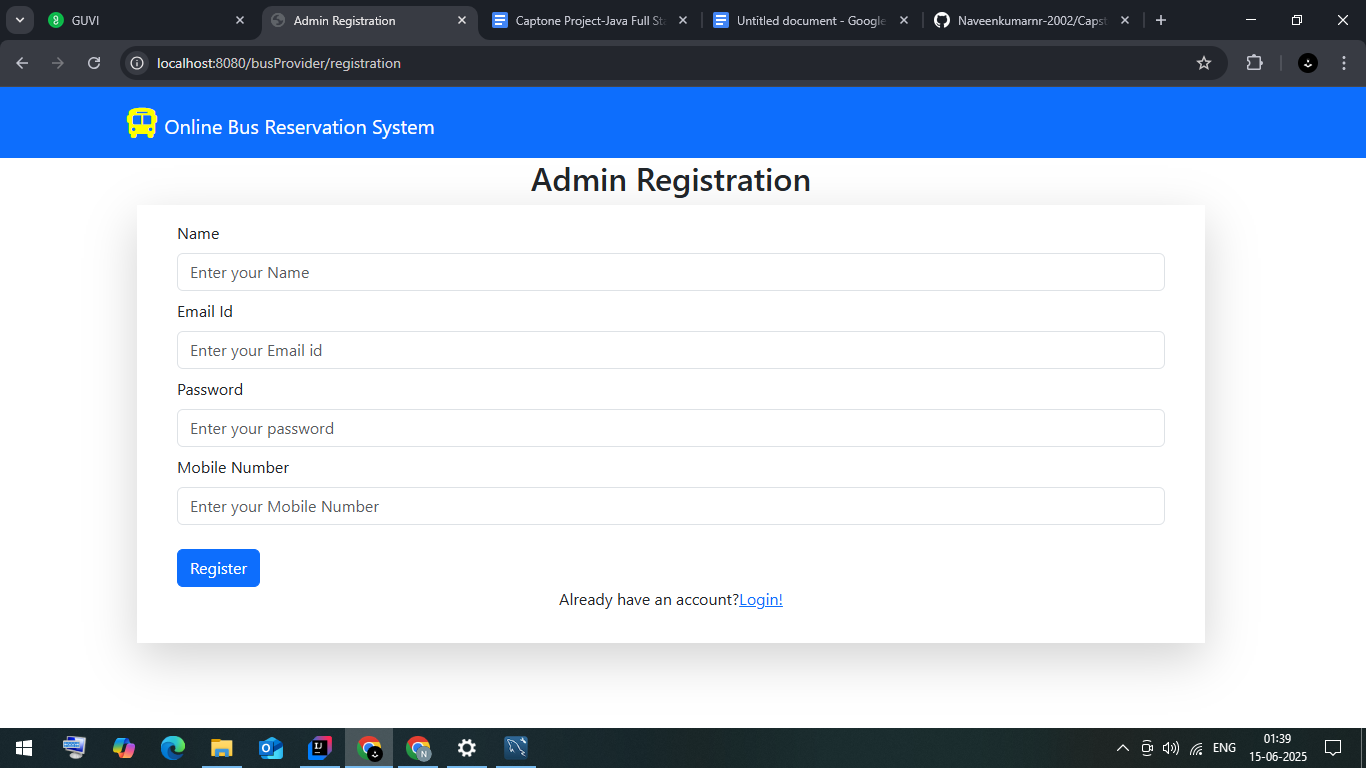
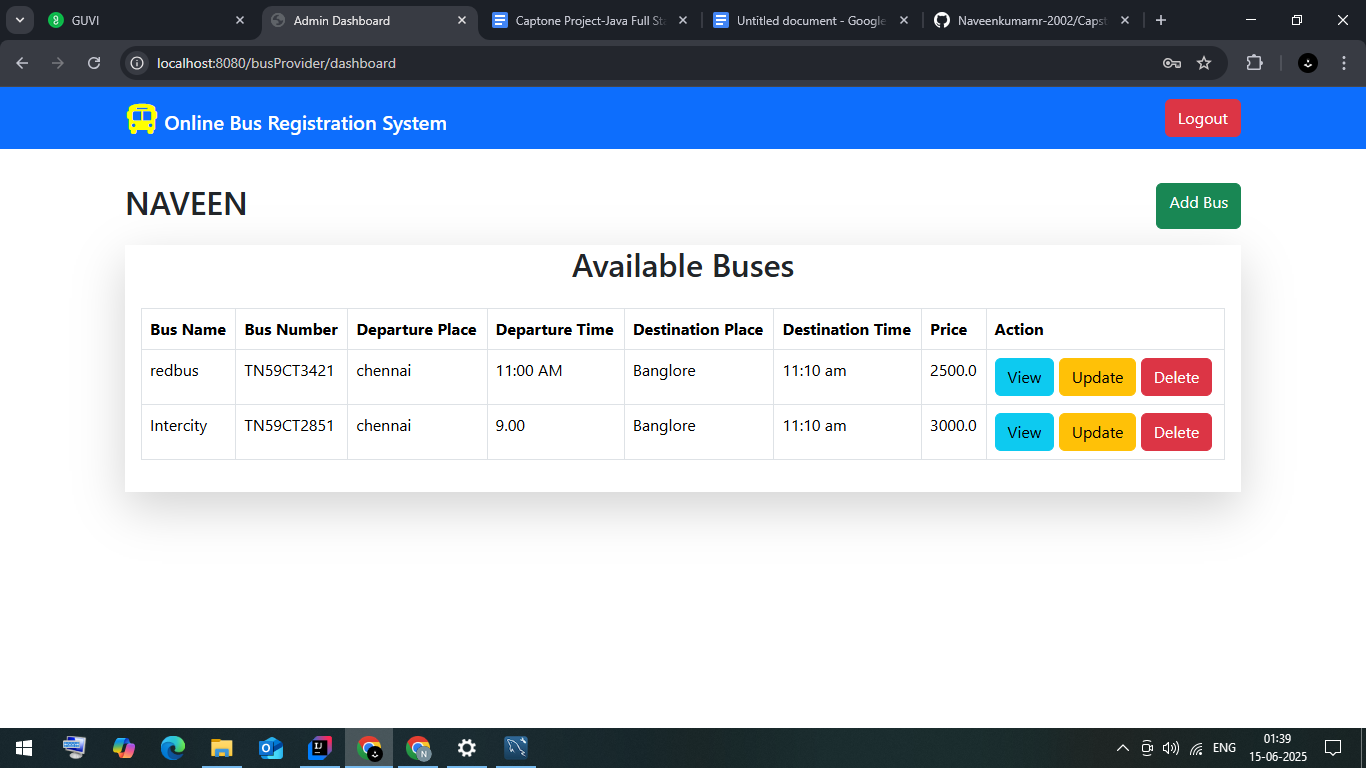
**</div>**

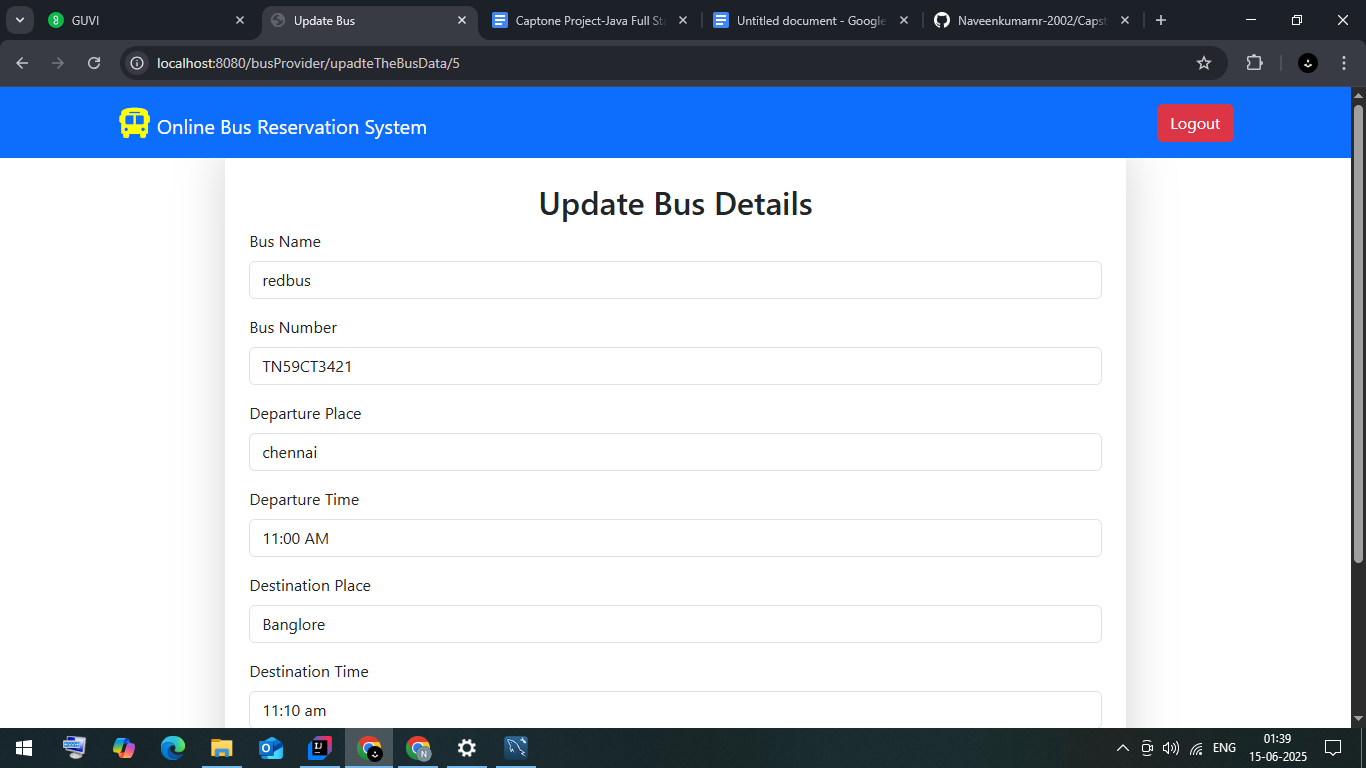
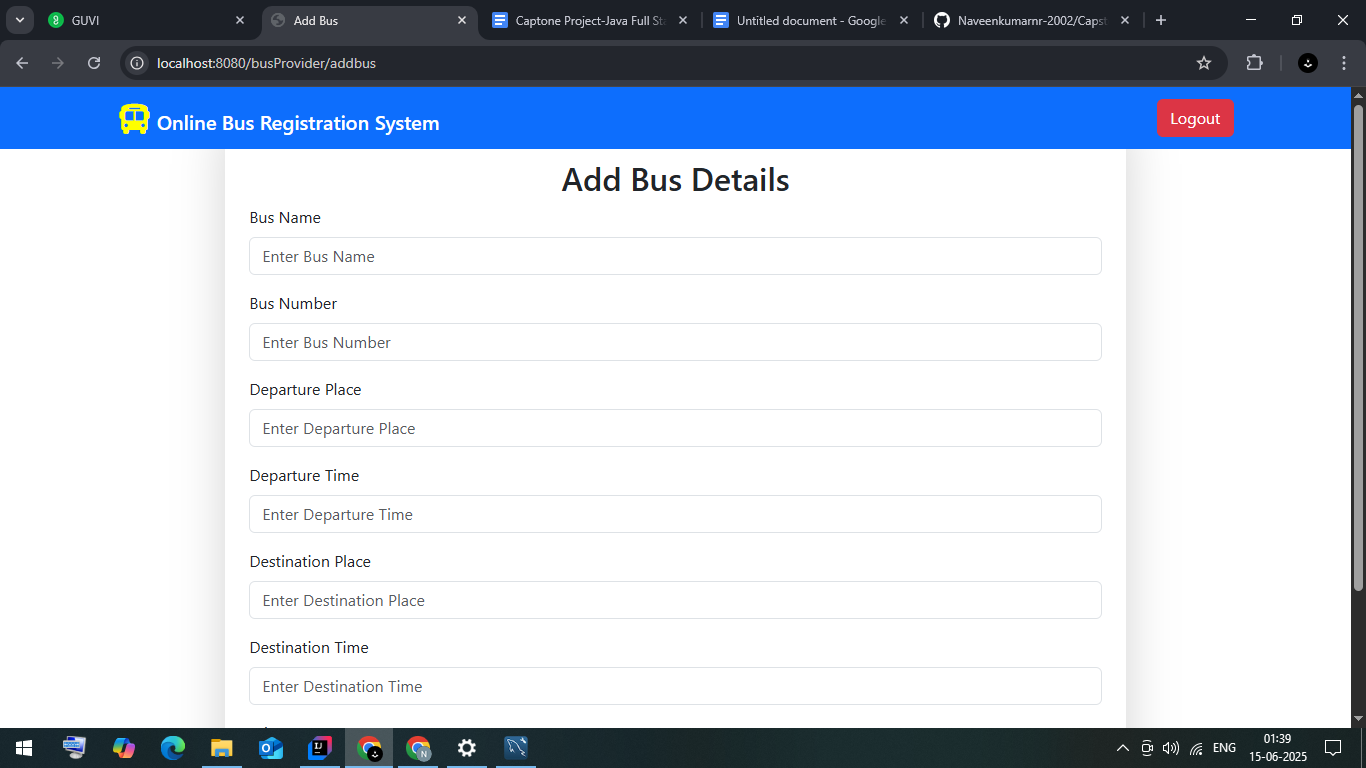
**</div>**

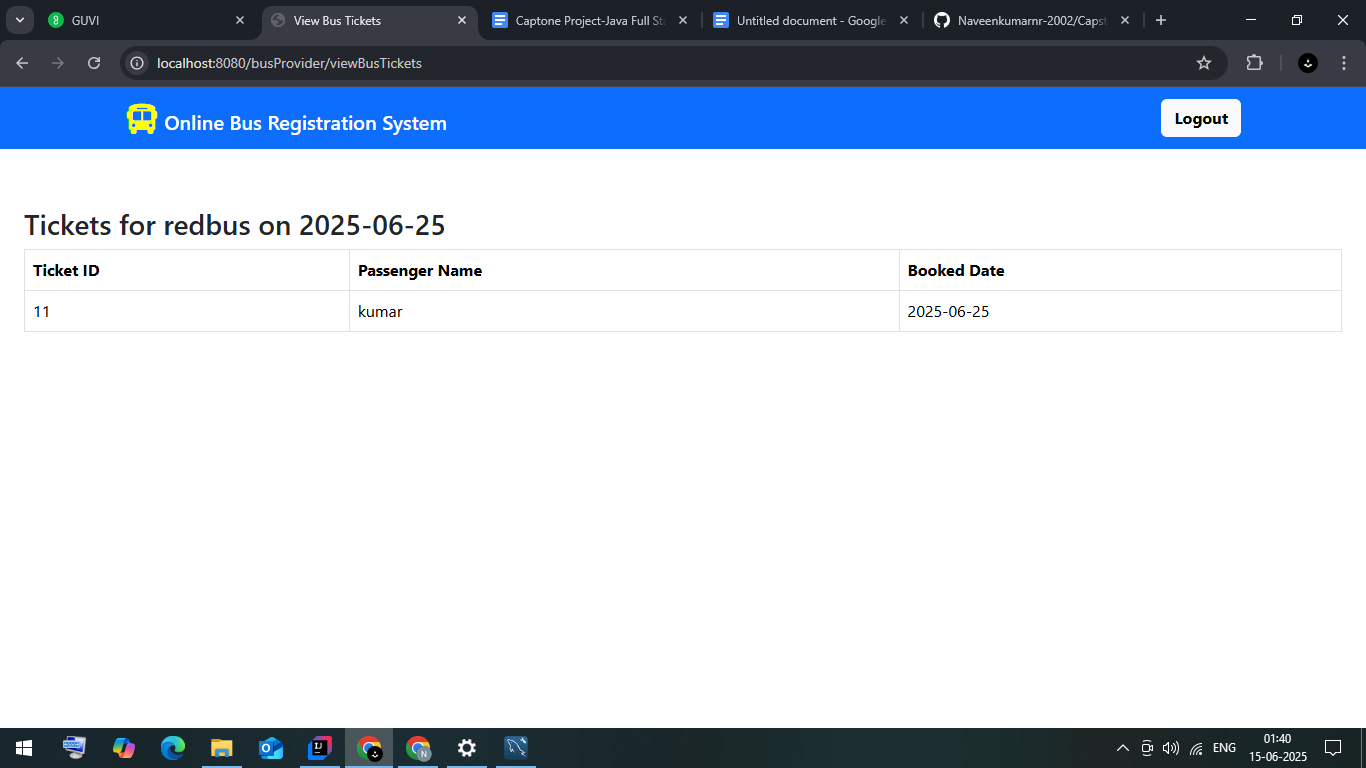
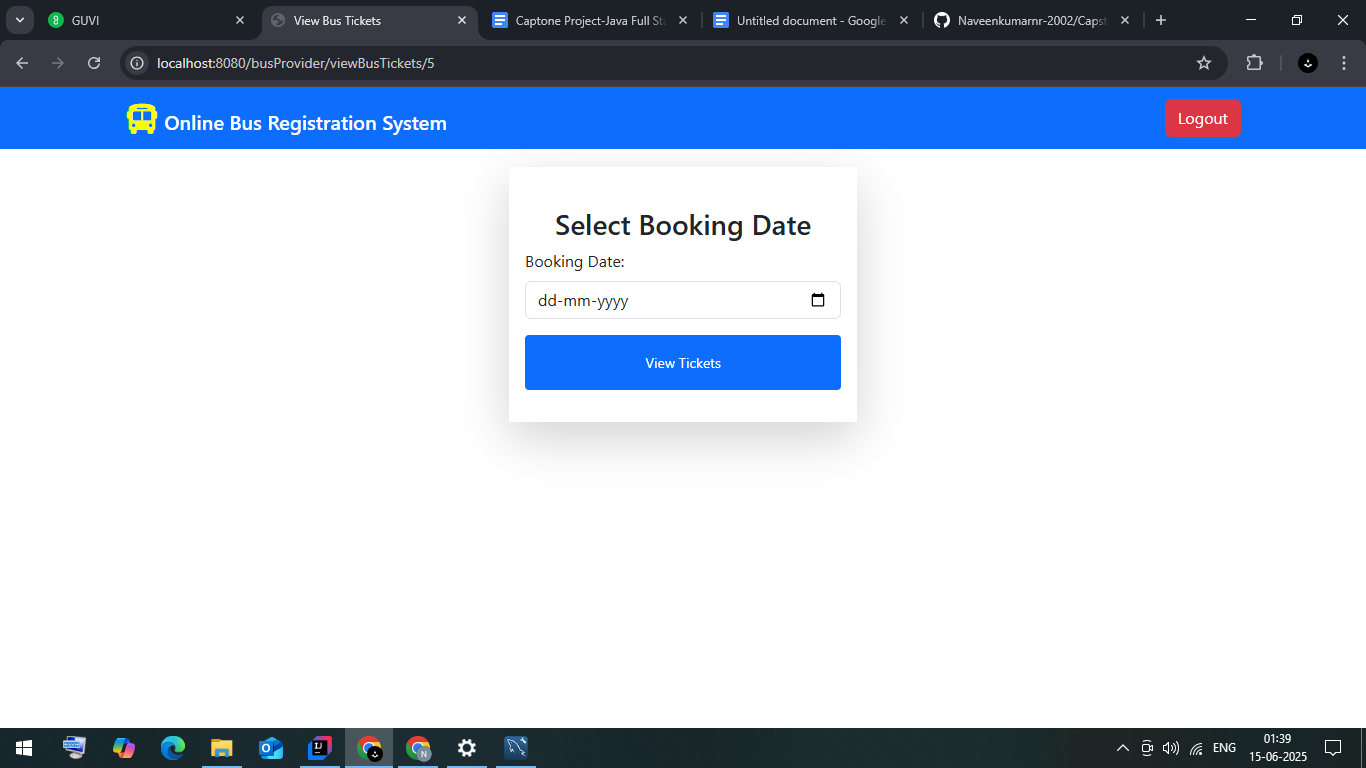
**</body>**

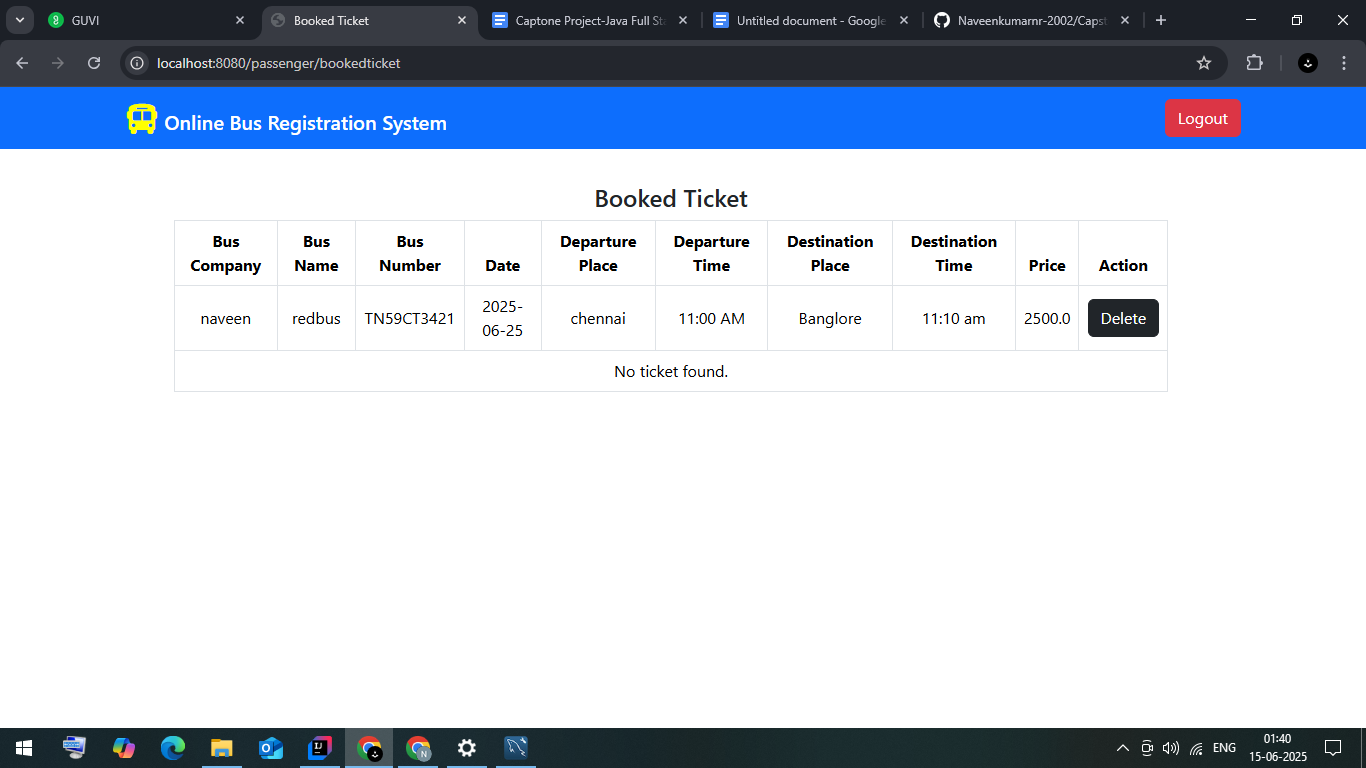
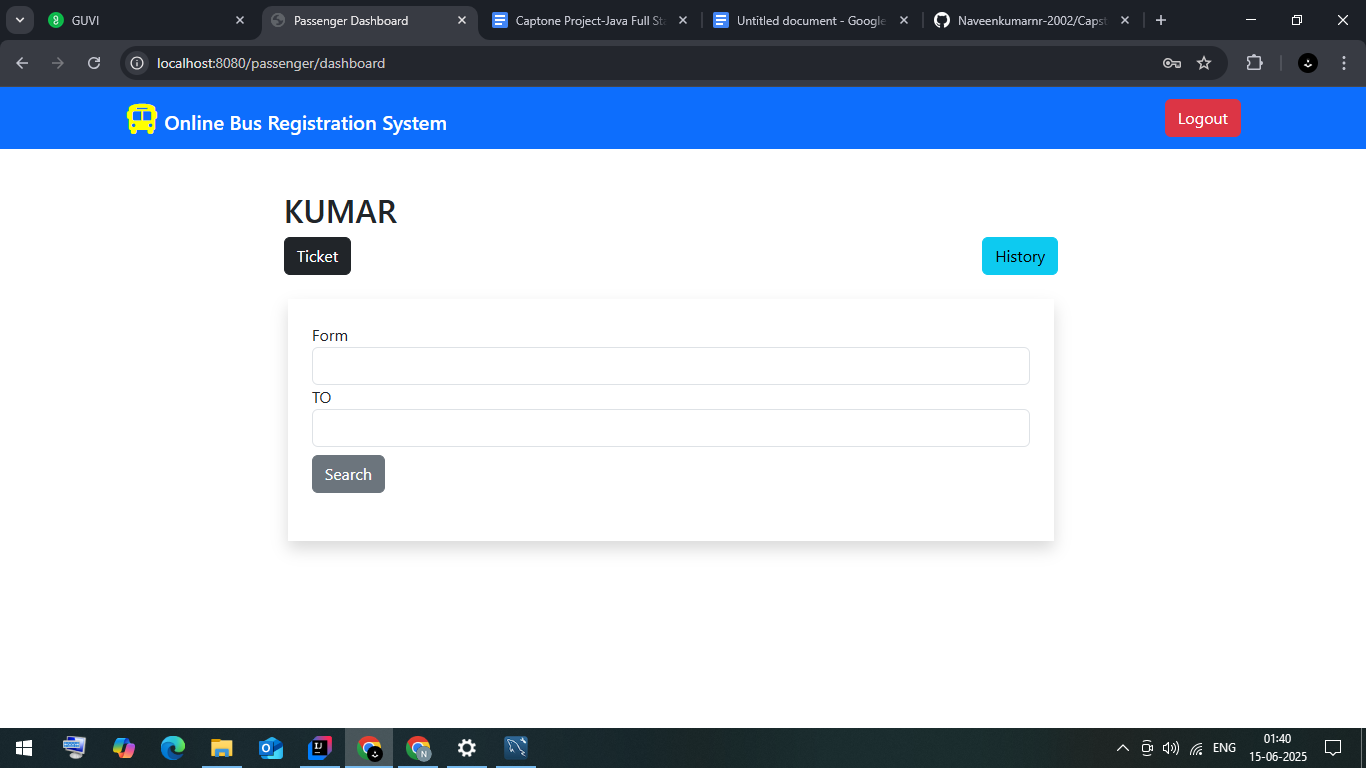
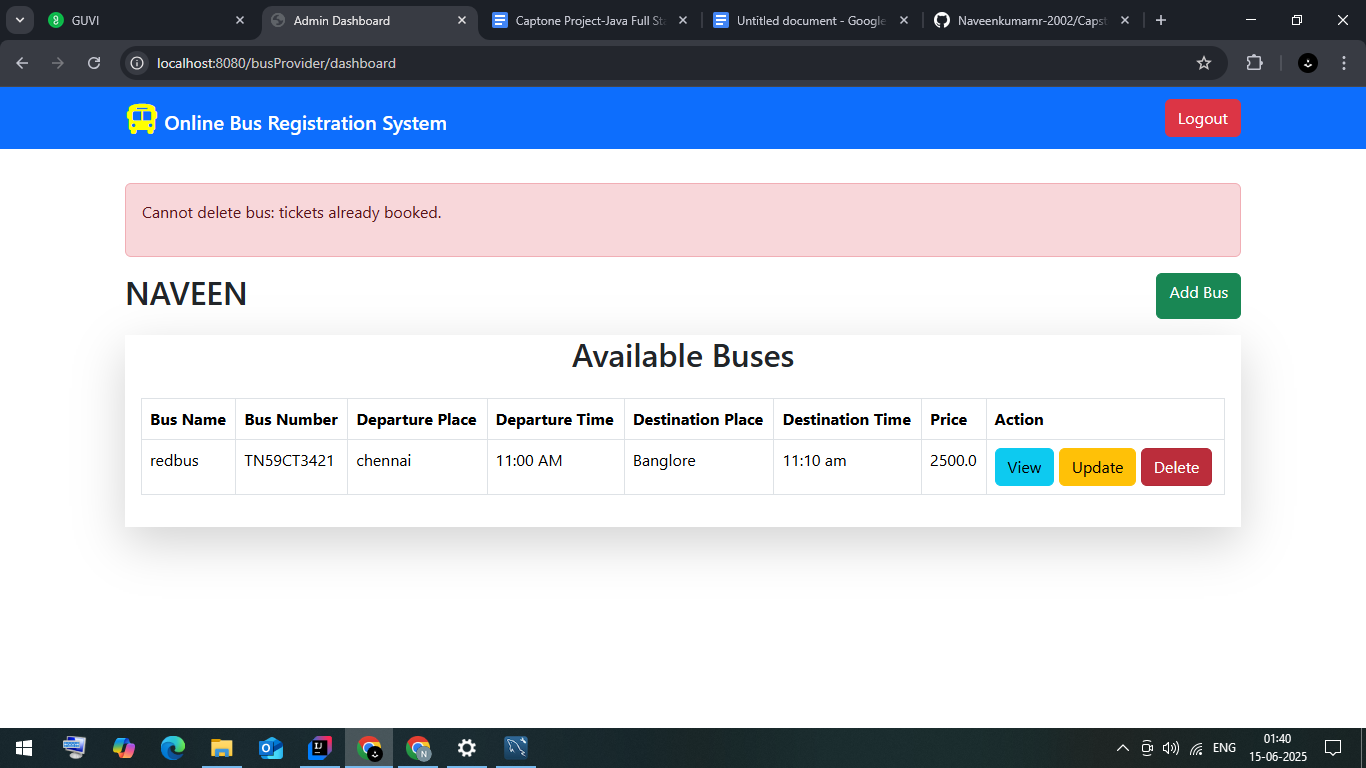
**Outputs**

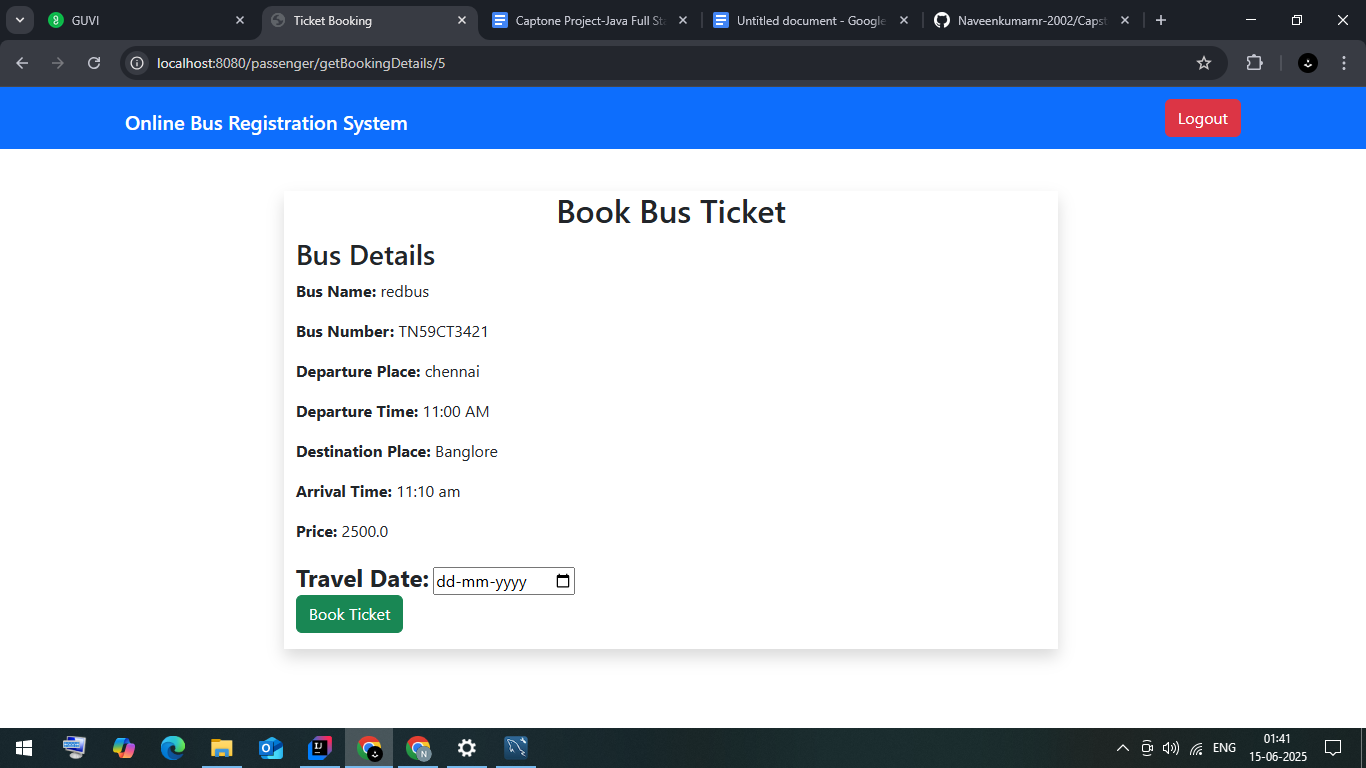
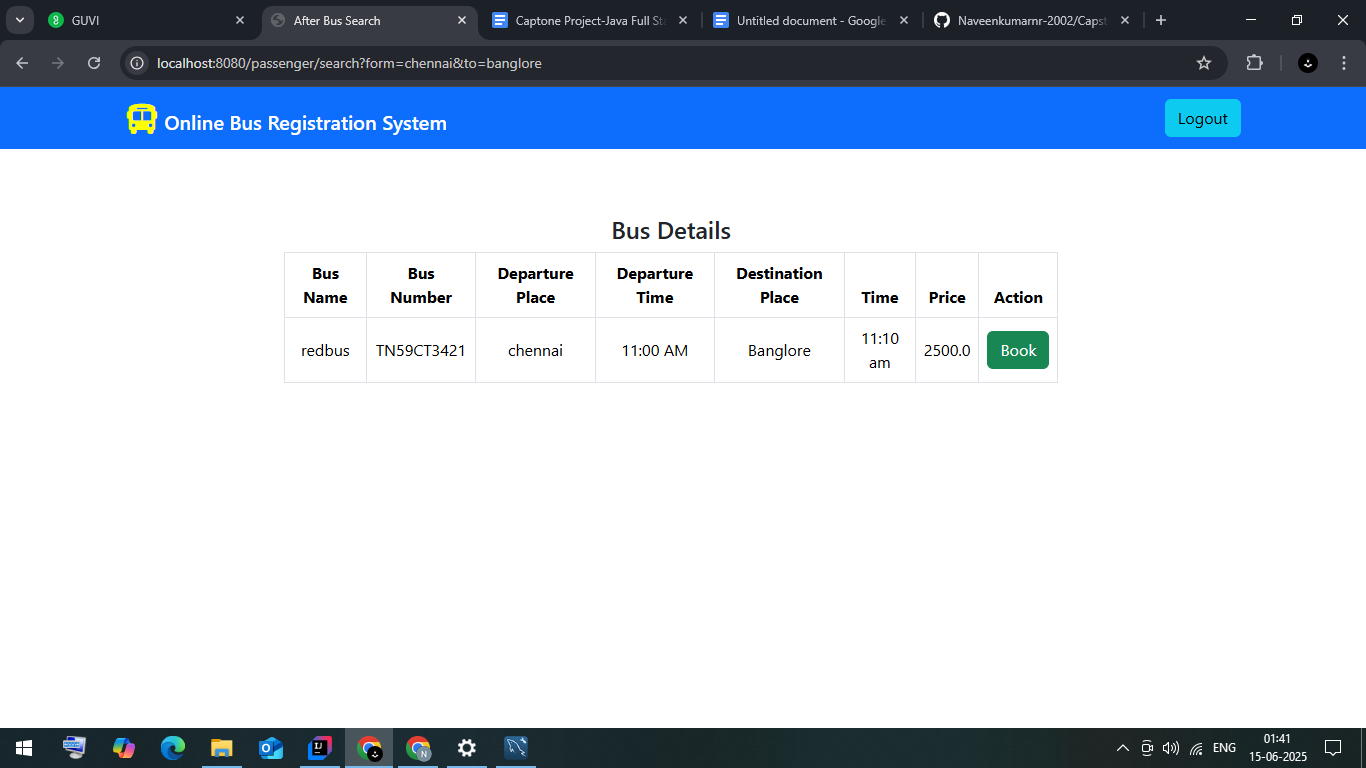
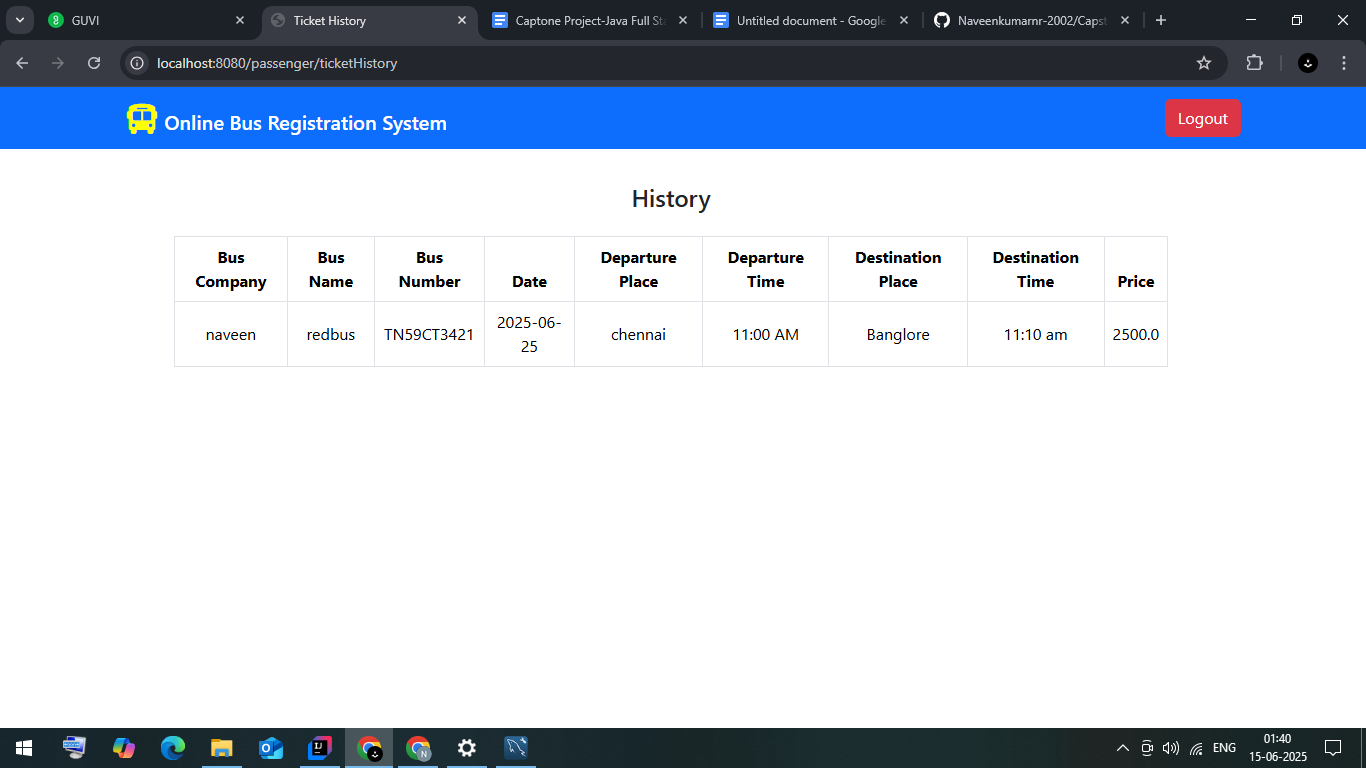
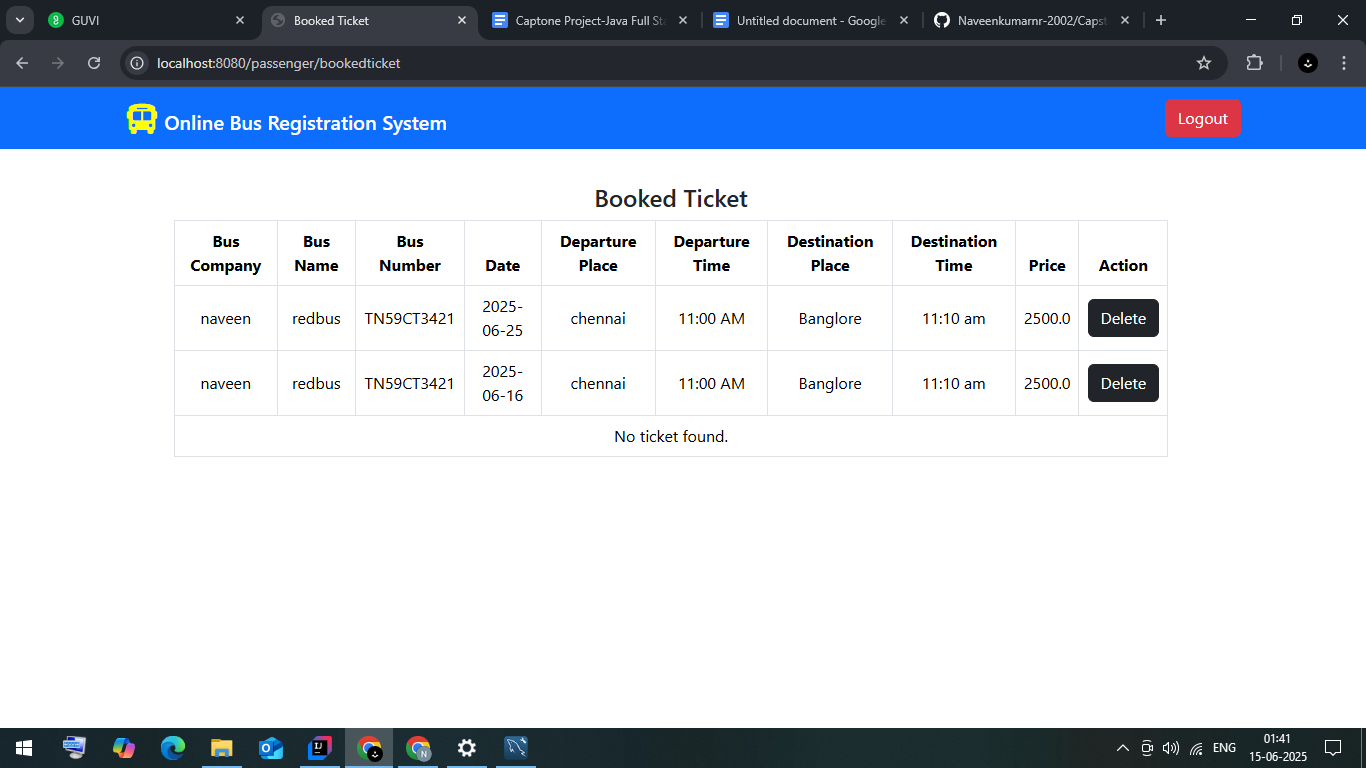
****

****

****

****

****

****