



ONLINE BUS BOOKING SYSTEM

PRESENTED BY GROUP :2



Biswojeet Behera

ID(2567240)

Garapati Keertana

ID(2568863)

Gaurav Kumar

ID(2567859)

Kiran Raj V

ID(2567416)

Kommu Mahesh

ID(2569143)

Navya Manju M

ID(2567544)

Nookanaboyena Karthik

ID(2569105)

Obili Govindu Gari Manmadha Reddy

ID(2566995)

TABLE OF CONTENTS

1. INTRODUCTION

2.TOOLS AND TECHNOLOGIES

3.SCOPE OF THE PROJECT

4.DATABASE SCHEMA

5. SCREENSHOTS

6. POSTMAN TESTING

7.JUNIT

8.CONCLUSION



The Online Bus Booking System is a web-based platform that revolutionizes the way people book and manage their bus travel. It provides a seamless and convenient experience for both travelers and bus operators. With this system, users can effortlessly search for available routes, select preferred seats , book the seats. This eliminates the need for manual ticketing and long queues, the Online Bus Booking System aims to streamline bus travel, making it more accessible and efficient for everyone.

The Scope of the online bus booking system is:

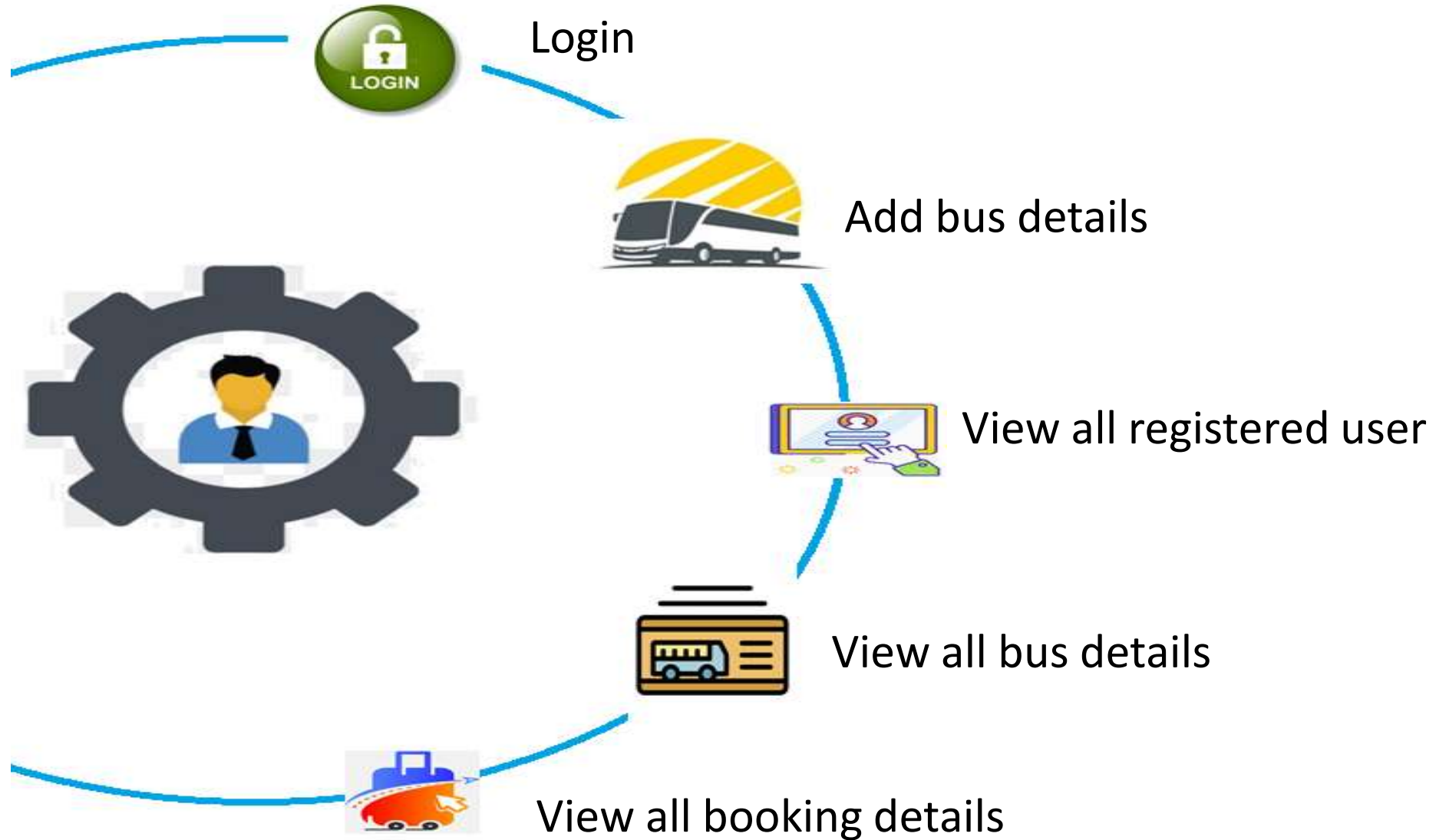
A user should be able to

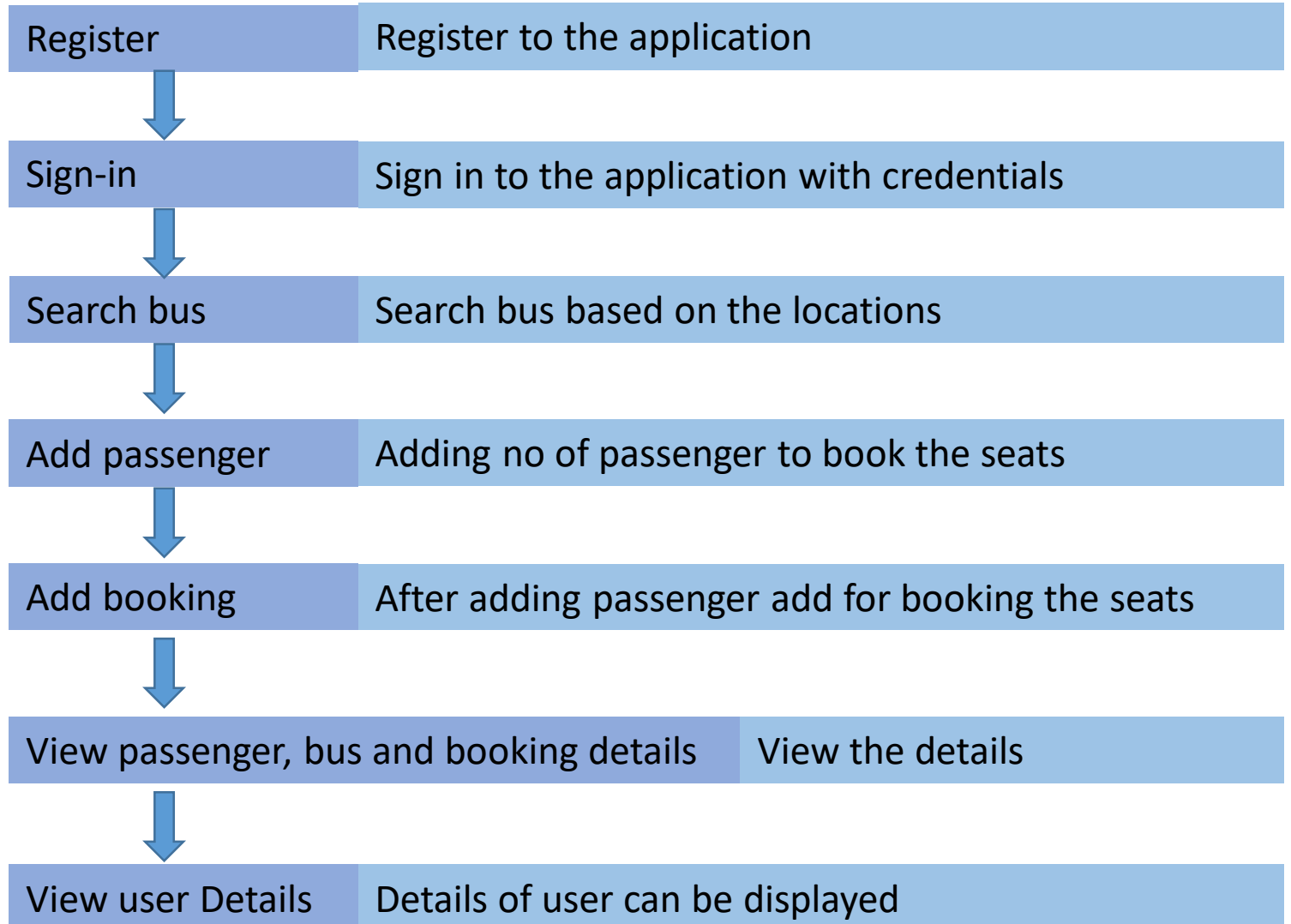
- Login to the system through the first page of the application.
- User can update the details along with the password after logging into the system.
- User can be able to create a new login for the accessing the reservation facility.
- Able to choose the seats which are available.



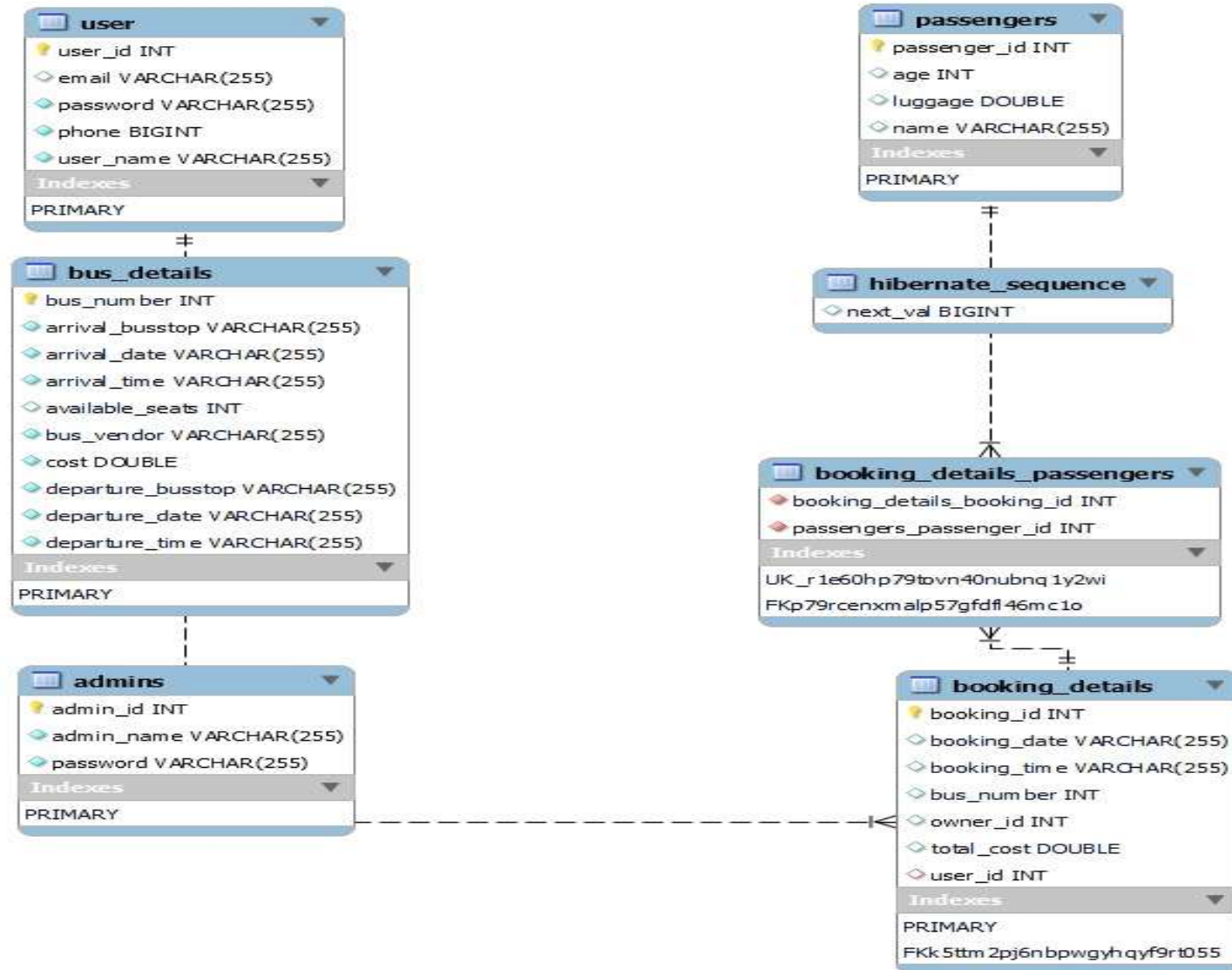
- Front End: Angular
- Server-Side: Spring Boot
- Backend: MySQL , Hibernate
- Server: Tomcat
- Operating System: Windows 10/11
- Developing Method: Agile Methodology

ADMIN ROLE

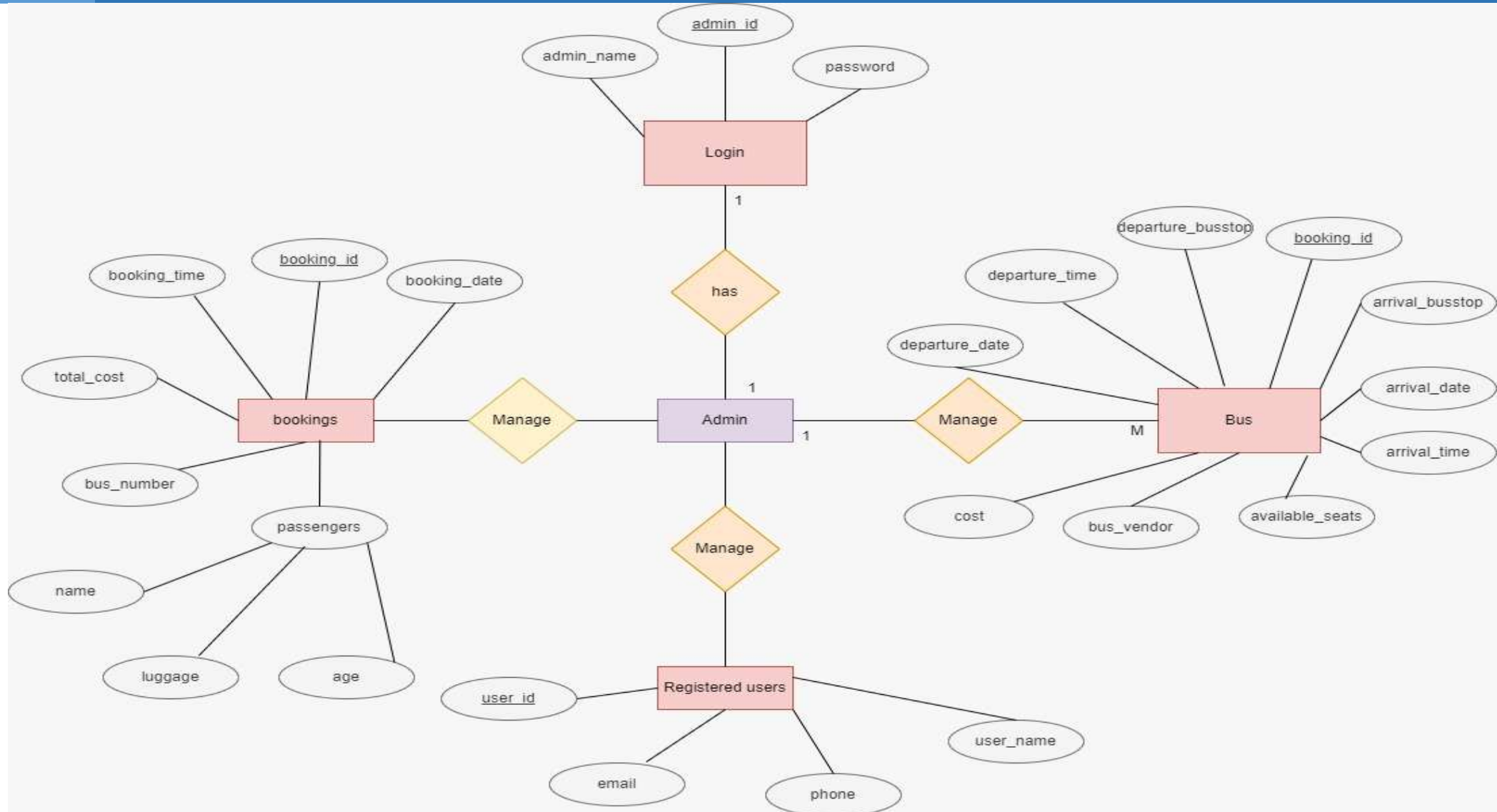




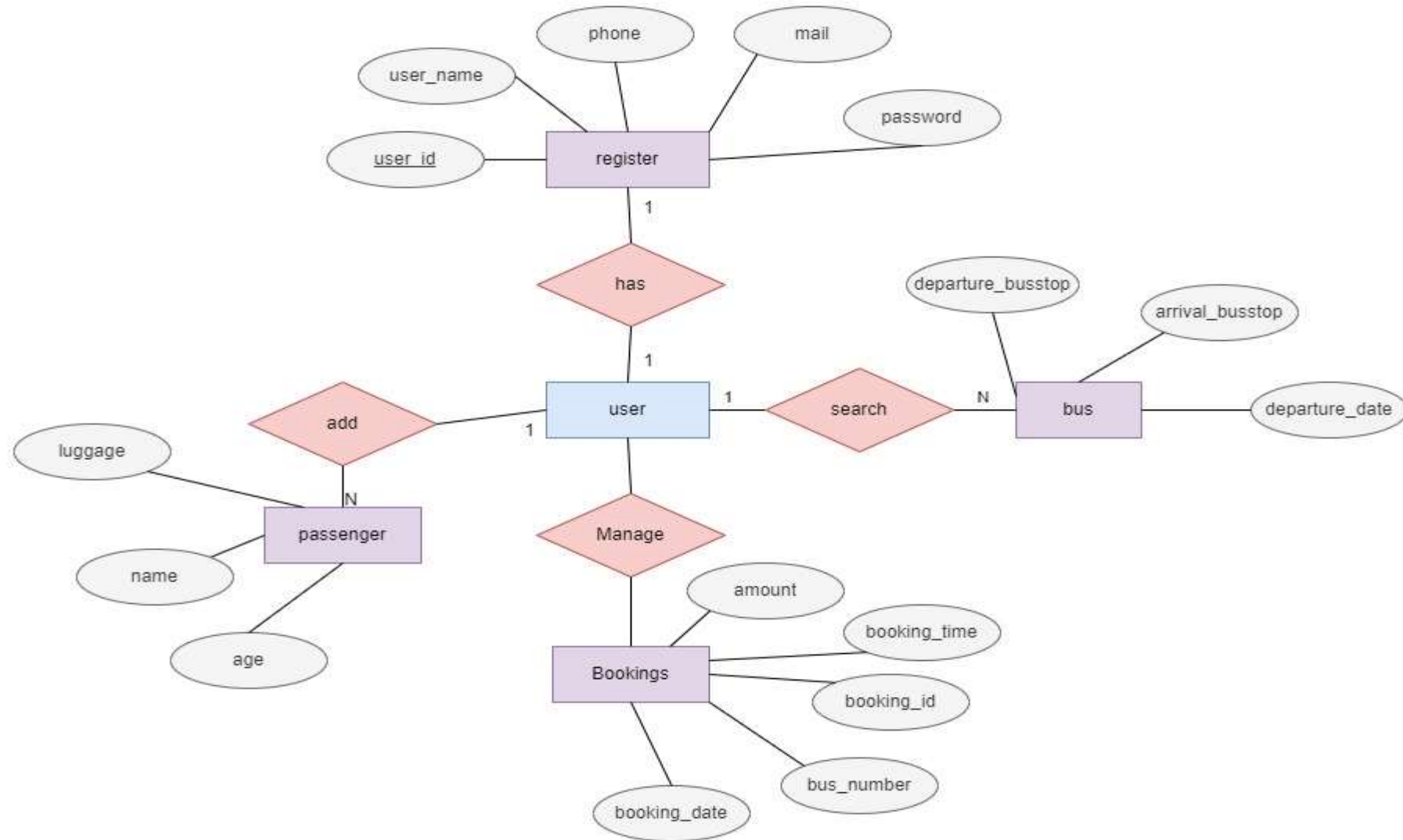
CLASS DIAGRAM



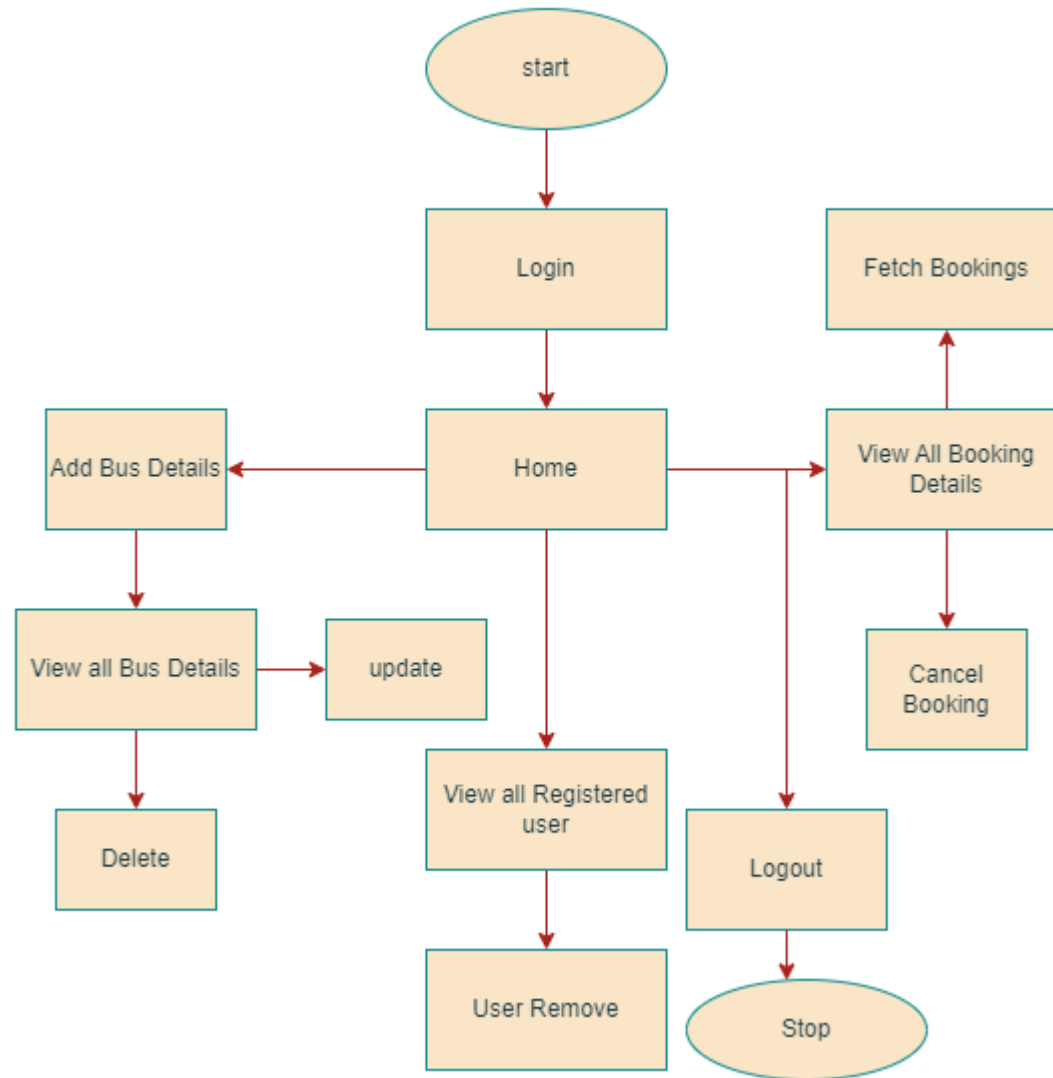
ER DIAGRAM OF ADMIN



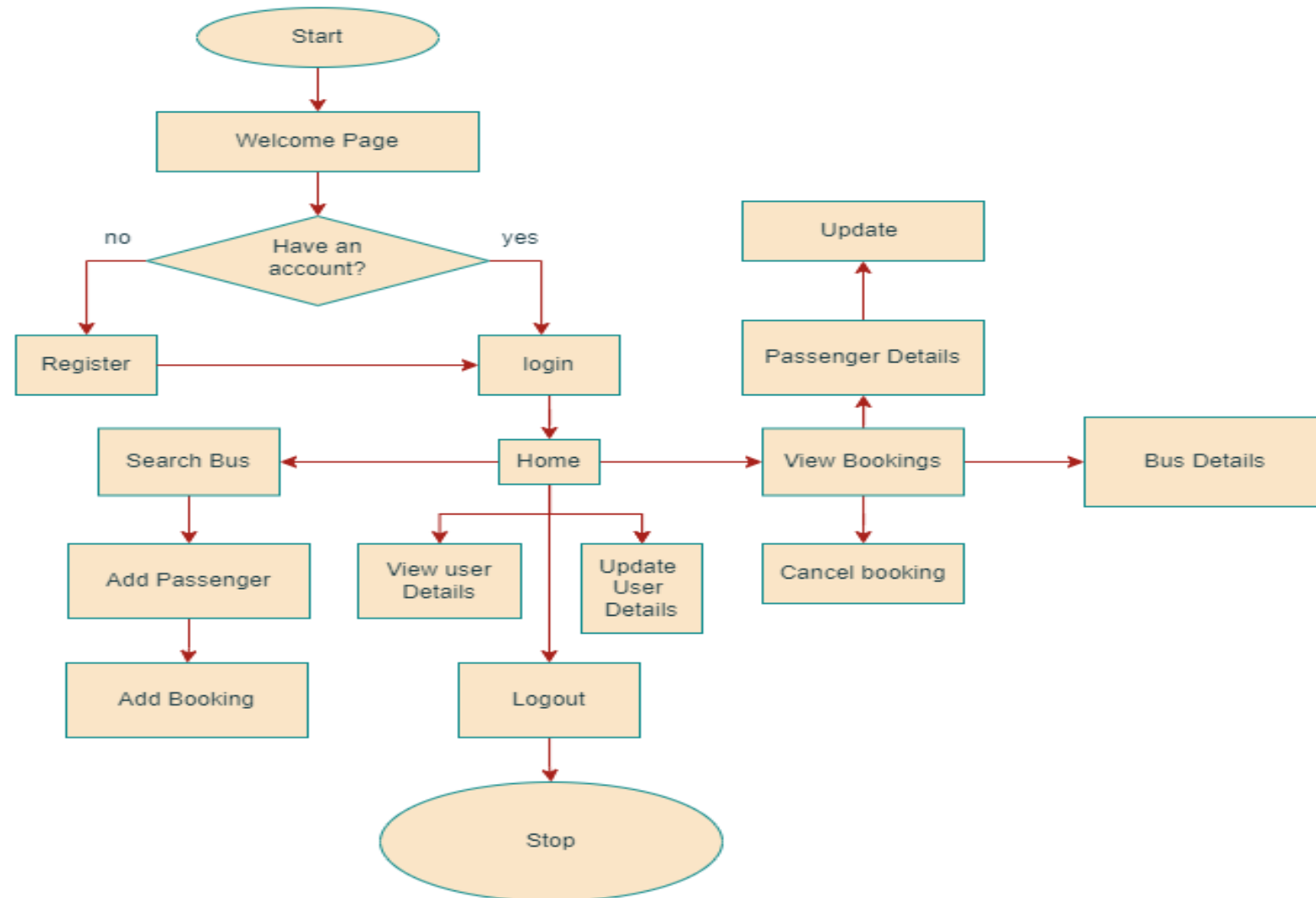
ER DIAGRAM OF USER

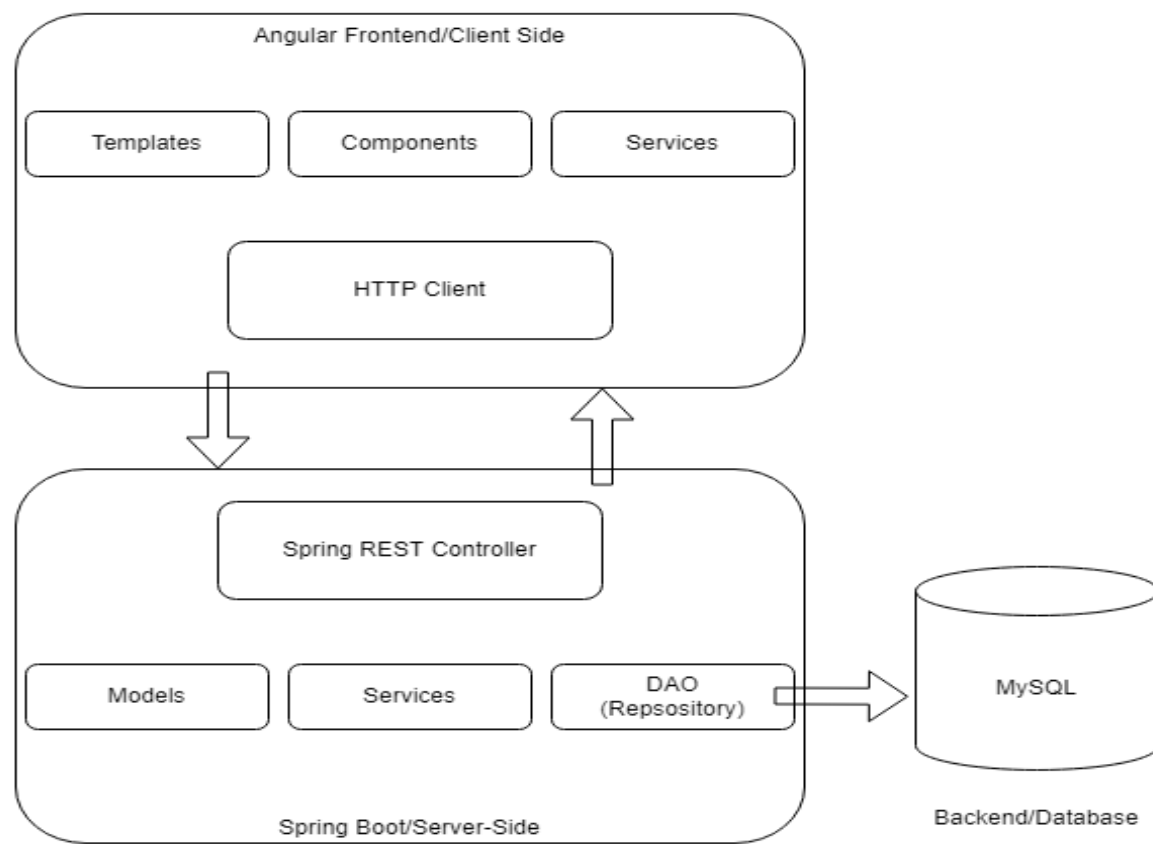


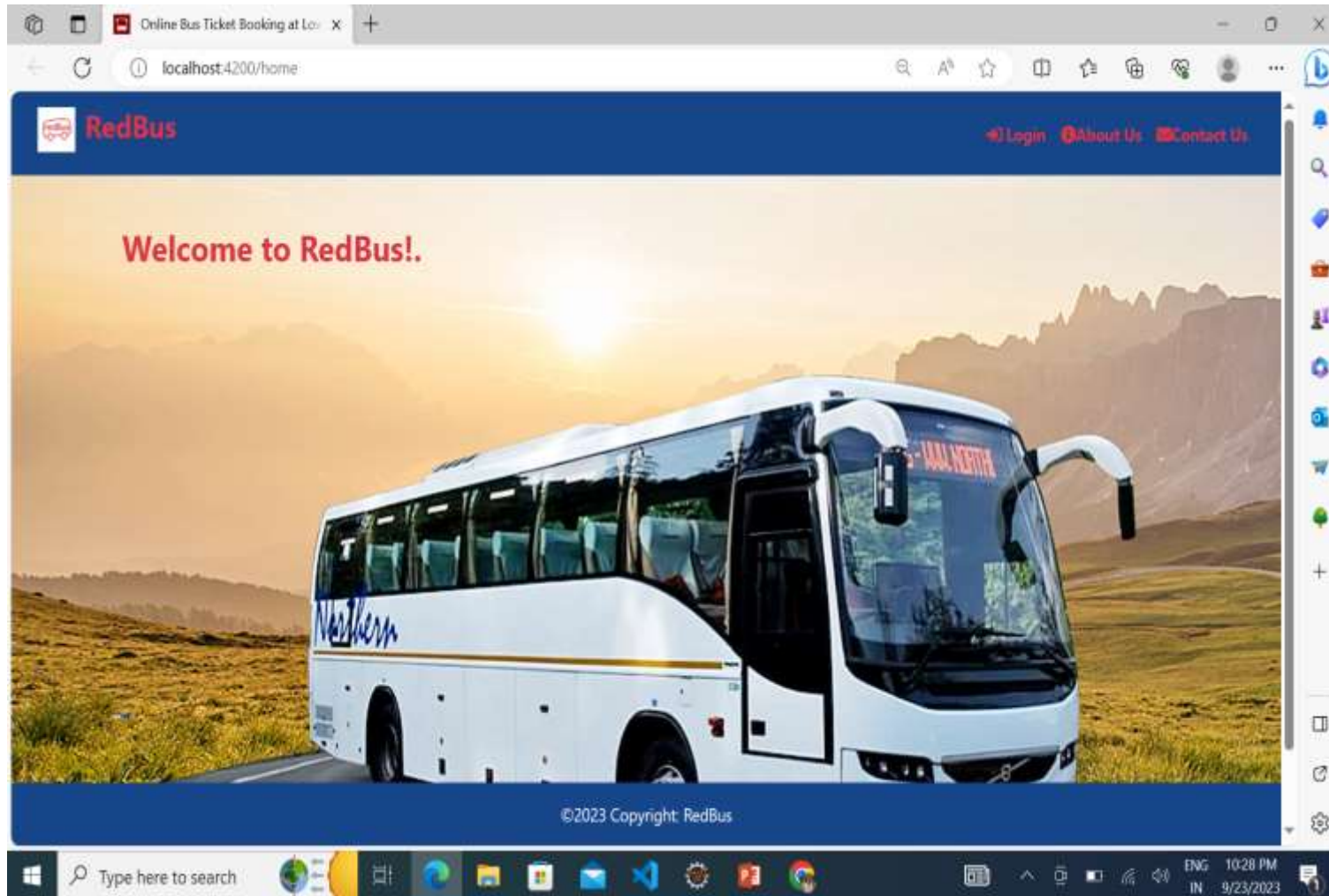
ADMIN FLOW CHART



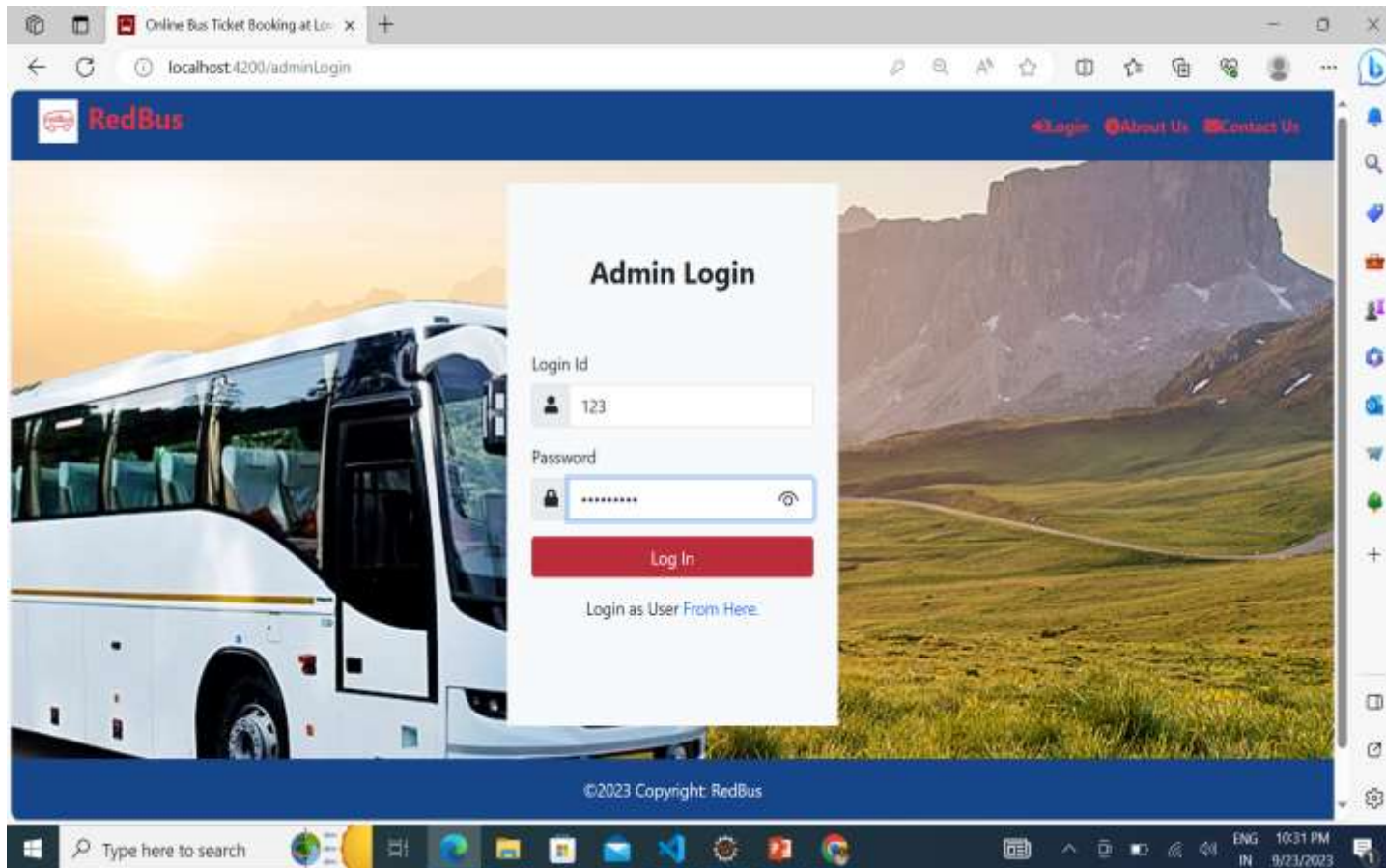
USER FLOW CHART





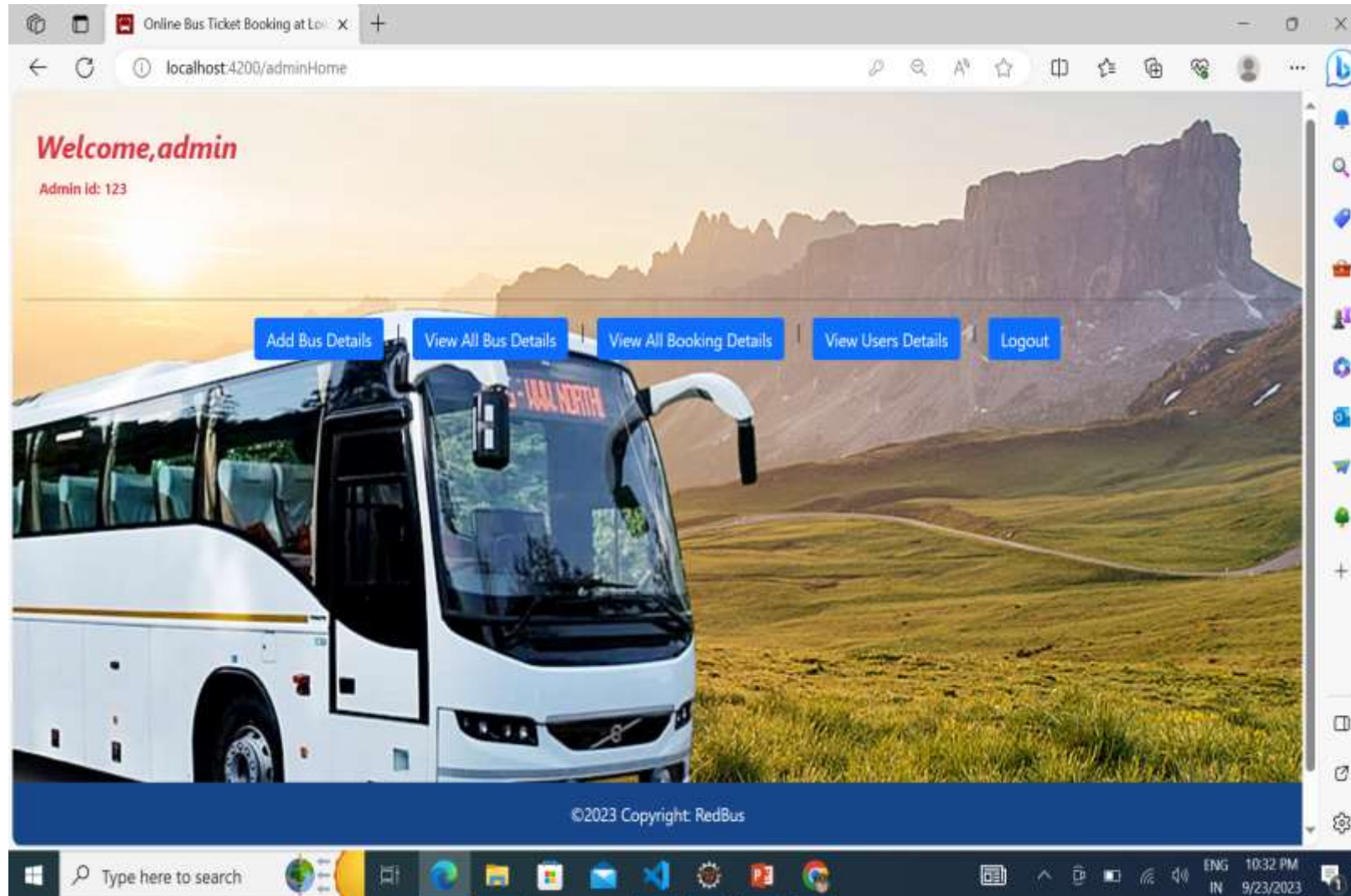


- This is the first module of this application where user has Login ,about us and contact us details.

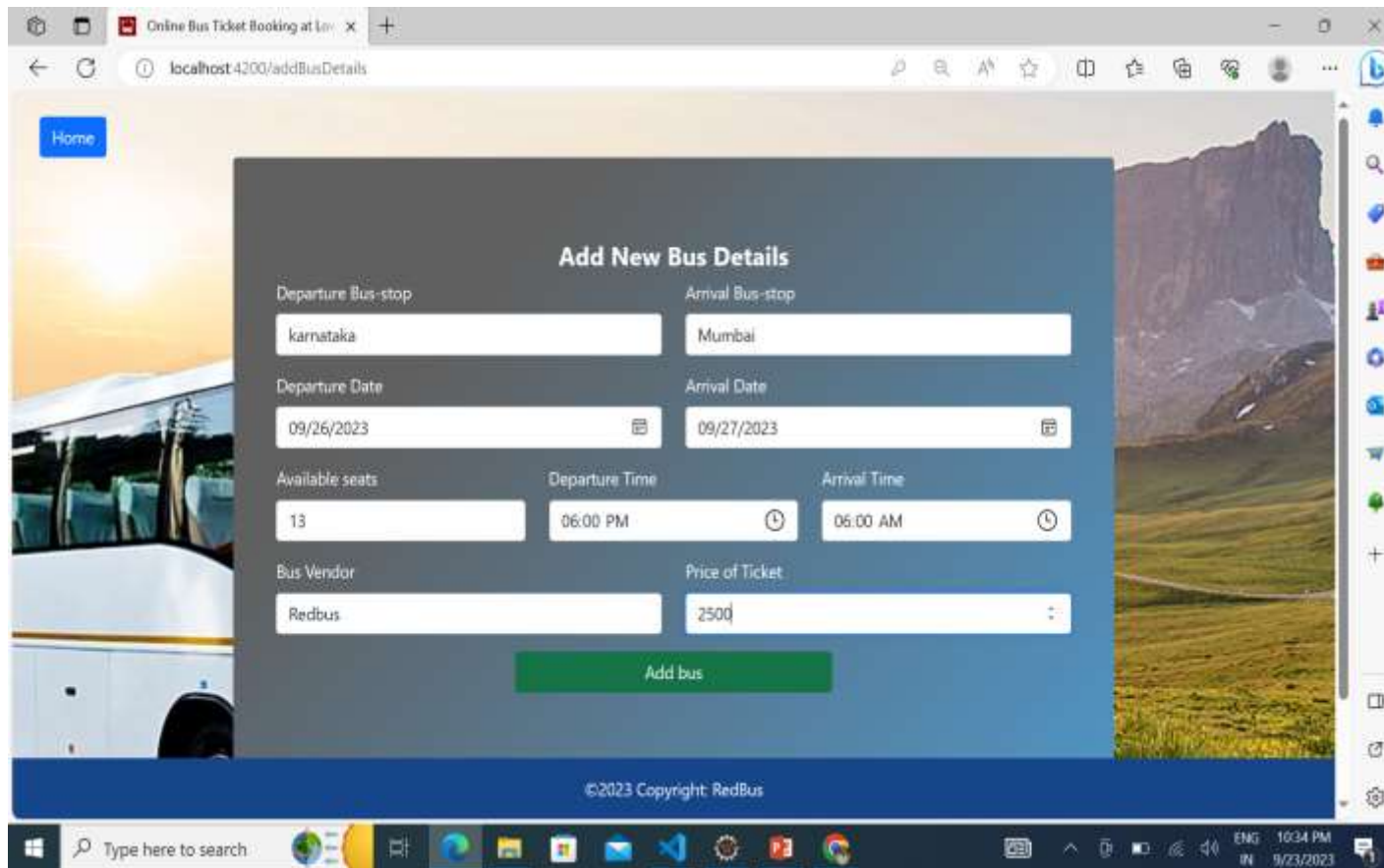


Admin Login bus booking:

- This booking system is a module in this project, a admin want to enter into this software project by entering his id and password.
- He has the complete control over the application.



- This module deals with addition and deletion of buses and deletion of user and booking details
- Can view the user details ,booking details and bus details.
- Only admin can perform these functionalities.



The screenshot shows a web browser window with the title 'Online Bus Ticket Booking at RedBus'. The address bar shows 'localhost:4200/addBusDetails'. The page features a 'Home' button in the top left corner. The main content area is a form titled 'Add New Bus Details' with the following fields:

- Departure Bus-stop: karnataka
- Arrival Bus-stop: Mumbai
- Departure Date: 09/26/2023
- Arrival Date: 09/27/2023
- Available seats: 13
- Departure Time: 06:00 PM
- Arrival Time: 06:00 AM
- Bus Vendor: Redbus
- Price of Ticket: 2500

A green 'Add bus' button is located at the bottom of the form. The footer of the page reads '©2023 Copyright RedBus'. The Windows taskbar at the bottom shows the search bar, task view, and various application icons. The system tray on the right indicates the date and time as 'ENG 10:34 PM IN 9/23/2023'.

- This booking module is responsible for adding the bus related details such as departure bus stop, arrival bus stop, date, available seats, timing and price.

Online Bus Ticket Booking at Local X

localhost:4200/addBusDetails

Home

localhost:4200 says
are you sure you want to add this bus?

OK Cancel

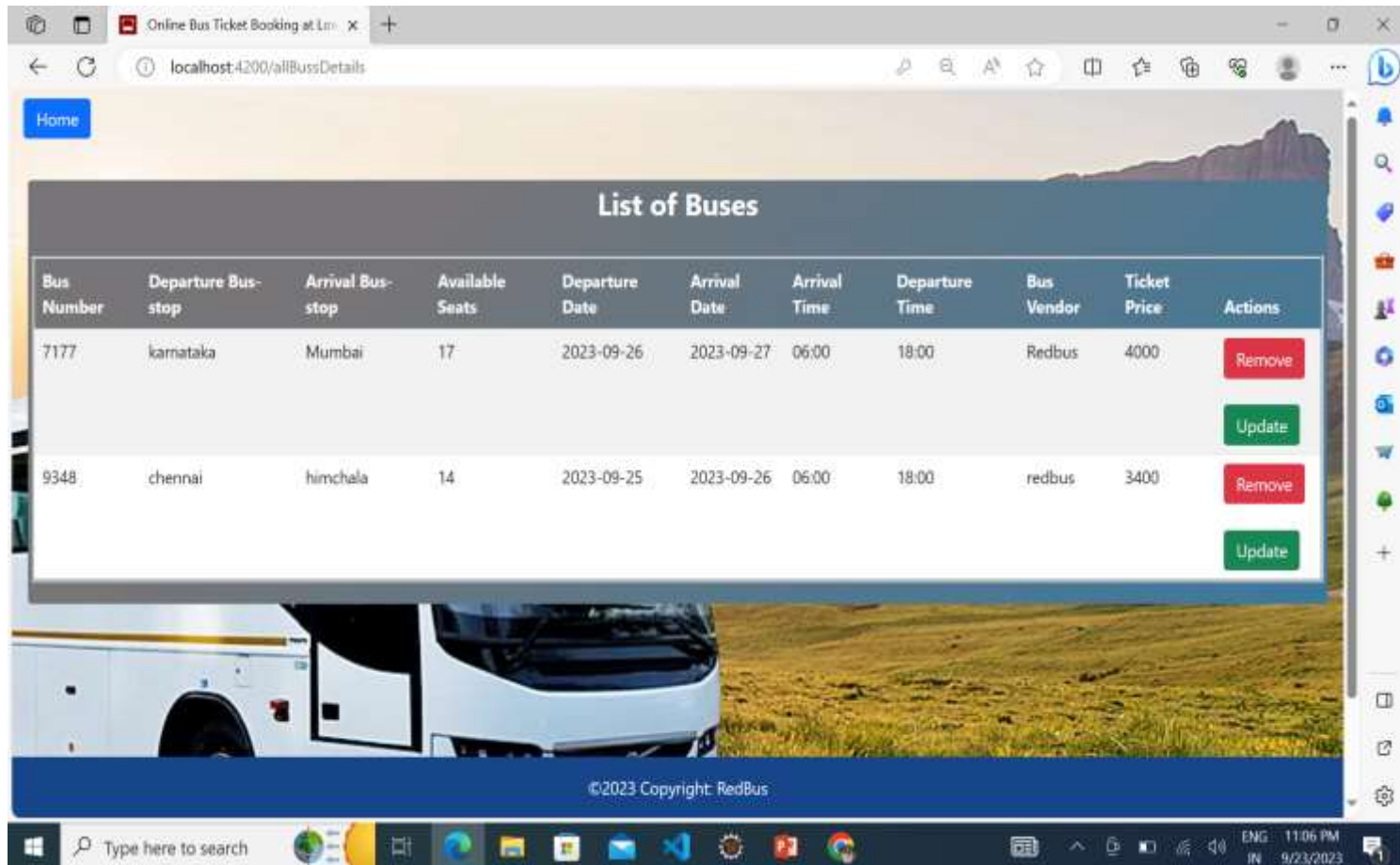
Add New Bus Details

Departure Bus-stop:	Arrival Bus-stop:	
<input type="text" value="karnataka"/>	<input type="text" value="Mumbai"/>	
Departure Date:	Arrival Date:	
<input type="text" value="09/26/2023"/>	<input type="text" value="09/27/2023"/>	
Available seats:	Departure Time:	Arrival Time:
<input type="text" value="13"/>	<input type="text" value="06:00 PM"/>	<input type="text" value="06:00 AM"/>
Bus Vendor:	Price of Ticket:	
<input type="text" value="Redbus"/>	<input type="text" value="2500"/>	

Add bus

©2023 Copyright: RedBus

- After adding the bus details we will get the confirmation message that bus details has added.



Online Bus Ticket Booking at Lim x

localhost:4200/allBussDetails

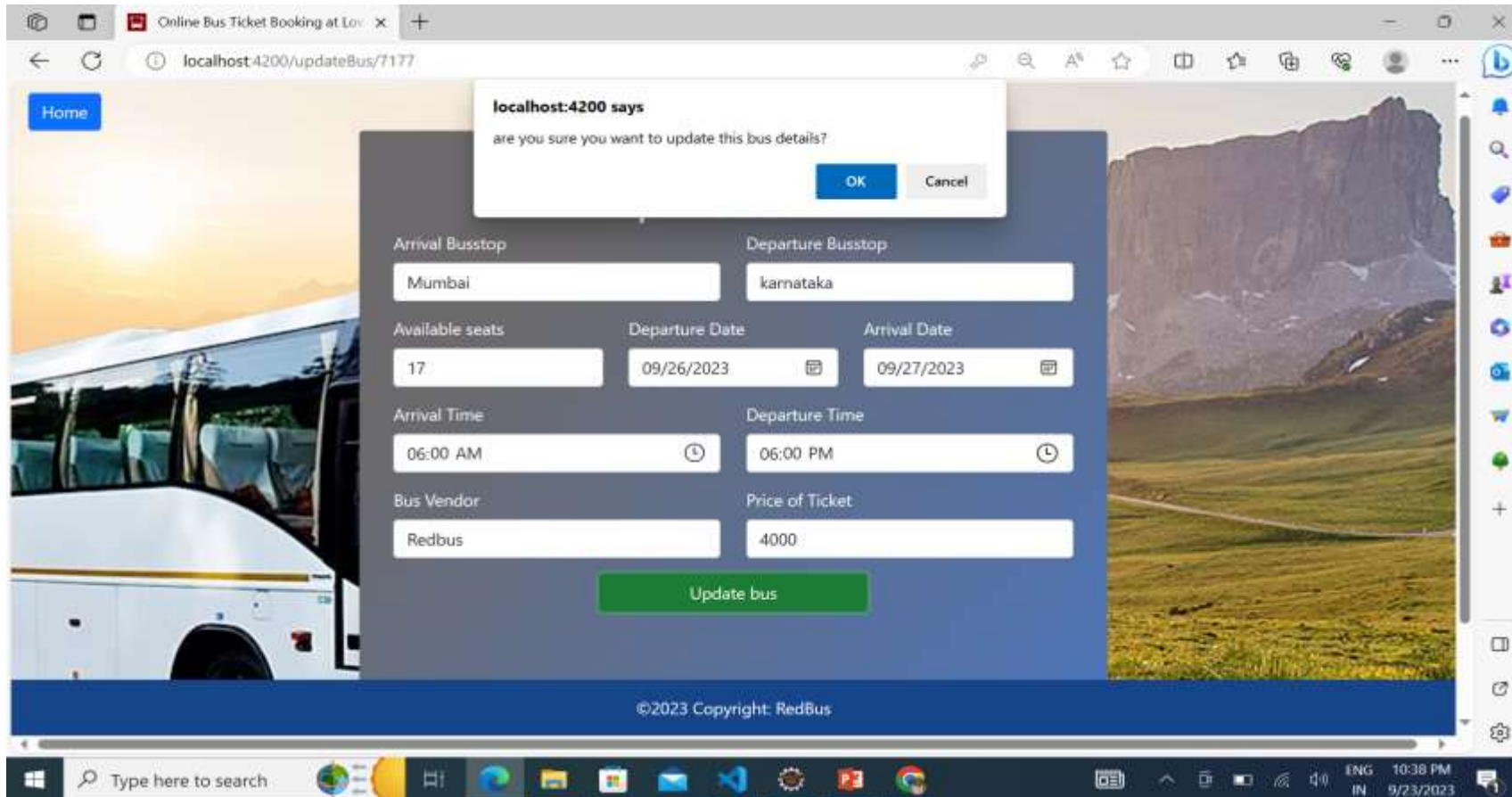
Home

List of Buses

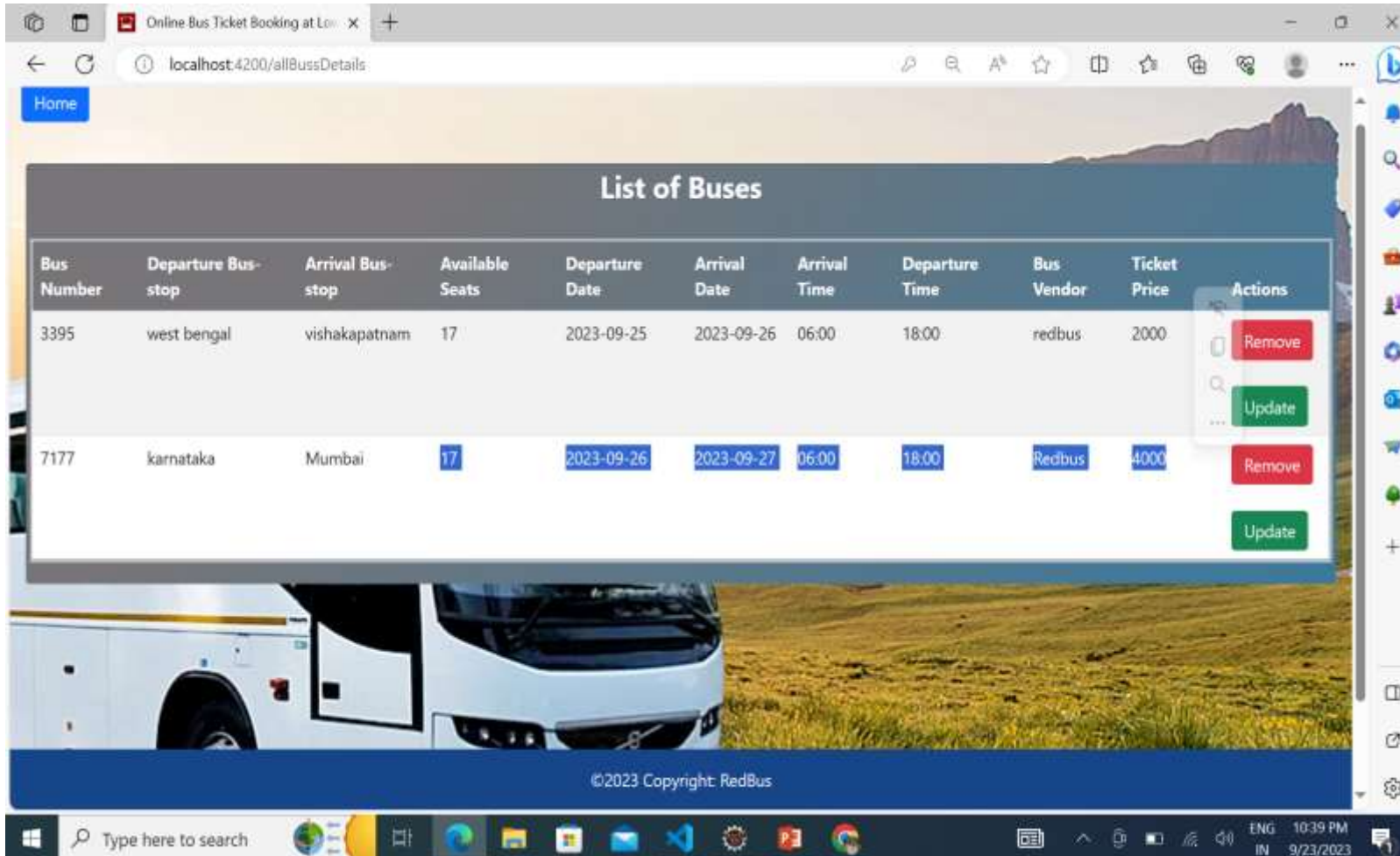
Bus Number	Departure Bus-stop	Arrival Bus-stop	Available Seats	Departure Date	Arrival Date	Arrival Time	Departure Time	Bus Vendor	Ticket Price	Actions
7177	karnataka	Mumbai	17	2023-09-26	2023-09-27	06:00	18:00	Redbus	4000	<button>Remove</button> <button>Update</button>
9348	chennai	himchala	14	2023-09-25	2023-09-26	06:00	18:00	redbus	3400	<button>Remove</button> <button>Update</button>

©2023 Copyright: RedBus

- This booking module has the details of buses which was added by the admin ,where he can remove or update the bus details.



- This module has the updation of bus details where admin can update the details with the confirmation message .



The screenshot shows a web browser window with the address bar displaying "localhost:4200/allBussDetails". The page has a "Home" button and a "List of Buses" section. Below this section is a table with the following columns: Bus Number, Departure Bus-stop, Arrival Bus-stop, Available Seats, Departure Date, Arrival Date, Arrival Time, Departure Time, Bus Vendor, Ticket Price, and Actions. The table contains two rows of bus data. The first row has Bus Number 3395, Departure Bus-stop west bengal, Arrival Bus-stop vishakapatnam, Available Seats 17, Departure Date 2023-09-25, Arrival Date 2023-09-26, Arrival Time 06:00, Departure Time 18:00, Bus Vendor redbus, and Ticket Price 2000. The second row has Bus Number 7177, Departure Bus-stop karnataka, Arrival Bus-stop Mumbai, Available Seats 17, Departure Date 2023-09-26, Arrival Date 2023-09-27, Arrival Time 06:00, Departure Time 18:00, Bus Vendor Redbus, and Ticket Price 4000. The Actions column for each row contains a "Remove" button and an "Update" button. The background of the page features a white bus in a field. The footer of the page displays "©2023 Copyright: RedBus". The Windows taskbar at the bottom shows the search bar, task view button, and several application icons, along with the system clock indicating 10:39 PM on 9/23/2023.

Bus Number	Departure Bus-stop	Arrival Bus-stop	Available Seats	Departure Date	Arrival Date	Arrival Time	Departure Time	Bus Vendor	Ticket Price	Actions
3395	west bengal	vishakapatnam	17	2023-09-25	2023-09-26	06:00	18:00	redbus	2000	<button>Remove</button> <button>Update</button>
7177	karnataka	Mumbai	17	2023-09-26	2023-09-27	06:00	18:00	Redbus	4000	<button>Remove</button> <button>Update</button>

- After updating the bus details here admin can see the updated bus details.

The screenshot shows a web browser window with the address bar displaying 'localhost:4200/allBussDetails'. A modal dialog box is open, asking for confirmation to remove a bus. The dialog text reads: 'localhost:4200 says are you sure you want to remove this bus?' with 'OK' and 'Cancel' buttons.

Below the dialog, a table lists bus details:

Bus Number	Departure Bus-stop	Arrival Bus-stop	Available Seats	Departure Date	Arrival Date	Arrival Time	Departure Time	Bus Vendor	Ticket Price	Actions
3395	west bengal	vishakapatnam	17	2023-09-25	2023-09-26	06:00	18:00	redbus	2000	<button>Remove</button> <button>Update</button>
7177	karnataka	Mumbai	17	2023-09-26	2023-09-27	06:00	18:00	Redbus	4000	<button>Remove</button> <button>Update</button>

The background of the application shows a white bus in a field. The footer displays '©2023 Copyright: RedBus'.

- In this module ,admin can cancel the added bus details with the confirmation message.

Online Bus Ticket Booking at Localhost:4200

Home

List of Buses

Bus Number	Departure Bus-stop	Arrival Bus-stop	Available Seats	Departure Date	Arrival Date	Arrival Time	Departure Time	Bus Vendor	Ticket Price	Actions
7177	karnataka	Mumbai	17	2023-09-26	2023-09-27	06:00	18:00	Redbus	4000	<button>Remove</button> <button>Update</button>

©2023 Copyright: RedBus

- After canceling the bus details admin can see the canceled bus details

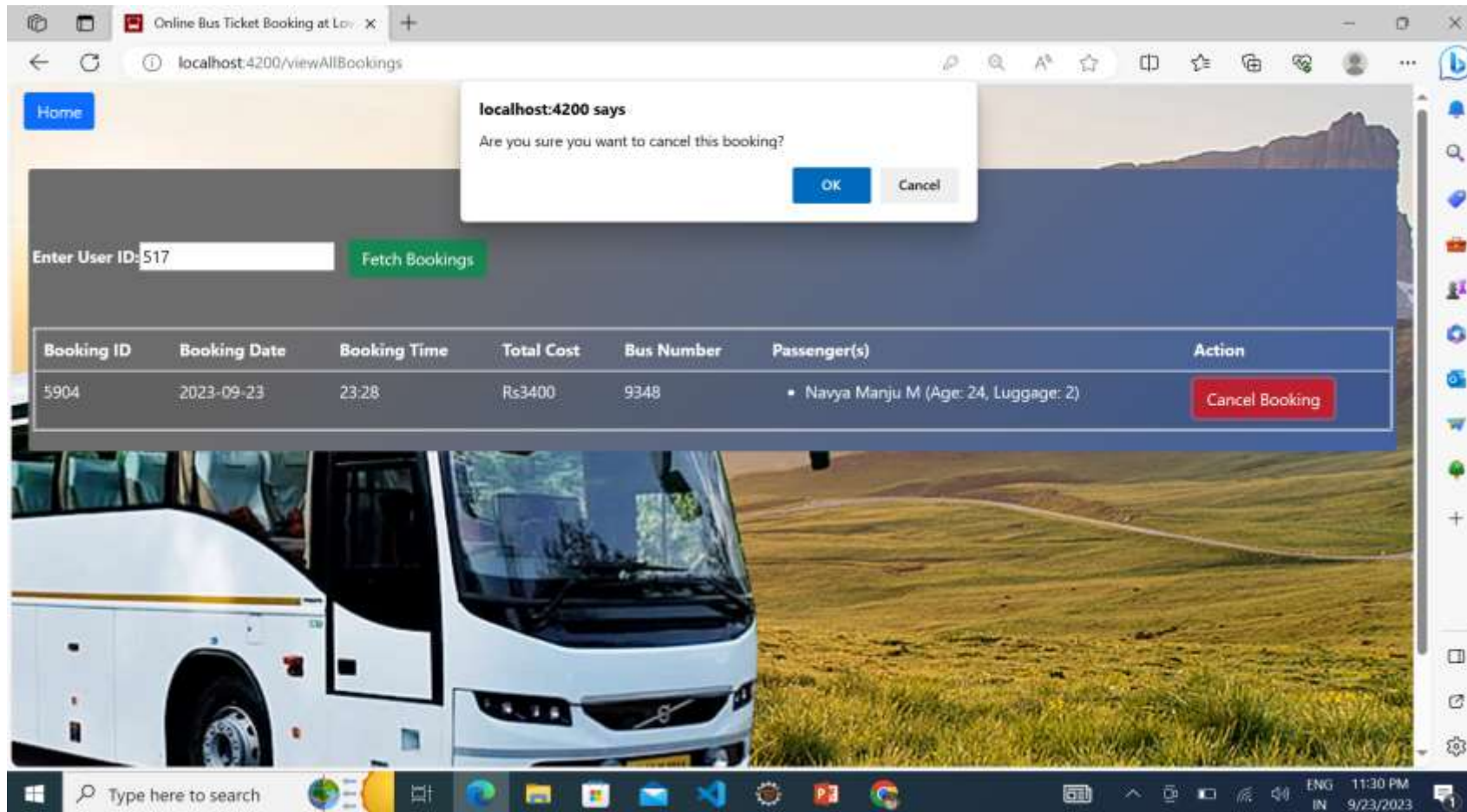


The screenshot shows a web browser window with the address bar displaying 'localhost:4200/viewAllBookings'. The page has a 'Home' button in the top left corner. The main heading is 'List of Bookings'. Below the heading, there is a search bar with the text 'Enter User ID: 517' and a green 'Fetch Bookings' button. Below the search bar, there is a table with the following data:

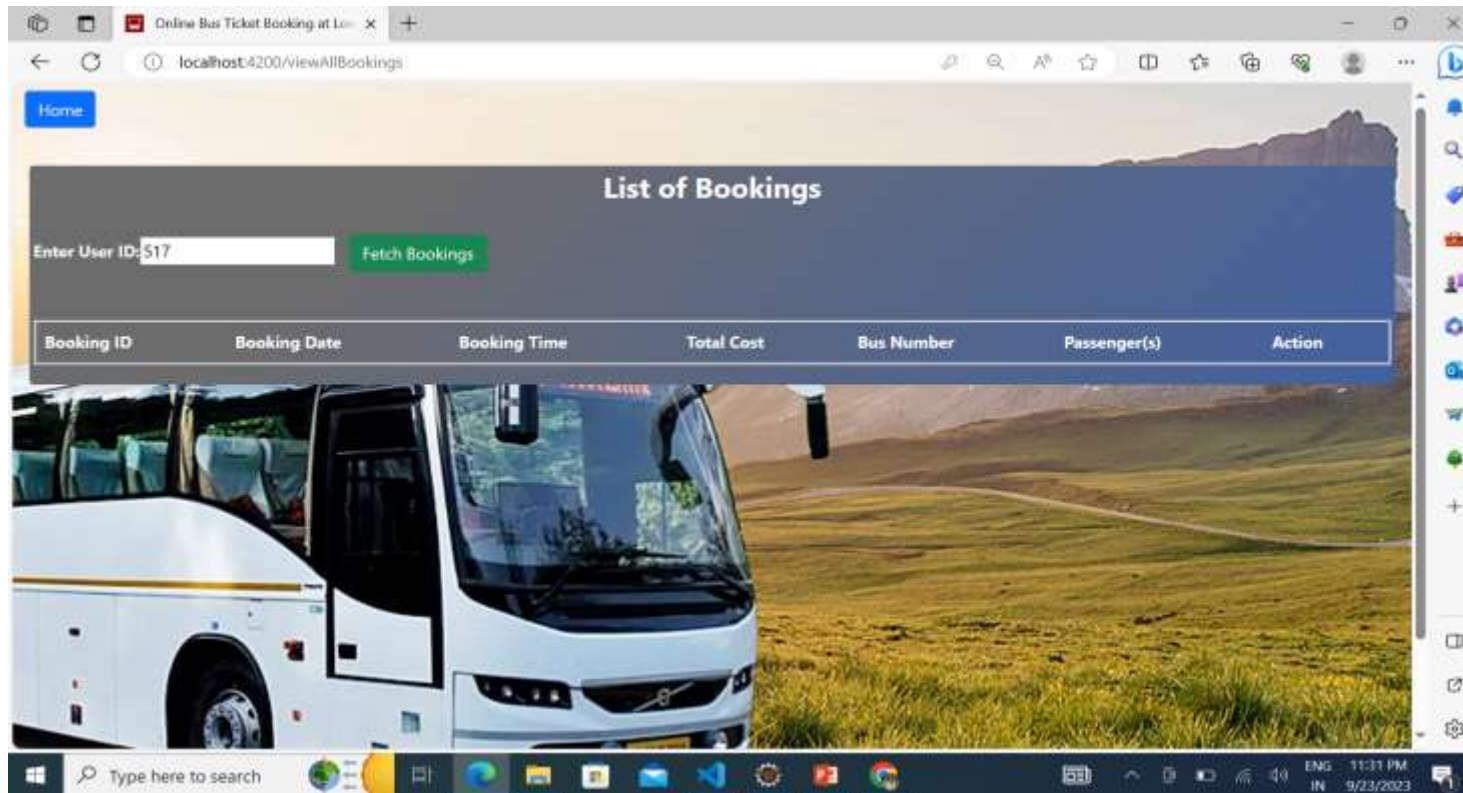
Booking ID	Booking Date	Booking Time	Total Cost	Bus Number	Passenger(s)	Action
5904	2023-09-23	23:28	Rs3400	9348	• Navya Manju M (Age: 24, Luggage: 2)	Cancel Booking

The background of the page features a white bus parked in a grassy field. The Windows taskbar is visible at the bottom of the screen.

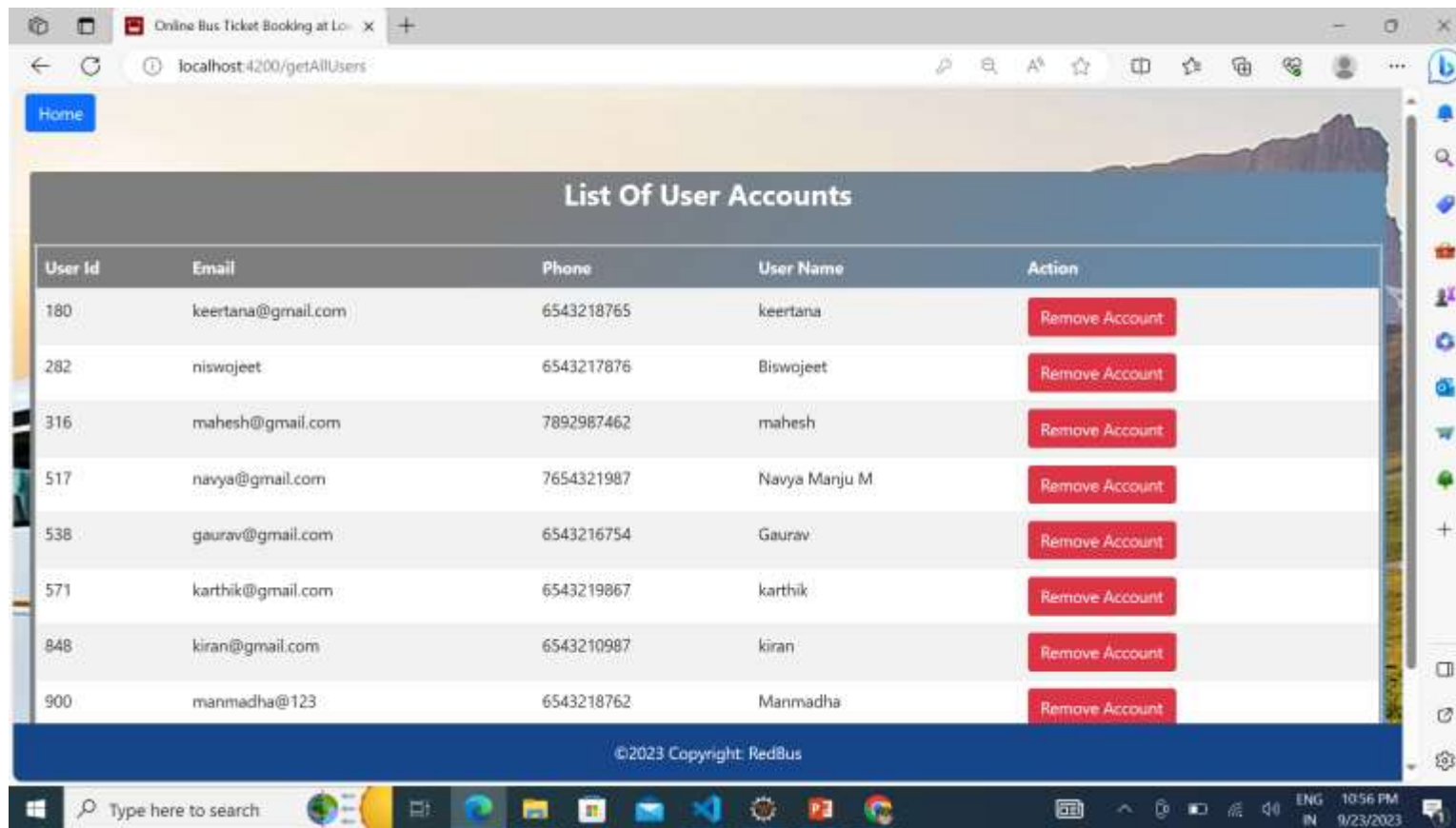
- This booking module has the list of bookings which was done by the user .
- Here admin can get the booking details by entering the user id .
- After entering the id the booking details were fetched with booking id,date ,booking time,total cost,passenger.
- Admin can cancel the booking .



- After canceling the bookings admin can view confirmation message



- After canceling the bookings admin can see the canceled bookings



Online Bus Ticket Booking at Localhost:4200/getAllUsers

Home

List Of User Accounts

User Id	Email	Phone	User Name	Action
180	keertana@gmail.com	6543218765	keertana	Remove Account
282	niswojeet	6543217876	Biswojeet	Remove Account
316	mahesh@gmail.com	7892987462	mahesh	Remove Account
517	navya@gmail.com	7654321987	Navya Manju M	Remove Account
538	gaurav@gmail.com	6543216754	Gaurav	Remove Account
571	karthik@gmail.com	6543219867	karthik	Remove Account
848	kiran@gmail.com	6543210987	kiran	Remove Account
900	manmadha@123	6543218762	Manmadha	Remove Account

©2023 Copyright: RedBus

- This booking module has the list of user accounts which was registered by the user with user id,email,phone,name.
- Here admin has option to remove the user account .

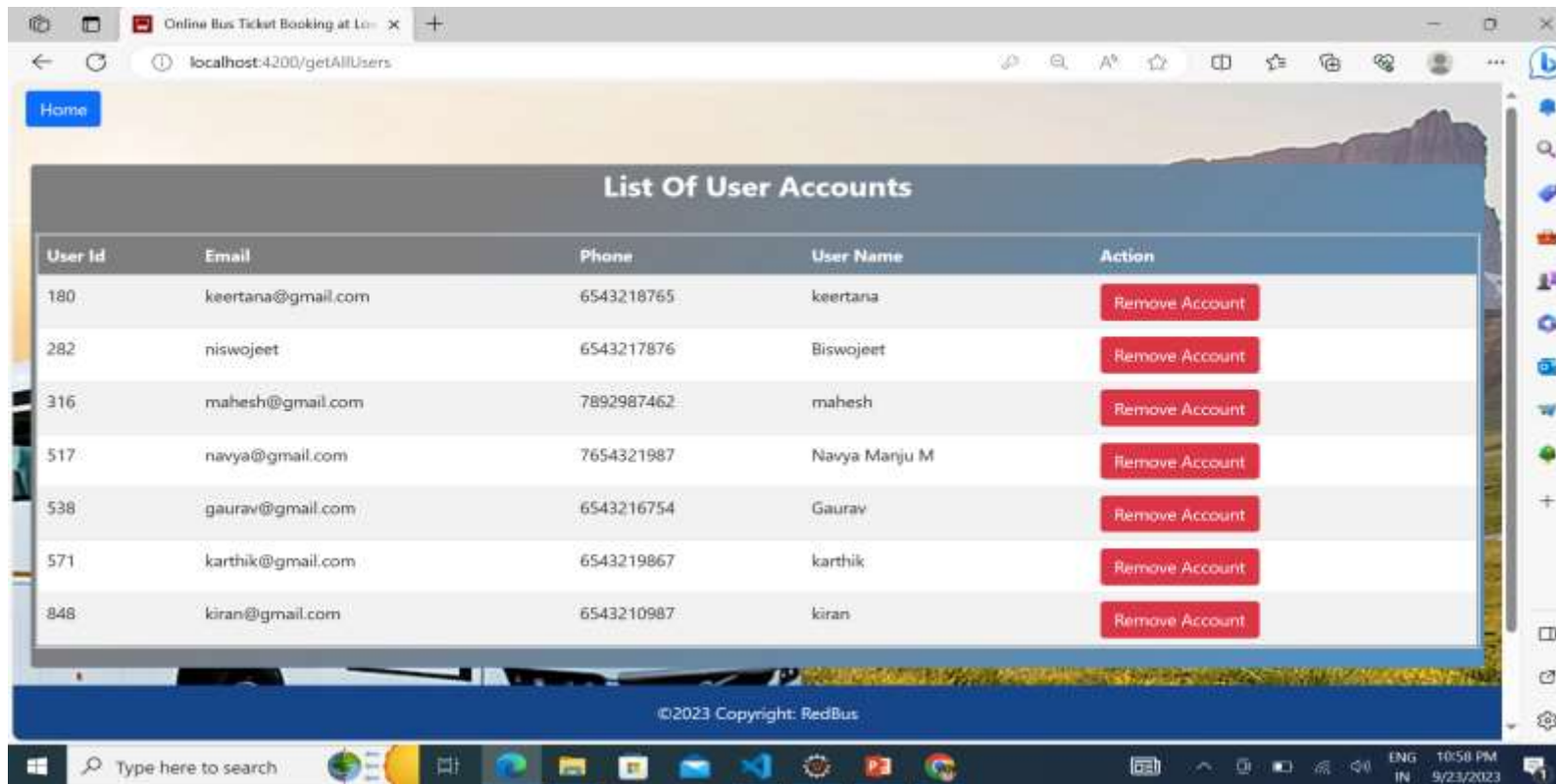
The screenshot shows a web browser window with the address bar displaying 'localhost:4200/getAllUsers'. The page contains a table with user details and an 'Action' column. A modal dialog box is open, asking for confirmation to remove an account.

User Id	Email			Action
180	keertana@gmail.com			Remove Account
282	niswojeet	6543217876	Biswojeet	Remove Account
316	mahesh@gmail.com	7892987462	mahesh	Remove Account
517	navya@gmail.com	7654321987	Navya Manju M	Remove Account
538	gaurav@gmail.com	6543216754	Gaurav	Remove Account
571	karthik@gmail.com	6543219867	karthik	Remove Account
848	kiran@gmail.com	6543210987	kiran	Remove Account
900	manmadha@123	6543218762	Manmadha	Remove Account

localhost:4200 says
Are you sure you want to remove this account?
OK Cancel

©2023 Copyright: RedBus

- After removing the account details admin will get the confirmation message.



The screenshot shows a web browser window with the title 'Online Bus Ticket Booking at Lo...'. The address bar shows 'localhost:4200/getAllUsers'. The page has a 'Home' button and a 'List Of User Accounts' section. Below this is a table with columns: User Id, Email, Phone, User Name, and Action. The table contains 7 rows of user data. Each row has a red 'Remove Account' button in the Action column. The footer of the page says '©2023 Copyright: RedBus'.

User Id	Email	Phone	User Name	Action
180	keertana@gmail.com	6543218765	keertana	Remove Account
282	niswojeet	6543217876	Biswojeet	Remove Account
316	mahesh@gmail.com	7892987462	mahesh	Remove Account
517	navya@gmail.com	7654321987	Navya Manju M	Remove Account
538	gaurav@gmail.com	6543216754	Gaurav	Remove Account
571	karthik@gmail.com	6543219867	karthik	Remove Account
848	kiran@gmail.com	6543210987	kiran	Remove Account

- After removing the account details the admin can see the removed account details.

Online Bus Ticket Booking at Localhost:4200

localhost:4200/addUser

RedBus

localhost:4200 says
Your User Id is: 538 Kindly Note it down for future use...

OK

Gaurav

Email:
gaurav@gmail.com

Phone Number:
+91 6543216754

Password:

Confirm password:

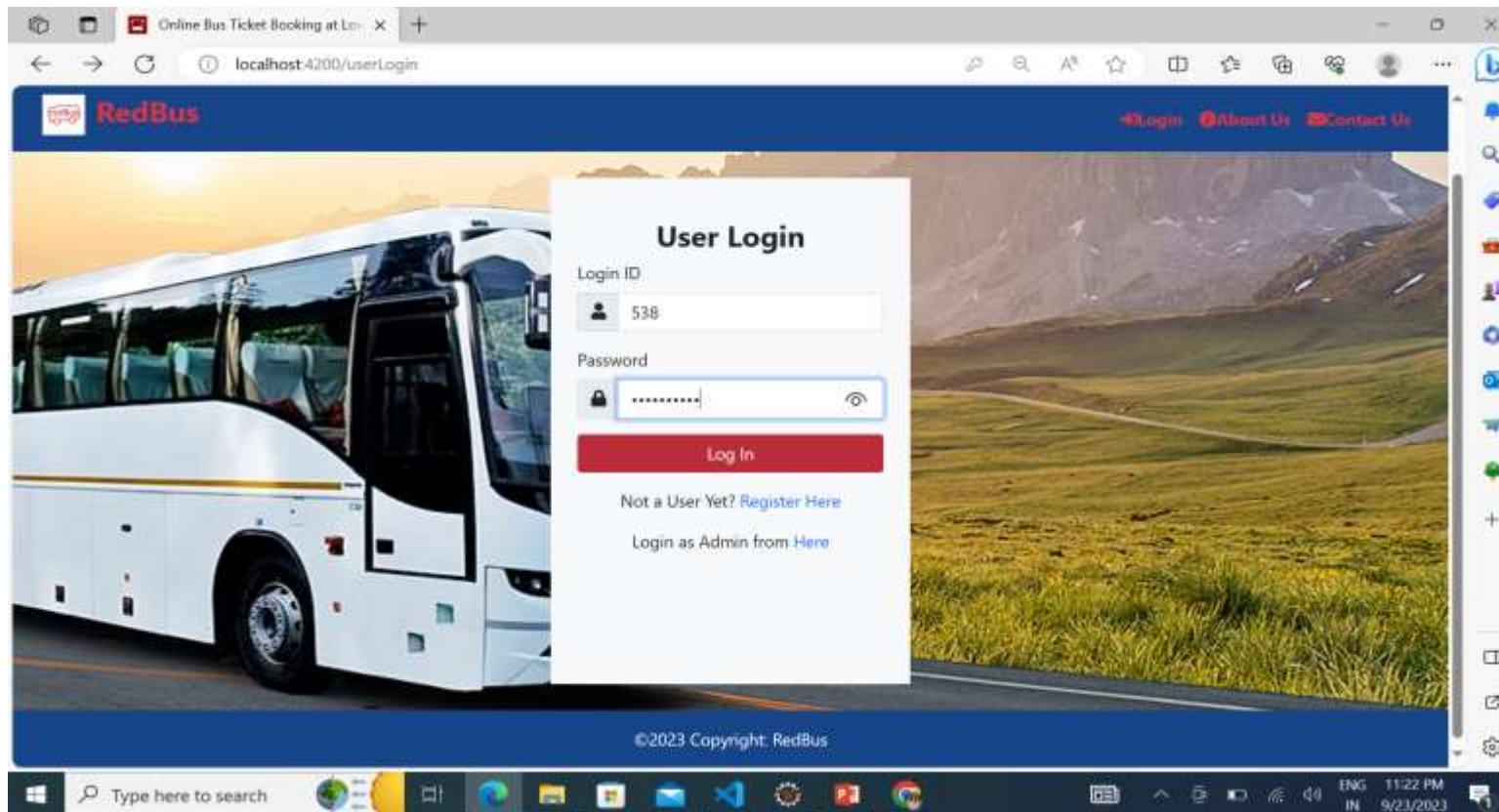
Register

Already a Registered User? [User Login](#)

©2023 Copyright: RedBus

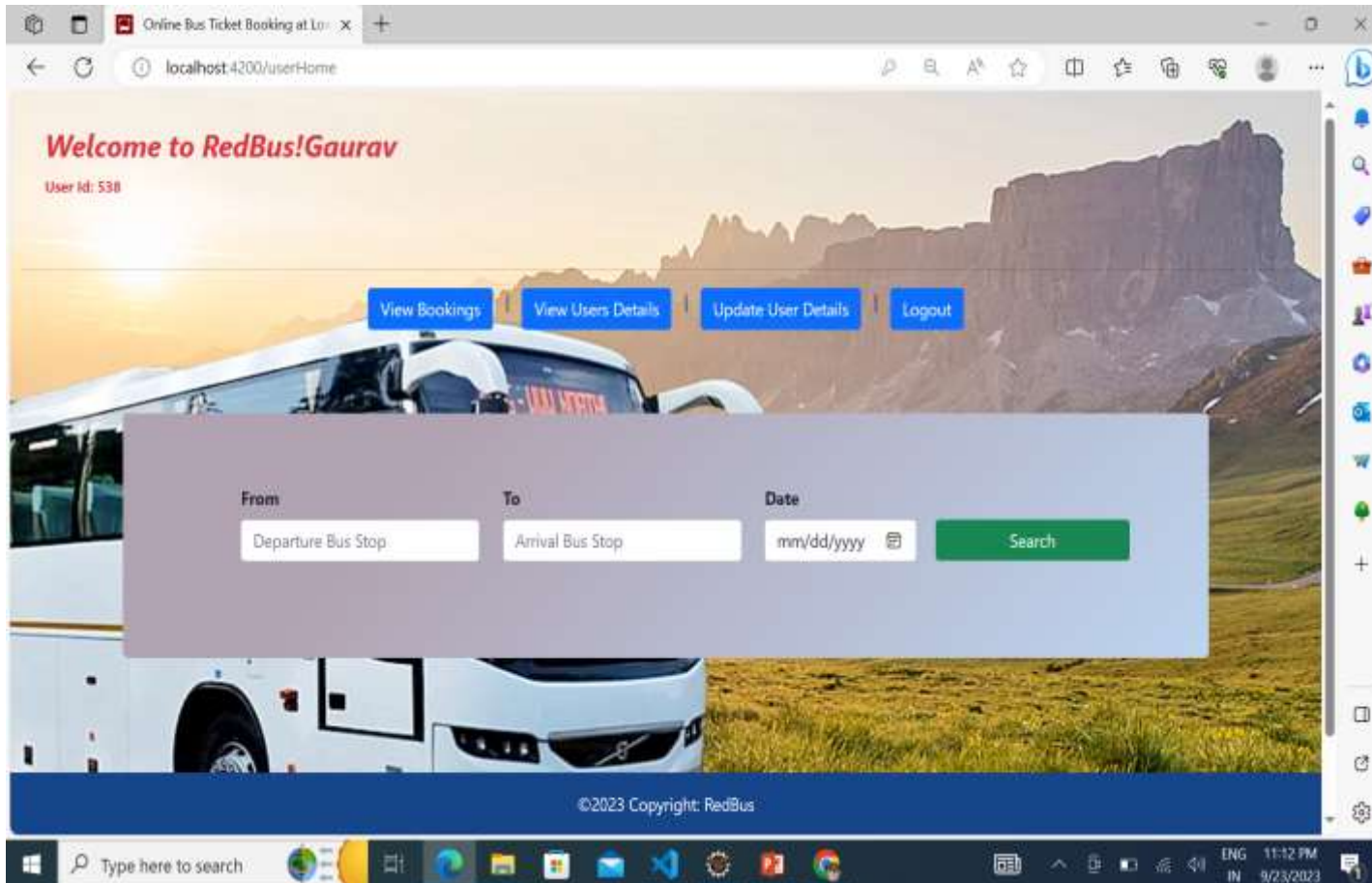
Registration module:

- In this module he/she need to register to the application to get into the application.
- After registering user will get the user id to login to the application.



Login bus booking Module:

- In this module a person can want to enter into this software project .
- he/she must have pass through this module, in other word we can say that this is authentication module because the user authenticates with his/her password and username.



- This module deals with addition of bookings by searching bus details and deletion of booking details.
- Can view the user details ,booking details and update user details.

Online Bus Ticket Booking at Low...

localhost:4200/userHome

From: To: Date:

Bus Number: 9348
From: chennai
To: himchala
Departure date: 2023-09-25
Departure Time: 18:00
Arrival Date: 2023-09-26
Arrival Time: 06:00
Cost: ₹3,400.00

©2023 Copyright: RedBus

- After searching the bus details user will get the bus details to add the passenger.

The screenshot shows a web browser window with the address bar displaying 'localhost:4200/addPassengers/9348'. The page has a 'Home' button in the top left corner. The main content area is titled 'Add Passengers' and features a form for 'Passenger 1'. The form includes three input fields: 'Name' with the value 'gaurav', 'Age' with the value '23', and 'luggage' with the value '4'. Below the 'Name' field is a red 'remove' button. At the bottom of the form are two red buttons: 'add Booking' and 'Add More'. The footer of the page contains the text '©2023 Copyright: RedBus'. The Windows taskbar is visible at the bottom of the screen.

Home

Add Passengers

Passenger 1

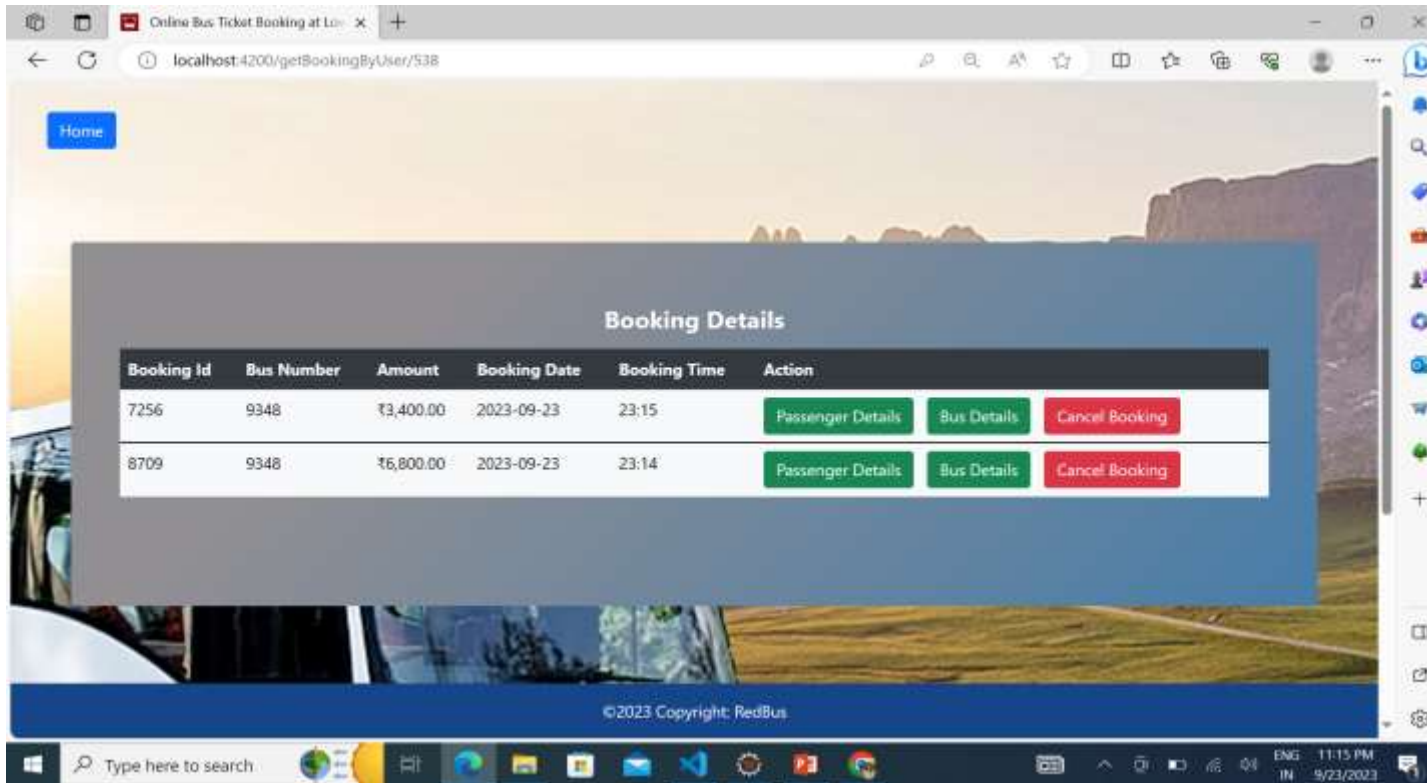
Name: gaurav Age: 23 luggage: 4

remove

add Booking Add More

©2023 Copyright: RedBus

- After viewing bus details user can add the passenger by entering name,age,luggage.
- And user has the option to book for another passenger and remove the added passenger.



The screenshot shows a web browser window with the URL `localhost:4200/getBookingByUser/538`. The page displays a "Booking Details" modal with a table of bookings. The table has columns for Booking Id, Bus Number, Amount, Booking Date, Booking Time, and Action. Two bookings are listed: one with Booking Id 7256 and another with Booking Id 8709. Each booking row has three buttons: "Passenger Details" (green), "Bus Details" (green), and "Cancel Booking" (red). The background of the page shows a scenic view of a bus and a landscape. The footer of the page reads "©2023 Copyright: RedBus".

Booking Id	Bus Number	Amount	Booking Date	Booking Time	Action
7256	9348	₹3,400.00	2023-09-23	23:15	Passenger Details Bus Details Cancel Booking
8709	9348	₹6,800.00	2023-09-23	23:14	Passenger Details Bus Details Cancel Booking

- After booking done the user can view the booking details with passenger details, and user can view the bus details which was he/she booked.
- And he/she will get the option to cancel the booking.

The screenshot displays a web application titled "Online Bus Ticket Booking at Le...". The browser address bar shows "localhost:4200/getBookingByUser/538". The main content area features a "Booking Details" modal with a table listing two bookings. Below this, a "Passenger Details" section shows a single passenger record for "gaurav".

Booking Id	Bus Number	Amount	Booking Date	Booking Time	Action
7256	9348	₹3,400.00	2023-09-23	23:15	Passenger Details Bus Details Cancel Booking
8709	9348	₹6,800.00	2023-09-23	23:14	Passenger Details Bus Details Cancel Booking

Passenger ID	Passenger Name	Passenger Age	Passenger Luggage	Action
10	gaurav	23	4	Update

©2023 Copyright: RedBus

- After booking user can view the passenger details by clicking the passenger button.

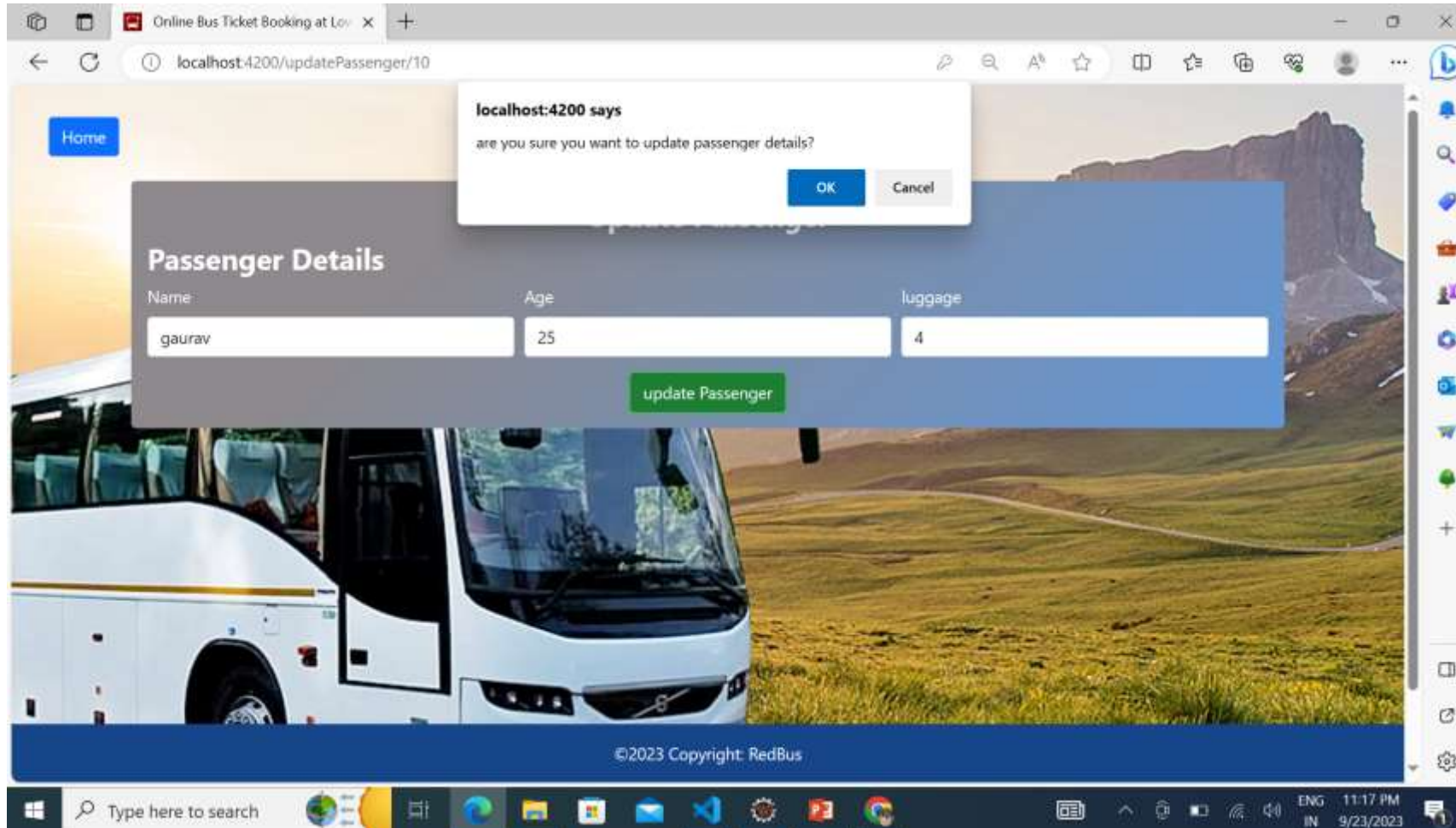
The screenshot displays a web browser window with the title "Online Bus Ticket Booking at Local" and the URL "localhost:4200/getBookingByUser/538". The main content area is titled "Booking Details" and contains a table with two rows of booking information. Each row has buttons for "Passenger Details", "Bus Details", and "Cancel Booking". Below the table, the bus details are listed: Bus Number: 9348, From: chennai, To: himchala, Departure date: 2023-09-25, Departure Time: 18:00, Arrival Date: 2023-09-26, and Arrival Time: 06:00. The footer of the application shows "©2023 Copyright: RedBus".

Booking Id	Bus Number	Amount	Booking Date	Booking Time	Action
7256	9348	₹3,400.00	2023-09-23	23:15	Passenger Details Bus Details Cancel Booking
8709	9348	₹6,800.00	2023-09-23	23:14	Passenger Details Bus Details Cancel Booking

Bus Number: 9348
From: chennai
To: himchala
Departure date: 2023-09-25
Departure Time: 18:00
Arrival Date: 2023-09-26
Arrival Time: 06:00

©2023 Copyright: RedBus

- After booking user can view the bus details by clicking the bus details button.



The user has the option to update the passenger details. After updating, he/she will get the confirmation message.

The screenshot shows a web browser window with the URL `localhost:4200/getBookingByUser/538`. The page displays a modal titled "Booking Details" with a table of bookings and a sub-modal for "Passenger Details".

Booking Id	Bus Number	Amount	Booking Date	Booking Time	Action
7256	9348	₹3,400.00	2023-09-23	23:15	Passenger Details Bus Details Cancel Booking
8709	9348	₹6,800.00	2023-09-23	23:14	Passenger Details Bus Details Cancel Booking

Below the bookings table, the "Passenger Details" sub-modal is open, showing a table with passenger information:

Passenger ID	Passenger Name	Passenger Age	Passenger Luggage	Action
10	gaurav	25	4	Update

The footer of the application shows "©2023 Copyright: RedBus".

- After updation user can see the updated details.

localhost:4200 says
are you sure you want to cancel booking?

OK Cancel

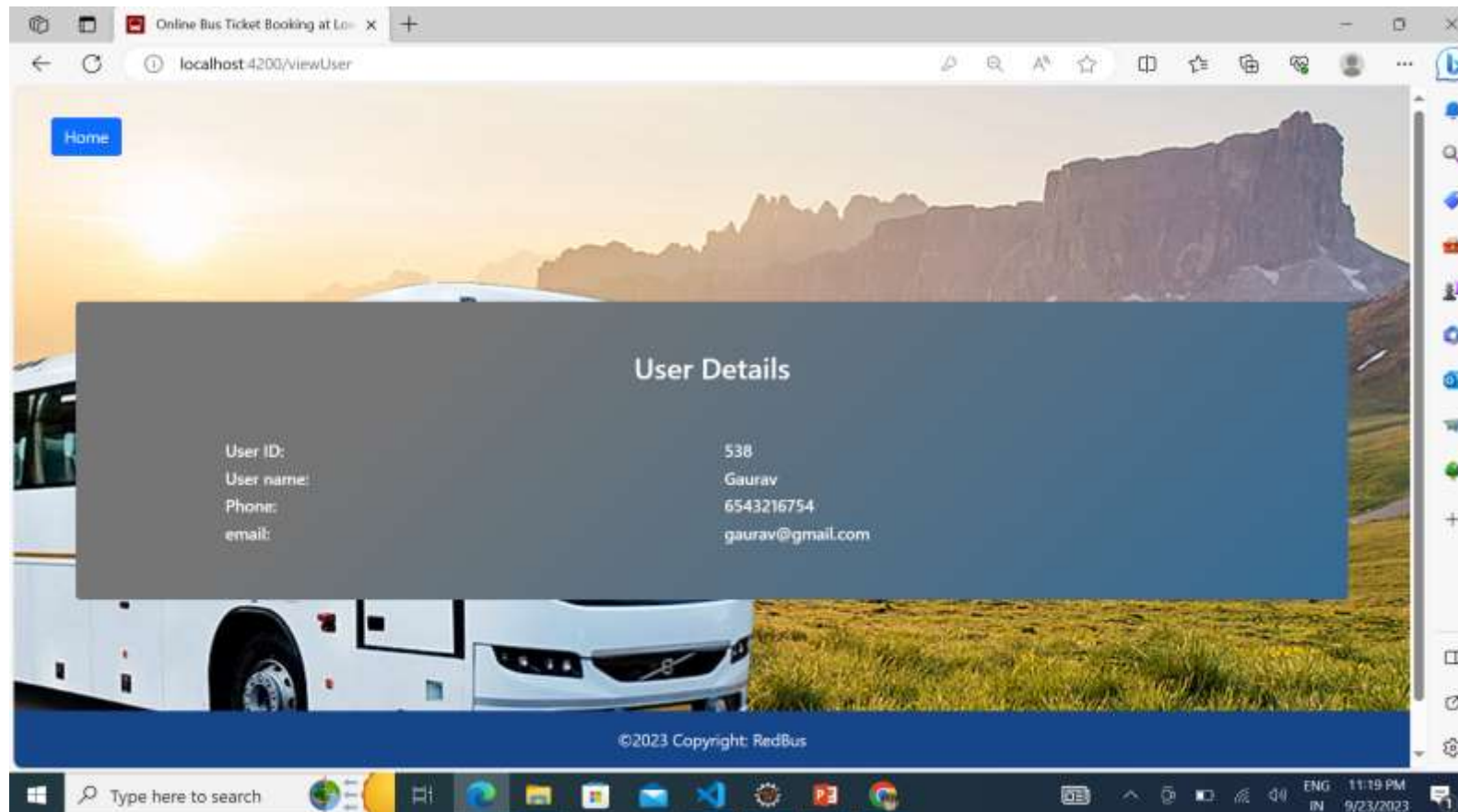
Booking Id	Bus Number	Amount	Booking Date	Booking Time	Action
7256	9348	₹3,400.00	2023-09-23	23:15	Passenger Details Bus Details Cancel Booking
8709	9348	₹6,800.00	2023-09-23	23:14	Passenger Details Bus Details Cancel Booking

Passenger Details:

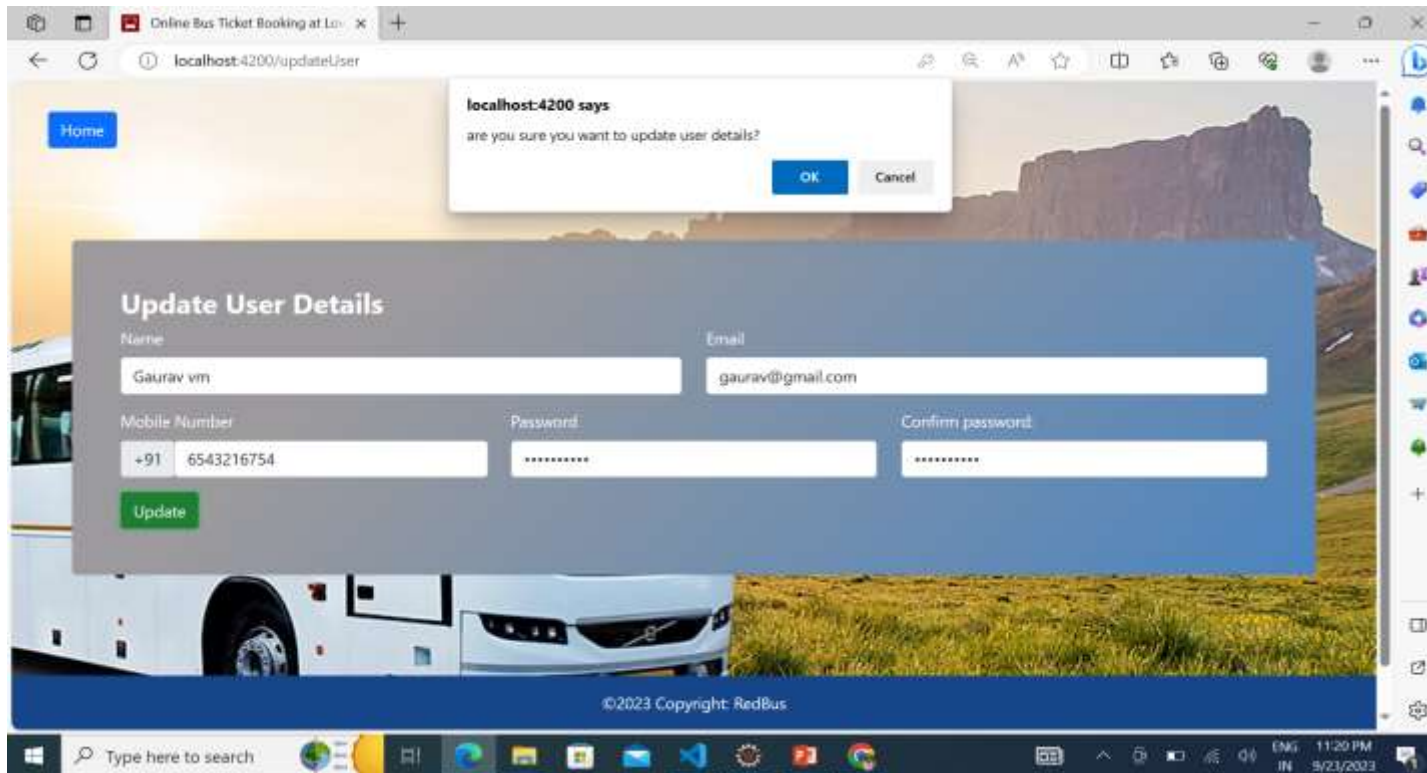
Passenger ID	Passenger Name	Passenger Age	Passenger Luggage	Action
10	gaurav	25	4	Update

©2023 Copyright: RedBus

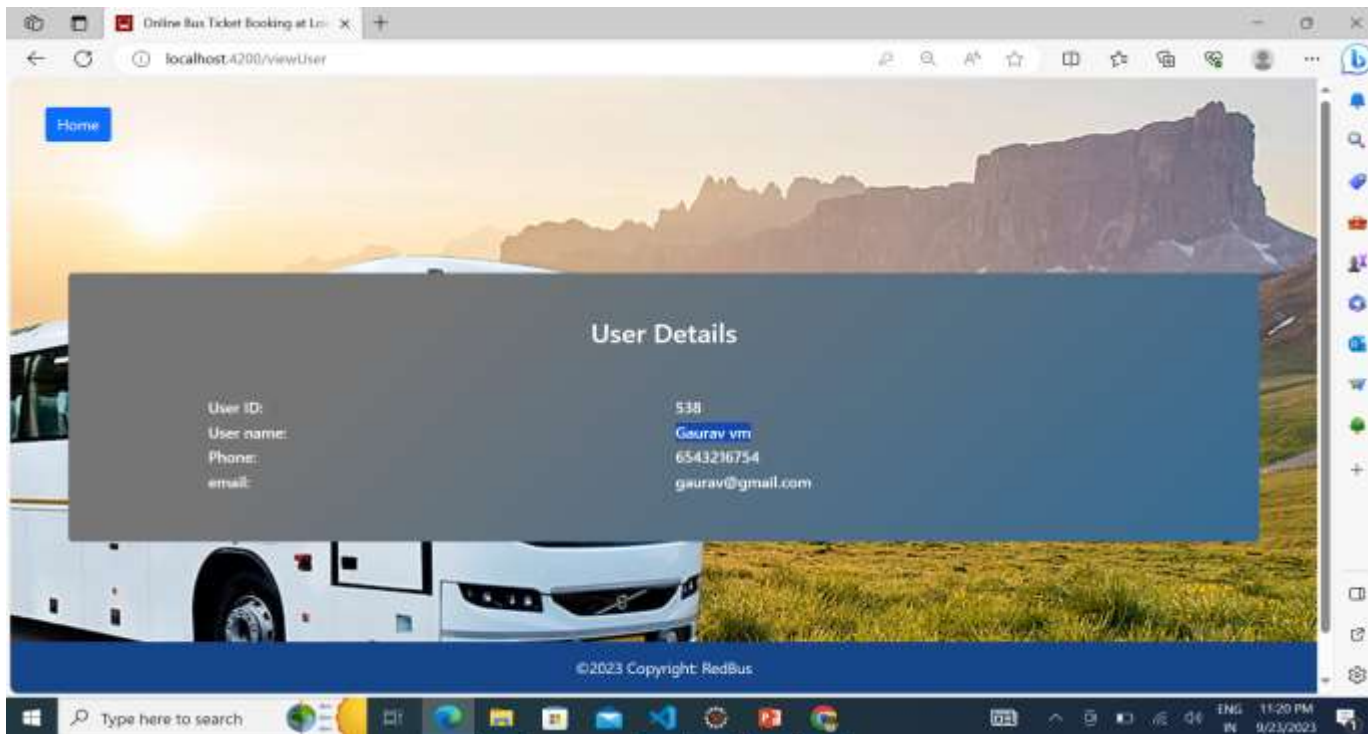
- After canceled booking the user will get the confirmation message.



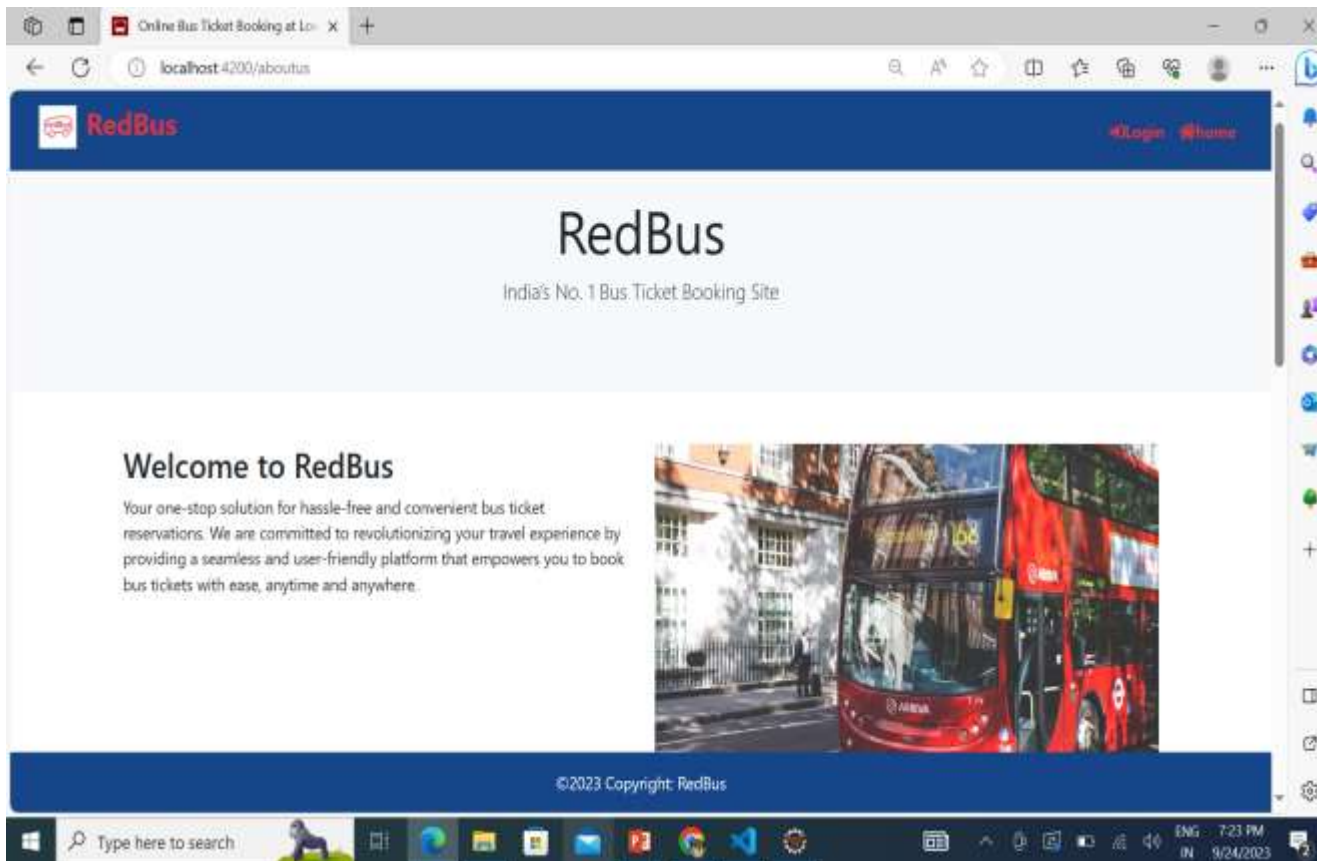
- This booking module has the user details which was registered.



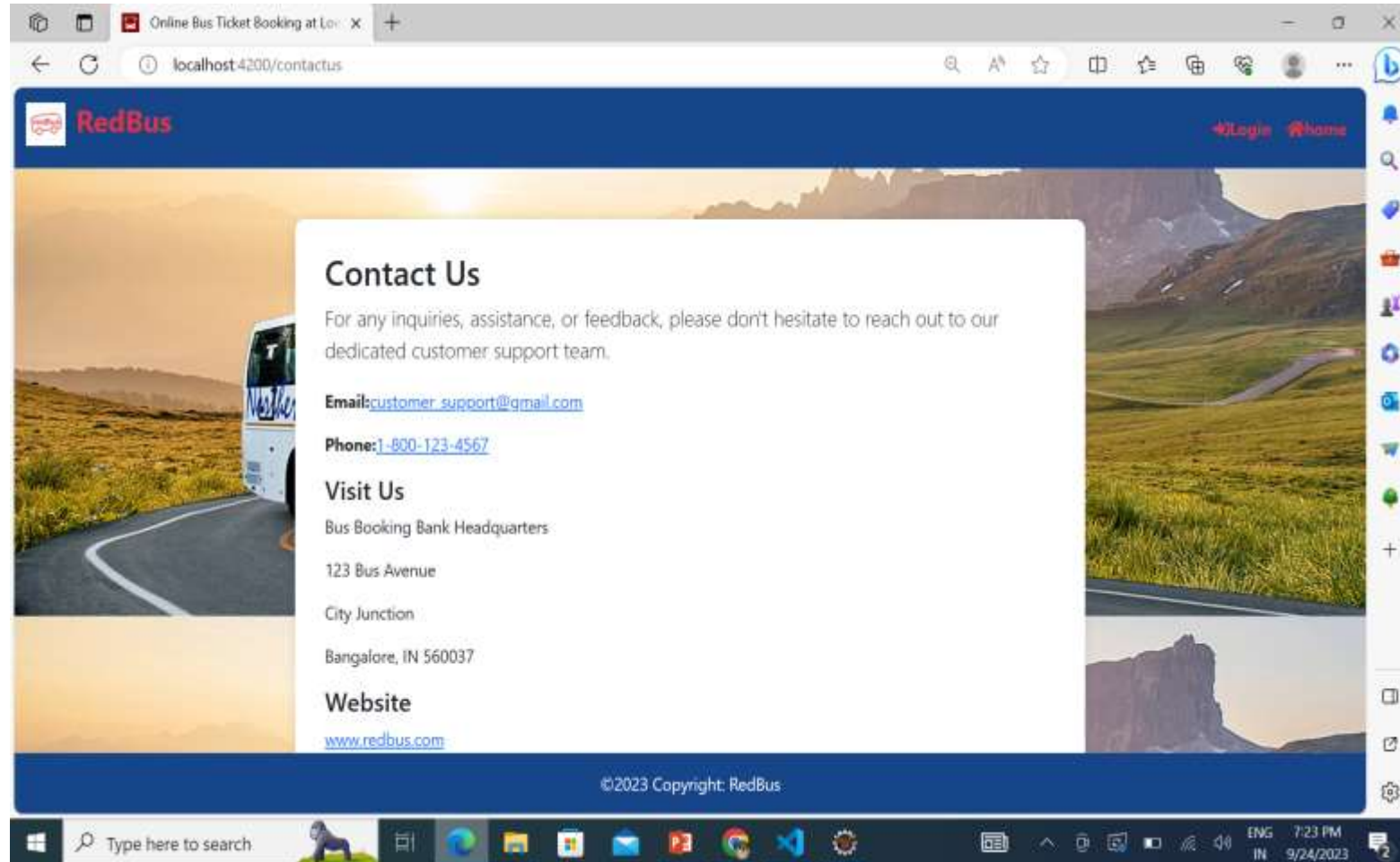
- User has the option to update the registered details with confirmation message.



- After updating the user details he can view the updated user details.

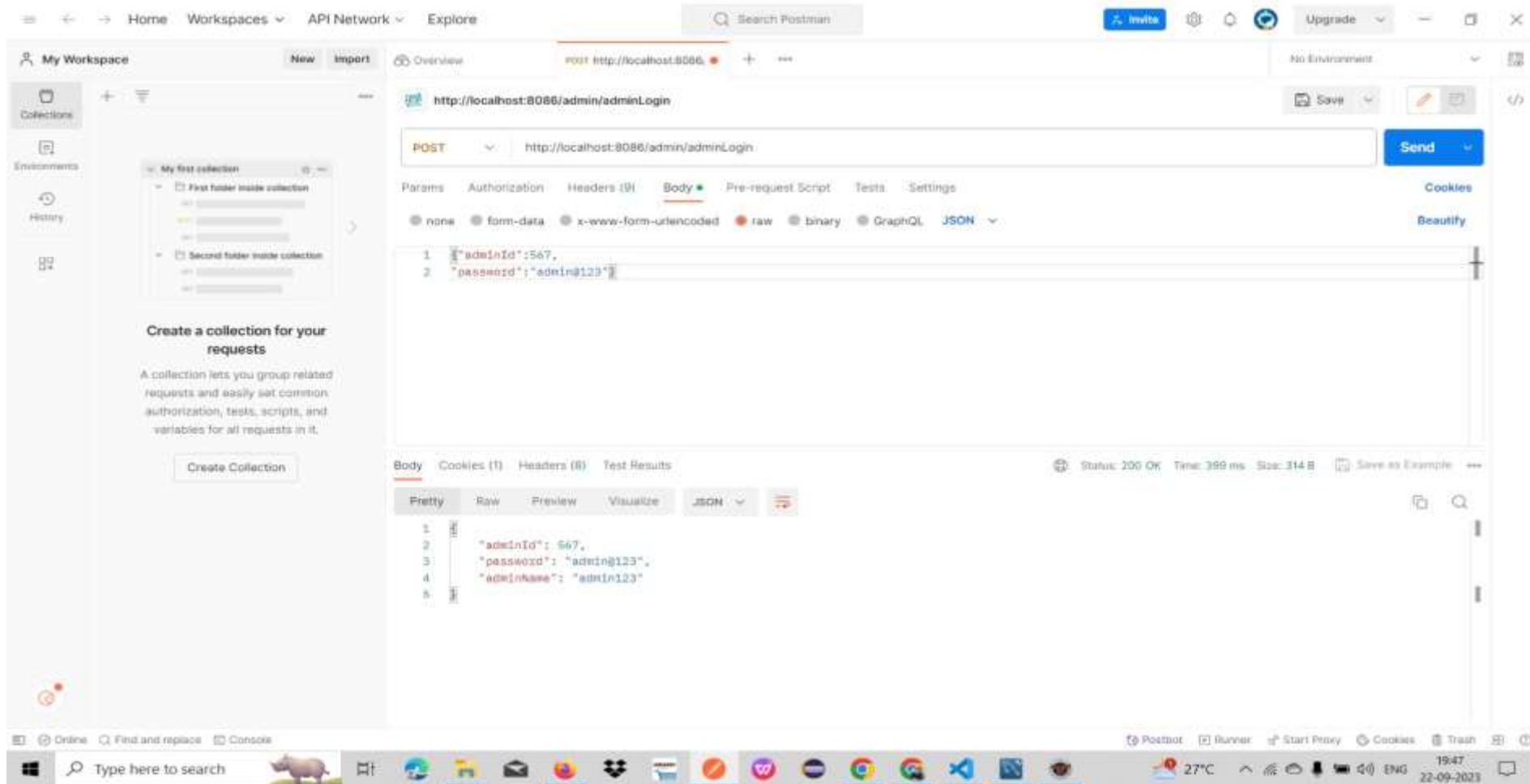


- This module has the information of the web application ,ie RedBus.



- This module includes the contact information of the web application.

Postman Testing of Admin Login.



The screenshot displays the Postman application interface. The top navigation bar includes links for Home, Workspaces, API Network, and Explore, along with a search bar and utility buttons like Invite, Upgrade, and Save. The left sidebar shows the 'My Workspace' section with options for Collections, Environments, History, and a 'Create Collection' button. The main workspace is divided into two panes. The top pane shows a REST client request to `http://localhost:8086/admin/getAdmin/567` using the GET method. The request body is set to JSON and contains the following data:

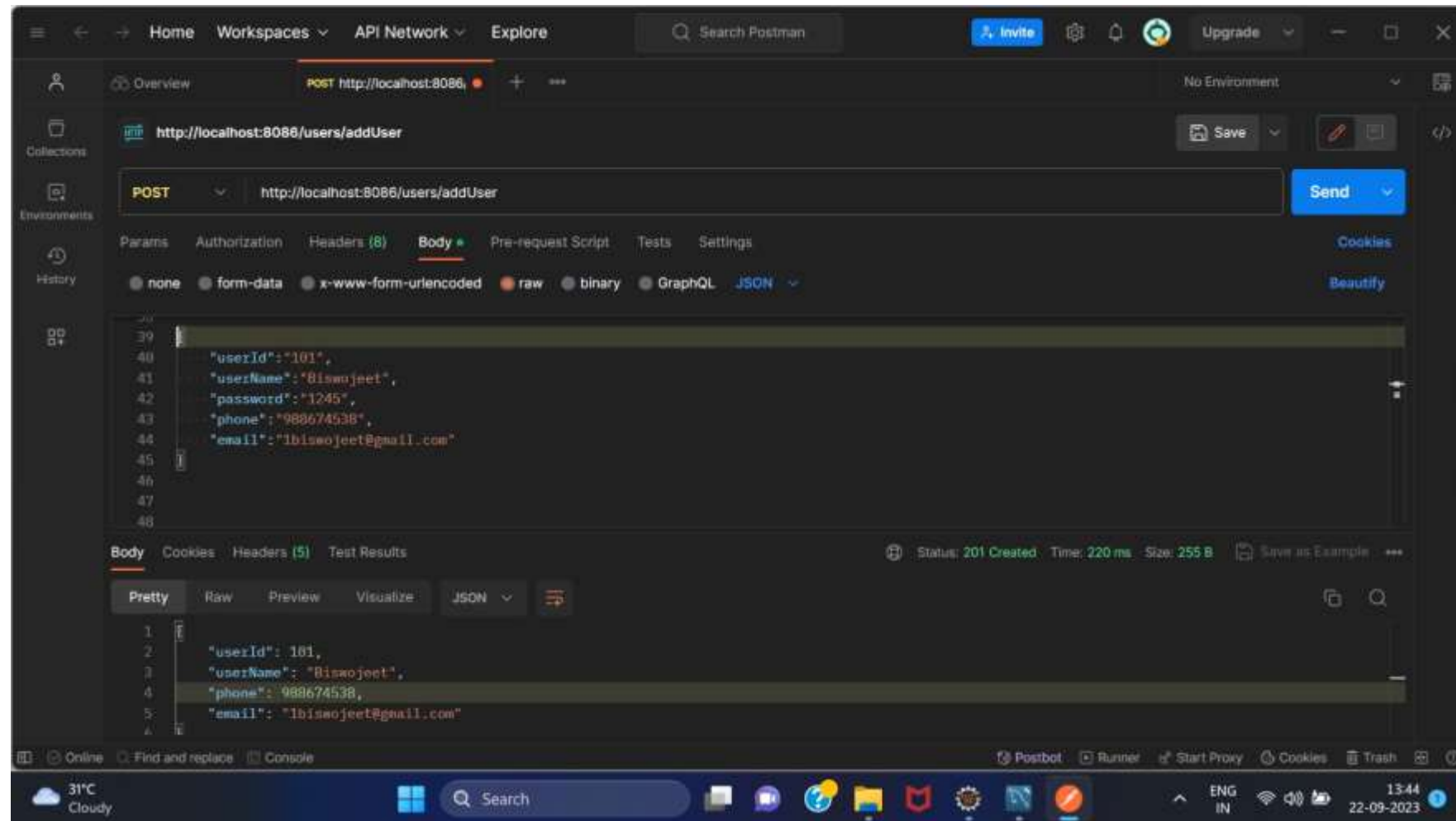
```
1 { "adminId": 567,  
2   "password": "admin@123" }
```

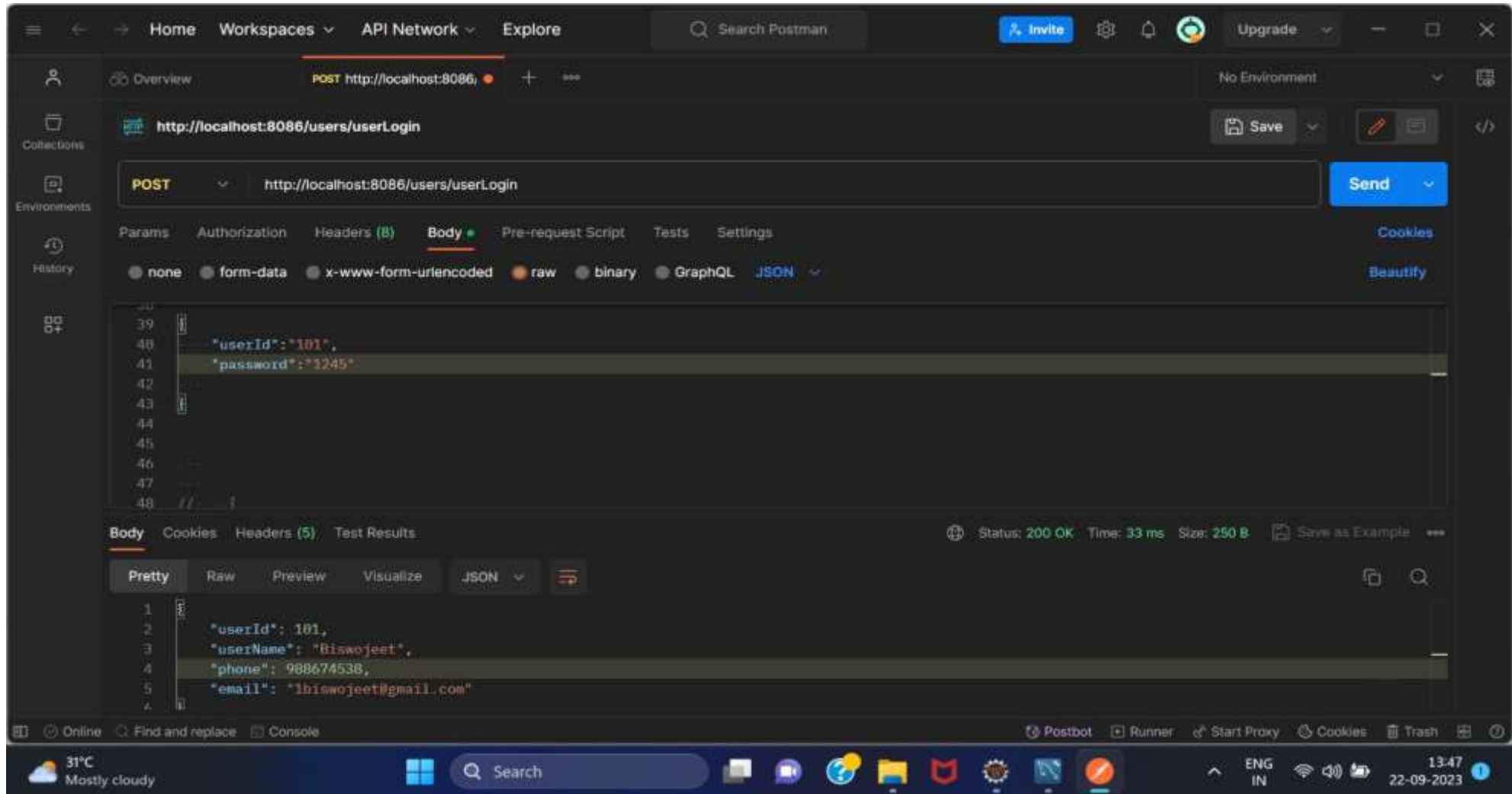
The bottom pane shows the response body in JSON format:

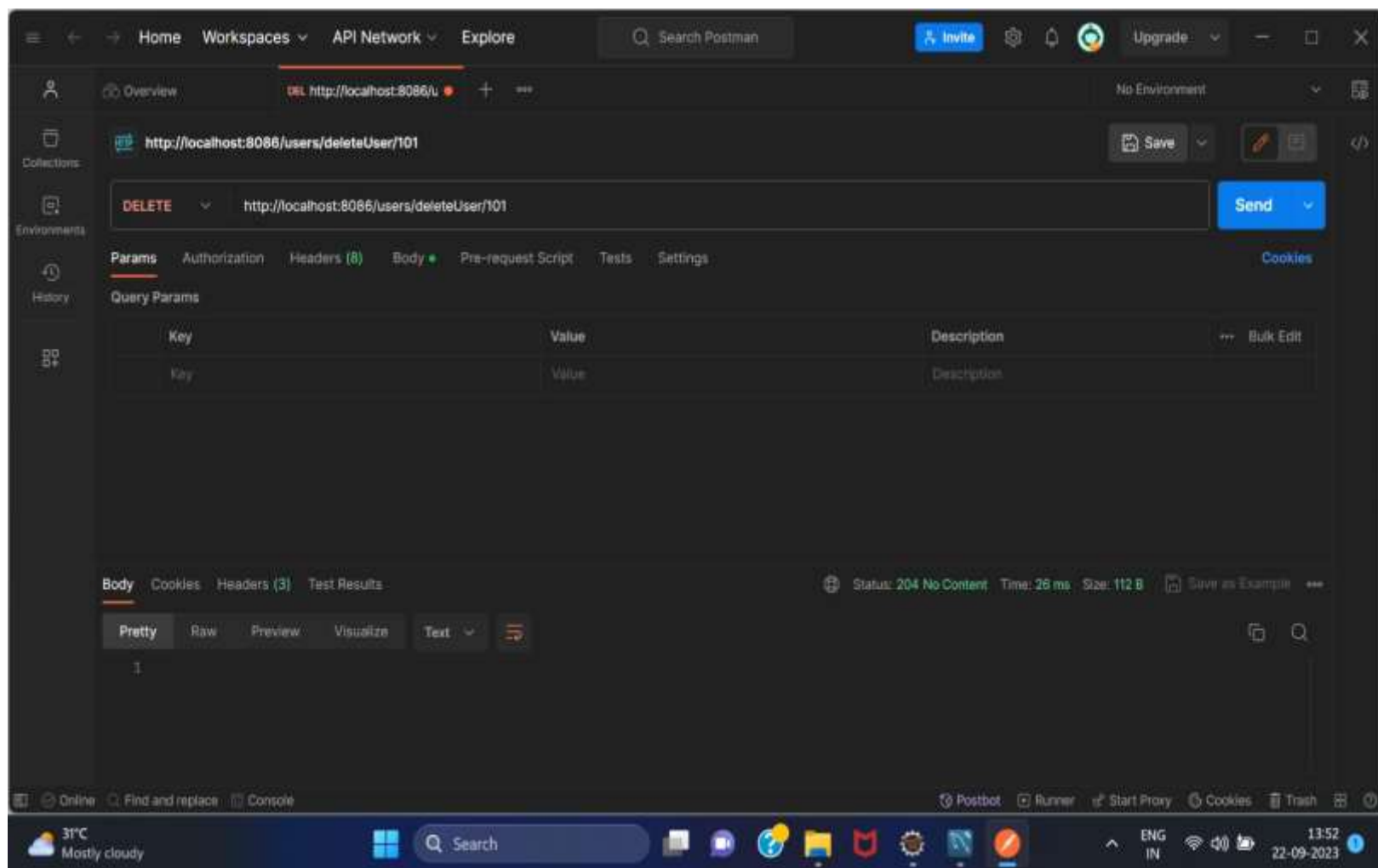
```
1 {  
2   "adminId": 567,  
3   "password": "admin@123",  
4   "adminName": "admin123"  
5 }
```

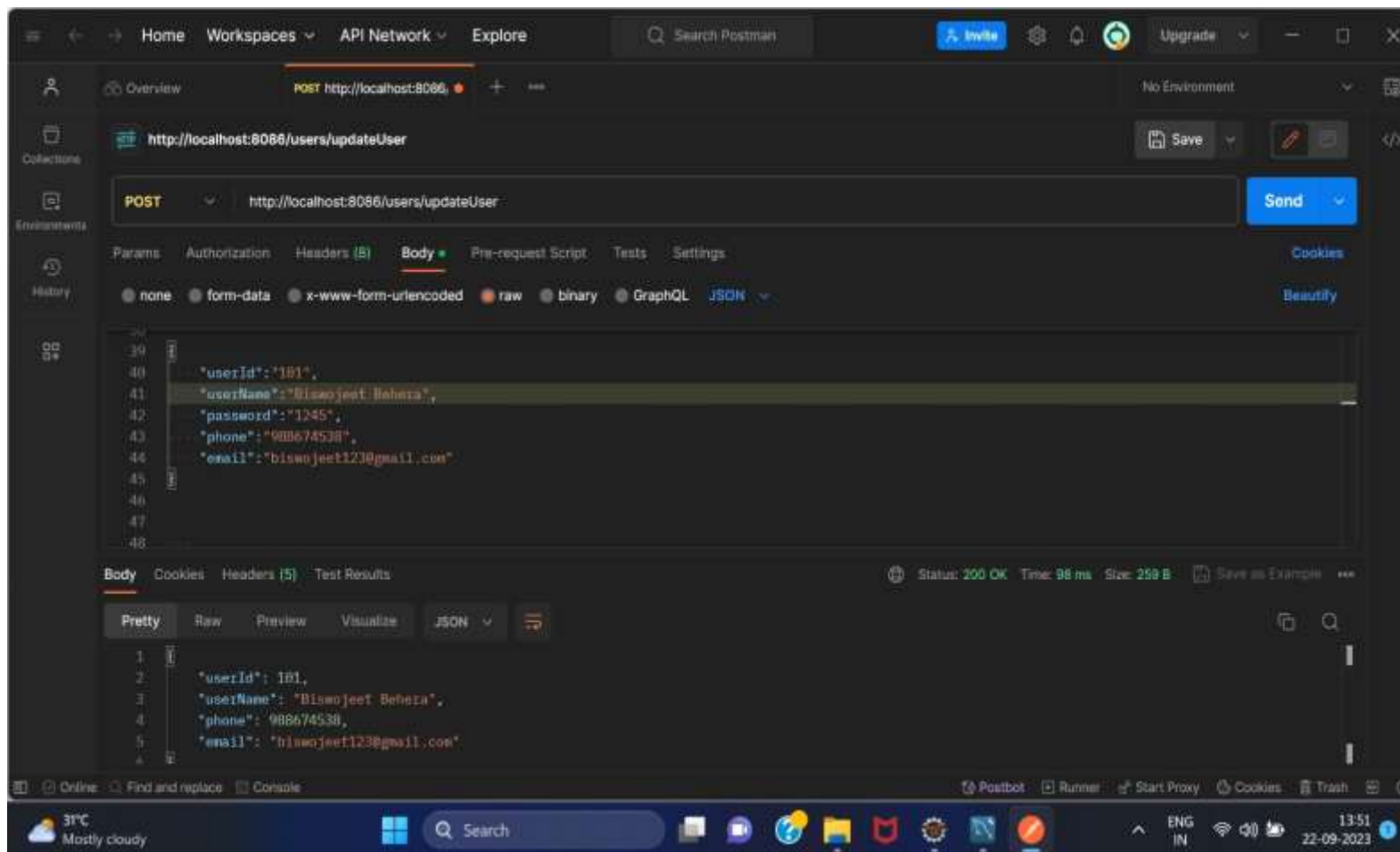
The status bar at the bottom indicates a successful response with status 200 OK, a time of 30 ms, and a size of 314 B. The Windows taskbar at the very bottom shows the system clock at 19:48 on 22-09-2023.

Postman Testing Of User Login

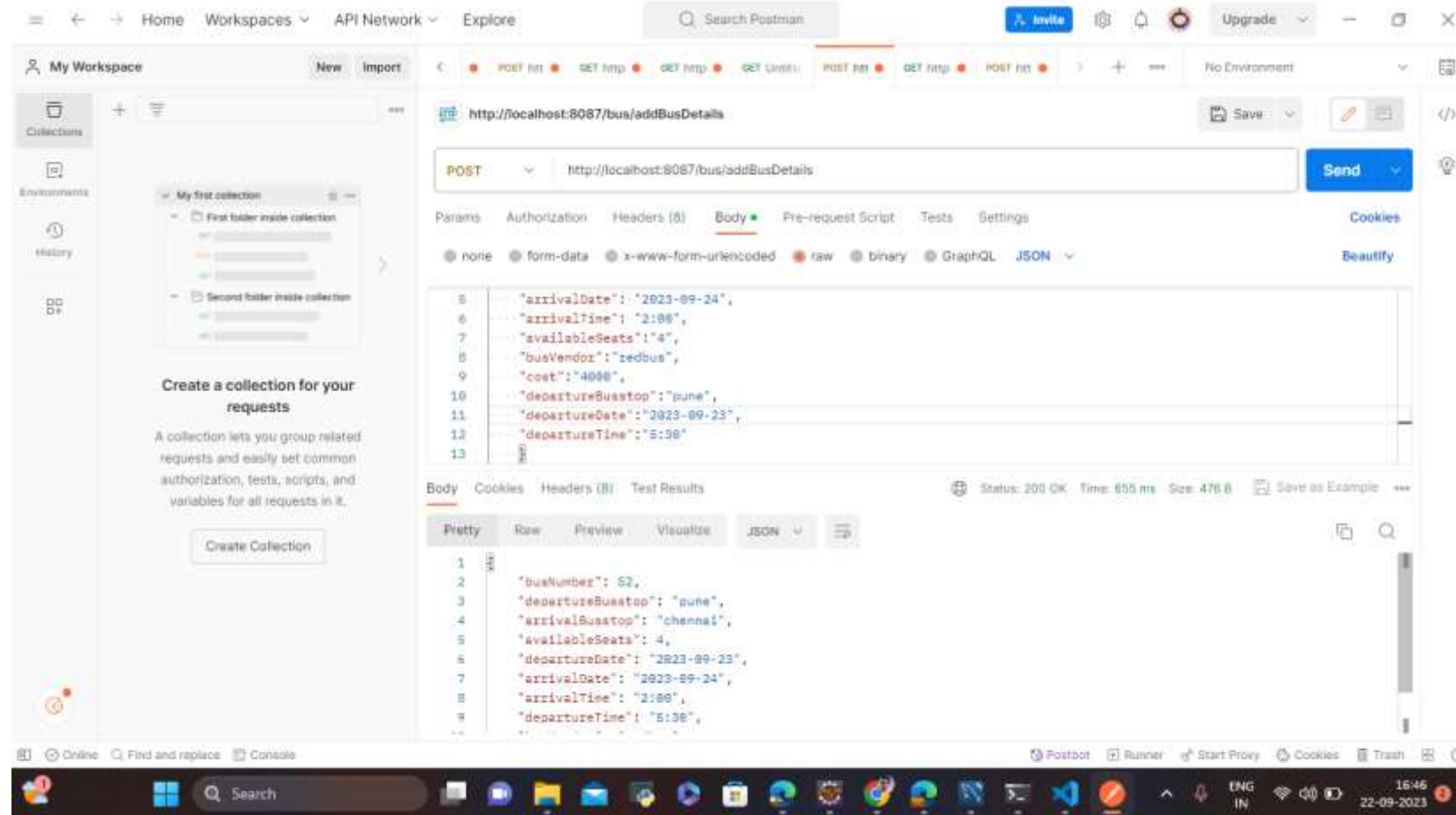


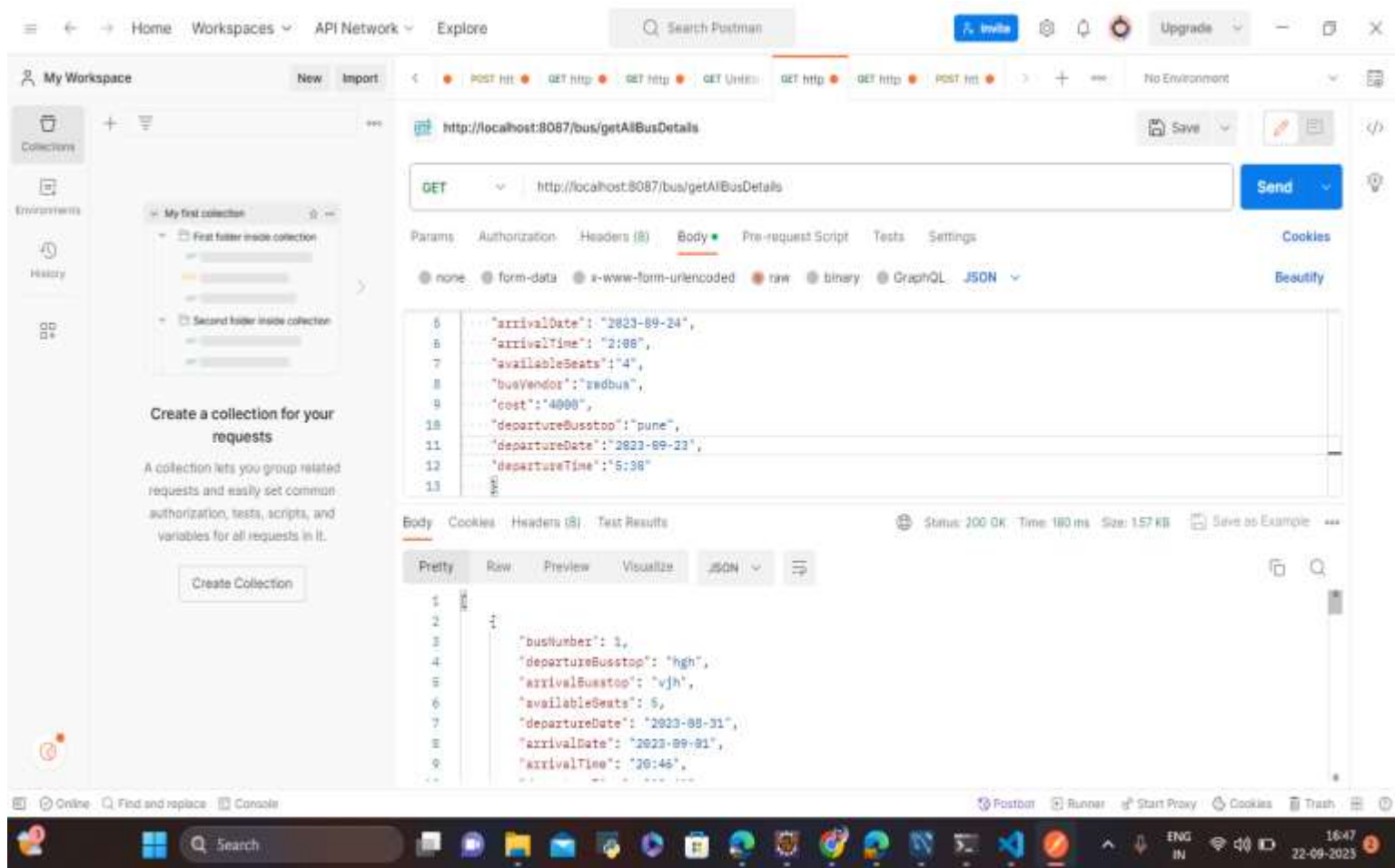






Postman Testing Of Bus Details





The screenshot displays the Postman API client interface. The top navigation bar includes links for Home, Workspaces, API Network, and Explore, along with a search bar and utility buttons like Invite, Upgrade, and a close button. The left sidebar shows the 'My Workspace' section with options for New, Import, Collections, Environments, and History. A collection named 'My first collection' is expanded, showing two folders: 'First folder inside collection' and 'Second folder inside collection'. Below the sidebar, a prompt encourages creating a collection for requests.

The main workspace shows a POST request to `http://localhost:8087/bus/updateBusDetails`. The request is configured with the following details:

- Method:** POST
- URL:** `http://localhost:8087/bus/updateBusDetails`
- Body:** JSON format, containing the following data:

```
1 {
2   "busNumber": 1,
3   "departureBusstop": "hyderabad",
4   "arrivalBusstop": "rajahmundry",
5   "arrivalDate": "2023-09-24",
6   "arrivalTime": "2:00",
7   "availableSeats": 4,
8   "departureDate": "2023-09-23",
9   "departureTime": "5:30",
10  }
```

The response is displayed in the bottom right pane, showing a status of 200 OK, a time of 107 ms, and a size of 484 B. The response body is formatted as JSON:

```
1 {
2   "busNumber": 1,
3   "departureBusstop": "hyderabad",
4   "arrivalBusstop": "rajahmundry",
5   "availableSeats": 4,
6   "departureDate": "2023-09-23",
7   "arrivalDate": "2023-09-24",
8   "arrivalTime": "2:00",
9   "departureTime": "5:30",
10  }
```

The bottom status bar includes links for Online, Find and replace, Console, Postbot, Runner, Start Proxy, Cookies, Trash, and a help icon. The system tray at the bottom shows the date and time as 16:48 on 22-09-2023.

The screenshot displays the Postman application interface. The top navigation bar includes 'Home', 'Workspaces', 'API Network', and 'Explore'. A search bar labeled 'Search Postman' is present. On the left sidebar, there are sections for 'My Workspace' (with 'New' and 'Import' buttons), 'Collections' (showing a tree view of 'My first collection' with sub-folders and requests), 'Environments', 'History', and a 'Create a collection for your requests' button. The main workspace shows a 'DELETE' request to 'http://localhost:8087/bus/deleteBusDetails/2'. The 'Body' tab is selected, showing a JSON payload with bus details. The bottom status bar indicates a successful response: 'Status: 200 OK', 'Time: 66 ms', 'Size: 212 B'. The Windows taskbar at the bottom shows the system clock as 16:49 on 22-09-2023.

My Workspace New Import

Search Postman Invite Upgrade

Home Workspaces API Network Explore

My first collection

- First folder inside collection
 - GET
 - POST
 - GET
- Second folder inside collection
 - GET

Create a collection for your requests

A collection lets you group related requests and easily set common authorization, tests, scripts, and variables for all requests in it.

Create Collection

http://localhost:8087/bus/deleteBusDetails/2

DELETE http://localhost:8087/bus/deleteBusDetails/2 Send

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded **raw** binary GraphQL JSON Beautify

```
3 busNumber: 1,  
4 arrivalBusstop: "rajahmundry",  
5 arrivalDate: "2023-09-24",  
6 arrivalTime: "2:00",  
7 availableSeats: "4",  
8 busVendor: "redbus",  
9 cost: "1200",  
10 departureBusstop: "hyderabad",  
11 departureDate: "2023-09-23",  
12 departureTime: "5:30"
```

Body Cookies Headers (7) Test Results Status: 200 OK Time: 66 ms Size: 212 B Save as Example

Pretty Raw Preview Visualize Text 1

Online Find and replace Console Postbot Runner Start Proxy Cookies Trash

Search 16:49 22-09-2023

The screenshot displays the Postman application interface. At the top, there's a navigation bar with 'Home', 'Workspaces', 'API Network', and 'Explore'. A search bar labeled 'Search Postman' is also present. On the right, there are buttons for 'Invite', 'Upgrade', and a dropdown menu.

The main workspace is titled 'My Workspace' and includes 'New' and 'Import' buttons. The left sidebar shows 'Collections', 'Environments', and 'History'. A 'Your collection' panel is open, showing an 'Authorization' tab with 'API Key' selected. Below this, a message says 'Create a collection for your requests' with a 'Create Collection' button.

The central area shows a POST request to 'http://localhost:8086/user/addBooking/677/7531'. The request body is in JSON format, containing booking details. The 'Body' tab is selected, showing the raw JSON data. Below the request, the 'Test Results' tab is active, displaying the response body in JSON format.

Request Body (JSON):

```
{
  "bookingId": 1,
  "bookingDate": "2023-09-30",
  "bookingTime": "10:00 AM",
  "totalCost": 50.0,
  "busNumber": 123,
  "ownerId": 1,
  "passengers": [
    {
      "name": "Passenger1",
    }
  ]
}
```

Response Body (JSON):

```
{
  "bookingId": 9205,
  "bookingDate": "2023-09-21",
  "bookingTime": "15:59",
  "totalCost": 4000.0,
  "busNumber": 7531,
  "ownerId": 677,
  "passengers": [
    {
      "passengerId": 7,
    }
  ]
}
```

The status bar at the bottom indicates 'Status: 200 OK', 'Time: 124 ms', and 'Size: 400 B'. There are also buttons for 'Save as Example' and a search icon.

The screenshot displays the Postman API client interface. At the top, there are navigation links: Home, Workspaces, API Network, and Explore. A search bar labeled "Search Postman" is present. On the right, there are buttons for "Invite", settings, a bell icon, a circular logo, and an "Upgrade" button.

The left sidebar shows "My Workspace" with "New" and "Import" buttons. Below this are sections for "Collections", "Environments", "History", and a "Create a collection for your requests" prompt. The prompt includes the text: "A collection lets you group related requests and easily set common authorization, tests, scripts, and variables for all requests in it." and a "Create Collection" button.

The main area shows a request configuration for a DELETE method to the URL `http://localhost:8086/user/deleteBooking/6752/677`. The request is saved in "No Environment". The "Params" tab is active, showing a table for Query Params:

	Key	Value	Description	...	Bulk Edit
	Key	Value	Description		

Below the table, the "Body" tab is active, showing the response status: "Status: 200 OK", "Time: 238 ms", and "Size: 131 B". There is a "Save as Example" button. The response is displayed in "Pretty" format, showing the number "1".

The screenshot displays the Postman API client interface. The top navigation bar includes links for Home, Workspaces, API Network, and Explore, along with a search bar and utility buttons like 'Invite', 'Upgrade', and 'No Environment'. The left sidebar shows the 'My Workspace' section with a 'Collections' panel containing 'My first collection' and a 'Create a collection for your requests' button. The main workspace shows a GET request to `http://localhost:8086/user/findBus/mumbai/delhi/2023-09-28`. The 'Query Params' table is empty. The 'Body' tab is selected, showing a JSON response in 'Pretty' format. The response status is 200 OK, with a time of 22 ms and a size of 478 B. The JSON body contains bus details.

Query Params

Key	Value	Description
Key	Value	Description

Body | Cookies | Headers (8) | Test Results

Status: 200 OK | Time: 22 ms | Size: 478 B | Save as Example

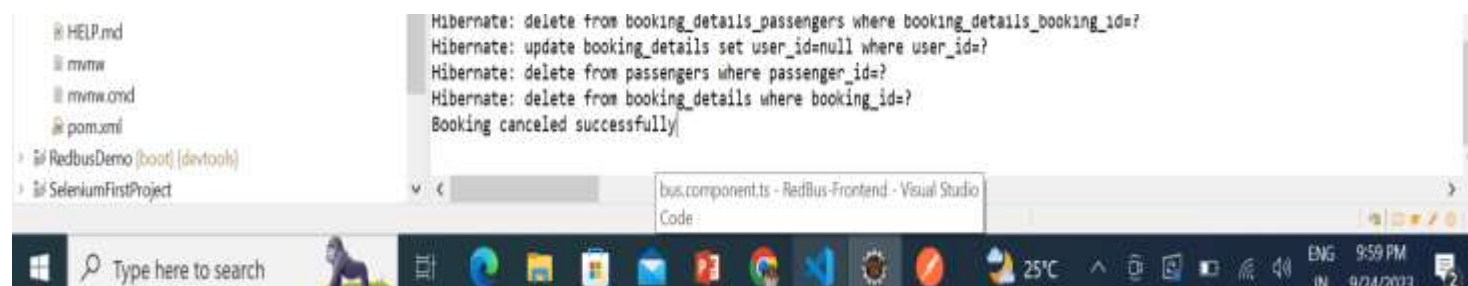
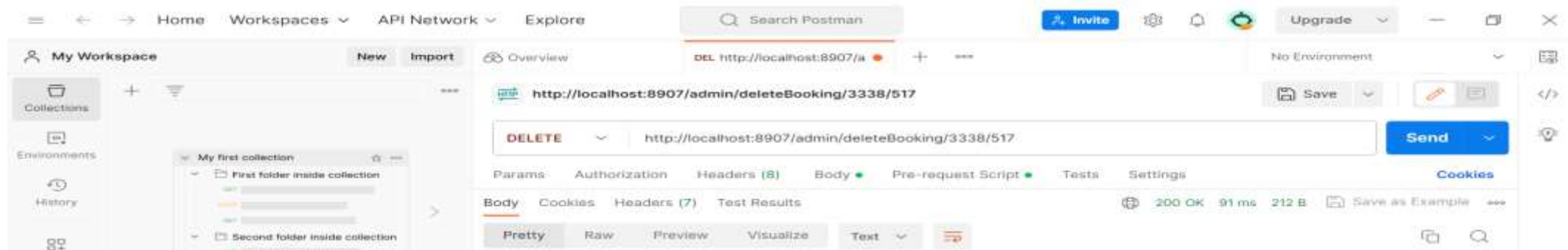
```
1 {
2   "busNumber": 6651,
3   "departureBusstop": "delhi",
4   "arrivalBusstop": "mumbai",
5   "availableSeats": 20,
6   "departureDate": "2023-09-28",
7   "arrivalDate": "2023-09-29",
8   "arrivalTime": "08:30",
9   "departureTime": "19:27",
10  "busVendor": "Ram",
11  "cost": 3500.0
12 }
```

Postman Testing of View all Booking Details

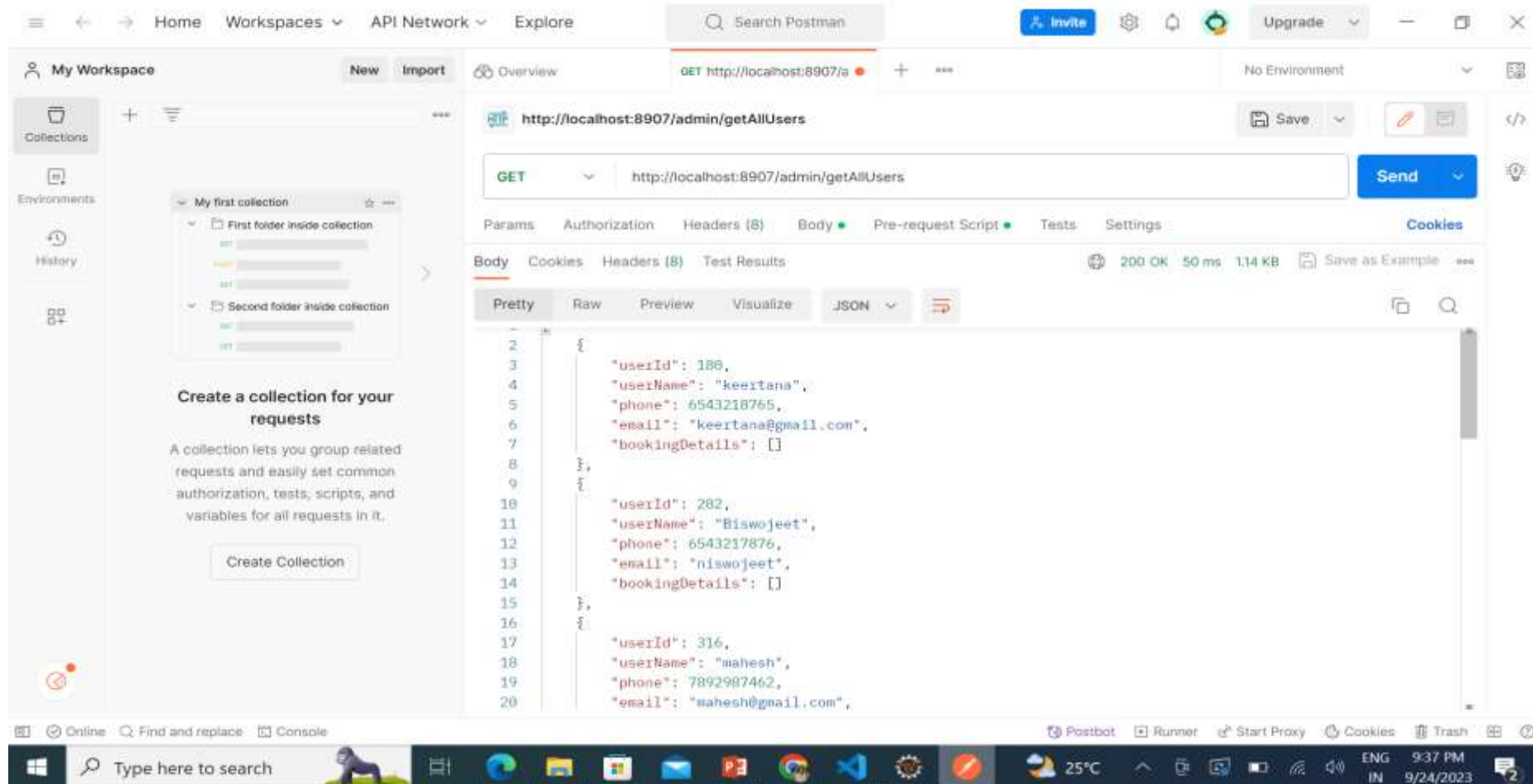
The screenshot displays the Postman application interface. On the left, the 'My Workspace' sidebar shows a collection named 'My first collection' with two folders: 'First folder inside collection' and 'Second folder inside collection'. Below this, a prompt says 'Create a collection for your requests' with a 'Create Collection' button. The main area shows a GET request to 'http://localhost:8907/admin/getBookingByUser/517'. The request is saved and has a 'Send' button. Below the URL bar, tabs for 'Params', 'Authorization', 'Headers (8)', 'Body', 'Pre-request Script', 'Tests', and 'Settings' are visible. The 'Body' tab is selected, showing a JSON response in 'Pretty' format. The response status is '200 OK' with a response time of '33 ms' and a size of '452 B'. The JSON data is as follows:

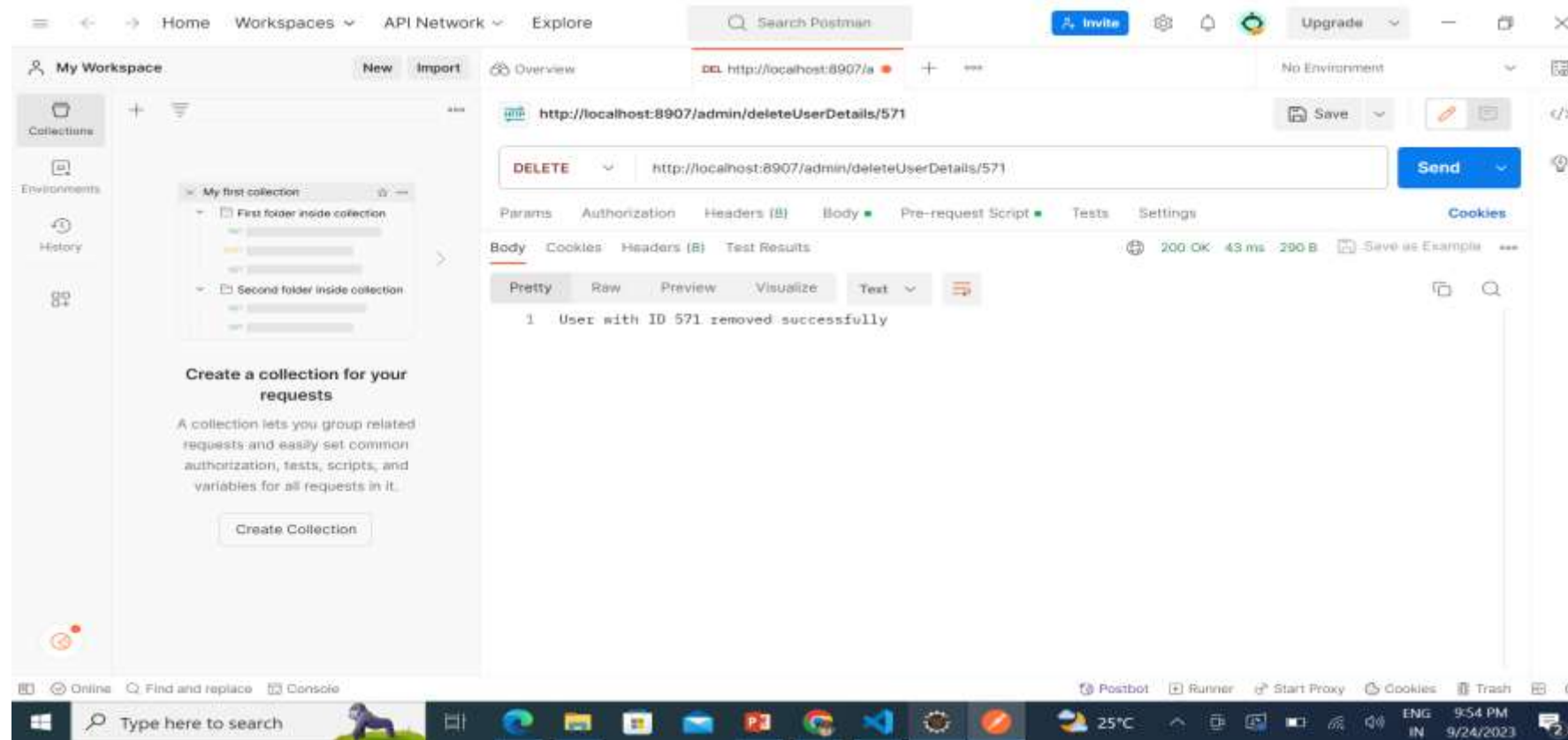
```
1 {
2   "bookingId": 3338,
3   "bookingDate": "2023-09-24",
4   "bookingTime": "21:34",
5   "totalCost": 3400.0,
6   "busNumber": 9348,
7   "ownerId": 517,
8   "passengers": [
9     {
10      "passengerId": 13,
11      "name": "Navya Manju M",
12      "age": 24,
13      "luggage": 2.0
14    }
15  ]
16 }
17
18 }
```

The bottom of the screen shows the Windows taskbar with the search bar and various application icons. The system tray on the right indicates the date and time as '9:35 PM 9/24/2023'.

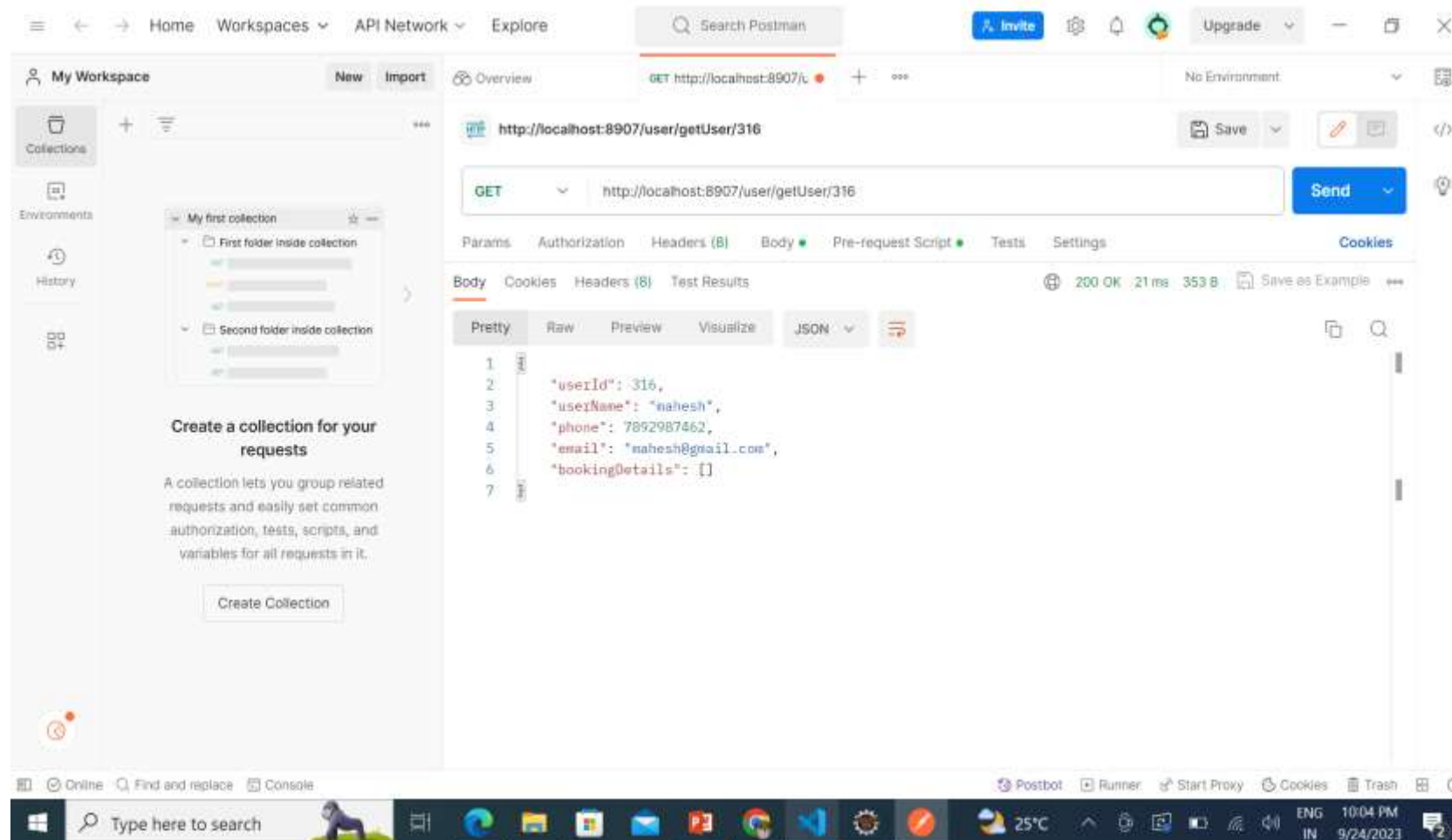


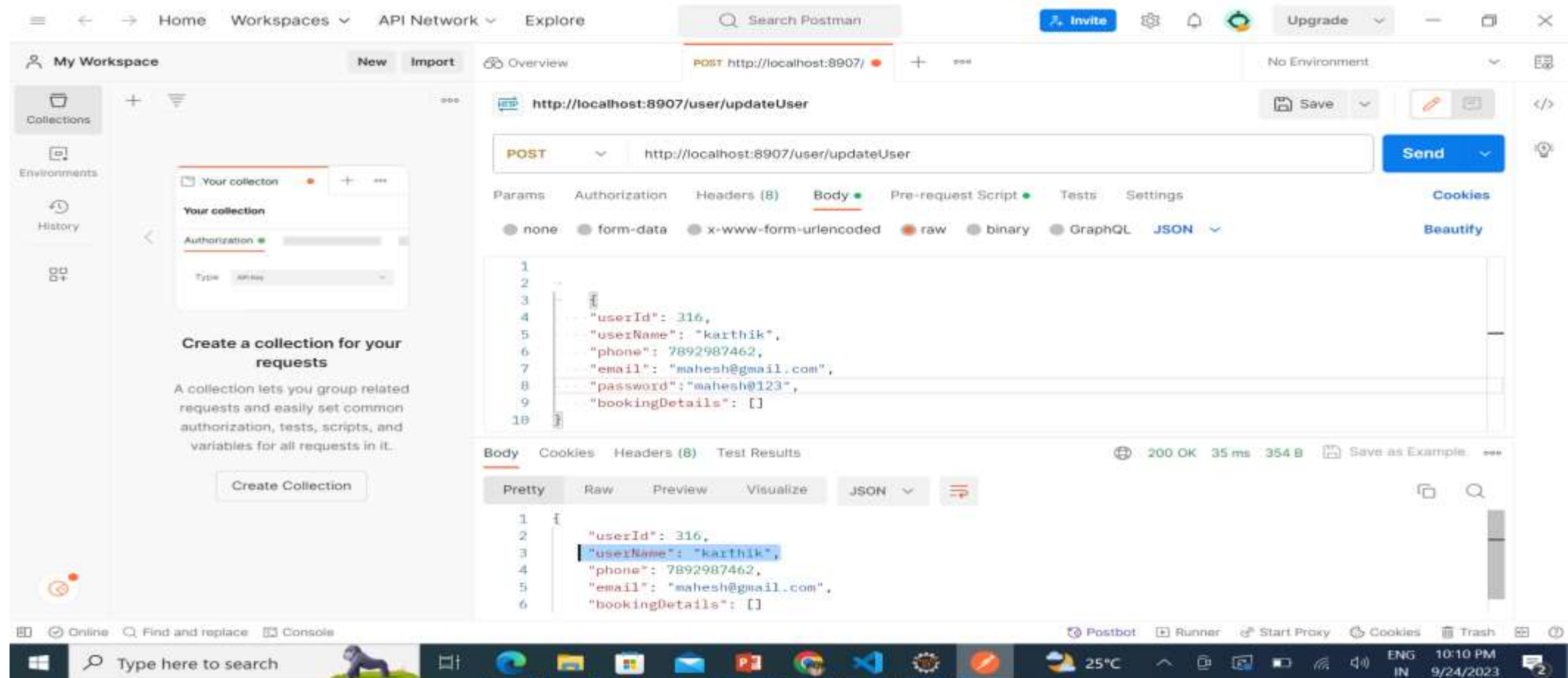
Postman Testing Of View Registered User

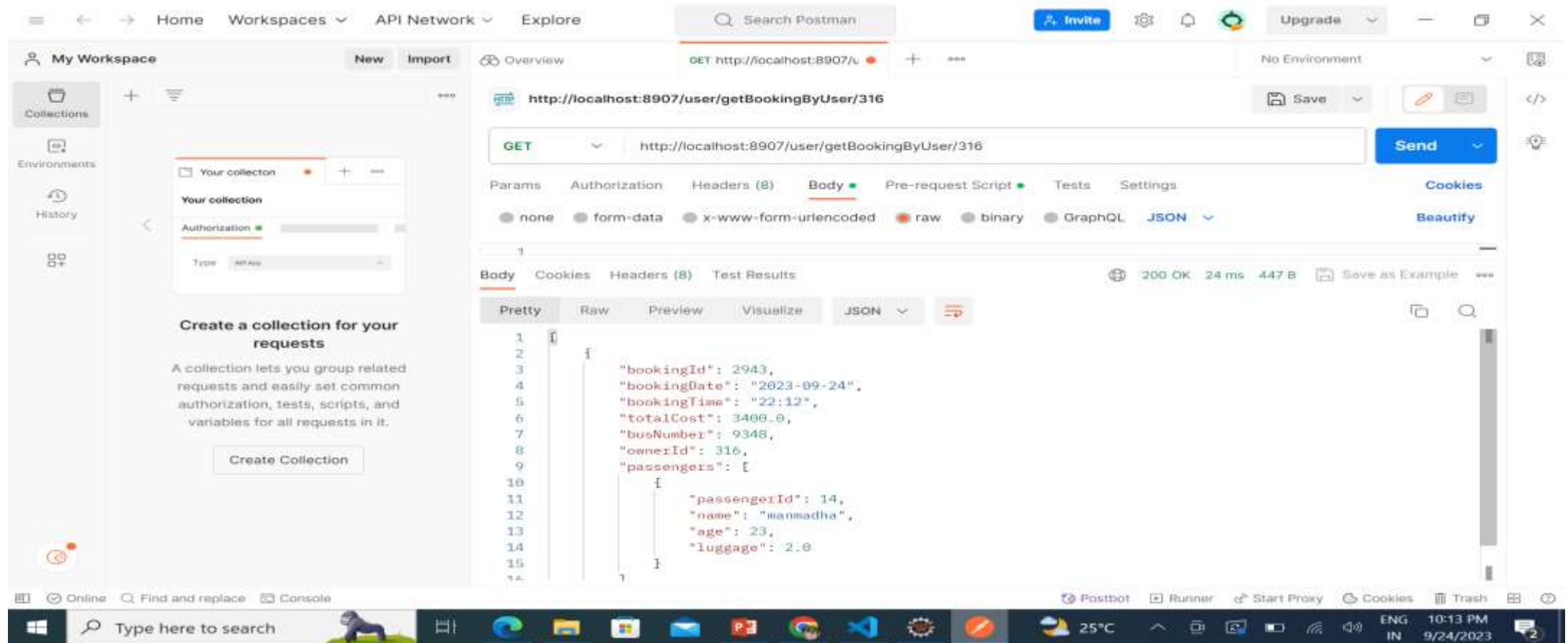


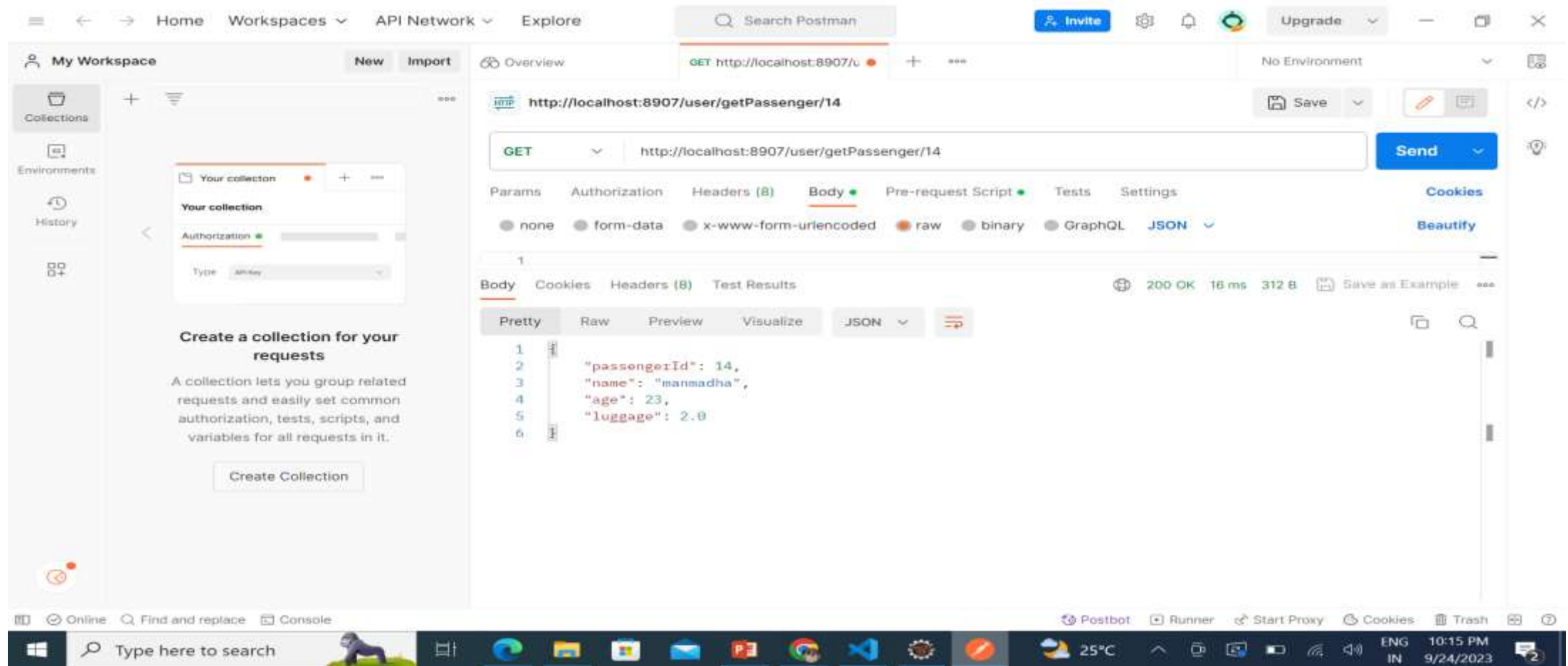


Postman Testing Of View all User







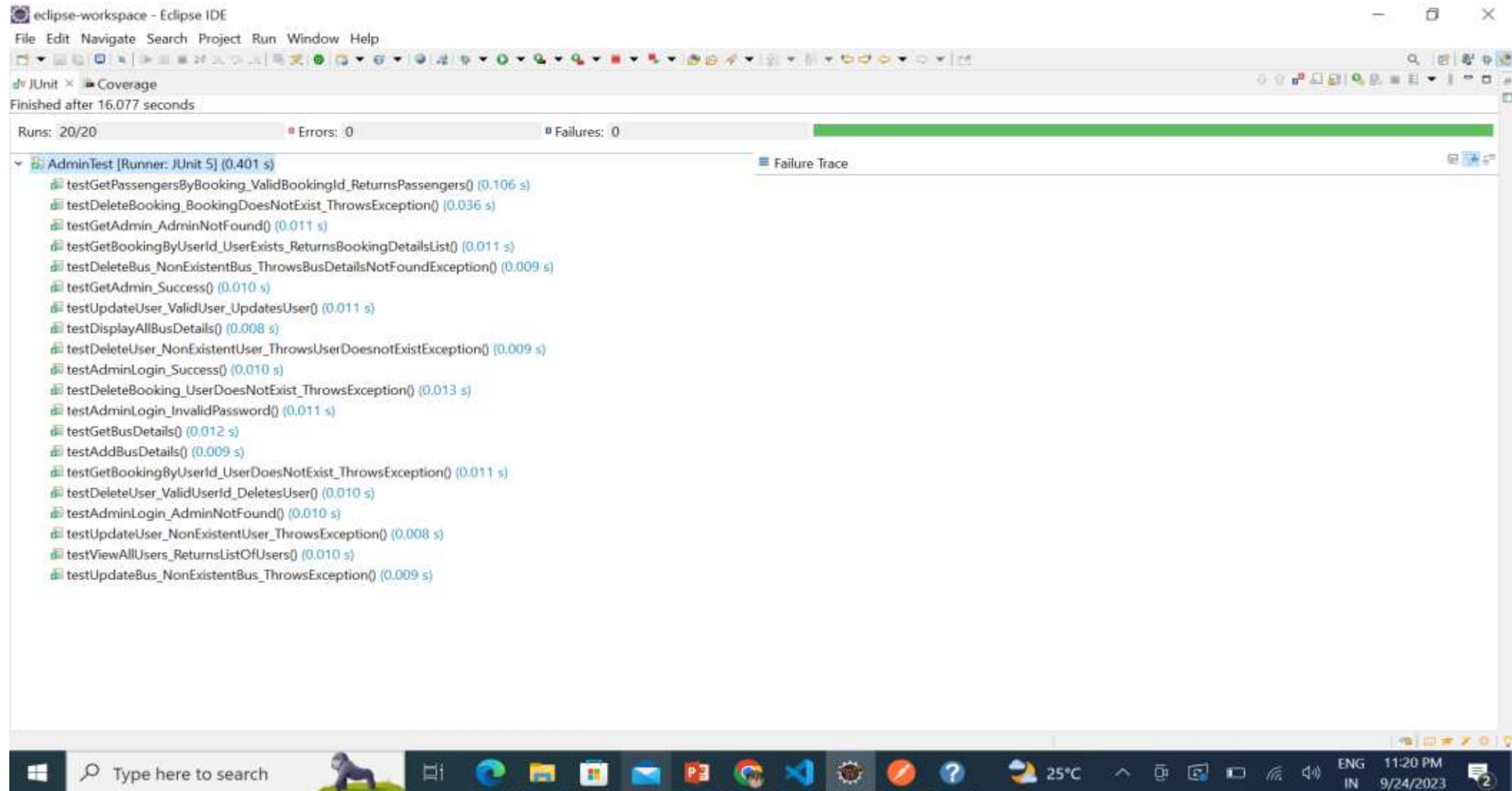


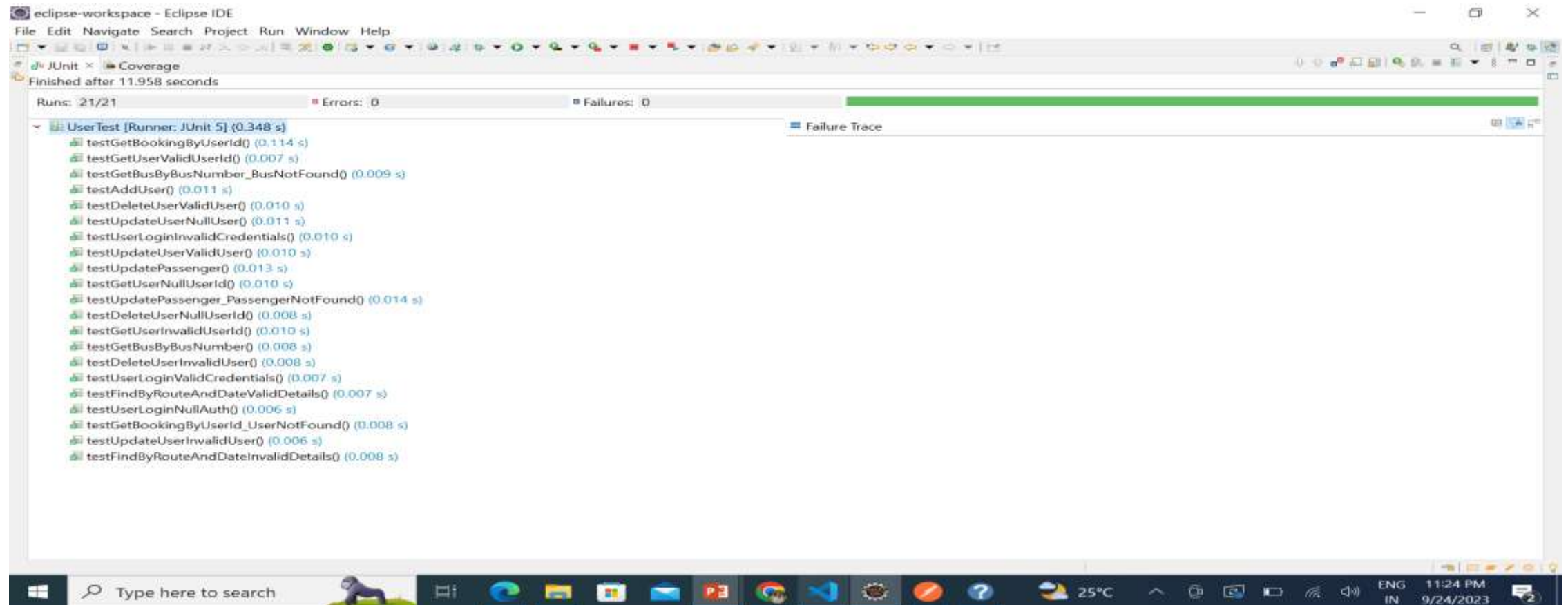
Postman Testing of View bus Details(User-Side)

The screenshot displays the Postman application interface. On the left sidebar, the 'My Workspace' section is active, showing a 'Your collection' dialog box with 'Authorization' set to 'API key'. Below this, a prompt says 'Create a collection for your requests' with a 'Create Collection' button. The main workspace shows a GET request to `http://localhost:8907/user/getBusByNumber/9348`. The request is saved, and the 'Body' tab is selected, showing a JSON response. The response status is 200 OK, with a response time of 36 ms and a body size of 485 B. The JSON response is as follows:

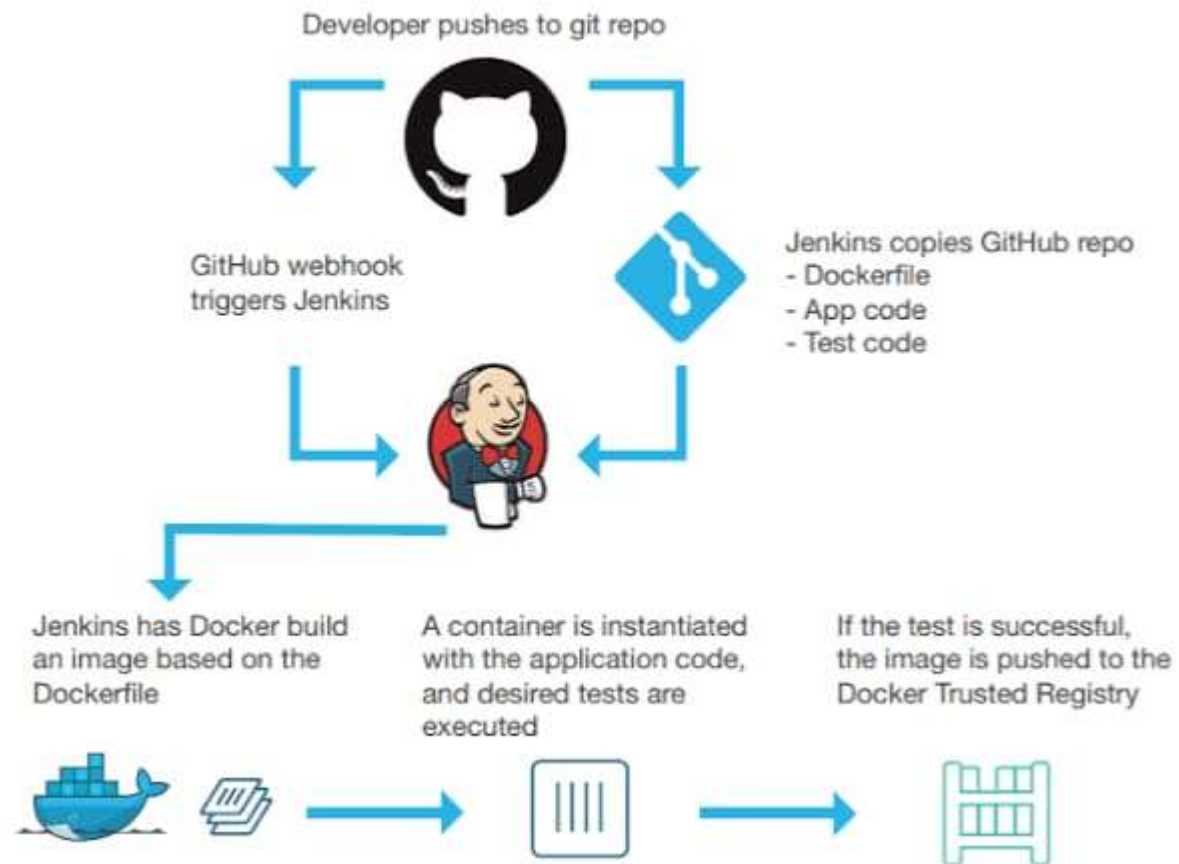
```
1 {
2   "busNumber": 9348,
3   "departureBusstop": "chennai",
4   "arrivalBusstop": "himchala",
5   "availableSeats": 14,
6   "departureDate": "2023-09-25",
7   "arrivalDate": "2023-09-26",
8   "arrivalTime": "06:00",
9   "departureTime": "18:00",
10  "busVendor": "redbus",
11  "cost": 3400.0
12 }
```

The bottom of the screen shows the Windows taskbar with the search bar and various application icons. The system clock indicates 10:16 PM on 9/24/2023.

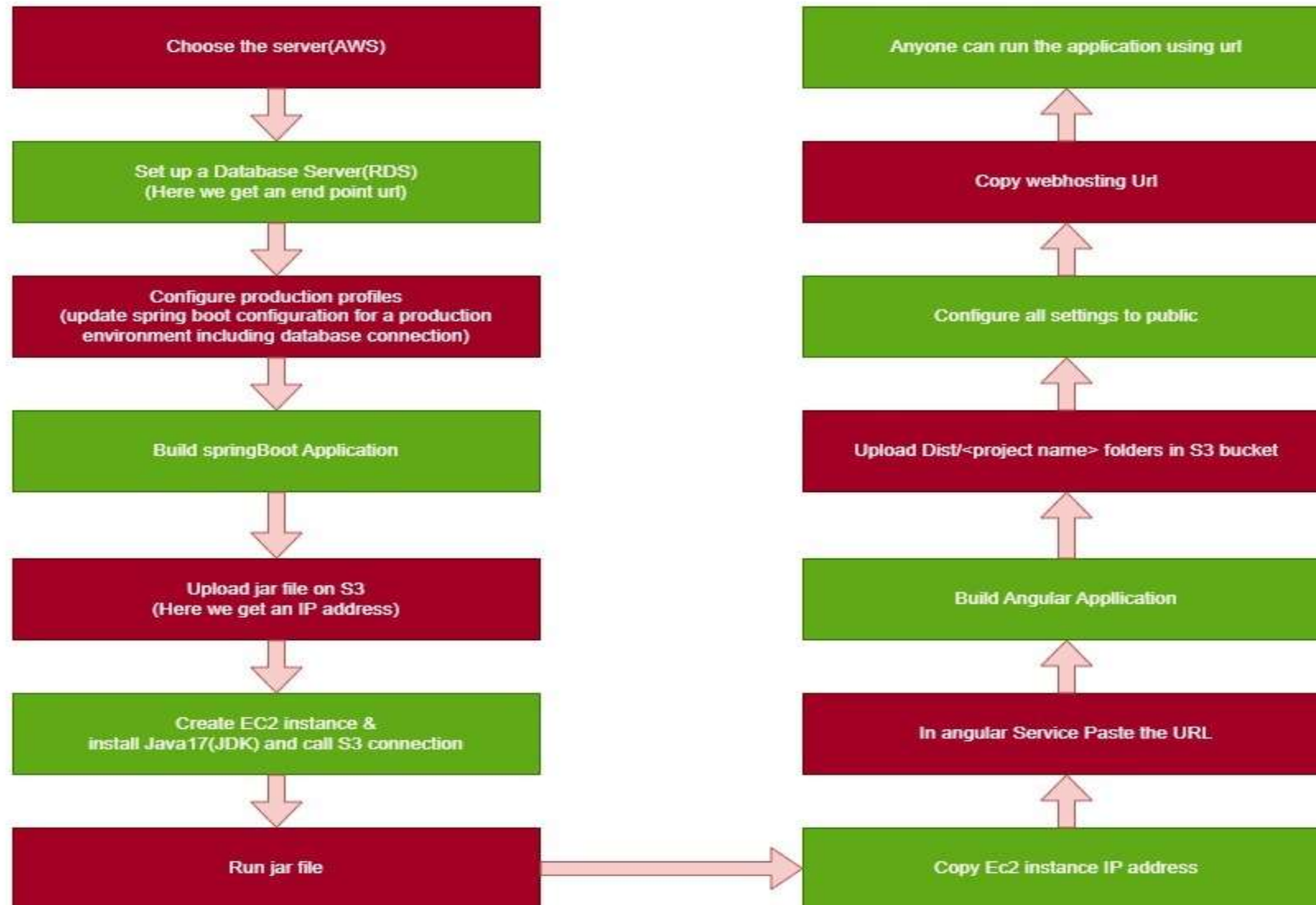




DEPLOYMENT ON JENKINS



DEPLOYMENT ON AWS



- RedBus travels is online bus booking system which provides online booking of bus.
- This system is user friendly and accurate.
- This system is efficient in reservation.
- The information about the user making a booking can be obtained by clicking a button, removing some of the difficulties that come with the manual system.



Thank
you