**MOVEINSYNC CASE STUDY ASSIGNMENT:**

BUS BOOKING SYSTEM

SNAPSEATS

**NAME:** YADLAPALLI SIDDHARTHA

**ROLL NO.:** 210001081

**PROJECT OVERVIEW:**

SnapSeats, is an innovative Bus Booking System, revolutionizes the travel reservation experience by providing a secure and efficient platform for passengers and administrators alike. With a strong emphasis on robust authentication, SnapSeats ensures a safe user environment.

SnapSeats, a secured bus booking system, features a user-friendly authentication process with dedicated login and sign-up pages. The admin dashboard empowers administrators to effortlessly manage buses—adding, deleting, and updating their details. Admins can also access the seating plans, visually identifying occupied and vacant seats. On the user dashboard, passengers can conveniently search for buses based on source, destination, and date. The availability of seats is visually represented using color codes: green for 60% or less occupancy, yellow for 60% to 90%, and red for over 90%, indicating almost full. After booking, users receive fare details and a confirmed ticket under their profile. SnapSeats facilitates ticket cancellations and provides a "My Trips" section for users to conveniently review their booked tickets. This comprehensive system ensures a seamless and intuitive bus booking experience for users.

**TOOLS AND FRAMEWORKS USED:**

* HTML, CSS, Javascript
* NodeJS, Express JS
* MySQL
* BootStrap

**FUNCTIONALITIES INCLUDED:**

* AUTHENTICATION:

The system's authentication process is managed through dedicated login and sign-in pages. Upon initiating a connection with the database server, the login page captures user email and password information. Subsequently, the system verifies these credentials in the user table. If a match is identified, the user is redirected to the user dashboard, and a boolean variable, **user\_login**, is set to true. This variable becomes instrumental in regulating access to other pages, ensuring that users must be logged in to navigate through the system.

In cases where user credentials are not located in the user table, the system then queries the admin table. If the requisite details are found in the admin table, the system redirects the user to the admin dashboard. Simultaneously, it sets the **admin\_login** variable to true, serving as a key determinant for accessing various admin-specific pages within the system. This multifaceted authentication mechanism enhances the security and segregation of user and admin functionalities.

Regarding the sign-in page, it collects user-provided information including name, contact details, email, and password, and subsequently inserts these details into the user table. Following the successful sign-in process, the system dynamically redirects the user to the appropriate page based on their user type. This streamlined approach ensures an efficient and personalized onboarding experience for users.

* **ADMIN DASHBOARD**

The admin dashboard serves as the central hub for administrators to manage and monitor the bus services. The following key features are available in the admin dashboard.

* + - **ADDING A NEW BUS ROUTE:** The admin can add a new bus to the system by providing relevant details such as bus number, route, departure and arrival times, and seating capacity. Each bus is assigned a unique bus number for easy identification.
    - **DELETE EXISTING BUSES:** The admin has the authority to delete existing buses from the system. This functionality is useful for removing buses that are no longer in service or undergoing maintenance.
    - **SEATING INFORMATION:** The admin can view the seating arrangement of a specific bus by entering its bus number. The seating information includes the layout of seats, seat numbers, and the current booking status (occupied or available).
* **USER DASHBOARD:**

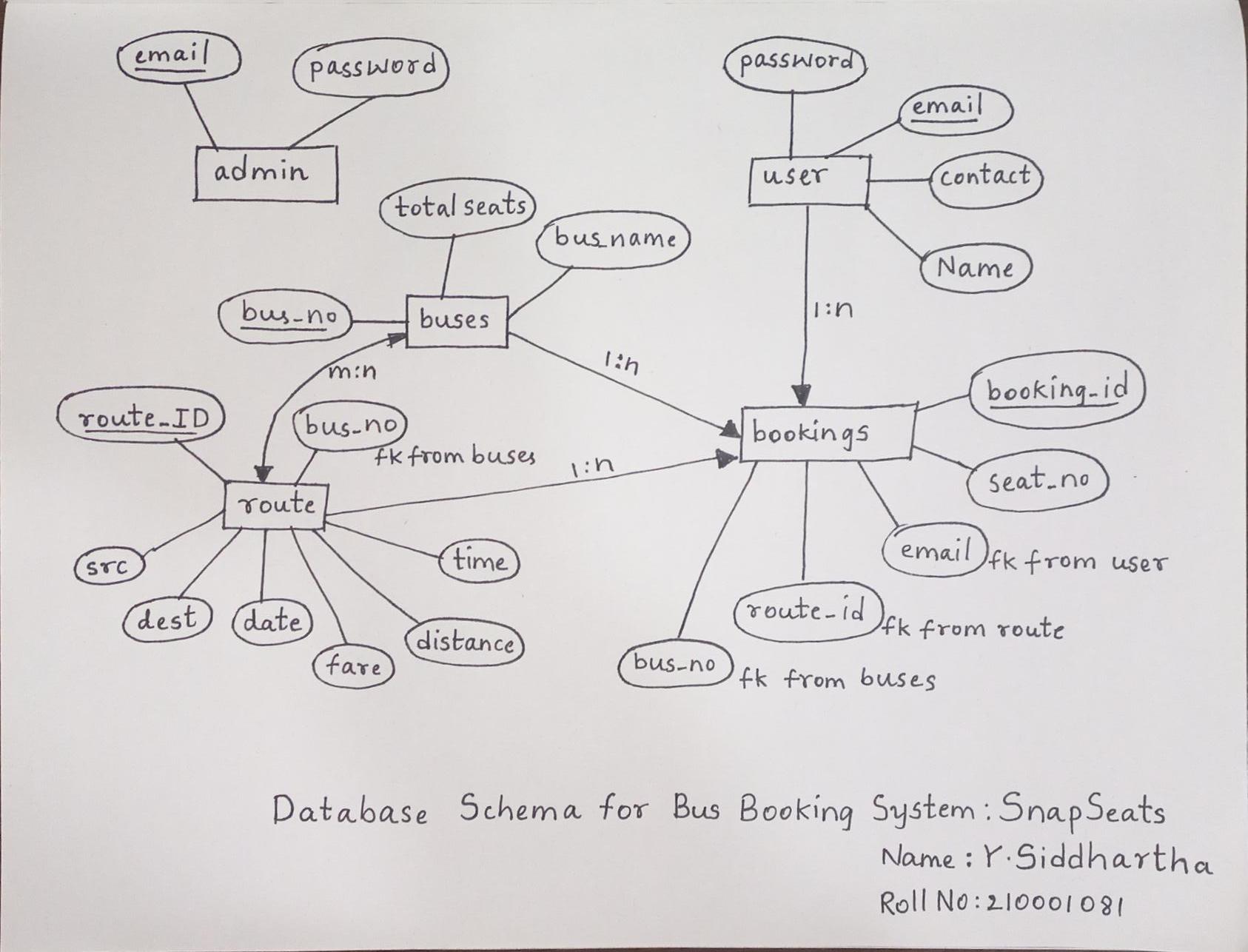
The user dashboard is designed to provide a user-friendly interface for customers to search for bus services, make bookings, and manage their reservations. The following features are available in the user dashboard.

* + - **SEARCH FOR BUS SERVICES:** Users can search for bus services based on criteria such as departure and arrival locations, date. The search results display available buses, their details, and the corresponding ticket prices.
    - **MAKE BOOKINGS:** Users can select a preferred bus from the search results and proceed to book seats for their journey. The system ensures real-time seat availability updates and allows users to choose their preferred seats.

**ADDITIONAL FUNCTIONALITIES:**

1. The color indication feature is integrated into the bus card within the user dashboard. A green bar is displayed when the occupancy is below 60%, signaling ample availability. A yellow color bar appears for occupancy levels between 60% and 90%, serving as a cautionary indicator for users to note moderate filling. The presence of a red bar, triggered when occupancy exceeds 90%, indicates that the bus is almost full, prompting users to act promptly.
2. The user has the flexibility to select a preferred seat number, as opposed to the system automatically assigning a specific distinctive seat to them.
3. The system enhances user experience by presenting a convenient pop-up displaying comprehensive ticket details, including passenger information and the source-destination particulars. Moreover, users have the added convenience of **downloading the ticket in PDF format** if they wish to retain a digital copy for their records or convenience. This dual functionality not only provides immediate access to essential trip information but also offers users the flexibility to manage their tickets in a format of their preference.

**DATABASE SCHEMA:**



The backend schema of the snapseats is defined as snapseats\_database in the MySQL. It has five entity sets, admin, user, buses, bookings, routes.

**User Table:**

**email (Primary Key):** Unique identifier for each user.

**password**: Stores the user's password securely.

**name:** Holds the user's name.

**contact:** Stores the user's contact number.

**Primary Key Constraint:** Ensures that each user is uniquely identified by their email address.

**Admin Table:**

**email (Primary Key):** Unique identifier for each admin.

**password:** Stores the admin's password securely.

**Primary Key Constraint:** Ensures that each admin is uniquely identified by their email address.

**Buses Table:**

**bus\_no (Primary Key):** Unique identifier for each bus.

**bus\_name:** Holds the name of the bus.

**total\_seats:** Indicates the total number of seats in the bus.

**Primary Key Constraint**: Ensures that each bus is uniquely identified by its bus number.

**Route Table:**

**route\_id (Primary Key, Auto Increment):** Unique identifier for each route.

**src:** Source location of the bus route.

**dest**: Destination location of the bus route.

**journey\_date:** Date of the journey.

**fare:** Fare for the journey.

**distance:** Distance covered by the route.

**journey\_time:** Duration of the journey.

**bus\_no (Foreign Key):** Links to the bus\_no in the Buses table.

**Primary Key Constraint:** Ensures that each route is uniquely identified by its route\_id.

**Foreign Key Constraint:** Connects the bus\_no in the Route table to the bus\_no in the Buses table.

**Bookings Table:**

**booking\_id (Primary Key, Auto Increment):** Unique identifier for each booking.

**bus\_no (Foreign Key):** Links to the bus\_no in the Buses table.

**seat\_no:** Indicates the seat number booked.

**route\_id (Foreign Key):** Links to the route\_id in the Route table.

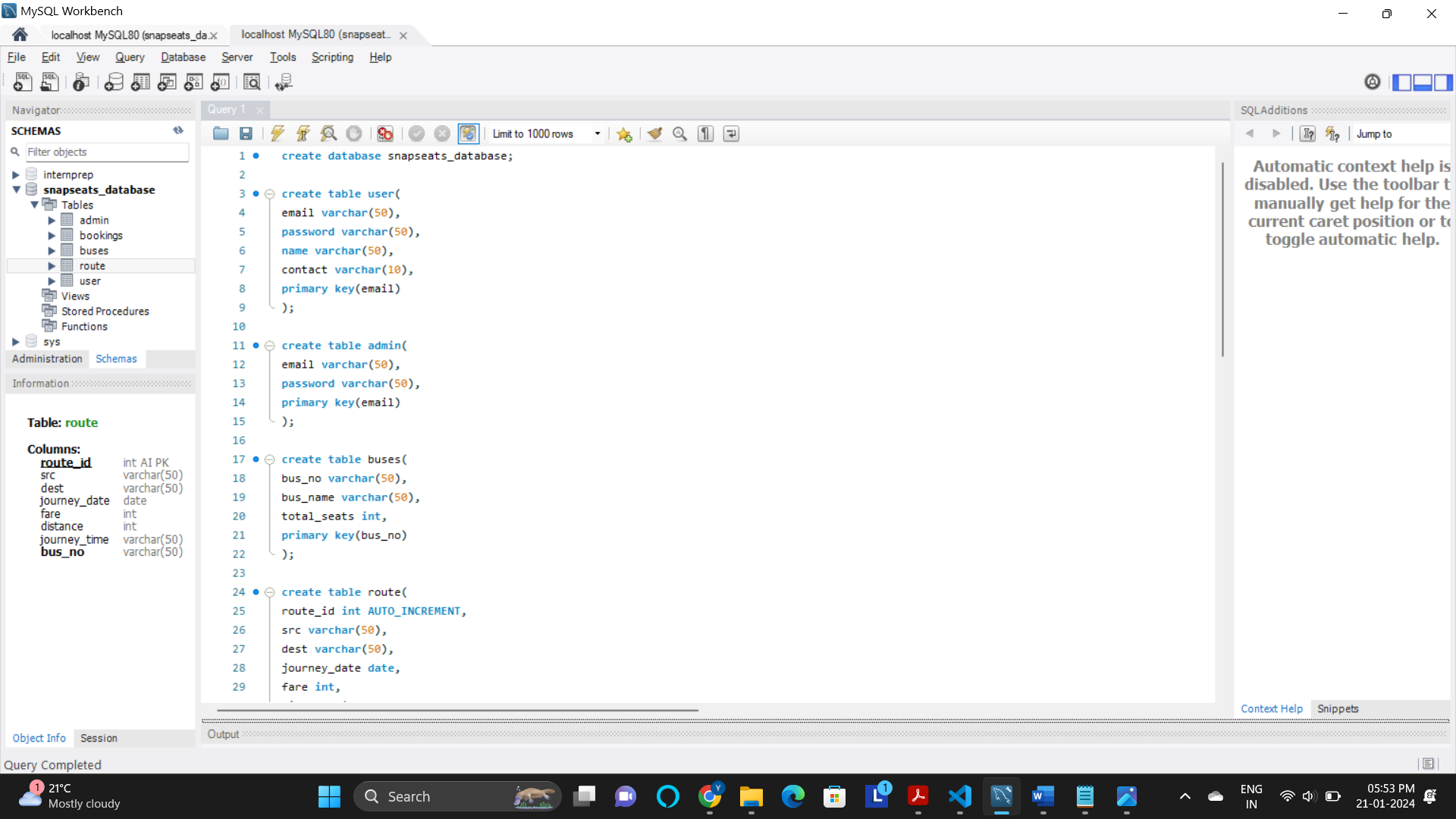
**email (Foreign Key):** Links to the email in the User table.

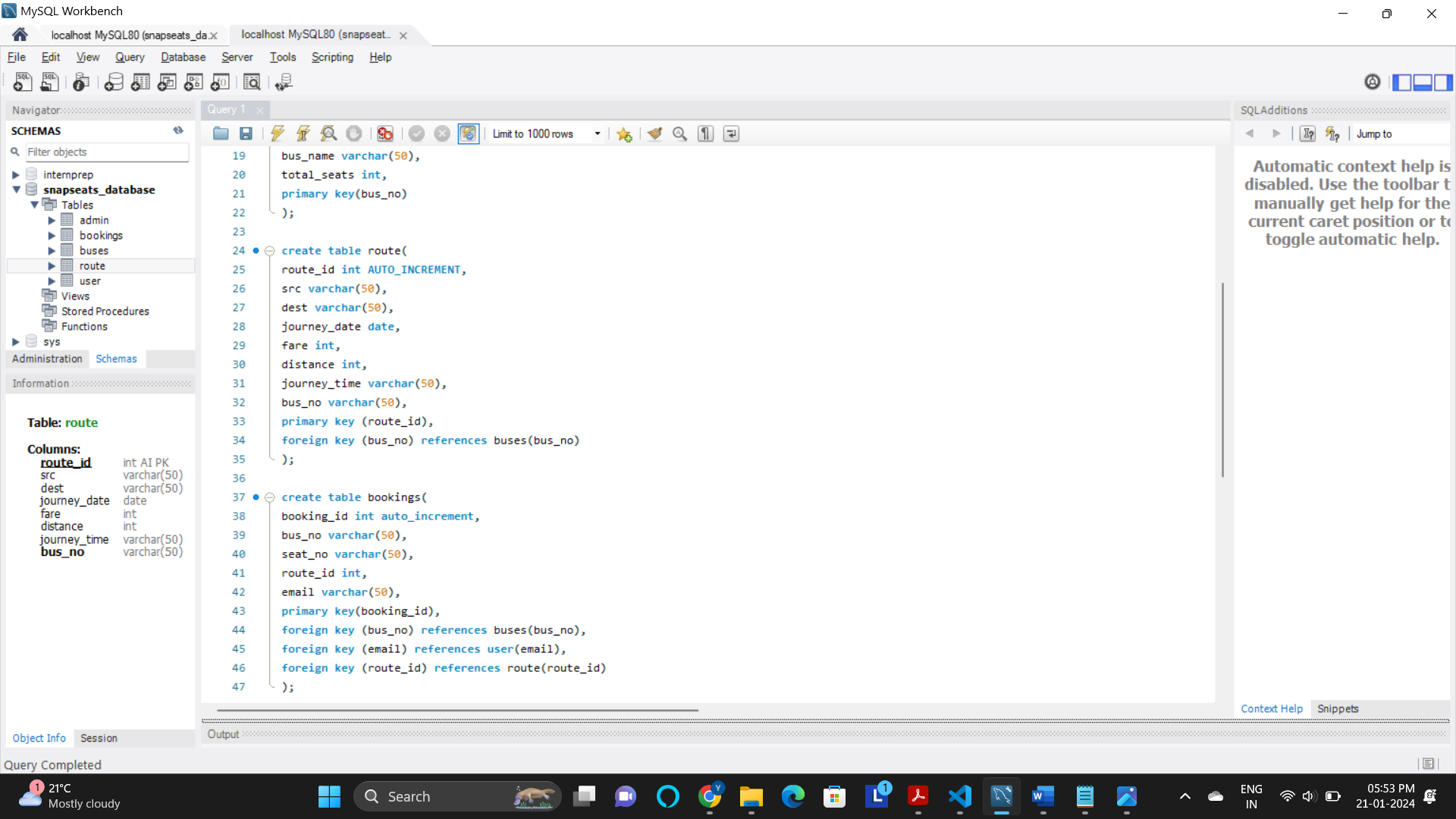
**Primary Key Constraint:** Ensures that each booking is uniquely identified by its booking\_id.

**Foreign Key Constraints:** Connects the bus\_no, route\_id, and email to their respective tables.

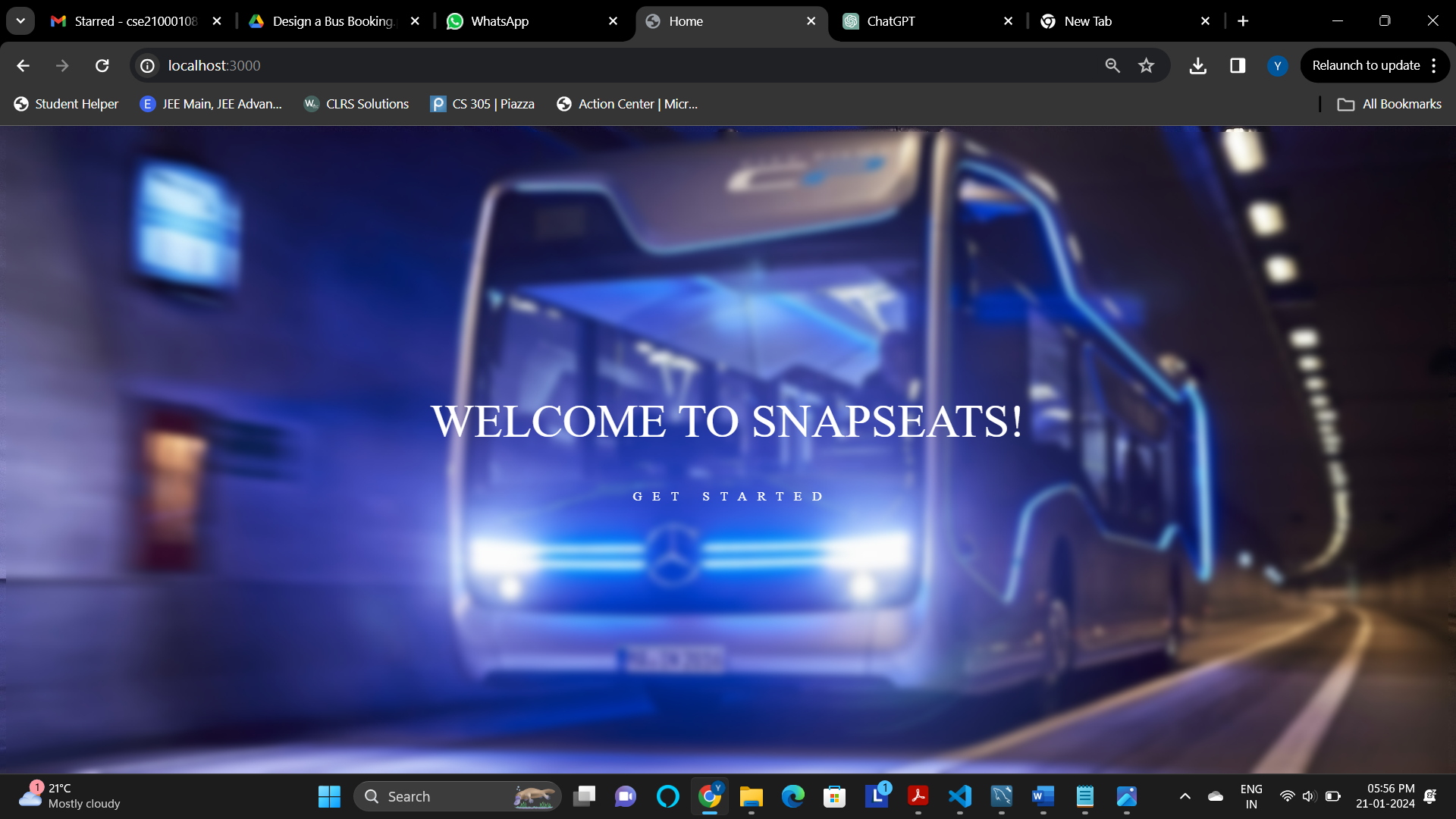
This database structure allows for the management of users, admins, buses, routes, bookings, and user booking history in an organized and relational manner.

SCREENSHOTS OF DATABASE SQL CODE:

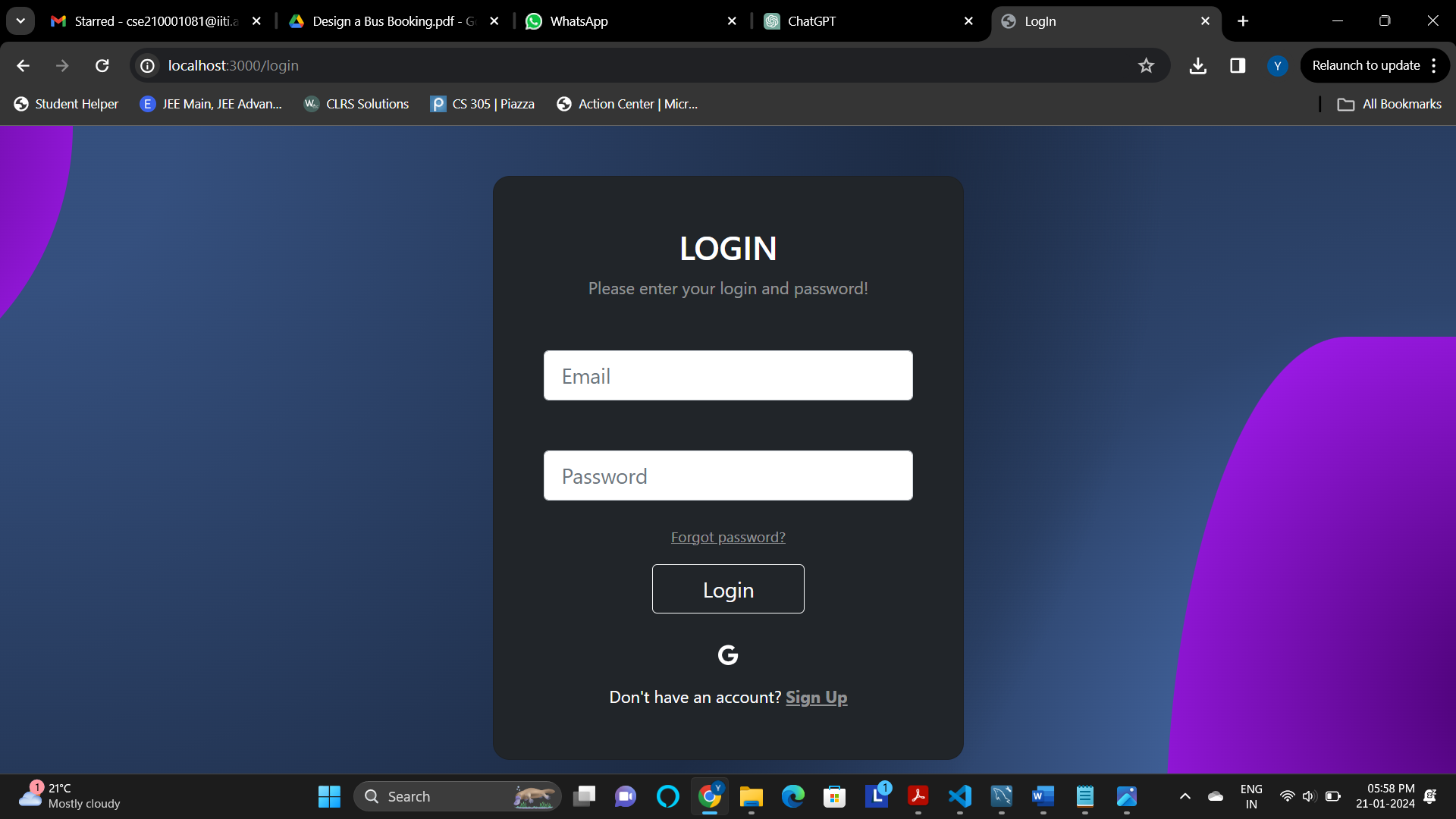




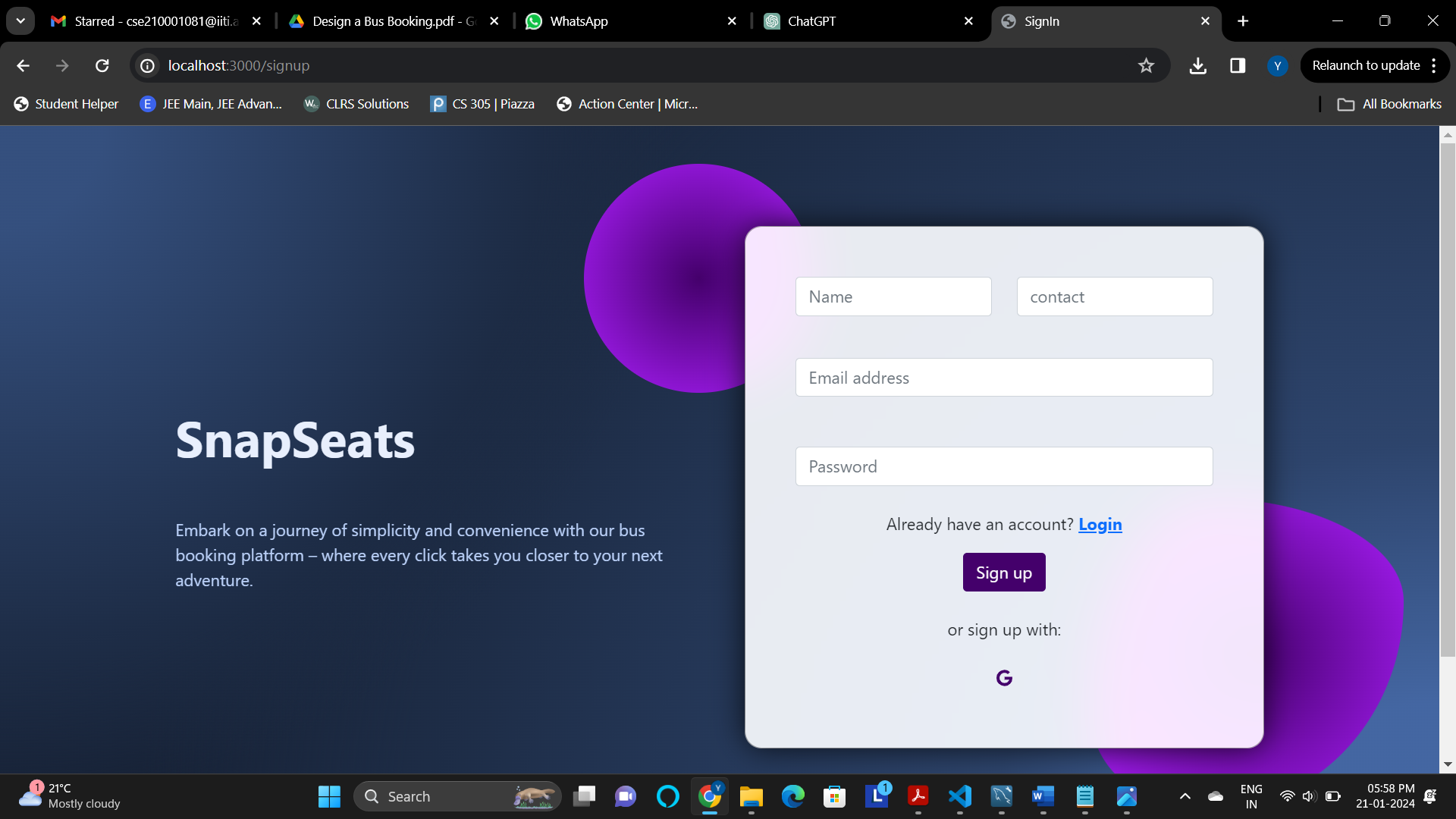
SCREESHOTS OF HOME PAGE AND AUTHENTICATION PAGES:

HOME PAGE:

Get started button redirects to the login page

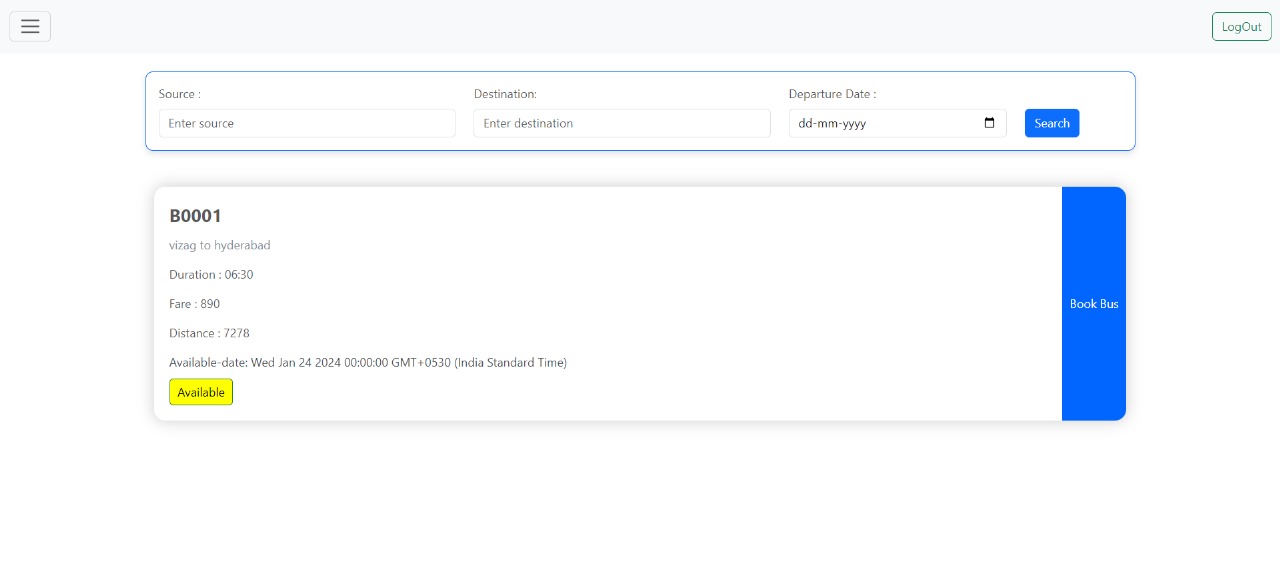
LOGIN PAGE:

Takes in the user credentials and redirects him to the dashboard based on the user type

SIGNIN PAGE:

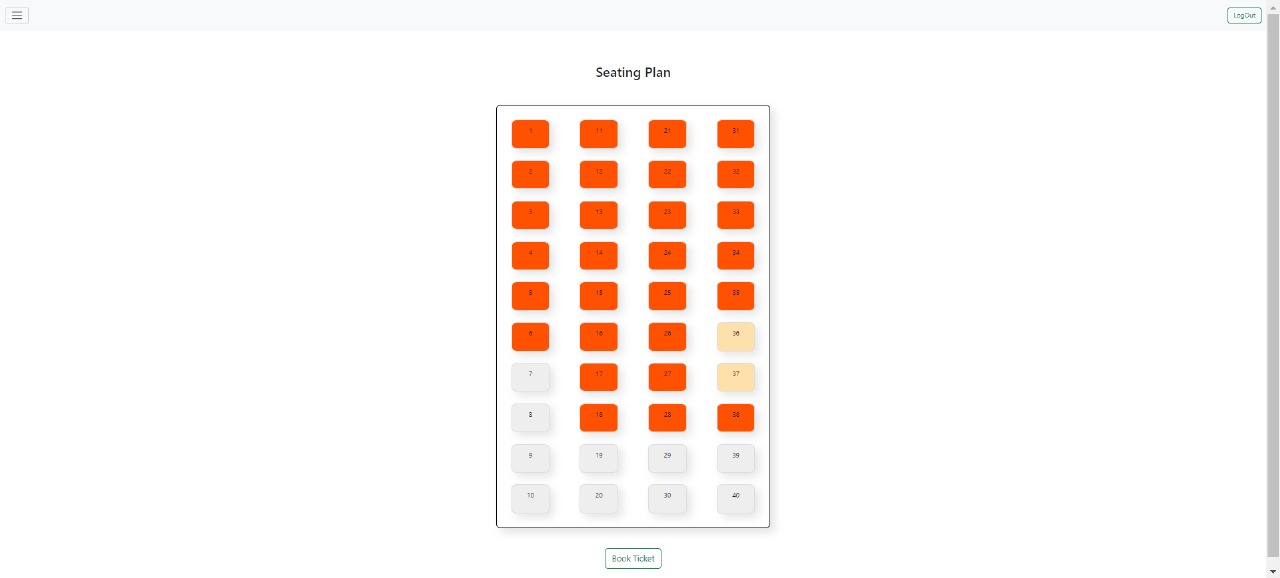
Takes in the information of the new user and saves it to the database. Then redirects to the dashboard based on the new user type

USER PAGE:

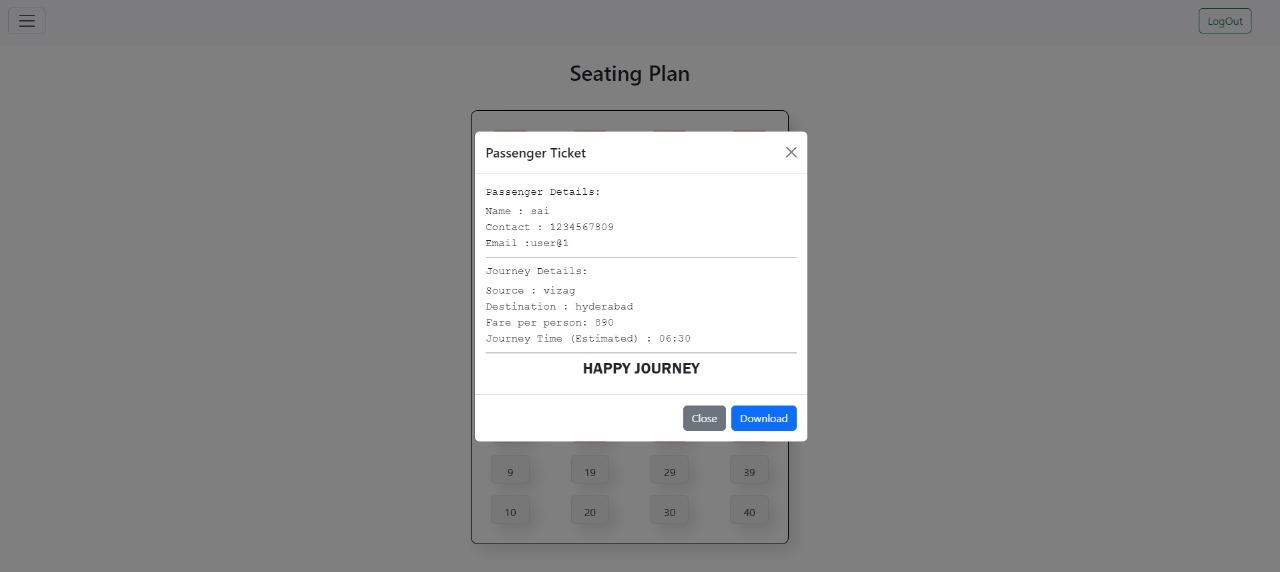
SEARCH AND BOOK TICKETS:

Once the user enters the source, destination and departure date and searches, it displays the buses available. based on the number of seats booked, it displays the **available color** button in red, yellow, green colors. Once he presses book bus button, it redirects to the seating plan where user can select the seats.

CHOOSE SEAT:

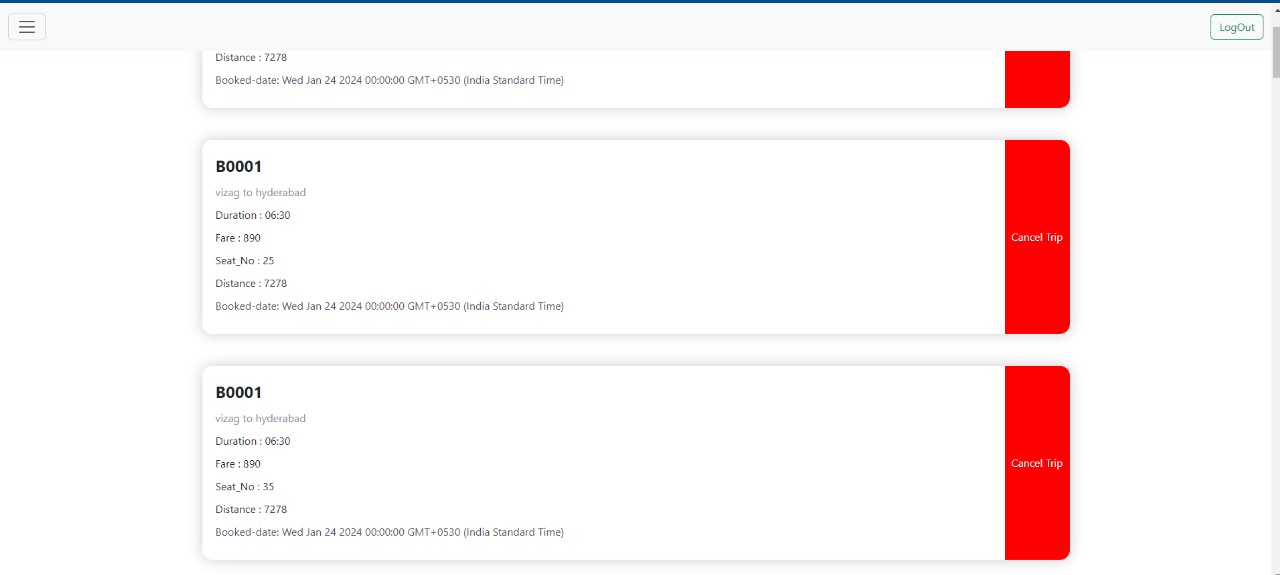


We can even select multiple seats and book them. Once we select seats and book them, it displays a pop up of the ticket.

DISPLAY AND DOWNLOAD TICKETS:

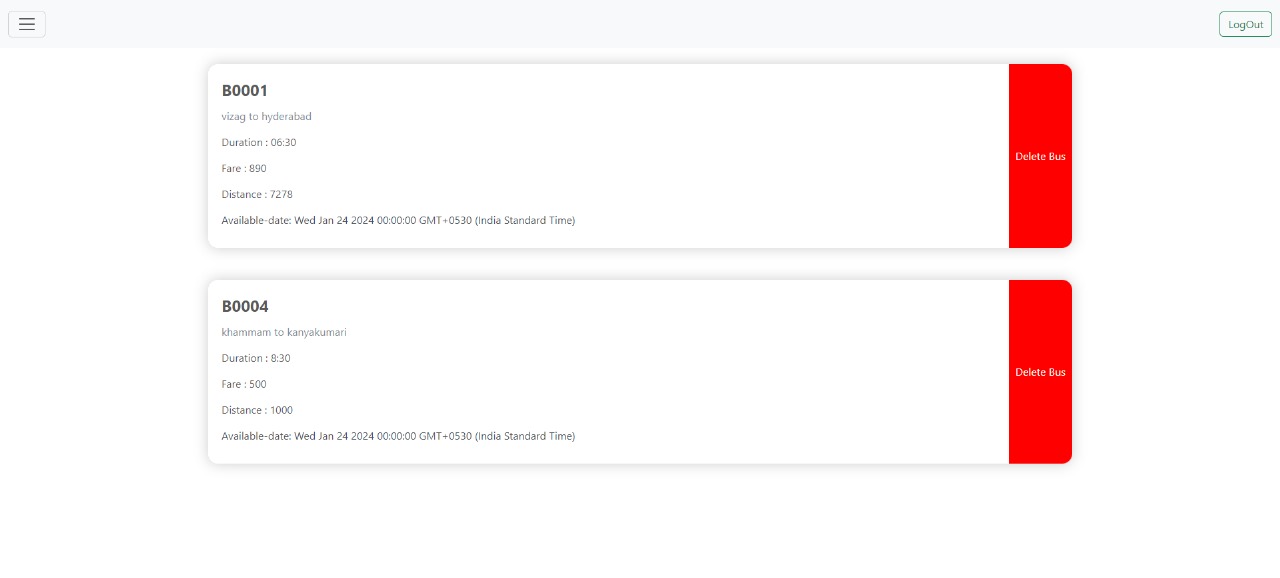
We can even download the ticket in the pdf format.

SEE BOOKING HISTORY:

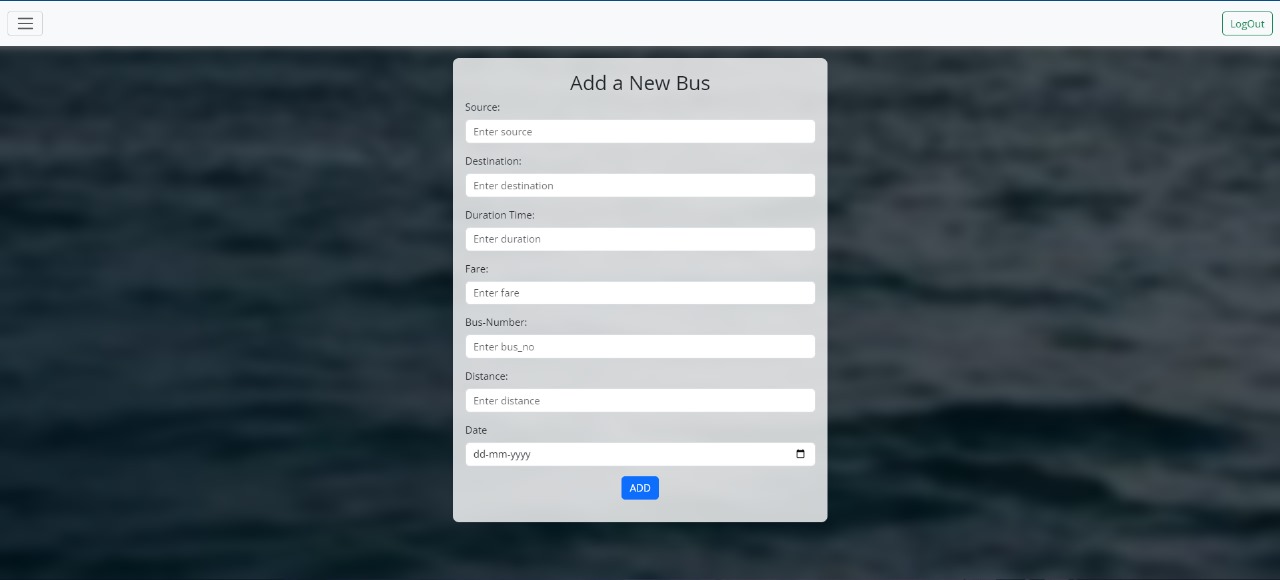


Here, the user can see all his tickets and even cancel the tickets by pressing the cancel ticket button.

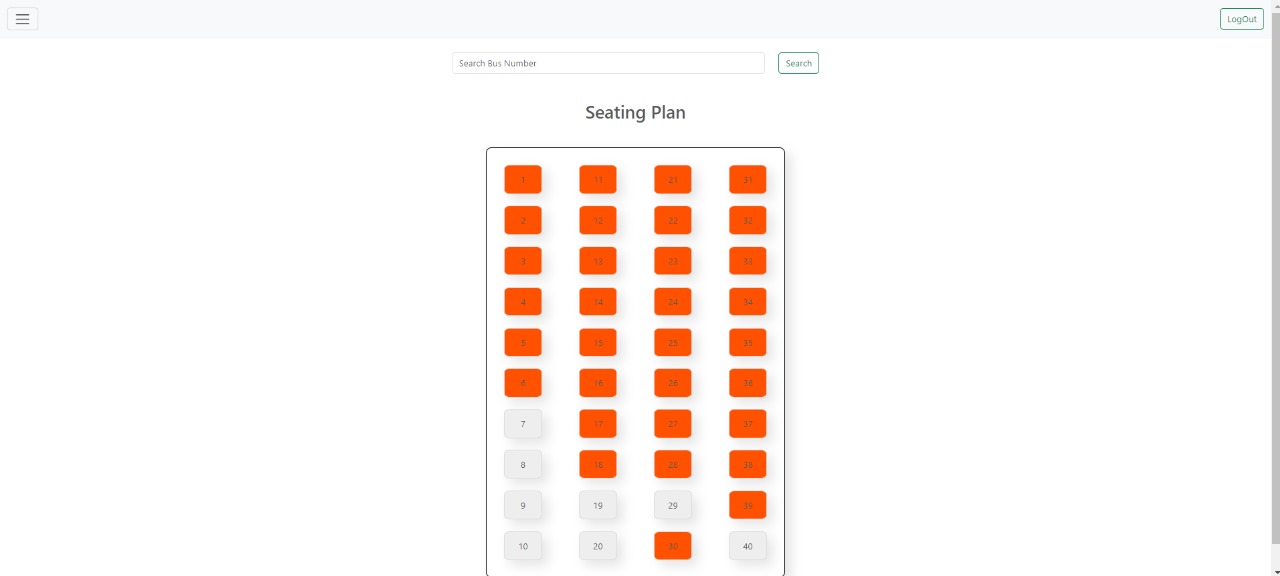
ADMIN PAGE:

VIEW CURRENT BUSES SCHEDULE AND DELETE A BUS:

This is the main page of the admin dashboard, it displays the bus details and gives the privilages to delete the bus route as well

ADD NEW BUS:

This page is to add a new bus route by the admin.

VIEW SEATING PLAN AND SEATING AVAILABILITY OF THE BUS:­­

In this page the admin can view the seating availability of the bus no. searched by the admin.

­­­­