The Bus Ticket Booking System is a web and mobile application designed to simplify the

process of booking bus tickets. It allows users to search for available tickets, view detailed information, and book specific seats. The backend is built with Node.js and Express.js, while the frontend uses Flutter for the mobile app and standard web technologies for the web interface. Key features include real-time seat availability updates, filtering options based on travel details, and an administrative panel for managing schedules. The system uses RESTful APIs for all ticket operations, ensuring a smooth and efficient booking process.

1. **Sign in page:**

A page that takes input from user and compare it to the data existing in a json file.

A screenshot of a computer

Description automatically generated

Figure 1 sign in page.

1. **Buses page**

In this page you can do the following.

* Search for specific bus with any/all the following criteria (From city, To city, Date, Time, bus class) (see figure 2)
* Add new tickets using the sliding panel. (see figure 3)
* When clicking on any of the tickets you will be able to book a specific seat on that bus. (see figures 4, 5)

A screenshot of a computer

Description automatically generated

Figure 2 Bus page

A screenshot of a computer

Description automatically generated

Figure 3 adding a bus in the buses screen.

A screenshot of a computer

Description automatically generated

A screenshot of a game

Description automatically generatedFigure 4 select the specific ticket to book.

A screenshot of a game

Description automatically generated

Figure 5 example of set of booked seats.

1. **Report page.**

In this page you find a set of helpful statistics with an interactive graph that shows how the sales are going for different buses.

A screenshot of a graph

Description automatically generated

Figure 6 report page.

1. **Profile page.**

Thing is fancy here just a typical profile with data coming from json file, and a picture of a handsome lad.

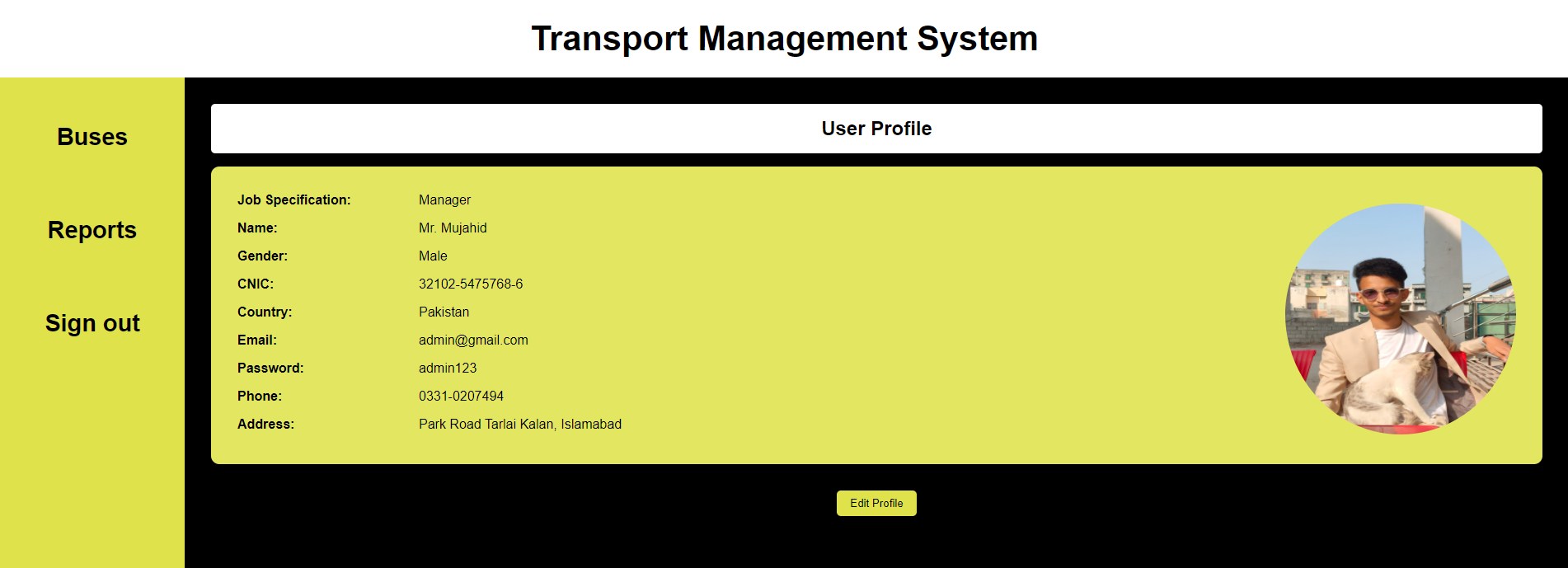


Figure 7 profile page.

1. **End points for the API.**

* Get All Tickets (GET /tickets)

Retrieve all available tickets.

Link: <https://webproject-123-41fb57c20018.herokuapp.com/tickets>

A screen shot of a computer program

Description automatically generated

* Get a Specific Ticket by ID (GET /tickets/:id)

Description: Retrieve details of a specific ticket by ID.

Link: <https://webproject-123-41fb57c20018.herokuapp.com/tickets/1>

A computer screen shot of white text

Description automatically generated

* Add a New Ticket (POST /tickets)

Description: Add a new ticket to the system.

Link: <https://webproject-123-41fb57c20018.herokuapp.com/tickets>

Message body example:

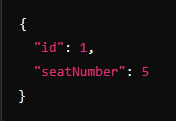
A computer screen shot of text

Description automatically generated

* Book a Specific Seat on a Ticket (POST /book-seat)

Link: <https://webproject-123-41fb57c20018.herokuapp.com/book-seat>

Message body example:



1. **Mobile app (User side)**

The supposedly user application to book a ticket.

A screenshot of a phone

Description automatically generatedA screenshot of a phone

Description automatically generatedA screenshot of a phone

Description automatically generated

Figure 8 application interface.