SCHOOL OF INFORMATION TECHNOLOGY

MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY

WEST BENGAL

Software Engineering Project

On

"Online Railway Reservation System"

(2022)

MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY, WEST BENGAL

By

Sachin Sarkar [Roll : 30084320001]

Arpan Datta [Roll : 30084320003]

Anirban Debnath [Roll : 30084320010]

Sayan Mondal [Roll : 30084320012]

From

2nd Year, 4th Semester

BSc IT (Data Science)

Abstract

The Indian Railways (IR) carries about millions of passengers in reserved accommodation every day. The Computerised Railway Reservation System facilities the booking and cancellation of tickets. These tickets can be booked or cancelled for journeys commencing in any part of India and ending in any other part.

In the given project we will be developing a website which will help users to find train details, book and cancel tickets and the exact rates of their tickets to the desired destination as well as admin can add , edit and remove train details.

With the help of online booking people can book their tickets online through internet, sitting in their home by a single click of mouse.

Table of Contents

Introduction

General Overview

Our website has various kinds of information that helps regarding booking of tickets via railways.

Fist of all someone can login to our website in two ways, one is normal user and one is admin. For new users there is a sign-up option to create a new account.

A normal user will be able to search trains, see the train fare, the arrival and departure time of the train, they can book the tickets and also check the ticket status, after booking user can easily cancel the ticket (if possible). User can edit their personal information like name, age, address, user id etc and they can also change their account password.

Now let's talk about Admin, an admin will be able to add new trains in train list and they can also able to remove trains from the train list.

Objectives

The objective of the railway reservation system Project is to design and develop a software product for booking railway reservation tickets and also cancel that easily with one click.

That is –

1. To search the trains.
2. To check the train's arrival, departure time.
3. To check the train's source and destination.
4. To check the availability of the ticket.
5. To calculate fare.
6. To book the ticket.
7. To check the booked tickets.
8. To cancel the ticket if necessary.

Feasibility Study

Here we check the feasibility study in two ways such as technical feasibility and economical feasibility.

Technical Feasibility

We can strongly say that it is technically feasible, science there will not be much difficulty getting required resources for the development and maintaining the system as well. All the resources needed for the development of the software as well as the maintenance of the same is easily available in online(free of cost), here we are utilizing the resources which are available in our device already.

Economical Feasibility

Development of this application is highly economically feasible. We not needed spend much more for the development of the system already available. The only thing is to be done is making an environment for the development with an effective supervision. If we are doing so, we can attain the maximum usability of the corresponding resources. Even after the development, we will not be in a condition to invest more in this. Therefore, the system is economically feasible.

Software Requirements Specification

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS purpose ,scope, definitions, acronyms, abbreviations, references and overview of SRS.A **Software Requirements Specification** (**SRS**) - a requirements specification for a software system - is a complete description of the behaviour of a system to be developed. It includes a set of use cases that describe all the interactions the users will have with the software. Use cases are also known as functional requirements . In addition to use cases, the SRS also contains non-functional (or supplementary) requirements. Non-functional requirements are requirements which impose constraints on the design or implementation (such as performance engineering requirements, quality standards, or design constraints). The aim of this document is to gather and analyse and give an in-depth insight of the complete Marvel Electronics and Home Entertainment software system by defining the problem statement in detail. This is a documentation of the project Railways Reservation System done sincerely and satisfactorily by my group members. A Software has to be developed for automating the manual **Railway Reservation System**.

Purpose

User:

The user purpose of used the railway reservation system is to create search train ,book ticket, cancel ticket ,booking history, edit your details, reset password and logout.

Admin:

An administrator of a railway reservation system is a person who can run everything through system edits, some of which are show trains, show users , Add train ,Edit train , Remove train, Remove users and log out, etc.

Scope

Reservation Clerk is a person to create and Cancel Reservation by entering Login Password . Admin is a person who updates the Train Information by entering his own Password . The system can handle only reservation and train details efficiently and it doesn't contain other details about the railway administration . The main purpose of this system is

- search train
- book ticket
- cancel ticket
- booking history
- edit your details
- reset password
- logout

The Seats of book tickets cannot be more than the seats of Train at that date . This is a constraint that has to be followed the Clerk when he creates the Reservation . For that purpose he wants to check the seats remaining present in the Train .

The scope of this system in creating Reservation is that , from any Railway Station we can Create Reservation , which is updated automatically in all the stations . Hence , there is no confusion to the Reservation Clerk in all the stations to create the Reservation . This can be possible by maintaining Global Database . Clerks present at different can access the global database and the clerks can easily understand the remaining reservation seats . It provides the ability to create reservation from different places for a train .

Product & its functions

The Product of this project is Railway Reservation System , which is to Search Trains, Book Tickets, Cancel Tickets, Booking History, Edit Your Details, Reset Password , Logout.

Benefits and Goals

The Benefit of this project is to reduce the work of Reservation Clerk and it is easy to check the remaining seats present in the Train and easily view the Reservation Status of the Train and tells to the passenger . The Goals of this project is that , from any Railway Station we can Create Reservation , which is updated automatically in all the stations Hence , there is no confusion to the Reservation Clerk in all the stations to create the Reservation.

Glossary

This should define all technical terms and abbreviations used in the document

- DFD :- Data Flow Diagram
- ERD :- Entity Relationship Diagram
- SRS :- Software Requirements Specification

Product Description

Reservation Clerk receives details from the passenger and he checks to see if there are any reserved seats left. When seats are available, passengers are allotted seats with reservations. Then slip to the passenger and he will update the details of the reservation. If seats are not available, place the reservation on the 'Waiting List'. Then if the reservation clerk gets the details from the passenger to cancel the reservation and updates the passenger. After the reservation was cancelled, the reservation was made with the reservation details and the passengers were placed on the waiting list.

The reservation clerk tell the reservation status to passengers who are in waiting list . The manager updates the train information and he generates the report of the train , reservation details .

Product Functioning

- The reservation clerk takes the detail from the passenger .
- The reservation clerk checks whether the seats are reaming or not .
  If seats are available , reservation clerk create the reservation and updates the reservation details .
- If seats are not available . he place the reservation in waiting list .
  Reservation clerk cancels the reservation and update the reservation details .
- The manager updates the train information and generates the report.

Functions of Project

There are seven functionalities by the Railway Reservation system:

- **search train:** Passengers search the train according to their own time to reach their destination.
  1. put the train source and destination the search box ,
  2. click on find

- **book ticket:** Reservation form has to be filled by passenger. If seats are available entries like train name, number, destination are made.

- **cancel ticket:** If the user has made a ticket reservation on a date, and the user later feels that he will not be able to go that day, then the user can cancel his ticket

- **booking history:** If a user has booked a ticket, the history of that reservation will be automatically viewed directly in the user's booking history on that website.

- **edit your details :** If the user wants to save a new seat and if he thinks he needs to edit the profile details of his profile, he can go to the Edit option.

- **reset password :** If an old user or a new user goes to the website and forgets or does not remember his password while logging in, he can reset the password and change his old password to give a new password.

- **Logout:** Users can log out by going to the logout option if they do not get any benefit from this website or if they have any problem and users can log out if they have a lot of shots, that is, 3 or more email accounts, and if they think they can keep one email account and delete the rest.

Function Requirements

performance requirements :

- **User Satisfaction:** - The system is such that it stands up to the user expectations.
- **Response Time: -**The response of all the operation is good. This has been made possible by careful programming.
- **Error Handling:** - Response to user errors and undesired situations has been taken care of to ensure that the system operates without halting.
- **Safety and Robustness:** - The system is able to avoid or tackle disastrous action. In other words, it should be foul proof. The system safeguards against undesired events, without human intervention.
- **Portable:** - The software should not be architecture specific. It should be easily transferable to other platforms if needed.
- **User friendliness:** - The system is easy to learn and understand. A native user can also use the system effectively, without any difficulties.

Hardware requirements :

For the hardware requirements the SRS specifies the logical characteristics of each interface and the software product and the hardware components. It specifies the hardware requirements like memory restrictions, cache size, the processor, RAM size etc... those are required for the software to run.

Minimum Hardware Requirements

Processor: (Intel core i3 11generation / AMD Ryzen 3 and above)

Hard disk drive 40 GB

RAM 4 MB

Cache 212 kb

Printer

Wifi connection

Preferred Hardware Requirements

Processor: (Intel core i5 11generation/ AMD Ryzen 5 and above)

Hard disk drive 256 GB

RAM 8 GB

Cache 512 kb

Software requirements :

The software requirement of this are as follows:

- Windows 10 , 11
- SQL
- **MYSQL (Server, Workbench,** CLI **)**
- Python (libraries ; Pandas, Numpy, Streamlit, Backend : MySQL Connector)
- **VISUAL STUDIO CODE**

Logical Database Requirements :

The following information is to be stored in the database:

- Login Table(Passenger ID, User-Name, Password)
- Passenger Table -> (Passenger IDs , Names , Age , Gender, Address, Contacts)
- Train Table-> (Train Numbers , Name, Source, Destination, Source-Time, Destination-Time , Total-Seats, Available-Seats, Fare)
- Tickets Table (Tickets-Number, Passenger-Id, Train-Number, Date, Number-of- Passenger, Seats-Number, Total Price).

Non-Function Requirements

Security:

The system use SSL (secured socket layer) in all transactions that include any confidential customer information. The system must automatically log out all customers after a period of inactivity. The system should not leave any cookies on the customer's computer containing the user's password. The system's back-end servers shall only be accessible to authenticated management.

Reliability:

The reliability of the overall project depends on the reliability of the separate components. The main pillar of reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes.

Availability:

The system should be available at all times, meaning the user can access it using a web browser, only restricted by the down time of the server on which the system runs. A customer friendly system which is in access of people around the world should work 24 hours. In case of a of a hardware failure or database corruption, a replacement page will be shown. Also in case of a hardware failure or database corruption, backups of the database should be retrieved from the server and saved by the Organizer.

Maintainability:

A commercial database is used for maintaining the database and the application server takes care of the site. In case of a failure, a re-initialization of the project will be done. Also the software design is being done with modularity in mind so that maintainability can be done efficiently.

Supportability:

The code and supporting modules of the system will be well documented and easy to understand. Online User Documentation and Help System Requirements.

Project Scheduling

Project-task scheduling is a significant project planning activity. It comprises deciding which functions would be taken up when.

With our Analysis on Project tasks based on the Objective and SRS, we finalize our Project Task Scheduling as given below :

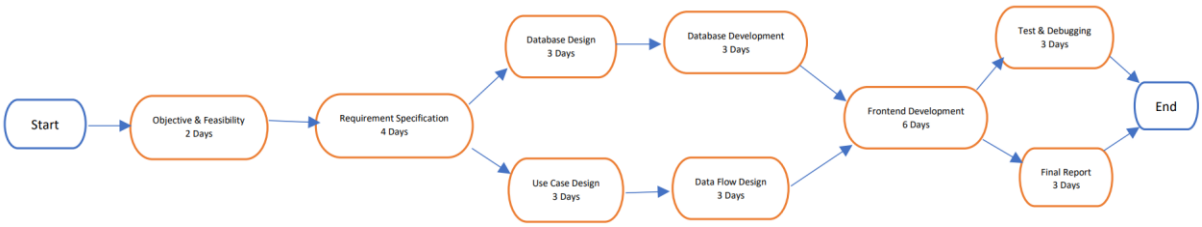
1. Start – 15th May, 2022
2. Objective and Feasibility Analysis – (15th May – 17th May)
3. Requirement Analysis and Specification – (17th May – 21st May)
4. Database Design – (21st May – 24th May)
5. Use Case Design – (21st May – 24th May)
6. Data Flow Design – (24th May – 27th May)
7. Database Development – (24th May – 27th May)
8. Frontend Development – (27th May – 2nd June)
9. Test and Debugging – (2nd June – 5th June)
10. Final Report – (2nd June – 5th June)
11. End – 5th June, 2022

The Pert Chart and Gantt Chart is given below –

Pert Chart

A PERT chart, also known as a PERT diagram, is a tool used to schedule, organize, and map out tasks within a project. PERT stands for program evaluation and review technique. It provides a visual representation of a project's timeline and breaks down individual tasks and shows the dependency among this individual tasks.

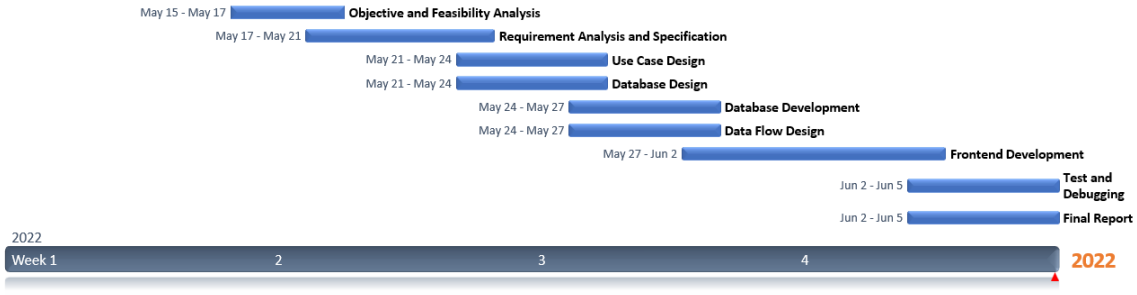
Here is the Pert Chart as per our Planned Project Schedule :



Gantt Chart

A Gantt chart is a horizontal bar chart developed as a production control tool. Frequently used in project management, a Gantt chart provides a graphical illustration of a schedule that can be used to plan, coordinate and track tasks in a project.

Here is the Gantt Chart as per our Planned Project Schedule :

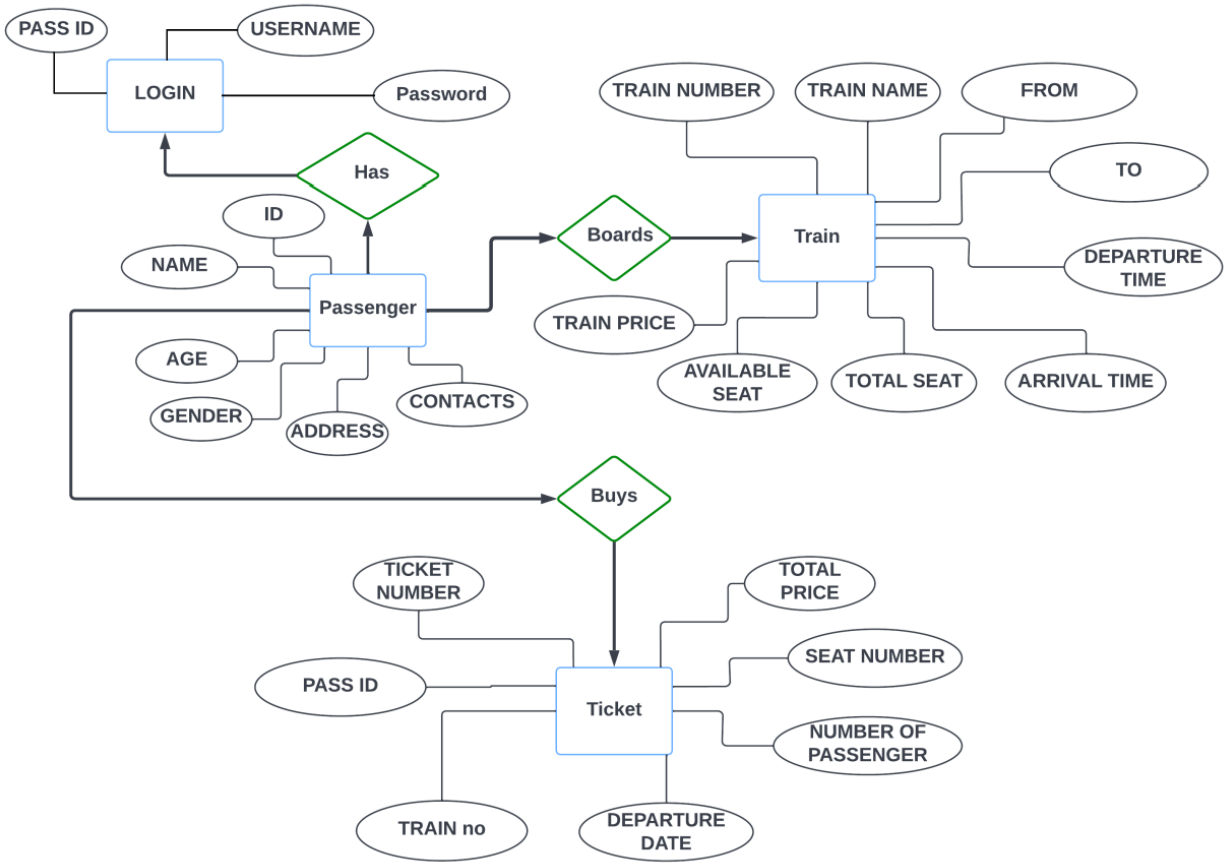


Design

Entity Relationship Design

An entity relationship diagram (ERD), also known as an Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that implemented as a database.

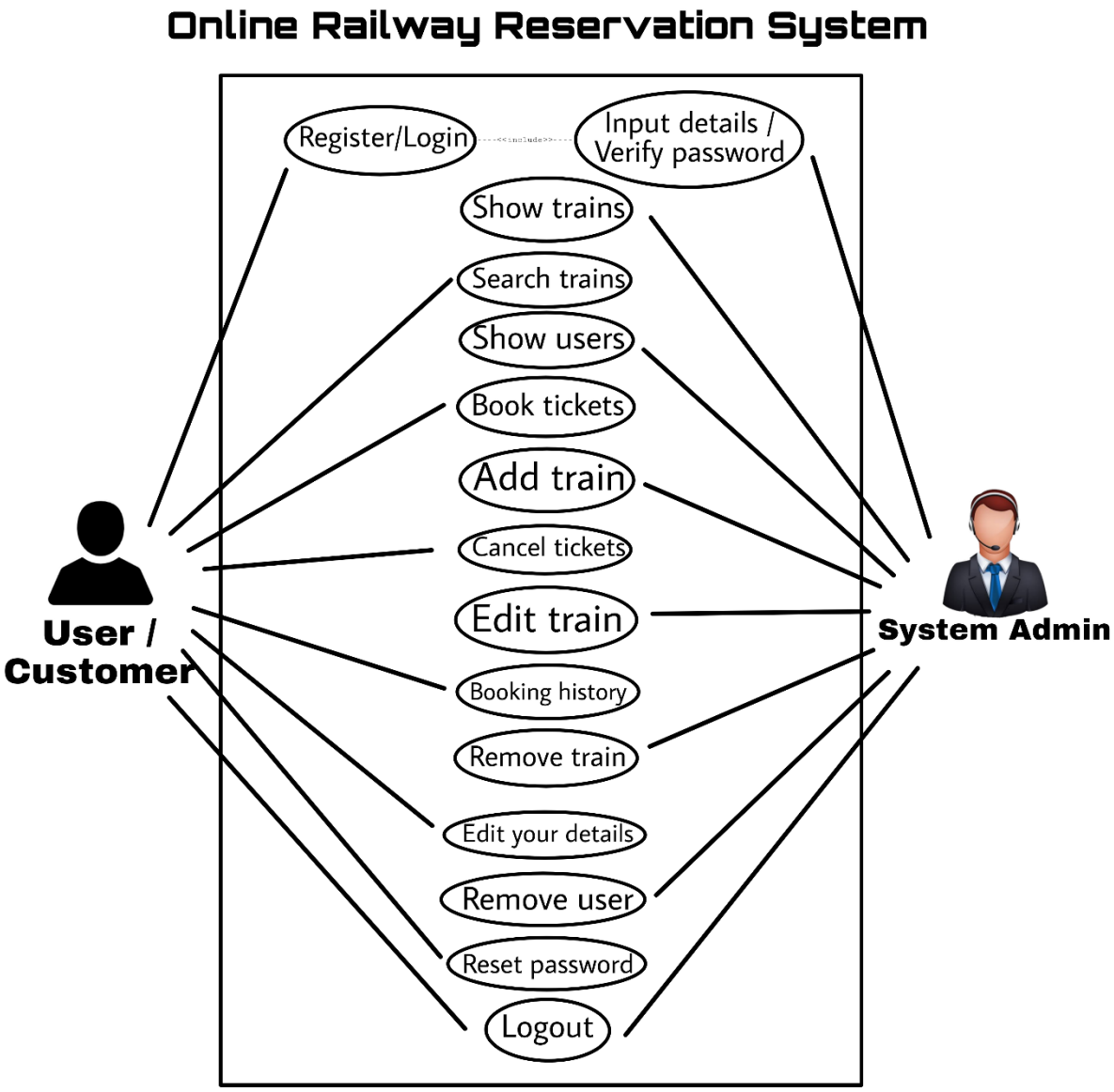
Here is the ER Diagram that designed to implement our Database in MySQL Server for our "Railway Reservation System".



Use Case Design

A **use case diagram** is **used to represent the dynamic behavior of a system**. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application.

Here is the Use Case Diagram that designed to implement our website for our “Railway Reservation System”.



Data Flow Diagram

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both. It shows how data enters and leaves the system, what changes the information, and where data is stored.

A DFD usually divided into 3 levels like level 0, level 1, level 2.

DFD – Level 0 :

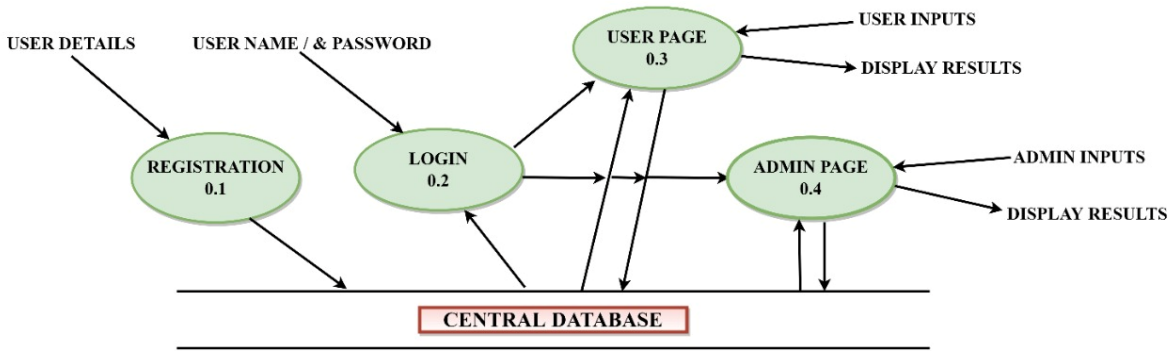
It is also known as a context diagram. It's designed to be an abstraction view, showing the system as a single process with its relationship to external entities. It represents the entire system as a single bubble with input and output data indicated by incoming/outgoing arrows.

Here is the DFD of 0 level for our system :

DFD – Level 1 :

In 1-level DFD, the context diagram is decomposed into multiple bubbles/processes. In this level, we highlight the main functions of the system and breakdown the high-level process of 0-level DFD into subprocesses.

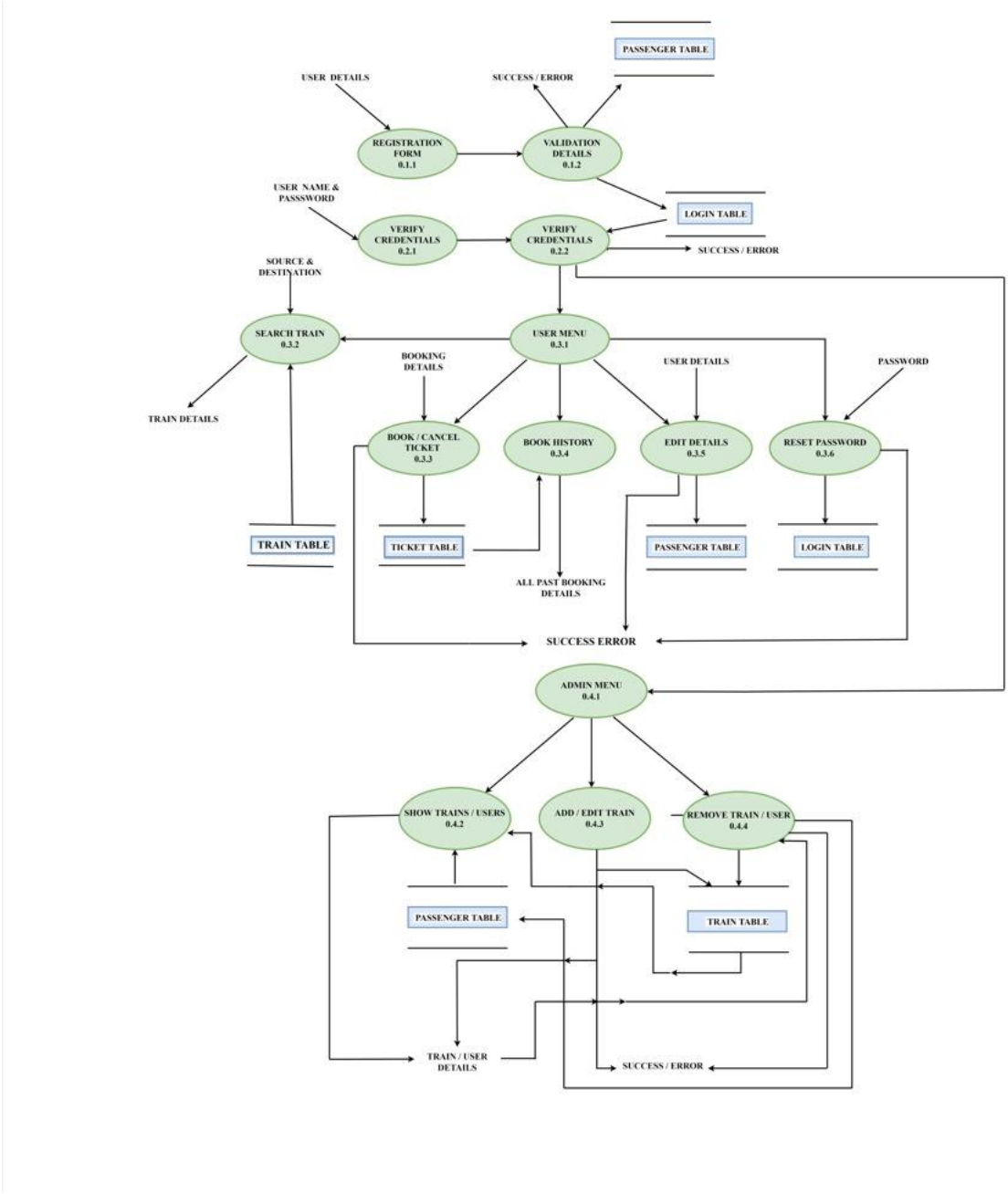
Here is the DFD of 1 level for our system :



DFD – Level 2 :

2-level DFD goes one step deeper into parts of 1-level DFD. It can be used to plan or record the specific/necessary detail about the system's functioning.

Here is the DFD of 2 level for our system :

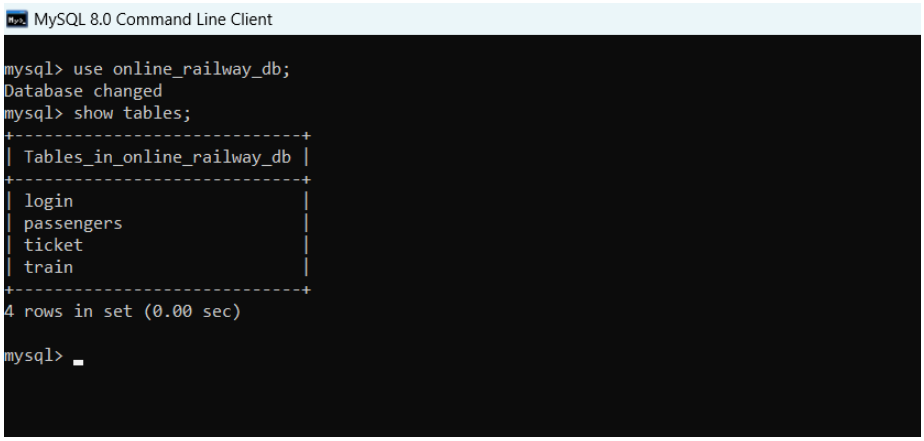


Coding and Implementation

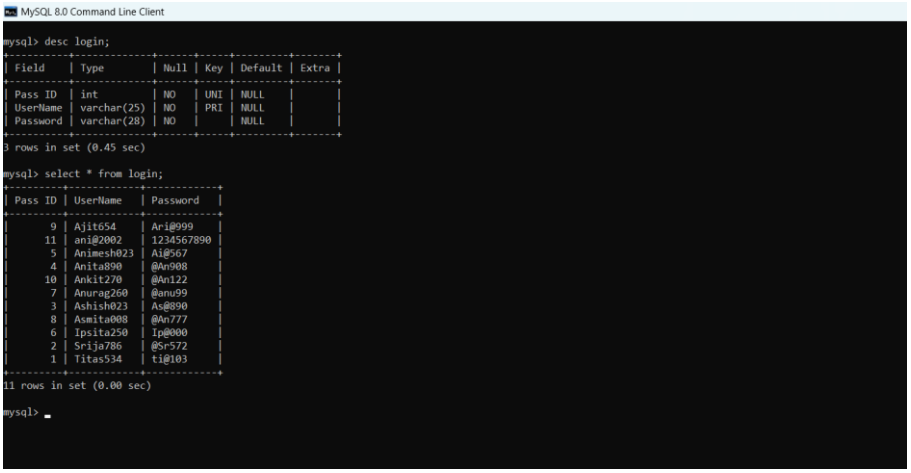
As per our Objective, the Software Requirement Specification and Design, we implemented the entire Railway Reservation System where we developed a website integrated with backend database as per we designed through Python and MySQL.

Here are the screenshot for both Backend database and Frontend Website given below :

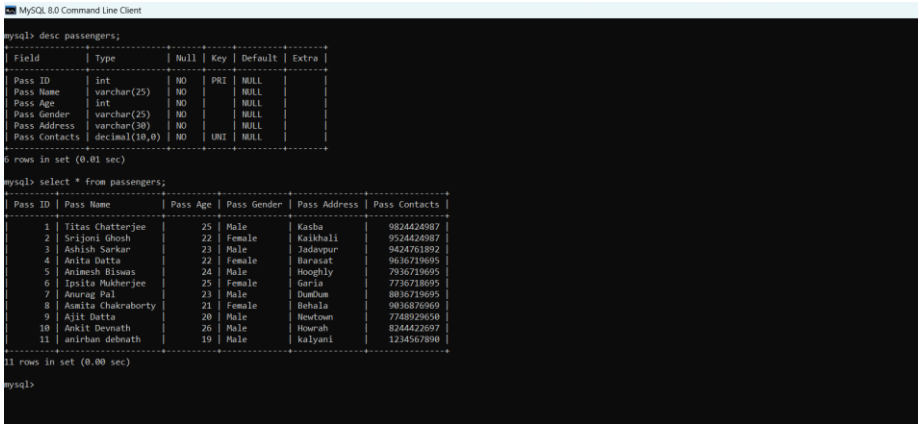
Backend or Database Images



List of all tables that our system uses

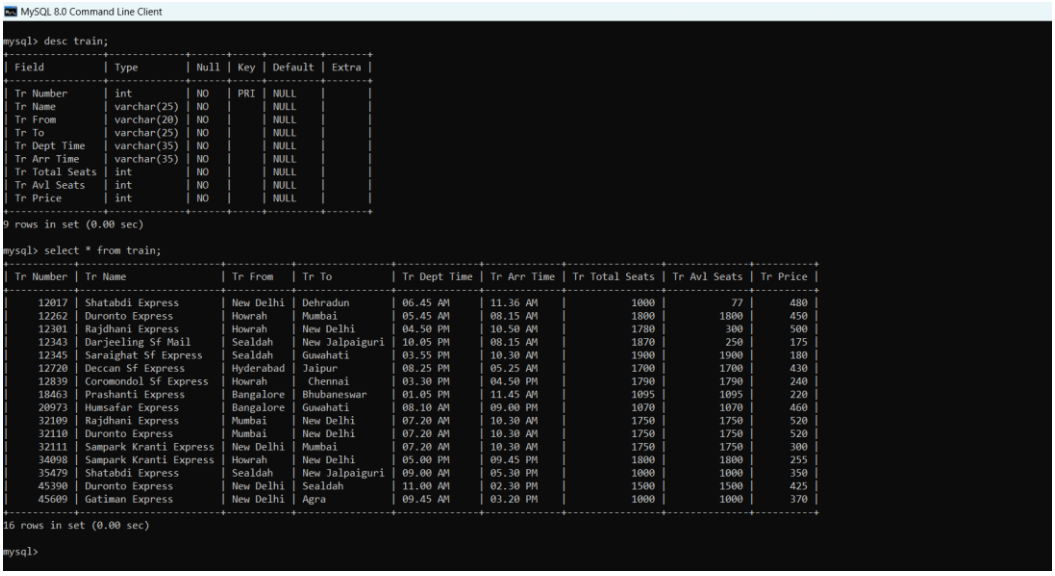


Login table with schema and data



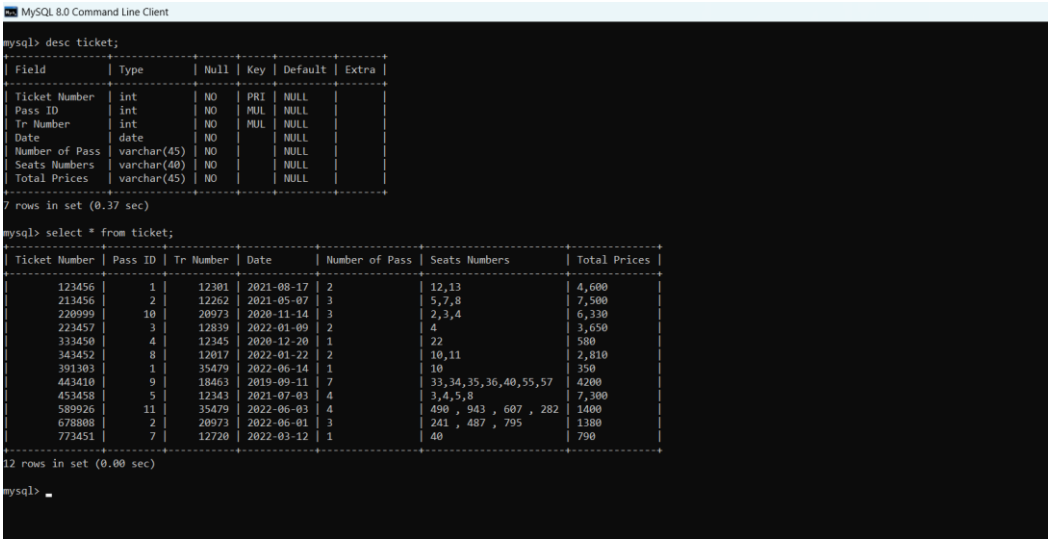
```
MySQL 8.0 Command Line Client

mysql> desc passengers;
+---------------+--------------+------+-----+---------+-------+
| Field         | Type         | Null | Key | Default | Extra |
+---------------+--------------+------+-----+---------+-------+
| Pass ID       | int          | NO   | PRI | NULL    |       |
| Pass Name     | varchar(25)  | NO   |     | NULL    |       |
| Pass Age      | int          | NO   |     | NULL    |       |
| Pass Gender   | varchar(25)  | NO   |     | NULL    |       |
| Pass Address  | varchar(30)  | NO   |     | NULL    |       |
| Pass Contacts | decimal(10,0) | NO   | UNI | NULL    |       |
+---------------+--------------+------+-----+---------+-------+
6 rows in set (0.01 sec)

mysql> select * from passengers;
+---------+-----------------+----------+-------------+--------------+--------------+
| Pass ID | Pass Name       | Pass Age | Pass Gender | Pass Address | Pass Contacts |
+---------+-----------------+----------+-------------+--------------+--------------+
|       1 | Titas Chatterjee |       25 | Male        | Kasba        |   9824424987 |
|       2 | Srijoni Ghosh   |       22 | Female      | Kaikhali     |   9524424987 |
|       3 | Ashish Sarkar   |       23 | Male        | Jadavpur     |   9424761892 |
|       4 | Anita Datta     |       22 | Female      | Barasat      |   9636719695 |
|       5 | Animesh Biswas  |       24 | Male        | Hooghly      |   7936719695 |
|       6 | Ipsita Mukherjee |       25 | Female      | Garia        |   7736718695 |
|       7 | Anurag Pal      |       23 | Male        | DumDum       |   8036719695 |
|       8 | Asmita Chakraborty |       21 | Female      | Behala       |   9036876969 |
|       9 | Ajit Datta      |       20 | Male        | Newtown      |   7748929650 |
|      10 | Ankit Devnath   |       26 | Male        | Howrah       |   8244422697 |
|      11 | anirban debnath |       19 | Male        | kalyani      |   1234567890 |
+---------+-----------------+----------+-------------+--------------+--------------+
11 rows in set (0.00 sec)

mysql>
```

Passengers table with schema and data



```
MySQL 8.0 Command Line Client

mysql> desc train;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| Tr Number     | int         | NO   | PRI | NULL    |       |
| Tr Name       | varchar(25) | NO   |     | NULL    |       |
| Tr From       | varchar(20) | NO   |     | NULL    |       |
| Tr To         | varchar(25) | NO   |     | NULL    |       |
| Tr Dept Time  | varchar(35) | NO   |     | NULL    |       |
| Tr Arr Time   | varchar(35) | NO   |     | NULL    |       |
| Tr Total Seats | int         | NO   |     | NULL    |       |
| Tr Avl Seats  | int         | NO   |     | NULL    |       |
| Tr Price      | int         | NO   |     | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
9 rows in set (0.00 sec)

mysql> select * from train;
+-----------+---------------------+-----------+-----------+---------------+---------------+----------------+--------------+----------+
| Tr Number | Tr Name             | Tr From   | Tr To     | Tr Dept Time  | Tr Arr Time   | Tr Total Seats | Tr Avl Seats | Tr Price |
+-----------+---------------------+-----------+-----------+---------------+---------------+----------------+--------------+----------+
|     12017 | Shatabdi Express    | New Delhi | Dehradun  | 06.45 AM      | 11.36 AM      |           1000 |           77 |      480 |
|     12262 | Duronto Express     | Howrah    | Mumbai    | 05.45 AM      | 08.15 PM      |           1800 |         1800 |      450 |
|     12301 | Rajdhani Express    | Howrah    | New Delhi | 04.50 PM      | 10.50 AM      |           1780 |          300 |      500 |
|     12343 | Darjeeling Sf Mail  | Sealdah   | New Jalpaiguri | 10.05 PM | 08.15 AM      |           1870 |          250 |      175 |
|     12345 | Saraighat Sf Express | Sealdah   | Guwahati  | 03.55 PM      | 10.30 AM      |           1900 |         1900 |      180 |
|     12720 | Deccan Sf Express   | Hyderabad | Jaipur    | 08.25 PM      | 05.25 AM      |           1700 |         1700 |      430 |
|     12839 | Coromondol Sf Express | Howrah    | Chennai   | 03.30 PM      | 04.50 PM      |           1790 |         1790 |      240 |
|     18463 | Prashanti Express   | Bangalore | Bhubaneswar | 01.05 PM    | 11.45 AM      |           1095 |         1095 |      220 |
|     20973 | Humsafar Express    | Bangalore | Guwahati  | 08.10 AM      | 09.00 PM      |           1070 |         1070 |      460 |
|     32109 | Rajdhani Express    | Mumbai    | New Delhi | 07.20 AM      | 10.30 AM      |           1750 |         1750 |      520 |
|     32110 | Duronto Express     | Mumbai    | New Delhi | 07.20 AM      | 10.30 AM      |           1750 |         1750 |      520 |
|     32111 | Sampark Kranti Express | New Delhi | Mumbai    | 07.20 AM      | 10.30 AM      |           1750 |         1750 |      300 |
|     34098 | Sampark Kranti Express | Howrah    | New Delhi | 05.00 PM      | 09.45 PM      |           1800 |         1800 |      255 |
|     35479 | Shatabdi Express    | Sealdah   | New Jalpaiguri | 09.00 AM | 05.30 PM      |           1000 |         1000 |      350 |
|     45390 | Duronto Express     | New Delhi | Sealdah   | 11.00 AM      | 02.30 PM      |           1500 |         1500 |      425 |
|     45609 | Gatiman Express     | New Delhi | Agra      | 09.45 AM      | 03.20 PM      |           1000 |         1000 |      370 |
+-----------+---------------------+-----------+-----------+---------------+---------------+----------------+--------------+----------+
16 rows in set (0.00 sec)

mysql>
```

Train table with schema and data

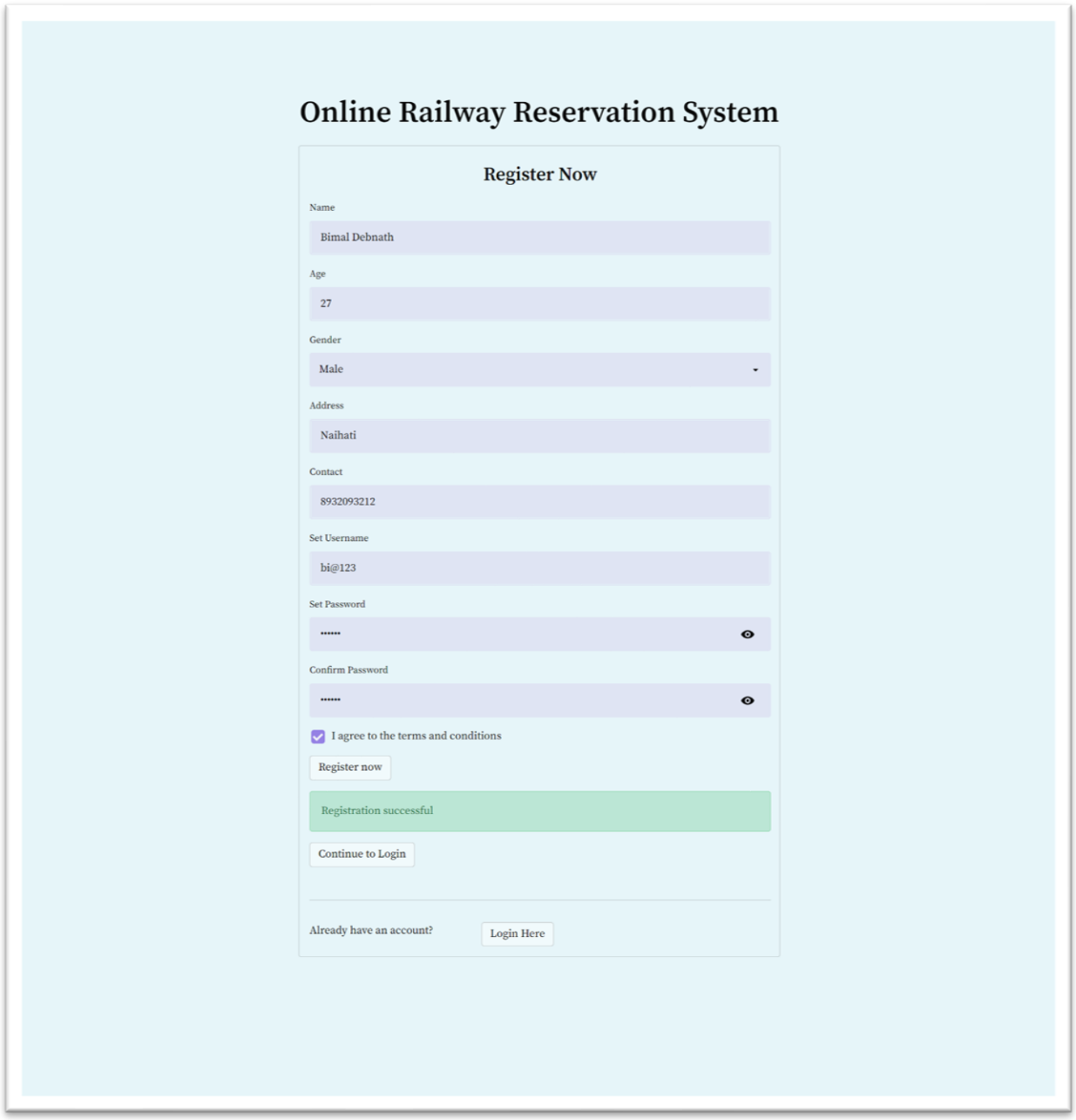


MySQL 8.0 Command Line Client

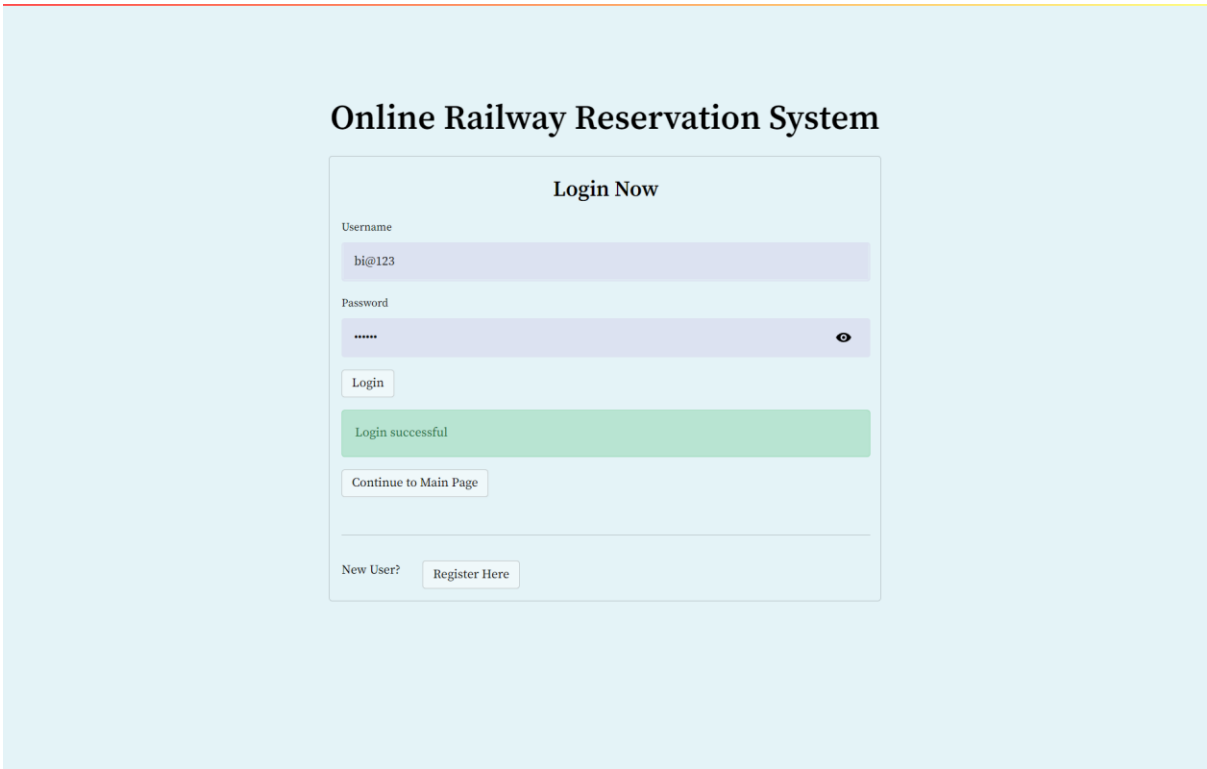
```
mysql> desc ticket;
+----------------+-------------+------+-----+---------+-------+
| Field          | Type        | Null | Key | Default | Extra |
+----------------+-------------+------+-----+---------+-------+
| Ticket Number  | int         | NO   | PRI | NULL    |       |
| Pass ID        | int         | NO   | MUL | NULL    |       |
| Tr Number      | int         | NO   | MUL | NULL    |       |
| Date           | date        | NO   |     | NULL    |       |
| Number of Pass | varchar(45) | NO   |     | NULL    |       |
| Seats Numbers  | varchar(40) | NO   |     | NULL    |       |
| Total Prices   | varchar(45) | NO   |     | NULL    |       |
+----------------+-------------+------+-----+---------+-------+
7 rows in set (0.37 sec)

mysql> select * from ticket;
+--------------+---------+-----------+------------+----------------+-------------------+--------------+
| Ticket Number | Pass ID | Tr Number | Date       | Number of Pass | Seats Numbers     | Total Prices |
+--------------+---------+-----------+------------+----------------+-------------------+--------------+
|       123456 |       1 |     12301 | 2021-08-17 | 2              | 12,13             | 4,600        |
|       213456 |       2 |     12262 | 2021-05-07 | 3              | 5,7,8             | 7,500        |
|       220999 |      10 |     20973 | 2020-11-14 | 3              | 2,3,4             | 6,330        |
|       223457 |       3 |     12839 | 2022-01-09 | 2              | 4                 | 3,650        |
|       333450 |       4 |     12345 | 2020-12-20 | 1              | 22                | 580          |
|       343452 |       8 |     12017 | 2022-01-22 | 2              | 10,11             | 2,810        |
|       391303 |       1 |     35479 | 2022-06-14 | 1              | 10                | 350          |
|       443410 |       9 |     18463 | 2019-09-11 | 7              | 33,34,35,36,40,55,57 | 4200      |
|       453458 |       5 |     12343 | 2021-07-03 | 4              | 3,4,5,8           | 7,300        |
|       589926 |      11 |     35479 | 2022-06-03 | 4              | 490 , 943 , 607 , 282 | 1400     |
|       678808 |       2 |     20973 | 2022-06-01 | 3              | 241 , 487 , 795   | 1380         |
|       773451 |       7 |     12720 | 2022-03-12 | 1              | 40                | 790          |
+--------------+---------+-----------+------------+----------------+-------------------+--------------+
12 rows in set (0.00 sec)

mysql>
```

Ticket table with schema and data

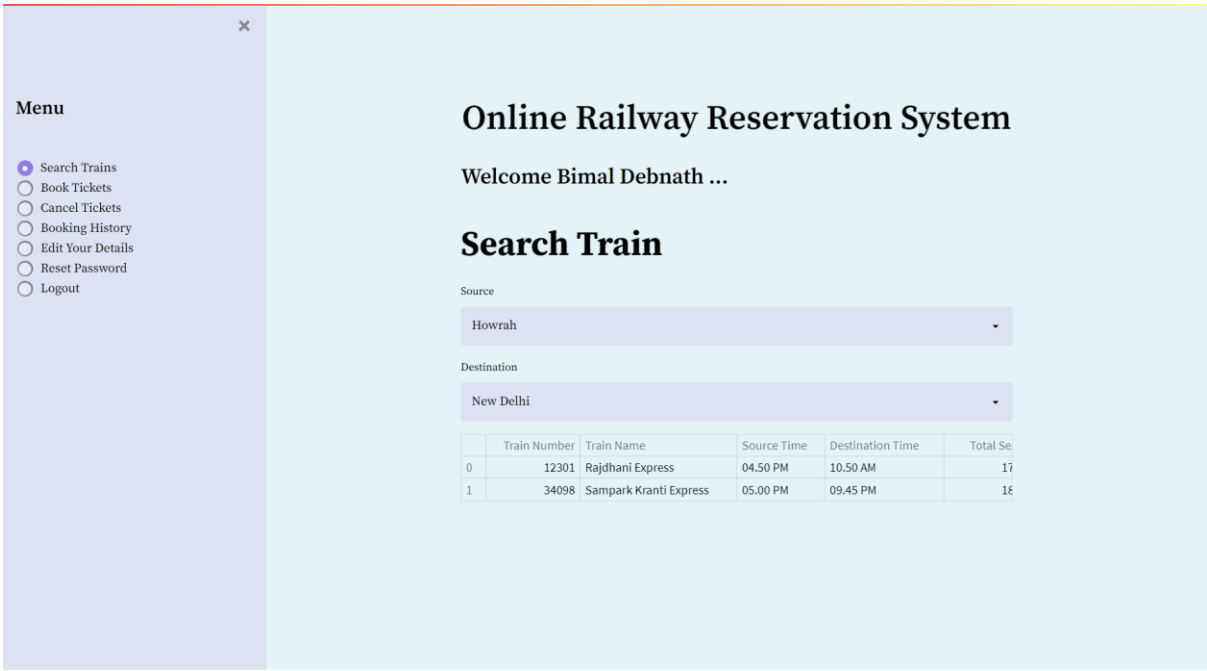
Frontend Images



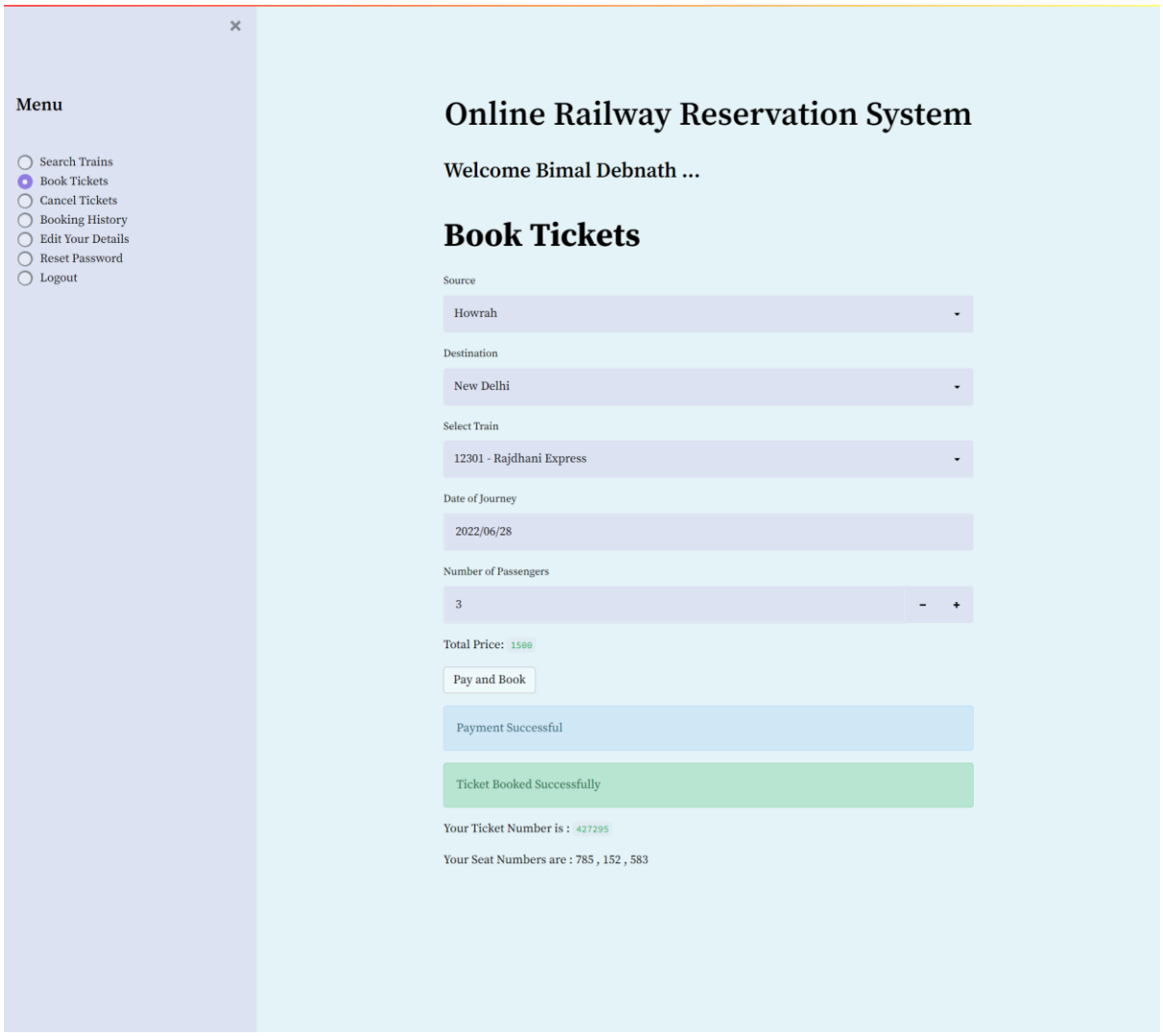
The Registration Form Page



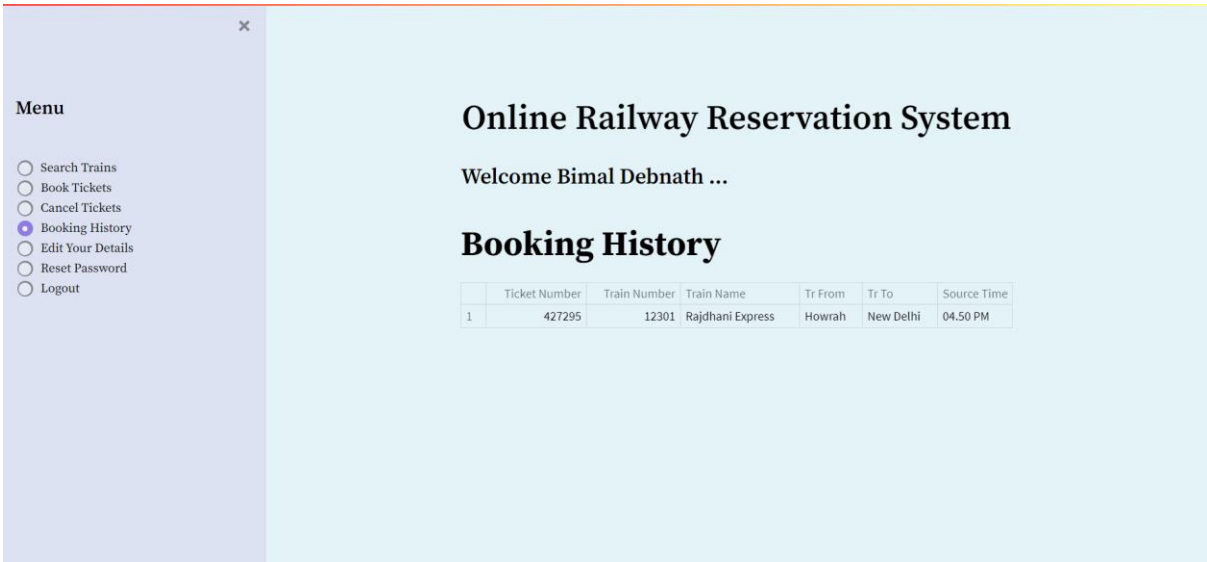
The Login Form Page



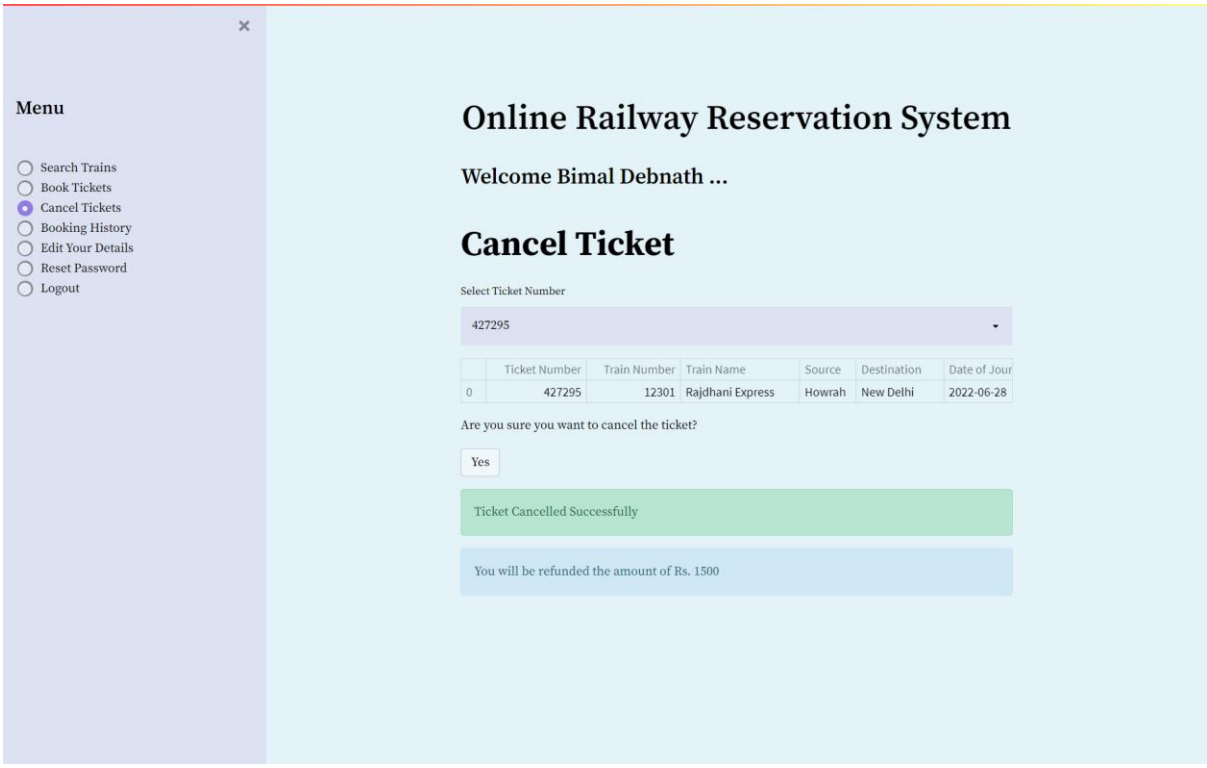
User Main Menu : Search Trains



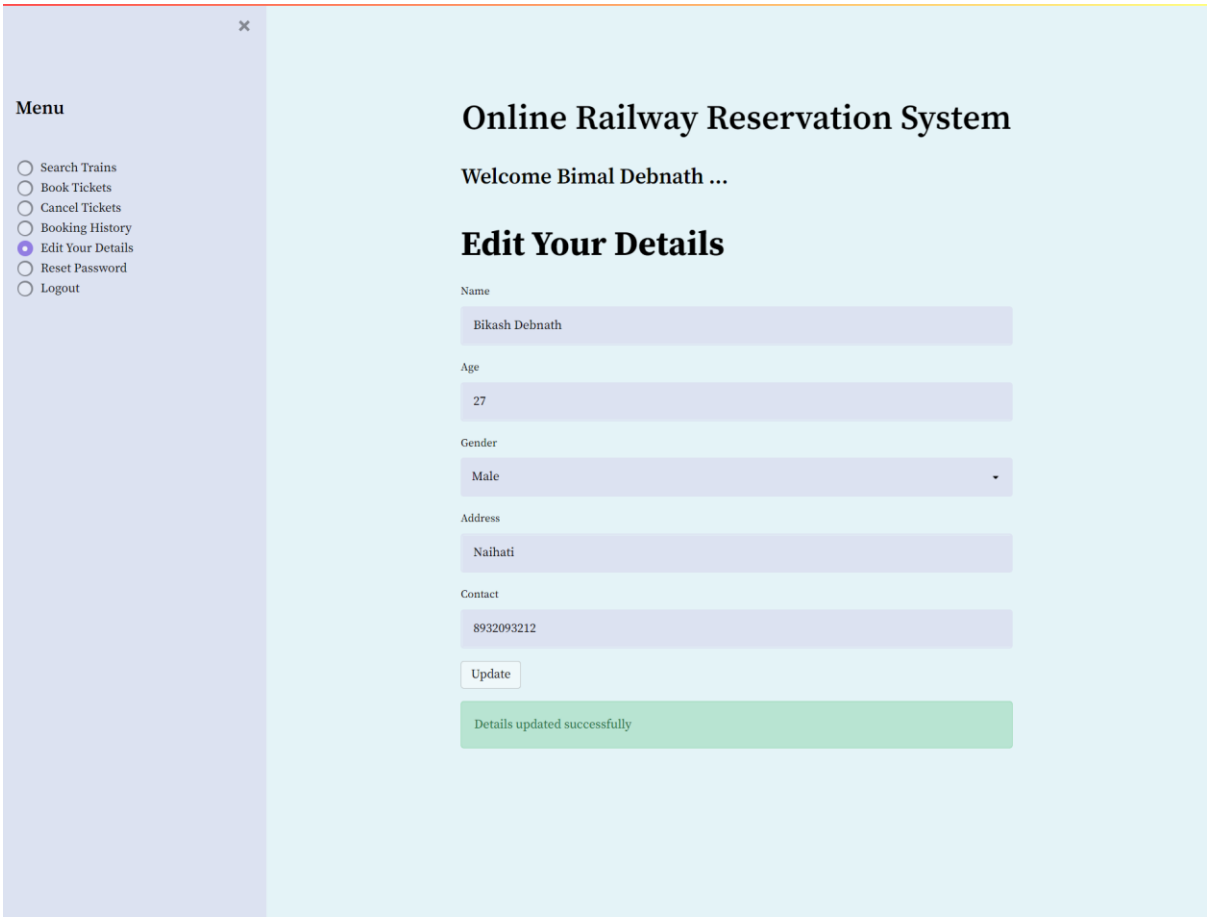
User Main Menu : Book Tickets



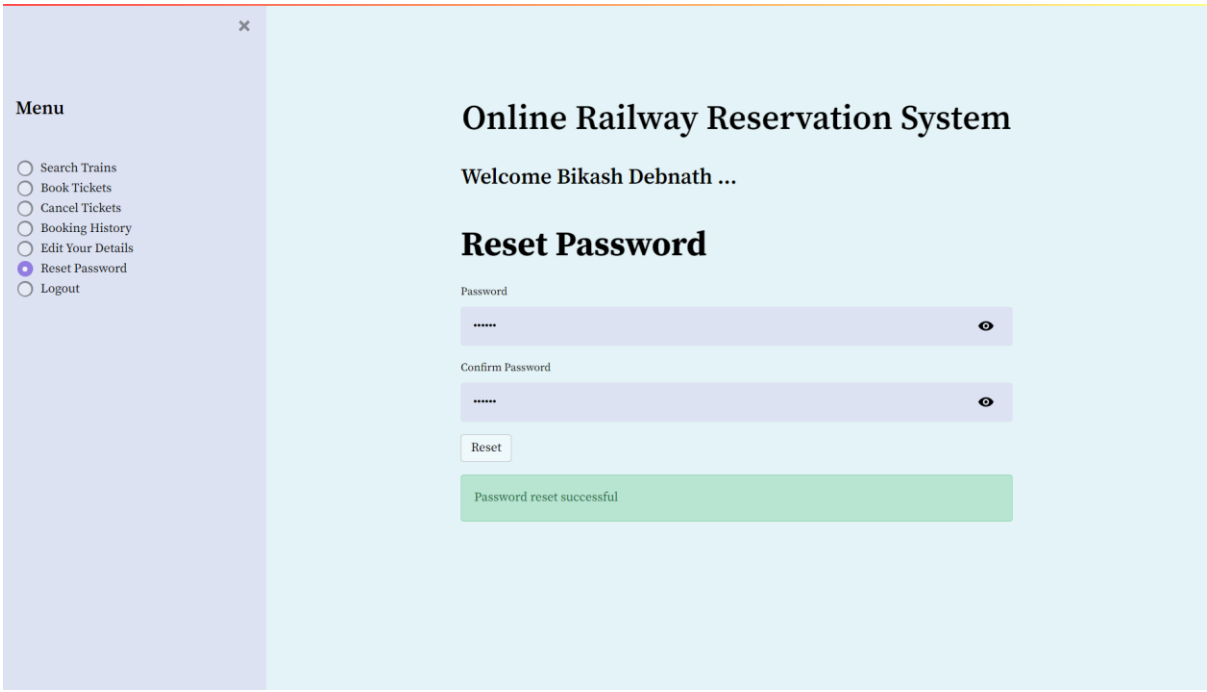
User Main Menu : Booking History



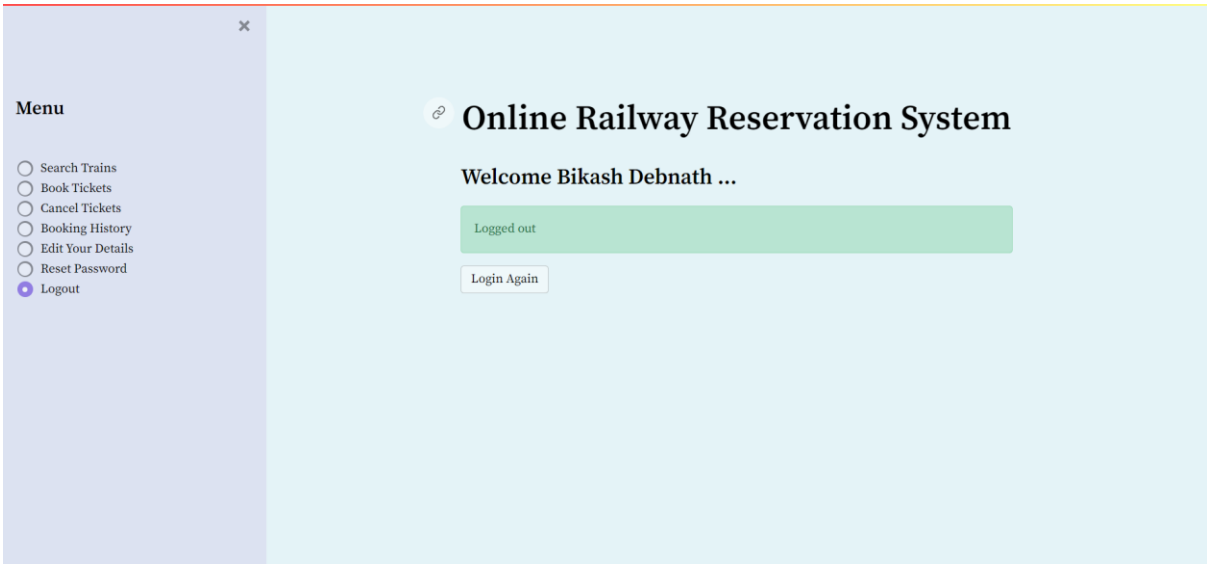
User Main Menu : Cancel Tickets



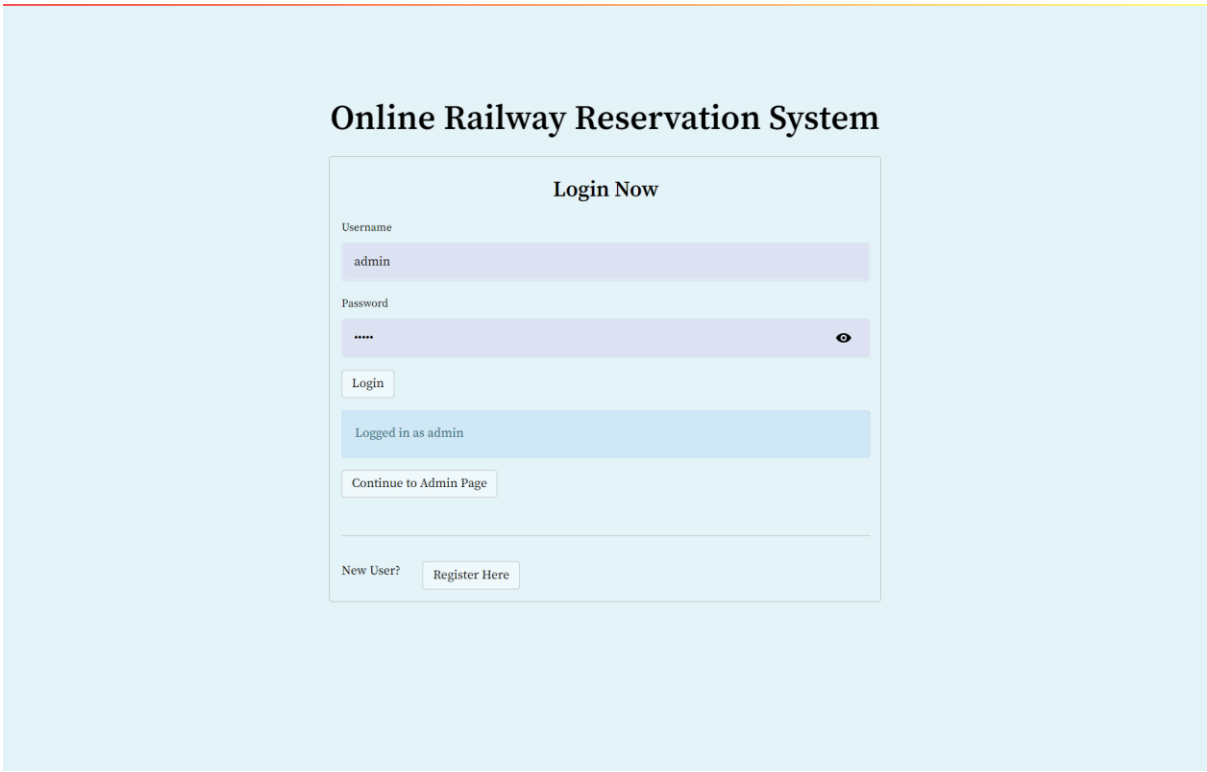
User Main Menu : Edit User details



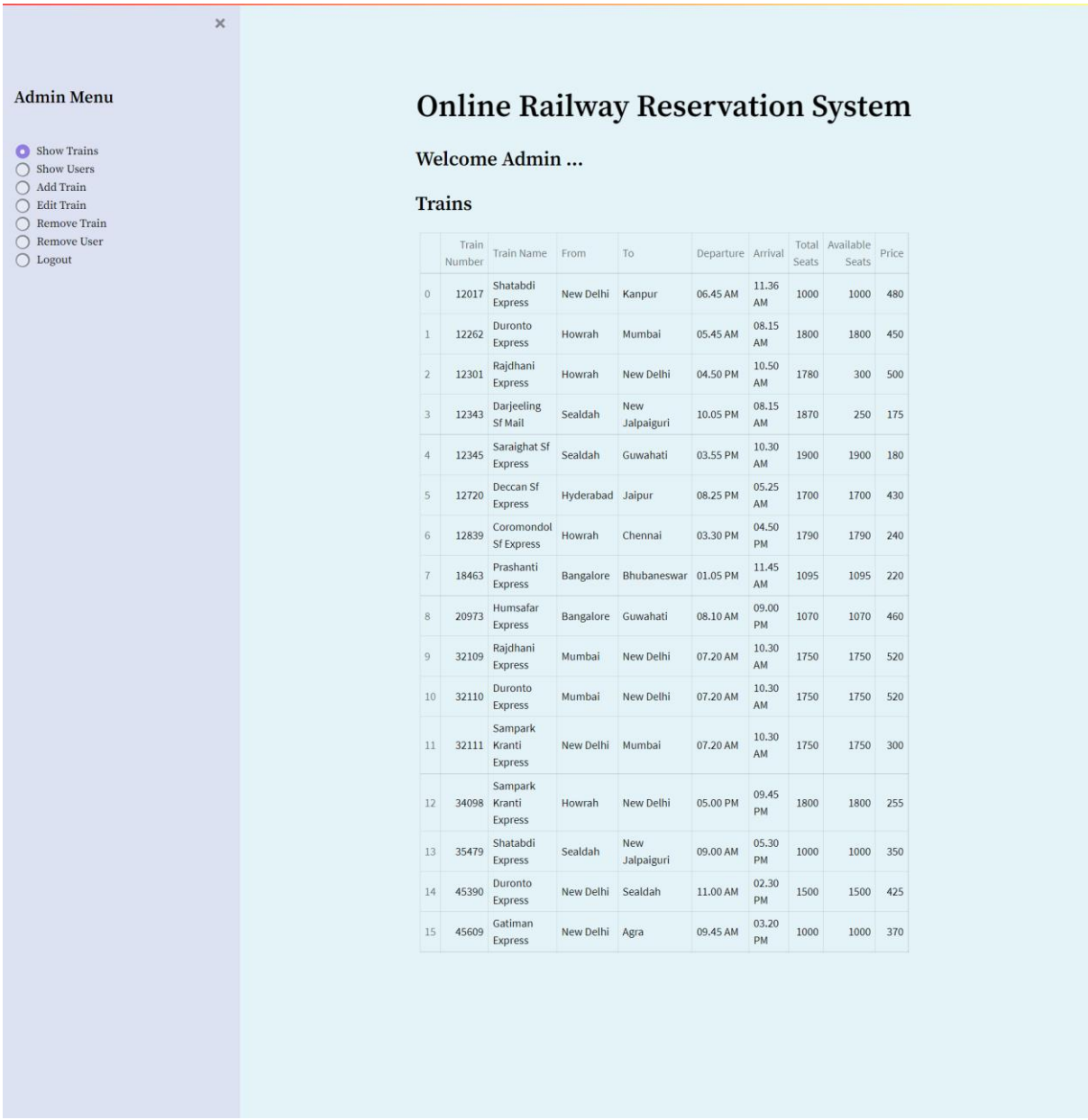
User Main Menu : Reset Password



User Main Menu : Log Out

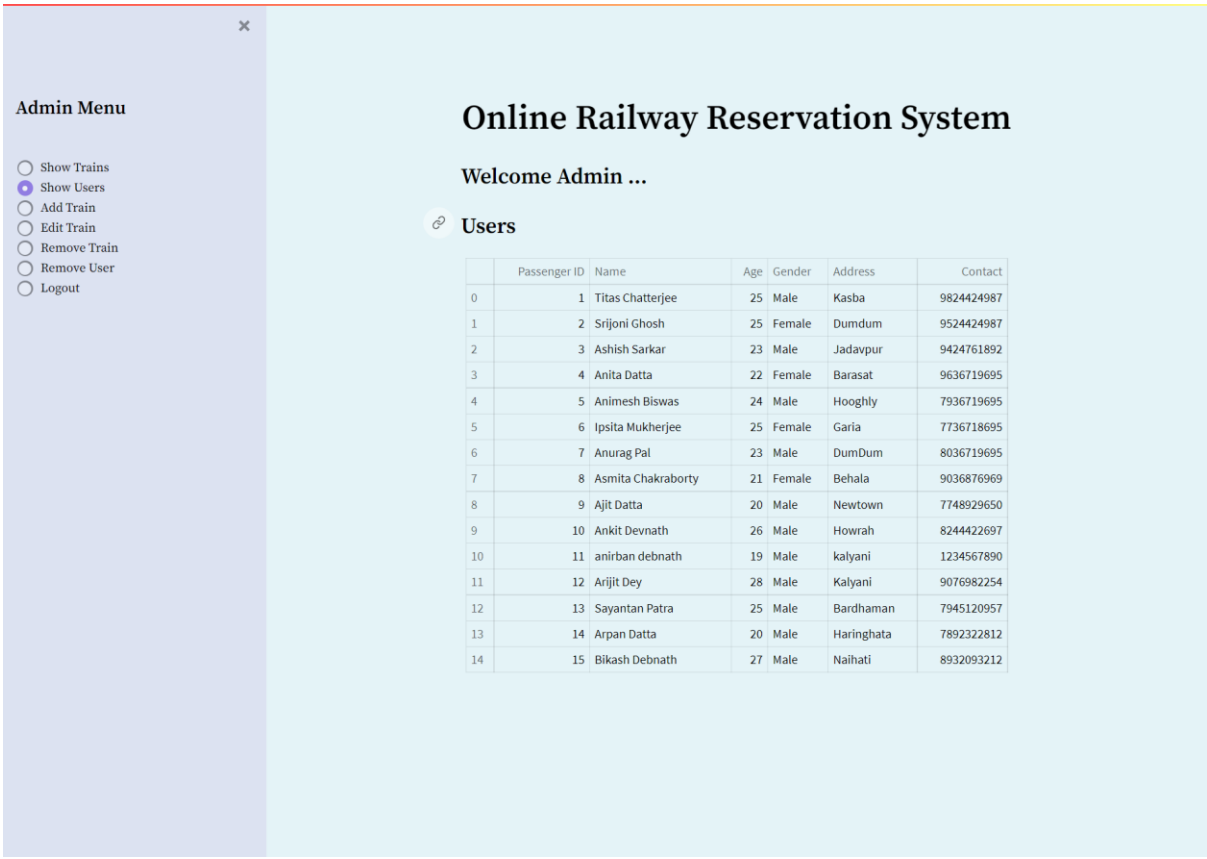


Login As Admin



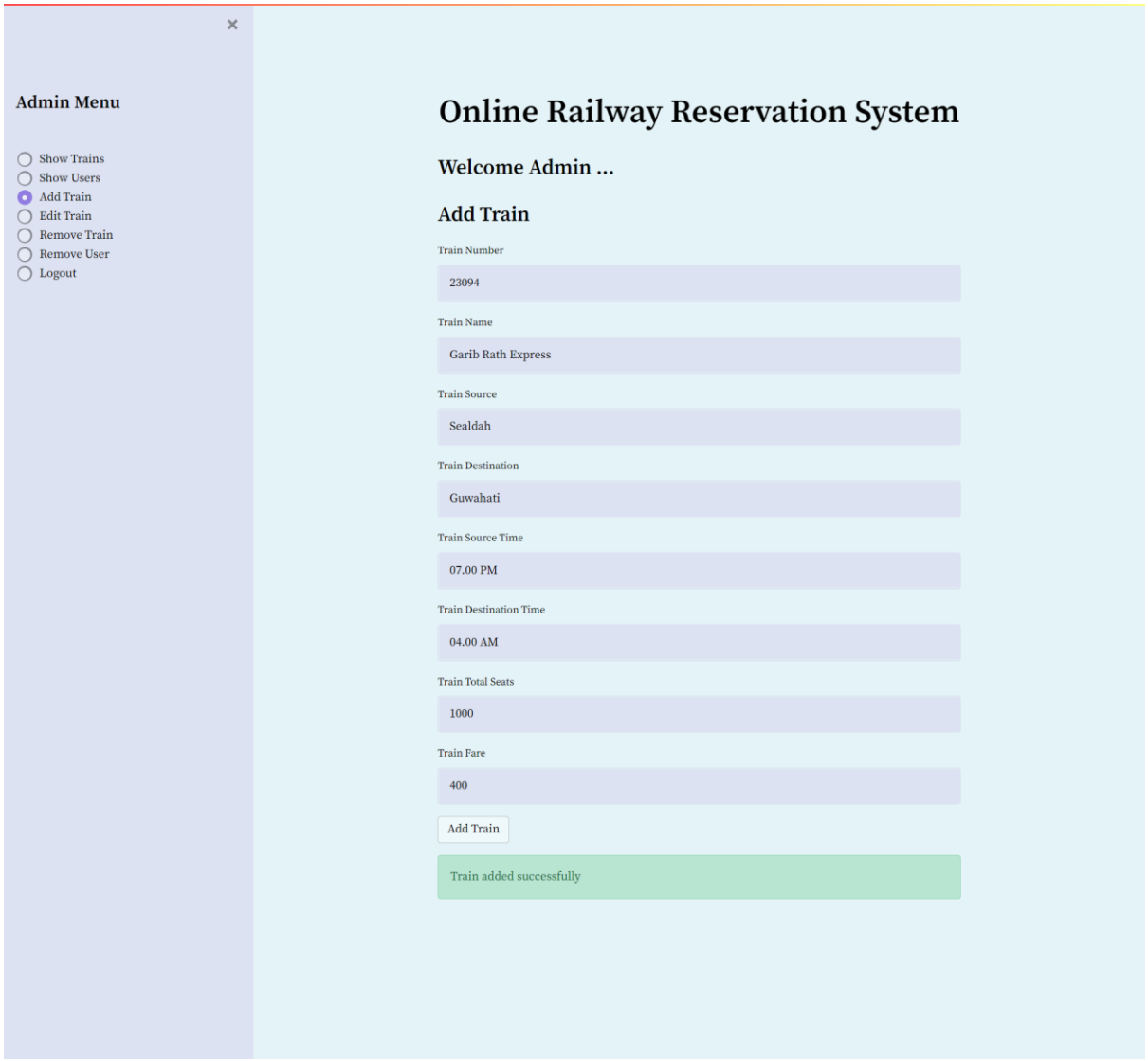
	Train Number	Train Name	From	To	Departure	Arrival	Total Seats	Available Seats	Price
0	12017	Shatabdi Express	New Delhi	Kanpur	06.45 AM	11.36 AM	1000	1000	480
1	12262	Duronto Express	Howrah	Mumbai	05.45 AM	08.15 AM	1800	1800	450
2	12301	Rajdhani Express	Howrah	New Delhi	04.50 PM	10.50 AM	1780	300	500
3	12343	Darjeeling Sf Mail	Sealdah	New Jalpaiguri	10.05 PM	08.15 AM	1870	250	175
4	12345	Saraighat Sf Express	Sealdah	Guwahati	03.55 PM	10.30 AM	1900	1900	180
5	12720	Deccan Sf Express	Hyderabad	Jaipur	08.25 PM	05.25 AM	1700	1700	430
6	12839	Coromondol Sf Express	Howrah	Chennai	03.30 PM	04.50 PM	1790	1790	240
7	18463	Prashanti Express	Bangalore	Bhubaneswar	01.05 PM	11.45 AM	1095	1095	220
8	20973	Humsafar Express	Bangalore	Guwahati	08.10 AM	09.00 PM	1070	1070	460
9	32109	Rajdhani Express	Mumbai	New Delhi	07.20 AM	10.30 AM	1750	1750	520
10	32110	Duronto Express	Mumbai	New Delhi	07.20 AM	10.30 AM	1750	1750	520
11	32111	Sampark Kranti Express	New Delhi	Mumbai	07.20 AM	10.30 AM	1750	1750	300
12	34098	Sampark Kranti Express	Howrah	New Delhi	05.00 PM	09.45 PM	1800	1800	255
13	35479	Shatabdi Express	Sealdah	New Jalpaiguri	09.00 AM	05.30 PM	1000	1000	350
14	45390	Duronto Express	New Delhi	Sealdah	11.00 AM	02.30 PM	1500	1500	425
15	45609	Gatiman Express	New Delhi	Agra	09.45 AM	03.20 PM	1000	1000	370

Admin Main Menu : Show Trains

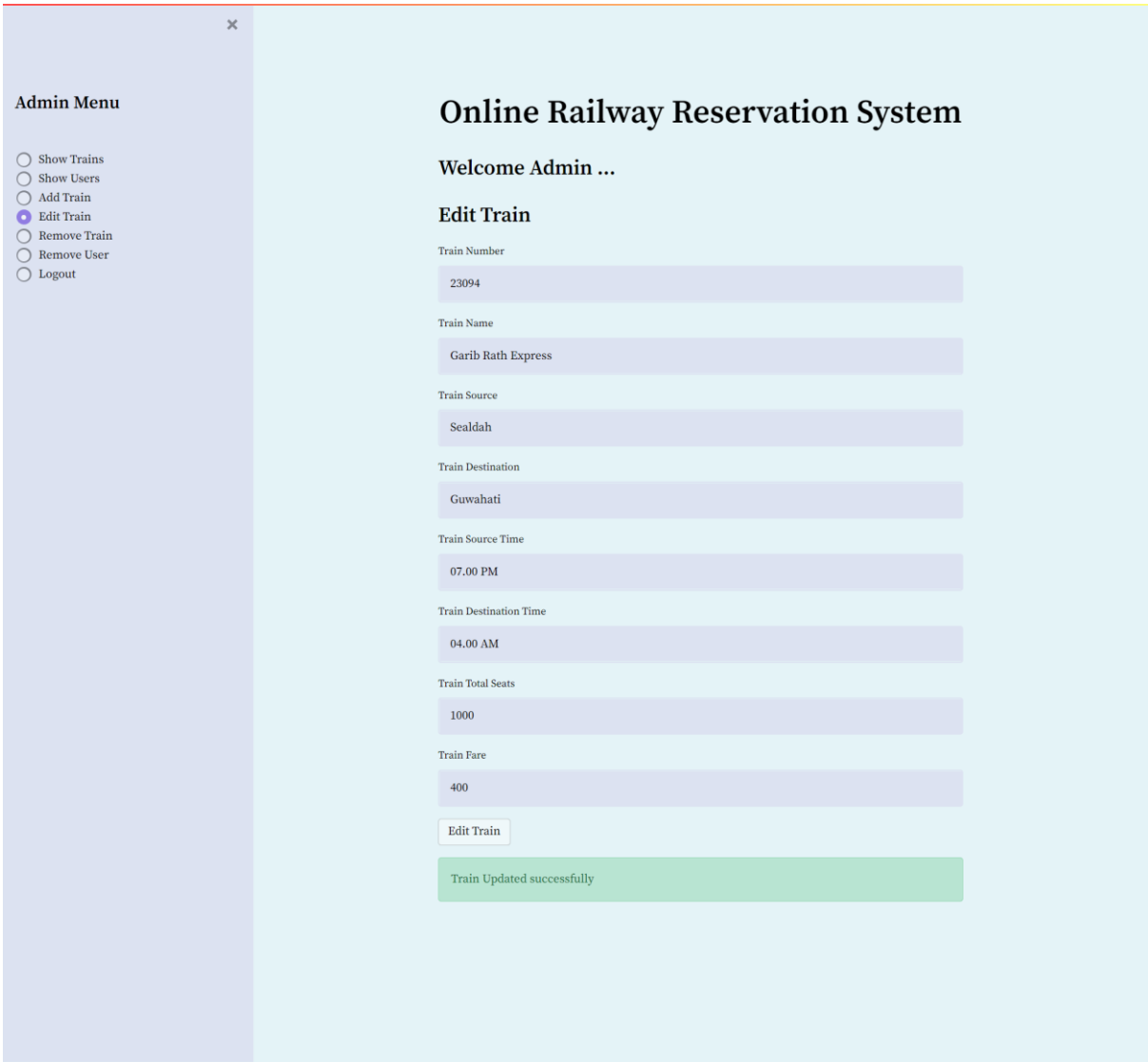


	Passenger ID	Name	Age	Gender	Address	Contact
0	1	Titas Chatterjee	25	Male	Kasba	9824424987
1	2	Srijoni Ghosh	25	Female	Dumdum	9524424987
2	3	Ashish Sarkar	23	Male	Jadavpur	9424761892
3	4	Anita Datta	22	Female	Barasat	9636719695
4	5	Animesh Biswas	24	Male	Hooghly	7936719695
5	6	Ipsita Mukherjee	25	Female	Garia	7736718695
6	7	Anurag Pal	23	Male	DumDum	8036719695
7	8	Asmita Chakraborty	21	Female	Behala	9036876969
8	9	Ajit Datta	20	Male	Newtown	7748929650
9	10	Ankit Devnath	26	Male	Howrah	8244422697
10	11	anirban debnath	19	Male	kalyani	1234567890
11	12	Arijit Dey	28	Male	Kalyani	9076982254
12	13	Sayantan Patra	25	Male	Bardhaman	7945120957
13	14	Arpan Datta	20	Male	Haringhata	7892322812
14	15	Bikash Debnath	27	Male	Naihati	8932093212

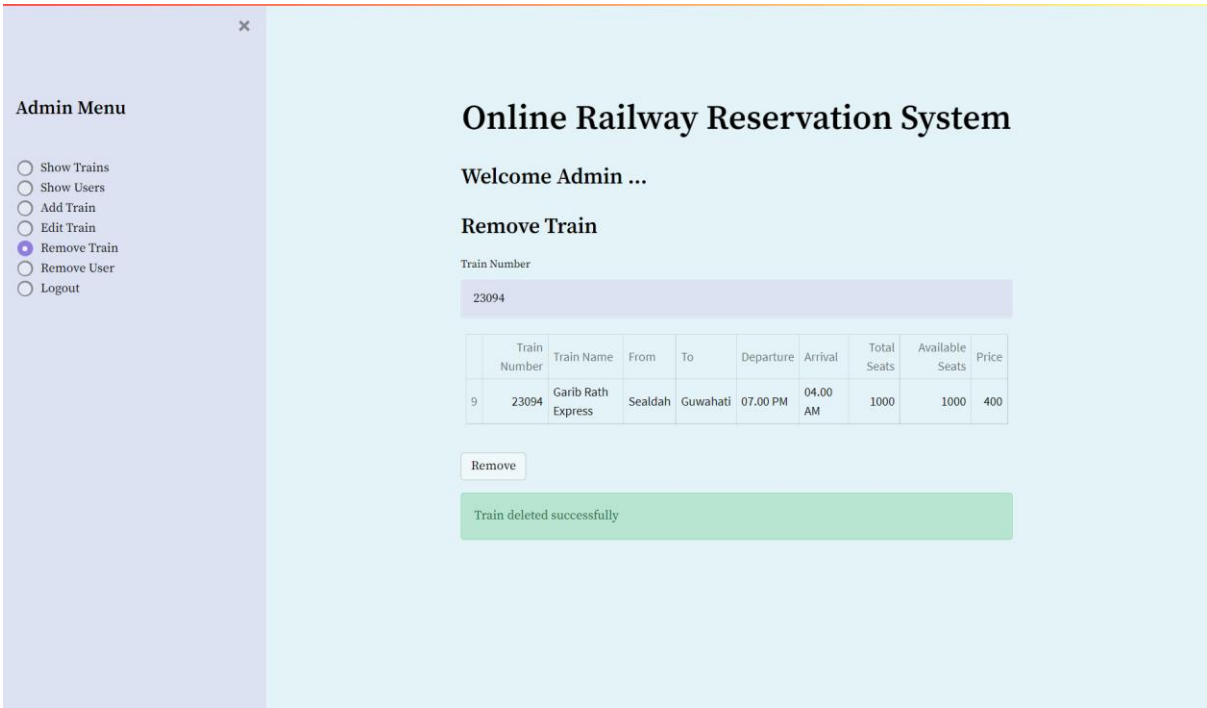
Admin Main Menu : Show Users



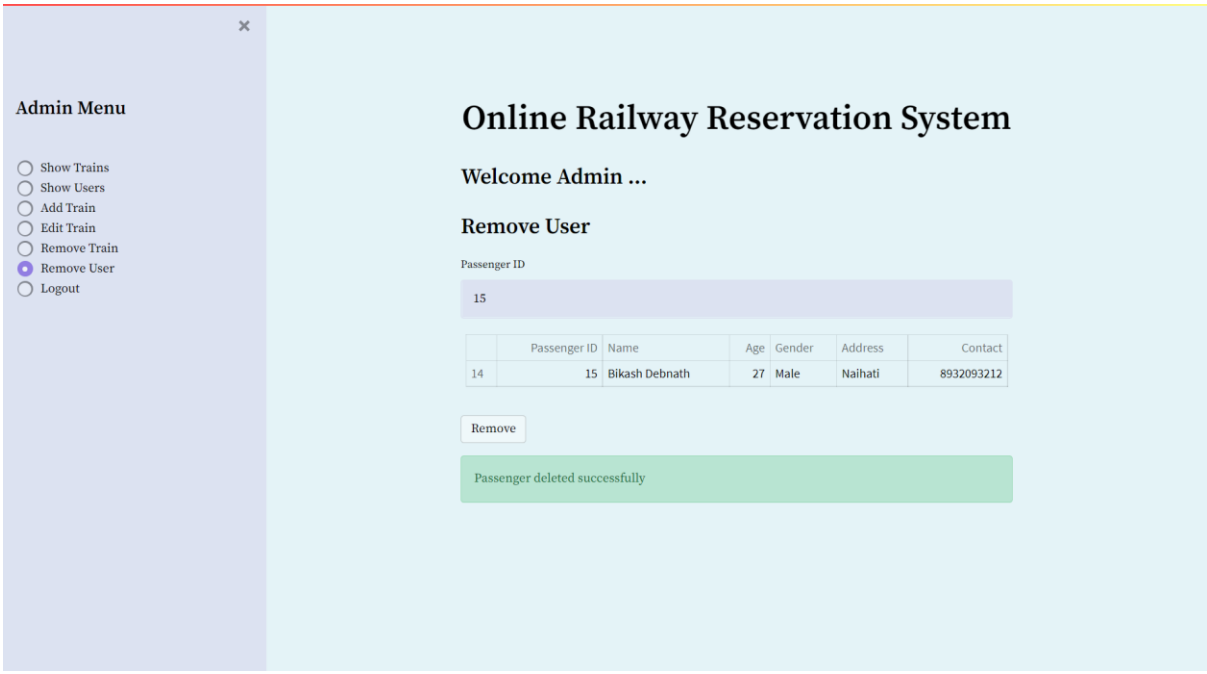
Admin Main Menu : Add a Train



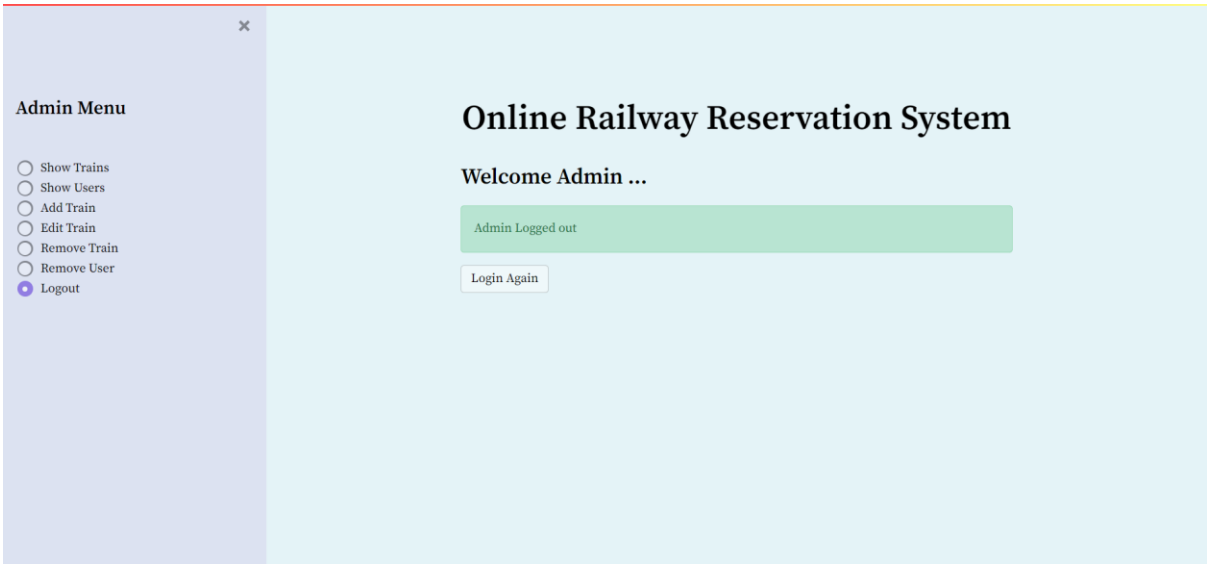
Admin Main Menu : Update a Train



Admin Main Menu : Remove a Train



Admin Main Menu : Remove a User



Admin Main Menu : Admin Log Out

Testing

Software testing is **the process of evaluating and verifying that a software product or application does what it is supposed to do**. The benefits of testing include preventing bugs, reducing development costs and improving performance.

There are 4 types of testing we have done –

1. Unit Testing
2. Integration Testing
3. System Testing
4. Acceptance Testing

Unit Testing

Starting from the bottom the first test level is "Unit Testing". It involves checking that each feature specified in the "Component Design" has been implemented in the component. In theory an independent tester should do this, but in practice the developer usually does it, as they are the only people who understand how a component works. The problem with a component is that it performs only a small part of the functionality of a system, and it relies on co-operating with other parts of the system, which may not have been built yet. To overcome this, the developer either builds, or uses special software to trick the component into believe it is working in a fully functional system.

The summary of unit tests is provided below:

For User :

Unit	Purpose	Tested
Search Trains	To search available trains between source and destination.	Yes
Book Ticket	To book a train reservation ticket for a particular date.	Yes
Cancel Ticket	To cancel a existing ticket but for future journey.	Yes
Booking History	To check the booking history of a user from his registration.	Yes
Edit Details	Edit user personal details.	Yes
Reset Password	Reset user login password.	Yes
Logout	Log out	Yes

For Admin :

Unit	Purpose	Tested
Show Trains	Show all trains details from the database.	Yes
Show Users	Show all users details from the database.	Yes
Add Train	Add a train into the database.	Yes
Edit Train	Edit or Update train details.	Yes
Remove Train	Remove a train entirely from database.	Yes
Remove User	Remove a user entirely from database.	Yes
Logout	Log Out	Yes

Integration Testing

As the components are constructed and tested they are then linked together to check if they work with each other. It is a fact that two components that have passed all their tests, when connected to each other produce one new component full of faults. These tests can be done by specialists, or by the developers. Integration Testing is not focused on what the components are doing but on how they communicate with each other, as specified in the "System Design". The "System Design" defines relationships between components. The tests are organized to check all the interfaces, until all the components have been built and interfaced to each other producing the whole system. Thus this test was successfully done. No conflicts or inconsistencies were detected.

System Testing

Once the entire system has been built then it has to be tested against the "System Specification" to check if it delivers the features required. It is still developer focused, although specialist developers known as systems testers are normally employed to do it. In essence System Testing is not about checking the individual parts of the design, but about checking the system as a whole. In fact it is one giant component.

System testing can involve a number of specialist types of test to see if all the functional and non - functional requirements have been met.

In addition to functional requirements these may include the following types of testing for the non - functional requirements:

- Performance - Are the performance criteria met?
- Volume - Can large volumes of information be handled?
- Stress - Can peak volumes of information be handled?
- Documentation - Is the documentation usable for the system?
- Robustness - Does the system remain stable under adverse circumstances?

The system was found to perform its function properly under all conditions.

Acceptance Testing

Acceptance Testing checks the system against the "Requirements". It is similar to systems testing in that the whole system is checked but the important difference is the change in focus:

Systems testing checks that the system that was specified has been delivered. Acceptance Testing checks that the system will deliver what was requested. The customer should always do acceptance testing and not the developer. The customer knows what is required from the system to achieve value in the business and is the only person qualified to make that judgment. This testing is more of getting the answer for whether is the software delivered as defined by the customer. It's like getting a green flag from the customer that the software is up to the expectation and ready to be used.

Results

Online railway ticket reservation system was successfully designed and developed as per the specifications. The system was found to work satisfactorily without any errors under all conditions.

Conclusion

Railway reservation system is one of the most emerging System which is very very helpful for every rail passenger. The main aim of developing Reservation system is to provide all information that is required by the users. User friendliness is a must that is the user must get the details without complicated searching procedures. Other important requirements of software are data security, extensibility and maintainability. All these features are included in this web application.

Working through this kind of project we learns a lot about Databases, Database management systems, relational models, Different Database Views , Queries handling, Streamlit software, MySQL server, front end and back end handling, SRS, ER diagram and a great practical experience during the project.