



DATABASE AND MANAGEMENT SYSTEMS

PROJECT

RAILWAY RESERVATION SYSTEM



GROUP 53

MEMBERS:

Krishnam Omar | Gurmehak Kaur | Ishita Sindhwani | Darsh Parikh
2020309 | 2020298 | 2020305 | 2020560



Project report

Database Management Systems

Scope of project:

Project goal:

To build an interactive web application for the train reservation system. It would be based on the IRCTC website and enable people to make reservations and payments for the railway services online with ease.

Deliverables:

Our application would provide various facilities to the users. They will be required to sign up by providing some compulsory information for our database to create their account. It will allow them to look up for different trains and their prices for the travel route. It would enable users to choose the train they want to book, pay for it online, check the travel time, select AC/non AC coaches for travel and also to cancel the ticket. Moreover, our application would have features that will let user see the arrival and departure time and which terminal and station will be the final stop.

Tasks:

Our tasks entail collecting and storing user information to ensure smooth functioning. It would also require us to store information about various trains, their routes and timings to show the user relevant information when they search for it. Moreover, tracking the number of seats left in various trains and their routes would be required to check how many seats are available for sale.

Processes:

It would require us to create a database and various tables to store and edit the information via user input and to update the values for smooth and error free functioning.



Assumptions:

For a user to use our application they must be literate, be able to use a computer and know online banking (have a bank account) to make payments.

Stakeholders:

The users who will book the tickets will be the main stakeholders. Moreover, the railway department and government will also be key stakeholders.

Entities:

Strong entities:

- Train: Primary key is Train_id
- User: Primary key is Adhaar_no
- Station: Primary key is Station_id

Weak entities:

- Passenger : Is a weak entity since it depends on the User entity.
- Route : Is a weak entity since it depends on the train entity.
- Coach : Is a weak entity since it depends on the Train entity.
- Terminal : Is a weak entity since it depends on the Station entity.

Entity constraints:

All the primary keys (underlined) are unique and non null.

Primary keys in various tables are:

Adhaar

Route_id

Ticket_id

Station_id

Terminal_id

Train_id



Coach_id

All foreign keys correspond to some primary key in the tables.

Foreign keys in various tables are:

Start_station_id
End_station_id
Start_terminal_id
End_terminal_id
Adhaar_no
Coach_id
Destination_id

The table specifies the type of data an attribute can store like varchar, int , et cetera.

On deletion of the primary key, foreign key also gets deleted by cascading.

Relationships established, roles and entity participation type:

- Boards : Relationship between passenger, train and route as passenger boards a train for a particular route.
- Consists: Relationship between train and coach as train consists of various coaches.
- Has: Relationship between terminal and station as station has various terminals.
- Travels: Relationship between Station and route as train stops at various stations in its route.
- Is A: Relationship between user and passenger as a person with an account only can book a ticket.

- One to many:
 - User and Passenger
 - Station and Terminal
 - Train and coach
 - Coach and passenger

- Many to many:



- Route and Terminal
- The entities Coach and Train are related via the Consists relationship. The participation of Train and Coach in Consists is total relationship. Hence, there is total participation.
- The entities Train, Route and Passenger are related via the Boards relationship. There is total participation of Route and Passenger in Boards whereas there is partial participation of Train.
- The entities Route and Terminal are related via the Travels relationship. There is total participation of Route whereas partial participation of Terminal.
- The entities Users and Passenger are related via the Is A relationship. There is total participation of Passenger whereas partial participation of Users.

Ternary relationship:

Boards : It is a ternary relationship because it establishes a relationship between passenger, train and the route.

Data Population:

We used the following for generating data for database:

<https://generatedata.com/generator>

Tables creation SQL:



```
drop database Railway_System;
create database Railway_System;
show databases;

use railway_system;
create table Train(
    Train_id int primary key,
    Train_name varchar(250),
    Total_seats int,
    Available_seats int,
    Arrival_time datetime,
    Departure_time datetime
);

create table Coach(
    Coach_id int primary key,
    Coach_name varchar(50),
    Number_of_Seats int,
    Train_id int,
    foreign key(Train_id) references Train(Train_id) on delete cascade
);

ALTER TABLE Coach
ADD Train_name varchar(250);
```



```
ALTER TABLE Train
ADD Source_id int;

ALTER TABLE Train
ADD Destination_id int;

create table Station(
    Station_id int primary key,
    Station_name varchar(250),
    No_of_terminals int
);

ALTER TABLE Train
ADD FOREIGN KEY(Source_id)
REFERENCES Station(Station_id)
ON DELETE SET NULL;

ALTER TABLE Train
ADD FOREIGN KEY(Destination_id)
REFERENCES Station(Station_id)
ON DELETE SET NULL;

create table Terminal(
    Terminal_id int primary key,
    Station_id int,
    foreign key(Station_id) references Station(Station_id) on delete cascade
);
```



```
create table Users(
    Adhaar_no int primary key,
    Username varchar(250),
    e_mail varchar(250),
    Mobile int,
    DOB date,
    First_name varchar(250),
    Last_name varchar(250)
);

create table Route(
    Route_id int primary key,
    Number_of_seats int,
    Start_terminal_id int,
    End_terminal_id int,
    Arrival_time datetime,
    Departure_time datetime,
    Start_station_id int,
    End_station_id int,
    foreign key(Start_station_id) references Station(Station_id) on delete set null,
    foreign key(End_station_id) references Station(Station_id) on delete set null,
    foreign key(Start_terminal_id) references Terminal(Terminal_id) on delete set null,
    foreign key(End_terminal_id) references Terminal(Terminal_id) on delete set null
);
```

```
ALTER TABLE Route
ADD Train_id int;

ALTER TABLE Route
ADD FOREIGN KEY(Train_id)
REFERENCES Train(Train_id)
ON DELETE SET NULL;

create table Passenger(
    Ticket_id int primary key,
    Adhaar_no int,
    foreign key(Adhaar_no) references Users(Adhaar_no) on delete cascade,
    Date_of_Booking datetime,
    Coach_id int,
    foreign key(Coach_id) references Coach(Coach_id) on delete set null,
    Route_id int,
    foreign key(Route_id) references Route(Route_id) on delete set null,
    Start_station_id int,
    End_station_id int,
    foreign key(Start_station_id) references Station(Station_id) on delete set null,
    foreign key(End_station_id) references Station(Station_id) on delete set null,
    Start_terminal_id int,
    End_terminal_id int,
    foreign key(Start_terminal_id) references Terminal(Terminal_id) on delete set null,
    foreign key(End_terminal_id) references Terminal(Terminal_id) on delete set null
);
```



```
ALTER TABLE Route
DROP Number_of_seats;

ALTER TABLE Route
ADD Seats_General int;
ALTER TABLE Route
ADD Seats_AC1 int;
ALTER TABLE Route
ADD Seats_AC2 int;

Alter table Passenger
Add Train_id int;

ALTER TABLE Passenger
ADD FOREIGN KEY(Train_id)
REFERENCES Train(Train_id)
ON DELETE SET NULL;

select * from Train;
select * from Route;
select * from Coach;
select * from Station;
select * from Terminal;
select * from Users;
select * from Passenger;
show tables;
```



TABLES WITH DATA:

Coach:

Coach_id	Coach_name	Number_of_Seats	Train_id	Train_name
103821	Genral	60	10382	"Chennai Mail"
103822	AC_1	60	10382	"Chennai Mail"
103823	AC_2	60	10382	"Chennai Mail"
109301	Genral	80	10930	"Gujarat Express"
109302	AC_1	70	10930	"Gujarat Express"
119631	Genral	70	11963	"Jhelum Express"
119632	AC_1	40	11963	"Jhelum Express"
119633	AC_2	30	11963	"Jhelum Express"
127031	Genral	200	12703	"Rajdhani Express"
144841	Genral	90	14484	"Shatabdi Express"
144842	AC_1	100	14484	"Shatabdi Express"
165191	Genral	100	16519	"Shan-e-Punjab Express"
165192	AC_1	70	16519	"Shan-e-Punjab Express"
168471	Genral	60	16847	"Goa Express"
168472	AC_1	50	16847	"Goa Express"
168473	AC_2	50	16847	"Goa Express"

Passenger:

Ticket_id	Adhaar_no	Date_of_Booking	Coach_id	Route_id	Start_station_id	End_station_id	Start_terminal_id	End_terminal_id	Train_id
112219	546789017	"2023-01-04 15:33:04"	127031	1270301	100	101	1001	1011	12703
114603	100012389	"2021-07-01 03:09:38"	168473	1684702	112	116	1122	1161	16847
178050	909911345	"2022-11-21 09:10:00"	144842	1448401	100	108	1001	1082	14484
178949	988100984	"2022-09-06 20:50:32"	165191	1448403	106	109	1061	1092	16519



Route:

Route_id	Start_terminal_id	End_terminal_id	Arrival_time	Departure_time	Start_station_id	End_station_id	Train_id	Seats_General	Seats_AC1	Seats_AC2
1038201	1092	1102	"2022-04-19 07:46:22"	"2022-04-19 08:06:12"	109	110	10382	60	60	60
1038202	1102	1111	"2022-04-19 09:15:32"	"2022-04-19 09:32:17"	110	111	10382	60	60	60
1038203	1111	1121	"2022-04-19 12:42:17"	"2022-04-19 13:00:57"	111	112	10382	60	60	60
1038204	1121	1132	"2022-04-19 15:25:17"	"2022-04-19 15:52:58"	112	113	10382	60	60	60
1038205	1132	NULL	"2022-04-19 17:14:21"	NULL	113	NULL	10382	0	0	0
1093001	1192	1021	"2022-02-18 11:26:36"	"2022-02-18 11:35:36"	119	102	10930	80	70	0
1093002	1021	1182	"2022-02-18 13:06:36"	"2022-02-18 13:27:56"	102	118	10930	80	70	0
1093003	1182	1132	"2022-02-18 15:56:37"	"2022-02-18 16:29:25"	118	113	10930	80	70	0
1093004	1132	NULL	"2022-02-18 20:12:09"	NULL	113	NULL	10930	80	70	0
1196301	1122	1151	"2022-09-06 22:06:30"	"2022-09-06 22:19:37"	112	115	11963	70	40	30
1196302	1151	1202	"2022-09-07 13:09:29"	"2022-09-07 13:26:49"	115	120	11963	70	40	30
1196303	1202	1001	"2022-09-07 17:41:59"	"2022-09-07 18:02:06"	120	100	11963	70	40	30
1196304	1001	1211	"2022-09-08 00:21:03"	"2022-09-08 00:57:43"	100	121	11963	70	40	30
1196305	1211	NULL	"2022-09-08 10:10:32"	NULL	121	NULL	11963	0	0	0
1270301	1001	1011	"2022-07-21 10:31:31"	"2022-07-21 11:05:31"	100	101	12703	199	0	0
1270302	1011	1021	"2022-07-22 12:31:55"	"2022-07-22 13:00:31"	101	102	12703	199	0	0
1270303	1021	1032	"2022-07-22 15:15:55"	"2022-07-22 15:32:55"	102	103	12703	199	0	0
1270304	1032	1041	"2022-07-22 16:15:55"	"2022-07-22 16:40:55"	103	104	12703	199	0	0
1270305	1041	NULL	"2022-07-22 17:50:55"	NULL	104	NULL	12703	0	0	0
1448401	1001	1051	"2022-01-09 02:44:06"	"2022-01-09 03:02:23"	100	105	14484	90	99	0
1448402	1051	1061	"2022-01-09 03:41:46"	"2022-01-09 03:58:21"	105	106	14484	90	99	0
1448403	1061	1072	"2022-01-09 05:15:09"	"2022-01-09 05:30:17"	106	107	14484	90	99	0
1448404	1072	1082	"2022-01-09 06:18:47"	"2022-01-09 06:38:46"	107	108	14484	90	99	0
1448405	1082	NULL	"2022-01-09 07:11:51"	NULL	108	NULL	14484	0	0	0
1651901	1172	1001	"2022-08-25 11:42:13"	"2022-08-25 11:50:03"	117	100	16519	100	70	0
1651902	1001	1011	"2022-08-25 18:32:53"	"2022-08-25 18:51:23"	100	101	16519	100	70	0
1651903	1011	1181	"2022-08-25 18:32:53"	"2022-08-25 18:50:06"	101	118	16519	100	70	0
1651904	1181	1131	"2022-08-25 23:21:53"	"2022-08-26 00:05:53"	118	113	16519	100	70	0
1651905	1131	NULL	"2022-08-26 04:17:43"	NULL	113	NULL	16519	0	0	0
1684701	1141	1122	"2022-11-07 16:35:00"	"2022-11-07 17:00:00"	114	112	16847	60	50	50
1684702	1122	1152	"2022-11-08 07:46:31"	"2022-11-08 08:14:41"	112	115	16847	60	50	49
1684703	1152	1161	"2022-11-09 10:44:41"	"2022-11-09 11:25:33"	115	116	16847	60	50	49
1684704	1161	1001	"2022-11-09 18:19:43"	"2022-11-09 18:36:03"	116	100	16847	60	50	50
1684705	1001	NULL	"2022-11-09 20:16:53"	NULL	100	NULL	16847	0	0	0

**Station:**

Station_id	Station_name	No_of_terminals
100	New_Delhi	1
101	Kota	1
102	Vadodara	1
103	Panvel	2
104	Madgaon	2
105	Meerut	1
106	Roorkee	2
107	Haridwar	2
108	Dehradun	2
109	Chennai	2
110	Guntakal	2
111	Wadi	1
112	Pune	2
113	Mumbai	2
114	Vasco	2
115	Bhopal	2
116	Agra	1
117	Amritsar	2
118	Surat	2
119	Ahmedabad	2
120	Jhansi	2
121	Jammu	1

**Terminal:**

Terminal_id	Station_id
1001	100
1011	101
1021	102
1031	103
1032	103
1041	104
1042	104
1051	105
1061	106
1062	106
1071	107
1072	107
1081	108
1082	108
1091	109
1092	109
1101	110
1102	110
1111	111
1121	112
1122	112
1131	113
1132	113
1141	114
1142	114
1151	115
1152	115
1161	116
1171	117
1172	117
1181	118
1182	118
1191	119
1192	119
1201	120
1202	120
1211	121



Train:

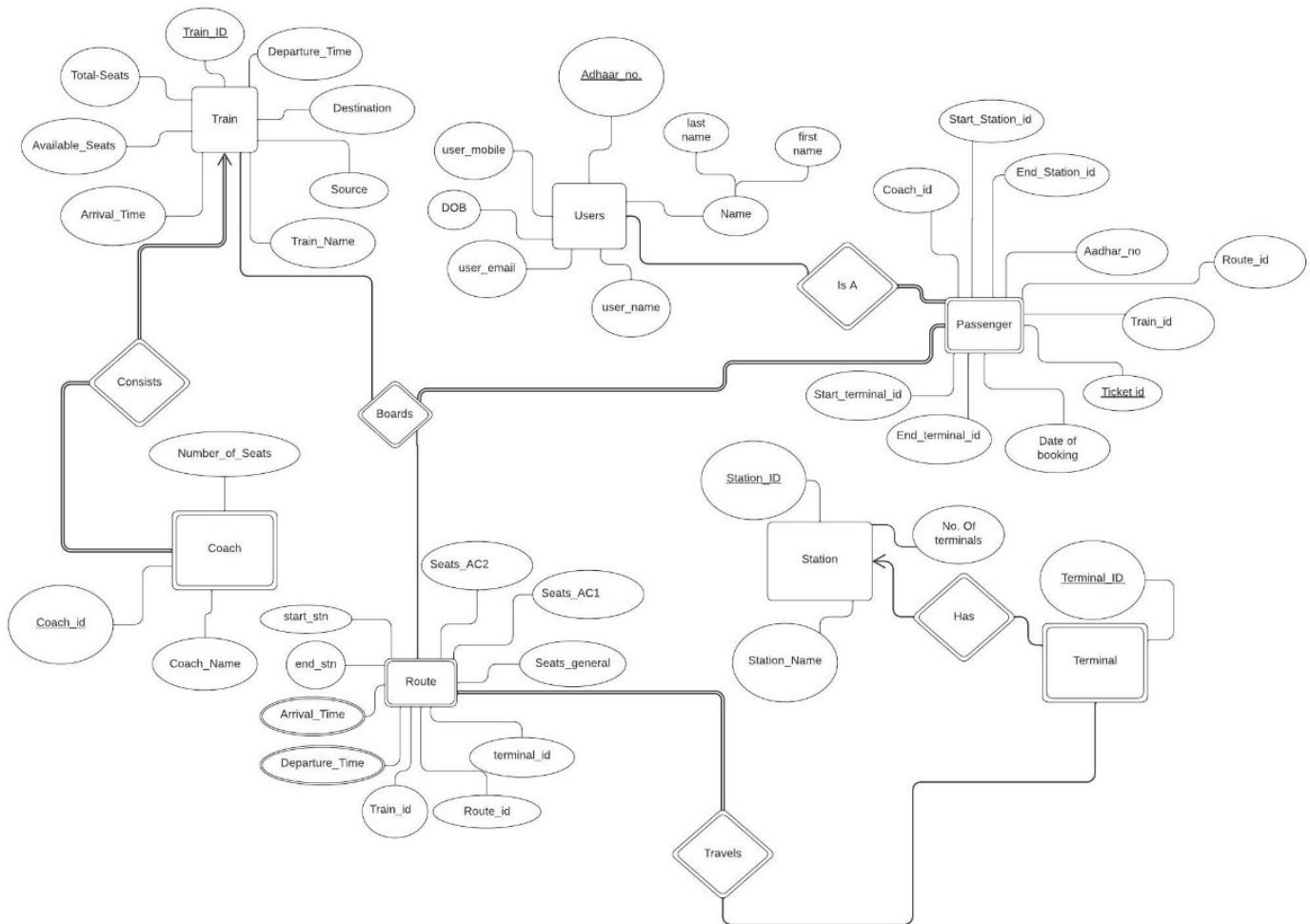
Train_id	Train_name	Total_seats	Available_seats	Arrival_time	Departure_time	Source_id	Destination_id
10382	"Chennai Mail"	180	103	"2022-04-19 17:14:21"	"2022-04-19 08:06:12"	109	113
10930	"Gujarat Express"	150	91	"2022-02-18 20:12:09"	"2022-02-18 11:35:36"	119	113
11963	"Jhelum Express"	140	71	"2022-09-08 10:10:32"	"2022-09-06 22:19:37"	112	121
12703	"Rajdhani Express"	200	67	"2022-07-22 17:50:55"	"2022-07-22 11:05:31"	100	104
14484	"Shatabdi Express"	190	101	"2022-01-09 07:11:51"	"2022-01-09 03:02:23"	100	108
16519	"Shan-e-Punjab Express"	170	158	"2022-08-26 04:17:43"	"2022-08-25 11:50:03"	117	113
16847	"Goa Express"	160	83	"2022-11-09 20:16:53"	"2022-11-07 17:00:00"	114	100

User:

Adhaar_no	Username	e_mail	Mobile	DOB	First_name	Last_name
100012389	Iknoxx_00	Iknoxxx@aol.edu	566872435	11-08-1999	Ivor	Knox
256982140	octopus	abcsample123@gmail.com	123586012	31-07-2002	Ishita	Sindhwani
546789017	Emmassey_12	Emmanuel2030@google.co.uk	676796781	05-10-1998	Emmanuel	Massey
699621213	GoodCiara_45	ciara.good@aol.org	645170617	30-10-1996	Ciara	Good
909911345	MattyCam_34	matty.cam@hotmail.com	511582733	06-11-2000	Matthew	Cameron
988100984	TimRado_99	timothy567@icloud.net	227121618	16-05-1997	Timothy	Alvarado

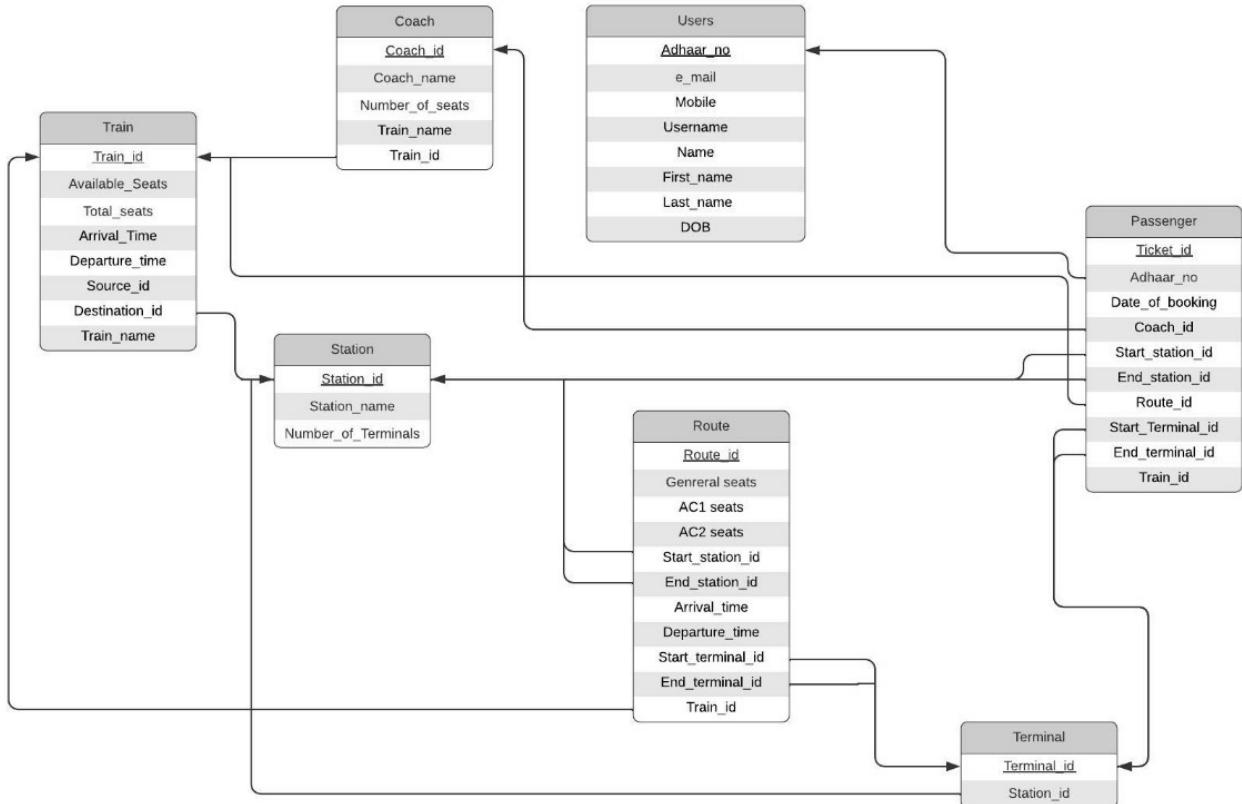


ER Diagram:





Relational Schema:





FRONTEND:

We started by creating the UI/UX of our web application using Figma.

This helped us visualize and create a basic template for all the screens.

We then proceeded to use html and css to edit and style our plan and form a website from it.

We used buttons and hyperlink to traverse to various webpages and link them with each other.

We created our first page the website user to decide if they want to login as a passenger or they want to login as an admin. They also have the option to view the available trains by inputting the departure and arrival destination and searching for them. There is also a contact us and about us page where we have given the hyperlinks to the respective pages.

Login/Register as a User

When the user logs in as a user they are required to give the information about adhaar, date of birth, et cetera. All this information goes to our database. They then can check available trains and add their preference of time and station. They then can see a page where they can confirm their booking details and their ticket is booked. This makes a user a passenger in our database. User can also see their information they added while registering and edit it if they want.

Login as admin

If we login as an admin we can make various changes and add/delete components.

They can add information about stations and trains. For trains they need to specify the number of seats and coaches of general, AC1 and Ac2 type.

For adding route, they need to add station and arrival and departure time of each of them and terminals at which train will stop.

BACKEND:

The app.py file handles all the backend part. In our backend we have linked everything with our database and created links between that and our webpage using flask. For example for the user login page, after making the suitable html and css files, we defined a function in python named user_info . this function takes the adhaar no and password input . If the the adhaar input by user matches any of the adhaar numbers stored in our database(we check this using an embedded sql query),we render it another screen that displays other details of the user.Then , that screen enables user to

book a ticket and view the tickets booked. We have exhaustively made use of buttons and hoverings that improve the UI and also help in transition between screens.



Indexing: We have done indexing in the following cases

A screenshot of a terminal window titled "indexing.sql". It contains the following SQL code:

```
use railway_system;

create index user_speed on users(Adhaar_no, password);
create index train_speed on train(Train_id, Train_name);
create index route_speed on route(Route_id, Train_id, Start_station_id, End_station_id);
create index passenger_speed on passenger(Ticket_id, Adhaar_no, Train_id);
```

Embedded SQL queries:

We have used a lot of embedded sql queries to create the functionality and flow of our application. All the buttons and information that is fed, the relevant sql queries run on our database and display information about what is asked. The queries to add data are also present as user can book ticket and add their information and the admin can add and edit information about routes. All the other information about available trains and ticket is filled through these as well

A screenshot of the Trainetic website. The header features the Trainetic logo, navigation links for "Admin Login", "Contact Us", and "About Us", and two buttons: "REGISTER" and "LOGIN". The main content area has a large image of a steam locomotive pulling a train through a snowy landscape. Overlaid on the image is the text "TRAINETIC" in large letters, followed by "SAFETY | SECURITY | PUNCTUALITY". To the left of the image is a search form with fields for "From Station", "To Station", and a "Search" button.



[Admin Login](#) [Contact us](#) [About Us](#)



USER LOG IN

LOGIN

BACK

5656

878478

Keep me logged in

LOGIN

OR LOGIN USING



[About Us](#)[Contact us](#)

USER INFORMATION

NAME: Gurmehak chadha

Date of Birth: 3000-05-04

ADHAAR NUMBER: 5656

Username: gmehak04

Phone number: 9811

E-mail: amrit2013@icloud.com

PRESS

TICKET INFORMATION

[Browse](#)

BROWSE TRAINS

[Logout](#)



[Contact us](#) [About Us](#)

BROWSE TRAINS

FROM STATION

surat

TO STATION

mumbai

BACK

CHECK



[Admin Login](#) [Contact us](#) [About Us](#)



ABOUT US

[BACK](#)

Travel is fun and when it comes to train travel, we go into the bypaths and un-trodden depths of wilderness and travel explorations to tell the world the glories of our journey!

TRAINETIC aims to serve its passengers with inclusive and comprehensive information that helps those making informed decisions, thus simplifying their train travels.



[Admin Login](#) [Contact us](#) [About Us](#)

CONTACT US

[BACK](#)



darsh20560@iiitd.ac.in



krishnam20309@iiitd.ac.in



gurmehak20298@iiitd.ac.in ishita20305@iiitd.ac.in



[User Login](#)[Contact us](#)[About Us](#)

ADMIN LOG IN

[BACK](#)

LOGIN

 __octopus__ monkey_man Keep me logged in[LOGIN](#)

OR LOGIN USING

[Contact Us](#)[About Us](#)

WELCOME ADMIN

[Log out](#)[ADD TRAIN](#)[ADD STATION](#)



Contact us

About Us



BACK

STATION NAME

bengluru

NUMBER OF
TERMINALS

2

OK



STATION ADDED
SUCCESSFULLY!

ADMIN_HOME



Contributions:

All members were present in the meetings and worked together on all the aspects of the project. Work was equally distributed and each member did their part with dedication.