

RAILWAY RESERVATION WEBSITE

Aditya Ramachandran, 2019103502

ShruthiG, 2019103061

Varsha SSKM, 2019103603

Batch –P

CREATION OF RELATIONS: (attached Create_tables.sql file)

```
Create_tables.sql
1  CREATE DATABASE railwaydb;
2  USE railwaydb;
3
4  CREATE TABLE TRAIN
5  (TRAIN_NO VARCHAR(15) PRIMARY KEY,
6   EXPRESS_NAME VARCHAR(20),
7   FC_SEATS INT,
8   SC_SEATS INT,
9   FC_PRICE INT,
10  SC_PRICE INT);
11
12  CREATE TABLE EXPRESS
13  (EXPRESS_NO INT NOT NULL AUTO_INCREMENT PRIMARY KEY ,
14   TRAIN_NO VARCHAR(15),
15   FROMM VARCHAR(20),
16   TOO VARCHAR(20),
17   STARTING_TIME TIMESTAMP,
18   ARRIVAL_TIME TIMESTAMP,
19   FC_SEATS_REMAINING INT,
20   SC_SEATS_REMAINING INT,
21   FOREIGN KEY (TRAIN_NO) REFERENCES TRAIN(TRAIN_NO)
22  );
23
24  ALTER TABLE EXPRESS AUTO_INCREMENT = 10001;
```

```
26  CREATE TABLE USER_DETAILS
27  (USER_ID VARCHAR(15) PRIMARY KEY,
28   USERNAME VARCHAR(20),
29   EMAIL_ID VARCHAR(25),
30   PASSWORD VARCHAR(25)
31  );
32
33  CREATE TABLE BOOKING
34  (BOOKING_ID INT NOT NULL AUTO_INCREMENT PRIMARY KEY ,
35   EXPRESS_NO INT,
36   USER_ID VARCHAR(20),
37   TRAIN_NO VARCHAR(15),
38   TOTAL_FC_SEATS INT,
39   TOTAL_SC_SEATS INT,
40   TOTAL_COST INT,
41   FOREIGN KEY (EXPRESS_NO) REFERENCES EXPRESS(EXPRESS_NO),
42   FOREIGN KEY (USER_ID) REFERENCES USER_DETAILS(USER_ID),
43   FOREIGN KEY (TRAIN_NO) REFERENCES TRAIN(TRAIN_NO)
44  );
```

RUNNING Create_tables.sql FILE ON MYSQL TERMINAL:

```
mysql> source C:\Users\DeLL\Desktop\DBMS PROJECT\Create_tables.sql
Query OK, 1 row affected (0.01 sec)

Database changed
Query OK, 0 rows affected (0.02 sec)

Query OK, 0 rows affected (0.03 sec)

Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0

Query OK, 0 rows affected (0.02 sec)

Query OK, 0 rows affected (0.03 sec)

mysql> SELECT * FROM EXPRESS;
Empty set (0.00 sec)

mysql> SHOW TABLES;
+-----+
| Tables_in_railwaydb |
+-----+
| booking              |
| express              |
| train                |
| user_details         |
+-----+
4 rows in set (0.01 sec)
```

Our project has four relations BOOKING, EXPRESS, TRAIN, USER_DETAILS. The relations TRAIN and EXPRESS are static tables which will be available to user as they request for a particular route from the webpage form. The USER_DETAILS relation gets populated as the user enters details in the SignIn.html webpage. The BOOKING relation gets populated as and when the users book for trains based on availability of trains.

INSERTING TUPLES IN TRAIN RELATION: (attached insert_train.sql file)

```
insert_train.sql
1  INSERT INTO TRAIN VALUES('S12345','Vaigai',10,25,400,250);
2  INSERT INTO TRAIN VALUES('N21023','Rajdhani',11,20,500,200);
3  INSERT INTO TRAIN VALUES('W39014','Konkan',15,30,300,200);
4  INSERT INTO TRAIN VALUES('E45698','Howrah',10,20,500,300);
5  INSERT INTO TRAIN VALUES('N56913','Ganga-Kaveri',15,30,400,250);
6  INSERT INTO TRAIN VALUES('S67813','Mysuru',20,35,400,250);
7
8  SELECT CONCAT (EXPRESS_NAME,' Express') FROM TRAIN;
9
```

RUNNING insert_train.sql FILE ON MYSQL TERMINAL:

```
mysql> source C:\Users\DeLL\Desktop\DBMS PROJECT\insert_train.sql
Query OK, 1 row affected (0.01 sec)
```

```
Query OK, 1 row affected (0.00 sec)
```

```
Query OK, 1 row affected (0.00 sec)
```

```
Query OK, 1 row affected (0.00 sec)
```

```
Query OK, 1 row affected (0.00 sec)
```

```
Query OK, 1 row affected (0.00 sec)
```

```
+-----+
| CONCAT (EXPRESS_NAME,' Express') |
+-----+
| Howrah Express                    |
| Rajdhani Express                  |
| Ganga-Kaveri Express              |
| Vaigai Express                    |
| Mysuru Express                    |
| Konkan Express                    |
+-----+
```

```
6 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM TRAIN;
```

```
+-----+-----+-----+-----+-----+-----+
| TRAIN_NO | EXPRESS_NAME | FC_SEATS | SC_SEATS | FC_PRICE | SC_PRICE |
+-----+-----+-----+-----+-----+-----+
| E45698   | Howrah       | 10       | 20       | 500      | 300      |
| N21023   | Rajdhani     | 11       | 20       | 500      | 200      |
| N56913   | Ganga-Kaveri | 15       | 30       | 400      | 250      |
| S12345   | Vaigai       | 10       | 25       | 400      | 250      |
| S67813   | Mysuru       | 20       | 35       | 400      | 250      |
| W39014   | Konkan       | 15       | 30       | 300      | 200      |
+-----+-----+-----+-----+-----+-----+
```

```
6 rows in set (0.00 sec)
```

INSERTING TUPLES IN EXPRESS RELATION: (attached insert_express.sql file)

```
insert_express.sql
1  INSERT INTO EXPRESS(TRAIN_NO,FROMM,TOO,STARTING_TIME,ARRIVAL_TIME,FC_SEATS_REMAINING,SC_SEATS_REMAINING)
2  SELECT
3  'S12345','Madurai','Chennai','2021-05-23 23:30:00','2021-05-24 05:29:00',
4  FC_SEATS , SC_SEATS FROM TRAIN WHERE TRAIN_NO = 'S12345';
5
6  INSERT INTO EXPRESS(TRAIN_NO,FROMM,TOO,STARTING_TIME,ARRIVAL_TIME,FC_SEATS_REMAINING,SC_SEATS_REMAINING)
7  SELECT
8  'N21023','Delhi','Bangalore','2021-06-05 18:30:00','2021-06-07 03:00:00',
9  FC_SEATS, SC_SEATS FROM TRAIN WHERE TRAIN_NO = 'N21023';
10
11 INSERT INTO EXPRESS(TRAIN_NO,FROMM,TOO,STARTING_TIME,ARRIVAL_TIME,FC_SEATS_REMAINING,SC_SEATS_REMAINING)
12 SELECT
13 'W39014','Nagpur','Bhopal','2021-06-10 15:30:00','2021-06-10 23:45:00',
14 FC_SEATS,SC_SEATS FROM TRAIN WHERE TRAIN_NO = 'W39014';
15
16 INSERT INTO EXPRESS(TRAIN_NO,FROMM,TOO,STARTING_TIME,ARRIVAL_TIME,FC_SEATS_REMAINING,SC_SEATS_REMAINING)
17 SELECT
18 'E45698','Kolkata','Hyderabad','2021-06-13 08:30:00','2021-06-14 05:29:00',
19 FC_SEATS,SC_SEATS FROM TRAIN WHERE TRAIN_NO = 'E45698';
20
21 INSERT INTO EXPRESS(TRAIN_NO,FROMM,TOO,STARTING_TIME,ARRIVAL_TIME,FC_SEATS_REMAINING,SC_SEATS_REMAINING)
22 SELECT
23 'N56913','Rameshwaram','Varanasi','2021-06-15 12:30:00','2021-06-18 03:30:00',
24 FC_SEATS,SC_SEATS FROM TRAIN WHERE TRAIN_NO = 'N56913';
25
26 INSERT INTO EXPRESS(TRAIN_NO,FROMM,TOO,STARTING_TIME,ARRIVAL_TIME,FC_SEATS_REMAINING,SC_SEATS_REMAINING)
27 SELECT
28 'S67813','Chennai','Mysore','2021-06-19 10:00:00','2021-06-19 18:10:00',
29 FC_SEATS,SC_SEATS FROM TRAIN WHERE TRAIN_NO = 'S67813';
```

RUNNING insert_express.sql FILE ON MYSQL TERMINAL:

```
mysql> source C:\Users\DeLL\Desktop\DBMS PROJECT\insert_express.sql
Query OK, 1 row affected (0.01 sec)
Records: 1 Duplicates: 0 Warnings: 0

Query OK, 1 row affected (0.01 sec)
Records: 1 Duplicates: 0 Warnings: 0

Query OK, 1 row affected (0.00 sec)
Records: 1 Duplicates: 0 Warnings: 0

Query OK, 1 row affected (0.00 sec)
Records: 1 Duplicates: 0 Warnings: 0

Query OK, 1 row affected (0.00 sec)
Records: 1 Duplicates: 0 Warnings: 0

Query OK, 1 row affected (0.00 sec)
Records: 1 Duplicates: 0 Warnings: 0
```

```
mysql> SELECT *FROM EXPRESS;
```

EXPRESS_NO	TRAIN_NO	FROMM	TOO	STARTING_TIME	ARRIVAL_TIME	FC_SEATS_REMAINING	SC_SEATS_REMAINING
10002	S12345	Madurai	Chennai	2021-05-23 23:30:00	2021-05-24 05:29:00	10	25
10003	N21023	Delhi	Bangalore	2021-06-05 18:30:00	2021-06-07 03:00:00	11	20
10004	W39014	Nagpur	Bhopal	2021-06-10 15:30:00	2021-06-10 23:45:00	15	30
10005	E45698	Kolkata	Hyderabad	2021-06-13 08:30:00	2021-06-14 05:29:00	10	20
10006	N56913	Rameshwaram	Varanasi	2021-06-15 12:30:00	2021-06-18 03:30:00	15	30
10007	S67813	Chennai	Mysore	2021-06-19 10:00:00	2021-06-19 18:10:00	20	35

```
6 rows in set (0.00 sec)
```

CONNECTION BETWEEN FRONTEND AND DATABASE (BACKEND FILES):

We have created a server.js file using Nodejs as the language which links the various webpages of our project. From the homepage we can go to the Sign in or Login page.

When a user signs in with credentials filled in the form, the credentials filled by the user gets added as a tuple into the USER_DETAILS relation. If the user enters a set of credentials that is not already existing in the USER_DETAILS relation, the user will be directed to a success page whereas if the user_id or username already exists in the USER_DETAILS relation, then the user will be directed to an error page and they can try to Sign In again.

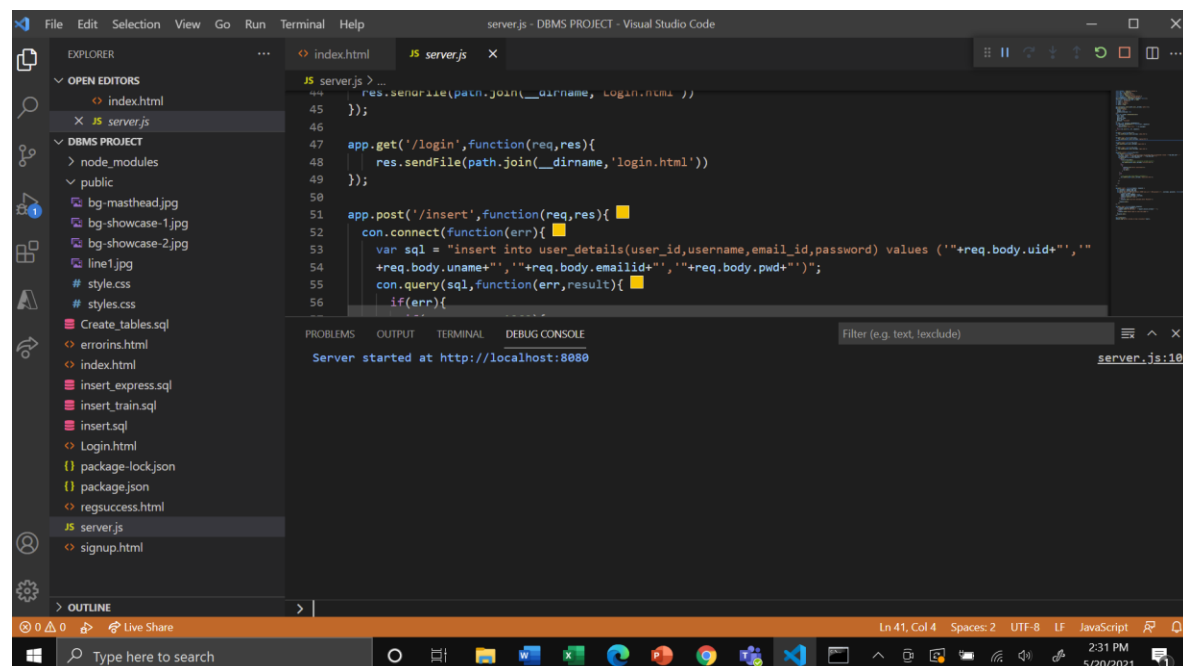
When a user enters the details to Login, the authentication script is written in server.js file which checks if the given user_id and password exist in the same combination in the USER_DETAILS relation. Upon finding a match, the user is directed to success message and in the case of not match found, the user is asked to re enter the correct credentials.

The working explained above is demonstrated below and the respective files are attached

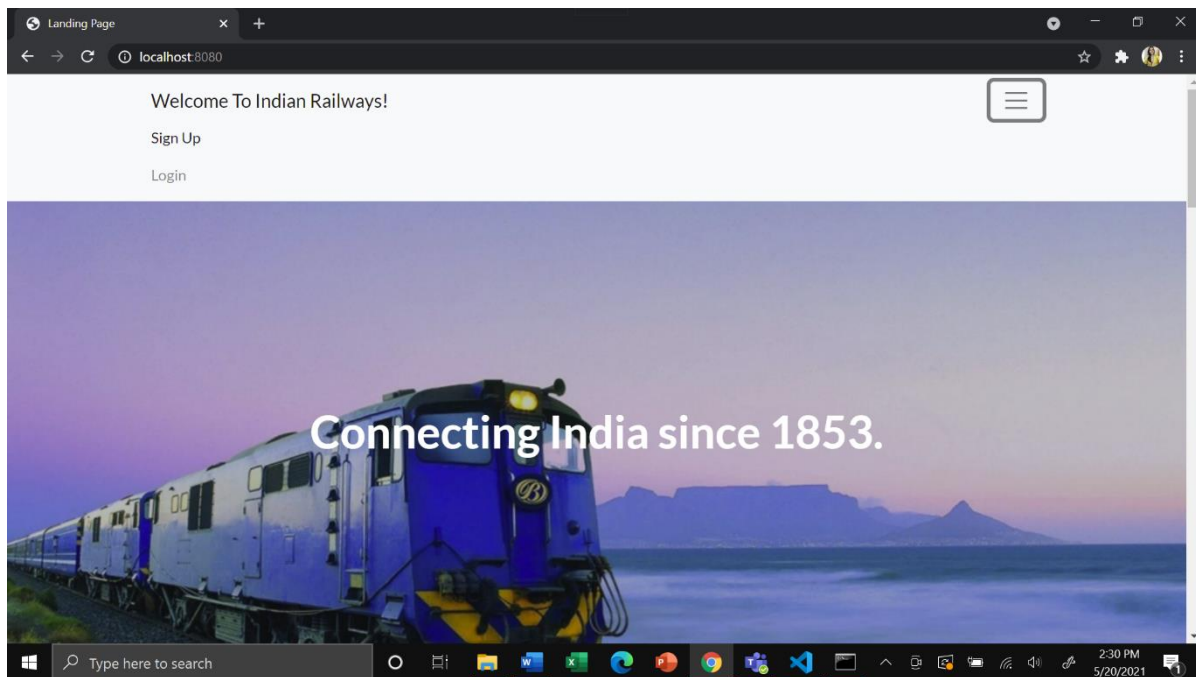
USER_DETAILS RELATION CURRENTLY:

```
mysql> SELECT *FROM USER_DETAILS;  
Empty set (0.01 sec)
```

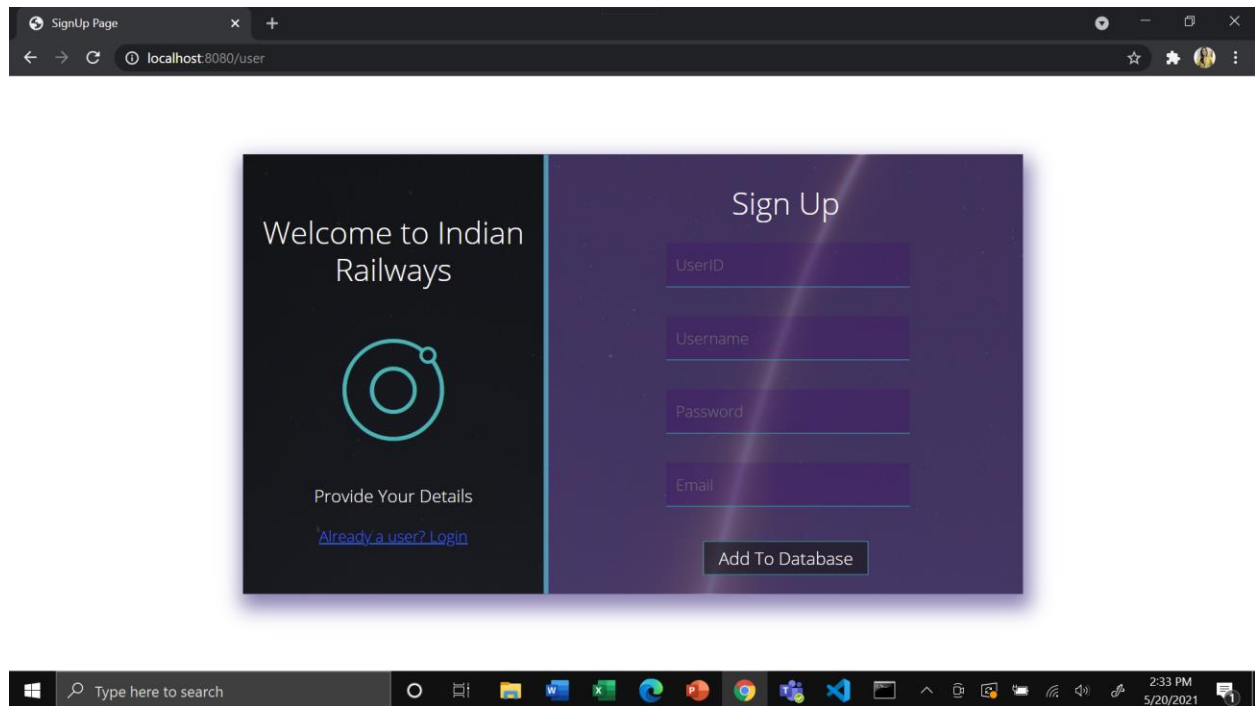
STARTING THE SERVER:



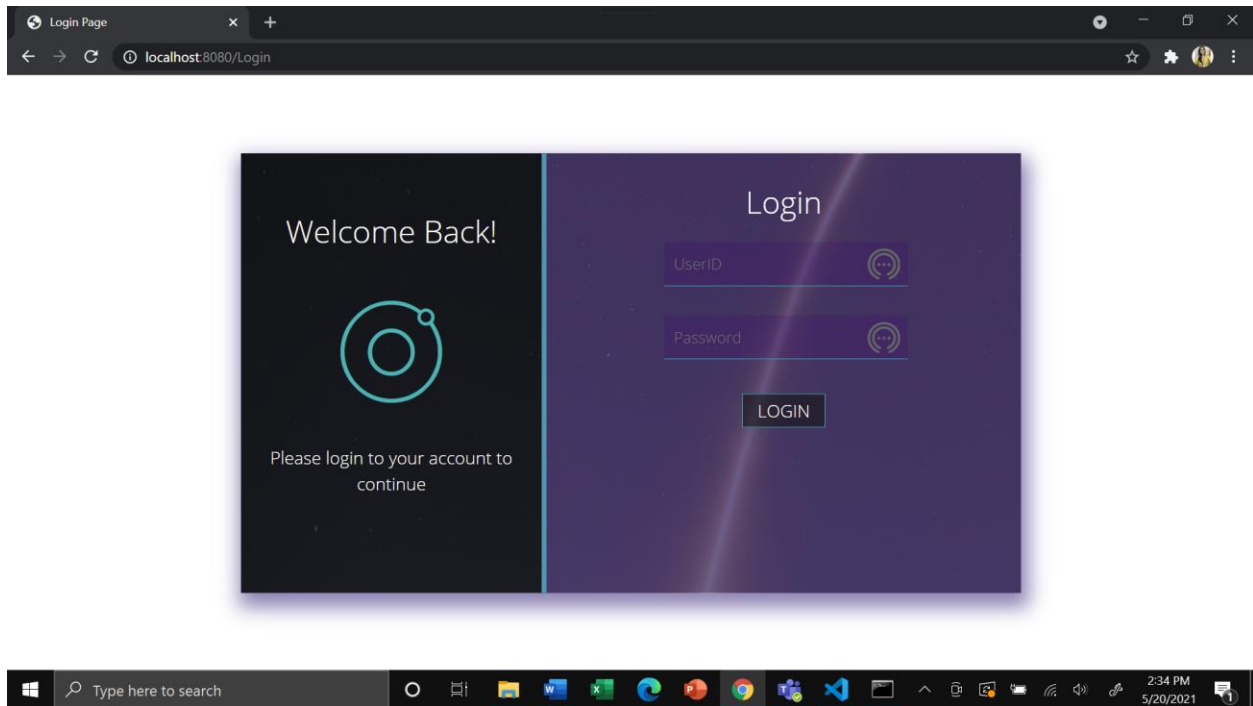
LANDING PAGE FRONTEND:



SIGN UP PAGE FRONT END:

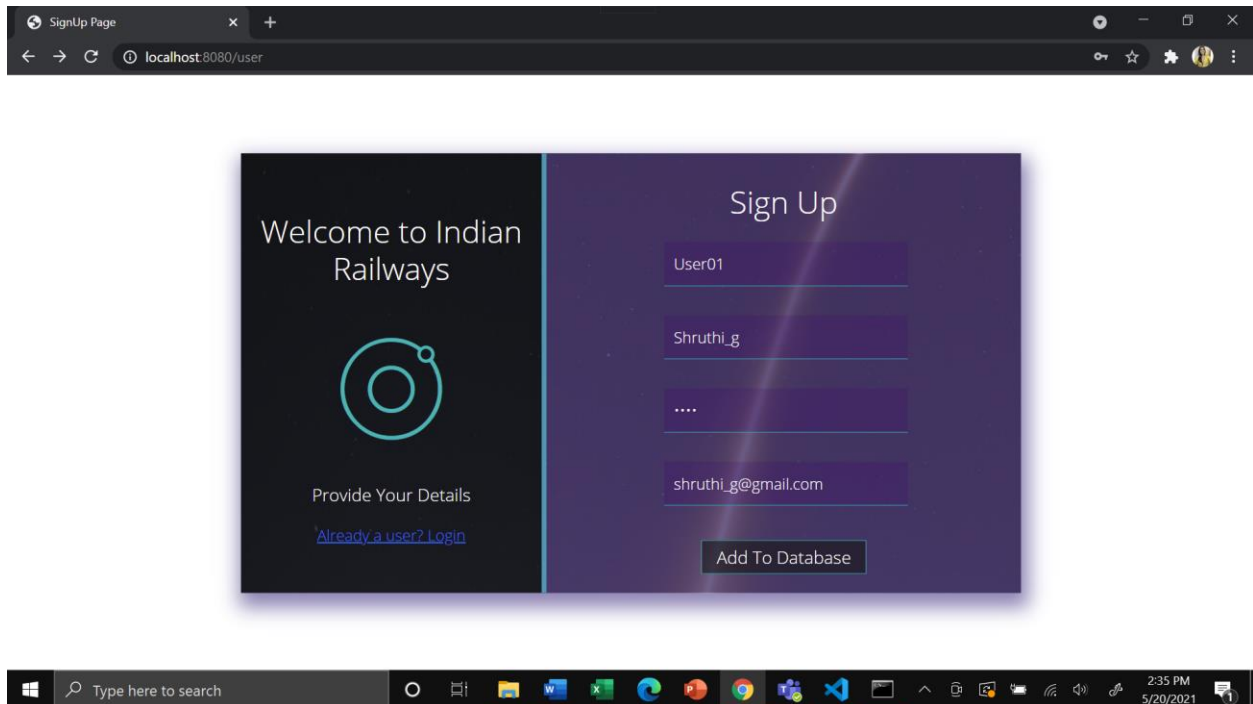


LOGIN PAGE FRONT END:



SHOWING THE CONNECTION BETWEEN FRONTEND AND DATABASE:

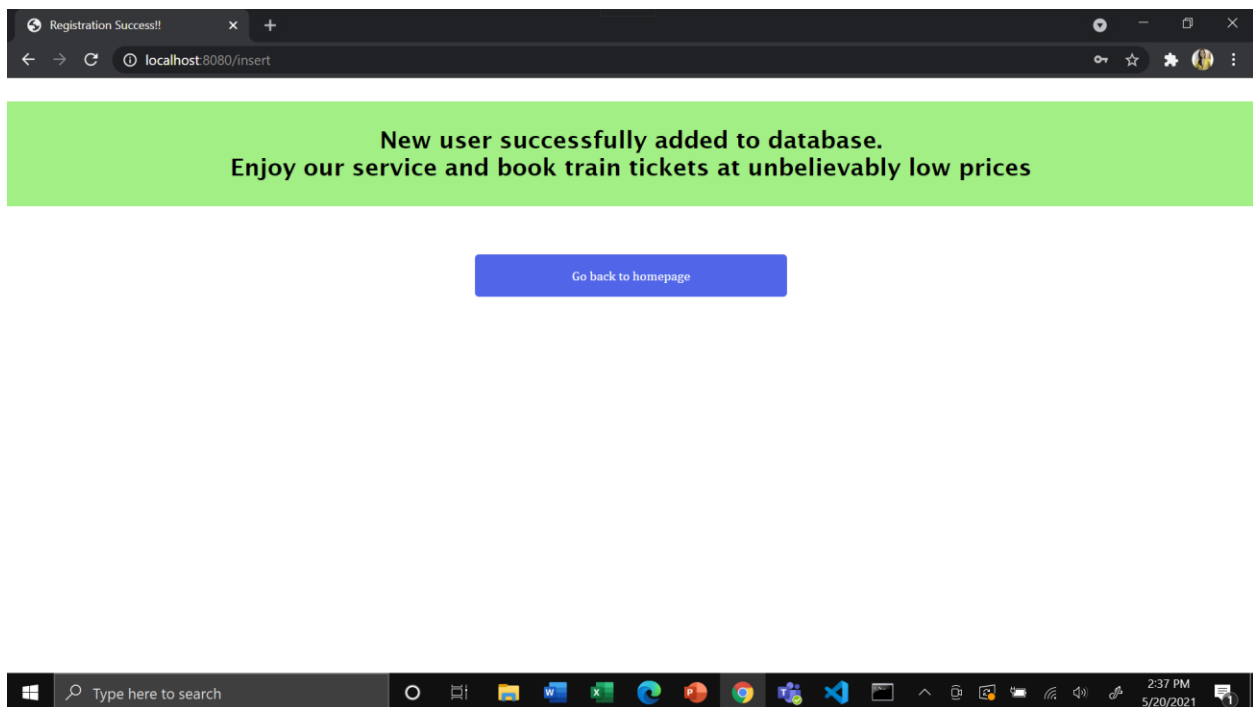
ENTERING DETAILS IN SIGN UP FORM:



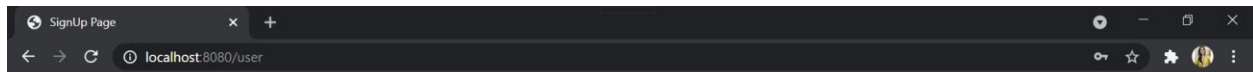
USER_DETAILS RELATIONS AFTER FILLING THE FORM:

```
mysql> SELECT *FROM USER_DETAILS;
+-----+-----+-----+-----+
| USER_ID | USERNAME | EMAIL_ID | PASSWORD |
+-----+-----+-----+-----+
| User01 | Shruthi_g | shruthi_g@gmail.com | pwd1 |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

UPON ADDING THE USER IS DIRECTED TO SUCCESS PAGE:



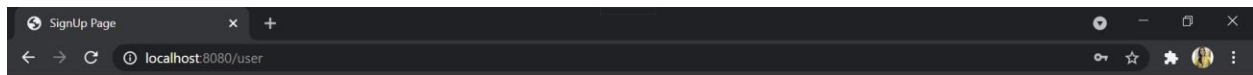
ADDING TWO MORE USERS:

A screenshot of a web application interface. On the left, a dark blue sidebar contains the text 'Welcome to Indian Railways', a circular logo, and the text 'Provide Your Details' with a link '[Already a user? Login](#)'. The main area has a purple background with the title 'Sign Up'. It contains four input fields: 'User02', 'Aditya_r', '....', and 'aditya_r@gmail.com'. Below these fields is a button labeled 'Add To Database'.


**New user successfully added to database.
Enjoy our service and book train tickets at unbelievably low prices**

[Go back to homepage](#)





Welcome to Indian Railways



Provide Your Details

[Already a user? Login](#)

Sign Up

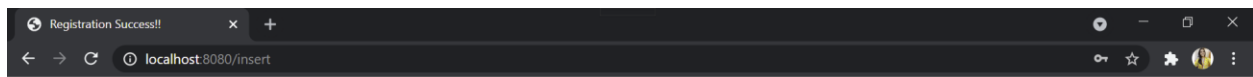
User03

Varsha_sskm

....

varsha_sskm@gmail.com

Add To Database



New user successfully added to database.
Enjoy our service and book train tickets at unbelievably low prices

Go back to homepage



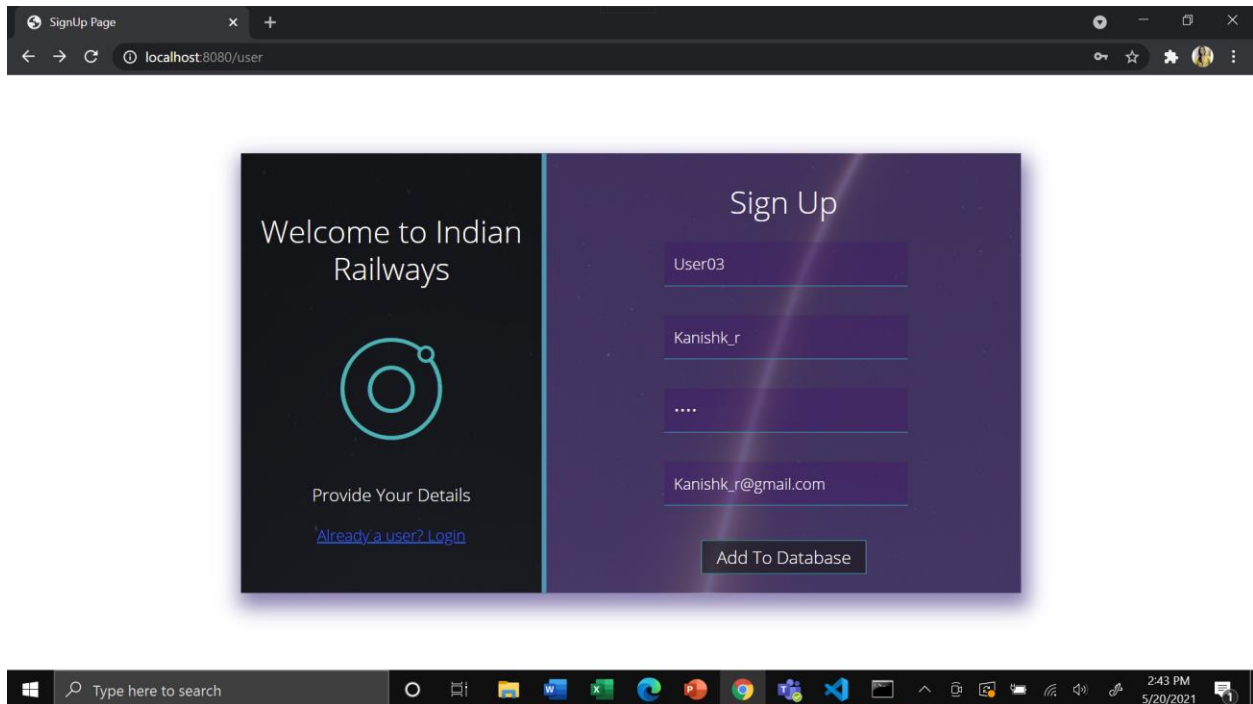
USER_DETAILS RELATION AFTER POPULATION OF DATA:

```
mysql> SELECT *FROM USER_DETAILS;
```

USER_ID	USERNAME	EMAIL_ID	PASSWORD
User01	Shruthi_g	shruthi_g@gmail.com	pwd1
User02	Aditya_r	aditya_r@gmail.com	pwd2
User03	Varsha_sskm	varsha_sskm@gmail.com	pwd3

```
3 rows in set (0.00 sec)
```

SHOWING THE ERROR WHEN WE TRY TO SIGN UP WITH ALREADY EXISTING CREDENTIALS:

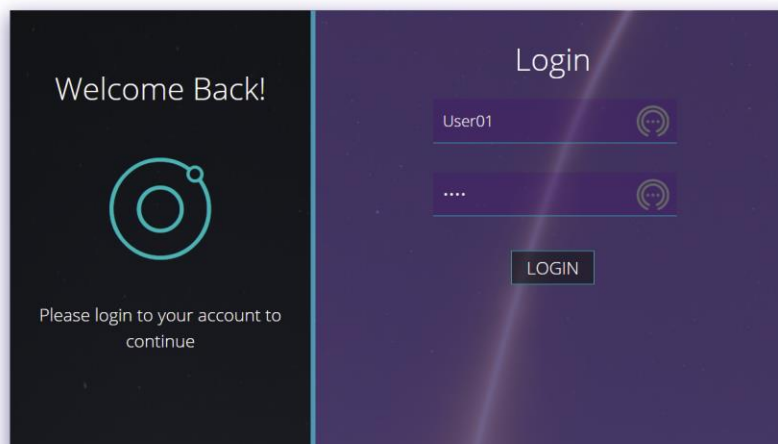
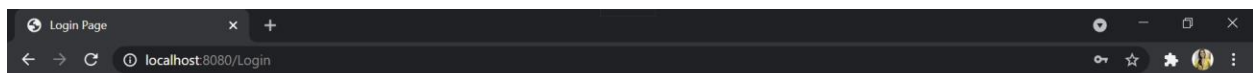




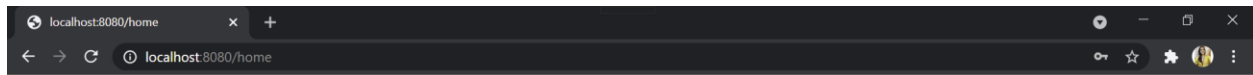
Given user ID exists.

Please re-enter your information

LOGIN AUTHENTICATION:



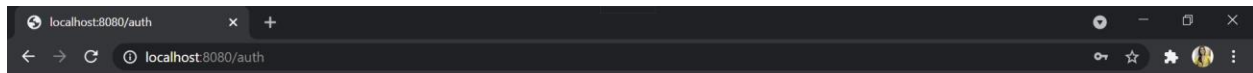
WHEN CORRECT CREDENTIALS ARE ENTERED:



Welcome back, User01!



WHEN INCORRECT CREDENTIALS ARE ENTERED:



Incorrect Username and/or Password!



SQL QUERIES USED TO INSERT TUPLES FROM SIGN UP FORM:

```
app.post('/insert',function(req,res){
  con.connect(function(err){
    var sql = "insert into user_details(user_id,username,email_id,password) values ('"+req.body.uid+"','"+
    +req.body.uname+"','"+req.body.emailid+"','"+req.body.pwd+"')";
    con.query(sql,function(err,result){
      if(err){
        if(err.errno==1062){
          res.sendFile(path.join(__dirname,'errorins.html'));
        }
        else
        {
          console.error("Error inserting!\n");
          throw err;
          res.end();
        }
      }
      else
      {
        res.sendFile(path.join(__dirname,'regsuccess.html'));
      }
    });
  });
});
});
```

SQL QUERIES USED FOR AUTHENTICATION:

```
app.post('/auth', function(request, response) {
  var username = request.body.uid;
  var password = request.body.pwd;
  con.query('SELECT * FROM user_details WHERE user_id = ? AND password = ?', [username, password],
  function(error, results, fields) {
    if (results.length > 0) {
      request.session.loggedin = true;
      request.session.username = username;
      response.redirect('/home');
    } else {
      response.send('Incorrect Username and/or Password!');
    }
    response.end();
  });
});

app.get('/home',function(request,response){
  if (request.session.loggedin) {
    response.send('Welcome back, ' + request.session.username + '!');
  } else {
    response.send('Please login to view this page!');
  }
  response.end();
});
```

CONSOLIDATED RELATION CREATED AND POPULATED USING SQL QUERIES:

```
mysql> SELECT *FROM USER_DETAILS;
```

USER_ID	USERNAME	EMAIL_ID	PASSWORD
User01	Shruthi_g	shruthi_g@gmail.com	pwd1
User02	Aditya_r	aditya_r@gmail.com	pwd2
User03	Varsha_sskm	varsha_sskm@gmail.com	pwd3

```
3 rows in set (0.00 sec)
```

```
mysql> SELECT *FROM EXPRESS;
```

EXPRESS_NO	TRAIN_NO	FROMM	TOO	STARTING_TIME	ARRIVAL_TIME	FC_SEATS_REMAINING	SC_SEATS_REMAINING
10002	S12345	Madurai	Chennai	2021-05-23 23:30:00	2021-05-24 05:29:00	10	25
10003	N21023	Delhi	Bangalore	2021-06-05 18:30:00	2021-06-07 03:00:00	11	20
10004	W39014	Nagpur	Bhopal	2021-06-10 15:30:00	2021-06-10 23:45:00	15	30
10005	E45698	Kolkata	Hyderabad	2021-06-13 08:30:00	2021-06-14 05:29:00	10	20
10006	N56913	Rameshwaram	Varanasi	2021-06-15 12:30:00	2021-06-18 03:30:00	15	30
10007	S67813	Chennai	Mysore	2021-06-19 10:00:00	2021-06-19 18:10:00	20	35

```
6 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM TRAIN;
```

TRAIN_NO	EXPRESS_NAME	FC_SEATS	SC_SEATS	FC_PRICE	SC_PRICE
E45698	Howrah	10	20	500	300
N21023	Rajdhani	11	20	500	200
N56913	Ganga-Kaveri	15	30	400	250
S12345	Vaigai	10	25	400	250
S67813	Mysuru	20	35	400	250
W39014	Konkan	15	30	300	200

```
6 rows in set (0.00 sec)
```