

Indian Institute of Information Technology, Allahabad

Database Management System

PROJECT REPORT

Tours & Travels Web Application

Project Team :

IIB2019007 : Aditya Raj

IIB20190021 : Hitika Kumar

IIT2019018 : B Chetan Rao

IIB2019001 : Harsh Mahajan

INDEX

1. Introduction
2. Technology used
 - 2.1 Frontend Development
 - 2.2 Backend Development
 - 2.3 Database
3. ER Diagram
4. Schema of each table required
5. Normalization of Tables and DataFrames
6. Functional components of Project
 - 5.1 Functional requirements of members
 - 5.2 Functional requirements of Administrator

1. Introduction

We have developed an on-line tour and travel web application for the people. Online tour and travels system Provides many services to the people. Our online tour and travels system has these five key features:

- (1) It provides the destination information and must visit tourist places to the people;
- (2) It gives the information about the tour packages;
- (3) It provides the accommodation facility to the user;
- (4) It provides the facility of online booking ;
- (5) We have designed the system such that it is easy to maintain and upgrade.

2. Technology used

2.1 Frontend Development

For the frontend development we have used Google Material Design. Material is a design system created by Google to help teams build high-quality digital experiences for Android, iOS, Flutter, and the web. Material Components are interactive building blocks for creating a user interface, and include a built-in states system to communicate focus, selection, activation, error, hover, press, drag, and disabled states. Component libraries are available for Android, iOS, Flutter, and the web. Components cover a range of interface needs, including:

- **Display:** Placing and organizing content using components like cards, lists, and sheets.
- **Navigation:** Allowing users to move through the product using components like navigation drawers and tabs.
- **Actions:** Allowing users to perform tasks using components such as the floating action button.
- **Input:** Allowing users to enter information or make selections using components like text fields, chips, and selection controls.
- **Communication:** Alerting users to key information and messages using components such as snackbars, banners, and dialogs.

2.2 Backend Development

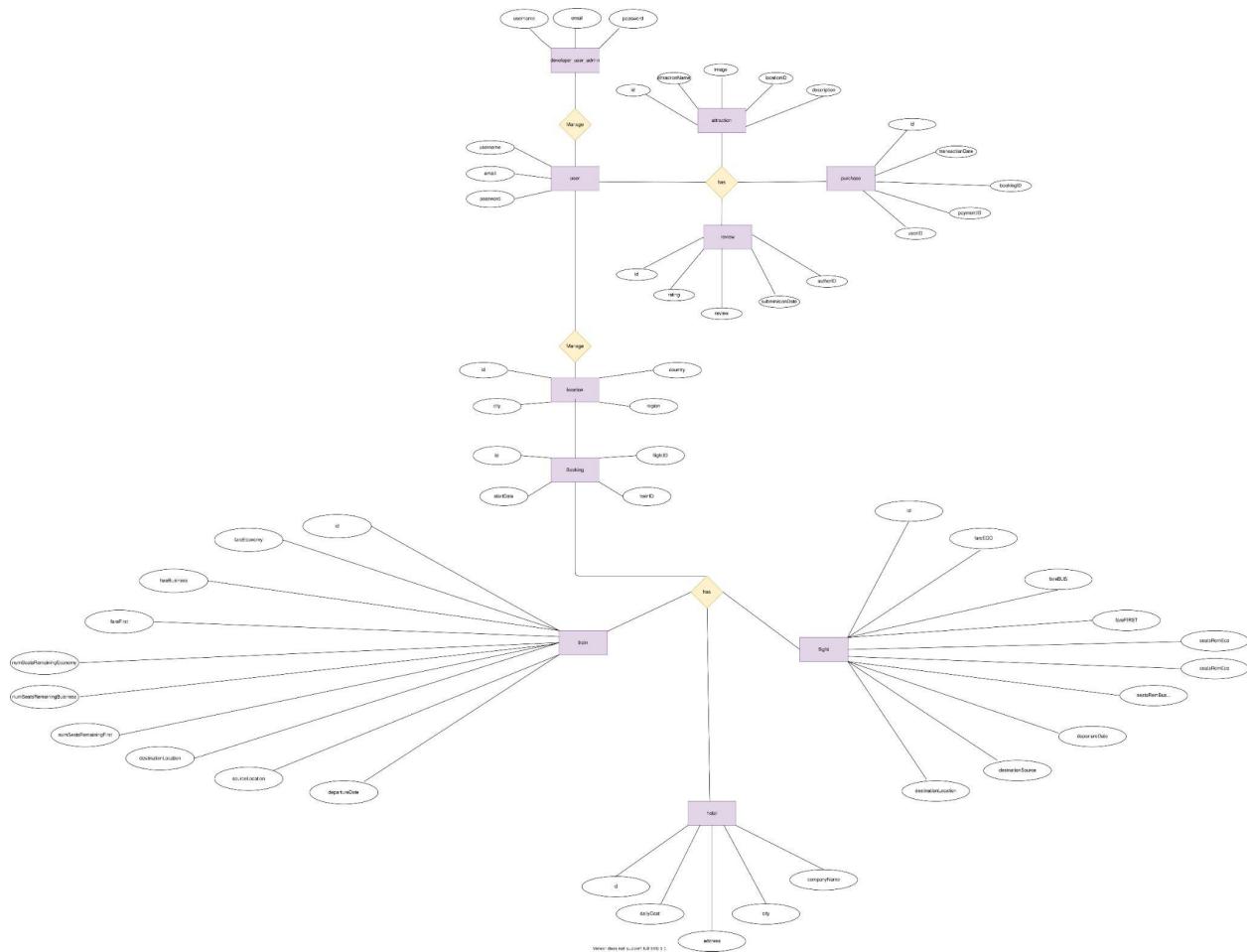
For backend development we have used Django Framework. Django is a high-level Python Web framework that encourages rapid development and

clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It's free and open source.

2.3 Database

For database management we have used PostgreSQL. PostgreSQL also known as postgres is a powerful, open source object-relational database system with over 30 years of active development that has earned it a strong reputation for reliability, feature robustness, and performance.

3. ER Diagram



ENTITIES	ATTRIBUTES
Attraction	<u>id</u> ,attractionName,attractionDescription,image,location_id
Booking	<u>id</u> ,startDate,Flight_id,Hotel_id,Train_id
Flight	<u>id</u> ,departureDate,sourceLocation,destinationLocation,fareEconomy,fareBusiness,fareFirst,numSeatsRemainingEconomy,numSeatsRemainingBusiness,numSeatsRemainingFirst
Hotel	<u>id</u> ,dailyCost,address,city,companyName
Location	<u>id</u> ,city,region,country,image
Payment	<u>id</u> ,amount,paymentType,cardNo
Purchase	<u>id</u> ,transactionDate,type,bookingID_id,paymentID_id,userID_id
Review	<u>id</u> ,rating,review,submissionDate,author_id
Train	<u>id</u> ,departureDate,sourceLocation,destinationLocation,fareEconomy,fareBusiness,fareFirst,numSeatsRemainingEconomy,numSeatsRemainingBusiness,numSeatsRemainingFirst
User	username, <u>email</u> ,password
Developer_user_admin	username, <u>email</u> ,password

4. Schema of each table required

1) User Table

```
class user(models.Model):  
  
    username = models.CharField(max_length=40)  
  
    email = models.CharField(max_length=35,  
unique=True, primary_key=True)  
  
    password = models.CharField(max_length=20)
```

2) Location Table

```
class location(models.Model):  
  
    id = models.IntegerField(auto_created=True,  
primary_key=True, unique=True)  
  
    city = models.CharField(max_length=30)  
  
    region = models.CharField(max_length=2)  
  
    country = models.CharField(max_length=2,  
default='US')  
  
    image = models.CharField(max_length=200)
```

3)Attraction Table

```
class attraction(models.Model):  
  
    id = models.IntegerField(auto_created=True,  
primary_key=True, unique=True)  
  
    location = models.ForeignKey(  
  
        location, on_delete=models.CASCADE)  
  
  
  
  
    attractionName =  
models.CharField(max_length=30)  
  
    attractionDescription =  
models.CharField(max_length=1000)  
  
    image = models.CharField(max_length=200)
```

4)Flight Table

```
class flight(models.Model):  
  
    id = models.IntegerField(auto_created=True,  
primary_key=True, unique=True)  
  
    departureDate = models.DateField()
```

```

    sourceLocation =
models.CharField(max_length=30)

    destinationLocation =
models.CharField(max_length=30)

    fareEconomy = models.DecimalField(max_digits=6,
decimal_places=2)

    fareBusiness =
models.DecimalField(max_digits=6, decimal_places=2)

    fareFirst = models.DecimalField(max_digits=6,
decimal_places=2)

    numSeatsRemainingEconomy =
models.IntegerField()

    numSeatsRemainingBusiness =
models.IntegerField()

    numSeatsRemainingFirst = models.IntegerField()

```

5)Hotel Table

```

class hotel(models.Model):

    id = models.IntegerField(auto_created=True,
primary_key=True, unique=True)

    dailyCost = models.DecimalField(max_digits=6,
decimal_places=2)

```

```
address = models.CharField(max_length=30)

city = models.CharField(max_length=30)

companyName = models.CharField(max_length=30,
default='hotel')
```

6)Train Table

```
class train(models.Model):

    id = models.IntegerField(auto_created=True,
primary_key=True, unique=True)

    departureDate = models.DateField()

    sourceLocation =
models.CharField(max_length=30)

    destinationLocation =
models.CharField(max_length=30)

    fareEconomy = models.DecimalField(max_digits=6,
decimal_places=2)

    fareBusiness =
models.DecimalField(max_digits=6, decimal_places=2)

    fareFirst = models.DecimalField(max_digits=6,
decimal_places=2)
```

```
    numSeatsRemainingEconomy =
models.IntegerField()

    numSeatsRemainingBusiness =
models.IntegerField()

    numSeatsRemainingFirst = models.IntegerField()
```

7)Review Table

```
class review(models.Model):
    id = models.IntegerField(auto_created=True,
primary_key=True, unique=True)
    rating = models.IntegerField()
    review = models.CharField(max_length=1000)
    author = models.ForeignKey(user,
on_delete=models.CASCADE)
    submissionDate =
models.DateField(auto_now=True)
```

8)Payment Table

```
class payment(models.Model):
    id = models.IntegerField(auto_created=True,
primary_key=True, unique=True)
    PAYMENT_TYPES = [('credit', 'Credit'),
('debit', 'Debit')]
```

```
amount = models.DecimalField(max_digits=6,
decimal_places=2)
paymentType =
models.CharField(choices=PAYMENT_TYPES,
max_length=6)
cardNo = models.CharField(max_length=16)
```

9)Booking Table

```
class booking(models.Model):
    id = models.IntegerField(auto_created=True,
primary_key=True, unique=True)
    startDate = models.DateTimeField(auto_now=True)
    # TRANSPORTATION_TYPES = [('flight'),
('train')]
    # classType = models.CharField(max_length=10)

    Flight = models.ForeignKey(
        flight, on_delete=models.CASCADE,
default=None, null=True)
    Train = models.ForeignKey(
        train, on_delete=models.CASCADE,
default=None, null=True)
    Hotel = models.ForeignKey(
```

```
    hotel, on_delete=models.CASCADE,  
default=None, null=True)
```

10)Purchase Model Table

```
class purchase(models.Model):  
  
    transactionDate =  
models.DateTimeField(auto_now=True)  
        userID = models.ForeignKey(user,  
on_delete=models.CASCADE)  
        Type = models.CharField(max_length=10,  
default="", null=True)  
        bookingID = models.ForeignKey(  
            booking, on_delete=models.CASCADE)  
        paymentID = models.ForeignKey(  
            payment, on_delete=models.CASCADE)
```

11)Admin Table

```
class user_admin(models.Model):  
    username = models.CharField(max_length=40, default "")  
    email = models.CharField(max_length=35, unique=True, primary_key=True)  
    password = models.CharField(max_length=20)
```

6. Normalization of Tables and DataFrames

Step 1:- Ensuring First Normal Form in Tables

1. First Normal form states that every table column in the database should have a unique value, there should be no repetition in values.
2. All the databases created for the subject of this project have a single attribute and will give only a single output with respect to primary key ensuring 1st normal form.

Step 2:- Ensuring Second Normal Form in Tables:-

1. Second Normal form states that every non primary key attribute should be fully functional dependent on primary key, i.e, it should not be dependent on part of a candidate key.
2. To ensure 2nd normal form in our database, no combination of keys are used as primary key, a separate id is provided as a primary key in most cases.
3. Since every table has mostly separate ids (like Booking id in booking Table, purchase id in Purchase table etc..) or single Email like primary Key in user database, so point of partial dependency Even comes here

Step 3:- Ensuring Third Normal Form

1. The table is already in second Normal form, so we have to ensure our database schema has no transitive dependency.
2. Since every table has a single key as a primary key, most likely to be ID or similar stuff, every other key (primary or non-primary) is directly dependent on that, so no transitive things are there.
3. For table inter-dependency also, only primary key on any table were used as foreign key in any other table.

For example:-

Booking table, purchase table and user table are there.

All have Booking_ID, purchase_id and Email as respective single primary key on which all other keys are directly dependent, so, there is no partial dependency neither the transitive dependency is present.

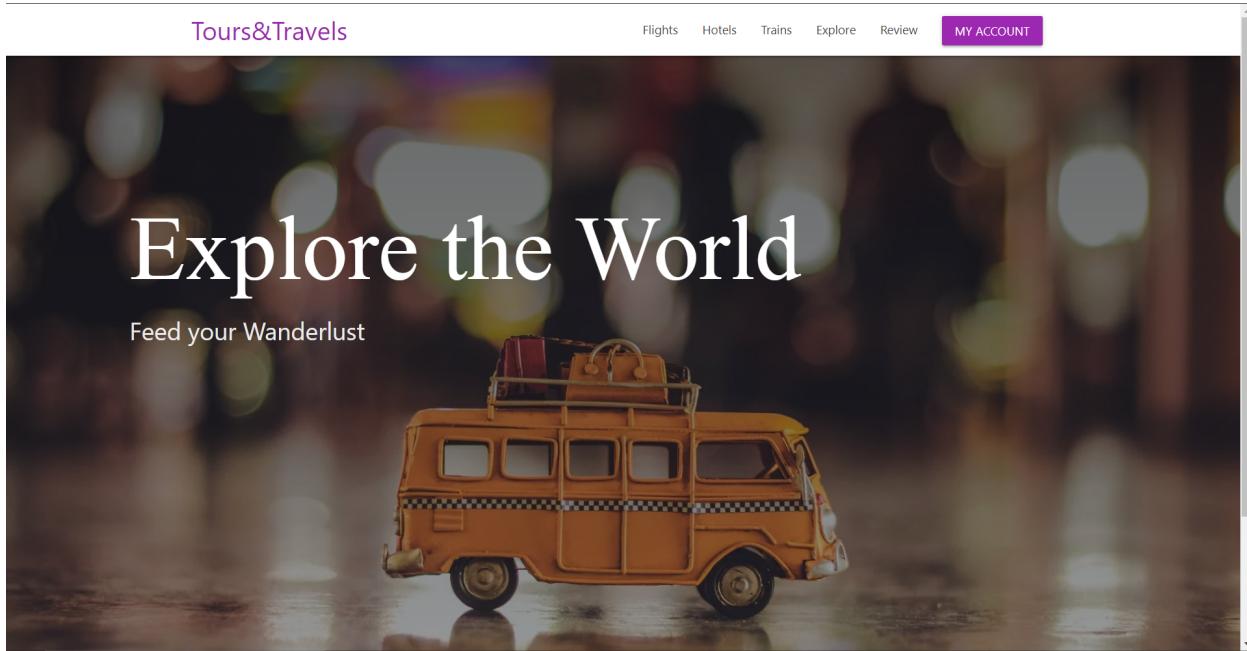
For inter-connectivity, bookind_id and purchase_id act as foreign key in the user database, so no issue as of here also.

In this way, We have a database that not only follows **3rd normal form**, but also follows **BCNF** due to all candidate and non-primary key dependent on a single primary key.

6. Functional components of the project/Implementation

6.1 Functional Requirements covered for Users

HOME PAGE: The home page of the website.



SIGN UP: SignUp feature allows users to login into the system successfully by filling in the credentials required i.e Name,Email,Password.

Sign Up

Name

Email

Password

[HAVE AN ACCOUNT? LOG IN](#)

Team TMinus0

We are a team of college students working hard to provide the best possible service to users.

Made by- Harsh Mahajan Aditya Raj B Chetan Rao Hitika Rajesh Kumar

LOGIN: Users can login into the website by filling the proper credentials i.e email and password.

Log In

Email

Password

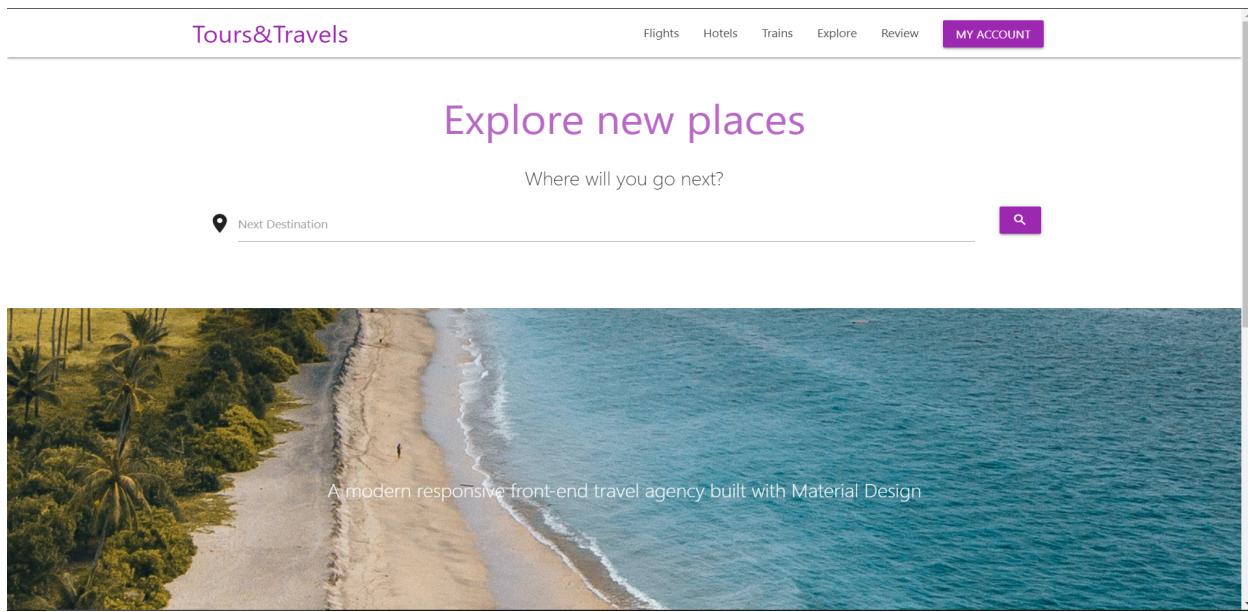
[CHANGE PASSWORD](#)[DON'T HAVE AN ACCOUNT? REGISTER NOW](#)

Team TMinus0

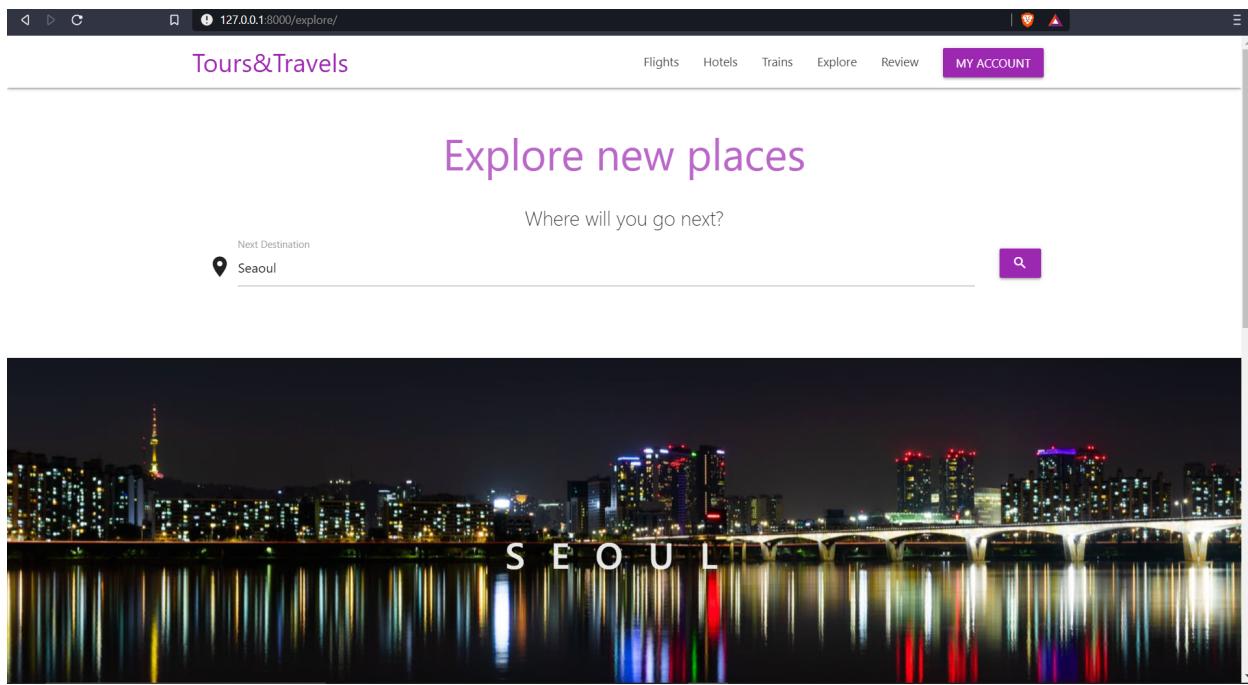
We are a team of college students working hard to provide the best possible service to users.

Made by- Harsh Mahajan Aditya Raj B Chetan Rao Hitika Rajesh Kumar

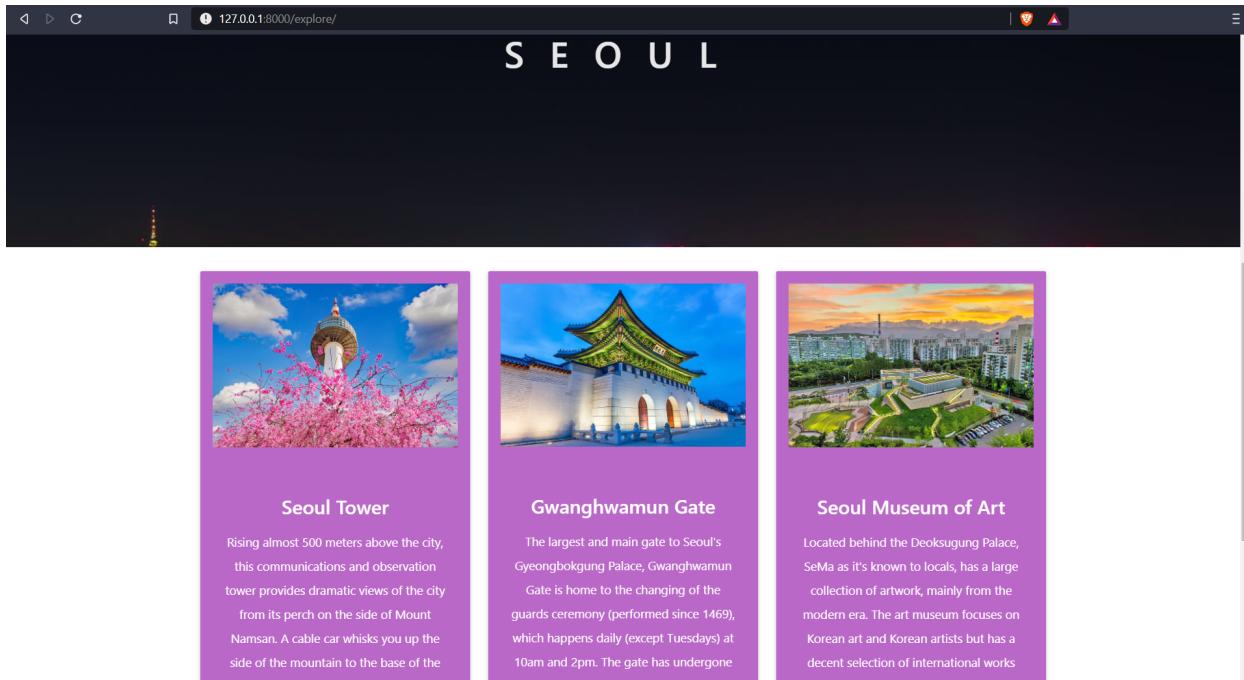
EXPLORE: Helps users to make best choices of destination for their travel .



SEARCHING THE DESTINATION: User can search his fav place of destination.



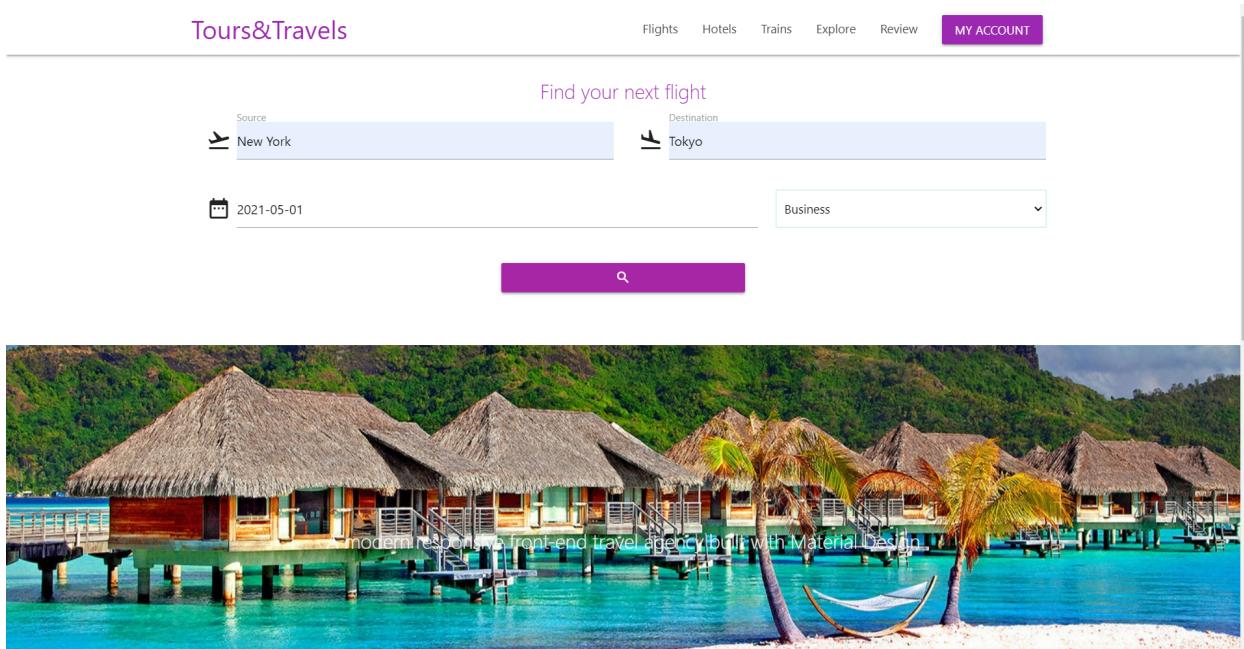
Users will also have the feature to have the knowledge of the famous places in that location.



The screenshot shows a travel exploration page for Seoul. At the top, there is a large banner with the word "SEOUL" in white capital letters. Below the banner, there are three cards, each featuring a purple header and a small image. The first card is titled "Seoul Tower" and shows an image of the tower with cherry blossoms in the foreground. The second card is titled "Gwanghwamun Gate" and shows an image of the gate at night. The third card is titled "Seoul Museum of Art" and shows an aerial view of the building. Each card has a brief description below the title.

Attraction	Description
Seoul Tower	Rising almost 500 meters above the city, this communications and observation tower provides dramatic views of the city from its perch on the side of Mount Namsan. A cable car whisks you up the side of the mountain to the base of the tower.
Gwanghwamun Gate	The largest and main gate to Seoul's Gyeongbokgung Palace, Gwanghwamun Gate is home to the changing of the guards ceremony (performed since 1469), which happens daily (except Tuesdays) at 10am and 2pm. The gate has undergone
Seoul Museum of Art	Located behind the Deoksugung Palace, SeMa as it's known to locals, has a large collection of artwork, mainly from the modern era. The art museum focuses on Korean art and Korean artists but has a decent selection of international works

SEARCHING THE FLIGHT/TRAIN: Users can search for the flight or train by entering the details required.



The screenshot shows a travel agency search interface. At the top, there is a navigation bar with links for "Tours&Travels", "Flights", "Hotels", "Trains", "Explore", "Review", and "MY ACCOUNT". Below the navigation bar, there is a search form with fields for "Source" (set to "New York") and "Destination" (set to "Tokyo"). There are also fields for "Date" (set to "2021-05-01") and "Class" (set to "Business"). A purple search button is located below the form. At the bottom of the page, there is a large image of a tropical resort with overwater bungalows and palm trees.

BOOK THE FLIGHT/TRAIN:Users can book for the flight or train and get the information like availability of seats .

The screenshot shows a travel booking website with a purple header bar. The main search area has fields for 'Source' (New York) and 'Destination' (Tokyo), a date selector ('2021-05-01'), and a dropdown for 'Business' class. A purple search button with a magnifying glass icon is centered below these fields. Below the search area, a summary box displays the search parameters: Source: New York, Destination: Tokyo, Departure Date: May 1, 2021, Departure Time: (empty), Class: business, and Fare: 300.00. A 'BOOK NOW' button is located at the bottom of this summary box. At the very bottom of the page, there is a purple footer bar with the text 'Team TMinus0' and a message: 'We are a team of college students working hard to provide the best possible service to users.'

MAKE PAYMENT FOR FLIGHT/TRAIN:Users can make payment for the flight or train by entering the details required and confirm their booking.

The screenshot shows a payment confirmation screen from the same travel booking website. It features a purple header bar and a purple footer bar. The main content area is titled 'Current Booking' and shows a summary of the trip: New York to Tokyo on May 1, 2021, in Business class for Rs 300.00. A 'CONFIRM' button is located at the bottom of this section. The purple footer bar at the bottom contains the text 'Team TMinus0' and the same message: 'We are a team of college students working hard to provide the best possible service to users.' Below this, it lists the names of the team members: Made by - Harsh Mahajan, Aditya Raj, B Chetan Rao, and Hitika Rajesh Kumar.

VIEW TICKET HISTORY: Users can view its ticket history.

Tours&Travels

Flights Hotels Trains Explore Review MY ACCOUNT

Current Booking

Card Number Card Type

CONFIRM

Booking successful

Type	Class	Source	Destination	Book Date	Total	Card Number
Flight	business	New York	Tokyo	May 11, 2021, 9:52 a.m.	Rs 300.00	123456789

Team TMinus0
We are a team of college students working hard to provide the best possible service to users.
Made by:- Harsh Mahajan Aditya Raj B Chetan Rao Hitika Rajesh Kumar

SEARCHING THE HOTEL: Users can search for the flight or train by entering the details required.

Tours&Travels

Flights Hotels Trains Explore Review MY ACCOUNT

Find your travel accommodations

Location: Ladakh

2021-05-18 2021-05-31

Search



BOOK THE HOTEL: Users can search for the hotel by entering the details required.

Find your travel accommodations

Location

Ladakh

2021-05-18

2021-05-31



Tminus0

LadakhCity

2000.00

[BOOK NOW](#)

Team TMinus0

We are a team of college students working hard to provide the best possible service to users.

Made by-

Harsh Mahajan

Aditya Raj

B Chetan Rao

Hitika Rajesh Kumar

MAKE PAYMENT: Users can make payment for the hotel by entering the details required and confirm their booking.

Current Booking

Card Number

1234567890

Debit

LadakhCity Ladakh

Rs 2000.00

[CONFIRM](#)

Team TMinus0

We are a team of college students working hard to provide the best possible service to users.

Made by-

Harsh Mahajan

Aditya Raj

B Chetan Rao

Hitika Rajesh Kumar

VIEW BOOKED HISTORY: Users can view its ticket history.

Current Booking

<input type="text"/> Card Number	<input type="text"/> Card Type
----------------------------------	--------------------------------

CONFIRM

Booking successful

Type	Class	Source	Destination	Book Date	Total	Card Number
Hotel	NULL	NULL	Ladakh	May 11, 2021, 9:59 a.m.	Rs 2000.00	1234567890

Team TMinus0

We are a team of college students working hard to provide the best possible service to users.

Made by:- Harsh Mahajan Aditya Raj B Chetan Rao Hitika Rajesh Kumar

REVIEW:Users can give their reviews about the services provided by this website and rate accordingly :)

Review our service

<input type="text"/> Review	<input type="text"/> Great Website	<input type="text"/> 5
-----------------------------	------------------------------------	------------------------

SUBMIT

Harsh says...

Ratings: 5 | April 30, 2021

Greate website

hitika says...

Ratings: 5 | May 8, 2021

hitika

hitika says...

Ratings: 5 | May 8, 2021

VIEW TOTAL HISTORY OF ALL TIMES:Users can view their preferred email.

Preferences Payment Methods History	<p>Preferences</p> <hr/> <p>Email Login h@gmail.com</p>
---	---

Team TMinus0

We are a team of college students working hard to provide the best possible service to users.

Made by: Harsh Mahajan Aditya Raj B Chetan Rao Hitika Rajesh Kumar

VIEW TOTAL HISTORY OF ALL TIMES: Users can view their booking history of all time.

Preferences Payment Methods History	<p>History</p> <hr/> <table border="1"> <thead> <tr> <th>Type</th><th>Transportation</th><th>Source</th><th>Destination</th><th>Book Date</th><th>Total</th><th>Card Number</th></tr> </thead> <tbody> <tr> <td>Flight</td><td>business</td><td>New York</td><td>Tokyo</td><td>May 8, 2021, 8:10 a.m.</td><td>Rs 300.00</td><td>2134567890</td></tr> <tr> <td>Flight</td><td>business</td><td>Ladakh</td><td>Seoul</td><td>May 9, 2021, 3:40 p.m.</td><td>Rs 2000.00</td><td>234567890</td></tr> <tr> <td>Flight</td><td>business</td><td>New York</td><td>Tokyo</td><td>May 11, 2021, 9:52 a.m.</td><td>Rs 300.00</td><td>123456789</td></tr> <tr> <td>Hotel</td><td>NULL</td><td>NULL</td><td>Ladakh</td><td>May 11, 2021, 9:59 a.m.</td><td>Rs 2000.00</td><td>1234567890</td></tr> </tbody> </table>	Type	Transportation	Source	Destination	Book Date	Total	Card Number	Flight	business	New York	Tokyo	May 8, 2021, 8:10 a.m.	Rs 300.00	2134567890	Flight	business	Ladakh	Seoul	May 9, 2021, 3:40 p.m.	Rs 2000.00	234567890	Flight	business	New York	Tokyo	May 11, 2021, 9:52 a.m.	Rs 300.00	123456789	Hotel	NULL	NULL	Ladakh	May 11, 2021, 9:59 a.m.	Rs 2000.00	1234567890
Type	Transportation	Source	Destination	Book Date	Total	Card Number																														
Flight	business	New York	Tokyo	May 8, 2021, 8:10 a.m.	Rs 300.00	2134567890																														
Flight	business	Ladakh	Seoul	May 9, 2021, 3:40 p.m.	Rs 2000.00	234567890																														
Flight	business	New York	Tokyo	May 11, 2021, 9:52 a.m.	Rs 300.00	123456789																														
Hotel	NULL	NULL	Ladakh	May 11, 2021, 9:59 a.m.	Rs 2000.00	1234567890																														

Team TMinus0

We are a team of college students working hard to provide the best possible service to users.

Made by: Harsh Mahajan Aditya Raj B Chetan Rao Hitika Rajesh Kumar

VIEW TOTAL HISTORY OF ALL TIMES: Users can view their payment history of all time.

Preferences	Payment	
Payment Methods		
History		
	Credit	XXXX-XXXX-XXXX-1234
	Debit	XXXX-XXXX-XXXX-1234

Team TMinus0

We are a team of college students working hard to provide the best possible service to users.

Made by- Harsh Mahajan Aditya Raj B Chetan Rao Hitika Rajesh Kumar

CHANGE PASSWORD:Users can change their password.

Change Password

Email

Old Password

New Password

SUBMIT ➤

Team TMinus0

We are a team of college students working hard to provide the best possible service to users.

Made by- Harsh Mahajan Aditya Raj B Chetan Rao Hitika Rajesh Kumar

6.2 Functional Requirements covered for Admin

LOGIN: Admin can login into the system by putting in the correct details.

Tours&Travels

Flights Hotels Trains Explore Review **MY ACCOUNT**

Log In

Email

Password

SUBMIT ➤

[CHANGE PASSWORD](#)

[DON'T HAVE AN ACCOUNT? REGISTER NOW](#)

Team TMinus0

We are a team of college students working hard to provide the best possible service to users.

Made by- Harsh Mahajan Aditya Raj B Chetan Rao Hitika Rajesh Kumar



PRINT: Admin can view all the booking history of all users .

Tours&Travels							
User	Type	Class	Source	Destination	Book Date	Total	Card Number
harshmahajan8989@gmail.com	Hotel	NULL	NULL	Tokyo	April 30, 2021, 2:54 p.m.	Rs 1500.00	1234123
harshmahajan8989@gmail.com	Hotel	NULL	NULL	Tokyo	April 30, 2021, 4:01 p.m.	Rs 1500.00	1341234
harshmahajan8989@gmail.com	Train	business	New York	Tokyo	April 30, 2021, 4:13 p.m.	Rs 300.00	49520458043
harshmahajan8989@gmail.com	Hotel	NULL	NULL	Tokyo	April 30, 2021, 4:25 p.m.	Rs 1500.00	1351234123
harshmahajan8989@gmail.com	Hotel	NULL	NULL	Tokyo	April 30, 2021, 4:27 p.m.	Rs 1500.00	1341234123412
harshmahajan8989@gmail.com	Train	economy	New York	Tokyo	April 30, 2021, 7:09 p.m.	Rs 200.00	135352312341234
harshmahajan8989@gmail.com	Flight	business	New York	Tokyo	April 30, 2021, 7:11 p.m.	Rs 300.00	13412341234
harshmahajan8989@gmail.com	Hotel	NULL	NULL	Tokyo	April 30, 2021, 7:12 p.m.	Rs 1500.00	5234523452

Team TMinus0

We are a team of college students working hard to provide the best possible service to users.

Made by- Harsh Mahajan Aditya Raj B Chetan Rao Hitika Rajesh Kumar

UPDATE DETAILS: Admin can update the details of all the functionalities.