

# **Flight Booking Application USE MERN STACK**

**Done By**

**Praveen Kumar.G**

**Prasanth.J**

**Vishwa.E**

**Yuvaraj.P**



# Introduction to the MERN Stack



## MongoDB

Database for storing flight data, user information, and bookings.



## React

Frontend framework for creating a dynamic and responsive user interface.



## Flight Search & Booking

Core functionality allowing users to search for flights and make reservations.



## Express.js

Backend framework for handling API requests and server-side logic.



## Node.js

JavaScript runtime environment for executing the server-side code.



## User Accounts

User registration, login, and profile management capabilities.



# Overview of the Flight Booking Application

1

**User Registration and Authentication**

2

**Flight Search and Availability**

3

**Itinerary Management and Updates**

4

**Notifications and Reminders**

# User Registration and Authentication

## Secure Sign-up

Implement user registration with email verification and password hashing

## Seamless Login

Allow users to log in securely using their registered credentials

## Password Recovery

Provide a mechanism for users to reset their forgotten passwords



# Flight Search and Availability

## Flight Search

Allow users to search for flights based on origin, destination, and travel dates

## Availability Check

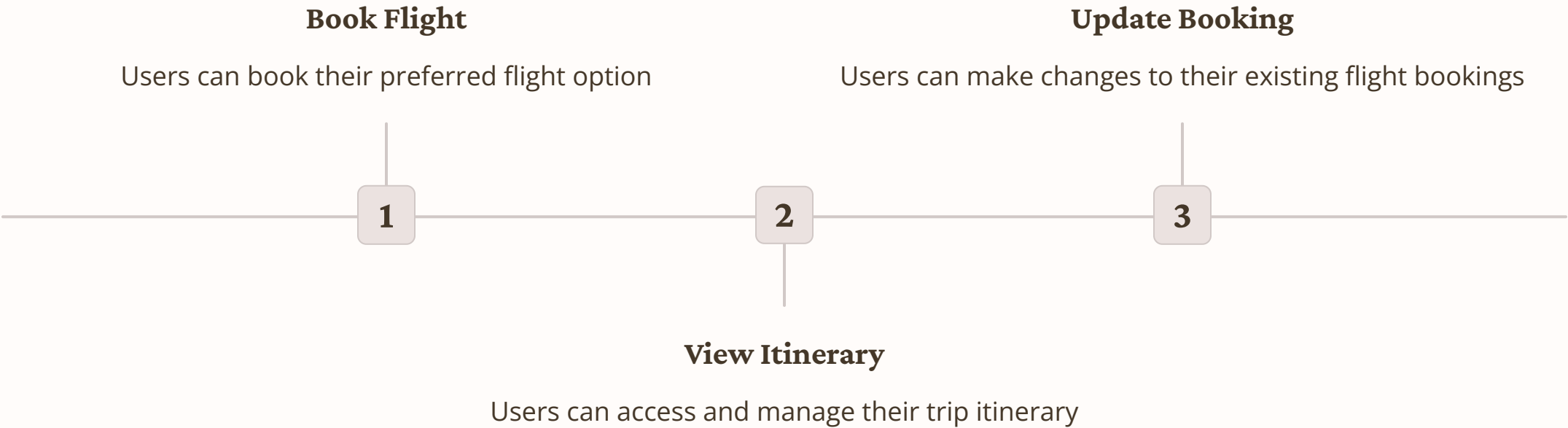
Display real-time flight availability and pricing information

## Filtering and Sorting

Provide options to filter and sort search results by various criteria



# Itinerary Management and Updates





# Notifications and Reminders



## Flight Confirmation

Users receive email confirmation for their bookings



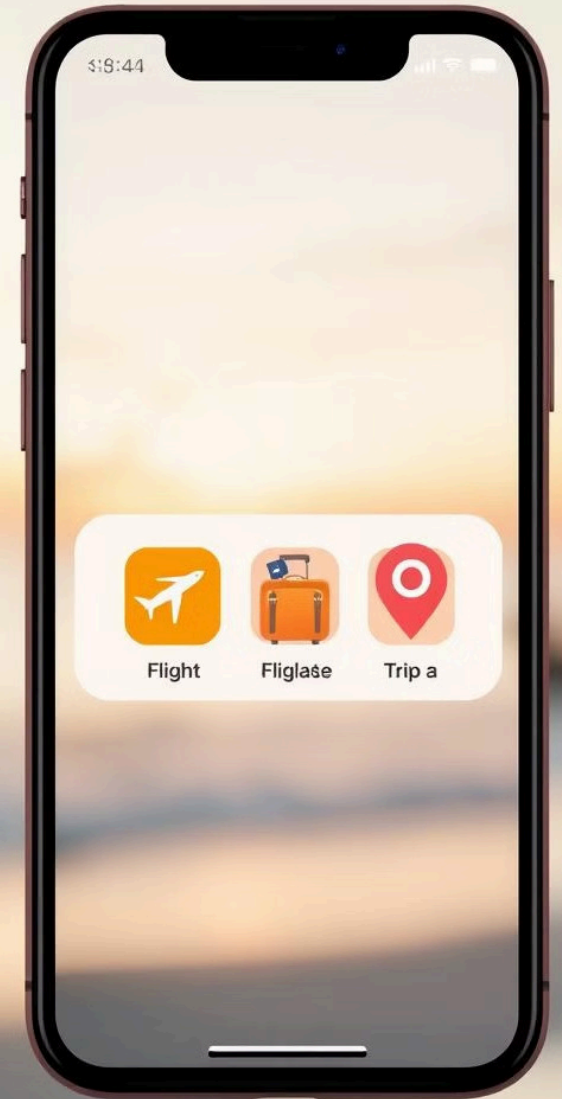
## Trip Reminders

Users get timely notifications about their upcoming trips

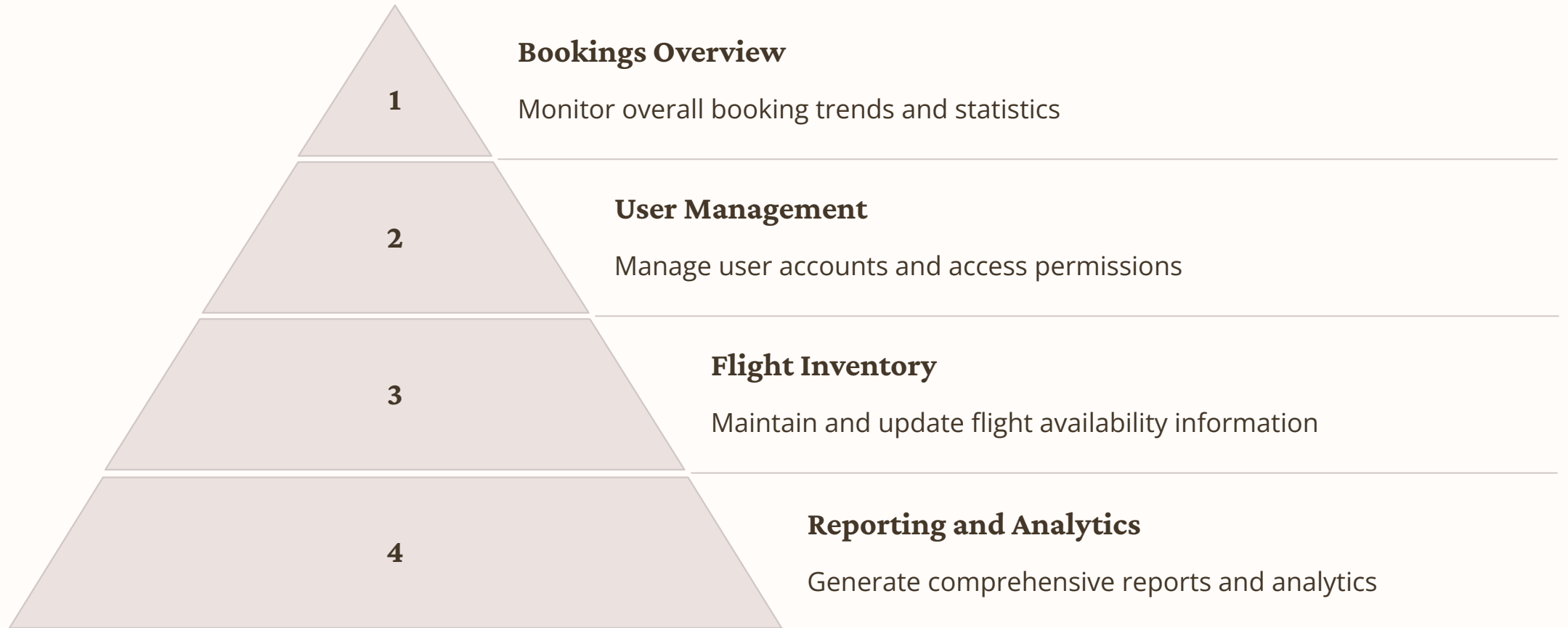


## Mobile Alerts

Users can opt-in for push notifications on their mobile devices



# Admin Dashboard and Analytics





# Execution Part

1

## Backend Development

Implement the server-side logic using Express and Node.js

2

## Frontend Development

Build the user interface with React and supporting technologies

3

## Database Integration

Integrate the MongoDB database to store and retrieve data

4

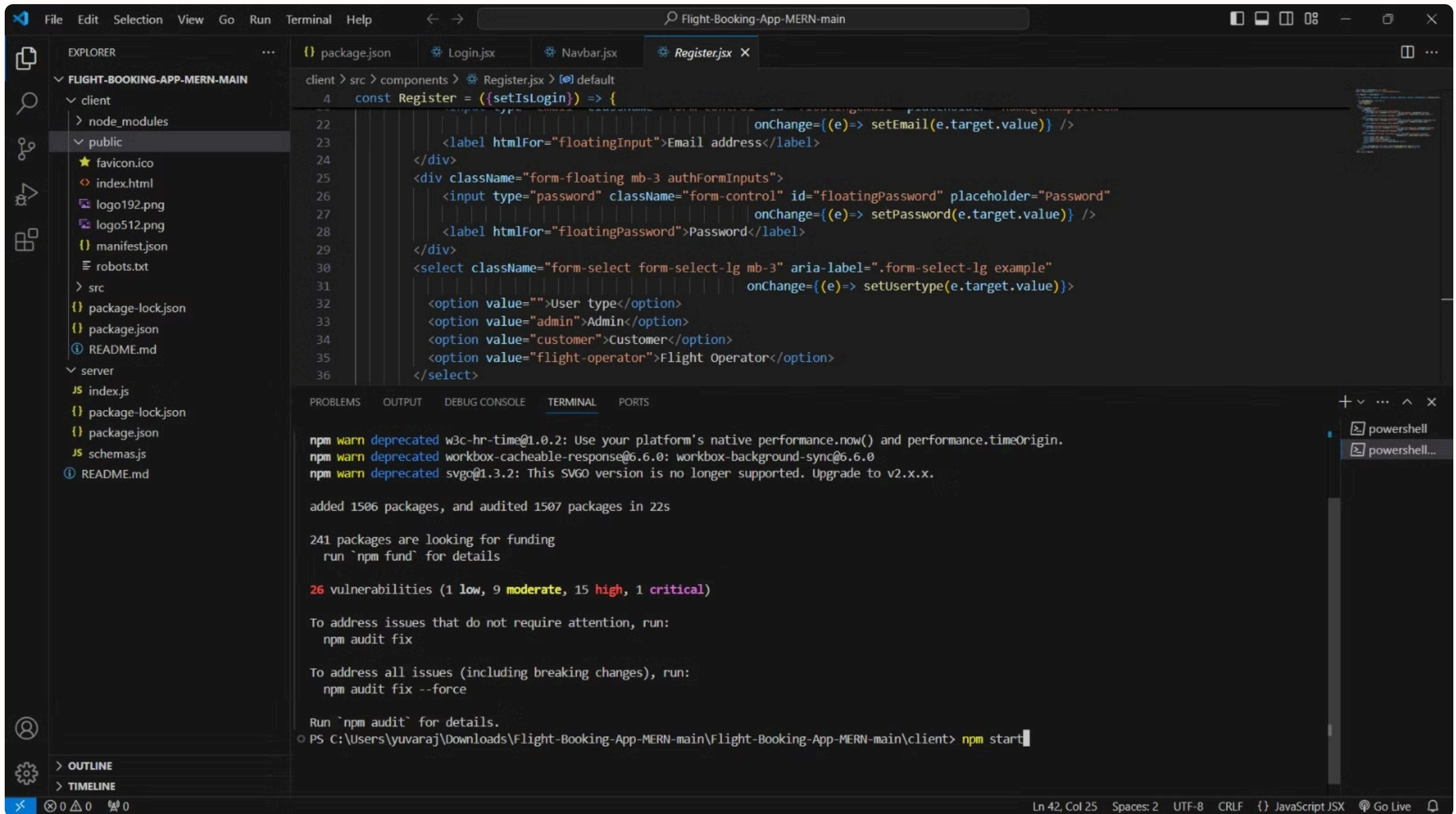
## Testing and Deployment

Conduct comprehensive testing and deploy the application



# Execution

## CLIENTSIDE



The screenshot shows a Visual Studio Code editor with the following components:

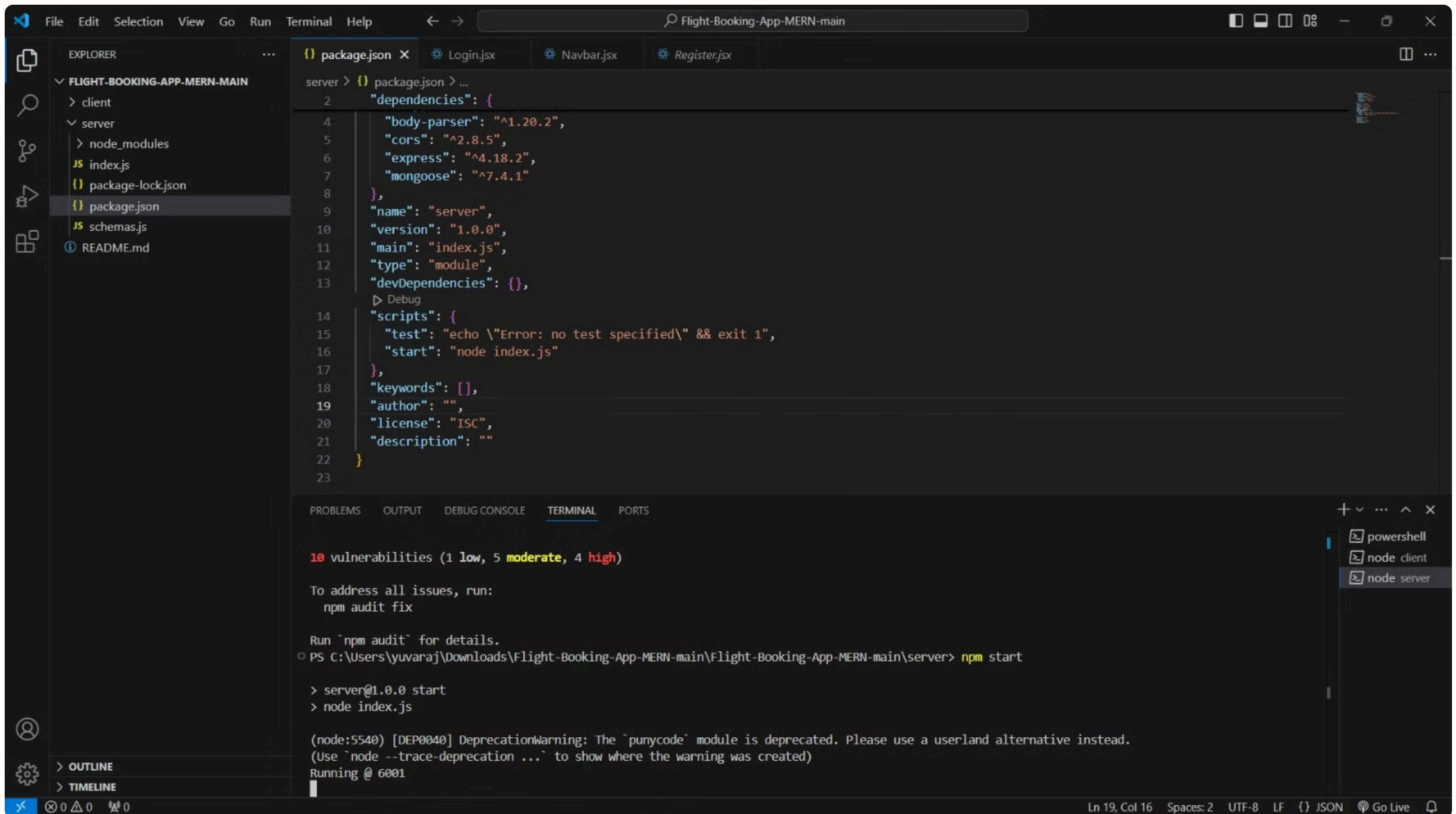
- EXPLORER:** Displays the file structure of the 'FLIGHT-BOOKING-APP-MERN-MAIN' project. The 'client' directory is expanded, showing 'node\_modules', 'public', and 'src'. The 'public' directory contains files like 'favicon.ico', 'index.html', 'logo192.png', 'logo512.png', 'manifest.json', and 'robots.txt'. The 'src' directory contains 'package-lock.json', 'package.json', 'README.md', 'index.js', 'package-lock.json', 'package.json', and 'schemas.js'.
- Register.jsx:** The active file in the editor. It shows a React component with a form for registration. The form includes an email input, a password input, and a select dropdown for user type. The code is as follows:

```
4  const Register = ({setIsLogin}) => {  
22      <input type="email" className="form-control" id="floatingInput" placeholder="Email" value={email} onChange={(e) => setEmail(e.target.value)} />  
23      <label htmlFor="floatingInput">Email address</label>  
24    </div>  
25    <div className="form-floating mb-3 authFormInputs">  
26      <input type="password" className="form-control" id="floatingPassword" placeholder="Password" value={password} onChange={(e) => setPassword(e.target.value)} />  
27      <label htmlFor="floatingPassword">Password</label>  
28    </div>  
29    <div>  
30      <select className="form-select form-select-lg mb-3" aria-label=".form-select-lg example" value={usertype} onChange={(e) => setUserType(e.target.value)}>  
31        <option value="">User type</option>  
32        <option value="admin">Admin</option>  
33        <option value="customer">Customer</option>  
34        <option value="flight-operator">Flight Operator</option>  
35      </select>  
36    </div>  
  }  
}
```
- TERMINAL:** Shows the output of an npm audit command. The output includes warnings about deprecated packages and a list of vulnerabilities. The command being executed is `npm start`.

```
npm warn deprecated w3c-hr-time@1.0.2: Use your platform's native performance.now() and performance.timeOrigin.  
npm warn deprecated workbox-cacheable-response@6.6.0: workbox-background-sync@6.6.0  
npm warn deprecated svgo@1.3.2: This SVGO version is no longer supported. Upgrade to v2.x.x.  
  
added 1506 packages, and audited 1507 packages in 22s  
  
241 packages are looking for funding  
  run `npm fund` for details  
  
26 vulnerabilities (1 low, 9 moderate, 15 high, 1 critical)  
  
To address issues that do not require attention, run:  
  npm audit fix  
  
To address all issues (including breaking changes), run:  
  npm audit fix --force  
  
Run `npm audit` for details.  
PS C:\Users\yuvaraj\Downloads\Flight-Booking-App-MERN-main\Flight-Booking-App-MERN-main\client> npm start
```

# Execution

## Serverside



The screenshot shows a Visual Studio Code editor window with the following components:

- EXPLORER:** Displays the file structure of the project "FLIGHT-BOOKING-APP-MERN-MAIN". The "server" folder is expanded, showing files like "index.js", "package-lock.json", "package.json", "schemas.js", and "README.md".
- Editor:** The "package.json" file is open, showing the following content:

```
server > {} package.json > ...
2  "dependencies": {
4    "body-parser": "^1.20.2",
5    "cors": "^2.8.5",
6    "express": "^4.18.2",
7    "mongoose": "^7.4.1"
8  },
9  "name": "server",
10 "version": "1.0.0",
11 "main": "index.js",
12 "type": "module",
13 "devDependencies": {},
14 "scripts": {
15   "test": "echo \"Error: no test specified\" && exit 1",
16   "start": "node index.js"
17 },
18 "keywords": [],
19 "author": "",
20 "license": "ISC",
21 "description": ""
22 }
23
```
- TERMINAL:** The terminal window shows the output of running "npm start" in the "server" directory. It displays a deprecation warning and the command "node index.js".

```
10 vulnerabilities (1 low, 5 moderate, 4 high)

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
PS C:\Users\yuvaraj\Downloads\Flight-Booking-App-MERN-main\Flight-Booking-App-MERN-main\server> npm start

> server@1.0.0 start
> node index.js

(node:5540) [DEP0040] DeprecationWarning: The `punycode` module is deprecated. Please use a userland alternative instead.
(Use `node --trace-deprecation ...` to show where the warning was created)
Running @ 6001
```
- PORTS:** The ports panel on the right shows three ports: "powershell", "node client", and "node server".

# Software Interfaces

**HOME PAGE**

**LOGIN PAGE**

**REGISTRATION PAGE**

**ADMIN INTERFACE**

**OPERATOR INTERFACE**

**CLIENT INTERFACE**



# HOMEPAGE

Tata Air Flights

[Home](#) [Login](#)

## Embark on an Extraordinary Flight Booking Adventure!

Unleash your travel desires and book extraordinary Flight journeys that will transport you to unforgettable destinations, igniting a sense of adventure like never before.

☐ Return journey

Departure City  
Select

Destination City  
Select

Journey date  
dd-mm-yyyy

Search

# LOGIN PAGE

## Login

Email address

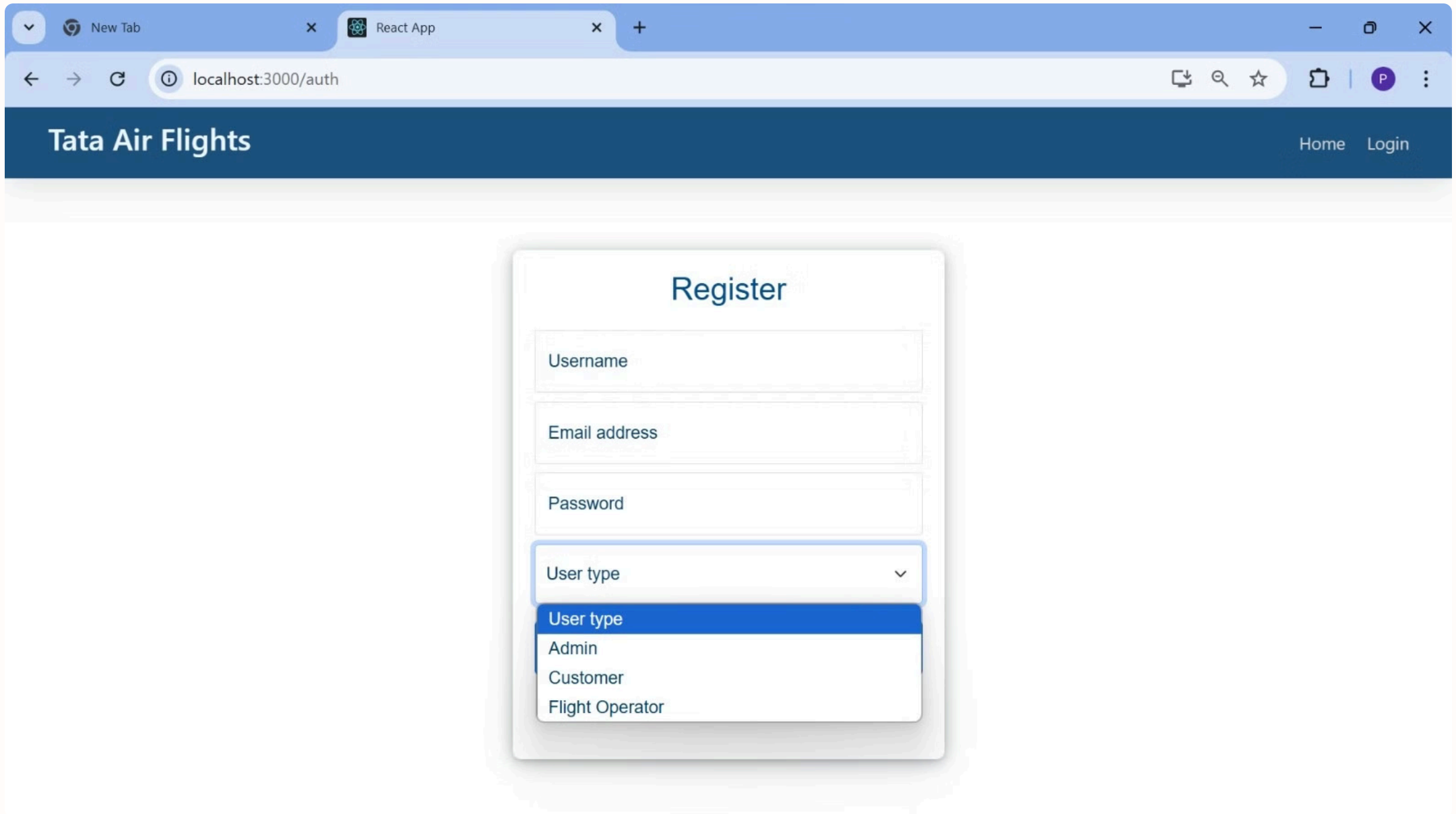
Password

Sign in

Not registered? [Register](#)



# REGISTRATION PAGE



The screenshot shows a web browser window with two tabs: 'New Tab' and 'React App'. The address bar displays 'localhost:3000/auth'. The page features a dark blue header with the 'Tata Air Flights' logo on the left and 'Home' and 'Login' links on the right. The main content area is white and contains a 'Register' form. The form has four input fields: 'Username', 'Email address', 'Password', and a 'User type' dropdown menu. The dropdown menu is open, showing three options: 'Admin', 'Customer', and 'Flight Operator'. The 'User type' dropdown is highlighted with a blue border and a blue background for the selected option.

Register

Username

Email address

Password

User type

User type

Admin

Customer

Flight Operator

# ADMIN INTERFACE

Tata Air Flights(Admin)

HomeUsersBookingsFlightsLogout

Users

9

View all

Bookings

4

View all

Flights

9

View all

New Operator Applications

Operator name:operator12

Operator email:operator123@gmail.com

Approve

Reject

Operator name:operator154

Operator email:operator154@gmail.com

Approve

Reject

Tata Air Flights(Admin)

HomeUsersBookingsFlightsLogout

Bookings

Booking ID: 6731b28f21ddc09b24394d93

Mobile: 902565207

Flight Id: FD8479

On-boarding: Chennai

Passengers:

1. Name: Praveen, Age: 21

2. Name: yuvaraj, Age: 21

Booking date: 2024-11-11

Journey Time: 12:56

Booking status: confirmed

Cancel Ticket

Email: gpkpraveenkumar143@gmail.com

Flight name: Operator\_2

Destination: Banglore

Seats: B-1, B-2

Journey date: 2024-11-12

Total price: 13800

Booking ID: 6731b25f21ddc09b24394d71

Mobile: FD8475

Flight Id: FD8479

On-boarding: Chennai

Passengers:

1. Name: Prasanth , Age: 21

2. Name: Ananth, Age: 21

3. Name: yuvaraj, Age: 21

Booking date: 2024-11-11

Journey date: 2024-11-12

Journey Time: 12:56

Total price: 0

Booking status: cancelled

# OPERATOR INTERFACE

Bookings  
2  
View all

Flights  
3  
View all

New Flight  
(new route)  
Add now

## Bookings

Booking ID: 6731b28f21ddc09b24394d93  
Mobile: 902565207      Email: gpkpraveenkumar143@gmail.com  
Flight Id: FD8479      Flight name: Operator\_2  
On-boarding: Chennai      Destination: Bangalore  
Passengers:  
1. Name: Praveen, Age: 21  
2. Name: yuvaraj, Age: 21  
Booking date: 2024-11-11      Journey date: 2024-11-12  
Journey Time: 12:56      Total price: 13800  
**Booking status: cancelled**

Booking ID: 6731b25f21ddc09b24394d71  
Mobile: FD8475      Email: gpkpraveenkumar13@gmail.com  
Flight Id: FD8479      Flight name: Operator\_2  
On-boarding: Chennai      Destination: Bangalore  
Passengers:  
1. Name: Prasanth, Age: 21  
2. Name: Ananth, Age: 21  
3. Name: yuvaraj, Age: 21  
Booking date: 2024-11-11      Journey date: 2024-11-12  
Journey Time: 12:56      Total price: 0  
**Booking status: cancelled**

Flight Name  
Operator\_2

Flight Id  
FD8475

Departure City  
Bhopal

Departure Time  
07:00

Destination City  
Jaipur

Arrival time  
11:00

Total seats

Base price

# CLIENT INTERFACE

Tata Air Flights

HomeBookingsLogout

Unleash your travel desires and book extraordinary Flight journeys that will transport you to unforgettable destinations, igniting a sense of adventure like never before.

Return journey

Departure City

Chennai

Destination City

Banglore

Journey date

20-11-2024

Search

Available Flights

Praveen\_05

Flight Number: FD8475

Start : Chennai

Departure Time: 11:00

Destination : Banglore

Arrival Time: 11:50

Starting Price: 2000

Available Seats: 50

Book Now

Operator\_2

Flight Number: FD8479

Start : Chennai

Departure Time: 12:56

Destination : Banglore

Arrival Time: 13:58

Starting Price: 2300

Available Seats: 50

Book Now

Tata Air Flights

HomeBookingsLogout

Book ticket

Flight Name: Praveen\_05

Flight No: FD8475

Base price: 2000

Email

gpkpraveenkumar143@gmail.com

Mobile

9023456778

No of passengers

1

Journey date

20-11-2024

Seat Class

Economy class

Passenger 1

Name

Manivasagan

Age

52

Total price: 2000

Book now

Tata Air Flights

HomeBookingsLogout

Bookings

Booking ID: 67320fc62add74bb4fefc129

Mobile:

9023456778

Email:

gpkpraveenkumar143@gmail.com

Flight Id: FD8475

Flight name: Praveen\_05

On-boarding: Chennai

Destination: Banglore

Passengers:

Seats: E-1

1. Name: Manivasagan, Age: 34

Booking date: 2024-11-11

Journey date: 2024-11-20

Journey Time: 11:00

Total price: 2000

Booking status: confirmed

Cancel Ticket

Booking ID: 6731b28f21ddc09b24394d93

Mobile:

902565207

Email:

gpkpraveenkumar143@gmail.com

Flight Id: FD8479

Flight name: Operator\_2

On-boarding: Chennai

Destination: Banglore

Passengers:

1. Name: Praveen, Age: 21

2. Name: yuvaraj, Age: 21

Booking date: 2024-11-11

Journey date: 2024-11-12

Journey Time: 12:56

Total price: 13800

Booking status: cancelled

Made with Gamma



# Conclusion and Future Enhancements

## **Streamlined Experience**

Seamless flight booking and management for users

## **Efficient Administration**

Comprehensive tools for administrators to manage the application

## **Scalable and Extensible**

Designed to handle growth and support future feature additions

