# Ram booking – Online Flight Booking System Full Stack MERN Project Documentation

---

- **1. Introduction**

- **Project Title:** Ram Booking – Online Flight Booking and Reservation System **Team Member:** Sriram

---

- **2. Project Overview Purpose**

**.Ram booking** is designed to digitize and streamline the entire flight ticket booking process. The system eliminates traditional offline or agent-based booking methods by offering a fast, secure, and user-friendly online platform.

- It provides passengers, airlines, and administrators seamless tools for:
- Real-time flight search
- Seat availability checking
- Ticket booking and cancellation
- User profile management
- Airline management & admin monitoring **Ram booking aims to:**
- Allow users to search flights based on **source, destination, date, and airline**.
- Provide passengers with a smooth interface for **booking, cancelling, and viewing tickets**.
- Provide airlines a structured portal to **manage flights, schedules, and seat availability**.
- Enable administrators to **manage flights, users, bookings, and airline approvals**.
- Improve workflow efficiency with fast cloud-based access and reduced manual errors.
- Ensure secure & authenticated interactions using a solid MERN architecture. **Features**
- JWT-based user authentication
- Flight search & filter system
- Real-time seat availability
- Passenger dashboard (bookings, cancellations)
- Airline dashboard (manage flights, schedules, seats)
- Admin panel (manage users, airlines, flights)
- Secure REST API
- Booking history & e-ticket generation

---

**3. Architecture**

- **Frontend (React.js):**
- React with Hooks and Context API
- Axios for API communication
- Bootstrap/Material UI for UI components • React Router DOM + Protected Routes
- **Backend (Node.js + Express.js):**

- RESTful API architecture
- Authentication middleware
- Role-based access control (user/airline/admin) **Database (MongoDB + Mongoose):**
- Collections:
- Users
- Flights
- Bookings
- Airlines
- Schema validation using Mongoose models.

---

### 4. Setup Instructions

- **Prerequisites**
- Node.js (v16+)
- MongoDB / MongoDB Atlas
- Git
- **Installation Steps**
- Clone the repository
- Install dependencies in both **client** & **server** folders
- npm install
- Create .env in backend:
- MONGO_DB="mongodb_connection_string"
- JWT_SECRET="your_secret_key"
- Start both frontend & backend
- Frontend: npm start
- Backend: nodemon index.js

---

### 5. Folder Structure

- **Client (React):**
- /src/components – UI Components
- /src/pages – Flight pages, booking pages
- /src/context – Global state management
- /src/utils – Helper functions
- **Server (Node.js):**
- /routes – API endpoints
- /controllers – Business logic
- /models – Mongoose Schemas
- /middleware – Auth & security

### 6. Running the Application

- **Frontend:** npm start
- **Backend:** nodemon index.js

### 7. API Documentation

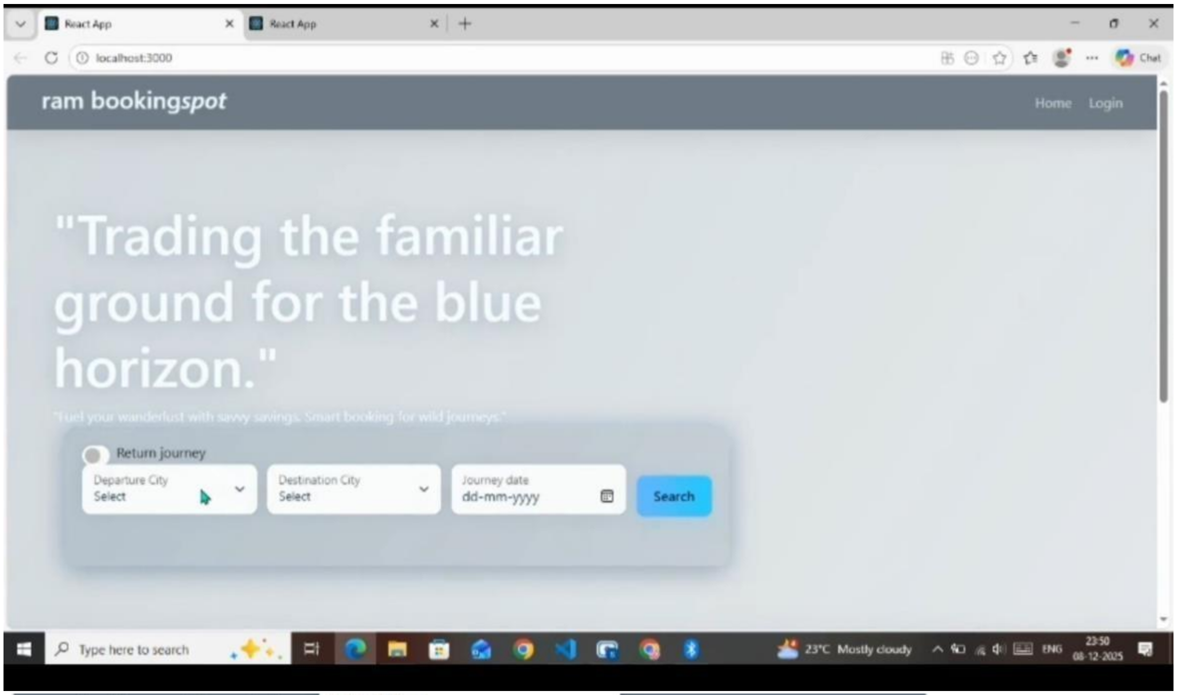
- **User APIs**
- POST /api/user/register – Create account
- POST /api/user/login – Login user
- GET /api/user/profile – Fetch user profile
- **Flight APIs**
- GET /api/flight/search – Search flights
- POST /api/flight/add – Add flight (Airline/Admin)
- POST /api/flight/update – Update flight details
- **Booking APIs**
- POST /api/booking/create – Book ticket
- GET /api/booking/user-bookings – User booking history
- POST /api/booking/cancel – Cancel booking
- **Admin APIs**
- GET /api/admin/get-airlines – View all airlines
- POST /api/admin/approve-airline – Approve/Reject airline
- GET /api/admin/get-users – View all users
- GET /api/admin/all-bookings – View all bookings

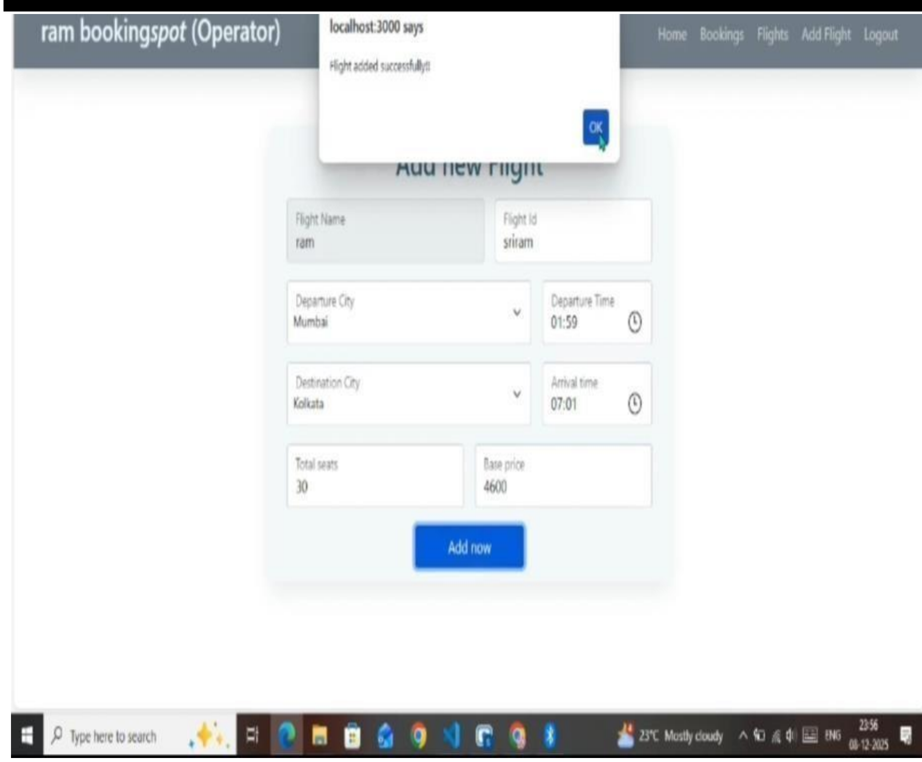
### 8. Authentication

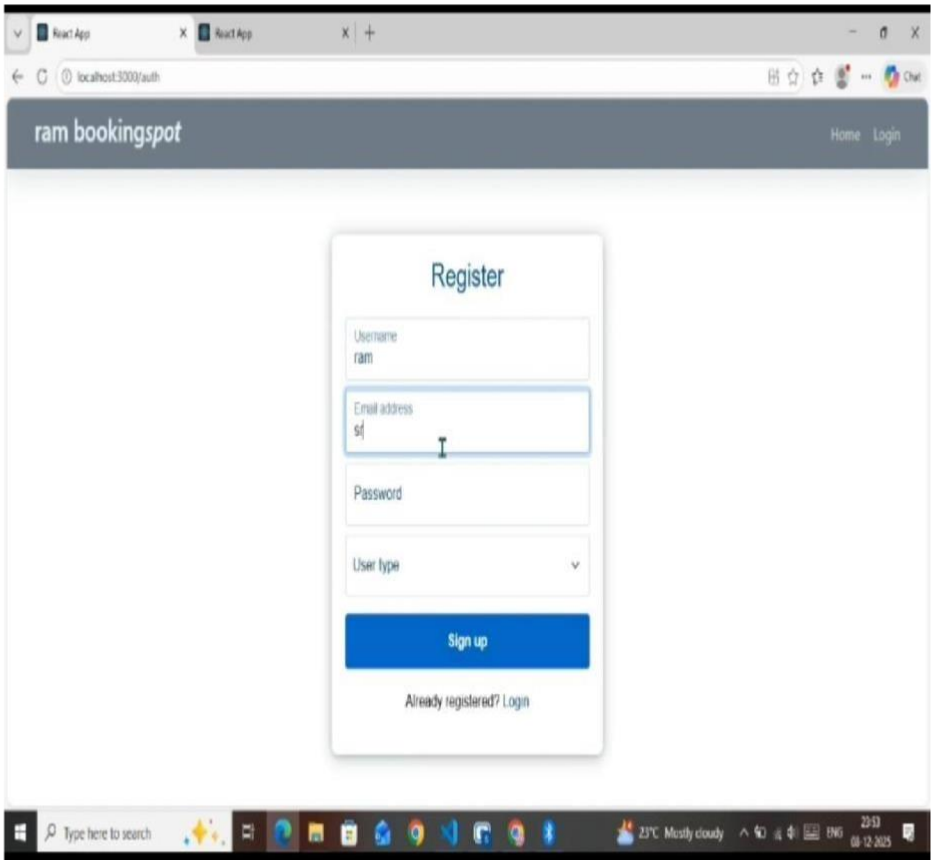
- JWT-based authentication • Tokens stored in Local Storage
- Role validation:
- **User** (Passenger)
- **Airline** • **Admin**

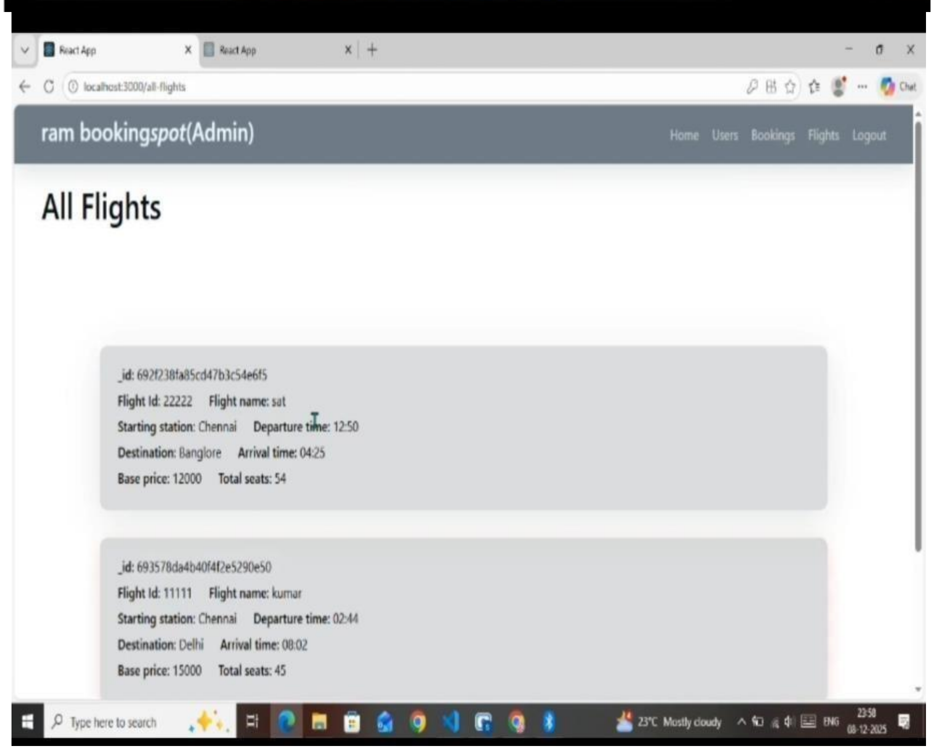
- **9. User Interface**

- *(Add screenshots of UI pages: Flight Search, Booking Page, Dashboard, etc.)*


React App
React App
localhost:3000
ram bookingspot
Home
Login
"Trading the familiar ground for the blue horizon."
"Fuel your wanderlust with savvy savings. Smart booking for wild journeys."
Return journey
Departure City
Select
Destination City
Select
Journey date
dd-mm-yyyy
Search
Type here to search
23°C Mostly cloudy
ENG
23:50
08-12-2025


ram bookingspot (Operator)
localhost:3000 says
Flight added successfully!
OK
Home
Bookings
Flights
Add Flight
Logout
Add new Flight
Flight Name
ram
Flight Id
sriram
Departure City
Mumbai
Departure Time
01:59
Destination City
Kolkata
Arrival time
07:01
Total seats
30
Base price
4600
Add now
Type here to search
23°C Mostly cloudy
ENG
23:56
08-12-2025


React App
localhost:3000/auth
ram bookingspot
Home
Login
Register
Username
ram
Email address
Password
User type
Sign up
Already registered? Login


React App
localhost:3000/all-flights
ram bookingspot(Admin)
Home
Users
Bookings
Flights
Logout
All Flights
_id: 692f238fa85cd47b3c54e6f5
Flight Id: 22222 Flight name: sat
Starting station: Chennai Departure time: 12:50
Destination: Banglore Arrival time: 04:25
Base price: 12000 Total seats: 54
_id: 693578da4b40f4f2e5290e50
Flight Id: 11111 Flight name: kumar
Starting station: Chennai Departure time: 02:44
Destination: Delhi Arrival time: 08:02
Base price: 15000 Total seats: 45

### 10. Testing

- Manual functional testing
- Endpoint testing with Postman
- JWT authentication testing

### 11. Screenshots or Demo

- *(Attach screenshots or provide demo/video link)*

### 12. Known Issues

- Occasional delay in seat availability update
- Airline dashboard UI improvements needed

### 13. Future Enhancements

- Add online payment gateway integration
- Add boarding pass QR generation
- Live flight tracking
- Add email/SMS updates for booking & cancellations