Planning Phase

Project Logic and Planning

Date	26 June 2025
Team id	LTVIP2025TMID20321
Project Name	Flight Finder
Maximum Marks	2 Marks

1. Overall Workflow Logic

1. **User Registration & Login**
   - Users and operators register via the form.
   - Admin must approve operator accounts.
   - JWT is used for authentication post-login.
2. **Flight Search & Booking**
   - Users enter: departure city, destination city, and journey date.
   - System filters and shows relevant flight data from MongoDB.
   - User selects flight, enters passenger details, and books.
   - Booking confirmation is shown and saved in collection of booking.
3. **Admin Actions**
   - Can view all users, flights, and bookings.
   - Approves or rejects flight operator requests.
   - Adds or edits flight details.
4. **Flight Operator Actions**
   - After admin approval, logs in to their dashboard.
   - Can add new flights, edit existing ones, or delete them.
   - Can view bookings made on their flights.
5. **User Actions**
   - Can view, cancel, or rebook their tickets.
   - Can check confirmation, journey details, and price summary.

2. Component Logic (MERN Stack)

Frontend (React.js)

- Handles routing, forms, and state.
- Uses Axios to call backend APIs.
- React Router for navigation and protection (e.g., /admin, /bookings).

Backend (Node.js + Express)

- Manages API routes like /register, /login, /fetch-flights, /book-ticket.
- Applies JWT-based route protection.
- Handles business logic for approval, booking, filtering, etc.

Database (MongoDB + Mongoose)

- Collections:
  - users: login data, role, approval
  - flights: flight info
  - bookings: user bookings

3. Role-Based Access Logic

Role	Permissions
Guest	Register, Login
User	Search flights, book, cancel, view bookings

Role	Permissions
Flight Operator	Add/edit/delete flights (post approval), view bookings
Admin	View all data, approve operators, manage users and flights

Sample Flight Booking Flow

1. **Input**: from, to, journeyDate
2. **System**: Filters flights from MongoDB
3. **User**: Selects a flight → enters email, mobile, no. of passengers
4. **System**: Calculates total price = basePrice × passengers × class multiplier
5. **User**: Confirms booking