Requirement Gathering and Analysis Phase
Solution Architecture

Date	8 July 2024
Team ID	SWTID1720170691
Project Name	Flight Booking APP
Maximum Marks	2

Solution Architecture:

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.

- Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.

- Define features, development phases, and solution requirements.

- Provide specifications according to which the solution is defined, managed, and delivered.

The flight booking management system is developed to streamline the booking process, manage real-time flight information, and enhance customer experience. This solution architecture illustrates the code components used in creating the application and the role each part plays in making the system functional.

Each component of the solution architecture diagram is crucial for the usability of the application. Solution architecture is a systematic approach to designing a solution that aligns with a business's needs and goals. It helps developers define the requirements and strategies needed to implement a software solution.

Solution Architecture Diagram

Flight Booking System

Backend Technologies

Node.js
- A runtime environment for executing JavaScript code server-side.
- Allows building scalable and fast network applications.
- Utilizes event-driven, non-blocking I/O model.

Express
- A fast, unopinionated, minimalist web framework for Node.js.
- Facilitates handling routes, requests, and middleware.
- Provides robust set of features to develop web applications.

Database

MongoDB
- A NoSQL, document-oriented database.
- Stores data in JSON-like format, making it flexible.
- MongoDB provides high performance and easy scalability.

Frontend Technologies

HTML
- The standard markup language for creating web pages.
- Provides the structure for the web application.
- Tags used to embed images, hyperlinks, and other content.

CSS
- Styles the HTML elements to create an attractive user interface.
- Controls layout, fonts, colors, and spacing.
- Can be used with pre-processors like Sass.

JavaScript
- A programming language that enables dynamic behavior on web pages.
- Can manipulate HTML and CSS after the page has loaded.
- Facilitates user interactions, animations, and AJAX operations.

API Endpoints

User Management
- `/users/signup`: Endpoint for user registration.
- `/users/login`: Endpoint for user authentication.
- `/users/profile`: Endpoint to access user profile.

Flight Management
- `/flights/search`: Endpoint to search available flights.
- `/flights/book`: Endpoint to book a flight.
- `/flights/cancel`: Endpoint to cancel a booking.

User Interface

Homepage
- Displays search form for flights.
- Offers login and sign-up options.

Search Results
- Shows available flights based on user criteria.
- Allows selection and booking of flights.

Booking Confirmation
- Displays booking details.
- Provides options to download ticket and view booking history.

Profile Page
- Shows user details and booking history.
- Allows users to update their info and manage bookings.

Middleware

Authentication
- Middleware to verify user credentials.
- Secures private routes like profile and booking management.

Error Handling
- Catches and processes application errors.
- Includes custom messages and logging mechanisms.