GlobeTrek

Parthib Barua
BEng (Hons) Software & Electronic Engineering

Ollscoil Teicneolaíochta an Atlantaigh

Atlantic Technological University

GLOBETREK
FIND YOUR PLACE

Summary

GlobeTrek is a smart hotel management web app that simplifies travel planning. It offers various interesting features.

1. **Find & Book Hotels with Ease**
   - Search hotels by city with integrated sorting system based on price and ratings.
2. **Explore Hotels Before Booking**
   - View high-quality images and additional features with price
3. **Real-Time Navigation & Map Integration with weather updates**
   - Built-in Google Maps for real time navigation which also features near-by places in the map
   - Real-Time weather updates based on current location

With the user dashboard, it enables efficient management of past reservations, allowing users to review the hotels and modify their choices of stay.

Tools & Technologies

1. **Frontend**
   React.js: A highly efficient and dynamic user interface
   Tailwind CSS: A utility first CSS framework for styling
   Axios: Handles API requests for efficient communication
2. **Backend**
   Node.js & Express.js: Fast and scalable backend for handling hotel and user management
   JWT (JSON Web Token): Ensure secure authentication
   MongoDB Atlas: A cloud based NoSQL database for storing user data and hotel details
3. **External APIs & Integrations**
   Google Maps API: Enables interactive map views, real-time navigation and nearby location search
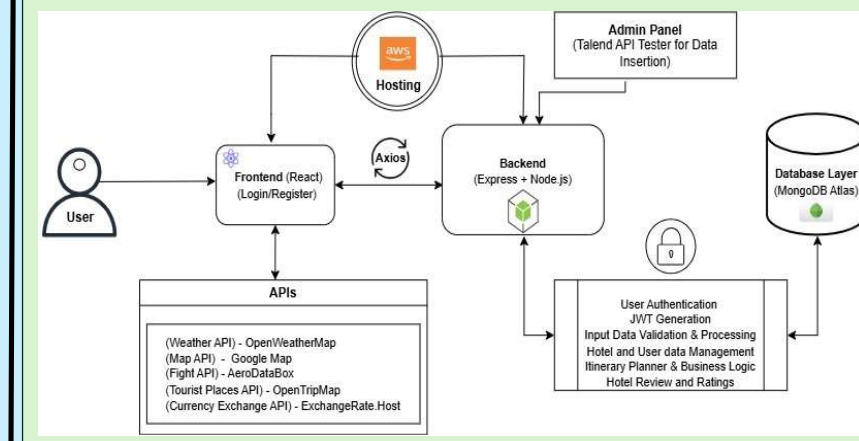   **OpenWeatherMap API:** Real-time weather updates
   Skyscanner API: Integrates flight booking services
4. **Hosting**
   AWS (Amazon Web Services): Hosts the application

Architecture Diagram

UI-Showcase

How it Works

- **Smooth Frontend-Backend Interaction**
  1. When a user searches for hotels or flights, React.js sends a request via **Axios** to the **Node.js** backend
  2. The backend processes the request, interacts with **MongoDB Atlas**, and retrieves relevant data
  3. Then response is sent back to the frontend, which updates the UI dynamically using **React State Management**
- **Secure User Authentication with JWT & Cookies**
  1. When a user logs in, their credentials are verified in **MongoDB**
  2. On successful login, a **JWT (JSON Web Token)** is generated by backend and sends it to the frontend
  3. The token is stored in **HTTP-only cookies**, preventing Cross-site Scripting
- **Efficient Data Fetching & Caching:**
  1. **MongoDB Atlas** handles large-scale data storage, allowing quick retrieval of hotel listings, user reviews, and bookings
  2. **Smart cached responses** reduce repeated API calls by temporarily storing user search data which helps to minimize load-times and ensures smooth, real-time browsing experience