



TRAVEL AGENCY MANAGEMENT SYSTEM

FULL-STACK WEB APPLICATION

TravelBugz



PRESENTED BY:

Yug Sarda
(2023UCP1663)

Tanuj Tenguriya
(2023UCP1649)

Khushal Saini
(2023UCP1678)

Introduction to the Project

Project Overview

- A web-based solution for managing travel bookings and customer interactions.
- Objective: Streamline travel agency operations and improve customer experience.
- Built with the MERN Stack: MongoDB, Express.js, React.js, and Node.js.



Project Structure



Travel-Agency-Management/

Backend/

- src/
 - src/ controllers: / Business logic for routes
 - models/ Database models (e.g. Mongoose schemas shemes.)
 - routes/ API routes (bookings, users, packages, etc.)
 - services/ External services (e.g, email, payment)
 - middlewares/ Authentication, validation, error handling
 - config/ Config / helper functions
 - utils/ Helper functions
- .env (local only)
- server.js / index.js Main server entry point
- package.json

Frontend/

- public/ Static assets, index.html
- src/
 - assts/ Images, icons, etc, (images)
 - components/ Reusable components (Navbar, Card, etc.)
 - pages/ Main pages (Home, Bookings, AdminDashboard)
 - services/ API calls (axios instances, etc.)
 - context/ Global state (e.g. auth, bookings)
 - routes/ Route configs
- App.js Root component
- index.js Entry point (ReactDOM.render)

README.md

settings

File structure

@reallygreatsite

Key Features



- User Authentication (Signup/Login)
- Admin dashboard to manage packages and bookings
- Dynamic package listings with filters
- Booking system connected to MongoDB
- Users can book packages, choose travel details, and view their booking history.

Technologies Used

- React.js & React Router DOM - For building a dynamic, component-based frontend with smooth navigation.
- Tailwind CSS & Framer Motion - For modern UI styling and animations.
- Axios - To handle HTTP requests between frontend and backend.
- Express.js & Node.js - Backend framework to build RESTful APIs.
- MongoDB & Mongoose - NoSQL database with schema modeling for storing users, packages, and bookings.
- JWT & dotenv - For secure authentication and environment variable management.
- Git & GitHub - For version control and collaborative development.
-

Methodology and Approach

Users browse packages via React frontend

Built using React.js with dynamic UI components for filtering and searching travel packages.

Bookings are made and sent to backend API

Axios is used to send HTTP requests to the backend (e.g., for bookings, login/signup)

Express backend handles logic and routes

Backend APIs are organized using MVC pattern with authentication and role-based access

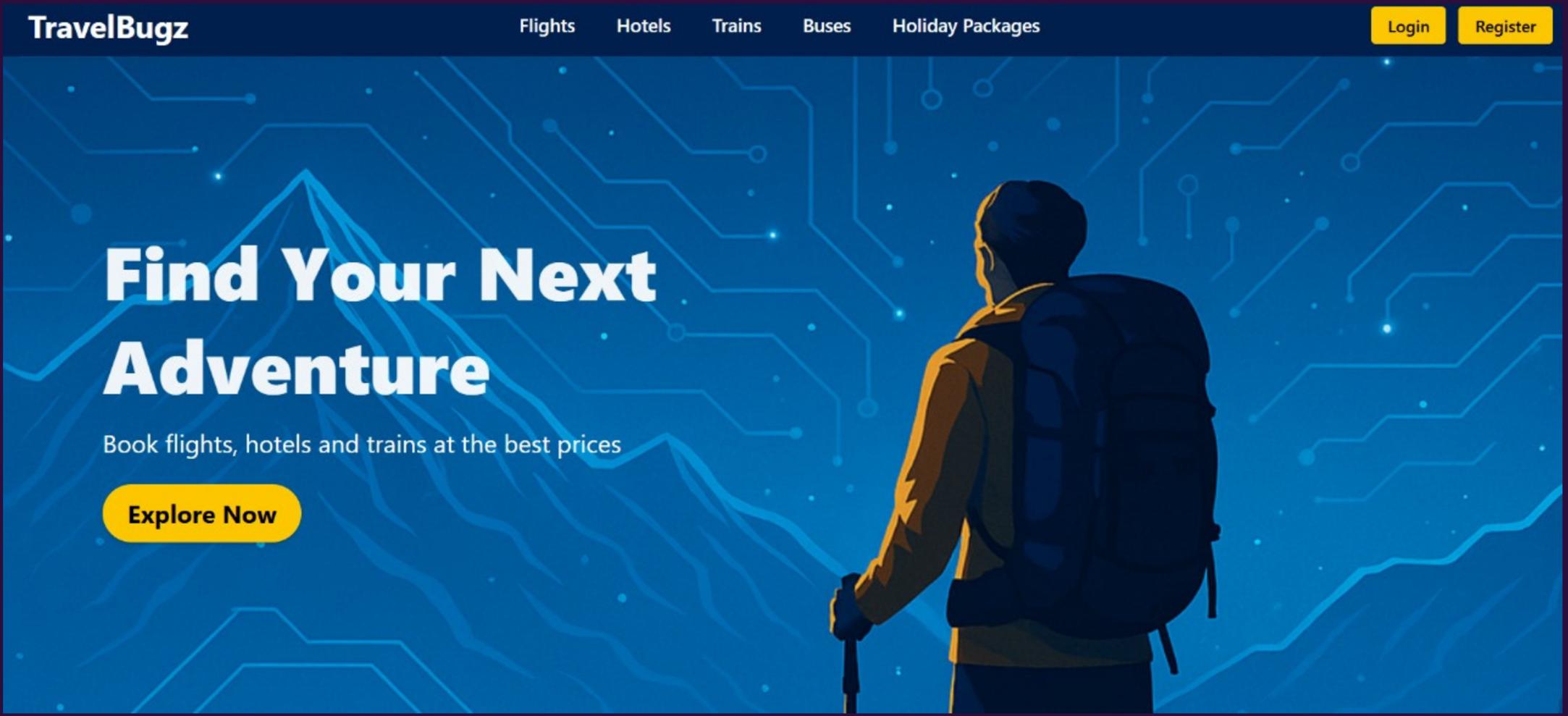
MongoDB stores user, booking, and package data

Mongoose is used to define schemas and interact with the database

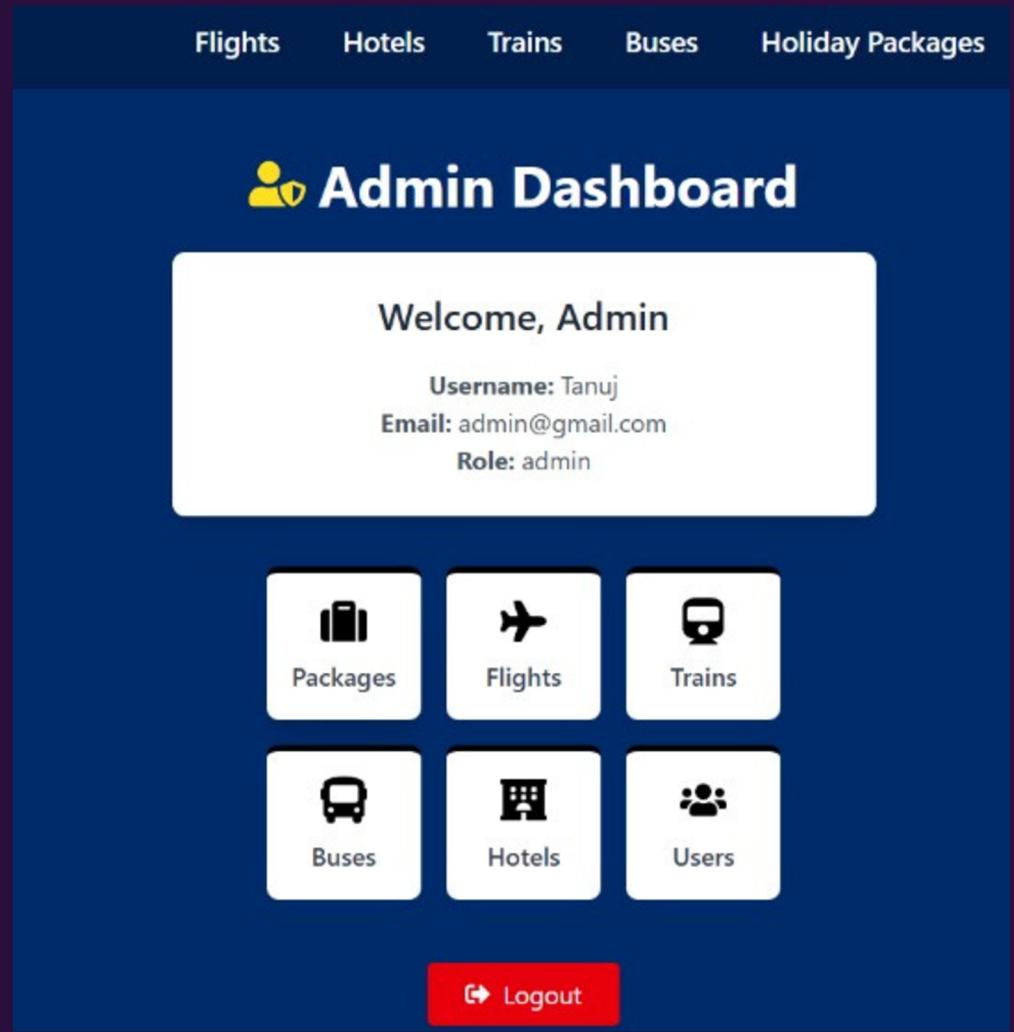


Highlights

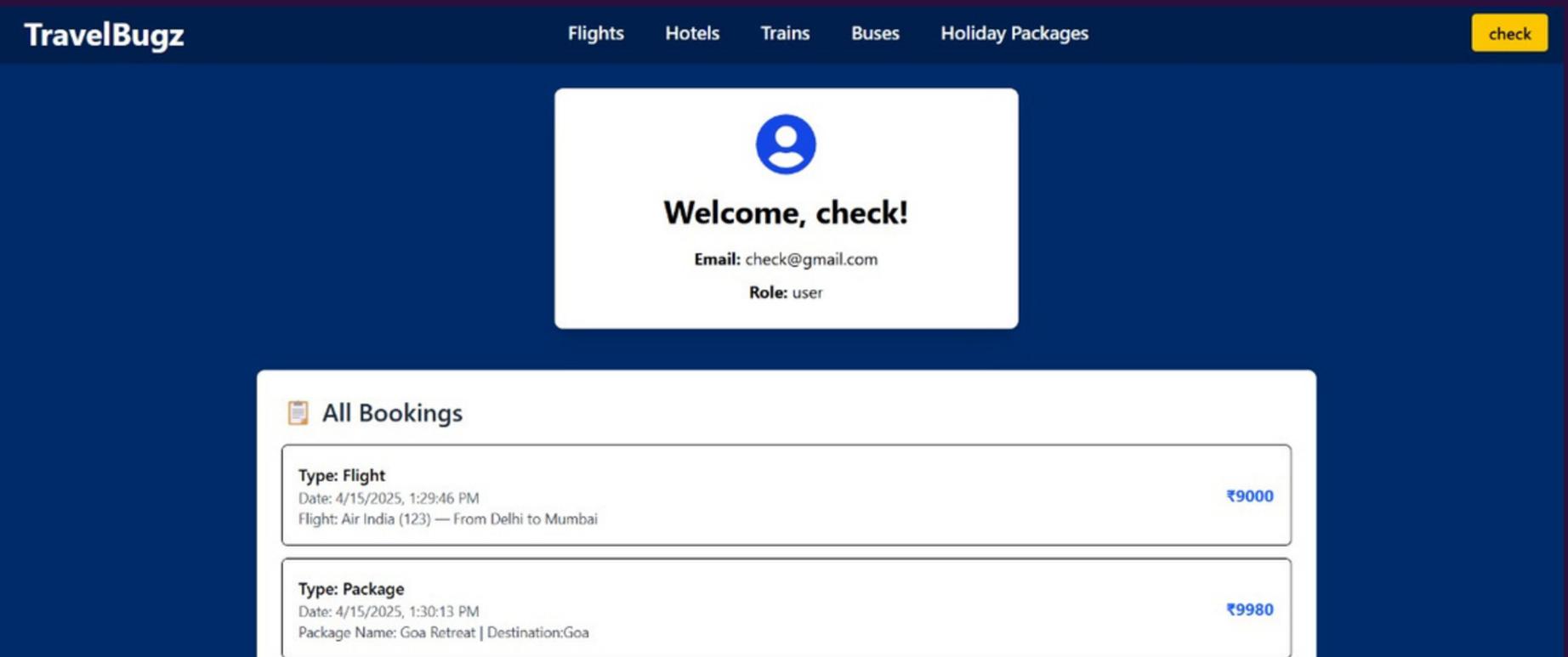
Home page with various options



The TravelBugz home page features a blue background with a digital circuit board pattern. In the center, there's a silhouette of a person with a backpack walking up a mountain. The text "Find Your Next Adventure" is prominently displayed in large white letters. Below it, a smaller text says "Book flights, hotels and trains at the best prices". A yellow button labeled "Explore Now" is located at the bottom left. At the top, there's a navigation bar with links for Flights, Hotels, Trains, Buses, and Holiday Packages, along with Login and Register buttons.



The Admin Dashboard has a dark blue header with navigation links for Flights, Hotels, Trains, Buses, and Holiday Packages. The main area starts with a "Welcome, Admin" message and user details: Username: Tanuj, Email: admin@gmail.com, Role: admin. Below this are six square buttons with icons: Packages (luggage), Flights (airplane), Trains (train), Buses (bus), Hotels (building), and Users (people). At the bottom right is a red "Logout" button.



The User Dashboard shows a welcome message for "check" with their email (check@gmail.com) and role (user). Below this is a section titled "All Bookings" showing two entries: a flight from Delhi to Mumbai for ₹9000 and a package to Goa for ₹9980. The dashboard also includes a navigation bar with links for Flights, Hotels, Trains, Buses, and Holiday Packages, and a "check" button at the top right.

Admin Dashboard View

User Dashboard



Challenges faced

- Setting up Git & GitHub syncing
- Installing and configuring Node, npm, MongoDB
- Connecting frontend and backend, especially managing state across components and APIs
- Implementing secure login systems for admins and users
- Ensuring input validation both on client and server side to avoid errors and injection attacks

Future Improvements

- Integrate online payment system
- Improve UI with animations
- Add search filters and user ratings
- Deploy using Render or Vercel



Project Outcomes and Benefits



Project Planning

Understood the importance of organizing tasks and setting milestones

Technical Skills

Gained hands-on experience with frontend, backend, and database integration

Problem Solving

Tackled real-world challenges like user validation and data management

Teamwork

Improved collaboration and communication within the team

Documentation & Testing

Understood the significance of proper documentation and system testing

Thank You

