

Chasing Horizons

Description

Chasing Horizons is a comprehensive web application designed for managing and booking travel-related services, including hotels and tours. It provides distinct interfaces and functionalities for regular users, hotel managers, and administrators.

Group Members

Lokesh Kudipudi - S20230010135

Meghana Kuruva - S20230010134

Rohin Sai Bogadi - S20230010047

Thotam Reddy Palla - S20230010177

Sneha Sri Sai Vadde - S20230010246

GitHub URL

<https://github.com/Lokesh-Kudipudi/FFSD-SEM4>

Types of Users

The platform caters to three main types of users:

1. **Regular Users/Customers:** Can browse and book hotels and tours, manage their trips, and update their settings.
2. **Hotel Managers/Owners:** Can manage their hotel listings, view bookings, and manage room availability.
3. **Administrators:** Have oversight of the entire platform, including managing users, hotels, tour packages, and viewing analytics.

Dashboards

The application features dedicated dashboards for each user type:

1. **Admin Dashboard:** (`Views/dashboard/admin/`)
 - Analytics (`analytics.ejs`)
 - Customer Management (`customers.ejs`)
 - Hotel Management (`hotelManagement.ejs`)
 - Package Management (`package.ejs` , `packages.ejs`)
2. **Hotel Manager Dashboard:** (`Views/dashboard/hotelManager/`)
 - Booking Management (`booking.ejs`)
 - Room Management (`roomsAdd.ejs` , `roomsIndex.ejs`)
3. **User Dashboard:** (`Views/dashboard/user/`)
 - My Trips (`myTrips.ejs`)
 - User Settings (`settings.ejs`)

Implemented Features

- **User Authentication:** Secure sign-in and sign-up for all user types (`auth/signIn.html` , `auth/signUp.html` , `auth/signUpHotelManager.html` , `middleware/authentication.js`).
- **Hotel Browsing and Booking:** Users can search, view details, and book hotels (`Controller/hotelController.js` , `Model/HotelModel.js` , `Views/hotels/`).
- **Tour Browsing and Booking:** Users can explore and book tour packages (`Controller/tourController.js` , `Model/tourModel.js` , `Views/tours/`).
- **Booking Management:** Users can view their bookings; Hotel Managers and Admins can manage all bookings (`Controller/bookingController.js` , `Model/bookingModel.js`).
- **Hotel Property Management:** Hotel Managers can add and manage their rooms and hotel details (`Views/dashboard/hotelManager/roomsAdd.ejs` , `roomsIndex.ejs`).
- **Admin Control Panel:** Comprehensive management of users, hotels, tours, and analytics (`Controller/adminController.js` - inferred, `Views/dashboard/admin/`).
- **User Profile Management:** Users can update their personal information and settings (`Views/dashboard/user/settings.ejs`).
- **Contact System:** A way for users to contact support or make inquiries (`Controller/contactController.js` , `Model/contactModel.js` , `public/html/contact.html`).
- **Recommendation Engine:** Suggests tours or hotels to users (`Views/recommendation.ejs` , `public/js/recommendation.js`).

Future Ideas

- **Enhanced AI Features:** Further leverage the Gemini API (`api/gemini.js`) for personalized recommendations, travel planning assistance, or customer support.
- **Advanced Analytics & Reporting:** More detailed and visual analytics for admins and hotel managers.
- **Real-time Notifications:** For booking confirmations, changes, and platform updates.
- **Expanded Payment Gateway Options:** Integration with more payment providers.
- **Social Media Integration:** Sharing trips, reviews, or logging in with social accounts.
- **Mobile Applications:** Native or PWA mobile apps for users and managers.
- **Loyalty Program:** Rewards for frequent users.
- **Multi-language Support:** To cater to a global audience.
- **Interactive Maps:** For browsing hotels and tour locations.