

Assignment_ID: assignment_category_0004

Project Overview

We're seeking a skilled Hotel Booking Platform Developer to lead the creation of a modern, user-centric booking platform. Your role involves designing and implementing frontend and backend systems to ensure seamless functionality across devices. Key responsibilities include feature development and database management. You should have proficiency in frontend technologies (HTML, CSS, JavaScript, React.js) and backend development using Node.js, Express.js, and MongoDB. Experience with Firebase authentication, JWT token implementation, and familiarity with Tanstack Query and animation libraries are advantageous. You will play a crucial role in shaping the future of hotel accommodation booking, enhancing user experiences, and ensuring the success of this innovative platform. Join us in shaping the future of online hotel booking with cutting-edge technology and unparalleled user experience.

🚩: 0 [If we have any update we will mention it here]

Key Rules:

- Include a minimum of 18 notable GitHub commits on the client side
- Include a minimum of 8 notable GitHub commits on the server side
- Add a meaningful readme.md file with the name of your website and a live site URL. Include a minimum of five bullet points to feature your website. Write your selected category's name here.
- Make it responsive for all devices. You need to make it responsive for mobile, tablet, and desktop views.
- Use the Environment variable to hide the Firebase config keys and MongoDB credentials.
- Don't use any Lorem ipsum text; you can not use the default alert to show any error or success message.

Main Features

Homepage Design

1. **Banner:** Use images or videos of hotel rooms to create a visual experience for users.
2. **Map:** Include a map that displays the hotel's location.
3. **Newsletter Signup:** Encourage users to subscribe to your newsletter for updates, deals, and exclusive offers.
4. **Featured Rooms:** Highlight a selection of featured rooms with images and descriptions. Use a "Book Now" button for quick access.
5. **User Reviews:** (see challenges part)

User Authentication

- Users can create an account with email and password.
- Users can log in using Google authentication via Firebase.

Navigation Bar

- The navigation bar should have links to the "Rooms" page and "My Bookings" page.
- The "My Bookings" page should be accessible only to authenticated users.
- About Us and Contact Us are optional but highly recommended.

Rooms Page

- Display a list of available rooms.
- There will be an image of the room. After clicking an individual picture, (it will redirect the user to the room details page).
- Implement a filter system to filter rooms by price range. (filter should be implemented on the server side).
- Users can view the total review (total review count of that room).
- Users can post reviews for rooms after booking. [See details](#)

Room Details Page

- Room Description
- Price per Night
- Room Size
- Availability
- Room Images
- Special Offers (if available)
- There will be a **Book Now**. On clicking Book Now the user can book a room. Make sure the room will be **unavailable** when someone books the room.
- Reviews (if unavailable else show any meaningful message) [to know how users will add reviews, follow [this](#)]
- Users can select a room and choose the booking date (single day) using a date picker.
- The system should ensure that users can only book available rooms.
- After clicking the book now button, a Modal will pop up where Users can view the room summary of their booking and a **confirm Button**. By Clicking the confirm Button user can book that room. [room summary means what they choose like price, date, room description, etc.]

My Bookings Page

- Display the list of the rooms booked by the logged-in user.
- Users can **cancel** the booking.
- Upon clicking the **Cancel** button, prompt the user to confirm the cancelation with a confirmation modal.
- Users can post reviews from here. [See details](#)
- Next to each booking, provide an "Update Date" button.
- Clicking the "Update Date" button should allow the user to select a new date for the booking, similar to the process for updating bookings in the "Rooms" section.
- Upon successful modification, update the booking details in the database and notify the user of the update using modal or toast.

Review System

- Users can post reviews for rooms only they have booked.
- Reviews should include a username, rating, comment, and timestamp.
 - The username field will be non-editable. Take the value from the logged-in user.
 - Rating should be a number filed between 1-5
- Reviews should be shown on the room details page for other users.

Access Control

- Users who are not logged in cannot book a room. If the user clicks the book button, they will be redirected to the login page.
- Users who are not logged in can only view basic room details but cannot post reviews.

404 Page 🚀

- Create a 404 page with an exciting jpg/gif and a "Back to home" button that redirects the user to the home page.

Package

- Implement a minimum of the 2 packages given below.
 - Update the browser tab title, meta-data ([Helmet](#))
 - Animation library AOS.
 - Map Library. ([pigeon-maps](#), [leaflet](#))

Challenges Part 🌟

User Reviews

- Display authentic user reviews and ratings to build trust on the home page. Consider using a testimonial carousel for variety. (show the reviews from the database). Make sure the reviews are sorted in descending order based on the timestamp. (the latest reviews should be shown first)
- Check the docs for [sorting](#) from the server side.

Special Offers and Promotions

- Prominently showcase special offers, discounts, and promotions on the home page. Use eye-catching images.
- When users will visit the home page they will see the **Special Offers and Promotions** in a popup or modal. See this [image](#)

Booking Cancellation 🚫

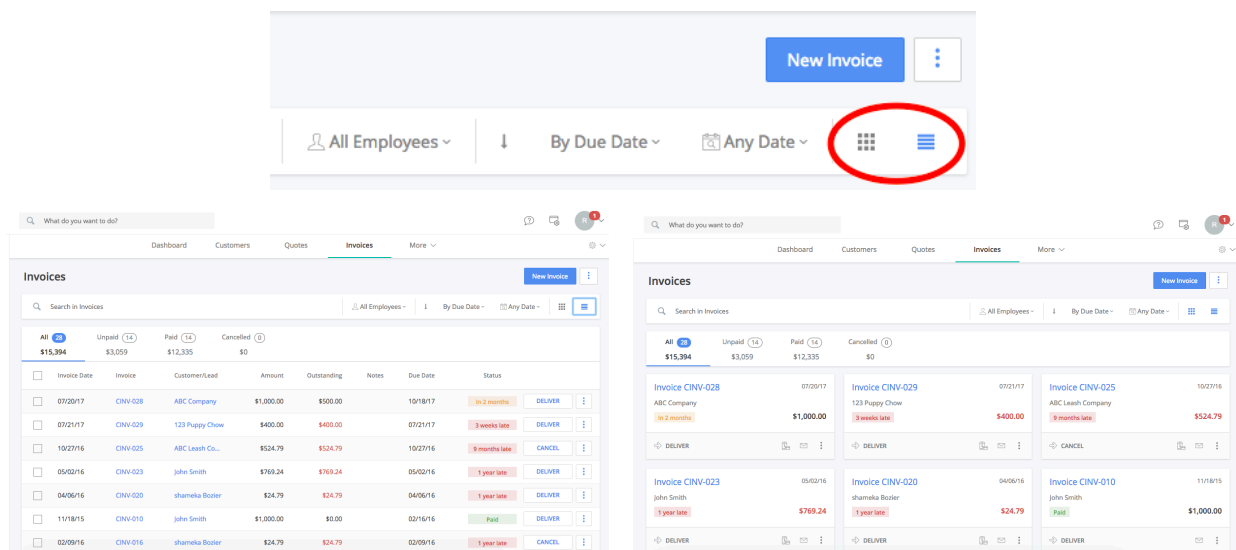
- Users can cancel a booking 1 day before the booked date. Suppose the user booked a room on 9/11/2024. If he wants to cancel the booking then he can cancel within 7/11/2024.
- Make sure after canceling the room the room will be available again.
- you can use [momentjs](#) to compare the time.

JWT Authentication 🔑

- Upon login, you will create a JWT token and store it on the client side. You will send the token with the call and verify the user. Implementing 401 and 403 is optional. Ensure you have implemented the JWT token, create a token, and store it on the client side for both email/password-based authentication and social login. You must implement JWT on your private routes.

Optional Part

1. Users can book the date duration. (Suppose the user can select 12/5/2024 - 15-5/2024).
2. **Gallery:** Create a gallery page with images of the hotel, rooms, amenities, and events. Users can explore and visualize their stay.
3. **Animation** 🎉
 - Use animation in the home page sections.
4. Make a sort system based on the available offers, and date range on the rooms page.
5. Make a system to toggle card to table view or vice versa.



Remember to maintain a consistent design and user experience throughout your website. Tailor these additional features and recommendations to suit your hotel booking platform's unique characteristics and goals.

Guidelines

1. Save time on the website idea. Just spend 15-20 minutes deciding, find a sample website, and start working on it.
2. Do not waste much time finding the right image. You can always start with a simple idea. Make the website and then add different images.
3. Don't look at the overall task list. Just take one task at a time and do it. Once it's done, pick the next task. If you get stuck on a particular task, move on to the next task.
4. Stay calm, think before coding, and work sequentially. You will make it.
5. Be strategic about the electricity issue.
6. Use ChatGPT to generate JSON data. You can use ChatGPT for other purposes as well.
7. Initially, you can add data to the MongoDB. Then you can use the data from the fetch call. Other collections will depend on you.

No Pain, No Gain

"The most beautiful moments in life come after going through hardships and challenges."

What to submit :

- Your assignment ID/Variant
- Your client-side code GitHub repository
- Your server-side code GitHub repository
- Your live website link