Exploro

Internship Assignment: Travel Website Development

Overview

As part of the selection process for the internship, you are required to create a fully functional travel website. This assignment is designed to evaluate your technical skills, creativity, and ability to implement real-world requirements into a software solution. Please read the assignment details carefully and adhere to the requirements.

Objective

Build a travel website where users can browse, book, and manage trips. Additionally, implement functionality for trip organizers to register and add trips. The assignment will involve creating both frontend and backend components with the specified tech stack.

Requirements

Core Functionalities

1. **Trip Listing and Details**:
    - Display a list of upcoming trips on the landing page.
    - Each trip should have a detailed view containing information such as:
        - Trip name
        - Description
        - Dates
        - Price
        - Available slots
        - Cancellation policy
2. **Trip Booking**:
    - Allow users to add trips to a cart.
    - Implement a checkout process to confirm booking.
    - Ensure that only authenticated users can book trips.
3. **Authentication**:
    - Unauthenticated users can browse trips and view details.
    - To book a trip, users must log in or sign up.
4. **Trip Organizer Registration**:
    - Provide a registration flow for trip organizers.
    - Registered organizers should have access to a dashboard to:
        - Add new trips
        - View their added trips
        - Edit or delete trips

5. **Booking Management and Cancellation**:
   - Authenticated users should have a section to view all their booked trips.
   - Implement cancellation functionality with the following policies:
     - Full refund if cancelled 15 days prior to the trip date.
     - 50% refund if cancelled 7 days prior.
     - No refund if cancelled less than 7 days prior.
6. **Payment Handling**:
   - Implement a dummy payment system to store payment details.
   - Code should be modular and extensible for integrating third-party payment gateways in the future.
7. **Website Layout**:
   - **Landing Page**:
     - Include information about the travel company.
     - List upcoming trips with an option to view details or add to cart.
   - **User Dashboard**:
     - For customers to view and manage their bookings.
     - For organizers to manage their trips.

Technical Stack

- **Frontend**:
  - JavaScript, CSS, React
- **Backend**:
  - Any technology of your choice (Node.js, Java, Python, etc.)

Good-to-Have Features (Optional)

1. **Concurrency Handling**:
   - If a trip has only one slot left, ensure that simultaneous booking attempts by different users result in only one successful booking.
2. **Session Management**:
   - Implement login sessions that expire after a specified duration (e.g., 7 days), requiring users to log in again.
3. **Optimized Load Time**:
   - Ensure that the landing page loads quickly by optimizing assets (images, CSS, JavaScript) and implementing lazy loading for content.

Deliverables

- **Frontend**:
  - Fully functional React-based user interface.
  - Responsive design that works on both desktop and mobile devices.
- **Backend**:

  - APIs for handling user authentication, trip data, bookings, and organizer functionalities.
  - Database schema for storing trips, users, bookings, and payment details.
- **Documentation**:
  - Include a README file with:
    - Setup instructions for the frontend and backend.
    - Explanation of your approach and any assumptions made.
    - Instructions on running the application.
  - API documentation for all implemented endpoints.

Evaluation Criteria

- **Functionality**:
  - How well the core functionalities are implemented.
  - Adherence to the requirements.
- **Code Quality**:
  - Readability, modularity, and adherence to best practices.
- **UI/UX Design**:
  - User-friendly and visually appealing design. Visual appearance can be ignored a bit if the core functionality is working as expected.
- **Scalability**:
  - Extensibility of the code for future features like payment gateway integration.
- **Optional Features**:
  - Bonus points for implementing concurrency handling, session management, or load-time optimization.

Submission Guidelines

1. Submit your assignment as a GitHub repository.
2. The repository should include:
   - Separate folders for frontend and backend code.
   - A SQL/NoSQL script or schema for the database.
   - Clear documentation.
3. Ensure the repository is private and share access with the provided email address.
4. Deadline: **22 Dec 2024**.

Important Notes

- This assignment is designed to test your problem-solving and development skills. Plagiarism or using pre-built solutions will result in disqualification.
- While ChatGPT or similar tools can be used for guidance, the implementation must be your own.

- Feel free to reach out for clarifications regarding the assignment requirements.

Good luck, and we look forward to your submissions!