Reachify
Step into The Future of Marketing
info@reachifyinnovations.com
www.reachifyinnovations.com

Assignment

Thank you for your interest in the Full Stack Intern position at Reachify! This role requires proficiency across frontend and backend development, as well as experience with API integration. Below is a structured assignment that will help us assess your skills in building a full-stack application, with a focus on practical integration of the frontend and backend.

Please Note: Originality is crucial. If any part of this assignment appears copied from external sources or AI-generated without unique input, it may result in disqualification.

Full Stack Intern Assignment – Core Full Stack Development

Objective:

This assignment will assess your ability to develop a simple full-stack application using React for the frontend and FastAPI (or Django) for the backend, implementing basic CRUD functionality and integrating the two. You'll also create a minimal API testing plan and demonstrate your familiarity with Git for version control.

Assignment Instructions:

>> Section 1: Frontend Development with React

1. Create a Basic React App
   - Set up a React application that displays a list of items fetched from a backend API.
   - Use React Hooks (e.g., 'useState ' and 'useEffect ') to manage the fetching and display of data.
   - Ensure the list updates in real-time when data changes.
2. Component Design
   - Develop a reusable component that displays individual items from the list with custom styling using either CSS Modules or styled-components .
   - Implement a "Delete" button in each component to remove items from the list.
3. State Management and API Error Handling
   - Implement error handling for the API request, showing an error message if the fetch request fails.
   - Add a "loading" indicator while data is being fetched.
   - Extra: Use React Context API to manage and share state globally across components.

>> Section 2: Backend Development with Python

1. API Development with FastAPI or Django
   - Create a simple API with Python that provides a list of items (like a list of products or users).
   - Ensure the API has the following endpoints:
     - 'GET /items ': to retrieve all items.
     - 'POST /items ': to add a new item.
     - 'DELETE /items/{id} ': to delete an item by ID.

- Implement input validation and error handling for each endpoint.

2. Authentication & Security Basics
   - Add basic token-based authentication (e.g., JWT) to protect API endpoints.
   - Ensure only authenticated users can add or delete items.
3. Connecting Frontend to Backend
   - Connect the React frontend application developed in Section 1 to this API.
   - Fetch data from the API to display it in the frontend, and ensure that the "Delete" and "Add" functionality works properly.

>> **Section 3: Deployment on Docker/Microsoft Azure**

2. Continuous Integration/Continuous Deployment (CI/CD)
   - Set up a CI/CD pipeline using GitHub Actions or Azure Pipelines to automatically deploy the application on commits to the main branch.
3. Containerization with Docker
   - Create a Dockerfile for both the React frontend and Python backend, making sure each service runs independently in its own container.

Submission Format:

Please provide the GitHub repository link containing both the frontend and backend code. Name the repository as **“[Your Name] - Reachify Full Stack Assignment”** and ensure that each component has a clear setup guide.

Additional Request: Best Project links

Please include links of top 3 best projects of yours that demonstrate your frontend or backend work. You can include this in the README or as a separate file in the repository.

Evaluation Breakdown:

- **Frontend Development with React (40%)**
- **Backend Development with Python (40%)**
- **Deployment (20%)**

Thank you for taking the time to complete this assignment. We look forward to seeing your skills in full-stack development!