

Project Title: Youth Professional Platform

Description:

This platform connects youth with job opportunities, training, and professional networking. It also offers an admin interface to manage users, jobs, and training resources.

Functional Requirements

1. User Side (Frontend)

1. Homepage:

- a. A landing page explaining the platform's purpose and benefits.
- b. Highlights latest job postings, training courses, and active users.
- c. Includes "Sign Up," "Log In," and "Search Jobs" buttons.

2. User Registration & Login:

- a. Registration form with user type (individual or company).
- b. Collects user details:
 - i. For Individuals: Name, email, phone, skills, education, and CV upload.
 - ii. For Companies: Business name, address, contact details, and job offerings.

3. User Dashboard:

a. Job Opportunities:

- i. Search, filter, and apply for jobs.

b. Networking:

- i. Search for other users by skill or industry and send connection requests.

c. Learning Section:

- i. Access training courses (videos, documents).

d. Tokens & Rewards:

- i. View earned tokens (if blockchain is integrated).

4. Search and Filters:

- a. Advanced filtering options for jobs and courses (e.g., by skill, location, industry).

2. Admin Side (Backend)

1. Admin Dashboard:

- a. Overview of platform usage:
 - i. Total users, active users, jobs posted, and courses accessed.

2. Manage Jobs:

- a. Review, approve, or reject job postings.

3. Manage Users:

- a. View user profiles and deactivate accounts violating terms.

4. Analytics & Reports:

- a. Generate reports on user activity, popular skills, and job trends.

5. Manage Learning Resources:

- a. Add/edit training materials (videos, PDFs, etc.).

Non-Functional Requirements

1. Performance:

- a. Must handle up to 10,000 active users simultaneously.

2. Scalability:

- a. Should allow easy expansion for additional features or users.

3. Security:

- a. Use JWT authentication for secure logins.
- b. Data encryption for user credentials.

4. Responsiveness:

- a. Works seamlessly on mobile, tablet, and desktop devices.

Technology Stack

1. Frontend:

- a. **ReactJS** for a dynamic and user-friendly interface.

2. Backend:

- a. **Django/DRF** for API management and data processing.

3. Database:

- a. **PostgreSQL** to manage user profiles, jobs, and courses.

4. Hosting:

- a. **AWS** or **Vercel** for reliable and scalable deployment.

5. Blockchain Integration (Optional):

- a. **Ethereum** or **Binance Smart Chain** for tokenization.

Flow Diagram Components

1. User Workflow:

- a. Homepage → Sign Up/Login → User Dashboard → Search Jobs, Networking, Learning, or Token Management.

2. Admin Workflow:

- a. Admin Login → Admin Dashboard → Manage Jobs, Users, Reports, or Learning Resources.

Deliverables

1. Prototype Design:

- a. Wireframes for both user and admin interfaces.

2. Core Features Development:

- a. User registration, job search, and networking.
- b. Admin job management and user analytics.

3. Deployment:

- a. Fully functional platform hosted online.