

## Overview

As part of this assessment, you will build a **Job & Networking Portal** — a minimal full-stack web application inspired by **LinkedIn, Upwork, and AngelList**, enhanced with:

- User profiles, job listings, and social feed
- AI-powered features like job matching and skill extraction
- Web3 wallet connection & blockchain payment integration
- Optional on-chain logging or smart contract functionality

This assignment is your chance to showcase **full-stack, Web3, AI, and product thinking skills** to qualify for the **RizeOS Core Team Internship**.

## Tech Stack Flexibility

Layer	Options
Frontend	React.js, Tailwind CSS
Backend	Node.js, Express.js
Database	MongoDB or PostgreSQL
Blockchain	✓ Choose One: <b>Solana</b> / <b>Ethereum</b> / <b>Polygon</b>
Wallet	✓ Choose One: <b>Phantom (for Solana)</b> / <b>MetaMask (for EVM chains)</b>
Smart Contracts	(Optional) Solana Program in Rust / Solidity Smart Contract
AI / ML	NLP-based logic for skill parsing, similarity matching, etc.

**You are encouraged to choose the stack and features that reflect your strengths and help create a better user-centric product.**

## Tasks

### Task 1: Full-Stack Product Development

#### Module 1: Authentication & Profile Management

- User registration/login (JWT/session)
- Create/edit profile with:
  - Name, bio, LinkedIn URL
  - Manual or AI-extracted **skills**
  - **Public wallet address** (from MetaMask/Phantom)

## Module 2: Job Posting + Feed

- Authenticated users can:
  - Post jobs (title, description, skills, budget/salary)
  - View job listings and user posts (e.g., career advice, updates)
  - Filter by skill, location, and tags
- Store data in the backend securely

## Module 3: Blockchain Payment Integration

- Before posting a job:
  - Connect their **wallet (MetaMask or Phantom)**
  - Pay platform fee (e.g., **0.01 SOL or 0.001 ETH/MATIC**) to the **admin wallet**
- Use Web3.js, Ethers.js, or Solana Web3.js to:
  - Initiate & sign the transaction
  - Confirm it on the blockchain
  - Enable post submission only after success
- **Optional:** Log payment on backend and/or via smart contract on-chain

## Module 4: AI Enhancements *(Pick any or all)*

Feature	Description
Job ↔ Applicant Matching	NLP model to show "match score" based on job description and candidate bio
Resume Skill Extraction	Parse the uploaded resume or bio to auto-fill the top skills
Smart Suggestions	Recommended jobs or connections based on profile

Feel free to experiment with other AI features that can make the user journey smoother or smarter, and help your product stand out among other candidates.

## Task 2: Go-To-Market (GTM) & Monetization Strategy

Prepare a **detailed GTM strategy** that covers:

- **Target user base** and user personas
- **3-month roadmap** to reach the first **10,000 users**
- **Marketing** plan with only **₹5,000 budget**
- **Revenue Streams** (minimum 2 – e.g., ₹150/month subscriptions, token boosts, commissions)

## Deliverable: 15–20 Minute Demo Video Link (Mandatory)

This should include:

1. **Introduction** — About you and your thought process
2. **Tech Walkthrough**
  - Frontend, backend, AI, and Web3 code explained
  - DB, APIs, wallet integration, smart contract (if done)
3. **AI & Unique Feature Demo**
  - Show how your AI / UX feature adds value
4. **UI/UX Explanation** — What you prioritized and why
5. **GTM & Revenue Plan**
  - Monetization model, conversion plan, scalability
6. **Roadmap & Fundraising (Optional)**
  - Vision beyond MVP and how you'd raise funds

## Final Deliverables

Item	Description
Deployed App	Frontend/backend on <b>Vercel / Netlify / Render</b>
GitHub Repo	Clean code with a proper README and run instructions
AI/ML Logic	Code snippets or an explanation of integration
Wallet Integration	MetaMask/Phantom connection + verified testnet transaction
Admin Wallet	Your devnet/testnet admin wallet for testing payments
Demo Video	<b>15–20 min</b> , clear explanations (Google Drive link)
(Optional) Smart Contract	Source code + deployed testnet address

## Evaluation Criteria

Category	Weight
Core Functionality & APIs	30%
Web3 Integration	20%
AI Features (Quality)	20%
UI/UX Design	10%

Code Quality	10%
Documentation & Demo	10%

## Timeline

- **Total Duration:** 3–4 days from the day you receive this assignment

## Final Notes

This assignment reflects **real-world product building** — from development to GTM strategy.

Go beyond the basic functionality if you wish to **stand out**:

- Show ownership, problem-solving, and product-thinking
- Deliver a clean, modular, scalable version with attention to **user experience**

We're excited to see what you build!