

# Moveo Coding Task - AI Crypto advisor

## Concept -

Build a web app that serves as a personalized crypto investor dashboard. The app gets to know the user through a short onboarding quiz and then shows daily AI-curated content tailored to their interests. Users can give feedback (thumbs up/down) to help improve future recommendations. A fun meme is also shown dynamically each time the dashboard updates.

## Pages & Features

### 1. Login/Signup

- Register with email, name, and password.
- Login page with basic authentication (JWT or session-based).

### 2. Onboarding (after first login) - Ask a few short questions:

- What crypto assets are you interested in?
  - What type of investor are you? (e.g. HODLer, Day Trader, NFT Collector)
  - What kind of content would you like to see? (e.g. Market News, Charts, Social, Fun)
- Save answers in the DB as user preferences.

### 3. Daily Dashboard

- Display 4 sections based on user preferences:
  - Market News (CryptoPanic API or static fallback)
  - Coin Prices (CoinGecko API)
  - AI Insight of the Day (free LLM like OpenRouter or Hugging Face)
  - Fun Crypto Meme
- Each section includes voting, stored in DB for future model improvements.

## Technical Guidelines

- Frontend: React/Angular
- Backend: Any language/framework
- Database: SQLite, PostgreSQL, or MongoDB
- Use only free public APIs and free-tier AI tools
- Focus on clean UX, readable code, and good structure

Suggested Free APIs:

- Coin data: CoinGecko API
- News: CryptoPanic
- AI: OpenRouter, Hugging Face Inference API
- Memes: Reddit scraping or static JSON

## Deployment

Your project must be deployed and publicly accessible. Suggested free services:

- Frontend: Vercel or Netlify
- Backend + DB: Render, Railway, or Glitch

## Deliverables

- Public GitHub repository
- Deployed app URL
- If you use any AI tools (e.g., ChatGPT, Cursor, GitHub Copilot, etc.) during this assignment, please include a summary of your interactions. This will help us better understand your thinking process and how you collaborate with such tools.
- Access to DB

- **Bonus** - Add suggestion about the process of training, related to this dashboard app, how feedback is stored for future model improvements (no need to implement)

**Good luck!**