Project Overview: Cryptocurrency Watchlist

Project Goals

- Create a user-friendly cryptocurrency watchlist that allows users to track real-time prices.
- Enable users to add cryptocurrencies into their own watchlist.
- Implement price alert systems that notify users when a cryptocurrency reaches a specified price.
- Ensure seamless API integration for live price updates.
- Provide a responsive and minimalistic design for both desktop and mobile users.

Target Customers

- Beginner to intermediate cryptocurrency investors who need a simple tracking tool.
- Students and tech enthusiasts interested in learning about cryptocurrency price trends.
- Individuals who want a lightweight alternative to complex trading platforms.

Customer Needs Addressed

- A straightforward platform to monitor cryptocurrency movements.
- A simple way to set and receive price alerts without complex trading tools.
- A clutter-free, easy-to-navigate interface focused on watchlist functionality.

Competition Analysis

- Competitors:
  - CoinMarketCap
  - CoinGecko
  - Binance
- Limitations of the competitors:
  - Most of these platforms include overwhelming data, making it difficult for casual users to focus on simple price tracking.
- Unique Selling Point:
  - A distraction-free interface focused only on watchlist and price alerts, without the complexities of advanced trading charts.

Primary Differentiation

- Minimalist UI tailored for users who only want to monitor and set alerts, without additional features like portfolio tracking or in-depth analytics.
- Open-source and customizable for users who may want to expand capabilities.

Vision Statement

- "A sleek and intuitive crypto watchlist that makes tracking and alerts effortless."

Sprint Planning

- **Sprint 1: Core Watchlist Implementation**
- **Goal:** Establish project structure and watchlist functionality.
    - o   Set up Django and Database.
    - o   Integrate cryptocurrency API for real-time tracking.
    - o   Allow users to add and remove cryptocurrencies from their watchlists.
- **Sprint 2: User Authentication & UI Desing**
- **Goal:** Implement Authentication and refine UI.
    - o   Develop CRUD Operations.
    - o   Develop sing-up, login, and logout features.
    - o   Desing a responsive, user-friendly interface.
    - o   Ensure mobile compatibility.
- **Sprint 3: Pricer Alerts & Notifications**
- **Goal:** Implement price alert system.
    - o   Allow users to set price targets.
    - o   Develop background processes to check prices and trigger price alerts.
    - o   Display notifications for met conditions.
- **Sprint 4: Testing, Optimization & Deployment**
- **Goal:** Finalize and deploy the application.
    - o   Conduct testing and debugging.
    - o   Optimize database and API performance.
    - o   Deploy using PythonAnywhere.
    - o   Gather user feedback for future improvements.