**Frontend Development with React.js**

**Project Documentation for CRYPTOVERSE - Cryptocurrency Dashboard**

**1. Introduction**

* **Project Title:** CRYPTOVERSE
* **Team Members:**
* Syed Mohammed Mohaideen A – (Team Leader)
* Adithya J
* Kamarajan B
* Bala B

**2. Project Overview**

* **Purpose:**  
  Cryptoverse is a cryptocurrency dashboard that provides historical price analysis over five years. It enables investors to track price trends, analyze volatility, and make informed investment decisions.
* **Features:**
  + Historical price data visualization using charts.
  + Search and filter cryptocurrencies.
  + Interactive UI with seamless navigation.
  + Cryptocurrency details page with statistics and links.
  + Redux Toolkit for state management.

**3. Architecture**

* **Component Structure:**
  + Home.js: Displays global crypto stats and top 10 cryptocurrencies.
  + Cryptocurrencies.js: Lists all cryptocurrencies with a search feature.
  + CryptoDetails.js: Shows cryptocurrency details, price trends, and statistics.
  + LineChart.js: Visualizes historical price trends.
  + Navbar.js: Handles navigation.
* **State Management:**
  + Uses **Redux Toolkit** for global state management.
  + API requests are managed through a Redux API slice (cryptoApi).
* **Routing:**
  + **react-router-dom** is used for client-side routing.
  + Routes:
    - /: Home Page
    - /cryptocurrencies: Cryptocurrency List
    - /crypto/:coinId: Cryptocurrency Details

**4. Setup Instructions**

* **Prerequisites:**
  + **Node.js & npm:** Install from [Node.js Official Site](https://nodejs.org/en/download/)
  + **Git:** Install from [Git Official Site](https://git-scm.com/downloads)
  + **React.js:** Installed via npx create-react-app
* **Installation:**

1. Clone the repository:

* + git clone <https://github.com/SyEDSmM/KRYPTOS.git>

2. Navigate to the project directory:

* cd cryptoverse

3. Install dependencies:

* npm install

4. Start the development server:

* npm start # or npm run dev (if using Vite)

5. Open the app in a browser:

* <http://localhost:5173/>

**5. Folder Structure**

cryptoverse/

│── src/

│ ├── components/ # Reusable components (Navbar, LineChart, Loader)

│ ├── pages/ # Page components (Home, Cryptocurrencies, CryptoDetails)

│ ├── services/ # API calls using Redux Toolkit

│ ├── assets/ # Images and static resources

│ ├── App.js # Main application file

│ ├── index.js # Entry point of React App

│ └── store.js # Redux store setup

│── public/

│── package.json

│── README.md

* **Utilities:**
  + services/cryptoApi.js: Manages API calls using Redux Toolkit.
  + utils/: Contains helper functions.

**6. Running the Application**

* **Frontend:**
* npm start # In the client directory
* or (if using Vite):
* npm run dev

**7. Component Documentation**

* **Key Components:**
  + Home.js: Displays global cryptocurrency statistics and trending coins.
  + Cryptocurrencies.js: Lists cryptocurrencies with a search feature.
  + CryptoDetails.js: Shows individual cryptocurrency price trends.
  + LineChart.js: Renders interactive price trend charts.
  + Navbar.js: Navigation component.
* **Reusable Components:**
  + LineChart.js: Displays price trends using **react-chartjs-2**.
  + Loader.js: Displays a loading spinner.

**8. State Management**

* **Global State:**
  + Managed using **Redux Toolkit**.
  + **cryptoApi** slice handles API requests and caching.
* **Local State:**
  + Uses useState and useEffect for component-specific logic.
  + Example: CryptoDetails.js manages the selected time period for historical data.

**9. User Interface**

* **Screenshots/Demo Links:**
  + **Homepage:** Displays global crypto stats and trending cryptocurrencies.
  + **Cryptocurrency List:** Shows all available cryptocurrencies.
  + **Crypto Details Page:** Displays charts, statistics, and useful links.

📌 [Demo Video](https://drive.google.com/file/d/1n8VS9pRXdYQjXQ4amb65U0zN4DJsdnz3/view?usp=drivesdk)  
📌 [Execution Video](https://drive.google.com/file/d/1oebbU-0uiulxV-qLdli-VC_Db_A6q-E-/view?usp=drivesdk)

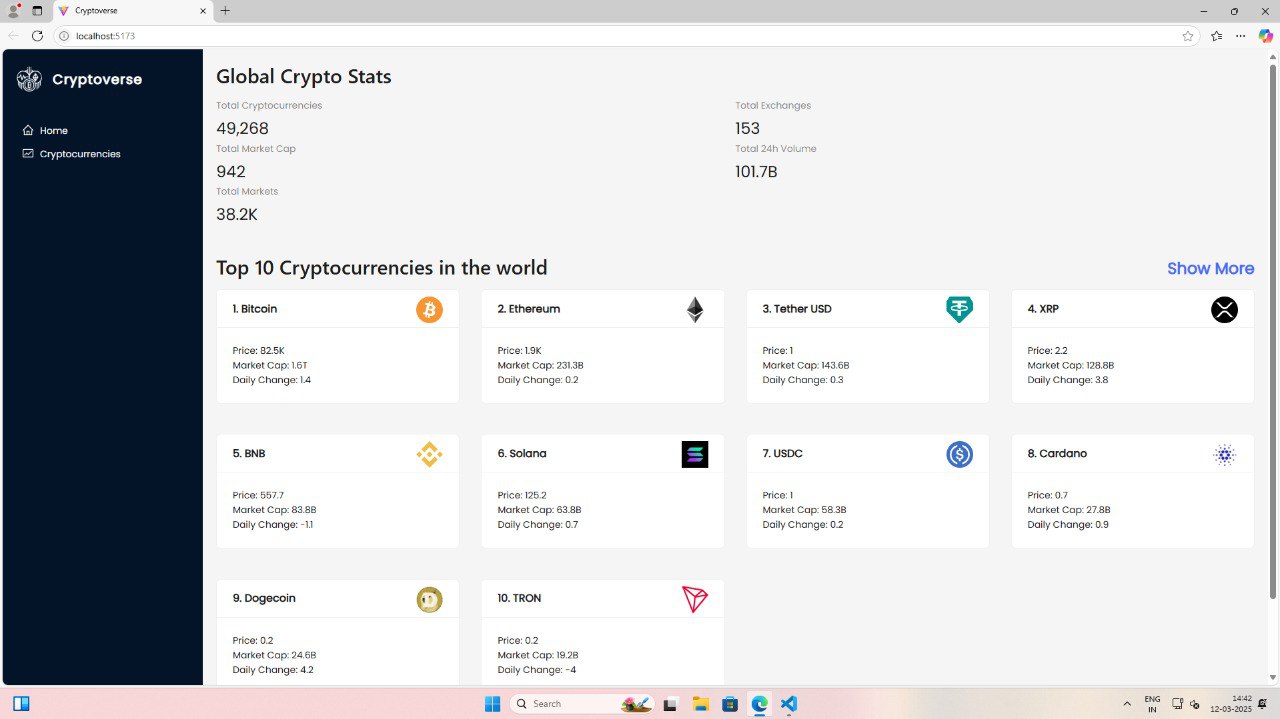
**10. Styling**

* **CSS Frameworks/Libraries:**
  + **Ant Design (antd)**: Provides UI components.
  + **Styled-Components**: Used for styling individual components.
* **Theming:**
  + Dark/light mode can be implemented in future updates.

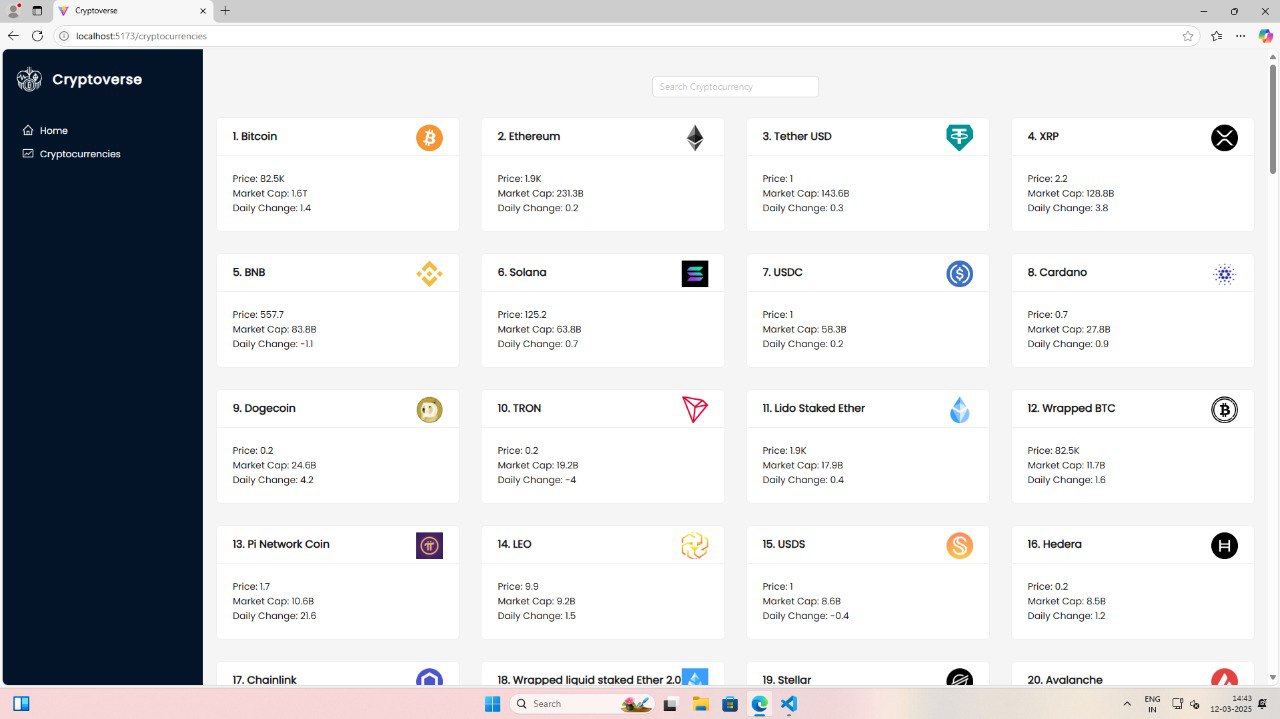
**11. Testing**

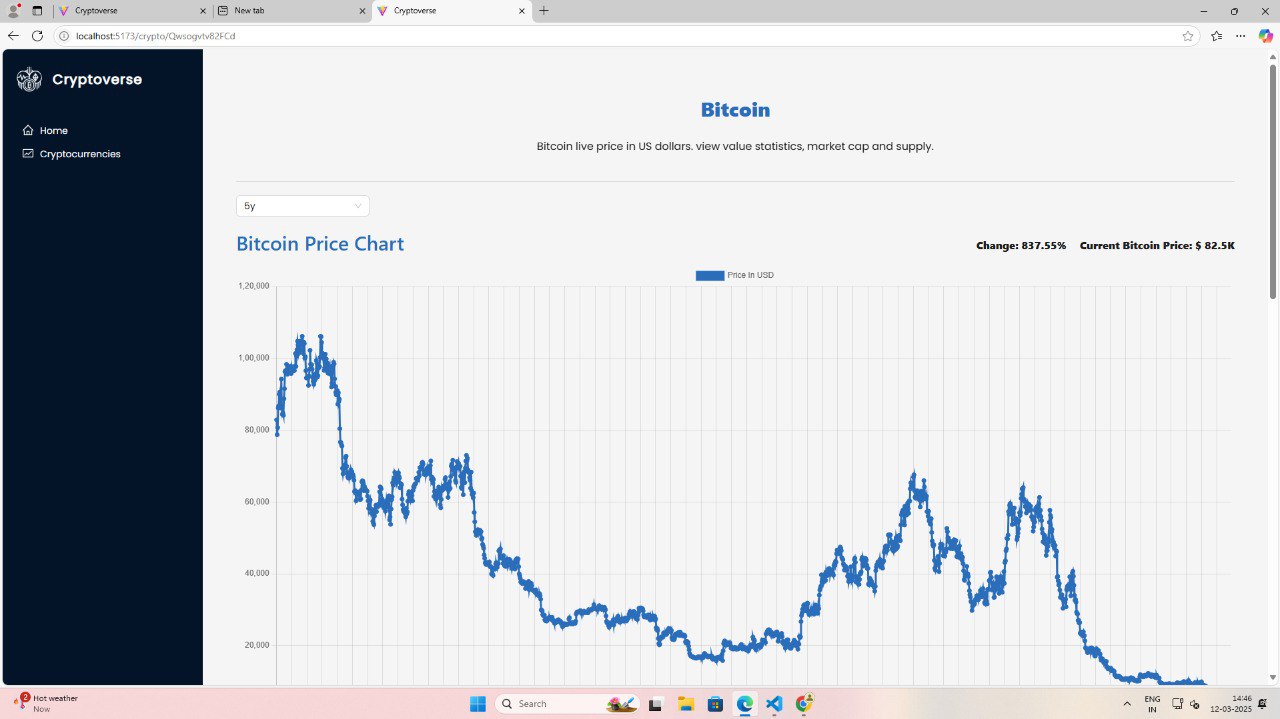
* **Testing Strategy:**
  + Unit testing with **Jest**.
  + Component testing with **React Testing Library**.
  + End-to-end testing with **Cypress** (future enhancement).
* **Code Coverage:**
  + Use jest --coverage for coverage reports.

**12. Screenshots or Demo**

* **Demo Link:**  
  📌 [Cryptoverse Demo](https://drive.google.com/file/d/1n8VS9pRXdYQjXQ4amb65U0zN4DJsdnz3/view?usp=drivesdk)
* **Screenshots:**  
  📌 [Homepage](https://drive.google.com/file/d/1oiRM4Afr1271hxJlnM_zA65H-BosN1vG/view?usp=drivesdk)

📌 [Cryptocurrency List](https://drive.google.com/file/d/1opGUT3Yf2VT_Wwd7SecrHIPjhrDoo2pJ/view?usp=drivesdk)



📌 [Crypto Details Page](https://drive.google.com/file/d/1yFMpLxJh4-_G7ik_lx7jmIVWJeeVp3zJ/view?usp=sharing)

**13. Known Issues**

* No documented bugs in the existing system.

**14. Future Enhancements**

* **Real-time Price Updates:**
  + Use WebSockets to fetch live data.
* **Additional Filtering Options:**
  + Users can filter by market cap, volume, and performance.
* **Enhanced UI/UX:**
  + Dark mode and custom themes.
* **Expand Historical Data:**
  + Include more timeframes for better analysis.