Frontend Development with React.js

Project Documentation

**Introduction**

**Project Title:** Cryptocurrency Dashboard

**Team Members:**

**Team Leader : KATHIRVEL M**

**Gmail: mahaliagamm@gmail.com**

**Fronted end developer: RAJESHKUMAR R**

**Gmail : rk3108038@gmail.com**

**Back end developer : HEMATHRI E**

**Gmail : hemathri28@gmail.com**

**Ui and UX : BALAJI K**

**Gmail : 143santhoshsandy143@gmail.com**

**Project Overview**

**Purpose:** The Cryptocurrency Dashboard is a React-based web application that provides real-time cryptocurrency market data, trends, and price details.

**Features:**

* Display a list of top cryptocurrencies.
* View detailed information about each cryptocurrency.
* Interactive line charts for market trends.
* Search functionality for cryptocurrencies.
* Responsive design for various screen sizes.

**Architecture**

**Component Structure:**

* Navbar.jsx - Navigation bar.
* Home.jsx - Home page.
* Cryptocurrencies.jsx - Displays list of cryptocurrencies.
* CryptoDetails.jsx - Detailed information on selected cryptocurrency.
* LineChart.jsx - Graph visualization of trends.
* Loader.jsx - Loading animation component.

**State Management:**

* Uses **Redux Toolkit** (store.js) to manage global state.
* cryptoApi.js handles API requests.

**Routing:**

* Uses react-router-dom for navigation between pages.

**Setup Instructions**

**Prerequisites:**

* Node.js (Latest LTS version recommended)
* NPM or Yarn

**Installation:**

1. Clone the repository: git clone [repository URL]
2. Navigate to the project folder: cd project-directory
3. Install dependencies: npm install
4. Start the development server: npm run dev

**Folder Structure**

* **Client:** React frontend with components, pages, and assets.
* **Services:** API handling with cryptoApi.js.
* **Store:** Redux store configuration in store.js.

**Running the Application**

* Start the frontend server: npm run dev

**Component Documentation**

**Key Components:**

* Cryptocurrencies.jsx: Fetches and displays a list of cryptocurrencies.
* CryptoDetails.jsx: Fetches and displays detailed information on a selected cryptocurrency.
* LineChart.jsx: Renders graphical market trends.

**Reusable Components:**

* Loader.jsx: Displays a loading animation.
* Navbar.jsx: Persistent navigation bar across pages.

**State Management**

**Global State:**

* Redux Toolkit handles API state and caching.

**Local State:**

* Managed via React's useState and useEffect hooks.

**User Interface**

* **Cryptocurrencies List Page**
* **Crypto Details Page with Line Charts**
* **Navigation Bar for Routing**

**Styling**

* Uses **CSS Modules** for component-specific styling.
* Implements responsive design using Flexbox and Grid.

**Testing**

**Testing Strategy:**

* Uses Jest and React Testing Library for unit tests.

**Code Coverage:**

* Ensures component behavior through snapshot and unit tests.

**Future Enhancements**

* Dark mode support.
* User authentication.
* Additional financial indicators in charts.

**Screenshots or link to a deployed demo**

https://drive.google.com/file/d/1d8mcMIbk67LU7dDYE9ha0jii4V\_PoU\_h/view?usp=drive\_link













