**Frontend Development with React.js**

**Project Documentation format**

* **Introduction**
* **Project Title**: Cryptoverse: A Cryptocurrency Dashboard.
* **Team Members**: Team Size : 5
* Team Leader : MOHANA PRIYA T Y

Emailid: yoganandhan.t@gmail.com

* Team member : GAYATHRI K

Emailid: gayee1222@gmail.com

* Team member : SUJITHA R

Emailid: r.sujitha3724@gmail.com

* Team member : SABITHA R

Emailid: ramsabitha03@gmail.com

* Team member : SANDHIYA K

Emailid: ksandhiya101@gmail.com

* **Project Overview**
* **Purpose**: CrytoVerse is a cryptocurrency dashboard built using React.js, HTML, and CSS. It provides real-time data, analytics, and insights into various cryptocurrencies. The platform aims to help users track market trends, view live price updates, and analyze 5 years historical data efficiently and represented through linechart.
* **Features**:
* Live cryptocurrency price tracking
* Market cap and volume analysis
* Interactive charts and graphs
* Search and filter functionality
* User-friendly interface
* **Architecture**
* **Component Structure**: The application follows a modular React component structure with reusable components for charts and search filters.
* **State Management:**CrytoVerse uses Rapidapi for global state management, ensuring efficient data handling and reactivity.
* **Routing**: The application is structured with react-router-dom for smooth navigation between pages like Home, Market Trends, and Coin Details..
* **Setup Instructions**
* **Prerequisites**: Ensure you have the following installed:
* Node.js (Latest LTS version)
* npm or yarn
* **Installation**: 1. Clone the repository:

git clone <https://github.com/SSC369/cryptoverse>

cd cryptoverse-dashboard

* 2. Install dependencies:

npm install

* 3. Start the development server:

npm start.

* **Folder Structure**
* **Client**:
* /src – Main source directory
* /components – Reusable UI components
* /services – Main application Services
* /assets – Images, icons, and styles
* /cryptoapi– Helper functions and API handlers
* **CryptoAPI**:
* API calls are handled through cryptoapi.js
* Custom hooks for fetching data and managing state
* **Running the Application**
* To start the React app locally:

npm start

* **Component Documentation**
* **Key Components**:
* Header: Navigation bar with search functionality.
* Dashboard: Displays live cryptocurrency data.
* CoinDetails: Detailed information about a selected cryptocurrency.
* **Reusable Components**:Chart.js Graphs – Used for visualizing trends.
* **State Management**
* **Global State** Managed using Rapidapi to store and update cryptocurrency data efficiently..
* **Local State**: Handled within individual components using useState for UI interactions.
* **User Interface**
* The dashboard has an intuitive design with interactive charts, tables, and a responsive layout..
* **Styling**
* **CSS Frameworks/Libraries**:
* CSS Frameworks/Libraries:
* CSS / React.js for responsive design.
* Styled Components for dynamic styling.
* **Testing**
* **Testing Strategy**:
* Unit Testing with Jest
* Integration Testing with React Testing Library
* **Code Coverage**: Ensures optimal test coverage using jest-coverage.
* **Screenshots or Demo**
* We attached Demo link video.
* **Known Issues**
* API rate limits may cause temporary delays in data fetching
* Performance optimization needed for large data sets.
* **Future Enhancements**
* Add user authentication for watchlists.
* Implement real-time websocket updates.
* Integrate AI-based price predictions.