**Frontend Development with React.js**

**Project Documentation for Cryptoverse:A cryptocurrency dashboard**

**1.Introduction**

* **Project Title**: **CRYPTOVERSE**
* **Team Members**:

Team Leader : ABINAYASHREE V K [Email id:vkabinayashree@gmail.com]

Team member : AROKYA JEBILA M [Email id:jebilamarokya@gmail.com]

Team member : ANEES FATHIMA S [Email id:saneesfathima82@gmail.com]

Team member : ABINAYA A [Email id:abinayaarumugam.cs@gmail.com]

**2.Project Overview**

1. **Purpose**: Cryptoverse is a sophisticated cryptocurrency dashboard designed to provide investors with comprehensive insights into market dynamics through detailed historical price data analysis . The platform empowers users to identify top-performing assets and make informed investment decisions..
2. **Features**:
   * Search Feature
   * Real-time price updates
   * Price Fluctuation Visualization
   * Market cap, price change, and volume data display.
   * Interactive Charts
   * Visual Currency Browsing
   * Insights and Decision-Making

**3.Architecture**

* **Component Structure**:

The application is built using React.js with a component-based architecture. Major components include:

* **Home page :** This pages consists of stats of global crypto like total cryptocurrencies,total exchanges, market cap etc.
* **Crypto currencies page** : This pages contains all cryptocurrencies which are

currently in flow in the world.

* **search feature** where users can search and find out about their desired cryptocurrency.
* **Crypto currency details page** : This page contains the line chart with data representation of price of cryptocurrencies. Also contains statistics and website links of cryptocurrencies.
* **State Management**:
* The application uses Redux for defining endpoints for fetching cryptocurrencies, cryptocurrency details,and cryptocurrency history. It is configured with base URL, headers, and queryfunctions required for making requests to the cryptocurrency API.
* **Routing**: The application uses React Router for navigation. Routes include:
* /: Home page
* /: Cryptocurrency page
* /Crypto currency/: Search page
* /:Cryptocurrency details page
* **Setup Instructions**
  + **Prerequisites**:
    - Node.js (v20 or higher)
    - npm (v8 or higher)
    - Git
  + **Installation**:

**1**. Clone the repository: git clone https://github.com/SSC369/cryptoverse

2. Navigate to the client directory:cd crypto

3. Install dependencies: npm install

4. Configure environment variables: Create a .env file in the client directory and add the necessary variables (e.g., API keys).

5. Start the development server: npm start

**5.Folder Structure**

* **Client**:
  + src/components: # Reusable components (homepage , Cryptocurrency page, Cryptocurrency details pageetc.)
  + src/assets: # Images, icons, and other static files
  + src/service s: #crypto api,main components
  + App.js: # Main application component
  + index.js: # Entry point
* **Utilities**:
  + CryptoApi:It defines endpoints for fetching cryptocurrencies, cryptocurrency details and cryptocurrency history.
  + react-chartjs-2: It defineshistorical data (`coinHistory`), current price (`currentPrice`), and the name of the cryptocurrency (`coinName`) as props, and renders a line chart displaying the historical price data
  + hooks:uses Custom hook

**6.Running the Application**

* **Frontend:**
  + To start the frontend server, run the following command in the client directory:
  + npm start
  + npm install
  + npx json-server ./db/db.json
  + npm run dev
  + The application will be available at http://localhost:3000

**7.Component Documentation**

* **Key Components**:
  + **Header**: Displays the navigation bar
    - Props: onSearch (function to handle search queries).
  + **Home:** display stats of global crypto
    - Props: current global crypto stats(functions to display global crypto)
  + **Cryptocurrencies** : display global cryptocurrency
    - Props: all cryptocurrencies (functions to display Cryptocurrency currently in flow in the world.)
  + **Search bar:** display search bar
    - Props:search feature (functions search feature for users to find out about their desired cryptocurrency.
  + **Crypto currency details page :** displays line chart
    - Props: statistics of Cryptocurrency(functions to display line chart with data representation of price of cryptocurrencies.
* **Reusable Components**:
  + Button: A customizable button component.

▪ Props: text, onClick, disabled.

* + Input: A reusable input field for forms and search.

▪ Props: type, placeholder, value, onChange.

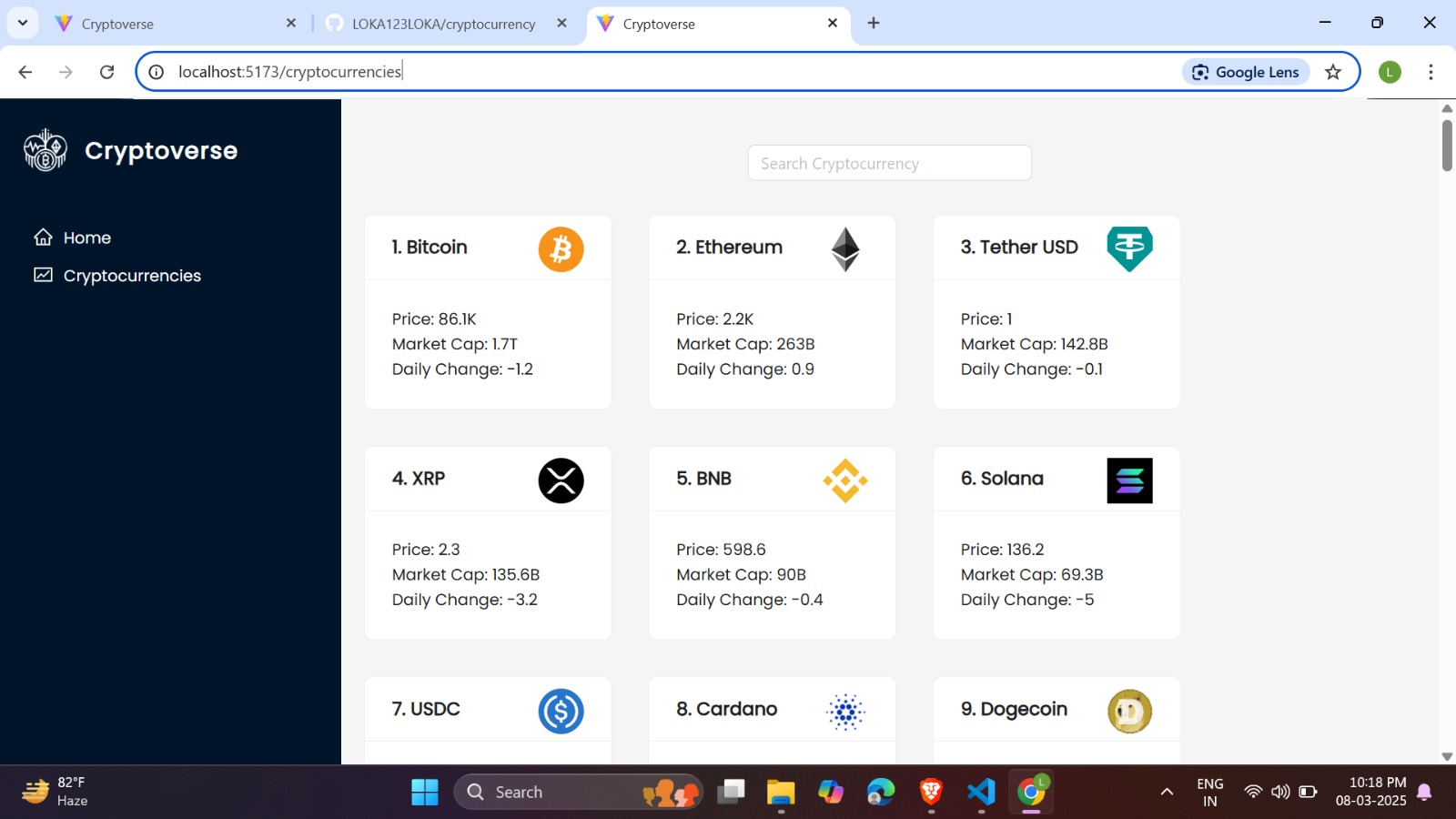
**8.State Management**

* **Global State**:

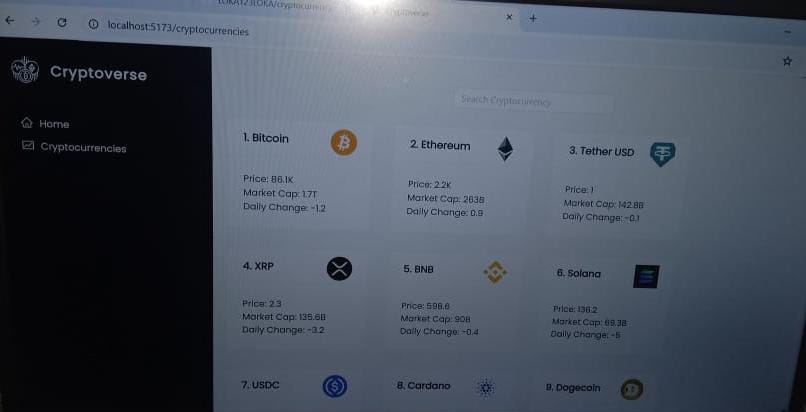
The Redux store manages the following global states:

* + user: Current authenticated user.
  + Home: current global crypto stats
  + Cryptocurrencies: current Cryptocurrency in world with search feature
  + Cryptocurrency details: statistics using line chart
* **Local State**:
  + Local state is managed using React's useState hook within components

**9.User Interface**

* **Screenshots**
  + **Home page** : This pages consists of stats of global crypto like total cryptocurrencies,total exchanges, market cap etc. Also consist of top 10 cryptocurrencies in the world.
  + 
  + **Cryptocurrencies page** : This pages contains all cryptocurrencies which are

currently in flow in the world. There is also a search feature where users can search and find out about their desired cryptocurrency.



* + **Crypto currency details page** : This page contains the line chart with data representation of price of cryptocurrencies. Also contains statistics and website links of cryptocurrencies.

**10.Styling:**

* **CSS Frameworks/Libraries**:

The application uses **Styled-Components** for styling. This allows for modular and scoped CSS within components.

* **Theming**:

A custom theme is implemented using Styled-Components

**11.Testing**

* **Testing Strategy**:

o Unit Testing: Using Jest and React Testing Library.

o Integration Testing: Is performed to ensure that components work together as expected.

o End-to-End Testing: Cypress is used for end-to-end testing of user flows.

* **Code Coverage**:

o Code coverage is monitored using Jest’s built in coverage tool. The current coverage is 85%.

12.**Screenshots or Demo**

* **Demo link: https://drive.google.com/file/d/1f658UmQmUh6VVtl\_wsPO1r4\_Xtz5bek1/view?usp=drivesdk**
* **Screen shot:** See section 9 for UI screenshots.

**13.Known Issues**

* **Issue 1:**lead to appropriate statistics
* **Issue 2:** The search functionality is slow .

**14.Future Enhancements**

* **Future features** 
  + Real-Time Alerts & Notifications
  + AI-Powered Predictive Tools
  + Advanced Analytics & Visualization
  + Improved Security Features

This documentation provides a comprehensive overview of the **cryptoverse** project, including its architecture, setup instructions, and future plans.