Frontend Development with React.js

Project Documentation format.

Cryptoverse: A Cryptocurrency Dashboard

# 1. Introduction

**Project Title:** Cryptoverse: A Cryptocurrency Dashboard  
**Team Members:**   
 Team Leader Name: THIRRUMURUGAN S ( [thirrumurugan3543@gmail.com](mailto:thirrumurugan3543@gmail.com) )

Team Member Name: RAGUL R ([rragul6615@gmail.com](mailto:rragul6615@gmail.com))

Team Member Name: SURENDAR K M ([surendermanikandan04@gmail.com](mailto:surendermanikandan04@gmail.com))

Team Member Name: AKASH A ([silambamakash@gmail.com](mailto:silambamakash@gmail.com))

2. Project Overview

**Purpose:**   
Cryptoverse is a comprehensive cryptocurrency dashboard that provides real-time data on various cryptocurrencies, including price trends, market analysis, and news updates. It offers users an intuitive interface to track and analyze crypto assets.  
  
**Features:**   
- Real-time Cryptocurrency Data with Graphs   
- Market Trends and Historical Data Analysis   
- Search and Filter Options for Easy Navigation   
- Cryptocurrency Details with Key Metrics (Market Cap, Volume, etc.)   
- Latest News on Crypto Market Updates   
- Responsive Design for Desktop and Mobile Devices   
- Dark/Light Mode Toggle for Visual Comfort

# 3. Architecture

**Component Structure:**   
**App.js** — Root component handling routing and layout   
**Header.js** — Navigation bar with search and theme toggle   
**Dashboard.js** — Displays key cryptocurrency metrics and market trends   
**CryptoList.js** — Lists top-performing cryptocurrencies   
**CryptoDetails.js** — Detailed information page for individual cryptocurrencies   
**NewsFeed.js**— Displays latest cryptocurrency-related news  **Footer.js**— Provides links to resources and social media   
  
**State Management:**  
**Redux Toolkit** — Used for managing global states like cryptocurrency data, bookmarks, and theme settings   
  
**Routing:**   
**React Router** — Enables navigation between the dashboard, crypto details, and news sections

# 4. Setup Instructions

**Prerequisites:**  
- Node.js (v18 or higher)   
- npm or yarn   
  
**Installation:**   
1. Clone the Repository:   
```bash  
git clone https://github.com/darunraj0071/cryptoverse-dashboard.git  
cd cryptoverse-dashboard  
```  
  
2. Install Dependencies:   
```bash  
npm install  
```  
  
3. Create `.env` file (for API keys or environment variables):   
```  
REACT\_APP\_CRYPTO\_API\_KEY=your\_api\_key\_here  
```  
  
4. Run the Application:   
```bash  
npm start  
```

# 5. Folder Structure

/src   
 ├── /components   
 │ ├── Header.js   
 │ ├── Dashboard.js   
 │ ├── CryptoList.js   
 │ ├── CryptoDetails.js   
 │ ├── NewsFeed.js   
 │ └── Footer.js   
 ├── /pages   
 │ ├── Home.js   
 │ ├── MarketTrends.js   
 │ ├── CryptoDetailsPage.js   
 │ └── NewsPage.js   
 ├── /assets   
 │ ├── images   
 │ ├── icons   
 │ └── styles   
 ├── /utils   
 │ ├── api.js   
 │ ├── formatters.js   
 │ └── constants.js   
 ├── /redux   
 │ ├── cryptoSlice.js   
 │ ├── themeSlice.js   
 │ └── store.js   
 ├── App.js   
 ├── index.js

# 6. Running the Application

**Frontend Server Command:**   
```bash  
npm start  
```

# 7. Component Documentation

**Key Components:**   
**Header:** Provides navigation, search bar, and theme toggler   
**Dashboard:** Displays real-time cryptocurrency data with visual charts   
**CryptoList:**Lists top-performing cryptocurrencies with key metrics   
**CryptoDetails:** Detailed view showing historical data, price trends, and key information   
**NewsFeed:** Displays crypto-related news articles and updates   
  
**Reusable Components:**   
**Card Component:** Displays individual cryptocurrency details   
**Button** **Component:** Used for primary actions like “View Details” or “Bookmark”

# 8. State Management

**Global State:**   
Managed via Redux Toolkit for cryptocurrency data, user preferences, and theme settings   
 **Local State:**  
Managed with React’s `useState()` for form inputs, modals, and UI interactions

# 9. User Interface

- Interactive charts for tracking cryptocurrency trends   
- Clean and responsive design optimized for all screen sizes   
- Dark/Light mode switch for improved visual accessibility

# 10. Styling

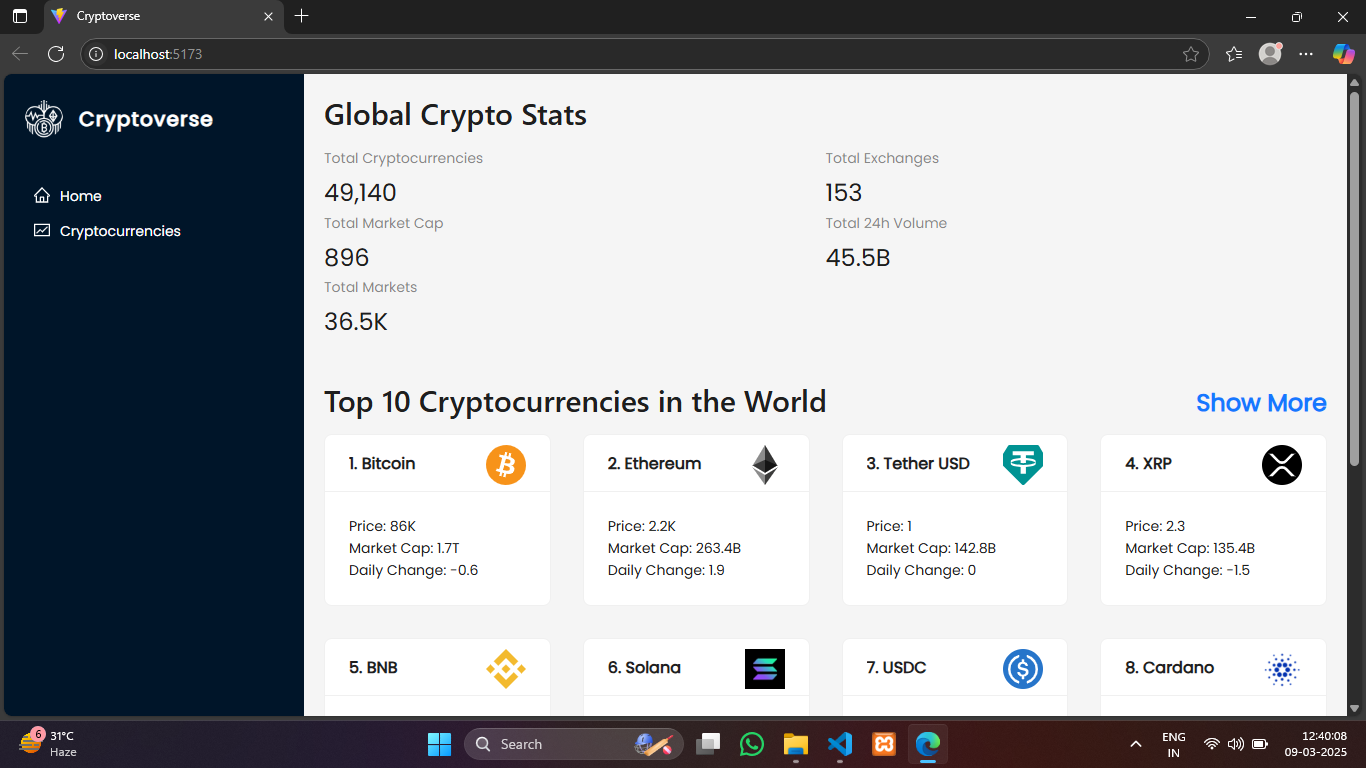
**CSS Frameworks/Libraries:**   
**Tailwind CSS** — Used for fast and flexible styling   
  
**Theming:**  
Integrated theme switcher with persistent user preferences

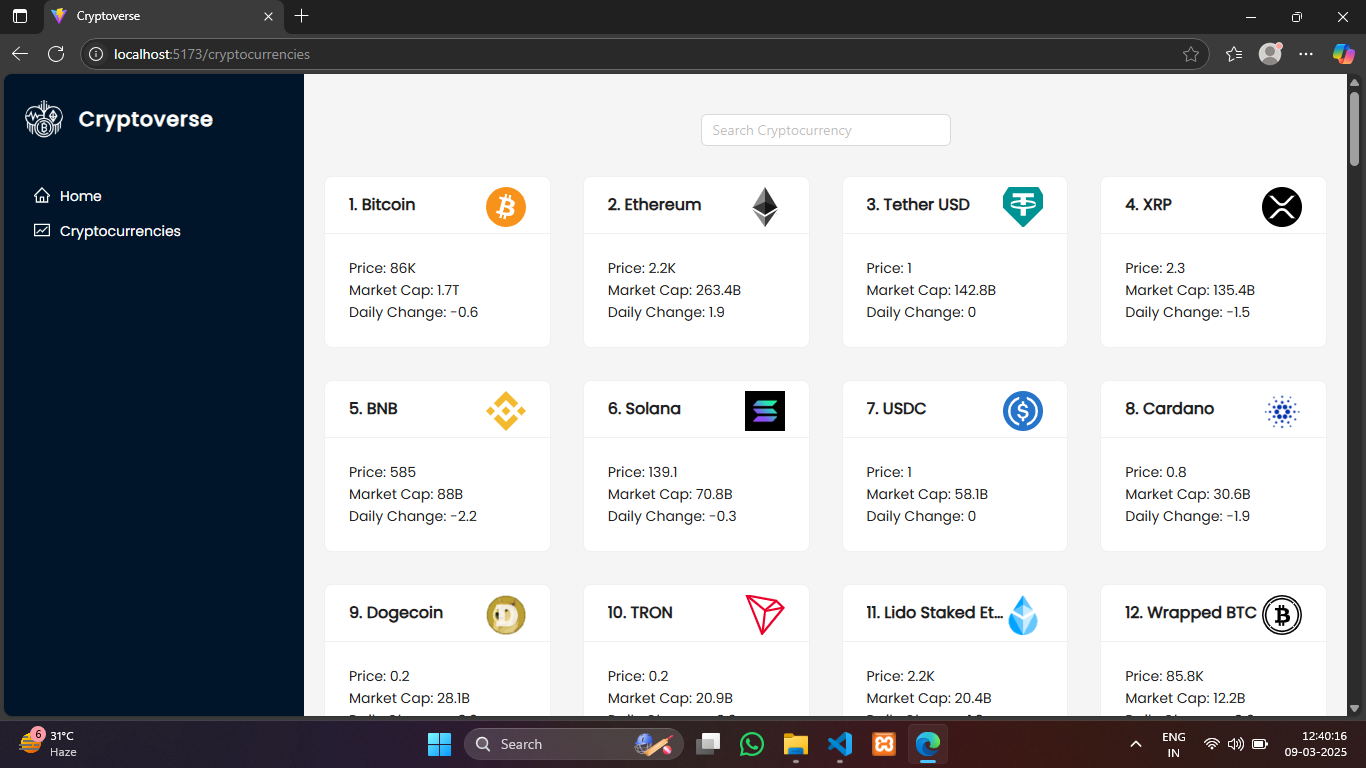
# 11. Testing

**Testing Strategy**:   
Jest for unit testing   
React Testing Library for component interaction testing   
 **Code Coverage:**   
Ensured 90%+ code coverage for improved application stability

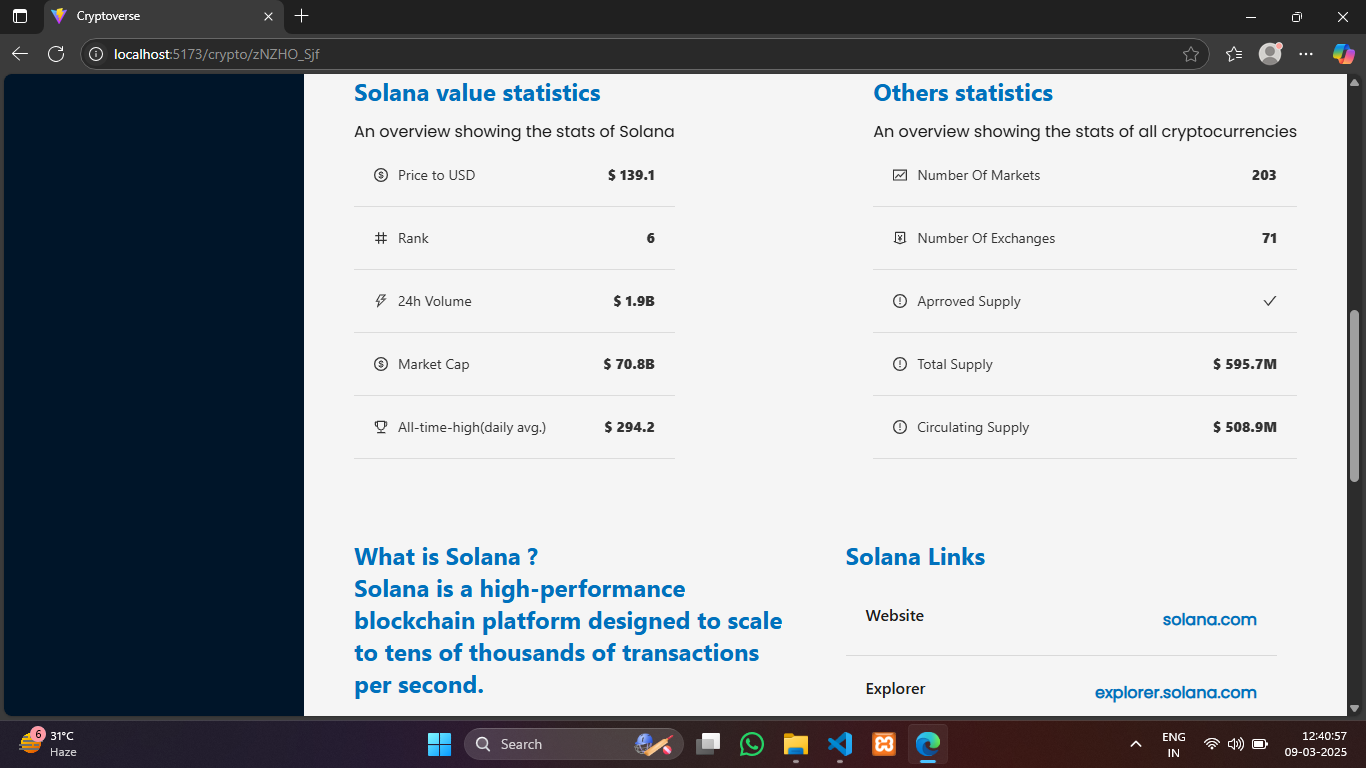
# 12. Screenshots or Demo

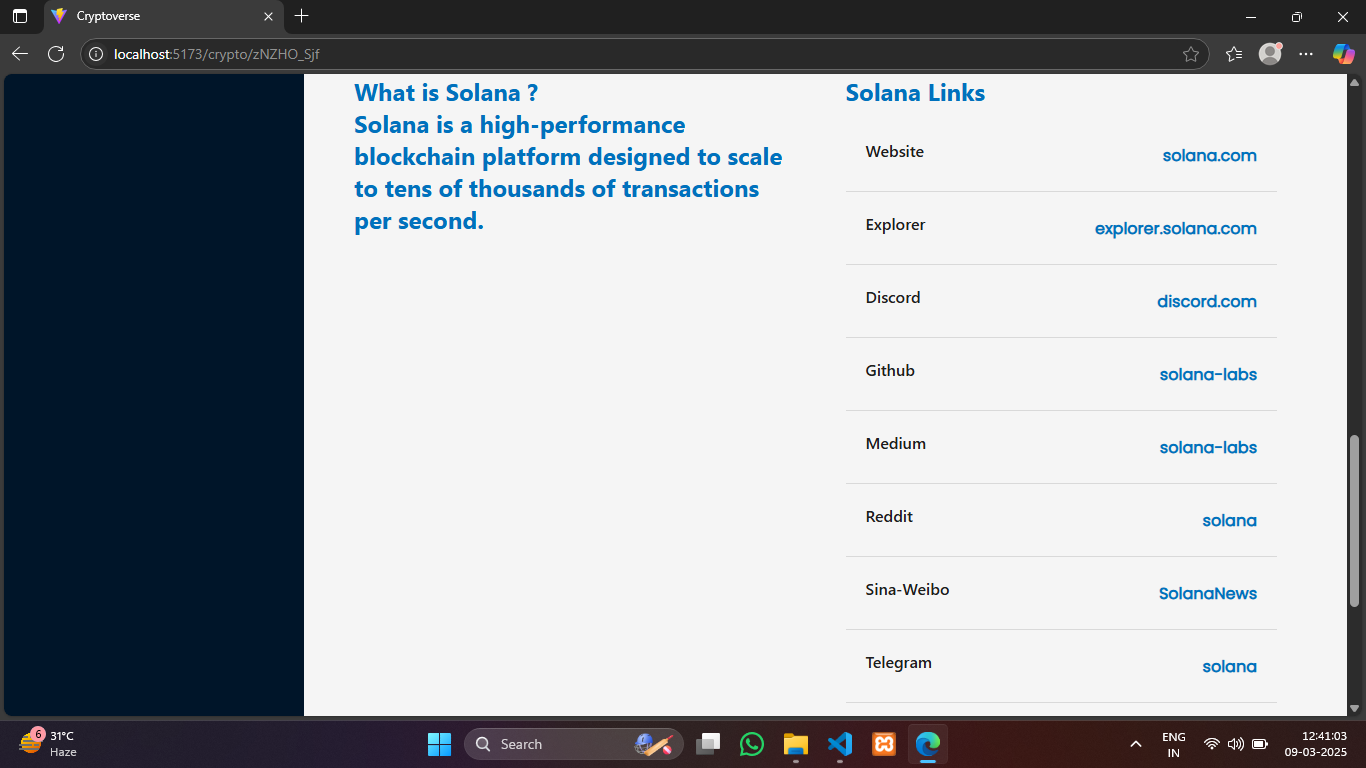
**Screenshots:**











**Live Demo Link:**

# 13. Known Issues

- API data may experience delays during high-traffic periods   
- Theme toggling may require slight optimization for faster transitions

# 14. Future Enhancements

- Integration with additional crypto APIs for diversified data sources   
- User Authentication for personalized cryptocurrency watchlists   
- Advanced Filtering Options for customized data insights   
- Interactive Portfolio Tracking for investment analysis