

Team ID: SWTID1741249032148951

Team Members:

Iyyanar.M

Sachin

Mohammed Kaif

Mohammed mushraf

Project Documentation: Cryptoverse

1. Introduction

Cryptoverse is a React-based web application built using Vite. It leverages modern frontend technologies and state management tools to create an efficient and interactive user experience.

2. Project Structure

The project is organized as follows:

- index.html: Entry point for the application, referencing the main JavaScript file.
- package.json: Contains project metadata, scripts, and dependencies.
- vite.config.js: Configuration file for Vite.
- README.md: Project description and setup guide.
- src/: Contains source code files including components, pages, and services.
- public/: Stores static assets like images and icons.

3. Installation & Setup

To install and set up the project, follow these steps:

Prerequisites:

- Node.js (latest stable version recommended)
- npm or yarn package manager

Steps:

1. Clone the repository:
`git clone <repository-url>`
`cd crypto`
2. Install dependencies:
`npm install`
3. Start the development server:
`npm run dev`
4. To build for production:
`npm run build`
5. To preview the build:
`npm run preview`
6. To check for linting issues:
`npm run lint`

4. Dependencies

The project uses the following dependencies:

- React (^18.2.0)
- React-DOM (^18.2.0)
- React-Router-DOM (^6.19.0)
- Redux Toolkit (^1.9.7)
- Ant Design (^5.11.2)
- Axios (^1.6.2)
- Chart.js (^4.4.0)
- Moment.js (^2.29.4)
- UUID (^9.0.1)
- Millify (^6.1.0)
- HTML React Parser (^5.0.6)

Development Dependencies:

- ESLint (^8.53.0)
- Vite (^5.0.0)
- @vitejs/plugin-react (^4.2.0)
- @types/react (^18.2.37)
- @types/react-dom (^18.2.15)

5. Configuration

Vite Configuration

Located in `vite.config.js`:

```
import { defineConfig } from 'vite';
import react from '@vitejs/plugin-react';
```

```
export default defineConfig({
  plugins: [react()],
});
```

ESLint Configuration

The project enforces linting rules using ESLint. Ensure code quality by running:

```
npm run lint
```

6. Additional Information

- The project is private as indicated in `package.json`.
- Uses ESLint for code linting and ensuring best practices.
- Supports HMR (Hot Module Replacement) through Vite.
- Uses Redux Toolkit for state management.
- React Router is used for client-side routing.

7. API & Data Fetching

The project fetches cryptocurrency data using Axios. API endpoints should be defined in a separate service file within the `src/services` folder.

8. Contribution Guidelines

- Fork the repository.
- Create a feature branch.

- Commit changes with meaningful messages.
- Push to your branch and create a pull request.

9. Conclusion

Cryptoverse is structured for scalability and performance using modern web technologies. Follow the setup guide to start contributing or using the application.