**Mini Project on**

**React Cryptocurrency Application**

**Jawaharlal Nehru Technological University Anantapur,**

# Ananthapuramu

in partial fulfillment of the requirements for the award of the degree of

**BACHELOR OF TECHNOLOGY IN**

**INFORMATION TECHNOLOGY**

*Submitted by*

|  |  |
| --- | --- |
| **21121A1222** | **C. Sai Ganesh** |
| **21121A1223** | **C. Ajay** |
| **21121A1224** | **C. Siva Rama Krishna** |
| **21121A1225** | **D. Vinathi** |
| **21121A1223** | **E. Sravanthi** |
| **21121A1227** | **G. Hemanth** |
| **21121A1224** | **G. Kishore** |

*Under the Supervision of*

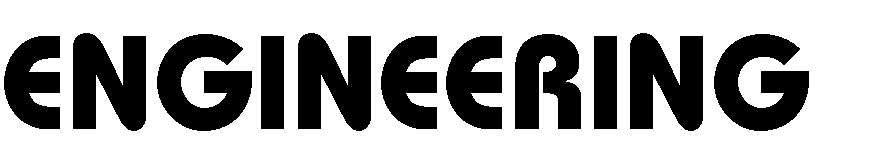
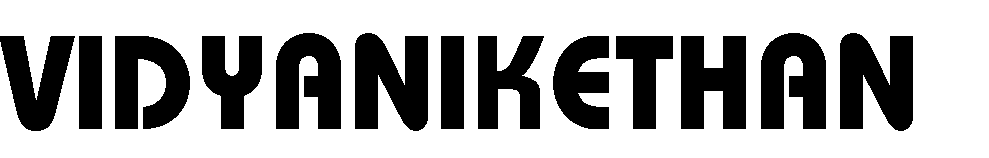
**Mr. Yuvraj Naik,** *M.Tech(Ph.D)*

**Assistant professor**

Department of Information Technology



**INFORMATION TECHNOLOGY**



(Affiliated to JNTUA, Ananthapuramu, Approved by AICTE, Accredited by NBA & NAAC) Sree Sainath Nagar, Tirupati – 517 102, A.P., INDIA

2023-2024

**TABLE OF CONTENTS**

**TITLE PAGE NO**

|  |  |
| --- | --- |
| **Abstract** | **3** |
| **Introduction** | **4-5** |
| **Proposed System** | **6** |
| **Algorithm** | **7-8** |
| **Procedure** | **9** |
| **Steps** | **10-11** |
| **Uses** | **12** |
| **Effiency** | **13** |
| **Output** | **14** |
| **Conclusion** | **15** |
| **References** | **15** |

# Fig.no Figure Name

|  |  |
| --- | --- |
| 1 | Home Page |
| 2 | All coins |

**Abstract**

## The React framework to provide users with real-time information and insights into various cryptocurrencies. Leveraging the power of React, this application offers an intuitive and interactive user interface, allowing users to track cryptocurrency prices, monitor market trends, and manage their cryptocurrency portfolios efficiently. The application fetches data from cryptocurrency APIs and displays it in a visually appealing manner, with features such as live price updates, historical price charts, and detailed coin information. Users can also customize their experience by setting up personalized alerts and notifications for price movements. With its responsive design and seamless user experience, the React Cryptocurrency Application aims to cater to both novice and experienced cryptocurrency enthusiasts, empowering them to make informed decisions in the dynamic world of digital assets.

## Keywords:

## React, Cryptocurrency, Web application, Real-time

## Cryptocurrency market, Coin information

## 

## 

## INTRODUCTION

Leveraging the power of React, a popular JavaScript library for building user interfaces, this application offers a seamless and intuitive user experience for cryptocurrency enthusiasts and investors.

In today's dynamic cryptocurrency market, staying informed about price fluctuations, market trends, and coin performances is crucial for making informed investment decisions. The React Cryptocurrency Application aims to address this need by offering a feature-rich platform where users can easily track the prices of their favorite cryptocurrencies, monitor market trends, and manage their portfolios effectively.

With its responsive design and interactive interface, the React Cryptocurrency Application allows users to explore detailed information about different cryptocurrencies, including live price updates, historical price charts, market capitalization, trading volume, and more. Users can customize their experience by setting up personalized alerts and notifications for price movements, ensuring that they never miss an opportunity to act on market changes.

Whether you are a seasoned cryptocurrency trader or a novice investor looking to explore the world of digital assets, the React Cryptocurrency Application provides a user-friendly platform to stay informed and make informed decisions in the fast-paced and ever-evolving cryptocurrency market.

React.js, commonly referred to as React, is a powerful JavaScript library developed by Facebook for building user interfaces (UIs). Its importance in modern web development stems from several key factors:

1. Component-Based Architecture: React adopts a component-based architecture, where UIs are composed of reusable and independent components. This modular approach makes it easier to manage complex UIs, improve code organization, and facilitate code reusability, resulting in faster development and easier maintenance.
2. Virtual DOM: React utilizes a virtual DOM (Document Object Model) to efficiently update the UI. Instead of directly manipulating the browser's DOM, React creates a lightweight representation of the DOM in memory and compares itwith the actual DOM to determine the minimal set of changes needed. This approach minimizes DOM manipulation and improves application performance, especially for large and dynamic applications.
3. Declarative Syntax: React's declarative syntax allows developers to describe the desired UI based on the application's state. This abstraction simplifies UI development by focusing on what the UI should look like rather than how to achieve it. As a result, code becomes more predictable, easier.
4. Community and Ecosystem: React has a vibrant and active community of developers, which has led to the creation of a rich ecosystem of libraries, tools, and resources. This ecosystem includes state management libraries like Redux, routing libraries like React Router, and UI component libraries like Material-UI. The abundance of resources and community support makes it easier for developers to build complex applications and solve common challenges.
5. Cross-Platform Compatibility: React can be used to build not only web applications but also mobile applications using frameworks like React Native. By leveraging the same component-based architecture and developer tools, developers can create native mobile apps for iOS and Android platforms using JavaScript and React.
6. Performance Optimization: React's virtual DOM and efficient rendering algorithms help optimize performance by minimizing DOM updates and re-renders. Additionally, React provides tools like React Profiler and React DevTools to analyze and optimize application performance, ensuring smooth and responsive user experiences.
7. SEO-Friendly\*\*: With the rise of server-side rendering (SSR) and static site generation (SSG) techniques, React applications can be made SEO-friendly. By pre-rendering HTML on the server or at build time, React apps can improve search engine indexing and visibility, leading to better search engine rankings and discoverability.

Overall, React.js is important in modern web development due to its simplicity, performance, scalability, and extensive ecosystem, making it a preferred choice for building dynamic and interactive.

**Proposed System**

The proposed system for the Application encompasses a range of features aimed at delivering a comprehensive and user-friendly platform for cryptocurrency enthusiasts. It begins with robust user authentication and authorization mechanisms to ensure secure access, followed by a customizable dashboard offering real-time updates on cryptocurrency prices, market trends, and performance indicators. Users can track and manage their cryptocurrency portfolios seamlessly, with options for adding personalized price alerts to stay informed about market movements. Integrated news feeds and market insights keep users abreast of the latest developments, while customization options, responsive design, and stringent security measures enhance the overall user experience. Thorough testing, feedback collection, and continuous improvement processes are integral to ensuring the application's reliability, functionality, and user satisfaction. Ultimately, the proposed system aims to empower users to make informed decisions in the dynamic cryptocurrency market landscape.

**Algorithm**

Algorithm: React Cryptocurrency Application

**Step 1 ->** Initialization:

Import necessary libraries and dependencies, including React, Axios for API requests, and any additional libraries for UI components.

- Set up the project structure, including components, routes, and state management.

**Step 2 ->** User Authentication:

2a. Implement user authentication functionality, including login, registration, and logout features.

2b. Use authentication tokens or cookies to manage user sessions securely.

**Step 3 ->** Dashboard Design:

3a. Create a dashboard component to display an overview of the cryptocurrency market.

3 b. Design UI elements to show real-time price updates, market trends, and performance indicators.

**Step 4 ->** Cryptocurrency Listings:

4a. Integrate cryptocurrency APIs to fetch real-time data for various cryptocurrencies.

4b. Implement a component to display cryptocurrency listings with details such as name, price, market cap, and volume.

**Step 5. ->** Portfolio Management:

5a. Develop features to allow users to track and manage their cryptocurrency portfolios.

5b. Implement functionalities for adding, removing, and updating portfolio holdings.

5c. Display portfolio performance metrics and charts to visualize gains/losses over time.

**Step 6 ->** Price Alerts:

6a. Create components for users to set up personalized price alerts for specific cryptocurrencies.

6b. Implement functionality to monitor cryptocurrency prices and trigger alerts based on user-defined thresholds.

6c. Integrate notification systems to deliver alerts via email, SMS, or push notifications.

**Step 7 ->** News and Insights:

7a. Integrate news APIs or RSS feeds to fetch the latest cryptocurrency news and market insights.

7b. Design UI components to display curated news articles, analysis, and updates within the application.

**Step 8 ->** Customization Options:

8a. Provide users with options to customize their dashboard layout, theme, and language preferences.

8b. Implement settings pages to allow users to adjust personalization settings according to their preferences.

**Step 9 ->** Responsive Design:

9a. Ensure that the application UI is responsive and accessible across various devices and screen sizes.

9b. Utilize responsive design techniques such as media queries and flexbox to adapt the layout for different viewports.

**Step 10 ->** Security Measures:

10a. Implement security measures to protect user data, including encryption for sensitive information and secure authentication methods.

10b. Follow best practices for web application security to prevent common vulnerabilities such as XSS and CSRF attacks.

**Step 11 ->** Testing and Quality Assurance:

11a. Conduct thorough testing of the application, including unit tests, integration tests, and end-to-end tests.

11b. Perform usability testing to gather feedback on the user experience and identify areas for improvement.

**Step 12 ->** Deployment:

12a. Prepare the application for deployment to a web hosting platform or server.

12b Configure deployment settings and optimize performance for production environments.

**Step 13 ->** Feedback and Iteration:

13a. Collect user feedback and analytics data to identify areas for improvement.

13b. Iterate on features, functionality, and UI design based on user input to enhance the overall user experience.

End Algorithm

**Module**

**Steps to Create React Cryptocurrency Application:**

1. Set Up Environment:

- Create an empty folder on your desktop named "crypto-react-app".

- Open Visual Studio Code or any preferred code editor.

- Drag and drop the empty folder into the code editor.

- Open the terminal in Visual Studio Code.

2. Initialize React App:

- Clear the terminal.

- Run `npx create-react-app` to initialize an empty React application in the current directory.

3. Folder Structure:

- Inside the created React app, navigate to the "src" folder.

- Delete the "src" folder.

- Create a new folder named "src".

4. Create Index.js:

- Inside the "src" folder, create a new file named "index.js".

- Import React and the main App component.

- Render the App component using ReactDOM.

5. Add App Component:

- Create a new file named "App.js" in the "src" folder.

- Install necessary dependencies using `npm install`.

- Install ant design, Redux, Axios, Chart.js, html-react-parser, millify, react-chartjs-2.

- Import required components and libraries in App.js.

- Create the main layout for the application.

6. Develop Navigation Bar:

- Create a new folder named "components" inside the "src" folder.

- Inside the "components" folder, create a new file named "Navbar.jsx".

- Implement the Navbar component using ant design for navigation.

7. Styling with CSS:

- Create a new file named "App.css" in the "src" folder.

- Add necessary CSS styles for the layout and responsiveness.

8. Homepage Development:

- Create a new file named "Homepage.jsx" in the "components" folder.

- Import required components and libraries.

- Fetch cryptocurrency data using Redux toolkit and API calls.

- Display global cryptocurrency stats and top cryptocurrencies.

9. Redux Toolkit and API Development:

- Install Redux Toolkit using `npm install @reduxjs/toolkit`.

- Develop Redux store, reducers, and actions for fetching cryptocurrency data.

- Make API requests to fetch real cryptocurrency data from a rapid API.

- Utilize Redux state management for predictable and centralized data handling.

10. Testing and Debugging:

- Test the application by running `npm start`.

- Debug any errors or issues encountered during development.

- Ensure proper functionality and responsiveness across different devices.

11. Deployment:

- Once the application is fully developed and tested, deploy it to a hosting service or platform.

- Ensure proper configuration and optimization for production deployment.

- Monitor application performance and user feedback for further improvements.

By following these step-by-step procedures, you can create a React cryptocurrency application with efficient data fetching, state management, and user interface components.

**Application**

The React cryptocurrency application serves several purposes and offers various benefits, including:

1. Real-time Market Data: Users can access real-time data on cryptocurrency prices, market trends, and performance indicators, enabling them to make informed investment decisions.
2. Portfolio Management: The application allows users to track and manage their cryptocurrency portfolios efficiently. Users can monitor their holdings, view performance metrics, and analyze their investment strategies.
3. Price Alerts: Users can set up personalized price alerts for specific cryptocurrencies. This feature enables users to stay informed about price movements and take timely actions based on their investment goals.
4. News and Insights: Integrated news feeds and market insights provide users with up-to-date information on the latest developments in the cryptocurrency market. Users can access curated news articles, analysis, and updates to stay informed about market trends and events.
5. Customization Options: The application offers customization options, allowing users to personalize their dashboard layout, theme, and language preferences. This enhances the user experience and enables users to tailor the application to their preferences.
6. Security: The application implements robust security measures to protect user data

**Result**

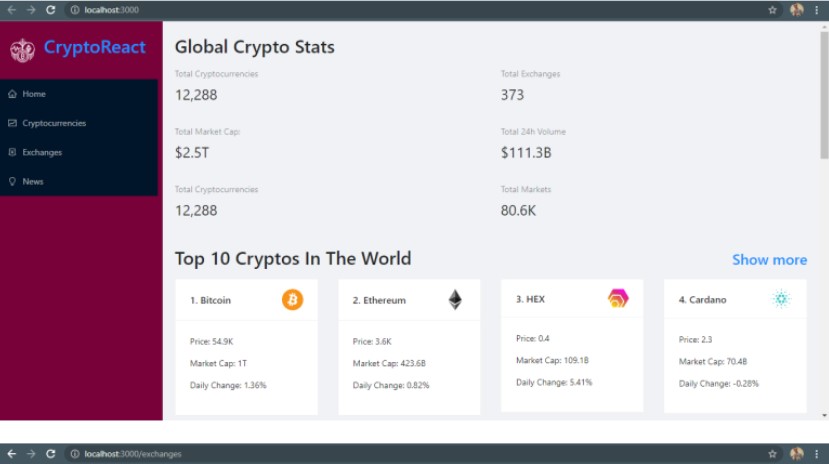


Fig 1: Home Page



Fig 2: All coins

**Conclusion**

To deploy the React Cryptocurrency Application, start by building the project using the terminal in Visual Studio Code. Once the build process completes, locate the generated "build" folder in your project directory. Then, navigate to your chosen hosting service platform, such as Netlify, and log in to your account. Access the deployment section or dashboard, and upload the "build" folder by dragging and dropping it onto the platform. Wait for the upload to finish, and your React application will be deployed on the web. You'll receive a unique URL for your deployed application, which you can share with others to access the live version. This allows users to explore cryptocurrency data, charts, and news conveniently. By following these steps, you can efficiently deploy your React Cryptocurrency Application and make it accessible to a broader audience online.

**References**

1. <https://www.interviewbit.com/blog/react-projects/>

2. <https://github.com/Megh2507/React-Crypto-App>

3. <https://chat.openai.com/c/90427ef6-2ef9-4b45-8019-4337e8daca34>

4. <https://blog.ldtalentwork.com/2021/10/13/how-to-create-a-cryptocurrency-app-using-react/>