

# CRYPTOBYTE

(DASHBOARD)

**Team ID :** SWTID1741152182

**Team Size :** 5

**Team Leader :** ZAHEERUDEEN S

**Team member :** ROCHAN KRISHNAMOORTHY

**Team member :** HARIHARAN P

**Team member :** MONESH R

**Team member :** YUVAN RAJ R

ZAHEERUDEEN S - sirajudeenzahir21@gmail.com

ROCHAN KRISHNAMOORTHY - rochankrishnamoorthy@gmail.com

HARIHARAN P - harish28070@gmail.com

MONESH R - moneshravi123@gmail.com

YUVAN RAJ R - yuvanramesh111@gmail.com

## Introduction

A crypto currency dashboard that displays historical price data over the past five years is

powerful tool for investors seeking a comprehensive understanding of market dynamics.

This feature-rich interface offers users a detailed historical perspective on the performance of various crypto currencies, enabling insightful analysis and informed decision-making.

Through visually intuitive charts and graphs, the dashboard allows for effective comparisons of multiple crypto currencies, aiding in the identification of top performers and overall market trends.

### **Description:**

CRYPTOBYTE is a sophisticated cryptocurrency dashboard designed to provide investors with comprehensive insights into market dynamics through detailed historical price data analysis spanning five years. Featuring visually intuitive charts, interactive tools, and seamless

### **Scenario based introduction:**

Imagine waking up in a world where traditional banks no longer exist. Instead of relying on middlemen, people transact directly with one another using a digital currency that isn't controlled by any government or institution. You walk into a store, scan a QR code, and instantly send payment from your digital wallet—no waiting, no fees, and no need for a bank.

navigation, the platform empowers users to identify top-performing assets and make informed investment decisions.

### **Technical architecture:**

The CRYPTOBYTE is built on a decentralized architecture powered by blockchain technology and a combination of distributed computing, cryptographic security, and consensus mechanisms.



### **Project goals and objectives:**

Project Goals Decentralization & Trustless Economy

Build a permissionless system where users can transact and interact without intermediaries.

Implement DAO (Decentralized Autonomous Organization) governance to ensure community-driven decisions

### **Scalability & High-Performance Infrastructure**

Utilize Layer 2 solutions (Rollups, State Channels) and sharding for high-speed transactions.

Ensure low transaction fees and fast settlement times for mass adoption

## **Interoperability & Cross-Chain Integration**

Enable seamless asset transfers across different blockchain networks.

Develop bridges and compatibility with major ecosystems like Ethereum, Solana, and Polkadot.

## **Key features:**

### **Decentralization & Trustless Transactions**

No central authority; transactions are verified by blockchain consensus.

Users have full control over their assets without intermediaries.

## **Blockchain Infrastructure**

Built on a scalable, high-performance blockchain (Layer 1 or Layer 2).

Supports smart contracts for automation and transparency.

## **Smart Contracts & Automation**

Self-executing contracts with predefined rules (e.g., Ethereum's Solidity).

Enables decentralized finance (DeFi), NFTs, and governance.

## Interoperability & Cross-Chain Integration

Compatible with multiple blockchains (Ethereum, Binance Smart Chain, Solana, Polkadot). Bridges and wrapped tokens for seamless asset transfers.

### Pre-Requisites

To develop InsightStream, the following tools and **technologies are required**:

**Node.js & npm:** Install and configure for JavaScript execution.

Download: <https://nodejs.org/en/download/>

#### Installation instructions:

<https://nodejs.org/en/download/package-manager/> **React.js:**

Create a dynamic user interface.

HTML, CSS, & JavaScript: Basic web development knowledge is essential.

**Git Version Control:** Use platforms like GitHub for collaboration.

**Git:** Download and installation instructions can be found at:

<https://git-scm.com/downloads/>

**Development Environment:** Recommended editors include Visual Studio Code, Sublime Text, or WebStorm.

**Visual Studio Code:** Download from

<https://code.visualstudio.com/download>

**Sublime Text:** Download from

<https://www.sublimetext.com/download>

**WebStorm:** Download from

<https://www.jetbrains.com/webstorm/download>

**Navigate to the project directory and install dependencies:**

```
cd news-app-react npm install
```

**Start the development server:** npm

```
start
```

Open a browser and visit **<http://localhost:3000>** to access the application.

## **Project Structure**

**The project is organized into four primary folders:**

**Components:** Houses reusable UI elements.

**Context:** Contains the Context API for global state management.

**Pages:** Stores page-specific components that map to different URLs.

**Styles:** Includes all CSS files for styling.

## **Project Milestones**

**Milestone 1:** Project Setup & Configuration

Install necessary tools including React.js, React Router DOM, Axios, and Tailwind CSS.

### **Set up the basic project structure.**

Open the project folder to install necessary tools. In this project, we use:

- React.js
- React Router Dom
- React Icons
- Bootstrap/Tailwind CSS
- Axios

For further reference, use the following resources

<https://react.dev/learn/installation> <https://react-bootstrap-v4.netlify.app/gettingstarted/introduction/>  
<https://axios-http.com/docs/intro>  
<https://reactrouter.com/en/main/start/tutorial>

## **Milestone 2: Development Phase**

- **Conceptualization & Planning**

**Define the vision:** What will the Cryptoverse offer (gaming, social, commerce, etc.)?

**Select the blockchain:** Ethereum, Solana, or a custom chain?

**Identify token utility:** Governance, transactions, staking, etc.

Research market trends and competitors.

- **Blockchain & Smart Contract Development**

Develop smart contracts for tokens, NFTs, and in-game assets.

Build a native token and set up liquidity pools

## Fetching crypto currency via api:

**Create a redux store:** have a fully functional Redux store with a counter slice, and you can integrate it into your React app

```
1 import { configureStore } from "@reduxjs/toolkit";
2 import { cryptoApi } from "../services/cryptoApi";
3
4 export default configureStore({
5   reducer: {
6     [cryptoApi.reducerPath]: cryptoApi.reducer,
7   },
8   middleware: (getDefaultMiddleware) =>
9     getDefaultMiddleware().concat(cryptoApi.middleware),
10 });
11
```

## Create a component to show the details of cryptocurrency:

This code defines a React functional component called CryptoDetails responsible displaying detailed information



about a specific cryptocurrency.

```

1 import React from "react";
2 import milify from "millify";
3 import { Typography, Row, Col, Statistic } from "antd";
4 import { Link } from "react-router-dom";
5 const { Title } = Typography;
6 import { useGetCryptosQuery } from "../services/cryptoApi";
7 import Cryptocurrencies from "../Cryptocurrencies";
8 import Loader from "../Loader";
9
10 const Home = () => {
11   const { data, isFetching } = useGetCryptosQuery(10);
12   if (isFetching) return <Loader />;
13   const globalStats = data?.data?.stats;
14
15   return (
16     <>
17       <Title level={2} className="heading">
18         Global Crypto Stats
19       </Title>
20       <Row>
21         <Col span={12}>
22           <Statistic title="Total Cryptocurrencies" value={globalStats.total} />
23         </Col>
24         <Col span={12}>
25           <Statistic
26             title="Total Exchanges"
27             value={milify(globalStats.totalExchanges)}
28           />
29         </Col>
30         <Col span={12}>
31           <Statistic
32             title="Total Market Cap"
33             value={milify(globalStats.totalMarketCap)}
34           />
35         </Col>
36         <Col span={12}>
37           <Statistic
38             title="Total 24h Volume"
39             value={milify(globalStats.total24hVolume)}
40           />
41         </Col>
42         <Col span={12}>
43           <Statistic
44             title="Total Markets"
45             value={milify(globalStats.totalMarkets)}
46           />
47         </Col>
48       </Row>
49       <div className="home-heading-container">
50         <Title level={2} className="home-title">
51           Top 10 Cryptocurrencies in the world
52         </Title>
53         <Title level={3} className="show-more">
54           <Link to="/cryptocurrencies">Show More</Link>
55         </Title>
56       </div>
57       <Cryptocurrencies simplified />
58     </>
59   );
60 };
61
62 export default Home;

```

## **Create a Homepage:**

This component, named Home, is a React functional component responsible for rendering the home page of the cryptocurrency dashboard.

```

1 import React from "react";
2 import millify from "millify";
3 import { Typography, Row, Col, Statistic } from "antd";
4 import { Link } from "react-router-dom";
5 const { Title } = Typography;
6 import { useGetCryptosQuery } from "../services/cryptoApi";
7 import Cryptocurrencies from "../Cryptocurrencies";
8 import Loader from "../Loader";
9
10 const Home = () => {
11   const { data, isFetching } = useGetCryptosQuery(10);
12   if (isFetching) return <Loader />;
13   const globalStats = data?.data?.stats;
14
15   return (
16     <>
17       <Title level={2} className="heading">
18         Global Crypto Stats
19       </Title>
20       <Row>
21         <Col span={12}>
22           <Statistic title="Total Cryptocurrencies" value={globalStats.total} />
23         </Col>
24         <Col span={12}>
25           <Statistic
26             title="Total Exchanges"
27             value={millify(globalStats.totalExchanges)}
28           />
29         </Col>
30         <Col span={12}>
31           <Statistic
32             title="Total Market Cap"
33             value={millify(globalStats.totalMarketCap)}
34           />
35         </Col>
36         <Col span={12}>
37           <Statistic
38             title="Total 24h Volume"
39             value={millify(globalStats.total24hVolume)}
40           />
41         </Col>
42         <Col span={12}>
43           <Statistic
44             title="Total Markets"
45             value={millify(globalStats.totalMarkets)}
46           />
47         </Col>
48       </Row>
49       <div className="home-heading-container">
50         <Title level={2} className="home-title">
51           Top 10 Cryptocurrencies in the world
52         </Title>
53         <Title level={3} className="show-more">
54           <Link to="/cryptocurrencies">Show More</Link>
55         </Title>
56       </div>
57       <Cryptocurrencies simplified />
58     </>
59   );
60 };
61
62 export default Home;
63

```

## Project Execution

Once the development is complete, start the application using: `npm start`

This will launch the app at **`http://localhost:3000`**, displaying key components:

### USER SNIPPETS:

**Home page** : This page consists of stats of global crypto like total cryptocurrencies total exchanges, market cap etc. Also consist of top 10 cryptocurrencies in the world.

**Crypto currencies page** : This page 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