

Cryptocurrency Tracker

**A Project Report Submitted in Fulfillment of the Requirements for the Award of
Degree**

Of

Bachelor of Technology

In

Computer Science and Engineering

By

Jaffer Hassan Bhat(11192722)

Rakeshkumar Sah(11192664)

Shubham Srivastava(11192719)

Under the supervision of

Mr. Charanjit Singh

Assistant Professor , CSE MMEC



M.M. Engineering College, Mullana, Ambala

**Maharishi Markandeshwar (Deemed to be University), Mullana, Ambala, Haryana,
India**

May 2022

Cryptocurrency Tracker

A Project Report Submitted in Fulfillment of the Requirements for the Award of Degree

Of

Bachelor of Technology

In

Computer Science and Engineering

By

Jaffer Hassan Bhat(11192722)

Rakeshkumar Sah(11192664)

Shubham Srivastava(11192719)

Under the supervision of

Mr. Charanjit Singh

Assistant Professor, CSE,MME



M. M. Engineering College, Mullana, Ambala

**Maharishi Markandeshwar (Deemed to be University), Mullana, Ambala, Haryana,
India**

May 2022

Department of Computer Science & Engineering

M. M. Engineering College, Mullana, Ambala, Haryana, India

Candidate's Declaration

We hereby certify that the work which is being presented in the project entitled "**Cryptocurrency Tracker**" in fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering of M. M. Engineering College, Mullana, Ambala, Haryana, India is an authentic record of our own work carried out during a period from Jan 2022 to Apr 2022, under the supervision of **Mr. Charanjit Singh (Assistant Professor)**. The matter presented in this project report has not been submitted by us for the award of any degree of this or any other Institute/University.

Jaffer Hassan Bhat

(11192722)

Rakeshkumar Sah

(11192664)

Shubham Srivastava

(11192719)

This is to certify that the above statement made by the above candidates is correct to the best of my knowledge.

Date:

Name & Signature of supervisor

Assistant Professor

CSE MMEC

Department of Computer Science & Engineering

M. M. Engineering College, Mullana, Ambala, Haryana, India

Acknowledgement

We wish to express my deep sense of indebtedness and sincerest gratitude to our guide, **Mr. Charanjit Singh (Assistant Professor)** for his/her invaluable guidance and constructive criticism throughout this work. She/he has displayed unique tolerance and understanding at every step of progress and encourages us. We deem it our privilege to have carried out our project work under his/her able guidance.

We would especially like to thank **Dr. Sandip Kumar Goel (Professor and Head, CSE Department, MMEC)** without whom, this work would not have been as it is now.

As a Final Personal Note, We are grateful to our parents, who are inspirational to us in their understanding, patience and constant encouragement.

Jaffer Hassan Bhat(11192722)

Rakeshkumar Sah(11192664)

Shubham Srivastava(11192719)

TABLE OF CONTENTS

Candidate's Declaration	
Acknowledgement	
Supervisor Performa.....	
Abstract.....	
List of Figures	
Chapter 1. Introduction about Project	1-3
1.1. Overview.....	1
1.2. Background View.....	1
1.3. Problem Overview	2
1.4. Project Planning	2-2
1.5. Reason For Selection.....	3
Chapter 2. System Design	4-6
2.1 Design.....	4
2.2 User Characteristics.....	5
2.3 System Information	5
2.4 System Analysis.....	5-6
Chapter 3. Objectives of Project Work.....	7
Chapter 4. Hardware and Software Requirements	8
4.1. Hardware Required	8
4.2. Software Required	8
Chapter 5. Implementation Tools for Project.....	9-10
5.1. ReactJS.....	9
5.2. API.....	9
5.3. Redux.....	9
5.4. HTML.....	10
5.5. CSS.....	10
5.6. Ant-design.....	10

Chapter 6. Project Model View.....	11-18
6.1 Home.....	11-12
6.2 Cryptocurrency.....	13
6.3 Searching Cryptocurrency.....	14
6.4 Cryptocurrency Live Price Chart.....	15
6.5 Cryptocurrency Statistics and other info	16
6.6 Cryptocurrency Details and related websites link.....	17
6.7 Cryptocurrency Latest News.....	18
 Chapter 7 Conclusion and Future Enhancements.....	 19-20
7.1 Conclusion.....	19
7.2 Future scope.....	19-20
 APPENDIX A: Source Code.....	 21-51

CRYPTOCURRENCY TRACKER

Abstract:

For the last few years there is always a news about cryptocurrency what are they and some of them like Bitcoin and Ethereum has become a huge assets. The total market cap of all the listed cryptocurrencies is over \$2trillion, and it is huge and growing forever. For the people who are holding cryptocurrencies are millionaires are now-a-days. People really like to have cryptocurrencies, most are them are willing to buy and due to the volatile nature of cryptos the price of ups and down within each passing seconds. So, there is a great need to keep an eye on it. Our project gives the live tracking of cryptocurrencies how's the rate changing. There are always new cryptocurrencies listing and making huge profit. So it is also important to keep an eye on new cryptos.

Apart from these, the cryptos always make to the blogs and news website. Government across the globe has started accepting the virtual money, Indian government has also accepted the virtual money. That's why it will be growing forever. The project has the feature to check the top 100 cryptocurrencies according to the market cap. It also shows the live chart of cryptos. There are links provided to different websites related to each cryptocurrencies one can visit to the respected website to know more about the cryptocurrencies. There is also small description about each cryptocurrencies. We have exchange facilities, we can exchange from one cryptos to the other according to the market values(this feature is disabled because we need premium plan for this).

List of Figures

Fig No.	Figure Caption	Page No.
2.1	Design	4
6.1	Homepage	11
6.2	Homepage footer	12
6.3	Cryptocurrency	13
6.4	Searching cryptocurrency	14
6.5	100th cryptocurrency	14
6.6	Cryptocurrency live price chart	15
6.7	Cryptocurrency Statistics and Other Information	16
6.8	Cryptocurrency details and related websites links	17
6.9	Cryptocurrency Latest News	18

CHAPTER-1

INTRODUCTION

1.1 Overview

For the last few years cryptocurrency has been the most talked topic around the globe. Why, because of the high return investment to the investors . And it has just begun a few years and in the long run it is going to give huge return and probably the best investment in now-a-days and everyone is talking about that.

Blockchain has taken the world by storm as the people showing interest in cryptocurrencies. Cryptocurrencies are volatile in nature and in few minutes it can be growing up or going down in price. Any kind of actions by organizations or even by the government can affect the markets of cryptocurrencies.

1.2 Background View

The "cryptocurrency tracker" has been developed to track the listed cryptocurrency, can view live chart and exchange among cryptocurrencies. The application is reduced as much as possible to avoid errors while entering the data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is user-friendly.

Cryptocurrency tracker, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping.

Project is simply and user-friendly responsive website for one to easily track the cryptocurrencies their description all the latest news through news portal.

Cryptocurrency Tracker is fast gaining ground as an accepted and used paradigm. More and more people are using to track the real-time data , prices and market cap of various cryptocurrencies. So, it is becoming popular day by day among people.

The objective of this project to develop a general-purpose tracking website where any one can easily track one's favourite cryptocurrencies and if they want to exchange it with others they can also do the same using 3rd party exchange website.

1.3 Problem Overview

Our project is mainly based on these two problems one to give the real-time data, one can insight to the cryptos price at the real-time and the what was the price of it was 5 years ago.

Second one is based on the news, as mentioned earlier the nature of cryptocurrencies is volatile so news is important. To check all the latest news about each cryptocurrencies one have to surf hours on different websites. We have given the facilities of news related to cryptocurrencies. Like any websites e.g. New York Times or business insider or any other websites publish any article related to the any cryptocurrencies one can show in the same website ,now to check out other different websites. There is a search facilities given through which one can check out his favourite cryptocurrencies and it's news.

1.4 Project Planning

Our project is build in responsive manner. It can easily support any web browser on any operating system. It is great looking websites with right colour combination delighted to watch out. In the left side we have logo and navbar in the right side we have some data about the cryptos and below them there is top 10 cryptos list and there is also list of some news. There is footer section which navigates to different sections.

Since it is build using ReactJS which is library of JavaScript. One can easily go to different pages(components) without refreshing.

Cryptocurrency tracker, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities

rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources.

1.5 Reason for Selection

The reason for selecting this project is very specific. The cryptocurrency has been the hottest topic around the globe in recent times. To make a website that gives every details about cryptos at the same level. To make this project we learnt how to work with API . To make a proper use of the API and fetching the data to the website is a great thing to learn. One more thing we can upgrade this website to more features like adding wallets and some other new technologies to look it more dynamic and advance in nature.

CHAPTER -2

SYSTEM DESIGN

2.1 Design

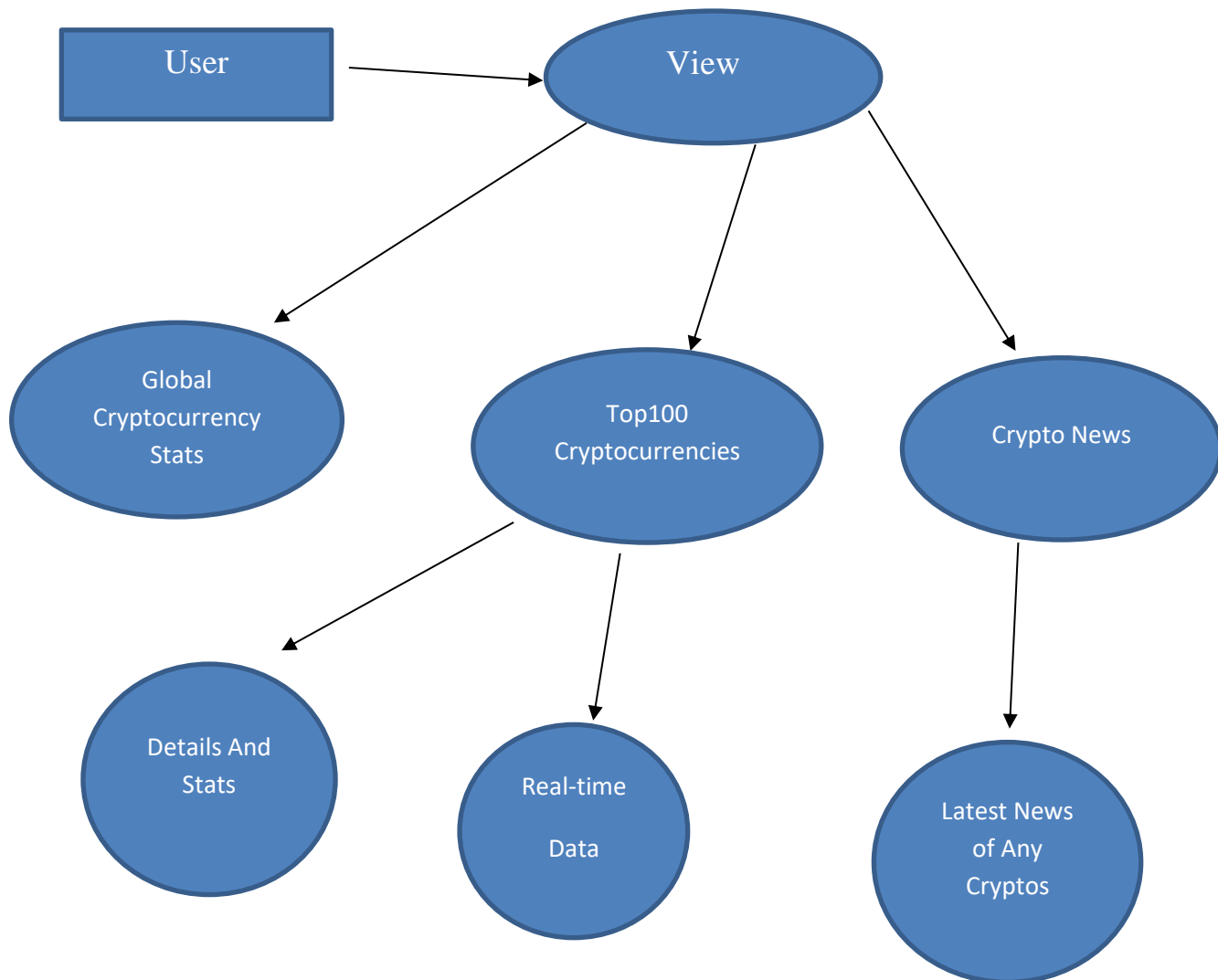


Fig 2.1.1

The system has divided into three main components. The components are Home Page , Cryptocurrency and News. In the Home Page section there are some global stats are

shown they are updated in real time. Second one is about cryptocurrencies and third part is about cryptocurrency news.

2.2 User Characteristics

Since this is just a website which shows all the details, stats and data about cryptocurrencies. So anyone can visit this website just searching on web browser. This website is made using only front-end so no user sign-in login option. In future when we add backend and wallets using web 3.0 technology we will add that section later. Now just user can visit all the pages(components) without sign in.

2.3 System Information

This system is an automated cryptocurrency tracking website. Through the software one can view the real-time data analytics, details ,news and can do exchanges. The system has following advantages:

- User Friendly Interface
- Search Facilities
- Real-time data analytics
- Look and Feel Environment

2.4 System Analysis

System analysis refers to the process of examining a situation with the intent of improving it through better procedures and methods. System analysis is the process of planning a new system either to replace or complement the existing system. System analysis is the of gathering and interpreting facts, diagnosing problems and using the information to re-comment improvements in the system . System analysis is conducted with following objectives in mind:

- Evaluate the system concept for feasibility.
- Perform economic and technical analysis.

- Allocate functions to hardware, software people, database and other system elements.
- Create a system definition that forms the foundation for all the subsequent engineering.

CHAPTER-3

OBJECTIVE OF THE PROJECT

The objective of this project is to create a cryptocurrency tracker web portal to show the real time stats. The web portal will have an online interface in the form of an cryptocurrency tracker that will allow users to check real time stats of any listed cryptocurrency.

The cryptocurrency tracker portal will have the following key features:

- A list of all the cryptocurrency that has been listed.
- A search engine on the website to allow customers to find any cryptocurrency.
- A track system to track the real-time stats of any cryptocurrency.
- An exchange option where user can exchange among other cryptocurrency.
- It tracks all the information of cryptocurrency and if any new cryptocurrency is it will
- Management of the website so that any problem occurred then it can be repaired in less time.

CHAPTER-4

HARDWARE AND SOFTWARE REQUIREMENTS

4.1 Hardware Required

- Processor : Pentium IV and above
- Ram : 2GB or above
- Hard Disk : 50GB or above
- Input Device: Keyboard, Mouse
- Output Device: Monitor

4.2 Software Required

- Operating System: Linux, Ubuntu , Mac , Windows 8 and above
- Frontend: ReactJS
- Toolkit: Redux
- API: Rapid API

CHAPTER-5

IMPLEMENTATION TOOLS FOR PROJECT

Tools

- **ReactJS**
- **API**
- **Redux**
- **HTML**
- **CSS**
- **Ant-design**

5.1 ReactJS

React is a declarative, efficient, and flexible JavaScript library for building user interfaces. ReactJS is a declarative, efficient, and flexible JavaScript library for building reusable UI components. It is an open-source, component-based frontend library responsible only for the view layer of the application. It was created by **Jordan Walke**, who was a software engineer at **Facebook**. Facebook developed ReactJS in **2011** in its newsfeed section, but it was released to the public in the month of **May 2013**. It lets you compose complex UIs from small and isolated pieces of code called “components”.

5.2 API

API is the acronym for Application Programming Interface, which is a software intermediary that allows two applications to talk to each other. We have used Rapid API.com where we have used coinranking and Bing search news to fetch data about the cryptos and news respectively.

5.3 Redux

Redux is an open-source JavaScript library used to manage application state. React uses Redux for building the user interface. React Redux is the official React binding for Redux. It allows React components to read data from a Redux Store, and dispatch **Actions** to the **Store** to update data. Redux helps apps to scale by providing a sensible way to manage state through a unidirectional data flow model. React Redux is conceptually simple.

5.4 HTML

HTML (HyperText Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content.

5.5 CSS

Stands for "Cascading Style Sheet." Cascading style sheets are used to format the layout of Web Pages. They can be used to define text styles, table sizes, and other aspects of Web pages that previously could only be defined in a page's HTML. CSS helps Web developers create a uniform look across several pages of a Web site. Instead of defining the style of each table and each block of text within a page's HTML, commonly used styles need to be defined only once in a CSS document.

5.6 Ant-design

It is a collection of components governed by UI standards for patterns, themes, styles, and design philosophies that can be reused to build applications. The Ant design system is an open source code for enterprise-level UI design languages and React UI library. It comes with a set of high-quality React components, with theme customization capability. It is built with i18n and has been localized for dozens of languages.

CHAPTER-6

PROJECT MODEL VIEW

6.1 Home



Fig 6.1

 Home

 Cryptocurrencies

 Exchanges

 News

SEC ramps up hiring to combat cryptocurrency fraud



The Securities and Exchange Commission is ramping up its fight against cryptocurrency-related fraud and cyber threats by expanding its staff...

Fox Business on MSN.com 3 hours ago

The Tale of a Crypto Executive Who Wasn't Who He Said He Was



He added that he had never bought cryptocurrency, let alone via Mr. Hannum. "I have never heard of him," he told me by email. "This is all a..."

New York Times 16 hours ago

Bentley University in Massachusetts now accepting cryptocurrency for tuition payments



half of americans think Cryptocurrency is here to stay. That's according to *** survey conducted by one poll for Stark. Net, *** platform de...

WCVB Channel 5 Boston 6 hours ago

Cryptoverse
All rights reserved.

[Home](#) [Cryptocurrencies](#) [News](#)

Fig 6.2

6.2 Cryptocurrency

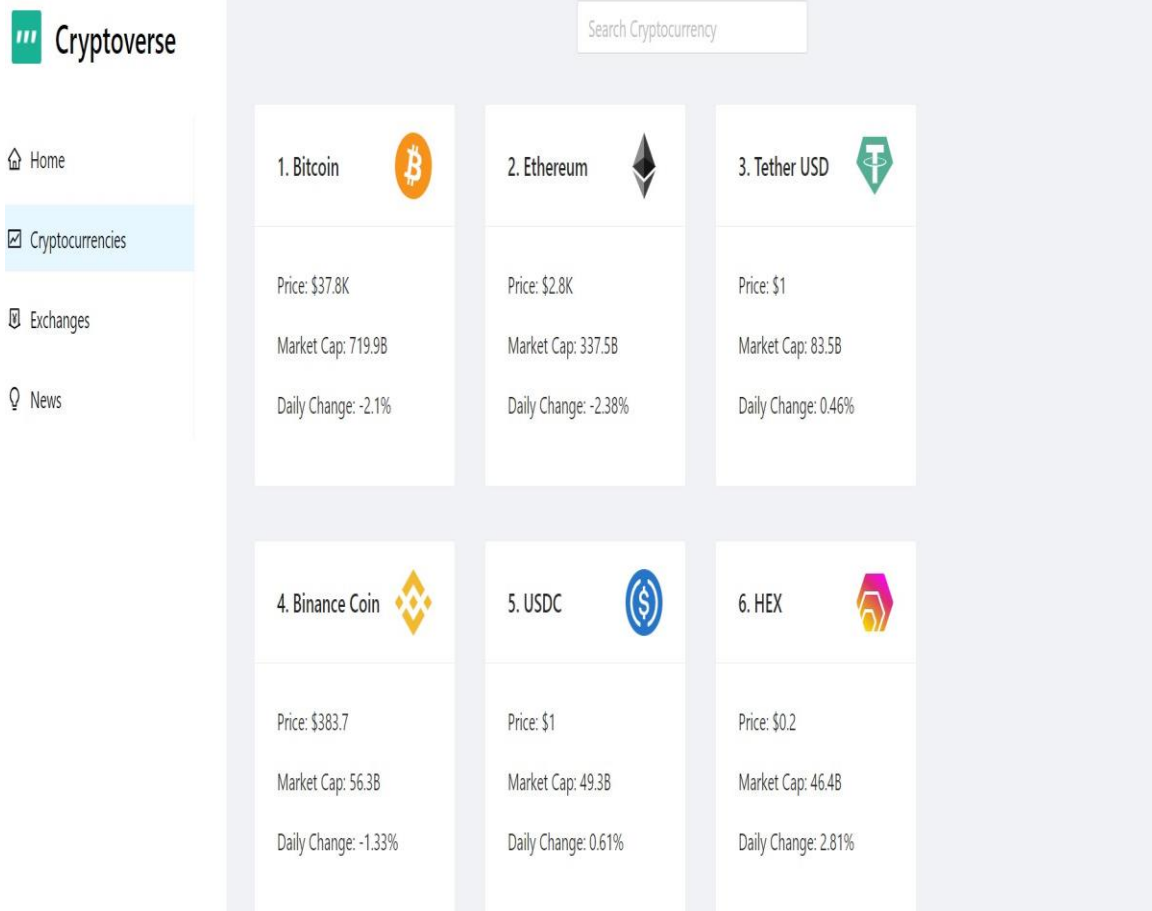


Fig 6.3

6.3 Searching Cryptocurrency

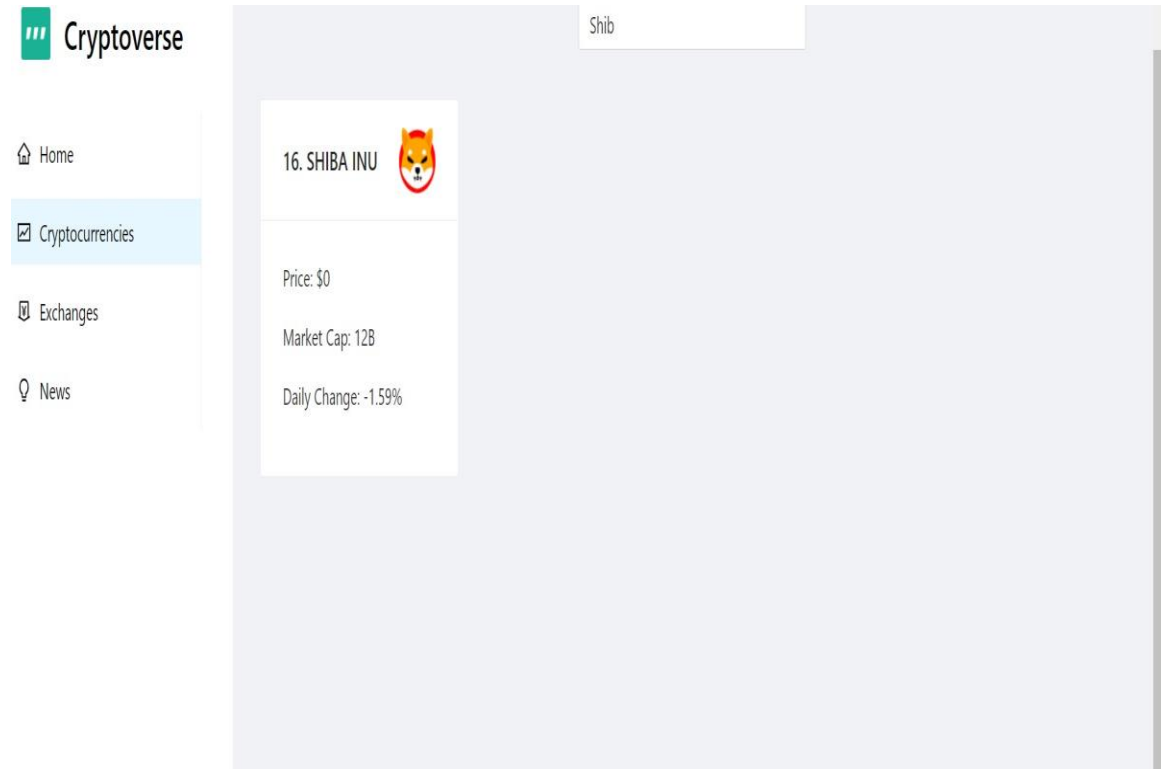


Fig 6.4

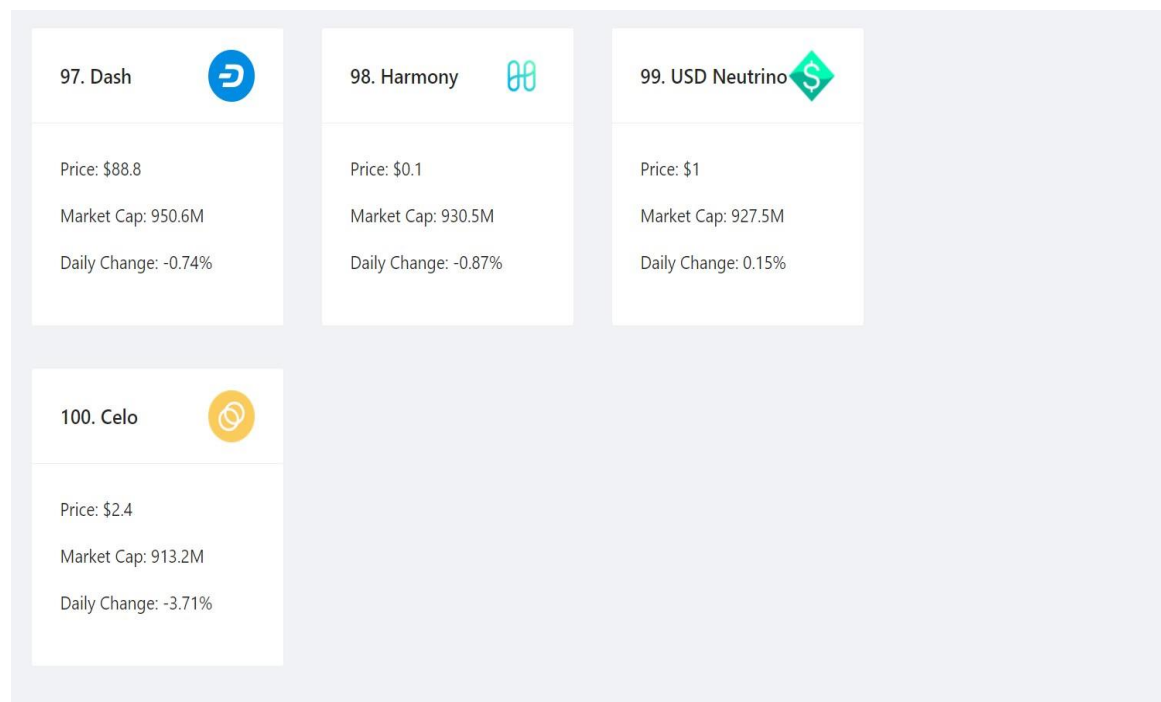


Fig 6.5

6.4 Cryptocurrency Live Price Chart

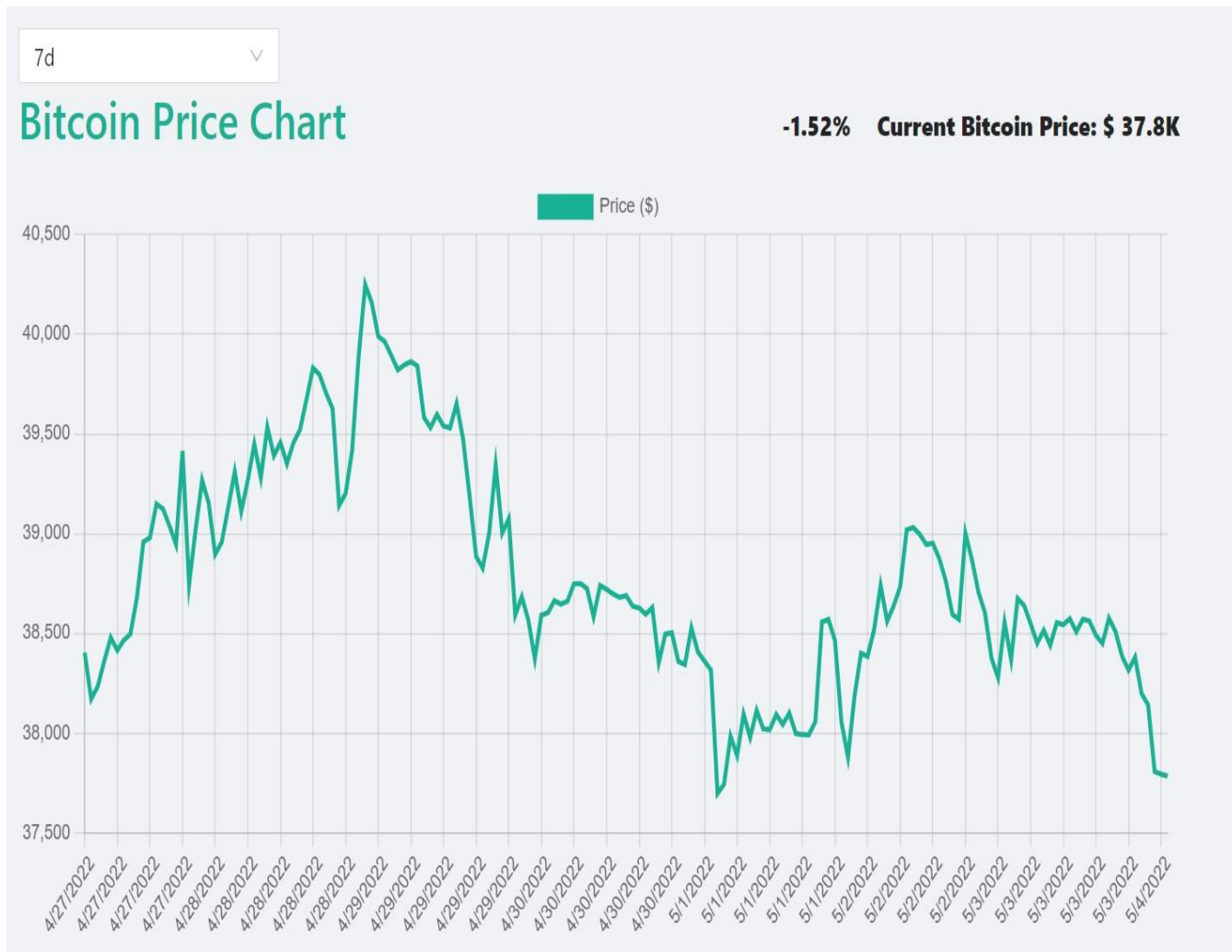


Fig 6.6

6.5 Cryptocurrency Statistics and other Information

Bitcoin Statistics		Other Stats Info	
An overview showing the statistics of Bitcoin, such as the base and quote currency, the rank, and All time high.		An overview showing more statistics of Bitcoin, such as the number of exchanges, the total and circulating supply.	
# Rank	1	📄 Number Of tickets	15173
💵 Price to USD	\$ 37.8K	🏛 Number Of Exchanges	215
💵 Market Cap	\$ 719.6B	⚠ Approved Supply	✓
⚡ 24h Volume	\$ 22B	⚠ Total Supply	19M
🏆 All time high	\$ 42.8K	⚠ Circulating Supply	19M

Fig 6.7

6.6 Cryptocurrency Details and Links to Related website


What is Bitcoin? <p>Bitcoin is the first digital currency that allows users to send and receive money, without the interference of a central bank or government. Instead, a network of thousands of peers is controlling the transactions; a decentralized system.</p> Why does bitcoin have value? <p>Bitcoin's useful qualities (decentralized, borderless, secure) aren't the only reason the coin is worth so much. Due to its scarcity (and it's hard to produce), Bitcoin is often nicknamed 'Digital Gold', in reference to 'classic' physical gold. Like gold, Bitcoin also has a finite supply of coins available; there will only ever be 21 million bitcoin. And now you know why the creation of new bitcoins is also called mining.</p> No inflation in bitcoin <p>Bitcoin was invented in response to a few concerns the inventor(s) had, such as inflation. Its supply is limited, so</p>	Bitcoin Links <table><tr><td>Website</td><td>bitcoin.org</td></tr><tr><td>Website</td><td>Bitcoin Whitepaper</td></tr><tr><td>Website</td><td>bitcoinmagazine.com</td></tr><tr><td>Bitcointalk</td><td>bitcointalk.org</td></tr><tr><td>Explorer</td><td>blockchain.com</td></tr><tr><td>Github</td><td>bitcoin/bitcoin</td></tr><tr><td>Reddit</td><td>r/bitcoin</td></tr></table>	Website	bitcoin.org	Website	Bitcoin Whitepaper	Website	bitcoinmagazine.com	Bitcointalk	bitcointalk.org	Explorer	blockchain.com	Github	bitcoin/bitcoin	Reddit	r/bitcoin
Website	bitcoin.org														
Website	Bitcoin Whitepaper														
Website	bitcoinmagazine.com														
Bitcointalk	bitcointalk.org														
Explorer	blockchain.com														
Github	bitcoin/bitcoin														
Reddit	r/bitcoin														

Fig 6.8

6.7 Cryptocurrency Latest News

Dogecoin

'Dogecoin' and Its Newly Made Value History as a Crypto




So, make a profit for many people like the moon! That's right; we're talking about Dogecoin, a cryptocurrency. Since mid-April, Dogecoin's p...

Deadline News

13 hours ago

Dogecoin Google searches peaked on Elon Musk's Twitter purchase




Online searches for 'Buy Dogecoin' shot up by almost 400 per cent on the day Elon Musk – one of the altcoin's biggest evangelists – bought s...

City A.M.

12 hours ago

Dogecoin price could fall to \$0.11 as the crypto world anticipates the upcoming FOMC meeting




Dogecoin price has investors on edge as the bulls have gone missing to start the month of May. Analysts have maintained that the Dogecoin pr...


FXStreet

an hour ago

Who Accepts Dogecoin as Payment?



Dogecoin Daily: Fed Meeting Weighs On Price, Dogefather



Dogecoin price provides buying opportunity before




Fig 6.9

18

CHAPTER-7

CONCLUSION AND FUTURE ENHANCEMENT

7.1 Conclusions

The project is only a humble venture to satisfy the needs in a user. Several user-friendly coding have also been adopted. This package shall prove to be a powerful package in satisfying all the requirements. This website provides a user-friendly version of cryptocurrency tracking. One need not to go through any user guide.

It provides real-time price details , how it has been behaving in days or even up to 5 years. There is also a news section provided where anyone can easily fetch the latest news. Searching option is provided in both cryptocurrencies and news section so that one can look upon his desired cryptocurrencies.

7.2 Future Scope

As cryptocurrency has huge potential and a process of upgradation similarly in that way our project has a huge potential to grow up. As now-a-days everything has been started working with web 3.0. We can also upgrade it to web 3.0 coping with decentralized app. We can also add Ethereum wallet or Meta-mask wallet so we don't have to use 3rd party crypto exchange process.

We can add all the exchanges done by all over the across here like through Binance or Coinbase. We can also add top losers and top gainers of the market so people looking for short-term investment can trade on the respected cryptos which are giving high return investment in short term.

We can also the prediction of certain cryptocurrencies in the upcoming years. This is a great idea if somehow it will predict the nearly accurate results people looking for investment for few years can rely on the data that shows by the predictor.

Following are the future scope:

- Backend can be added to make more user-friendly.
- Easily add wallets of Ethereum and Meta mask.
- Can be upgraded to web 3.0 to competing any modern website.
- NFT can also be added .
- Own cryptos exchange be added through web 3.0.
- Predictor can be added using ML.
- Can make it more robust.

APPENDIX-A SOURCE CODE

- **Index.JS**

```
import React from "react";
import ReactDOM from "react-dom";
import { BrowserRouter as Router } from "react-router-dom"; // In order to use links we
need to wrap our app with react-router.
import { Provider } from "react-redux";

import App from "./App";
import store from "./app/store";
import "antd/dist/antd.css";

ReactDOM.render(
  <React.StrictMode>
    <Router>
      <Provider store={store}>
        <App />
      </Provider>
    </Router>
  ,
  </React.StrictMode>,
  document.getElementById("root")
);
```


- **App.JS**

```
import React from "react";
import { Switch, Route, Link } from "react-router-dom";
import { Layout, Typography, Space } from "antd";
```

```
import {
  Navbar,
  Homepage,
  Cryptocurrencies,
  News,
  CryptoDetails,
} from "../components";
import "../App.css";
```

```
const App = () => {
  return (
    <div className="app">
      <div className="navbar">
        <Navbar />
      </div>
      <div className="main">
        <Layout>
          <div className="routes">
            <Switch>
              <Route exact path="/">
                <Homepage />
              </Route>
              <Route exact path="/cryptocurrencies">
                <Cryptocurrencies />
              </Route>
            </Switch>
          </div>
        </Layout>
      </div>
    </div>
  );
};
```

```

    </Route>
    <Route exact path="/crypto/:coinId">
      <CryptoDetails />
    </Route>
    <Route exact path="/news">
      <News />
    </Route>
  </Switch>
</div>
</Layout>
<div className="footer">
  <Typography.Title
    level={5}
    style={{ color: "black", textAlign: "center" }}
  >
    Cryptoverse
    <br />
    All rights reserved.
  </Typography.Title>
  <Space>
    <Link style={{ color: "#21A698" }} to="/">
      Home
    </Link>
    <Link style={{ color: "#21A698" }} to="/cryptocurrencies">
      Cryptocurrencies
    </Link>
    <Link style={{ color: "#21A698" }} to="/news">
      News
    </Link>
  </Space>
</div>

```

```

    </div>
  </div>
);
};

export default App;

```

- **CryptoApi**

```

import { createApi, fetchBaseQuery } from "@reduxjs/toolkit/query/react";

const cryptoApiHeaders = {
  "x-rapidapi-host": "coinranking1.p.rapidapi.com",
  "x-rapidapi-key": "5d814050a4mshfde5600c2696f56p1ce79fjsncb44eae09d34",
};

const baseUrl = "https://coinranking1.p.rapidapi.com";

const createRequest = (url) => ({ url, headers: cryptoApiHeaders });

export const cryptoApi = createApi({
  reducerPath: "cryptoApi",
  baseQuery: fetchBaseQuery({ baseUrl }),
  endpoints: (builder) => ({
    getCryptos: builder.query({
      query: (count) => createRequest(`/coins?limit=${count}`),
    }),
    // Note: To access this endpoint you need premium plan
    getExchanges: builder.query({
      query: () => createRequest('/exchanges'),
    }),
  }),
});

```

```

getCryptoDetails: builder.query({
  query: (coinId) => createRequest(`/coin/${coinId}`),
}),
getCryptoHistory: builder.query({
  query: ({ coinId, timeperiod }) =>
    createRequest(`coin/${coinId}/history?timePeriod=${timeperiod}`),
}),

}),
});

export const {
  useGetCryptosQuery,
  useGetCryptoDetailsQuery,
  useGetExchangesQuery,
  useGetCryptoHistoryQuery,
} = cryptoApi;

```

- **CryptoNewsAPI**

```

import { createApi, fetchBaseQuery } from "@reduxjs/toolkit/query/react";

const cryptoNewsHeaders = {
  "x-bingapis-sdk": "true",
  "x-rapidapi-host": "bing-news-search1.p.rapidapi.com",
  "x-rapidapi-key": "5d814050a4mshfde5600c2696f56p1ce79fjsncb44eae09d34",
};

const baseUrl = "https://bing-news-search1.p.rapidapi.com";

```

```

const createRequest = (url) => ({ url, headers: cryptoNewsHeaders });

export const cryptoNewsApi = createApi({
  reducerPath: "cryptoNewsApi",
  baseQuery: fetchBaseQuery({ baseUrl }),
  endpoints: (builder) => ({
    getCryptoNews: builder.query({
      query: ({ newsCategory, count }) =>
        createRequest(
          `/news/search?q=${newsCategory}&safeSearch=Off&textFormat=Raw&freshness=Day
          &count=${count}`
        ),
    }),
  }),
});

export const { useGetCryptoNewsQuery } = cryptoNewsApi;

```

- **Store.JS**

```

import { configureStore } from "@reduxjs/toolkit";

import { cryptoApi } from "../services/cryptoApi";
import { cryptoNewsApi } from "../services/cryptoNewsApi";

```

```

export default configureStore({
  reducer: {
    [cryptoApi.reducerPath]: cryptoApi.reducer,
    [cryptoNewsApi.reducerPath]: cryptoNewsApi.reducer,
  },
  // Adding the api middleware enables caching, invalidation, polling,
  // and other useful features of `rtk-query`.
  middleware: (getDefaultMiddleware) =>
    getDefaultMiddleware().concat(
      cryptoApi.middleware,
      cryptoNewsApi.middleware
    ),
});

```

- **HomePage.JSX**

```

import React from "react";
import millify from "millify"; // To format our numbers.
import { Typography, Row, Col, Statistic } from "antd";
import { Link } from "react-router-dom";
import { useGetCryptosQuery } from "../services/cryptoApi";
import { Cryptocurrencies, News } from "../components";
import Loader from "./Loader";

const { Title } = Typography;

const Homepage = () => {
  const { data, isFetching } = useGetCryptosQuery(10); // Declare our custom hook that
  we will use to dynamically display our data.
  const globalStats = data?.data?.stats;

```

```
if (isFetching) return <Loader />;
```

```
return (
```

```
  <>
```

```
    <Title level={2} className="heading">
```

```
      Global Crypto Stats
```

```
    </Title>
```

```
    <Row>
```

```
      <Col span={12}>
```

```
        <Statistic title="Total Cryptocurrencies" value={globalStats.total} />
```

```
      </Col>
```

```
      <Col span={12}>
```

```
        <Statistic
```

```
          title="Total Exchanges"
```

```
          value={millify(globalStats.totalExchanges)}
```

```
        />
```

```
      </Col>
```

```
      <Col span={12}>
```

```
        <Statistic
```

```
          title="Total Market Cap"
```

```
          value={millify(globalStats.totalMarketCap)}
```

```
        />
```

```
      </Col>
```

```
      <Col span={12}>
```

```
        <Statistic
```

```
          title="Total 24h Volume"
```

```
          value={millify(globalStats.total24hVolume)}
```

```
        />
```

```
      </Col>
```

```
    </Row>
```

```
    <div className="home-heading-container">
```

```

    <Title level={2} className="home-title">
      Top 10 Cryptocurrencies
    </Title>
    <Title level={3} className="show-more">
      <Link style={{ color: "#21A698" }} to="/cryptocurrencies">
        Show more
      </Link>
    </Title>
  </div>
  <Cryptocurrencies simplified />
  <div className="home-heading-container">
    <Title level={2} className="home-title">
      Latest Crypto News
    </Title>
    <Title level={3} className="show-more">
      <Link style={{ color: "#21A698" }} to="/news">
        Show more
      </Link>
    </Title>
  </div>
  <News simplified />
</>
);
};

export default Homepage;

```


- **Cryptocurrencies.JSX**

```
import React, { useEffect, useState } from "react";
import millify from "millify";
import { Link } from "react-router-dom";
import { Card, Row, Col, Input } from "antd";

import { useGetCryptosQuery } from "../services/cryptoApi";

import Loader from './Loader';

const Cryptocurrencies = ({ simplified }) => {
  const count = simplified ? 10 : 100;
  const { data: cryptosList, isFetching } = useGetCryptosQuery(count);
  const [cryptos, setCryptos] = useState();
  const [searchTerm, setSearchTerm] = useState("");

  useEffect(() => {
    setCryptos(cryptosList?.data?.coins);

    const filteredData = cryptosList?.data?.coins.filter((item) =>
      item.name.toLowerCase().includes(searchTerm)
    );

    setCryptos(filteredData);
  }, [cryptosList, searchTerm]);

  if (isFetching) return <Loader />;

  return (
```

◇

```
{!simplified && (  
  <div className="search-crypto">  
    <Input  
      placeholder="Search Cryptocurrency"  
      onChange={(e) => setSearchTerm(e.target.value.toLowerCase())}  
    />  
  </div>  
)}  
<Row gutter={[32, 32]} className="crypto-card-container">  
  {cryptos?.map((currency) => (  
    <Col  
      xs={24}  
      sm={12}  
      lg={6}  
      className="crypto-card"  
      key={currency.uuid}  
    >  
      <Link key={currency.uuid} to={`/${currency.uuid}`}>  
        <Card  
          title={` ${currency.rank}. ${currency.name} `}  
          extra={  
            <img className="crypto-image" src={currency.iconUrl} alt="" />  
          }  
          hoverable  
        >  
          <p>Price: ${millify(currency.price)}</p>  
          <p>Market Cap: {millify(currency.marketCap)}</p>  
          <p>Daily Change: {currency.change}%</p>  
        </Card>  
      </Link>  
    )  
  )  
}
```

```

        </Col>
      )))
    </Row>
  </>
);
};

export default Cryptocurrencies;

```

- **CryptoDetails.JSX**

```

import React, { useState } from "react";
import HTMLReactParser from "html-react-parser";
import { useParams } from "react-router-dom";
import millify from "millify";
import { Typography, Row, Col, Select } from "antd";
import {
  MoneyCollectOutlined,
  DollarCircleOutlined,
  ExclamationCircleOutlined,
  ThunderboltOutlined,
  StopOutlined,
  TrophyOutlined,
  BankOutlined,
  CheckOutlined,
  NumberOutlined,
} from "@ant-design/icons";

import {

```

```

    useGetCryptoDetailsQuery,
    useGetCryptoHistoryQuery,
  } from "../services/cryptoApi";

import Loader from './Loader';

import LineChart from "../LineChart";

const { Title, Text } = Typography;
const { Option } = Select;

const CryptoDetails = () => {
  const { coinId } = useParams();
  const [timeperiod, setTimeperiod] = useState("7d");
  const { data, isFetching } = useGetCryptoDetailsQuery(coinId);
  const { data: coinHistory } = useGetCryptoHistoryQuery({
    coinId,
    timeperiod,
  });
  const cryptoDetails = data?.data?.coin;

  console.log("cryptoDetails: ", cryptoDetails);

  if (isFetching) return <Loader />;

  const time = ["3h", "24h", "7d", "30d", "3m", "1y", "3y", "5y"];

  const stats = [
    { title: "Rank", value: cryptoDetails?.rank, icon: <NumberOutlined /> },
    {
      title: "Price to USD",

```

```

    value: `\$ ${cryptoDetails?.price && millify(cryptoDetails?.price)}`,
    icon: <DollarCircleOutlined />,
  },
  {
    title: "Market Cap",
    value: `\$ ${
      cryptoDetails?.marketCap && millify(cryptoDetails?.marketCap)
    }`,
    icon: <DollarCircleOutlined />,
  },
  {
    title: "24h Volume",
    value: `\$ ${
      cryptoDetails["24hVolume"] && millify(cryptoDetails["24hVolume"])
    }`,
    icon: <ThunderboltOutlined />,
  },

  {
    title: "All time high",
    value: `\$ ${
      cryptoDetails?.allTimeHigh?.price &&
      millify(cryptoDetails?.allTimeHigh?.price)
    }`,
    icon: <TrophyOutlined />,
  },
];

const genericStats = [
  {
    title: "Number Of tickets ",

```

```

    value: cryptoDetails?.numberOfMarkets,
    icon: <MoneyCollectOutlined />,
  },
  {
    title: "Number Of Exchanges",
    value: cryptoDetails?.numberOfExchanges,
    icon: <BankOutlined />,
  },
  {
    title: "Approved Supply",
    value: cryptoDetails?.supply?.confirmed ? (
      <CheckOutlined />
    ) : (
      <StopOutlined />
    ),
    icon: <ExclamationCircleOutlined />,
  },
  {
    title: "Total Supply",
    value: `${
      cryptoDetails?.supply?.total && millify(cryptoDetails?.supply?.total)
    }`,
    icon: <ExclamationCircleOutlined />,
  },
  {
    title: "Circulating Supply",
    value: `${
      cryptoDetails?.supply?.circulating &&
      millify(cryptoDetails?.supply?.circulating)
    }`,
    icon: <ExclamationCircleOutlined />,
  },

```

```

    },
  ];

  return (
    <Col className="coin-detail-container">
      <Col className="coin-heading-container">
        <Title level={2} className="coin-name">
          {data?.data?.coin.name} ({data?.data?.coin.symbol}) Price
        </Title>
        <p>
          {cryptoDetails.name} live price in US Dollar (USD). View value
          statistics, market cap and supply.
        </p>
      </Col>
      <Select
        defaultValue="7d"
        className="select-timeperiod"
        placeholder="Select Timeperiod"
        onChange={(value) => setTimeperiod(value)}
      >
        {time.map((date) => (
          <Option key={date}>{date}</Option>
        ))}
      </Select>
      <LineChart
        coinHistory={coinHistory}
        currentPrice={millify(cryptoDetails.price)}
        coinName={cryptoDetails.name}
      />
    <Col className="stats-container">
      <Col className="coin-value-statistics">

```

```

<Col className="coin-value-statistics-heading">
  <Title level={3} className="coin-details-heading">
    {cryptoDetails.name} Statistics
  </Title>
  <p>
    An overview showing the statistics of {cryptoDetails.name}, such
    as the base and quote currency, the rank, and All time high.
  </p>
</Col>
{stats.map(({ icon, title, value }) => (
  <Col className="coin-stats">
    <Col className="coin-stats-name">
      <Text>{icon}</Text>
      <Text>{title}</Text>
    </Col>
    <Text className="stats">{value}</Text>
  </Col>
))}
</Col>
<Col className="other-stats-info">
  <Col className="coin-value-statistics-heading">
    <Title level={3} className="coin-details-heading">
      Other Stats Info
    </Title>
    <p>
      An overview showing more statistics of {cryptoDetails.name}, such
      as the number of exchanges, the total and circulating supply.
    </p>
  </Col>
  {genericStats.map(({ icon, title, value }) => (
    <Col className="coin-stats">

```



```

    <Col className="coin-stats-name">
      <Text>{icon}</Text>
      <Text>{title}</Text>
    </Col>
    <Text className="stats">{value}</Text>
  </Col>
  )})
</Col>
</Col>
<Col className="coin-desc-link">
  <Row className="coin-desc">
    <Title level={3} className="coin-details-heading">
      What is {cryptoDetails.name}?
    </Title>
    {HTMLReactParser(cryptoDetails.description)}
  </Row>
  <Col className="coin-links">
    <Title level={3} className="coin-details-heading">
      {cryptoDetails.name} Links
    </Title>
    {cryptoDetails.links?.map((link) => (
      <Row className="coin-link" key={link.name}>
        <Title level={5} className="link-name">
          {link.type}
        </Title>
        <a href={link.url} target="_blank" rel="noreferrer">
          {link.name}
        </a>
      </Row>
    ))}
  </Col>

```

```

        </Col>
      </Col>
    );
  };

export default CryptoDetails;

```

- **Exchanges.JSX**

```

import React from 'react';
import millify from 'millify';
import { Collapse, Row, Col, Typography, Avatar } from 'antd';
import HTMLReactParser from 'html-react-parser';

import { useGetExchangesQuery } from '../services/cryptoApi';
import Loader from './Loader';

const { Text } = Typography;
const { Panel } = Collapse;

const Exchanges = () => {
  const { data, isFetching } = useGetExchangesQuery();
  const exchangesList = data?.data?.exchanges;
  // Note: To access this endpoint you need premium plan
  if (isFetching) return <Loader />;

  return (
    <
      <Row>

```

```

    <Col span={6}>Exchanges</Col>
    <Col span={6}>24h Trade Volume</Col>
    <Col span={6}>Markets</Col>
    <Col span={6}>Change</Col>
  </Row>
  <Row>
    { /*{exchangesList.map((exchange) => (
      <Col span={24}>
        <Collapse>
          <Panel
            key={exchange.uuid}
            showArrow={false}
            header={{
              <Row key={exchange.uuid}>
                <Col span={6}>
                  <Text><strong>{exchange.rank}</strong></Text>
                  <Avatar className="exchange-image" src={exchange.iconUrl} />
                  <Text><strong>{exchange.name}</strong></Text>
                </Col>
                <Col span={6}>${millify(exchange.volume)}</Col>
                <Col span={6}>{millify(exchange.numberOfMarkets)}</Col>
                <Col span={6}>{millify(exchange.marketShare)}%</Col>
              </Row>
            }}
          >
            {HTMLReactParser(exchange.description || "")}
          </Panel>
        </Collapse>
      </Col>
    )} */ }
  </Row>

```

```

    </>
  );
};

export default Exchanges;

```

- **LineChart.JSX**

```

import React from "react";
import { Line } from "react-chartjs-2";
import { Col, Row, Typography } from "antd";

```

```

import {
  Chart as ChartJS,
  CategoryScale,
  LinearScale,
  PointElement,
  LineElement,
  Tooltip,
  Legend,
} from "chart.js";

```

```

ChartJS.register(
  CategoryScale,
  LinearScale,
  PointElement,
  LineElement,
  Tooltip,
  Legend
);

```

```
);
```

```
const { Title } = Typography;
```

```
const LineChart = ({ coinHistory, currentPrice, coinName }) => {
```

```
  const coinPrice = [];
```

```
  const coinTimestamp = [];
```

```
  const len = coinHistory?.data?.history?.length;
```

```
  const yLen = coinHistory?.data?.history?.length;
```

```
  for (let i = len - 1; i > 0; i--) {
```

```
    coinPrice.push(coinHistory?.data?.history[i].price);
```

```
  }
```

```
  for (let i = yLen - 1; i > 0; i--) {
```

```
    coinTimestamp.push(
```

```
      new Date(
```

```
        coinHistory?.data?.history[i].timestamp * 1000
```

```
      ).toLocaleDateString()
```

```
    );
```

```
  }
```

```
  const data = {
```

```
    labels: coinTimestamp,
```

```
    datasets: [
```

```
      {
```

```
        label: "Price ($)",
```

```
        data: coinPrice,
```

```
        fill: false,
```

```
        backgroundColor: "#19b294",
```

```
        borderColor: "#19b294",
```

```
        pointRadius: 0,
```

```

    },
  ],
};

```

```

const options = {
  scales: {
    yAxes: [
      {
        ticks: {
          beginAtZero: true,
        },
      },
    ],
  },
};

```

```

return (
  <◇
    <Row className="chart-header">
      <Title level={2} className="chart-title">
        {coinName} Price Chart{" "}
      </Title>
      <Col className="price-container">
        <Title level={5} className="price-change">
          {coinHistory?.data?.change}%
        </Title>
        <Title level={5} className="current-price">
          Current {coinName} Price: $ {currentPrice}
        </Title>
      </Col>
    </Row>
  </◇
)

```

```

    <Line data={data} options={options} />
  </>
);
};

```

```
export default LineChart;
```

- **Loader.JSX**

```
import React from 'react';
import { Spin } from 'antd';
```

```
const Loader = () => (
  <div className="loader">
    <Spin />
  </div>
);
```

```
export default Loader;
```

- **Index.JS**

```
export { default as Homepage } from "./Homepage";
export { default as Navbar } from "./Navbar";
export { default as News } from "./News";
export { default as Cryptocurrencies } from "./Cryptocurrencies";
export { default as CryptoDetails } from "./CryptoDetails";
export { default as Exchanges } from "./Exchanges";
```

- **Navbar.JSX**

```
import React, { useState, useEffect } from "react";
import { Button, Menu, Typography, Avatar } from "antd";
import { Link } from "react-router-dom";
import {
  HomeOutlined,
  FundOutlined,
  MoneyCollectOutlined,
  MenuOutlined,
  BulbOutlined,
} from "@ant-design/icons";

import icon from "../images/3commas_logo.jpg";

const Navbar = () => {
  const [activeMenu, setActiveMenu] = useState(true);
  const [screenSize, setScreenSize] = useState(null);

  useEffect(() => {
    const handleResize = () => setScreenSize(window.innerWidth);
    window.addEventListener("resize", handleResize);
    handleResize();
    return () => window.removeEventListener("resize", handleResize);
  }, []);

  useEffect(() => {
    if (screenSize < 768) {
      setActiveMenu(false);
    }
  }, [screenSize]);
}
```



```

    } else {
      setActiveMenu(true);
    }
  }, [screensize]);

return (
  <div className="nav-container">
    <div className="logo-container">
      <Avatar
        shape="square"
        src={icon}
        style={{ verticalAlign: "middle" }}
        size="medium"
      />
      <Typography.Title level={3} className="logo">
        <Link style={{ color: "#000000" }} to="/">
          Cryptoverse
        </Link>
      </Typography.Title>
      <Button
        className="menu-control-container"
        style={{
          background: "#21A698",
          paddingTop: "0px",
          marginLeft: "2px",
        }}
        onClick={() => setActiveMenu(!activeMenu)}
      >
        <MenuOutlined />
      </Button>{ " " }
    </div>
  </div>

```

```

{activeMenu && (
  <Menu style={{ background: "#fff" }} theme="light">
    <Menu.Item
      style={{ color: "#000000" }}
      key="1"
      icon={<HomeOutlined />}
    >
      <Link style={{ color: "#000000" }} to="/">
        Home
      </Link>
    </Menu.Item>
    <Menu.Item
      style={{ color: "#000000" }}
      key="2"
      icon={<FundOutlined />}
    >
      <Link style={{ color: "#000000" }} to="/cryptocurrencies">
        Cryptocurrencies
      </Link>
    </Menu.Item>
    <Menu.Item
      style={{ color: "#000000" }}
      key="1"
      icon={<MoneyCollectOutlined />}
    >
      <Link style={{ color: "#000000" }} to="/exchanges">
        Exchanges
      </Link>
    </Menu.Item>
    <Menu.Item
      style={{ color: "#000000" }}

```

```

        key="3"
        icon={<BulbOutlined />}
      >
        <Link style={{ color: "#000000" }} to="/news">
          News
        </Link>
      </Menu.Item>
    </Menu>
  )}
</div>
);
};

```

```
export default Navbar;
```

- **News.JSX**

```

import React, { useState } from "react";
import { Select, Typography, Row, Col, Card } from "antd";
import moment from "moment";

import Loader from "../Loader";

import { useGetCryptosQuery } from "../../services/cryptoApi";
import { useGetCryptoNewsQuery } from "../../services/cryptoNewsApi";

const { Text, Title } = Typography;
const { Option } = Select;

```

```

const News = ({ simplified }) => {
  const [newsCategory, setNewsCategory] = useState("Cryptocurrency");
  const { data } = useGetCryptosQuery(100);
  const { data: cryptoNews } = useGetCryptoNewsQuery({
    newsCategory,
    count: simplified ? 6 : 18,
  });

  if (!cryptoNews?.value) return <Loader />;

  return (
    <Row gutter={[24, 24]}>
      {!simplified && (
        <Col span={24}>
          <Select
            showSearch
            className="select-news"
            placeholder="Enter your crypto here..."
            optionFilterProp="children"
            onChange={(value) => setNewsCategory(value)}
            filterOption={(input, option) =>
              option.children.toLowerCase().indexOf(input.toLowerCase()) >= 0
            }
          >
            <Option value="Cryptocurrency">Cryptocurrency</Option>
            {data?.data?.coins?.map((currency) => (
              <Option value={currency.name}>{currency.name}</Option>
            ))}
          </Select>
        </Col>
      )}
    >
  )
}

```

```

{cryptoNews.value.map((news, i) => (
  <Col xs={24} sm={12} lg={8} key={i}>
    <Card hoverable className="news-card">
      <a href={news.url} target="_blank" rel="noreferrer">
        <div className="news-image-container">
          <Title className="news-title" level={5}>
            {news.name.length > 100
              ? `${news.name.substring(0, 100)}...`
              : news.name}
          </Title>
          <img src={news?.image?.thumbnail?.contentUrl} alt="" key={i} />
        </div>
        <p>
          {news.description.length > 140
            ? `${news.description.substring(0, 140)}...`
            : news.description}
        </p>
        <div className="provider-container">
          <div>
            <Text className="provider-name">
              {news.provider[0]?.name}
            </Text>
          </div>
          <Text>
            {moment(news.datePublished).startOf("ss").fromNow()}
          </Text>
        </div>
      </a>
    </Card>
  </Col>
)})

```

```
    </Row>
  );
};

export default News;
```