CRYPTOTRAK

A Project Report

Submitted in partial fulfillment of the
Requirements for the award of the Degree of

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)

By

Name of The Student: Patil Abhishek Narayan
Seat Number: 8914 Division B

Under the esteemed guidance of

PROF – Mr. Omkar Sherkhane

DEPARTMENT OF INFORMATION TECHNOLOGY
Mahatma Education Society's
Pillai College of Arts Commerce & Science (Autonomous), New Panvel
(Affiliated to University of Mumbai)
PANVEL 410206
MAHARASHTRA
YEAR 2021-2022

Mahatma Education Society's

Pillai College of Arts Commerce & Science (Autonomous), New Panvel

(Affiliated to University of Mumbai

NEW PANVEL MAHARASHTRA 410206

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled. "**CryptoTrak**", is bonafied work of **PATIL ABHISHEK NARAYAN** bearing SeatNo.: (8914) submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATIONTECHNOLOGY from University of Mumbai.

Internal Guide **Coordinator**

External Examiner

Date: **College Seal**

ACKNOWLEDGEMENT

I Mr. Patil Abhishek Narayan student of PILLAI COLLEGE OF ARTS, COMMERCE & SCIENCE, NEW PANVEL would like to express my sincere gratitude towards our college's Information Technology Department.

I would like to thank Mrs. Deepika Sharma H.O.D. (I.T. & C.S.) for granting me the opportunity to build project for the college. Last but not least I thank our Lecturer (Mr. Omkar Sherkhane) for his constant support during this project. The project would have not been completed without the dedication, creativity and the enthusiasm my family provided me.

Yours faithfully,

Patil Abhishek Narayan

DECLARATION

I hereby declare that the project entitled, “**CryptoTrack**” done at **Pillai College, New Panvel**, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY) to be submitted as final semester project as part of our curriculum.

Name and Signature of the Student

TABLE OF CONTENTS

Chapter 1
Introduction

CryptoTrak is a website about cryptocurrency. Here you can track any cryptocurrency, anything from bitcoin to dogecoin. Also, in this website you can get latest news about cryptocurrency, calculate your portfolio and use graphs to track the cryptocurrency.

The main advantage of this website is that all the data, from news to the price of the cryptocurrency is live and updated through an API. This website was created for all the crypto lovers who can always be updated on their favorite cryptocurrency.

As everything has became online from purchases to movie streaming. Privacy is the main concern, here cryptocurrency comes in picture. Cryptocurrencies hold the promise of making it easier to transfer funds directly between two parties, without the need for a trusted third party like a bank or credit card company. These transfers are instead secured by the use of public keys and private keys and different forms of incentive systems, like Proof of Work or Proof of Stake. So, we created this website to track cryptocurrency.

Background

Cryptocurrency is a form of payment that can be exchanged online for goods and services. Many companies have issued their own currencies, often called tokens, and these can be traded specifically for the good or service that the company provides. You’ll need to exchange real currency for the cryptocurrency to access the good or service.

Cryptocurrencies work using a technology called blockchain.
Blockchain is a decentralized technology spread across many computers that manages and records transactions. Part of the appeal of this technology is its security.

More than 10,000 different cryptocurrencies are traded publicly, according to CoinMarketCap.com, a market research website.

And cryptocurrencies continue to proliferate, raising money through initial coin offerings, or ICOs. The total value of all cryptocurrencies on Aug. 18, 2021, was more than $1.9 trillion.

Supporters see cryptocurrencies such as Bitcoin as the currency of the future and are racing to buy them now, presumably before they become more valuable

Bitcoin is by far the most popular cryptocurrency, followed by other cryptocurrencies such as Ethereum, Litecoin, and Cardano.

Objective

- To track Cryptocurrency information.
- To give live of the desired cryptocurrency updates to the user.
- To get proper feedback from the user and work on it.
- Display live news about cryptocurrency.
- Provide a good user interface and user experience.
- Maintaining proper record of the users.
- Providing User tool that can convert a cryptocurrency to any other currency
- Keep the website updated.
- Consistently using the data of API.
- Using proper navigation.
- Trying to eliminate data redundancy.
- The customer can search their favourite cryptocurrency.
- Customer can track price ups and downs details using graphs.
- Create and maintain crypto portfolio.
- Have proper privacy for user data.
- Update portfolio according to the crypto price ups and downs.

Purpose

The purpose of cryptotrak is to monitor the current value of different digital currencies and tokens. This website provides previous data as well, allowing users to compare current prices against older values. The customer can easily track their favourite cryptocurrency using this app and also, they can maintain their portfolio here which will be updated in real time. The data of the portfolio and other things like queries and messages from the users will be saved in a database which will be accessible always to the admins.

Cryptotrak provide users or investors with a whole range of information, including daily trading prices, market capitalization, trading volume, and price charts. Cryptotrak will provide live news of the all the famous cryptocurrencies using a news API.

To provide good user-friendly experience. This website uses proper UI UX functions which makes it easier to use at any time.

Graphs will be provided for each cryptocurrency, from which users can track the previous prices ups and downs of cryptocurrency, from which they can make proper decision for their investment in crypto.

Cryptocurrency price trackers should use reliable data and consistently update their data.

To display data of cryptocurrency like market cap, price, 24-hour trading volume, the circulating supply, the percentage change in value over the previous 24 hours, and a seven-day price graph. The graphs will be user friendly and while be designed using proper JavaScript library.

To provide a currency converter that can be used for example: converting the bitcoin price to rupees or dollars etc.

Scope

The scope of this website is very high. In this modern world of technology, everything is going digital and so is currencies. So, to track this so-called digital currency this kind of tracking website is very useful. The scope of this website is to track cryptocurrency 24x7.

Also, user can maintain their portfolio and convert their cryptocurrency price into regular currency price.

The data that will be provide of a cryptocurrency will be extracted from an API, which makes the data live. Everything from trading prices, market capitalization, trading volume, and price charts will be provided by the API.

New changes like adding additional features like bulls and bears graphs, proper search engine can be added to the website.

Applicability

This software is applicable in the field of Blockchain. It helps people to gain

information about the latest products of technology. It promotes the use of latest technology

and in turn helps in the development of the country.

Chapter 2

System Planning

SURVEY OF TECHNOLOGIES

Frontend:

HTML:

HTML is a programming language that stands for Hypertext Markup Language**.** This is a relatively simplistic yet powerful programming language that allows web develops and website owners to create the structure of their website. If you want a sentence in a blog post to receive a bold font, HTML could be used to achieve this. When using this system, any text files can be tagged for effects, graphics, fonts, or colours. If you've never used any kind of programming language in the past, HTML would be a great place to start. Keep in mind that HTML markups are enhanced by CSS and JavaScript, which is why it's recommended that you learn this programming language first

What Can You Do With HTML?

The main reason why it's important to know HTML is because it allows you as a website owner to create the basic structure of you website, from which you can use any other coding language to enhance the design.

- With a basic understanding of HTML, you can significantly increase the control and flexibility that you have with your website.
- You'll be able to make your own changes by having knowledge of this code.
- Some platforms will automatically implement many of the more complex functions of a website design, which leaves you with the ability to tend to the minor changes that can be made with HTML.

CSS:

CSS is a style sheet language that's focused almost entirely on improving the presentation of HTML elements. While CSS is a static programming language, it can be used to make your website appear visually pleasing and modern**.** If you want your website to have a sleek and stylish design, CSS is the programming language to use. With a basic understanding of CSS, you can make changes to the overall look and feel of your website, which is useful when you want to match the appearance of your website with the aesthetics of your brand.

What Can You Do With CSS?

Even though CSS is primarily used to create static visual effects, the latest versions of the programming language offer some functionality for the creation of very simple animations.

- If you want to change the colour of some of the text on a web page, basic CSS would allow you to do so by adding "style" code to the HTML code that's already found on your page.
- Along with the presentation of HTML, CSS can also be used to alter the layout and formatting of your website.
- Once you know the basics of CSS, you should be capable of improving the look of your website and all of the HTML that you have already added to the design.

JAVASCRIPT:

JavaScript is among the most important programming languages for website owners who want to make their site more interactive, which makes this the most complex of the three standard programming languages in this article**.** While HTML provides the structure for a website and CSS allows you to control the presentation of a site, the JavaScript programming language gives you the tools that you need to alter the behaviour of different elements that are found on a website page. With this tool, you can add a layer of interactivity to the CSS and

HTML elements on your website, which is highly recommended if you want your website to appeal to the modern customer.

REACT

The top tier of the MERN stack is React.js, the declarative JavaScript framework for creating dynamic client-side applications in HTML. React lets you build up complex interfaces through simple Components, connect them to data on your backend server, and render them as HTML.

Reacts strong suit is handling stateful, data-driven interfaces with minimal code and minimal pain, and it has all the bells and whistles you'd expect from a modern web framework: great support for forms, error handling, events, lists, and more.

BACKEND:

EXPRESS and NODE JS

The next level down is the Express.js server-side framework, running inside a Node.js server. Express.js bills itself as a "fast, unopinionated, minimalist web framework for Node.js," and that is indeed exactly what it is. Express.js has powerful models for URL routing (matching an incoming URL with a server function), and handling HTTP requests and responses.

By making XML HTTP Requests (XHRs) or GETs or POSTs from your React.js front-end, you can connect to Express.js functions that power your application. Those functions in turn use MongoDB's Node.js drivers, either via callbacks for using Promises, to access and update data in your MongoDB database.

MONGO DB:

The next level down is the Express.js server-side framework, running inside a Node.js server. Express.js bills itself as a "fast, unopinionated, minimalist web framework for Node.js," and that is indeed exactly what it is. Express.js has powerful models for URL routing (matching an incoming URL with a server function), and handling HTTP requests and responses.

By making XML HTTP Requests (XHRs) or GETs or POSTs from your React.js front-end, you can connect to Express.js functions that power your application. Those

functions in turn use MongoDB's Node.js drivers, either via callbacks for using Promises, to access and update data in your MongoDB database.

FACT FINDING TECHNIQUES

1. OBSERVATION:

I observed sites like coinmarket and coinlab. I used API to get the data of crypto currency and its news using these websites, news API and crytptocurrency API.

2. REVIEWING EXISTING DOCUMENTS

We reviewed various websites and applications that people use to buy products and get information. We even reviewed newspapers and applications to get some idea about how to start this project.

3. INTERVIEWS

I have Interview various investors of cryptocurrency, according to them information like market cap, price which are there in the current website are not sufficient, they said more information are needed like graphs which will show the historical data. They also said they portfolio should be real time.

I have decided to add the graphs and portfolio requirement in future development of the website

QUESTIONNAIRE

1. Are you satisfied with the working of this software?

2. Do you like the way the data is represented?

3. What do you feel about this software?

4. Does this website cater to all your needs?

5. Do you have any suggestions to improve this software?

6. Is this software convenient to use?

7. Does this software give you the adequate information you're looking for?

8. What feature of this software you like most?

9. How do you think this software will affect your business?

FEASIBILITY STUDY

TECHNICAL FEASIBILITY –

In Technical Feasibility current resources both hardware software along with required technology are analysed/assessed to develop project. This technical feasibility study gives report whether there exists correct required resources and technologies which will be used for project development. Along with this, feasibility study also analyzes technical skills and capabilities of technical team, existing technology can be used or not, maintenance and up-gradation is easy or not for chosen technology etc.

In my project I have used React for Frontend and Node js and mongo DB with express which are quite accessible and also react is maintained by Facebook so the scope is high for react

ECONOMIC FEASIBILITY –

In Economic Feasibility study cost and benefit of the project is analysed. Means under this feasibility study a detail analysis is carried out what will be cost of the project for development which includes all required cost for final development like hardware and software resource required, design and development cost and operational cost and so on. After that it is analysed whether project will be beneficial in terms of finance for organization or not.

Considering economic feasibility, API keys used are either free or low cost. Hosting the website Is also cheaper nowadays using sites like HEROKU.

LEGAL FEASIBILITY –

In Legal Feasibility study project is analysed in legality point of view. This includes analysing barriers of legal implementation of project, data protection acts or social media laws, project certificate, license, copyright etc. Overall it can be said that Legal Feasibility Study is study to know if proposed project conform legal and ethical requirements.

All the images, videos and API source are mentioned , also there is a terms and condition page

SCHEDULE FEASIBILITY –

In Schedule Feasibility Study mainly timelines/deadlines is analysed for proposed project which includes how many times teams will take to complete final project which has a great impact on the organization as purpose of project may fail if it can’t be completed on time.

FINANCIAL FEASIBILITY

What is Financial Feasibility?

A study on whether a project is viable after taking into consideration its total costs and probable revenues. If the revenues cover the costs of the project, then the project is visible. For instance, the cost of expanding production will entail $500,000; however, the expected increase in revenue will be $1,000,000. Thus, the project is feasible as its revenue more than cover its costs.

This website is financially feasible for everyone as it will be free of cost.

STAKEHOLDERS

OWNER

Owner is the one who owns the website. He will always monitor what is happening in the system.

ADMIN

Admin is the one who controls all the data entered by the users in the website forms.

Admin filters all the data received and provide genuine information to the customer. Therefore, the admin has complete control over the website.

Also manages user data and all other API that are used

DEVELOPERS

Developers are stakeholders as they will keep on working on the project for future development. If any issues occur in the system, they will find the problem and try to find a solution for it.

USERS

Users are the ones who really matters in the successful run of the software. They are the ones who decide the future of the website.

CHAPTER 3
Requirement and Analysis.

PROBLEM DEFINITION

In the current system the information provide is very less, if we consider a website showing details of cryptocurrency, only small data is shown. Data like market price, current price and ups and down are shown. Also, we can't track our portfolio.

DISADVANTAGES WITH CURRENT SYSTEMS:

1. Less data is shown.
2. Users can't track their portfolio.
3. Users can't convert normal currencies into crypto.
4. It's not just about the crypto data, it's also about the news of crypto, user should be updated with the news of cryptocurrency.

REQUIREMENT SPECIFICATION

In the proposed system the customer can sit in the comfort of his home and check out the available cryptocurrency. They can compare different cryptocurrency and also maintain their portfolio. They can track their cryptocurrency using graphs which are provided by the website.

MODULES OF PROPOSED SYSTEM

Admin Module:

Admin Module is the workspace where admin can do all the works from tracking user to managing API. Also, it tracks visits from users.

Admin Module consists of the following module:

1. **Portfolio Module:**
   In portfolio module the user's cryptocurrency portfolio is managed.
2. **Comment Module:**
   In comment module the comments from users are managed. These comments we get from the comment section.
3. **Contact Message Module:**
   This module is used to track contact messages.

User Module:

This module is for user which consists of following modules:

1. **Home Module:**
   This module will show a brief introduction the website, i.e., what the website do and will have navigation to all other module.
2. **Tracker Module:**
   This module will have the main part of the website which is crypto currency tracker also the module will lead to the specific cryptocurrency page which chose by the user.
3. **Specific Cryptocurrency Module**:

This module is dynamically generated as per the user choice, i.e. when a user choose a cryptocurrency from the tracker, it will lead to this module where it consists of more information about the same crypto currency.

4. **Contact module:**

   In this module the user will add their queries, which will be answered by the owner.

5. **Portfolio Module:**

   In this module user can add their portfolio

6. **News Module:**

   In this module user can view news about cryptocurrency.

7. **Converter Module:**

   In this module user can convert their favourite cryptocurrency to any other currency. For example, to convert BTC to USD, user have to select the specific currency and add the amount.

PLANNING AND SCHEDULING

GANTT CHART

🟦

TARGET

🟧

TIME TAKEN

Sr.No	Topic	July			August			September
		1/6-10/6	11/6-20/6	21/6-31/6	1/7-10/7	11/7-20/7	21/7-31/7	1/8-10/8
1	**Introduction**	🟦 🟧	🟦 🟧	🟦 🟧				
1.1	Background		🟦 🟧	🟦 🟧				
1.2	Objective		🟦	🟦 🟧	🟦 🟧			
1.3	Purpose		🟦 🟧	🟦 🟧	🟦 🟧			
1.4	Scope		🟦 🟧	🟦 🟧	🟦 🟧			
1.5	Applicability			🟦 🟧	🟦 🟧	🟧		
2	**System Planning**							
2.1	Survey of Technology		🟦 🟧	🟦 🟧	🟦 🟧			
2.2	Fact Finding Techniques			🟦	🟦 🟧	🟦 🟧		
2.3	Feasibility Study				🟦 🟧	🟦 🟧		
2.4	Stakeholder				🟦 🟧	🟦 🟧		
3	**Requirement & Analysis**							

3.1	Problem Definition		🟦 🟧	
3.2	Requirement		🟦 🟧	
3.3	Planning & Scheduling		🟦 🟧	
3.4	Software & Hardware Requirement		🟦 🟧	
3.5	Conceptual Notes		🟦 🟧	

4	**System Design**			
4.1	Data Design			🟦 🟧
4.2	Data Integrity & Constraints			🟦 🟧
4.3	User Interface & Design			🟦 🟧
4.4	Security Issues			🟦 🟧
4.5	Test Cases			🟦 🟧

SOFTWARE AND HARDWARE

REQUIREMENTS HARDWARE REQUIREMENT

DEVELOPER SIDE

	PROCESSOR	RAM	DISK SPACE
CLIENT SIDE	1.6 GHz or faster processor	1GB RAM	200MB disk space
SERVER SIDE	Server Environment Capable Hardware	2GB RAM	As per the size of the database

SOFTWARE REQUIREMENT

FRONT END	Visual Studio Code
BACKEND	Mongo DB atlas
OPERATING SYSTEM	Windows 10

CONCEPTUAL MODELS

USE CASE DIAGRAM:

What is Use Case Diagram?

To model a system, the most important aspect is to capture the dynamic behavior. Dynamic behavior means the behavior of the system when it is running/operating.

Only static behavior is not sufficient to model a system rather dynamic behavior is more important than static behavior. In UML, there are five diagrams available to model the dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature, there should be some internal or external factors for making the interaction.

These internal and external agents are known as actors. Use case diagrams consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system.

Hence to model the entire system, a number of use case diagrams are used.

the purpose of use case diagram is to capture the dynamic aspect of a system. However, this definition is too generic to describe the purpose, as other four diagrams (activity, sequence, collaboration, and State chart) also have the same purpose. We will look into some specific purpose, which will distinguish it from other four diagrams.

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analyzed to gather its functionalities, use cases are prepared and actors are identified.

When the initial task is complete, use case diagrams are modelled to present the outside view.

In brief, the purposes of use case diagrams can be said to be as

follows –

- Used to gather the requirements of a system.
- Used to get an outside view of a system.
- Identify the external and internal factors influencing the system.
- Show the interaction among the requirements are actors.

Diagram:
Fig 3.1

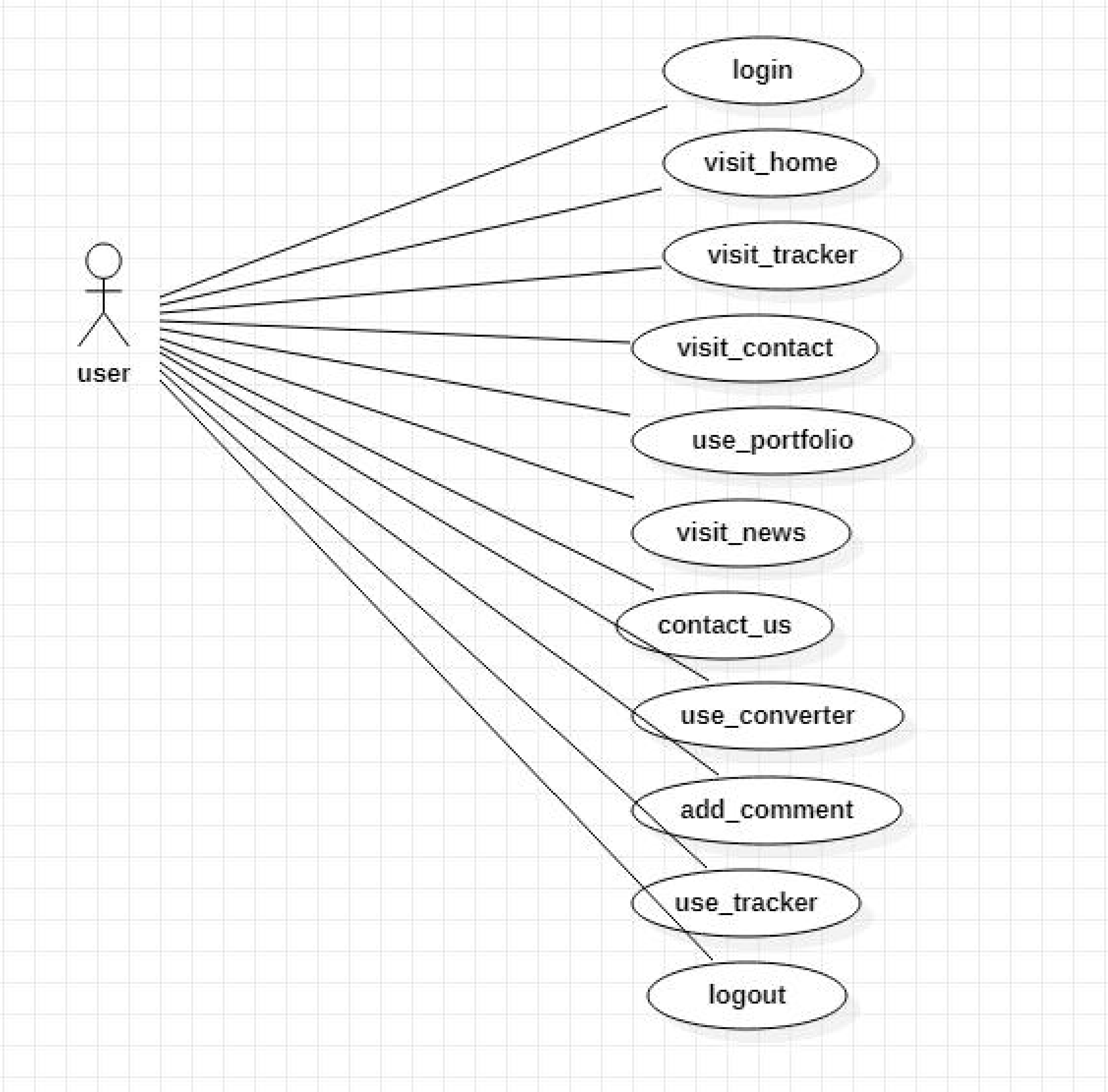
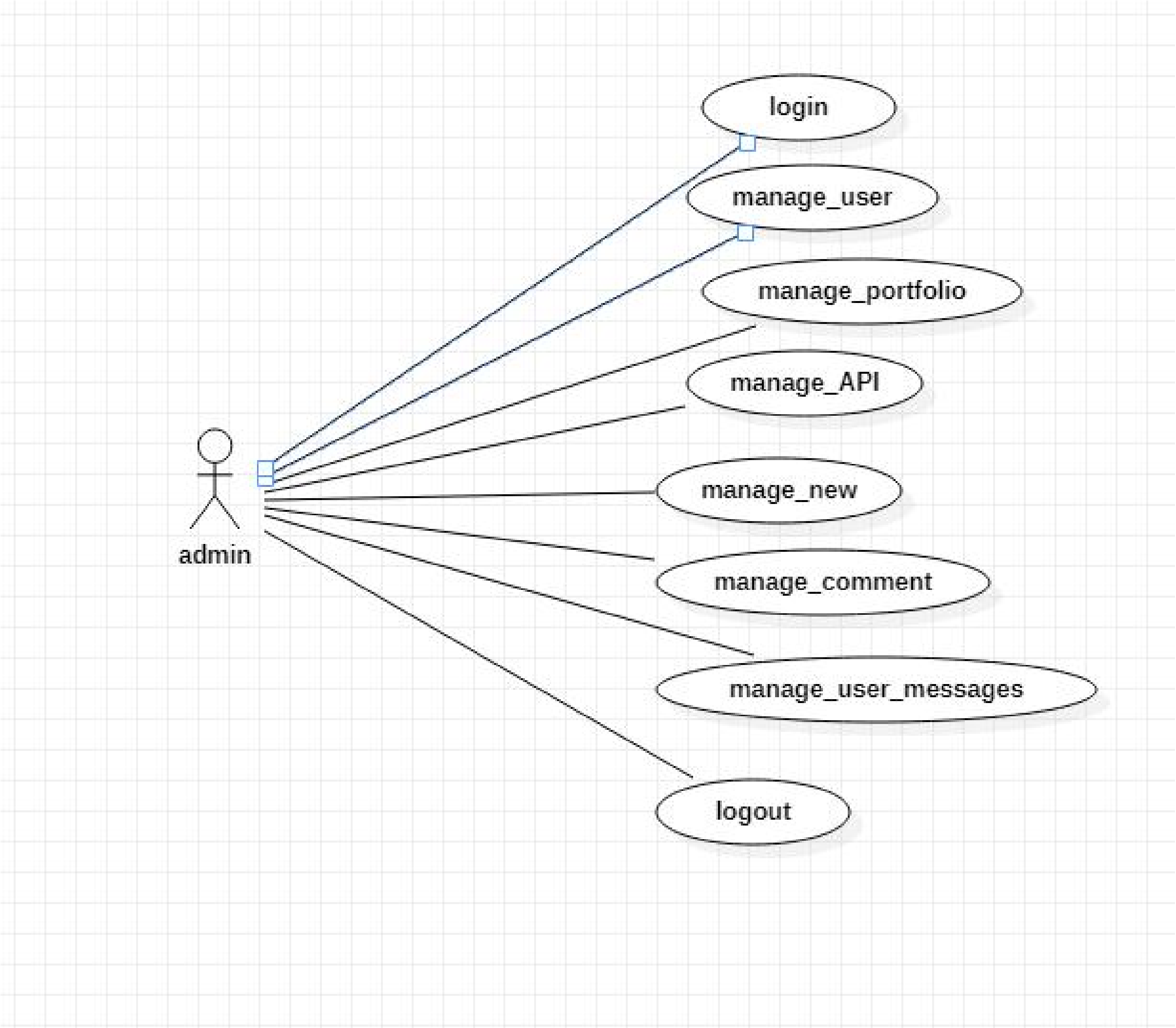


Fig 3.2


login
manage_user
manage_portfolio
manage_API
admin
manage_new
manage_comment
manage_user_messages
logout

ACTIVITY DIAGRAM:

We use Activity Diagrams to illustrate the flow of control in a system and refer to the steps involved in the execution of a use case. We model sequential and concurrent activities using activity diagrams. So, we basically depict workflows visually using an activity diagram. An activity diagram focuses on condition of flow and the sequence in which it happens. We describe or depict what causes a particular event using an activity diagram.

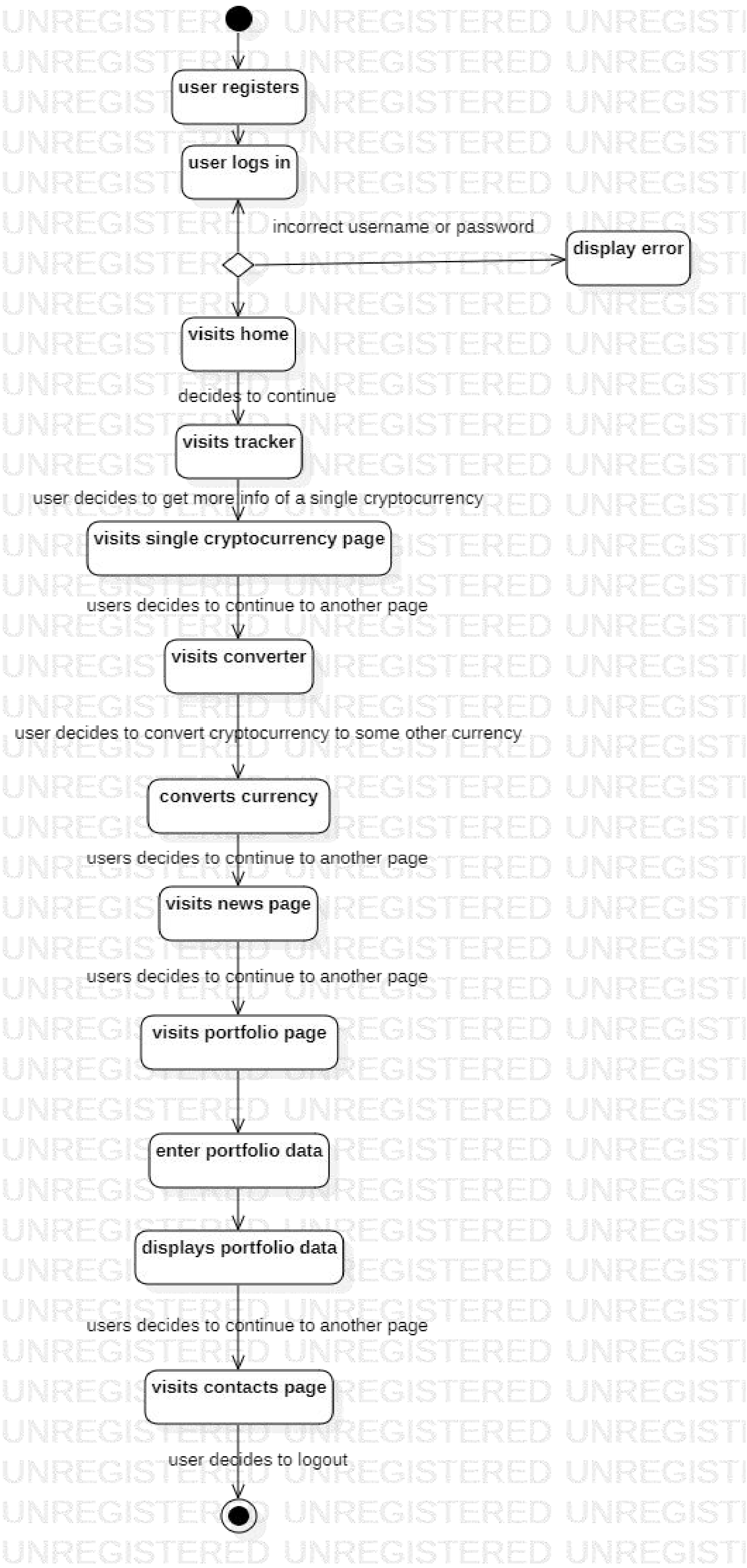
UML models basically three types of diagrams, namely, structure diagrams, interaction diagrams, and behaviour diagrams. An activity diagram is a behavioural diagram i.e. it depicts

the behaviour of a system.

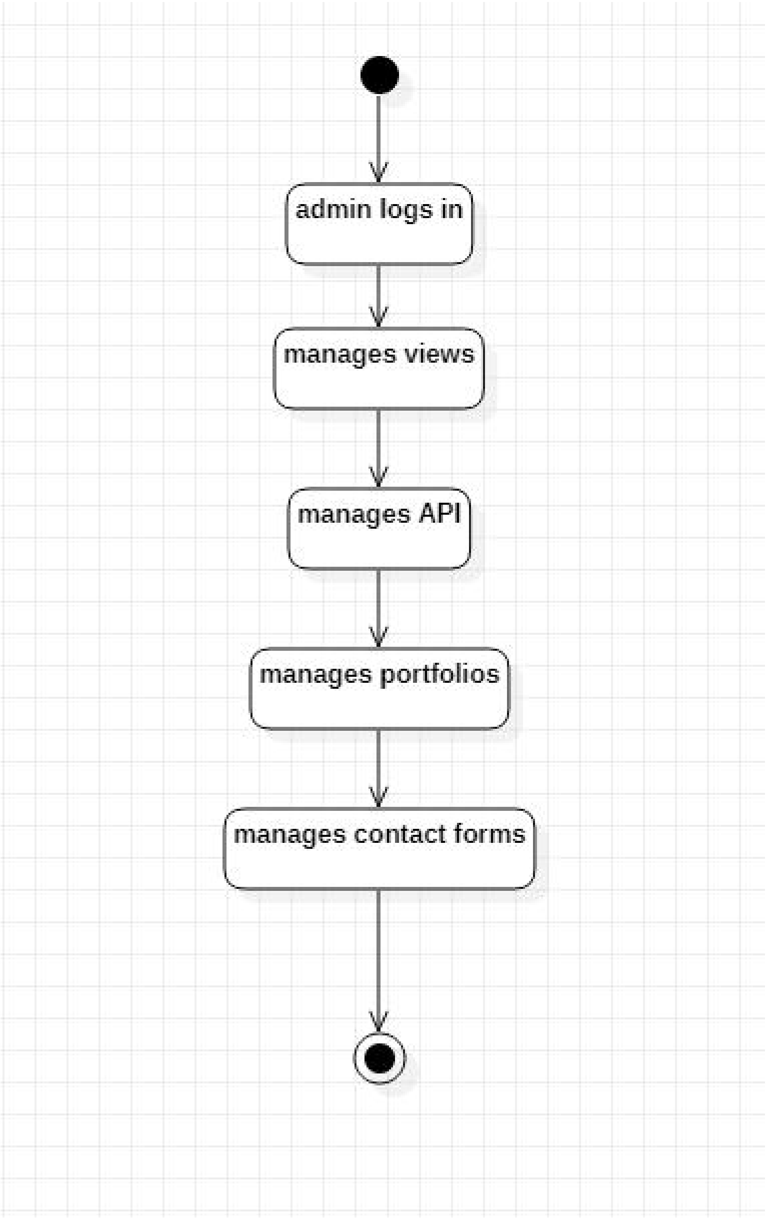
An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed. We can depict both sequential processing and concurrent processing of activities using an activity diagram. They are used in business and process modelling where their primary use is to depict the dynamic aspects of a system.

USERS ACTIVITY DIAGRAM:

User Activity Diagram: fig 3.4


user registers
user logs in
incorrect username or password
display error
visits home
decides to continue
visits tracker
user decides to get more info of a single cryptocurrency
visits single cryptocurrency page
users decides to continue to another page
visits converter
user decides to convert cryptocurrency to some other currency
converts currency
users decides to continue to another page
visits news page
users decides to continue to another page
visits portfolio page
enter portfolio data
displays portfolio data
users decides to continue to another page
visits contacts page
user decides to logout

Admin Activity Diagram: fig 3.5



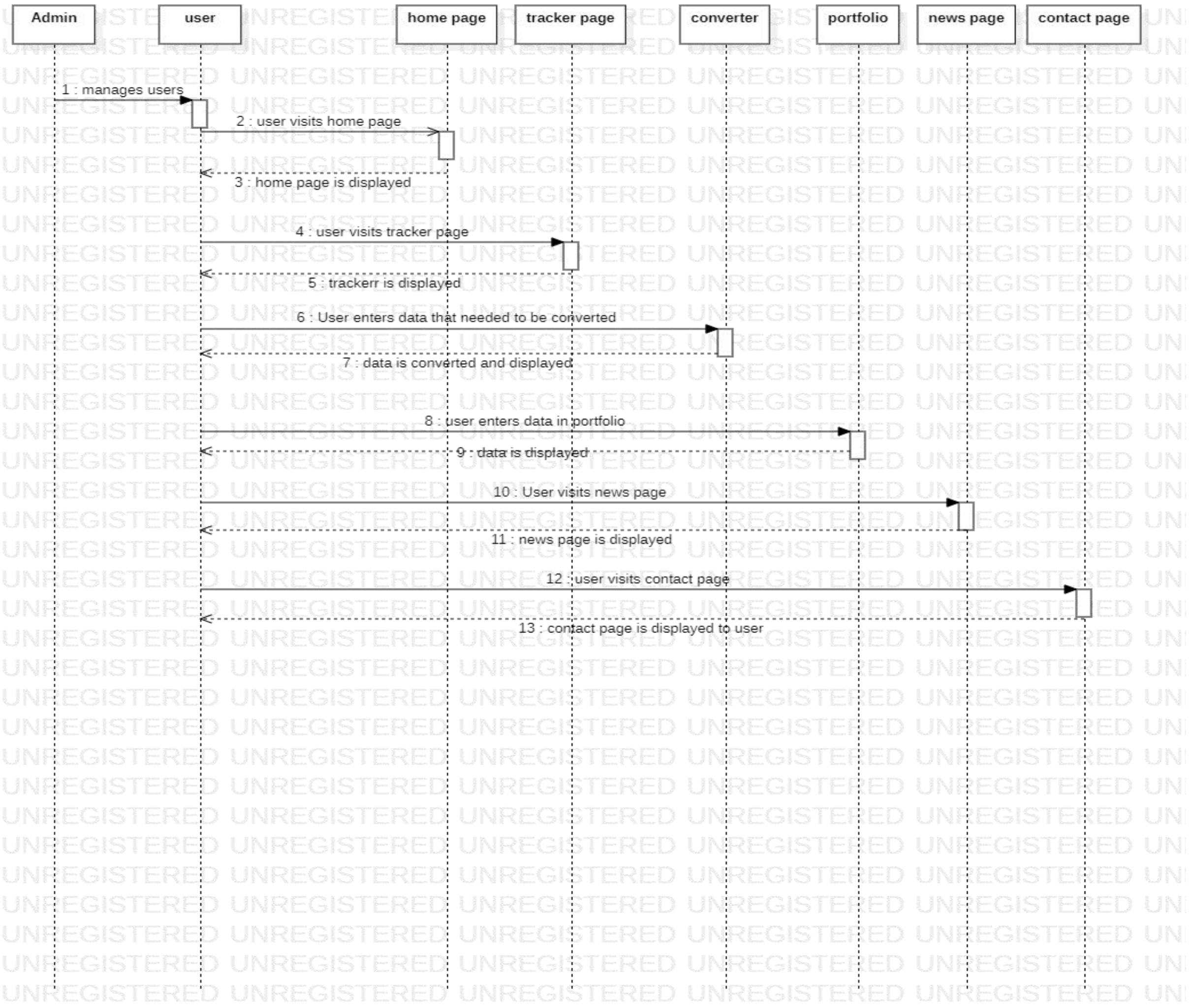
SEQUENCE DIAGRAM

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

A sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

DIAGRAM:

Fig 3.6


Admin
user
home page
tracker page
converter
portfolio
news page
contact page
1 : manages users
2 : user visits home page
3 : home page is displayed
4 : user visits tracker page
5 : trackerr is displayed
6 : User enters data that needed to be converted
7 : data is converted and displayed
8 : user enters data in portfolio
9 : data is displayed
10 : User visits news page
11 : news page is displayed
12 : user visits contact page
13 : contact page is displayed to user

COMPONENT DIAGRAM:

Component diagrams are different in terms of nature and behavior. Component diagrams are used to model the physical aspects of a system. Now the question is, what are these physical aspects? Physical aspects are the elements such as executables, libraries, files, documents, etc. which reside in a node.

Component diagrams are used to visualize the organization and relationships among components in a system. These diagrams are also used to make executable systems.

PURPOSE OF COMPONENT DIAGRAMS:

Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describes the components used to make those functionalities.

Thus, from that point of view, component diagrams are used to visualize the physical components in a system. These components are libraries, packages, files, etc.

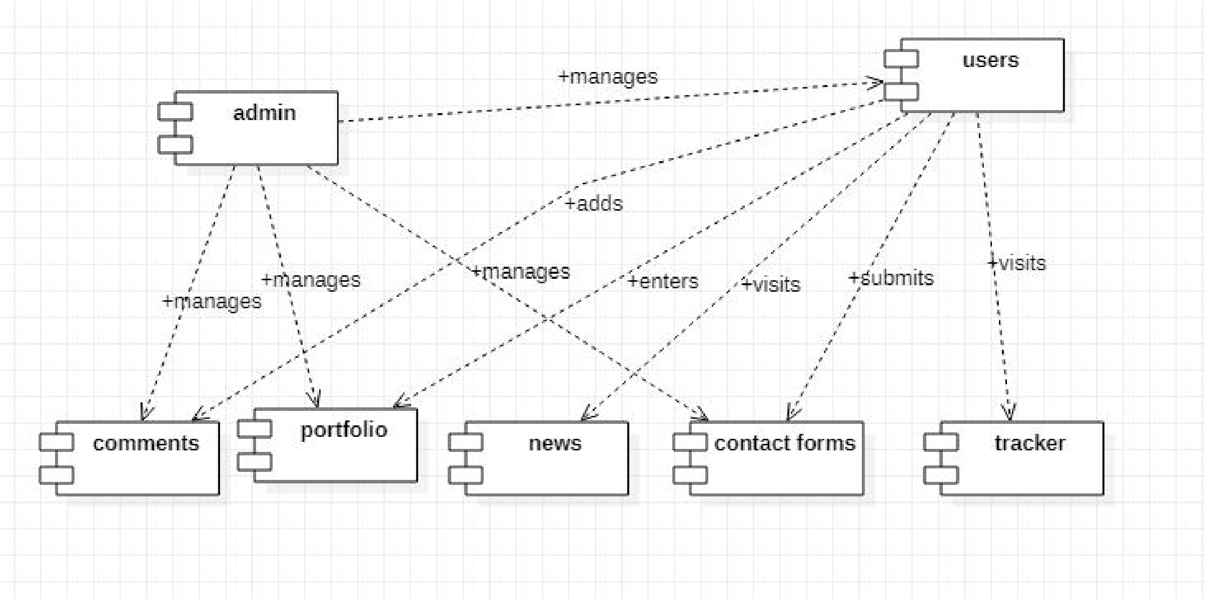
Component diagrams can also be described as a static implementation view of a system. Static implementation represents the organization of the components at a particular moment.

A single component diagram cannot represent the entire system but a collection of diagrams is used to represent the whole.

The purpose of the component diagram can be summarized as −

- Visualize the components of a system.
- Construct executables by using forward and reverse engineering.
- Describe the organization and relationships of the components.

DIAGRAM:


users
admin
+manages
+adds
+manages
+manages
+manages
+enters
+visits
+submits
+visits
comments
portfolio
news
contact forms
tracker

CHAPTER 4

System Design

DATA DESIGN

USER SCHEMA

UsersSchema {

Id: Number,

FirstName: String,

LastName: String,

Email: String,

UserName: String,

Password:String,

Comments: [CommentSchema],

Portfolio: [PortfolioSchema]

}

COMMENT SCHEMA

CommentSchema {

Comment: String,

CommentLoc: String,

CommentDates: Date,

CommentTime: Date

```
}
```

PORTFOLIO SCHEMA

```
PortfolioSchema {

CryptoName: String,

Amount: Number,

PortDates: Date,

PortTime: Date

}
```

CONTACT SCHEMA

```
contactSchema {

email : String,

subject: String,

Amount: Number,

contactDates: Date,

conttactTime: Date

}
```

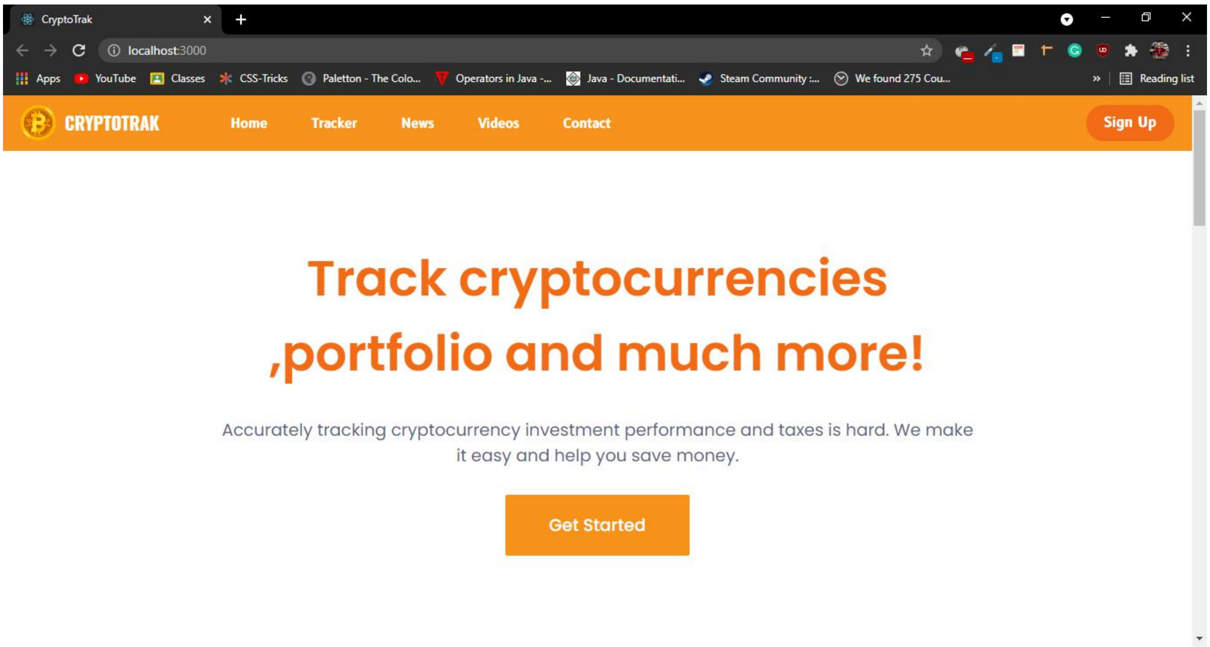
ADMIN SCHEMA

```
adminSchema{
UserName: String,

Password: String

}
```

USER INTERFACE AND DESIGN:

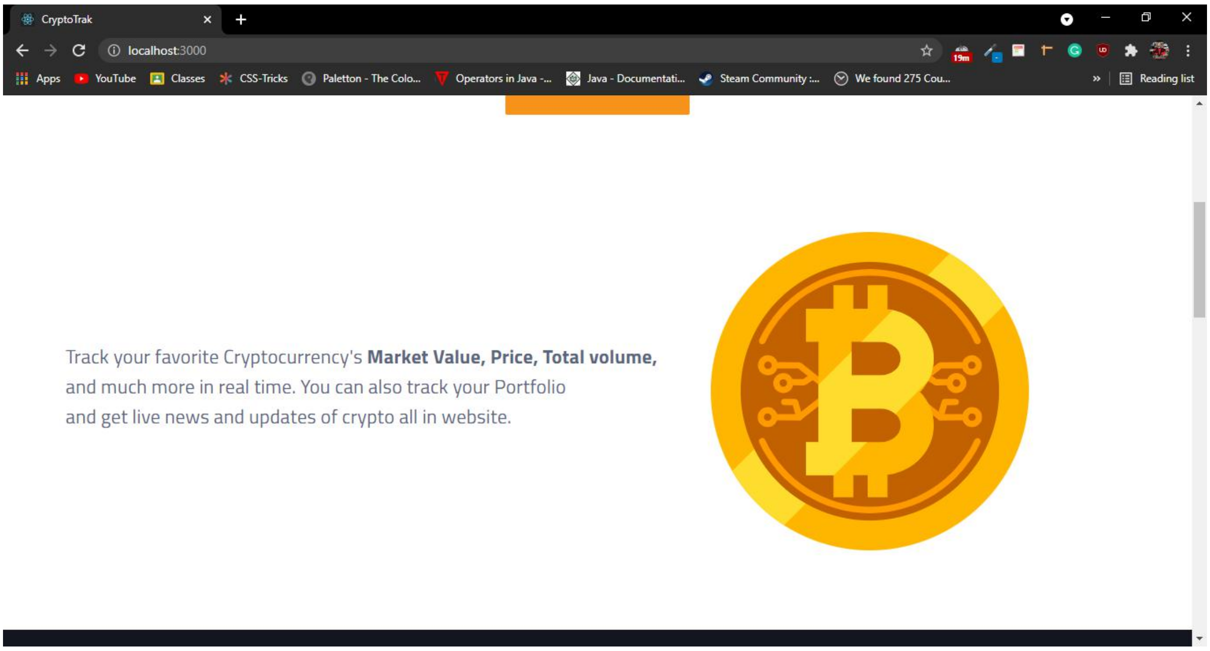
In the industrial design field of human-computer interaction, a user interface (UI) is the space where interactions between humans and machines occur. The goal of this interaction is to allow effective operation and control of the machine from the human end, whilst the machine simultaneously feeds back information that aids the operators' decision-making process. Examples of this broad concept of user interfaces include the interactive aspects of computer operating systems, hand tools, heavy machinery operator controls, and process controls. The design considerations applicable when creating user interfaces are related to, or involve such disciplines as, ergonomics and psychology.

HOME PAGE

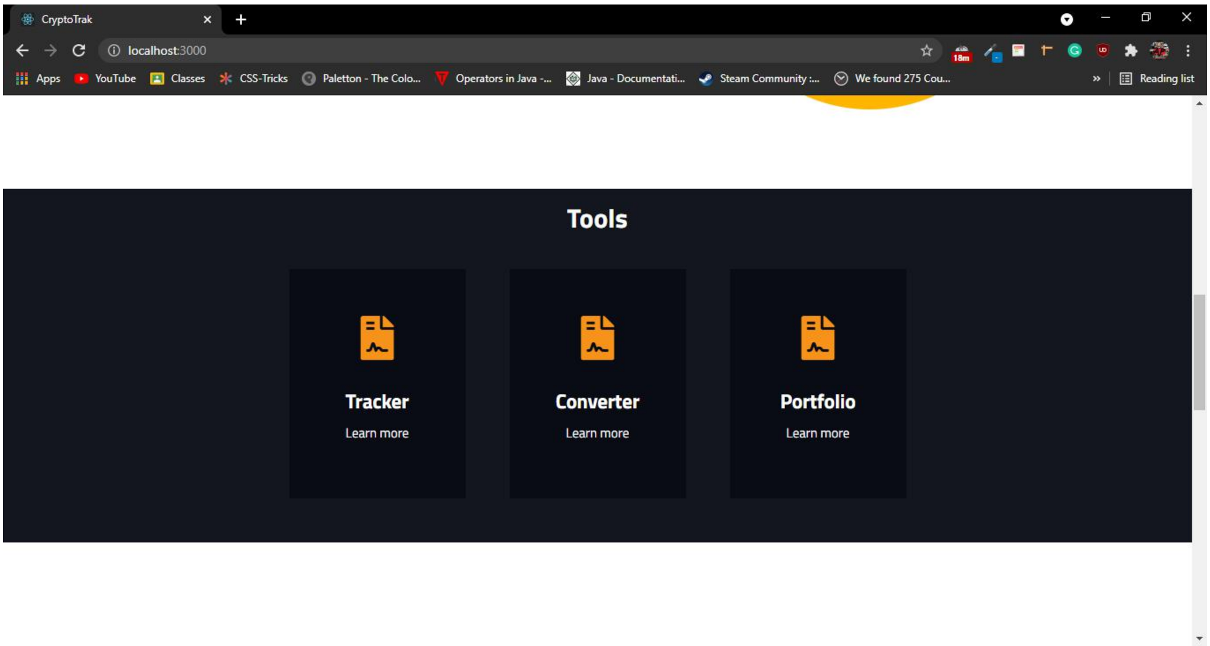


Track cryptocurrencies ,portfolio and much more!

Accurately tracking cryptocurrency investment performance and taxes is hard. We make it easy and help you save money.



Track your favorite Cryptocurrency's **Market Value, Price, Total volume,** and much more in real time. You can also track your Portfolio and get live news and updates of crypto all in website.



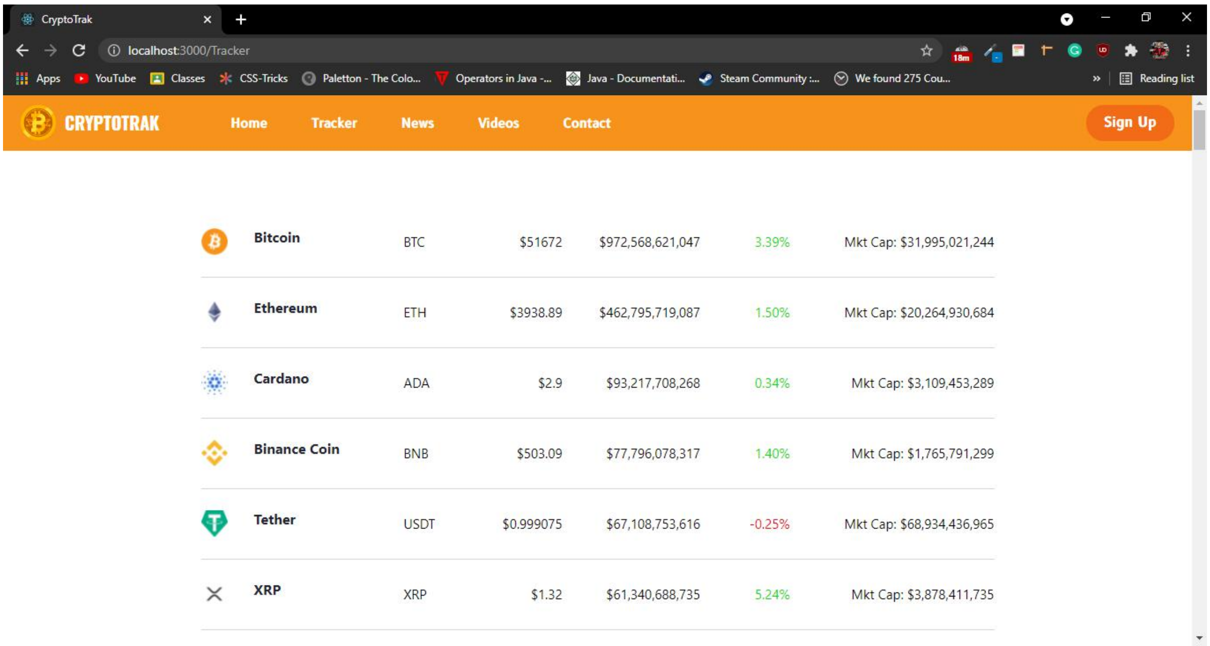
Tools

Tracker
Learn more

Converter
Learn more

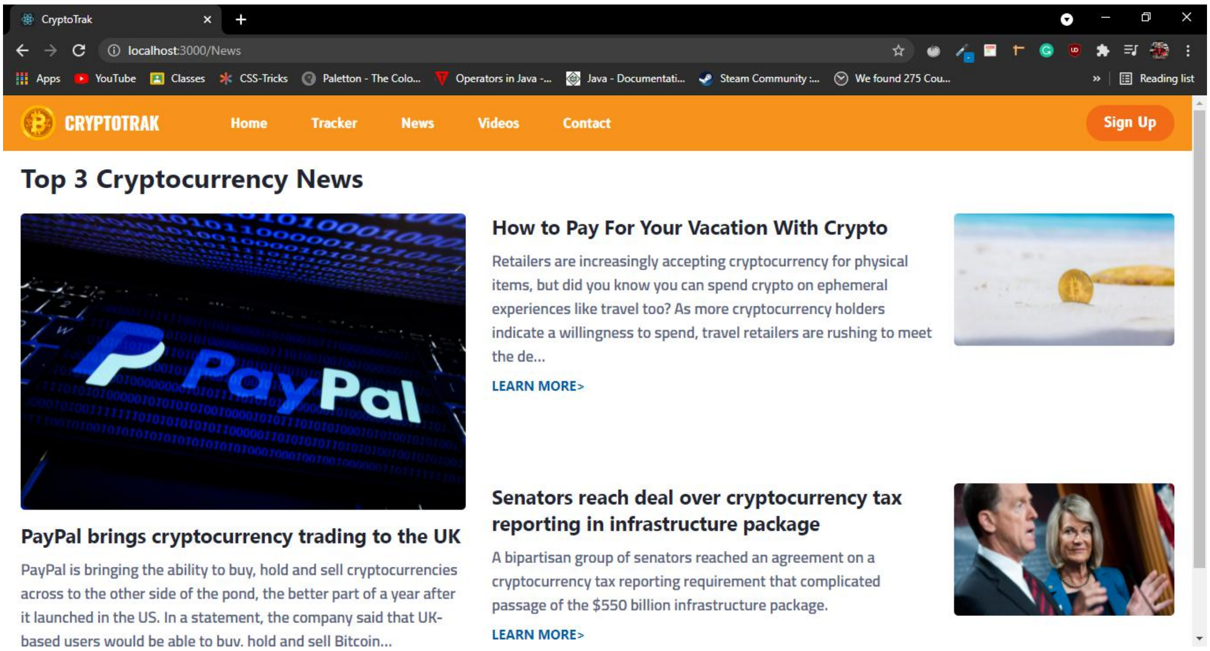
Portfolio
Learn more

TRACKER PAGE



Bitcoin	BTC	$51672	$972,568,621,047	3.39%	Mkt Cap: $31,995,021,244
Ethereum	ETH	$3938.89	$462,795,719,087	1.50%	Mkt Cap: $20,264,930,684
Cardano	ADA	$2.9	$93,217,708,268	0.34%	Mkt Cap: $3,109,453,289
Binance Coin	BNB	$503.09	$77,796,078,317	1.40%	Mkt Cap: $1,765,791,299
Tether	USDT	$0.999075	$67,108,753,616	-0.25%	Mkt Cap: $68,934,436,965
XRP	XRP	$1.32	$61,340,688,735	5.24%	Mkt Cap: $3,878,411,735

NEWS PAGE



Top 3 Cryptocurrency News

PayPal brings cryptocurrency trading to the UK

PayPal is bringing the ability to buy, hold and sell cryptocurrencies across to the other side of the pond, the better part of a year after it launched in the US. In a statement, the company said that UK-based users would be able to buy, hold and sell Bitcoin...

How to Pay For Your Vacation With Crypto

Retailers are increasingly accepting cryptocurrency for physical items, but did you know you can spend crypto on ephemeral experiences like travel too? As more cryptocurrency holders indicate a willingness to spend, travel retailers are rushing to meet the de...

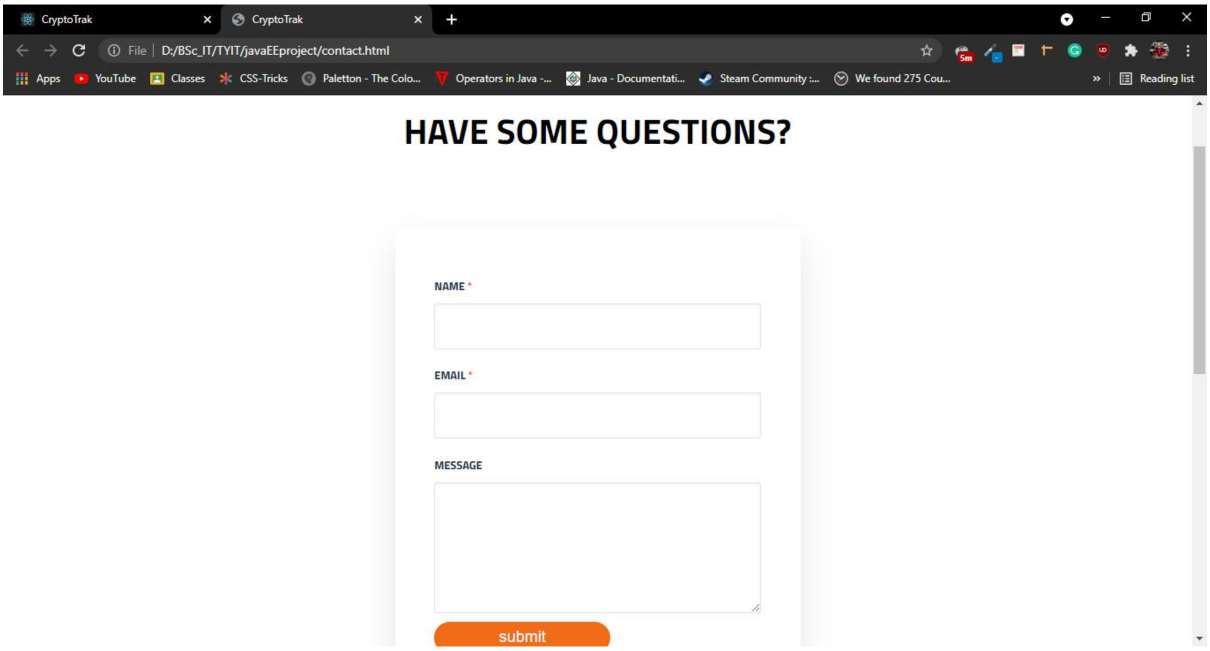
LEARN MORE>

Senators reach deal over cryptocurrency tax reporting in infrastructure package

A bipartisan group of senators reached an agreement on a cryptocurrency tax reporting requirement that complicated passage of the $550 billion infrastructure package.

LEARN MORE>

CONTACT PAGE



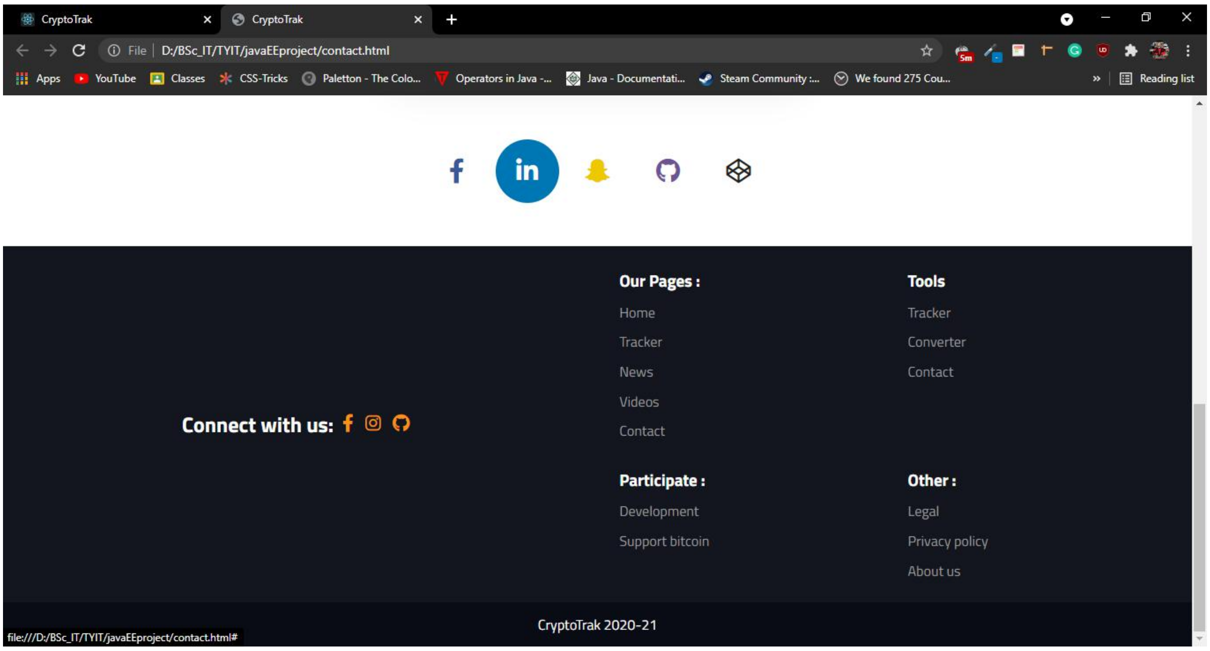
HAVE SOME QUESTIONS?

NAME *

EMAIL *

MESSAGE

submit



Connect with us:

Our Pages :
Home
Tracker
News
Videos
Contact

Tools
Tracker
Converter
Contact

Participate :
Development
Support bitcoin

Other :
Legal
Privacy policy
About us

CryptoTrak 2020-21

file:///D:/BSc_IT/TYIT/javaEEproject/contact.html#

SECURITY ISSUES

NOSQL INJECTION:

With traditional SQL injection, the attacker exploits unsafe user input processing to modify or replace SQL queries (or other SQL statements) that the application sends to a database engine. In other words, an SQL injection allows the attacker to execute commands in the database. Unlike relational databases, NoSQL databases don't use a common query language. NoSQL query syntax is product-specific and queries are written in the programming language of the application: PHP, JavaScript, Python, Java, and so on. This means that a successful injection lets the attacker execute commands not only in the database, but also in the application itself, which can be far more dangerous.

MongoDB uses the Binary JSON (BSON) data format and comes with a secure BSON query assembly tool. Queries are also represented as BSON objects (i.e. binary data), so direct string injection is not possible. However, MongoDB deliberately allows applications to run JavaScript on the server within the $where and mapReduce operations. The documentation is very clear that if this functionality is used, the developer should take care to prevent users from submitting malicious JavaScript. In other words, MongoDB deliberately includes a potential injection vector.

PREVENTING NOSQL INJECTION ATTACKS

The consequences of a successful MongoDB injection or other NoSQL injection attack can be even more serious than with traditional SQL injection. Attackers can not only extract data from the database, but also execute code in the context of the application, for example to perform denial of service attacks or even compromise admin user accounts and take control of the server. Such attacks are especially dangerous since NoSQL data stores are often a novelty to developers familiar only with relational database products, which increases the risk of insecure code.

Many popular NoSQL products are still young and under intense development, so it’s always a good idea to use the most recent version. For example, earlier versions of MongoDB were notoriously insecure on many levels and had serious injection vulnerabilities, while in more recent versions, security is taken much more seriously.

As is often the case in web application security, the best way to prevent NoSQL injection attacks is to avoid using unsanitized user inputs in application code, especially when building database queries. MongoDB, for example, has built-in features for secure query building without JavaScript.

BROKEN AUTHENTICATION

This is a collection of multiple problems that might occur during broken authentication, but they don’t all stem from the same root cause.

Assuming that anyone still wants to roll their own authentication code in 2014 (what are you thinking??), I advise against it. It is extremely hard to get right, and there are a myriad of possible pitfalls, just to mention a few:

- The URL might contain the session id and leak it in the referrer header to someone else.
- The passwords might not be encrypted either in storage or transit.
- The session ids might be predictable, thus gaining access is trivial.
- Session fixation might be possible.
- Session hijacking might be possible, timeouts not implemented right or using HTTP (no SSL security), etc...

INSECURE DIRECT OBJECT REFERENCES

This is a classic case of trusting user input and paying the price in a resulting security vulnerability. A direct object reference means that an internal object such as a file or database key is exposed to the user. The problem with this is that the attacker can provide this reference

and, if authorization is either not enforced (or is broken), the attacker can access or do things that they should be precluded from.

For example, the code has a download.php module that reads and lets the user download files, using a CGI parameter to specify the file name (e.g., download.php?file=something.txt). Either by mistake or due to laziness, the developer omitted authorization from the code. The attacker can now use this to download any system files that the user running PHP has access to, like the application code itself or other data left lying around on the server, like backups. Uh-oh.

Another common vulnerability example is a password reset function that relies on user input to determine whose password we're resetting. After clicking the valid URL, an attacker can just modify the username field in the URL to say something like "admin".

Incidentally, both of these examples are things I myself have seen appearing often "in the wild".

Prevention: Perform user authorization properly and consistently, and whitelist the choices. More often than not though, the whole problem can be avoided by storing data internally and not relying on it being passed from the client via CGI parameters. Session variables in most frameworks are well suited for this purpose.

SENSITIVE DATA EXPOSURE

This web security vulnerability is about crypto and resource protection. Sensitive data should be encrypted at all times, including in transit and at rest. No exceptions. Credit card information and user passwords should never travel or be stored unencrypted, and passwords

should always be hashed. Obviously the crypto/hashing algorithm must not be a weak one – when in doubt, web security standards recommend AES (256 bits and up) and RSA (2048 bits
and up).

And while it goes without saying that session IDs and sensitive data should not be traveling in the URLs and sensitive cookies should have the secure flag on, this is very important and cannot be over-emphasized.

CROSS SITE REQUEST FORGERY (CSRF)

This is a nice example of a confused deputy attack whereby the browser is fooled by some other party into misusing its authority. A 3rd party site, for example, can make the user's browser misuse it's authority to do something for the attacker.

In the case of CSRF, a 3rd party site issues requests to the target site (e.g., your bank) using your browser with your cookies / session. If you are logged in on one tab on your bank's homepage, for example, and they are vulnerable to this attack, another tab can make your browser misuse its credentials on the attacker's behalf, resulting in the confused deputy problem. The deputy is the browser that misuses its authority (session cookies) to do something
the attacker instructs it to do.

CROSS SITE SCRIPTING

This is a fairly widespread input sanitization failure (essentially a special case of common mistake #1). An attacker gives your web application JavaScript tags on input. When this input is returned to the user unsanitized, the user's browser will execute it. It can be as simple as crafting a link and persuading a user to click it, or it can be something much more sinister. On page load the script runs and, for example, can be used to post your cookies to the attacker.

THE IMPORTANCE OF ENCRYPTION

With using more and more technologies in our lives we are generating large amounts

of data, a great share of which is sensitive data. If someone else gets hold of that data you may

be at risk of financial fraud or identity theft.

Let us draw a parallel with real life.

Let us say you are going shopping. You visit the store whenever you want, some of the staff may or may not know your name if you are a regular. You choose your products and proceed to the checkout. No one from the store can get a hold of your money until you draw out your wallet and you hand the cashier your cash. In this situation you are relatively safe.

Only if you have a thief in your close vicinity, they may get a hold of your wallet.

Stealing your wallet will give them access to your cash which they can use with little consequence.

In the digital world, you may visit the online store whenever you want, but if you have shopped there before they probably have your personal information already – name, address, and quite possibly your phone number. They have it regardless of whether you are logged in or you are not.

If you have saved your credit card with that shop they have your financial information

at all times, too (whereas when you shop in a brick and mortar store your wallet will only be at

risk while you are there, shopping).

Instead of your money being at risk only from people in your immediate vicinity, you are now at risk from any hacker, who has access to the network, and at all times.

Encryption changes your information so that it is incomprehensible and therefore unusable to anyone who does not have the key.

Let’s say someone tries to steal your personal information, your name, if it was encrypted. It would be the equivalent of going into the store and asking for your name, but anyone who can give them your name speaks another, unknown language.

If they get a hold of your digital wallet it would be the equivalent of acquiring your cash, but your banknotes are in a currency they cannot use unless they go to the exchange bureau that you own, to convert it.

1. (TRIPLE) DES

Triple DES or 3DES is the successor to the DES algorithm, or the Data Encryption Standard. DES is an obsolete symmetric-key method of data encryption. It was phased out because of its over-exploited vulnerabilities. Originally, DES used to be a very popular encryption method. It is a creation of the engineers in IBM, from back in 1970. In 1977 it was adopted by the government of the USA as an encryption standard – the Federal Information Encryption Standard. DES was used for protecting unclassified but sensitive data. It was the first algorithm approved by the federal government for public disclosure. It gained popularity fast among businesses dealing with confidential information – financial institutions, telecommunication companies, other software and Internet providers, etc. DES encrypts data by grouping it into blocks and encoding them simultaneously by applying the key to the entire

block – instead of bit by bit. The blocks are always 64-bit. The key could be applied to all blocks simultaneously or another model could be used where the encryption of one block is made dependent on the result of the previous one.

In DES, the same key is used to encipher and decipher the message. Decoding is nothing more than applying the same steps in reverse direction and reverse order. What is more, the key options are limited. The keys are always 64-bit, however, 8 parts of the key will be limited in variation in order to be used for key validation, which makes it an equivalent of a 56-bit key.

Which means, the choice of encryptions keys is limited and therefore all information, encrypted via DES is prone to brute force attacks. Because of this serious flaw, scientists had concerns about DES being adopted as the standard even in the 1970's. However, it remained a very popular encryption method until the late 1990's.

2. RSA

The algorithm is named after its designers. In the 1970's Ron Rivest, Adi Shamir, and Leonard Adleman: Rivest-Shamir-Adleman came up with the encryption method while working for the Massachusetts Institute of Technology. In contrast to DES, RSA is asymmetric, which means the system is not using the same key for decryption and encryption. The algorithm works via generating a public and a private key. The public and private key are not identical, but are linked to each other. There is no difference which key is used for encryption and which is used for decryption. Both the private and the public key can be used to encode the message and the opposite key will have to be applied to decode it. In fact, the difference is elsewhere:

the public key, as its name suggests, can be introduced to the public, while the private key is the ‘secret’.

The high level of security RSA provides is ensured by the difficulty of factoring long strings, the product of factoring two large prime numbers. Both the private and the public keys are generated via multiplying two large numbers, creating a modulus. The modulus is used by both the public and the private key. The public key is created via the modulus, adding a public exponent, usually 65537. And the private key is created by the modulus, and a private exponent.

The computational difficulty of large integers makes RSA significantly safer than DES. As explained earlier, DES works with keys, equivalent to 56-bit, 3DES works with keys, equivalent to 112-bit, and RSA works with keys that are 1024- or 2048-bits long. Still, experts think 1024-bit keys are relatively vulnerable. The government and the IT industry recommend using 2048-bit keys only. RSA is used for SSH authentication, for SSL encryption, and for protecting sensitive data in various browsers.

TEST CASES:

SR. NO	FORM NAME	TEST CONDITION	STEP OR PROCEDURE	INPUT TEST DATA	ACTUAL OUTPUT	PASS / FAIL
1.	Login	Check login with invalid input	Enters username with wrong password	Username: Abhishek Password: Wrong pass	Message displayed: Wrong username or password	PASS
2.	Login	Check login with invalid input	Entered wrong username with correct password	Username: wrongusername Password: Password123	Message displayed: Wrong username or password	PASS
3	Registration	Check for invalid email	Enters email with no @	Email: abhishekgmail.com	Message displayed :Invalid email address	PASS
4.	Registration	check is both password and confirm password are correct	correct password with wrong password confirmed	password: rightpass password confirmed: wrongpass	message displayed: passwords doesn't match	PASS
5	Contact	Check for invalid email	Enters email with no @	Email: abhishekgmail.com	Message displayed: Invalid email address	PASS
6	portfolio	Check for invalid	Enters currency in	Currency: -19	Message Displayed:	PASS

		input	negative value		Invalid amount	
7	signup check is user repeats same email	check is user repeats same email	repeats email of a existing user	email: jeffreymukkath65 @gmail.com	message displayed: email already exists	PASS

Chapter 5

5.1 Coding Details.

5.2 Coding Efficiency

App.js

```
import './App.css';
import Header from './components/Header'

import Home from './components/Home';
import News from './components/News';
import Contact from './components/Contact';
import Video from './components/Videos';
import Tracker from './components/Tracker';
import {BrowserRouter as Router , Switch , Route} from 'react-router-dom'



function App() {
  return (
    <Router>
    <div className="App">

      <Switch>
<Route path="/" exact component={Home}/>
<Route path="/Tracker" component={Tracker}/>
<Route path="/News" component={News}/>
<Route path="/Videos" component={Contact}/>
<Route path="/Contact" component={Contact}/>
</Switch>


    </div>
    </Router>
  );
}

export default App;
```

Button.js

```
import React from 'react'
import './Button.css'

const Button = (props) => {
    return (
        <div>
            <button style={props.style} className="btn">{props.name}</button>
        </div>
    )
}

export default Button
```

Button.css

```
@import
url("https://fonts.googleapis.com/css2?family=B612:ital,wght@0,400;0,700;1,400;
1,700&display=swap");

@import
url("https://fonts.googleapis.com/css2?family=Poppins:ital,wght@0,100;0,200;0,
300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,
700;1,800;1,900&display=swap");

.btn {
  background: #f26b16;
  border-radius: 1000px;
  width: 100px;
  color: #ffffff;
  padding: 10px 15px;
  border: none;
  text-align: center;
  font-family: "B612", sans-serif;
  font-weight: 700;
  font-size: 16px;
}

.btn:hover {
  opacity: 0.7;
}
/*# sourceMappingURL=Button.css.map */
```

Card.js

```
import React from 'react'
import './Card.css'

const Card = (props) => {
    return (
        <div>
```

```
            <div className="card-body">
                <img src={props.image} alt={props.alt} />
                <h2>{props.h2}</h2>
                <p>{props.p}</p>
                <a>LEARN MORE > </a>

            </div>
        </div>
    )
}

export default Card
```

Card.css

```
@import
url("https://fonts.googleapis.com/css2?family=Poppins:ital,wght@0,100;0,200;0,
300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,
700;1,800;1,900&display=swap");
.card-body {
  border-radius: 5px;
  margin-bottom: 20px;
}

.card-body img {
  width: 100%;
  border-radius: 5px;
  margin-bottom: 10px;
}

.card-body h2 {
  margin-bottom: 10px;
  color: #222531;
}

.card-body p {
  margin-bottom: 20px;
  font-family: "Titillium Web", sans-serif;
  font-weight: 600;
  color: #58667e;
  font-size: 1.1rem;
}

.card-body a {
  font-weight: bold;
  color: #005da6;
}

.card-body a:hover {
  border-bottom: solid;
}
/*# sourceMappingURL=Card.css.map */
```

Coin.js

```
import React from 'react';
import './Coin.css';

const Coin = ({
  name,
  price,
  symbol,
  marketcap,
  volume,
  image,
  priceChange
}) => {
  return (
    <div className='coin-container'>
      <div className='coin-row'>
        <div className='coin'>
          <img src={image} alt='crypto' />
          <h1>{name}</h1>
          <p className='coin-symbol'>{symbol}</p>
        </div>
        <div className='coin-data'>
          <p className='coin-price'>${price}</p>
          <p className='coin-volume'>${volume.toLocaleString()}</p>

          {priceChange < 0 ? (
            <p className='coin-percent red'>{priceChange.toFixed(2)}%</p>
          ) : (
            <p className='coin-percent green'>{priceChange.toFixed(2)}%</p>
          )}

          <p className='coin-marketcap'>
            Mkt Cap: ${marketcap.toLocaleString()}
          </p>
        </div>
      </div>
    </div>
  );
};

export default Coin;
```

Coin.css

```
.coin-container {
  display: flex;
  justify-content: center;
}

.coin-row {
  display: flex;
  flex-direction: row;
  justify-content: start;
  align-items: center;
  height: 80px;
  border-bottom: 1px solid #d7d7d7;
  width: 900px;
}

.coin {
  display: flex;
  align-items: center;
  padding-right: 24px;

  min-width: 300px;
}

.coin h1 {
  font-size: 16px;
  width: 150px;
}

.coin img {
  height: 30px;
  width: 30px;
  margin-right: 10px;
}

.coin-symbol {
  text-transform: uppercase;
}

.coin-data {
  display: flex;
  text-align: right;
  justify-content: space-between;
  width: 100%;
}

.coin-price {
  width: 110px;
}

.coin-volume {
  width: 155px;
}

.coin-marketcap {
  width: 230px;
}

.coin-percent {
  width: 100px;
}
```

```
.red {
  color: #f00606;
}

.green {
  color: #11d811;
}
```

Color.css

```
@mixin gradiant_primary {
  /* Permalink - use to edit and share this gradient:
https://colorzilla.com/gradient-editor/#2989d8+22,207cca+62,1e5799+100 */
  background: rgb(41, 137, 216); /* Old browsers */
  background: -moz-linear-gradient(
    45deg,
    rgba(41, 137, 216, 1) 22%,
    rgba(32, 124, 202, 1) 62%,
    rgba(30, 87, 153, 1) 100%
  ); /* FF3.6-15 */
  background: -webkit-linear-
    gradient(
      45deg,
      rgba(41, 137, 216, 1) 22%,
      rgba(32, 124, 202, 1) 62%,
      rgba(30, 87, 153, 1) 100%
    ); /* Chrome10-25,Safari5.1-6 */
  background: linear-gradient(
    45deg,
    rgba(41, 137, 216, 1) 22%,
    rgba(32, 124, 202, 1) 62%,

    rgba(30, 87, 153, 1) 100%

  ); /* W3C, IE10+, FF16+, Chrome26+, Opera12+, Safari7+ */

  filter: progid:DXImageTransform.Microsoft.gradient( startColorstr='#2989d8',
endColorstr='#1e5799',GradientType=1 ); /* IE6-9 fallback on horizontal
gradient */

}


@import
url("https://fonts.googleapis.com/css2?family=Poppins:ital,wght@0,100;0,200;0,
300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,
700;1,800;1,900&display=swap");


$primary-color: #f7931a;

$secondary-color: #f26b16;

$link-color: rgb(0, 93, 166);

$font-color-p: #58667e;

$font-color-h: #222531;

$primary-color-faded: #f1e6db;


$b612-font: "B612", sans-serif;

$tillium-font: "Titillium Web", sans-serif;

$poppins-font: "Poppins", sans-serif;


$shadow: 0 1px 5px rgba(104, 104, 104, 0.8);

$shadow-orange: 0 1px 5px rgba(223, 112, 21, 0.5);


$tools-bg: #13161f;

$tools-card: #090c14;
```

Contact.js

```
import React from 'react'

import Header from './Header'

import Footer from './Footer';

import './Contact.css'


function Contact() {

    return (

        <div>

            <Header></Header>


            <form>

  <input name="name" type="text" class="feedback-input" placeholder="Name"
/>

  <input name="email" type="text" class="feedback-input" placeholder="Email"
/>

  <textarea name="text" class="feedback-input"
placeholder="Comment"></textarea>

  <input type="submit" value="SUBMIT"/>

</form>


            <Footer></Footer>

        </div>

    )

}


export default Contact
```

Contact.css

```
@import url(https://fonts.googleapis.com/css?family=Montserrat:400,700);
```

```scss
@import "./colors.scss";

form {
  max-width: 420px;
  margin: 50px auto;
}

.feedback-input {
  color: white;
  font-family: Helvetica, Arial, sans-serif;
  font-weight: 500;
  font-size: 18px;
  border-radius: 5px;
  line-height: 22px;
  background-color: transparent;
  border: 2px solid $primary-color;
  transition: all 0.3s;
  padding: 13px;
  margin-bottom: 15px;
  width: 100%;
  box-sizing: border-box;
  outline: 0;
}

.feedback-input:focus {
  border: 2px solid $tools-bg;
}

textarea {
  height: 150px;
  line-height: 150%;
  resize: vertical;
}

[type="submit"] {
  font-family: "Montserrat", Arial, Helvetica, sans-serif;
  width: 100%;
  background: $primary-color;
  border-radius: 5px;
  border: 0;
  cursor: pointer;
  color: white;
  font-size: 24px;
  padding-top: 10px;
  padding-bottom: 10px;
  transition: all 0.3s;
  margin-top: -4px;
  font-weight: 700;
}
[type="submit"]:hover {
  background: $secondary-color;
}
```

Footer.js

```
import React from 'react'
import  { Link } from 'react-router-dom';
import './Footer.css';
const Footer = () => {
    return (
        <div>
        <div className="Footer">
```

```
<div className="social" >
    <h2>Connect with us: </h2>
<div>
    <i class="fab fa-facebook-f"></i>
    <i class="fab fa-instagram"></i>
    <i class="fab fa-github"></i>
</div>
</div>

<div className="links">
<ul>
    <li><h3>Our Pages :</h3></li>
        <Link className="Link" to="/">
            <li><a href="">Home</a></li>

            </Link>
            <Link className="Link" to="/Tracker">
            <li><a href="">Tracker</a></li>

            </Link>
            <Link className="Link" to="/News">
            <li><a href="">News</a></li>

            </Link>
            <Link className="Link" to="/Videos">
            <li><a href="">Videos</a></li>

            </Link>
            <Link className="Link" to="/Contacts">
            <li><a href="">Contact</a></li>
```

```
                </Link>

            </ul>

            <ul>
        <li><h3>Tools</h3></li>

        <Link className="Link" to="/News">
                <li><a href="">Tracker</a></li>

                </Link>
                <Link className="Link" to="/Videos">
                <li><a href="">Converter</a></li>

                </Link>
                <Link className="Link" to="/Contacts">
                <li><a href="">Contact</a></li>

                </Link>

            </ul>

            <ul>
        <li><h3>Participate :</h3></li>
        <Link className="Link" to="/News">
                <li><a href="">Development</a></li>

                </Link>
                <Link className="Link" to="/Videos">
                <li><a href="">Support Cryptotrak</a></li>
```

```
                    </Link>




                </ul>
                <ul>
            <li><h3>Other :</h3></li>

            <Link className="Link" to="/News">
                    <li><a href="">Legal</a></li>

                    </Link>
                    <Link className="Link" to="/Videos">
                    <li><a href="">Privacy policy</a></li>

                    </Link>
                    <Link className="Link" to="/Contacts">
                    <li><a href="">About us</a></li>

                    </Link>

                </ul>




        </div>
        </div>
        <div className="footer_end"><p>CryptoTrak 2020-21</p></div>
        </div>
    )
}
```

```
export default Footer
```

Footer.css

```
@import "./colors.scss";

.Footer {
  display: grid;
  padding: 20px 0px;
  background-color: $tools-bg;
  h2,
  h3 {
    font-family: $tillium-font;
    color: white;
  }
  grid-template-columns: 1fr 1fr;
}

.social {
  display: flex;
  justify-content: center;
  align-items: center;
  h2 {
    padding-right: 10px;
  }
  i {
    color: $primary-color;
    font-size: 20px;
    padding-right: 10px;
  }
```

```
}
.links {
  display: grid;
  grid-template-columns: 1fr 1fr;
  grid-gap: 20px;
  margin: 0 20px;
  li {
    list-style: none;
    h3 {
      font-weight: 50px;
    }
  }
  a {
    text-decoration: none;
    color: #9d9d9d;
    font-weight: 100px;
  }
}

.footer_end {
  display: flex;
  background: $tools-card;
  height: 50px;
  font-family: $tillium-font;
  align-items: center;
  justify-content: center;
  color: white;
}

//lapi
@media only screen and (max-width: 1024px) {
}
//tablet
@media only screen and (max-width: 768px) {
  .Footer {
    .social {
      margin-bottom: 30px;
    }
    display: flex;
    flex-direction: column;
  }
}

//mobile
@media only screen and (max-width: 425px) {
  .links {
    display: flex;
    flex-direction: column;
    justify-content: center;
  }
}
```

Header.js

```
import React from 'react'
import  { Link } from 'react-router-dom';
import logo from '../assests/logo.svg'
import './Header.css'
import Button from './Button'

const Header = () => {
    function menu_toggle(){
```

```
document.querySelector('.main-menu').classList.toggle('show');
}

return (
    <div className="Header">

    <header className="Navbar">
        <img src={logo} alt="" className="logo" />
        <a className="company_name" href="">CRYPTOTRAK</a>

        <nav>
            <ul className="main-menu">
            <Link className="Link" to="/">
                <li><a href="">Home</a></li>

                </Link>
                <Link className="Link" to="/Tracker">
                <li><a href="">Tracker</a></li>

                </Link>
                <Link className="Link" to="/News">
                <li><a href="">News</a></li>




                </Link>
                <Link className="Link" to="/Contact">
                <li><a href="">Contact</a></li>

                </Link>
                </ul>

            </nav>


            <Button className="buttons" style={{width:'100px'}} name="Sign
Up"> </Button>

            <div onClick={menu_toggle} className="menu-btn"> <i class="fa fa-
bars" ></i>

            </div>

        </header>

        </div>

    )

}


export default Header
```

Header.css

```
//font-family: 'Lato', sans-serif;

@import "colors";


@mixin gradiant_primary {

  /* Permalink - use to edit and share this gradient:
https://colorzilla.com/gradient-editor/#2989d8+22,207cca+62,1e5799+100 */

  background: rgb(41, 137, 216); /* Old browsers */

  background: -moz-linear-gradient(

    45deg,

    rgba(41, 137, 216, 1) 22%,

    rgba(32, 124, 202, 1) 62%,

    rgba(30, 87, 153, 1) 100%

  ); /* FF3.6-15 */

  background: -webkit-linear-gradient(

    45deg,
    rgba(41, 137, 216, 1) 22%,
    rgba(32, 124, 202, 1) 62%,
    rgba(30, 87, 153, 1) 100%
  ); /* Chrome10-25,Safari5.1-6 */
  background: linear-gradient(
    45deg,
    rgba(41, 137, 216, 1) 22%,
    rgba(32, 124, 202, 1) 62%,
    rgba(30, 87, 153, 1) 100%
  ); /* W3C, IE10+, FF16+, Chrome26+, Opera12+, Safari7+ */
  filter: progid:DXImageTransform.Microsoft.gradient( startColorstr='#2989d8',
endColorstr='#1e5799',GradientType=1 ); /* IE6-9 fallback on horizontal
gradient */
}

.Navbar {
  display: flex;
  width: 100%;
  height: 10vh;
  background-color: $primary-color;
  //margin: auto;
  padding: 20px;

  align-items: center;
  justify-content: space-between;

  .logo {
    width: 40px;
    height: 40px;
  }

  .company_name {
    font-family: "Oswald", sans-serif;
    font-weight: bold;
    font-size: 20px;
    display: flex;
    flex: 1;
    color: white;
    margin-left: 10px;
    text-decoration: none;
  }
  nav {
    display: flex;
    flex: 7;
    margin-left: 50px;
    ul {
      display: flex;
      flex-direction: row;
    }
    .Link {
      text-decoration: none;
      li {
        list-style: none;
      }

      a {
        text-decoration: none;

        margin-right: 50px;
        font-family: "B612", sans-serif;
        font-weight: 700;
        padding-bottom: 2px;
        color: white;
      }
      a:hover {
        border-bottom: solid;
      }
    }
  }
}
.menu-btn {
  cursor: pointer;

  top: 10px;
  right: 40px;
  z-index: 2;
  display: none;
  i {
    font-size: 30px;
    color: white;
  }
}

@media only screen and (max-width: 1024px) {
}
//tablet
@media only screen and (max-width: 768px) {
  .menu-btn {
    display: block;
  }

  nav {
    ul.main-menu {
      display: block;
      position: absolute;
      top: 0;
      left: 0;
      background: $tools-bg;
      width: 50%;
      height: 100%;
      border-right: #ccc 1px solid;
      opacity: 0.9;
      padding: 30px;
      transform: translateX(-500px);
      transition: transform 0.5s ease-in-out;
    }
    ul.main-menu.show {
      transform: translateX(-20px);
    }
  }

  Button {
    position: absolute;
    top: 0;
    left: 0;
    transform: translateX(-500px);
    transition: transform 0.5s ease-in-out;
  }
}

//mobile
@media only screen and (max-width: 425px) {
}
```

Home.js

```
import React from 'react'


import Button from './Button';

import './Home.css'

import Header from './Header'

import Card from './Card.js';


import Home_pic from './home_pic.svg';

import logo2 from '../assests/hom_pic.svg';

import tracker_home from '../assests/tracker_home.jpg';

import news_home from '../assests/news_home.jpg';

import contact_home from '../assests/contact_us_home.jpg';

import github from '../assests/github.png'

import Footer from './Footer';

import logo from '../assests/logo.svg';



const Home = () => {


//     const texts = ["Track cryptocurrencies ,portfolio a much more!"];

// let count=  0;

// let index =0;

// let currentText = '';

// let letter = '';


// function type(){


//     if(count === texts.length){

//         count = 0;

//     }
//    currentText = texts[count];


//     letter = currentText.slice(0,++index);
//  document.querySelector('.type').textContent=letter;


//     if(letter.length === currentText.length){
//         count++;


//     }
//     setTimeout(type,100)

// };

    return (
        <div className="Home" >



            <div className="home">


            <Header></Header>
            <div className="home_img">
            <h2 > Track cryptocurrencies ,portfolio and
                much more!</h2>
                <p>Accurately tracking cryptocurrency investment performance
and taxes is hard. We make it easy and help you save money.</p>
                 <a href="#">Get Started</a>
            </div>



               </div>
               <section id="intro">
                   <div>
                       <p>Track your favorite Cryptocurrency's <b>Market Value,
Price, Total volume,</b>
                           <br/> and much more in real time. You can also
track your Portfolio
                           <br/> and get live news and updates of crypto all
in website.</p>
                   </div>
                       <img src={logo} alt="logo" />
               </section>

               <section className="tools ">
                     <h1>Tools</h1>
                   <div className="cardscontainer">

                   <div className="cards c1">
                       <i class="fas fa-file-contract"></i>
                   <h1>Tracker</h1>
                   <a >Learn more</a>
                   </div>

                   <div className="cards c2">
                       <i class="fas fa-file-contract"></i>

                   <h1>Converter</h1>
                   <a >Learn more</a>
                   </div>

                   <div className="cards c3">
                       <i class="fas fa-file-contract"></i>
```

```
                    <h1>Portfolio</h1>

                    <a >Learn more</a>

                      </div>


                    </div>


                </section>



                <div className="cards_container">

                    <Card image={tracker_home} alt="Tracker" h2="Tracker"
p="Start tracking your favorite cryptocurrency now."></Card>

                    <Card image={news_home} alt="News" h2="News" p="Get Live
news of cryptocurrency."></Card>

                    <Card image={contact_home} alt="Contact Us" h2="Contact Us"
p="Do you have any querries? Contact us."></Card>

                    <Card image={tracker_home} alt="Tracker" h2="portfolio"
p="Start tracking your favroute cryptocurrency now"></Card>


                </div>




                <div id="github">

                    <div>

                    <h1>Contribute to Cryptotrack</h1>

                    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit.

                         Dolorum natus, quaerat quos expedita modi rem minima
pariatur soluta

                         harum et fuga delectus

                         maxime cupiditate, blanditiis cum ex id possimus
quibusdam!</p>

                    <a
href="https://github.com/abhishek3011patil/3rdyearproject"><span><i class="fab
fa-github"></i></span>Github</a>
```

```
                    </div>

                    <img src={github} alt="github" />
                    </div>

                    <Footer></Footer>


        </div>
    )
}

export default Home
```

Home.css

```
@import "./colors.scss";

// .cryptotrack {
//   scroll-snap-type: y mandatory;
//   overflow-y: scroll;
//   height: 100vh;
// }

.Home {
  scroll-snap-type: y mandatary;
  overflow-y: scroll;
  height: 100vh;
}

.home {
```

```
.home_img {
  width: 100%;
  height: 90vh;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
  // background-color: $primary-color-faded;
  padding: 0px 240px;
  font-family: $poppins-font;
  h2 {
    font-size: 3.5rem;
    color: $secondary-color;

    text-align: center;
    margin-bottom: 30px;
    font-weight: 600;
  }
  p {
    margin-bottom: 30px;

    text-align: center;
    font-size: 1.2rem;
    color: $font-color-p;
  }
  a {
    text-decoration: none;
    color: white;
    background-color: $primary-color;
    border-radius: 2px;
    padding: 20px 50px;
```

```
      font-size: 1.2rem;
      font-weight: 500;
    }
    a:hover {
      background-color: $secondary-color;
    }
  }
}

.nav_home {
  height: 80vh;
}

.cards_container {
  display: grid;
  grid-template-columns: repeat(4, 1fr);
  grid-gap: 20px;
  //background-color: $primary-color;
  margin-top: 20px;
  padding: 100px;
}

#intro {
  display: grid;
  grid-template-columns: 60% 40%;
  padding: 0px 30px;
  margin-bottom: 50px;
  div {
    display: flex;
    flex-direction: row;
    justify-content: center;
    align-items: center;
    p {
      font-family: $tillium-font;
      font-weight: 500;
      color: $font-color-p;
      margin-right: 20px;
      font-size: 1.4rem;
      margin-bottom: 20px;
    }
  }
  img {
    width: 70%;
    margin: 30px 0px;
    margin-bottom: 40px;
  }
}

// tools
.tools {
  background: $tools-bg;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;

  h1,
  a {
    color: white;
    font-family: $tillium-font;
    font-weight: 300px;
    margin-bottom: 50px;
  }

  .cardscontainer {
    display: grid;
    grid-template-columns: 1fr 1fr 1fr;
    grid-gap: 50px;

    margin-bottom: 50px;

    .cards {
      background-color: $tools-card;
      display: flex;
      flex-direction: column;
      justify-content: center;
      align-items: center;
      width: 200px;
      height: 260px;

      i {
        font-size: 50px;
        color: $primary-color;
        margin-bottom: 20px;
      }
      h1,
      a {
        margin-bottom: 10px;
      }
    }
  }
}
```

```
.tools > h1 {
  font-size: 1.9rem;
}

//tools end

//github start
#github {
  background: $primary-color-faded;
  display: grid;
  grid-template-columns: 60% 40%;
  margin: 0px 40px;
  padding: 20px 40px;
  margin-bottom: 50px;

  h1,
  p,
  a {
    color: $font-color-h;
    font-family: $tillium-font;
    font-weight: 300px;
    margin: 0px;
    margin-bottom: 50px;
  }
  h1 {
    font-size: 2.2rem;
  }
  p {
    color: $font-color-p;
    font-size: 1.4rem;
  }
  img {
    width: 70%;
    margin-left: 60px;
  }
  a {
    text-decoration: none;
    color: black;
    font-size: 1rem;
    padding: 5px 4px;
    i {
      padding-right: 8px;
      color: black;
    }
    border-style: solid;

    border-color: black;
    border-width: 2px;
  }
  a:hover {
    i {
      color: $secondary-color;
    }
    color: $secondary-color;
    border-color: $secondary-color;
    box-shadow: $shadow-orange;
  }
}
//github end

//lapi
@media only screen and (max-width: 1024px) {
}
//tablet
@media only screen and (max-width: 768px) {
  .cards_container {
    grid-template-columns: repeat(2, 1fr);
  }

  #intro {
    display: flex;
    flex-direction: column-reverse;
    justify-content: center;
    align-items: center;
    p {
      text-align: center;
    }
    img {
      width: 50%;
    }

    .home_img {
      h2 {
        font-size: 2rem;
        color: aqua;
      }
    }
  }

  .tools {
    .cardscontainer {
      display: grid;
      grid-template-columns: 1fr 1fr 1fr 1fr;
      grid-template-rows: 1fr 1fr;
      grid-template-areas:
        "c1 c1 c2 c2"
        ". c3 c3 .";
    }
    .c1 {
      grid-area: c1;
    }
    .c2 {
      grid-area: c2;
    }
    .c3 {
      grid-area: c3;
    }
  }

  #github {
    display: flex;
    flex-direction: column-reverse;
    justify-content: center;
    align-items: center;
    img {
      width: 50%;
      margin-bottom: 20px;
      margin-left: 0px;
    }
  }
}

//mobile
@media only screen and (max-width: 425px) {
  .cards_container {
    grid-template-columns: 1fr;
  }
  .home_img {
    h2 {
      font-size: 2rem;
    }
    p {
      font-size: 1rem;
    }
  }

  .tools {
    .cardscontainer {
      display: flex;
      flex-direction: column;
    }
  }

  #github {
    h1 {
      font-size: 1.5rem;
      margin-bottom: 10px;
    }
    p {
      font-size: 1rem;
    }
  }
}
```

News.js

```
import React, { useState, useEffect } from 'react';


import Header from './Header'

import Card from './Card'

import './News.css'

import tracker_home from '../assests/tracker_home.jpg';

import Footer from './Footer';

import axios from 'axios';


const News = () => {


//     const [news, setNews] = useState("");

//   const [search, setSearch] = useState('');


//   useEffect(() => {

//     axios

//       .get(

//         'https://newsapi.org/v2/top-
headlines?country=us&language=en&apiKey=0540e3d2f70f498ea4ceae0b34322b96'

//       )

//       .then(res => {

//         setNews(res.data);

//         console.log(res.data);


//       })

//       .catch(error => console.log(error));

//   }, []);


//  const abc= res.data;
```

```
const key ="0540e3d2f70f498ea4ceae0b34322b96";
//   console.log(abc)
//   //news.map((new)=>(console.log(new.articles)))

const [items, setItems] = useState([])
const [isLoading, setIsLoading] = useState(true)
const [query, setQuery] = useState('')

useEffect(() => {
  const fetchItems = async () => {


    setIsLoading(true)
    const result = await axios(
      `https://newsapi.org/v2/everything?q=cryptocurrency&apiKey=${key}`
    )



    setItems(result.data)
    setIsLoading(false)

  }

  fetchItems()

}, )


if (!items.articles) {
  return <span>Loading...</span>;
```

```
}



    return (




        <div>
            <Header></Header>



            <h1>Top 3 Cryptocurrency News</h1>




            <div className="container">
                <div className="Card">


               <Card  image={items.articles[0].urlToImage}
               alt="Tracker"
               h2={items.articles[0].title}
               p={items.articles[0].description} />
                  </div>
               <div className="card-hori card-hori1">
                 <div>
                 <h2>{items.articles[1].title}</h2>
                 <p> {items.articles[1].description}</p>
                 <a href={items.articles[1].url}>LEARN MORE></a>
                 </div>
                 <img src={items.articles[1].urlToImage} alt="new world" />
```

```
                    </div>


                    <div className="card-hori card-hori2">
                    <div>
                    <h2>{items.articles[2].title}</h2>
                    <p> {items.articles[2].description}</p>
                    <a href={items.articles[2].url}>LEARN MORE></a>
                    </div>
                    <img src={items.articles[2].urlToImage} alt="new world" />





                    </div>

                 </div>

                 <Footer></Footer>






             </div>
        )
    }

export default News
```

News.css

```
@import "./colors.scss";

h1 {
  height: 5vh;
  color: $font-color-h;
  margin: 10px 20px;
}

.container {
  display: grid;
  padding: 20px;
  min-width: 100%;
  grid-template-areas:
    "Card cardhori1"
    "Card cardhori2";

  grid-gap: 30px;
}

.Card {
  grid-area: Card;
}

.card-hori {
  display: flex;
  flex-direction: row;
  img {
    width: 250px;
    height: 150px;
    border-radius: 5px;
    margin-bottom: 10px;
  }
  p {
    font-family: $tillium-font;
    font-weight: 600;
    color: $font-color-p;
    margin-right: 20px;
    font-size: 1.1rem;
    margin-bottom: 10px;
  }

  h2 {
    margin-right: 20px;
    margin-bottom: 10px;
    color: $font-color-h;
  }
  a {
    font-weight: bold;
    color: $link-color;
    text-decoration: none;
  }
  a:hover {
    border-bottom: solid;
  }
}

.card-hori1 {
  grid-area: cardhori1;
}
```

```
.card-hori2 {
  grid-area: cardhori2;
}


//lapi
@media only screen and (max-width: 1024px) {
}
//tablet
@media only screen and (max-width: 768px) {
  .container {
    grid-template-areas:
      "Card Card"
      "cardhori1 cardhori2";
  }


  .card-hori {
    flex-direction: column-reverse;
  }
}
@media only screen and (max-width: 425px) {
  .container {
    grid-template-areas:
      "Card "
      "cardhori1 "
      "cardhori2";
  }
}
```

Tracker.js

```
import React, { useState, useEffect } from 'react';
import axios from 'axios';
import './tracker.css';
import Coin from './Coin';
import Header from './Header';
import Footer from './Footer';

function App() {
  const [coins, setCoins] = useState([]);
  const [search, setSearch] = useState('');

  useEffect(() => {
    axios
      .get(
        'https://api.coingecko.com/api/v3/coins/markets?vs_currency=usd&order=
market_cap_desc&per_page=10&page=1&sparkline=false'
      )
      .then(res => {
        setCoins(res.data);
        console.log(res.data);
      })
      .catch(error => console.log(error));
  }, []);

  const handleChange = e => {
    setSearch(e.target.value);
  };

  const filteredCoins = coins.filter(coin =>
    coin.name.toLowerCase().includes(search.toLowerCase())
  );

  return (
      <div>  <Header></Header>


           <div className='coin-app'>


      {filteredCoins.map(coin => {
        return (
          <Coin
            key={coin.id}
            name={coin.name}
            price={coin.current_price}
            symbol={coin.symbol}
            marketcap={coin.total_volume}
            volume={coin.market_cap}
            image={coin.image}
            priceChange={coin.price_change_percentage_24h}
          />
        );
      })}
    </div>
    <Footer></Footer>
    </div>
  );
}

export default App;
```

Tracker.css

```
@import
url("https://fonts.googleapis.com/css2?family=Montserrat&display=swap");


.coin-app {
  display: flex;
  flex-direction: column;
  align-items: center;
  margin-top: 64px;
  color: black;
}

.coin-search {
  margin-bottom: 64px;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-content: center;
}

.coin-text {
  margin-bottom: 32px;
  text-align: center;
}

.coin-input {
  padding-left: 16px;
  width: 300px;
  height: 50px;
  border-radius: 4px;
  border: none;
  background-image: linear-gradient(
    -225deg,
    #ac32e4 0%,
    #7918f2 48%,
    #4801ff 100%
  );

  color: #e2e2e2;
}

.coin-input::placeholder {
  color: #e2e2e2;
}
```

5.3 Testing Approach

5.3.1 Unit Testing

Unit testing deals with testing a unit or module as a whole. This would test the interaction of

many functions but do confine the test within one module.

In our application we tested each module particularly, to see whether there are any bugs or error

while executing the module. Changes were made if any error was found. We went through a pattern of

testing which was defined in system testing chapter.

5.3.2 Integrated Testing

Integrated Testing brings all the modules together into a special testing environment, then checks

for errors, bugs and interoperability. It deals with tests for the entire application. After completing unit

testing, we combined all the modules together for testing the entire application. Through this technique

we came to know about the application limits and features are tested here.

5.4 Modifications and Improvements

- Easy access to the website.
- Admin can check the registered users.
- Make easier for users to buy the products.
- For more improvement’s users can give their feedback.

Chapter 6

Conclusion and Future Work

We successfully created that tracks cryptocurrencies.

We implemented react in our website.

In future we will add more feature like portfolio, graphs , etc.

Their will be a mobile application for this website in near future.

Limitations of the System

- ➢ This website cannot be used if the Internet Connection is off.
- ➢ At least, 1 mbps of internet connection is necessary.

Chapter 7

References

1. Guidance from Sir Omkar Sherkane

2. www.tutorialspoint.com

3. www.youtube.com

4. www.w3schools.com

5. www.stackoverflow.com