CRYPTOCURRENCY WALLET

An Internship Training and Summer Project Report submitted to

COMPUTER SCIENCE AND ENGINEERING

in partial fulfilment of the requirements

for the award of the Degree of

BACHELOR OF ENGINEERING

Submitted by

Manju Priya S (19101062)
Shahana R (19101098)
Sowmiya V (19101105)
Sowmya R (19101106)

Under the guidance of

Ms.Thamaraiselvi A., M.E., AP/CSE

Mr.A.Sabarishwaran, M.C.A., Technical Consultant

VIVEKANANDHA
COLLEGE OF ENGINEERING FOR WOMEN
[Autonomous]
Approved by AICTE, New Delhi and Accredited by NBA [CSE,ECE,EEE,IT,BT]
Affiliated to Anna University, Chennai-25,
Elayampalayam, Tiruchengode, Namakkal Dt. – 637205

NOVEMBER 2022

VIVEKANANDHA COLLEGE OF ENGINEERING FOR WOMEN

[Autonomous]

Elayampalayam, Tiruchengode, Namakkal Dt. – 637205

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Certificate

This is to certify that the Internship Training and Summer Project Report entitled **"CRYPTOCURRENCY WALLET",** in partial fulfilment of the requirements for the award the Degree of **BACHELOR OF ENGINEERING** is a record of original training undergone by **MANJU PRIYA S (19101062), SHAHANA R (19101098), SOWMIYA V (19101105),SOWMYA R(19101106)** during the year **2022** of her study in the Department of **COMPUTER SCIENCE AND ENGINEERING**, **Vivekanandha College of Engineering for Women** under my supervision and the report has not formed the basis for the award of any Degree/Fellowship or other similar title to any candidate of any University.

Place: Elayampalayam **Signature of Guide**

Date: **Ms.Thamaraiselvi A., M.E., AP/CSE**

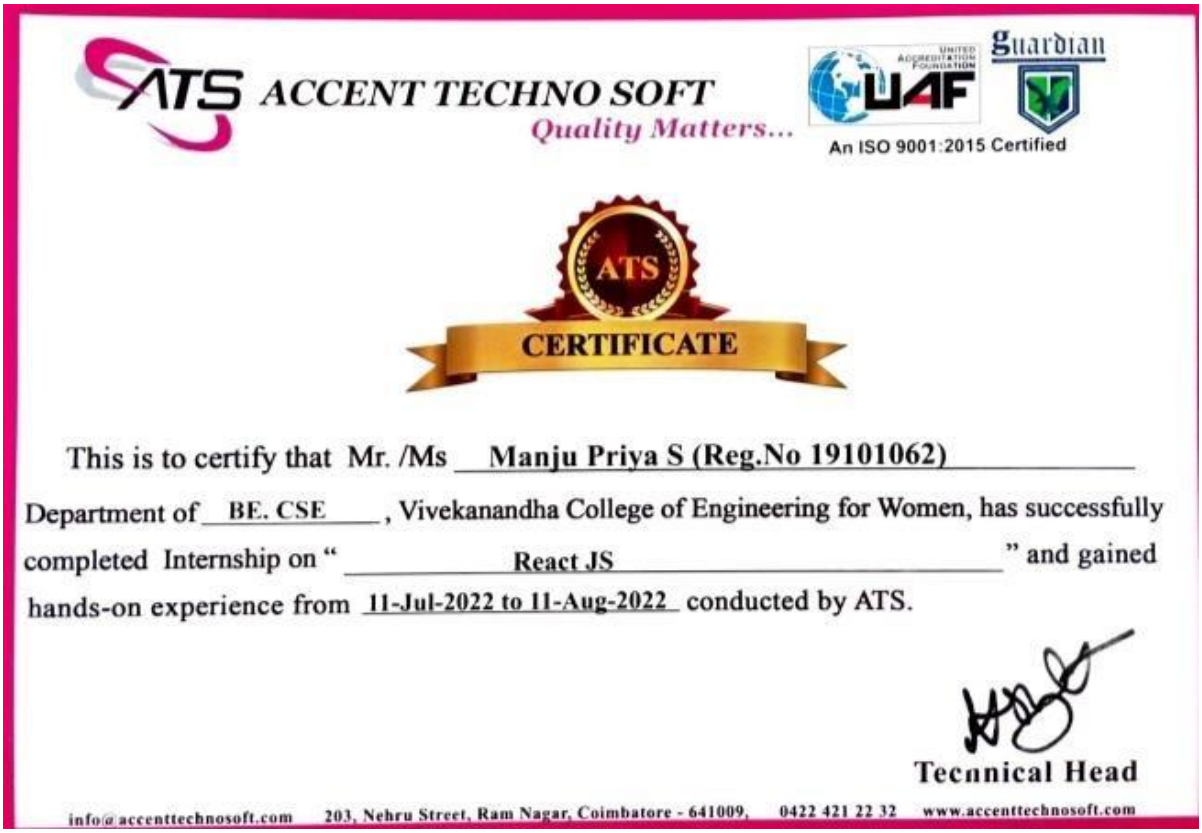
Head of the Department

Dr.C.POONGODI, M.E., Ph.D.

Submitted to the Department of **COMPUTER SCIENCE AND ENGINEERING**, **Vivekanandha College of Engineering for Women** for the examination held on __ _____

INTERNAL EXAMINER

Company Certificate



ACCENT TECHNO SOFT
Quality Matters...

An ISO 9001:2015 Certified

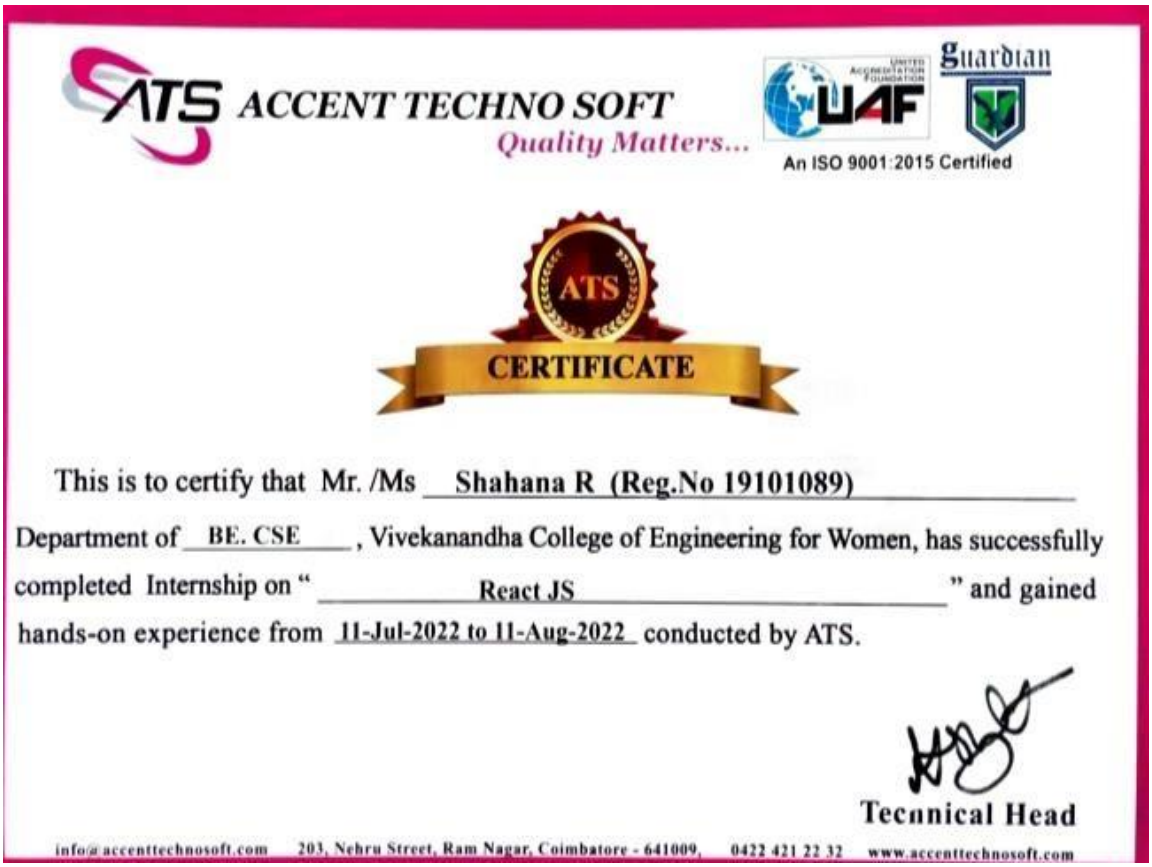
CERTIFICATE

This is to certify that Mr. /Ms **Manju Priya S (Reg.No 19101062)**

Department of **BE. CSE**, Vivekanandha College of Engineering for Women, has successfully completed Internship on " **React JS** " and gained hands-on experience from **11-Jul-2022 to 11-Aug-2022** conducted by ATS.

Technical Head

info@accenttechnosoft.com 203, Nehru Street, Ram Nagar, Coimbatore - 641009, 0422 421 22 32 www.accenttechnosoft.com



ACCENT TECHNO SOFT

Quality Matters...

An ISO 9001:2015 Certified

Guardian

ATS

CERTIFICATE

This is to certify that Mr. /Ms Shahana R (Reg.No 19101089)

Department of BE. CSE, Vivekanandha College of Engineering for Women, has successfully completed Internship on " React JS " and gained hands-on experience from 11-Jul-2022 to 11-Aug-2022 conducted by ATS.

Technical Head

info@accenttechnosoft.com 203, Nehru Street, Ram Nagar, Coimbatore - 641009, 0422 421 22 32 www.accenttechnosoft.com

ACCENT TECHNO SOFT

Quality Matters...

An ISO 9001:2015 Certified

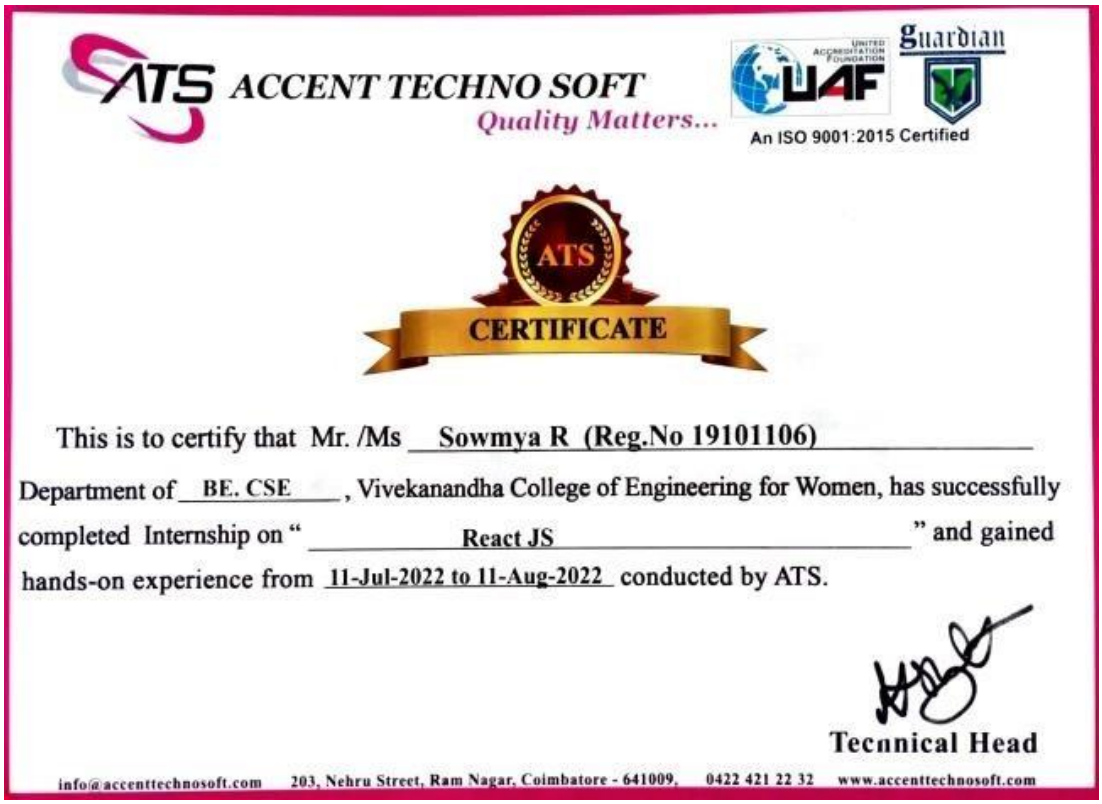
CERTIFICATE

This is to certify that Mr. /Ms **Sowmiya V (Reg.No 19101105)**

Department of **BE. CSE**, Vivekanandha College of Engineering for Women, has successfully completed Internship on "**React JS**" and gained hands-on experience from **11-Jul-2022 to 11-Aug-2022** conducted by ATS.

Technical Head

info@accenttechnosoft.com 203, Nehru Street, Ram Nagar, Coimbatore - 641009, 0422 421 22 32 www.accenttechnosoft.com


ACCENT TECHNO SOFT
Quality Matters...
An ISO 9001:2015 Certified
Guardian
ATS
CERTIFICATE
This is to certify that Mr. /Ms Sowmya R (Reg.No 19101106)
Department of BE. CSE , Vivekanandha College of Engineering for Women, has successfully completed Internship on " React JS " and gained hands-on experience from 11-Jul-2022 to 11-Aug-2022 conducted by ATS.
Technical Head
info@accenttechnosoft.com 203, Nehru Street, Ram Nagar, Coimbatore - 641009, 0422 421 22 32 www.accenttechnosoft.com

DECLARATION

We, **MANJU PRIYA S, SHAHANA R , SOWMIYA V, SOWMYA R** hereby declare that the Internship Training and Summer Project Report, entitled **" CRYPTOCURRENCY WALLET "** , submitted to the **Vivekanandha College of Engineering for Women** in partial fulfilment of the requirements for the award of the Degree of **BACHELOR OF ENGINEERING** is a record of original training undergone by me during the period **JULY 2022** under the supervision and guidance of **Ms.Thamaraiselvi A.,B.E.,M.E.,AP/CSE,** Department of **COMPUTER SCIENCE AND ENGINEERING** , **Vivekanandha College of Engineering for Women** and it has not formed the basis for the award of any Degree/Fellowship or other similar title to any candidate of any University.

Place: **Signature of the Students**

Date:

ACKNOWLEDGEMENT

We thank and praise the Lord Almighty for providing us Knowledge, strength and all the necessary facilities to do this report successfully.

We are immensely grateful to our **Chairman and Secretary**. "**VidyaRathna" Prof. Dr.M.KARUNANITHI, B.Pharm., M.S., Ph.D., D.Litt**., Vivekanandha Educational Institutions, who is our inspiration.

We are extremely grateful to our beloved **Executive Director Professor S.KUPPUSWAMY, B.E., M.Sc (ENGG.), Dr.Ing(France).,** for his motivation and guidance of our Internship Training and Summer Project.

We wish to express our profound thanks to our beloved Principal, **Dr.KCK.VIJAYAKUMAR, M.E., Ph.D., MIE,** for all the facilities and support provided during the period of our Internship Training and Summer Project.

We would like to acknowledge our **Head of the Department Dr.C.POONGODI, M.E., Ph.D.,** for the encouragement and support for completing the project successfully.

We wish to thank our Internship Training and Summer Project **Coordinators Mr.P.S.PRAKASH,B.E.,M.E.,(Ph.D)AP/CSE** and **Ms.S.SINDUJA,B.E.,M.E., AP/CSE** for their kind support and guidance in completion of our Internship Training and Summer Project.

We wish to thank our Internship Training and Summer Project **Faculty guideMs.Thamaraiselvi A M.E., AP/CSE** Departmentof Computer Science and Engineering, and **Industrial guide Mr.A.Sabarishwaran, M.C.A., Technical Consultant**for their kind support and guidance in completion of our Internship Training and Summer Project.

We are thankful and fortunate enough to get constant encouragement, support and guidance from our parents and all Teaching and non Teaching staff members of Department of Computer Science and Engineering who helped us in successfully completing our project.

ABSTRACT

A cryptocurrency wallet is an application that functions as a walletfor your cryptocurrency. It is called a wallet because it is used similarly to a wallet you put cash and cards in. Instead of holding these physical items, it stores the passkeys you use to sign for your cryptocurrency transactions and provides the interface that lets you access your crypto.Modern cryptocurrency wallets make the blockchain accessible to everyone. When cryptocurrency was first introduced, sending cryptocurrency was a manual task that required entering long keys. Today, the software does most of it for you. The first wallet was that of Bitcoin's developer, Satoshi Nakamoto. The second wallet belonged to Hal Finney, who corresponded with Nakamoto and reportedly was the first to run the Bitcoin client software wallet. Nakamoto sent him 10 bitcoin as a test, and the cryptocurrency craze began.

CONTENTS

CHAPTER NO	PARTICULARS	PAGE NO
i	CERTIFICATE	ii
ii	COMPANY CERTIFICATE	iii
iii	DECLARATION	vii
iv	ACKNOWLEDGEMENT	viii
v	ABSTRACT	11
vi	LIST OF FIGURES	12
vii	LIST OF ABBREVIATIONS	13
01	INTRODUCTION	14
	1.1 OVERVIEW	14
	1.2 OBJECTIVE	14
	1.3 SCOPE	14
02	COMPANY PROFILE	15
	2.1 ABOUT	15
	2.2 TECHNOLOGY TRAINING METHODOLOGIES	15
	2.3 TECHNOLOGY OFFERED	15
03	LITERATURE SURVEY	16
	3.1 LITERATURE REVIEW	16
	3.2 SUMMARY 0F LITERATURE SURVEY	18
04	SYSTEM DESIGN	19
	4.1 EXISTING SYSTEM	19

	4.2 PROPOSED SYSTEM	19
05	SYSTEM REQUIREMENTS	21
	5.1 HARDWARE REQUIREMENTS	21
	5.2 SOFTWARE REQUIREMENTS	21
06	MODULE DESCRIPTION	22
	6.1 NEED FOR CRYPTO	22
	6.2 CORE FEATURES	24
	6.3 PROPOSING FEATURES	27
	6.4 TECH STACK FOR CRYPTO WALLET DEVELOPMENT	29
	6.5 STAGES OF BUILDING CRYPTO WALLET	29
	6.6 WORKING OF CRYPTO	31
	6.7 ANALYSIS OF CRYPTO WALLET PROJECT	32
07	IMPLEMENTATION AND RESULTS	33
08	CONCLUSION & FUTURE WORK	48
09	REFERENCES	49

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
01	PROPOSED SYSTEM	07
02	NEED FOR CRYPTO CURRENCY	10
03	CORE FEATURES	11
04	ADVANCING FEATURES	14
05	TECH STACK FOR CRYPTO WALLET DEVELOPMENT	16
06	WORKING OF CRYPTO WALLET	18
07	ANALYSIS OF CRYPTO WALLET	19
08	INSTALLATION OF NPM	20
09	INSTALLATION OF REACT	21
10	PACKAGES	21
11	CRYPTOWALLET	41
12	ENCRYPTION	41

LIST OF ABBREVIATIONS

ABBRIEVIATED FORM	EXPANDED FORM
NFT	NON -FUNGIBLE TOKEN
UI/UX	USER INTERFACE
APP	APPLICATION
JSX	JAVA SCRIPT XML
API	APPLICATION PROGRAMMING INTERFACE
MPC	MARGINAL PROPENSITY TO CONSUME

CHAPTER-1

INTRODUCTION

1.1 OVERVIEW

A cryptocurrency wallet refers to a physical medium, device, service, or application that maintains private and/or public passwords for crypto transactions. In addition to the basic purpose of storing keys, it also makes the owner of the digital pseudonymous.

It comes in various forms – from hardware crypto wallets like the Ledger, which resembles a USB stick, to applications on mobile devices that make purchasing and holding digital assets, such as cryptocurrencies, as easy as using a credit card online.

1.2 OBJECTIVE

The main purpose of cryptocurrency is to reduce the risk involved in traditional currency . It is very easy to use and can access it anywhere and anytime .All need is to have smart phone and good connection . In cryptocurrency the power and the responsibilities are in the hands of the currency holder , and they help in solving real world problems .

1.3 SCOPE

Crypto wallets help to store confidential keys, maintaining crypto safe and available. Inshort, is nothing but a secure place to store your proof of ownership. They also permit todeliver, receive, and spend cryptocurrencies like Bitcoin, Ethereum, and many more. Cryptowallets vary from simple-to-use apps to more complicated security solutions

CHAPTER-2

COMPANY PROFILE

Accent Techno Soft (ATS) provides a wide range of solutions in IT Consulting, technology and Operations space for our clients. To enhance the business value of our service offerings to our customers, we have formed strategic alliances with industry bodies, technologies vendors and system integrators. Through these partnerships we are able to deliver industry-best end–toend solutions to our customers.

2.1 TECHNOLOGY TRAINING METHODOLOGIES

ATS is unique in its training and delivery methodology as it is the first in India to provide the participant hi -end technology training followed by a rigorous internship in its development center. This gives the participant a real time software development experience and makes him a “Complete Software Professional”.

- Corporate Training
- IT Training for Students & Professional
- Internship Program
- Workshops
- Industrial Visit / In-Plant Training
- Campus to Corporate Connect

2.2 TECHNOLOGIES OFFERED

- JAVA/J2EE
- Rich Internet Application (RIA)
- .Net
- Web Designing
- PHP/MySQL
- Software Testing
- Hardware & Networking
- Data Warehousing
- Perl/Python

CHAPTER-3

LITERATURE SURVEY

3.1 LITERATURE REVIEW

Blockchain and Cryptocurrencies: Model, Techniques, and Applications[2018]

General idea

A survey of current cryptocurrencies to understand blockchain & its different types.

Advantages: Provides different incentive models, ecosystem & applications of the blockchain. Explains blockchain in a layered architecture. Limitations: Does not provide any solid architecture for its stated application.

A Brief Survey of Cryptocurrency Systems[2017]

General idea

It evaluates the strengths, weaknesses, and possible threats to all major mining strategy. It outlines how Cryptocurrencies mine, where they have comparable performance and assurance, and where they have unique threats and strengths

Advantages: 1. Currently, major Cryptocurrencies use Proof of Work, Proof of Stake or a combination of the both for mining. 2. A combination of the both is found to be effective. 3. Typically memory-intensive hash functions have been found to be faster mining algorithms. Limitations: A majority of hash algorithms are CPU-intensive and the others are memory intensive. 2. While Proof of Work is resource intensive, Proof of Stake cannot act independently. 3. Cryptocurrencies are still experimenting with their mining protocols and algorithms to optimize their performance. No full proof algorithm has been found yet

Blockchain: Future of Financial and Cyber Security[2016]

General idea

This paper explains the concept, characteristics, need for Blockchain and how Bitcoin works. It attempts to highlights role of Blockchain in shaping the future of banking.

Advantages:1.The decrease in device cost 2.Increases computing power

Limitations: 1. If an attack was done by an attacker then there will be a loss of all bitcoins, we can't recover it because the government is not involved in

Bitcoin: A Peer-to-Peer Electronic Cash System[2008]

General idea

A distributed peer to peer system working under the blockchain framework

Advantages: Cryptocurrency without any central authority. Successful POW mechanism. Limitations: The cost of POW consensus protocol will keep increasing as more people join the network.

Trust Your Wallet : A New Online Wallet Architecture for Bitcoin[2017]

General idea

It introduces a wallet which is highly secured by Multiple signatures.

Advantages: The scalability of disaster recovery center Limitations: If we lost one of the keys then we are not able to recover that key.

A survey on the security of blockchain systems[2017]

General idea

Detail survey of the security issues in current systems and existing solutions

Advantages: A careful comparison between bitcoin and ethereum. Different aspects of system vulnerability Limitations: Cryptocurrency will need more methods to achieve security and privacy..

3.1.SUMMARY OF THE SURVEY

The above content describes the literature survey. This describes about the details about the paper with the general idea of the paper with its advantages & limitations. It helps to understand the workings of the current system & the limitations of the system which are required to overcome for betterment of the system. The literature survey helps us to bring up with more new features to the current existing system of cryptocurrency wallet.

CHAPTER-4

SYSTEM DESIGN

Software design is a way through which the requirements can be converted into a proper representation. This representation can be used to create architecture and detailed design diagrams through which further process of the project can be carried out.

4.1.EXISTING SYSTEM

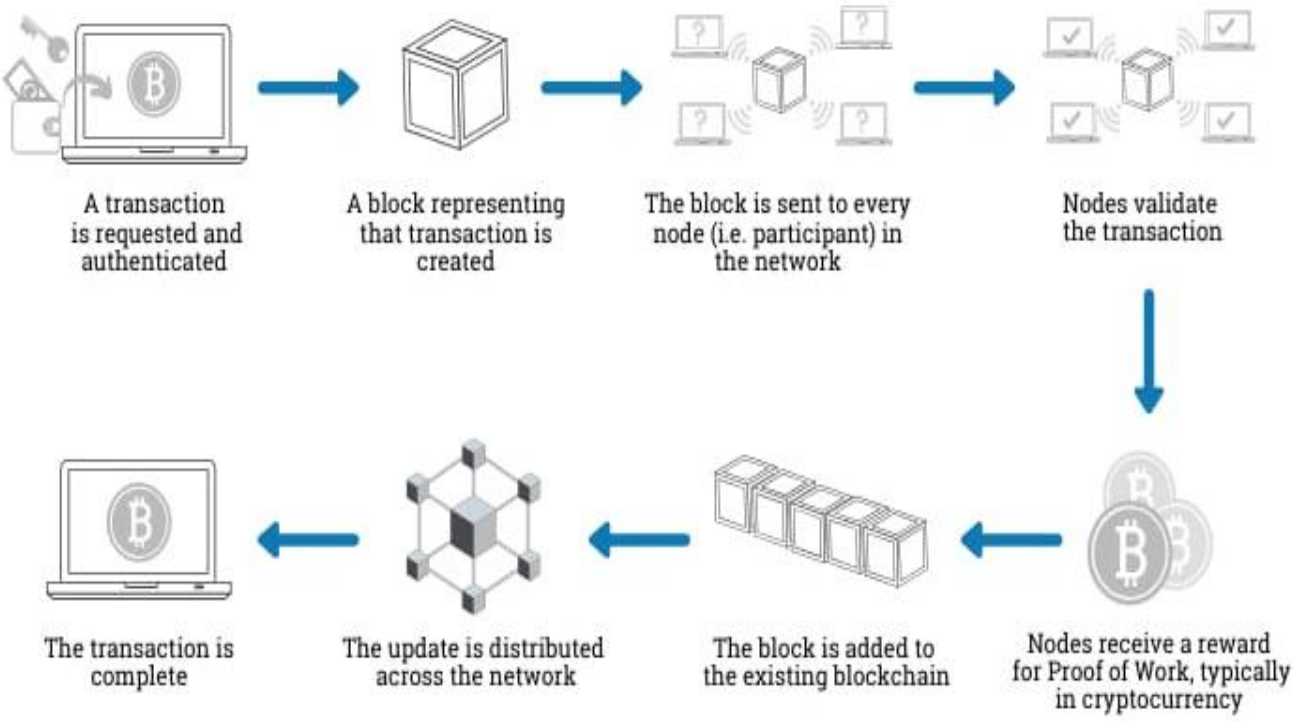
Cryptocurrency wallets are software applications on computers or mobile devices such as phones or tablets. They use an internet connection to access the blockchain network for the cryptocurrency you're using.

Cryptocurrencies are not "stored" anywhere—they are bits of data stored in a database. These bits of data are scattered all over the database; the wallet finds all of the bits associated with your public address and sums up the amount for you in the app's interface.

Sending and receiving cryptocurrency is very easy using these applications. You can send or receive cryptocurrency from your wallet using various methods. Typically, you enter the recipient's wallet address, choose an amount to send, sign the transaction using your private key, add an amount to pay the transaction fee, and send it.

4.2.PROPOSED SYSTEM

As shown in figure, suppose party A wants to transfer some currency over to party B. Then by the wallet interaction, party A can start the transaction over to party B. The transaction is represented online as a block. The block is then broadcasted to every party in the network. The network as a whole will approve the transaction to be valid or invalid by using the distributed ledger. Then by some consensus protocol block will get added to the distributed ledger. In this way, the transaction for transferring money from A to B is completed successfully



4.1 System Archietechture

Activity diagram:

In figure , user access wallet website and authenticate by referencing the database. After a successfullogin, the user can input transaction details HERE BEFORE THE USER PROVIDING INPUT TRANSACTION DETAILS USER IS GIVEN A PASSWORD TO START A TRANSACTION and then start a transaction. The important details are verified with database and transaction is created in cryptocurrency network. The miners will then verify and validate the transactions. After that block gets added into blockchain. The changes are acknowledged by cryptocurrency model. The transaction details are recorded into the database. The result is displayed to the user by the wallet

CHAPTER-5

SYSTEM REQUIREMENTS

5.1. HARDWARE REQUIREMENTS

The hardware requirements for executing this model are:

- RAM – 4 GB
- Processor – Intel(R) Core(TM) i3
- Processor speed – 3.60 GHz

5.2. SOFTWARE REQUIREMENTS

The programming language used to develop this application is Python and the IDE used is Spyder Notebook.

- Operating System – Windows 10
- Programming Language – front end ReactJS
- HTML Hypertext Markup Language
- NPM Node Package Manager
- NPX Node Package Execute
- UI/UX DESIGN
- VISUAL STUDIO CODE

CHAPTER-6

MODULE DESCRIPTION

The system can be divided into six modules.

Module1: NEED FOR CRYPTOCURRENCY WALLET
Module2: CORE FEATURE
Module3: PROPOSING FEATURE
Module4: TECHNOLOGY FOR WEB APP
Module5: STAGES OF IMPLEMENTATION
Module6: PROJECT ANALYSIS

6.1. NEED FOR CRYPTOCURRENCY WALLET DEVELOPMENT

In the last few years, we have witnessed a rapid growth of the cryptocurrency market. Total market capitalizationpeaked in 2021 and continues to surprise traders and investors today. The involvement of both companies and ordinary users in the cryptocurrency market increases, which means that more and more people in the world get access to this financial instrument. As a result, the demand for high-quality and reliably protected solutions for storing crypto assets rises. According to Zion Market Research, the market of crypto wallets is expected to grow to 686 million in the next 6 years.

The crypto wallet is designed to solve current problems, such as unreliable token storage on exchanges, and complex transaction and conversion systems. In a high-quality cryptocurrency wallet application, tokens can be transferred from one address to another or converted in just a few clicks. The internal security protocol takes care of the security of the funds, and seed phrases and strong passwords, which only the user has access to, create an additional layer of protection

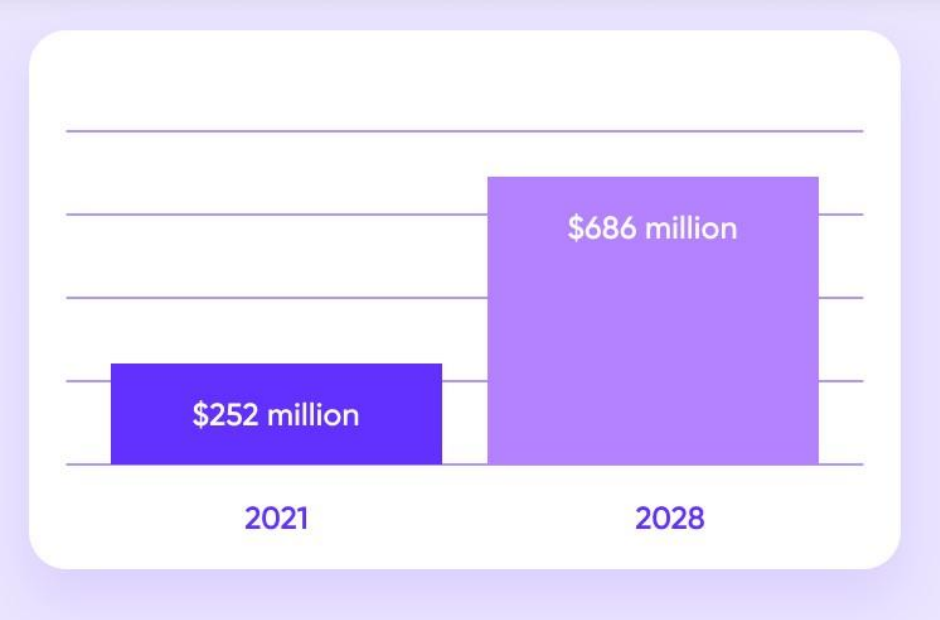


Figure 6.1The statistics expect the market to grow

From a developer's perspective, it's profitable to make a cryptocurrency wallet application because they can earn money on commissions during transfers and conversions of currencies and advertising. That's why the idea to create a cryptocurrency wallet today seems promising.

6.2.Core features

6.2 core features

What functions should a cryptocurrency wallet application have? As a rule, it is created to avoid problems that are associated with trading and storing funds on the crypto exchange or marketplace and to enhance asset security. Based on this, the main requirements for the application are:

A unique access key. Every wallet should provide its users with unique public and private keys. There should be an opportunity to add a personal password or two-factor authentication.

Session duration. You need to limit the session duration for security purposes. This practice exists in banking applications. When some time passes without user activity, the application closes automatically. To resume the session, a user needs to re-enter the login password.

Several supported cryptocurrencies. There are single-currency crypto wallets, such as Bitcoin wallet or Ethereum wallet, but now their popularity is not as high as that of multi-currency alternatives. It is often convenient for investors and traders to have a diversified portfolio of assets and store them in one place. The most efficient crypto wallets constantly add new tokens to their list due to the growing demand from the crypto community.

Buying and selling cryptocurrencies. The crypto wallet application should make it possible to buy and sell cryptocurrency assets, execute buy and sell orders, and organize trades between users.

Transaction options. You need to allow sending cryptocurrencies to other wallets using the wallet number (public key). The transfer of cryptocurrencies from one user to another is an integral part of every cryptocurrency wallet application.

QR codes. QR codes simplify functions such as making a transaction, receiving a transfer of assets to your address, logging into the crypto wallet app, securely authorizing a user on the platform, or confirming any other actions.

Push notifications. Notifications may include one-time codes to confirm transactions, as well as information about successful crediting or debiting assets or an error. The push notification is important for maintaining security. If some assets have been debited from the account without the user's consent, the notification will immediately inform the owner about it.

Constant updates of the conversion rate. Due to the high volatility of crypto assets, the exchange rate is constantly changing. Crypto wallets allow users to transfer money in different ways, for example, in Bitcoin or altcoins. Your application needs to synchronize with the blockchain network and show up-to-date information about conversion rates at the time of the transaction.

Compatibility with iOS and Android. Launch the app simultaneously on the most popular operating systems: iOS and Android. You can achieve it withReact Nativethat helps developers use one code base to develop two apps.

6.3 ADVANCING FEATURES

Besides the main mandatory functions, you can also stand out from the competition by adding the following features:

6.3 advancing features

Commission calculation. Some crypto wallets have a field for calculating the commission or they automatically show how much you need to pay depending on the transaction amount. This feature adds convenience to the user and makes the system more transparent.

Conversion of fiat into cryptocurrency. Not all wallets allow you to convert US dollars or other fiat currencies into cryptocurrencies. However, such a feature can add popularity to your wallet and perfectly suit beginners in the world of crypto.

Payment gateways. A crypto payment gateway is a tool that allows users to pay with crypto for goods and services and merchants to accept crypto payments. Advanced crypto payment providers offer to automatically convert assets into fiat currencies.

Collaboration with cryptocurrency exchanges (DEXes). Many wallets need to be connected to the web version of a decentralized exchange orNFT marketplaceto buy and sell assets on this website. A simplified or automatic connection to popular exchangers will be a great addition to your service.

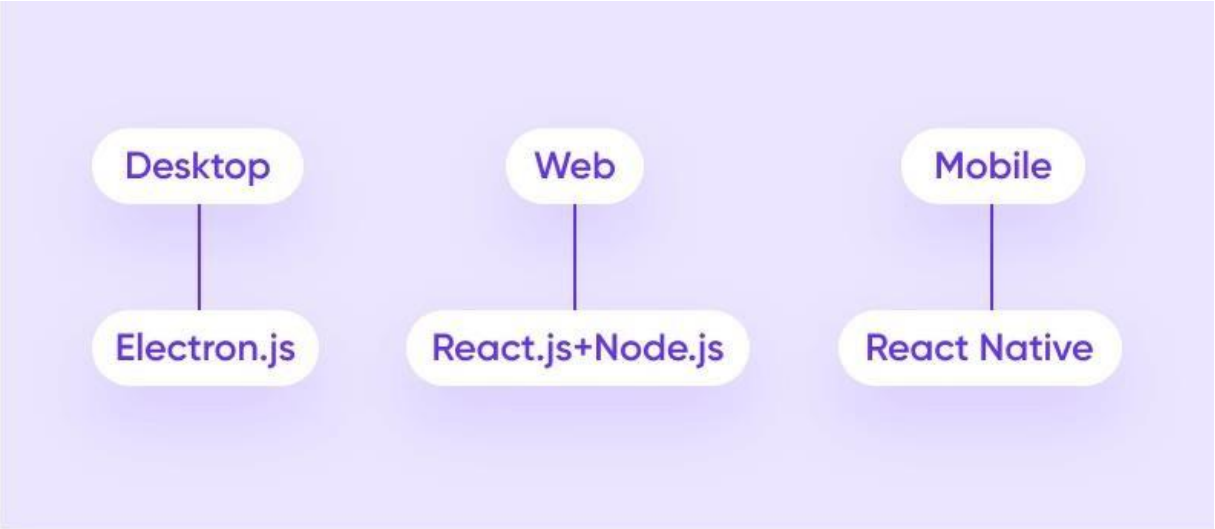
Several accounts for cryptocurrencies. To diversify the portfolio, investors and traders buy and store several different cryptocurrencies at the same time. You can add the ability to create multiple accounts for different currencies to your crypto wallet application. So there will be a division between assets.

Staking. Many platforms, including Binance and Coinbase, have added the ability to stake cryptocurrencies using a cryptocurrency wallet. Storing cryptocurrencies to maintain network performance and receive rewards for this is a popular passive income among crypto enthusiasts.

6.4 Tech stack for crypto wallet development

To develop a crypto wallet application, you will need to choose either a desktop, a web or a mobile app. Let's explore some useful and advanced technologies that can be used for every type of app.

React.js and Node.js for web apps. React.js and Node.js are open-source JavaScript libraries for web applications. React is a popular frontend library for creating interactive user interfaces. It offers regular updates and a convenient toolset. Node.js is a backend development environment that allows specialists to create scalable web applications with high performance, flexibility, and interoperability.



6.4 Tech stack for crypto wallet development

6.5 How to build a crypto wallet in 5 steps

Let’s take a look at the crypto wallet app development stages.

Step 1. Analyze the project. The development of both simple and more complex crypto wallets begins with the analysis of available data adjusted for user requests. Share your idea, define a target audience, choose the app’s features, define and set the terms of reference, and set timeframes and a budget for the crypto wallet development.

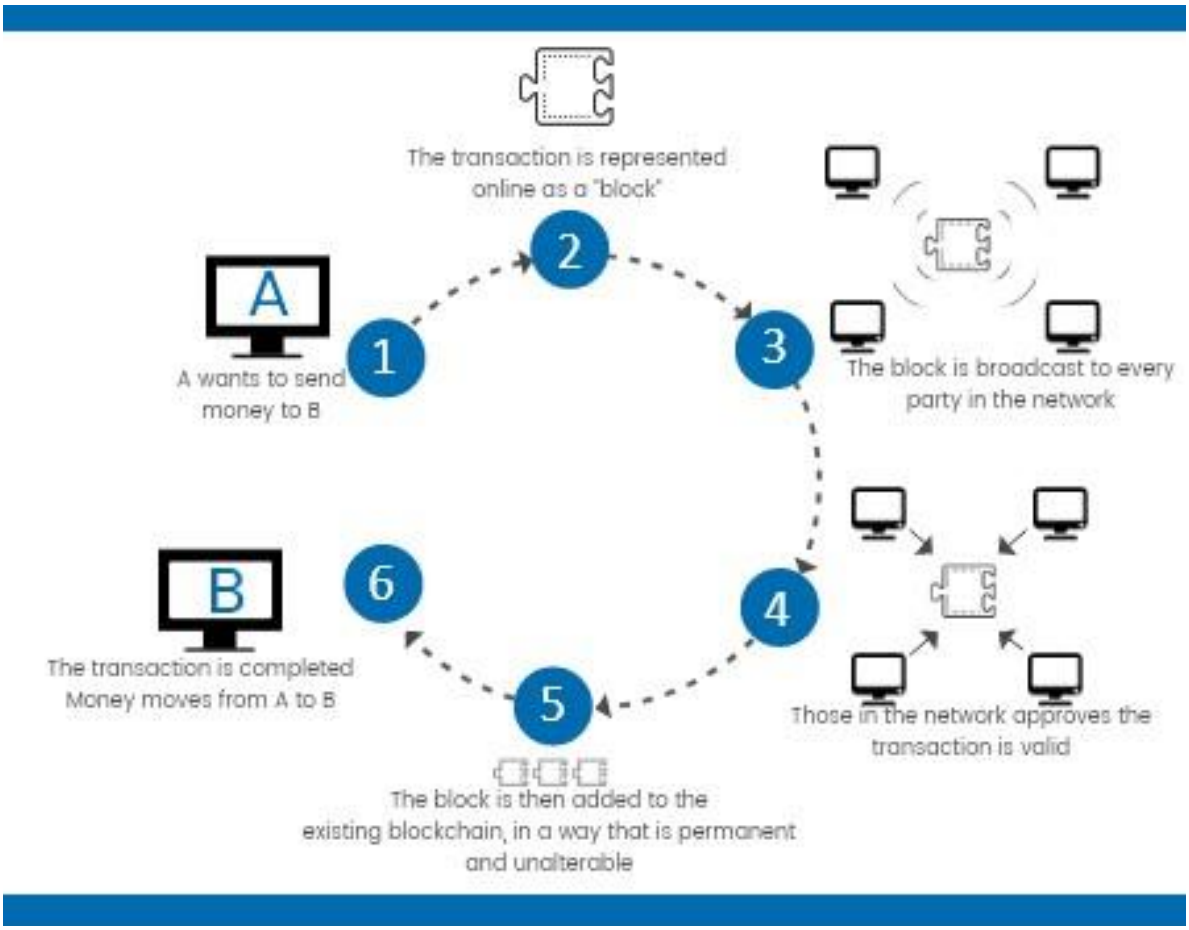
Step 2. Create a design. You need to find design references, set your expectations for the future app, and design the crypto wallet application with your UI/UX team.

Step 3. Develop a web or mobile application. Find developers that will work on architecture and the interface on both the client and server sides.

Step 4. Test the app. Testing will allow you to control all aspects of the application. You need to have the QA testers look for any possible bugs and fix them before the app is released.

Step 5. Release the app and get feedback. Launch your crypto wallet app and receive feedback from the first users. Then, you can come up with ideas to improve the app**.**

6.6 working of crypto wallet



6.6 working of crypto wallet

Blockchain is an digital database which keeps an unchangeable record of data operations. Group of these operations called “blocks”. Here is general flow of how it works:

If someone want to send money using crypto wallet. Transaction is represented as block.

1. The Blocks which is broadcast to every party in network.
2. Those in the network approves the transaction is valid or not.
3. If block is valid, it will be assigned to receiving crypto wallet
4. Received block is permanent and unalterable.
6. Commit the transaction

6.7 ANALYSIS OF CRYPTO WALLET PROJECT

The application development market offers many solutions of different price categories. Our company can share examples of ready-made crypto wallet applications.

The estimations the cost of the project from the first day of the customer's request. We need to discuss all aspects of the work, app functions, preferred design, and personal wishes. If you are going to develop applications for a crypto wallet, you should familiarize yourself with the action plan. Based on the core features discussed above, our team estimates the pace and timing of crypto wallet development as follows:

Stage	Estimation in hours	Estimation in weeks	Approximate costs
Project analysis	0	0	$0
UI/UX design	137 hours	5 weeks	$6 840
App development	1050 hours	12 weeks	$62 100
QA (Quality Assurance)	360 hours	in parallel with the development	$7 200
Project management	during the whole project	during the whole project	$4 950

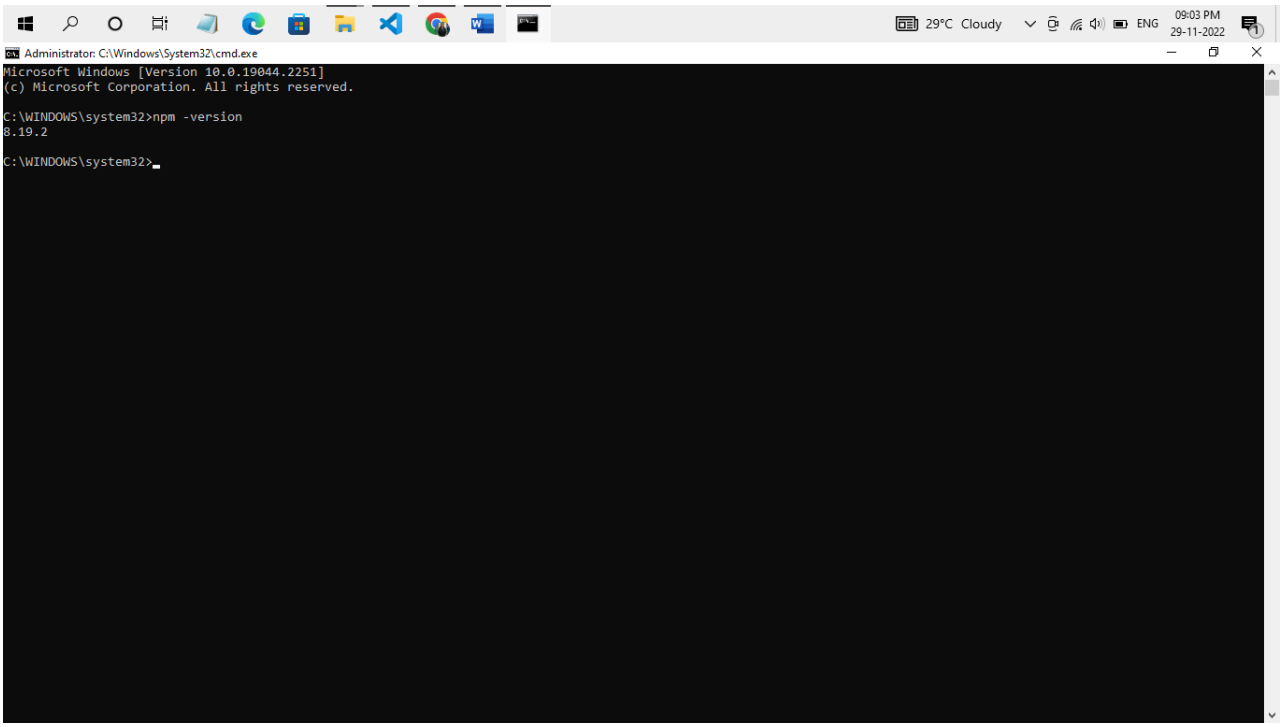
6.7 analysis of crypto wallet project

CHAPTER-7

IMPLEMENTATION AND RESULTS

In the process of implementation ,it is required to install the software which is node,js from which it is accessible for react.js

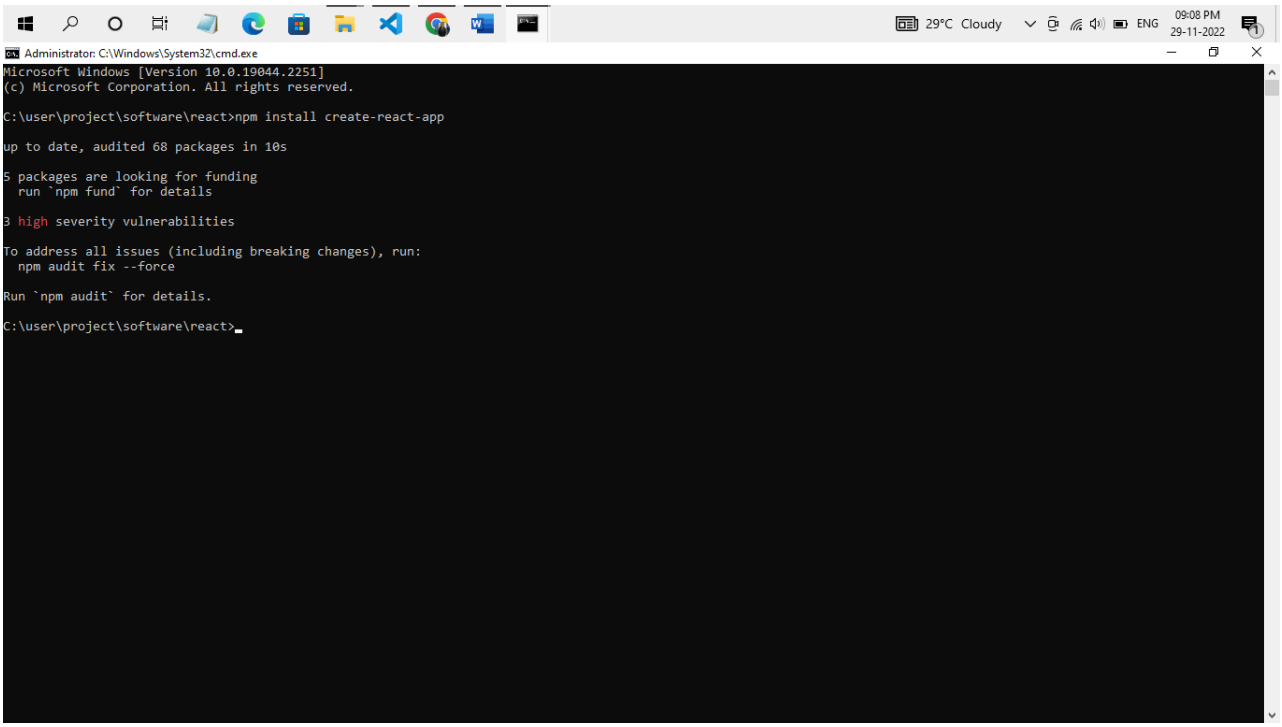
Installation process of npm



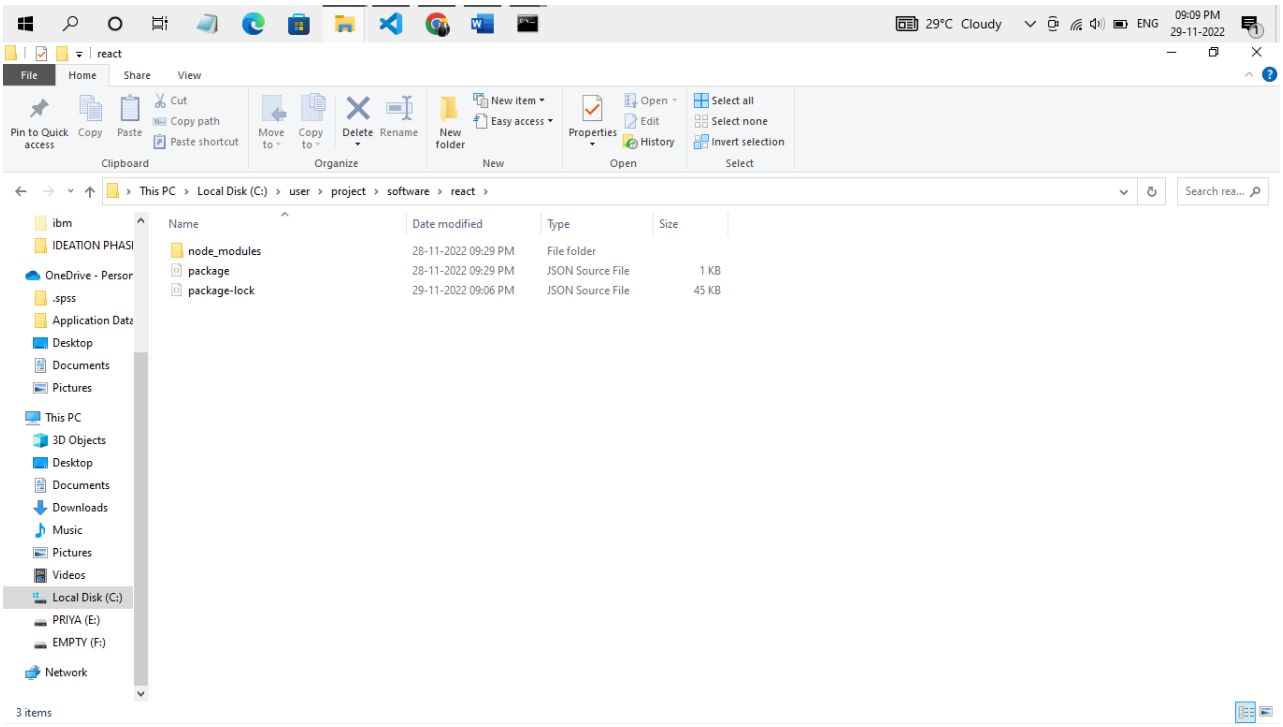
7.1 Installation process of npm

Installation of react.js

Command to install react npx create-react-app my-app



7.2 Installation of react.js



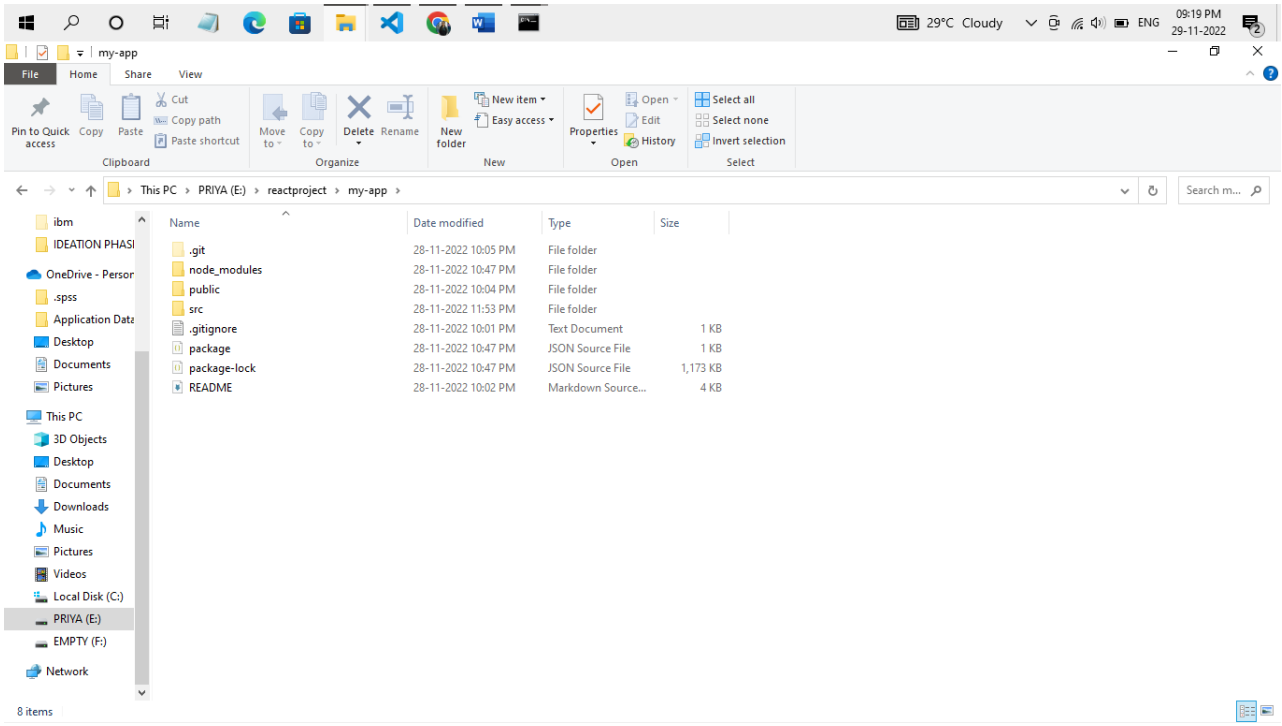
7.3 packages are installed

Create React App is a comfortable environment for **learning React**, and is the best way to start building **a new single-page application** in React.

It sets up your development environment so that you can use the latest JavaScript features and need to have Node >= 14.0.0 and npm>= 5.6 on your machine. To create a project, run:

```
npx create-react-app my-app
```

Command used to create a react app
npx create-react-app my-app
cd my-app
npm



7.4 folder of the reactapp

Once the application is installed we can create a react app using the above command and the packages are been installed within the folder by using command promt .

And coding can be implemented with help of visual code where each blocks are having a separate coding which is presented below

After you click on Sign, you will see the public key from the wallet on the screen.

Public

index.html

```
!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <meta name="theme-color" content="#000000" />
    <meta
      name="description"
      content="Web site created using create-react-app"
    />
    <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />
    <!--
      manifest.json provides metadata used when your web app is installed on a
      user's mobile device or desktop. See
https://developers.google.com/web/fundamentals/web-app-manifest/
    -->
    <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
    <!--
      Notice the use of %PUBLIC_URL% in the tags above.
      It will be replaced with the URL of the `public` folder during the build.
      Only files inside the `public` folder can be referenced from the HTML.

      Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC_URL%/favicon.ico" will
      work correctly both with client-side routing and a non-root public URL.
      Learn how to configure a non-root public URL by running `npm run build`.
    -->
    <link href="https://cdn.jsdelivr.net/npm/remixicon@2.5.0/fonts/remixicon.css"
rel="stylesheet">
    <title>React App</title>
  </head>
  <body>
    <noscript>You need to enable JavaScript to run this app.</noscript>
    <div id="root"></div>
    <!--
      This HTML file is a template.
      If you open it directly in the browser, you will see an empty page.

      You can add webfonts, meta tags, or analytics to this file.
      The build step will place the bundled scripts into the <body> tag.

      To begin the development, run `npm start` or `yarn start`.
      To create a production bundle, use `npm run build` or `yarn build`.
    -->
  </body>
```

```
</html>
```

Src

Src>components

Src>components>convertdetails>

Src>components>convertdetails>convertdetails.jsx

```
import ConversionCard from "components/ConversionCard/ConversionCard";
import "./ConvertDetails.scss";
constConvertDetails = () => {
  return (
    <div className="ConvertDetails Flex Col">
      <div className="ConvertDetailsRow">
        {[1, 2, 3].map((item, i) => (
          <ConversionCard key={i} />
        ))}
      </div>
    </div>
  );
};

export default ConvertDetails;
```

Src>components>convertdetails>convertdetails.scss

```
ConvertDetails {
    grid-area: convert-details;
    overflow-x: auto;
    .ConvertDetailsRow {
      display: flex;
      flex-direction: flex;
    }
  }

  @mediaonlyscreen and (min-width: 1170px) {
    .ConvertDetails {
      padding-top: 50px;
      .ConvertDetailsRow {
        flex-direction: row;
        width: max-content;
      }
    }
  }
```

Src>components>conversationcard>

Src>components>conversationcard>conversationcard.jsx

```
import"./ConversionCard.scss";
constConversionCard = () => {
  return (
    <divclassName="ConversionCard Flex Col">
      <divclassName="Flex Row">
        <divclassName="Flex Col Mt20">
          <divclassName="ConversionToFrom Flex ItemsCenter">
            <div>
              <divclassName="LabelBlock">
                <pclassName="Label">Bitcoin</p>
                <pclassName="Name">5.04 BTC ($53 378.91)</p>
              </div>
            </div>
            <divclassName="Icon BgLightGray2 Flex JustifyCenterItemsCenterMlAuto
Mr5">
              <iclassName="ri-bit-coin-fill"></i>
            </div>
            <iclassName="ri-arrow-down-s-line ColorDarkGray3"></i>
          </div>
          <divclassName="My30">
            <pclassName="Font600 FontSize10 ColorLightGray1">Your spend</p>
            <divclassName="Flex Row WFullJustifyBetween">
              <pclassName="FontTransformY ColorDarkGray3 FontSize1-4 Font600">
                3.000000
              </p>
              <pclassName="FontTransformY ColorDarkGray3 FontSize1-4 Font600">
                BTC
              </p>
            </div>
            <hrclassName="Purple"/>
            <divclassName="Flex Row WFullJustifyBetween">
              <pclassName="FontLato Font600 FontSize10 ColorLightGray3">
                19 392.20
              </p>
              <pclassName="FontLato Font600 FontSize10 ColorLightGray3">USD</p>
            </div>
          </div>
        </div>
        <divclassName="Flex Col ItemsCenter Mx30">
          <divclassName="Flex VLine"></div>
          <divclassName="Icon BgDarkGray3 Flex JustifyCenterItemsCenter">
            <iclassName="ri-repeat-2-fill ColorLightGray4"></i>
          </div>
          <divclassName="Flex VLine"></div>
        </div>
```

```
        <divclassName="Flex Col Mt20">
          <divclassName="ConversionToFrom Flex ItemsCenter">
            <div>
              <divclassName="LabelBlock">
                <pclassName="Label">Ethereum</p>
                <pclassName="Name">160 ETH ($32 672.99)</p>
              </div>
            </div>
            <divclassName="Icon BgLightGray2 Flex JustifyCenterItemsCenterMlAuto
Mr5">
              <iclassName="ri-copper-diamond-fill"></i>
            </div>
            <iclassName="ri-arrow-down-s-line ColorDarkGray3"></i>
          </div>
          <divclassName="My30">
            <pclassName="Font600 FontSize10 ColorLightGray1">You get</p>
            <divclassName="Flex Row WFullJustifyBetween">
              <pclassName="FontTransformY ColorLightGray5 FontSize1-4 Font600">
                94.426101
              </p>
              <pclassName="FontTransformY ColorLightGray5 FontSize1-4 Font600">
                ETH
              </p>
            </div>
            <hrclassName="Light"/>
            <divclassName="Flex Row WFullJustifyBetween">
              <pclassName="FontLato Font600 FontSize10 ColorLightGray3">
                19 392.20
              </p>
              <pclassName="FontLato Font600 FontSize10 ColorLightGray3">USD</p>
            </div>
          </div>
        </div>
      </div>
      <divclassName="Flex Row ItemsCenter Mt40">
        <div>
          <pclassName="Font600 FontSize10 ColorLightGray1 Mb5">Market rate</p>
          <pclassName="FontLatoFontTransformY Font700 FontSize12 ColorDarkGray3">
            1 BTC = 31.47 ETH
          </p>
        </div>
        <divclassName="Ml30">
          <pclassName="Font600 FontSize10 ColorLightGray1 Mb5">
            Conversion fee
          </p>
          <pclassName="Font700 FontSize12 ColorDarkGray3 FontTransformYFontLato">
            12.938539 BNB
          </p>
        </div>
```

```
        <buttonclassName="MlAutoConvertBtn">Convert Now</button>
      </div>
    </div>
  );
};

exportdefaultConversionCard;
```

Src>components>conversationcard>conversationcard.scss

```
@import"/src/Colors.scss";

.ConversionCard {
  background: $light-gray-7;
  margin-right: 40px;
  padding: 40px30px;
  position: relative;
  hr {
    border-bottom: 0;
    margin: 5px0;
  }
  hr.Purple {
    border-top: 2pxsolid$light-purple;
  }
  hr.Light {
    border-top: 2pxsolid$light-gray-2;
  }
  .ConversionToFrom {
    background: $white;
    padding: 20px;
    .LabelBlock {
      margin-right: 20px;
    }
    .Name {
      font-family: "Lato";
      font-weight: 600;
      font-size: 10px;
      color: $light-gray-3;
      margin-top: 5px;
    }
    .Icon {
      padding: 10px;
      border-radius: 50%;
      color: $light-purple-2;
      font-size: 1.2rem;
    }
  }
  .VLine {
```

```
    border-left: 1pxsolid$light-gray-2;
    height: 40%;
  }
  .Icon {
    padding: 10px;
    border-radius: 50%;
  }
  .ConvertBtn {
    position: absolute;
    background: $dark-gray-3;
    color: $light-gray-2;
    border: none;
    padding: 15px45px;
    font-weight: 600;
    right: -20px;
  }
}
```

Src>components>navbar

Src>components>navbar>navbar.jsx

```
import"./NavBar.scss";

constNavBar = () => {
  return (
    <navclassName="NavBar">
      <li>
        <iclassName="ri-stack-fill"></i>
      </li>
      <li>
        <iclassName="ri-layout-masonry-line"></i>
      </li>
      <li>
        <iclassName="ri-repeat-2-fill active"></i>
      </li>
      <li>
        <iclassName="ri-layout-top-line"></i>
      </li>
      <li>
        <iclassName="ri-checkbox-blank-line"></i>
      </li>
      <li>
        <iclassName="ri-user-line"></i>
      </li>
    </nav>
  );
```

```
};

exportdefaultNavBar;
```

Src>components>navbar

Src>components>navbar>navbar.scss

```
@import"/src/Colors.scss";

.NavBar {
  display: none;
  grid-area: nav;
  background: $dark-gray-1;
  flex-direction: column;
  justify-content: space-between;
  align-items: center;
  padding: 50px0;
  width: 70px;
  border-radius: 30px;
  li {
    list-style: none;
  }
  i {
    color: $dark-gray-2;
    font-size: 1rem;
    position: relative;
    display: flex;
    align-items: center;
    flex-direction: column;
  }
  li:first-childi {
    color: $white;
  }
  i.active::after {
    position: absolute;
    content: "";
    width: 4px;
    height: 4px;
    background: $white;
    border-radius: 50%;
    bottom: -8px;
  }
}

@mediaonlyscreen and (min-width: 1170px) {
  .NavBar {
```

```
    display: flex;
  }
}
```

Src>components>steps>

Src>components>steps>steps.jsx

```
import"./Steps.scss";

constSteps = () => {
  return (
    <divclassName="Steps Flex JustifyBetweenItemsCenter">
      <divclassName="Label">
        <pclassName="Small">1st Step</p>
        <spanclassName="Active">Select</span>
      </div>
      <divclassName="Line Active"></div>
      <divclassName="Label">
        <pclassName="Small">2nd Step</p>
        <span>Confirm</span>
      </div>
      <divclassName="Line"></div>
      <divclassName="Label">
        <pclassName="Small">3rd Step</p>
        <span>Success</span>
      </div>
    </div>
  );
};

exportdefaultSteps;
```

Src>components>steps>steps.scss

```
@import"/src/Colors.scss";

.Steps {
  grid-area: steps;
  padding: 20px;
  .Label {
    span {
      font-size: 14px;
      color: $light-gray-1;
    }
    span.Active {
```

```
      font-weight: 600;
      color: $dark-gray-3;
    }
  }
  .Line {
    width: 20%;
    height: 1px;
    background: $light-gray-2;
    position: relative;
  }
  .Line.Active::after {
    content: "";
    width: 50%;
    height: 1px;
    background: $light-gray-4;
    position: absolute;
  }
}

@mediaonlyscreen and (min-width: 1170px) {
  .Steps {
    padding-top: 55px;
  }
}
```

Src>components>walletitem

Src>components>walletitem.jsx

```
import"./WalletItem.scss";

constWalletItem = ({ icon, label, name, value, usd }) => {
  return (
    <divclassName="WalletItem My30">
      <divclassName="Flex ItemsCenter">
        <divclassName="IconWrapper Flex JustifyCenterItemsCenter">{icon}</div>
        <divclassName="Ml15">
          <pclassName="ColorDarkGray3">{label}</p>
          <pclassName="FontLato Font600 FontSize10 ColorLightGray3 Mt5">
            {name}
          </p>
        </div>
        <divclassName="Flex Col MlAutoItemsEnd">
          <pclassName="Mb5 FontLatoFontTransformY Font700 FontSize12
ColorDarkGray3">
            {value}
```

```
          </p>
          <pclassName="FontLato Font600 FontSize10 ColorLightGray3">{usd}</p>
        </div>
      </div>
    </div>
  );
};

exportdefaultWalletItem;
```

Src>components>walletitem.scss

```
@import"/src/Colors.scss";

.WalletItem {
  .IconWrapper {
    padding: 10px;
    background: $white;
    border-radius: 50%;
  }
}
```

Src>components>walletlist

Src>components>walletlist.jsx

```
importWalletItemfrom"components/WalletItem/WalletItem";
import"./WalletList.scss";
constlist = [
  {
    icon: (
      <iclassName="Icon ri-bit-coin-fill"style={{ color:"#FA996E" }}></i>
    ),
    label:"BTC",
    name:"Bitcoin",
    value:"5.040000",
    usd:"$35 578.91",
  },
  {
    icon: (
      <i
        className="Icon ri-copper-diamond-fill"
        style={{ color:"#353535" }}
      ></i>
    ),
    label:"ETH",
```

```
    name:"Ethereum",
    value:"160.020000",
    usd:"$36 578.91",
  },
  {
    icon: (
      <iclassName="Icon ri-bit-coin-fill"style={{ color:"#9B9B9B" }}></i>
    ),
    label:"LTC",
    name:"Litecoin",
    value:"135.890000",
    usd:"$7 035.02",
  },
  {
    icon:<iclassName="Icon ri-stack-fill"style={{ color:"#EBBA23" }}></i>,
    label:"BNB",
    name:"Binancecoin",
    value:"5.040000",
    usd:"$35 578.91",
  },
];
constWalletList = () => {
  return (
    <divclassName="WalletList">
      <pclassName="FontSize20 ColorDarkGray3 My20">Wallets</p>
      <divclassName="Flex Row JustifyBetweenItemsCenter Mt40">
        <div>
          <divclassName="Flex Row ItemsCenter">
            <pclassName="ColorDarkGray3">USD</p>
            <iclassName="ri-arrow-drop-down-line ColorDarkGray3"></i>
          </div>
          <pclassName="Font600 FontSize10 ColorLightGray1 Mt5">Market Value</p>
        </div>
        <divclassName="Flex Col ItemsEnd">
          <pclassName="FontTransformYFontLato Font700 FontSize12 ColorDarkGray3
Mb5">
            $ 159 001.212
          </p>
          <pclassName="Font600 FontSize10 ColorLightGray1 Mt5">
            Total Balance
          </p>
        </div>
      </div>
      <hr/>
      <div>
        {list.map((item, i) => (
          <WalletItemkey={i}{...item}/>
        ))}
      </div>
```

```
    </div>
  );
};

exportdefaultWalletList;
```

Src>components>walletlist.scss

```
@import"/src/Colors.scss";

.WalletList {
  grid-area: wallet-list;
  padding: 20px20px020px;
  hr {
    border-top: 1pxsolid$light-gray-2;
    border-bottom: 0;
    margin: 30px0;
  }
  .Icon {
    font-size: 1.3rem;
  }
}
```

Src>App.js

```
importlogofrom'./logo.svg';
import'./App.scss';

functionApp() {
  return (
    <divclassName="App">
      <divclassName="Container"></div>
    </div>
  );
}

exportdefaultApp;
```

Src>App.scss

```
@import"/src/Colors.scss";

.App {
  ::-webkit-scrollbar {
    height: 2px;
```

```
  width: 2px;
}
::-webkit-scrollbar-track {
  border-radius: 10px;
  background: $light-gray-2;
}
::-webkit-scrollbar-thumb {
  border-radius: 10px;
  background: $light-gray-3;
}
background: $light-gray-6;
width: 100vw;
height: auto;
display: flex;
justify-content: center;
align-items: center;
.Container {
  overflow: hidden;
  width: 100%;
  border: 6pxsolid$black;
  border-radius: 60px;
  display: grid;
  grid-template-columns: 100%;
  grid-template-rows: autoautoauto;
  grid-template-areas:
    "wallet-list"
    "steps"
    "convert-details";
}

.Flex {
  display: flex;
}
.Row {
  flex-direction: row;
}
.Col {
  flex-direction: column;
}
.WFull {
  width: 100%;
}
.JustifyBetween {
  justify-content: space-between;
}
.JustifyCenter {
  justify-content: center;
}
.ItemsEnd {
```

```
  align-items: flex-end;
}
.ItemsCenter {
  align-items: center;
}
.FontTransformY {
  transform: scaleY(1.5);
}
.Font400 {
  font-weight: 400;
}
.Font600 {
  font-weight: 600;
}
.Font700 {
  font-weight: 700;
}
.FontSize1-4 {
  font-size: 1.4rem;
}
.FontSize10 {
  font-size: 10px;
}
.FontSize12 {
  font-size: 12px;
}
.FontSize20 {
  font-size: 20px;
}
.BgDarkGray3 {
  background: $dark-gray-3;
}
.BgLightGray2 {
  background: $light-gray-2;
}
.ColorLightGray1 {
  color: $light-gray-1;
}
.ColorLightGray3 {
  color: $light-gray-3;
}
.ColorLightGray4 {
  color: $light-gray-4;
}
.ColorLightGray5 {
  color: $light-gray-5;
}
.ColorDarkGray3 {
  color: $dark-gray-3;
```

```
  }
  .Mx30 {
    margin: 030px;
  }
  .My20 {
    margin: 20px0;
  }
  .My30 {
    margin: 30px0;
  }
  .Mt5 {
    margin-top: 5px;
  }
  .Mt20 {
    margin-top: 20px;
  }
  .Mt30 {
    margin-top: 30px;
  }
  .Mt40 {
    margin-top: 40px;
  }
  .Mr5 {
    margin-right: 5px;
  }
  .Mb5 {
    margin-bottom: 5px;
  }
  .Ml15 {
    margin-left: 15px;
  }
  .Ml30 {
    margin-left: 30px;
  }
  .MlAuto {
    margin-left: auto;
  }
}

@mediaonlyscreen and (min-width: 1170px) {
  .App {
    height: 100vh;

    .Container {
      padding: 20px;
      width: 67%;
      height: 80%;
      display: grid;
      grid-template-columns: 70px25%auto;
```

```
      grid-template-rows: 10%90%;
      grid-template-areas:
        "nav wallet-list steps"
        "nav wallet-list convert-details";
    }
  }
}
```

Src>App.test.js

```
import{ render, screen } from'@testing-library/react';
importAppfrom'./App';

test('renders learn react link', () => {
  render(<App/>);
  constlinkElement = screen.getByText(/learn react/i);
  expect(linkElement).toBeInTheDocument();
});
```

Src>Colors.scss

```
$dark-gray-1: #1d222c;
$dark-gray-2: #5e636c;
$dark-gray-3: #1f242b;
$light-gray-1: #9ba1b0;
$light-gray-2: #e8e9ef;
$light-gray-3: #858d9f;
$light-gray-4: #c1bfd9;
$light-gray-5: #78889f;
$light-gray-6: #f5f3fb;
$light-gray-7: #fcfbfe;
$light-purple: #aba9da;
$light-purple-2: #635fd1;
$white: #fff;
$black: #000;
```

Src>index.js

```
importReactfrom"react";
importReactDOMfrom"react-dom/client";
import"./index.scss";
importAppfrom"./App";
importreportWebVitalsfrom"./reportWebVitals";
```

```
constroot = ReactDOM.createRoot(document.getElementById("root"));
root.render(
  <React.StrictMode>
    <App/>
  </React.StrictMode>
);

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

src>index.scss

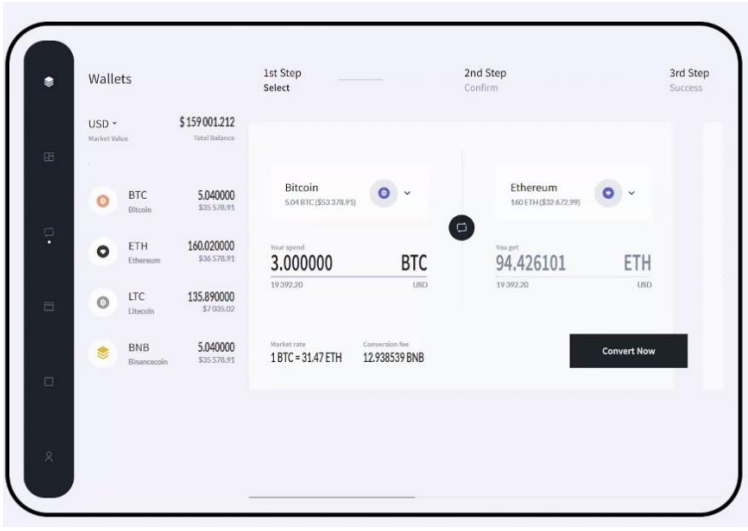
```
@importurl('https://fonts.googleapis.com/css2?family=Lato:wght@400;700&family=Sour
ce+Sans+Pro:wght@400;600;700&display=swap');


*, html, body {
  margin: 0;
  box-sizing: border-box;
  font-family: 'Source Sans Pro', sans-serif;
}

.FontLato {
  font-family: 'Lato', sans-serif;
}
```

App.js >bcrypt.js

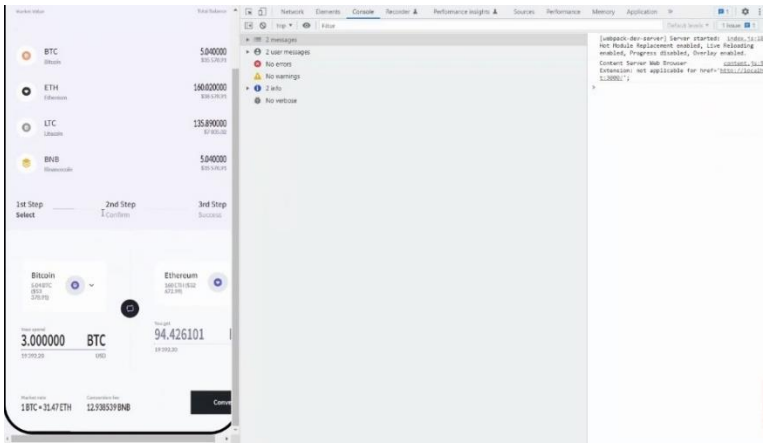
```
import
{
useRef
} from
'react'
        import './App.css'

        import bcrypt from 'bcryptjs'

        // SALT should be created ONE TIME upon sign up
        const salt = bcrypt.genSaltSync(10)
        // example =>  $2a$10$CwTycUXWue0Thq9StjUM0u => to be added always to the
        password hash

        function App() {
        constemailInputRef = useRef()
```

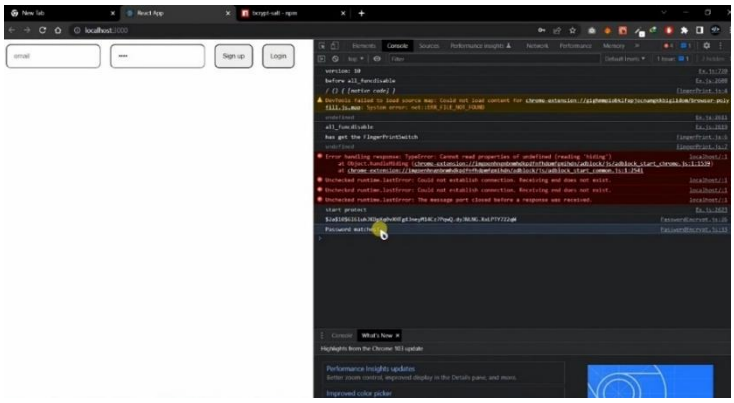
```
constpasswordInputRef = useRef()

  function handleLoginForm() {
const email = emailInputRef.current.value
const password = passwordInputRef.current.value
consthashedPassword = bcrypt.hashSync(password, '$2a$10$CwTycUXWue0Thq9StjUM0u') // hash created previously created upon sign up
fetch('https://api.sampleapis.com/beers/ale', {
     method: 'POST',
     headers: {
       Accept: 'application/json',
       'Content-Type': 'application/json',
     },
     body: JSON.stringify({
       email: email,
       password: hashedPassword,
     }),
   })
  }
  return (
<div className='App'>
<header className='App-header'>
<form>
<input style={{ padding: '15px', borderRadius: '10px', margin: '10px' }} ref={emailInputRef} type='email' placeholder='Email' />
<input style={{ padding: '15px', borderRadius: '10px', margin: '10px' }} ref={passwordInputRef} type='password' placeholder='Password' />
<button
          type='submit'
          style={{ padding: '15px', borderRadius: '10px', margin: '10px' }}
onClick={e => {
e.preventDefault()
handleLoginForm()
          }}>
          Log In
</button>
</form>
<span>Your new SALT: {salt}</span>
<br />
<span>
        Save this Salt, UPON sign up <br /> if you refresh it will generate a new SALT!!!
</span>
</header>
</div>
  )
}
Export default App
```

Output



7.5 cryptowallet





7.6 encryption

CHAPTER-8

CONCLUSION & FUTURE WORK

8.1 CONCLUSION

Cryptocurrency is an impressive technical achievement , but it remains a monetory experiment . even if cryprtocurrencies survive, they may not fully displace flat currencies. They provide an interesting new perspective from which to view economic questions surrounding currency governance , the characteristics of money , the political economy of financial intermediaries and the nature of currency competition.

Although the necessary part , the most of the reputable wallets are not monetized per se. additional services like currency exchange or merchant tools, provide revenue generating capabilities .

Safety of a wallet is a prime concern for a user. For which it is been given a password after successful login of the user and before giving the input transactions details .

8.2 FUTURE WORK

In this project, the future of wallet :crypto which can be a remote control model in upcomimg developments . As it can bring the features like multisig/MPC, customer happieness, passwordless,100% mobile ,non- custodial and multi-assest and multi function. The main focus for the future work is using BIOMETRIC(not password) is the future . and bringing up the all – in – one web 3 wallet like send &receive ,buy&sell , trade & earn ,connect to dapps all in one idea pharse application .this is the ideal super application for crypto developers **Commission calculation.** Some crypto wallets have a field for calculating the commission or they automatically show how much you need to pay depending on the transaction amount. This feature adds convenience to the user and makes the system more transparent.

CHAPTER-9 REFERENCES

[1] ^Jumpupto:[ab]"*European Blockchain Services Infrastructure(EBSI)*".*European Commission*. *Retrieved 24 July 2020.*

[2] ^Yadav, Nagendra Singh &Goar, Vishal &Kuri, Manoj. (2020). CryptoWallet: A Perfect Combination with Blockchain and Security Solution for Banking. International Journal of Psychosocial Rehabilitation. 24. 6056-6066. 10.37200/IJPR/V24I2/PR2021078.

[3] ^*Guler, Sevil (2015).*"*Secure Bitcoin Wallet*"*(PDF). UNIVERSITY OF TARTU FACULTY OF MATHEMATICS AND COMPUTER SCIENCE Institute of Computer Science Computer Science Curriculum: 48 – via core.ac.uk.*

[4] ^Shaik, Cheman. (2020). Securing Cryptocurrency Wallet Seed Phrase Digitally with Blind Key Encryption. International Journal on Cryptography and Information Security. 10. 1-10. 10.5121/ijcis.2020.10401.

[5] ^Jokić, Stevo&Cvetković, Aleksandar Sandro &Adamović, Saša&Ristić, Nenad&Spalević, Petar. (2019). Comparative analysis of cryptocurrency wallets vs traditional wallets. Ekonomika. 65. 10.5937/ekonomika1903065J.

[6] *Deepanshu, Bhatt (2019).*"*Best DApp Browsers to use in 2019*".*Archived**from the original on 2020-10-21.*

[7] Bitcoin Wallets: What You Need to Know About the Hardware".*The Daily Dot*. 2018-11-20. Retrieved 2019-03-10.

[8] *"Bitcoin Startup Predicts Cryptocurrency Market Will Grow By $100 Billion in 2018". Fortune.*

[9] Gutoski, Gus; Stebila, Douglas."Hierarchical deterministic Bitcoin wallets that tolerate key leakage"(PDF). *iacr.org*. International Association for Cryptologic Research. Retrieved 2 November 2018.

[10] Air-gap jumpers on cyber.bgu.ac.il

[11] *Wuille, Pieter (11 February 2012).*"*BIP-0032: Hierarchical Deterministic Wallets*".*GitHub*. *Retrieved 17 October 2021.*

[12] *Buterin, Vitalik(26 November 2013).*"*Deterministic Wallets, Their Advantages and theirUnderstatedFlaws*".*Bitcoin Magazine*. *Retrieved 17 October 2021.*