A Project Report

on

**CRYPTOCURRENCY-ATM**

Submitted in partial fulfilment of the requirements for the award of the degree

of

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**BY**

**MOHAMMED MOHTESHAM ALI 160319733066**

**AKBAR MIRZA SALEEM BAIG 160319733065**

**MIRZA YAHYA BAIG 160319733039**

**Under the guidance**

**of**

**Mrs. Afroze Begum**

**Assistant Professor**

**Department Of CSE, DCET**

****

**Department of Computer Science Engineering**

**Deccan College of Engineering and Technology**

**(Affiliated to Osmania University)**

**Hyderabad**

**2021-2022**

# CERTIFICATE

This is to certify that the project report entitled **Crypto-ATM** being submitted by

**Akbar Mirza Saleem Baig (160319733065), Mohammed Mohtesham Ali (160319733066) and Mirza Yahya Baig (160319733039)** in partial fulfillment for the award of the Degree of Bachelor of Engineering in Computer Science by the Osmania University is a record of Bonafede work carried out by them under my guidance and supervision.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree or Diploma.

Internal Guide Head of CSE Department

**Mrs. Afroze Begum**    **Dr. Syed Raziuddin**

Asst. Professor Professor

Department of CSE Department of CSE

DCET, Hyderabad DCET, Hyderabad

# DECLARATION

This is to certify that the work completed in the mini project titled “**Cryptocurrency ATM**” is a record of work done by us in the Department of Computer Science at Deccan College of Engineering & Technology. As partial fulfillment of the requirements for the degree of Bachelor of Engineering in Computer Science & Engineering.

The results presented in this dissertation have been verified and are found to be satisfactory. The results embodied in this dissertation have not been submitted to any other university for the award of any degree or diploma.

Mirza Yahya Baig 160319733039

Akbar Mirza Saleem Baig 160319733065

Mohammed Mohtesham Ali 160319733066

# ACKNOWLEDGEMENT

I am thankful to Principal **Dr. M.A. MALIK** for providing excellent infrastructure and a nice atmosphere for completing this project successfully.

I am thankful to Head of the department **Dr. SYED RAZIUDDIN** for providing the necessary facilities during the execution of our project work.

This project would not have been a success without my internal guide. So, I would extend my deep sense of gratitude to my internal guide **Mrs. Afroze Begum** and **Mr.Musthafa Baig** for the effort they took in guiding me in all the stages of completion of our project work. We are grateful for their valuable suggestions, advice, guidance and constructive ideas in each and every step, which was indeed a great need towards the successful completion of the project.

I convey my heartfelt thanks to the lab staff for allowing me to use the required equipment whenever needed.

Finally, I would like to take this opportunity to thank the Almighty God through the work.

# ABSTRACT

As the ever-growing demand for [Decentralized Digital Assets] grows more and more, the reliance on Blockchain Technology has become significant in moving & recording these asset transactions happening on the Blockchain Network from Cryptocurrencies or Hardware wallets.

As the world of finance moves closer towards mass adoption for this type of technology it has increased the complexity of operating this hence making it not usable by less tech-savvy users.

Hence why we have introduced a cleaner & more time-saving approach for everyone using these decentralized assets like cryptocurrencies etc. Which will help them in converting their coins into FIAT currency like INR & have it deposited within the user's bank account directly within a matter of few seconds using our online Cryptocurrency-ATM website built on FLASK [A Python Web Framework] & various other frameworks and languages to support the full functioning of this site security.

Facts: Today, there are over 5,000 Bitcoin ATMs across the globe, according to Coin ATM Radar. From New York to Hong Kong, Moscow to London, and Buenos Aires to Sydney, you will find Bitcoin ATMs. Even the small African nation of Djibouti is home to a Bitcoin teller machine. The more bitcoin awareness and adoption grows, the more entrepreneurs are installing Bitcoin ATMs in an attempt to benefit from the” bitcoin gold rush.”

Contents

[CERTIFICATE 2](#_Toc95905308)

[DECLARATION 3](#_Toc95905309)

[ACKNOWLEDGEMENT 4](#_Toc95905310)

[ABSTRACT 5](#_Toc95905311)

[1. INTRODUCITON 8](#_Toc95905312)

[1.1 Introduction 8](#_Toc95905313)

[1.1 Problem Statement 8](#_Toc95905314)

[1.2 Objective 8](#_Toc95905315)

[1.3 Scope 9](#_Toc95905316)

[2. LITERATURE SURVEY 10](#_Toc95905317)

[2.1 PYTHON 10](#_Toc95905318)

[2.2 FLASK 10](#_Toc95905319)

[2.3 JAVASCRIPT 10](#_Toc95905320)

[2.4 HTML5 10](#_Toc95905321)

[2.6 CSS3 11](#_Toc95905322)

[2.7 API 11](#_Toc95905323)

[3. SYSTEM ANALYSIS 12](#_Toc95905324)

[2.1 Software Development Life Cycle (SDLC) 12](#_Toc95905325)

[4. SYSTEM SPECIFICATION 14](#_Toc95905326)

[4.1 Feasibility Study 14](#_Toc95905327)

[4.2 Hardware Specification 14](#_Toc95905328)

[4.3 Software Specification 14](#_Toc95905329)

[5. SYSTEM DESIGN 15](#_Toc95905330)

[5.1 UML Diagrams 15](#_Toc95905331)

[Types of UML Diagrams 15](#_Toc95905332)

[5.2 System Development 20](#_Toc95905333)

[6. IMPLEMENTATION 22](#_Toc95905334)

[6.1 Module Description 22](#_Toc95905335)

[6.2 Code 22](#_Toc95905336)

[7. TEST STRATEGIES AND RESULTS 64](#_Toc95905337)

[7.2 VALIDATION (TEST CASES) 67](#_Toc95905338)

[7.3 Integration Testing 69](#_Toc95905339)

[7.4 Screenshots 70](#_Toc95905340)

[8. CONCLUSION & FUTURE ENHANCEMENT 77](#_Toc95905341)

[8.1 Conclusion 77](#_Toc95905342)

[8.2 Future Enhancement 77](#_Toc95905343)

[REFERENCES 78](#_Toc95905344)

# INTRODUCITON

## 1.1 Introduction

This " Cryptocurrency-ATM" Mini Project is a flask-based web application running on the Python micro framework. This website was built to ease fluidity of buying & exchanging digital assets on the TRON Network which has the lowest transactions fees of 1USDT i.e., Approx. ₹75, the site has been constructed in such a manner that is usable by everyone with a matter of a few clicks of the mouse! This site manages to save a substantial amount of effort, time and money for the prospective asset investor, and above all that, takes away the biggest headache of dealing with KYC registration waiting periods, exchanges complex buying platforms and lastly, price fluctuations of these digital assets.

Our website also displays information price & market capitalization changes to the Top 10 Stable Coins in the world.

## 1.1 Problem Statement

When buying Cryptocurrencies from exchanges like WazirX or Binance, the users are required to wait for at least 2 days for a KYC registration, to use their facility which may even be rejected or not be supported by many other platforms due to country restrictions and so forth. However, our website doesn’t include KYC rules since it is given out to exclusive members decided by us. So, the users are free to buy or exchange their cryptocurrencies without the need of KYC.

In exchanges the rate of the cryptocurrency coins may fluctuate in seconds.

However, our website provides a constant rate for the stable coins being fixed at ₹75.

## 1.2 Objective

The main objective of this website is to improve & equip our youth with a more leading-edge platform for their financial future because as we move forward towards the future, there is high chance that adaptions to digital finances like cryptocurrencies will move faster as the world has only seen a global crypto ownership rate at an average of 3.9%, with an over 300 million crypto users worldwide. You can also see many companies like Tesla, PayPal, Wikipedia, MicroStrategy etc. Have already adopted to cryptocurrencies in the present and are adding more with each dip. Ex: Biggest Legal tender to Bitcoin is country El Salvador.

## 1.3 Scope

In the ever-expanding world of finance, many physical cryptocurrency ATMs have been installed everywhere from Canada to the UAE. Since these ATMs aren't easily accessible or installed in all locations, we decided to develop a web version of these ATMs that is more accessible & easier to use.

# 2. LITERATURE SURVEY

## 2.1 PYTHON

Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with its use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming.

## 2.2 FLASK

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

## 2.3 JAVASCRIPT

JavaScript abbreviated JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. Over 97% of websites use JavaScript on the client side for web page behavior, often incorporating third-party libraries.

## 2.4 HTML5

HTML5 is a markup language used for structuring and presenting content on the World Wide Web. It is the fifth and last major HTML version. HTML5 includes detailed processing models to encourage more interoperable implementations; it extends, improves, and rationalizes the markup available for documents and introduces markup and application programming interfaces (APIs) for complex web applications.

## 2.6 CSS3

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.

Bootstrap is a CSS framework directed at responsive, mobile-first front-end web development. It contains CSS and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

## 2.7 API

An application programming interface (API) is a connection between computers or between computer programs. It is a type of software interface, offering a service to other pieces of software. A document or standard that describes how to build or use such a connection or interface is called an API specification. A computer system that meets this standard is said to implement or expose an API. The term API may refer either to the specification or to the implementation.

Our website usesAPI’s of Paytm, Binance and WazirX.

# 3. SYSTEM ANALYSIS

System analysis is the analysis of the problem that the organization will try to solve with an information system. It consists of defining the problem, identifying its causes, specifying the solution, and identifying the information requirements that must be met by a system solution. System analysis is a very important stage because other stages later will depend on the analysis stage. Analyzing the software includes many things. First of all, we study the system to understand the concept of the system and also, to specify the requirement of the project. After we specify the requirement and collect the data required; we start prepare the Project Initiation Report (PIR). At this report we consider what are the methodology, hardware, software and operating system will use for the project. Later on, we prepare project requirement Specification (PRS).

## 2.1 Software Development Life Cycle (SDLC)

There are various software development approaches defined and designed which are used/employed during development process of software, these approaches are also referred as “Software Development Process Models” (e.g. Waterfall model, Incremental model, V-model, iterative model etc). Each process model follows a particular life cycle in order to ensure success in process of software development. Software life cycle models describe phases of the software cycle and the order in which those phases are executed. Each phase produces deliverables required by the next phase in the life cycle. Requirements are translated into design. Code is produced according to the design which is called development phase. After coding and development the testing verifies the deliverable of the implementation phase against requirements.

**Life Cycle Phases**

There are following six phases in every Software development life cycle model:

* Requirement gathering and analysis
* Design
* Implementation or coding
* Testing
* Deployment
* Maintenance

Requirement gathering and analysis: Business requirements are gathered in this phase. This phase is the main focus of the project managers and stake holders. Meetings with managers, stake holders and users are held in order to determine the requirements like; who is going to use the system? How will they use the system? What data should be input into the system? What data should be output by the system? These are general questions that get answered during a requirements gathering phase.

After requirement gathering these requirements are analyzed for their validity and the possibility of incorporating the requirements in the system to be development is also studied. Finally, a Requirement Specification document is created which serves the purpose of guideline for the next phase of the model.

Design In this phase the system and software design is prepared from the requirement specifications which were studied in the first phase. System Design helps in specifying hardware and system requirements and also helps in defining overall system architecture. The system design specifications serve as input for the next phase of the model.

Testing After the code is developed it is tested against the requirements to make sure that the product is actually solving the needs addressed and gathered during the requirements phase. During this phase unit testing, integration testing, system testing, acceptance testing are done.

Deployment After successful testing the product is delivered / deployed to the customer for their use.

Maintenance Once when the customer starts using the developed system then the actual problems comes up and needs to be solved from time to time. This process where the care is taken for the developed product is known as maintenance.

# 4. SYSTEM SPECIFICATION

## 4.1 Feasibility Study

Feasibility study is about the viability of a system. The proposed system has to be examined for its technical, economical and operational feasibility. It is evaluated from developer and user’s point of view.

* **Economic Feasibility**

Economic analysis is most frequently used for evaluation of the effectiveness of the system. This part of feasibility study gives the top management the economic justification for the new system.

**Operational Feasibility**

Operational Feasibility is a must, because it ensures that the implementation of the software as an application should be high.

* **Technical Feasibility**

It is the process of assessing the development application ability to construct a proposed system.

## 4.2 Hardware Specification

PROCESSOR: Intel Pentium or more

RAM: 2GB RAM or Later

VIDEO - 1024x768, 24-bit colors

KEYBOARD - Standard 104 Keys (QWERTY)

Hard Disk: PC with 20GB

## 4.3 Software Specification

Operating System Server: Windows 8 or later

Server Tools: MS Virtual Studio code

# 5. SYSTEM DESIGN

## 5.1 UML Diagrams

The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system.

UML offers a way to visualize a system's architectural blueprints in a diagram, including elements such as:

1. any activities (jobs)
2. individual components of the system;
3. and how they can interact with other software components;
4. how the system will run;
5. how entities interact with others (components and interfaces);
6. external user interface.

## Types of UML Diagrams

1. Use Case Diagram

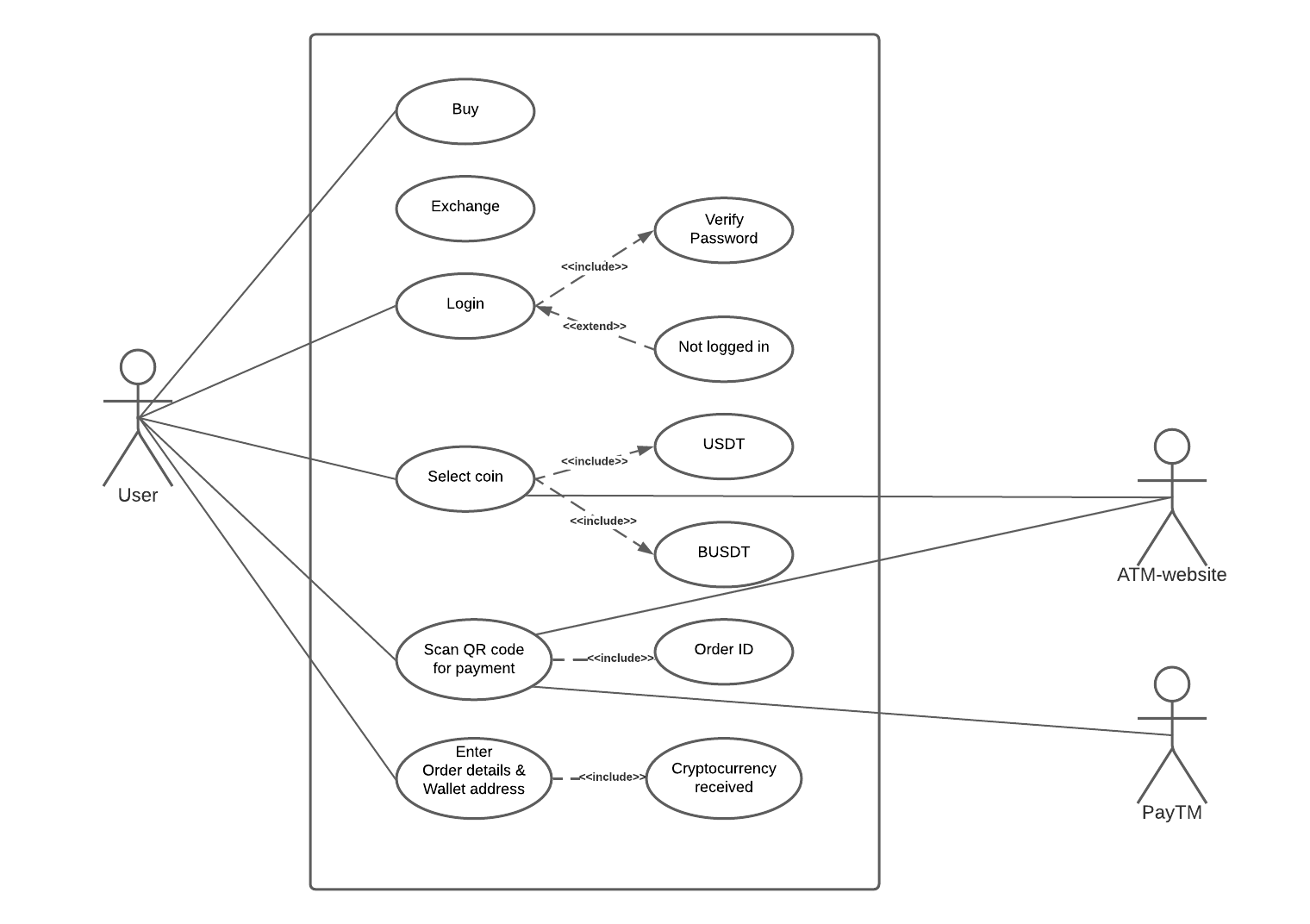
A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.

There exist 3 types of relationships

1) Association

2) Dependency

3) Generalization



Use Case Diagram for Cryptocurrency-ATM

**2. Activity Diagram**

An activity diagram is a special case of state diagram. An activity diagram is like a flow machine showing the flow a control from one activity to another. An Activity diagram is used to model dynamic aspects of the system

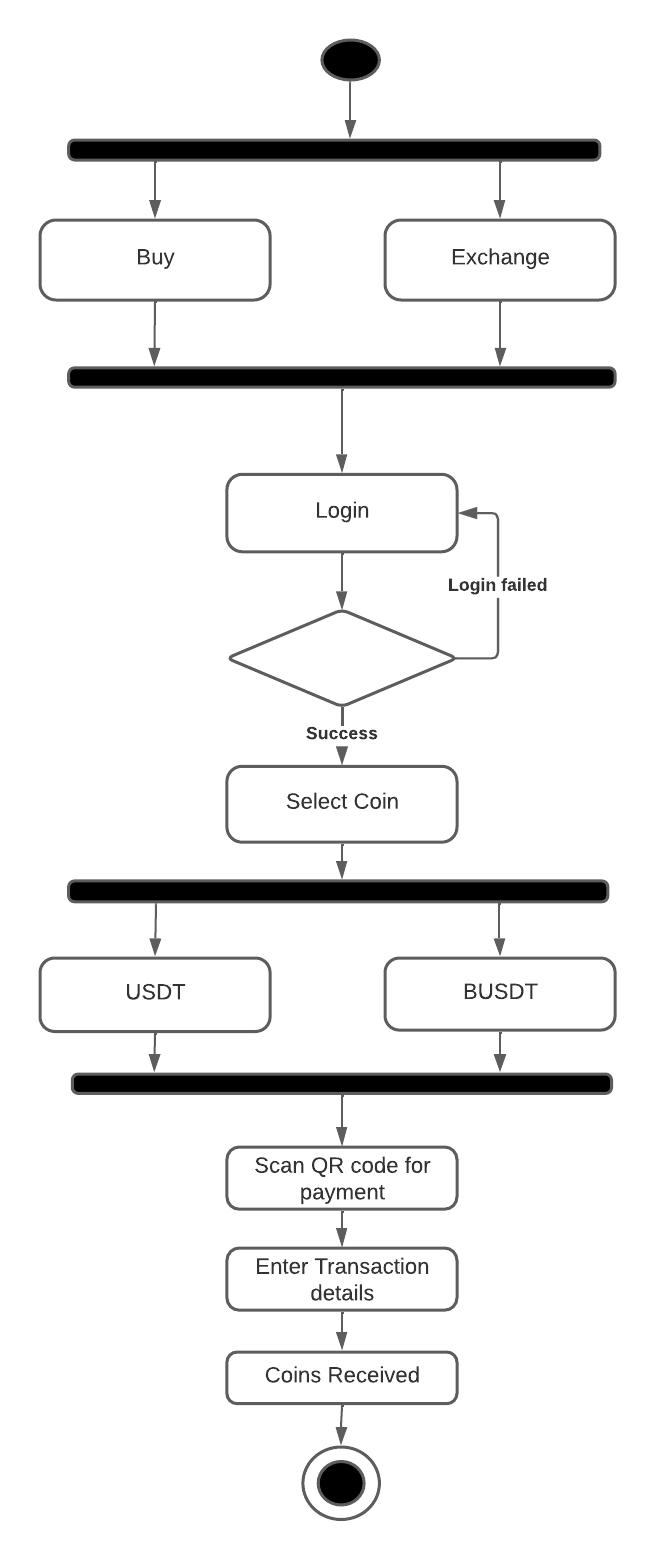
.

Activity Diagram Contains:

1) Activity states and action states

2) Transition

3) Object

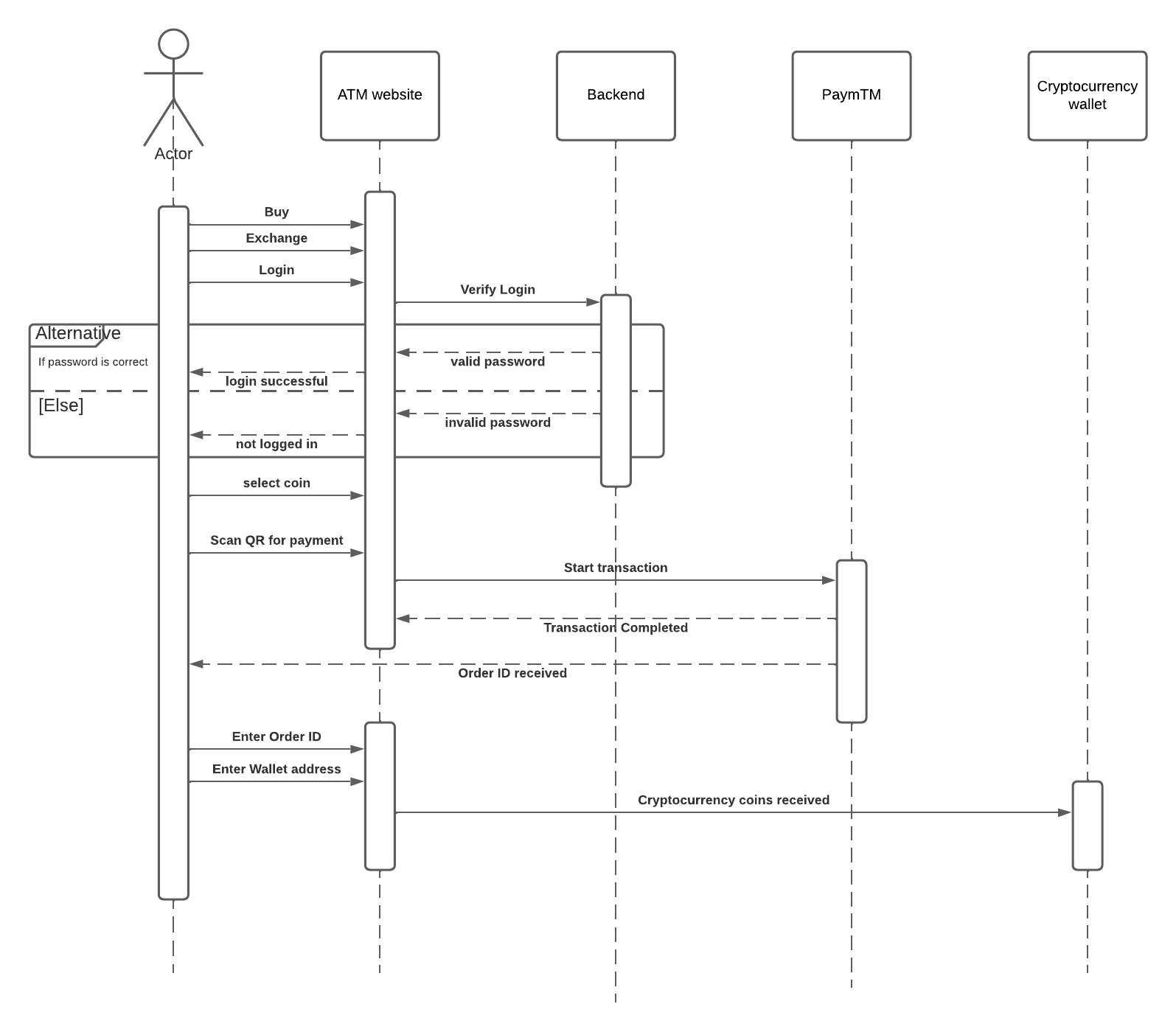


Activity Diagram for Cryptocurrency-ATM

**3. Sequence Diagram**

Sequence Diagram is a graphical view of a scenario that shows object interaction in a time-based sequence, what happens first, what happens next.

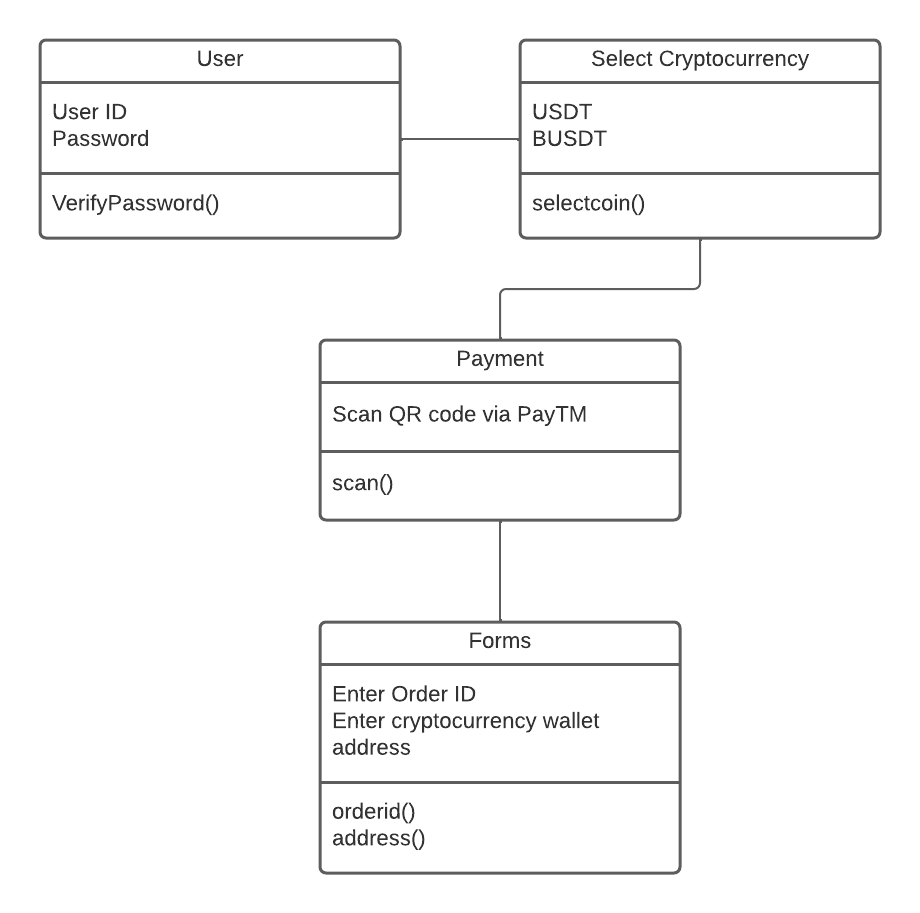
Sequence Diagrams establish the roles of objects and help provide essential information to determine class responsibilities and interfaces.

A sequence diagram has two dimensions: vertical placement represents time and horizontal placement represents different object

Sequence Diagram for Cryptocurrency-ATM

**4. Class Diagram**

A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.



Class Diagram for Cryptocurrency ATM

## 5.2 System Development

Software Engineers have been trying various tools, methods and procedures to control the process of software development in order to build high quality software with high productivity. This method provides “how it is” for building the software while the tools provide automated or semi automated support for the methods. They are used in all stages of software development process, namely, planning, analysis, design, development and maintenance. The software development procedure integrates the methods and tools together and enables rational and timely development of the software system.

They provide the guidelines as how to apply these methods and tools, how to produce the deliverable at each stage, what controls to apply, and what milestones to use to assess the performance of the program. There exist a number of software development paradigms each using a different set of methods and tools. The selection of a particular paradigm depends on the nature of application of the programming language used for the controls and the deliverables required. The development of such successful systems depends not only on the use of appropriate methods and techniques but also the developers’ commitment to the objective of the system. A successful system must: -

1. Satisfy the user requirements

2. Be easy to understand by user and operator

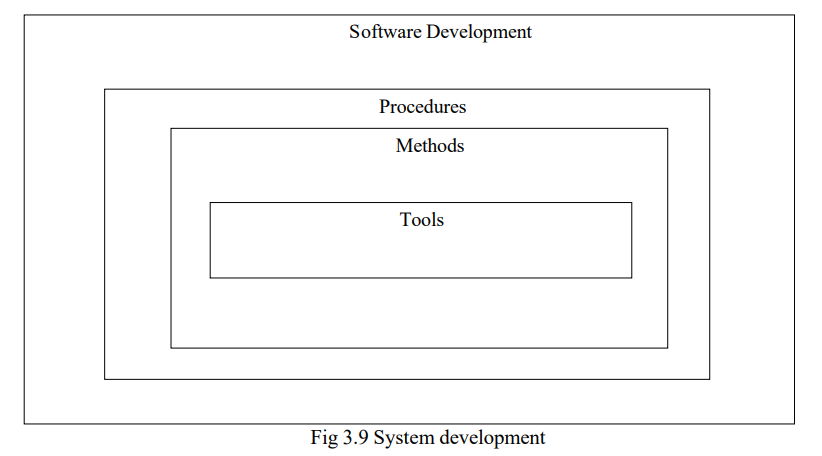
3. Be easy to operate

4. Have a good user interface

5. Be easy to modify

6. Be expandable

7. Have adequate security control against the misuse of data

****

# 6. IMPLEMENTATION

## 6.1 Module Description

The Modules of the project are various Development units which are built separately and integrated together at the end to form a complete Application. There are mainly 4 modules in this system

**LOGIN MODULE**

In this, the user has to enter their user ID and password to login to our website.

**PRODUCT MODULE**

In this module, the user has to select one of the given cryptocurrencies for purchasing.

**TRANSACTION MODULE**

In this module, the user has to scan the QR code which embedded in our website to purchase the cryptocurrency. After the transaction is successful, the user will get the Order ID.

**FORMS MODULE**

In this, the user is directed to enter the order ID which is given when the transaction is successful. Also, the user also required to provide their cryptocurrency wallet address for storing the cryptocurrency.

**LOGOUT MODULE**

After completion of a successful transaction the user logs out of the session by using the logout button.

## 6.2 Code

**Python files:**

**1. Main App.py:**

from flask import Flask

from app\_blueprint import app\_blueprint

app = Flask(\_\_name\_\_)

app.secret\_key = 'somesecretkeythatonlyishouldknow'

app.register\_blueprint(app\_blueprint)

if \_\_name\_\_ == "\_\_main\_\_":

app.run()

**2. Inherits blueprint:**

########################### IMPORT ###################################

import requests

import json

from paytmchecksum import generateSignature

from flask import Blueprint, render\_template, request, flash

from keys import merchant\_id,merchant\_key

app\_blueprint = Blueprint('app\_blueprint',\_\_name\_\_)

from binance import Client

import json

#################### IMPORTS END ################################

################### HTML ROUTES ################################

@app\_blueprint.route('/coins')

def coins():

return render\_template("coins.html")

@app\_blueprint.route('/')

@app\_blueprint.route('/home')

def home():

return render\_template("home.html")

@app\_blueprint.route('/form', methods = ['POST', 'GET'])

def form():

return render\_template("form.html")

@app\_blueprint.route('/buy')

def buy():

return render\_template("buy.html")

@app\_blueprint.route('/success')

def success():

return render\_template("success.html")

@app\_blueprint.route('/failure')

def failure():

return render\_template("failure.html")

########################################### LOGIN ROUTES ##################################################

from flask import (

Flask,

redirect,

render\_template,

request,

session,

url\_for,

flash

)

from flask\_socketio import SocketIO, emit

from flask\_login import current\_user, logout\_user

from datetime import timedelta

#Step – 1(import necessary library)

from flask import (Flask, render\_template, request, redirect, session,flash)

#Step – 2 (configuring your application)

app = Flask(\_\_name\_\_)

app.secret\_key = 'ItShouldBeAnythingButSecret'

socketio = SocketIO(app)

app.config['PERMANENT\_SESSION\_LIFETIME'] = timedelta(minutes = 12)

#step – 3 (creating a dictionary to store information about users)

user = {"username": "username", "password": "password"}

#Step – 4 (creating route for login)

@app\_blueprint.route('/login', methods = ['POST', 'GET'])

def login():

if request.method != 'POST':

return render\_template("index.html")

username = request.form.get('username')

password = request.form.get('password')

if username == user['username'] and password == user['password']:

session.permanent = True

session['user'] = username

return redirect('/deposit')

flash("Login Failed", "info")

return redirect('/login')

#Step -5(creating route for deposit and logout)

@app\_blueprint.route('/deposit')

def deposit():

if('user' in session and session['user'] == user['username']):

return render\_template("deposit.html")

return '<h1>You are not logged in.</h1>'

#Step -6(creating route for logging out)

@app\_blueprint.route('/logout')

def logout():

session.pop('username',None)

flash("You were logged out.", "info")

return redirect(url\_for("app\_blueprint.login"))

@socketio.on('disconnect')

def disconnect\_user():

logout\_user()

session.pop('ItShouldBeAnythingButSecret', None)

#################### TEST OR FUNCTION ROUTES ######################

from keys import api\_key, api\_secret

client = Client(api\_key, api\_secret)

######################### Binance- API CALLS ###########################

def paytm\_check\_money(order\_id,amt):

print(order\_id)

paytmParams = {

'body': {

"mid": merchant\_id,

"orderId": str(order\_id)

}

}

checksum = generateSignature(json.dumps(paytmParams["body"]), merchant\_key)

paytmParams["head"] = {

"signature" : checksum

}

post\_data = json.dumps(paytmParams)

url = "https://securegw.paytm.in/v3/order/status"

response = requests.post(url, data = post\_data, headers = {"Content-type": "application/json"}).json()

amt\_sent = response['body']['txnAmount']

return float(amt\_sent)==int(amt)

@app\_blueprint.route('/buycrypto',methods=["GET","POST"])

def paytmchecker():

order\_id = request.form['tid']

amt = request.form['Wallet']

try:

if paytm\_check\_money(order\_id,amt):

addr = request.form['Email']

client.withdraw(

coin='USDT',

# address=email,

address=f'<{addr}>',

amount=amt)

return redirect(url\_for("app\_blueprint.sucess"))

else:

return redirect(url\_for("app\_blueprint.failure"))

except Exception:

return redirect(url\_for("app\_blueprint.failure"))

**3. Keys.py**

###################### Binacne API KEYS ##############################

api\_key = "g257kS41SLmFPNWi1tUaFbPSTjiKYj3Rb1D4kLuHFsgAmsn5FoZ628E6yXvtYaV7"

api\_secret = "935PKy1hPks2MTA5SuItUeB9QT4A2ZFItGySJkyc2wdIkVdos2S0jYNawbH0IjIU"

############################ PAYTM Keys ###################################

merchant\_id = "MVFOAP42279449625656"

merchant\_key = "fh8BvsjqHnZfxVo9"

**HTML Pages in Sequence:**

**1. Home. html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta content="width=device-width, initial-scale=1.0" name="viewport">

<title>Crypto-ATM - Home Page</title>

<meta content="" name="Cryptocurreny ATM By Deccan">

<meta content="" name="Crpto atm">

<!-- Favicons -->

<link href="staic/img/favicon.png" rel="icon">

<link href="staic/img/apple-touch-icon.png" rel="apple-touch-icon">

{% include 'css.html' %}

</head>

<body>

<!-- ======= Header ======= -->

<header>

{% include 'navbar.html' %}

</header>

<!-- End #header -->

<!-- ======= Hero Section ======= -->

<section id="hero">

<div class="hero-container">

<h1>Welcome</h1>

<h2>Click The Button Below To Get Started</h2>

<a href="login" class="btn-get-started scrollto">Cryptocurrency-Banking</a>

</div>

</section><!-- #hero -->

<main id="main">

<!-- ======= About Us Section ======= -->

<section id="about" class="about">

<div class="container">

<div class="section-title">

<h2><b>The Rise And Development Of Crypto-ATMs In The Present Future</b></h2>

</div>

<div class="row">

<div class="col-lg-6 order-1 order-lg-2">

<img src="static/img/about-img.jpg" class="img-fluid" alt="">

</div>

<div class="col-lg-6 pt-4 pt-lg-0 order-2 order-lg-1">

<p>

As the ever-growing demand for [Decentralized Digital Assets] grows more and more, the reliance on Blockchain Technology has become significant in moving & recording these asset transactions happening on the Blockchain Network from Cryptocurrencies or Hardware wallets.

<br><br>

As the world of finance moves closer towards mass adoption for this type of technology it has increased the complexity of operating this hence making it not usable by less tech-savvy users.

<br><br>

Hence why we have introduced a cleaner & more time-saving approach for everyone using these decentralized assets like cryptocurrencies etc. Which will help them in converting their coins into FIAT currency like INR & have it deposited within the user's bank account directly within a matter of few seconds using our online Cryptocurrency-ATM website built on FLASK [A Python Web Framework] & various other frameworks and languages to support the full functioning of this site security.

<br><br>

Facts: Today, there are over 5,000 Bitcoin ATMs across the globe, according to Coin ATM Radar. From New York to Hong Kong, Moscow to London, and Buenos Aires to Sydney, you will find Bitcoin ATMs. Even the small African nation of Djibouti is home to a Bitcoin teller machine. The more bitcoin awareness and adoption grows, the more entrepreneurs are installing Bitcoin ATMs in an attempt to benefit from the” bitcoin gold rush.”

</p>

</div>

</div>

</div>

</section><!-- End About Us Section -->

<!-- ======= Services Section ======= -->

<section id="services" class="services section-bg">

<div class="container">

<div class="section-title">

<h2><b>Services of Crypto-ATM</b></h2>

<p>Crypto ATMs use an internet-based cryptocurrency portal to manage transactions.The users deposit their cash and the machine converts the cash into crypto and sends the funds to your Cypto wallet wherein you send and receive all your crypto funds.</p>

</div>

<div class="row">

<div class="col-lg-4 col-md-6 icon-box">

<div class="icon"><i class="bi bi-emoji-laughing-fill"></i></div>

<h4 class="title"><a href="">Easy to use</a></h4>

<p class="description">All you have to do is follow a simple and clear step by step process when using our ATMs. You do not need to set up an account, online wallet, or deal with lengthy public keys.

Your receipt contains a QR code, with the public and private key, which can be scanned and immediately recognized by standard wallet apps, if you have one.</p>

</div>

<div class="col-lg-4 col-md-6 icon-box">

<div class="icon"><i class="bi bi-currency-dollar"></i></div>

<h4 class="title"><a href="">Network fees</a></h4>

<p class="description">The transaction fees on a Crypto-ATM are generally between 0.1 and 0.2 USDT which is equal to 8-14 INR per transaction.</p>

</div>

<div class="col-lg-4 col-md-6 icon-box">

<div class="icon"><i class="bi bi-cash-coin"></i></div>

<h4 class="title"><a href="">Lowest Deposit & Withdrawal Amount</a></h4>

<p class="description">Generally the minimum Withdrawal amount is $20 and Deposit amount is $2</p>

</div>

<div class="col-lg-4 col-md-6 icon-box">

<div class="icon"><i class="bi bi-shield"></i></div>

<h4 class="title"><a href="">Security</a></h4>

<p class="description">Currently, the cryptocurrency ATM infrastructure is running on Heroku, a platform that is operated and managed by developers.

To send, receive & record transactions, the cryptocurrency nodes are connected to the "TRON" Network which has one of the lowest transaction fees.

The use of Zero Knowledge ATMs will provide even greater anonymity for security while preserving greater decentralization in the future.</p>

</div>

<div class="col-lg-4 col-md-6 icon-box">

<div class="icon"><i class="bi bi-lightning-charge-fill"></i></div>

<h4 class="title"><a href="">Immediacy</a></h4>

<p class="description">Cryptocurrency ATMs can complete transactions in few minutes.

For buy order, as soon as money has been inserted into our ATM, a purchase is executed for the total payment amount.

If you’re looking to make a sell order, cryptocurrency ATMs make it so that you no longer need to wait several days to receive cash.</p>

</div>

</div>

</div>

</section><!-- End Services Section -->

<!-- ======= Call To Action Section ======= -->

<!-- ======= Frequently Asked Questions Section ======= -->

<section id="faq" class="faq section-bg">

<div class="container">

<div class="section-title">

<h2><b>Frequently Asked Questions</b></h2>

</div>

<ul class="faq-list">

<li>

<div data-bs-toggle="collapse" class="collapsed question" href="#faq1">Is Crypto ATM Reliable? <i class="bi bi-chevron-down icon-show"></i><i class="bi bi-chevron-up icon-close"></i></div>

<div id="faq1" class="collapse" data-bs-parent=".faq-list">

<p>

Crypto ATMs are one of the safest ways to buy, send or sell coins. First, instant transactions protect you from Crypto coins volatility. Secondly, passwords and 2-factor authentication protect your account from others.

</p>

</div>

</li>

<li>

<div data-bs-toggle="collapse" href="#faq2" class="collapsed question">How long does a Crypto ATM transaction take? <i class="bi bi-chevron-down icon-show"></i><i class="bi bi-chevron-up icon-close"></i></div>

<div id="faq2" class="collapse" data-bs-parent=".faq-list">

<p>

Crypto ATM purchases are instant. Account verification takes only a few minutes, and purchasing coins is even faster. However, coin sales must be confirmed on the blockchain. This confirmation involves a third party.

</p>

</div>

</li>

<li>

<div data-bs-toggle="collapse" href="#faq3" class="collapsed question">Is there any fee while making a transaction in Crypto ATM? <i class="bi bi-chevron-down icon-show"></i><i class="bi bi-chevron-up icon-close"></i></div>

<div id="faq3" class="collapse" data-bs-parent=".faq-list">

<p>

As with any bitcoin exchange, Crypto ATMs also charge a fee per transaction. Depending on the Crypto ATM or kiosk provider, these fees can be as low as 6.99% or as high as 20%.

</p>

</div>

</li>

<li>

<div data-bs-toggle="collapse" href="#faq4" class="collapsed question">Do Crypto-ATM accept any type of debit/credit cards? <i class="bi bi-chevron-down icon-show"></i><i class="bi bi-chevron-up icon-close"></i></div>

<div id="faq4" class="collapse" data-bs-parent=".faq-list">

<p>

Crypto-ATM kiosks look like traditional ATMs, but do not connect to a bank account and instead connect the user directly to a Crypto wallet or exchange.

</p>

</div>

</li>

</ul>

</div>

</section><!-- End Frequently Asked Questions Section -->

<!-- ======= Our Team Section ======= -->

<section id="team" class="team">

<div class="container">

<div class="section-title">

<h2><b>Team</b></h2>

<p>DCET students have developed an innovative project called Crypto-ATM, which allows you to conduct easy deposits and purchases without the hassle of an exchange.</p>

</div>

<div class="row gy-4">

<div class="col-lg-4 col-md-6">

<div class="member">

<img src="static/img/team/team-1.jpg" alt="">

<h4>Mohammed Mohtesham Ali</h4>

<span>Project Coordinator</span>

<span>16031973-3066</span>

<p>Exploring the Bounds Of Technology <br> As A CS Student.</p>

<div class="social">

<a href=""><i class="bi bi-twitter"></i></a>

<a href=""><i class="bi bi-facebook"></i></a>

<a href=""><i class="bi bi-instagram"></i></a>

<a href="https://www.linkedin.com/in/-mohtesham-ali/"><i class="bi bi-linkedin"></i></a>

</div>

</div>

</div>

<div class="col-lg-4 col-md-6">

<div class="member">

<img src="static/img/team/team-2.jpg"alt="">

<h4>Akbar Mirza</h4>

<span>Assistant Coordinator</span>

<span>16031973-3065</span>

<p>

Aspiring Software Developer.

<br>

</p>

<div class="social">

<a href=""><i class="bi bi-twitter"></i></a>

<a href=""><i class="bi bi-facebook"></i></a>

<a href=""><i class="bi bi-instagram"></i></a>

<a href="https://www.linkedin.com/in/akbar-mirza-b17386193/"><i class="bi bi-linkedin"></i></a>

</div>

</div>

</div>

<div class="col-lg-4 col-md-6">

<div class="member">

<img src="static/img/team/team-3.jpg" alt="">

<h4>Mirza Yahya Baig</h4>

<span>Documentation & Quality Control</span>

<span>160319733039</span>

<p>

Programmer's are tools for converting caffeine into code

</p>

<div class="social">

<a href=""><i class="bi bi-twitter"></i></a>

<a href=""><i class="bi bi-facebook"></i></a>

<a href=""><i class="bi bi-instagram"></i></a>

<a href=""><i class="bi bi-linkedin"></i></a>

</div>

</div>

</div>

</div>

</div>

</section><!-- End Our Team Section -->

<!-- ======= Contact Us Section ======= -->

<section id="contact" class="contact section-bg">

<div class="container">

<div class="section-title">

<h2><b>Contact Us</b></h2>

<p>We realize that the Crypto-ATM is a new concept for you all, and you might have questions.Any questions regarding the project can be directed to Project Coordinator Mohtesham Ali.</p>

</div>

<div class="row">

<div class="col-lg-4 col-md-6">

<div class="contact-about">

<h3><b>Crypto-ATM</b></h3>

<img src="static\img\download.png" alt="" class="customer-care">;

</div>

</div>

<div class="col-lg-3 col-md-6">

<div class="info">

<div>

<i class="bi bi-geo-alt"></i>

<p>Deccan College of Engineering & Technology<br>Hyderabad,Telangana 500001</p>

</div>

<div>

<i class="bi bi-envelope"></i>

<p>mohteshamali38@yahoo.com</p>

</div>

<div>

<i class="bi bi-phone"></i>

<p>+917093752144</p>

</div>

</div>

</div>

<div class="col-lg-5 col-md-12">

<form action="forms/contact.php" method="post" role="form" class="php-email-form">

<div class="form-group">

<input type="text" name="name" class="form-control" id="name" placeholder="Your Name" required>

</div>

<div class="form-group mt-3">

<input type="email" class="form-control" name="email" id="email" placeholder="Your Email" required>

</div>

<div class="form-group mt-3">

<input type="text" class="form-control" name="subject" id="subject" placeholder="Subject" required>

</div>

<div class="form-group mt-3">

<textarea class="form-control" name="message" rows="5" placeholder="Message" required></textarea>

</div>

<div class="my-3">

<div class="loading">Loading</div>

<div class="error-message"></div>

<div class="sent-message">Your message has been sent. Thank you!</div>

</div>

<div class="text-center"><button type="submit">Send Message</button></div>

</form>

</div>

</div>

</div>

</section><!-- End Contact Us Section -->

<!-- ======= Map Section ======= -->

<section class="map">

<div style="width: 100%"><iframe width="100%" height="600" frameborder="0" scrolling="no" marginheight="0" marginwidth="0" src="https://maps.google.com/maps?width=100%25&amp;height=600&amp;hl=en&amp;q=deccan%20college%20of%20engineering+(My%20Business%20Name)&amp;t=&amp;z=15&amp;ie=UTF8&amp;iwloc=B&amp;output=embed"><a href="https://www.gps.ie/sport-gps/">swimming watch</a></iframe></div>

</section><!-- End Map Section -->

</main><!-- End #main -->

<!-- ======= Footer ======= -->

{% include 'footer.html' %}

<!-- Vendor JS Files -->

<script src="static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>

<script src="static/vendor/glightbox/js/glightbox.min.js"></script>

<script src="static/vendor/isotope-layout/isotope.pkgd.min.js"></script>

<script src="static/vendor/swiper/swiper-bundle.min.js"></script>

<script src="static/vendor/php-email-form/validate.js"></script>

<!-- Template Main JS File -->

<script src="static/js/main.js"></script>

</body>

</html>

**3. Footer.html**

<!-- ======= Footer ======= -->

<footer id="footer">

<div class="container">

<div class="copyright">

&copy; Copyright <strong><span>Crypto-ATM</span></strong>. All Rights Reserved 2021-2022

</div>

<div class="credits">

<!-- All the links in the footer should remain intact. -->

<!-- You can delete the links only if you purchased the pro version. -->

<!-- Licensing information: https://bootstrapmade.com/license/ -->

<!-- Purchase the pro version with working PHP/AJAX contact form: https://bootstrapmade.com/free-one-page-bootstrap-template-amoeba/ -->

Designed by <a href="https://bootstrapmade.com/">Crypto-Team Using @Boot-Strap</a>

</div>

</div>

</footer><!-- End #footer -->

<a href="#" class="back-to-top d-flex align-items-center justify-content-center"><i class="bi bi-arrow-up-short"></i></a>

**4. Navbar.html**

<!-- ======= Page Alignment ======= -->

<header id="header" class="fixed-top d-flex align-items-center">

<div class="container d-flex align-items-center">

<div class="logo me-auto">

<h1><a href="home">Crypto-ATM</a></h1>

</div>

</div>

<!-- End #Page Alignment -->

<!-- ======= NavBar ======= -->

<nav id="navbar" class="navbar navbar-expand-lg navbar-dark bg-25564f">

<a class="navbar-brand" href="home">Home</a>

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavAltMarkup" aria-controls="navbarNavAltMarkup" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarNavAltMarkup">

<div class="navbar-nav">

<a class="nav-item nav-link" href="coins">Stable-Coins</a>

<a class="nav-item nav-link" href="#services">Services & Fees</a>

<a class="nav-item nav-link" href="#contact">Support</a>

<a class="nav-item nav-link" href="#"></a>

<ul class="nav navbar-nav navbar-right">

{% if session['logged\_in'] == True %}

<li><a href="/logout">Logout <span class="sr-only">(current)</span></a></li>

{% else %}

<li><a href="/login">Login <span class="sr-only">(current)</span></a></li>

{% endif %}

</ul>

<a class="nav-item nav-link disabled" href="#" tabindex="-1" aria-disabled="true" style = "color:rgba(4, 243, 16, 0.685);">Status:<span class="sr-only" style = "color:red;">Offline</span></a>

</div>

</div>

<i class="bi bi-list mobile-nav-toggle"></i>

</nav>

<!-- End NavBar -->

**5. Index.html**

<!DOCTYPE html>

<html lang="en">

<head>

<title>ATM-Login</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<!--===============================================================================================-->

<link rel="icon" type="image/png" href="static/images/icons/favicon.ico"/>

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css" href="static/vendor/bootstrap/css/bootstraplogin.min.css">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css" href="static/fonts/font-awesome-4.7.0/css/font-awesome.min.css">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css" href="static/fonts/Linearicons-Free-v1.0.0/icon-font.min.css">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css" href="static/vendor/animate/animate.css">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css" href="static/vendor/css-hamburgers/hamburgers.min.css">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css" href="static/vendor/animsition/css/animsition.min.css">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css" href="static/vendor/select2/select2.min.css">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css" href="static/vendor/daterangepicker/daterangepicker.css">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css" href="static/css/util.css">

<link rel="stylesheet" type="text/css" href="static/css/main.css">

<!--===============================================================================================-->

</head>

<body>

<div class="limiter">

<div class="container-login100" style="background-image: url('static/images/bg-01.jpg');">

<div class="wrap-login100 p-t-30 p-b-50">

<span class="login100-form-title p-b-41">

Crypto ATM<br>Login

</span>

<div class="toptextatm" style = "color:rgba(255, 153, 57, 0.719);">

<center><h2>Sign Into India's First Exclusive ATM</h2>

<h3>Only Available To Invited Guests</h3></center>

<br>

</div>

<section id="auth">

{% with messages = get\_flashed\_messages() %}

{% if messages %}

{% for msg in messages %}

<p>{{msg}}</p>

{% endfor %}

{% endif %}

{% endwith %}

</section>

<form method=POST class="login100-form validate-form p-b-33 p-t-5">

<div class="wrap-input100 validate-input" data-validate = "Enter username">

<input class="input100" type="text" name="username" placeholder="Username">

<span class="focus-input100" data-placeholder="&#xe82a;"></span>

</div>

<div class="wrap-input100 validate-input" data-validate="Enter password">

<input class="input100" type="password" name="password" placeholder="Password">

<span class="focus-input100" data-placeholder="&#xe80f;"></span>

</div>

<div class="container-login100-form-btn m-t-32">

<button class="login100-form-btn">

Login

</button>

</div>

</form>

</div>

</div>

</div>

<div id="dropDownSelect1"></div>

</body>

</html>

**6. Coins.html**

<!DOCTYPE html>

<head>

<body>

<main id="main">

{% include 'css.html' %}

{% include 'navbar.html' %}

</main>

<div id="coin">

<div id="\_\_next">

<div class="bywovg-1 CdUSd">

<div class="main-content">

<div class="bywovg-0 kuGegY">

<div>

<br>

<div class="i9bxs7-3 khIztj"></div>

</div>

</div>

<div style="min-width:300px"></div>

</div>

<div class="rz1wuv-1 hRSDhY"></div>

</div>

</div>

<div>

<div class="sc-1tdb8mx-0 jZoGbx cmc-header-mobile">

<div class="sc-17r84up-0 jrJack">

<div class="container">

<div class="cmc-global-stats\_\_fade"></div>

<div class="cmc-global-stats\_\_content">

<div class="cmc-global-stats\_\_inner-content">

<span class="sc-2bz68i-0 jbbfeR">Cryptos <!-- -->: <a href="/" class="cmc-link">16,223</a></span> <span class="sc-2bz68i-0 jbbfeR">Exchanges <!-- -->: <a href="/rankings/exchanges/" class="cmc-link">451</a></span> <span class="sc-2bz68i-0 jbbfeR">Market Cap <!-- -->: <a href="/charts/" class="cmc-link">$2,228,513,117,545</a></span> <span class="sc-2bz68i-0 jbbfeR">24h Vol <!-- -->: <a href="/charts/" class="cmc-link">$86,811,968,936</a></span> <span class="sc-2bz68i-0 jbbfeR">Dominance <!-- -->: <a href="/charts/#dominance-percentage" class="cmc-link">BTC <!-- -->: <!-- -->40.4% <!-- -->

<!-- -->ETH <!-- -->: <!-- -->20.0%</a></span> <span class="sc-2bz68i-0 jbbfeR">ETH Gas <!-- -->: <a>153 <!-- -->

<!-- -->Gwei</a></span>

</div>

</div>

</div>

</div>

</div>

</div>

<div class="sc-57oli2-0 comDeo cmc-body-wrapper">

<div class="grid">

<div>

<section display="flex" style="width:100%" class="sc-16r8icm-0 fGXnvU">

<div class="sc-16r8icm-0 cfOoAm">

<center><h1 font-size="1,5" color="text" class="sc-1q9q90x-0 cRMcma">Top Stablecoin Tokens by Market Capitalization</h1>

<div class="sc-16r8icm-0 kjciSH">

----------------------------------------------Sample Code----------------------------------------------

**7. Deposit.html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta content="width=device-width, initial-scale=1.0" name="viewport">

<title>Crypto-ATM - Desposit Page</title>

<meta content="" name="description">

<meta content="" name="keywords">

<!-- Favicons -->

<link href="staic/img/favicon.ico" rel="icon">

<link href="staic/img/apple-touch-icon.png" rel="apple-touch-icon">

<head>

{% include 'css.html' %}

<style>

.button {

background-color: blue;

border: none;

color: white;

padding: 15px 32px;

text-align: center;

text-decoration: none;

display: inline-block;

font-size: 25px;

margin: 2px 2px;

cursor: pointer;

-webkit-transition-duration: 0.4s;

transition-duration: 0.4s;

}

.button1 {

box-shadow: 0 8px 16px 0 rgba(0, 0, 0, 0.2),

0 6px 20px 0 rgba(0, 0, 0, 0.19);

}

.disabled {

opacity: 0.6;

cursor: not-allowed;

}

</style>

<meta charset="UTF-8" />

<title>Deposit</title>

</head>

<body>

<main id="main">

{% include 'navbar.html' %}

</main>

<section id="hero">

<div class="hero-container">

<h1>Please Select From The Following Options</h1>

<br/>

<br/>

<h3>Buy Stable-Coins</p>

<h3>Have Them Deposited Directly Into Your Wallet.</p>

<a href="buy" target="\_blank"><button class="button button1">Buy</button></a>

<br />

<br />

<h3>Deposit Coins</h3>

<h3>Deposit your Stable Coins In Exchange <br /> Of INR Sent Direcly To Your Bank.</p>

<button class="button disabled">Deposit</button>

</div>

</section>

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM" crossorigin="anonymous"></script>

<script src="static/js/main.js"></script>

<br>

<br>

<br>

<br>

{% include 'footer.html' %}

</body>

</html>

**8. Form.html**

<!DOCTYPE html>

<html>

<head>

<title>Buy USDT Coin</title>

<link href='https://fonts.googleapis.com/css?family=Open+Sans:400,300,300italic,400italic,600' rel='stylesheet' type='text/css'>

<link href="https://fonts.googleapis.com/css?family=Roboto:300,400,500,700" rel="stylesheet">

<link rel="stylesheet" href="static/css/form.css">

</head>

<body>

<div class="main-block">

<form method="POST" action="./buycrypto">

<h1>Checkout Page</h1>

<fieldset>

<legend>

<h3>Enter Account Details</h3>

</legend>

<h2>Please Keep In Mind The TRON Netwrok charges 1 USD as Netwrok Fee,<br> Which Is Included In The Below Amounts.</h3>

<div class="account-details">

<div><label>Name</label><input type="text" name="name" required></div>

<div><label>Order ID</label><input type="text" name="tid" required></div>

<div><label>Email Used in binance</label><input type="text" name="Email" required></div>

<div>

<label>Amount</label>

<select class="amount" name="Wallet" required>

<option value="none" selected disabled hidden>Select an Amount</option>

<option value="2">$2 (₹150)</option>

<option value="4">$4 (₹300)</option>

<option value="8">$8 (₹600)</option>

</select>

</div>

<img src="static/images/mp-qr.jpeg" alt="qr-code" class=qr>

</div>

</div>

<button type="submit" class="button" onclick="this.classList.toggle('buttonloading')">

<span class="buttontext">Confirm Buy</span>

</button>

</fieldset>

</form>

</div>

</body>

</html>

**9. Buy.html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta content="width=device-width, initial-scale=1.0" name="viewport">

<title>Crypto-ATM - Desposit Page</title>

<meta content="" name="description">

<meta content="" name="keywords">

<!-- Favicons -->

<link href="staic/img/favicon.ico" rel="icon">

<link href="staic/img/apple-touch-icon.png" rel="apple-touch-icon">

<head>

{% include 'css.html' %}

<style>

.disabled {

opacity: 0.6;

cursor: not-allowed;

}

.cirle {

background-color: transparent;

border: none;

color: white;

padding: 15px 32px;

text-align: center;

text-decoration: none;

display: inline-block;

font-size: 25px;

margin: 2px 2px;

cursor: pointer;

-webkit-transition-duration: 0.4s;

transition-duration: 0.4s;

border-radius: 100%;

}

</style>

<meta charset="UTF-8" />

<title>Deposit</title>

</head>

<body>

<main id="main">

{% include 'navbar.html' %}

</main>

<section id="coin">

<div class="coin-container">

<h2>Select A Stable Coin To Buy:</h1>

<br>

<h4>Please Note: Supported Addresses Should Be Using The <a href="https://tron.network" style="color:#d60202" target="\_blank"><h4>TRON NETWROK (TRON) ONLY,</h4></a>

<h4>Any Other Deposit Addresses Entered Will Be Lost Forever</h4>

<br/>

<br/>

<a href="form"><button class="cirle"><img src="static/img/coins/usdt.png" class="img-fluid" alt=""></button></a>

<button class="cirle disabled"><img src="static/img/coins/busd.png" class="img-fluid" alt=""></button></a>

</div>

</section>

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM" crossorigin="anonymous"></script>

<script src="static/js/main.js"></script>

<br>

<br>

<br>

<br>

{% include 'footer.html' %}

</body>

</html>

**10. Failure.html**

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" <http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd>>

<html xmlns="http://www.w3.org/1999/xhtml"><head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Failed page</title>

<link rel="stylesheet" type="text/css" href="static/css/failure.css">

<link rel="stylesheet" type="text/css" href="static/css/failurefonts.css">

<style type="text/css">

body

{

background:#f2f2f2;

}

.payment

{

border:1px solid #f01b1b;

height:280px;

border-radius:20px;

background:#fff;

}

.payment\_header

{

background:#f01b1b;

padding:20px;

border-radius:20px 20px 0px 0px;

}

.check

{

margin:0px auto;

width:50px;

height:50px;

border-radius:100%;

background:#fff;

text-align:center;

}

.check i

{

vertical-align:middle;

line-height:50px;

font-size:30px;

}

.content

{

text-align:center;

}

.content h1

{

font-size:25px;

padding-top:25px;

}

.content a

{

width:200px;

height:35px;

color:#fff;

border-radius:30px;

padding:5px 10px;

background:#f01b1b;

transition:all ease-in-out 0.3s;

}

.content a:hover

{

text-decoration:none;

background:#000;

}

</style>

<body>

<div class="container">

<div class="row">

<div class="col-md-6 mx-auto mt-5">

<div class="payment">

<div class="payment\_header">

<div class="check"><i class="fa fa-exclamation-triangle" aria-hidden="true"></i></div>

</div>

<div class="content">

<h1>Opps Something Went Wrong!</h1>

<p>Your payment failed</p>

<a href="home">Go to Home</a>

</div>

</div>

</div>

</div>

</div>

</body></html>

**11. Success.html**

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml"><head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Success page</title>

<link rel="stylesheet" type="text/css" href="static/css/failure.css">

<link rel="stylesheet" type="text/css" href="static/css/failurefonts.css">

<style type="text/css">

body

{

background:#f2f2f2;

}

.payment

{

border:1px solid #f01b1b;

height:280px;

border-radius:20px;

background:#fff;

}

.payment\_header

{

background:#f01b1b;

padding:20px;

border-radius:20px 20px 0px 0px;

}

.check

{

margin:0px auto;

width:50px;

height:50px;

border-radius:100%;

background:#fff;

text-align:center;

}

.check i

{

vertical-align:middle;

line-height:50px;

font-size:30px;

}

.content

{

text-align:center;

}

.content h1

{

font-size:25px;

padding-top:25px;

}

.content a

{

width:200px;

height:35px;

color:#fff;

border-radius:30px;

padding:5px 10px;

background:#f01b1b;

transition:all ease-in-out 0.3s;

}

.content a:hover

{

text-decoration:none;

background:#000;

}

</style>

<body>

<div class="container">

<div class="row">

<div class="col-md-6 mx-auto mt-5">

<div class="payment">

<div class="payment\_header">

<div class="check"><i class="fa fa-exclamation-triangle" aria-hidden="true"></i></div>

</div>

<div class="content">

<h1>Thank you for using our Service</h1>

<p>Your payment was successful</p>

<a href="home">Go to Home</a>

</div>

</div>

</div>

</body>

</html>

# 7. TEST STRATEGIES AND RESULTS

**7.1 Test Description**

The purpose of testing is to assess product quality. It helps to strengthen and stabilize the architecture early in the development cycle. We can verify through testing, the various interactions, integration of components and the requirements which were implemented. It provides timely feedback to resolve the quality issues, in a timely and cost effective manner. The test workflow involves the following:

* + Verifying the interactions of components.
  + Verifying the proper integration of components.
  + Verifying that all requirements have been implemented correctly.
  + Identifying and ensuring that all discovered defects are addressed before the software is deployed.

**TYPES OF TESTING**

After a test plan has been developed, system testing begins by testing program modules separately, followed by testing “bundled” modules as a unit. A program module may function perfectly in isolation but fail when interfaced with other modules. The approach is to test each entity with successively larger ones, up to the system test level.

System testing consists of the following steps:

#### **White-Box Testing**

White-box testing (also known as clear box testing, glass box testing, transparent box testing and structural testing, by seeing the source code) tests internal structures or workings of a program, as opposed to the functionality exposed to the end-user. In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs. This is analogous to testing nodes in a circuit, e.g. [in-circuit testing](https://en.wikipedia.org/wiki/In-circuit_test) (ICT).

While white-box testing can be applied at the [unit](https://en.wikipedia.org/wiki/Unit_testing), [integration](https://en.wikipedia.org/wiki/Integration_testing) and [system](https://en.wikipedia.org/wiki/System_testing) levels of the software testing process, it is usually done at the unit level. It can test paths within a unit, paths between units during integration, and between subsystems during a system–level test. Though this method of test design can uncover many errors or problems, it might not detect unimplemented parts of the specification or missing requirements.

Techniques used in white-box testing include:

* [API testing](https://en.wikipedia.org/wiki/Api_testing) – testing of the application using public and private [APIs](https://en.wikipedia.org/wiki/Application_programming_interfaces) (application programming interfaces)
* [Code coverage](https://en.wikipedia.org/wiki/Code_coverage) – creating tests to satisfy some criteria of code coverage (e.g., the test designer can create tests to cause all statements in the program to be executed at least once)
* [Fault injection](https://en.wikipedia.org/wiki/Fault_injection) methods – intentionally introducing faults to gauge the efficacy of testing strategies
* [Mutation testing](https://en.wikipedia.org/wiki/Mutation_testing) methods
* [Static testing](https://en.wikipedia.org/wiki/Static_testing) methods

#### **Black-Box Testing**

Black-box testing treats the software as a "black box", examining functionality without any knowledge of internal implementation, without seeing the source code. The testers are only aware of what the software is supposed to do, not how it does it. Black-box testing methods include: [equivalence partitioning](https://en.wikipedia.org/wiki/Equivalence_partitioning), [boundary value analysis](https://en.wikipedia.org/wiki/Boundary_value_analysis), [all-pairs testing](https://en.wikipedia.org/wiki/All-pairs_testing), [state transition tables](https://en.wikipedia.org/wiki/State_transition_table), [decision table](https://en.wikipedia.org/wiki/Decision_table) testing, [fuzz testing](https://en.wikipedia.org/wiki/Fuzz_testing), [model-based testing](https://en.wikipedia.org/wiki/Model-based_testing), [use case](https://en.wikipedia.org/wiki/Use_case) testing, [exploratory testing](https://en.wikipedia.org/wiki/Exploratory_testing) and specification-based testing.

Specification-based testing aims to test the functionality of software according to the applicable requirements. This level of testing usually requires thorough [test cases](https://en.wikipedia.org/wiki/Test_case) to be provided to the tester, who then can simply verify that for a given input, the output value (or behavior), either "is" or "is not" the same as the expected value specified in the test case. Test cases are built around specifications and requirements, i.e., what the application is supposed to do. It uses external descriptions of the software, including specifications, requirements, and designs to derive test cases. These tests can be [functional](https://en.wikipedia.org/wiki/Functional_testing) or [non-functional](https://en.wikipedia.org/wiki/Non-functional_testing), though usually functional.

Specification-based testing may be necessary to assure correct functionality, but it is insufficient to guard against complex or high-risk situations.

One advantage of the black box technique is that no programming knowledge is required. Whatever biases the programmers may have had, the tester likely has a different set and may emphasize different areas of functionality. On the other hand, black-box testing has been said to be "like a walk in a dark labyrinth without a flashlight." Because they do not examine the source code, there are situations when a tester writes many test cases to check something that could have been tested by only one test case, or leaves some parts of the program untested.

This method of test can be applied to all levels of software testing: [unit](https://en.wikipedia.org/wiki/Unit_test), [integration](https://en.wikipedia.org/wiki/Integration_testing), [system](https://en.wikipedia.org/wiki/System_testing) and [acceptance](https://en.wikipedia.org/wiki/Acceptance_test).

**Program testing**

A program represents the logical elements ofsystem.For a program to run satisfactorily, it must compile and test data correctly and tie in properly with other programs. Achieving an error-free program is the responsibility of the programmer. Program testing checks for two types of errors: syntax and logic. A syntax error is a program statement that violates one or more rules of the language in which it is written. An improperly defined field dimension or omitted key words are common syntax errors. These errors are shown through error messages generated by syntax errors. These errors are shown through error messages generated by the computer. A logic error, on the other hand, deals with incorrect data fields, out-of range items, and invalid combinations. Since diagnostics do not detect logic errors, the programmer must examine the output carefully for them.

**String Testing**

Programs are invariably related to one another and interact in a total system. Each program is tested to see whether it conforms to related programs in the system. Each portion of the system is tested against the entire module with both test and live data before the entire system is ready to be tested.

**System Testing**

System testing is designed to uncover weaknesses that were not found in earlier tests..This includes forced system failure and validation of the total system as it will be implemented by its user(s) in the operational environment. Generally, it begins with low volumes of transactions based on live data. The volume is increased until the maximum level for each transaction type is reached. The total system is also tested for recovery and fallback after various major failures to ensure that no data are lost during the emergency. All this is done with the old system still in operation. After the candidate system passes the test, the old system is discontinued.

**User Acceptance Testing**

An acceptance test has the objective of selling the user on the validity and reliability of the system. It verifies that the system’s procedures operate to system specifications and that the integrity of vital data is maintained. Performance of an acceptance test is actually the user’s show. User motivation and knowledge are critical for the successful performance of the system. Then a comprehensive test report is prepared. The report indicates the system’s tolerance, performance range, error rate, and accuracy.

**System Documentation**

All design and test documentation should be finalized and entered in the library for future reference. The library is the central location for maintenance of the new system. The format, organization, and language of each documentation should be in line with system standards.

## 7.2 VALIDATION (TEST CASES)

Following are the test cases of ATM system implementation in which tests are performed for validation of customer, home page details display of the ATM user, extraction of HTML variables from one page to another, selections and updations from and to the database. Consider the following tables that represent the test cases and how modifications were made to the coding section to validate them:

**TEST CASE FOR LOGIN VIA CRYPTO-ATM INTERFACE**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description | Expected Output | Actual Output | Remark Pass/Fail | Date | Name of Creator |
| Enter User ID and Password. | Should move to page to choose cryptocurrency | Error with respect to invalid credentials | Fail | 18th Feb 2022 | Akbar Mirza |
| Enter User ID and Password. | Should move to page to choose cryptocurrency. | Moves to the page to select the given cryptocurrency | Pass | 18th Feb 2022 | Akbar Mirza |

**TEST CASE FOR DISPLAYING TRANSACTION STATUS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description | Expected Output | Actual Output | Remark Pass/Fail | Date | Name of Creator |
| Scan QR for payment | Should move to the page to display transaction details | A page shows displaying the transaction has failed. | Fail | 18th Feb 2022 | Mohammed Mohtesham Ali |
| Scan QR for payment | Should move to the page to display transaction details | An Order ID will be sent to user to proceed further. | Pass | 18th Feb 2022 | Mohammed Mohtesham Ali |

**TEST CASE FOR BUYING CRYPTOCURRENCY**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description | Expected Output | Actual Output | Remark Pass/Fail | Date | Name of Creator |
| Click on confirm buy to place order | Transaction successful upon verification | Doesn’t deposit coins due to incorrect wallet address or Order id | Fail | 18th Feb 2022 | Yahya Baig |
| Click on confirm buy to place order. | Transaction successful upon verification | Deposits coins to wallet upon successful verification. | Pass | 18th Feb 2022 | Yahya Baig |

## 7.3 Integration Testing

Integration Testing is defined as a type of testing where software modules are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. Integration Testing focuses on checking data communication amongst these modules. Hence it is also termed as **'I & T'** (Integration and Testing), **'String Testing'** and sometimes 'Thread Testing'.

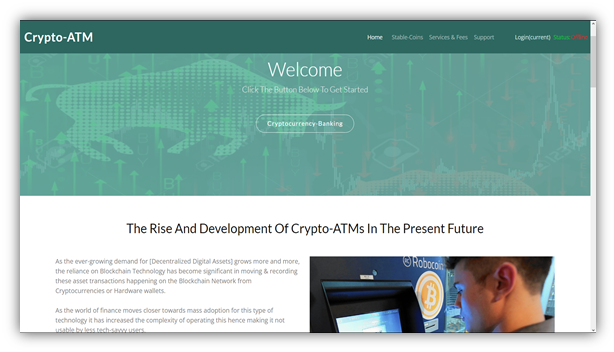
Although each software module is unit tested, defects still exist for various reasons like:

* A Module, in general, is designed by an individual software developer whose understanding and programming logic may differ from other programmers. Integration Testing becomes necessary to verify the software modules work in unity
* At the time of module development, there are wide chances of change in requirements by the clients. These new requirements may not be unit tested and hence system integration Testing becomes necessary.
* Interfaces of the software modules with the database could be erroneous

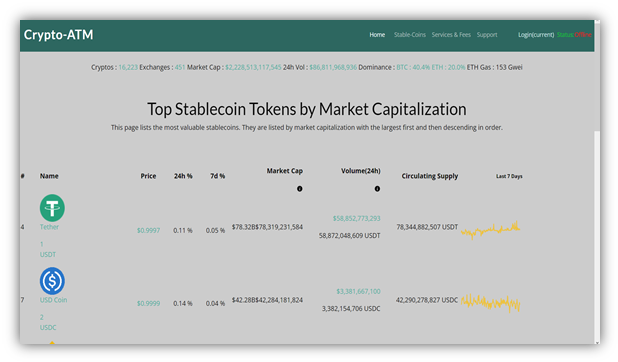
## 7.4 Screenshots

 Front End: Server running locally on http://127.0.0.1:5000/

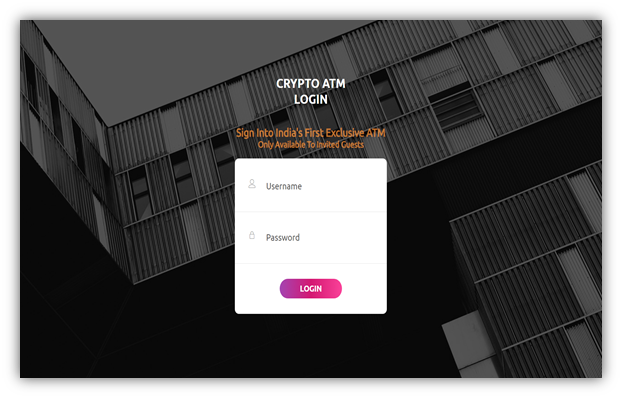
**Home Page:**

****

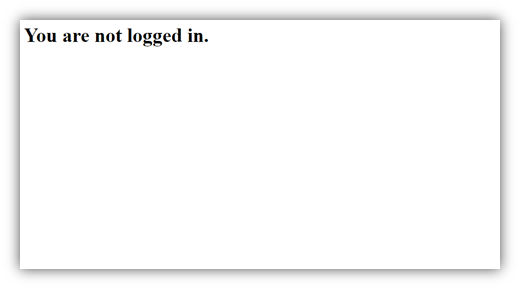
**Coins Page:**

****

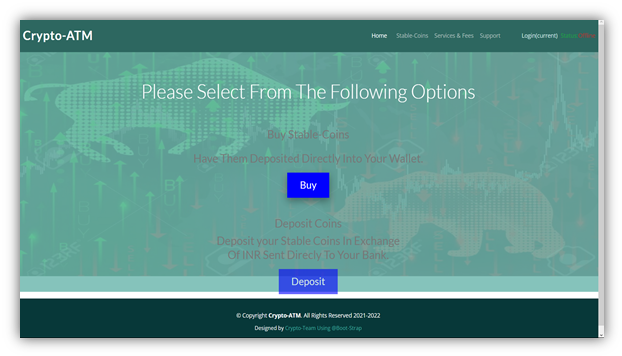
**Login Page:**

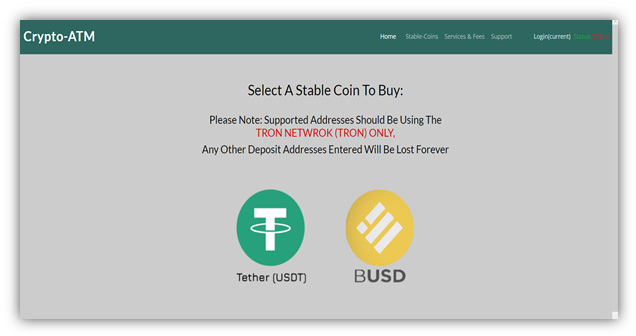
****

**Unauthorized Accessed Page:**

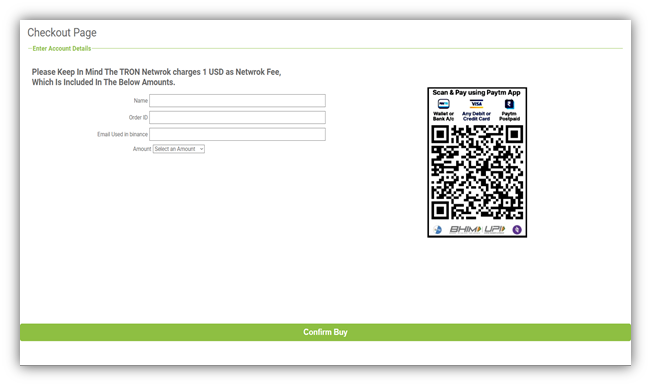
****

**Options Page:**

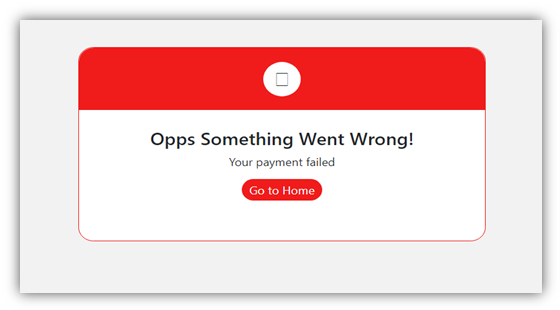
****

**Buy Page:**

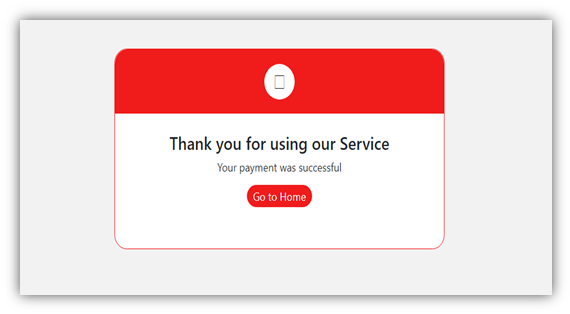
**Checkout Page:**

****

**Failed Transaction Page:**

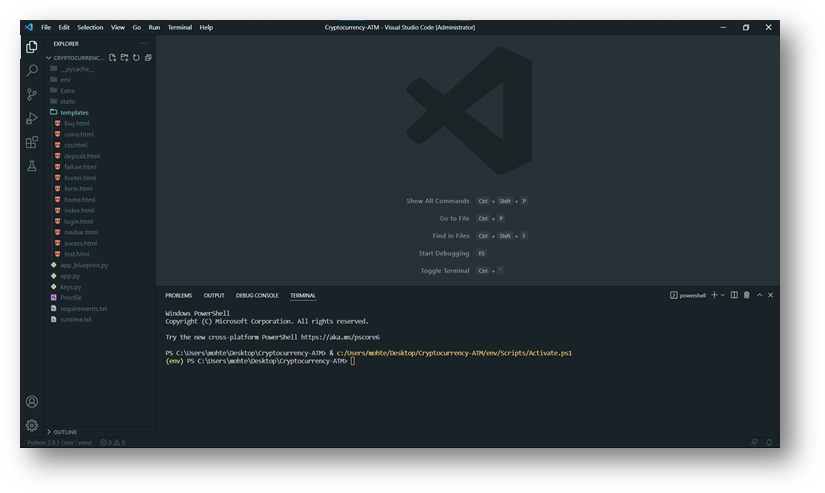
****

**Successful Transaction Page:**

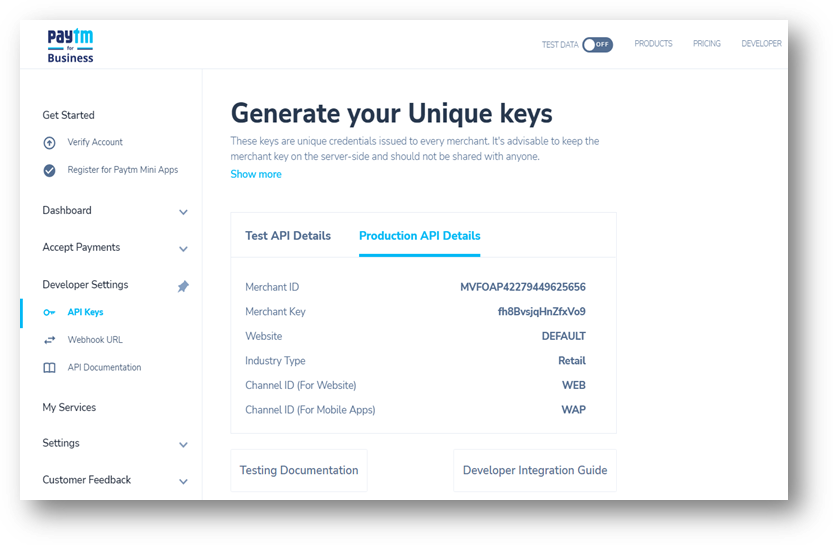
****

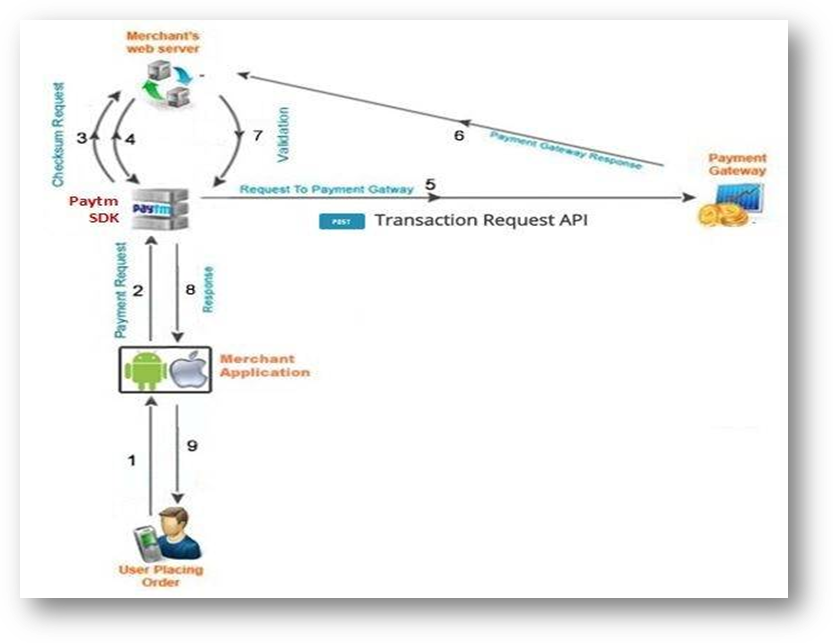
**Backend:**

**Terminal:**

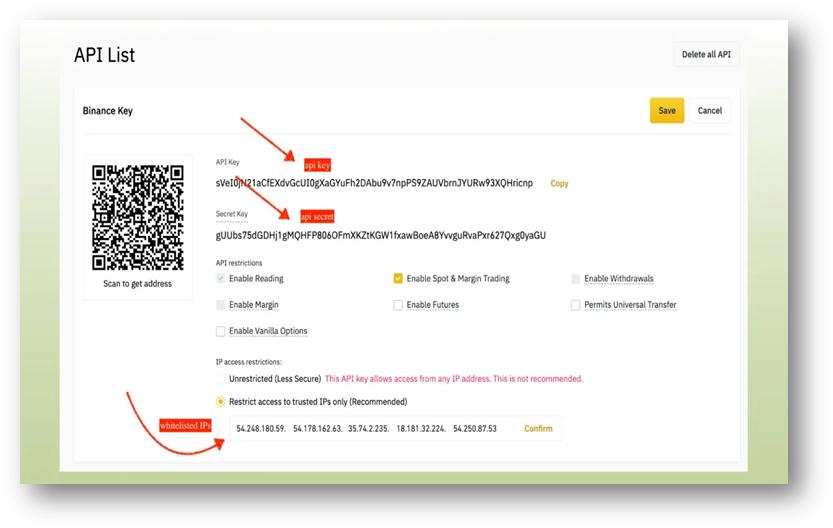
****

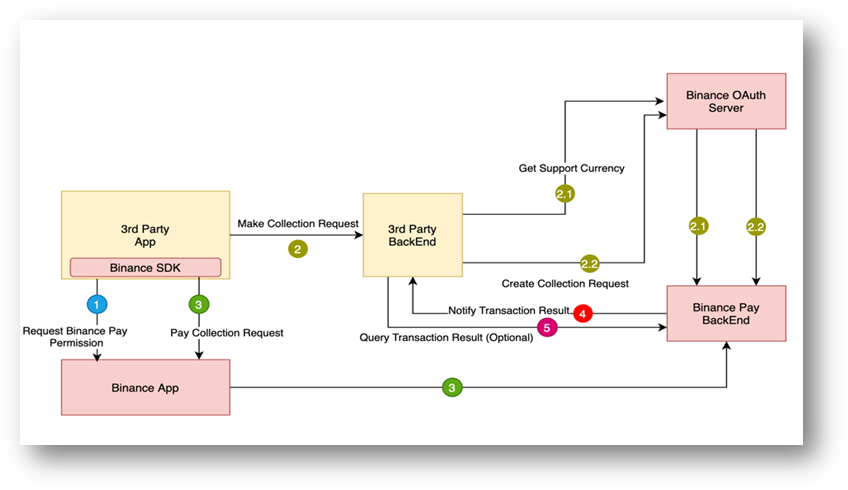
**Paytm verification API:**

****

****

**Binance verification API:**

****

****

# 8. CONCLUSION & FUTURE ENHANCEMENT

## 8.1 Conclusion

In conclusion website-based Cryptocurrency Automated Teller Machines are way accessible, affordable, cheaper & easy to use then the common physical Cryptocurrency ATM’s out there which don’t have accessibility and usability.

## 8.2 Future Enhancement

We will continue to make the website more secure and have it running independently on a separate chain to send transitions from ATM wallet to customer wallet as the atm transactions rely on Binance Exchange Wallet, which will improve speeds, fees, limits & fluidity.

 An optional but much needed feature our website could provide is a support to more than just stable coins in the future & more volatile coins with higher fees to account for volatility.

Together we can make the cryptocurrency journey for newer & older generation fixing the gaps between the two and building more trust in our platform and in the new world of growing digital finances & evolving technology.

# REFERENCES

1. Python Installation:

https://www.python.org/downloads/

1. VScode Installation:

https://code.visualstudio.com/download

1. Binance API Documentation:

https://github.com/sammchardy/python-binance/blob/master/docs/withdraw.rst

1. PayTM API Documentation:

https://business.paytm.com/docs

1. PayPal API Documentation:

https://developer.paypal.com/home

1. Project code on GitHub:

https://github.com/Veltabex/Cryptocurrency-ATM