# Final Project Documentation

# Shanchay no money, no life.....

## Submission By:


**Name** : K. M. Ashiful Islam Istiuk

**I'd no** : 201-35-3017

**Department** : Software Engineering

**Section** : A


## *Submitted To:*

**Course Teacher's name** : Md. Shohel Arman

**Designation** : Senior Lecturer

# APPROVAL

This project titled "**Shanchay**", submitted by K M Ashiful Islam Istiuk, ID: 201-35-3017 to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Software Engineering and approved as to its style and contents.

# DECLARATION

I hereby declare that I have done this project under the supervision of Md. Shohel Arman, Senior Lecturer, Department of Software Engineering, Daffodil International University. I also declare that this project or any part of this is unique and has not been submitted elsewhere for the award of any degree.

Name: K M Ashiful Islam Istiuk
ID: 201-35-3017
Department of Software Engineering
Faculty of Science & Information Technology
Daffodil International University

**Certified by:**

Md. Shohel Arman
Senior Lecturer
Department Of Software Engineering
Faculty of Science & Information Technology
Daffodil International University

# ACKNOWLEDGEMENT

At first, I am blessed that I successfully moved towards the last semester. I am pleased with my almighty. First, at the beginning of university life, I have learned a lot about software development as well as computer science-related knowledge from my university's knowledgeable teachers and helpful course mates. Teachers teach us ethics, morality, and politeness as well as software knowledge and related knowledge. I must be thankful to my parents and my family to give me the opportunity and always be to myself. My family always supports me. I am highly indebted to Md. Shohel Arman for his guidance and constant supervision as well as for providing necessary information regarding the project & for his support in completing the project. My supervisor supports me to make this project "**Shanchay**" a successful end. My thanks and appreciations go to my course mates in developing the project and people who have willingly helped me out with their abilities.

# ABSTRACT

"**Recommendation Based Scholarship Website for Bangladeshi Student**" is an educational project. Our main motivation is to make a recommendation-based scholarship website so that Bangladeshi students can easily get scholarships in abroad. The system will automate scholarships for students according to their inputs so that both technical and non-technical students can complete the admission process themselves without taking any help form agencies. It is developed of scholarship departments for the group of colleges. Scholarship Officers can add data regarding the scholarships and students. Institution sort out eligible students for the scholarships. It will be an inducement to improve the quality of the study and thereby attaining high standards.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# CHAPTER 1: INTRODUCTION

## 1.1 Project Overview

- This project is developed for who are interested on savings. Though via some Avenue are very popular, majority of people are not find the easiest platform to save money. To have a general awareness it will be more convenient if each and everyone use **Shanchay.** Istiuk is  a student. He is a wastrel person. He wastes his money much way. He eats unhealthy food, buy unnecessary things, purchases game, dirking unnecessary soft drinks, going to restaurant and so on. For his habits he does not make any savings. After Semester his all friends going for trip. Some of friend buying gadget. But he cannot because he has no savings. Every Time He wants to save his money but he cannot restrain himself from spending money.

.

## 1.2 Project Purpose

The project "**Shanchay**" is a software package developed for saving money. The software is very helpful to save the money. The project gives the opportunity to buy coin based on their money. And after sometime user can withdraw coin.

### 1.2.1 Background

In the current context, Money is very important for our life. And savings money is not an easy task.

### 1.2.2 Benefits and Beneficiaries

Through the system we can easily maintain the data without any loss or damage.

By using this system time consumption will be reduced.

By using this website User can easily save their money.

Admin can easily check the reward of an users and can update on the user Dash borad.

### 1.2.3 Goals

The goal or aim of our project is to automate scholarships for students according to their inputs. The project is aiming -

To allow Bangladeshi students to easily access scholarship

details. To maintain the data without any loss or damage.

To reduce the time for finding a suitable scholarship.

To create a network amongst Bangladeshi students so that more students get the opportunity to study in abroad.

## 1.3 Stakeholders

There are many members are associate with this project. They have helped to develop the system directly or indirectly.

### 1.3.1 Internal Stakeholders:

1. Admin

### 1.3.2 External Stakeholders:

2. User

# CHAPTER 2: USER CLASS AND CHARACTERISTICS

## 2.1 Admin

User can buy ,withdraw select transaction method through the system.

## 2.3 Admin

This module deals with the issue coin. he stores all the coin regarding the reward done by the student.

# CHAPTER 3: SOFTWARE REQUIREMENTS SPECIFICATION

Requirements analysis is the best process of identifying the user satisfaction from the system. So, requirements analysis is an important part of project management. When I selected this project, I thought about some specific software requirement, like as –

Who is the stakeholder of this project?
Is it helpful for them or not?
Functional and non- functional
requirements Maintenance of the system
Is it efficient for using?

## 3.1 Functional Requirement

The Functional Requirements for the system is given below:

| FR-01 | **Registration** |
|---|---|
| Description | After entering the provided/ define URI in any internet browser, in the software index page the user must have to register him first. |
| Stakeholders | Admin , User |

| FR-02 | **Login** |
|---|---|
| Description | After every time of access he has to login first before entering the main function of the application. It is mandatory to login for all users. |
| Stakeholders | Admin , User |

| FR-03 | **Dash Board** |
|---|---|
| Description | User can see their information and can update their profile. Admin access user's profile for add coin information. |
| Stakeholders | Admin , User |

| FR-04 | **Buy Coin** |
|---|---|
| Description | Every user can buy coin based on their money with their favourite transaction method.(1 coin = 80tk.) |
| Stakeholders | User |

| FR-05 | Withdraw |
|---|---|
| Description | User can withdraw their money by withdraw coin if they reached at least 100 by their favourite method. |
| Stakeholders | User |

| FR-06 | Transaction Method |
|---|---|
| Description | It is an important part of our system. User can buy and withdraw their coin by using transaction method. There are all kind of transaction method in our system. |
| Stakeholders | User. |

| FR-07 | Reward |
|---|---|
| Description | If any user achieved 100 coin the system will give 10 coin extra as gift. If 200 coin him received 20 coins extra. Admin will calculate the reward and update on the user's profile. |
| Stakeholders | Admin , User |

## 3.2 Non – Functional Requirements

The non – functional requirements for the project is given below:

**Table 1.2 Non Functional Requirements**

| NFR01 | Usability |
|---|---|
| **Description** | The system is designed with completely automated process. Hence there is no or less intervention. |
| **Priority** | High |

| NFR02 | Reliability and Security |
|---|---|
| **Description** | The system is more reliable because of the qualities that are inherited from the chosen platform Dot net. And it provides secure access of confidential data with unique id and password. |
| **Priority** | High |

| NFR03 | Availability |
|---|---|
| **Description** | The system should work 24/7 as user can get access and service. |
| **Priority** | High |

| NFR04 | Accuracy |
|---|---|
| **Description** | Data or process requirement concerned with defining the precision which the solution will record or produce data |
| **Priority** | High |

## 3.3 Performance Requirements

A requirement that specifies a performance characteristic that a system or system or system Component must possess; for example, speed, accuracy, frequency.

### 3.3.1 Speed and Latency Requirements

The system is required a fair amount of speed especially while browsing

**Table 1.3 Speed and Latency Requirement**

| | |
|---|---|
| PR-01 | **The Landing page will response within a second** |
| Description | While the user's browsing the system the landing page will show within a second. It also depends on user's internet connection. |
| Stakeholders | User, Admin. |

### 3.3.2 Precision and Accuracy Requirements

There are no specific precision and accuracy requirements

### 3.3.3 Capacity Requirements

The system is able to manage all the information of Students.

**Table 1.4 Capacity Requirements**

| | |
|---|---|
| PR-02 | **Initially the system will store 30,000 user information.** |
| Description | The information of user will be stored in database. |
| Stakeholders | User , Admin. |

## 3.4 Dependability Requirements

The flexibility of current frameworks encourage system architects to enable reconfiguration mechanisms that refocus the available, safe resources to support the most critical services rather than over-provisioning to build failure-proof system. Therefore, these requirements are essential.

**Table 1.5 Dependability Requirements**

### 3.4.1 Reliability and Availability

In order to smooth operations the system must be available around the clock.

| | |
|---|---|
| DR-01 | **The system must be available 24x7** |
| Description | The system must be available 24 hours in a day<br>The system must be updated regularly |
| Stakeholders | Admin |

### 3.4.2 Robustness and Fault Tolerance Requirements

The system will almost ensure 0% crush in any single minor error and don't give any wrong calculation.

| | |
|---|---|
| DR-02 | **The system handles over access and system errors** |
| Description | Sometimes multiple users can over access to this system. The system can handle multiple user access |
| Stakeholders | N/A |

### 3.4.3 Safety Critical Requirements

There are no specific safety critical requirements.

## 3.5 Maintainability and Supportability

Supportability is the degree to which system design characteristics and planned logistics resources meet system requirements. Supportability is the capability of a total system design to support operations and readiness needs throughout the life-cycle of a system at an affordable cost.

**Table 1.6 Maintainability and Supportability Requirements**

### 3.5.1 Maintenance Requirements

| | |
|---|---|
| DR-01 | **The system helps to update any information in any time** |
| Description | The admin post any events and can enable to change or update any information in any situation |
| Stakeholders | Admin |

### 3.5.2 Supportability Requirements

In order to understand the system's behavior on a technical support required by the system operator. The reason for reading them might be

System malfunction has occurred and the system operator has to find the exact point of time when this happened

System produces wrong results and the developers must be able to reproduce the data flow through the system

Hacker tried to breach the system's security mechanisms and the system operator must understand what he did

### 3.5.3 Adaptability Requirements

There are no specific adaptability requirements

## 3.6 Security Requirements

There are no access requirements beside those that have been outlined in the below:

The software must validate all user input to ensure it does not exceed the size specified for that type of input

The server must authenticate every request accessing the restricted Web pages

After authenticating the browser, the server must determine whether that browser is authorized to access the requested restricted Web pages

The system must have security controls to protect against denial-of-service attacks

The system must encrypt sensitive data transmitted over the Internet between the server and the browser

To get access to this system or a specific module the system must provide a central authentication mechanism. In order to prevent anyone to exploit stolen all users password must be encrypted in hash process.

**Table 1.7 Security Requirements**

### 3.6.1 Access Requirements

To get access to the system, the system provides authorization/authentication way. This system uses various modules.

| SR-01 | **The system provides security strategies** |
|---|---|
| Description | The system is designed in way that allows all modules to access a mechanism that provides security services. |
| Stakeholders | User, Admin |

### 3.6.2 Privacy Requirements

The system provides a protection of the database in the server. However, the system will have to increment this level of protection because of the personal data mode available on the system & the larger share of people that will be having access to it through the system's registration. The user's privacy will be granted by the limited access that the log in process is going to give to the database

| | |
|---|---|
| SR-02 | **All data will be protected** |
| Description | The main requirement in the context is the generation of user's data for analysis. |

## 3.7 Usability and Human Integrity Requirements

These Requirements define how to meet the physical and cognitive needs of the intended Users of your website or application.

**Table 1.8 Usability and Human Integrity Requirements**

### 3.7.1 Ease of Use Requirements

The system is easy to use and can easily be understandable.

| | |
|---|---|
| UH-01 | **The system must be usable for craft workers with all associate stakeholders** |
| Description | The system indicates the several possibilities that the user has to go on in using the system. The members are allowed to undo any of the operation. |
| Stakeholders | User, Admin |

### 3.7.2 Understand-ability and Politeness Requirements

| | |
|---|---|
| UH-02 | **The features of Scholarship system** |
| Description | The system is more efficiently ease of use more added features .The system is understand-ability for both user. The system will not use any term that is not specified in this system. |
| Stakeholders | Admin |

### 3.7.3 User Documentation

| | |
|---|---|
| DR-01 | **The system developer documentation** |
| Description | To develop this project we have specified requirement of user documentation. The teams are involved to this project documentation. |
| Stakeholders | System Developer |

## 3.8 Look and Feel Requirements

The look and feel requirements describe the intended spirit, the mood, or the style of the product's appearance. These requirements specify the intention of the appearance, and are not a detailed design of an interface.

**Table 1.9 Look and Feel Requirements**

### 3.8.1 Appearance Requirements

| | |
|---|---|
| DR-01 | **Labels of mandatory fields must be bold** |
| Description | Labels of mandatory fields must be bold to identify them as being of mandatory. |
| Stakeholders | User, Admin |

### 3.8.2 Style Requirements

We will provide a web based user interface. This requirement does not only define the necessity to use a CSS but although the requirements regarding the CSS's content as well as CSS framework like bootstrap.

| | |
|---|---|
| DR-01 | **The look and feel must be controllable using style sheet** |
| Description | The styling of the elements of the web based user interface will be defined using CSS, JS and bootstrap. |
| Stakeholders | Admin |

## 3.9 Operational and Environmental Requirements

This requirements focus on how the users will operate the system, including interfaces and interoperability with other systems. The requirements establish how well and under what Conditions the system must perform.

### 3.9.1 Expected Physical Requirements

There are no specific expected physical requirements.

### 3.9.2 Requirement for Interfacing with Adjacent System

There is no specific interfacing with adjacent system requirements

### 3.9.3 Release Requirements

There are no specific release requirements but in the project schedule section it was described briefly.

## 3.10 Legal Requirements

These requirements consider any violence of rules and regulation and which rules should be followed to maintain this system

### 3.10.1 Compliance Requirements

There are no specific compliance requirements

### 3.10.2 Standard Requirements

There are no specific standard requirements

# CHAPTER 4: SYSTEM ANALYSIS

## 4.1 Use Case Diagram

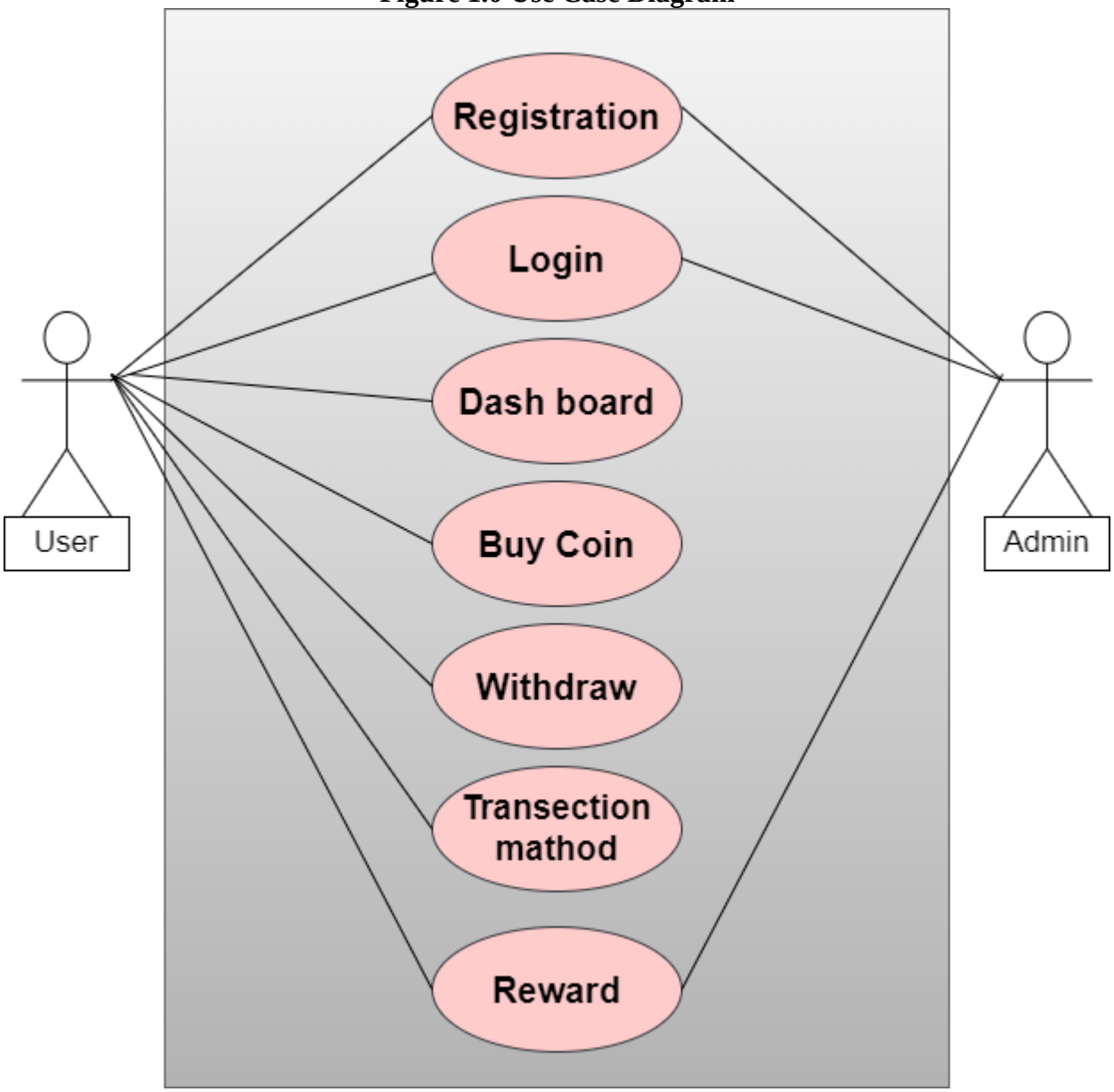
The Case Diagram is used to identify the primary elements and processes that form the system. The primary elements are termed as “actors” and the processes are called “use case”. The Use Case Diagram shows which actor interact with each case.

In this project,

**Primary Actors:** User
**Secondary Actors:** Admin

According to the primary and secondary actors, the Use Case Diagram with use for the project is given below:

**Figure 1.0 Use Case Diagram**



## 4.2 Use Case Description

<table>
<tr><td>Use Case</td><td>Registration & Login</td></tr>
<tr><td>Goal</td><td>Create a new user account into the system.</td></tr>
<tr><td>Preconditions</td><td>User must enter all the required information to create account.</td></tr>
<tr><td>Success End Condition</td><td>Successfully created new user account.</td></tr>
<tr><td>Failed End Condition</td><td>Sorry not created new user account.</td></tr>
<tr><td>Primary Actor:<br>Secondary Actor:</td><td>Admin, User.</td></tr>
<tr><td>Trigger:</td><td>Request to register new user.</td></tr>
<tr><td>Description/ Main Success Scenario:</td><td>
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Download the app from the given link.</td></tr>
<tr><td>2</td><td>Install into the mobile and allow all the necessary permission.</td></tr>
<tr><td>3</td><td>Enter all the required information.</td></tr>
<tr><td>3</td><td>Push Sign In Button.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flaws:</td><td>
<table>
<tr><th>Step</th><th>Branching Action</th></tr>
<tr><td>3x</td><td>Permission not allows.</td></tr>
<tr><td>3x1</td><td>Entered information is not valid.</td></tr>
</table>
</td></tr>
<tr><td>Quality Requirements:</td><td>
<table>
<tr><th>Step</th><th>Requirement</th></tr>
<tr><td>1</td><td>User must be entered all the required information.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td>Use Case</td><td>Dash Board</td></tr>
<tr><td>Goal</td><td>To see overview of whole the system.</td></tr>
<tr><td>Preconditions</td><td>User must be registered and Login into the system.</td></tr>
<tr><td>Success End Condition</td><td>Able to use the system smoothly.</td></tr>
<tr><td>Failed End Condition</td><td>It will be more difficult to use the system.</td></tr>
<tr><td>Primary Actor:<br>Secondary Actor:</td><td>User.</td></tr>
<tr><td>Trigger:</td><td>Request to open the system.</td></tr>
<tr><td>Description/ Main Success Scenario:</td><td>
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Registration for the system.</td></tr>
<tr><td>2</td><td>Log into the system.</td></tr>
<tr><td>3</td><td>Tap Coin to see total number of coin.</td></tr>
<tr><td>4</td><td>Tap reward for get reward.</td></tr>
<tr><td>5</td><td>Tap Buy Coin for buy coin (1 coin =80tk.).</td></tr>
<tr><td>6</td><td>Tap withdraw for withdraw coin.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flaws:</td><td>
<table>
<tr><th>Step</th><th>Branching Action</th></tr>
<tr><td>4x</td><td>If user has not enough coins he cannot get any reward.</td></tr>
<tr><td>5</td><td>If user has less than 100 coins he cannot withdraw the coins.</td></tr>
</table>
</td></tr>
<tr><td>Quality Requirements:</td><td>
<table>
<tr><th>Step</th><th>Requirement</th></tr>
<tr><td>1</td><td>Performance, Speed and Latency.</td></tr>
</table>
</td></tr>
</table>

| **Use Case** | **Buy Coin** |
|---|---|
| **Goal** | Buying coin to save money. |
| **Preconditions** | User must be registered and Login into the system. |

<table>
<tr><td>Success End Condition</td><td>Buy more money and make more savings.</td></tr>
<tr><td>Failed End Condition</td><td>If system fails to perform the future the full project will be failed.</td></tr>
<tr><td>Primary Actor:<br>Secondary Actor:</td><td>User.</td></tr>
<tr><td>Trigger:</td><td>Request to buy coin.</td></tr>
<tr><td>Description/ Main Success Scenario:</td><td><table><tr><th>Step</th><th>Action</th></tr><tr><td>1</td><td>Log into the system.</td></tr><tr><td>2</td><td>Tap Buy Coin for buy coin.</td></tr><tr><td>3</td><td>Chose Transaction method.</td></tr><tr><td>4</td><td>Enter Amount and password.</td></tr></table></td></tr>
<tr><td>Alternative Flaws:</td><td><table><tr><th>Step</th><th>Branching Action</th></tr><tr><td>3</td><td>User can be chosen wrong Trisection method.</td></tr><tr><td>4</td><td>User can be entering wrong password.</td></tr></table></td></tr>
<tr><td>Quality Requirements:</td><td><table><tr><th>Step</th><th>Requirement</th></tr><tr><td>1</td><td>Performance, Speed and Latency.</td></tr></table></td></tr>
</table>

| **Use Case** | **Withdraw** |
|---|---|
| **Goal** | Withdraw the savings. |
| **Preconditions** | User must have to more than 100 coins. |
| **Success End Condition** | User will receive his savings. |
| **Failed End Condition** | User cannot get his savings and user lose the trust about the system. |
| **Primary Actor:**<br>**Secondary Actor:** | User. |
| **Trigger:** | Request to withdraw coin. |

<table>
<tr><td>Description/ Main Success Scenario:</td><td>
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Log into the system.</td></tr>
<tr><td>2</td><td>Tap withdraw for withdraw the coins.</td></tr>
<tr><td>3</td><td>Chose Transaction method.</td></tr>
<tr><td>4</td><td>Enter Amount and password.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flaws:</td><td>
<table>
<tr><th>Step</th><th>Branching Action</th></tr>
<tr><td>3</td><td>User can be chosen wrong Trisection method.</td></tr>
<tr><td>4</td><td>User can be entering wrong password.</td></tr>
</table>
</td></tr>
<tr><td>Quality Requirements:</td><td>
<table>
<tr><th>Step</th><th>Requirement</th></tr>
<tr><td>1</td><td>Performance, Speed and Latency.</td></tr>
</table>
</td></tr>
</table>

| **Use Case** | **Transaction Method** |
|---|---|
| **Goal** | To Buy and Withdraw the coins. |
| **Preconditions** | User has to account on exciting Transaction method. |
| **Success End Condition** | User will be able make transaction easily. |
| **Failed End Condition** | User cannot be make transaction easily. |
| **Primary Actor:**<br>**Secondary Actor:** | User. |
| **Trigger:** | Request to withdraw coin. |

<table>
<tr><td>Description/ Main Success Scenario:</td><td>
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Log into the system.</td></tr>
<tr><td>2</td><td>Tap Buy coin or withdraw.</td></tr>
<tr><td>3</td><td>Chose Transaction method.</td></tr>
<tr><td>4</td><td>Enter Account details and password.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flaws:</td><td>
<table>
<tr><th>Step</th><th>Branching Action</th></tr>
<tr><td>3</td><td>User can be chosen wrong Trisection method.</td></tr>
<tr><td>4</td><td>User can be entering wrong password.</td></tr>
</table>
</td></tr>
<tr><td>Quality Requirements:</td><td>
<table>
<tr><th>Step</th><th>Requirement</th></tr>
<tr><td>1</td><td>Performance, Speed and Latency.</td></tr>
</table>
</td></tr>
</table>

| **Use Case** | **Reward** |
|---|---|
| **Goal** | Grow interest to the users for making more savings. |
| **Preconditions** | User has to more than 100 coins. |
| **Success End Condition** | User will be able make transaction easily. |
| **Failed End Condition** | User cannot be make transaction easily. |
| **Primary Actor:**<br>**Secondary Actor:** | User. |
| **Trigger:** | Request to get reward. |

<table>
<tr><td>Description/ Main Success Scenario:</td><td>
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Log into the system.</td></tr>
<tr><td>2</td><td>Tap get reward.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flaws:</td><td>
<table>
<tr><th>Step</th><th>Branching Action</th></tr>
<tr><td>3</td><td>User has not enough coin .</td></tr>
</table>
</td></tr>
<tr><td>Quality Requirements:</td><td>
<table>
<tr><th>Step</th><th>Requirement</th></tr>
<tr><td>1</td><td>Performance, Speed and Latency.</td></tr>
</table>
</td></tr>
</table>

## 4.3 Activity Diagram

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent.

**Figure 2.1 Activity Diagram for Registration**

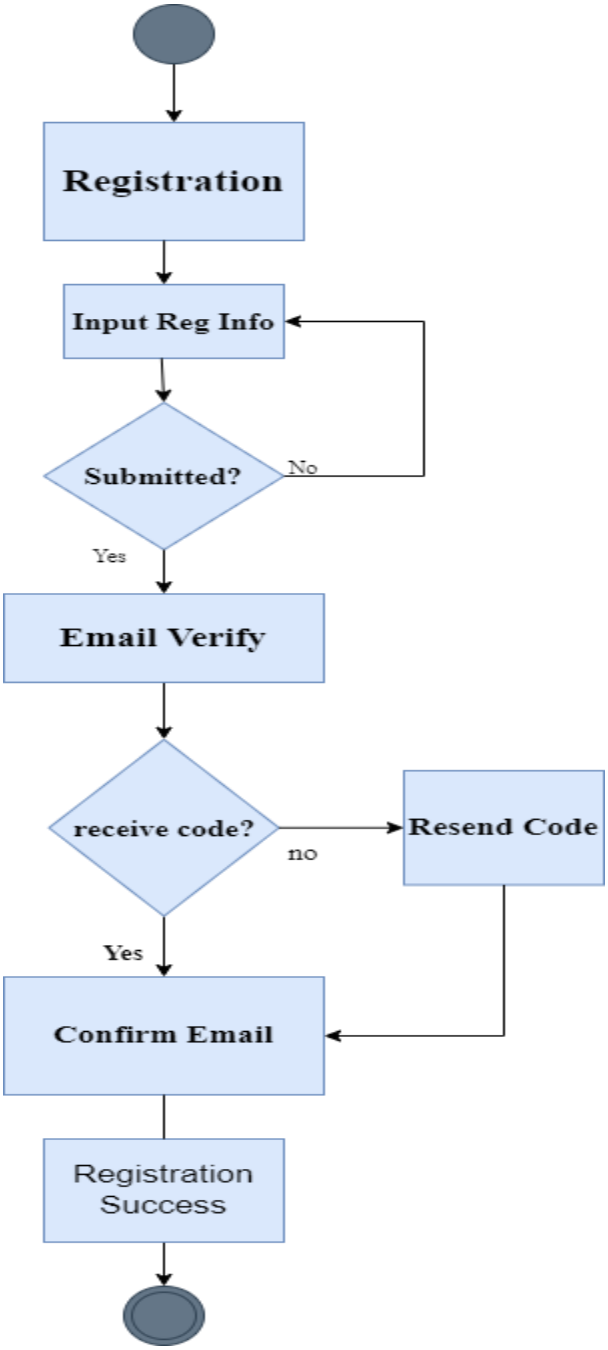



**Figure 2.2 Activity Diagram for Login:**

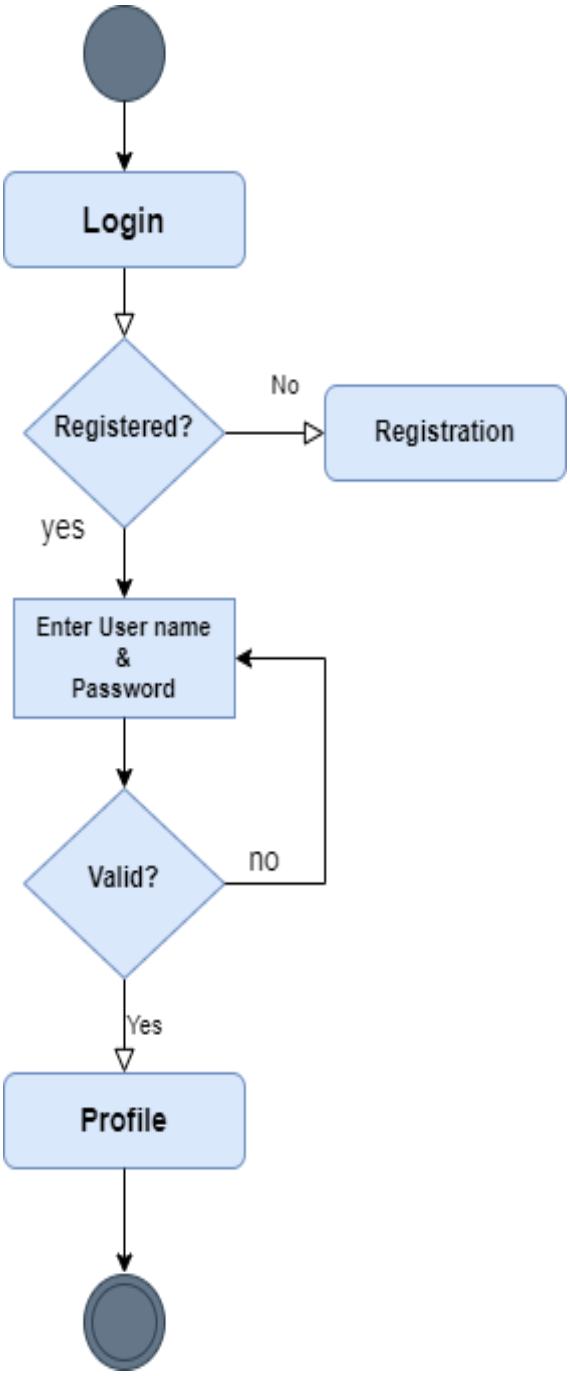



**Figure 2.3 Activity Diagram for Dashboard:**

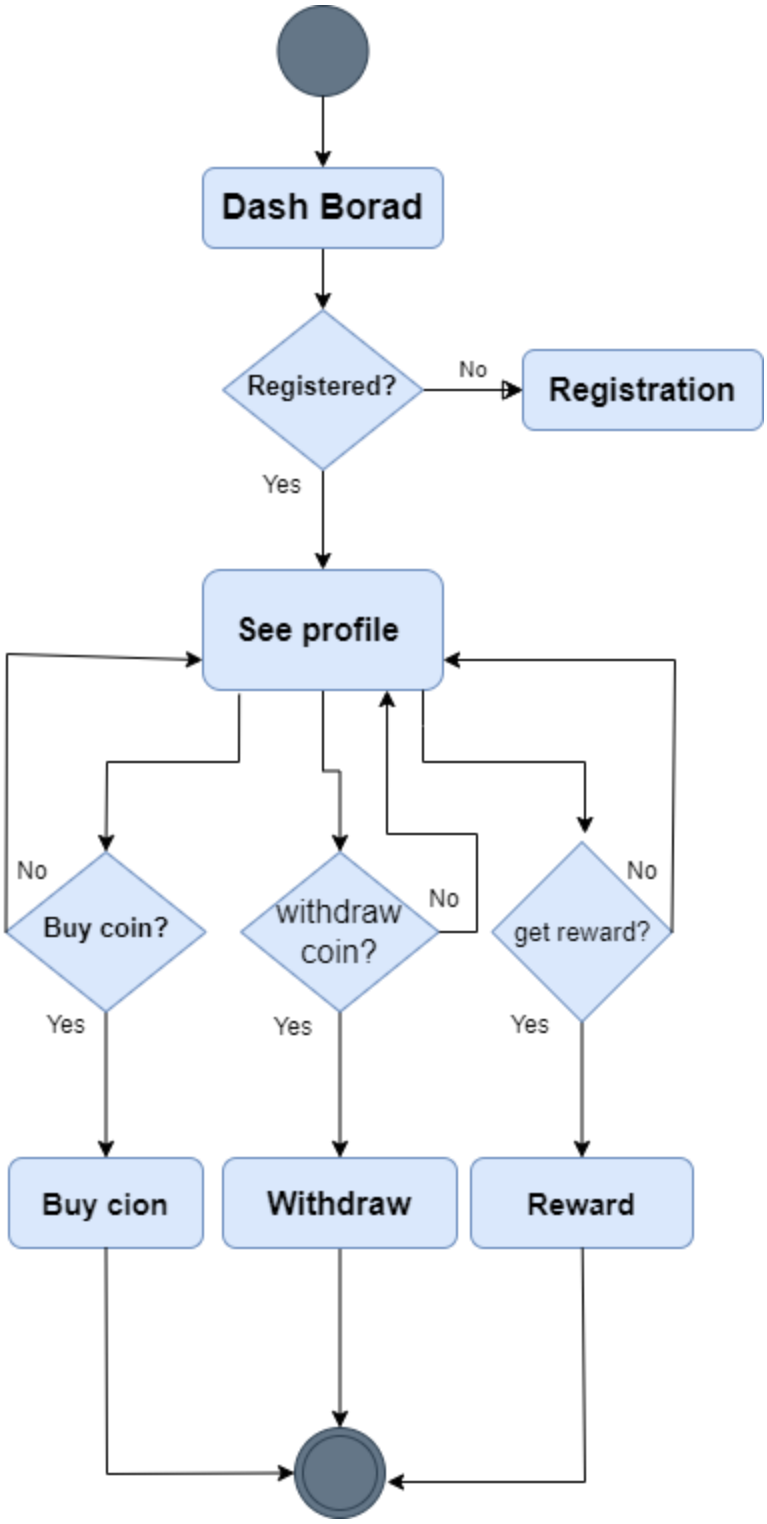



**Figure 2.4 Activity Diagram for Buy Coin:**

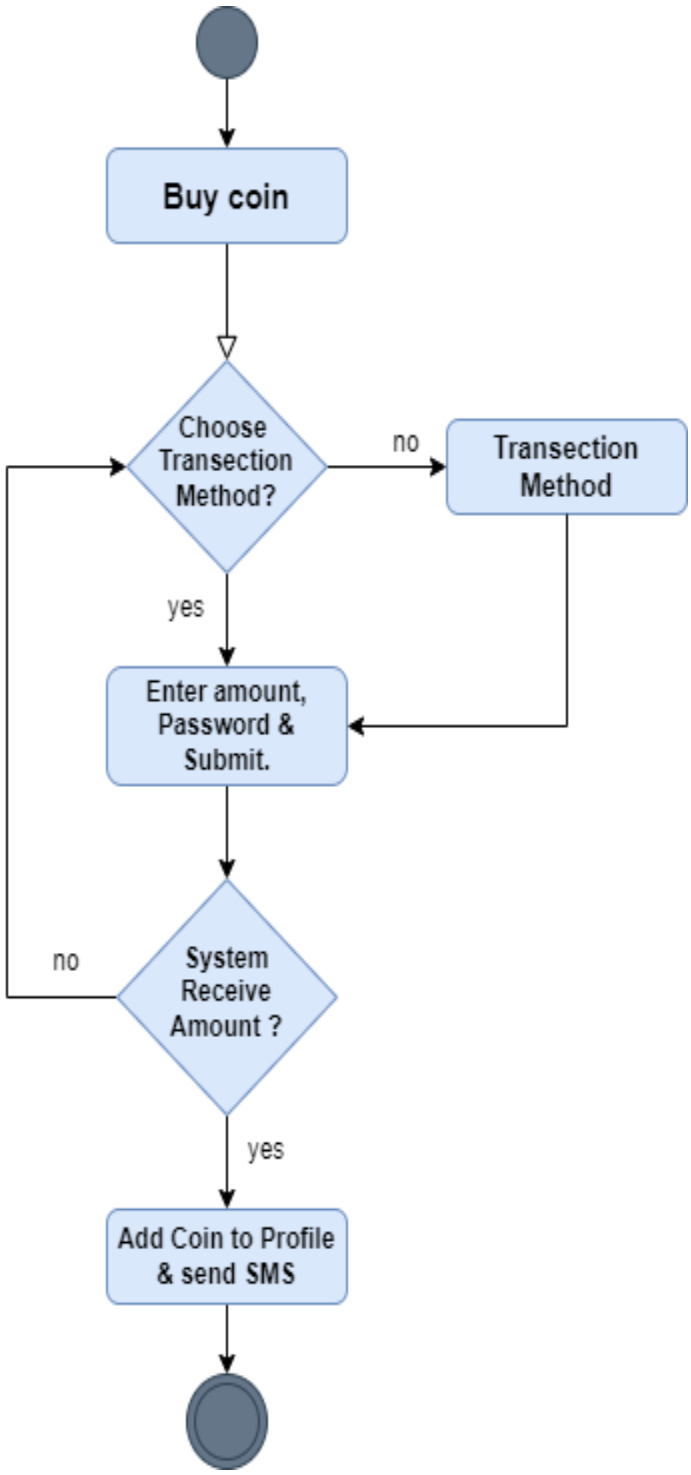



**Figure 2.2 Activity Diagram for Withdraw:**

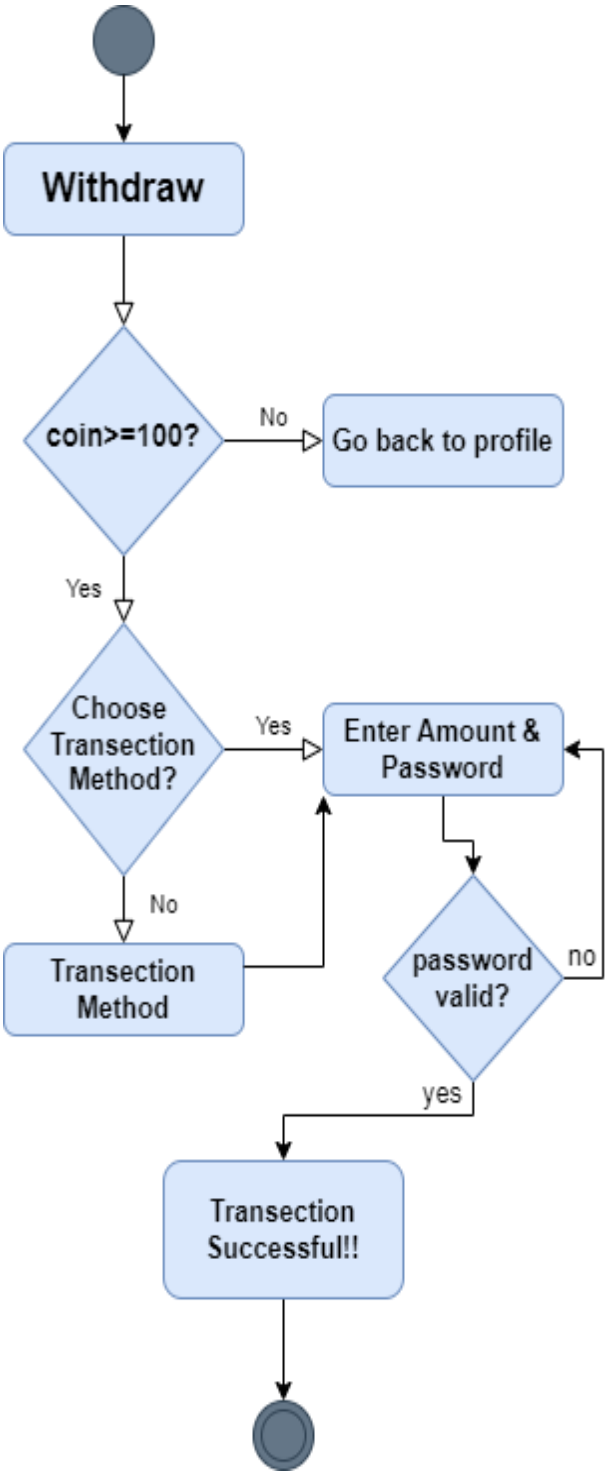



**Figure 2.2 Activity Diagram for Transaction Method:**

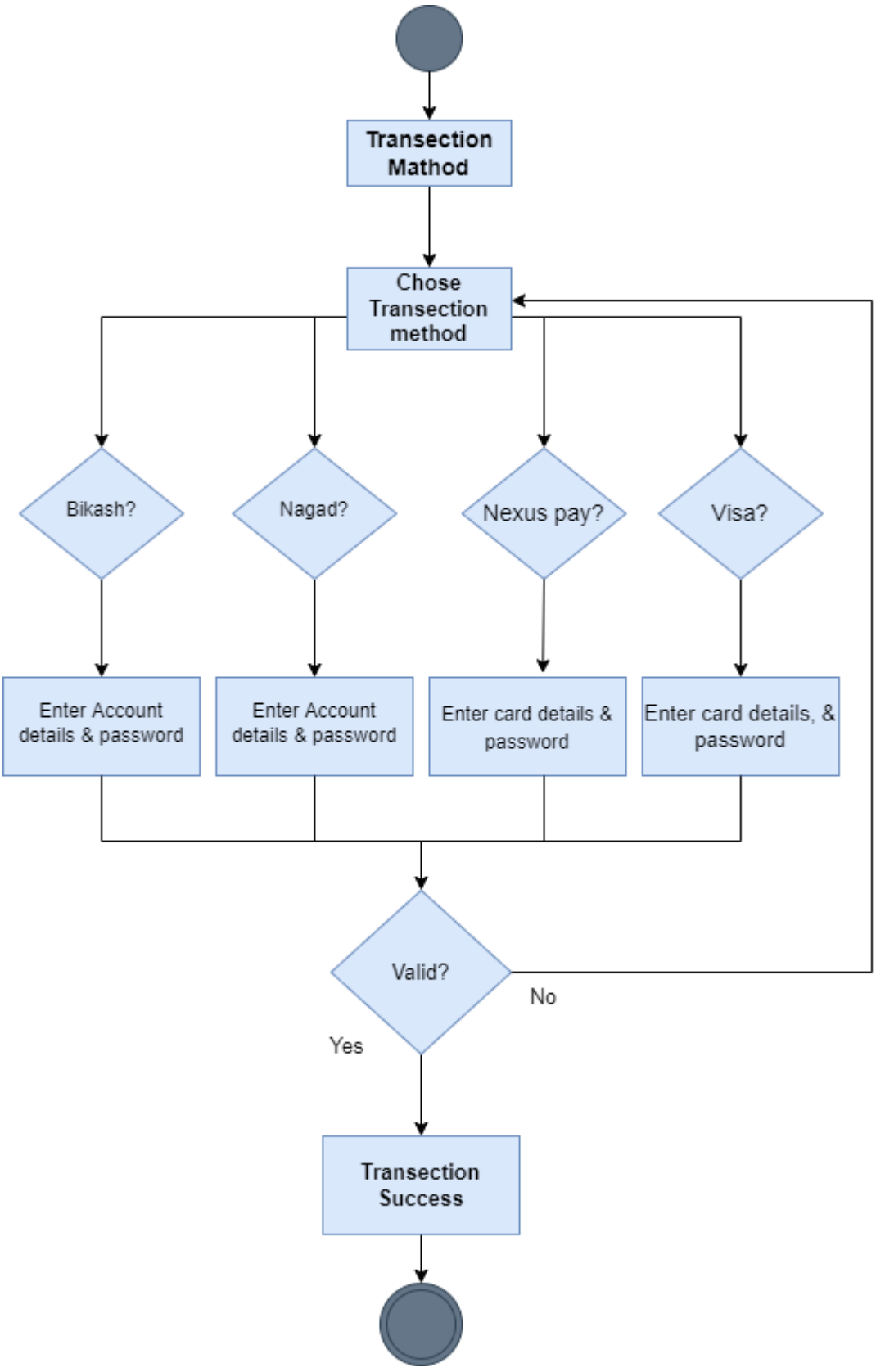
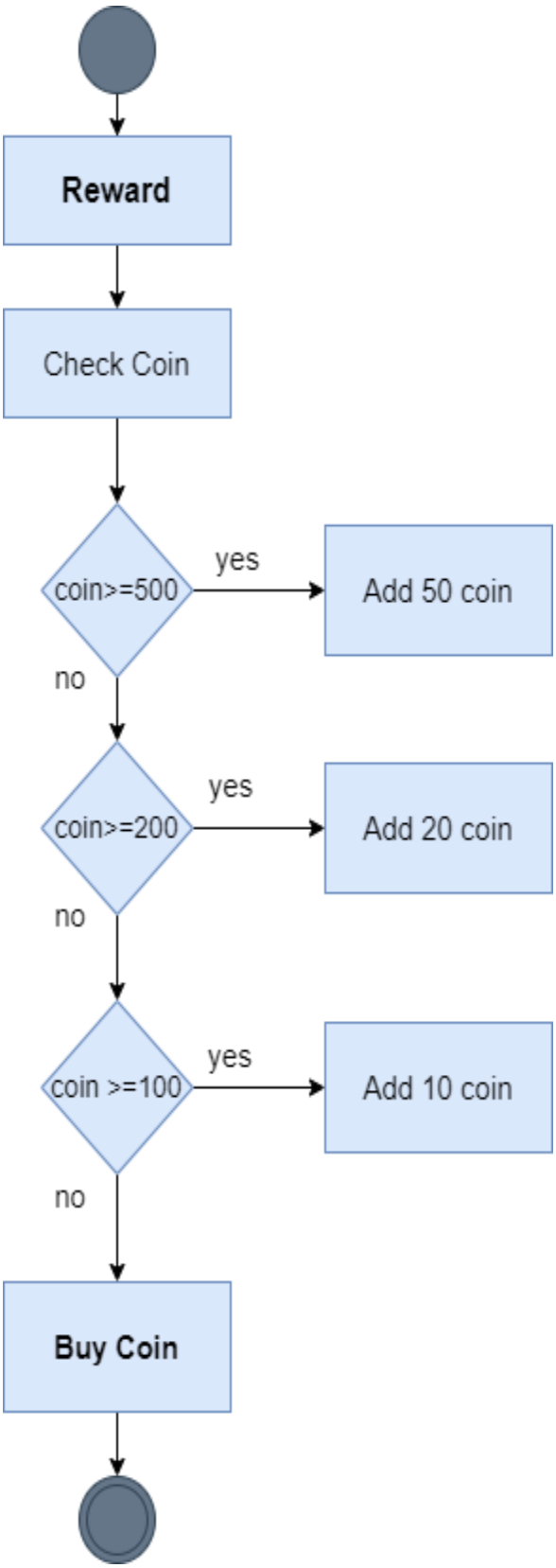



**Figure 2.2 Activity Diagram for Reward:**


Reward
Check Coin
coin>=500
yes
Add 50 coin
no
coin>=200
yes
Add 20 coin
no
coin >=100
yes
Add 10 coin
no
Buy Coin

# CHAPTER 5: SYSTEM DESIGN SPECIFICATION

## 5.1 Sequence Diagram

Sequence diagram show the process in sequential way that it's actor done. Sequence Diagram describe the sequence system to database.

**Figure 3.1 Log in**

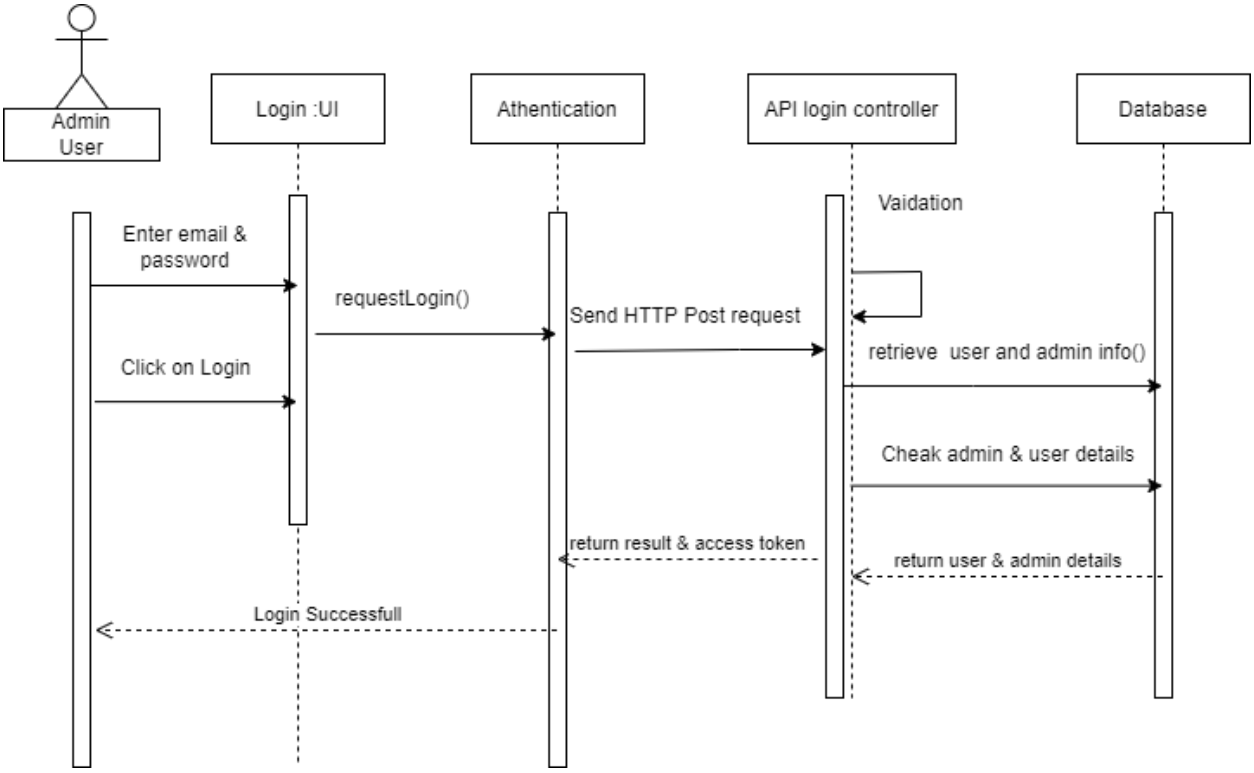


**Figure 3.2 Buy Coin**

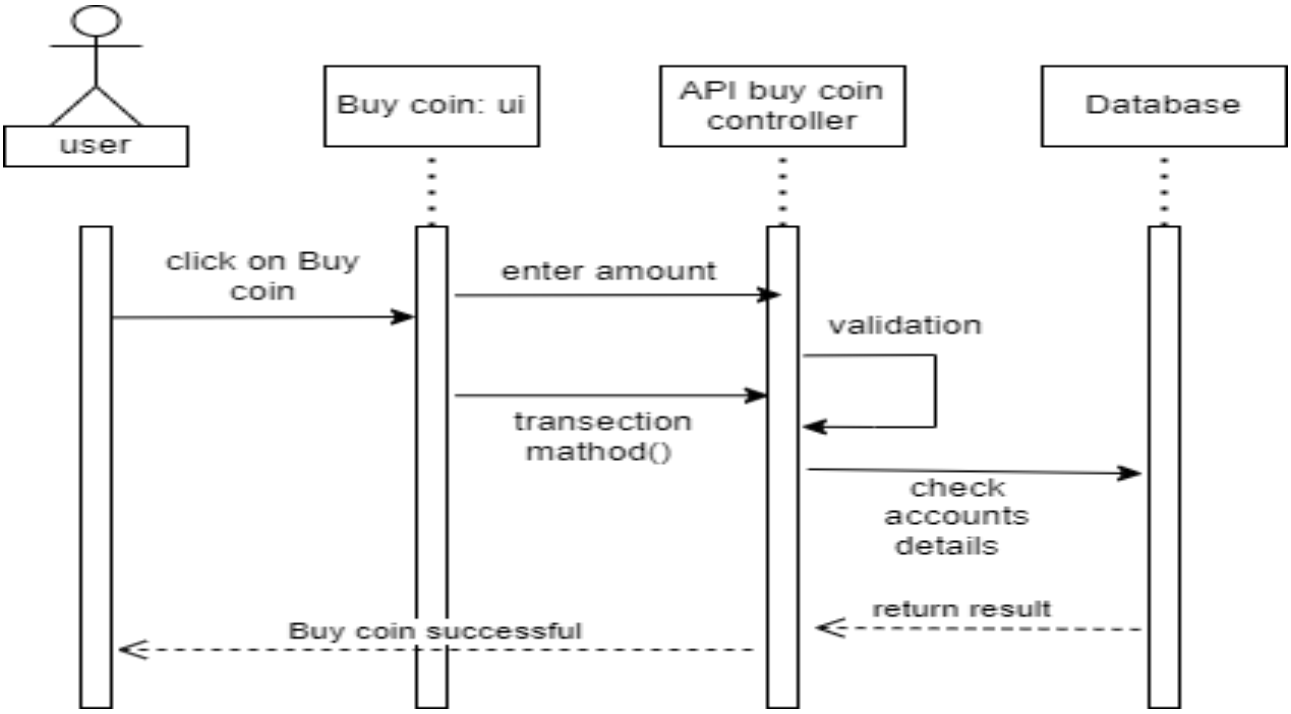
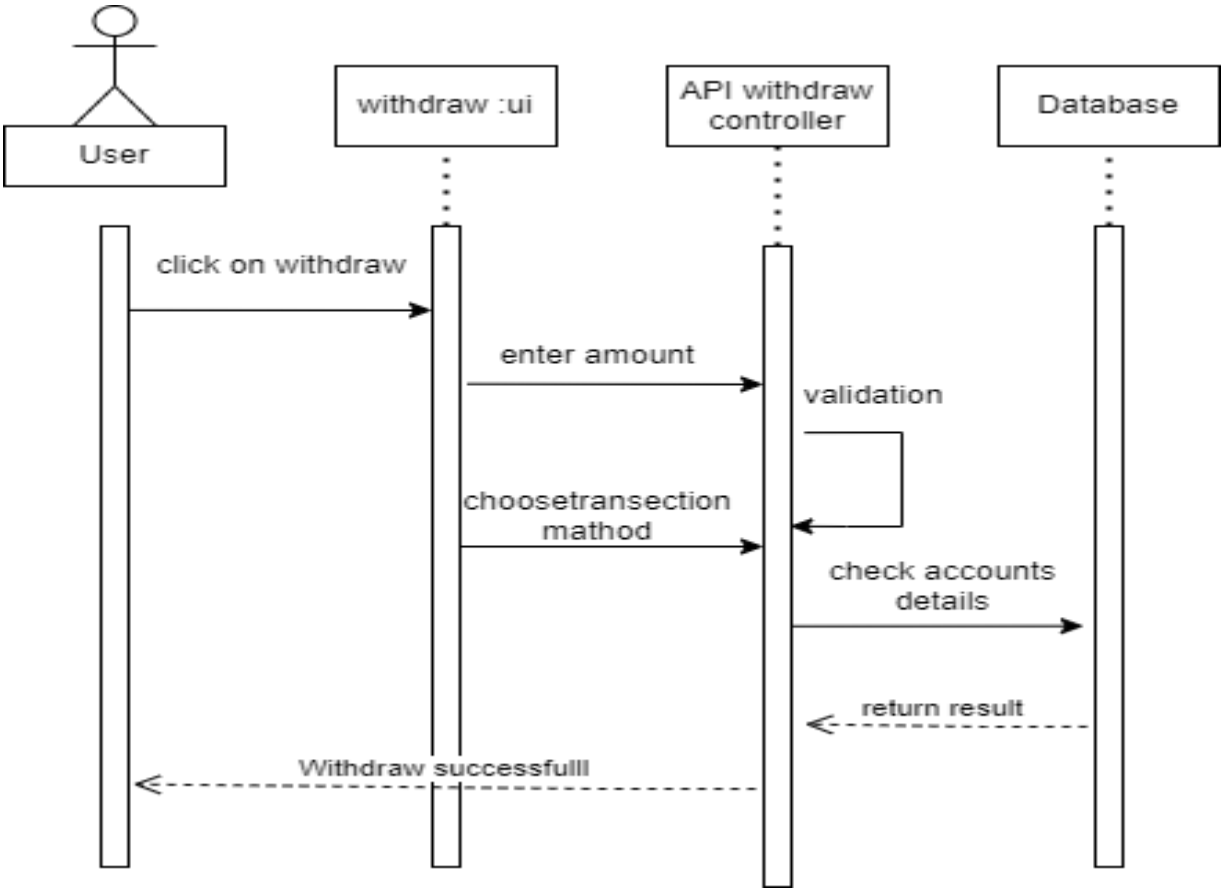
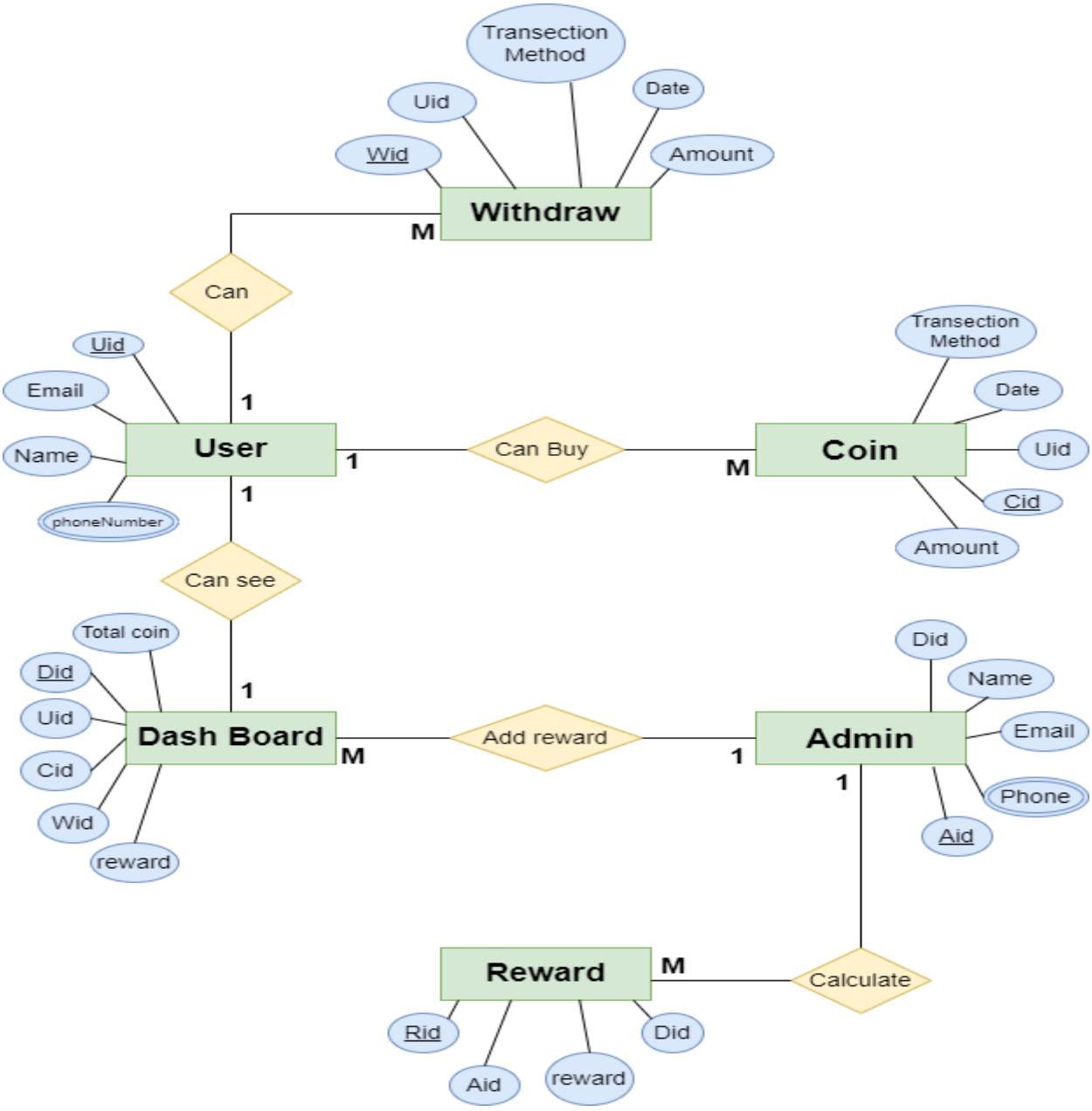


**Figure 3.3 Withdraw Coin**



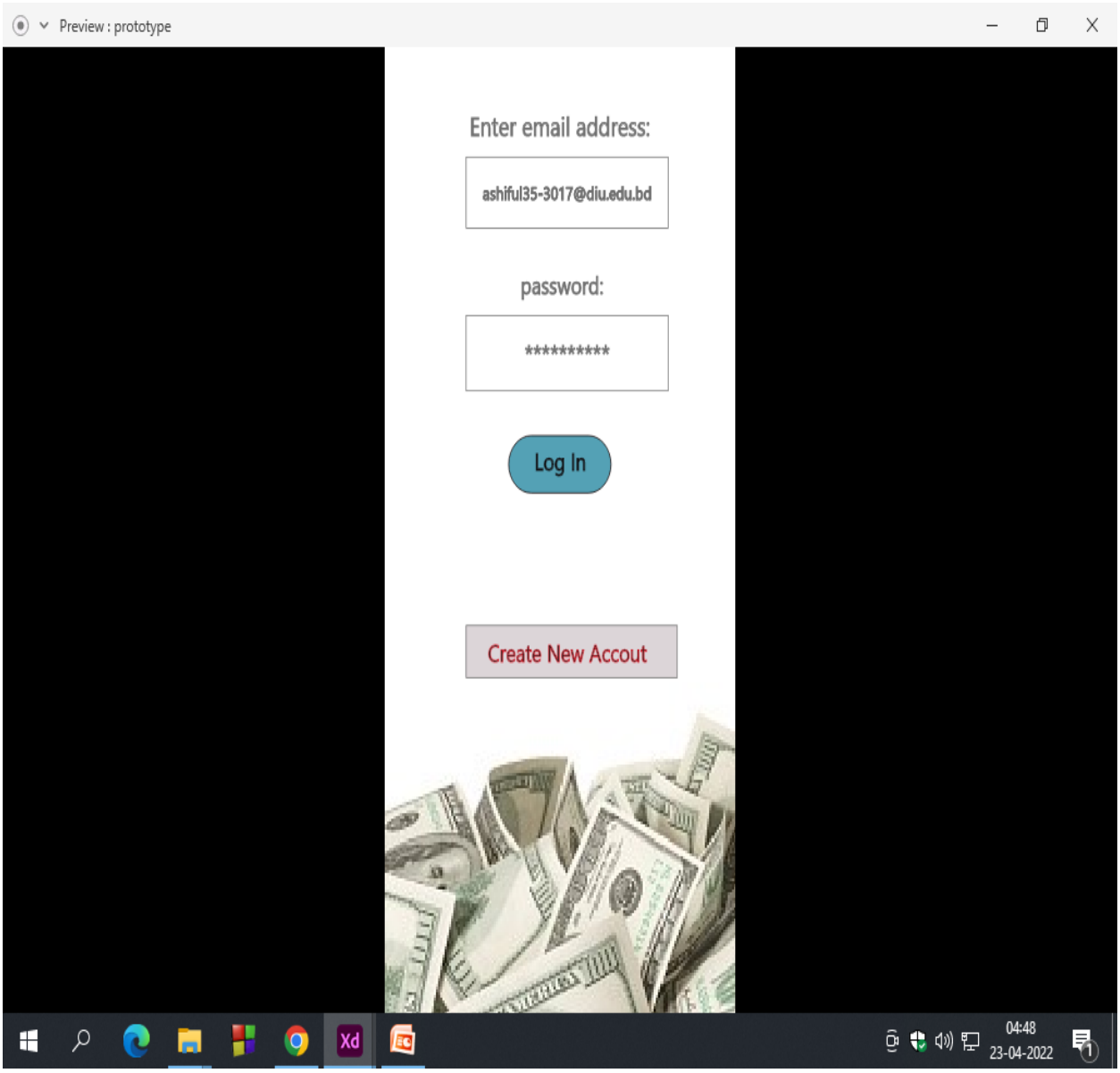
## 5.2 Entity Relationship Diagram:

**Figure 4.0 Entity Relationship Diagram**

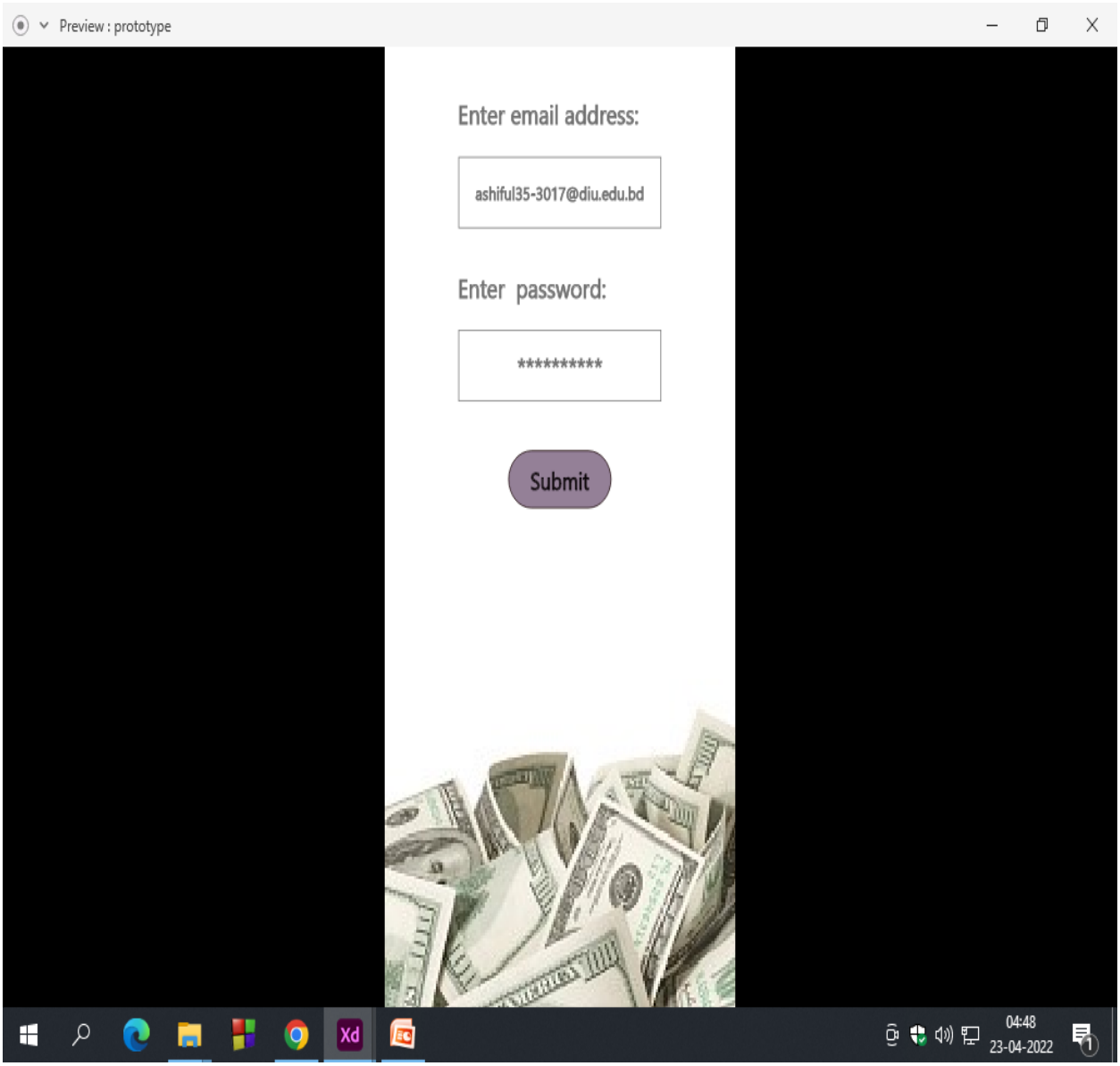



**Shanchay**

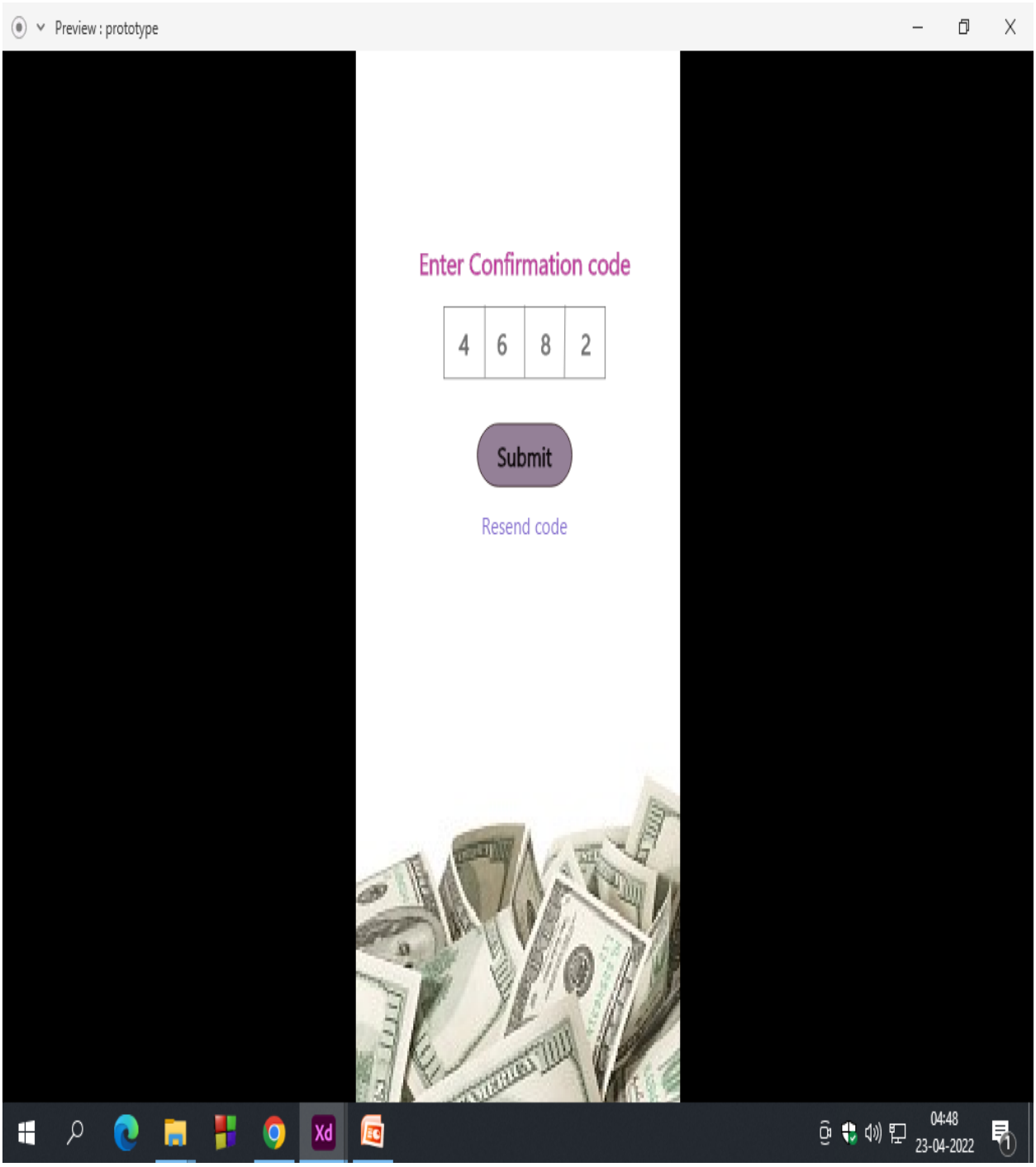
## 5.2Project Manual:



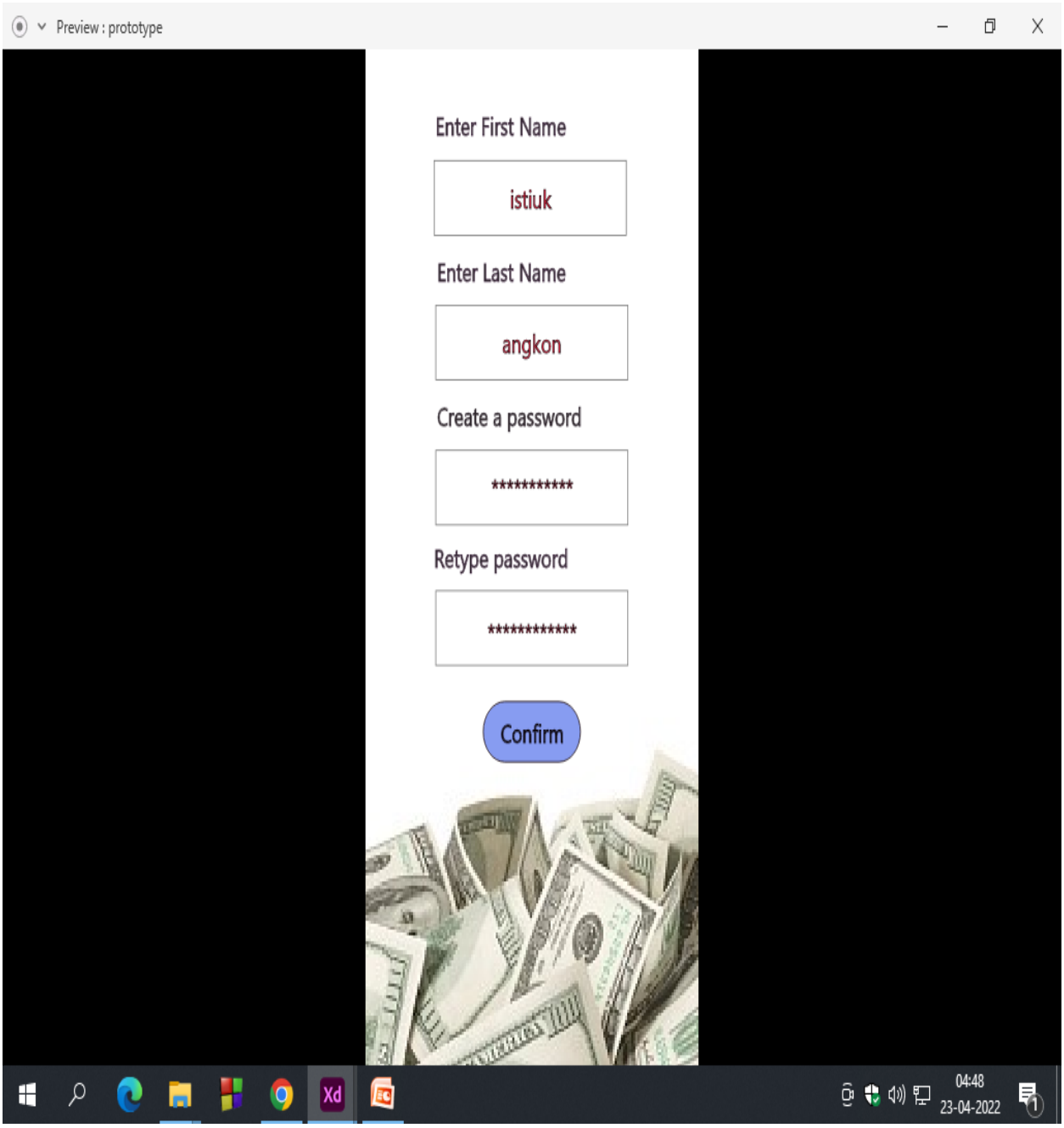

At first you have to log in into the system. If you have not any account you can choose Create new account.



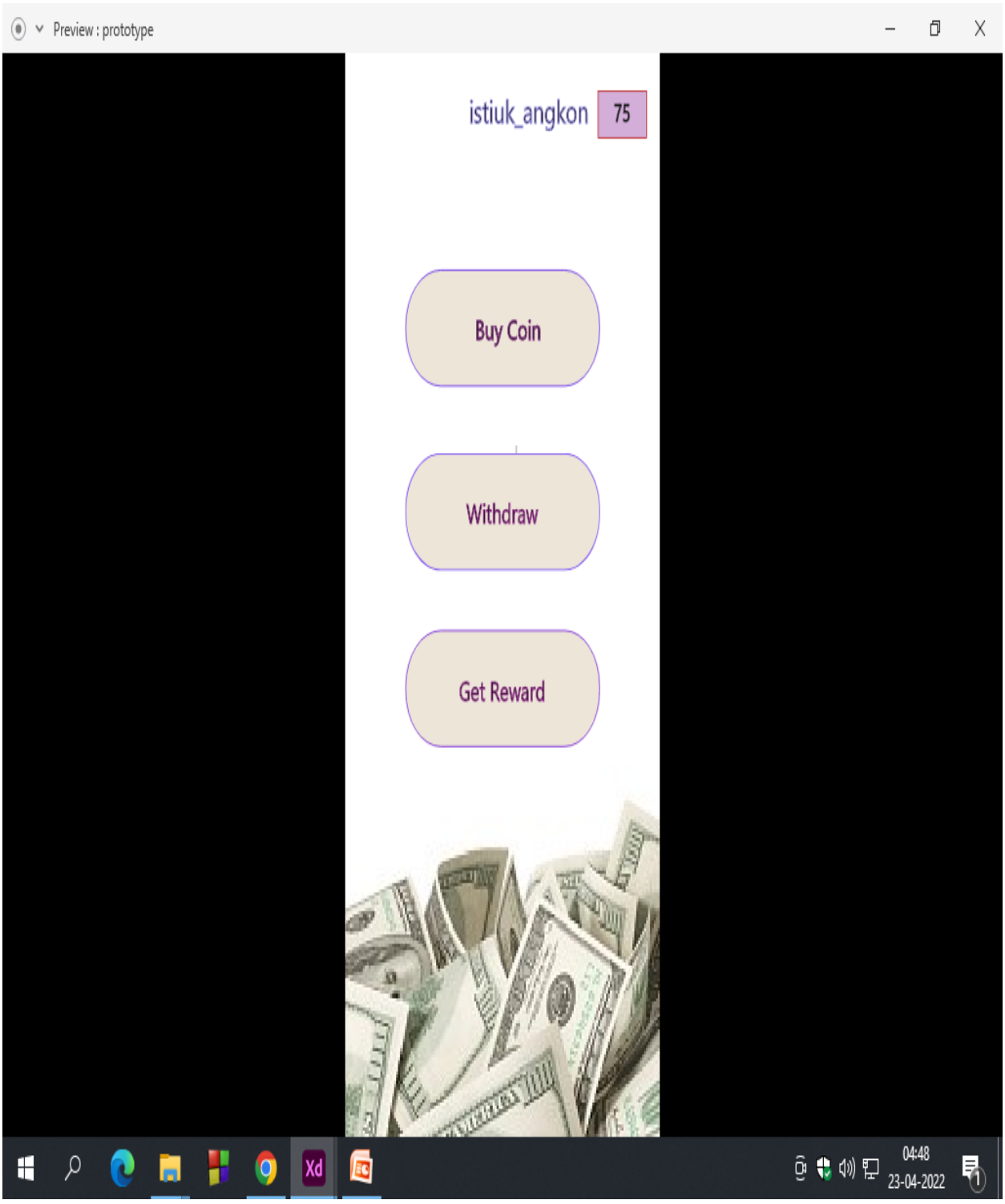

You have to input Email address and password for confirmation and submit.



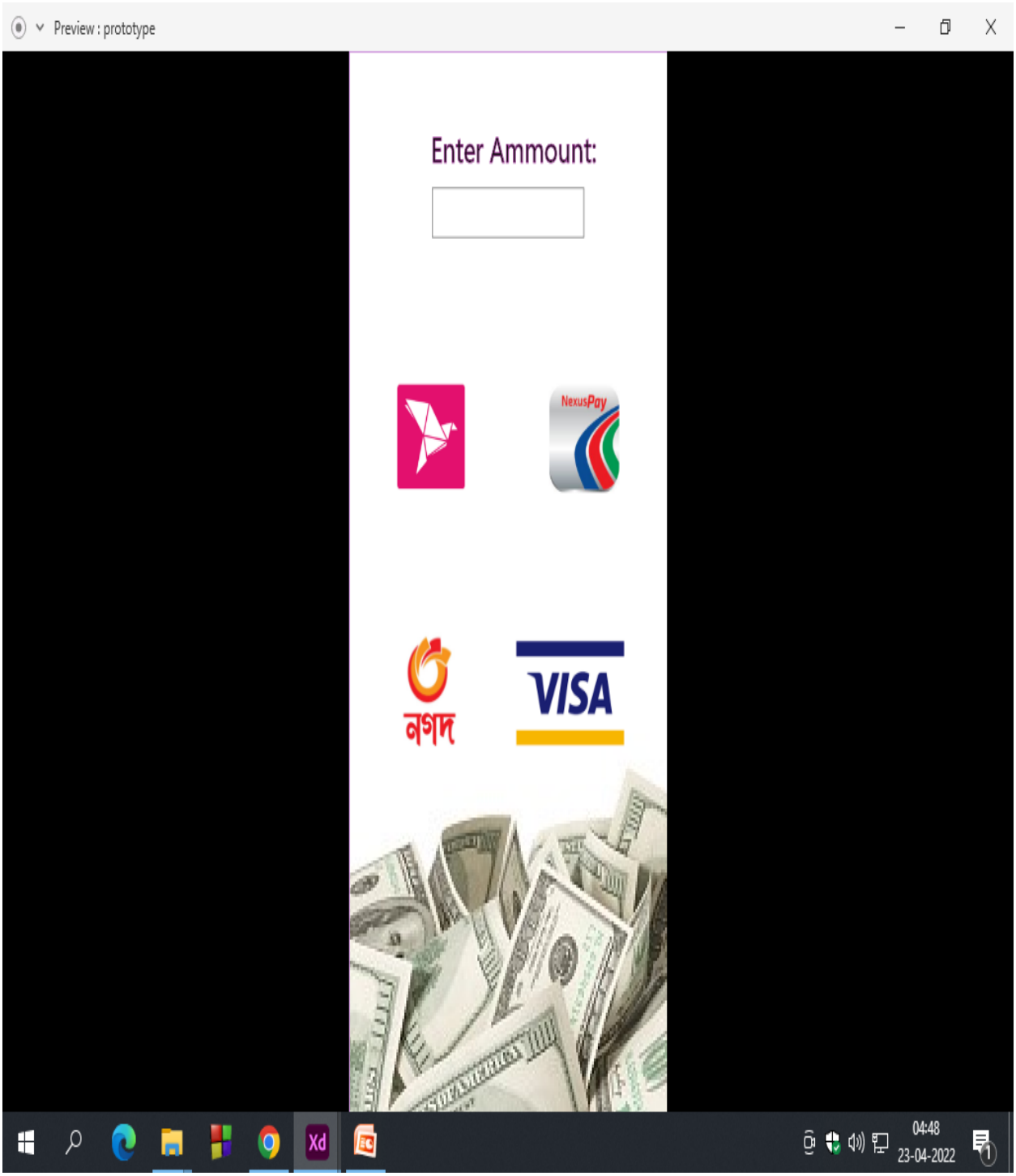

Input confirmation code. If you cannot get any code than click resend.



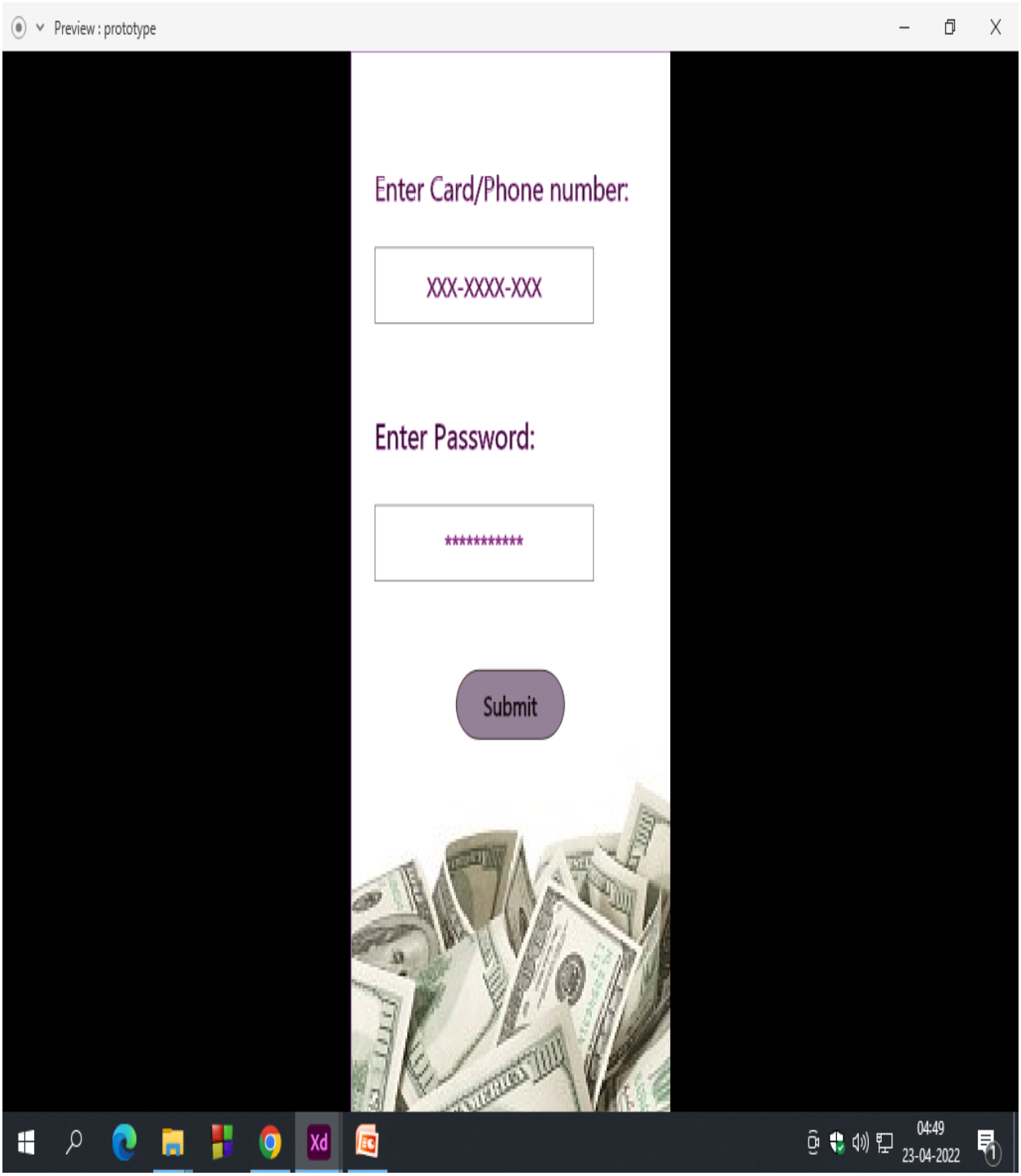

After submit confirmation code you have to input your user name and set a password and click submit button. You successfully done your registration.



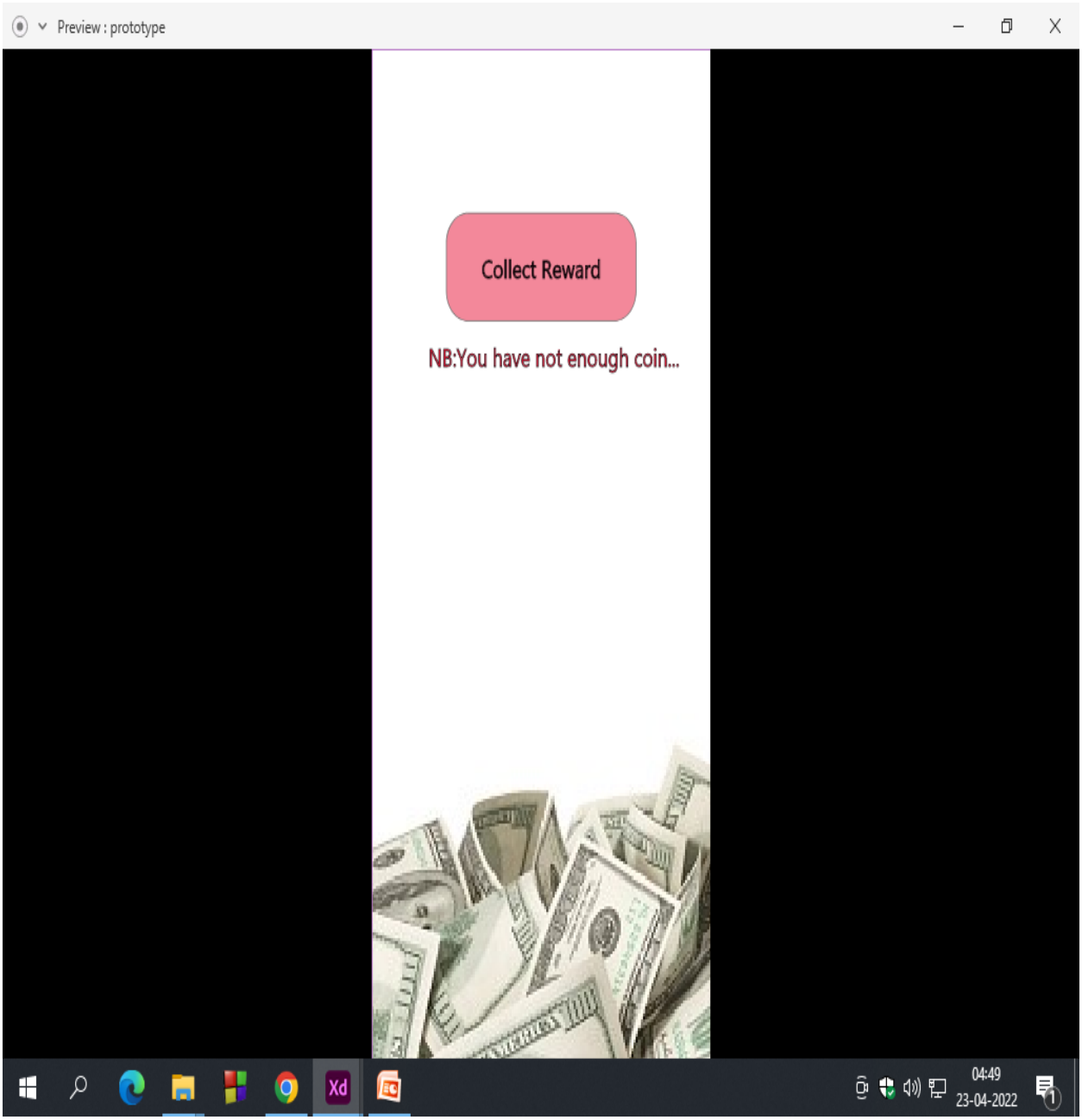

After complete registration and Log in you can see the Dashboard of the system. You can choose Buy Coin for buying coin, you can choose withdraw for withdraw coin, you can choose Reward for get reward.




You have to enter amount and choose any of transaction method between them.




Enter your Accounts details and password and submit. You successfully complete Buy or Withdraw coin.




After clicking reward you can reward if you have required coins for getting reward. Click the get reward button. And back to Dashboard.

# CHAPTER 7: PROJECT SUMMARY

## 7.1 Limitations

The main limitation is system cannot handle million data and signals at time.

## 7.2 Obstacles and Achievement

**Obstacles:**

Learning new technology and new environment is a big issue.
It's very difficult to complete a work within limited time.
To collect requirements is a very tough

**Achievements:**

Successfully built a project.
Learnt a new technology.
Deploy a project is a new experience.
Learned the real-life experience by uploading project on the live server. Know about document and the development process.

## 7.3 Future Work

Though the system was developed as much as needed and its work properly. But I have to add something new features to make the systems fulfilment. The future work will include some major changes

# REFERENCES

1. https://www.w3schools.com/

2. https://laravel.com/

3. https://vuex.vuejs.org/

4. https://vuejs.org/

5. https://router.vuejs.org/

6. https://github.com/

7. https://www.google.com/

# Thank You