B.Sc. Engg. Project

A Project on Banking Management System

Submitted by

Tunazzinur Rahman Kabbo (ID: 19202103268)
Md. Zobayer Hasan Nayem (ID: 19202103274)
Md. Imran Nazir (ID: 19202103248)

Submitted to

Department of Computer Science & Engineering

(In partial fulfillment of the requirements for the degree of

Bachelor of Science in Computer Science & Engineering)

Department of Computer Science & Engineering

Bangladesh University of Business & Technology (BUBT)

Dhaka 1216

April 26,2022

Acknowledgment

"When a task is completed successful" makes everyone happy. But the happiness will be gold without glitter if we didn't state the persons who have supported us to make it a success. Success will be crowned to people who made it a reality but the people whose constant guidance and encouragement made it possible will be crowned first on the eve of success. This acknowledgment transcends the reality of formality when we would like to express deep gratitude and respect to all those people behind the screen who guided, inspired and helped me for the completion of our project work. We consider ourselves lucky enough to get such a good project. This project would add as an asset to our academic profile. We express our gratitude to the help of our supervisor Shadha Anzum Shanto, for his constant supervision, guidance and co-operation throughout the project and for giving constant motivation and valuable help through the project work.We extend our sincere gratitude to our parents who have encouraged us with their blessings to do this project successfully. Finally, we would like to thank to all our friends, all the teaching and non-teaching staff members of the CSE Department, for all the timely help, ideas and encouragement which helped throughout in the completion of project.

Abstract

We hereby declare that the project entitled "Banking Management System" submitted in partial fulfillment by us for the degree B.Sc. Engineering in Computer Science and Engineering in the faculty of Computer Science and Engineering of Bangladesh University of Business and Technology (BUBT) under the guidance of our supervision of Ms. Sadah Anzum Shanto,Lecturer,department of Computer Science and Engineering is our own work and it contains no material which has been accepted for the award to the candidates of any other disciplines expect few references which is taken from various books and authors to enrich our knowledge about the topic of our project.

Banking Management System

We hereby declare that the Project on Banking Management System studies on software development submitted in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering of Bangladesh University of Business and Technology (BUBT),under the guidance of our supervisor Ms. Sadah Anjum Shanto, Lecturer, department of Computer Science and Engineering, is our own work and that it contains no material which has been accepted for the award to the candidate(s) of any other degree or diploma, except where due reference is made in the text of the project. To the best of our knowledge, it contains no materials previously published or written by any other person except where due reference is made in the project.

Tunazzinur Rahman Kabbo	Md. Zobayer Hasan Nayem	Md. Imran Nazir
ID: 19202103268	ID: 19202103274	ID: 19202103248
Intake: 44	Intake: 44	Intake: 44

Copyright

© Copyright by Tunazzinur Rahman Kabbo(ID:19202103268), Md. Zobayer Hasan Nayem(ID: 19202103274), Md. Imran Nazir (ID: 19202103248)

All Right Reserved.

Dedication

Dedicated to our parents, teachers, friends and who loved us for all their love and inspiration.

Certificate

This is to certify that Tunazzinur Rahman Kabbo (ID:19202103268), Md. Zobayer Hasan Nayem (ID: 19202103274)), Md. Imran Nazir (ID: 19202103248),were belong to the department of Computer Science and Engineering, have completed their Project on Banking Management System satisfactorily in partial fulfillment for the requirement of Bachelor of Science in Computer Science and Engineering of Bangladesh University of Business and Technology in the year 2022.

Supervisor
Ms. Sadah Anzum Shanto
Lecturer
Department of Computer Science and Engineering
Bangladesh University of Business and Technology

Approval

A Project on "Banking Management System" is submitted by Tunazzinur Rahman Kabbo ID:19202103268, Md. Zobayer Hasan Nayem(ID: 19202103274)), Md. Imran Nazir (ID: 19202103248) Department of Computer Science and Engineering, Bangladesh University of Business and Technology under the supervision of Ms. Sadah Anzum Shanto, Lecturer, Department of Computer Science and Engineering, has been acknowledged as palatable for the partial fulfillment of the necessity for the degree of Bachelor of Science in Computer Science and Engineering and endorsed as to its style and substance.

Supervisor
Ms. Sadah Anzum Shanto
Lecturer
Department of Computer Science and Engineering
Bangladesh University of Business and Technology

Chairman
Md. Saifur Rahman
Assistant Professor & Chairman
Department of Computer Science and Engineering
Bangladesh University of Business and Technology

Acronyms List

BMS Banking Management System

DBMS Database Management System

API Application Programming Interface

UI User Interface

ERD Entity Relationship Diagram

SQL Structure Query Language

GUI Graphical User Interface

SDLC Software Development Life Cycle

UML Unified Modeling Language

DFD Data Flow Diagram

Contents

Acknowledgment **i**

Abstract **ii**

Declaration **iii**

Copyright **iv**

Dedication **v**

Certificate **vi**

Acronyms List **viii**

List of Figures **viii**

List of Tables **viii**

1 Introduction **1**

1.1 Importance of the system 1

1.2 Major functions of bank management system 2

1.3 Objectives 3

1.4 Audience 3

1.5 Reason of The System 3

1.6 Scope of the system 4

1.7 Motivation 5

1.8 System Development Life Cycle 5

1.9 SDLC of our system . . . 7
1.9.1 Identifying problems, opportunities and objectives . . . 7
1.9.2 Analyzing system needs . . . 8
1.9.3 Designing the recommended system . . . 8
1.9.4 Developing and documenting software . . . 9
1.9.5 Testing and maintaining the system . . . 10
1.9.6 Implementing and evaluating the system . . . 12

2 Methodology **13**
2.1 Feasibility Analysis . . . 13
2.1.1 Technical Feasibility . . . 14
2.1.2 Operational Feasibility . . . 15
2.1.3 Economical Feasibility . . . 16
2.2 Context Level Diagram . . . 16
2.2.1 Context Level Diagram for our System . . . 17
2.3 Human Information Requirements . . . 19
2.4 UML Diagrams . . . 20
2.4.1 Sequence Diagram for Banking Management System . . . 20
2.4.2 Activity Diagram for User all Process . . . 21
2.4.3 Use Case Diagram for Banking Management System . . . 22
2.4.4 Sequence Diagram for Banking Management System . . . 23
2.4.5 Class Diagram for Banking Management System . . . 24

3 Implementation **25**
3.1 Software Tools . . . 25
3.1.1 Xampp . . . 25
3.1.2 Visual Studio Community . . . 26
3.1.3 MySQL Connector . . . 26
3.1.4 C# programming Language . . . 27
3.1.5 .Net Frame Work . . . 27
3.2 Software Review . . . 28

3.2.1 User Registration 28
3.2.2 User login 28
3.2.3 User Dashboard 29
3.2.4 User Withdraw 29
3.2.5 User Transfer 29
3.2.6 User Bill pay 30
3.2.7 User Payment 30
3.2.8 User Crypt Currency 30
3.2.9 User Add Money 31
3.2.10 User Donation 31
3.2.11 User Profile 31
3.2.12 Admin Login 32
3.2.13 Admin Dashboard 32

4 Conclusion **33**

4.1 Advantages of Our System 33
4.2 Limitation 35
4.3 Future Work 36
4.4 Developers contribution final words 36
4.4.1 Reference 37

List of Figures

1.1 Seven Phases of SDLC Model 6

2.1 Symbols used in Context Diagram 17
2.2 Context Level Data Flow Diagram 18
2.3 Sequence Diagram for Appointment Confirmation 20
2.4 Activity Diagram for User all Process 21
2.5 Usecase Diagram for Banking Management System 22
2.6 Sequence Diagram for Banking Management System 23
2.7 Class Diagram for Banking Management System 24

3.1 Xampp Interface 26

Chapter 1

Introduction

Banking management system allows consumers to perform quick self-service transactions. It can be used for day to day transactions. Moreover, depositing, withdrawing or even to know the information of an account like the balance amount, etc.

1.1 Importance of the system

To serve the banking industry, we introduced this system.If you say why?

The question arises why banking system was needed. Because, Banking made transactions simple and safer.Not only that, there are many more functions available on this system which will be discussed beneath. Bank relationship management provides processes and technology that help a company track and manage all their bank relationships by: Providing a single view of all accounts and activities with a bank, worldwide.

1.2 Major functions of bank management system

- Accepting deposits.
- Lending loans and advances.
- Transfer of funds.
- Login.
- Validation.
- Get balance information.
- Withdrawal of money.
- Customer info.
- Digital currency.
- Credit card services.
- Administrative Control

1.3 Objectives

Bank management governs various concerns associated with bank in order to maximize profits. The concerns broadly include liquidity management, asset management, liability management and capital management. Out motive is to provide digitalize banking system with crypt and many more things by understanding consumer needs. As the world changing rapidly with technological things, so generation need a better system with technological benefits.

1.4 Audience

Our main audiences are –

- Employee
- Business Man
- House-wife
- Student
- Non-Employee

1.5 Reason of The System

After using normal banking system, we faced many issues and there are lackings available. For that reason, we couldn't take the full benefit of the system. We friends get on a meeting discussed about it. Moreover that we had gone to public to know their opinion and confirmed that they are not happy with the current system. So, we made a decision to redesign the system with thinking about user needs.

- Deposit made easy.
- Withdrawals with bank and MFS.
- Transfer of funds.

- Login.
- Validation.
- Get balance information.
- Withdrawal of money.
- Customer info.
- Digital currency.
- Credit card services.
- Administrative Control.

1.6 Scope of the system

- Through this project we can manage crypt-currency easily with a real-time crypt exchange rates.
- The users can avail crypt wallet when they register in the program.
- They can transfer a valid transaction easily with a secure method.
- Users can easily update their credentials and user information easily.
- Users can avail two-way authentication as a extra layer of security.
- Users can take fast loan and can pay the amount through EMI.
- For payment, users can easily go through a verification system. In which the user's information will be secured.
- Users can easily add money from Mobile Banking Apps or from any Cards.

1.7 Motivation

- Today our world is in tune with modernity and if we want to keep pace with this we have to be with it. Now we have this digital system so that people can carry their money without any worries.
- We have taken this kind of project because many people open bank accounts without understanding much. They face a lot of difficulties with money, so they often give money to another account.
- In addition, it is taken by us so that any user can use this application very easily.
- A unique advantage here is that those who want to work with digital money have done it very nicely in this project. As a result, they can easily buy this money with money. And they can easily see the value of it.
- We are adding another feature here that allows users to easily add money from their digital wallets, such as bKash, Cash, Way, Rocket, and MasterCard, which is not the case with many banks.
- Apart from this, students in our country cannot open an account without their NID cards. As a result, they cannot accumulate their savings. But in our banking service, a student can open an account with their ID card. And he can accept all kinds of services.
- Many banks charge a fee to pay any bill, but we do not capture any charges in this system. They can exchange money completely free of charge.

1.8 System Development Life Cycle

System Development Life Cycle (SDLC) is a process for planning, creating, testing, and deploying an information system. A life cycle model is a diagram that depicts all of the steps involved in moving a software product through its life cycle stages. It also describes the framework in which these methods will be implemented. It has seven steps consisting of problem identification, determining human information, analyzing system needs, designing, developing and testing the system and evaluating it.

For each project, the developer must select an appropriate life cycle model and then monitor it. Without the adoption of a detailed life cycle model, the development of a system would not be systematic and disciplined. When working on a system, members of the team must be aware of when and what they should do. Otherwise, anarchy and project failure would result. Each phase's entry and exit criteria are specified in a software life cycle model. Only if the stage-entry criteria for a phase have been met can it begin. The entry and departure criteria for each stage cannot be identified without a software life cycle model. It's difficult for systems development managers to keep track of the project's development without software life cycle models.

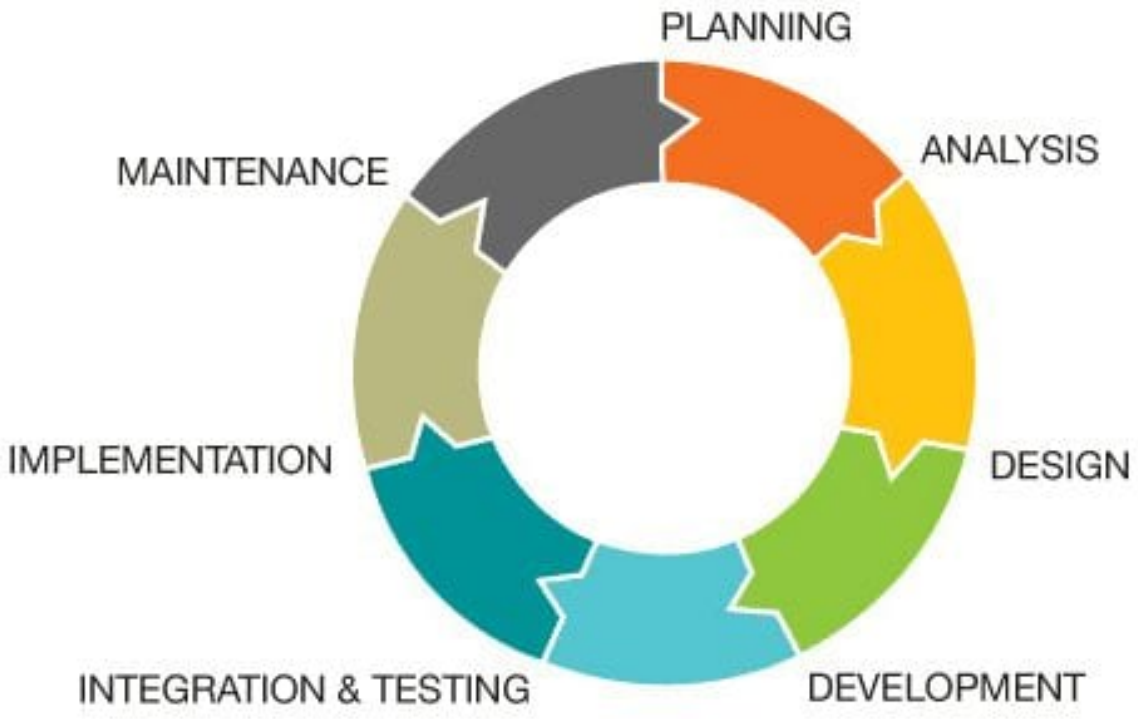


Figure 1.1: Seven Phases of SDLC Model

SDLC Cycle represents the process of developing software. SDLC framework includes the following phases:

Phase 1: Identifying problems, opportunities and objectives

In this phase, the problems are identified like what kind of operations will be done, how it will be done, its opportunities and objectives, etc.

Phase 2: Determining human information requirement

The analyst next moves on to determining the human needs of the users involved, using in-

terviewing, sampling, and investigating hard data, as well as questionnaires as well as less obtrusive methods.

Phase 3: Analyzing system needs

During this phase, the systems analyst prepares a systems proposal that summarizes what has been learned about present systems' users, usability, and utility; gives cost-benefit analyses of alternatives; and makes suggestions on what should be done (if anything).

Phase 4: Designing the recommended system

The systems analyst uses the information collected earlier to accomplish the logical design of the information system by using techniques of good form and Web page or screen design.

Phase 5: Developing and documenting software

The analyst collaborates with programmers to create any custom software that is required. During this phase, the analyst collaborates with users to create useful software documentation.

Phase 6: Testing and maintaining the system

Before the information system can be used, it must be tested. It is much less costly to catch problems before the system is signed over to users.

Phase 7: Implementing and evaluating the system

In this last phase of systems development, the analyst helps implement the information system. Evaluation is included as part of this final phase of the SDLC.

1.9 SDLC of our system

Our system is Banking Management System. Now, the SDLC of our system –

1.9.1 Identifying problems, opportunities and objectives

Keeping track the all activities and their record on paper and error. It is also very efficient and a time consuming process of observing continuous increase in number of client visiting the bank. Recording and maintaining all the client record highly unreliable, inefficient.The problem facing the current manual system is difficult to update and maintain, inconsistent data, insecurity, difficult to impose different various data files and difficult to data backup. It is against this backdrop that automated database system I developed to addressed the

problem.Keeping track the all activities and their record on paper and error. It is also very efficient and a time consuming process of observing continuous increase in number of client visiting the bank.

Benefits of Proposed System:

The Bank Management System is an application for maintaining a person's account in a bank. The system provides the access to the costumer to create an account, deposit/withdraw the cash from his account, also to convert currency. The following documentation provides the specification of the system.

1.9.2 Analyzing system needs

- We will create a method in the system through which this issue will go directly to the user's OTP.
- We need to create a database here that will keep a record of how many transactions we have done and the process it has.
- Buy Crypto-Currency, there will be a wallet, there we will have all the transaction records, and this record will be in the database.
- To see all these types of work, we need to make two options in admin: wallet and email verification.
- Perhaps we can solve the problem by creating this panel.

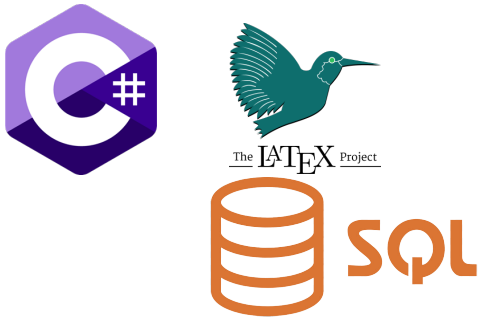
1.9.3 Designing the recommended system

- ER Diagram
- DFD
- Activity Diagram
- Class Diagram
- Usecase Diagram

- Sequence Diagram
- Class Diagram

1.9.4 Developing and documenting software

We have used **C#** for the coding part and **LaTeX** for the documenting part. To handle the database we used **SQL**.



C#

C# is an object-oriented, component-oriented programming language. C# provides language constructs to directly support these concepts, making C# a natural language in which to create and use software components. Since its origin, C# has added features to support new workloads and emerging software design practices. At its core, C# is an object-oriented language. We can define types and their behavior.

LaTeX

LaTeX is a document preparation system for high-quality typesetting. It is most often used for medium-to-large technical or scientific documents but it can be used for almost any form of publishing. LaTeX is not a word processor! Instead, LaTeX encourages authors not to worry too much about the appearance of their documents but to concentrate on getting the right content.

SQL

SQL stands for Structured Query Language. SQL is used to communicate with a database. According to ANSI, it is the standard language for relational database management systems. SQL statements are used to perform tasks such as update data on a database, or retrieve data from a database.

1.9.5 Testing and maintaining the system

Not only does it help to prevent mistakes and defects to the software being developed, it also ensures that customers receive the highest quality products, services and solutions.

Software Quality Assurance

Quality is defined as the product or services that should be ”fit for use and purpose.” Quality is all about meeting the needs and expectations of customers concerning functionality, design, reliability, durability, and price of the product.

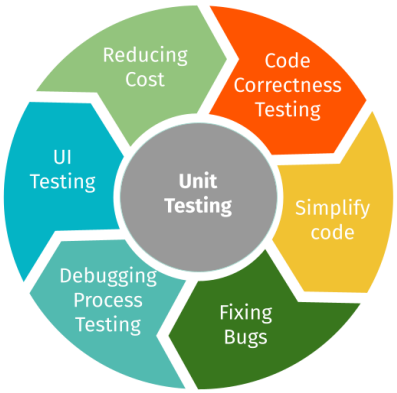
Attributes of Software Quality Assurance

1. Functionality

2. Reliability
3. Usability
4. Efficiency
5. Maintainability
6. Portability

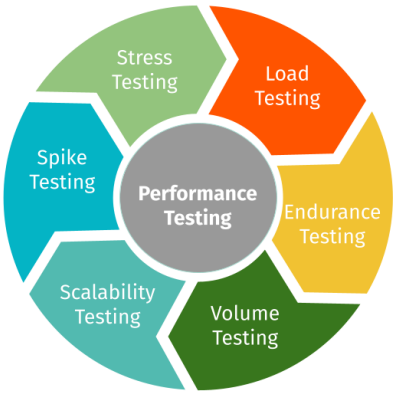
Unit testing

Unit testing focuses on testing individual parts/units of a system. Any function, procedure, method, or module can be a unit to undergo unit testing for determining its correctness and expected behavior. Unit testing is the first testing that we performed during the development phase.



Performance testing

Performance testing is carried out to determine the speed, stability, and scalability of our system. The overall goal of this testing is to check the performance of system by ensuring CPU utilization, database handling, page load speed, server resource utilization, etc.



After that our system goes under maintenance which covers a wide variety of activities, including removing program and design errors, updating documentation and test data and updating user support.

1.9.6 Implementing and evaluating the system

Implementation includes user notification, user training, installation of hardware, installation of software onto production computers, and integration of the system into daily work processes. This phase involves training users to handle the system. The analyst needs to plan for a smooth conversion from the old system to the new one. Evaluation is included as part of this final phase of the SDLC mostly for the sake of discussion. Actually, evaluation takes place during every phase. A key criterion that must be satisfied is whether the intended users are indeed using the system.

Chapter 2

Methodology

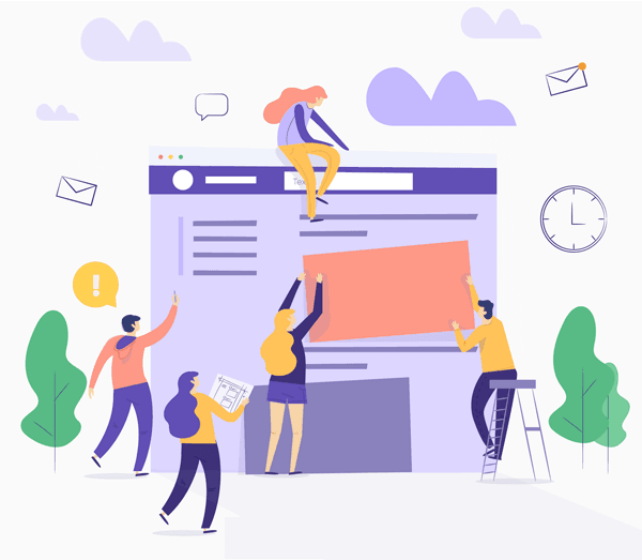
2.1 Feasibility Analysis

A feasibility study is an analysis that considers all of a project's relevant factors—including economic, technical, legal, and scheduling considerations to ascertain the likelihood of completing the project successfully.

2.1.1 Technical Feasibility

The users of Banking Management System (BMS) are used to with windows operating system. So, we are launching our software in the availability and mostly used technology. It's easy for them to operate the software and complete their needs. Moreover, they have physical access of the banks and ATM's but to upgrade the present banking system, we have introduced the BMS software. As our users have internet access, they can easily sign up and login via email. By this, they can easily interface our software and explore all the functionalities of BMS. So, our solution is technically feasible.

2.1.2 Operational Feasibility



Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

Functionalities we've introduced are transferring money from one to account to other, withdrawing, bill paying, donation etc. Also, we're trying to implement crypto currency exchange services. If we can properly implement as mentioned things, then we will get an outcome from this. So, when our software will be put it service, it will be able to operate easily.

So, our BMS will be operationally feasible.

2.1.3 Economical Feasibility

Economic feasibility is a kind of cost-benefit analysis of the examined project. So in our BMS system we've picked a budget to complete this whole project. The budget includes separate cost limits for different activities. We've to count money for system analyst time. Also, when we studied the system what things will be needed for this project, we had to provide time and energy on to it. Then we have buy hardware related things to run the software, a stable internet connection. Moreover, this, we have to use software's like Visual Studio, for analysis used browsers etc. For all of this budget and time, we've to count x amount of money. And they are in to our budget. So, our BMS will be economically feasible.

2.2 Context Level Diagram

A context diagram is a visual representation of the relationship between data and processes. This diagram has three main components which include external entities, system processes, and data flows.

1. **External Entity-** an element in the system diagram that inputs data into the information system and retrieves processed data.

2. **Process-** refers to the entire process of the system. This is responsible for processing and distributing information to the entities of the system context diagram.

3. **Data Flow-** this element depicts the flow of the data within the system. It is supported by text to show what type of data is being sent.

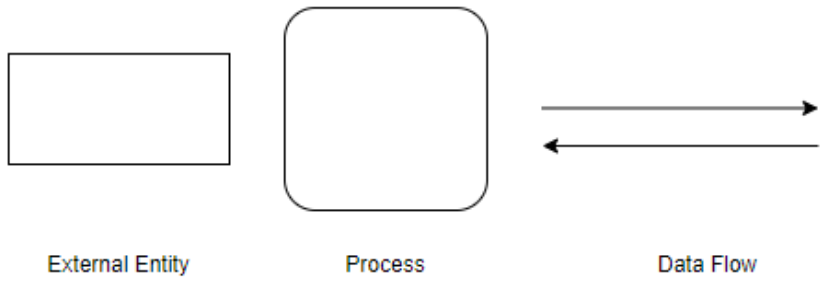


Figure 2.1: Symbols used in Context Diagram

2.2.1 Context Level Diagram for our System

Context level data flow diagram of our Banking Management System —

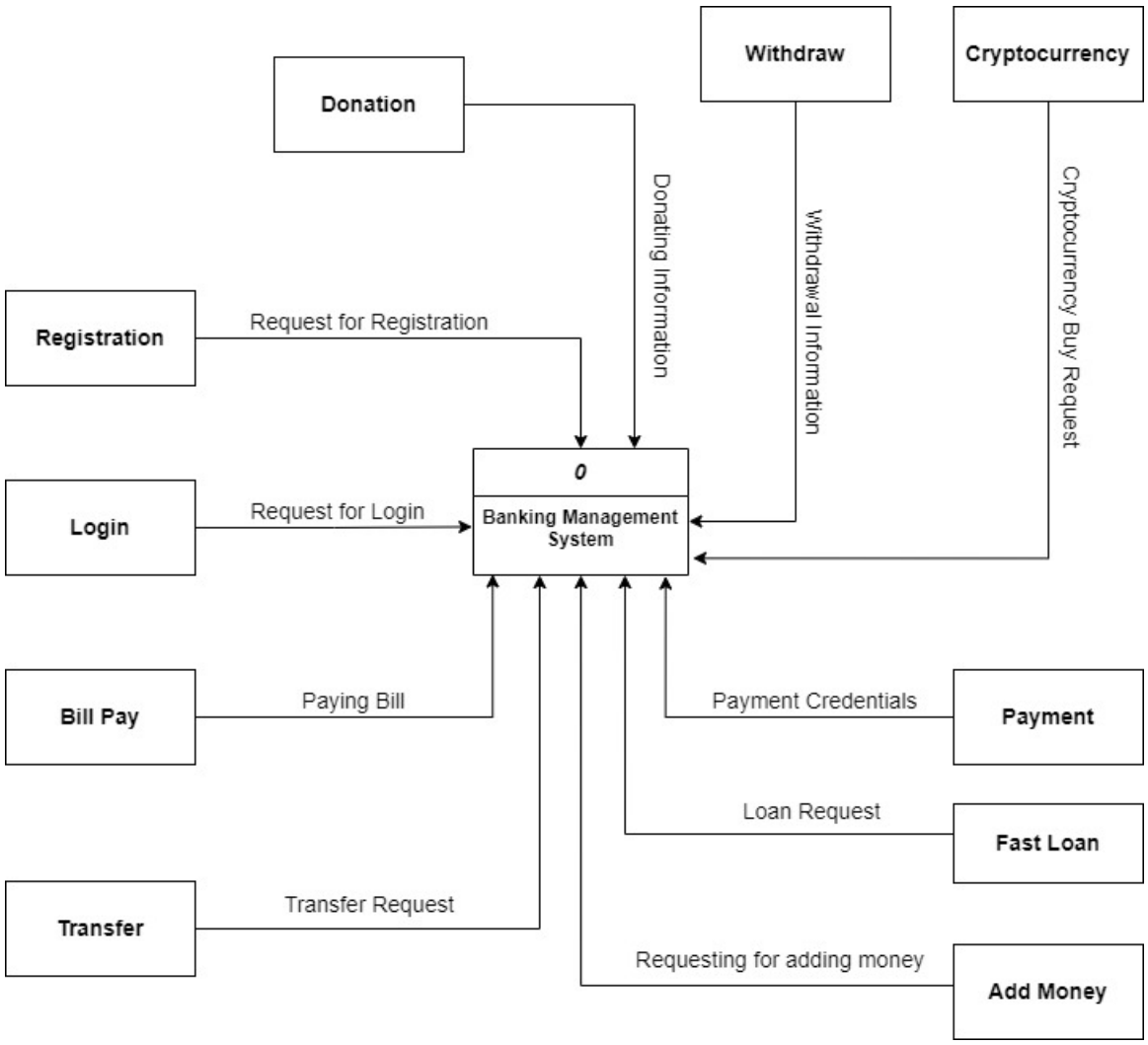


Figure 2.2: Context Level Data Flow Diagram

Scenario : This system is about Hospital Management System. There are three entities: Receptionist,Doctor and Medicine companies. Receptionist mainly set appointment, download the doctor given prescription for patient, add patient information and modify them. Receptionist can suggest patient clinic for test and medical representative names.Medicine companies can provide medical representatives names and prices of various medicines. Doctor can see all the appointment list of his and can update his profile. Doctor also gives prescription. Admin can add, modify patient information, appointments and can add, modify, delete doctor information. Also admin can add new admin through registration.

2.3 Human Information Requirements

Human needs are an important part of our software, through which we can catch the problems of our software, and our primary objective is to satisfy the users with it. We have some problems with this system.

1. For the user's safety, we are looking at the issue OTP system with Email.
2. For this security, we will verify the user's email when he first registers in this system.
3. We will send this OTP in the mail when the user transfers money, and it will do so when he gives it for verification. Yes, this user is the actual owner.
4. Also, since Crypt-Currency will be bought through our system, so a wallet is required for this.
5. We will make this wallet with registration so that the user does not have to do it separately.
6. The wallet will show how much money the user has bought dollars and how many Crypt-Currency dollars he has bought.

2.4 UML Diagrams

UML, which stands for Unified Modeling Language, is a way to visually represent the architecture, design, and implementation of complex software systems.

2.4.1 Sequence Diagram for Banking Management System

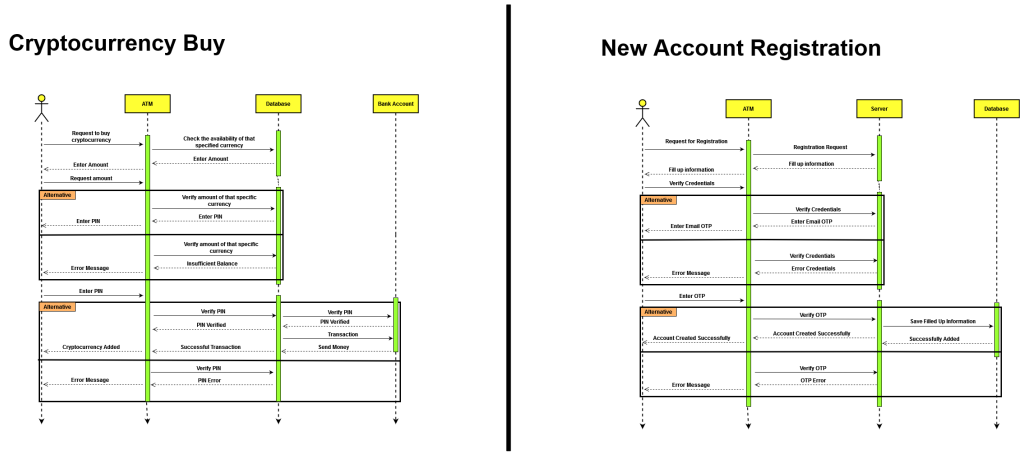


Figure 2.3: Sequence Diagram for Appointment Confirmation

2.4.2 Activity Diagram for User all Process

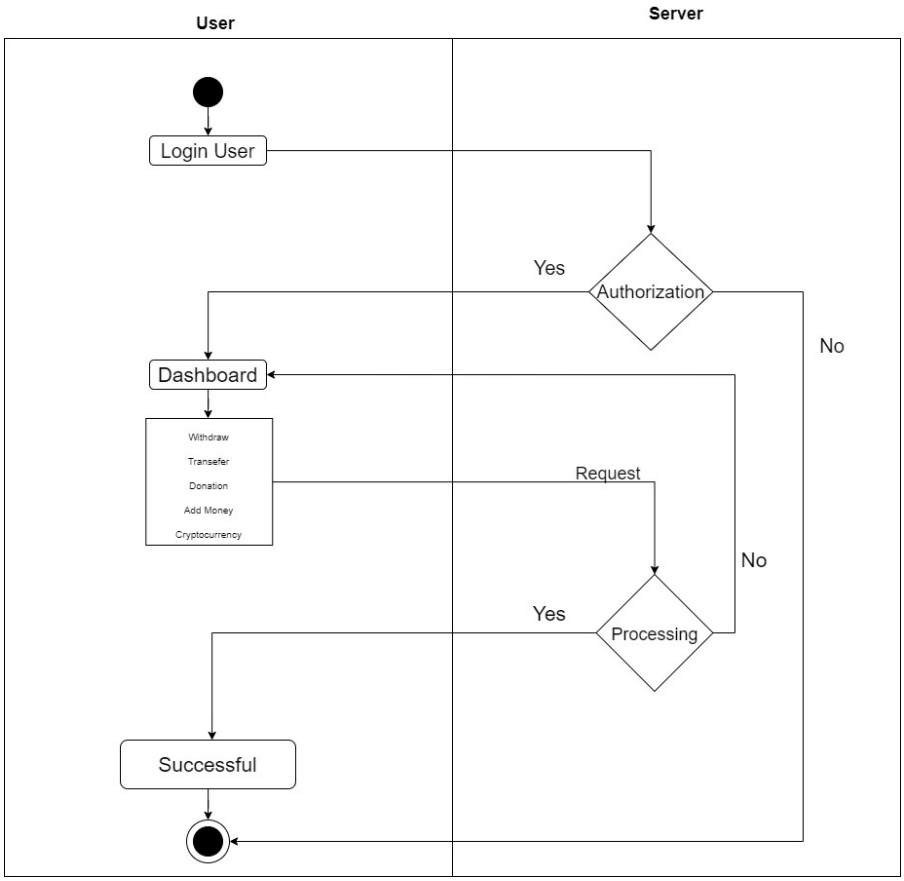


Figure 2.4: Activity Diagram for User all Process

2.4.3 Use Case Diagram for Banking Management System

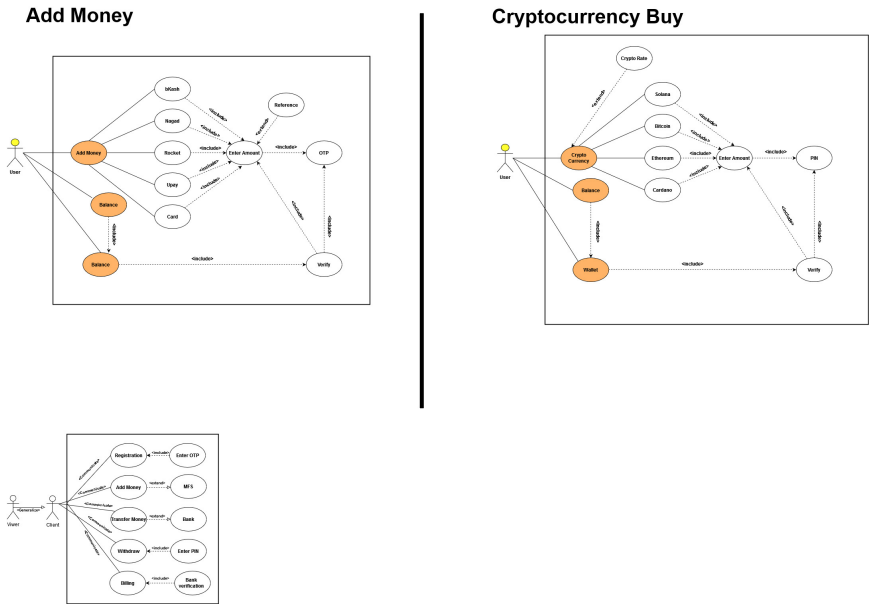


Figure 2.5: Usecase Diagram for Banking Management System

2.4.4 Sequence Diagram for Banking Management System

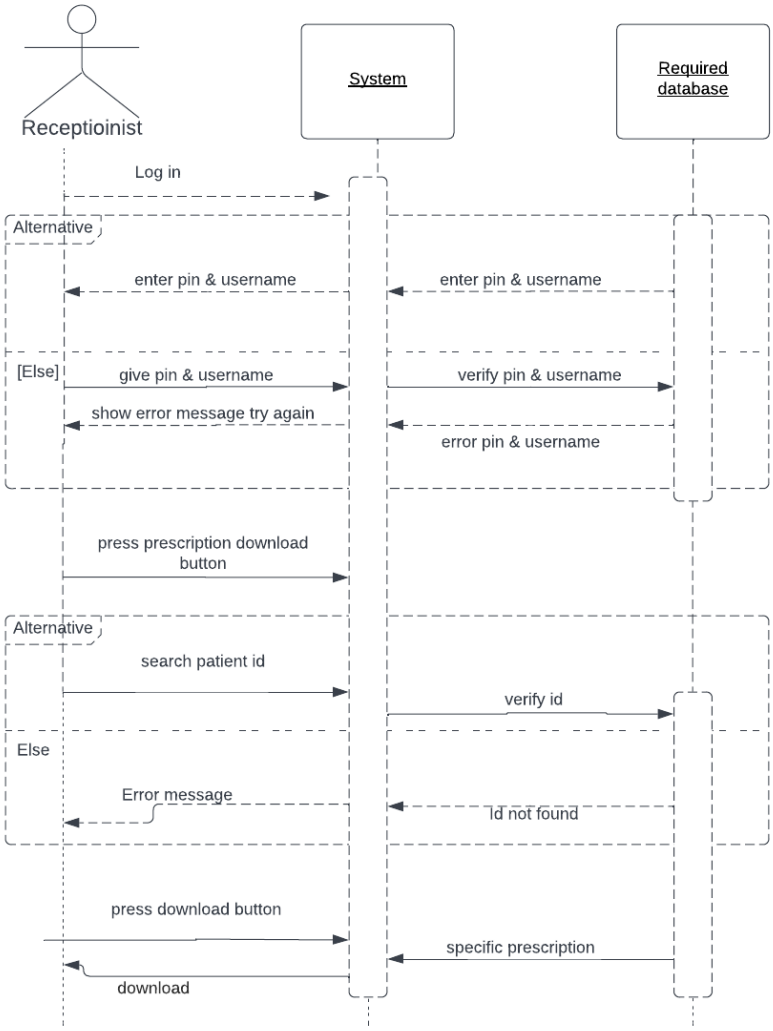


Figure 2.6: Sequence Diagram for Banking Management System

2.4.5 Class Diagram for Banking Management System

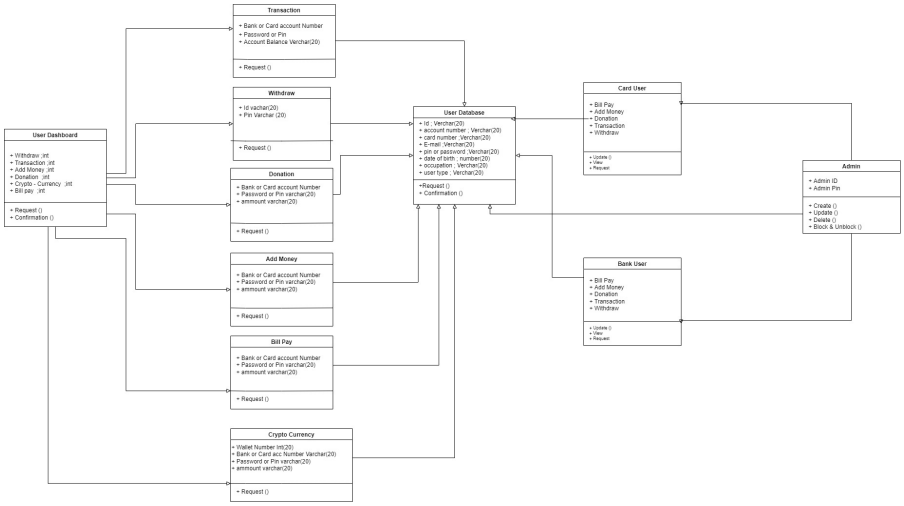


Figure 2.7: Class Diagram for Banking Management System

Chapter 3

Implementation

3.1 Software Tools

We developed a software that why people are easily using his payment system. They can easily deposit his money, transfer his money, also he can buy Cryptocurrency, he can easily exchange BDT to USD. he can check daily Cryptocurrency Rate.We have also developed a E-mail Verification System, That's why a user can any transaction safely. It's a OTP Verification System.

3.1.1 Xampp

XAMPP is a cross-platform web server that is free and open-source. XAMPP is a short form for Cross-Platform, Apache, MySQL, PHP, and Perl. XAMPP is a popular cross-platform web server that allows programmers to write and test their code on a local web server. It was created by Apache Friends, and the public can revise or modify its native source code. It includes MariaDB, Apache HTTP Server, and interpreters for PHP and Perl, among other computer languages. Because of XAMPP's simplicity of deployment, a developer can quickly and easily install a WAMP or LAMP stack on an operating system, with the added benefit that common add-in apps like WordPress and Joomla can also be loaded.

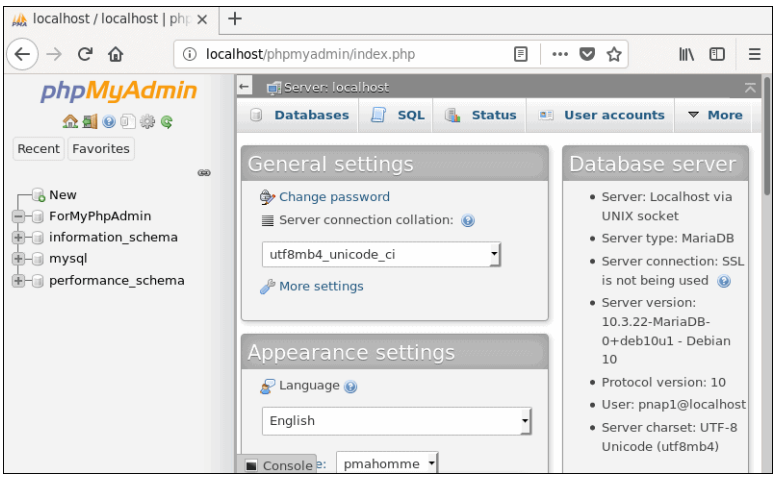


Figure 3.1: Xampp Interface

3.1.2 Visual Studio Community

Visual Studio is an Integrated Development Environment (IDE) developed by Microsoft to develop GUI (Graphical User Interface), console, Web applications, web apps, mobile apps, cloud, and web services, etc. With the help of this IDE, you can create managed code as well as native code. It uses the various platforms of Microsoft software development software like Windows store, Microsoft Silver light, and Windows API, etc. It is not a language-specific IDE as you can use this to write code in C, C++, VB(Visual Basic), Python, JavaScript, and many more languages. It provides support for 36 different programming languages. It is available for Windows as well as for mac OS.

3.1.3 MySQL Connector

MySQL Connector/NET 8.0.23 is the latest General Availability release of the MySQL Connector NET 8.0 series. This version supports .NET 5.0 and the X Dev API, which enables application developers to write code that combines the strengths of the relational and document models using a modern, No SQL-like syntax that does not assume previous experience writing traditional SQL.

3.1.4 C# programming Language

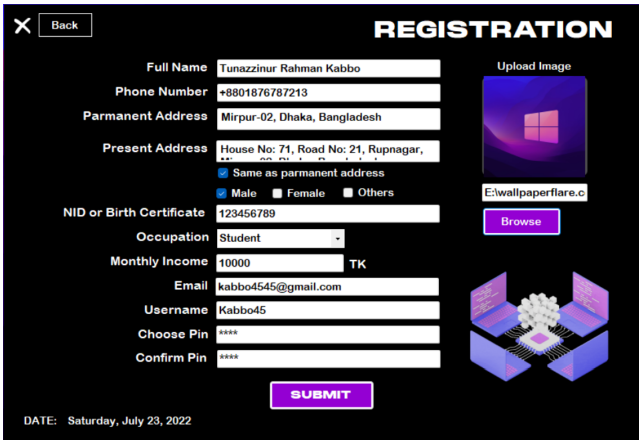
C (pronounced "See Sharp") is a modern, object-oriented, and type safe programming language. C-sharp enables developers to build many types of secure and robust applications that run in .NET. C has its roots in the C family of languages and will be immediately familiar to C, C++, Java, and JavaScript programmers. This tour provides an overview of the major components of the language in C-sharp 8 and earlier. If you want to explore the language through interactive examples, try the introduction to C-sharp tutorials. C-sharp is an object-oriented, component-oriented programming language. C-sharp provides language constructs to directly support these concepts, making C-sharp a natural language in which to create and use software components. Since its origin, C-sharp has added features to support new workloads and emerging software design practices. At its core, C-sharp is an object-oriented language. You define types and their behavior.

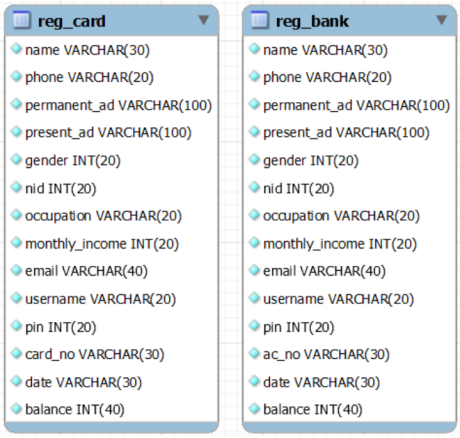
3.1.5 .Net Frame Work

NET is a framework to develop software applications. It is designed and developed by Microsoft and the first beta version released in 2000. It is used to develop applications for web, Windows, phone. Moreover, it provides a broad range of functionalities and support. This framework contains a large number of class libraries known as Framework Class Library (FCL). The software programs written in .NET are executed in the execution environment, which is called CLR (Common Language Run time). These are the core and essential parts of the .NET framework. This framework provides various services like memory management, networking, security, memory management, and type-safety.

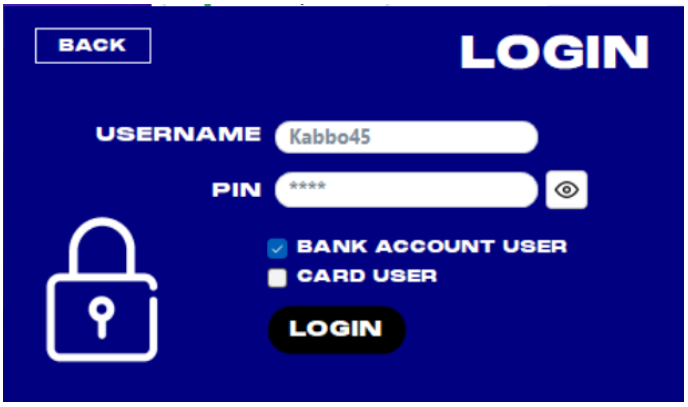
3.2 Software Review

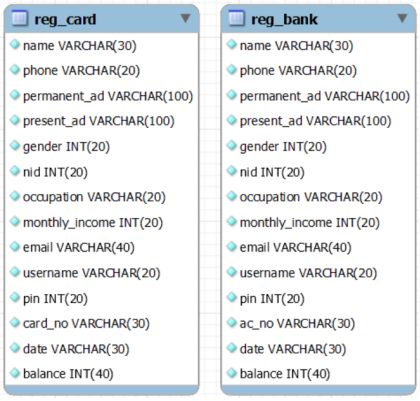
3.2.1 User Registration



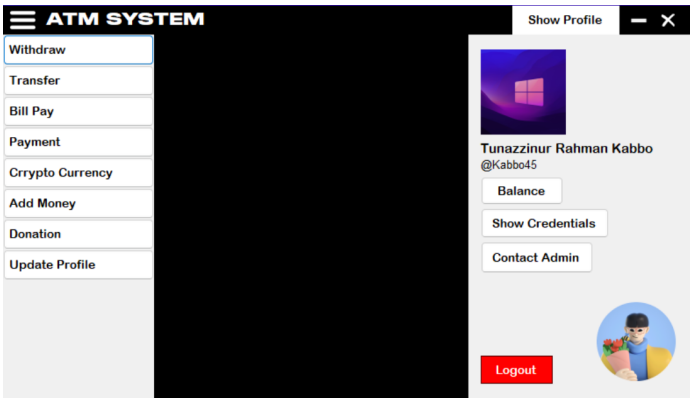


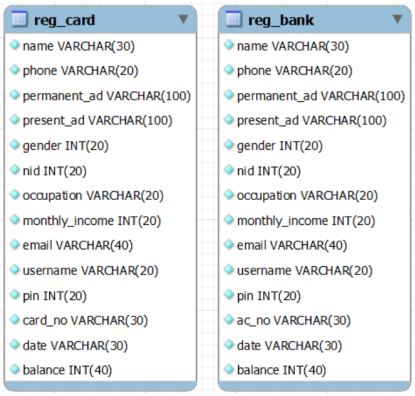
3.2.2 User login



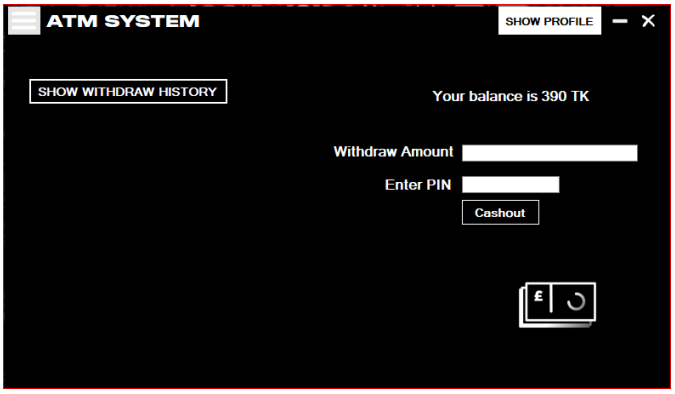


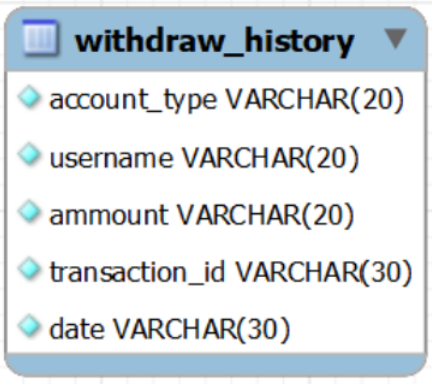
3.2.3 User Dashboard



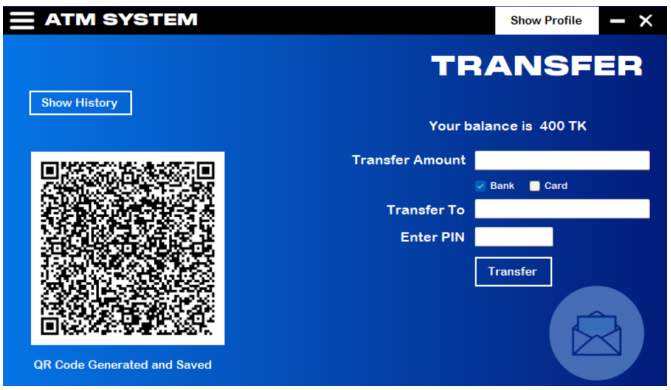


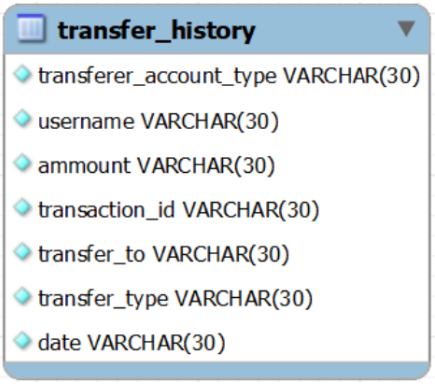
3.2.4 User Withdraw



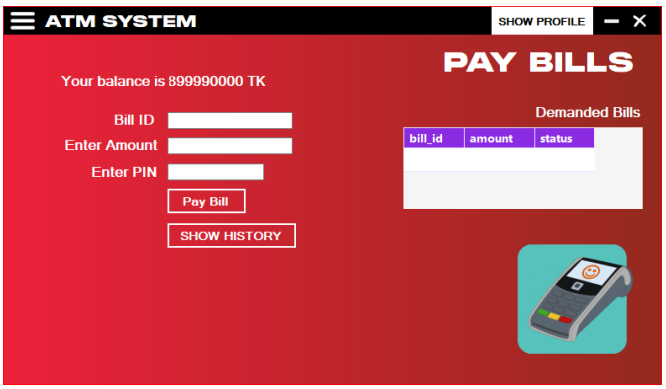


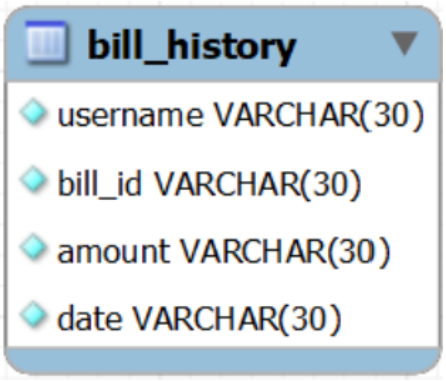
3.2.5 User Transfer



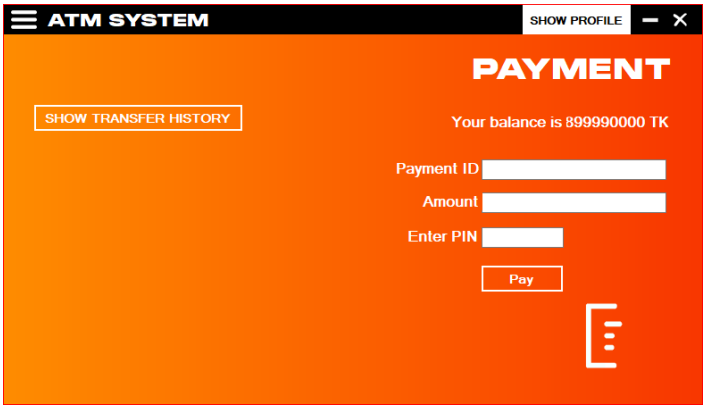


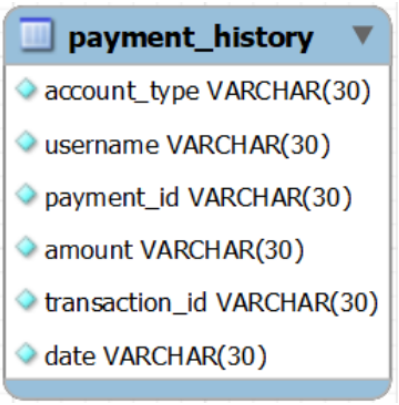
3.2.6 User Bill pay



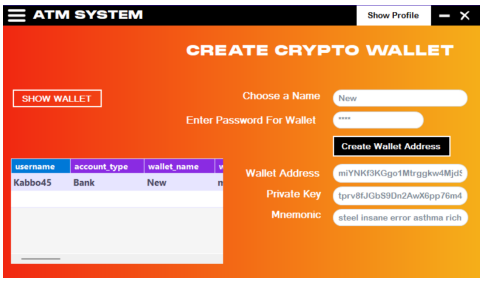


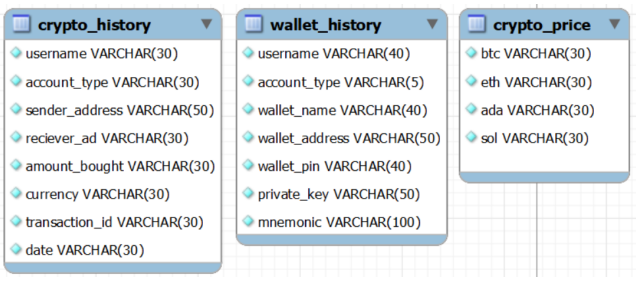
3.2.7 User Payment



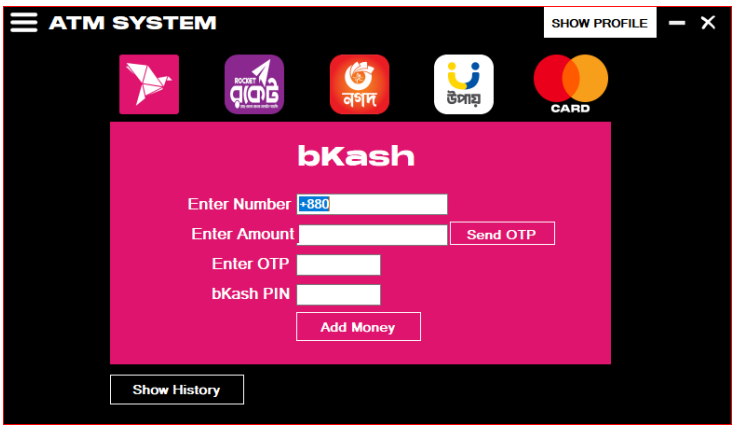


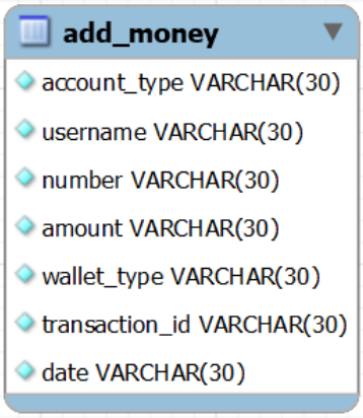
3.2.8 User Crypt Currency



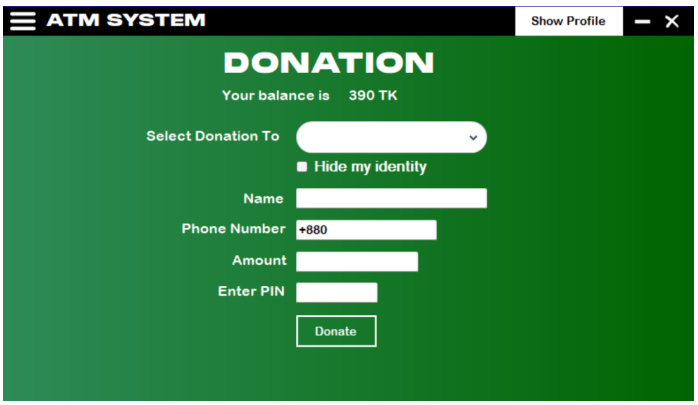


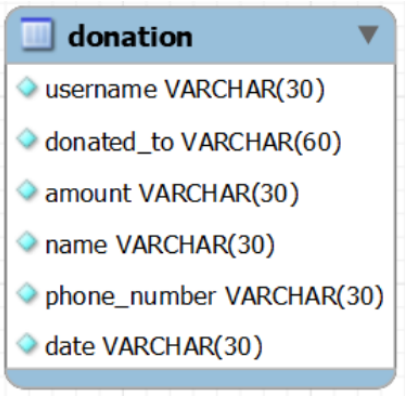
3.2.9 User Add Money



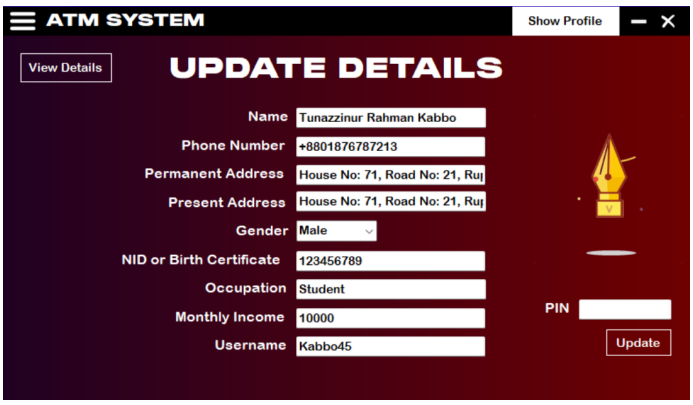


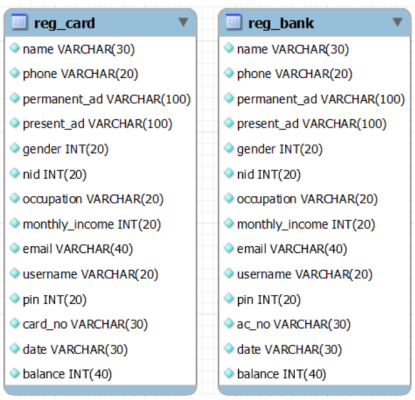
3.2.10 User Donation



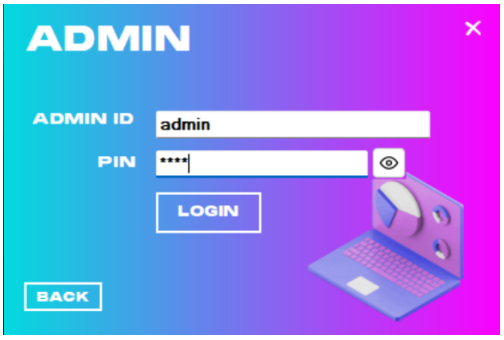


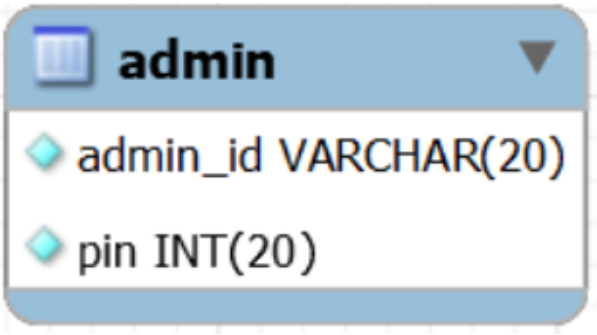
3.2.11 User Profile



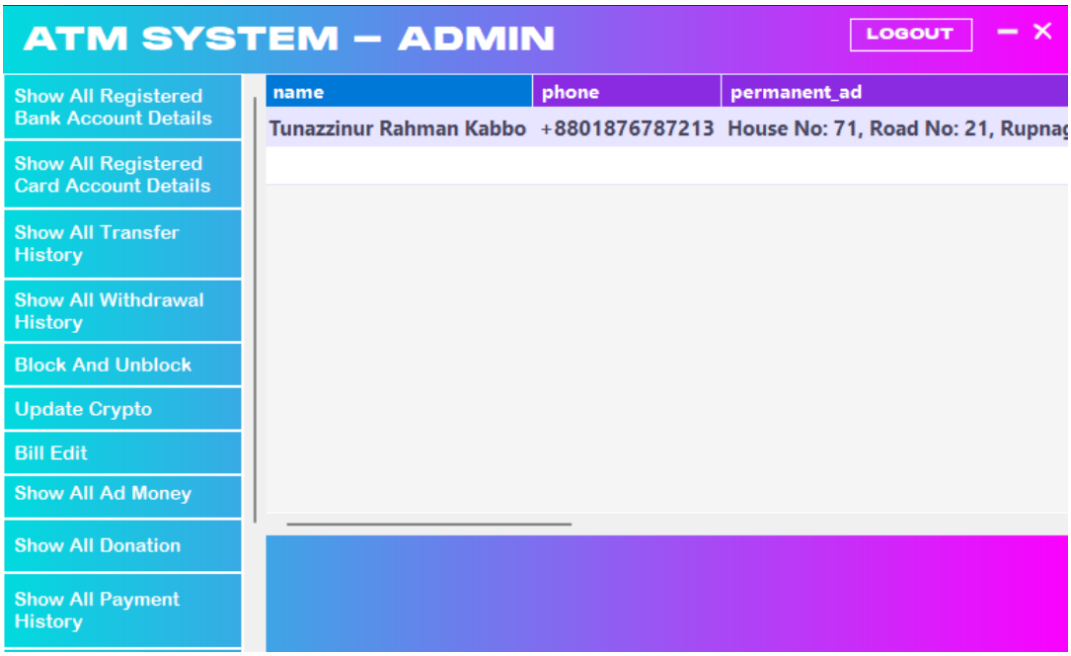


3.2.12 Admin Login





3.2.13 Admin Dashboard



Chapter 4

Conclusion

4.1 Advantages of Our System

While things have been busy schedule for working people as of late, they still manage to make some time for bank works when they have some leisure or downtime to spare.

- A user can complete the transition from any part of the world.
- Users can send crypto currency using our system from anywhere. Can create crypto wallets.
- We will need some kind of passport to use this system. Besides, passport will be required to use dual Currency.
- A user can easily transfer money through email verification.

- To use this facility, any user can install our software and use it.
- Help user do banking transaction.
- Help user do banking transaction.
- Online shopping
- Help user do banking transaction.

The software is easy to use, and makes banking simple both for the bank and the customers. The functionality of integration helps production of statements, reports, and messages via email.

Reduction in Cost of Operation

It helps to reduce the workforce that a bank requires at each branch to run its operations professionally.

24/7 Availability

The key advantage of banking software is that the customers can make transactions around the clock. They could access their account from any part of the world and make a transaction at their ease.

4.2 Limitation

- Users cannot trade crypto in our system.
- Currently we have 4 crypto-currencies in our wallet.
- Add money is in a prototype phase.
- No Bug-reporting option.

4.3 Future Work

Consumers' growing desire to access financial services from digital channels has led to a surge in new banking technologies that are reconceptualizing the banking industry.

- More user-friendly system.
- Option for reporting bugs.
- Option for feedback.

4.4 Developers contribution final words

We started this project when we had very few concepts about the C language and its different usage. Over time, we've worked to make our project more compact and more user-friendly. During this period, we learned various C programming concepts and now in position to design in the error free program successfully. Overall, the team feels that the project is pretty successful although we did face a few problems here and there but we came to know the various C programming techniques.

- More user-friendly system.
- Option for reporting bugs.
- Crypt-Currency Trading

4.4.1 Reference

- Stack Overflow
- GitHub
- W3School