Tribhuvan University

Faculty of Humanities and Social Science

Online Banking System

A Project Report

Submitted to:

Department of Computer Application

Pascal National College

In partial fulfillment of the requirements for the Bachelors in Computer Application

Submitted by:

Sangam Subedi | 6-2-1226-21-2021

Aaditya Khatri | 6-2-1226-01-2021

Under the Supervision of

Kabindra Koirala

Tribhuvan University

Faculty of Humanities and Social Science

Pascal National College

Supervisor's Recommendation

I hereby recommend that this project prepared under my supervision by Aaditya Khatri and Sangam Subedi entitled "**Online Banking System**" in partial fulfillment of the requirements for the degree of Bachelor of Computer Application is recommended for the final evaluation.

Kabindra Koirala

Project Supervisor

BCA Department

Pascal National College, Lalitpur

Tribhuvan University

Faculty of Humanities and Social Science

Pascal National College

Letter of Approval

This is to certify that this project prepared by Aaditya Khatri and Sangam Subedi entitled "**Online Banking System**" in partial fulfillment of the requirements for the degree of Bachelor in Computer Application has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

SIGNATURE of Supervisor	**SIGNATURE of HOD/ Coordinator**
SIGNATURE of Internal Examiner	**SIGNATURE of External Examiner External**

ABSTRACT

Online Banking System combines a number of processes and systems to automate and easily manage all the user information provided with registration, withdrawal, fund transfer and account management. This paper presents the design and implementation of the Online Banking System aimed to provide efficient financial services to the users. The system integrates a user-friendly interface accessible via web. The primary objectives of the system are to help manage accounts of the user, fund transfer, withdrawal, balance inquiry and view statement functionalities.

Water methodology is implemented for this system that includes a series of starting with requirement analysis, design, implementation and testing. Requirement analysis is used to require essential information about how the system is implementing its requirements like functional, non-functional. Design phase consists of how the information is transformed into system design specifications. It can be illustrated using various tools. Implementation phase is implemented through various programming languages and testing is done.

ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to our supervisor **Mr. Kabrindra Koirala** who gave us the golden opportunity to do this wonderful project on the topic of Online Banking System, which also helped us in doing a lot of research and we came to know about so many new tools and technologies.

We would like to express our special thanks to our College Principal **Mr. Ashok Kumar Panth** who gave us permission for doing this project.

I am highly indebted to **Pascal National College** for their guidance and constant supervision as well as for providing necessary information regarding the Project and support in the completion. We would also like to thank our parents and friends who helped us a lot in finalizing this project within the limited time frame.

In the end, we would also like to thank Tribhuvan University for giving us this opportunity via the course of Computer Application to help us understand the project ethics at this early stage and helped us to evaluate my knowledge and expand it a little more.

Yours sincerely,

Aaditya Khatri

Sangam Subedi

Table of Contents

List of Abbreviations ..... **vi**
List of Figures ..... **vii**
List of Tables ..... **viii**
Chapter 1: Introduction **1**
1.1 Introduction 1
1.2 Problem Statement 2
1.3 Objectives 2
1.4 Scope 2
1.5 Limitations 2
1.6 Report Organization 3
Chapter 2: Background Study and Literature Review **4**
2.1 Background Study 4
2.2 Literature Review 5
Chapter 3: System analysis and Design **6**
3.1 System analysis 6
3.1.1 Requirement Analysis 6
I. Functional Requirement 6
II. Non-functional Requirement 7
3.1.2 Feasibility Analysis 7
I. Technical 7
II. Operational 8
III. Economic 8
IV. Schedule 8
3.1.3 Data Modeling (ER diagram) 9
3.1.4 Process Modeling (DFD) 10
3.2 System Design 12
3.2.1 Architectural Design 12
3.2.2 Database Schema Design 13
3.2.3 Interface Design 14
3.2.4 Physical DFD 17
Chapter 4: Implementation and Testing **19**
4.1 Implementation 19
4.4.1 Tools Used 19
4.4.2 Implementation of Modules 21
4.2 Testing 22
4.2.1 Test Cases for Unit Testing 22
4.2.2 Test Cases for System Testing 24
Chapter 5 : Conclusion and Future Recommendations **25**
5.1 Lessons learned / Outcome 25
5.2 Conclusions 25
5.3 Future Recommendations 26
Appendices **27**
Screenshots 27

Source Codes 33
References 34

List of Abbreviations

PHP: Hypertext Preprocessor

HTML: Hypertext Markup Language

CSS: Cascading Stylesheet

JS: JavaScript

MySQL: My Structured Query Language

DFD: Data Flow Diagram

ERD: Entity Relationship Diagram

List of Figures

Figure 1: Waterfall Model **6**
Figure 2: Gantt Chart **8**
Figure 3: ER-Diagram of Online Banking System **9**
Figure 4: Context or Zero level DFD **10**
Figure 5: Level 1 DFD **11**
Figure 6: Architectural Design **11**
Figure 7: Schema Design **12**
Figure 8: Wireframe for Dashboard **13**
Figure 9: Wireframe for Client side **14**
Figure 10: Wireframe for Fund Transfer **15**
Figure 11: Wireframe for Login page **15**
Figure 12: Context Diagram **16**
Figure 13: Level 1 DFD **17**

List of Tables

Table 1: Test case for Registration of an user **21**
Table 2: Test case for login of an user **21**
Table 3: Test case for Account Management **22**
Table 4: Test case for add of an user (Success) **22**
Table 5: Test case for add of an user (Failure) **22**
Table 6: Test case for Fund Transfer **23**
Table 7: Test Case for UI Elements **23**
Table 8: Test case for error handling **23**

Chapter 1: Introduction

1.1 Introduction

As we know that, An “Online Banking System” is a digital platform that allows the users/customers to access and manage their financial accounts over the internet. It provides a convenient way for users to perform various banking transactions without physically visiting a bank branch.

Online Banking System is a digital platform that enables individuals to conduct various financial transactions and manage their accounts over the internet. This system provides users with the convenience of accessing banking services from the comfort of their homes or on-the-go through mobile devices. Users can view account balances, access transaction histories, transfer funds between accounts and withdrawal amount.

Security features such as multi-factor authentication and encryption are implemented to ensure the confidentiality and integrity of user data. Mobile banking applications further enhance accessibility, allowing users to perform banking tasks using smartphones and tablets.

Overall, an online banking system streamlines financial processes, offering a range of services including account management, withdrawal and fund transfers while adhering to regulatory standards and providing robust customer support.

1.2 Problem Statement

In the context of Nepal, the internet and online banking systems face a myriad of challenges that contribute to their limited accessibility and effectiveness. One major hurdle is the limited internet penetration, particularly in rural and remote areas, where users encounter difficulties accessing online banking services due to poor connectivity.

Additionally, low levels of digital literacy present another obstacle, making the online banking process intimidating for some users. Security concerns further impede the system's functionality, with risks ranging from phishing attacks to identity theft and data breaches. Inadequate encryption and weak authentication methods exacerbate these risks, making online transactions more vulnerable than traditional methods.

A crucial factor contributing to the challenges is the lack of trust in online banking systems, stemming from worries about data privacy, security breaches, and the potential for financial fraud. To overcome this, transparent communication, robust security measures, and consistent regulatory compliance are essential to reassure users. Lastly, customer support challenges, including language barriers, limited accessibility in remote areas, and varying digital literacy levels, add another layer of complexity to the overall effectiveness of online banking in Nepal. Addressing these issues requires a holistic approach that encompasses infrastructure development, educational initiatives, and regulatory improvements.

1.3 Objectives

- To develop a system that helps to create and manage accounts.
- To develop a system that consists of functionalities like payment, balance inquiry and view statement.

1.4 Scope

- This system will incorporate web accessibility features, ensuring users can seamlessly access and manage their banking services.
- This system will design and implement some of the core functionalities such as user registration, account creation and basic account management.

1.5 Limitations

- No QR Scan Feature for payment system at the moment

- No Linkup with bank as a complete service, as further integration costs money to buy APIs.

1.6 Report Organization

Introduction

This chapter deals with the introduction of the system with its objectives and limitations along with the reason why the system is made.

Background Study and Literature Review

This chapter summarizes the work that has been carried out in the field of online banking system and also describes the features about some existing applications related to the Online banking system.

System Analysis and Design

This chapter focuses on the different requirements of the system, which describes the functional, non-functional, feasibility analysis, Entity Relational diagram, Data Flow Diagram, design of the system with system architecture, database schema, and interface design.

Implementation and Testing

This chapter emphasizes tools used in system development, implementing details and results of test performed.

Conclusion and Future Recommendation

This chapter highlights a brief summary of lessons learnt, outcome and conclusion of the whole project and explains what has been done and what further improvements could be done.

Chapter 2: Background Study and Literature Review

2.1 Background Study

Online banking systems have revolutionized the way customers interact with their financial institutions, offering unparalleled convenience, accessibility, and efficiency. Traditionally, banking transactions required customers to visit physical branches during business hours, often leading to long wait times and limited flexibility. However, with the advent of online banking, customers now have the freedom to manage their finances anytime, anywhere, through secure digital platforms.

Online banking offers a range of benefits. One of the primary advantages is the ability to perform a wide range of financial transactions without the need to visit a physical branch. Whether it's checking account balances, transferring funds between accounts, paying bills, or applying for loans, online banking provides customers with a convenient and efficient way to manage their money.

2.2 Literature Review

Online banking has become an integral part of modern banking services, transforming the way customers interact with financial institutions. This literature review aims to provide an overview of key themes and findings in the existing body of research related to online banking systems. The review will focus on technological advancements, security measures, user experience, and recent developments in the field [1].

For this project, we conducted an analysis of a few relevant websites, i.e Nepal Bank . These websites offer a similar feature set and functionalities. As part of our research, we actively engaged with these websites as a user to identify their area of weakness and their strength. By assuming the user perspective, we found out the valuable understanding of the expectations and requirements of a high quality online bank management system.

According to the source, Nepal Bank, it has every function like this project but the system hasn't facilitated a good user interface which makes users not want to visit again.

A study about e-banking over 1999–2006 shows that the application of e-banking can improve banks' performance in terms of the growth in assets, reduction in operating expenses and portfolio enhancement [2]. Even in the 1990s, Sraeel (1996) emphasizes that creating virtual banking will not only create a new service delivery channel, but also lead to value creation to both banks and customers. AmatoMcCoy further argues that customers will be attracted to e-banking when the advanced e-banking services like e-transfer and e-bill options are available.

Today, the banking system in the country has become so technologically advanced that Almost all banking services are delivered through electronic platforms. Electronic banking in North Macedonia has a relatively recent history of its development [3].

Chapter 3: System analysis and Design

3.1 System analysis

This project will be done by using the Waterfall Model with additional development. It includes a series of starting with requirement analysis, design, implementation and testing. During requirement analysis, all the functional and nonfunctional requirements are analyzed and the system is developed according to the requirement then designing of the system is carried out. After the design process, implementation as well as coding and development part is started. Then after integrating the system the testing part is done.

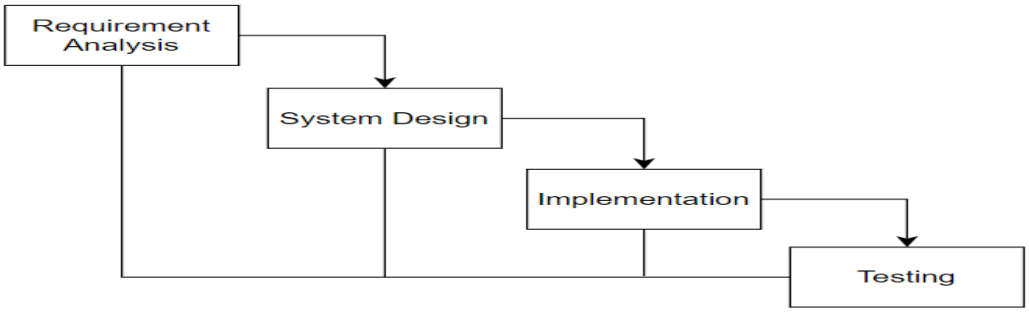


Figure 1: Waterfall Model

3.1.1 Requirement Analysis

I. Functional Requirement

These requirements are what of your website i.e. it includes the functions and core operations of the banking system which lets the users interact with the website. These are tangible, easy to quantify and understand, and typically define the behavior of a system based on user interactions. The system should provide following functionalities:

- Users should be able to register online while providing necessary information.
- Users should be able to create, view and manage their bank accounts.
- Users should be able to sign in with passwords.
- The system should support functions like fund transfer, withdrawal, checking balance, updating personal information.

- Admin should be able to manage and authenticate customer information and accounts.

II. Non-functional Requirement

It describes the general properties of a system. They are among the most important things to define when building the specification for an e-commerce platform, as most have a direct impact on the experience and satisfaction of your users. Here are some basic types of nonfunctional requirements:

- The system should be available 24 hours of the time.
- The system should be compatible with the latest version of popular browsers, including Opera, Chrome and Edge.
- The system should be able to handle multi-user connections during peak hours.
- Users should be able access their account after successful authentication.

3.1.2 Feasibility Analysis

Feasibility is the study of impact, which happens in the organization by the development of a system. The impact can be either positive or negative. When the positives nominate the negatives, then the system is considered feasible. Here the feasibility study can be performed in three ways such as technical feasibility, operational feasibility, economical feasibility and schedule feasibility..

I. Technical

It is technical feasible, since there will not be much difficulty in getting required resources for the development and maintaining the system as well. The project is being built by using simple technologies to reduce technological hindrance. The project is written on VScode editor. To deploy the application, the only technical aspect needed are mentioned below:

Operating System - Windows or any OS platform

RAM - 512MB or higher

Disk space - 1 GB

Higher Web Tools - HTML editors

Device - PC, Laptop

II. Operational

The project ensures user-friendly access with simple design and seamless integration into existing banking operations. It incorporates security with validation, complies with regulatory standards providing an environment for transactions. The system supports scalability to accommodate future growths.

III. Economic

Development of this application is highly economically feasible. We didn't spend that much money for its development. The only thing to be done is making an environment for the development with effective supervision.

IV. Schedule

This includes the project schedule and all time allocated for their completion. The Gantt chart is as follows:

	Start Date	End Date	Duration	Status
Research and Planning	11/21/2023	11/28/2023	7	Completed
Requirement Gathering	12/24/2023	1/7/2024	14	Completed
System Design	1/21/2024	2/10/2024	20	Completed
Development Phase 1	2/12/2024	3/10/2024	27	Completed
Review	3/12/2024	3/17/2024	5	Completed
Development Phase 2	3/26/2024	4/28/2024	33	Completed
Review	5/14/2024	5/19/2024	5	Completed
Testing and Revision	6/15/2024	6/18/2024	3	Completed

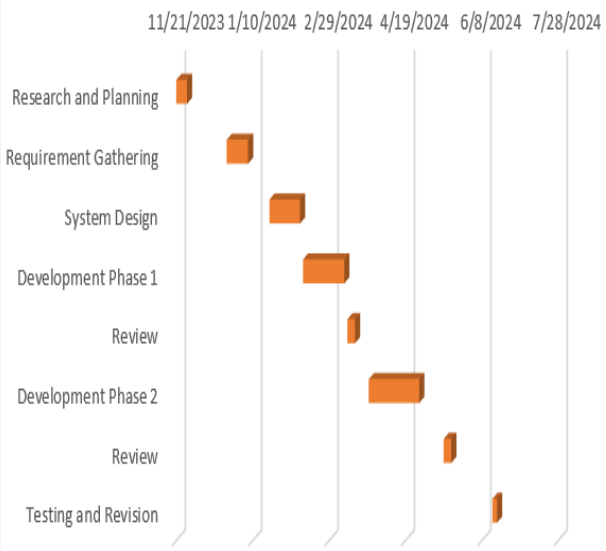


Figure 2: Gantt Chart

3.1.3 Data Modeling (ER diagram)

A data model is a mechanism that provides abstraction for database application. Data models define how data is connected to each other and how they are processed and stored in a system. There are different data models which are used to design and develop the database, but here we use ER (Entity Relationship) Model. E-R Model is a popular high level conceptual data model that consists of a collection of basic objects called entities and relationships among these objects. The ER diagram of online banking system is as shown below:

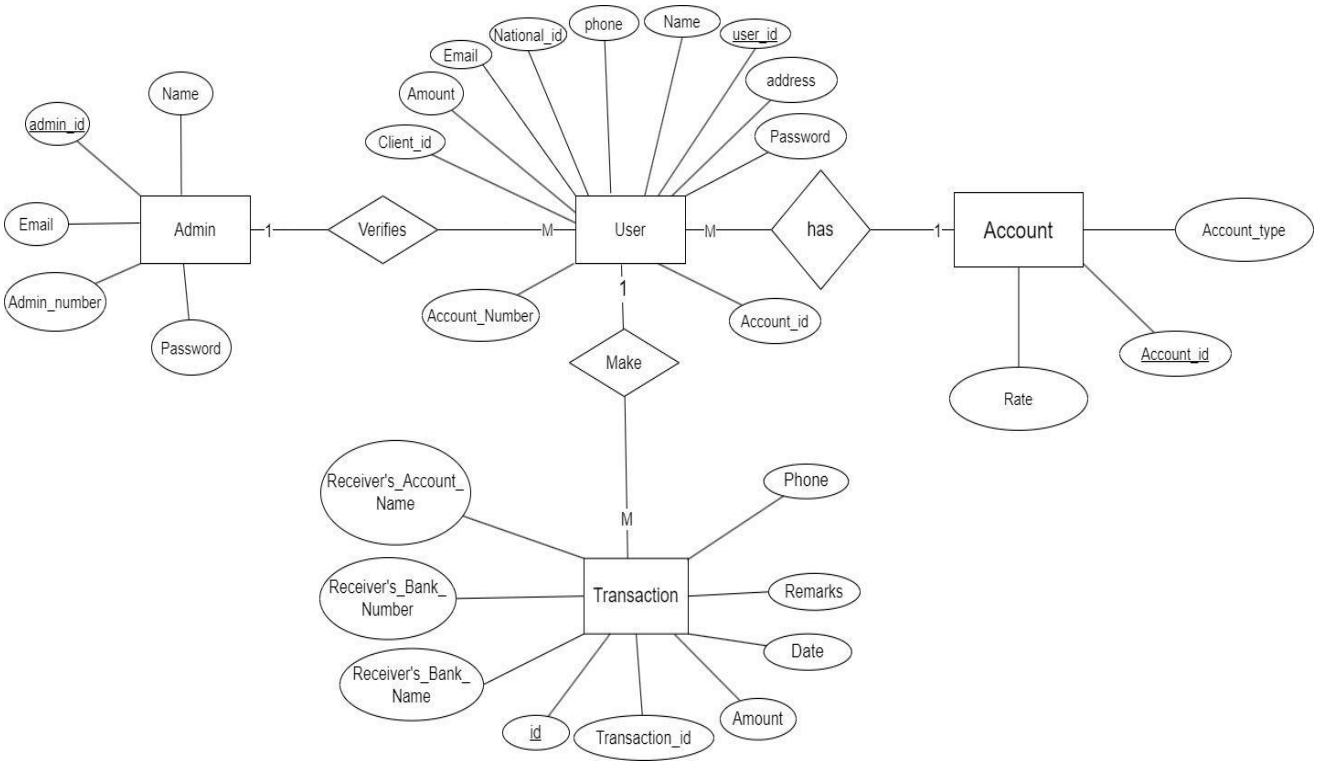


Figure 3: ER-Diagram of Online Banking System

There are three entities for the online banking system : User, Admin, Account and Transaction. User have attributes like user_id, name, address, phone, National_id, Email, password, Amount, Client_id, account_number, account_id whereas user_id remains the primary key. Account entities hold attributes like account_id, account_type, rate where account_id being the primary key. Transaction entity holds attributes like Receiver's_Bank_Number , Receiver's_Bank_Name,Receiver's_Account_Name, id, Transaction_id, Amount, Date, Remarks, phone whereas id being the primary key.Admin entity holds attributes like name, admin_id, email, admin_number, password whereas admin_id being the primary key. There is one to many relationship between account and users also between users and Transaction.

3.1.4 Process Modeling (DFD)

A DFD can be referred to as a Process Model. A data-flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. A data flow diagram (DFD) is a significant modeling technique for analyzing and constructing information processes illustrates this flow of information in a process based on the inputs and outputs. Initially a context diagram is drawn, which is a simple representation of the entire system under investigation. This is followed by a level 1 diagram; which provides an overview of the major functional areas of the business.

Context Diagram

This diagram shows the system in a single process within the external entities.

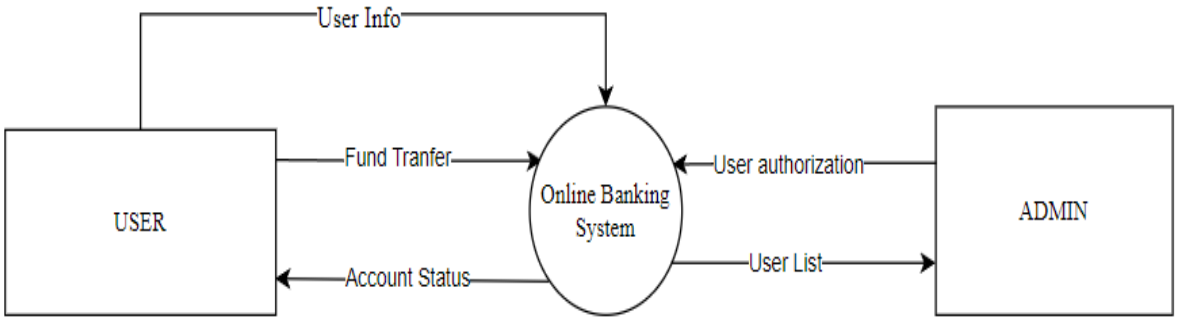


Figure 4: Context or Zero level DFD

Level 1 DFD

This diagram breaks down the main processes into more detailed sub processes.

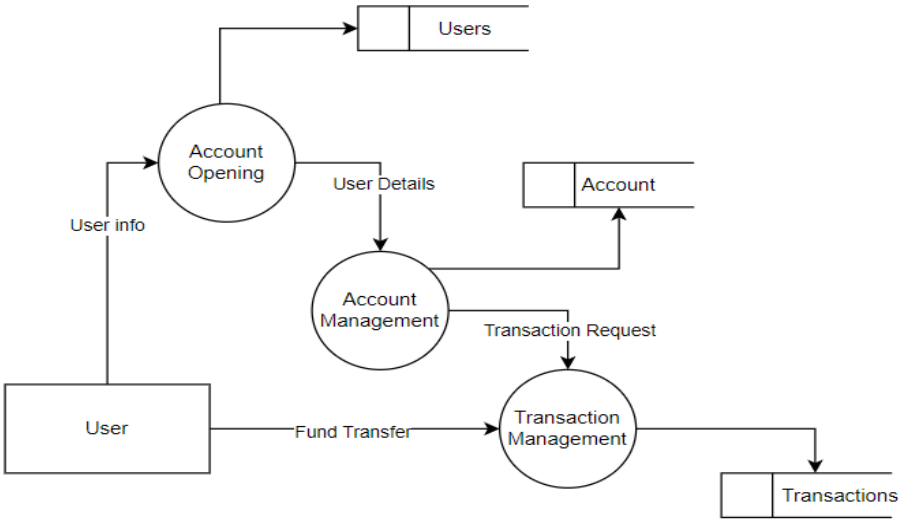


Figure 5: Level 1 DFD

3.2 System Design

To realize the different functional requirement of the system in graphical form, different design diagram of the systems has been prepared which are as follows:

3.2.1 Architectural Design

For this system, three tier architecture is used which includes presentation, server and database tier. In architectural design, basic structure is given.

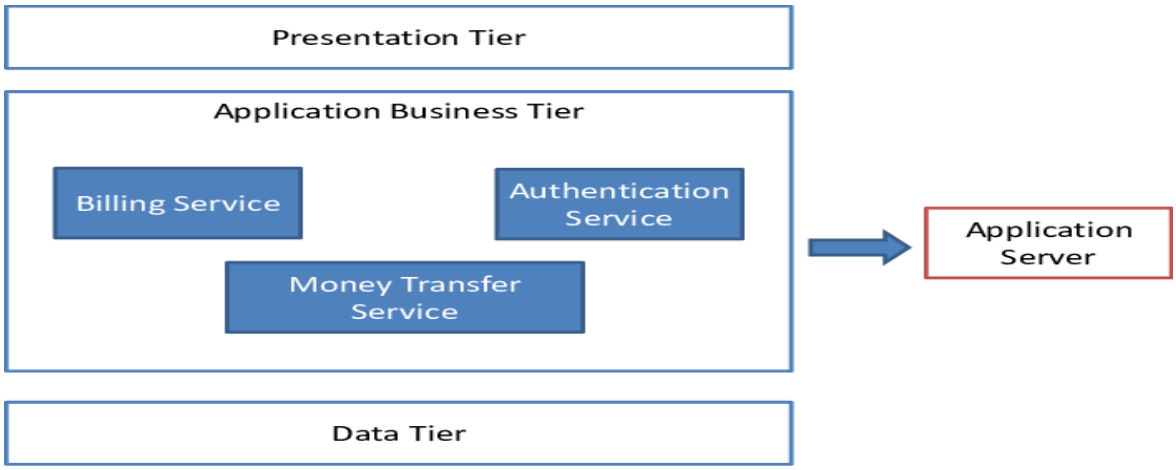


Figure 6: Architectural Design

3.2.2 Database Schema Design

The database schema design for the online banking system is represented below:

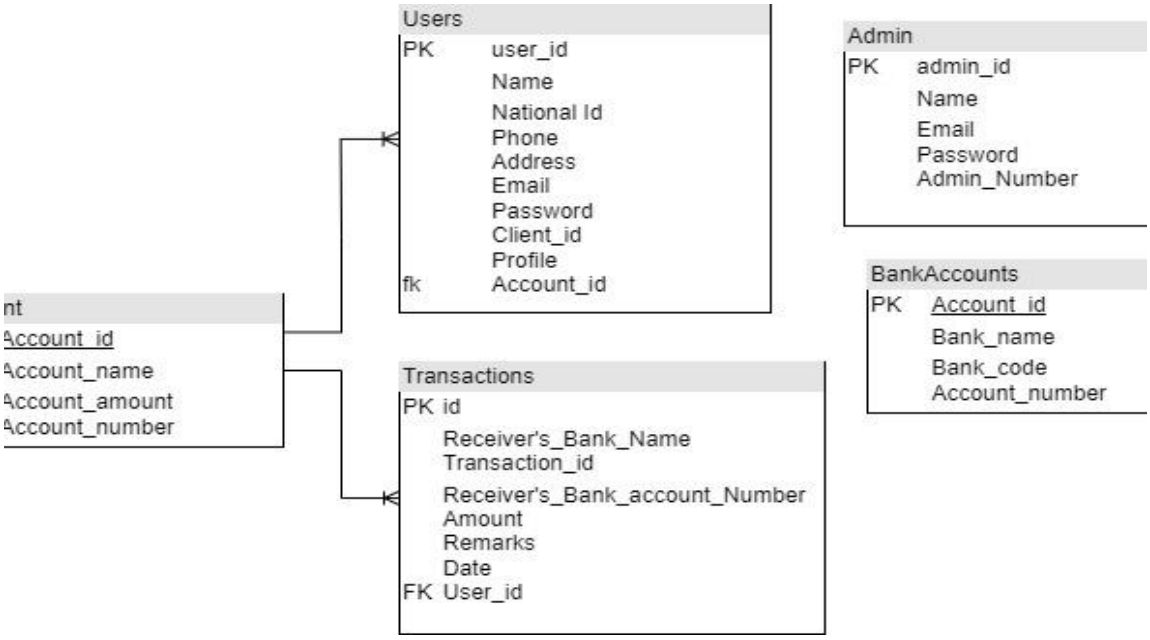


Figure 7: Schema Design

In the above schema, we can figure out there are five tables in the database: User, Account, Bank accounts, admin and Transaction. The User table has attributes like user_id which is the primary key for the user table with data type int and other attributes like name, address, phone, National_id, Email, password, Amount, Client_id, account_number, account_id. Similarly in the next table Account it contains primary key account_id as int data type and other attributes like account_type, Rate. Transaction table contains primary key id as int data type and other entities like Receiver's_Bank_Number, Receiver's_Bank_Name, Receiver's_Account_Name, Transaction_id, Amount, Date, Remarks, phone.Admin table contains primary key admin_id and other attributes like name, email, admin_number, password. BankAccounts table contains primary key account_id and other attributes like Bank_name, Bank_code, Account_number.

3.2.3 Interface Design

Interface design refers to the process of creating visual elements, interactions, and layouts through which users interact with a product or system. A simple wireframe of Online Banking system are represented below:

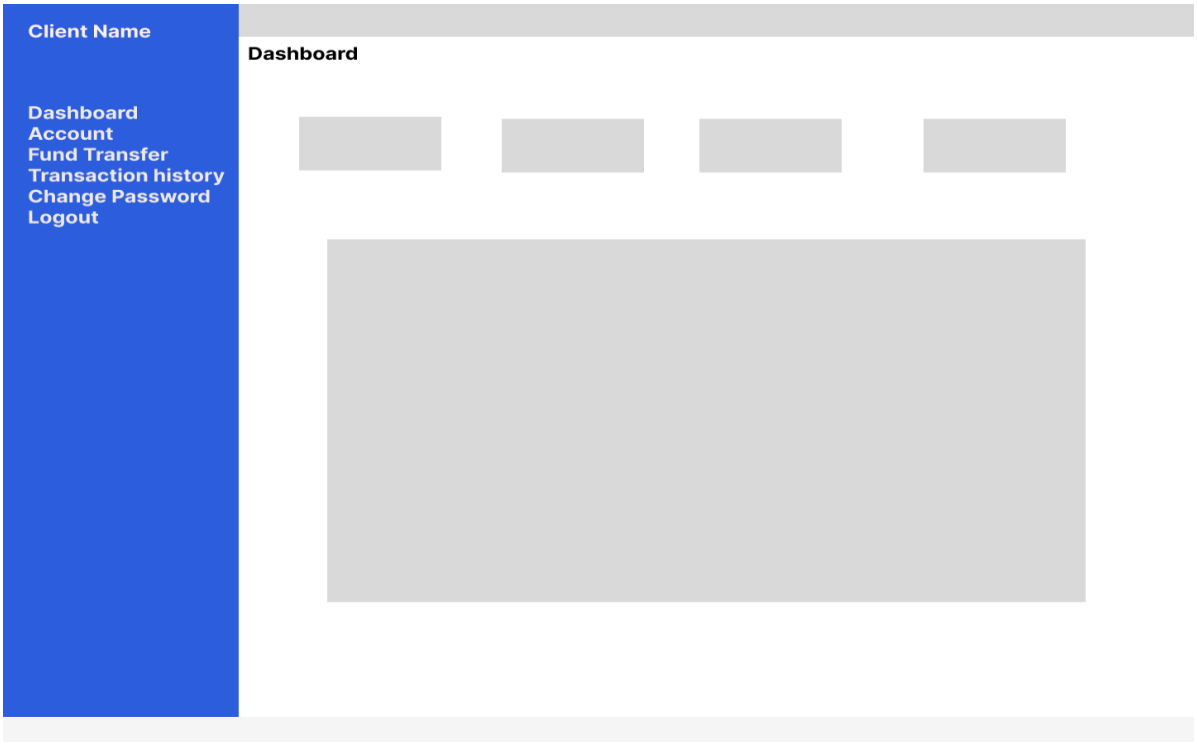


Figure 8: Wireframe for Dashboard

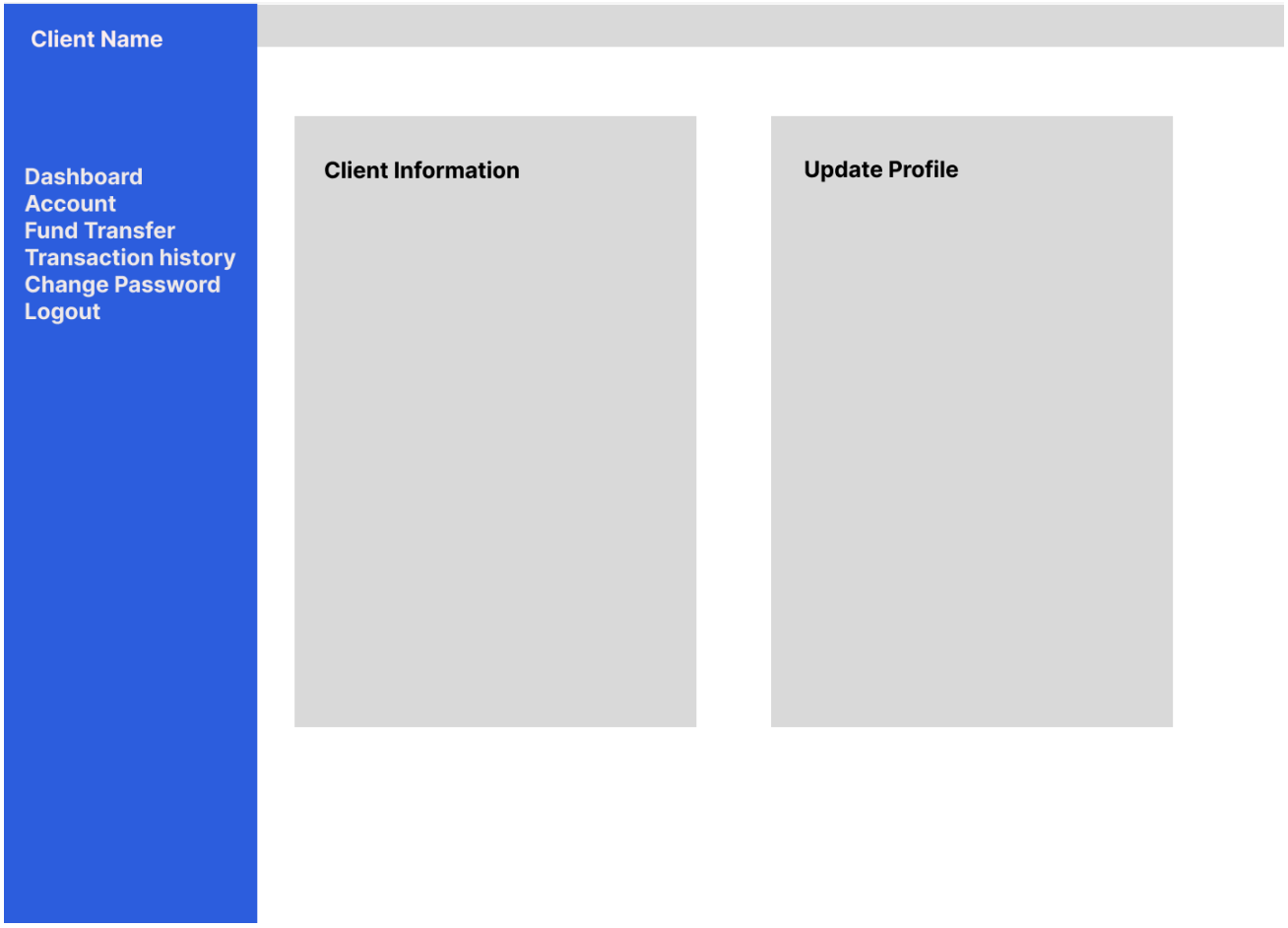


Figure 9: Wireframe for Client side

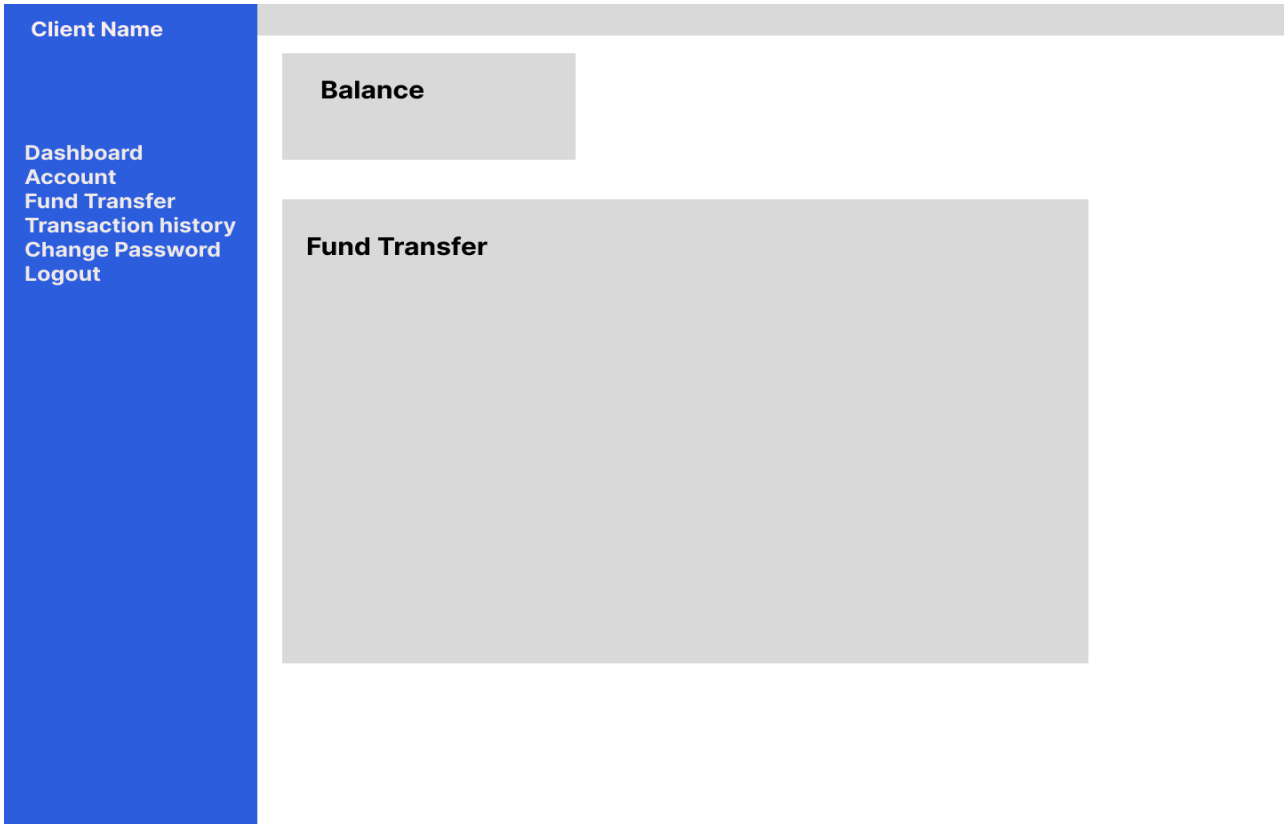


Figure 10: Wireframe for Fund Transfer

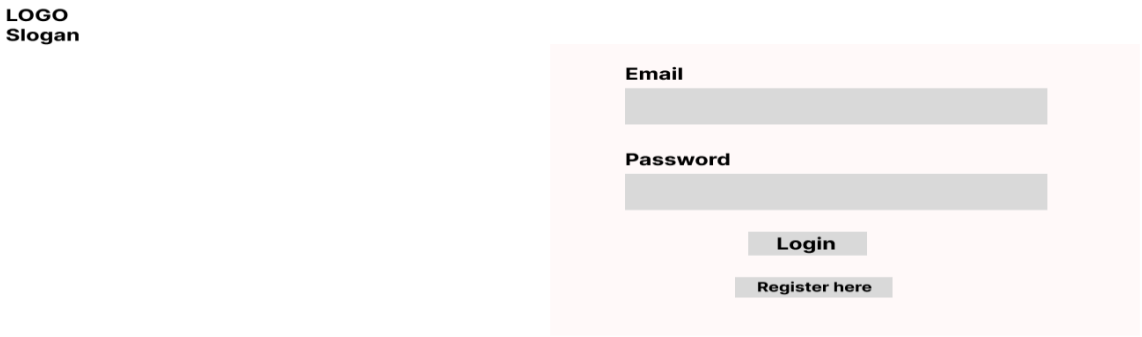


Figure 11: Wireframe for Login page

3.2.4 Physical DFD

It explains the best method to implement the business activities of the system. Moreover, it involves the physical implementation of devices and files required for the business processes. All of the below diagrams have been visualized for data processing and structured design of Online furniture shopping system and working flow.

Context Diagram

It is a basic overview of the whole system or process being analyzed or modeled. It is designed at a glance view of the product; confirm order showing the system as a single high-level process.

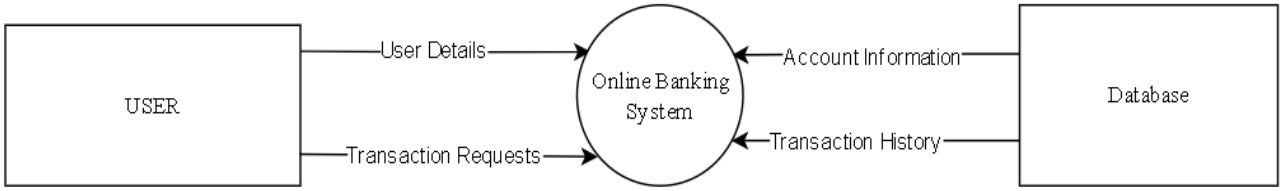


Figure 12: Context Diagram

Level 1 DFD

It shows how the system is divided into subsystems, each of which deals with one or more of the data flows to or from an external agent and which together provide all of the functionality of the system as a whole.

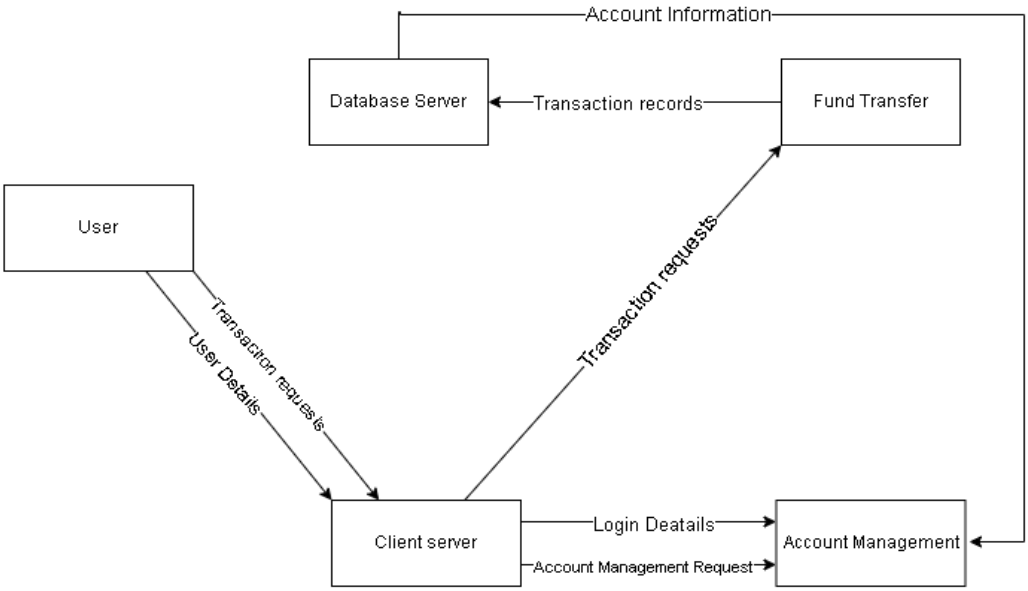


Figure 13: Level 1 DFD

Chapter 4: Implementation and Testing

4.1 Implementation

4.4.1 Tools Used

Following are the tools and framework used for accomplishment this project:

Front End Tools:

- **HTML**

We used HTML to structure the layout of web pages, define the elements (such as buttons, forms, text fields, and images), and present information to users. We also used elements such as <form>, <input>, <select>, and <button> to create forms and capture user data.

- **CSS**

CSS is used for designing different tags of html. It is also used to design different components by the help of class and id. By using css, we can control the text color, font style, the spacing between paragraphs, sizing of columns, layout designs, and many more.CSS, or Cascading Style Sheets, is a crucial technology in web development that helps in enhancing the design and presentation of a system, especially web-based systems.

- **JavaScript**

JavaScript is used for client-side validation and to make dynamic, interactive and responsive web pages. It is used to add dynamic behavior to the webpage and add special effects to the webpage.

Back End Tools:

- **PHP**

PHP is used for the backend purpose and for making dynamic web pages. It is used for server-side scripting purposes to add connectivity to the database and also used to encrypt the data, validate the user data, confirm users to go to certain pages, login pages. It also includes add, update and delete the data from the database.

Server

- **APACHE SERVER**

APACHE server is used to run php files and create fast and dynamic web pages.

Database

- **MYSQL**

MySQL is used for storing all the information required to the database in Work Progress Tracker. It is used for performing CRUD operations such as create, delete and update data from the database as requested by the user.

Documentation Tools

- **MS Office**

We have used MS Office for writing and editing the documentation of work progress tracker.

- **Draw.io**

This is used to generate diagrams for system analysis and design of Diagrams in order to save time since all components are available with drag and drop functions.

4.4.2 Implementation of Modules

There are four major modules within the Online Banking System: User Registration, Account Management, Transactions and Admin.

1. **User Registration**

   The user authentication module handles user registration, login and logout process. User's registered with their email, password and other verifications. Login functionality for the registered users with md5 secure hashing. Logout process exits the user from the system.

2. **Account Management**

   The account management module allows users to view their account, manage existing accounts and change passwords. User can view account balance, transaction history, and personal information. They can change their passwords.

3. **Transactions Module**

   The Transactions module allows users to transfer the fund from one account to another. If two accounts are from the same bank transaction can be allowed whereas if not then it's not allowed to transfer funds.

4. **Admin module**

   The admin module allows admin to manage user accounts in the system. Admin can also update the rate of bank accounts and as well as add new account numbers.

4.2 Testing

4.2.1 Test Cases for Unit Testing

Unit testing emphasizes the verification effort on the smallest unit of software design i.e.;a software component or module. Unit testing is performed in parallel with the coding phase. Unit testing tests units or modules not the whole software. I have tested each view/module of the application individually. As the modules were built up testing was carried out simultaneously, tracking out each and every kind of input and checking the corresponding output until module is working correctly

Registration:

Table 1: Test case for Registration of an user

S.N	Test Name	Input	Expected Outcome	Actual Output	Test Results
1.	Open Application	http://localhost/OnlineBanking/index.php	Login page	Login page	Pass
2.	Enter Invalid email,password	Email = aaditya123 Password = aka123	Registration Successful	Registration Failed	Pass
3.	Enter valid email, password	Email = akaaditya@gmail.com Password: Aaditya@123	Registration Successful	Registration Successful	Pass

Table 2: Test case for login of an user

S.N	Test Name	Input	Expected Outcome	Actual Output	Test Results
1.	Open Application	http://localhost/OnlineBanking/index.php	Login page	Login page	Pass
2.	Enter Invalid email,password	Email = aaditya123 Password = aka123	Login Successful, User Dashboard	Login Failed	Pass
3.	Enter valid email, password	Email = akaaditya@gmail.com Password: Aaditya@123	Login Successful, User Dashboard	Login Successful, User Dashboard	Pass

Account Management:

Table 3: Test case for Account Management

S.N	Test Name	Input	Expected Outcome	Actual Output	Test Results
1.	Enter invalid national id	National id = abc123	Update Successful	Invalid National id	Pass
2.	Enter valid name, national id, phone	Name = Aaditya Khatri National id = 123456 Phone = 9847874419	Update Successful	Update Successful	Pass

Adding of an user:

Table 4: Test case for add of an user (Success)

Test Name	User adding
Test Data	Name = Aaditya Khatri, Email = akaaditya@gmail.com, Password = Aaditya@123, Phone = 9847874419, Address = Kathmandu, National id = 123456, Account Type = Saving, Account Number = 1234567891234567
Expected result	Success
Test result	Success

Table 5: Test case for add of an user (Failure)

Test Name	User adding
Test Data	Name = Sangam837, Email = sangamsumbedi07@gmail.com, Password = sangam123, Phone = 12345678, Address = Lalitpur, National id =123456 , Account Type = Saving, Account Number = 123456789123569
Expected result	Invalid name, password, phone
Test result	Invalid name, password, phone

Fund Transfer:

Table 6: Test case for Fund Transfer

S.N	Test Name	Input	Expected Outcome	Actual Output	Test Results
1.	Enter invalid account number, invalid account name	Account Number = 123456789123a Account Name = Sangam12	Transaction Successful	Invalid account number and account name	Pass
2.	Enter invalid account number, invalid account name	Account Number = 1234567891234567 Account Name = Aaditya Khatri	Transaction Successful	Transaction Successful	Pass

4.2.1 Test Cases for System Testing

Table 7: Test Case for UI Elements

S.N	Test Name	Test Steps	Expected Outcome	Actual Output	Test Results
1.	Verify the UI elements on the dashboard page	1. Navigate to the dashboard after logging in 2. Observe the dashboard	The UI Elements should be displayed	The UI elements are displayed in the dashboard	Pass

Table 8: Test case for error handling

S.N	Test Name	Test steps	Expected Outcome	Actual Output	Test Results
1.	Verify the error handling	1. Enter an unregistered email and password in login page	Error message should be generated	Error message “Credentials do not match” is generated	Pass

Chapter 5: Conclusion and Future Recommendations

5.1 Lessons learned / Outcome

Every project makes us learn and gain knowledge in different aspects. In the following project, we have learned lots of problem-solving skills and learned things like team work, finding the solution on our own, proper use of guidelines, communication and writing skills and management of the team.

- **Teamwork**

  Since this is a team project, it teaches how to work with group members and develop the system together. We have learned how to work with a team and divide our tasks with each other and deal with the problems and errors that occur in this system.

- **Problem Solving Skills**

  From this project, we have learned lots of problem-solving skills and also learned to recognize different errors occur in this system and solve it

- **Writing Skills**

  We have learned how to prepare proposals and documentation related to projects and also learned to use different case tools for use case diagrams, schema diagrams, data flow diagrams, ER- diagrams and so on.

- **Manage time**

  The most important lesson learnt was management of time according to the complexity of the system components i.e. know which components to prioritize.

5.2 Conclusions

After the successful completion of the project, this system will help the customers to create and manage their accounts. Also this system will help the customer like payment, balance inquiry and view statements. Customers can easily fund transfer with the help of this system. With the help of this project, we learned different skills like team working, technical skills, project management, problem solving as well as management of the time.

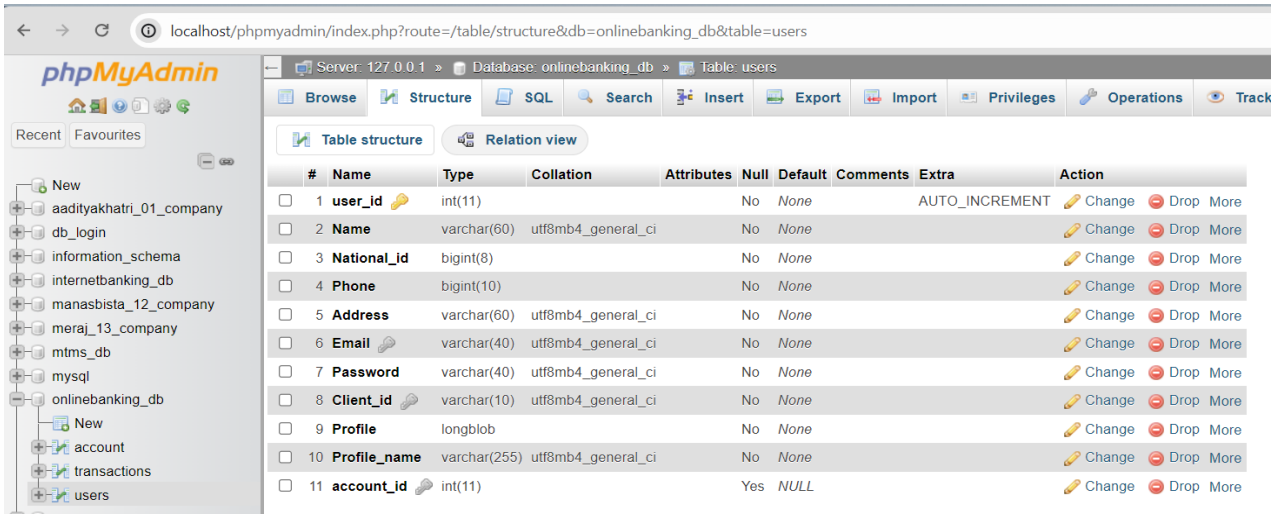
5.3 Future Recommendations

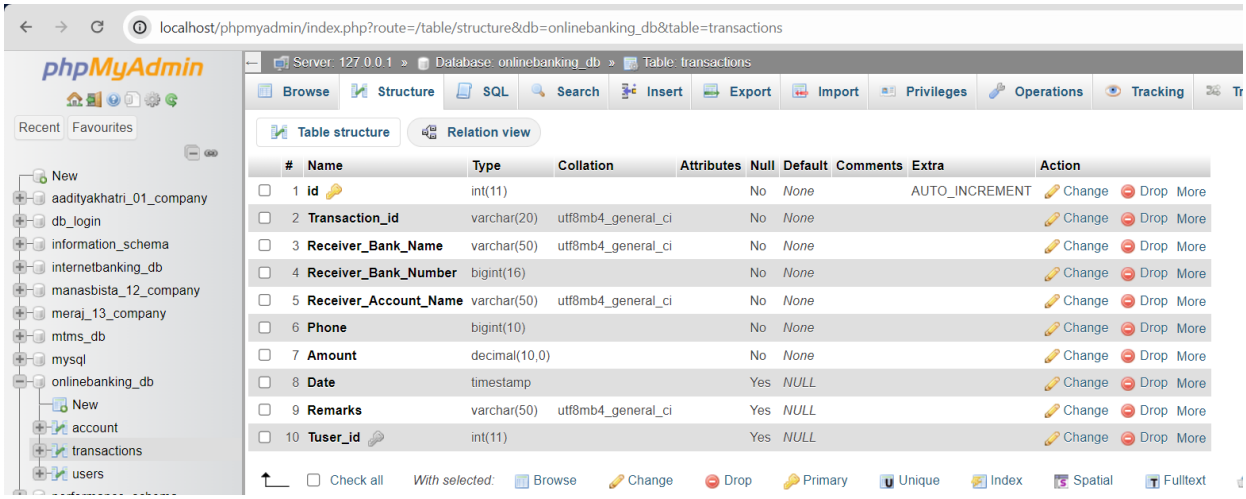
The development project could have been more efficiently handled with regards to design and development. The documentation process might have been better programming the project prior to any documentation. The system can be updated based on the user's requirements recommendation. The page load and server load speed might be improved. Some of the future recommendations are given below:

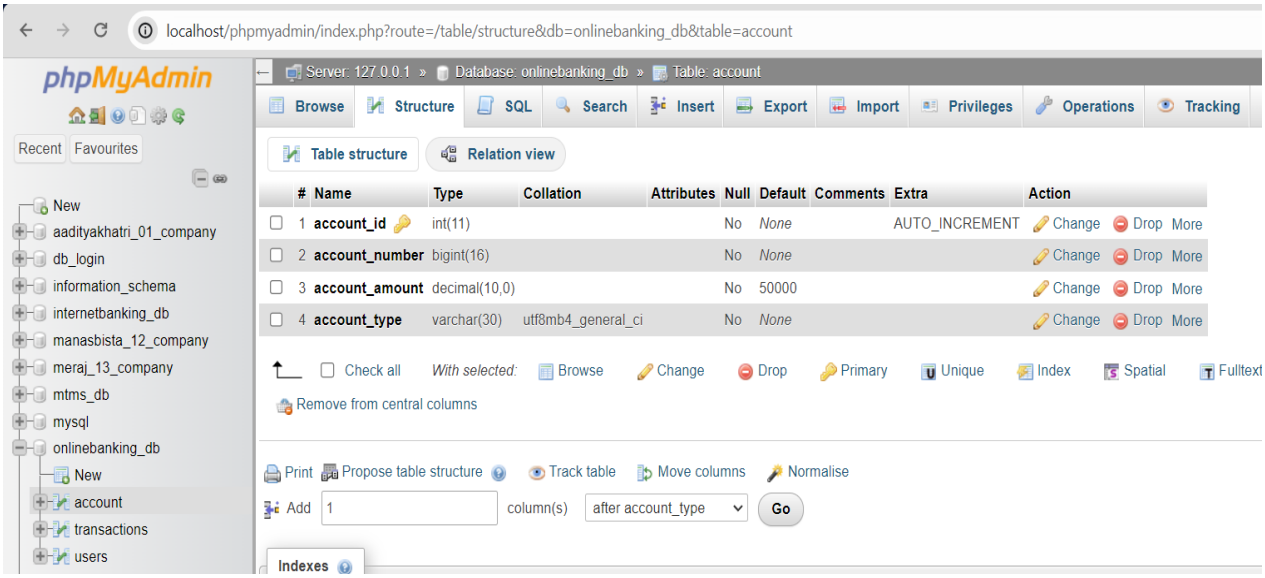
- KYC verification can be added
- Detail Report generator can be added
- More graphs can be added
- Bill payment services can be added

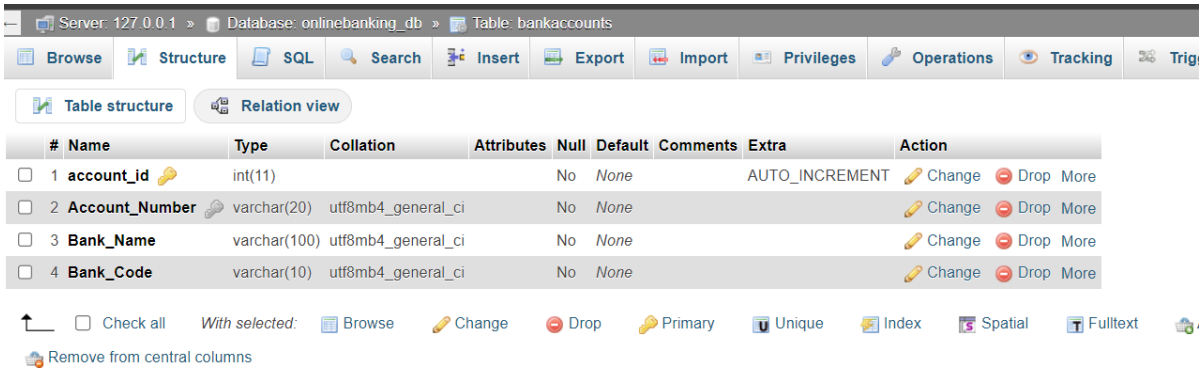
Appendices

Screenshots






localhost/phpmyadmin/index.php?route=/table/structure&db=onlinebanking_db&table=account
phpMyAdmin
Recent Favourites
New
aadityakhatri_01_company
db_login
information_schema
internetbanking_db
manasbista_12_company
meraj_13_company
mtms_db
mysql
onlinebanking_db
New
account
transactions
users
Server: 127.0.0.1 » Database: onlinebanking_db » Table: account
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking
Table structure Relation view
Name Type Collation Attributes Null Default Comments Extra Action
1 account_id int(11) No None AUTO_INCREMENT Change Drop More
2 account_number bigint(16) No None Change Drop More
3 account_amount decimal(10,0) No 50000 Change Drop More
4 account_type varchar(30) utf8mb4_general_ci No None Change Drop More
Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext
Remove from central columns
Print Propose table structure Track table Move columns Normalise
Add 1 column(s) after account_type Go
Indexes


Server: 127.0.0.1 » Database: onlinebanking_db » Table: bankaccounts
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking
Table structure Relation view
Name Type Collation Attributes Null Default Comments Extra Action
1 account_id int(11) No None AUTO_INCREMENT Change Drop More
2 Account_Number varchar(20) utf8mb4_general_ci No None Change Drop More
3 Bank_Name varchar(100) utf8mb4_general_ci No None Change Drop More
4 Bank_Code varchar(10) utf8mb4_general_ci No None Change Drop More
Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext
Remove from central columns

Register here

Name

Email

Password

Phone

Address

National ID Number

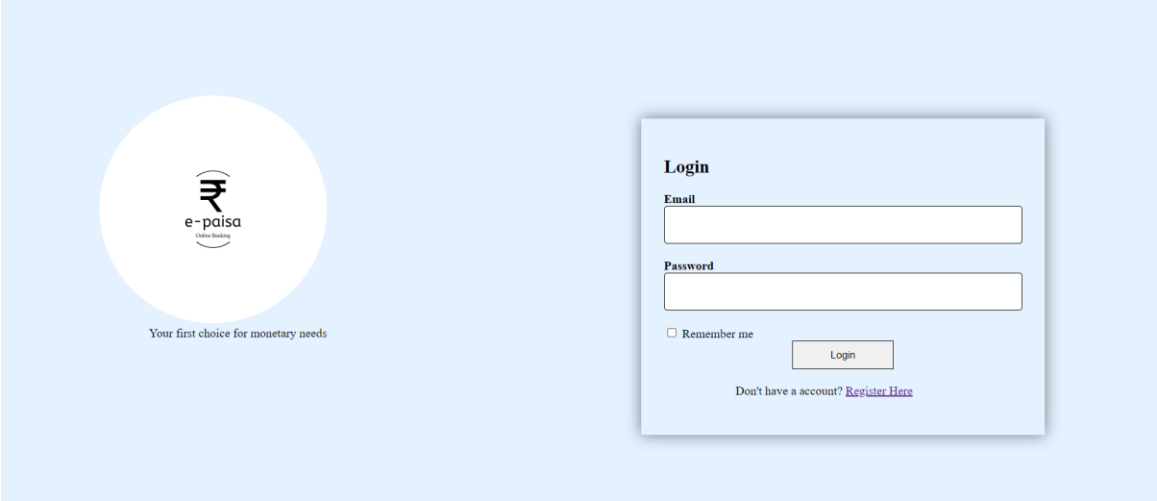
Account Type

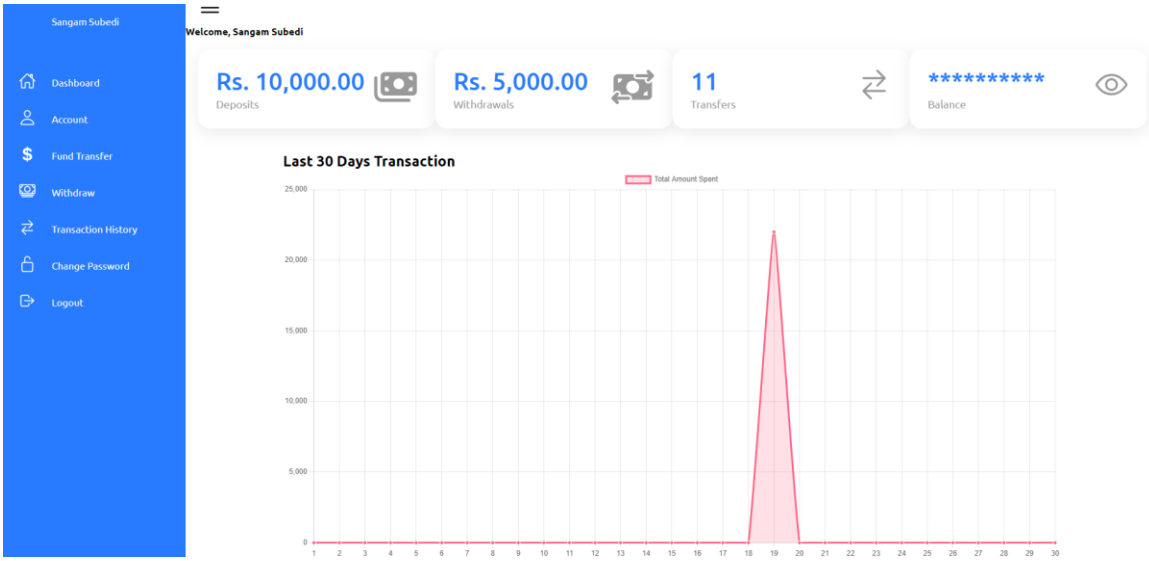
---Select Account Type---

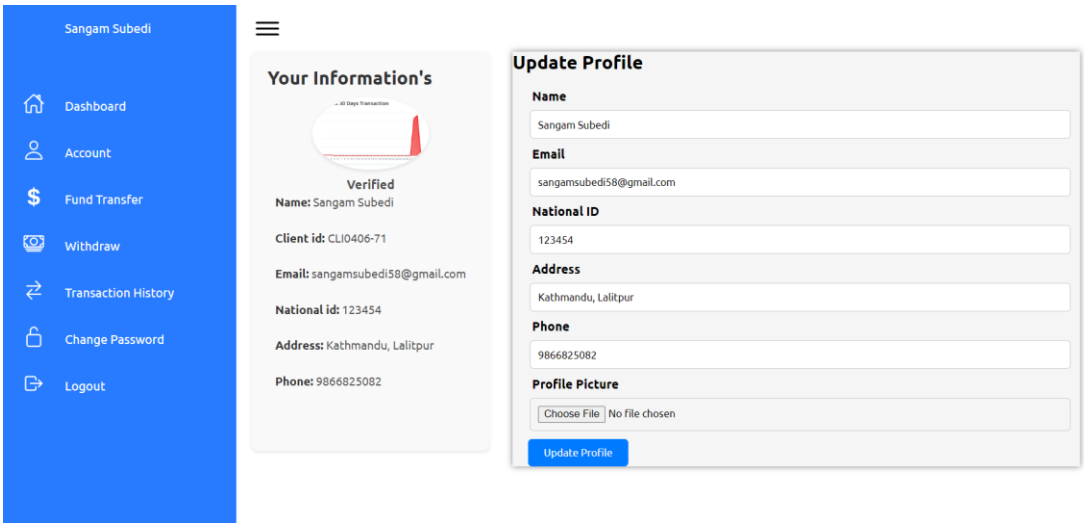
Account Number

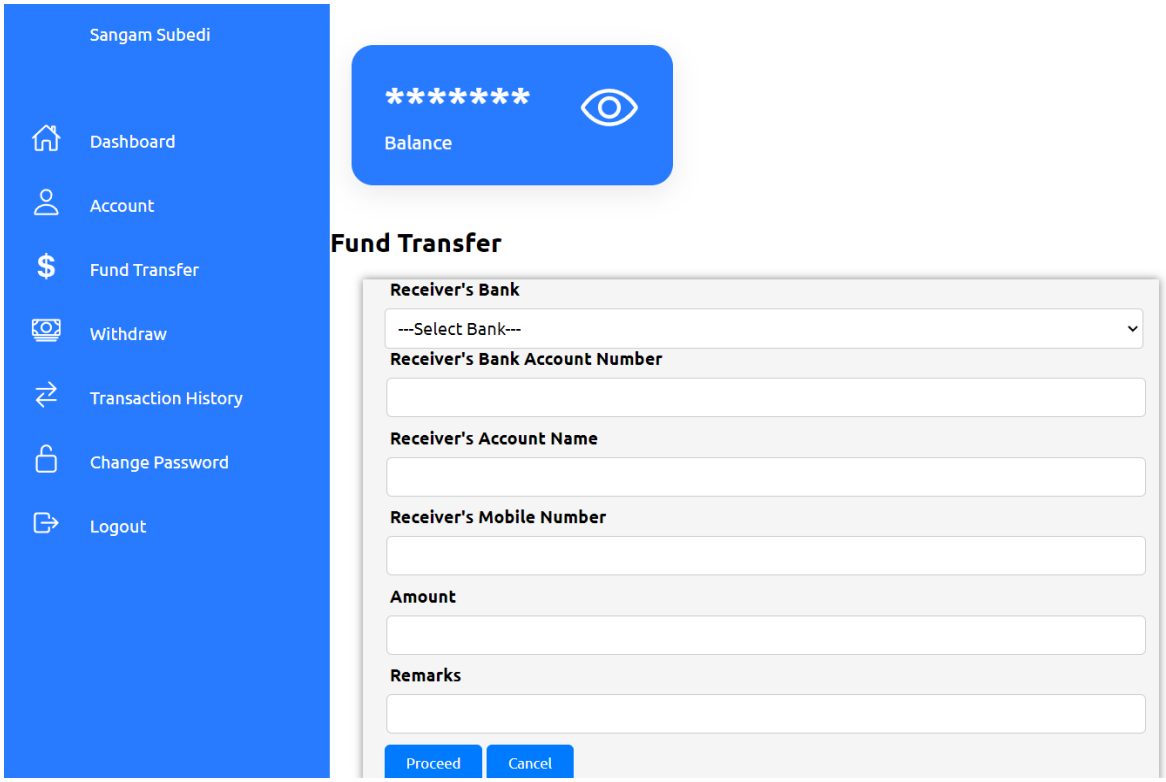
I accept terms and conditions

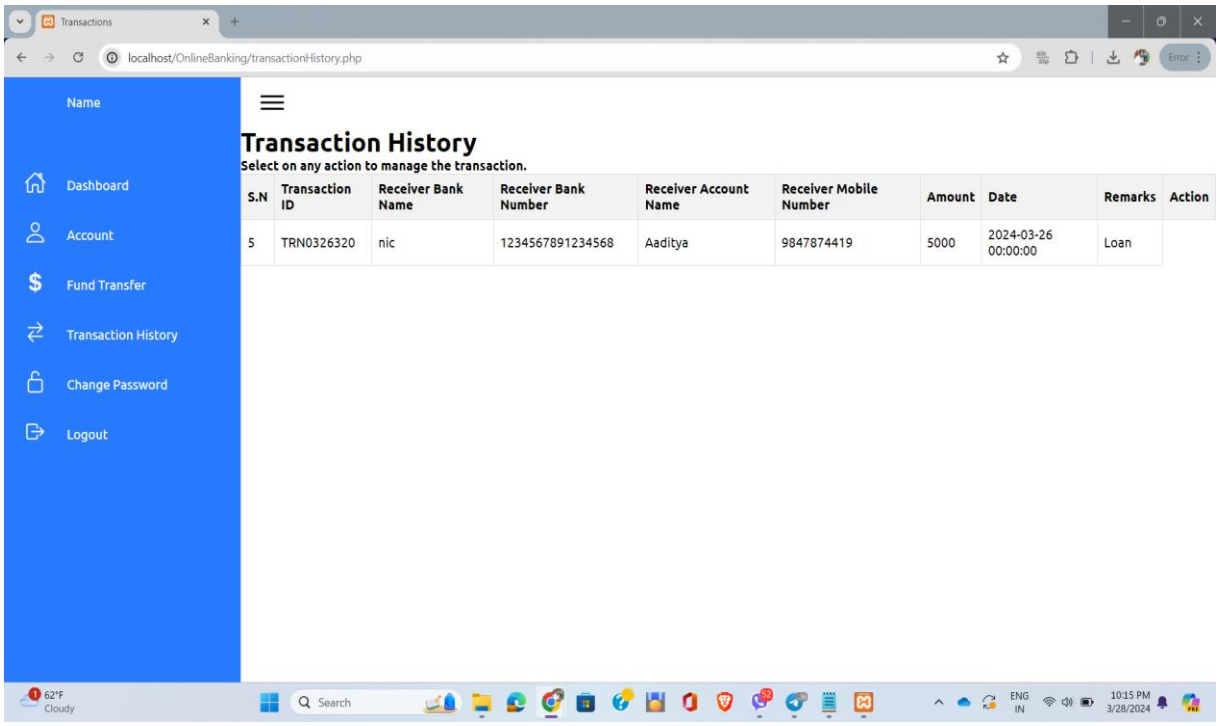
Submit Login

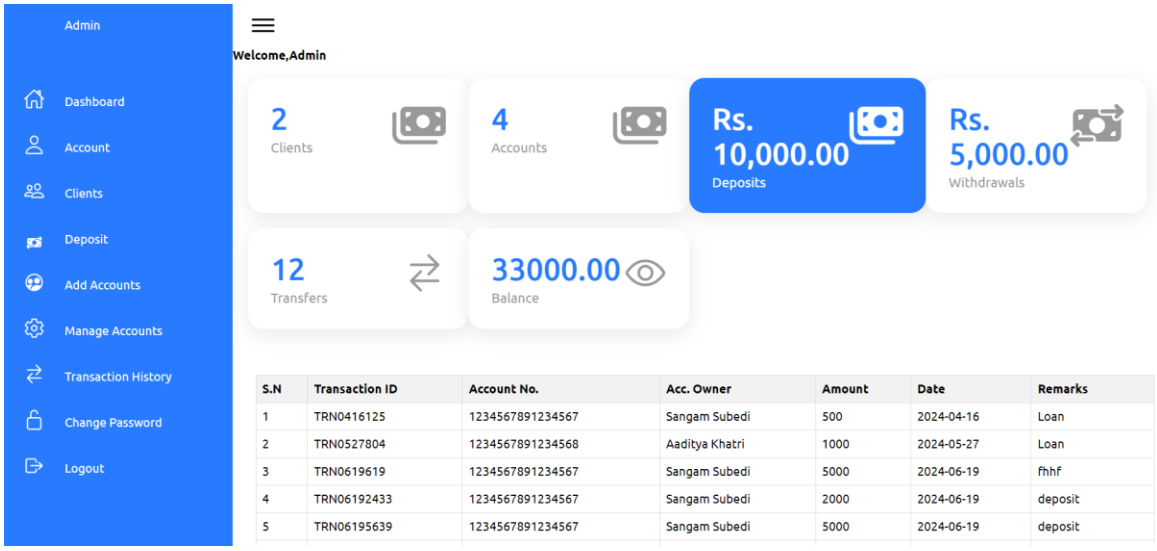

e-paisa
Online Banking
Your first choice for monetary needs
Login
Email
Password
Remember me
Login
Don't have a account? Register Here

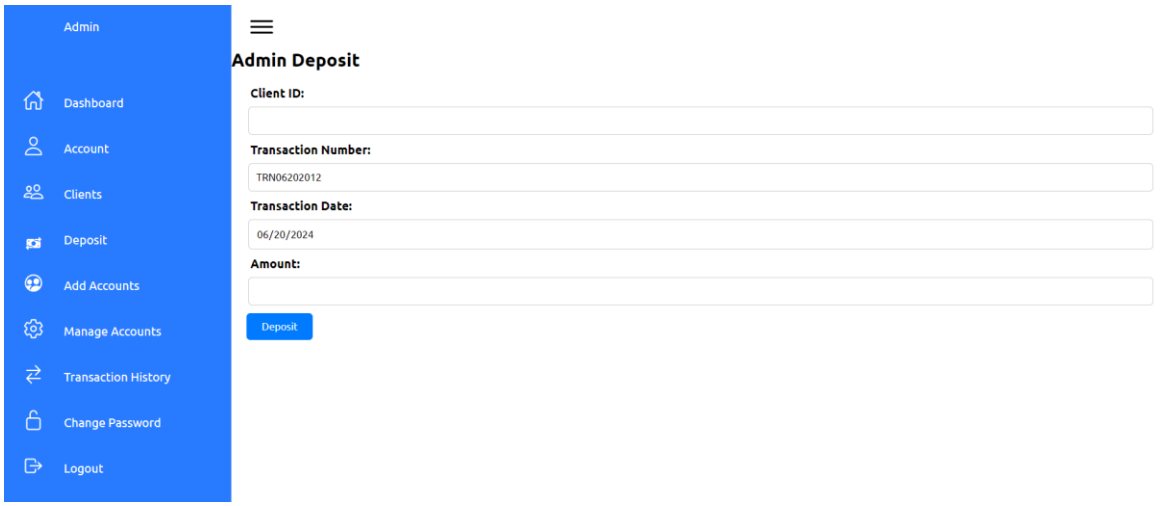

Sangam Subedi
Dashboard
Account
Fund Transfer
Withdraw
Transaction History
Change Password
Logout
Welcome, Sangam Subedi
Rs. 10,000.00
Deposits
Rs. 5,000.00
Withdrawals
11
Transfers
****
Balance
Last 30 Days Transaction
Total Amount Spent

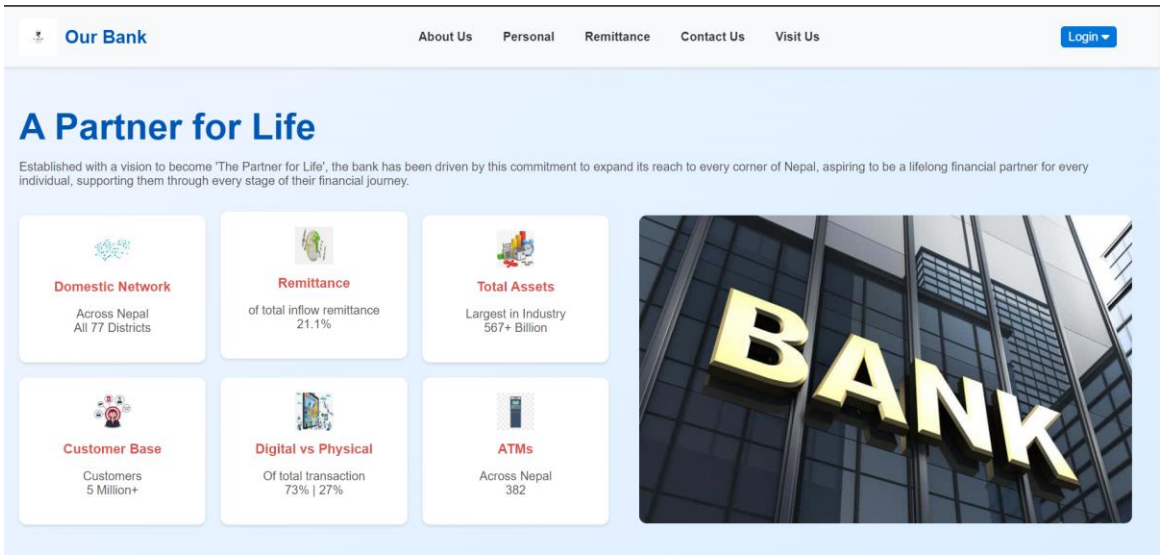

Sangam Subedi
Dashboard
Account
Fund Transfer
Withdraw
Transaction History
Change Password
Logout
Your Information's
Verified
Name: Sangam Subedi
Client id: CLI0406-71
Email: sangamsubedi58@gmail.com
National id: 123454
Address: Kathmandu, Lalitpur
Phone: 9866825082
Update Profile
Name
Sangam Subedi
Email
sangamsubedi58@gmail.com
National ID
123454
Address
Kathmandu, Lalitpur
Phone
9866825082
Profile Picture
Choose File
No file chosen
Update Profile


Sangam Subedi
Dashboard
Account
Fund Transfer
Withdraw
Transaction History
Change Password
Logout
****
Balance
Fund Transfer
Receiver's Bank
---Select Bank---
Receiver's Bank Account Number
Receiver's Account Name
Receiver's Mobile Number
Amount
Remarks
Proceed
Cancel


Transactions
localhost/OnlineBanking/transactionHistory.php
Name
Dashboard
Account
Fund Transfer
Transaction History
Change Password
Logout
Transaction History
Select on any action to manage the transaction.
S.N
Transaction ID
Receiver Bank Name
Receiver Bank Number
Receiver Account Name
Receiver Mobile Number
Amount
Date
Remarks
Action
5
TRN0326320
nic
1234567891234568
Aaditya
9847874419
5000
2024-03-26 00:00:00
Loan


Admin
Dashboard
Account
Clients
Deposit
Add Accounts
Manage Accounts
Transaction History
Change Password
Logout
Welcome,Admin
2
Clients
4
Accounts
Rs. 10,000.00
Deposits
Rs. 5,000.00
Withdrawals
12
Transfers
33000.00
Balance
S.N Transaction ID Account No. Acc. Owner Amount Date Remarks
1 TRN0416125 1234567891234567 Sangam Subedi 500 2024-04-16 Loan
2 TRN0527804 1234567891234568 Aaditya Khatri 1000 2024-05-27 Loan
3 TRN0619619 1234567891234567 Sangam Subedi 5000 2024-06-19 fhhf
4 TRN06192433 1234567891234567 Sangam Subedi 2000 2024-06-19 deposit
5 TRN06195639 1234567891234567 Sangam Subedi 5000 2024-06-19 deposit


Admin
Dashboard
Account
Clients
Deposit
Add Accounts
Manage Accounts
Transaction History
Change Password
Logout
Admin Deposit
Client ID:
Transaction Number:
TRN06202012
Transaction Date:
06/20/2024
Amount:
Deposit


Our Bank
About Us
Personal
Remittance
Contact Us
Visit Us
Login
A Partner for Life
Established with a vision to become 'The Partner for Life', the bank has been driven by this commitment to expand its reach to every corner of Nepal, aspiring to be a lifelong financial partner for every individual, supporting them through every stage of their financial journey.
Domestic Network
Across Nepal
All 77 Districts
Remittance
of total inflow remittance
21.1%
Total Assets
Largest in Industry
567+ Billion
Customer Base
Customers
5 Million+
Digital vs Physical
Of total transaction
73% | 27%
ATMs
Across Nepal
382
BANK

Source Codes

Login:

login.php

```php
<?php
session_start();
$err = [];
if(isset($_POST['login'])){
    require_once '../connection.php';
    if(isset($_POST['email']) && !empty($_POST['email']) && trim($_POST['email'])){
        $email = $_POST['email'];
        if(!filter_var($email, FILTER_VALIDATE_EMAIL)){
            $err['email'] = 'Please enter valid email';
        }
    } else{
        $err['email'] = 'Please enter email';
    }

    if(isset($_POST['password']) && !empty($_POST['password'])){
        $password = $_POST['password'];
        $encrypted_password = md5($password);
    } else{
        $err['password'] = 'Please enter password';
    }

    if(count($err) == 0){
        $sql = "SELECT user_id,name,email FROM users WHERE email = '$email' AND password = '$encrypted_password'";
        $result = $connection->query($sql);
        if($result->num_rows == 1){
            $row = $result->fetch_assoc();
            $_SESSION['admin_id'] = $row['user_id'];
            $_SESSION['admin_name'] = $row['name'];
            $_SESSION['admin_email'] = $row['email'];
            if(isset($_POST['remember'])){
                setcookie('admin_id',$row['user_id'],time()+86400*7);
                setcookie('admin_name',$row['name'],time()+86400*7);
                setcookie('admin_email',$row['email'],time()+86400*7);
            }
            header('Location:dashboard.php');
        } else{
            $msg = 'Credentials not match';
        }
    }
}
?>
```

```php
<h2>Register here</h2>
<form action="<?php echo $_SERVER['PHP_SELF'] ?>" method="post">
    <div>
        <div class="form-box">
            <label for="name">Name</label>
            <input type="text" name="name" id="name"/>
            <?php echo isset($err['name'])?$err['name']:''; ?>
        </div>
        <div class="form-box">
            <label for="email">Email</label>
            <input type="email" name="email" id="email"/>
            <?php echo isset($err['email'])?$err['email']:''; ?>
        </div>
        <div class="form-box">
            <label for="password">Password</label>
            <input type="password" name="password" id="password"/>
            <?php echo isset($err['password'])?$err['password']:''; ?>
        </div>
        <div class="form-box">
            <label for="phone">Phone</label>
            <input type="tel" name="phone" id="phone"/>
            <?php echo isset($err['phone'])?$err['phone']:''; ?>
        </div>
        <div class="form-box">
            <label for="address">Address</label>
            <input type="text" name="address" id="address"/>
            <?php echo isset($err['address'])?$err['address']:''; ?>
        </div>
        <div class="form-box">
            <label for="nationalID">National ID Number</label>
            <input type="number" name="nationalID" id="nationalID"/>
            <?php echo isset($err['nationalID'])?$err['nationalID']:''; ?>
        </div>
        <div class="form-box">
            <label for="accType">Account Type</label>
            <select name="accType" id="accType">
                <option value="">---Select Account Type---</option>
                <option value="saving" <?php if($accType === "saving")echo "selected" ?>>Saving</option>
                <option value="current" <?php if($accType === "current")echo "selected" ?>>Current</option>
            </select>
            <?php echo isset($err['accType'])?$err['accType']:''; ?>
        </div>
```

```php
$user_id = $_SESSION['admin_id'];
$m = date('m');
$d = date('d');
$Tid = "TRN{$m}{$d}" . rand(100, 999);
$current_date = date('Y-m-d');
$sql = "INSERT INTO transactions(Transaction_id, Receiver_Bank_Name, Receiver_Bank_Number, Receiver_Account_Name,
Phone, Amount, Date, Remarks, Tuser_id)
    VALUES ('$Tid', '$receiverBank', '$receiverBankAcNo', '$receiverAcName', '$phone', '$amount',
    '$current_date', '$remarks', '$user_id')";

mysqli_begin_transaction($connection);
if (mysqli_query($connection, $sql)) {
    $update_query_sender = "UPDATE Users SET Amount = Amount - $amount WHERE user_id = $user_id";
    $sender_update_result = mysqli_query($connection, $update_query_sender);

    // Update receiver's account balance
    $update_query_receiver = "UPDATE Users SET Amount = Amount + $amount WHERE Account_Number = '$receiverBankAcNo'";
    $receiver_update_result = mysqli_query($connection, $update_query_receiver);

    // Commit the transaction if all queries succeed
    if ($sender_update_result && $receiver_update_result) {
        mysqli_commit($connection);
        $msg = 'Transaction Successful. Money transferred successfully.';
    } else {
        // Rollback the transaction if any query fails
        mysqli_rollback($connection);
        $msg = 'Transaction failed. Please try again.';
    }
} else {
    mysqli_rollback($connection);
    $msg = 'Transaction failed. Please try again.';
}
```

```
<table cellspacing=0>
    <thead>
        <tr>
            <th>S.N</th>
            <th>Transaction ID</th>
            <th>Account No.</th>
            <th>Acc. Owner</th>
            <th>Amount</th>
            <th>Date</th>
            <th>Remarks</th>
        </tr>
    </thead>
    <tbody>
        <?php
        $count = 1;
        if ($result->num_rows > 0) {
            while ($row = $result->fetch_assoc()) {
                echo "<tr>";
                echo "<td>" . $count . "</td>";
                echo "<td>" . $row['Transaction_id'] . "</td>";
                echo "<td>" . $row['Acc_Number'] . "</td>";
                echo "<td>" . $row['Acc_Owner'] . "</td>";
                echo "<td>" . $row['Amount'] . "</td>";
                echo "<td>" . $row['Date'] . "</td>";
                echo "<td>" . $row['Remarks'] . "</td>";
                echo "</tr>";
                $count++;
            }
        } else {
            echo "<tr><td colspan='10'>No transactions found</td></tr>";
        }
        ?>
    </tbody>
</table>
```

```
<?php
session_start();
require_once '../connection.php';
if (!isset($_SESSION['admin_id'])) {
    header('location:login.php?err=1');
    exit;
}

$user_id = $_SESSION['admin_id'];
$sql = "SELECT transactions.*, users.Account_Number AS 'Acc_Number', users.Name AS 'Acc_Owner'
        FROM transactions
        INNER JOIN users ON transactions.Tuser_id = users.user_id";
$result = $connection->query($sql);

if (!$result) {
    trigger_error('Invalid query: ' . $connection->error);
}
?>
```

References

[1] B. Acharya, "DITALIZATION IN BANKING IN NEPAL," 2023.

[2] G. V. K. E. R. L. Krishnan Dandapani, "Internet banking services and credit union performance," *Managerial Finance*, vol. Vol. 34 No. 6, pp. pp. 437-446, 2008.

[3] A. M. Hyde, "Pimrindore," October 2015. [Online]. Available: https://www.pimrindore.ac.in/vol2%2Cissue2/Dr.Hyde.pdf.