****

**Tribhuvan University**

**Faculty of Humanities and Social Science**

**BASIC BANKING PROTOTYPE**

A PROJECT REPORT

**Submitted to**

**Department of Computer Application**

**Jaya Multiple Campus**

**Makalbari, Kathmandu**

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

**Submitted by**

Santosh Dahal

BCA 4th Semester

TU Reg. No.: 6-2-721-21-2021

Under the Supervision of

**Mukti Thapa**

****

**Tribhuvan University**

**Faculty of Humanities and Social Science**

**Jaya Multiple Campus**

Makalbari, Kathmandu

Bachelor in Computer Application

**SUPERVISOR’S RECOMMENDATION**

I hereby recommend that this project prepared under my supervision by **Santosh Dahal** entitled **“Basic Banking Prototype**” in the Partial Fulfillment of requirement for the degree of Bachelor in Computer Application is recommended for that final evaluation.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Mukti Thapa

Project Supervisor

BCA Department

Jaya Multiple Campus

****

**Tribhuvan University**

**Faculty of Humanities and Social Science**

**Jaya Multiple Campus**

Makalbari, Kathmandu

Bachelor in Computer Application

**LETTER OF APPROVAL**

This is certified that this project prepared by **Santosh Dahal** entitled “**Basic Banking Prototype**” in the Partial Fulfillment of requirement 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.

­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Mukti Thapa Subash Bista

Project Supervisor H.O.D

BCA Department Jaya Multiple Campus

Jaya Multiple Campus Makalbari, Kathmandu

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**External Examiner Internal Examiner**

**Mukti Thapa**

**ABSTRACT**

This document details the design and development of the Basic Banking Prototype, a database project employing PHP, HTML, CSS, and JS. Designed to enhance traditional banking functions, this prototype simulates standard bank operations while introducing improvements that streamline digital transactions and user interactions. The project aims to create a web-based platform that enhances efficiency in essential banking tasks, thereby reducing human error, manpower requirements, and transaction times. This leads to greater customer satisfaction by addressing common inefficiencies in daily banking operations. Operating on a Windows platform, the prototype uses MySQL for database management and XAMPP for server operations, ensuring a stable and secure environment for banking activities. With an integration of technologies like PHP, MySQL, HTML, CSS, and JavaScript, the prototype offers a reliable, scalable, and user-friendly banking experience. The Basic Banking Prototype extends beyond traditional functionalities to include advanced account management, automated fund transfers, and real-time balance updates, promoting a more comprehensive and flexible banking experience. This prototype represents a significant step forward in digital banking, aligning with technological trends and meeting modern customers' evolving needs.

**ACKNOWLEDGEMENT**

I would like to extend my deepest gratitude to my project supervisor, **Mukti Thapa** and BCA Coordinator **Subash Bista** whose guidance was invaluable in the conceptualization and execution of my project on **"Basic Banking Prototype"**. His expertise and insights were instrumental in shaping my approach and enabling me to explore innovative aspects of digital banking solutions.

I am also immensely thankful to the Campus Chief, **Mr. Bhawani Prasad Poudel**, for his unwavering support and for granting me the permission needed to undertake this significant project. His encouragement has been a great motivator throughout my journey. My sincere appreciation goes to Jaya Multiple Campus and specifically to the Department of Computer Application for providing the resources, technical support, and environment that fostered my research and development efforts. The continuous advice and feedback from the faculty members enriched my learning experience significantly.

I must express my gratitude towards the fellow students of Jaya Multiple Campus for their cooperation and encouragement, which played a crucial role in the successful completion of my project. A special thanks to my family and friends, who have been my constant source of support and encouragement. Their understanding and patience have been a great help in completing this project within the allocated timeframe.

Lastly, I would like to acknowledge Tribhuvan University and the Faculty of Humanities and Social Science for offering me the platform and curriculum that introduced me to the intricacies of project handling and implementation in the field of Computer Application, which significantly enhanced my knowledge and skill set.

Yours sincerely,

Santosh Dahal

**Table of Content**

**SUPERVISOR’S RECOMMENDATION**

**LETTER OF APPROVAL**

**ABSTRACT ----------------------------------------------------------------------------------------- i**

**ACKNOWLEDGEMENT ----------------------------------------------------------------------- ii**

**LIST OF ABBERVATIONS--------------------------------------------------------------------- v**

**LIST OF FIGURES ------------------------------------------------------------------------------ vi**

**LIST OF TABLES--------------------------------------------------------------------------------vii**

**CHAPTER 1: INTRODUCTION ---------------------------------------------------------------1**

* 1. Introduction……………………………………………………….…..…...1
  2. Problem Statement…………………………………………………...……1
  3. Objectives…………………………………………………………....…….2
  4. Scope and limitations………………………………………..……….……2

**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

3.1.2 Feasibility Analysis………………………………………….……….8

3.1.3 Data Modeling (ER-Diagram)………………………...………….…..9

3.1.4 Process Modeling(DFD)……………………………………..……...11

3.2 System Design………………………………………………..……….….12

3.2.1 Architectural Design………………………………………..……….12

3.2.2 System Flowchart………………………………………..………….13

3.2.3 Database Schema Design…………………………………..……….15

3.2.4 Interface Design……………………………………….……….…...16

**CHAPTER 4: IMPLEMENTITION AND TESTING------------------------------------ 17**

4.1 Implementation………………………………………/……….…………17

4.1.1 Tools Used…………………………………………………..……...17

4.1.2 Implementation Details and Modules……………………...….…….18

4.2 Testing…………………………………………………………...……….20

4.2.1 Purpose of Testing………………………………………..…….…...20

4.2.2 Test Case for Unit Testing…………………………………………...21

A)Test Case 1: Staff………………………………………...……….21

A.1 Staff Login Page………………………………………..……21

A.2 Approve Customer…………………………..……..…..……21

A.3 Delete Customer……………………………………..…..…..22

A.4 Credit Customer…………………………………...…..…….22

A.5 View Active Customer…………………………...……...…..22

A.6 Edit Customer Details…………………………………..…...23

B) Test Case 2: Customer…………………………..……………...23

B.1 Customer Login Page……………………………….…..….23

B.2 Apply for Debit Card……………………………..…..…….23

B.3 Register E-Banking…………………………………..…….24

B.4 Add Beneficiary………………………………..…………..24

B.5 Fund Transfer………………..……………..………………24

**CHAPTER 5: CONCLUSION AND FUTURE RECOMMENDATIONS-------------25**

5.1 Lesson Learn……………………………………..………………….……25

5.2 Conclusion……………………………………………………..…………25

5.3 Future Recommendation……………………………………………..…..26

**REFERANCE--------------------------------------------------------------------------------------27**

**LIST OF ABBREVIATIONS**

CRUD Create, Read, Update and Delete

CSS Cascading Style Sheet

DFD Data Flow Diagram

ERD Entity Relationship Diagram

HTML Hypertext Markup Language

JS Java Script

MySQL Microsoft Service Structured Query Language

OTP One Time Password

PHP **Hypertext Preprocessor**

UI User Interface

**LIST OF FIGURES**

Figure 3.1: Waterfall Methodology for Basic Banking Prototype…………...…….……..6

Figure 3.2: Use Case Diagram for Basic Banking Prototype…………………….……….7

Figure 3.3: Gantt Chart for Basic Banking Prototype……...……………………………...9

Figure 3.4: E-R Diagram for Basic Banking Prototype……………………………….…10

Figure 3.5: Level 0 DFD for Basic Banking Prototype…………………………..………11

Figure 3.6: Level 1 DFD for Basic Banking Prototype……………………………..……12

Figure 3.7: Architectural Design for Basic Banking Prototype………………….…..…..12

Figure 3.8: Flowchart of Basic Banking Prototype for Staff……………………..……...14

Figure 3.9: Flowchart of Basic Banking Prototype for Customer…………….……..…..14

Figure 3.10: Database Schema Diagram for Basic Banking Prototype…………..……...15

Figure 3.11: Main Page for Basic Banking Prototype………………..……...…………..11

Figure 3.12: Login page for Basic Banking Prototype……………………….……….…11

**LIST OF TABLES**

Table 3.1: Gantt Chart Table for Basic Banking Prototype………..……………...………9

Table 4.1: Staff Login Page of Basic Banking Prototype…………………………..…....21

Table 4.2: Approve Customer Account of Basic Banking Prototype …………….……..21

Table 4.3: Delete user account of Basic Banking Prototype……………………….….…22

Table 4.4: Credit Customer Account of Basic Banking Prototype …………….………..22

Table 4.4: View Active Customer of Basic Banking Prototype ………………...…...….22

Table 4.5: View Active Customer Account of Basic Banking Prototype…………..……27

Table 4.6: Edit Customer Details of Basic Banking Prototype…………………..….…...28

Table 4.7: Customer login page of Basic Banking System…………………………..…..28

Table 4.8: Apply for Debit Card of Basic Banking System…………………….………..28

Table 4.9: Register E-Banking of Basic Banking Prototype……………………….…….29

Table 4.10: Add Beneficiary of Basic Banking Prototype……………………..……...…29

Table 4.11: Fund Transfer of Basic Banking Prototype………………………....……….30

**CHAPTER 1**

**INTRODUCTION**

* 1. **Introduction**

The Basic Banking Prototype is a sophisticated database application designed to simulate the operations of a banking account system using web technologies like PHP, HTML, CSS, and JavaScript. This enhances the interface and ensures smooth user interactions, demonstrating the functionalities of an online banking system and streamlining the management of a person's bank account in a digital environment. The project integrates modern web technologies to deliver a seamless and intuitive banking experience, mirroring traditional banking's complexity and robustness with digital convenience.

This prototype offers a comprehensive suite of digital banking services, including account creation, transaction management, balance inquiries, and customer interactions, all within a secure framework adhering to industry-standard protocols. It ensures data security and transaction integrity, fostering user trust and confidence. Moreover, by employing responsive web design principles, the prototype guarantees accessibility across various devices, providing a consistent experience whether accessed via desktop, tablet, or smartphone. An administrative module for bank personnel is included, facilitating user account management, transaction monitoring, and report generation. This dual functionality streamlines operations and enhances service delivery, demonstrating how traditional banking can be transformed with modern technology to provide a scalable, efficient, and user-friendly solution.

* 1. **Problem Statement**

In the face of evolving digital demands, traditional banking systems are challenged by slow processing times, accessibility issues, and outdated technology, leading to decreased customer satisfaction and increased operational costs. These systems often require physical presence, limiting accessibility for customers in remote or underserved areas, and are hampered by security vulnerabilities that compromise data integrity and customer trust.

The Basic Banking Prototype addresses these challenges by leveraging advanced web technologies to enhance the functionality and security of traditional banking. This digital solution offers comprehensive, remote banking services through a user-friendly platform, eliminating the need for physical bank visits and integrating robust security measures to protect user data. By providing a responsive and secure online banking experience, this prototype aims to meet the demands of modern consumers and improve the efficiency of banking operations.

* 1. **Objective**

The objective of project is:

1. To Provide remote banking services to users, ensuring accessibility for customers in all regions.
2. To Streamline banking processes, reducing processing times and operational costs.
3. To Design an intuitive interface for seamless banking experiences.
   1. **Scope and Limitations**
      1. **Scope**
4. User Account Management: Allowing users to create, modify, and manage their banking accounts securely.
5. Transaction Handling: Facilitating various transactions such as fund transfers, and account inquiries.
6. Security Features: Implementing encryption protocols, user authentication mechanisms, and data validation to ensure the security and integrity of user information.
7. Cross-Platform Compatibility: Ensuring the platform is accessible across multiple devices and web browsers to accommodate diverse user preferences.
   * 1. **Limitations**
8. Limited Functionality: The prototype may not encompass all features and functionalities of a comprehensive banking system, such as investment management or loan processing.
9. Security Risks: While security measures are implemented, the prototype may still be vulnerable to cyber threats and data breaches, especially if not regularly updated and maintained.
10. Scalability Constraints: The prototype may face challenges in scaling up to accommodate a large user base or increasing transaction volumes, leading to potential performance issues.
11. Regulatory Compliance: Compliance with banking regulations and standards may not be fully addressed in the prototype, requiring additional validation and adjustments to meet legal requirements.
12. Dependency on Third-Party Services: The prototype may rely on external services or APIs for certain functionalities, which could introduce dependencies and potential points of failure.

**CHAPTER:2**

**BACKGROUND STUDY AND LITERATURE REVIEW**

**2.1 Background Study**

Online Banking Systems enable customers to perform financial transactions over the internet, providing 24/7 access to banking services. Users can check balances, transfer funds, and manage beneficiary accounts without visiting a bank branch. The main objectives are to provide hassle-free banking services, enhance banking operations digitally, and ensure secure and user-friendly access to banking features.

A case study of Nabil Bank's nBank system highlights these objectives. The system includes mobile check deposit, customizable alerts, and a focus on security through biometric logins and fraud monitoring. Key features include mobile banking via smartphone, integrated bill payment, customizable alerts, and 24/7 customer support.

Research in online banking emphasizes user experience, security measures, and intelligent tools. Enhancing user experience involves intuitive navigation and personalized dashboards. Security measures like encryption, two-factor authentication, and security audits prevent fraud. Intelligent tools streamline data entry and record-keeping for efficient account and transaction management.

The Basic Banking Prototype incorporates these principles to create a secure, efficient, and user-friendly system. In this prototype, there are two actors: staff and customer. Staff can log in, approve customers, view customers, delete customers, deposit to accounts, add beneficiaries, remove beneficiaries, transfer balances, change passwords, edit details, view statements, and log out. Similarly, customers can log in, view customers, deposit to accounts, add beneficiaries, remove beneficiaries, transfer balances, change passwords, edit details, view statements, and log out. This prototype aims to provide a streamlined and secure online banking experience for both staff and customers.

**2.2 Literature Review**

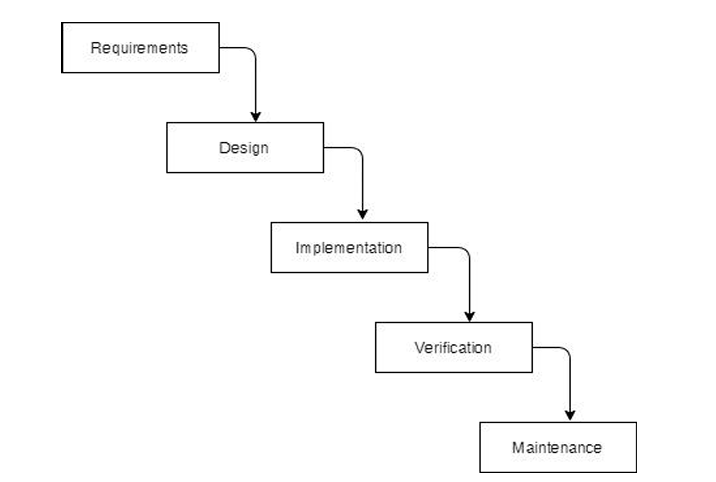
* In research by **Junaid Ashraf titled “Design and Implementation of an Online Banking System in PHP,”** it identifies the essential features and user requirements of an online banking system, focusing on internet banking functionalities. The aim of this study was to explore how user-friendly interfaces and secure transaction methods can enhance user satisfaction and trust in digital banking services.[1]
* In research by **Priya Sharma and Ankit Patel titled “Secure Online Transactions in Banking Systems: A Review,”** it defines the security measures necessary for online banking, emphasizing encryption, two-factor authentication, and regular security audits. This study highlights the importance of protecting user data and maintaining the integrity of financial transactions to prevent fraud and unauthorized access.[2]
* In research by **Kiran Kumar and Suman Reddy titled “User Experience in Online Banking: A Comparative Analysis,”** it contributes to understanding the impact of user interface design on the overall user experience in online banking platforms. This research argues that intuitive navigation, responsive design, and personalized dashboards significantly improve user engagement and satisfaction.[3]
* In research by **nbank Nabil Bank** it explores the application of intelligent support tools in managing digital resources. This study's findings can be related to the management of beneficiary accounts and transaction records in online banking systems, suggesting that automated tools can streamline data entry and record-keeping processes.[4]

**CHAPTER:3**

**SYSTEM ANALYSIS AND DESIGN**

**3.1 System Analysis**

This system is designed with the series of processes starting with requirement analysis, design, implementation, testing and maintenance. During requirement analysis, all the functional and nonfunctional requirement are analyzed and system is developed according to the requirement then designing of the system is carried out. After the design process, coding and development part is started then after integrating the system there is testing of the system. If the testing is positive then system is implemented otherwise some maintenance is done and system come in operation.



**3.1.1 Requirement Analysis**

The requirements are to be collected before starting projects’ development life cycle. To design and develop system, functional as well as non-functional requirement of the system has been studied.

1. **Functional Requirement**

Different functional requirement of the system has been identified and are listed as below:

1. **Hardware Requirements**

To successfully develop and use the baking project, there are specific hardware requirements for both the developer side and client side.

**Figure 3.1: Waterfall Methodology for Basic Banking Prototype**

* + 1. **Requirements Analysis**

1. **Functional Requirements**

Different functional requirement of the system has been identified and are listed as below:

**For Staff:**

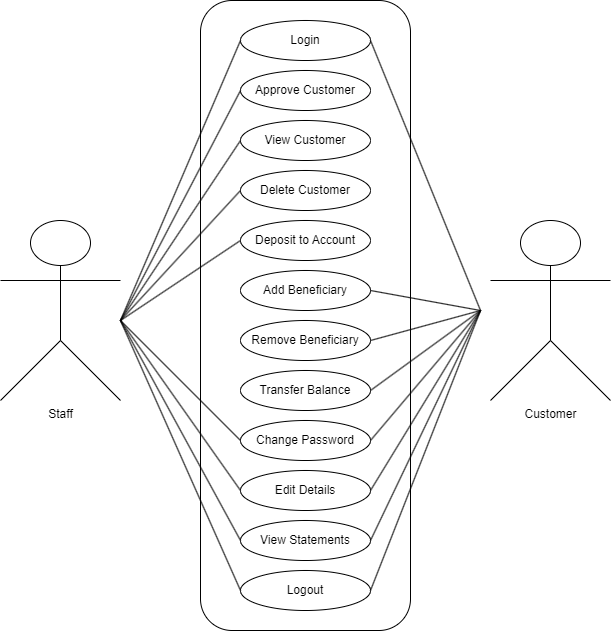
* The system should allow the system administrator to login and logout from the system.
* The system should allow the system administrator to add, delete, and update details and information of staff.
* The system should allow the system administrator to monitor the system user’s data.

**For Customer:**

* The system should create and manage Customer username and password.
* The system should be transfer fund to one to another account.
* The system should be managed and view transactions data of customers.

**USECASE DIAGRAM**

In the basic banking prototype, there are two actors: staff and customer. The staff can login, approve customer, view customer, delete customer, deposit to account, add beneficiary, remove beneficiary, transfer balance, change password, edit details, view statement, and logout. Similarly, the customer can login, view customer, deposit to account, add beneficiary, remove beneficiary, transfer balance, change password, edit details, view statement, and logout.



**Figure 3.2: Use Case Diagram of Basic Banking Prototype**

1. **Non-Functional Requirements**

Different non-functional requirement has been studied and identified and are listed as below:

* **Security:** The system is secure from outside attacks as authorized user and admin are allowed to access the data. Admin representative on duty can log into the system and have access to the sponsorship system but access to have various subsystems is protected by the user login screen that requires a username and password. This system uses at least 8-character passwords for security. Different validation process is used.
* **Performance:** The performance of the system is fast and accurate as in this system database is normalized so it provides fast operations.

**3.1.2 Feasibility Analysis**

1. **Technical Feasibility**

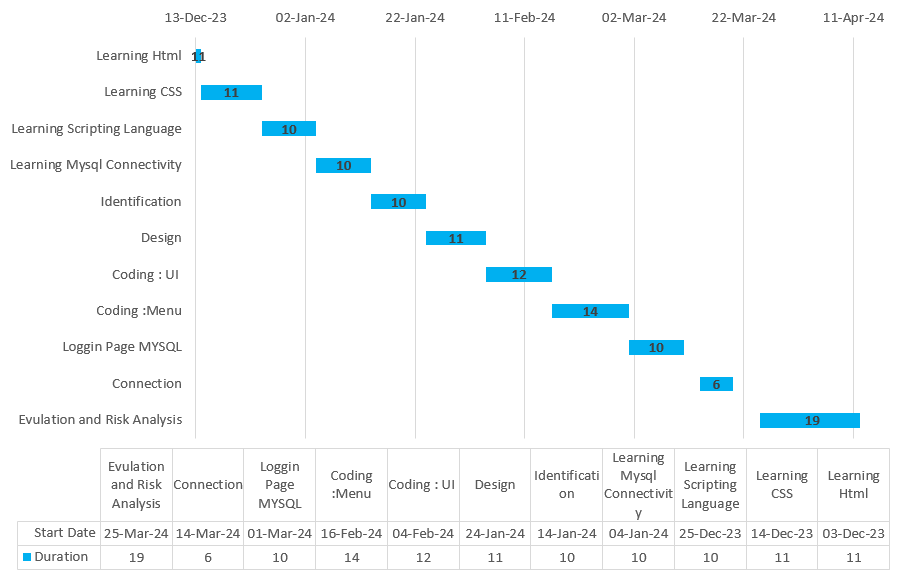
The system is technically feasible as the requirement for the development of the system is easily accessible. The necessary hardware and software required for the development and implementation of the system is available. The basic programming language which is suitable for project is available and the libraries required for project is capable of achieving the result that we are aiming for. All the existing resources can be used for the development and maintenance system.

1. **Economic Feasibility:**

The system is economically feasible and cost effective. As all the tools and resources required are either open sources or free. After the completion of the system organization didn’t need to deploy any new hardware and software as the required software and hardware. The existing resource of the system can be used.

**Table 3.1: Gantt chart Table for Basic Banking Prototype**

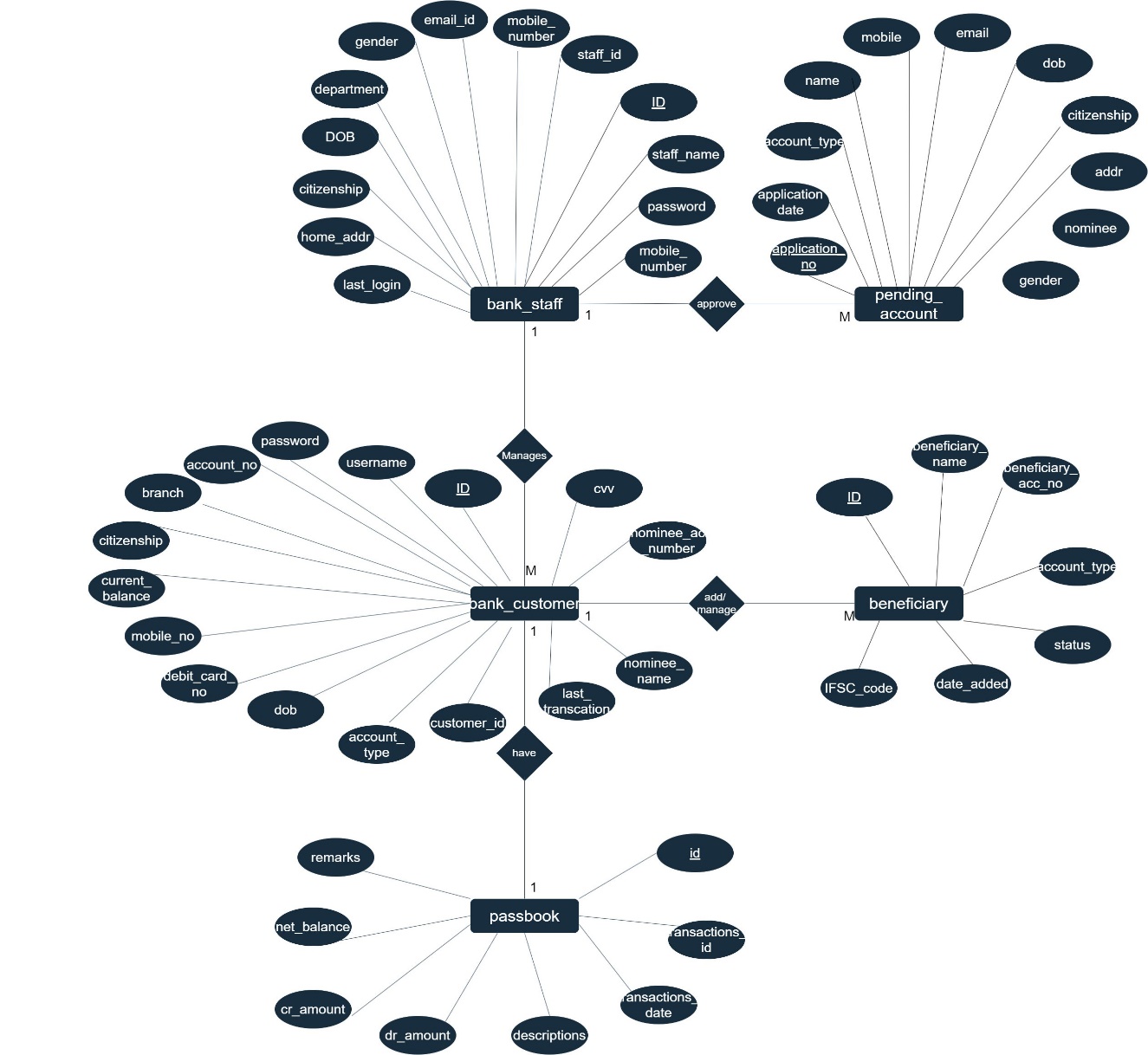
|  |  |
| --- | --- |
| **Task Name** | **Duration** |
| Getting Started | 2 weeks |
| System Design &Architecture | 2 weeks |
| Implementation | 7 weeks |
| Deployment | 4 weeks |
| Documentation | 12 weeks |



**Figure 3.3: Gantt chart for Basic Banking Prototype**

**3.1.3 Data Modeling**

The ER diagram for the **"Basic Banking Prototype"** project represents the key entities and relationships in a banking system. The primary entities include **‘bank\_staff’**, **‘bank\_customer’**, **‘pending\_account’**, **‘beneficiary’**, and **‘passbook’**. The ‘bank\_staff’ entity captures details such as email, mobile number, staff ID, name, department, date of birth, citizenship, home address, and last login. Bank staff members can approve multiple pending accounts and manage multiple bank customers. The ‘pending\_account’ entity contains information about accounts awaiting approval, including account type, application date, application number, and personal details like name, email, and mobile number.

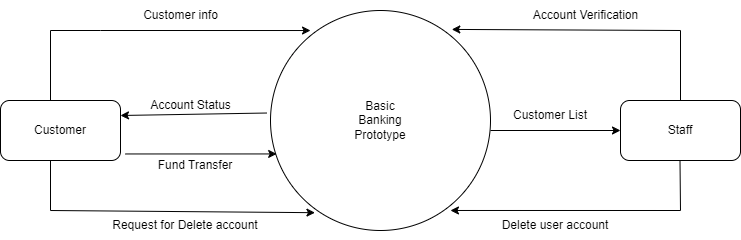
****The ‘bank\_customer’ entity is central, encompassing attributes like account number, branch, citizenship, current balance, mobile number, debit card number, date of birth, account type, username, password, CVV, nominee account number, nominee name, and last transaction. Bank customers can manage multiple beneficiaries and have a passbook associated with their account. The beneficiary entity records details of beneficiaries added by bank customers, including beneficiary name, account number, account type, status, IFSC code, and date added. Lastly, the passbook entity maintains transaction records for each customer, with attributes like transaction ID, transaction date, descriptions, credit and debit amounts, net balance, and remarks. The relationships between these entities, indicated by diamonds and lines, reflect various banking operations and data management activities within the system.

**Figure 3.4: ER-Diagram for Basic Banking Prototype**

**3.1.4 Process Modeling**

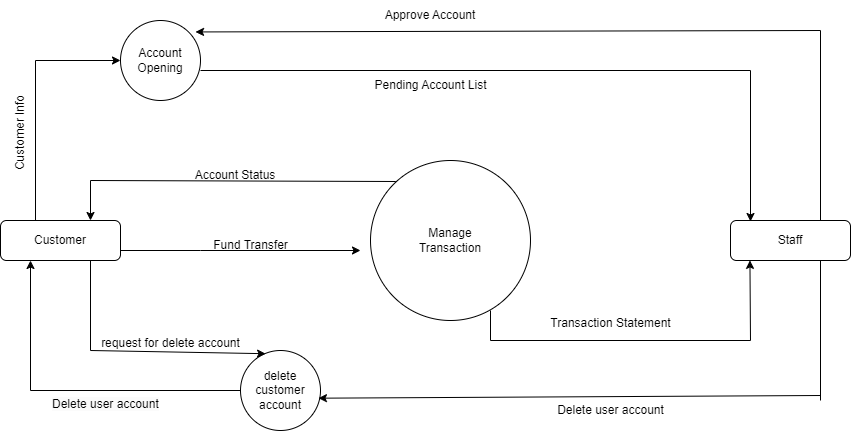
Process modeling is a technique used to represent and analyze the flow of activities, data, and decisions within a system or organization. It aims to provide a visual representation of how a process works, enabling stakeholders to understand, analyze and improve the process. It helps in optimizing processes, reducing errors, improving communication, and facilitating process automation initiatives. So, the process modeling of my project is given below:

**Zero Level DFD:** The Level 0 Data Flow Diagram (DFD) for the Basic Banking Prototype illustrates the system's high-level interactions with Customers and Staff. Customers provide information, request account deletions, and initiate fund transfers, while the system responds with account status updates. Staff members receive customer lists for account management and verify accounts. They also request account deletions, which the system processes. This DFD clearly depicts the essential data flows and interactions within the banking system.



**Figure 3.5: Level 0 DFD for Basic Banking Prototype**

* **First Level DFD:** This Level 1 Data Flow Diagram (DFD) depicts the basic interactions in a banking system. Customers open accounts, which staff approve and manage. Customers can transfer funds and check account status, while staff access transaction statements. Customers can also request account deletions, which staff process. The diagram shows the key flows of information between customers, staff, and the system.

****

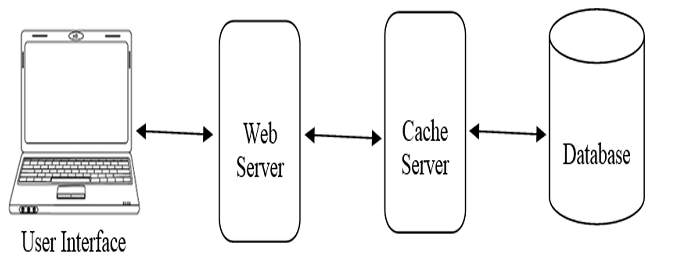
**Figure 3.6: Level 1 DFD for Basic Banking Prototype**

**3.2 System Design**

This system is designed with the series of processes starting with requirement analysis, design, implementation, testing and maintenance. During requirement analysis, all the functional and nonfunctional requirement are analyzed and system is developed according to the requirement then designing of the system is carried out. After the design process, coding and development part is started then after integrating the system there is testing of the system. If the testing is positive then system is implemented otherwise some maintenance is done and system come in operation.

**3.2.1 Architectural Design**

For this system, three tier architecture is used which includes user interface, web server and database. In architectural design, basic structure of the system is shown.



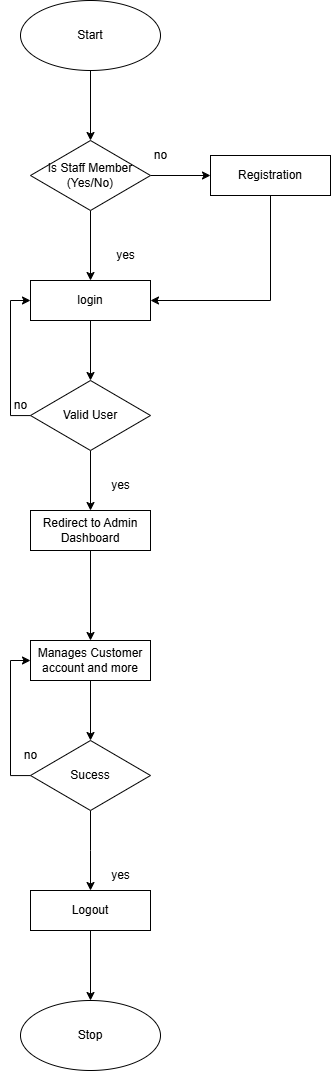
**Figure 3.7: Architectural Design for Basic Banking Prototype**

**3.2.2 System Flowchart**

The figures below depict the flowcharts of the basic banking prototype for staff and customers.

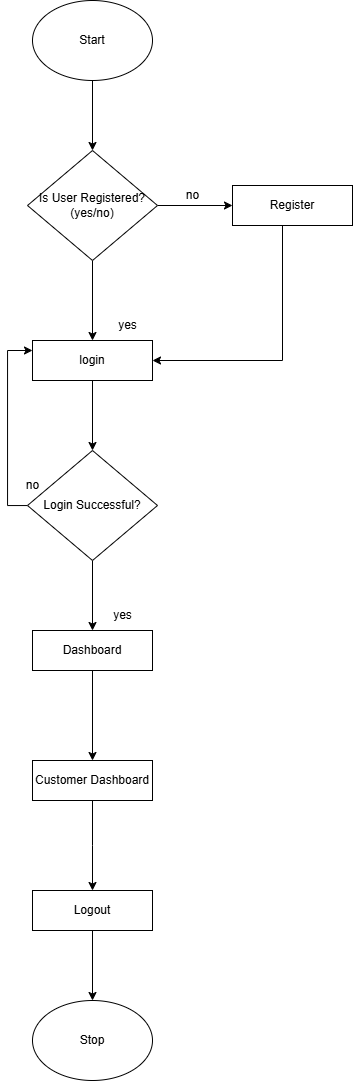
For staff, the process begins by determining if the person is a staff member. If not, they are directed to registration. Staff members proceed to the login step, and if the login is successful, they are redirected to the admin dashboard. Here, they can manage customer accounts and other administrative tasks. Upon completing their tasks, they can log out and end the process.

For customers, the process starts by checking if the user is registered. Unregistered users must register first. Registered users proceed to the login step, and if the login is successful, they are redirected to the customer dashboard. Customers can then access their account details and perform banking transactions. After completing their activities, they log out and end the process.

**For Admin**

**Figure 3.8: Flowchart of Basic Banking Prototype for Staff**

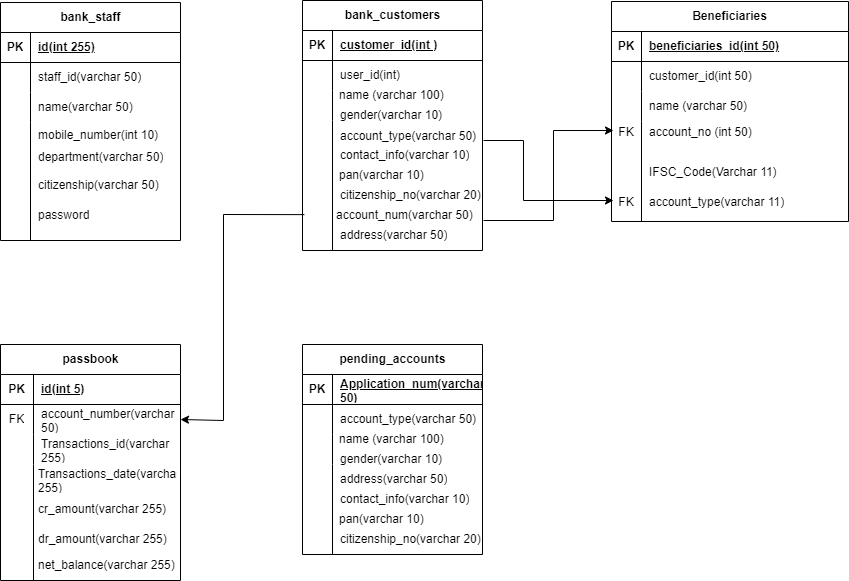
**For User**

****

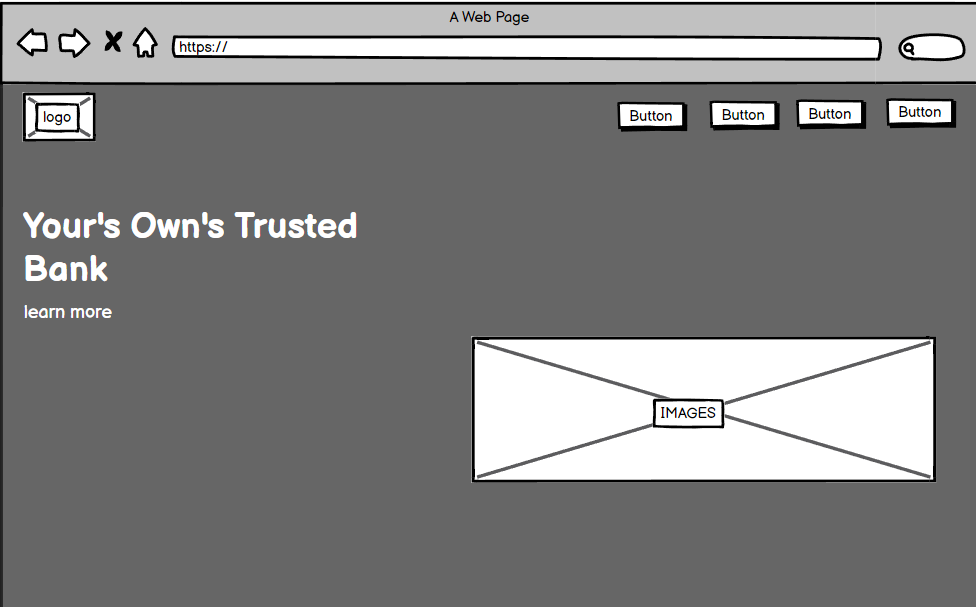
**Figure 3.9: Flowchart of Basic Banking Prototype for Staff**

**3.2.3 Database Schema Design**

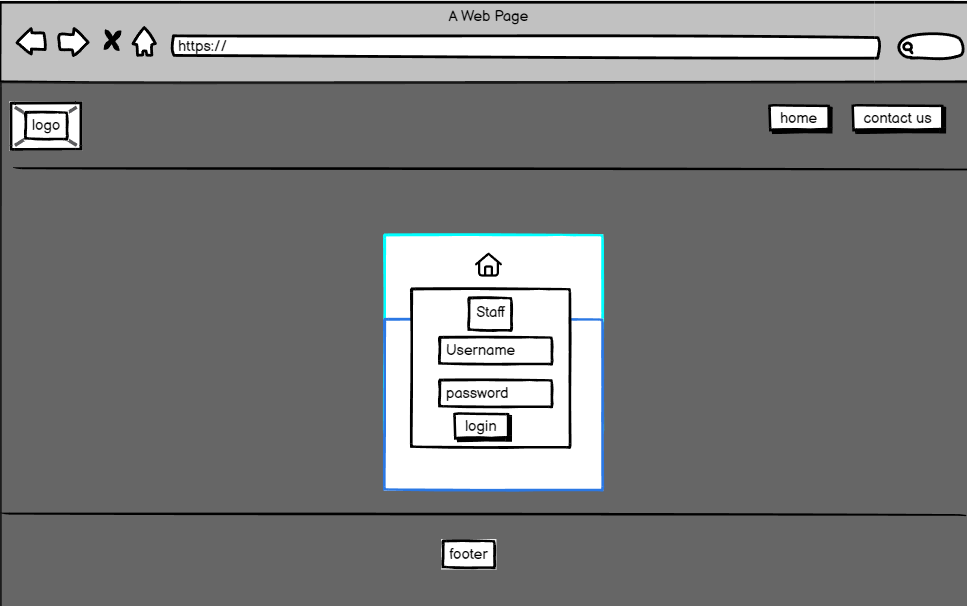
The figure below is the database schema design of Basic Banking Prototype. Database schema design is used to show basic structure of the system. In basic banking prototype, there are six tables in the databases each of them has their own fields where their id is primary key and if that id is used in another table, it becomes foreign key and foreign key are connected to another table with a line. There is data type of each entity and the foreign key in schema is represented by the arrow as shown in the diagram.

****

**Figure 3.10: Database Schema Design for Basic Banking Prototype**

**3.2.3 Interface Design**

**Figure 3.11: Main Page for Basic Banking Prototype**

****

**Figure 3.12: Login Page for Basic Banking Prototype**

**CHAPTER: 4**

**IMPLEMENTATION AND TESTING**

* 1. **IMPLEMENTATIONS**
     1. **Tools Used**

Different tools, application and technologies have been used in this project. And all of them are discussed below:

1. **Microsoft Visual Studio**

Microsoft Visual Studio is an Integrated Development Environment (IDE) developed by Microsoft to develop GUI (Graphical user interface), console, web application, web apps, mobile apps, cloud, and wed server, etc.

1. **XAMPP**

XAMPP is one of the widely used cross-platform web servers, which helps developers to create and test their programs on a local webserver. It was developed by the Apache Friends, and its native source code can be revised or modified by the audiences.

1. **Web Browser**

A web browser, or simply “browser,” is an application used to access and view websites. Common web browsers include Microsoft Internet Explorer, Google Chromes, Mozilla 29 Firefox, and Apple Safari.

**Front End Tools**

* **HTML**

HTML stands for Hypertext Markup Language in which it is the set of markup symbols or codes inserted into a file intended for displaying on the internet.

* **CSS**

CSS stands for a Cascading Style Sheet. Cascading style sheets are used to format the layout of Web Pages. They can be used to define text style, table sizes, and other aspects of web pages that previously could only be defined in a page’s HTML.

* **JavaScript**

JavaScript is a programming language commonly used in web development, it was originally developed by Netscape as a means to add dynamic and interactive elements to websites.

**Back End Tools**

* **PHP**

PHP stands for Hyper Pre-Processor (it is a recursive acronym, if you can understand what that means.) PHP is and HTML-embedded Web Scripting languages. This means PHP code can be inserted into the HTML of a Web pages.

**Database**

* **MySQL**

MySQL is a relational database management system based on SQL (Structured Query Language). This application is used for a wide range of purposed, including data 31 warehousing, e-commerce, and logging applications

* + 1. **Implementation Details of Modules**

The "Basic Banking Prototype" system comprises several modules, each with specific functionalities to manage different aspects of the banking operations. The modules are described as follows:

**Staff Module**

* **Staff edit/delete/view category**

This module equips staff with tools to efficiently manage customer accounts. Staff can update customer details like contact info and preferences. They can also delete inactive accounts securely, with all actions logged for accountability. Access to comprehensive customer profiles, including balances and transaction histories, is facilitated through robust search and filter features. Security measures include role-based access controls and detailed activity logging to ensure compliance and protect customer data, enhancing overall operational efficiency and customer service in banking operations.

* **Staff Manages Pending Account**

In this feature, staff are empowered to oversee and manage pending account applications effectively. Staff can review and approve pending accounts, ensuring adherence to strict verification protocols. If details such as mobile number, PAN number, or citizenship number match those of an existing account, staff are alerted to prevent duplicate accounts. Staff also have the authority to delete pending accounts that do not meet verification criteria or require further scrutiny. This process enhances security and prevents unauthorized or duplicate account creation, maintaining data integrity and regulatory compliance within the banking system.

* **Staff View Statement of Customer**

This functionality allows staff members to access and view detailed statements of customer accounts. Staff can retrieve comprehensive transaction histories, account balances, and other relevant details essential for customer service and operational purposes. This access enables staff to respond promptly to customer inquiries, verify transaction information, and provide accurate financial advice or assistance as needed. Security measures ensure that only authorized staff members can access sensitive customer data, maintaining confidentiality and regulatory compliance throughout the process.

* **Staff Credit Customer Account**

This feature enables staff members to directly credit funds into customer accounts. Staff can initiate and process credit transactions securely, ensuring accurate and timely deposits. This capability allows staff to manage routine and specific credit operations efficiently, maintaining detailed records and adhering to strict security protocols and compliance standards. This functionality enhances operational efficiency by facilitating seamless fund transfers and maintaining the integrity of customer financial transactions within banking operations.

**Customer Module**

* **Customer Add/Delete/View Beneficiary**

Customers can add, delete, and view beneficiaries within the banking system. This feature allows them to easily manage recipients for fund transfers, ensuring smooth transaction processes. It provides transparency by enabling customers to view beneficiary details, including account information, while maintaining security through authentication and authorization protocols.

* **Customer Transfer Funds**

Customers can securely transfer funds to added beneficiaries within the banking system. This feature allows customers to initiate and complete transactions easily, ensuring timely and accurate transfers between accounts. Authentication steps are integrated to maintain security and protect transaction integrity, enhancing convenience and efficiency in managing financial transfers.

* **Customer View Statement**

This feature allows customers to access and view their account statements conveniently within the banking system. Customers can review detailed transaction histories, account balances, and other essential financial details.

**Login Module**

In login module, we have implemented two sub modules they are staff login and customer login. Staff and customer log into the system using their valid username and password.

* 1. **TESTING**
     1. **Purpose of Testing**

The purpose of testing is to help in finalizing the software application or product against business and user requirements. Testing can be verification and validation or reliability estimation. Some of the objectives of software testing are given below:

* To check whether software which builds. It is as per the requirement or not.
* Finding defects form the software before customers find them out
* Defects get a fix from the developer.
* Preventing defects
* Gaining confidence about the level of quality
  + 1. **Test Care for Unit Testing**

1. **Test Case 1: Staff**

**A.1. Staff Login Page**

**Table 4.1: Staff Login Page of Basic Banking Prototype**

|  |  |
| --- | --- |
| Objective | Login into the web pages |
| Action | Entering login credentials i.e., Staff id and password. |
| Expected Result | To enter dashboard by signing in with user type Staff credentials. |
| Actual Result | Entered staff profile by signing in to the application with staff Credentials. |
| Conclusion | Test Successful |

**A.2. Approve Customer**

**Table 4.2: Approve Customer of Basic Banking Prototype**

|  |  |
| --- | --- |
| Objective | Approve the pending customer |
| Action | View pending account by application number and approve them |
| Expected Result | To Approve customer account |
| Actual Result | Approve Customer account then customer can get their account details. |
| Conclusion | Test Successful |

**A.3. Delete Customer**

**Table 4.3: Delete Customer of Basic Banking Prototype**

|  |  |
| --- | --- |
| Objective | Approve the customer account |
| Action | Login into staff panel and click on Delete Account |
| Expected Result | User account should be deleted |
| Actual Result | User account deleted and after that user can’t login with their details |
| Conclusion | Test Successful |

**A.4. Credit Customer**

**Table 4.4: Credit Customer of Basic Banking Prototype**

|  |  |
| --- | --- |
| Objective | Credit Balance on Customer Account |
| Action | Login into staff panel and click on Credit Section |
| Expected Result | User account should be Credit with balance |
| Actual Result | User account has credited and after that user can maker transactions |
| Conclusion | Test Successful |

**A.5. View Active Customer**

**Table 4.5: View Active Customer of Basic Banking Prototype**

|  |  |
| --- | --- |
| Objective | To view list of customers |
| Action | Login into staff panel and click on View Active Customer |
| Expected Result | Active Customer should be display |
| Actual Result | List of active customers will be displayed |
| Conclusion | Test Successful |

**A.6.Edit Customers Details**

**Table 4.6: Edit Customer Details of Basic Banking Prototype**

|  |  |
| --- | --- |
| Objective | To view list of customers |
| Action | Login into staff panel and click on Edit Details and enter customer account number |
| Expected Result | User details should be display and staff can edit their details |
| Actual Result | Customer details edited |
| Conclusion | Test Successful |

1. **Test Case:2 Customer**

**B.1.Customer Login Page**

**Table 4.7: Customer Login Page of Basic Banking Prototype**

|  |  |
| --- | --- |
| Objective | Login into the web pages |
| Action | Entering login credentials i.e. Customer id and password. |
| Expected Result | To enter dashboard by signing in with user type Customer credentials. |
| Actual Result | Entered customer profile by signing in to the application with staff Credentials. |
| Conclusion | Test Successful |

**B.2.Apply Debit Card**

**Table 4.8: Apply Debit Card for Basic Banking Prototype**

|  |  |
| --- | --- |
| Objective | To get debit card details |
| Action | Entering customer details to get debit card |
| Expected Result | To get debit cards details like: card number, cvv, etc. |
| Actual Result | Customer get their card details after entering the account details |
| Conclusion | Test Successful |

**B.3. Register E-Banking**

**Table 4.9: Register E-Banking of Basic Banking Prototype**

|  |  |
| --- | --- |
| Objective | To get login details for entering customer web pages |
| Action | Entering account details and debit card details to get login credentials |
| Expected Result | To get customer id and password to enter customer panel |
| Actual Result | Customer get their login credentials after entering the account details |
| Conclusion | Test Successful |

**B.4. Add Beneficiary**

**Table 4.10: Add Beneficiary on Basic Banking Prototype**

|  |  |
| --- | --- |
| Objective | To add beneficiary to make a transaction |
| Action | Login into customer panel and click into Add Beneficiary |
| Expected Result | Beneficiary account have to added |
| Actual Result | Beneficiary account added |
| Conclusion | Test Successful |

**B.5.Fund Transfer**

**Table 4.11: Fund Transfer on Basic Banking Prototype**

|  |  |
| --- | --- |
| Objective | To transfer fund to another accounts |
| Action | Login into customer panel and click to fund transfer section and select beneficiary to transfer fund |
| Expected Result | Fund should be transfer into preferred account |
| Actual Result | Fund Transfer into another account |
| Conclusion | Test Successful |

**CHAPTER:5**

**CONCLUSION AND FUTURE RECOMMENDATIONS**

* 1. **Lesson Learn**

While making this project we have learn many things. And they are listed down below:

* Learn about PHP, HTML, JavaScript, Jason, and MySQL server.
* Learn how to connect all the things.
* Learn about how to do research in the current market.
* Learn how to solve the problem related to codes
* Learn to implement a crud operation in a system
* Learn to how to completely workable project.
  1. **Conclusion**

In conclusion, the development of the "Basic Banking Prototype" represents a significant step forward in the evolution of financial services. This prototype not only enhances the accessibility and efficiency of banking transactions but also modernizes the way we manage personal and business finances. With intuitive user interfaces, robust security measures, and real-time processing capabilities, this banking model offers a more streamlined and user-friendly experience for customers.

However, the implementation of such banking technologies also comes with its own set of challenges, including the need for stringent data security, compliance with financial regulations, and the continuous updating of technology to thwart cyber threats. Despite these hurdles, the "Basic Banking Prototype" remains a promising development for financial institutions looking to innovate, enhance customer satisfaction, and expand their services globally. As technology continues to advance, the potential for further transformation in the banking sector is immense, paving the way for more secure, efficient, and inclusive financial environments.

* 1. **Future Recommendation**

The future of the "Basic Banking Prototype" holds considerable potential for enhancement and further innovation. To ensure its evolution aligns with the growing demands of the digital age, the following recommendations are proposed for future developments in this field:

* Integration of Advanced Analytics: Implement more sophisticated analytics tools to help users understand their spending habits, forecast future trends, and provide personalized financial advice based on data-driven insights.
* Enhanced Security Measures: Develop stronger cybersecurity measures, including biometric security systems and two-factor authentication, to protect user data and prevent fraud. Explore the use of blockchain technology for secure, transparent transactions.
* Artificial Intelligence Services: Incorporate AI to offer automated customer support and financial advisory services. AI can help in personalizing the banking experience, managing assets, and providing customers with proactive alerts about their financial health.
* Expansion of Services: Broaden the range of services offered, such as insurance, investment products, and retirement planning, to make the basic banking app a comprehensive financial tool for users at different stages of their financial journey.
* Sustainability Initiatives: Implement practices to reduce the environmental impact of banking operations, including paperless systems and incentivizing customers to engage in environmentally friendly transactions and investments.
* User Experience Customization: Enhance personalization of the user interface and experience to accommodate individual preferences and accessibility needs, making banking more intuitive for a diverse customer base.

By pursuing these recommendations, the "Basic Banking Prototype" can continue to evolve and significantly impact the financial industry, promoting more inclusive, secure, and innovative banking solutions.

**REFERENCES**

[1] Ashraf, Junaid. "Design and Implementation of an Online Banking System in PHP." [ONLINE] Available: [Junaid Ashraf - Software Development Manager - u-blox | LinkedIn](https://pk.linkedin.com/in/junaid-ashraf-682075a). Accessed: [2024].

[2] Sharma, Priya, and Patel, Ankit. "Secure Online Transactions in Banking Systems: A Review." [ONLINE] Available: [(PDF) A Review of Cyber Security Issues in Online Banking and Online Transactions (researchgate.net)](https://www.researchgate.net/publication/372548600_A_Review_of_Cyber_Security_Issues_in_Online_Banking_and_Online_Transactions). Accessed: [2023].

[3] Kumar, Kiran, and Reddy, Suman. "User Experience in Online Banking: A Comparative Analysis." [ONLINE] Available: [334f5e51acce0b4eef5a63b1845c6e672895.pdf (semanticscholar.org)](https://pdfs.semanticscholar.org/d0a5/334f5e51acce0b4eef5a63b1845c6e672895.pdf). Accessed:[22024].

[4] Nabil Bank. "Intelligent Support Tools in Managing Digital Resources: A Case Study." [ONLINE] Available: [https://www.nabilbank.com/services-payments/services/individuals/nbank](https://www.nabilbank.com/services-payments/services/individuals/nbank%20) Accessed: [2021-2024].

[5] "Financial Times" | [ONLINE] | Available[: https://www.ft.com/](https://d.docs.live.net/a81f7ae8be0f6923/Documents/:%20https:/www.ft.com/) | [Accessed - 2024]

[6] "The Economist - Finance and Economics" | [ONLINE] | Available: [https://www.economist.com/finance-and-economics](https://www.economist.com/finance-and-economics%20) | [Accessed 2023]

[7] "Investopedia" | [ONLINE] | Available: <https://www.investopedia.com> | [Accessed 2023]

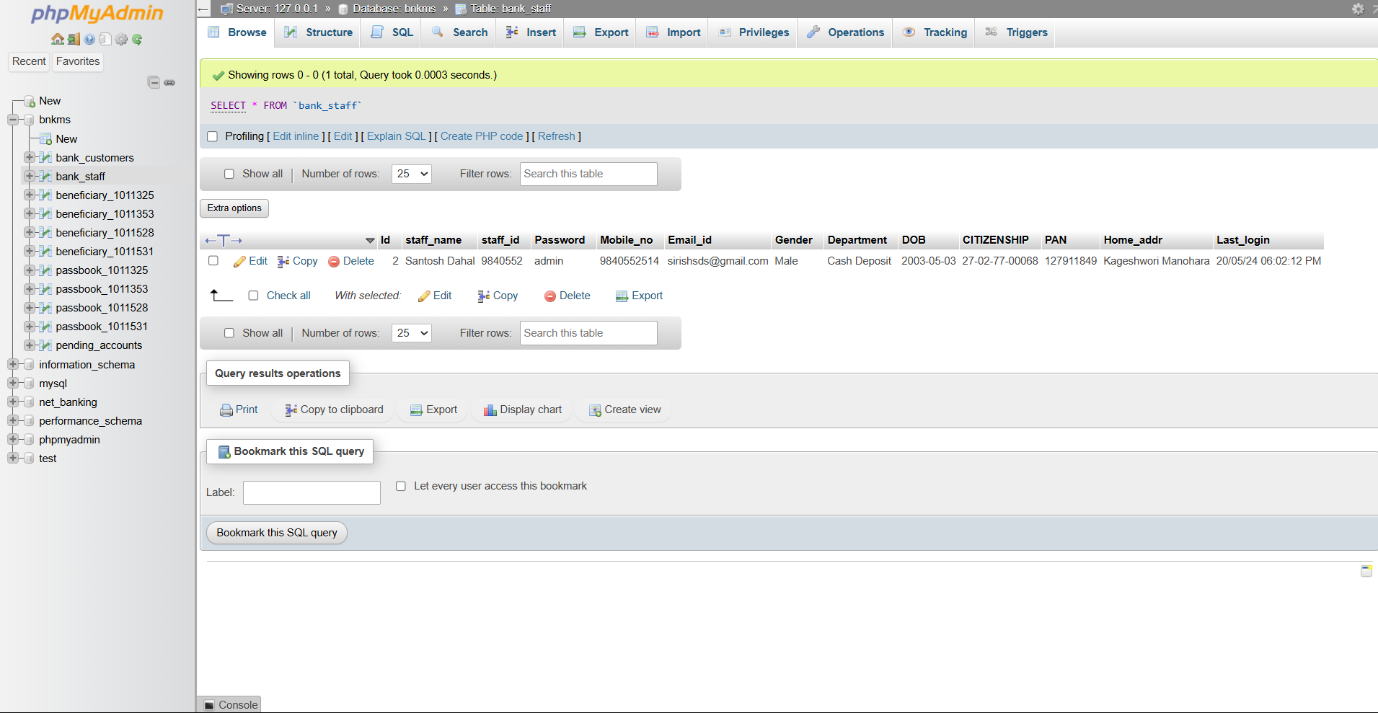
[8] "World Bank - Digital Finance" | [ONLINE] | Available: [https://www.worldbank.org/en/topic/financialsector/brief/digital-finance](https://www.worldbank.org/en/topic/financialsector/brief/digital-finance%20) | [Accessed 2024]

[9] "FinTech Futures" | [ONLINE] | Available: [https://www.fintechfutures.com/](https://www.fintechfutures.com/%20) | [Accessed 2024]

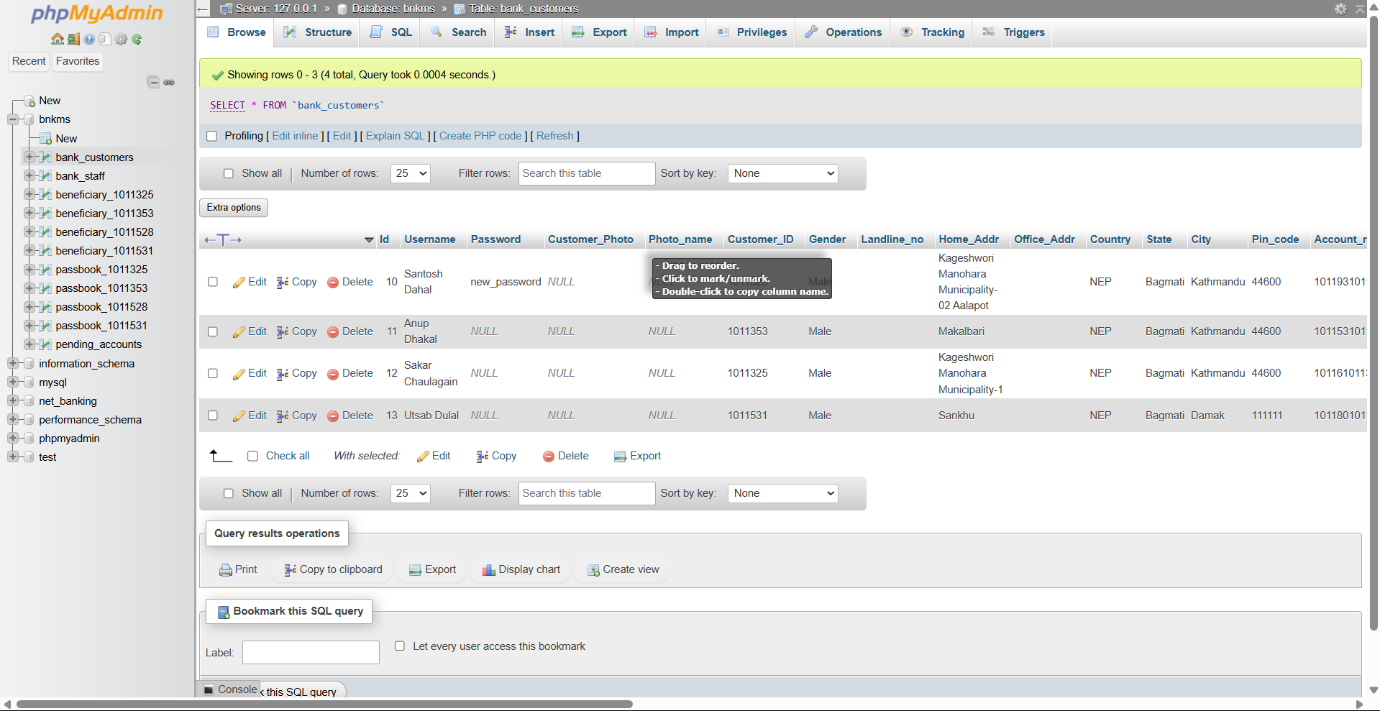
[10] "TechCrunch - Fintech" | [ONLINE] | Available[: https://techcrunch.com/tag/fintech/](https://d.docs.live.net/a81f7ae8be0f6923/Documents/:%20https:/techcrunch.com/tag/fintech/) | [Accessed 2024]

**APPENDIX: SYSTEM SCREENSHOTS**

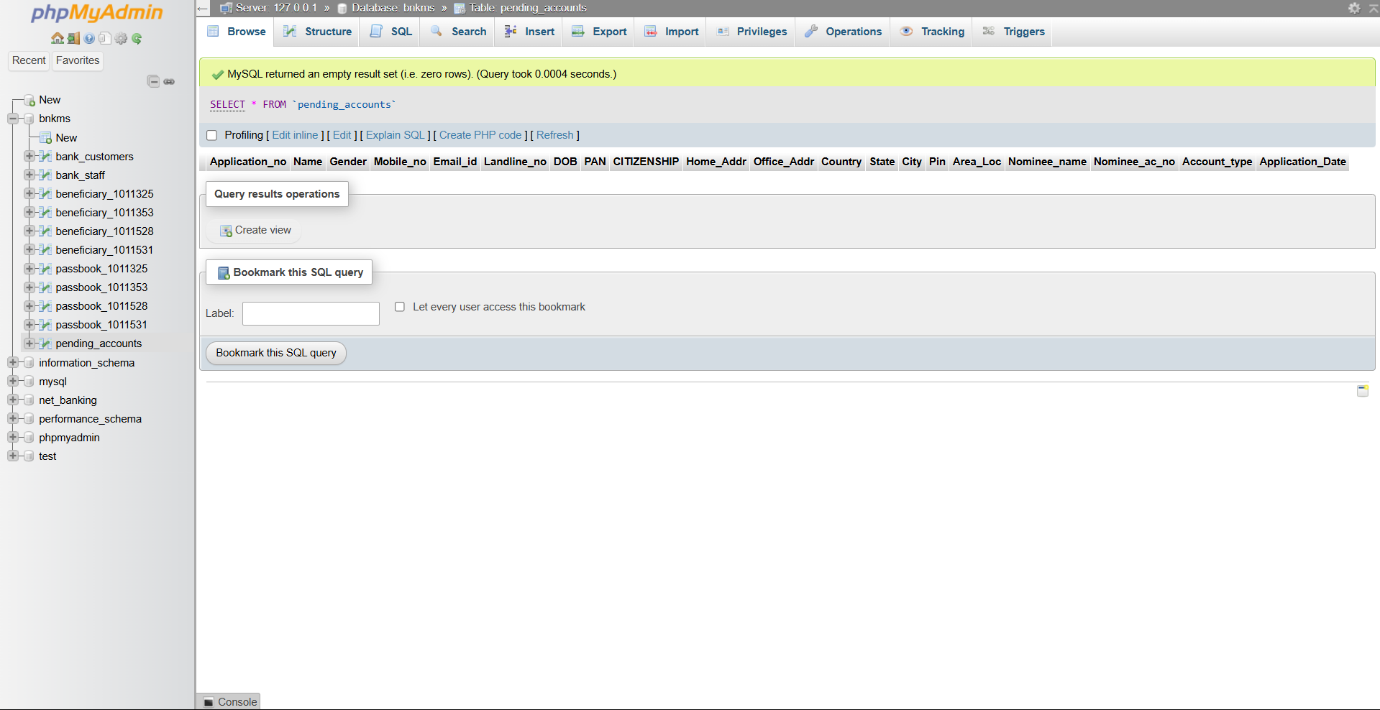
* Database Overview
* Bank Staff Database



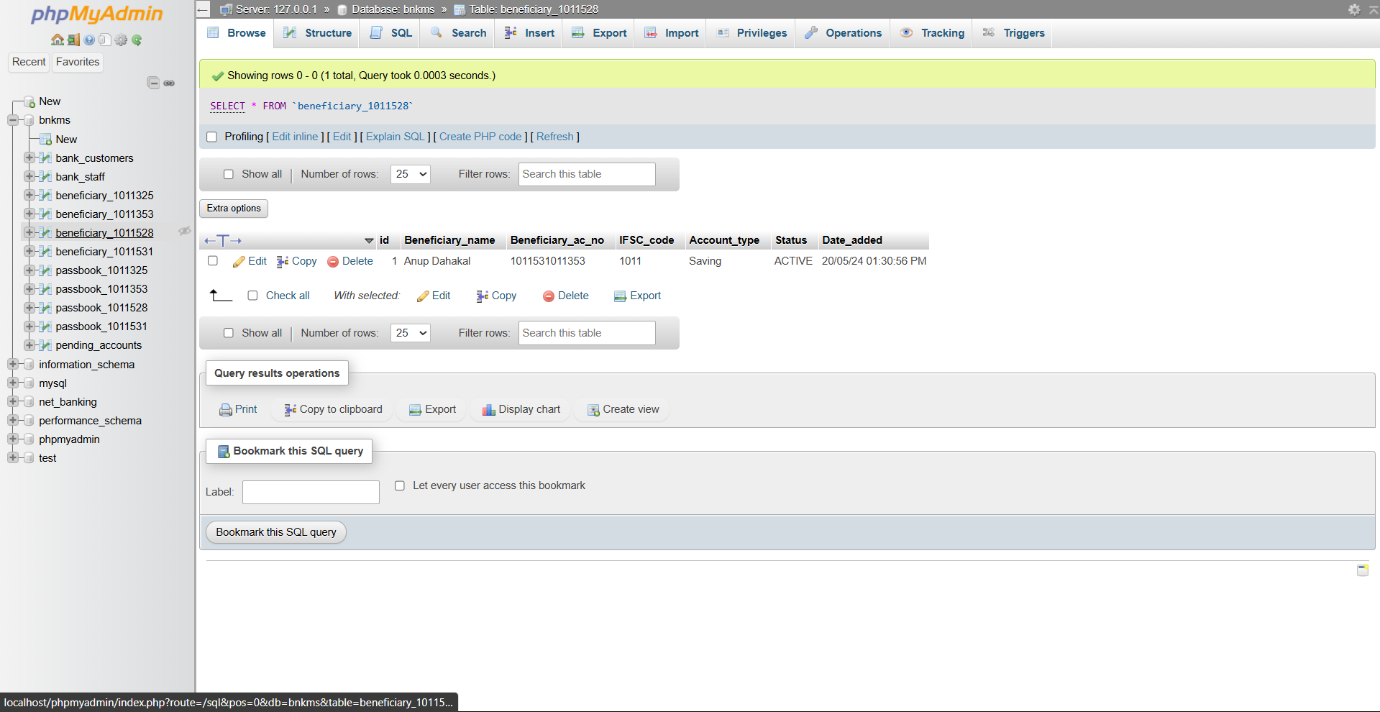
* Bank Customers Database



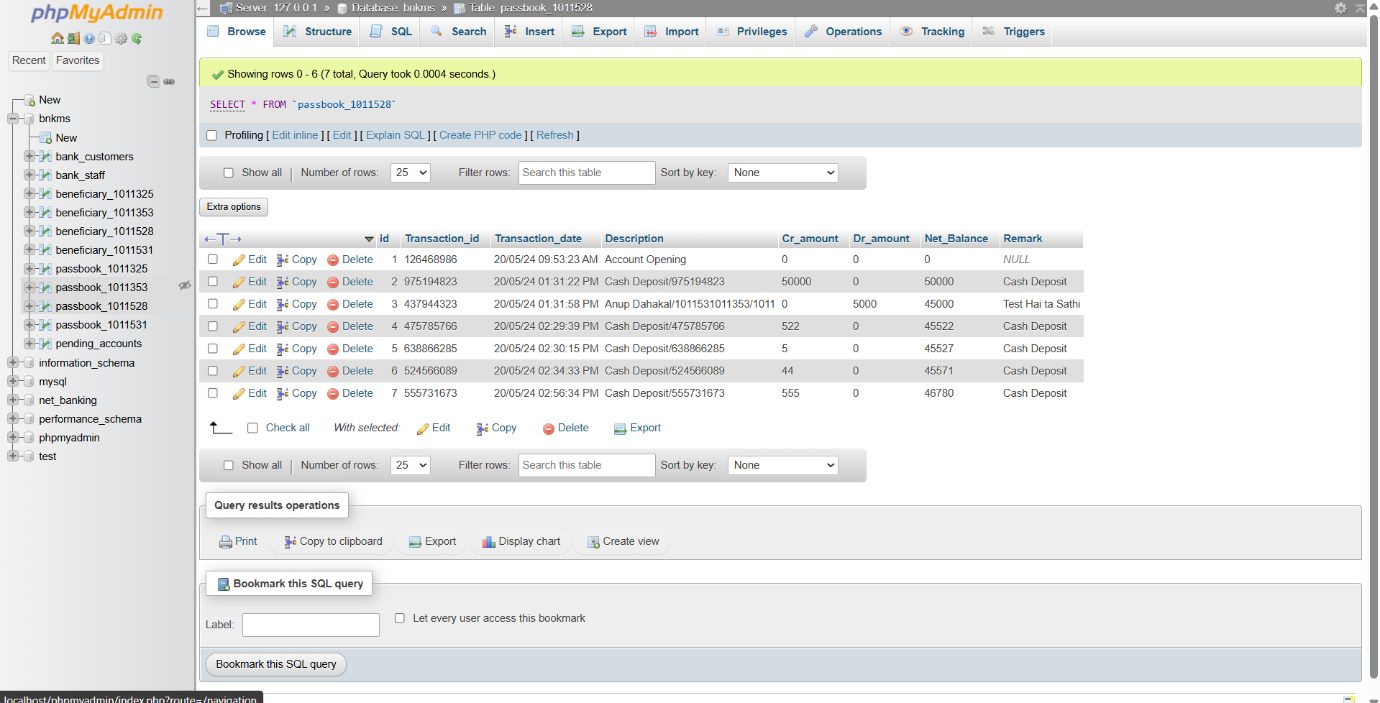
* Pending Account Database

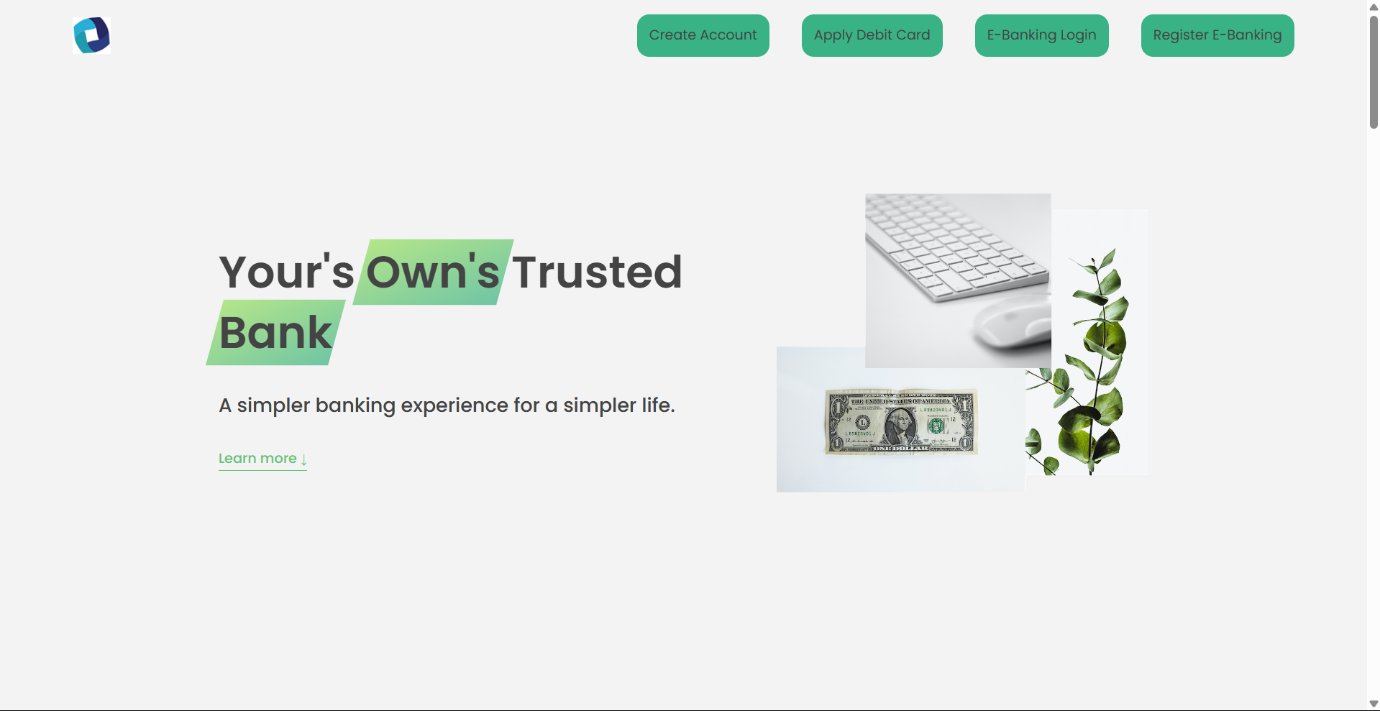


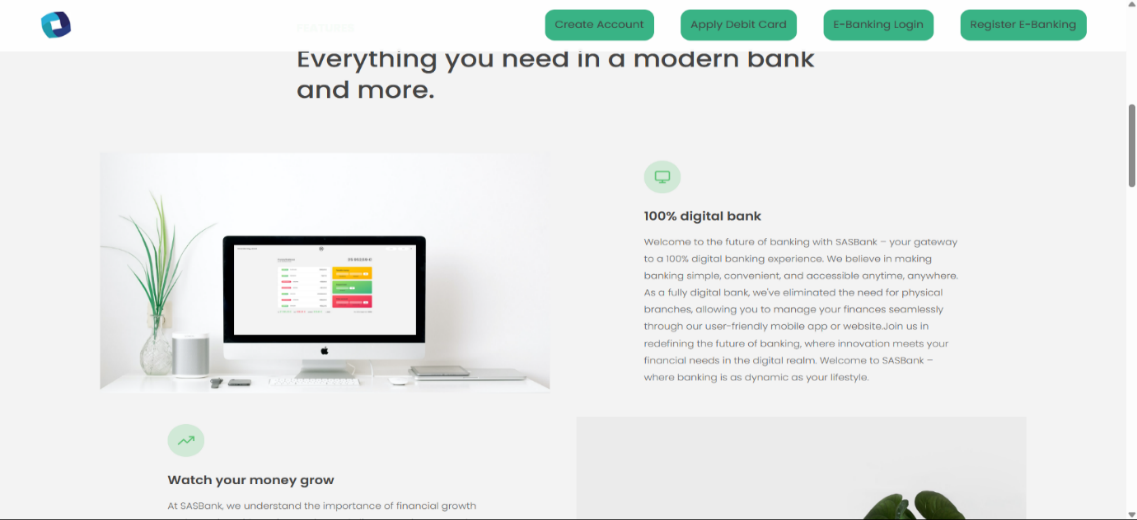
* Beneficiary Database

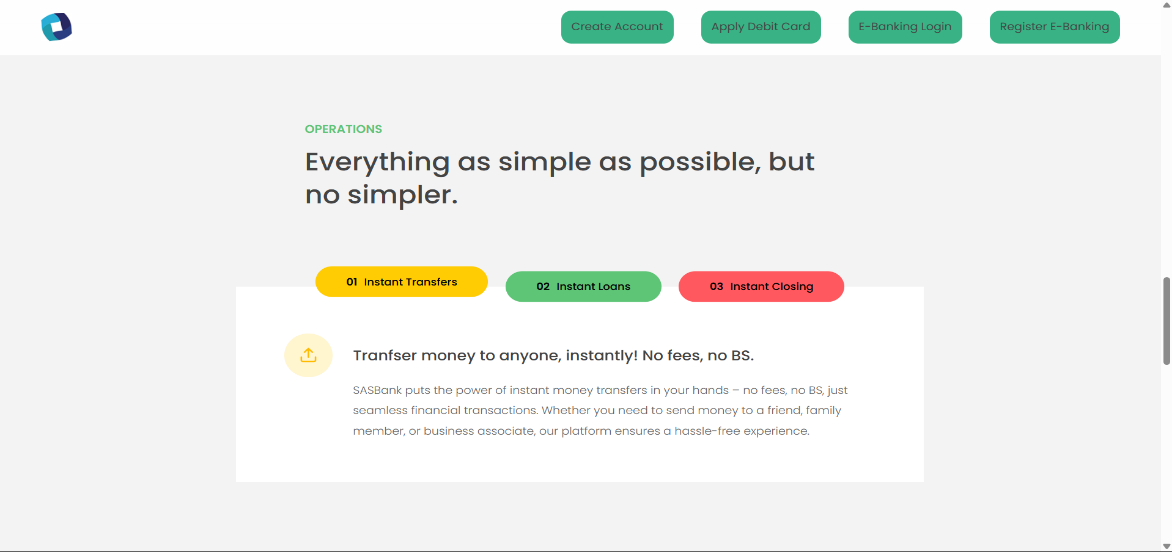


* Passbook Database

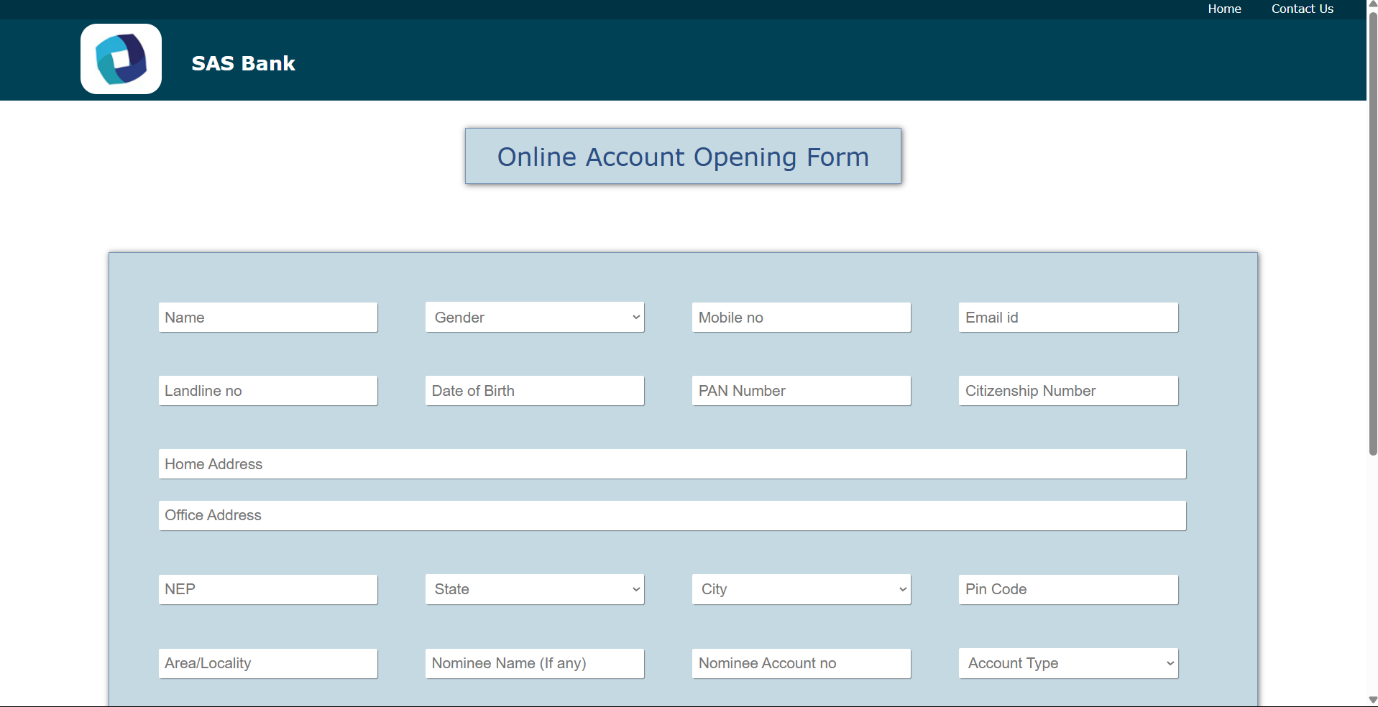


* Front End Overview
* Home Page

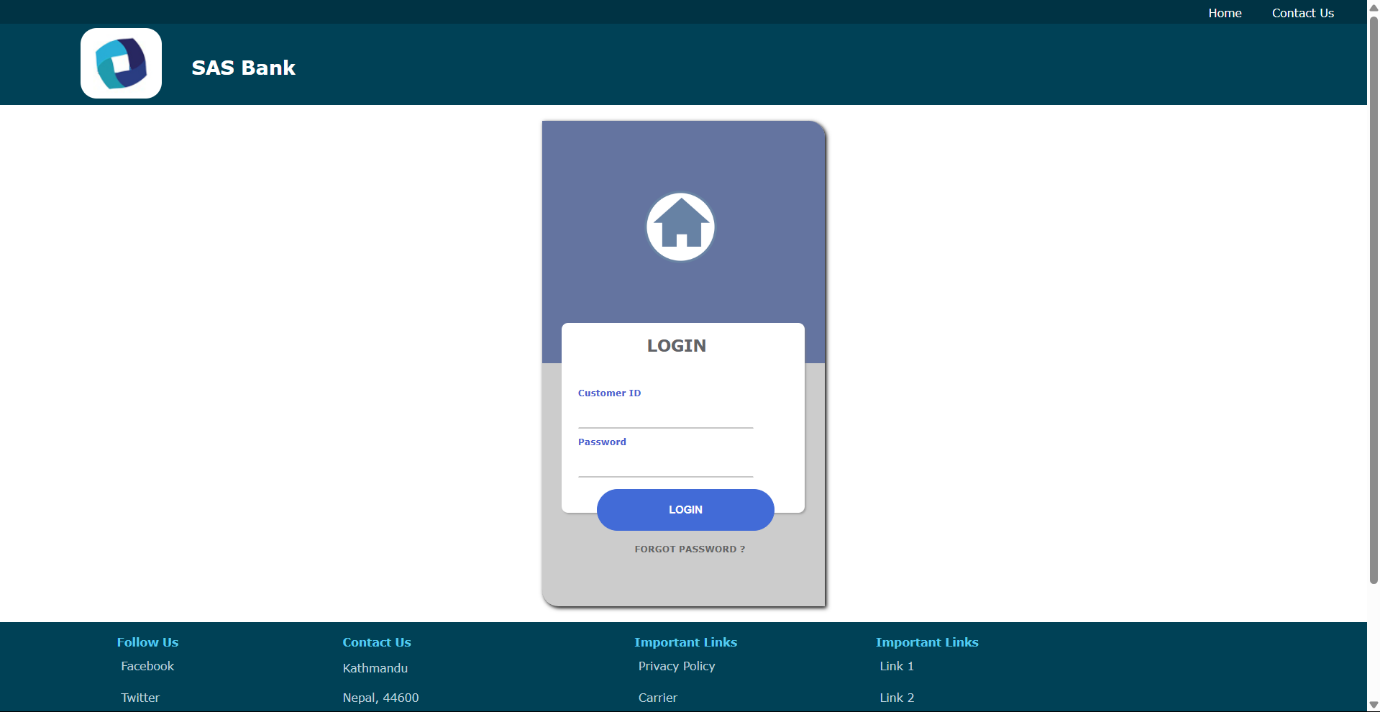




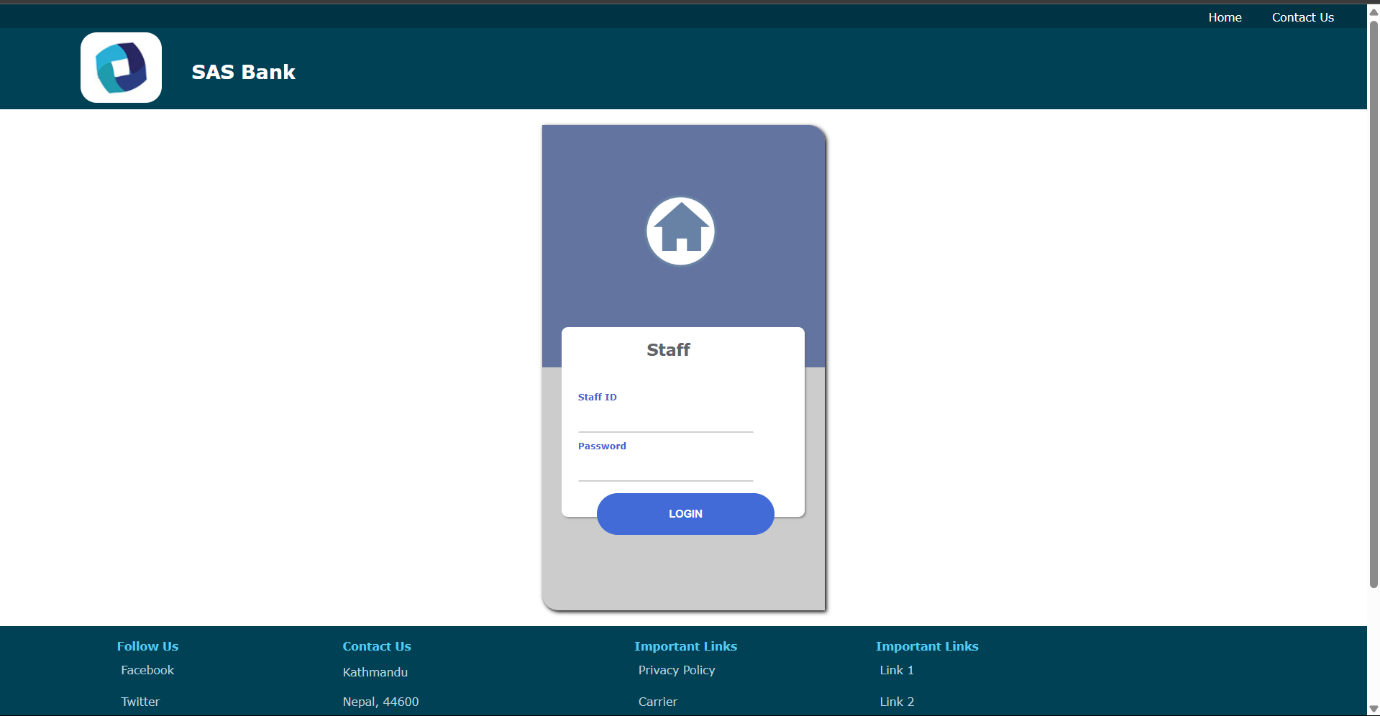
* **Account Registration Page**

****

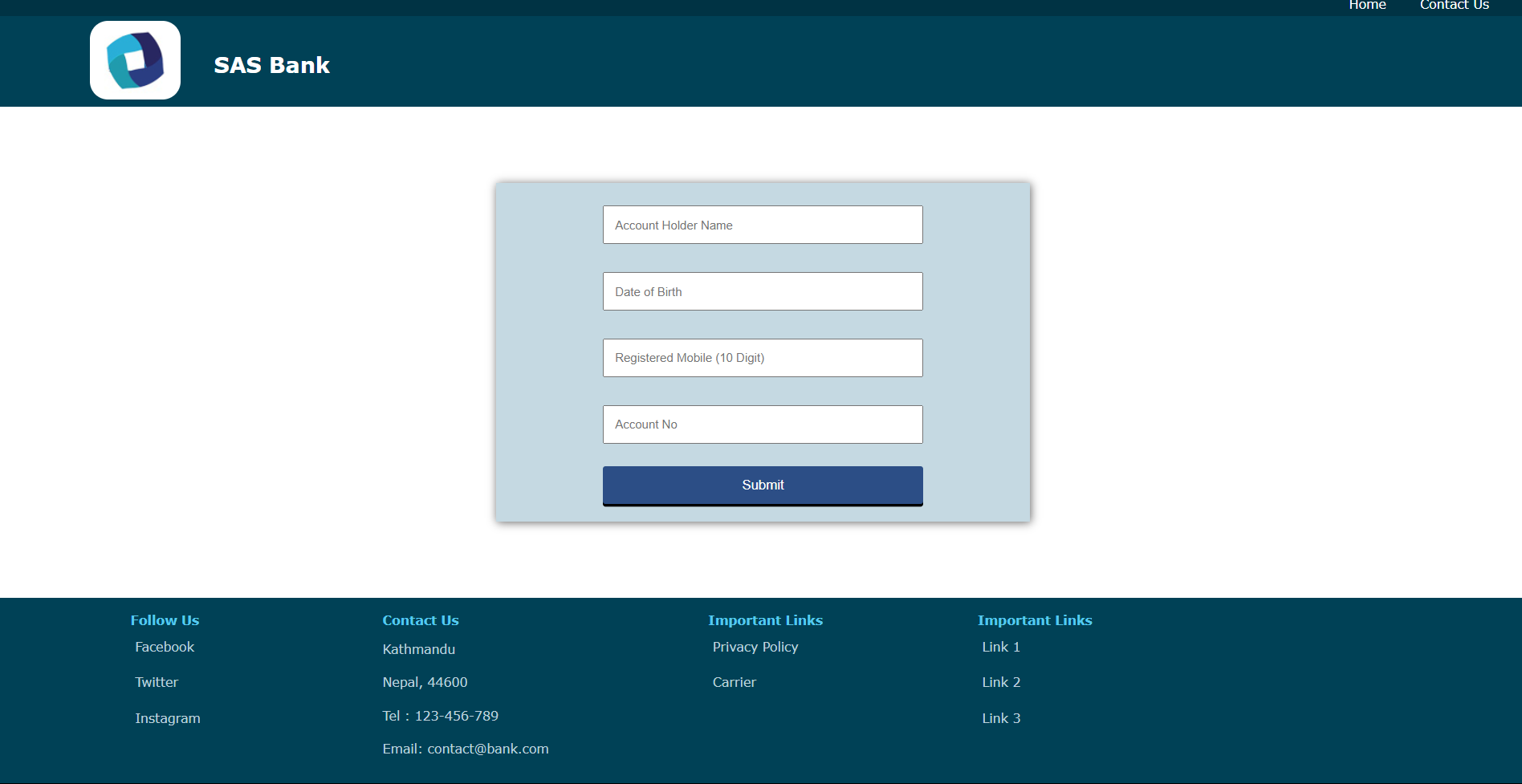
* Customer Login Page



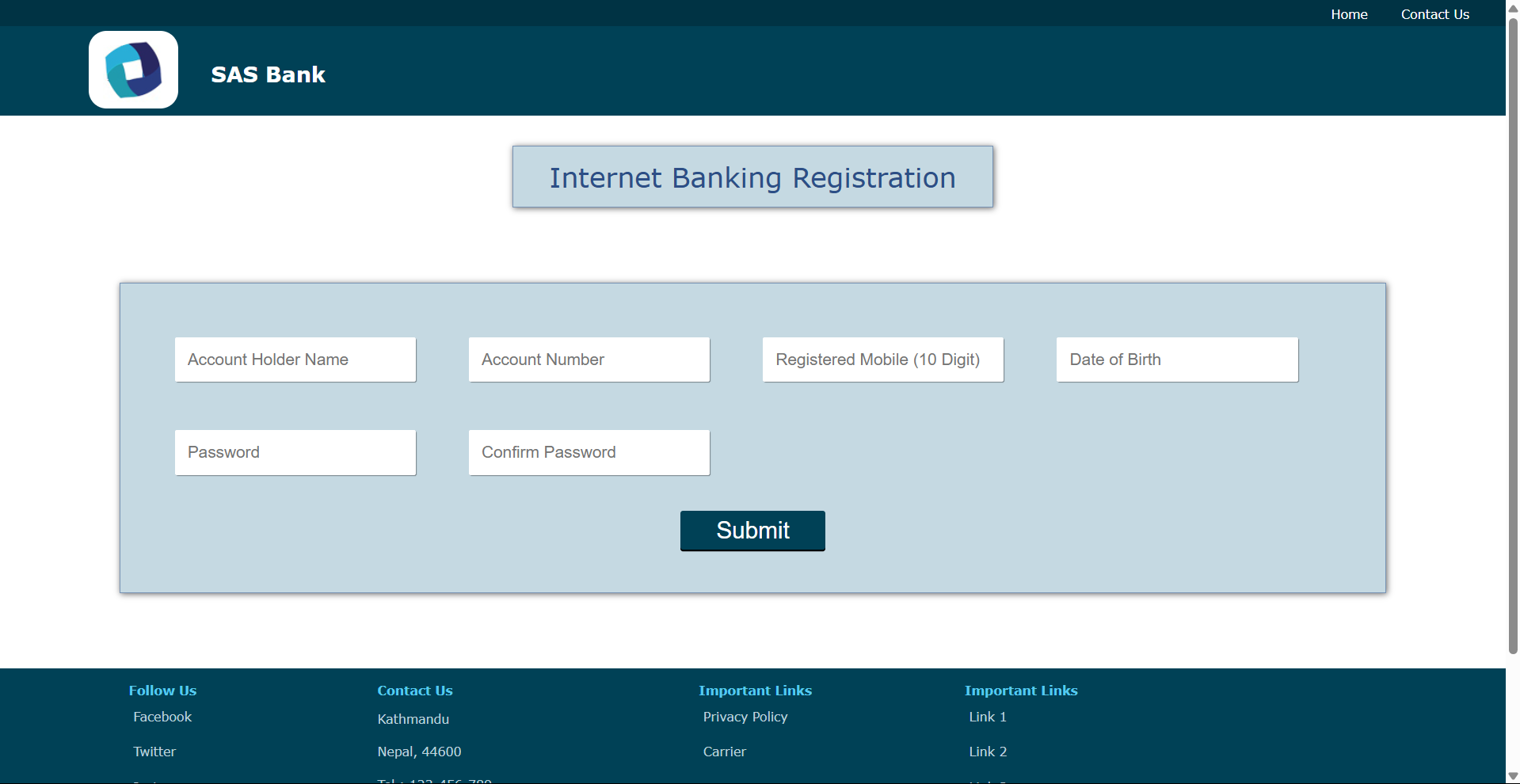
* Staff Login Page

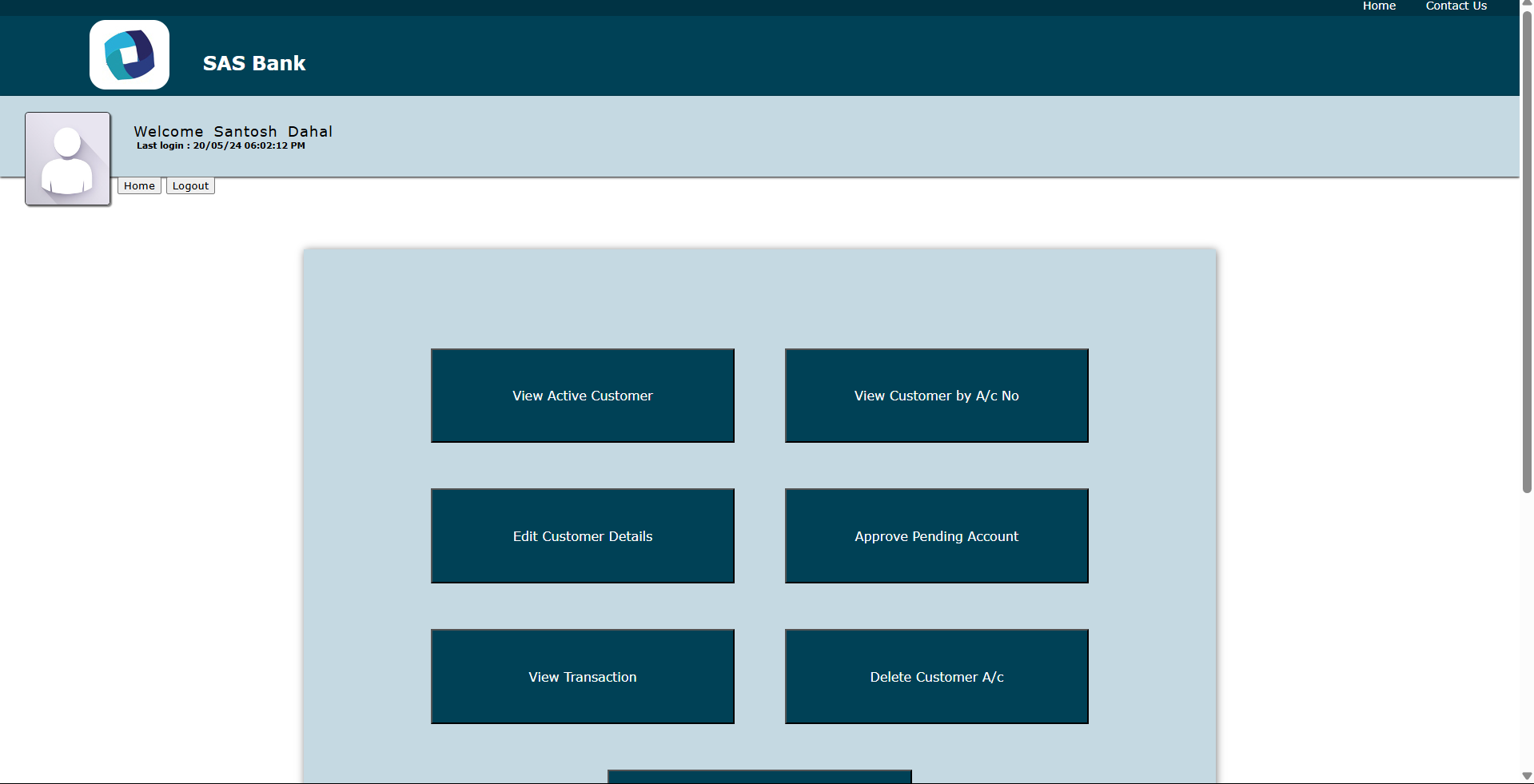


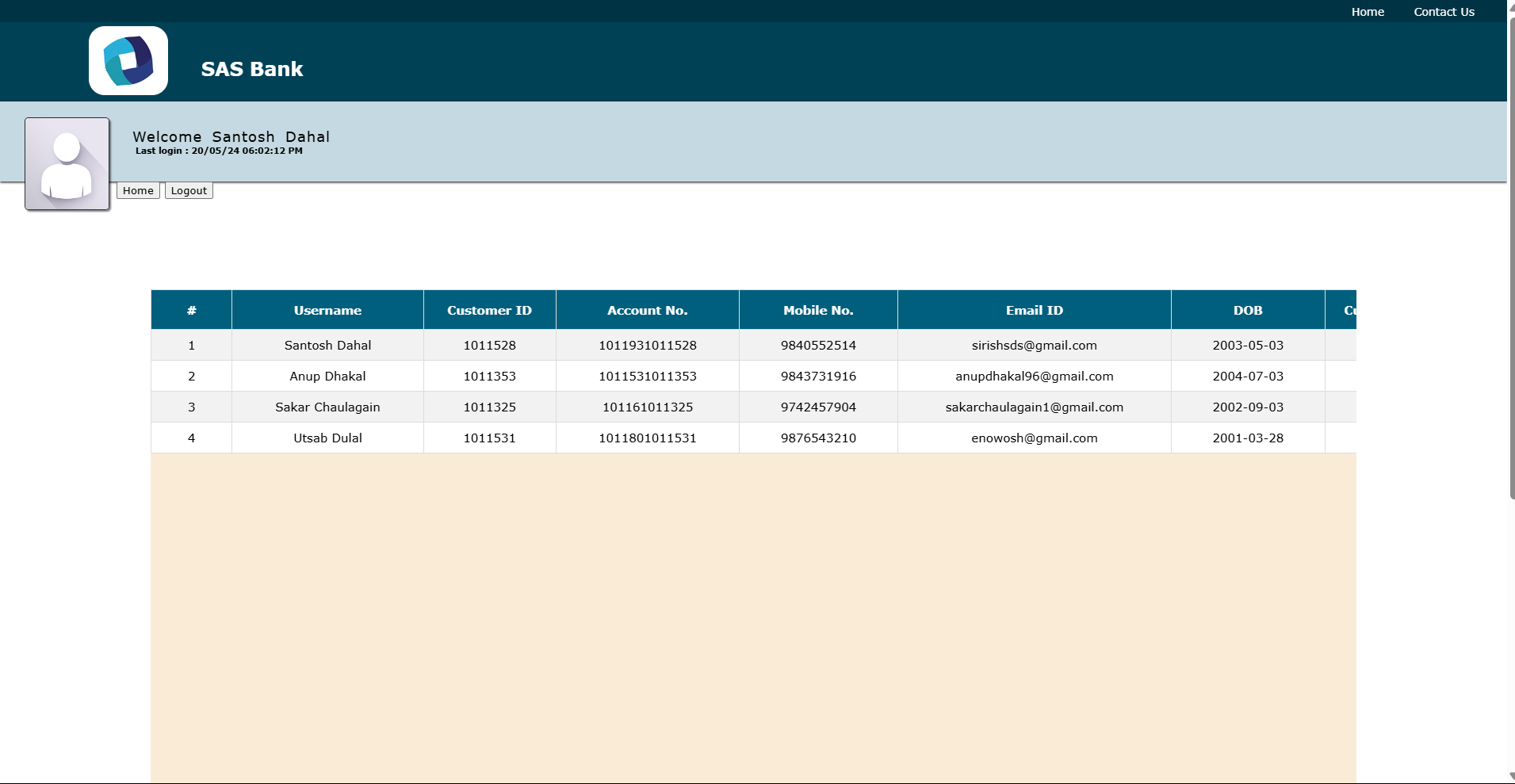
* Page for Debit Card Apply



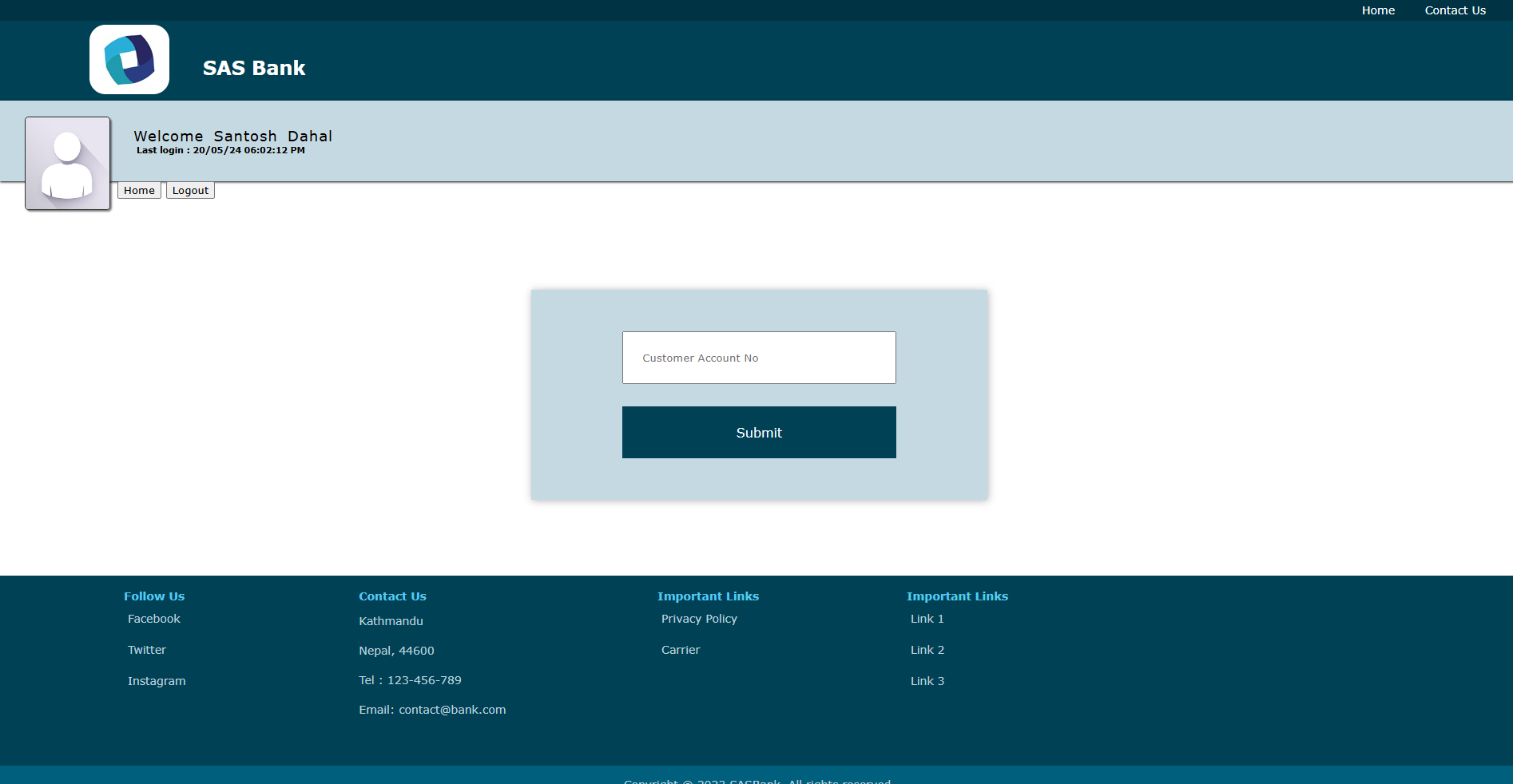
* E-Banking Registration Page



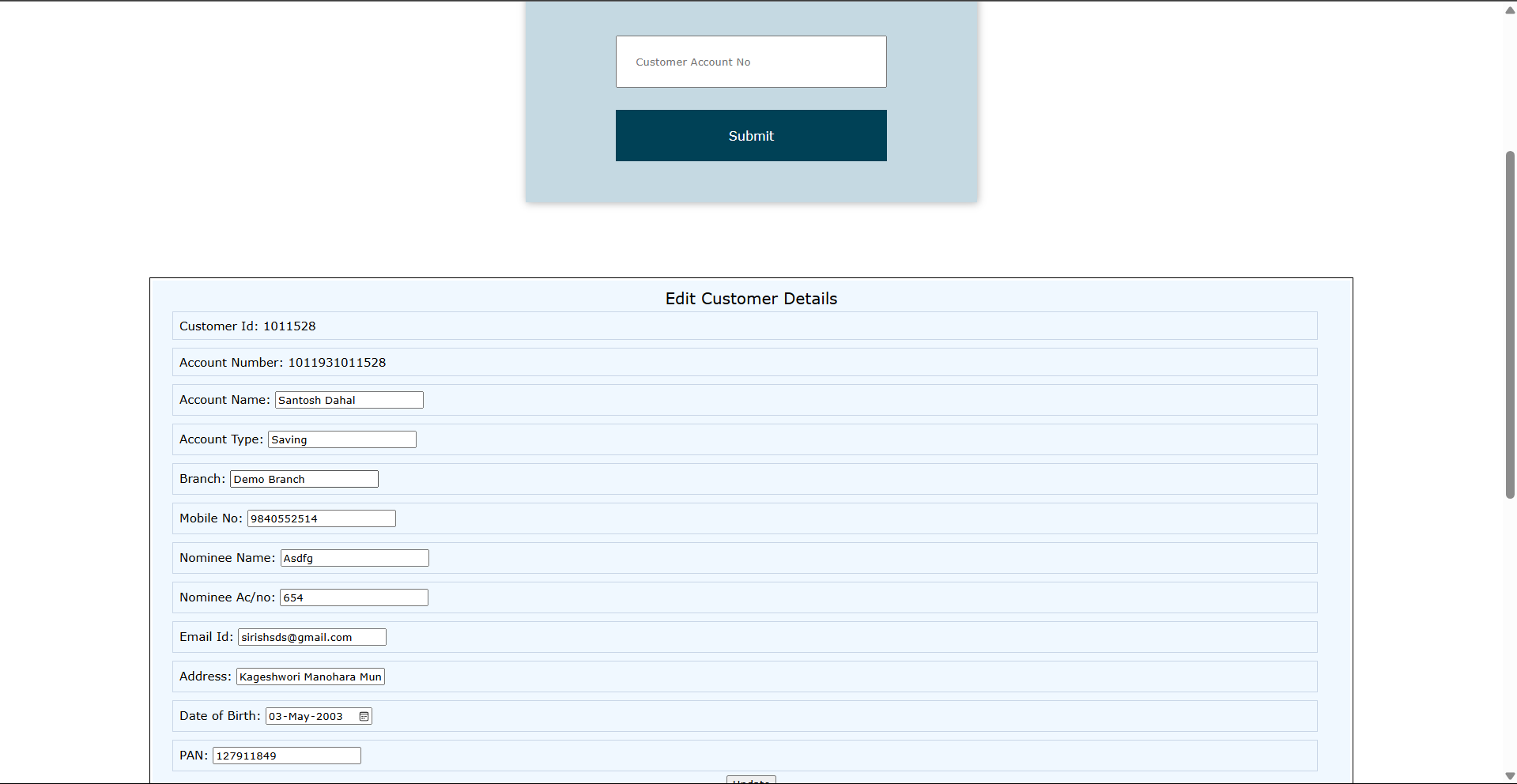
* Staff Panel Overview
* Dashboard
* View Active Customer Page



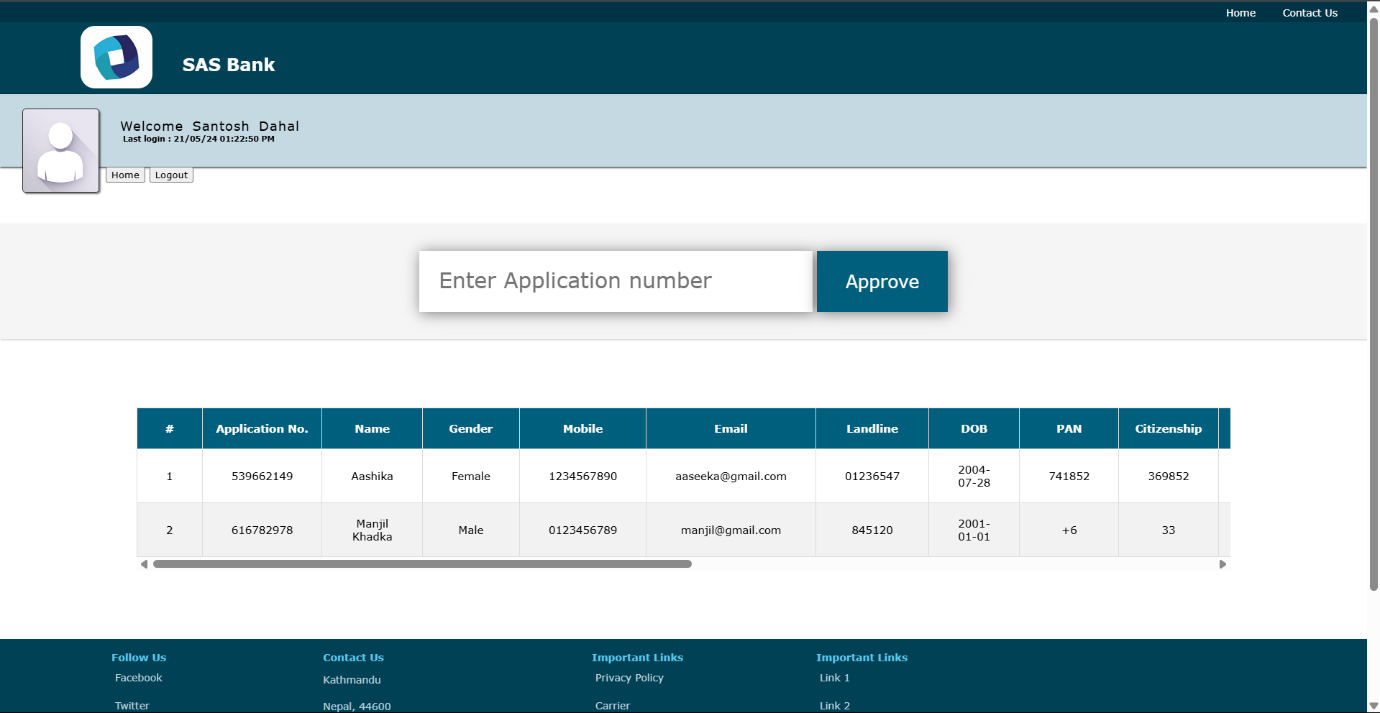
* View Customer by Account Number Page



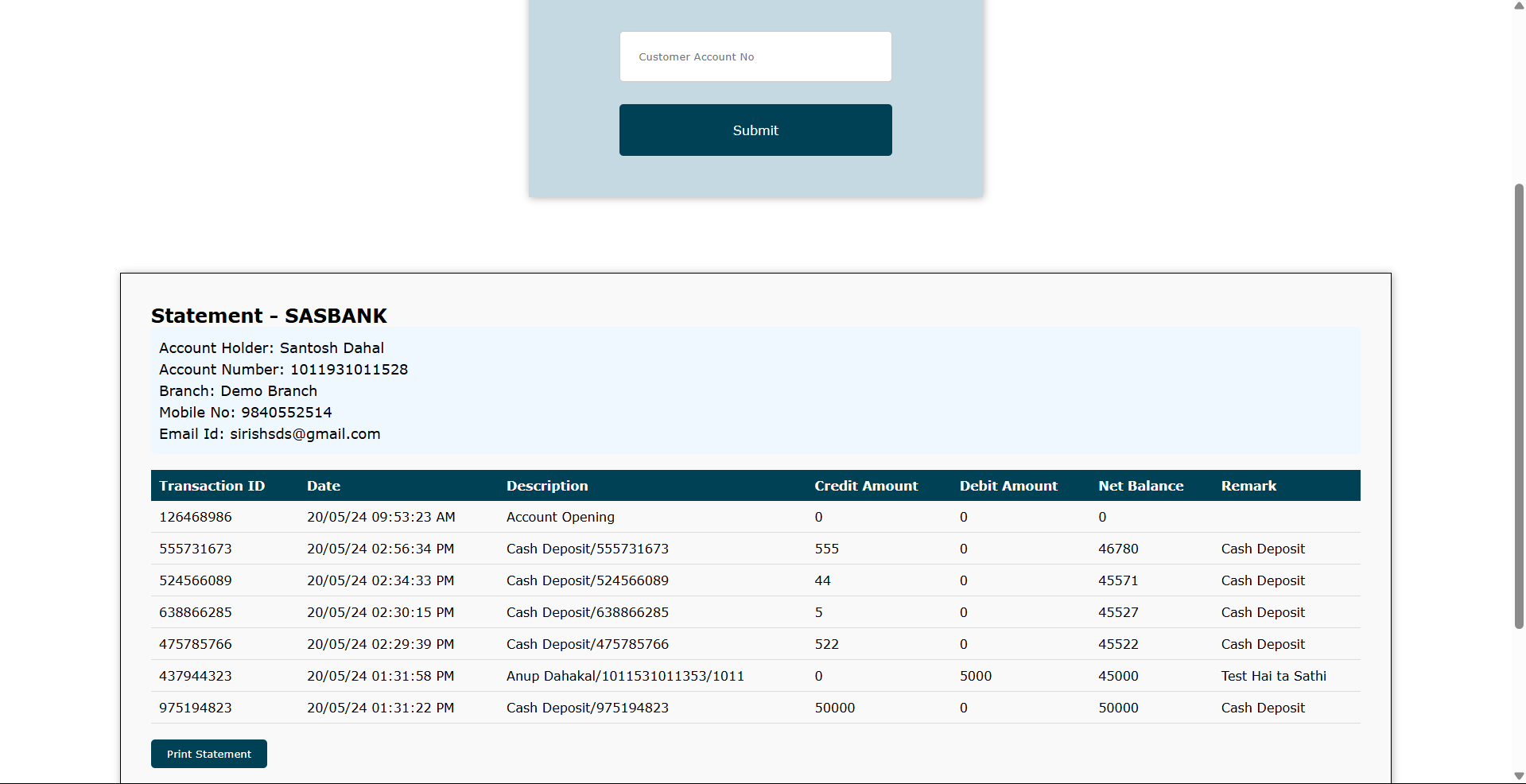
* Edit Customer Details Page



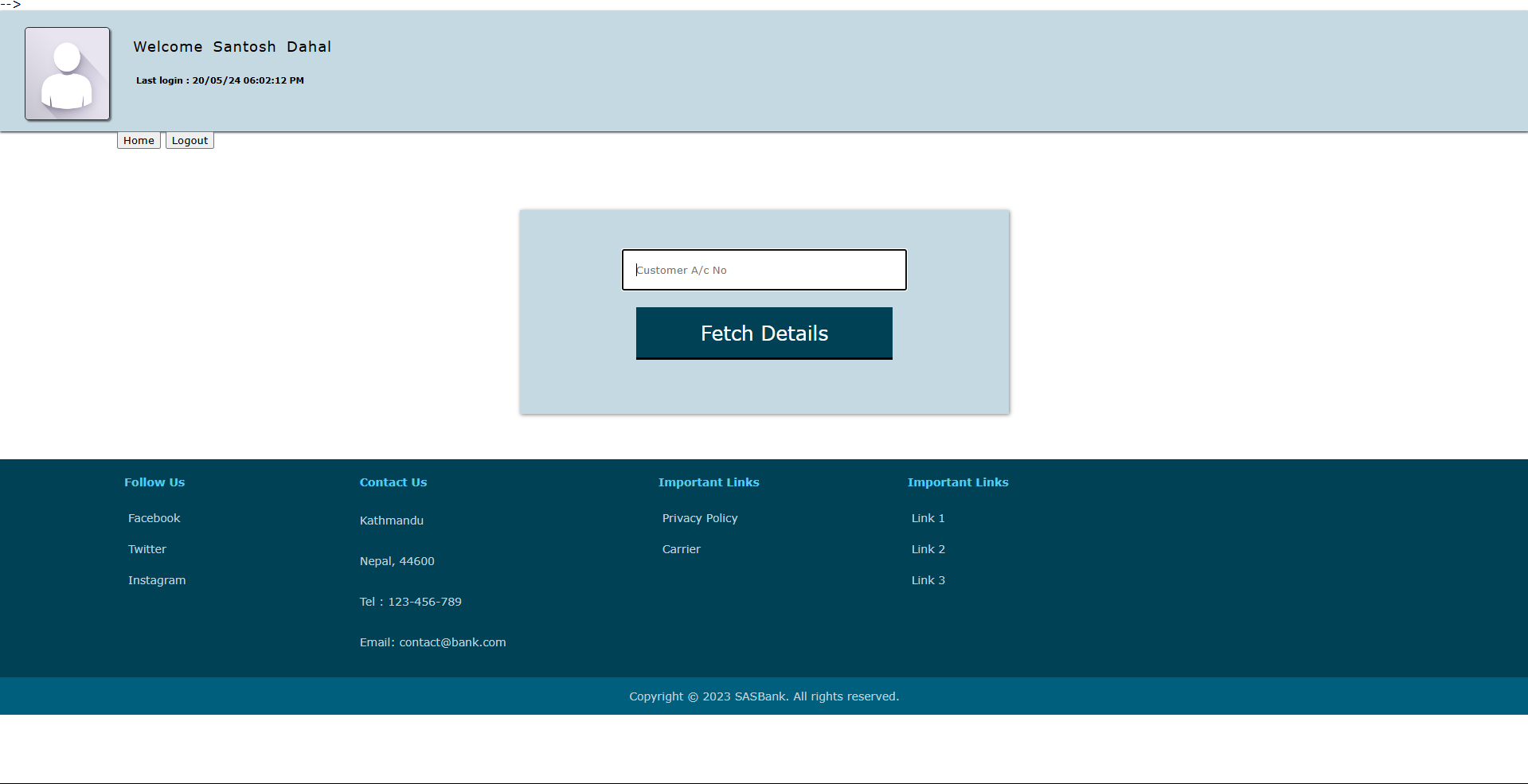
* Approve Pending Account Page



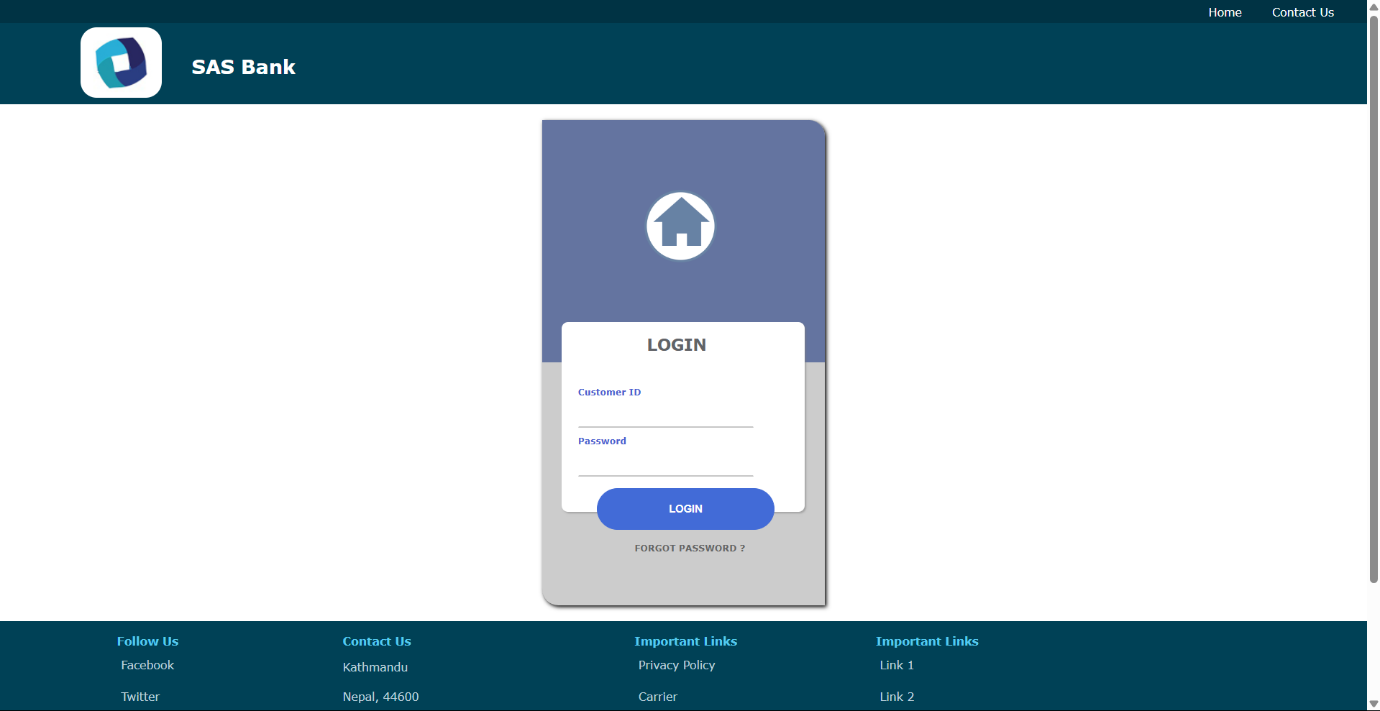
* View Customer Transaction Page



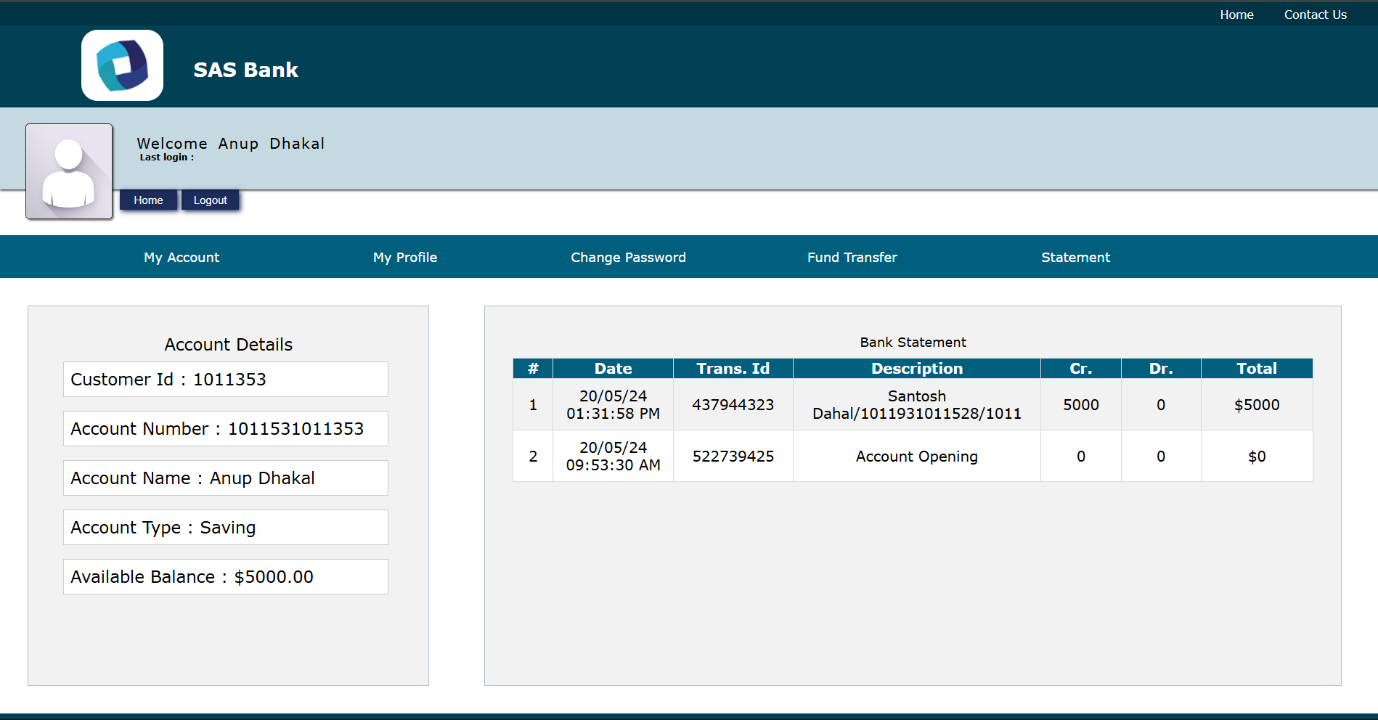
* Credit Customer Page



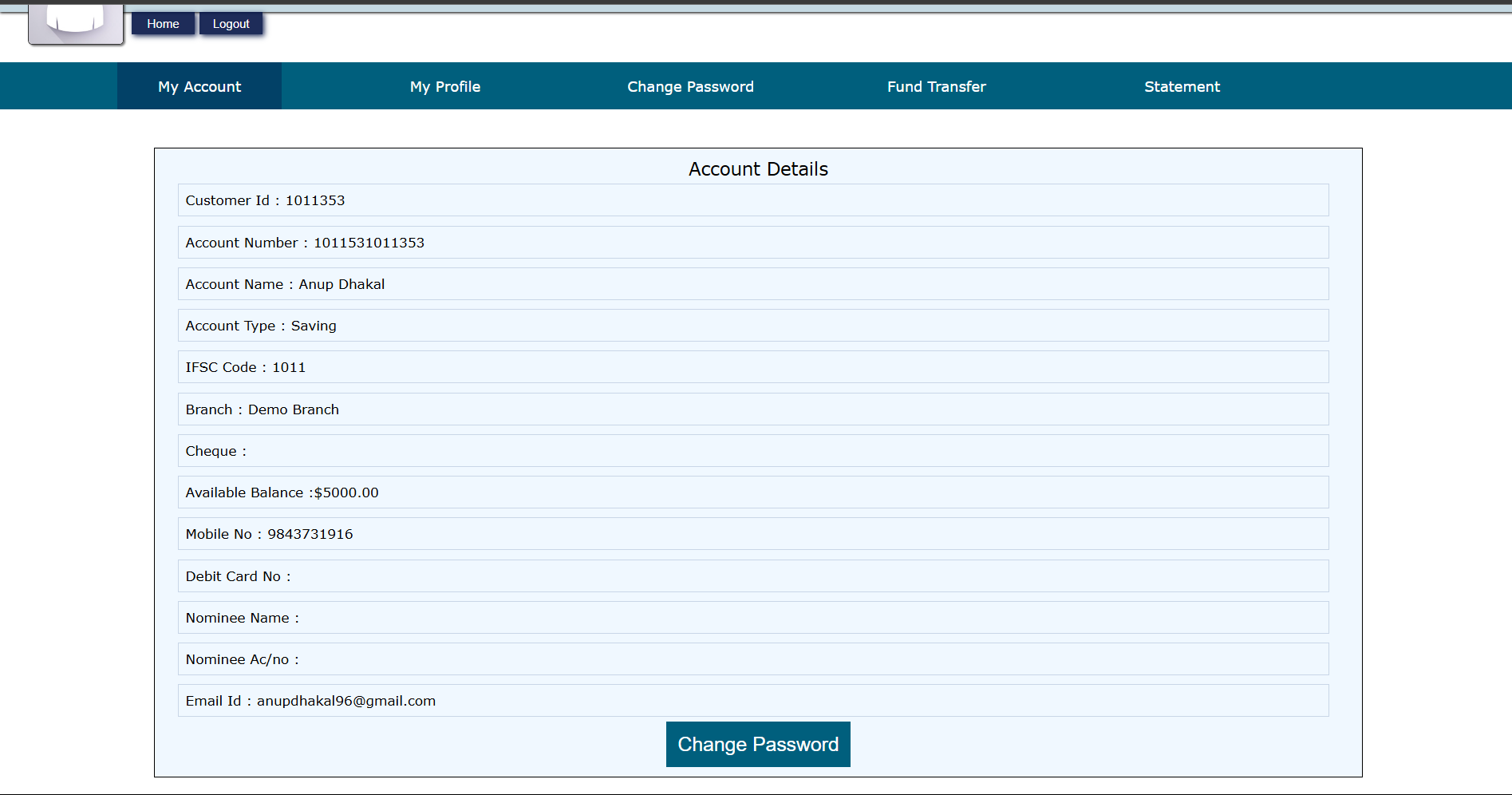
* Customer Panel Overview
* Login page



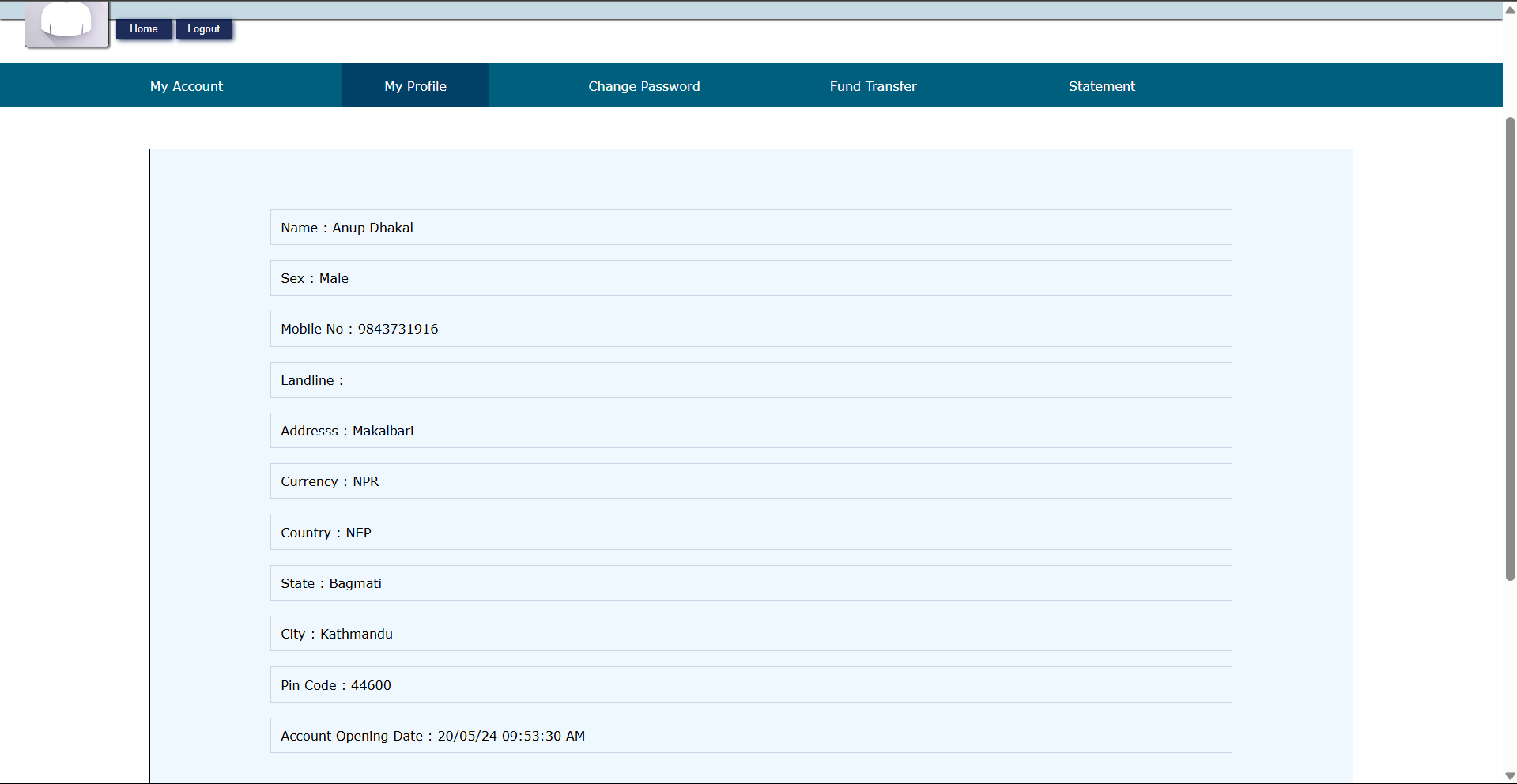
* Dashboard



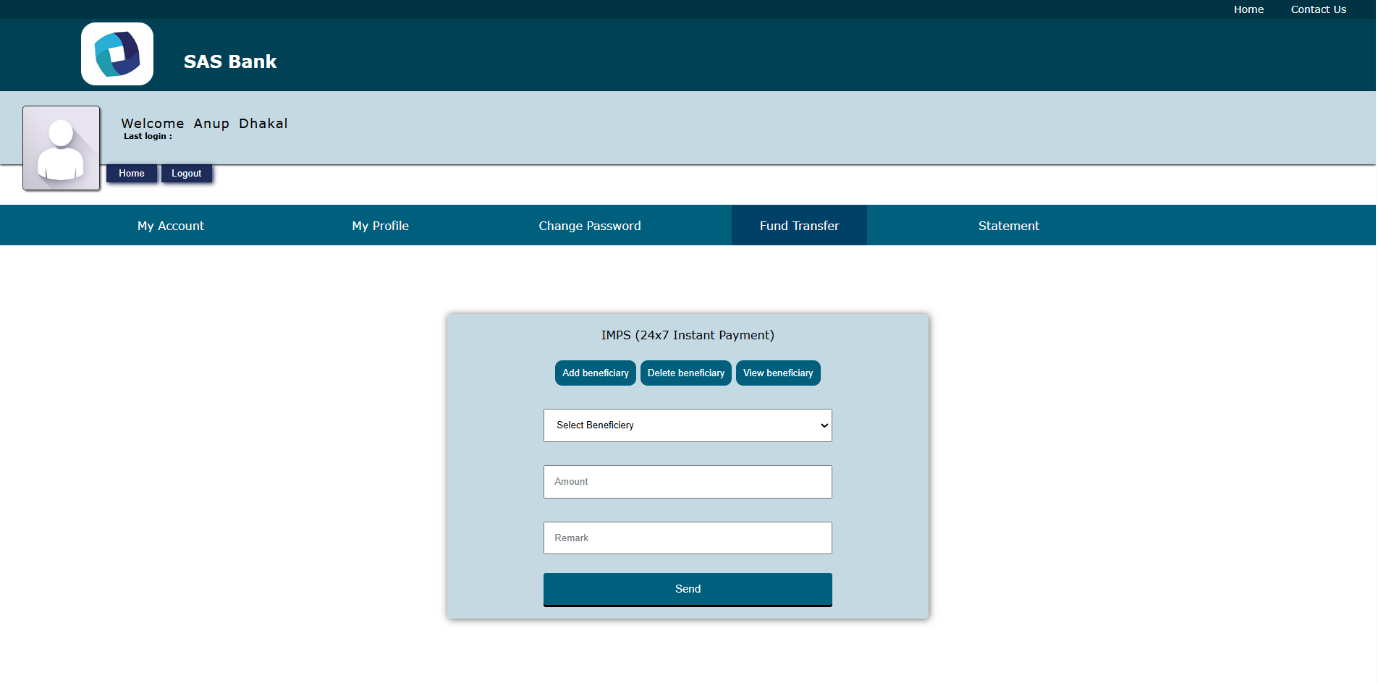
* My Account Section



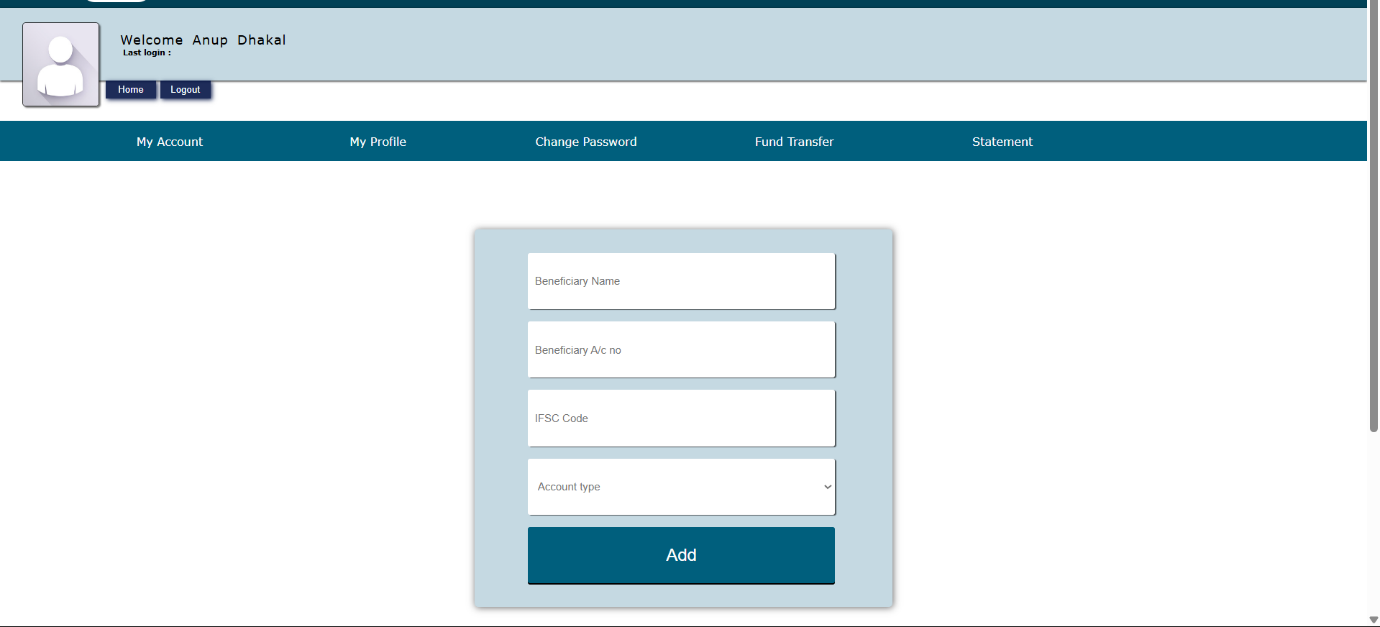
* My Profile Section



* Fund Transfer Section



* Add Beneficiary Section

****

* Statements Section

