**MOBILE BANKING SYSTEM**

**HIGH LEVEL DESIGN**

|  |
| --- |
| **MOBILE BANKING SYSTEM** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GUIDED BY-Mr.Biswa P Das** |  |  |  |  |
| **Date** | **version** | **Author** | **Brief Description of Changes** | **Approver Signature** |
| October19,2022 | 1.0 | Deepika, Himanshu, Thanuja, Sradha, Ritvik |  |  |
|  |  |  |  |  |

**Introduction**

**1 Purpose:**

Changing scenario has prompted customers to do banking from the safety of their home. Gone are the days when one needed to go to the bank, stand in long queues, waste valuable time and do transactions.

The present day banking system is more home based/mobile based. The effort is to minimize the banking time and maximize the level of service.

This mobile banking project is one that provides various banking functionality to a user with ease on his mobile phone. It allows the user to check his/her balance, view latest transactions, transfer funds, request for cheque book etc.

**1.1 Intended Audience:** This document is intended to be read by Customer.

**1.2 Acronyms/Abbreviations:**

|  |  |
| --- | --- |
| CUSTOMER | User |
| BANKER | Admin manages the customer details |

**1.3 Project Purpose:**

The purpose of this document is to show the requirements for the Online Banking System, in which customers can access all the banking facilities under one roof.

**1.4 Key Project Objectives:**

* Create and register for new account
* Validate password
* Make Transaction
* Request cheque book
* Check Transaction History
* Update personal information
* RD\_FD Accounts

**1.5 Project Scope :**

With the help of online banking, there are several indispensable services which are made available to customers, without them having to personally visit the bank.

Customers can perform financial transactions like transfer funds online and open a savings account among various other transactions.

Under non-financial transactions, customers can carry out several activities which may require going to the bank like applying for a new cheque book, getting account statements, and updating contact information.

**1.6 Functional Overview**

**Banker**

1. int customer\_maintenance() - This function is called by bank employees to manage all the customer data.

* int add\_data()- By using this function bank employees can add the credentials of a new customer.
* int delete\_data()- By using this function a banker can delete the record of existing customers.
* int edit\_data()- By using this function the banker can alter the details of the existing customer.
* int daily\_transaction() - By calling this function, the bank employee can get the report of all the transactions that has been done by all the customers on a particular day.
* int report\_min\_bal()- By using this function, a banker can get the report of all the accounts whose balance is low and generates a remainder for all that particular account.

**Customer**

(A)int customer()- By selecting this function from the main menu , the

Customer can get the view of all the facilities which he can avail.

* int reg\_process()- by calling this function customer can register himself for the registration of a new account by using aadhar card number .

User can either opt for a savings account or can go for a current account. User has to input the first name ,last name,Father name,email id,phone number,address

* int open\_acc()- After successful validation of all the documents ,an 8-digit account number will be generated automatically and the user is requested to set up a password for the same.
* int view\_acc()- By calling this function the user can see his personal details ,account number,available balance,type of account and other details.
* int Open\_RD\_FD()- Here by calling this function a customer can open an Recurring deposit account(RD) or a Fixed Deposit account(FD).the customer will get a separate account number for the new account and which will be linked with the SA or CA of the user.
* int interest\_rate()- using this option the customer can get an annual interest percentage according to RBI guidelines.
* Int close RD\_FD()- If the user wants to close the RD or FD accounts, then by calling this function the user can put a request for the same and after the completion of process the respective amount will be credited to the user’s existing SA or CA and the FD or RD account will be closed.
* int req\_chequebook()- If the user wants a cheque book the user can call this function and after submitting the required details customer can put in a request to issue a cheque book.
* int change\_details()- By calling this function the customer can update his personal details and can even change password.

(C) int transaction()- the user can use this function if he wants to do any type

the transaction

* int deposit()- If the customer wants to deposit an amount to his account , then by selecting this function he can do it easily.
* int withdrawal()-If the customer wants to withdraw an amount to his account , then by selecting this function he can do it easily.
* int trans\_history()- If the customer wants to see his transaction, the by going to this function, he can get a view of the last 10 transactions that have been done through that account.

**Design Overview:**

|  |  |
| --- | --- |
| Module name | Customer Registration |
| Handled by | Ritvik Saini |
| Description | The user register for new bank account |

|  |  |
| --- | --- |
| Module name | Password Validation |
| Handled By | Deepika Arumilli |
| Description | Validate the password which is set by the user |

|  |  |
| --- | --- |
| Module name | Add Data |
| Handled By | Ritvik Saini |
| Description | This is used to add customer record. |

|  |  |
| --- | --- |
| Module name | Edit Data |
| Handled By | Sradha R |
| Description | This is used to edit the existing data |

|  |  |
| --- | --- |
| Module name | Delete Data |
| Handled By | Chandragiri Thanuja |
| Description | This is used to delete the existing data |

|  |  |
| --- | --- |
| Module name | Transaction |
| Handled By | Himanshu Kamboj |
| Description | This module handles all types of transactions done |

|  |  |
| --- | --- |
| Module name | Request Cheque book |
| Handled By | Ritvik Saini |
| Description | This module allows user to issue cheque book request |

|  |  |
| --- | --- |
| Module name | FD and RD accounts |
| Handled By | Deepika Arumilli |
| Description | This module allow user to open a RD or FD account which is linked with existing SA or CA. |

|  |  |
| --- | --- |
| Module name | Transaction history |
| Handled By | Chandragiri Thanuja |
| Description | This module will allow user to see last 10 transaction done on the account |

|  |  |
| --- | --- |
| Module name | Low Balance Remainder |
| Handled By | Chandragiri Thanuja |
| Description | This module is here to alerts user those who are having low account balance |

|  |  |
| --- | --- |
| Module name | Last 10 Transactions |
| Handled By | Himanshu Kamboj, Deepika Arumilli |
| Description | This module displays the last 10 transactions of the customer account. |

|  |  |
| --- | --- |
| Module name | View Data |
| Handled By | Sradha R |
| Description | This module is used to view the customer details. |

|  |  |
| --- | --- |
| Module name | Daily Transaction History |
| Handled By | Himanshu Kamboj, Sradha R |
| Description | This module gives the daily transaction of the customer to the banker, |

**2.1 Design objective**

* Register for new account
* Set up password
* Make transaction
* Open RD and FD accounts
* Request for cheque book
* See last 10 transaction
* Add new data
* Modify the existing data
* Delete the existing data
* Set reminder for account having low balance

**2.2 Design alternative**

We have used dynamic linked lists because dynamic linked lists are faster than accessing direct files.

**2.3 User interface paradigms:**

The Mobile Banking system gives access to users to access the banking facilities in their mobile phones. The account details of a customer is stored in a single file.

The user can register for a new account and after registration, the user can set a password for the same. He can do various transactions and can also see the transaction history.

User can also update about new annual interest percentage issued by RBI and can open RD or FD account accordingly.

Users can also request for a checkbook and also be allowed to update the personal information if required.The user can also make a request to close the account.

**2.4 Error detection /Exception Handling**

New users should register before login or else it displays the no user found. Registered users have to login with valid credentials.

Otherwise, they will get invalid account number or password. After login, the user is able to access the banking facilities and be able to do the required operation.

**2.5 Performance:**

The system will work on the customer terminal. The performance depends on the hardware component of the user’s system.

**2.6 Maintenance:**

Very little maintenance could be required for this setup. An initial configuration will be the only system required interaction after the system is put together.

The only other user maintenance would be any changes to settings after setup, and any specified special cases where user settings or history need to be changed.

Physical maintenance on the system’s parts may be required, and would result in temporary loss of data or Internet. Upgrades of hardware and software should have little effect on this project but may result in downtime.

**3.Environment Description:**

**3.1 Time Zone Support:** IST- Kolkata

**3.2 Language Support:**  English

**3.3 User Desktop Requirements:**

a. 64-bit processor, 1 GHz or faster

b. At least 2 GB free hard drive space

c. At least 1 GB RAM

**3.4 Integration Requirements:**

1. Language: C

2. Tools: GDB , Valgrind, Makefile ,Splint ,vi editor

3. Compiler: gcc

4. Environment : Linux

**3.5 Configuration:**

**3.5.1: Operating System**: Any flavor of Linux.