**Online Banking System**

Sprint 1 and Sprint 2

***Submitted by***

Vaka Satwik Reddy BL.EN.U4AIE20070 Scrum Team

Varun Vinod Kulkarni BL.EN.U4AIE20071 Product Owner

Vimal c BL.EN.U4AIE20073 Scrum Team

Vishal s BL.EN.U4AIE20074 Scrum Master

Vamsi Krishna Vunnam BL.EN.U4AIE20075 Scrum Team

***In partial fulfilment of the requirements for the completion of Agile Use cases for***

**Project-based Course**

**19CSE314 Software Engneering**



AMRITA SCHOOL OF ENGINEERING, BENGALURU

AMRITA VISHWA VIDYAPEETHAM

**BENGALURU 560035**

May-2022

**Abstract**

The bank management system project is a program that keeps track of a customer’s bank account. It develops a project for resolving a customer’s financial applications in a banking environment to meet the needs of an end banking user by providing multiple ways to complete banking chores. This project’s main goal is to create software for a bank account management system. The goal of the bank management system project is to create an organic and optimal software of interaction between the various banking components. Depending on the bank’s policies, bank personnel and/or customers can utilize the Banking Management System. It can be utilized by multiple users if they have the necessary permissions.

The system allows customers to view their account balances, transaction history, and transfer funds between accounts. They can also pay bills, request loans, and update personal information. The online banking system uses advanced security measures such as authentication and encryption to protect customers' personal and financial information. Customers can access the system using their unique login credentials, and the system logs them out after a period of inactivity to prevent unauthorized access. Online banking systems have revolutionized the way people bank, reducing the need for physical visits to a bank branch and enabling customers to carry out transactions quickly and efficiently. The system is also environmentally friendly, as it reduces the use of paper-based transactions and documentation.

**Software Requirements**

Front End Languages: HTML, CSS

Back End Languages: Microsoft SQL, PHP

Operating system : Windows XP/7/10 (4Gb ram)

IDE: Visual Studio Code

Database: Microsoft SQL Server Management studio (XAMPP)

**Sprint: 1**

Should allow display of multiple accounts with balances and recent transactions.

* Manage the header functions.
  + Login Management
  + Alert/Messages Management
  + Account Management
  + Profile Management
* Upon clicking on login button or any button that requires the customer to be logged-in
  + open the login/register page.
  + enable logging-in [user-id, password, OTP (if applicable)]
  + upon successful login, redirect to the page intended.
  + upon failure, redirect to exception processing (retry, forgot password or register)
* Upon clicking the Alert/Messages Button
  + display the current list of alerts/messages/interactions.
* Upon clicking the Account Management Button
  + Show the account categories (savings, home loan, investments, insurance, credit cart, etc.) currently linked to this login, and for each category
    - allow expand to view the linked accounts in that category, along with the current balance
* Upon clicking the Profile Management Button
  + Show the personal information collect and allow options for editing (name, address, Dobb, current password, preferences)
  + Upon saving show the updated details

**Sprint: 2**

Should allow the user to perform a few banking transactions.

* Manage the account functions.
  + Transaction History
  + Fund Transfer
  + Make Payment
* Transaction History
  + Show the transaction history, for each transaction.
    - display additional information.
* Fund Transfer
  + Show the list of accounts from which funds can be transferred (savings, current)
  + Upon clicking Add new payment,
    - collect the information required for setting up a payee (Name, Bank Name, Branch Name, Account Number, etc.)
  + Upon selecting the payee
    - enter the amount that needs to be transferred
    - indicate if a standing instruction needs to be placed for recurring payments (periodicity and number of instances)
    - date on which the transfer should take place (or start for recurring payments)
    - Upon clicking the confirm button, show confirmation of request.
* Make a payment.

Upon clicking the Make a Payment Button, pay Bill, allowing payment from

* + - Saving Accounts
  + Select the account from which payment should be made.
  + Check if sufficient balance is available in the account.
  + Upon clicking the confirm button, process the payment.
* Loan Application
* User can apply for a loan.
* Different types of loan (like agriculture, education, business)
* Employee management
* Employee for deposit of money in customer’s table
* They will have a separate login and profile management.
* Employee can be able to see the Transactions done by all customers.