Date:

BANK MANAGEMENT SYSTEM

Overview:

The Bank Management System project is a comprehensive software solution designed to manage the daily operations of a bank. The system utilizes Python as the programming language, GUI (Graphical User Interface) for user interaction, and SQL as the database management system for storing and retrieving data. The primary objective of this project is to create a user-friendly and efficient system for bank employees to manage customer accounts, transactions, and reports.

The main objective of our project is to develop a GUI based application using python library(tkinter) and uses my SQL database to store the data. This project help the bank organization to provide seamless connectivity through internal sub domain. The Hardware requirement for this project is computer with a minimum of 2 GB RAM and 500 MB free disk space and Windows, Linux, or macOS as an Operating System. Software requirement are Python 3.x (latest version)
GUI Library: Tkinter.
SQL Database Management System: MySQL or SQLite
Python libraries: mysql-connector-python or sqlite3 for database connectivity
Coding Platform: Jupyter Notebook

Basic Functionality of our Project are:

1. Customer Registration.

This is the first module where users are registered to the bank. The registered customer detail are feeded into the bank database from its GUI interface. We have also provided validation functionality using python validator for each and every field so to provide insertion of fake or incorrect data therefore increases system performance and only allow authentic users for registration making our application secure and robust in all aspects. After successful registration system automatically generate a number which act as a account number for users.

2.Customer Login

This is second module where the users enter their credential to login to their dashboard and show their personal detail with the help of this module registered customer can apply for loan or perform multiple functions. Customer can check their balance deposited or withdrawal. This module help customer to stay connected with the bank.

3.Manager Login:

This module perform crucial task which include taking bigger decision such aa loan approvals maintaining staff or other official task.

4.Cashier Login:

This module is responsible for depositing and withdrawal of money from customer accounts. Hence Customer Login is directly connected with the cashier module any alter or changes in cashier module directly or indirectly affect customer login module. Cashier is also responsible for loan registration for the customers. The initial process of loan procedure start from cashier module i.e cashier help customer to enrolled for loan procedure their after manger may approve or disapprove the loan.

Date:

Detail of File:

Bankmanagement.py:

Basically in this file all the GUI as well as python backend code is written. This file mainly consists of database connection code and code performing CRUD(Create, Read, Update, Delete) operation.
Different module file are as follows:

1. lines no. 14 to 108 focuses on handling the customer registration thereby defining register_page (), and loginbtn() as a function. That ensure proper registration of users. From line 26 to 50 we also uses python validator for form validation for reference check to the mentioned linked at the bottom of this page. After all validation the data are inserted into database using code from line no.53 to 64. From lines 67 to 107 GUI code for customer is written.

2. from line no. 156 to 231 focuses in functionality of dashboard. The detail description is as follows:
In open_customer_dashboard(customer_data) function multiple functions are defined that perform different functionality for example function apply_loan() help customer to apply for loan and function submit_loan_application() help to register user for loan.Thereafter from line 218-230 GUI for customer dashboard. In this module multiple small function are declared such as update_address() and update_pin() for updation and deletion function.

3.Similarly from lines 234 onward individual dashboard and functions for manager cashier are declared which perform their functions some of the important functions are discussed below.

- Handle_request() function in manager dashboard directly fetched data from customer dashboard i.e both module are linked directly to convey messages.
- view_approved_loans() function in cashier module also establish direct connection with manger module and function if manger approve loan through their data cashier can show in their dashboard either the loan is approved or not. Lines 388 onward is the code for above mentioned features.

DATABASE

Database.sql

In this file a database name “bank” is created having multiple tables created such as

- cust_detail(to store customer data)
- managers(Hold manager credentials)
- cashiers(Hold cashier credential)
- loans(to hold loan amount of specific users)

In manager and cashier we have fixed their crediential imtially as:
insert into managers(FirstName, LastName,Username,PIN) values("Gulshan","Ray","manager", "manager_password");
similarly for cashier.
insert into cashiers(FirstName,LastName,UserName,PIN) values("Sahil","Raj","cashier", "cashier_password");

Link: https://github.com/gulshan-ray123/Bank_management-System/tree/main

Date:

SCREENSHOTS:

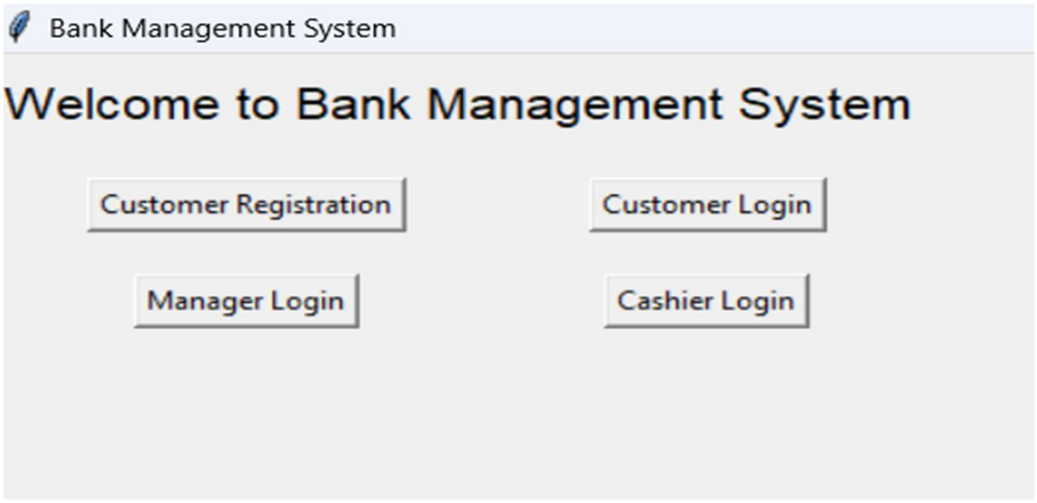

Fig 1. UI interface of our project.

Fig2. Customer registration form.

Date:

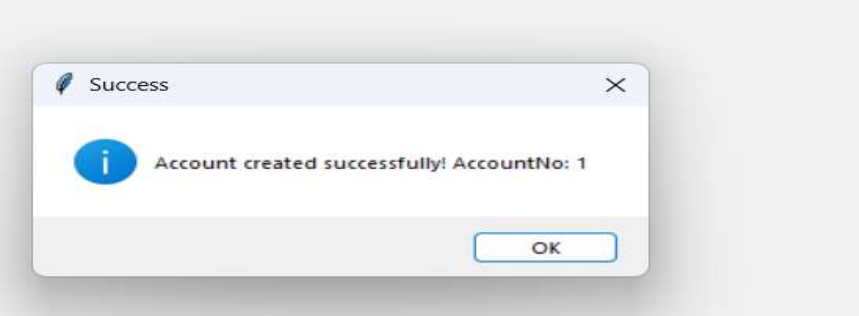

Fig3. After successful account creation account no. automatically generated.

Fig 4. Login page for customer.

Fig 5. Customer Dashboard.

Date:

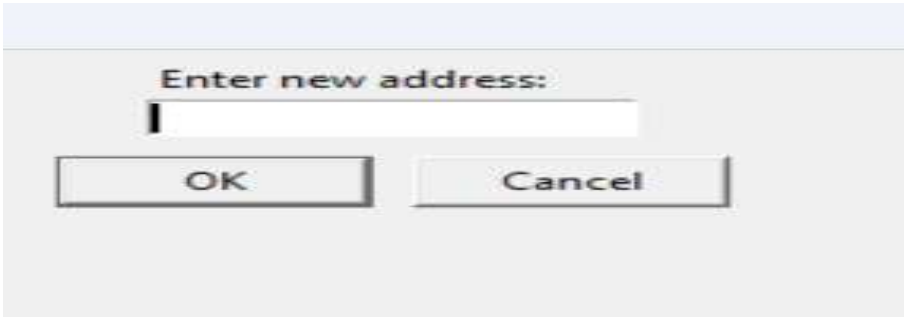

Fig 6.Updating form for address

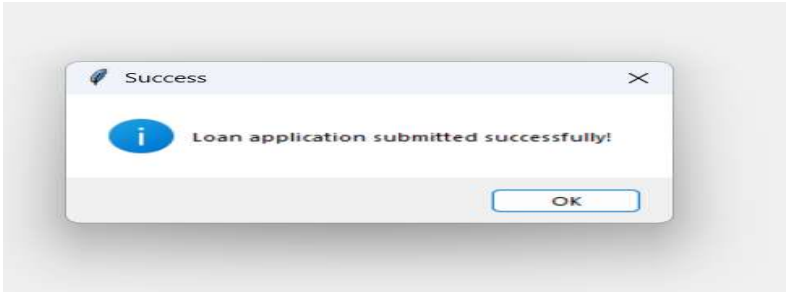

Fig 7. Applying Loan

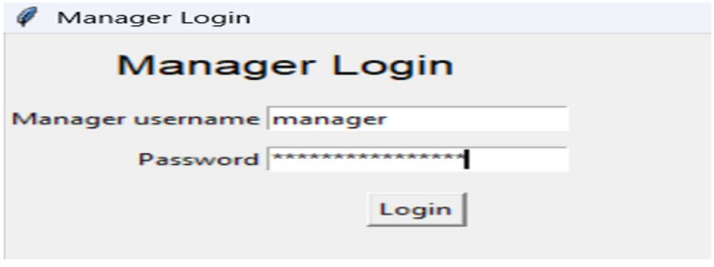

Fig 8. Manager Dashboard after login.

Date:

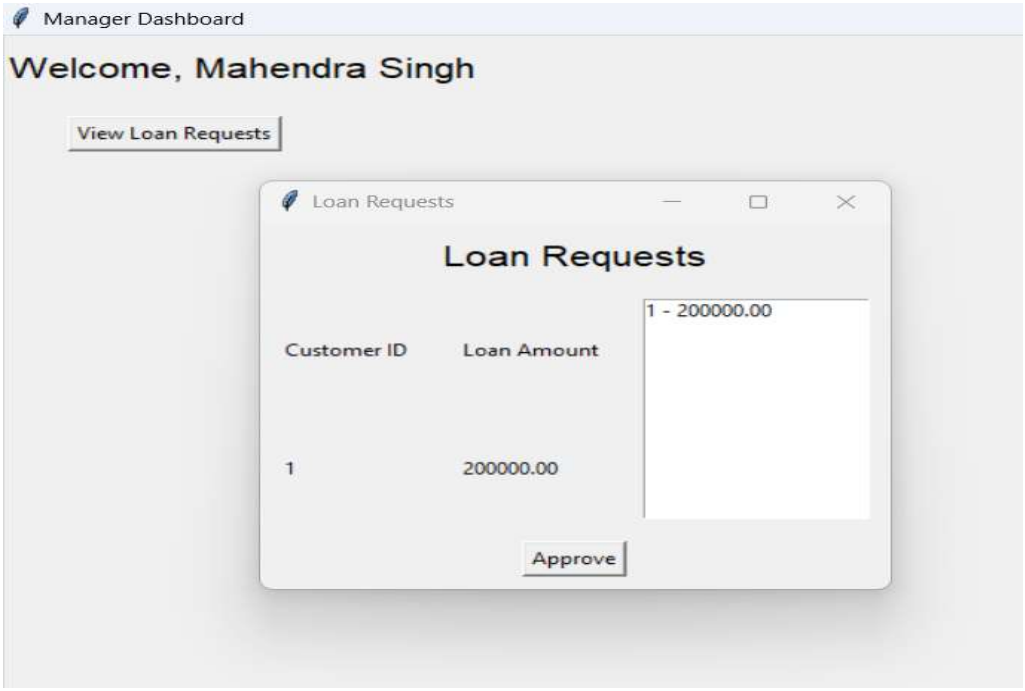

Fig 9. Manager showing loan request transferred from module customer dashboard.

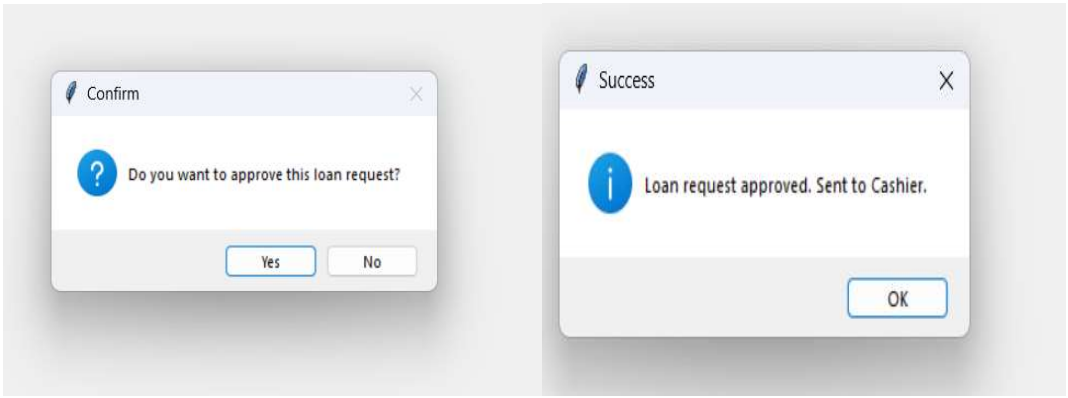

Fig 10. Confirmation of loan request.

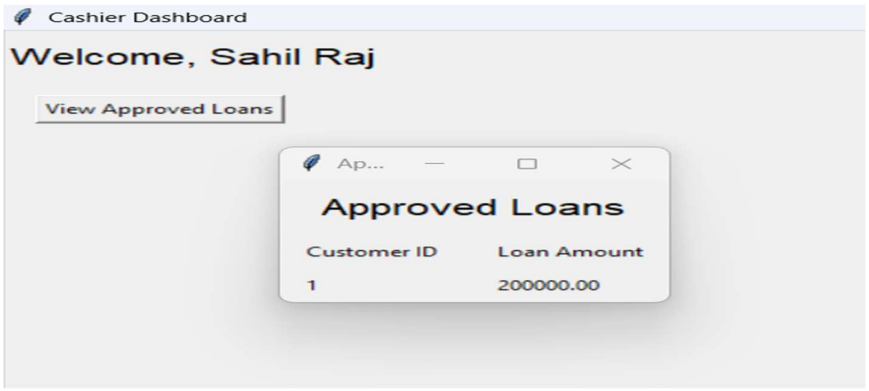

Fig11. Cashier dashboard where he can show loan approval.