

# **CALICUT CO-OPERATIVE BANK-LOAN MANAGEMENT SYSTEM**

**PROJECT THESIS**

**SUBMITTED**

**TO**

**AWH ENGINEERING COLLEGE  
KUTTIKKATTOOR, KOZHIKODE**

**IN PARTIAL FULFILMENT  
OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE  
OF**

**Master Of Computer Applications**

**BY**

**SIVAKAMI K**



**DEPARTMENT OF COMPUTER APPLICATIONS  
AWH ENGINEERING COLLEGE KUTTIKKATTOOR,  
KOZHIKODE  
DECEMBER 2023**

# DEPARTMENT OF COMPUTER APPLICATIONS



## AWH ENGINEERING COLLEGE KOZHIKODE

### CERTIFICATE

*This is to certify that this thesis entitled “CALICUT CO-OPERATIVE BANK-LOAN MANAGEMENT SYSTEM” submitted herewith is an authentic record of the thesis work done by SIVAKAMI K (AWH22MCA-2039) under our guidance in partial fulfillment of the requirements for the award of Master of Computer Applications from APJ Abdul Kalam Technological University during the academic year 2023.*

**Mrs. Sruti Sudevan.**

Assistant Professor

Dept. of Computer Applications

Head of the Department

**Mrs. Keerthi Sankar K.**

Assistant Professor

Dept. of Computer Applications

Project Guide

**External Examiner**

**Internal Examiner**

## **ACKNOWLEDGEMENT**

I express my sincere gratitude to our beloved principal **Dr. Sabeena M V** for providing me an opportunity with the required facilities for doing this project. I express my hearty thanks to **Mrs. Sruti Sudevan**, Head of the department of Computer Application, **Mrs. Keerthi Sankar K**, Assistant Professor for her guidance. I am thankful to all other staff of the MCA department for their encouragement, timely guidance, valuable suggestions and inspiring ideas given throughout this project. I am grateful to my friends for the way they have cooperated, expected me to achieve success and have always stirred my ambition to do the best. Above all, I am grateful to the almighty, who has showered His blessings on me throughout my life and throughout the project.

**SIVAKAMI K**

## **ABSTRACT**

The main objective of developing this project titled 'CALICUT CO-OPERATIVE BANK-LOAN MANAGEMENT SYSTEM' (CCB-LM) is to handle the all details of Loans in the bank. The project has been developed to smoothen the processing of Loans in banks. Customer can apply for a loan and after approved it they can track their details from online. Bank loan management system is an interface which facilitates a customer to apply for a loan online and to track the status from time-to-time. This system provides detail about the customers, their loan details, EMI details and its rate details. Getting a loan is a very tiring and complicated process in India. It may take weeks even months for loans to get approved and people have to visit the loan office again and again for document and verification. Using with this system admin can find customer details easily and it's a paperless system so workload is reduced. It is very helpful for those banking staffs who are in the charge of loan management, it provides a very reliable and convenient form for every loan and EMI related transaction and their related details. Interest rates and the loan details are also available at the click of a mouse. Bank loan management system is unique in such a way, it not only helps the customers but also the loan agency to check the pending, assign it to departments, complete the formalities and procedures between the departments and arrive at decisions to very fact.

# CONTENTS

	PAGE NO
1. INTRODUCTION	1
2. SYSTEM ANALYSIS	3
2.1 Existing System	4
2.2 Proposed System	4
2.3 Module Description	5
2.4 Sprint	6
2.5 User Stories	8
3. FEASIBILITY STUDY	9
3.1 Economical Feasibility	10
3.2 Technical Feasibility	10
3.3 Operational Feasibility	10
3.4 Software Feasibility	10
4. SOFTWARE ENGINEERING PARADIGM	11
4.1 Agile Model	12
4.2 Scrum	12
5. SYSTEM REQUIREMENT SPECIFICATIONS	13
5.1 Software Requirements	14
5.2 Hardware Requirements	14
6. SYSTEM DESIGN	15
6.1 Database Design	16
6.2 Tables	17
6.3 UML Design	19
6.4 Use Case Diagram	20
6.5 Scenario	21
6.6 Sequence Diagram	22
7. SYSTEM DEVELOPMENT	24
7.1 Coding	25
8. SYSTEM TESTING AND IMPLEMENTATION	26
8.1 Types of Testing	27
8.2 Implementation	28

9. SYSTEM MAINTENANCE	29
10. FUTURE ENHANCEMENT	31
11. CONCLUSION	33
12. APPENDIX	35
13. BIBLIOGRAPHY	44

# **INTRODUCTION**

## 1. INTRODUCTION

In the ever-evolving landscape of financial services, the efficient management of loans is paramount for both customers and financial institutions. Recognizing the complexities and time-consuming nature of loan processing, the 'CALICUT CO-OPERATIVE BANK-LOAN MANAGEMENT SYSTEM' (CCB-LM) emerges as a pivotal solution designed to streamline and enhance the entire loan management process within the banking sector. This project aims to revolutionize the traditional approach to loan applications by introducing a seamless online platform. Its primary objective is to empower customers to effortlessly apply for loans and, once approved, conveniently monitor the status of their applications in real-time. The Bank Loan Management System serves as an intuitive interface, providing comprehensive details about customers, their loan agreements, EMI (Equated Monthly Installment) specifics, and associated interest rates.

In essence, the 'CALICUT CO-OPERATIVE BANK-LOAN MANAGEMENT SYSTEM' is a groundbreaking initiative that not only benefits customers by simplifying their loan journey but also empowers banking institutions to operate with increased efficiency, transparency, and responsiveness in the dynamic realm of loan management.



# SYSTEM ANALYSIS

## **2. SYSTEM ANALYSIS**

### **2.1 Existing System**

At present the finance division does not have any automated system to manage the data of Loans. They have to rely on manual system which is time consuming. Existing system doesn't maintain details automatically and can't track details easily. It doesn't allow the customer to check their profile in proper way which leads customer dissatisfaction and any addition, deletion or modification to any data or retrieval of data requires lot of time and effort. Existing system doesn't contains functionalities of fast retrieval information such as customer details and maintenance of all the loan details so it involves lots of paperwork. Apart from administrative task being cumbersome, manual system of registration is also long and error-prone. Lots of time is required to manage customer info & details so it feels that existing system not accurate and therefore maintenance becomes very complicate. It used to take more time to find customers because there are required extra manual effort such as to hire candidate. It is tedious process as one has to inform the users to fill the correct information. There is no provision to prevent incorrect data entry in form and the data masters.

### **2.2 Proposed System**

The proposed system is a web based automated application and maintains a centralized repository of all related information. The system maintains the records of the loan details for every user. The customer can directly apply for a loan type from the list available. Here once the customer fills basic enquiry form, it reaches the bank server so he gets a login id password. The application is received by Admin to approval. The administrator now can check it and select whether to go forward or reject the candidate. This system can be controlled by the administrator.

Now the customer gets approval by verifying the uploaded documents to the site through his login. The customer just needs to upload needed documents online and can also track loan status. Once the loan is approved the customer need to submit a copy of documents uploaded online at the bank for verification before receiving loan amount.

## 2.3 Module Description

This project has 2 modules:

### ADMIN

- Login
- Manage Registrations
- Upload Loan Details
- Manage Loan Application
- Manage EMI History Details
- View Feedback

### USER

- Register
- Login
- Manage Profile
- View Loan Details
- Apply for Loan scheme
- View Loan Status
- View EMI History Details
- Add Feedback

## 2.4 Sprint

### Sprint 1

Module	Task	Pending Task of Any	Hours For Completion	Expected Date of Completion	Actual Date of Completion	Reason For Deviation
Requirement analysis	Observation and brain storming					
	Mvc architecture					
	Use case					
	Sequence diagram					
	User stories					
Admin, User	Table design					
	Installing packages					
	Login					
	Testing					

**Sprint 2**

Module	Task	Pending Task of Any	Hours For Completion	Expected Date of Completion	Actual Date of Completion	Reason For Deviation
Admin	Login					
	Manage Registrations					
	Upload Loan Details					
	Manage Loan Application					
	Manage EMI History Details					
	View Feedback					

**Sprint 3**

Module	Task	Pending Task of Any	Hours For Completion	Expected Date of Completion	Actual Date of Completion	Reason For Deviation
User	Register					
	Login					
	Manage Profile					
	View Loans					
	Apply Loan					
	View Loan Status					
	View EMI Detail					
	View Feedback					

## 2.5 User Stories

Calicut Co-operative Bank-Loan Management System is a web application which consists of two modules Admin and User. Admin should be responsible for uploading different loan details in the application. The admin should also be responsible for managing Users by authentication. The admin should be able to view the loan application of users and verify it. Then the admin can update the EMI details of the loan applied by the users and the can also view the user's feedback.

User should be able to sign-up by providing personal information and credentials and then log in to account using username and password. User should be able to manage profile. The user should also be able to view loan details, apply for loan scheme. Then the user should be able to view loan status and should view EMI history details. Finally the user should be able to add feedback.

# **FEASIBILITY STUDY**

### **3. FEASIBILITY STUDY**

An analysis of the ability to complete a project successfully, taking into account legal, economic, technological, scheduling, and other factors is considered a feasibility study. Rather than just diving into a project and hoping for the best, feasibility study allows project managers to investigate the possible negative and positive outcomes of a project before investing too much money and time.

#### **3.1 Economic Feasibility**

The economic requirements evaluate the costs involved in developing and implementing the system versus the benefits it will bring, such as time savings, reduced paperwork, and improved efficiency.

#### **3.2 Technical Feasibility**

The technical requirements for the system assess whether the proposed system can be technically implemented with the available technology and infrastructure. It also evaluates if the system can be integrated with existing systems in the bank.

#### **3.3 Operational Feasibility**

Determines whether the bank staff and customers can adapt to the new system easily. Also assess the training needs for the staff to effectively use and manage the loan management system.

#### **3.5 Software Feasibility**

Identifies the operating system's required to run the loan management system and check compatibility with the existing infrastructure. Assess the capability of the chosen DBMS to handle the volume of data generated by loan transactions and customer information.



# **SOFTWARE ENGINEERING PARADIGM**

## **4. SOFTWARE ENGINEERING PARADIGM**

The software engineering paradigm which is also referred to as a software process model or Software Development Life Cycle (SDLC) model is the development strategy that encompasses the process, methods and tools. SDLC describes the period of time that starts with the software system being conceptualized.

### **4.1 Agile Model**

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. At the end of the iteration, a working product is displayed to the customer and important stakeholders. In Agile, the tasks are divided to time boxes (small time frames) to deliver specific features for a release. Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.

### **4.2 Scrum**

Scrum is an agile framework for managing knowledge work, with an emphasis on software development. It is designed for teams of three to nine members, who break their work into actions that can be completed within time boxed iterations, called "sprints", no longer than one month and most commonly two weeks, then track progress and re-plan in 15-minute stand-up meetings, called daily scrums. Scrum is an iterative and incremental framework for managing product development. It defines "a flexible, holistic product development strategy where a development team works as a unit to reach a common goal", challenges assumptions of the "traditional, sequential approach to product development, and enables teams to self organize by encouraging physical co-location or close online collaboration of all team members, as well as daily face-to-face communication among all team members and disciplines involved.

# **SYSTEM REQUIREMENT SPECIFICATION**

## **5. SYSTEM REQUIREMENT SPECIFICATION**

### **5.1 Software Requirements**

- Operating System : Windows 7 or above
- Front End : HTML, CSS, JS
- Back End : Python
- IDE : PyCharm
- Web browser : Internet Explorer/Google Chrome/Firefox

### **5.2 Hardware Requirements**

- Processor : Intel core i3 or above
- Ram : 4 GB
- Storage : 10 GB Hard Disk

# **SYSTEM DESIGN**

## 6. SYSTEM DESIGN

System design is the first in the development phase for many engineered products or systems. It may define the process of applying various techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

### 6.1 Database Design

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database.

#### Normalization

It is a process of converting a relation to a standard form. The process is used to handle the problems that can arise due to data redundancy.

**Normal Forms:** These are the rules for structuring relations that eliminate anomalies.

#### 1. First Normal Form (1NF)

A relation is said to be in first normal form if the values in the relation are atomic for every attribute in the relation.

#### 2. Second Normal Form (2NF)

A relation is said to be in second Normal form if it is in first normal form and it should satisfy any one of the following rules.

- Primary key is not a composite primary key
- No non key attributes are present
- Every non key attribute is fully functionally dependent on a full set of primary keys.

#### 3. Third Normal Form (3NF)

A relation is said to be in third normal form if there exist no transitive dependencies.

#### Transitive Dependency:

If two non-key attributes depend on each other as well on the primary key then they are said to be transitively dependent.

## 6.2 Tables

### Login

Field Name	Type	Width	Constraints
ID	Int	100	Primary Key
Username	Varchar	100	Not Null
Password	Varchar	50	Not Null
Type	Varchar	50	Not Null

### User

Field Name	Type	Width	Constraints
ID	Int	100	Primary Key
Photo	Varchar	500	Not Null
First_Name	Varchar	20	Not Null
Last_Name	Varchar	20	Not Null
Gender	Varchar	10	Not Null
DOB	Date	30	Not Null
Email	Varchar	50	Not Null
Place	Varchar	50	Not Null
State	Varchar	30	Not Null
Country	Varchar	30	Not Null
PIN	Int	10	Not Null
Phone	Varchar	20	Not Null
LOGIN_Id	Int	100	Foreign Key
Status	Varchar	20	Not Null
Acc_No	Varchar	100	Not Null
IFSC	Varchar	30	Not Null

**Loan Details**

Field Name	Type	Width	Constraints
ID	Int	100	Primary Key
Loan_Name	Varchar	50	Not Null
Amount	Float	30	Not Null
Interest	Float	30	Not Null
EMI	Varchar	20	Not Null
Loan_Duration	Varchar	10	Not Null
Date	Date	30	Not Null
Guarantee	Varchar	30	Not Null

**Loan Application**

Field Name	Type	Width	Constraints
ID	Int	100	Primary Key
Aadhaar	Varchar	500	Not Null
Pan_card	Varchar	500	Not Null
Salary_Slip	Varchar	500	Not Null
LOANDETAILS_ Id	Int	100	Foreign Key
USER_Id	Int	100	Foreign Key
Status	Varchar	100	Not Null
Guarantee	Varchar	500	Not Null

**Bank**

Field Name	Type	Width	Constraints
ID	Int	100	Primary Key
Account	Varchar	100	Not Null
IFSC_CODE	Varchar	30	Not Null



**EMI History**

Field Name	Type	Width	Constraints
ID	Int	100	Primary Key
Amount_Received	Float	20	Not Null
Status	Varchar	30	Not Null
EMI_Pending	Int	20	Not Null
Due	Date	30	Not Null
LOANAPPLICATION_Id	Int	100	Foreign Key
USER_Id	Int	100	Foreign Key

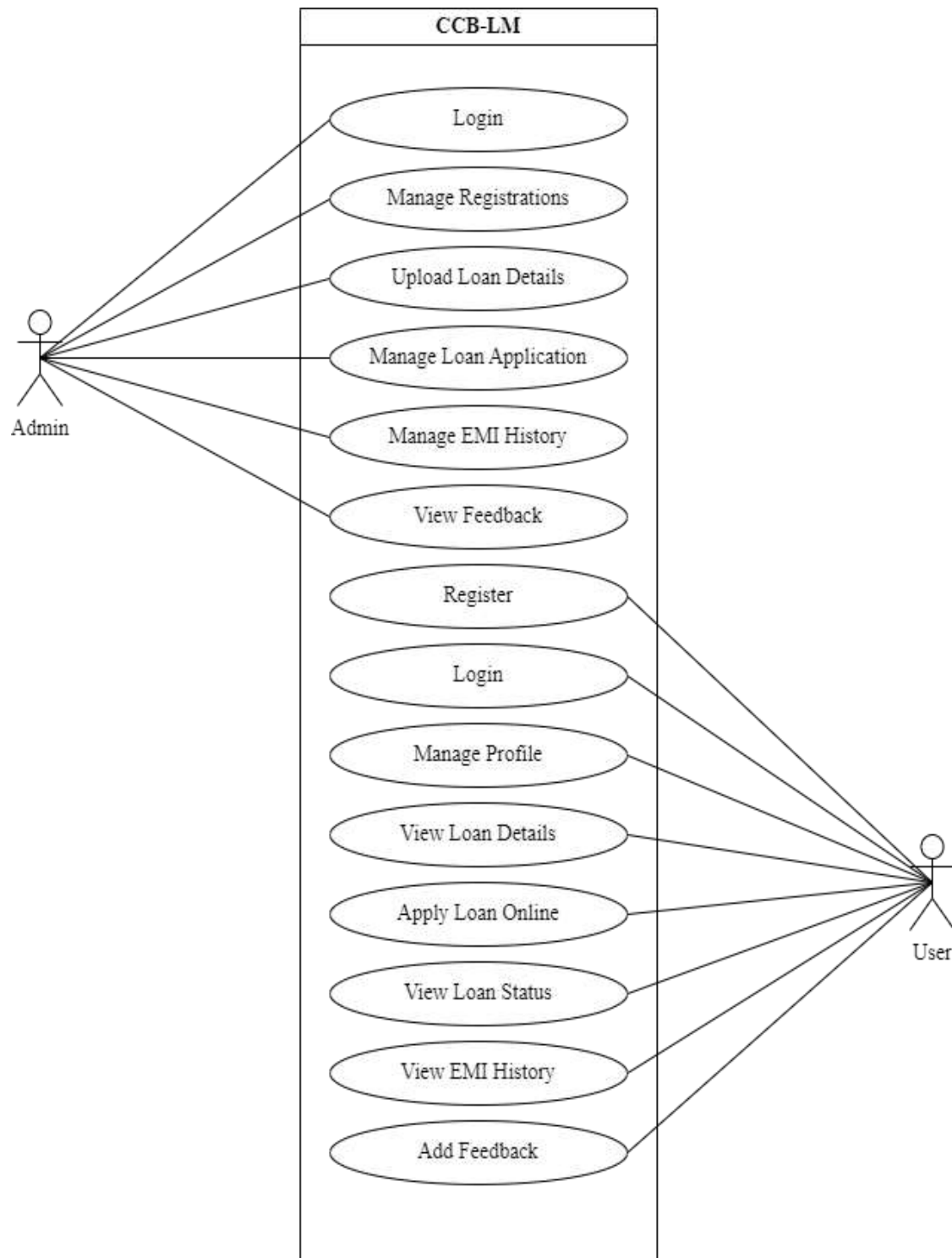
**Feedback**

Field Name	Type	Width	Constraints
ID	Int	100	Primary Key
Date	Date	30	Not Null
Feedback	Varchar	500	Not Null
USER_Id	Int	100	Foreign Key

**6.3 UML Design**

The Unified Modelling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artefacts of the software systems, as well as for business modelling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems. The UML is a very important part of developing object oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

## 6.4 Use Case Diagram



## 6.5 Scenario

### ADMIN

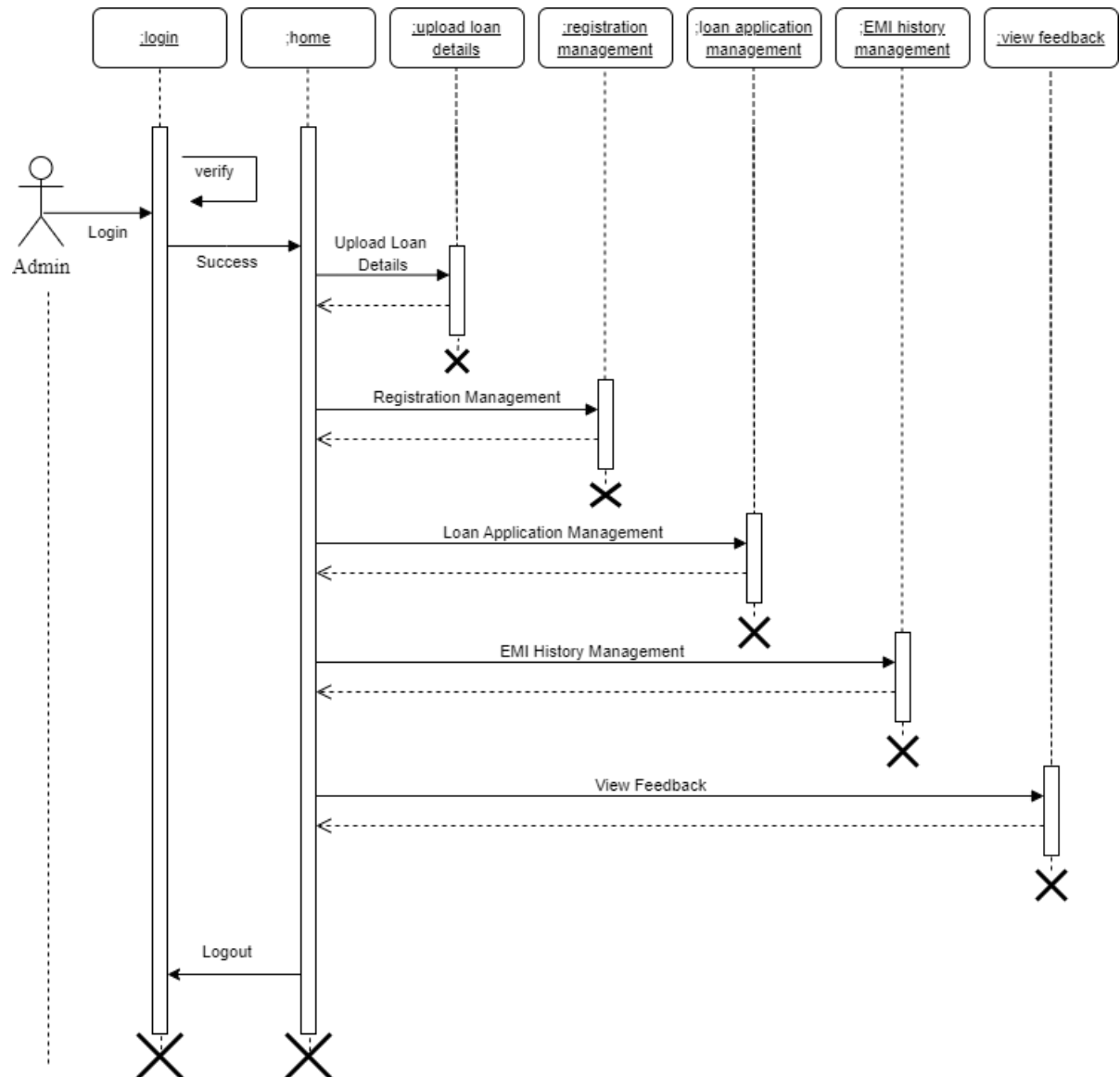
- Registration Management
- Upload Loan Details
- Loan Application Management
- EMI History Management
- View Feedback

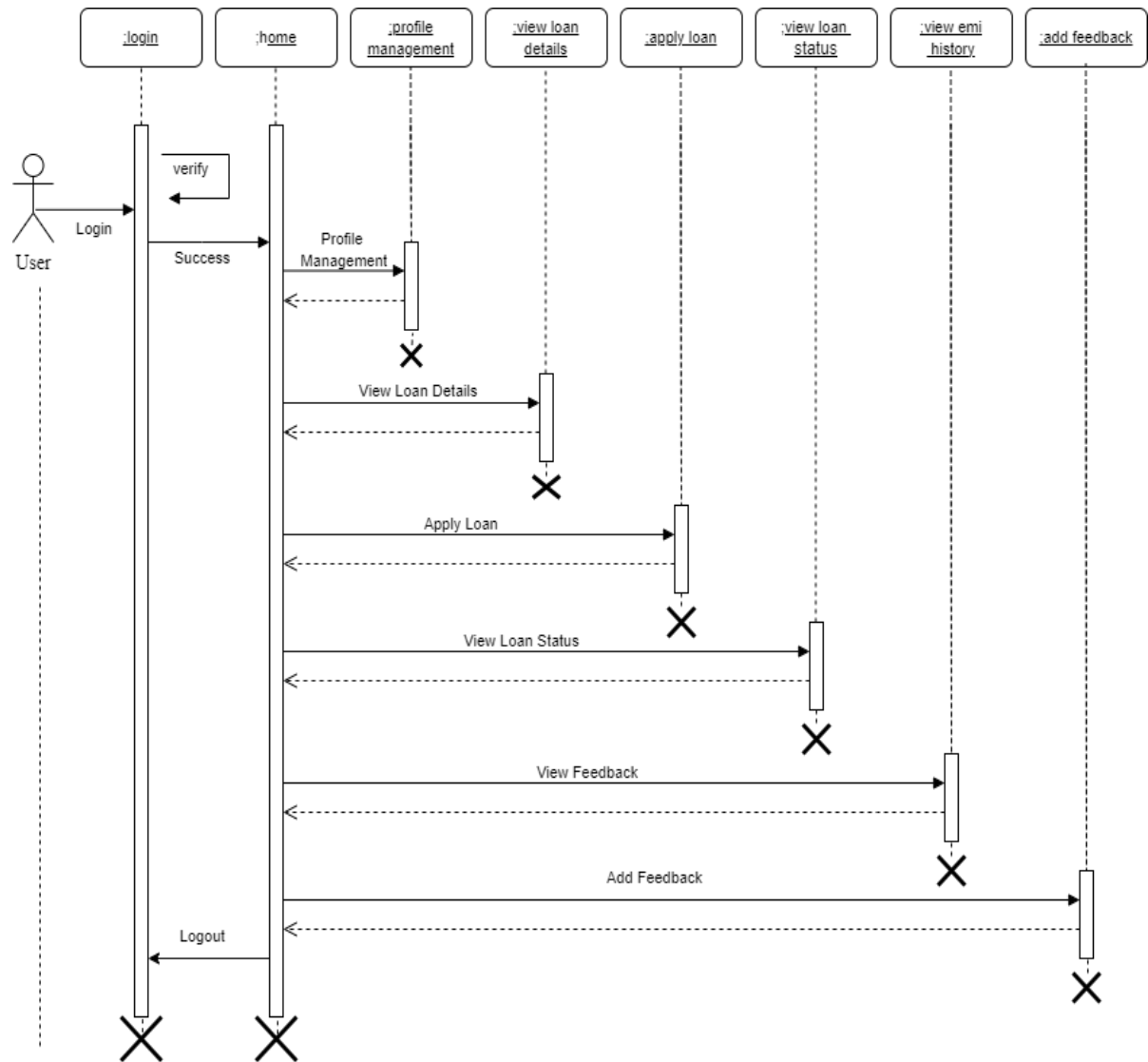
### USER

- Profile Management
- View Loan Details
- Apply for Loan
- View Loan Status
- View EMI History
- Add Feedback

## 6.6 Sequence Diagram

### Admin



**User**

# **SYSTEM DEVELOPMENT**

## **7. SYSTEM DEVELOPMENT**

System development is a series of operations to manipulate data to produce output from a computer system. The principal activities performed during the development phase can be divided into two major related sequences.

### **7.1 Coding**

The purpose of code is to facilitate the identification and retrieval of items of information. A code is an ordered collection of symbols designed to provide unique identification of an entity or an attribute.

#### **Python**

Python is a widely used high-level programming language for general purpose programming, created by Guido van Rossum and first released in 1991. An interpreted language, Python has a design philosophy that emphasises code readability and a syntax that allows programmers to express concepts in fewer lines of code.

#### **Libraries**

Python's large standard library, commonly cited as one of its greatest strengths, provides tools suited to many tasks. For Internet-facing applications, many standard formats and protocols such as MIME and HTTP are supported.

#### **Django**

Django is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel.

#### **MYSQL Client**

Licence GPL Platforms OS Independent Python versions Python 2.7 and 3.4+ PyPI.  
<https://pypi.org/project/mysqlclient/> MySQL client is a fork of MySQL python. It adds Python 3 support and fixes many bugs. It is the MySQL library that is recommended by the Django documentation

# **SYSTEM TESTING AND IMPLEMENTATION**



## **8. SYSTEM TESTING AND IMPLEMENTATION**

Testing is vital to the success of the system. It makes a logical assumption that if all the parts of the system are correct, the goal will be successfully achieved in this project. It is the stage of implementation, which ensures that the system works accurately and effectively before the live operation commences.

### **8.1 Types of Testing**

Different types of testing are:

- Unit testing
- Integration testing
- System testing
- Validation testing

#### **Unit Testing**

Verifies the correctness of individual units or components of the software. Ensures that each function, procedure, or module works as intended. Tests individual functions like loan application submission, approval logic, and EMI calculation.

#### **Integration Testing**

Verifies the interaction between different components and ensure they work seamlessly together. Tests the integration between the loan application module and the approval module. Verifies data flow and consistency between customer details, loan information, and EMI calculations.

#### **System Testing**

Verifies the system as a whole to ensure it meets the specified requirements. Tests the entire loan application process from submission to approval, including customer tracking. Verifies system response to simultaneous user interactions and stress testing for peak load scenarios.

### **Validation Testing**

Ensure that the developed system satisfies the specified requirements and meets the customer's needs. Validate that customers can successfully apply for loans online and track their status. Confirm that the system provides accurate loan details, EMI information, and interest rates.

## **8.2 Implementation**

Implementation is the stage of a project, when theoretical design is turned into a working system. The most crucial stage is achieving a successful system and confidence that the new system will work effectively. It involves careful planning, investigation of the manual system and a new system. Implementation means converting a new or revised system design into an operational one. The implementation includes all those activities that take place to convert from the old system to the new one.

There are several activities involved while implementing a project:

- Careful planning.
- Investigating the current system and its constraints on implementation.
- Design of methods to achieve the changeover.
- Training of the staff in the changeover procedure and evaluation of change over method

# **SYSTEM MAINTENANCE**

## 9. SYSTEM MAINTENANCE

Maintenance is making adaptation of the software for external changes (requirements changes or enhancements) and internal changes (fixing bugs). When changes are made during the maintenance phase all preceding steps of the model must be revisited.

There are three types of maintenance:

- Corrective (Fixing bugs/errors)
- Adaptive (Updates due to environment changes)
- Perfective (Enhancements, requirements changes)

Maintenance is an enigma of the system development. The definition of the software maintenance can be given describing four activities that are undertaken after the program is released for use. The maintenance activity occurs since it is unreasonable to assume that software testing will uncover all in a large system. The second activity that contributes to the definition of maintenance occurs since rapid changes are encountered in every aspect of computing. The third activity involves recommendation for new capabilities, modification to the existing functions and general enhancements when the software is used. The fourth maintenance activity occurs when software is changed to improve future maintainability or reliability.

# **FUTURE ENHANCEMENT**

## **10. FUTURE ENHANCEMENT**

The "CALICUT CO-OPERATIVE BANK-LOAN MANAGEMENT SYSTEM" (CCB-LM) could benefit from incorporating advanced artificial intelligence (AI) and machine learning (ML) capabilities in future. Implement a robust notification system to keep customers informed about the status of their loan applications, EMI payments, and any important updates. Additionally, the integration of chatbots powered by natural language processing (NLP) could enhance customer interactions, providing real-time assistance and guidance throughout the loan application journey. Allow customers to digitally sign documents, making the loan application and approval process entirely paperless.

Moreover, the implementation of blockchain technology could further bolster security and transparency, ensuring the integrity of transactions and providing a decentralized and tamper-resistant ledger. These enhancements would not only streamline the loan management process but also position the system at the forefront of technological innovation, offering a more efficient, secure, and customer-centric banking experience.

# CONCLUSION

## 11. CONCLUSION

The 'CALICUT CO-OPERATIVE BANK-LOAN MANAGEMENT SYSTEM' (CCB-LM) has been designed with the primary goal of streamlining the loan management process within the bank. By enabling customers to apply for loans online and track their status, the system aims to simplify and expedite what is traditionally a complex and time-consuming procedure. The paperless nature of the system reduces administrative workload, providing a convenient and reliable platform for both customers and banking staff. With features such as detailed customer information, loan specifics, EMI details, and instant access to interest rates, the system not only enhances the customer experience but also facilitates efficient decision-making and coordination among different departments within the loan agency.

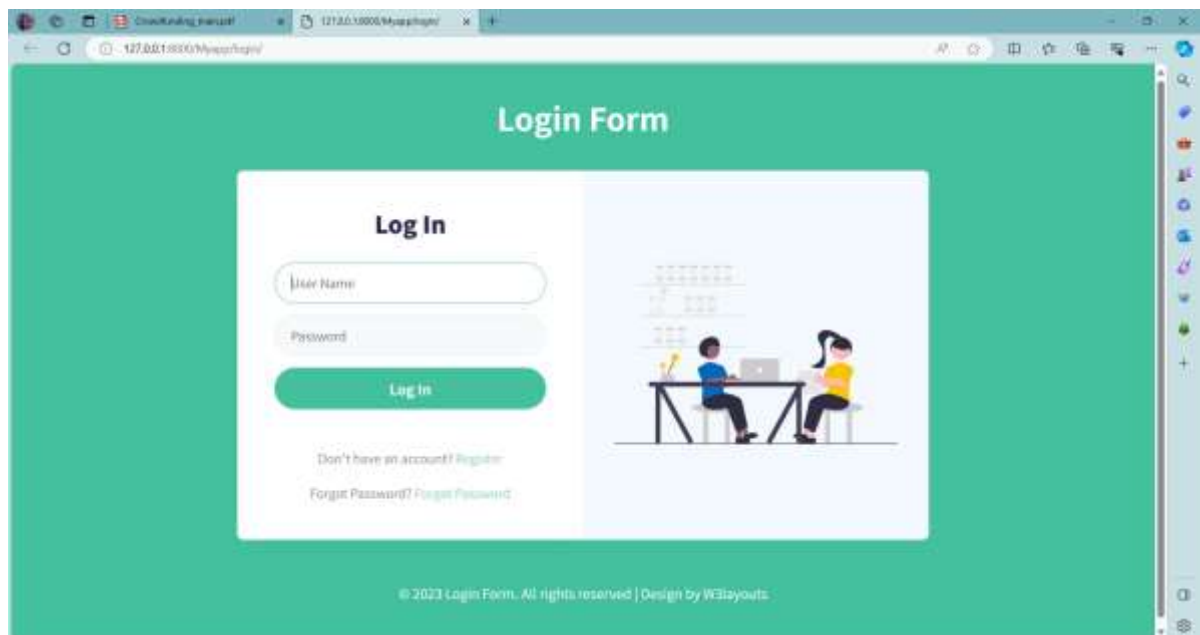
Overall, the Calicut Co-operative Bank Loan Management System stands out as a valuable tool that not only benefits individual customers but also contributes to the overall effectiveness and efficiency of the loan management process.



# APPENDIX

## 12.APPENDIX

### Admin Login



### Admin Home



## Admin Add Loan Details

Loan Name:

Loan Amount:

Interest:

EM:

Loan Duration:

Date:

Guarantee:

© 2023 Loans. All Rights Reserved | Design by W3layouts

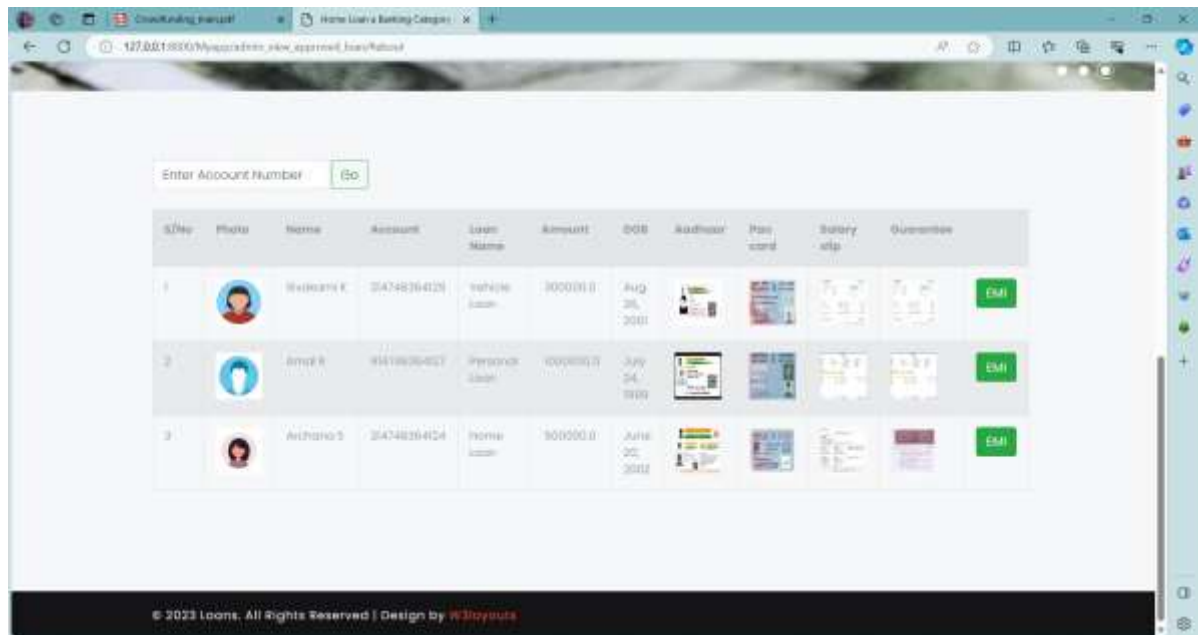
## Admin View Users



















Enter Account Number

S.No	Photo	Name	Account	ID	Gender	DOB	Email	Address	Phone		
1		Anu A	38478038432	CORP0026402	Female	Aug. 14, 2001	anu@gmail.com	Subashy Cairat Kand Kand 573012	9916300618	<input type="button" value="Approve"/>	<input type="button" value="Reject"/>
2		Vinod K	38478038432	CORP0026402	Male	July 14, 2001	vinod@gmail.com	Kingsommba Cairat Kand Kand 573012	9916300618	<input type="button" value="Approve"/>	<input type="button" value="Reject"/>
3		Kiran D	38478038432	CORP0026402	Male	Oct. 14, 2000	kiran@gmail.com	KandKand Cairat Kand Kand 573012	9916300618	<input type="button" value="Approve"/>	<input type="button" value="Reject"/>

© 2023 Loans. All Rights Reserved | Design by W3layouts

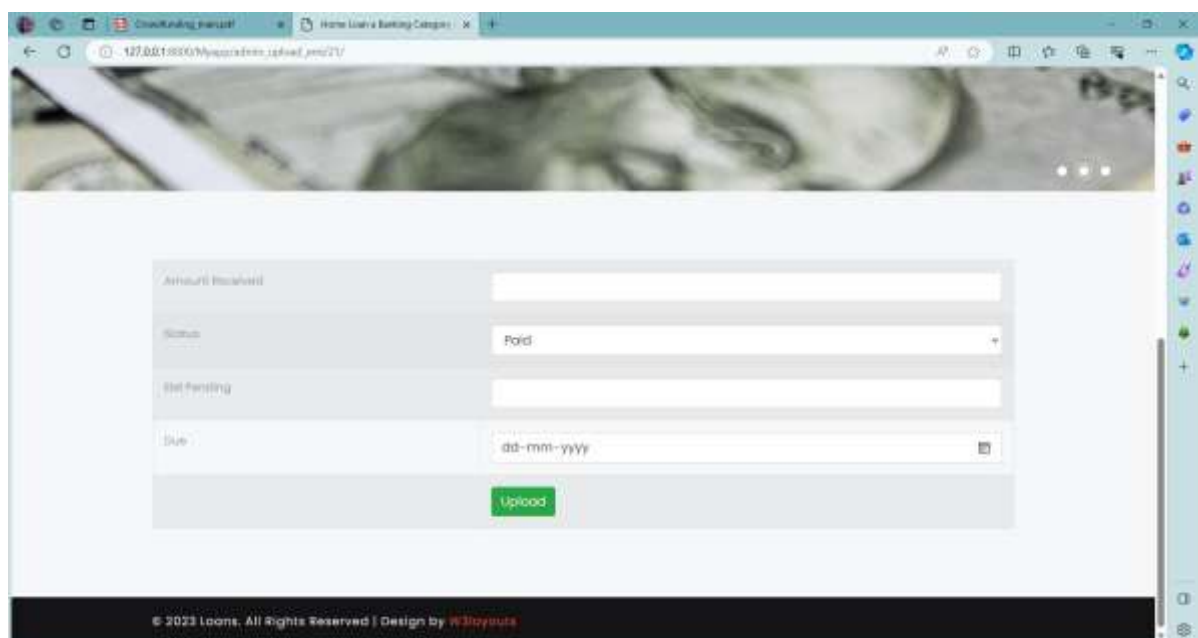
## Admin View Loan Application



S.No	Photo	Name	Account	Loan Name	Amount	DOB	Address	Pass word	Salary slip	Guarantee	EMI
1		Wishwanth K	25A748364125	Vehicle Loan	300000.0	Aug 26, 2001					
2		Amey B	43418636457	Personal Loan	100000.0	July 24, 1999					
3		Anitha K S	25A748364124	Home Loan	500000.0	June 22, 2002					

© 2023 Loans. All Rights Reserved | Design by W3layouts

## Admin Update EMI History



Amount Received:

Status:

EMI Pending:

Due:

© 2023 Loans. All Rights Reserved | Design by W3layouts

## Admin View Feedback

From: dd-mm-yyyy To: dd-mm-yyyy

S/No	Photo	Name	Email	Amount	Date	Feedback
1		Subash K	subashk@rediffmail.com	200000	Dec. 5, 2023	Very Useful website!
2		Anu	anu@gmail.com	50000	Dec. 5, 2023	I can't reply here

© 2023 Loans. All Rights Reserved | Design by W3layouts

## User Registration

CALICUT CO-OPERATIVE BANK-LOAN MANAGEMENT SYSTEM

### REGISTRATION FORM

First Name

Last Name

Age

Gender ☐ Female ☐ Male ☐ Others

DOB

Email

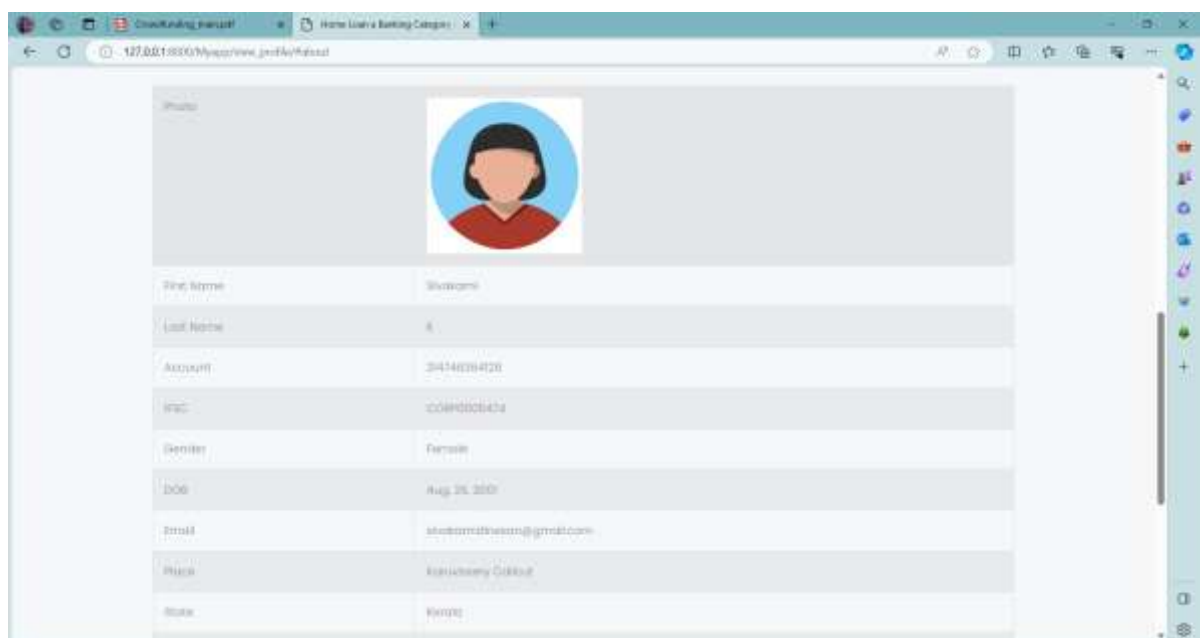
Place

State

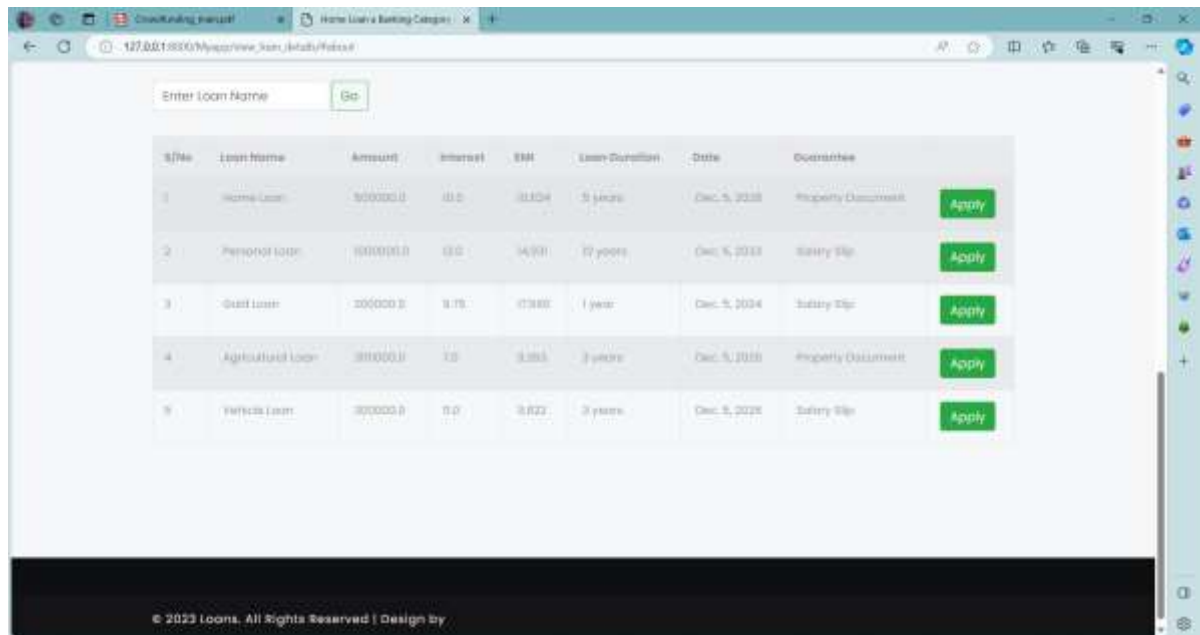
## User Home



## User Manage Profile



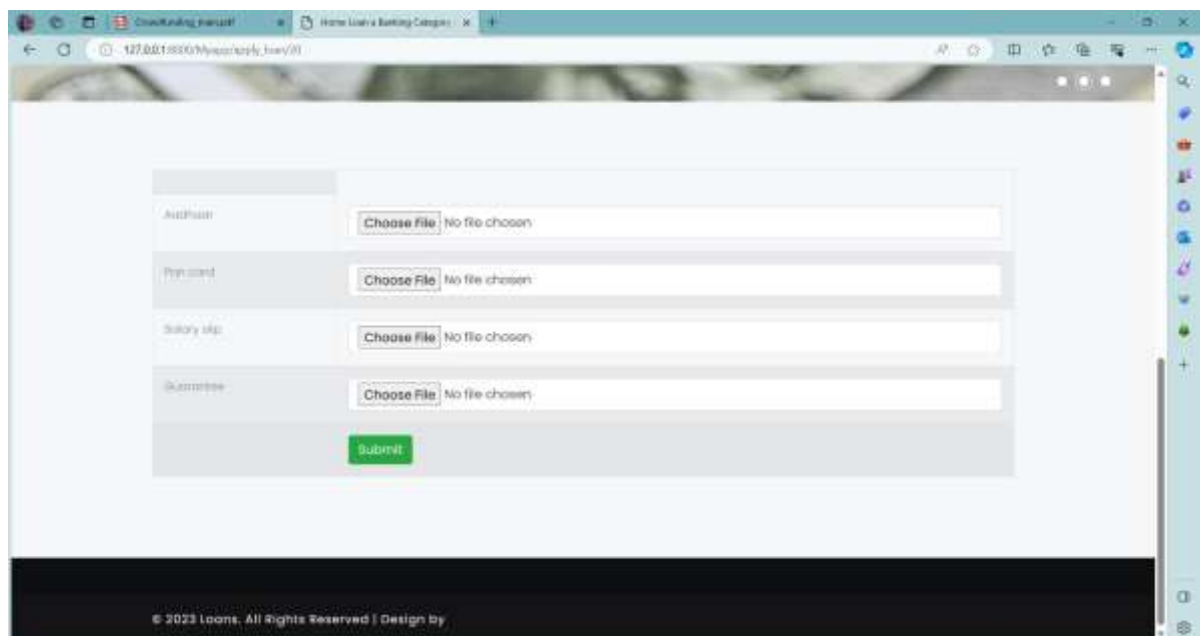
## User View Loan Details



S.No	Loan Name	Amount	Interest	EMI	Loan Duration	Date	Guarantee	
1	Home Loan	500000.0	0.0	3124	5 years	Dec. 5, 2023	Property Document	<a href="#">Apply</a>
2	Personal Loan	100000.0	0.0	1490	12 years	Dec. 5, 2023	Salary Slip	<a href="#">Apply</a>
3	Gold Loan	200000.0	0.75	1780	1 year	Dec. 5, 2024	Salary Slip	<a href="#">Apply</a>
4	Agricultural Loan	300000.0	0.0	3283	3 years	Dec. 5, 2023	Property Document	<a href="#">Apply</a>
5	Vehicle Loan	300000.0	0.0	3822	3 years	Dec. 5, 2023	Salary Slip	<a href="#">Apply</a>

© 2023 Loans. All Rights Reserved | Design by

## User Apply Loan



Aadhaar	<a href="#">Choose File</a>	No file chosen
Pan card	<a href="#">Choose File</a>	No file chosen
Salary slip	<a href="#">Choose File</a>	No file chosen
Guarantee	<a href="#">Choose File</a>	No file chosen

[Submit](#)

© 2023 Loans. All Rights Reserved | Design by



## User View Loan Status

Enter Loan Name

S/No	Photo	Name	Account	Loan Name	Amount	DOB	Address	Pan Card	Salary slip	Guaranties	Status
1		Sivakami K	204748364129	Vehicle Loan	300000.0	Aug. 26, 2000					Approved

© 2023 Loans. All Rights Reserved | Design by

## User View EMI History

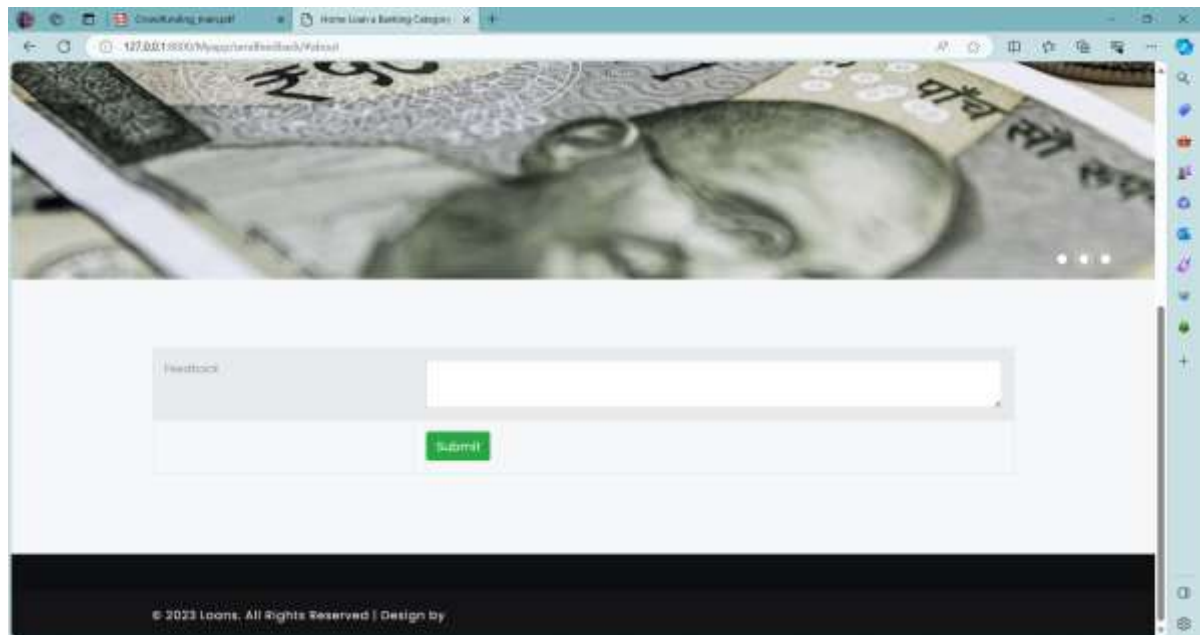
Enter the Status:

S/No	Loan Name	Amount Received	Status	EMI Pending	Due
1	Vehicle Loan	300000.0	Pending	6625	Jan. 5, 2024

© 2023 Loans. All Rights Reserved | Design by



## User Add Feedback



Feedback

Submit

© 2023 Loans. All Rights Reserved | Design by

# BIBLIOGRAPHY

## 13.BIBLIOGRAPHY

### WEBSITES

- [1] [https://www.w3schools.com/bootstrap/bootstrap\\_tables.asp](https://www.w3schools.com/bootstrap/bootstrap_tables.asp)
- [2] <https://getbootstrap.com/docs/4.0/components/buttons/>
- [3] [https://www.w3schools.com/python/python\\_intro.asp](https://www.w3schools.com/python/python_intro.asp)
- [4] [https://www.w3schools.com/sql/sql\\_ref\\_mysql.asp](https://www.w3schools.com/sql/sql_ref_mysql.asp)
- [5] <https://www.jetbrains.com/help/pycharm/getting-started.html>

### BOOKS

- [1] *Think Python: An Introduction to Software Design* - Allen B. Downey.
- [2] *Learning SQL* by Alan Beaulieu.
- [3] *Flask Web Development: Developing Web Applications with python*
- [4] *Head First Html and CSS* by Elizabeth Robson
- [5] *Code complete: a Practical Handbook of Software Construction* by Steve McConnell