Loan Prediction and Processing System

Team: Chutney
Manan Khasgiwale and Tavishi Priyam

Project Summary

Our project aims to integrate a machine learning predictive analysis with an online loan dispersal system. The system runs as a web app that includes a login page for the users. If the user is not already registered, it will prompt the user to register before they login.
After registering, users can fill out a loan prediction form with details such as applicant income, dependents, credit history, etc. The system will predict loan approval/rejection. If the loan is approved, the system will also predict an amount. After approval from the bank manager, the system will offer the users two different payment plans: monthly or quarterly. Once the payment plan has been finalised, the loan will be sanctioned and the amount will be disbursed. The web app will also send regular reminders to the users regarding payments and support the collection of these payments. Once the loan has been paid in full, the bank manager will approve the completion of payment and the client object is removed.

Project Requirements

The requirements and responsibilities of this system are:

- Register new user
- Get loan details from user
- Predict approval/rejection of loan
- Predict loan amount
- Contact Bank Manager for approval
- Sanction loan
- Provide user with payment plan options
- Disburse loan amount
- Collect payment
- Send out reminders
- Show balance

Main aim of the project is to get user details and predict the loan amount. It also takes care of loan disbursement and loan reminders/collection.

Users and Tasks: Use Cases

Keeping the WAVE rule in mind, we have decided on three actors for our system: client, bank system and bank manager. There are a total of eight use cases. Some of these are extending the main use cases and some are required.

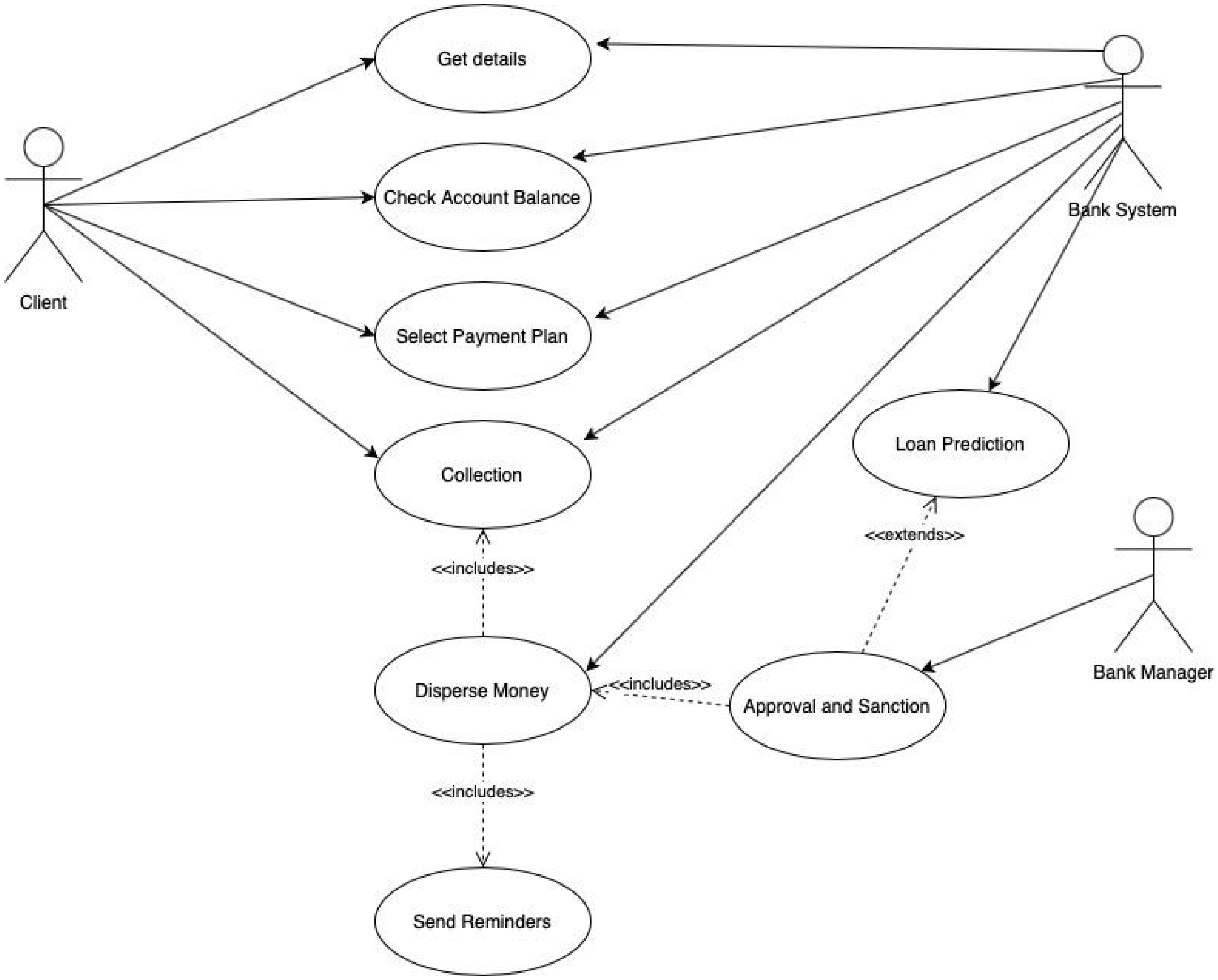


Figure 1: Use Case Diagram

UML Activity Diagram

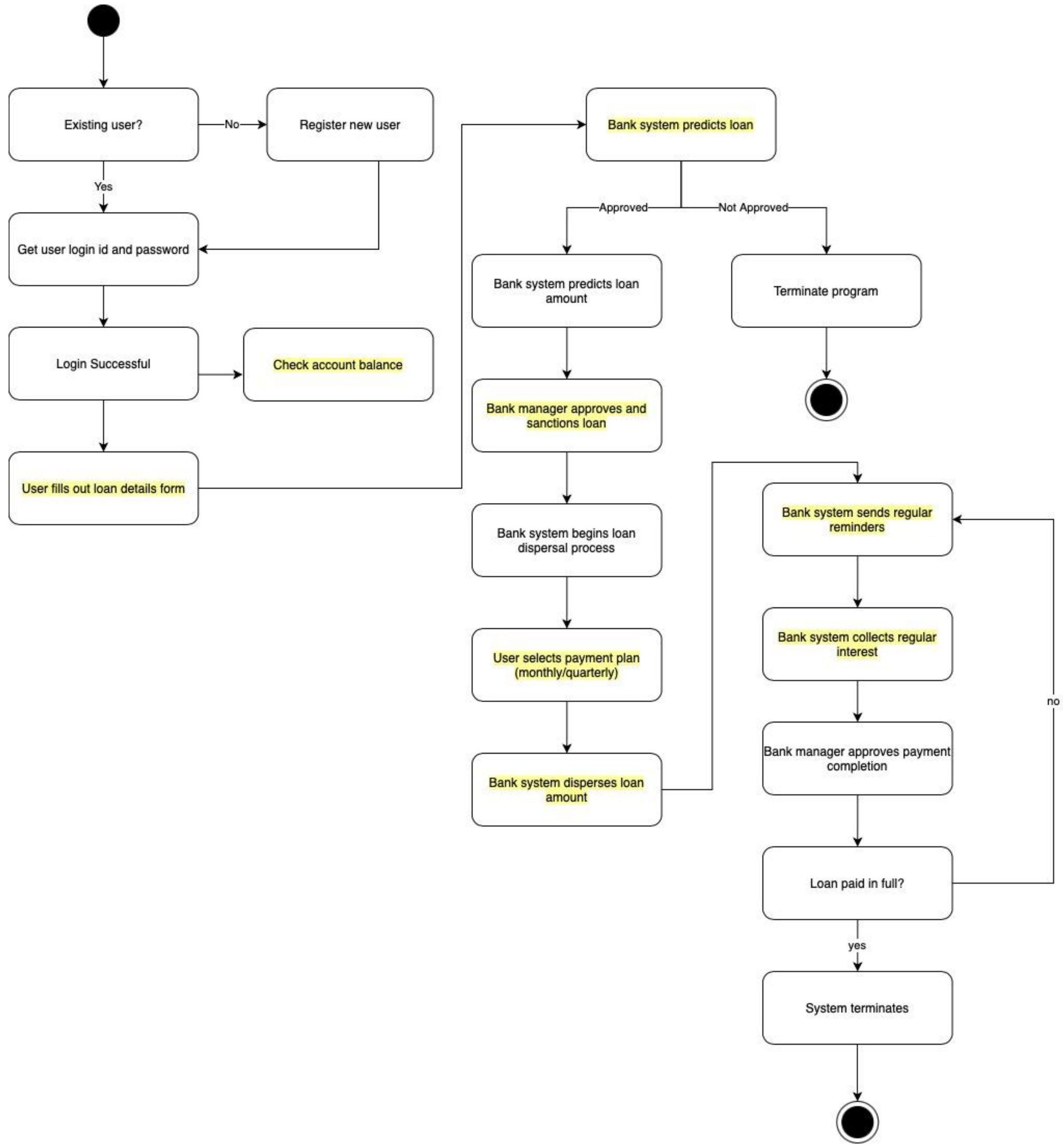

Existing user?
No
Register new user
Yes
Get user login id and password
Login Successful
Check account balance
User fills out loan details form
Bank system predicts loan
Approved
Not Approved
Bank system predicts loan amount
Terminate program
Bank manager approves and sanctions loan
Bank system begins loan dispersal process
User selects payment plan (monthly/quarterly)
Bank system disperses loan amount
Bank system sends regular reminders
Bank system collects regular interest
Bank manager approves payment completion
Loan paid in full?
no
yes
System terminates

Figure 2: UML Activity Diagram

The elements highlighted in yellow in the activity diagram are the use cases being serviced by the bank system, bank manager and the user.

Architecture Diagram

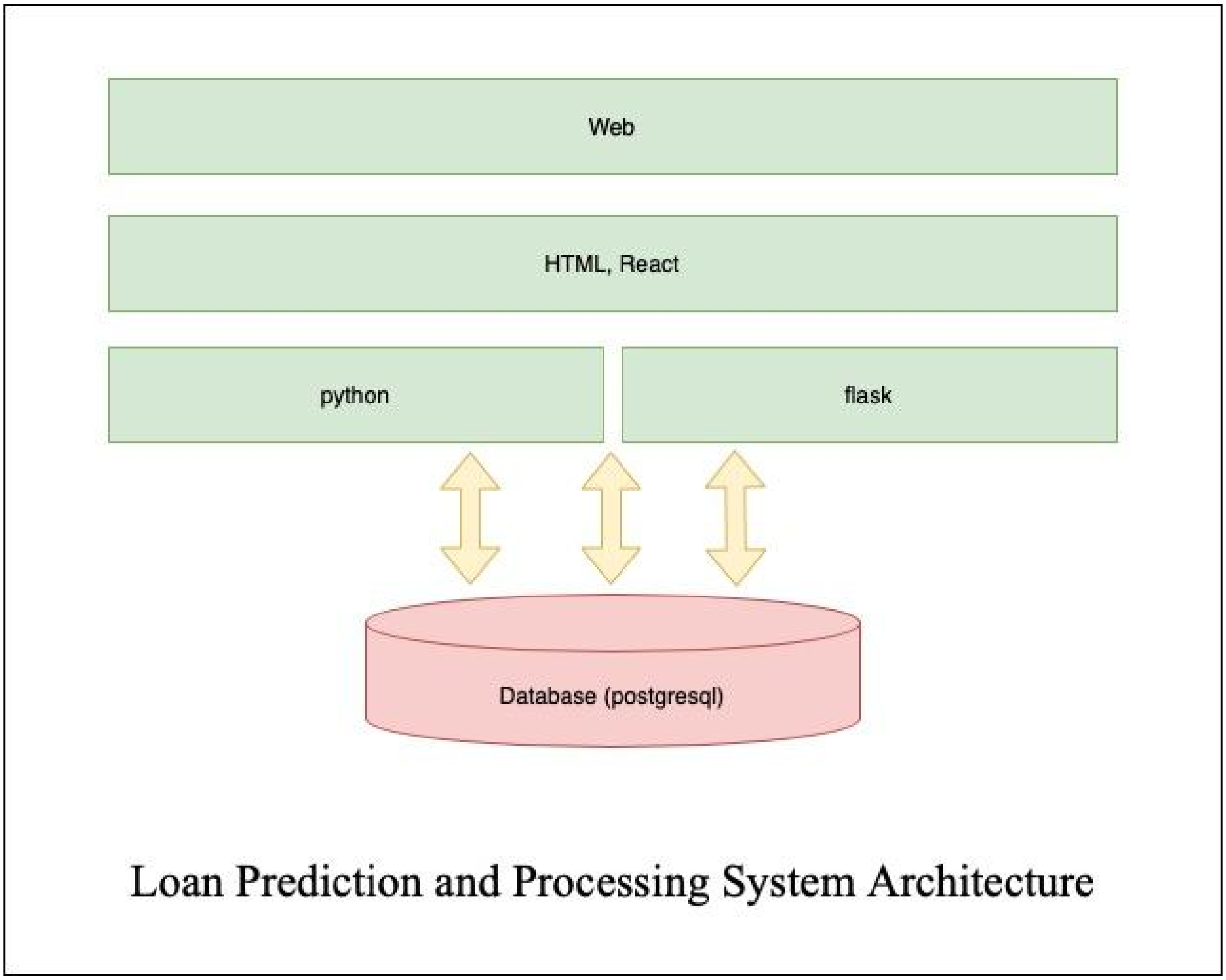


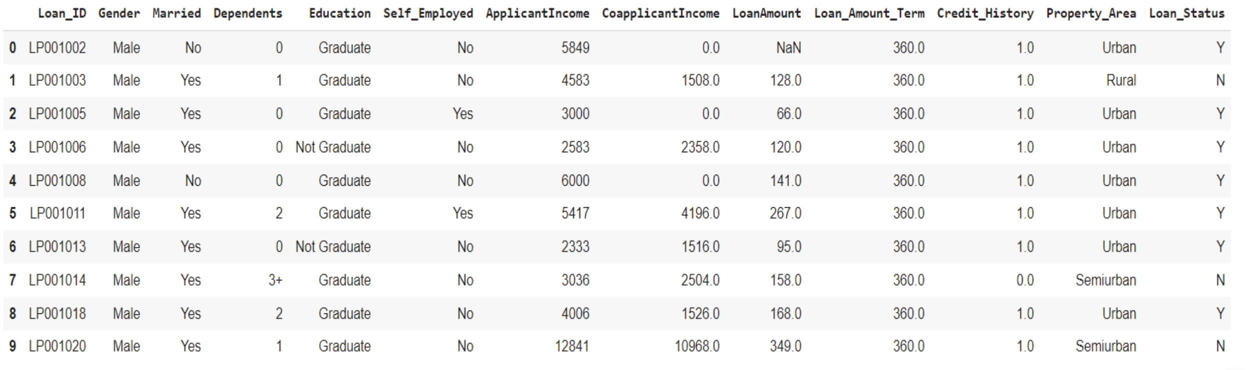
Figure 3: Architecture Diagram

In this project, we are developing a web application that serves desktop and mobile users. We are making use of React in combination with Python and flask to develop this web app. React will use flask as the backend to get the data from python code. To design the frontend, we are using React along with bootstrap and CSS.

Data Storage

For data storage, we’ll be deploying a **postgresql** server locally/cloud. We’ll have two main tables to store client related details, one for storing client information, balance, etc. and another for storing loan disbursement information.

Raw data for loan prediction algorithm:



	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
0	LP001002	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0	Urban	Y
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0	Rural	N
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0	Urban	Y
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0	Urban	Y
4	LP001008	Male	No	0	Graduate	No	6000	0.0	141.0	360.0	1.0	Urban	Y
5	LP001011	Male	Yes	2	Graduate	Yes	5417	4196.0	267.0	360.0	1.0	Urban	Y
6	LP001013	Male	Yes	0	Not Graduate	No	2333	1516.0	95.0	360.0	1.0	Urban	Y
7	LP001014	Male	Yes	3+	Graduate	No	3036	2504.0	158.0	360.0	0.0	Semiurban	N
8	LP001018	Male	Yes	2	Graduate	No	4006	1526.0	168.0	360.0	1.0	Urban	Y
9	LP001020	Male	Yes	1	Graduate	No	12841	10968.0	349.0	360.0	1.0	Semiurban	N

Figure 4: Loan Applicant Details

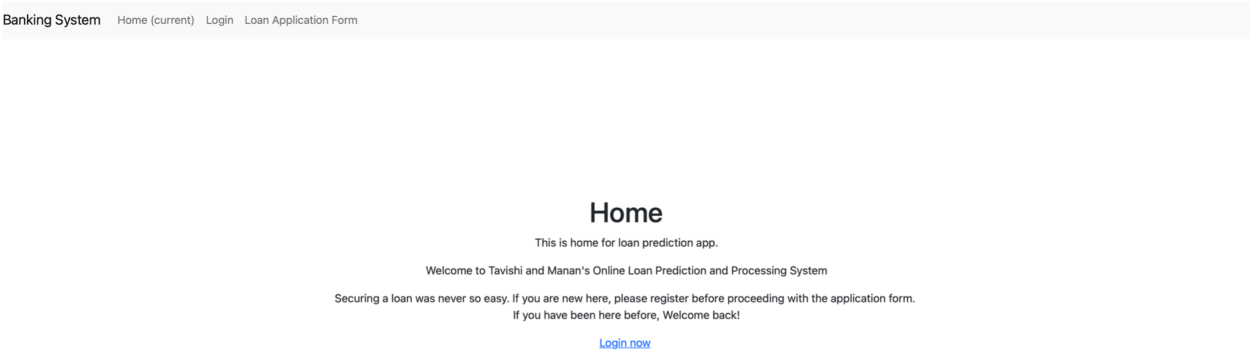
The table contains 13 columns, each describing a detail about the loan applicant. Each column acts as a variable that helps in predicting the loan for a client. Features such as graduate level education, high applicant and co-applicant income, low loan amount and positive credit history all contribute positively towards the loan prediction. The dataset being used to train the machine learning model has been taken from an online contest.

Entities in table:

- Client details: client id(primary key), name, date of birth
- Loan form details: loan id(primary key), client id(foriegn key), amount, balance left, payment duration

UI Mockup/Sketch

Home page: Illustrated below is a screenshot of the home page for our web application. This is a basic mockup of what the users will see when they run the application. Further changes will be made to improve the UI and design of the page.



Home

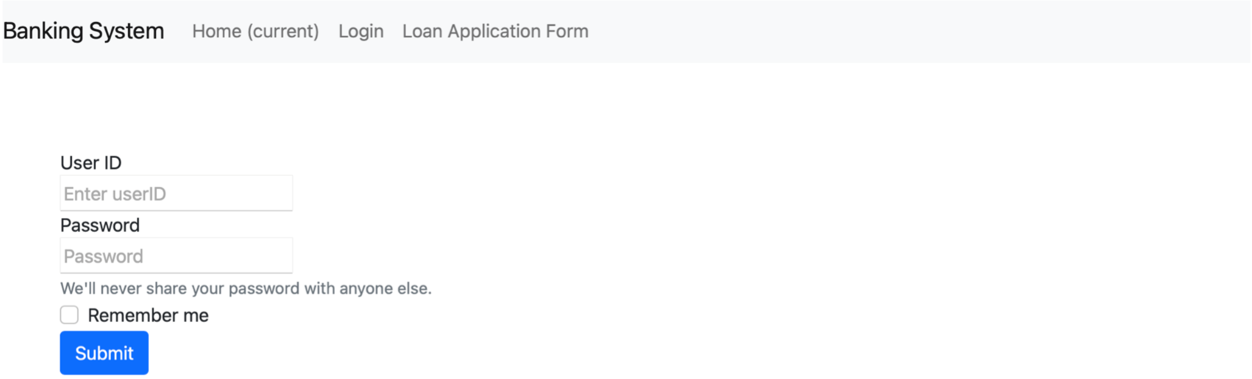
This is home for loan prediction app.

Welcome to Tavishi and Manan's Online Loan Prediction and Processing System

Securing a loan was never so easy. If you are new here, please register before proceeding with the application form.
If you have been here before, Welcome back!

Figure 5: Home page

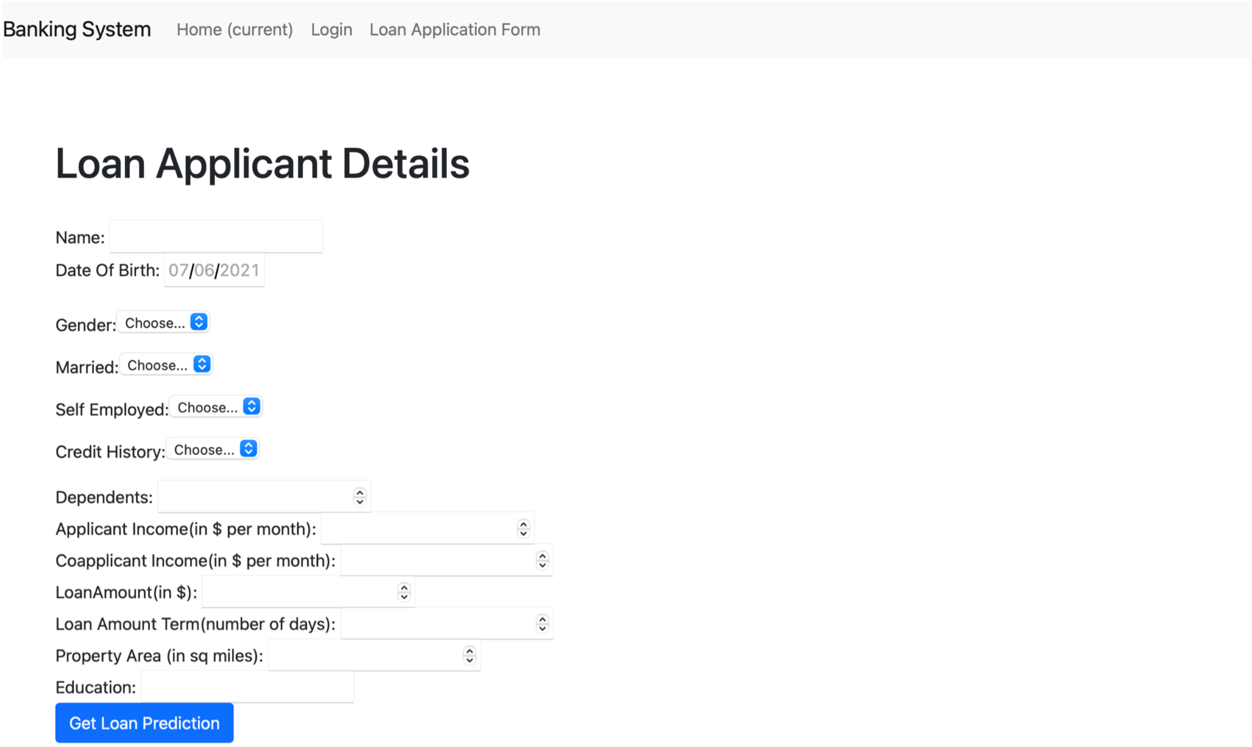
Login page: The home page redirects users to the login page where they can enter their userID and password to login. It will also prompt the user to register as new if the userID does not exist.



We'll never share your password with anyone else.

Figure 6: Login page

Loan Application Form: The loan application form can be accessed by the users by clicking on the corresponding button on the navigation bar at the top of the screen. The form asks for basic details of the applicant along with required information such as income, number of dependents, etc.



Loan Applicant Details

Name:

Date Of Birth: 07/06/2021

Gender: Choose...

Married: Choose...

Self Employed: Choose...

Credit History: Choose...

Dependents:

Applicant Income(in $ per month):

Coapplicant Income(in $ per month):

LoanAmount(in $):

Loan Amount Term(number of days):

Property Area (in sq miles):

Education:

Figure 7: Loan application form

All the pages displayed above are prototypes for the actual application. We plan on incorporating different design features in the UI.

UML Sequence Diagram

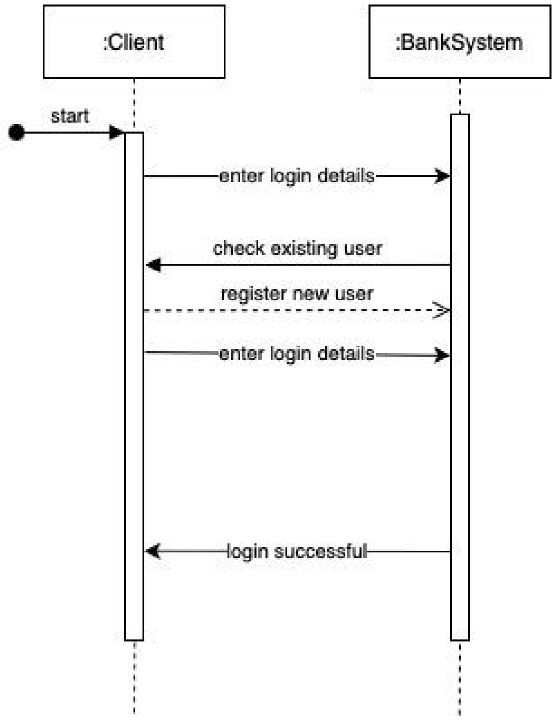


Figure 8: Login page interaction

The login page is essential to our web application since users cannot proceed to the loan prediction/processing without logging into the application.

The sequence diagram above depicts the interaction between the system and the user when the user enters login details. If they enter incorrect information, they will be redirected to the same page. If the user has not been registered yet, they will have the option to register themselves before they can proceed with the application. The login interaction is limited to the Client and the BankSystem. The BankManager does not play any role in this interaction.

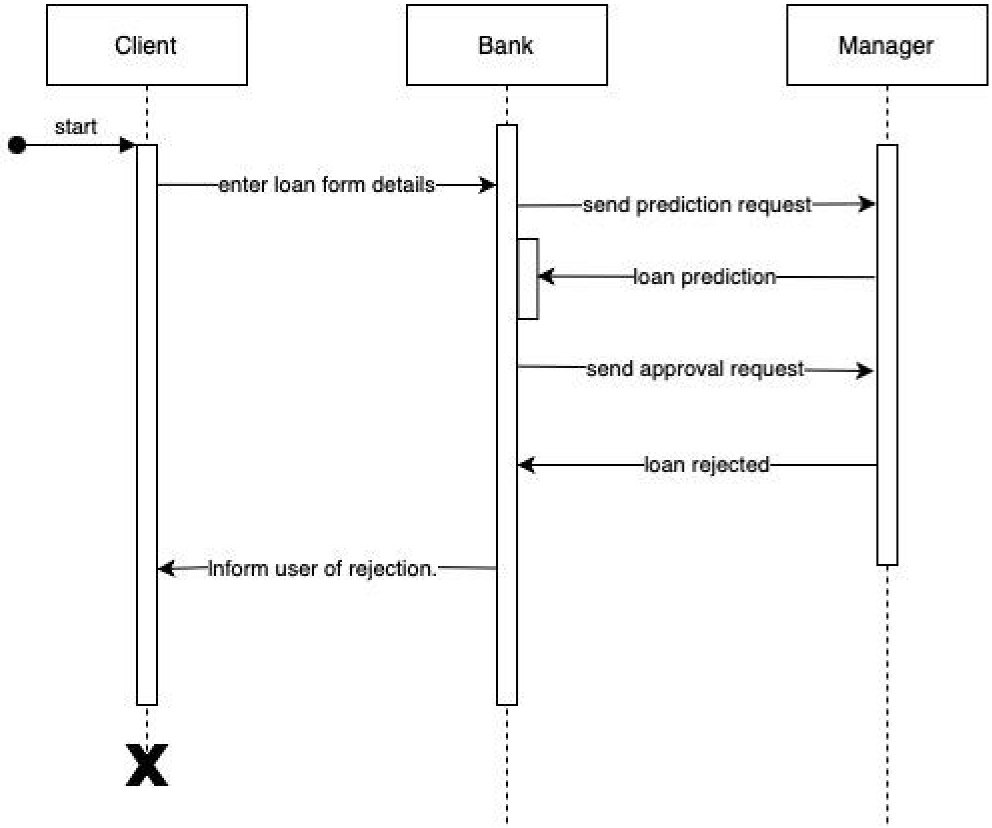


Figure 9: Loan rejection interaction

The loan applicant fills out the loan application form to get a prediction from the system. This sequence diagram depicts the interaction between the Client, BankSystem and the BankManager. The BankManager commands the BankSystem to run a prediction once the user has submitted the loan application form. Once the prediction is made, the BankManager either approves or rejects the sanction of the loan. If the loan is rejected, the client is informed of the rejection and the client object terminates..

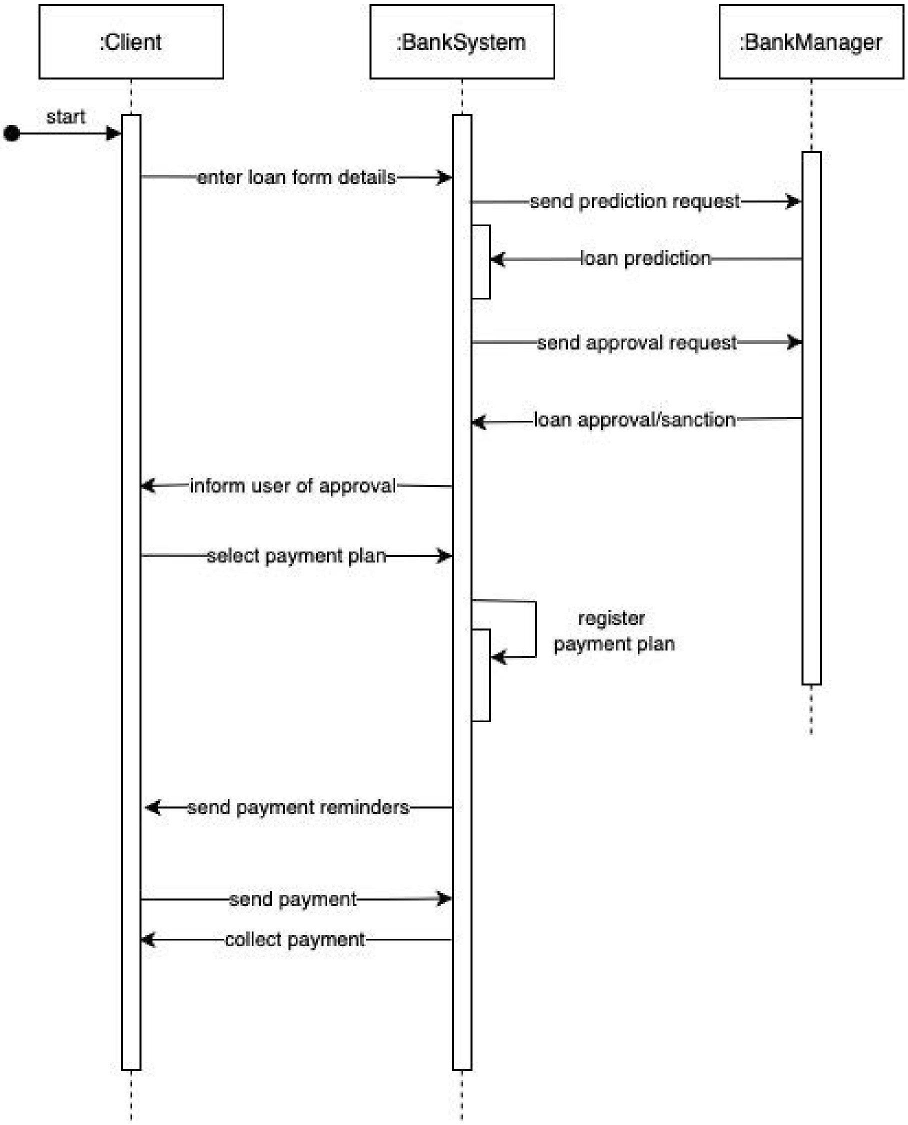


Figure 10: Loan approval interaction

The loan applicant fills out the loan application form to get a prediction from the system. This sequence diagram depicts the interaction between the Client, BankSystem and the BankManager. The BankManager commands the BankSystem to run a prediction once the user has submitted the loan application form. Once the prediction is made, the BankManager either approves or rejects the sanction of the loan. If the loan is approved, the user is informed of the approval and asked to select a payment plan. The selected payment plan is registered with the bank system and regular payments are collected. The BankSystem also sends out regular reminders to the user regarding the status of the loan and pending payments. Once the loan has been paid in full, the BankManager confirms it before the client object is terminated.

UML Class Diagram

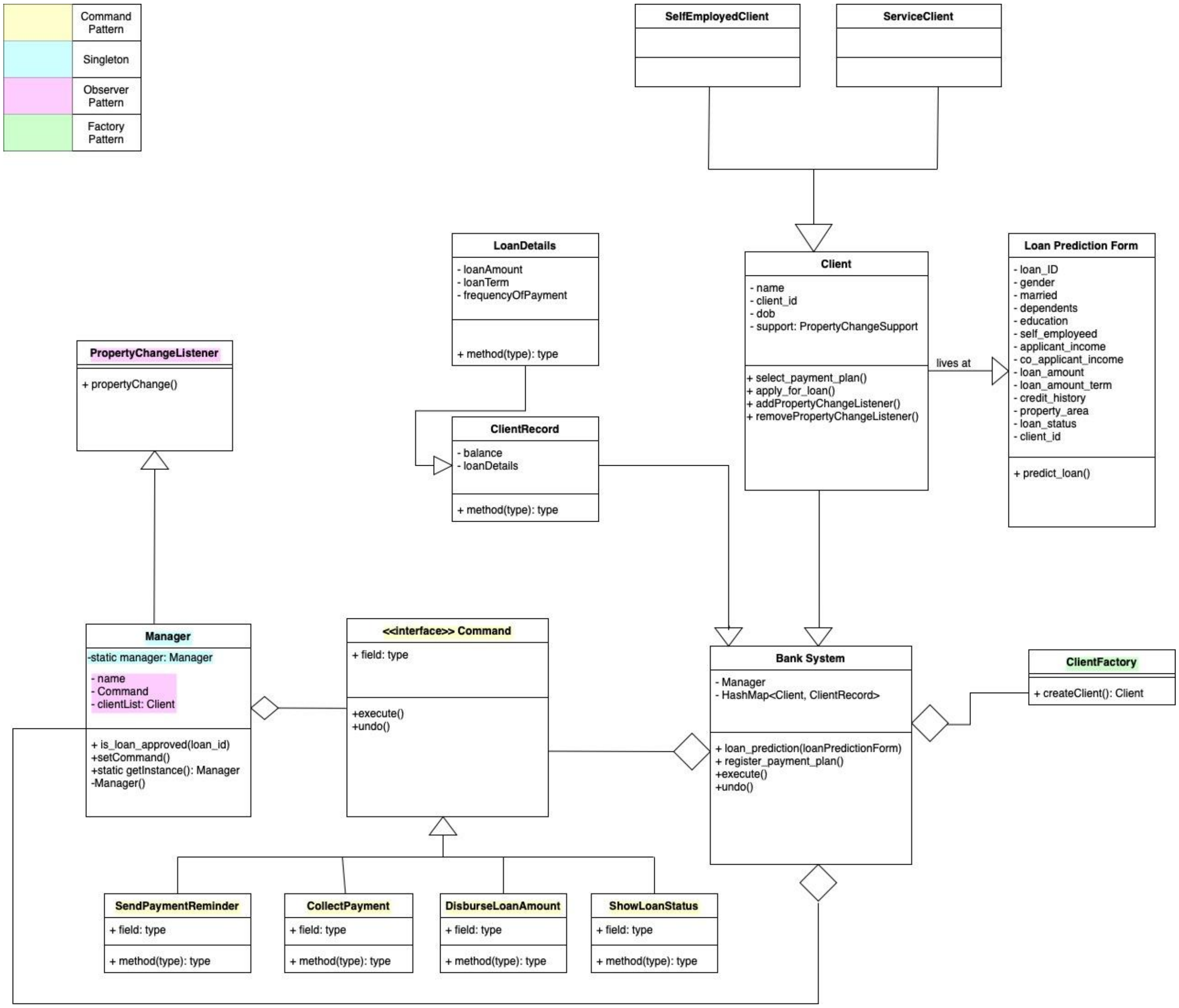


Figure 11: UML class diagram

Illustrated above is the class diagram for our web application. We are implementing four different design patterns in the initial prototype and plan on incorporating more as we proceed with the development process.

The code for the UI and basic python backend can be found on:
https://github.com/thatgeekyperson/OOADProjectSU21/tree/main/Project5.