Smart Lender Applicant Credibility Prediction for Loan Approval

Project Report

Team Members

Haritha T

Jayavarthini P

Nivaethetha P

Ragavi S

Table of Contents

1. **INTRODUCTION**
    a. Project Overview
    b. Purpose
2. **LITERATURE SURVEY**
    a. References
    b. Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
    a. Empathy Map Canvas
    b. Ideation & Brainstorming
    c. Proposed Solution
    d. Problem Solution fit
4. **REQUIREMENT ANALYSIS**
    a. Functional requirement
    b. Non-Functional requirements
5. **PROJECT DESIGN**
    a. Data Flow Diagrams
    b. Solution & Technical Architecture
    c. User Stories
6. **PROJECT PLANNING & SCHEDULING**
    a. Sprint Planning & Estimation
    b. Sprint Delivery Schedule
    c. Reports from JIRA
7. **CODING & SOLUTIONING**
    a. Feature 1
    b. Feature 2

    Database Schema (if Applicable)

8. **TESTING**
    a. Test Cases
    b. User Acceptance Testing
9. **RESULTS**
    a. Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**
    a. Source Code
    b. GitHub & Project Demo Link

Chapter 1: INTRODUCTION

1.1 Project Overview

The credit system governed by the banks is one of the most important factors which affect our country's economy and financial condition. Also, credit risk is one of the main functions of the banking community. The prediction of credit defaulters is one of the difficult tasks for any bank. This problem occurs when the banks need to provide loans to the customers who are in need of the money. But by forecasting the loan defaulters, the banks definitely may reduce their loss by reducing their non-profit assets.

1.2 Purpose

People who need loan and want to check whether they are eligible for loan or not

Chapter 2: LITERATURE SURVEY

2.1 References

[1] Sai Aparna Vangaveeti et al, In this paper, Logistic Regression algorithm had been used for predicting whether the loan is approved or not. They have used various input variables to get the efficient output. The output will be in form of binary (0 or 1). If the output is 0, then there is no approval. If the output is 1 means it is approved

[2] Kumar Arun et al, This paper's major goal is to forecast whether or not issuing a loan to a specific person will be secure. There are four components in this paper: (i)Data Collection (ii) Comparison of machine learning models on collected data (iii) Training of system on most promising model (iv) Testing. They have used six machine learning methods decision tree, random forest, support vector machine, neural network, adaboost,linear model

[3] Yamuna B et al, In this paper, they have used five different machine learning model to find the best fitting model. The five different models are linear regression, random forest, support vector machine, decision tree classifier, extreme gradient boost (XG Boost). Among all these machine learning algorithms XGBoost proved to be the best with accuracy 0.82

[4] Rekha A.G et al, In this paper, to analyze the relationship between financial transactions and loan defaults, two methods based on LSTM (Long Short Term Memory) and a hybrid neural network architecture are proposed. 94% accuracy is attained via bidirectional LSTMs with hybrid architecture. This work's utilization of hybrid neural network architectures offers the right path toward developing an early warning system through online loan default prediction

[5] Kumar T et al, In this paper, data from former clients of several banks who had loans accepted based on a set of criteria were used. The machine learning model is trained on the record to get reliable results. To forecast loan safety, the logistic regression approach is applied. The data is initially cleaned in order to remove any missing values from the data set. A data set of 1500 examples with 10 numerical and 8 categorical features were used to train the model. When crediting a loan to a customer,a number of factors, including the customer's CIBIL Score (Credit History), business value, and customer assets, have been considered

[6] Vaidya, This paper, discusses logistic regression and its mathematical representation. In order to implement the predictive and probabilistic approaches to a specific problem of loan approval prediction, his study uses logistic regression as a machine learning technique. This is to decide whether to approve a loan for a set of records belonging to an applicant. It also discusses a few other practical applications for the Machine Learning mode

[7] Chandan Soni et al, In this paper, a loan prediction system is presented that aids organizations in choosing whether to approve or deny loan requests from clients. The input variables such as loan ID, Marriage, loan amount, gender etc.., have been sent to train the model. The prediction is made using the

Decision Tree Algorithm

[8] Tanvir Anzum et al, In this paper, in order to predict fraudulent loan requests from

clients, they have used six machine learning algorithms such as Decision tree, Support vector machine, Random forest, K closest neighbors, Ada-Boost, and Logistic regression. The K-Nearest Neighbors algorithm provided accuracy of 83.75%, which was superior to the other five machine learning techniques [9] Jayapandian N et al, In this paper, they have used two different kind of datasets. One is used for training the model and the other dataset is used for testing the model. They used random forest, machine learning algorithm to train the model. The suggested random forest model offers a higher level of accuracy. This approach offers a level of accuracy that is 28% higher than that of conventional prediction

[10] Suresh Kumar et al, The primary goal of this paper is to decide whether or not the loan offered to a specific individual or organization should be approved. They have used two machine learning algorithms one is random

forest and another one is decision tree classifier. The parameter setting for decision tree are Min Split=20, Max Depth=30, Min Bucket=7. The parameter setting for random forest are Number of trees=450, number of variables=8

2.2 Problem Statement Definition

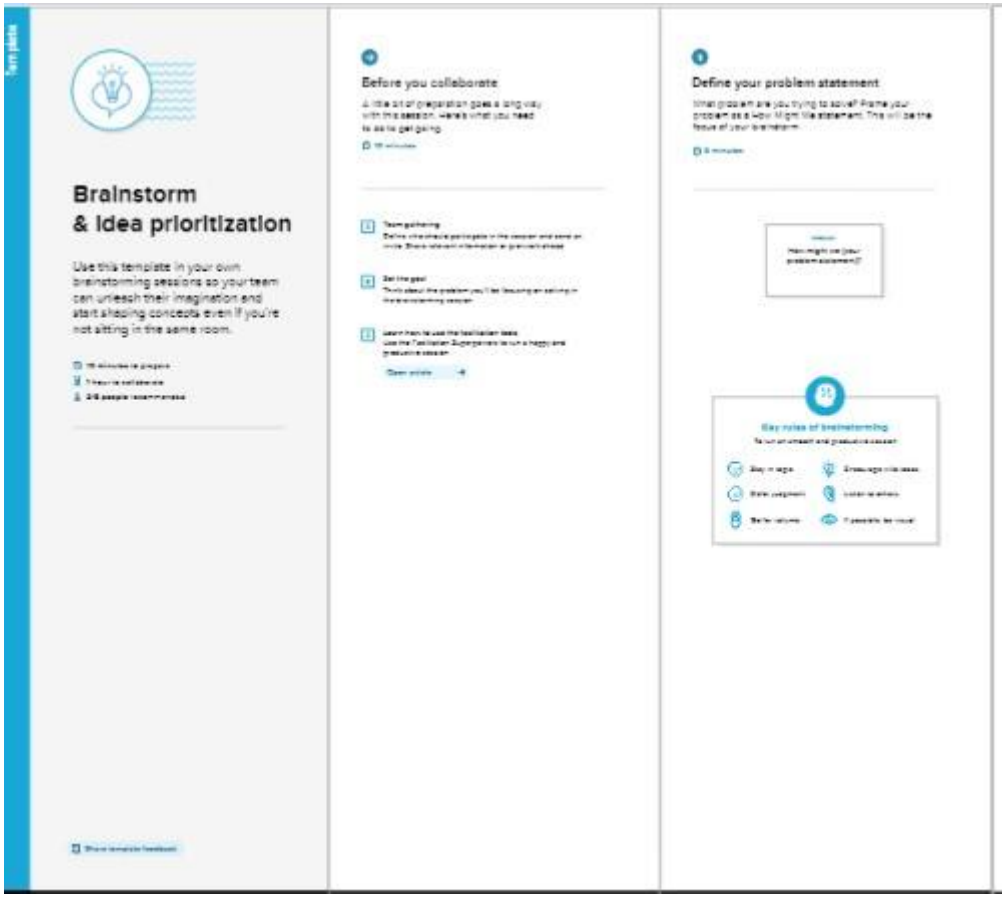
The credit system governed by the banks is one of the most important factors which affect our country's economy and financial condition. Also, credit risk is one of the main functions of the banking community. The prediction of credit defaulters is one of the difficult tasks for any bank. This problem occurs when the banks need to provide loans to the customers who are in need of the money. But by forecasting the loan defaulters, the banks definitely may reduce their loss by reducing their non-profit assets.

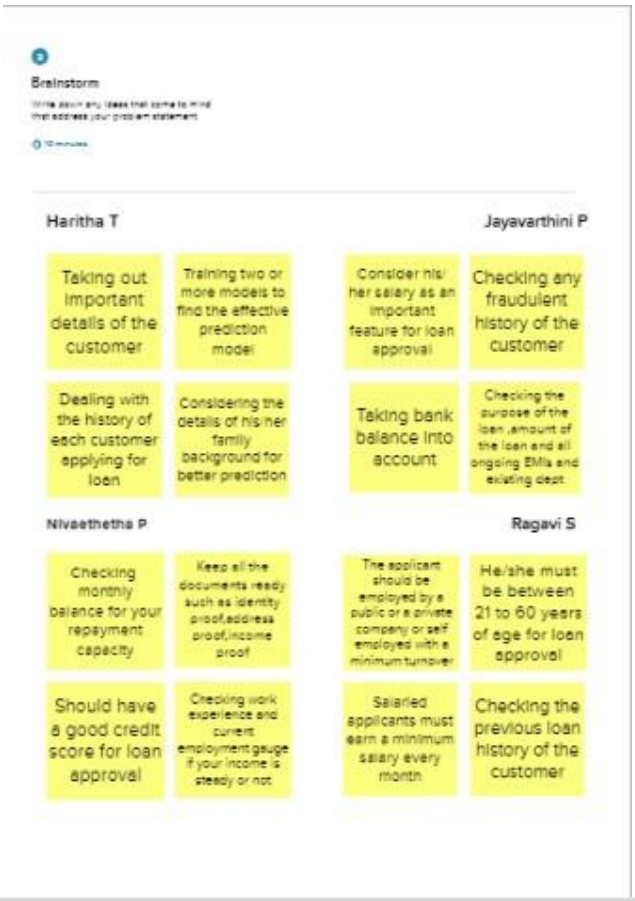
Chapter 3: IDEATION & PROPOSED SOLUTION

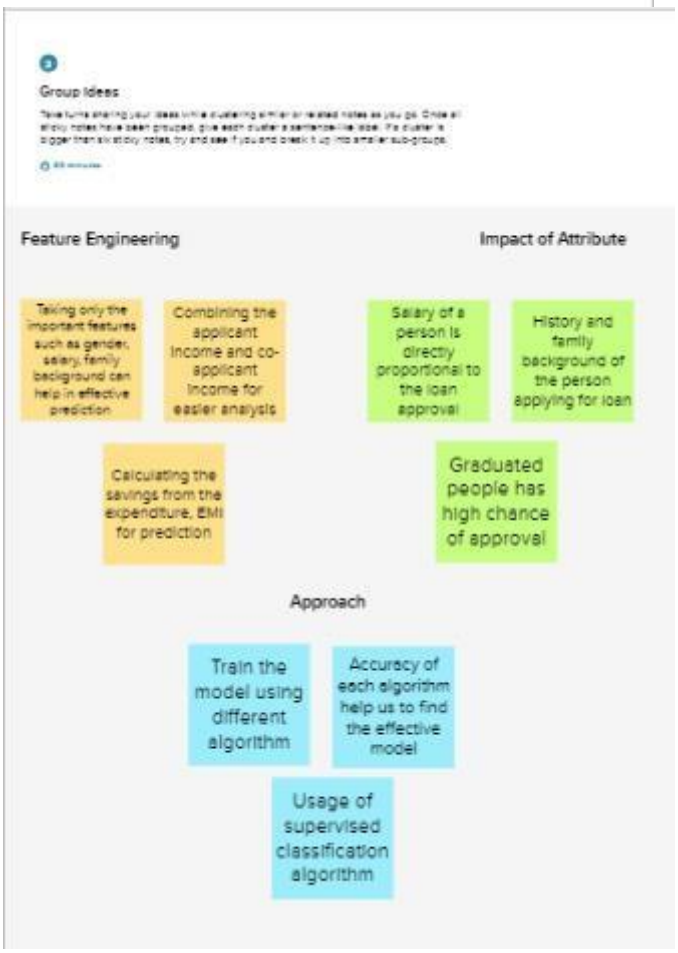
3.1Empathy Map Canvas

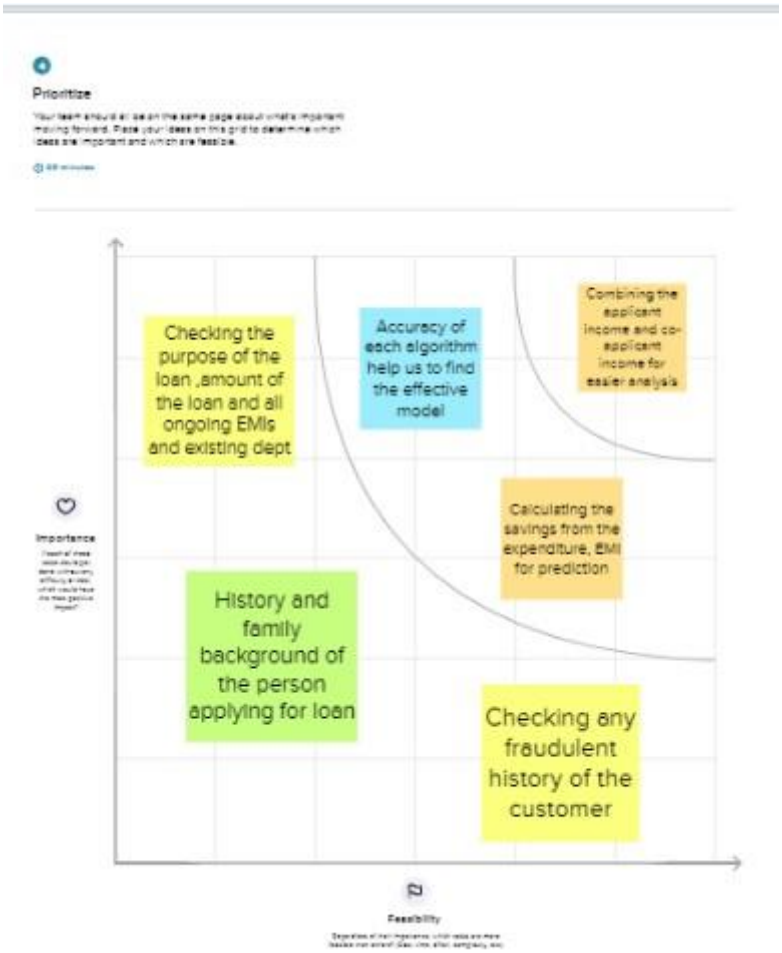
An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to help teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

3.2 Ideation & Brainstorming




Brainstorm
Haritha T
Taking out important details of the customer
Training two or more models to find the effective prediction model
Dealing with the history of each customer applying for loan
Considering the details of his/her family background for better prediction
Jayavarthini P
Consider his/her salary as an important feature for loan approval
Checking any fraudulent history of the customer
Taking bank balance into account
Checking the purpose of the loan ,amount of the loan and all ongoing EMIs and existing dept
Nivaethetha P
Checking monthly balance for your repayment capacity
Keep all the documents ready such as identity proof,address proof,income proof
Should have a good credit score for loan approval
Checking work experience and current employment gauge if your income is steady or not
Ragavi S
The applicant should be employed by a public or a private company or self employed with a minimum turnover
He/she must be between 21 to 60 years of age for loan approval
Salaried applicants must earn a minimum salary every month
Checking the previous loan history of the customer


Group Ideas
Feature Engineering
Taking only the important features such as gender, salary,family background can help in effective prediction
Combining the applicant income and co-applicant income for easier analysis
Calculating the savings from the expenditure, EMI for prediction
Impact of Attribute
Salary of a person is directly proportional to the loan approval
History and family background of the person applying for loan
Graduated people has high chance of approval
Approach
Train the model using different algorithm
Accuracy of each algorithm help us to find the effective model
Usage of supervised classification algorithm

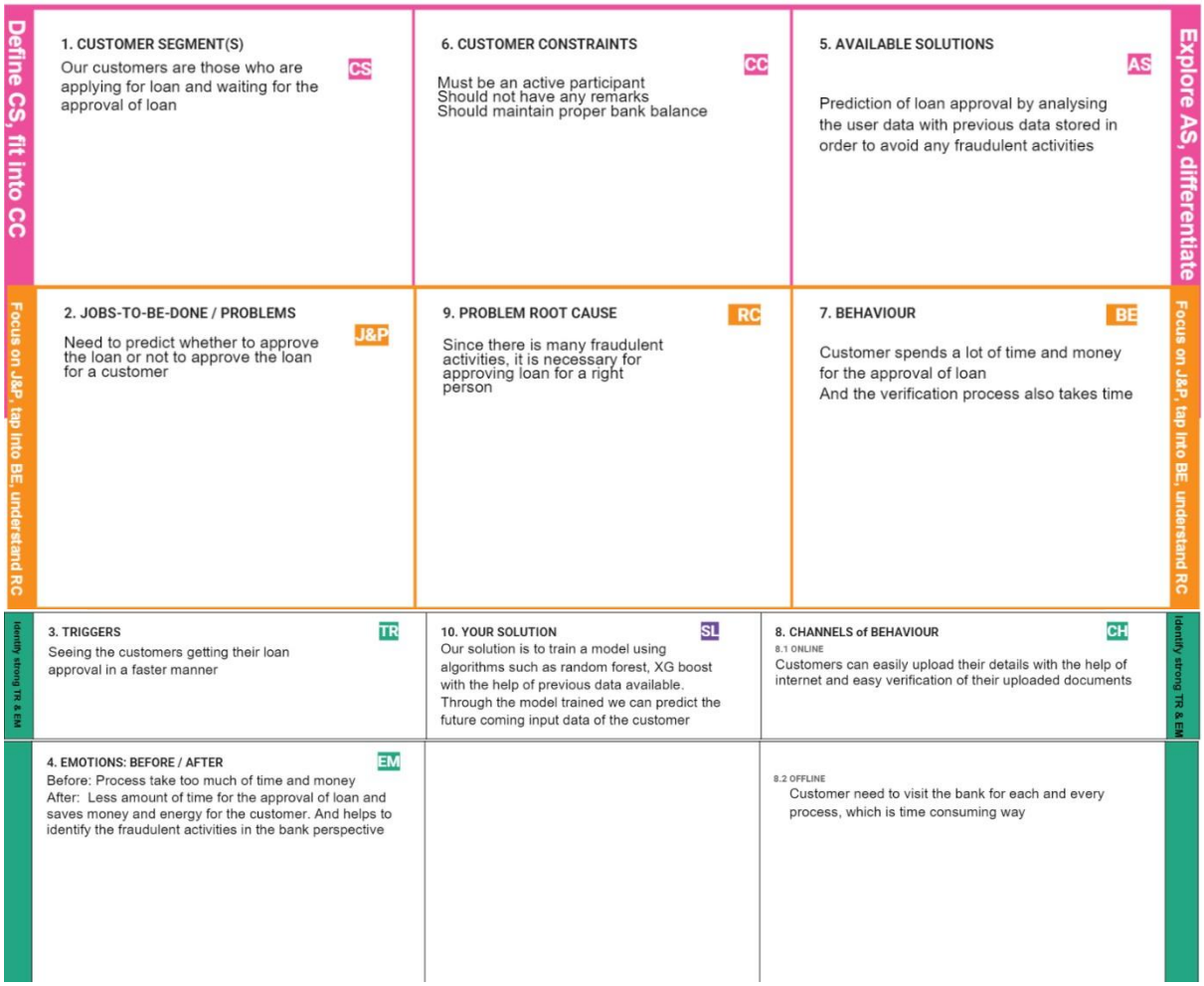


3.2 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To predict the loan approval for an applicant by considering features such as salary, gender, family background
2.	Idea / Solution description	By taking some important features such as salary, EMI, savings, employment, property area, gender, age and class label (supervised learning), we can apply a classification algorithm such as Random Forest, Decision Tree Classifier and XGBoost on the dataset to train the model. The trained model will help us in better prediction of loan approval

3.	Novelty / Uniqueness	Here we will be comparing the accuracy of all the algorithms to find the efficient model and to give a good customer experience in tracking the status of the loan
4.	Social Impact / Customer Satisfaction	The applicant can easily come to know their loan approval status. Minimal time and Manpower
5.	Business Model (Revenue Model)	The customer has to subscribe to the application if their loan amount is higher than the margin level and also if they are applying for the second time
6.	Scalability of the Solution	The applicant will be able to know about their eligibility for loan approval from home itself and can apply from anywhere. This will save time and reduce manpower for both applicant and the respective bank

3.4 Problem Solution fit



Chapter 4: REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement c)	Sub Requirement (Story / Sub-Task)
FR-1	User Requirement	To check the loan eligibility using the credit score, prediction for loan approval
FR-2	User Confirmation	Through one time verification and using captcha etc..,.
FR-3	Profile Updation	The user can update their profile when there is need to add any add-ons to it
FR-4	User Registration	The user gets login/signup using Gmail account or by using mobile number

FR-5	User Authentication	The OTP/verification code the user gets authenticated and OTP is used for mobile number registration
FR-6	Feedback Evaluation	The user provided feedbacks are used for evaluation of app performance and Updation is made over that

4.2 Non-functional Requirements:

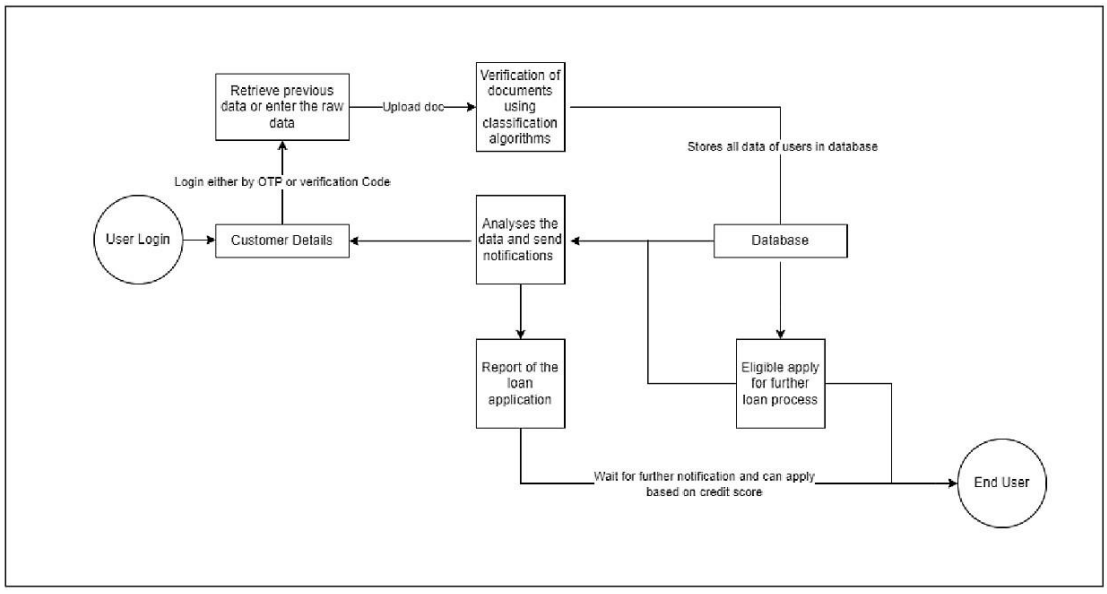
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	This application is mainly used to analyse cibil score and predict the eligibility for users to avail for loan approval by following community guidelines
NFR-2	**Security**	It uses OTP and verify code verification for each user and uses and hybrid security features over internet to safely maintain the updated documents of user
NFR-3	**Reliability**	Maintaining the app up to date for reliant future, durability and efficiency of the mobile app by releasing patch fix and software updates
NFR-4	**Performance**	It has a user-friendly interface and can check multiple persons cibil score parallels irrespective of server traffic. It stores the data collected over in an efficient database
NFR-5	**Availability**	It is platform independent and it is available where the users are able to wish it want to be. Depending upon the user requirements all services get offered
NFR-6	**Scalability**	Provides accurate prediction for user eligibility by using highly efficient algorithms and testing all the documents uploaded by the user at highly efficient rate

Chapter 5: PROJECT DESIGN

5.1 Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

DFD Level 0:



5.2 Solution & Technical Architecture

Solution Architecture:

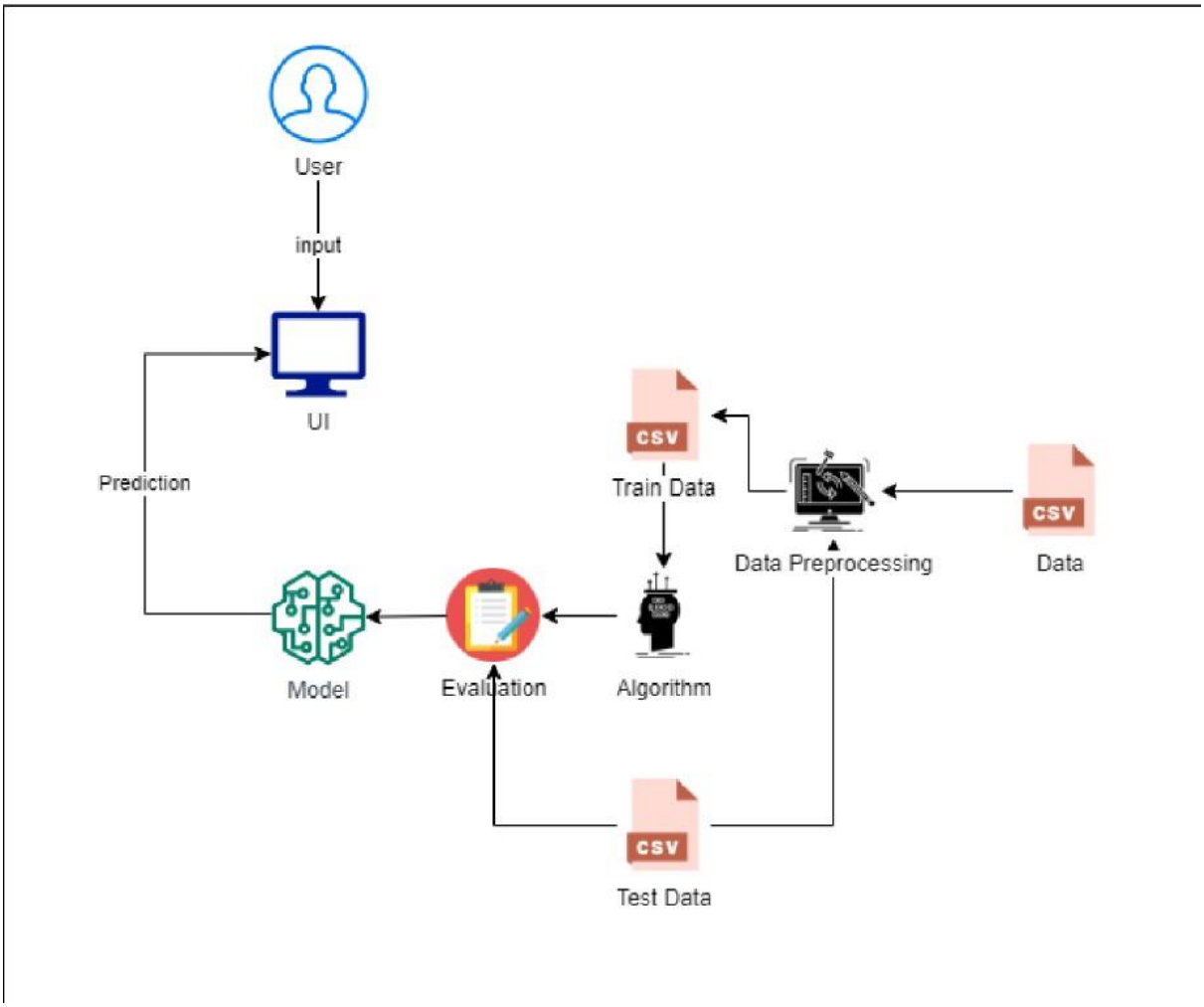
Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

● Find the best tech solution to solve existing business problems

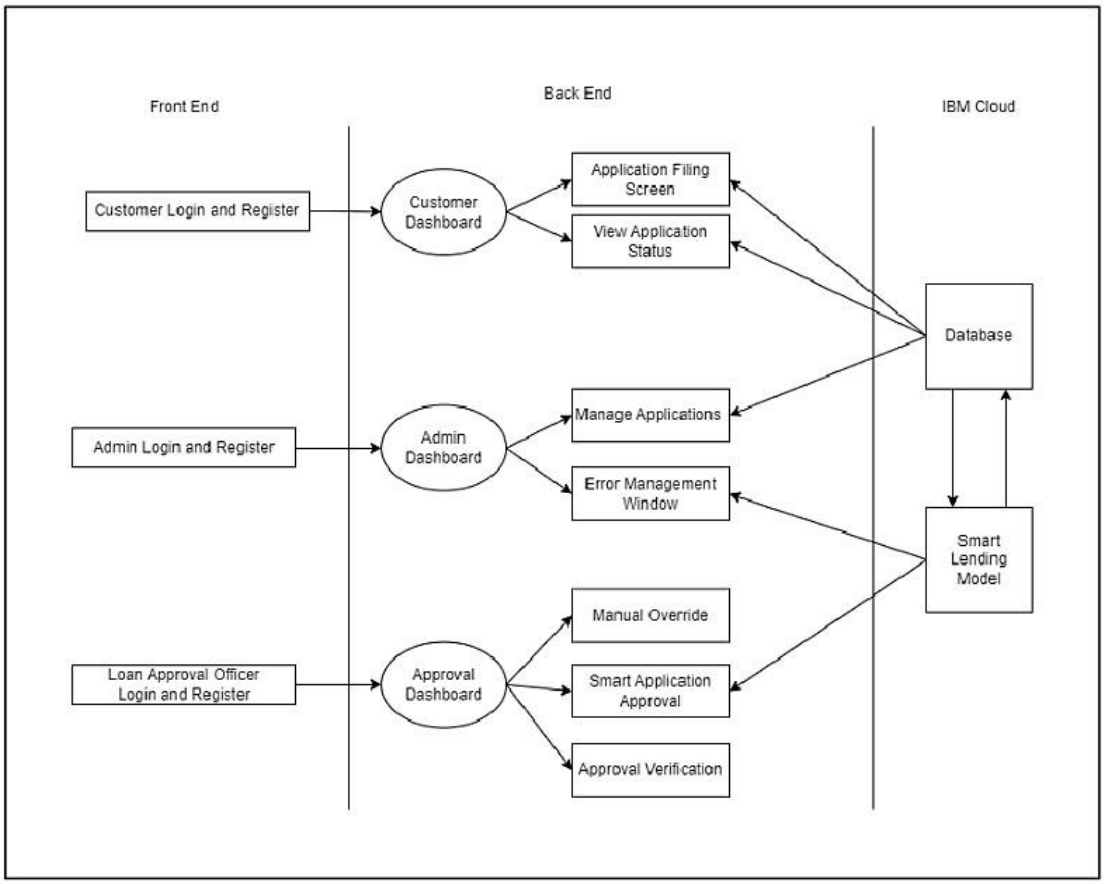
. ● Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders

. ● Define features, development phases, and solution requirements

. ● Provide specifications according to which the solution is defined, managed, and delivered.
Solution Architecture


User
input
UI
Prediction
Model
Evaluation
Algorithm
Train Data
Data Preprocessing
Data
Test Data
CSV
CSV
CSV

Technical Architecture



5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	As a user I can enter Gmail and set a password	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can get a code for confirmation	High	Sprint-1
		USN-3	Registration as a user can be confirmed using OTP or verification code	Aa a user can get OTP or verification code	Low	Sprint-1
	Login	USN-4	Users can login to web/mobile interface by storing or using the registered login credentials	Able to login	Medium	Sprint-1
		USN-5	As a user, I can log into the application by entering email & password	Can be able to login using Gmail	Medium	Sprint-1
	Dashboard	USN-6	As a user, I should be able to login the profile or status dashboard	Able to access dashboard account	Medium	Sprint-2
Customer (Web user)		USN-7	Checks the user feedback and provide essential technical support	Access the account/able to access the dashboard		Sprint-2
Loan Approval Executive	Automated analysis of score	USN-8	As a loan approval officer, I can make decisions by checking and monitoring all the feeded applications and getting to a prediction	Get a decision for loan prediction based on the details provided in the loan application	High	Sprint-3
		USN-9	As a admin cibil score which represents credit history plays major role in analysis	Cibil score/credit history place major role	High	Sprint-3
Admin	Login/Register	USN-10	As an admin I should be able to login with the unique email and password	Able to logged in	High	Sprint-4
	Dashboard	USN-11	As an admin I need the access of full authority towards the dashboard	Access the dashboard	Medium	Sprint-4

Chapter 6:PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story/Task	Story Points	Priority	Team Members
Sprint-1	Dataset	USN-4	Download the dataset	1	High	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-1		USN-5	Visualize the dataset	2	Low	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-1		USN-6	Pre-Process the dataset	3	Medium	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-1	Machine Learning Model	USN-7	KNN Model Building	5	High	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-2		USN-8	Decision Tree Model Building	5	High	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-2		USN-9	Naïve Bayes Model Building	5	High	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-2		USN-10	Fine Tuning of the Model	3	Low	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-2		USN-11	Evaluation and saving of the model	5	High	Haritha Jayavarthini Nivaethetha Ragavi

Sprint-3	Customer Interface	User	USN-12	Model Integration with flask	5	High	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-3			USN-1	As a user,I should be able to access the dashboard	3	Medium	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-3			USN-2	Selecting the loan type	3	Low	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-3			USN-3	Fill the application and check the eligibility for loan approval	5	High	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-4	Deployed website	the	USN-13	Register on IBM Cloud	3	Low	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-4			USN-14	Train the ML Model on IBM Cloud	5	Medium	Haritha Jayavarthini Nivaethetha Ragavi
Sprint-4			USN-15	Deploy the website on IBM Cloud	5	High	Haritha Jayavarthini Nivaethetha Ragavi

6.2 Sprint Delivery Schedule

Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date (Actual)**
Sprint-1	11	6 Days	24 Oct 2022	30 Oct 2022	11	30 Oct 2022
Sprint-2	18	6 Days	01 Nov 2022	07 Nov 2022	18	07 Nov 2022

Sprint-3	16	6 Days	04 Nov 2022	10 Nov 2022	16	10 Nov 2022
Sprint-4	16	6 Days	12 Nov 2022	18 Nov 2022	16	18 Nov 2022

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20(points per sprint). Let's calculate the team's average velocity(AV)per iteration unit(story points per day)

$$AV = \frac{\text{sprint duration}}{\text{velocity}} = \frac{20}{10} = 2$$

Our Project velocity

Sprint-1 = 11/6 = 1.833

Sprint-2 = 18/6 = 3

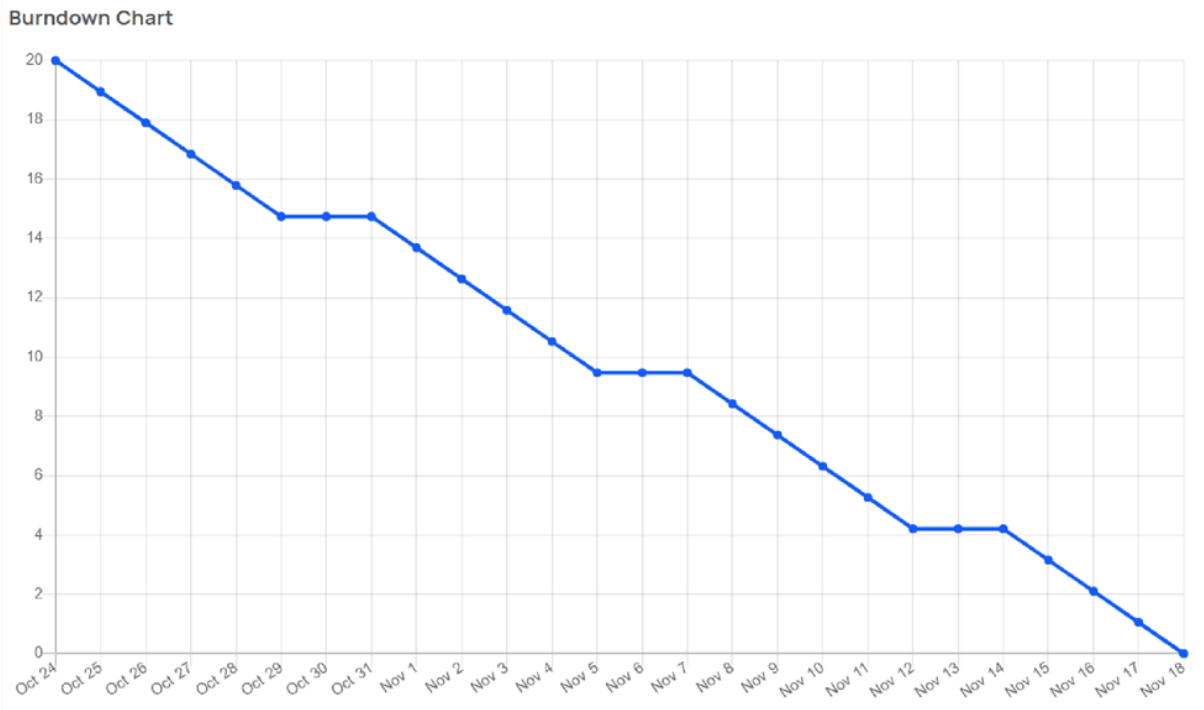
Sprint-3 = 16/6 = 2.67

Sprint-4 = 61/24 = 2.54

Burndown Chart

A burndown chart is a graphical representation of work left to do versus time.It is often used in agile software development methodologies such as Scrum.However,burndown charts can be applied to any project containing measurable progress over time.

Burndown Chart



Chapter 7: CODING & SOLUTIONING

7.1 Feature 1

app.py × model.py ×

```python
import ...
app= Flask(__name__, template_folder='templates')

model = pickle.load(open("model.pkl",'rb'))

@app.route('/')
def home():...
@app.route('/login.html')

@app.route('/home.html')
def home1():...
@app.route('/prediction.html')
def formpg():
    return render_template('prediction.html')

@app.route('/prediction.html',methods = ['POST'])
def predict():
    if request.method=='POST':
        name=request.form['Name']
        gender=request.form['gender']
        married=request.form['married']
        dependents=request.form['dependents']
        education=request.form['education']
        employed=request.form['employed']
        credit=request.form['credit']
        proparea=request.form['proparea']
        ApplicantIncome=float(request.form['ApplicantIncome'])
        CoapplicantIncome=float(request.form['CoapplicantIncome'])
        LoanAmount=float(request.form['LoanAmount'])
        Loan_Amount_Term=float(request.form['Loan_Amount_Term'])
```

app.py × model.py ×

```python
        employed = 0

    if dependents == '3+':
        dependents = 3
    if credit == 'Yes':
        credit = 1
    else:
        credit = 0
    if proparea == 'Urban':
        proparea = 2
    elif proparea == 'Rural':
        proparea = 0
    else:
        proparea = 1

    features = [gender,married,dependents,education,employed,ApplicantIncome,CoapplicantIncome,LoanAmount,Loan_Amount_Term,credit,p

    con_features = [np.array(features)]

    prediction = model.predict(con_features)
    print(prediction)
    if prediction==1:
        return render_template('approve.html',prediction_text ='Congratulations! '+name+' Your Loan is Approved')
    else:
        return render_template('reject.html',prediction_text ='Sorry '+name+' Your Loan is Denied')


if __name__ == "__main__":
    app.run(debug=True)
```

7.2 Feature 2

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Status</title>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css"/>
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiG
  <body>
    <h2 style="text-align:center">{{prediction_text}}</h2>
    <p style="text-align:center">
    <img src="../static/approved.jpeg" width="30%" height="75%">
    </p>
  </body>
</html>
```

Chapter 8: TESTING

8.1 Test Cases

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
tc01	Functional	Home Page	Verify user is able to click on Predict button		1.Enter URL and fill the form 2.Click on Predict button		Loan form should display	Working as expected	Pass				
tc02	Functional	Home Page	The web page is getting refreshed		1.Automatic page reload		Loan form must appear automatically after page reload	Working as expected	Fail	No steps needed	Y	BUG-1234	
tc03	Functional	Home page	Field address validation		1. Double-click on the E-mail address field		User should navigate to E-mail address field	Working as expected	Pass				
tc04	Functional	Output page	Loan Credibility predicted output		1. Click on eligibility chek button 2. View the predicted results		User should access the Loan credibility predicted result	Working as expected	Pass				

8.2 User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [ProductName] project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	9	3	1	2	15
Duplicate	0	0	4	0	4
External	1	2	0	0	3
Fixed	10	5	4	21	40
Not Reproduced	0	0	0	0	0
Skipped	0	0	0	0	0
Won't Fix	0	0	0	0	0
Totals	20	9	9	23	64

Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

Outsource Shipping	3	0	0	3
Exception Reporting	7	0	0	7

Final Report Output	4	0	0	4
Version Control	2	0	0	2

Chapter 9: RESULTS

9.1 Performance Metrics

Model Performance Testing:
Project team shall fill the following information in the model performance testing template.

S.No.	Parameter	Screenshot / Values
1.	Data Responsiveness	The ML model takes about 0.3 seconds to process the dataset. The credibility result is predicted in approximately 0.9 seconds
2.	Utilization of Data Filters	Sufficient data filters have been used for ideal model building
3.	Effective User Story	No of Scene Added - 15
4.	Descriptive Reports	No of Visualizations / Graphs - 13

Chapter 10: ADVANTAGES & DISADVANTAGES

Advantages:

Keep Control of the Company

A bank loans money to a business based on the value of the business and its perceived ability to service the loan by making payments on time and in full. Unlike with equity finance where the business issues shares, banks do not take any ownership position in businesses. Bank personnel also do not get involved in any aspect of running a business to which a bank grants a loan. This means you ghet to retain full management and control of your business with no external interference.

Bank Loan is Temporary

Once a business borrower has paid off a loan, there is no more obligation to or involvement with the bank lender unless the borrower wishes to take out a subsequent loan. Compare this with equity finance, where the company may be paying out dividends to shareholders for as along as the business exists.

Interest is Tax Deductible

The interest on business bank loans is tax-deductible. In addition, especially with fixed-rate loans, in which the interest rate does not change during the course of a loan, loan servicing payments remain the same throughout the life of the loan. This makes it easy for businesses to budget and plan for monthly loan payments. Even if the loan is an adjustable-rate loan, business owners can use a simple spreadsheet to compute future payments in the event of a change in rates

Disadvantages:

Tough to Qualify

One of the greatest disadvantages to bank loans is that they are very difficult to obtain unless a small business has a substantial track record or valuable collateral such as real estate. Banks are careful to lend only to businesses that can clearly repay their loans, and they also make sure that they are able to cover losses in the event of default. Business borrowers can be required to provide personal guarantees, which means the

borrower's personal assets can be seized in the event the business fails and is unable to repay all or part of a loan.

High Interest Rates

Interest rates for small-business loans from banks can be quite high, and the amount of bank funding for which a business qualifies is often not sufficient to completely meet its needs. The high interest rate for the funding a business does receive often stunts its expansion, because the business needs to not only service the loan but also deal with additional funding to cover funds not provided by the bank. Loans guaranteed by the U.S. Small Business Administration offer better terms than other loans, but the requirements to qualify for these subsidized bank loans are very strict

Chapter 11:

CONCLUSION

The analysis starts from data cleaning and processing missing value, exploratory analysis and finally model building and evaluation of the model. The best accuracy on public test set is when we get higher accuracy score and other performance metrics which will be found out. This project can help to predict the approval of bank loan or not for a candidate.

Chapter 12:

FUTURE SCOPE

In order to analyse the risk associated for the bank, credit evaluation largely involves gathering information about the customer and examining the project's technical, financial, and economic viability and this process developed a lot

Chapter 13:

APPENDIX

13.1 Source Code

index.html

```
<!doctype html>
<html lang="en">

<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="{{url_for('static', filename='/prediction.css')}}">
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0beta3/dist/css/bo
otstrap.min.css" rel="stylesheet"    integrity="sha384-
eOJMYsd53ii+scO/bJGFsiCZc+5NDVN2yr8+0RDqr0Ql0h+rP48ckxlpbzKgwra6"
crossorigin="anonymous">
  <link
href="https://unpkg.com/tailwindcss@^2/dist/tailwind.min.css"
rel="stylesheet">  <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/css/bootstrap.min.css"
integrity="sha384-
Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm"
crossorigin="anonymous">
  <title>Details Page</title>
</head>

<body>  <script>    function valid() {      var Ai =
document.getElementById("ApplicantIncome").value;      var Co =
document.getElementById("CoapplicantIncome").value;      var LA =
document.getElementById("LoanAmount").value;      var LT =
document.getElementById("Loan_Amount_Term").value;      if (Ai >
10000000000000000000000000000000000000000) {
alert("Applicant income is too large enter a valid number")
return false;
      }
      if (Co > 10000000000000000000000000000000000000000) {
alert("Coapplicant income is too large enter a valid number")
return false;
      }
```

```
      if (LA > 1000000000000000000000000000000000000000) {
alert("Loan Amount is too large enter a valid number")
return false;
      }
      if (LT > 1000000000000000000000000000000000000000) {
alert("loan amount term is too large enter a valid number")
return false;
      }
      var name =
document.getElementById("Name").value;       var
letters = /^[a-zA-Z]*$/;       if
(!name.match(letters)) {
        alert("Name must contain only alphabets")
return false;
      }
      var num = /^[0-9]+$/;       if (!Ai.match(num)) {
alert("Enter only valid numbers alphabets are not allowed  ")
return false;
      }
      if (!Co.match(num)) {         alert("Enter only valid numbers
alphabets are not allowed  ")         return false;
      }
      if (!LA.match(num)) {         alert("Enter only valid numbers
alphabets are not allowed  ")         return false;
      }
      if (!LT.match(num)) {         alert("Enter only valid numbers
alphabets are not allowed  ")         return false;
      }
      var mo = document.getElementById("mon").value;
var mn = /^[0-9]{10}$/;       if (!mo.match(mn)) {
alert("Please enter only 10 digit mobile number")
return false;
      }


    }
  </script>
  <section class="black body-font">
    <div class="container px-1 py-12 mx-auto">
<div class="flex flex-col text-center mb-10">

        <h1 class="Heading">Eligibility Check For Loan Approval</h1><br>
        <p class="fill">Fill Your Details</p>
      </div>
      <div>
</div>


      <form action='/prediction.html' method="post" onsubmit="return valid()"
class="px-24 mx-12">
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Name</label>
<input type="text" class="form-control" id="Name" name="Name" placeholder="Enter
your Name" required>
```

```
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Email ID</label>
<input type="email" class="form-control" id="email" name="email" placeholder="Enter
your Email ID" required>
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Mobile
Number</label>
          <input type="text" class="form-control" id="mon" name="mon"
placeholder="Enter your Mobile Number" required>
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Gender</label>
<select class="form-select" id="gender" name="gender" aria-label="Default select
example" required>
            <option selected>-- Select Gender --</option>
            <option value="Male">Male</option>
            <option value="Female">Female</option>
          </select>
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Married</label>
          <select class="form-select" id="married" name="married"
arialabel="Default select example" required>
            <option selected>-- Select Status --</option>
            <option value="Yes">Yes</option>
            <option value="No">No</option>
          </select>
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1"
class="formlabel">Dependents</label>
          <select class="form-select" id="dependents" name="dependents"
arialabel="Default select example" required>
            <option selected>-- Select Dependents --</option>
            <option value="0">0</option>
            <option value="1">1</option>
            <option value="2">2</option>
            <option value="3+">3+</option>
          </select>
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-
label">Education</label>
          <select class="form-select" id="education" name="education"
arialabel="Default select example" required>
            <option selected>-- Select Education --</option>
            <option value="Graduate">Graduate</option>
            <option value="Not Graduate">Not Graduate</option>
          </select>
        </div>
        <div class="mb-3">
```

```
          <label for="exampleFormControlInput1" class="form-label">Self
Employed</label>
          <select class="form-select" id="employed" name="employed"
arialabel="Default select example" required>
            <option selected>-- select Self Employed --</option>
            <option value="Yes">Yes</option>
            <option value="No">No</option>
          </select>
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Credit
History</label>
          <select class="form-select" id="credit" name="credit" aria-label="Default
select example" required>
            <option selected>-- select Credit History --</option>
            <option value="Yes">Yes</option>
<option value="No">No</option>

          </select>
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Property
Location</label>
          <select class="form-select" id="proparea" name="proparea"
arialabel="Default select example" required>
            <option selected>-- select Property Location --</option>
            <option value="Semiurban">Semiurban</option>
            <option value="Urban">Urban</option>
            <option value="Rural">Rural</option>
          </select>
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Enter Applicant
Income</label>
          <input type="text" class="form-control"
id="ApplicantIncome" name="ApplicantIncome"
placeholder="Applicant Income" required>

        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Enter
Coapplicant Income</label>
          <input type="text" class="form-control"
id="CoapplicantIncome" name="CoapplicantIncome"
placeholder="Co-applicant Income" required>
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Purpose of
loan</label>
          <select class="form-select" id="pur" name="pur" aria-label="Default
select example" required>
            <option selected>-- select the purpose of loan --</option>
            <option value="person">Personal Loan</option>
```

```
            <option value="Bussiness">Business Loan</option>
            <option value="Education">Education Loan</option>
            <option value="Home">Home Loan</option>
            <option value="Other">Other</option>
          </select>
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Enter Loan
Amount</label>
          <input type="text" class="form-control" id="LoanAmount" name="LoanAmount"
placeholder="Loan Amount" required>
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Enter Loan
Amount Term</label>
          <input type="text" class="form-control"
id="Loan_Amount_Term" name="Loan_Amount_Term"
placeholder="Loan Amount Term" required>         </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Enter Aadhar
Number</label>
          <input type="text" class="form-control" id="Adhar" name="Adhar"
placeholder="Aadhar Number" required>
        </div>
        <div class="mb-3">
          <label for="exampleFormControlInput1" class="form-label">Enter PAN Card
ID</label>
          <input type="text" class="form-control" id="PAN " name="PAN "
placeholder="PAN Card ID" required>
        </div>



        <br><br>
        <div class="mb-3">
          <button type="submit" value="PREDICT" class="btn btn-dark">Eligibility
Check</button>
        </div>
</form>



    </div>
  </section>
  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0beta3/dist
/js/bootstrap.bundle.min.js"     integrity="sha384-
JEW9xMcG8R+pH31jmWH6WWP0WintQrMb4s7ZOdauHnUtxwoG2vI5DkLtS3qm9Ekf"
crossorigin="anonymous"></script>

</body>
<style>
body{
    font-family: Arial, Helvetica, sans-serif;     font-
variant: small-caps;
```

```
}
</style>
</html>
```

app.py

```
from flask import
render_template,Flask,request import numpy as
np import pickle
from sklearn.preprocessing import scale
app= Flask(__name__, template_folder='templates')

model = pickle.load(open("model.pkl",'rb'))

@app.route('/')
def home():
    return render_template('home.html')
@app.route('/login.html')

@app.route('/home.html')
def home1():
    return render_template('home.html')
@app.route('/prediction.html')
def formpg():
    return render_template('prediction.html')

@app.route('/prediction.html',methods = ['POST'])
def predict():
    if request.method=='POST':
name=request.form['Name']
gender=request.form['gender']
married=request.form['married']
dependents=request.form['dependents']
education=request.form['education']
employed=request.form['employed']
credit=request.form['credit']
proparea=request.form['proparea']
        ApplicantIncome=float(request.form['ApplicantIncome'])
        CoapplicantIncome=float(request.form['CoapplicantIncome'])
        LoanAmount=float(request.form['LoanAmount'])
        Loan_Amount_Term=float(request.form['Loan_Amount_Term'])
if gender == 'Male':
        gender = 1
else:
        gender = 0

    if married == 'Yes':
        married = 1
else:
        married = 0
```

```
    if education == 'Graduate':
        education = 0
else:
        education = 1

    if employed == 'Yes':
        employed = 1
else:
        employed = 0

    if dependents == '3+':
        dependents = 3
if credit == 'Yes':
        credit = 1
else:
        credit = 0
if proparea == 'Urban':
        proparea = 2    elif
proparea == 'Rural':
        proparea = 0
else:
        proparea = 1

    features =
[gender,married,dependents,education,employed,ApplicantIncome,CoapplicantIncome,Loa
n Amount,Loan_Amount_Term,credit,proparea]
         con_features =
[np.array(features)]

    prediction =
model.predict(con_features)
print(prediction)    if prediction==1:
        return render_template('approve.html',prediction_text ='Congratulations!
'+name+' Your Loan is Approved')
else:
        return render_template('reject.html',prediction_text ='Sorry '+name+' Your
Loan is Denied')


if __name__ == "__main__":
app.run(debug=True)
```

13.2 GitHub& Project Demo Link GitHub link:

https://github.com/IBM-EPBL/IBM-Project-12158-1659438917

Project Demo Link:

https://youtu.be/VuhK9SC4M8Y