

Smart Lender

Applicant Credibility

Prediction for Loan Approval

IBM-Project-31088-1660196090

Team Members

Savitha A

Subbalakshmi G

Vinitha R

Sivapriya N

Table of Contents

1. INTRODUCTION

1.1 Project Overview

1.2 Purpose

2. LITERATURE SURVEY

2.1 References

2.2 Problem Statement Definition

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

4.2 Non-Functional requirements

5. PROJECT DESIGN

5.1 Data Flow Diagrams

5.2 Solution & Technical Architecture

5.3 User Stories

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

6.2 Sprint Delivery Schedule

6.3 Reports from JIRA

7. CODING & SOLUTIONING

7.1 Feature 1

7.2 Feature 2

Database Schema (if Applicable)

8. TESTING

8.1 Test Cases

8.2 User Acceptance Testing

9. RESULTS

9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

13. APPENDIX

13.1 Source Code

13.2 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

S.NO	TITLE	AUTHOR	ABSTRACT
1.	Modern Approach for Loan Sanctioning in Banks Using Machine Learning	Golak Bihari Rath. Debasish Das.BiswaRanjan Acharya.	<p>As the needs of people are increasing,the demand for loansin banks is also frequently gettinghigher every day. Usually, banks process the loan ofany applicant afterthe verification andchecking of its eligibility which is atough and time-taking process. In some cases, someapplicants default in payment resulting inloss of capital in banks. Machine learning approach would be an ideal solution to reduce human efforts and effective decision making in the loan approval process byimplementation of machine learning tools using classification algorithms to predictthe deserving applicants for loanapproval. In this paper, we build a system to construct amodel by training thesystem with records and approval results of the previously applied loan applicants. Model building is done by classification algorithms on the basis of some predictive features that categorize an outcome value as approve or</p>

			<p>disapprove. We found the logistic regression model has the best performance in comparison with other models and can be used as a predictive model reducing the risk factor in selecting the deserving applicants for loan repayment saving a lot of bank efforts and assets. Further, this model can be implemented in the banking sector allowing faster processing of loans.</p>
2.	<p>Loan Credibility Prediction System using Data Mining Techniques</p>	<p>Anuja Kadam. Pragati Namde. Sonal Shirke. Siddhesh Nandgaonkar. Dr. D. R. Ingle.</p>	<p>As we know that now-a-days there is a rapid growth in the banking sector, resulting in lots of people applying for bank loans. Finding out the applicant to whom the loan will be approved is a difficult process. Data mining techniques are becoming very popular nowadays because of the wide availability of huge quantities of data and the need for transforming such data into knowledge. Techniques of data mining are implemented in various domains such as retail industry, telecommunication industry, biological data analysis, etc. In this paper, we proposed a model which predicts loan</p>

			<p>approval/rejection of an applicant using data mining techniques. This can be done by training the model with the data of the previous records of the people applied for loan.</p>
3.	Loan Prediction using Decision Tree and Random Forest	<p>Kshitiz Gautam. Arun Pratap Singh. Keshav Tyagi. Mr. Suresh Kumar.</p>	<p>In India, the number of people or organizations applying for loans increases every year. The bank employees have to put in a lot of work to analyze or predict whether the customer can pay back the loan amount or not (defaulter or non-defaulter) in the given time. The aim of this paper is to find the nature or background or credibility of the client that is applying for the loan. We use exploratory data analysis techniques to deal with the problem of approving or rejecting the loan request or in short loan prediction. The main focus of this paper is to determine whether the loan given to a particular person or an organization shall be approved or not.</p>
4.	Logistic Regression Based Loan Approval Prediction	<p>Sai Aparna Vangaveeti. Naga Likitha Venna.</p>	<p>As we know that now-a-days there is a rapid growth in the banking sector, resulting in</p>

		Prasanna Naga Sri RamyaYajaman am Kidambi. Harika Marneni. Naga Satish KumarMaganti	lots of people applying for bank loans. Findingout the applicant to whom the loan will beapproved is a difficult process. In this paper,we proposed a modelwhich predicts loan approval/rejection of an applicant using machine learning techniques. This can be done by training the model with the data of the previous records of the peopleapplied for loan
--	--	---	---

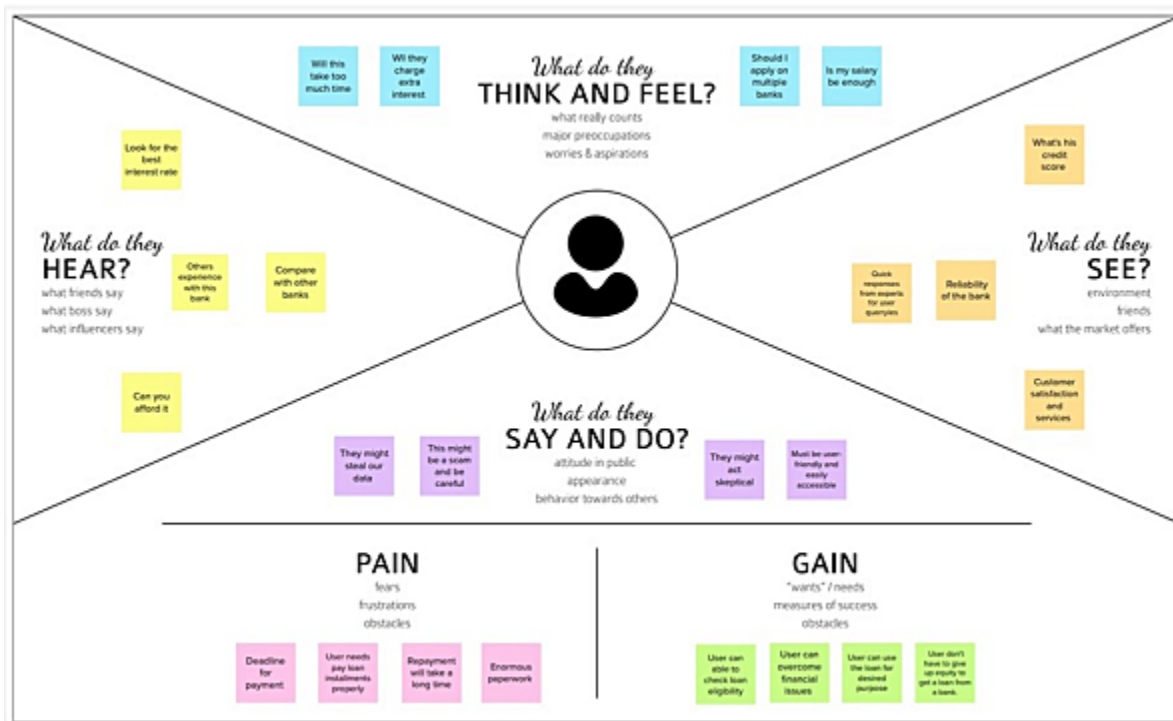
2.2 Problem Statement Definition

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

3.1 Empathy 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

Smart Lender - Applicant Credibility Prediction for Loan Approval

The system provides the most relevant attributes that help in determining whether to approve or reject the loan application. This aids in predicting the credibility of future customers. In the final stage, the designed system is tested with test set and the performance is assured.



Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

🕒 10 minutes



Team gathering

Define who should participate in the session and send an invite. Share relevant information on pre-work shared.



Set the goal

Think about the problem you'll be focusing on solving in the brainstorming session.



Learn how to use the facilitation tools

Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#) →



Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

🕒 5 minutes



Key rules of brainstorming

To run an smooth and productive session



Stay in topic.



Encourage wild ideas.



Defer judgment.



Listen to others.



Go for volume.



If possible, be visual.

3

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

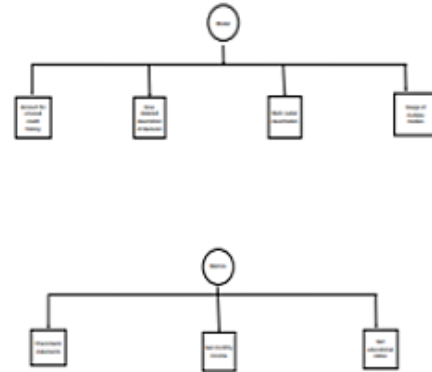


3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

20 minutes



Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

🕒 20 minutes

Lenders also typically require minutes and/or resolutions that document Board approval of the loan.	Loan applications ask for your annual income, and you can include money earned from part-time work.	If you meet a lender's minimum qualifications and want to see estimated rates and terms, you can pre-qualify for a personal loan.	If you're already a customer with the bank, you may receive an additional APR discount.	Lower APR than other types of personal loan lenders.	Missed payment does not hurt your credit score.
The lender will ask and will require documentation to confirm you have the finances in order to repay the loan.	Understanding credit score One person has one identity.	A credit card is technically a loan that is referred to as a revolving line of credit. Every time you use your card, your credit's buying limit sends you the money to pay for your purchase.	The documents asked for by any loan app are basic and minimal. These include your identity proof, address proof, and income proof.	Depending on your eligibility you can avail a loan amount of your own choice.	Such apps provide instant personal loan starting from ₹8K-10,000 to ₹20 Lakh. You can get a loan as per your requirement and eligibility.
Beware of the high origination fees associated with payday cash advances.	Consumers are easily confused about how debt consolidation works because the industry uses interchangeable terms.	Specific guidelines issued by different agencies determine how medical collections, appearing on your credit report will affect a mortgage approval and interest rate.	A debt consolidation loan means that one new lender pays off what you owe to multiple old creditors.	The critical approval criteria are a good credit score and sufficient income to cover your monthly payment along with your existing mortgage.	Seeing out and then repaying an installment loan on time should improve your score in the future, allowing you to borrow money at better rates.
Sometimes, if you have a bad credit score, or are currently unemployed, that it can be tricky to get approved for a loan. A great way to help ensure that you are approved is to get a cosigner.	Sometimes if you are getting a loan from a bank or creditor that is different from your own, providing them with a bank statement can help to improve the speed at which you can obtain your loan.	One of the best ways to get easily approved for any kind of loan is to have an extremely good credit rating.	People with benefit job security have the best approval odds because the lender can count on your income continuing through the end of the loan term.	Personal loans based on income and not a credit score can help borrowers with weak creditlines obtain quick funding for a vacation, an emergency cash etc..	If you ask the lender to disregard a critical underwriting factor, you must shine in this area.

TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.



3.3 Proposed Solution

S.No.	Parameter	Description
1	Problem Statement (Problem to be solved)	Insufficient Income . Not having steady source of income.
2.	Idea / Solution description	A steady source of income is the proof for any lender that you are capable of repaying your personal loan.
3.	Novelty / Uniqueness	Banks have set a minimum monthly salary cap.
.4.	Social Impact / CustomerSatisfaction	In the absence of a job, ensure that you have other sources of income.
5.	Business Model (RevenueModel)	Avoid applying for a loan when you have just switched jobs.
6.	Scalability of the Solution	Interest payouts or rental income, that will help you make EMI payments.

3.4 Problem Solution fit

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CC <ul style="list-style-type: none">Demographic segmentation- These include things like gender, age, family status, occupation, level of education, income level, religion, race, and ethnicity.Behavioral Segmentation-This includes both shopping behavior and purchase behavior.	6. CUSTOMER CONSTRAINTS CC <ol style="list-style-type: none">check loan document and put them in ordercategorize loanCheck for credit ratingEnter loan application into the systemThen loan approval or rejection decision is made	5. AVAILABLE SOLUTIONS CC <ol style="list-style-type: none">First of all identify the solutions for their problemsCustomer wants to increase their incomeMake a budget to help you resolve their financial problemsAvoid buying new thingsCustomers meet their advisor to discuss about their situation	Explore AS, differentiate
	2. JOBS-TO-BE-DONE / PROBLEMS AS <p>The loan is need to be provided to trustable borrower so the borrower need to be evaluated</p>	9. PROBLEM ROOT CAUSE AS <p>The loan borrower may not properly pay back loan if the loan is provided bend of the borrower capability.</p>	7. BEHAVIOUR AS <p>Verify whether the loan requested person is eligible for loan, based on the different parameters like person's economic potential, property support, financial performance, etc.</p>	
Focus on J&P, tap into BE, understand	Focus on J&P, tap into BE, understand			

3. TRIGGERS TR <p>The income source for the bank's is the interest of the loan which has been given to the loan requesters. So, the banks are willing to give the loan for trustable borrower.</p>	10. YOUR SOLUTION SL <ol style="list-style-type: none">Offer consistent companywide messagingProvide instructions for easy adoptionNurture customer relationshipsSolve for the right customers needs.Build feedback loops into every stage of the process.	8. CHANNELS of BEHAVIOUR CH <p>If some customers can take a online action by lending a money through web platforms or mobile apps, utilizing technology for authentication and credit evaluation.</p> <p>If some customers can take a offline action by offline payments are transaction processed asynchronously, offline payments are made via cash, checks, bank transfer, postal orders etc..</p>
--	--	---

<p>4. EMOTIONS: BEFORE / AFTER EM</p> <p>While evaluating the loan applicant the bankers struggle in deciding how to evaluate the loan applicant, which are the things to be considered and what are the criteria level needed to be checked.</p> <p>If the loan borrowers are not paying back the loan at specified intervals then the banks are not able to generate income which is necessary for maintaining the bank and providing the interest for the depositors, providing the interest for the depositors.</p>		<p>Flexibility for customer to pay through offline payment methods can help extend customers business.</p>
--	--	--

Chapter 4: REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	<ul style="list-style-type: none"> Registration through Form Registration through Gmail Registration through LinkedIn
FR-2	User Confirmation	<ul style="list-style-type: none"> Confirmation via Email Confirmation via OTP
FR-3	User Application	<ul style="list-style-type: none"> Filling of application Modification of application Verification of application
FR-4	Loan Issuance	<ul style="list-style-type: none"> Checking status of loan Loan Approval Loan Rejection

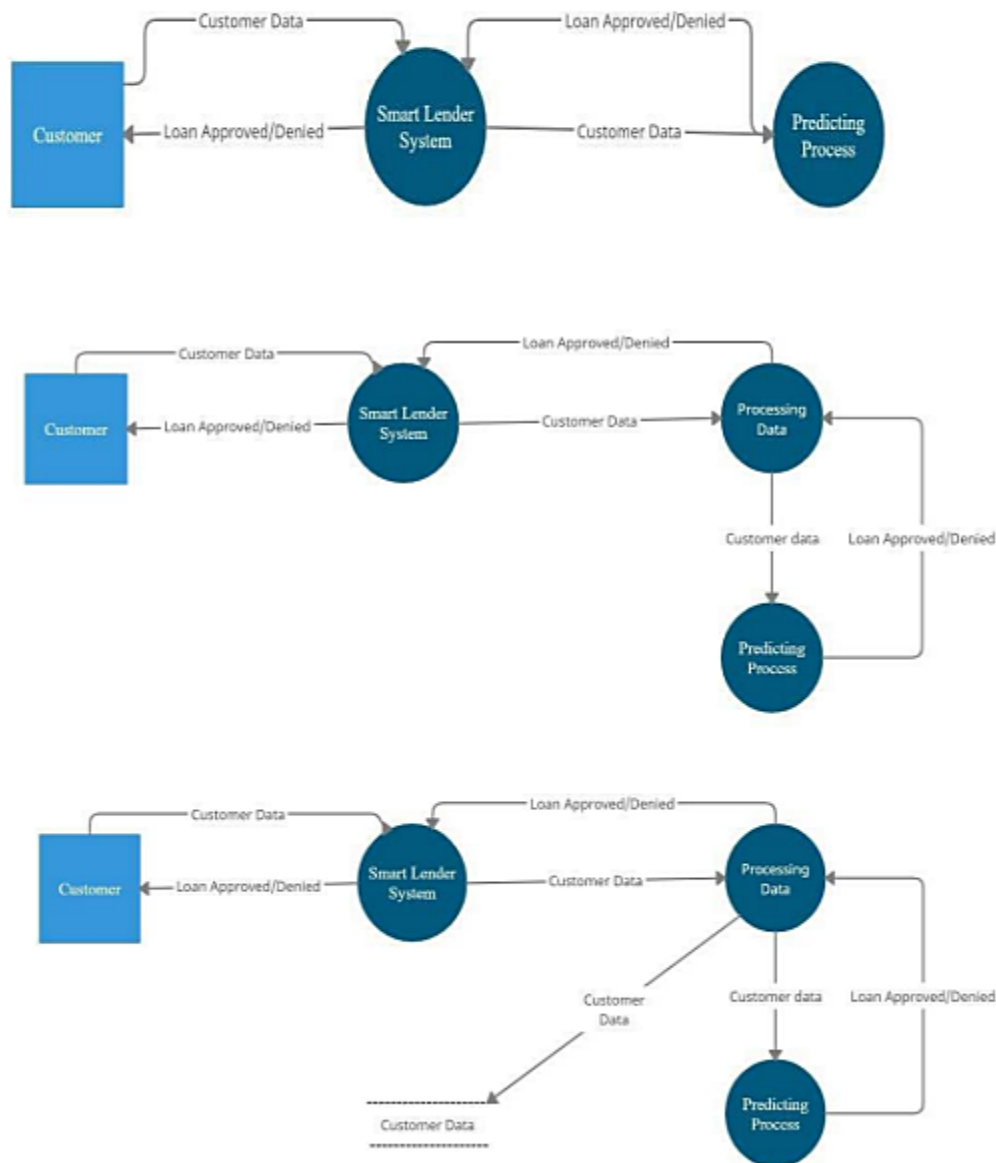
4.2 Non-Functional requirements

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	<ul style="list-style-type: none">• Easy to use and self-explaining website.• Easy navigation between pages.• Simple structure for faster access.
NFR-2	Reliability	<ul style="list-style-type: none">• Ensembling the outputs of various ML models.
NFR-3	Performance	<ul style="list-style-type: none">• Web Based Application.• Ability to indicate user inputs of erroneous data types.
NFR-4	Availability	<ul style="list-style-type: none">• Application is available 24 / 7 as it is hosted on IBM cloud.• Simple web browser is enough to access the website.
NFR-5	Scalability	<ul style="list-style-type: none">• Can be extended for other types of loans.• Aadhar and PAN verification can also be implemented.

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.

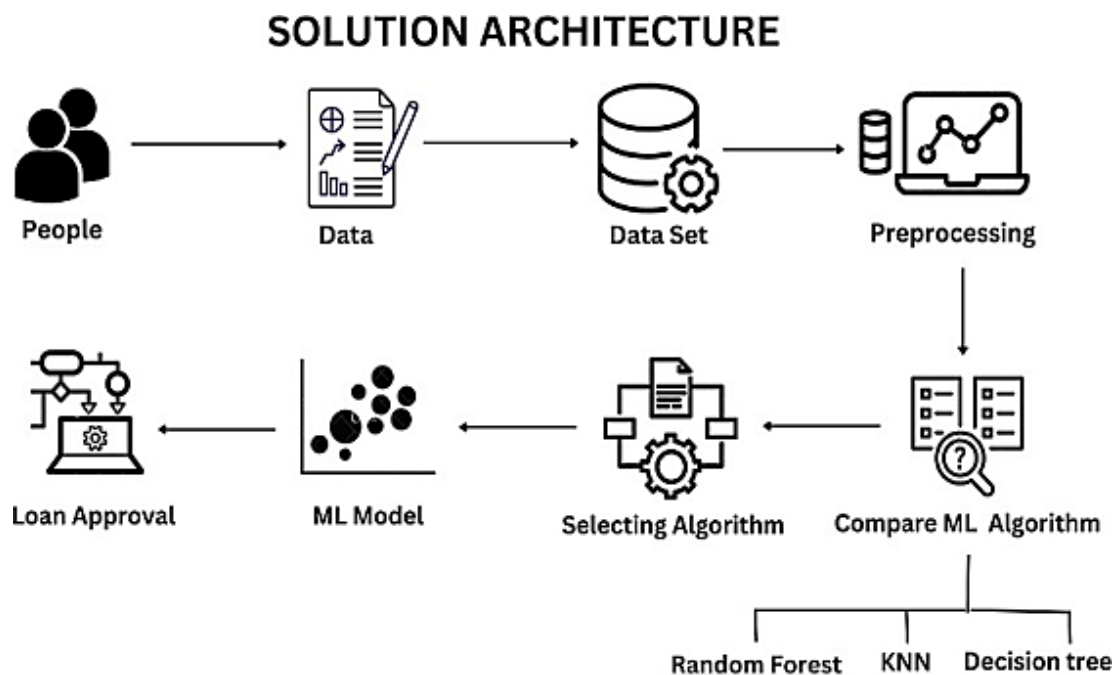


5.2 Solution & Technical

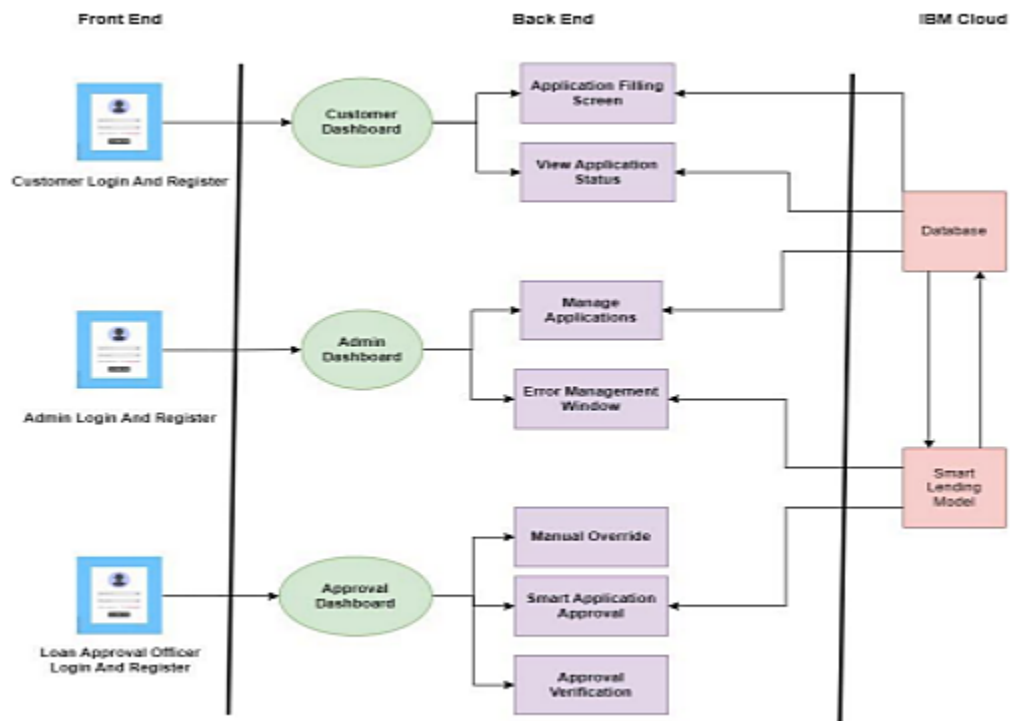
Architecture Solution Architecture:

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.



Technical Architecture:



5.3 User Stories

User Type	Functional Requirement	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Money lender (Web user)	Dashboard	USN-1	As a user, I should be able to access the dashboard.	Access the dashboard	Low	Sprint 3
		USN-2	Select the type of loan	Select the type of loan	Medium	Sprint 3
	Check for loan eligibility	USN-3	Fill the application with the details of the borrower.	Check the eligibility of the borrower.	High	Sprint 4
Borrower	Dashboard	USN-4	Should be able to access the dashboard.	Access the dashboard.	Low	Sprint 3
		USN-5	Choose the type of loan	Choose the type of loan	Medium	Sprint 3

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	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Savitha, Sivapriya, Subbalakshmi, Vinitha
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Savitha, Sivapriya, Subbalakshmi, Vinitha
Sprint-2		USN-3	As a user, I can register for the application through Facebook	2	Low	Savitha, Sivapriya, Subbalakshmi, Vinitha
Sprint-1		USN-4	As a user, I can register for the application through Gmail	2	Medium	Savitha, Sivapriya, Subbalakshmi, Vinitha
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	1	High	Savitha, Sivapriya, Subbalakshmi, Vinitha

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
	Dashboard					Savitha, Sivapriya, Subbalakshmi, Vinitha

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	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	29 Oct 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	29 Oct 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	29 Oct 2022

Velocity

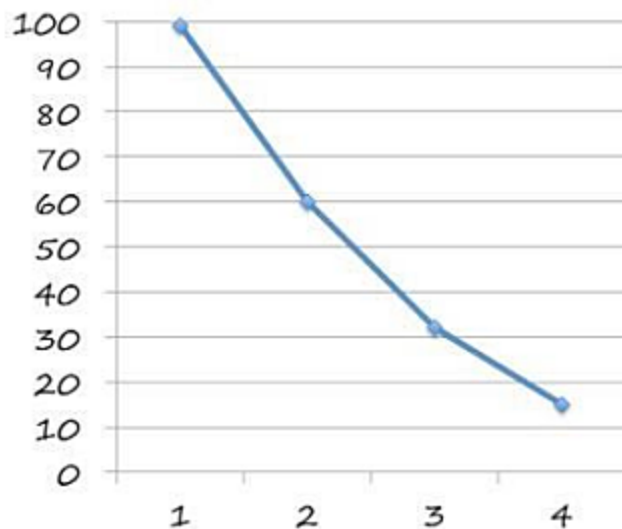
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

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

Our Project velocity

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)

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



In our project, there are 4 sprint activities.

This chart is drawn by taking

X - sprint and

Y - Pending hours.

Chapter 7: CODING & SOLUTIONING

7.1 Feature 1

The image displays a web application for "LOAN ELIGIBILITY PREDICTION" and its underlying Python code. The web interface at the top features a title "LOAN ELIGIBILITY PREDICTION" and a subtitle "FILL THE FORM FOR PREDICTION". Below this, there are several input fields for user information: NAME, EMAIL ID, MOBILE NUMBER, GENDER (a dropdown menu with "-- Select Gender --"), MARRIED (a dropdown menu with "-- Select Status --"), DEPENDENTS (a dropdown menu with "-- Select Dependents --"), and EDUCATION (a dropdown menu with "-- Select Education --").

The bottom portion of the image shows the Python code for the application, displayed in a VS Code editor. The code is a Flask application that handles a POST request to the "/predict" endpoint. It extracts form data for "proparea", "ApplicantIncome", "LoanAmount", and "Loan_Amount_Term". It then uses a series of conditional statements to assign values to "gender", "married", "education", "employed", "dependents", "credit", and "proparea" based on the form inputs. These values are combined into a "features" list, which is converted to a NumPy array and passed to a "model.predict" function. The application then renders either "approve.html" or "reject.html" based on the prediction result.

```
36 proparea = request.form['proparea']
37 ApplicantIncome = float(request.form['ApplicantIncome'])
38 CoapplicantIncome = float(request.form['CoapplicantIncome'])
39 LoanAmount = float(request.form['LoanAmount'])
40 Loan_Amount_Term = float(request.form['Loan_Amount_Term'])
41
42 if gender == 'Male':
43     gender = 1
44 else:
45     gender = 0
46
47 if married == 'Yes':
48     married = 1
49 else:
50     married = 0
51
52 if education == 'Graduate':
53     education = 0
54 else:
55     education = 1
56
57 if employed == 'Yes':
58     employed = 1
59 else:
60     employed = 0
61
62 if dependents == '3+':
63     dependents = 3
64 if credit == 'Yes':
65     credit = 1
66 else:
67     credit = 0
68 if proparea == 'Urban':
69     proparea = 2
70 elif proparea == 'Rural':
71     proparea = 0
72 else:
73     proparea = 1
74
75 features = [gender, married, dependents, education, employed, ApplicantIncome,
76             CoapplicantIncome, LoanAmount, Loan_Amount_Term, credit, proparea]
77
78 con_features = np.array(features)
79
80 prediction = model.predict(con_features)
81 print(prediction)
82 if prediction == 1:
83     return render_template('approve.html', prediction_text='Congratulations! '+name+' You are eligible for loan')
84 else:
85     return render_template('reject.html', prediction_text='Sorry '+name+' You are not eligible for loan')
86
87
88 if __name__ == '__main__':
89     app.run(debug=True)
```



```
EXPLORE
OPEN EDITORS
app.py
SPRINT 3
static
home.css
money.jpg
rejected.jpg
templates
approve.html
home.html
prediction.html
reject.html
app.py
loan_prediction...
model.pkl
model.py

app.py
1 from flask import render_template, Flask, request
2 import numpy as np
3 import pickle
4 from sklearn.preprocessing import scale
5 app = Flask(__name__, template_folder='templates')
6
7 model = pickle.load(open('model.pkl', 'rb'))
8
9
10 @app.route('/')
11 def home():
12     return render_template('home.html')
13
14
15 @app.route('/login.html')
16 @app.route('/home.html')
17 def home1():
18     return render_template('home.html')
19
20
21 @app.route('/prediction.html')
22 def formgg():
23     return render_template('prediction.html')
24
25
26 @app.route('/prediction.html', methods=['POST'])
27 def predict():
28     if request.method == 'POST':
29         name = request.form['name']
30         gender = request.form['gender']
31         married = request.form['married']
32         dependents = request.form['dependents']
33         education = request.form['education']
34         employed = request.form['employed']
35         credit = request.form['credit']
36         proarea = request.form['proarea']
37         ApplicantIncome = float(request.form['ApplicantIncome'])
38         CoapplicantIncome = float(request.form['CoapplicantIncome'])
39         LoanAmount = float(request.form['LoanAmount'])
40         LoanAmount_Term = float(request.form['LoanAmount_Term'])
41         if gender == 'Male':
42             gender = 1
43         else:
44             gender = 0
45
46         if married == 'Yes':
47             married = 1
48         else:
49             married = 0
50
51         if education == 'Graduate':
52             education = 0
53         else:
54             education = 1
55         if employed == 'Yes':
56             employed = 1
57         else:
58             employed = 0
59
60         X = np.array([name, gender, married, dependents, education, employed, credit, proarea, ApplicantIncome, CoapplicantIncome, LoanAmount, LoanAmount_Term]).reshape(1, -1)
61         X_scaled = scale(X)
62         prediction = model.predict(X_scaled)
63         if prediction == 0:
64             return render_template('reject.html', name=name)
65         else:
66             return render_template('approve.html', name=name)
```

7.2 Feature 2



Sorry subbalakshmi You are not eligible for loan



```
EXPLORE
  OPEN EDITORS
    prediction.html
  SPAN 3
    static
      home.css
      money.jpg
      rejected.jpg
    templates
      approve.html
      home.html
      prediction.html
      reject.html
    app.py
    loan_prediction...
    model.pkl
    model.py
  OUTLINE
  TIMELINE
  HTML SCRIPTS
  FILTERS

prediction.html
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206

<option selected== select Property Location ==>/option>
<option value="London">London</option>
<option value="Paris">Paris</option>
<option value="Mumbai">Mumbai</option>
</select>
</div>
<div class="mb-3">
<label form="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 form="exampleFormControlInput2" class="form-label">Enter Co-applicant Income</label>
<input type="text" class="form-control" id="CoapplicantIncome" name="CoapplicantIncome"
placeholder="Co-applicant Income" required>
</div>
<div class="mb-3">
<label form="exampleFormControlInput3" class="form-label">Purpose of loan</label>
<select class="form-select" id="purpose" name="purpose" aria-label="Default select example" required>
<option selected== select the purpose of loan ==>/option>
<option value="Personal">Personal Loan</option>
<option value="Business Loan">Business Loan</option>
<option value="Education Loan">Education Loan</option>
<option value="Home Loan">Home Loan</option>
<option value="Other">Other</option>
</select>
</div>
<div class="mb-3">
<label form="exampleFormControlInput4" 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 form="exampleFormControlInput5" 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 form="exampleFormControlInput6" class="form-label">Enter Aadhar Number</label>
<input type="text" class="form-control" id="Aadhar" name="Aadhar" placeholder="Aadhar Number" required>
</div>
<div class="mb-3">
<label form="exampleFormControlInput7" class="form-label">Enter PAN Card ID</label>
<input type="text" class="form-control" id="PAN" name="PAN" placeholder="PAN Card ID" required>
</div>
</div>
<div class="mb-3">
<button type="submit" value="PREDICT" class="btn btn-dark">Predict</button>
</div>
</form>
</div>

Python 3.10.0 64-bit
1.6.0 19
New / prediction.html Quokka
LF 1 Col 1 Rows 2 UTF-8 LF HTML Go Live Preview
```

```
EXPLORE
  OPEN EDITORS
    prediction.html
  SPAN 3
    static
      home.css
      money.jpg
      rejected.jpg
    templates
      approve.html
      home.html
      prediction.html
      reject.html
    app.py
    loan_prediction...
    model.pkl
    model.py
  OUTLINE
  TIMELINE
  HTML SCRIPTS
  FILTERS

prediction.html
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139

<form action="/prediction.html" method="post" onsubmit="return validate()" class="px-24 mx-12">
<div class="mb-3">
<label form="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 form="exampleFormControlInput2" 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 form="exampleFormControlInput3" class="form-label">Mobile Number</label>
<input type="text" class="form-control" id="mob" name="mob" placeholder="Enter your Mobile Number" required>
</div>
<div class="mb-3">
<label form="exampleFormControlInput4" 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 form="exampleFormControlInput5" class="form-label">Married</label>
<select class="form-select" id="married" name="married" aria-label="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 form="exampleFormControlInput6" class="form-label">Dependents</label>
<select class="form-select" id="dependents" name="dependents" aria-label="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 form="exampleFormControlInput7" class="form-label">Education</label>
<select class="form-select" id="education" name="education" aria-label="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 form="exampleFormControlInput8" class="form-label">Self Employed</label>
<select class="form-select" id="employed" name="employed" aria-label="Default select example" required>
<option selected== select Self Employed ==>/option>
<option value="Yes">Yes</option>
<option value="No">No</option>
</select>
</div>
</div>

Python 3.10.0 64-bit
1.6.0 19
New / prediction.html Quokka
LF 1 Col 1 Rows 2 UTF-8 LF HTML Go Live Preview
```

Chapter 8: TESTING

8.1 Test Cases

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
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				
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	
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				
Functional	Output page	Loan Credibility predicted output		1. Click on predict 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	Severity1	Severity2	Severity3	Severity4	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.

	Total Cases	Not Tested	Fail	Pass
Print Engine	5	0	0	5
Application	46	0	0	46
Security	4	0	0	4
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.

	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	Utilisation of DataFilters	Sufficient data filters have been used for ideal model building
3	Effective User Story	No of Scene Added - 15
4	Descriptive Reports	No of Visualisations / 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 service the loan by making payments on time and in full. Unlike with equity business issues shares, banks do not take any ownership position in businesses. Bank also do not get involved in any aspect of running a business to which a bank grants means you get to retain full management and control of your business with interference.

Bank Loan is Temporary

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

Interest is Tax Deductible

The interest on business bank loans is tax-deductible. In addition, especially with in which the interest rate does not change during the course of a loan, loan remain the same throughout the life of the loan. This makes it easy for businesses to plan for monthly loan payments. Even if the loan is an adjustable-rate loan, business 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 small business has a substantial track record or valuable collateral such as real estate careful to lend only to businesses that can clearly repay their loans, and they also 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

Source Codeindex.html

```
<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" aria-
label="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="form-
label">Dependents</label>

  <select class="form-select" id="dependents"
name="dependents" aria-label="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" aria-label="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" aria-label="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" aria-label="Default selectexample" 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 Co-applicantIncome</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 AmountTerm</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">Predict</button>

</div>

</form>

```

app.py

```

from flask import
render_template,
Flask, requestimport
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, LoanAmount, 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+''
```

```
You are eligible for loan')
```

```
else:
```

```
return render_template('reject.html', prediction_text='Sorry '+name+' You are not
```

```
eligible for loan')
```

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

GitHub& Project

Demo Link

Github Link:

<https://github.com/IBM-EPBL/IBM-Project-31088-1660196090>

Project Link:

https://drive.google.com/file/d/1_VpNEYa_2Ywbw9VBW8vvANW8GmeZcPsJ/view?usp=share_link