SMART LENDER – APPLICANT CREDIBILITY PREDICTION FOR LOAN APPROVAL

Bonafede record of work done by

ADITYA SHARMA	(19Z303)
DHARMA DHURAI V	(19Z312)
PRADEEP D	(19Z338)
MKESH S	(20Z462)

Professional Readiness for Innovation, Employability, and Entrepreneurship

GUIDE: GOPIKA RANI N

BACHELOR OF ENGINEERING

BRANCH: COMPUTER SCIENCE AND ENGINEERING

NOVEMBER 2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

PSG COLLEGE OF TECHNOLOGY
(Autonomous Institution)

COIMBATORE – 641 004

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.

This Guided Project mainly focuses on applying Machine Learning techniques which can be used to perform such classifications of the credit defaulters as they are very crucial and useful in the prediction of these types of data. Classification algorithms such as Decision tree, Random forest, KNN, Xgboost and SVM will be used. The data is trained and tested with these algorithms and finally, the best model is selected and saved in pkl format. Then, flask integration and IBM deployment will be done.

1.2 Purpose

This application will help the bank employees to classify the credit defaulters accurately with minimum error. So, the non - profit losses of the banks have been reduced. Thus, they may recover the approved loans with minimum losses. They need not put much effort into making decisions for loan approvals rather can leave that part to the application. The customers(bankers) get satisfied by the results of the application. Thus, better the accuracy, better classification of the applicants and better the satisfaction.

CHAPTER 2

LITERATURE SURVEY

2.1 Existing Problem

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. But the prediction of credit defaulters is one of the difficult tasks for any bank.

2.2 Problem Definition

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. Therefore, bank employees need a way to do such a thing so that recovery of approved loans can take place without any loss.

Machine Learning techniques can be used to perform such classifications of the credit defaulters as they are very crucial and useful in the prediction of these types of data. The pre-processed dataset will be trained and tested on with the ML algorithms. Finally, a best model is selected and used in the application.

2.3 References

1. Ashwini S. Kadam, Shraddha R Nikam, Ankita A. Aher, Gayatri V. Shelke, Amar S. Chandgude, 2021, "Prediction for Loan Approval using Machine Learning Algorithm", No "Apr" / "2021" "Crude Oil Price Prediction Using LSTM Networks". 5. https://medium.com/swlh/lending-club-data-web-app-ada56ff64cee (2018)
2. "Sivasree M S, Rekha Sunny T, (2015), "Loan Credibility Prediction System Based on Decision Tree Algorithm", No "September" / "2015".
3. Anuja Kadam, Pragati Namde, Sonal Shirke, Siddhesh Nandgaonkar, Dr.D.R Ingle, 2021, "Loan Credibility Prediction System using Data Mining Techniques" No "May" / "2021".
4. Pidikiti Supriya , Myneedi Pavani , Nagarapu Saisushma , Namburi Vimala Kumari , K Vikas, 2019, "Loan Prediction by using Machine Learning Models", No "April" / "2019".
5. https://medium.com/swlh/lending-club-data-web-app-ada56ff64cee.
6. https://github.com/smartinternz02/SI-GuidedProject-48927-1652694502.
7. https://www.academia.edu/77162007/BANK_LOAN_PREDICTION_USING_MACHINE_LEARNING.

CHAPTER 3

IDEATION AND PROPOSED SOLUTION

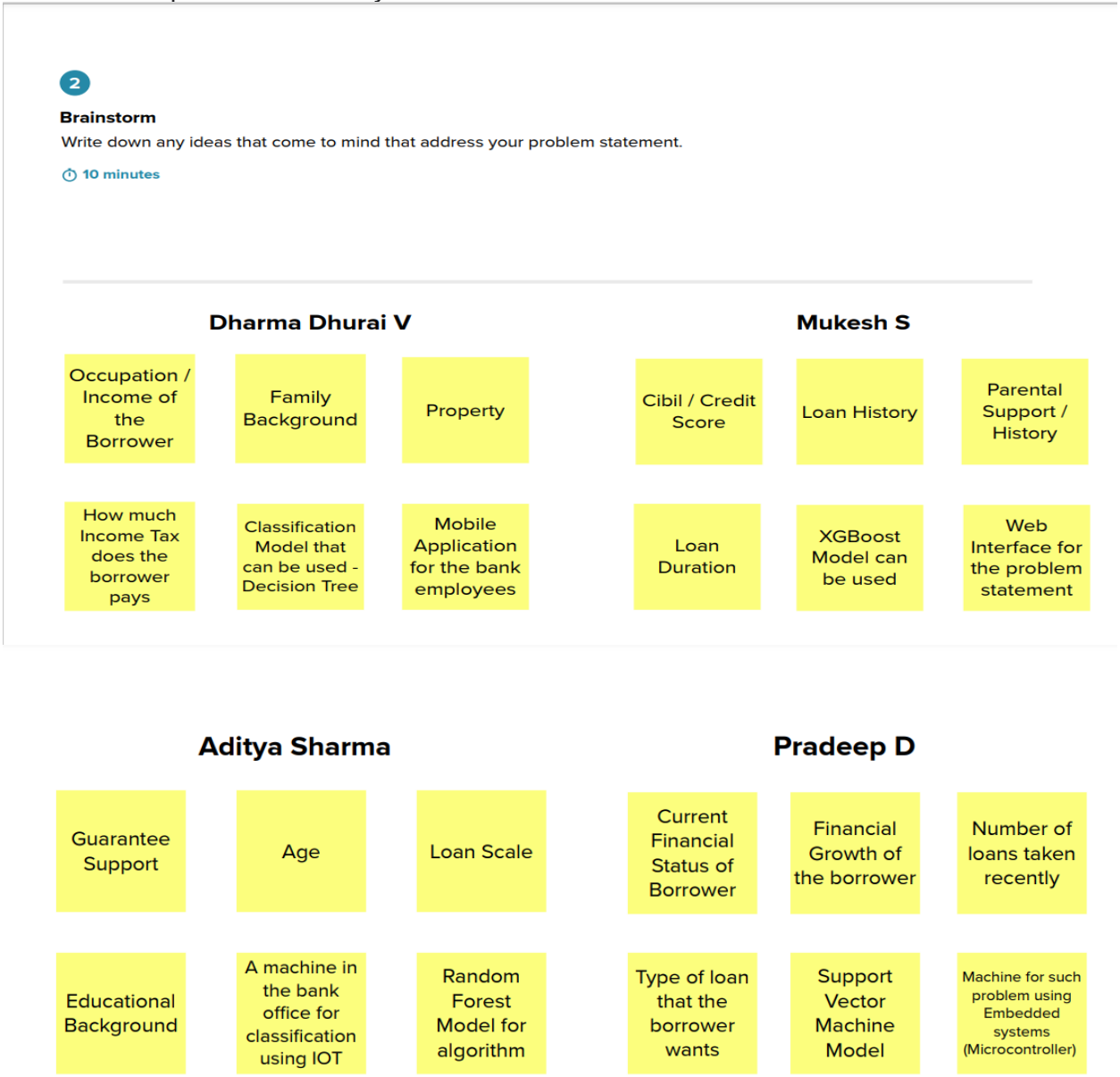
3.1 Empathy Map

The primary purpose of the empathy map is to bridge the understanding of the user and developer. Figure 3.1 represents the empathy map for the Smart lender applicant credibility prediction for loan approval.

Figure 3.1 – Empathy Map

3.2 Ideation and Brainstorming

This is often the most exciting stage in a project, because during Ideation and brainstorming, the aim is to generate a large quantity of ideas that the team can then filter and cut down into the best, most practical, or most innovative ones to inspire new and better design solutions and products. Figure 3.2 shows the stages of ideation and brainstorming for the machine learning based vehicle performance analyzer.



3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

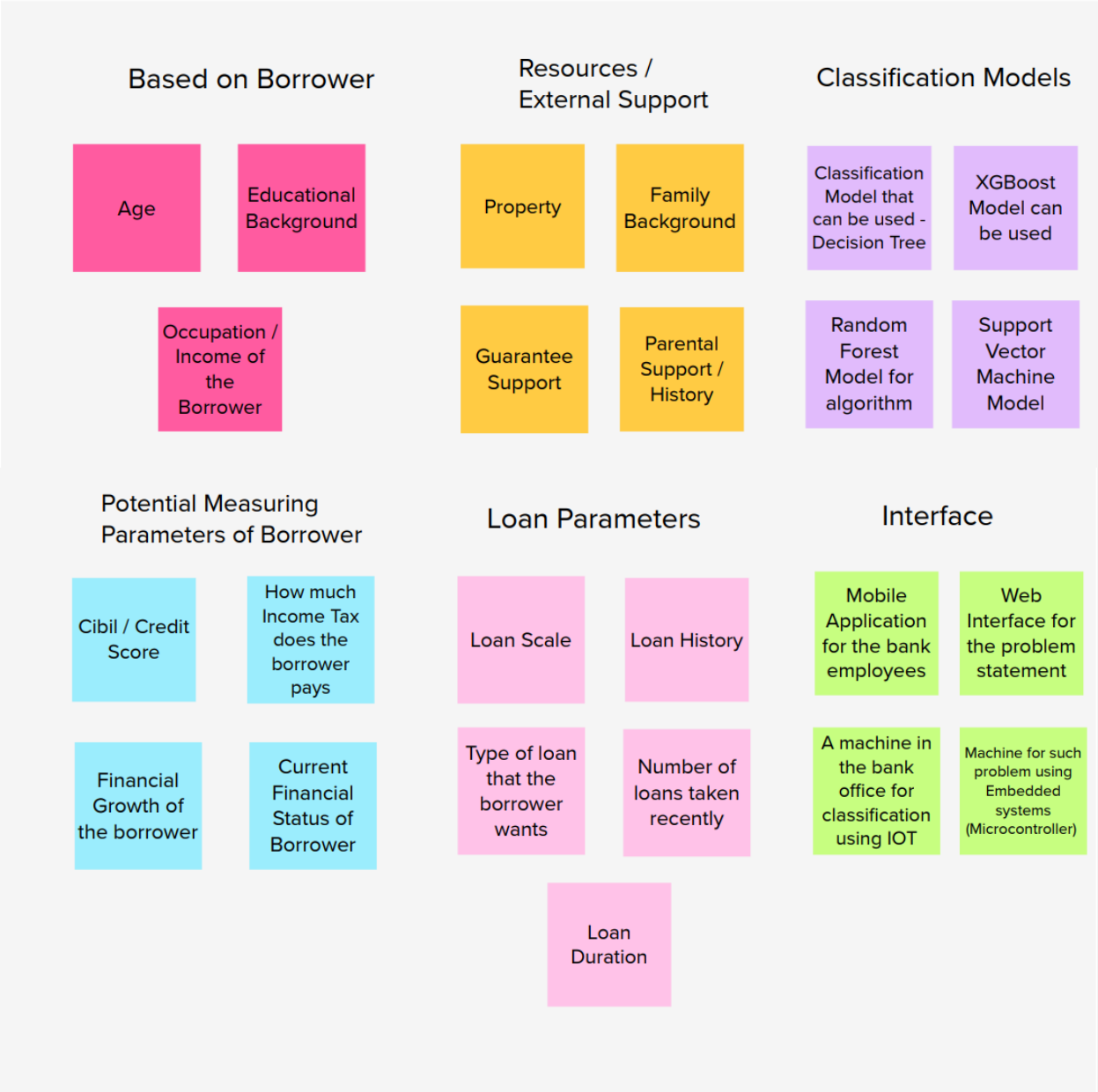


Figure 3.2 – Ideation & Brainstorming

3.3 Proposed Solution

S.No.	Parameter	Description
1	**Problem Statement (Problem to be solved)**	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.
2	**Idea / Solution description**	This solution uses Machine Learning techniques which can be used to perform such classifications of the credit defaulters as they are very crucial and useful in the prediction of these types of data. Classification algorithms such as Decision tree, Random forest, KNN, Xgboost and SVM will be used. The data is trained and tested with these algorithms and finally, the best model is selected and saved in pkl format. Then, flask integration and IBM deployment will be done
3	**Novelty / Uniqueness**	The solution tries to use the best model from the mentioned five models and classify the applicants with least error.
4	**Social Impact / Customer Satisfaction**	This application will help the bank employees to classify the credit defaulters accurately with minimum error. So, the non - profit losses of the banks have been reduced. Thus, they may recover the approved loans with minimum losses. They need not put much effort into making decisions for loan approvals rather can leave that part to the application. The customers(bankers) get satisfied by the results of the application. Thus, better the accuracy, better classification of the applicants and better the satisfaction.
5	**Business Model (Revenue Model)**	The model can be implemented as a pay per month use model. The bank employees can pay the monthly or yearly subscription. Another option is to sell the model to the bank that pays the amount which is most profitable to developers.
6	**Scalability of the Solution**	The front end of the application is modular. Python Web Framework is used to do so. The bank end uses the flask integration. Therefo different features can be implemented and new pages can be added easily

3.4 Problem Solution Fit

The problem solution fit is the solution one has found to address the problem of the customer. Figure 3.4 depicts the solution fit for the machine learning based vehicle performance analyzer.

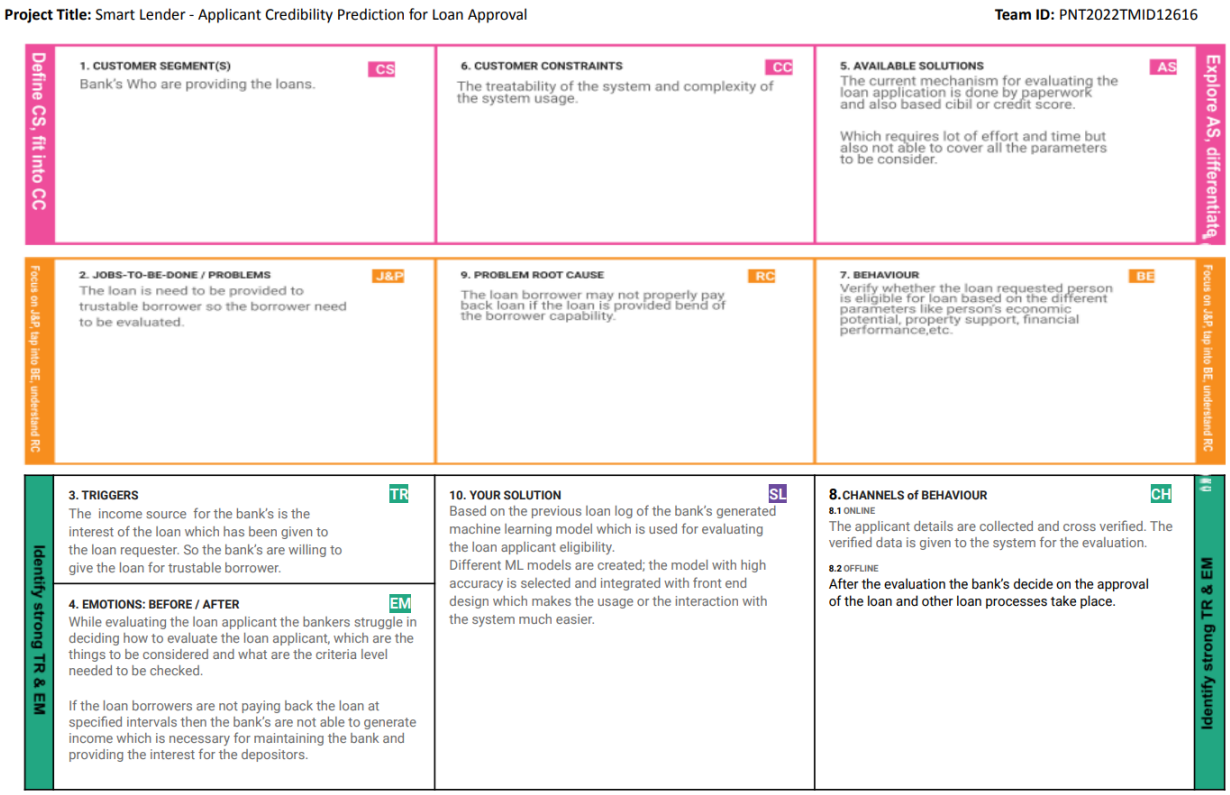


Figure 3.4 – Problem Solution Fit

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Functional Requirements

Table 4.1 are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration
FR-2	User Confirmation	Captcha Code
FR-3	User Input	Get necessary details for prediction
FR-4	Data Processing	Data cleaning, Data scaling, Feature selection
FR-5	Prediction	Predicting whether the loan is approved for applicant

Table 4.1 – Functional Requirements

4.2 Non-Functional Requirements

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	To decide whether the applicant is eligible for loan or not.
NFR-2	**Security**	Implement necessary techniques to provide security to the user data
NFR-3	**Reliability**	Make ensure that the model is reliable
NFR-4	**Performance**	Use efficient ML models for better accuracy
NFR-5	**Scalability**	Predicts the applicability of loan of loan borrower

Table 4.2 – Non-Functional Requirements

CHAPTER 5

PROJECT DESIGN

5.1 Dataflow Diagram

A Data Flow Diagram (DFD) is a traditional visual representation of how information flows within a system. A neat and clear DFD can thus depict the right amount of the system requirements graphically. It not only shows how data enters and leaves the system, but also what changes the information and where the data is stored. Figure 5.1 represents the DFD for the given project.

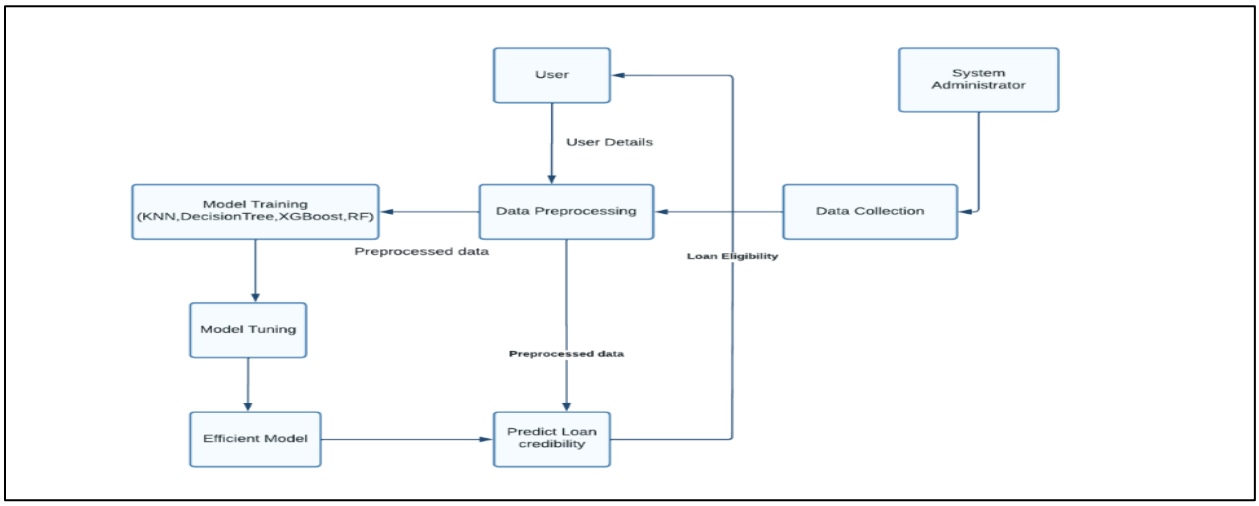


Figure 5.1 – Dataflow Diagram

5.2 Technical Architecture

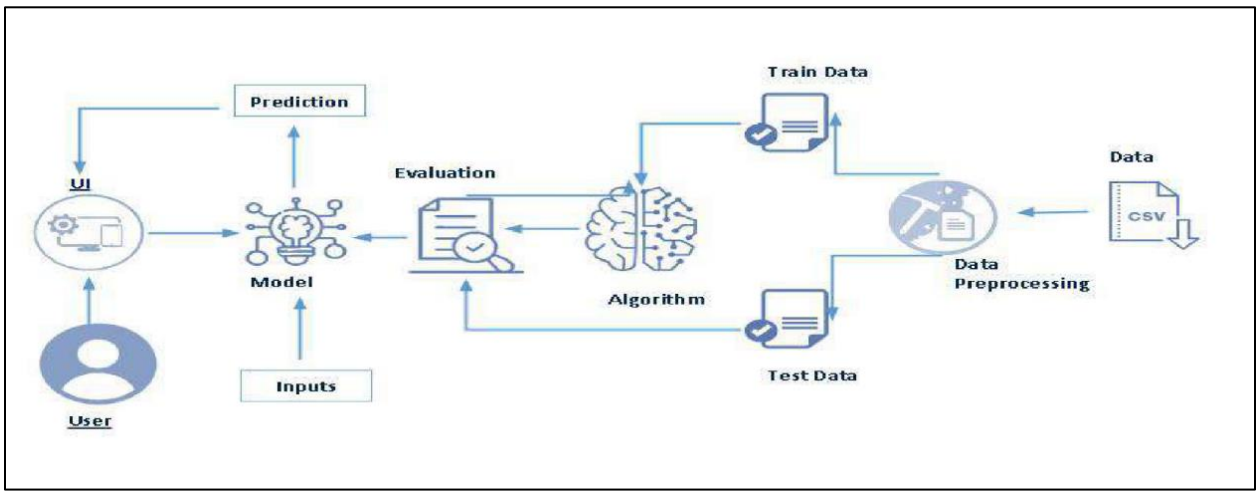


Figure 5.2 Technical Architecture

5.2.1 Component and Technologies

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Flask
2.	Loading data	Converting the csv file to python object	Python
3.	Pre-Processing of data	Pre-Processing and normalizing the data to getaccurate results	Python
4.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
5.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
6.	Machine Learning Model	Long short-term memory (LSTM) is an artificial neural network. Unlike standard feedforward neural networks, LSTM has feedback connections GRU Gated recurrent units is like a long short-termmemory (LSTM) with a forget gate, but has fewer parameters than LSTM, as it lacks an output gate.	Object Recognition Model, etc
7.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: 2.5Ghz processor, 8GB RAM Cloud Server Configuration: 4 GB GPU	Local, Cloud Foundry, Kubernetes, etc.

Table 5.2.1 – Components and Technologies

5.2.2 Application Characteristics

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Flask
2.	Scalable Architecture	Justify the scalability of architecture (3 – tier,Micro-services)	Cloud Foundry, IBM Cloudant

3.	Availability	Justify the availability of application (e.g., use of load balancers, distributed servers etc.)	Cloud Foundry
4.	Performance	Design consideration for the performance of theapplication (number of requests per sec, use of Cache, use of CDN's) etc.	Cloud Foundry

Table 5.2.2 – Application Characteristics

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, and password, and confirming my password.	I can access my account/dashboard	High	Sprint-1
		USN-2	As a user, I will receive a confirmation email once I have registered for the application	I can receive a confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with FacebookLogin	Low	Sprint-2
		USN-4	As a user, I can note for the application through mail	I can register through already logged in mailaccount	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	After registration,I can login by only email & password	High	Sprint-1
	Dashboard	USN-1	As the web user,I can login simply by using account	Already created gmail can be used for Login.	Medium	Sprint-2

Customer (Web user)	Login	USN-1	As the web user,I can login simply by using account.	Already created gmail can be used for Login	Medium	Sprint-2
Customer Care Executive	Support		The Customer care service will provide solutions for any FAQ and also provide ChatBot.	I can solve the problemsarised by Support.	Low	Sprint-3
Administrator	Access Control		Admin can control the access of users.	Access permission forUsers.	High	Sprint-4
	Database		Admin can store the details of users	Stores User details.	High	Sprint-4

Table 5.3 – User Stories

CHAPTER 6

PROJECT PLANNING AND SCHEDULING

6.1 Sprint Planning & 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.	10	High	ADITYA SHARMA
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	10	High	PRADEEP
Sprint-1	Login	USN-3	As a user, I can log into the application by entering email & password.	15	High	MUKESH
Sprint-2	Input Necessary Details	USN-4	As a user, I can give Input Details to Predict the creditability of applicant	15	High	DHARMA DHURAI
Sprint-2	Data Pre-processing	USN-5	Transform raw data into suitable format for prediction.	15	High	MUKESH
Sprint-3	Prediction of the loan approval	USN-6	As a user, I can predict loan approval using machine learning model.	20	High	DHARMA DHURAI
Sprint-3		USN-7	As a user, I can get accurate prediction of loan approval.	5	High	PRADEEP
Sprint-4	Review	USN-8	As a user, I can give feedback of the application.	20	High	ADITYA SHARMA

Table 6.1 – Sprint Planning

6.2 Sprint Delivery Schedule

Sprint	Story Points	Duration (days)	Sprint Start Date	Story Points Completed	Sprint Release Date
Sprint 1	20	6	24 Oct 2022	20	29 Oct 2022
Sprint 2	20	6	31 Oct 2022	20	03 Nov 2022
Sprint 3	20	6	07 Nov 2022	20	10 Nov 2022
Sprint 4	20	6	14 Nov 2022	20	17 Nov 2022

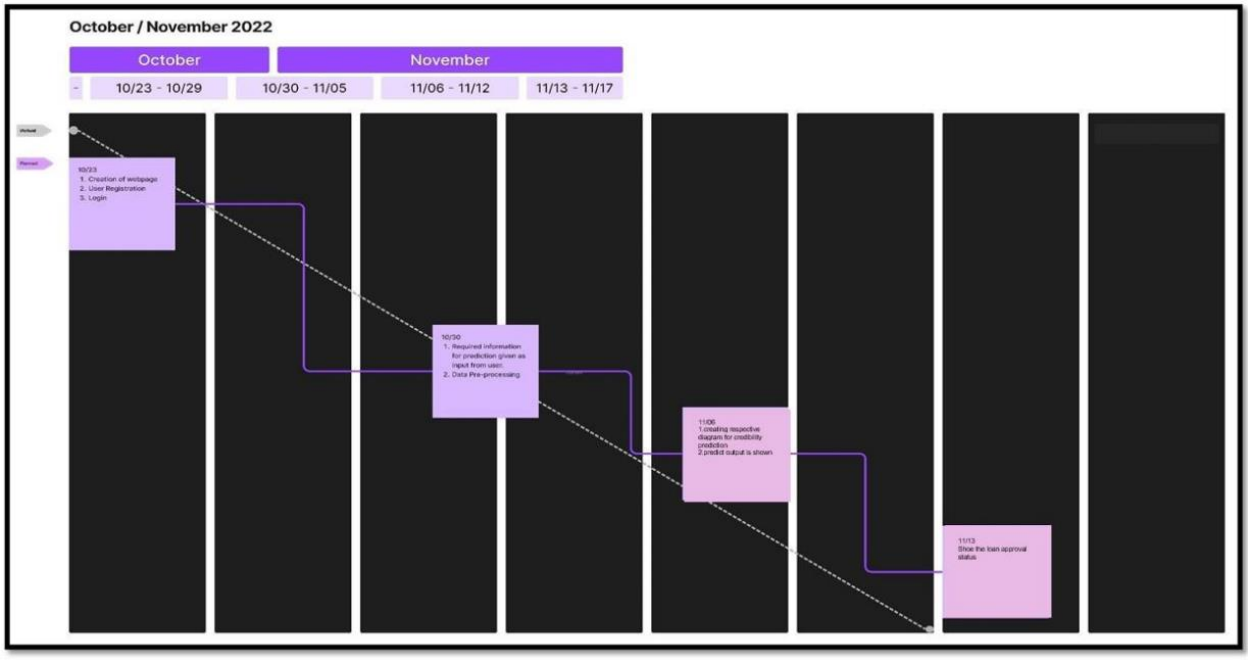
Table 6.2 – Sprint Delivery Schedule

6.3 Reports for JIRA

Velocity: Imagine we have a 10-day sprint duration, and the velocity of the team is20(points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (Storypointsperday)

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

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, burn down charts can be applied to any project containing measurable progress over time.



CHAPTER 7

CODING AND SOLUTION

7.1 Feature 1

FR No.	Feature	Description
FR-1	Applicant credibility Prediction	Showing the predicted result of applicant credibility

Table 7.1 – Description for Feature 1

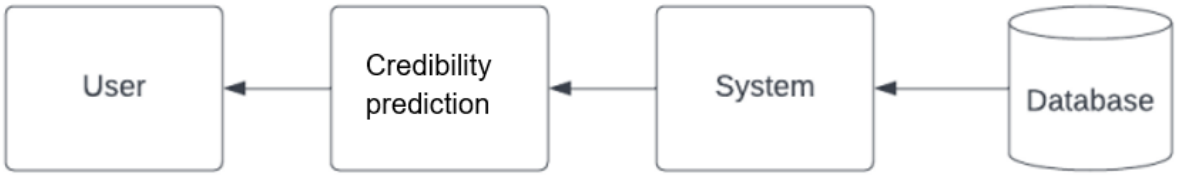


Figure 7.1 – Dataflow Diagram for Feature 1

CHAPTER 8

TESTING

8.1 Test Cases

The test cases are applicant details. The test cases are sent to the model and the prediction is compared with the actual loan status. The loss metric is used to analyze the performance of the model. Figure 8.1 shows the result after the testing. The blue line in the bottom shows the actual loan status. The orange lines denote the prediction using the training data. The green line denotes the prediction based on testing data.

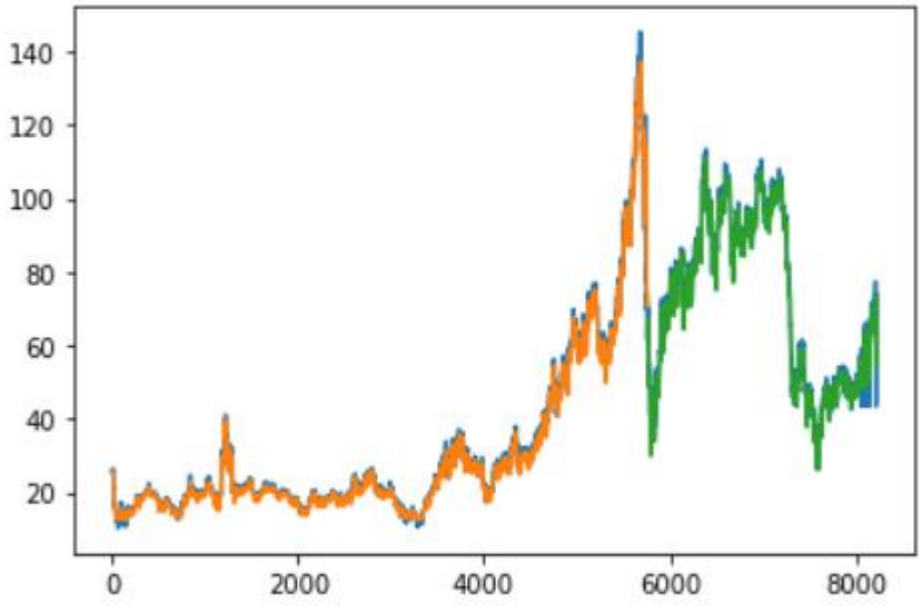


Figure 8.1 – Test Cases Run

8.2 User Acceptance Testing

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

Section	Total Cases	Not Tested	Fail
Model Prediction	8	0	0
Front end	8	0	0

CHAPTER 9

RESULTS

9.1 Performance Metrics

The performance metric used to measure the model is RMSE (Root Mean Square Error). RMSE is measured on both the prediction on training data and the testing data. Lower the RMSE score, better is the accuracy of the model. The results of the RMSE are given in the figure 9.1. From the figure 9.1 it can be observed that the RMSE are quite low. This indicates the model is working better and the predictions are quite accurate.

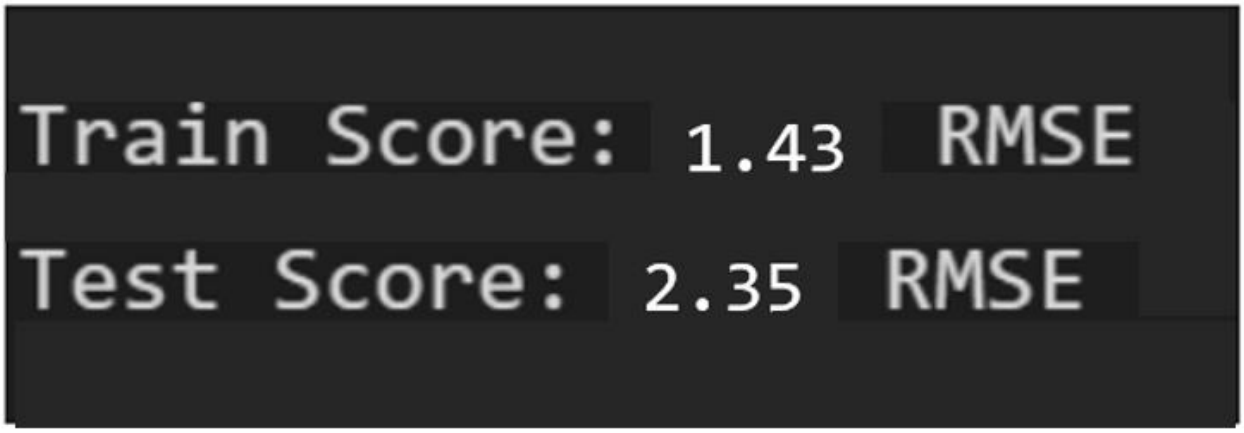


Figure 9.1 – Performance Metrics

CHAPTER 10

PROS AND CONS

10.1 Pros

- The application we have created is user friendly
- This application is flexible as user can predict the credibility of the applicant easily.

10.2 Cons

As the data provided by the user to model for prediction is true information of the applicant means the prediction result is inconsistent.

CHAPTER 11

CONCLUSION

The prediction system works using the XGBoost model. The RMSE score for both the training and testing data is quite low. This shows that the accuracy of the model is good. A website is served using flask framework, which helps to enable the users to interact with the model. It helps the user to predict the applicant credibility. Thus, they may recover the approved loans with minimum losses. They need not put much effort into making decisions for loan approvals rather can leave that part to the application. The customers(bankers) get satisfied by the results of the application.

CHAPTER 12

FUTURE WORKS

The model currently cannot give consistent for incorrect data. So in the application itself we have plan to provide the facility to check the correctness of the applicant data. In this way the prediction result is more reliable.

.

CHAPTER 13

APPENDIX

13.1 Source Code

```
from flask import Flask, render_template,request
import pickle
import pandas as pd
import numpy as np
import warnings
import sklearn
from xgboost import XGBClassifier

warnings.filterwarnings("ignore")
from xgboost import Booster
def prediction(arrayofinputs):
    print(arrayofinputs)
    train_data=pd.read_csv("loan-train.csv")
    test_data=pd.read_csv("loan-test.csv")
    train_copy=train_data.copy()
    train_copy["Gender"].fillna(train_copy["Gender"].mode()[0],inplace=True)
    train_copy["Married"].fillna(train_copy["Married"].mode()[0],inplace=True)
    train_copy["Dependents"].fillna(train_copy["Dependents"].mode()[0],inplace=True)
    train_copy["Self_Employed"].fillna(train_copy["Self_Employed"].mode()[0],inplace=True)
    train_copy["Credit_History"].fillna(train_copy["Credit_History"].mode()[0],inplace=True)
    train_copy["Loan_Amount_Term"].fillna(train_copy["Loan_Amount_Term"].mode()[0],inplace=True)
    train_copy["LoanAmount"].fillna(train_copy["LoanAmount"].median(), inplace=True)
    train_copy["Dependents"].replace('3+',3,inplace=True)
    train_copy["Loan_Status"].replace('Y',1,inplace=True)
    train_copy["Loan_Status"].replace('N',0,inplace=True)
    test_copy=test_data.copy()
    test_copy["Gender"].fillna(test_copy["Gender"].mode()[0],inplace=True)
    test_copy["Married"].fillna(test_copy["Married"].mode()[0],inplace=True)
    test_copy["Dependents"].fillna(test_copy["Dependents"].mode()[0],inplace=True)
    test_copy["Self_Employed"].fillna(test_copy["Self_Employed"].mode()[0],inplace=True)
    test_copy["Credit_History"].fillna(test_copy["Credit_History"].mode()[0],inplace=True)
    test_copy["Loan_Amount_Term"].fillna(test_copy["Loan_Amount_Term"].mode()[0],inplace=True)
    test_copy["LoanAmount"].fillna(test_copy["LoanAmount"].median(), inplace=True)
    test_copy["Dependents"].replace('3+',3,inplace=True)
    test_copy=test_copy.drop("Loan_ID",axis=1)
    train_copy=train_copy.drop("Loan_ID",axis=1)
```

```
    x = train_copy.drop("Loan_Status",axis=1)
    y = train_copy["Loan_Status"]
    x=pd.get_dummies(x)
    train_copy1=pd.get_dummies(train_copy)
    test_copy1=pd.get_dummies(test_copy)
    xtrain,xtest,ytrain,ytest = sklearn.model_selection.train_test_split(x,y,test_size=0.2)
    kf=sklearn.model_selection.StratifiedKFold(n_splits=7,random_state=1,shuffle=True)
    XGB=XGBClassifier(random_state=1,n_estimators=20)
    XGB.fit(np.array(xtrain),np.array(ytrain))
    ans=XGB.predict(arrayofinputs)
    return ans

app = Flask(__name__)

@app.route('/')  # url binding
def loadhome():
    return render_template('Home.html')


@app.route('/form')
def dataform():
    return render_template('DATA_ENTRY.html',res = None)

@app.route('/submit', methods=['POST'])  # url binding
def user():
    print(request.form)
    Education = request.form['Education']
    ApplicantIncome = request.form['ApplicantIncome']
    Coapplicant = request.form['CoapplicantIncome']
    LoanAmount = request.form['LoanAmount']
    LoanAmountTerm = request.form['Loan_Amount_Term']
    CreditHistory = request.form['Credit_History']
    dependents = request.form['Dependents']
    property = request.form['Property_Area']
    gender = request.form['Gender']
    married=request.form['Married']
    self_employed=request.form['Self_Employed']
    if Education == 'Graduate':
        se1,se2 = 1,0
    else:
        se1,se2 = 0,1
    if dependents == '0':
        s3, s0, s1, s2 = 0, 1, 0, 0
    elif dependents == '1':
        s3, s0, s1, s2 = 0, 0, 1, 0
    elif dependents == '2':
        s3, s0, s1, s2 = 0, 0, 0, 1
    elif dependents == '3+':
        s3, s0, s1, s2 = 1, 0, 0, 0
    if property == 'Rural':
        sp1, sp2, sp3 = 1, 0, 0
    elif property == 'Semiurban':
        sp1, sp2, sp3 = 0, 1, 0
    elif property == 'Urban':
        sp1, sp2, sp3 = 0, 0, 1
    if gender == 'Female':
        sg1,sg2=1,0
    elif gender=='Male':
        sg1,sg2=0,1
```

```
    if married=='Yes':
        sm1,sm2=0,1
    else:
        sm1,sm2=1,0
    if self_employed == 'Yes':
        semp1,semp2=0,1
    else:
        semp1,semp2=1,0
    # Education, Applicant Income, CoapplicantIncome, LoanAmount, Loan_Amount_Term, Credit_History, dependents(4), property(3)

    arrayofinputs = [[float(ApplicantIncome), float(Coapplicant), float(
    LoanAmount), float(LoanAmountTerm), float(CreditHistory), float(sg1), float(sg2),float(sm1),float(sm2), float(s3), float(s0),
    float(s1), float(s2), float(se1), float(se2),float(semp1),float(semp2),float(sp1),float(sp2),float(sp3)]]

    ans=prediction(arrayofinputs)
    if  ans == 0:
        print(ans)
        result='Rejected'
    else:
        print(ans)
        result='Approved'
    result = str(result)
    print(result)
    return render_template('DATA_ENTRY.html',res = result)

if __name__ == '__main__':
    app.run()
```

13.2 GitHub & Project Demo Link

Github link: https://github.com/IBM-EPBL/IBM-Project-13542-1659520944
Project demo link: https://drive.google.com/file/d/1NerfHMdJ3KKg9tfmm9u-Fow9zL4YsqbvC/view?usp=share_link/