

## **CHAPTER - 1**

### **INTRODUCTION**

#### **1.1 Project Overview :**

Loan sanctioning and credit scoring forms a multi-billion dollar industry -- in the US alone. With everyone from young students, entrepreneurs, and multi-million dollar companies turning to banks to seek financial support for their ventures, processing these applications creates a complex and cumbersome task for any banking institution. As of 2022, more than 20 million people in the US have active loans owing a collective debt of 178 billion dollars. Despite that, more than 20% of all applicants were denied loans. The loan approval or rejection has enormous ramifications for both the applicant and the bank, causing possible opportunity costs for both parties. Banks like Wells Fargo and Morgan Stanley have looked at the use of AI in determining lending risk and developing a loan prediction system in recent years to overcome human bias and delays in the application processing time.

#### **1.2 Purpose :**

To deal with the problem, we developed automatic loan prediction using machine learning techniques. We will train the machine with previous dataset. so machine can analyse and understand the process . Then machine will check for eligible applicant and give us result.

## **CHAPTER - 2**

### **LITERATURE SURVEY**

#### **2.1 Existing problem :**

Bank employees check the details of applicant manually and give the loan to eligible applicant. Checking the details of all applicants takes lot of time. The artificial neural network model for predict the credit risk of a bank. The Feed- forward back propagation neural network is used to forecast the credit default. The method in which two or more classifiers are combined together to produce a ensemble model for the better prediction. They used the bagging and boosting techniques and then used random

forest technique. The process of classifiers is to improve the performance of the data and it gives better efficiency. In this work, the authors describe various ensemble techniques for binary classification and also for multi class classification. The new technique that is described by the authors for ensemble is COB which gives effective performance of classification but it also compromised with noise and outlier data of classification. Finally they concluded that the ensemble based algorithm improves the results for training data set.

## **2.2 References :**

- 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".
- Sivasree M S, Rekha Sunny T, (2015), "Loan Credibility Prediction System Based on Decision Tree Algorithm", No "September" / "2015".
- Anuja Kadam, Pragati Namde, Sonal Shirke, Siddhesh Nandgaonkar, Dr.D.R Ingle, 2021, "Loan Credibility Prediction System using Data Mining Techniques" No "May" / "2021".
- Pidikiti Supriya , Myneedi Pavani , Nagarapu Saisushma , Namburi Vimala Kumari , K Vikas, 2019, "Loan Prediction by using Machine Learning Models", No "April" / "2019".
- <https://medium.com/swlh/lending-club-data-web-app-ada56ff64cee>
- <https://github.com/smartinternz02/SI-GuidedProject-48927-1652694502>
- [7https://www.academia.edu/77162007/BANK\\_LOAN\\_PREDICTION\\_USING MACHINE\\_LEARNING](https://www.academia.edu/77162007/BANK_LOAN_PREDICTION_USING_MACHINE_LEARNING)

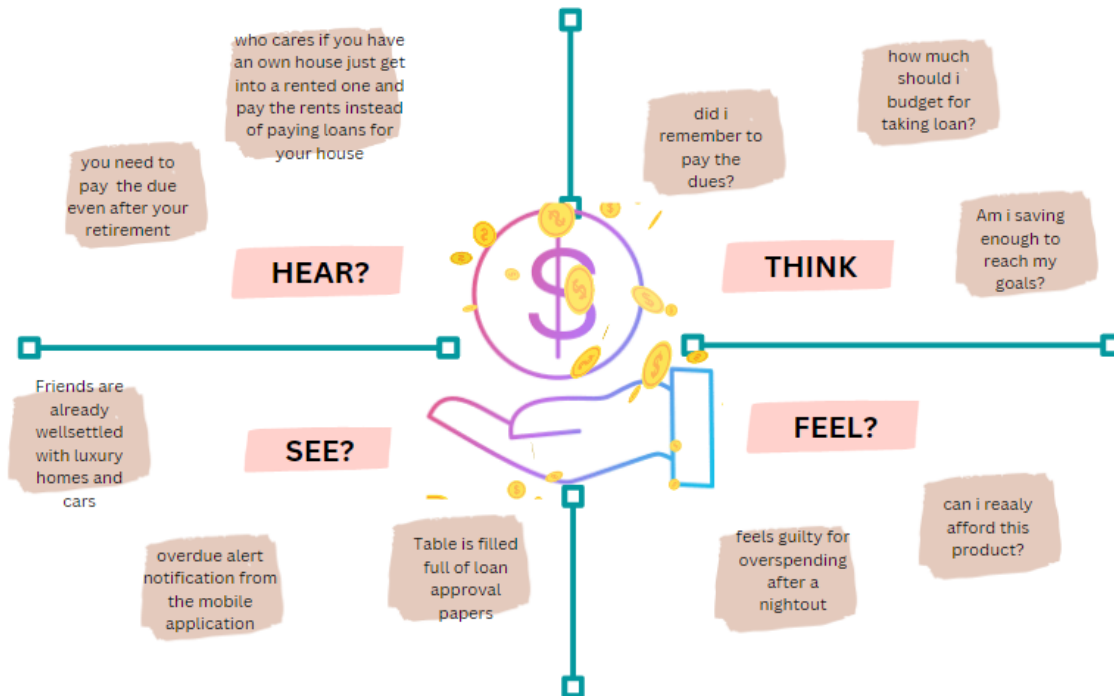
## **2.3 Problem statement definition :**

- Tracking or checking the status is difficult.
- Prone to human errors.
- Time consumption is high.
- Lot of paper works.
- Poor customer service due to lack of manpower

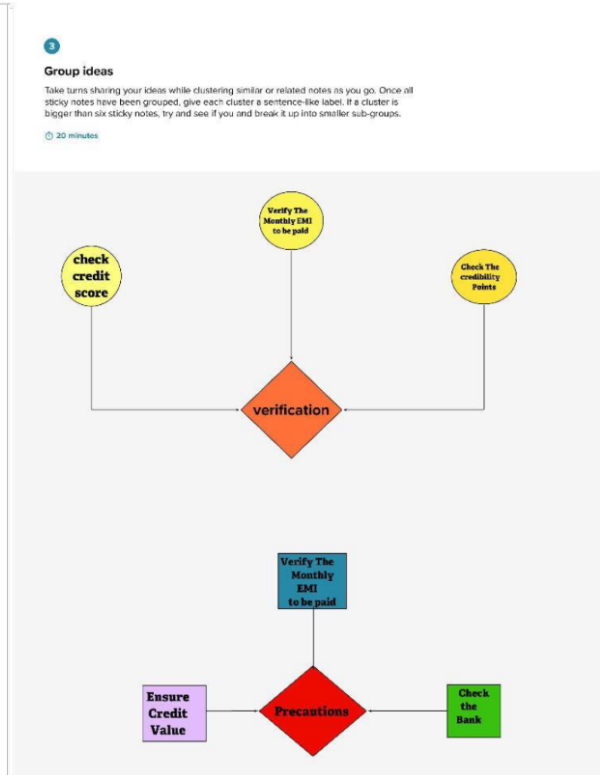
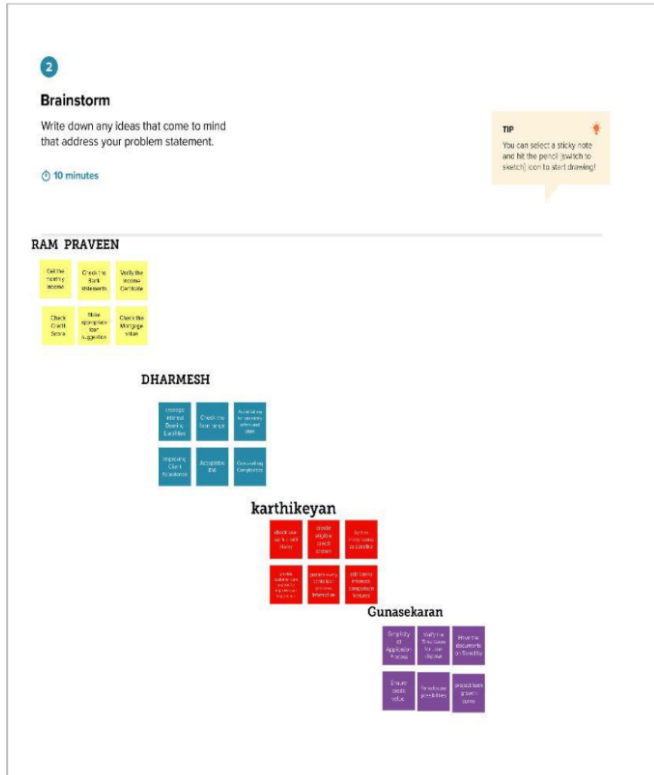
## CHAPTER - 3

### IDEATION AND PROPOSED SOLUTION

#### 3.1 Empathy Map Canvas :



#### 3.2 Ideation and Brainstorming :



### 3.3 Proposed Solution :

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	<ul style="list-style-type: none"> <li>• Tracking or checking the status is difficult.</li> <li>• Prone to human errors.</li> <li>• Time consumption is high.</li> <li>• Lot of paper works.</li> <li>• Poor customer service due to lack of manpower.</li> </ul>
2.	Idea / Solution description	<ul style="list-style-type: none"> <li>• Tracking or checking the status becomes easy.</li> <li>• Reduce the potential for human error.</li> <li>• Time consumption of the process will be reduced.</li> <li>• Reduces the paperwork to paperless.</li> <li>• Improve the effectiveness of customer service teams.</li> <li>• Fair eligibility prediction.</li> <li>• Highly scalable and provide data driven decisions to stakeholder and higher authority.</li> </ul> <p>We will be using classification algorithms such as Decision tree, Random Forest, KNN, and xgboost to achieve higher accuracy in predicting the model. We will train and test the data with these algorithms, tune by hyperparameter tuning. From this the above ideas are implemented.</p>
3.	Novelty / Uniqueness	As soon as the essential data are provided, the model will predict whether to approve the loan or not - By use of transfer learning.
4.	Social Impact / Customer Satisfaction	One of the most important factors which affect our country's economy and financial condition is the credit system governed by the banks. As we know credit risk evaluation is very crucial, there is a variety of techniques are used for risk level calculation. In addition, credit risk is one of the main functions of the banking community.
5.	Business Model (Revenue Model)	This model can be developed by minimum cost at the same time it will provide the peak performance, higher accuracy and the result will be more effective than traditional techniques.
6.	Scalability of the Solution	Banks need not to go through the background verification process of the applicant by using this model. The model will predict the customers data and their attributes like salary , credit score,etc.

### 3.4 Problem Solution Fit :

Define CS, fit into CC	<b>1. CUSTOMER SEGMENT(S)</b> Bank's Who are providing the loans. <b>CS</b>	<b>6. CUSTOMER CONSTRAINTS</b> The treatability of the system and complexity of the system usage. <b>CC</b>	<b>5. AVAILABLE SOLUTIONS</b> The current mechanism for evaluating the loan application is done by paperwork and also based cibil or credit score. <b>AS</b>  Which requires lot of effort and time but also not able to cover all the parameters to be consider.	Explore AS, differentiating
	<b>2. JOBS-TO-BE-DONE / PROBLEMS</b> The loan is need to be provided to trustable borrower so the borrower need to be evaluated. <b>J&amp;P</b>	<b>9. PROBLEM ROOT CAUSE</b> The loan borrower may not properly pay back loan if the loan is provided bend of the borrower capability. <b>RC</b>	<b>7. BEHAVIOUR</b> 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. <b>BE</b>	

## CHAPTER - 4 REQUIREMENT ANALYSIS

### Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Bank Website Registration through Gmail Registration through mobile Application
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Loan type	Personal Loan Education Loan
FR-4	User Details	Name, Address, Income, Occupation.
FR-5	Assets Proof	Agricultural land, Gold
FR-6	Verification	Verification of user Details which are provided above

**Non-functional Requirements:**

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Easy to access
NFR-2	Security	User proofs
NFR-3	Reliability	Based on the customer Income
NFR-4	Performance	Previous history of the user bank account
NFR-5	Availability	Based on the customer Address
NFR-6	Scalability	Based on the customer Assets proofs

## **CHAPTER - 5**

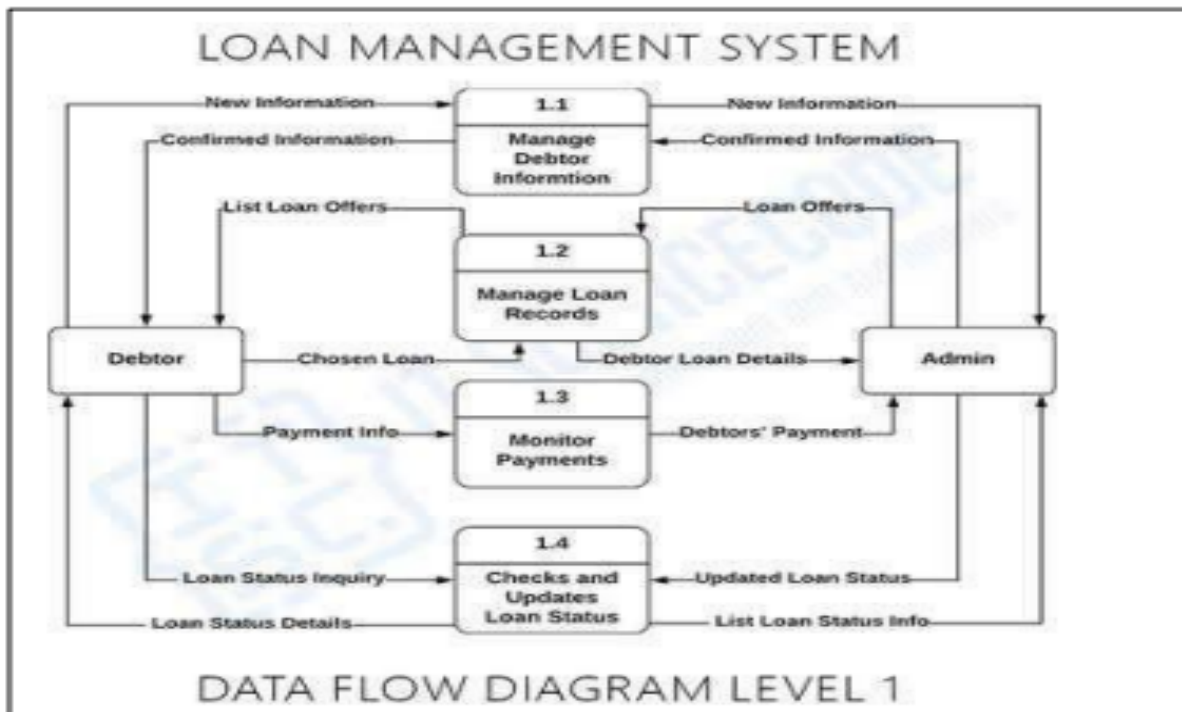
### **PROJECT DESIGN**

#### **5.1 Data Flow Diagrams :**

## Flow: (Simplified)

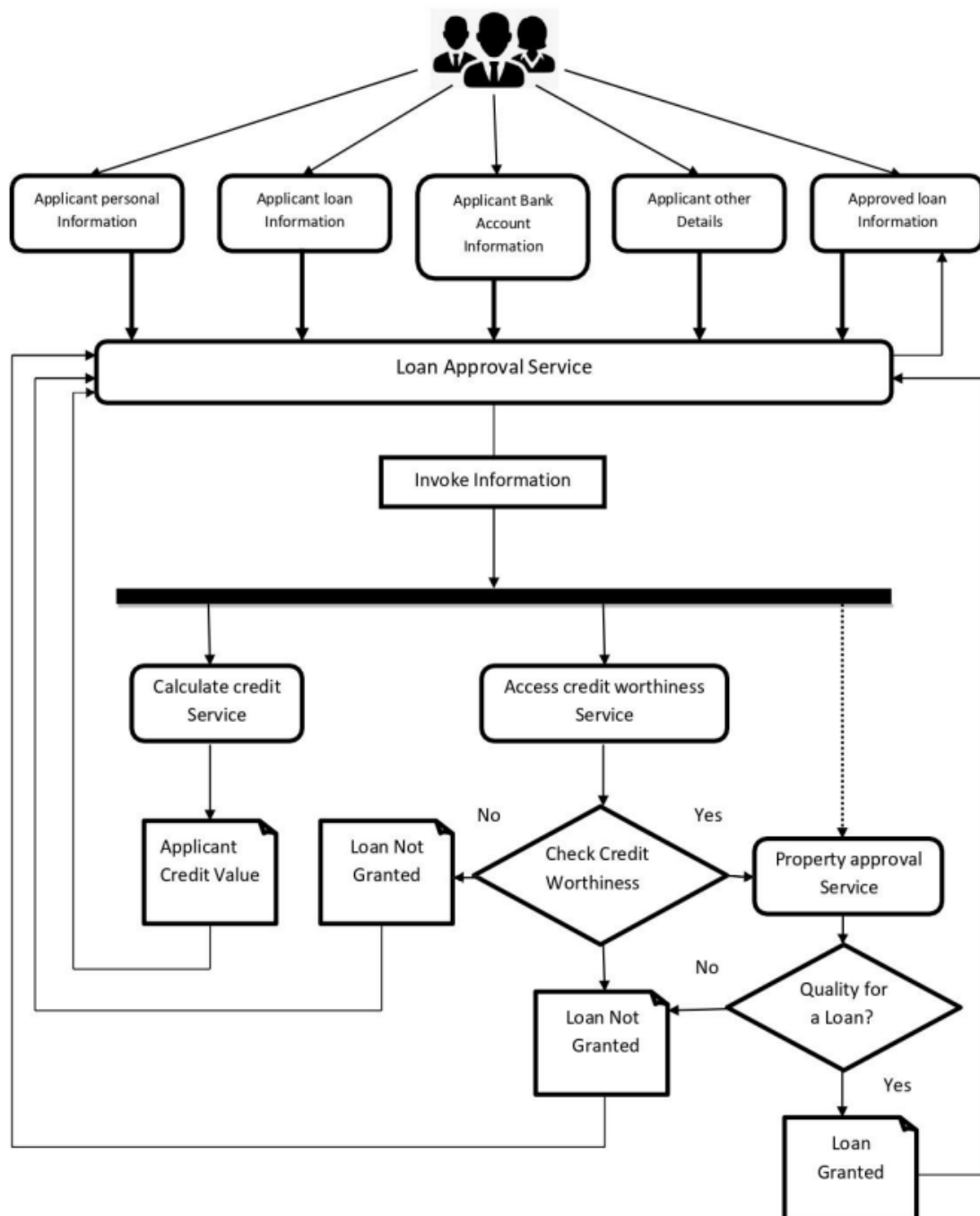


- The user can register in website by using Email and Mobile number.
- The user can Login by using Email and password as Registered in the respective website.
- The user will provide personal and financial details.
- User should upload the scanned documents.
- Then it will goes to approval process.
- Finally they will get loan closure certificate.





## 5.2 Solution and 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 loan application by entering my email/user number, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the loan application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the loan application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can receive the mail that you are registered in loan application.	Medium	Sprint-1

## **CHAPTER - 6**

### **PROJECT PLANNING AND SCHEDULING**

#### 6.1 Sprint Planning and Estimation :

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	3	High	Prathy P Durai M Vadivel M Afridi S
Sprint-1	Dashboard	USN-6	As a user, I should be able to access the dashboard with everything I am allowed to use.	2	Medium	Prathy P Durai M Vadivel M Afridi S
Sprint-1	Registration	USN-7	As a user, I can register for the application by entering my email, password, and confirming my password.	3	High	Prathy P Durai M Vadivel M Afridi S

## 6.2 Sprint Delivery Schedule :

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2	Register	USN-13	As a loan approval officer, I should be able to register myself as one using a unique email and password.	5	Medium	Prathy P Durai M Vadivel M Afridi S
Sprint-2	Login	USN-14	As a loan approval officer I should be able to login myself as one using a unique email and password.	5	Medium	Prathy P Durai M Vadivel M Afridi S
Sprint-3	Automated analysis of credit history	USN-15	As a loan approval officer, I can access the dashboard where I feed applications for loan prediction.	10	High	Prathy P Durai M Vadivel M Afridi S
Sprint-3		USN-16	As a loan approval officer, I can get a decision followed by some details for the decision when I feed an application for loan prediction.	15	High	Prathy P Durai M Vadivel M Afridi S
Sprint-4	Register	USN-17	As an admin, I should be able to register myself as one using a unique email and password.	2	Medium	Prathy P Durai M Vadivel M Afridi S
Sprint-4	Login	USN-18	As an admin I should be able to login myself as one using a unique email and password.	2	Medium	Prathy P Durai M Vadivel M Afridi S

## 6.3 Reports from JIRA :

**Project Tracker, Velocity & Burndown Chart: (4 Marks)**

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	28	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	10	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	25	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	6	19 Nov 2022

## CHAPTER - 7

### CODING AND SOLUTIONING

#### 7.1 Feature 1

```
body{
    margin:100px;
}

.form-label{
    color:rgb(235, 27, 97);
    font-weight: bolder;
    font-size: 18px;
}

.Heading{
    color:rgb(235, 27, 97);
    font-size: 30px;
    font-weight:bolder;
    padding: 10px;
    margin: 20px;
    font-family: 'Times New Roman', Times, serif;
    width: 100%;
}

.fill{
    color: black;
    font-size: 20px;
    font-weight:15px;
    font-family: 'Times New Roman', Times, serif;
}

#but{
    text-align: center;
    align-items: center;
    justify-content: center;
}

.mb-3{
    width:85%;
    padding-left:15%;
}

.form-control{
    margin:10px 2px;
    outline: solid 1px lightblue;
}
```

```

.form-select{
    margin:10px 2px;
    outline: solid 1px lightblue;
}
.but{
    background-color:rgb(235, 27, 97);
    color:white;
    border-radius: 10px;
    width:100px;
    border-color:white ;
    height:50px;
    padding: 10px;
}

#back{
    background-color:rgb(235, 27, 97);
    color:white;
    border-radius: 10px;
    width:100px;

}

```

## 7.2 Feature 2

```

@import
url('https://fonts.googleapis.com/css?family=Poppins:400,500,600,700&display=swap');
*{
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    font-family: 'Poppins', sans-serif;
}
html,body{
    display: grid;
    height: 100%;
    place-items: center;
    text-align: center;
    background:white;
}
.container{
    position: relative;
    width: 400px;

```

```
background:black;
padding: 20px 30px;
border: 1px solid #444;
border-radius: 5px;
display: flex;
align-items: center;
justify-content: center;
flex-direction: column;
}
.container .post{
display: none;
}
.container .text{
font-size: 25px;
color: #666;
font-weight: 500;
}
.container .edit{
position: absolute;
right: 10px;
top: 5px;
font-size: 16px;
color: #666;
font-weight: 500;
cursor: pointer;
}
.container .edit:hover{
text-decoration: underline;
}
.container .star-widget input{
display: none;
}
.star-widget label{
font-size: 40px;
color: #444;
padding: 10px;
float: right;
transition: all 0.2s ease;
}
input:not(:checked) ~ label:hover,
input:not(:checked) ~ label:hover ~ label{
color: #fd4;
}
input:checked ~ label{
color: #fd4;
}
```

```
input#rate-5:checked ~ label{
  color: #fe7;
  text-shadow: 0 0 20px #952;
}
#rate-1:checked ~ form header:before{
  content: "I just hate the service";
}
#rate-2:checked ~ form header:before{
  content: "I don't like the service ";
}
#rate-3:checked ~ form header:before{
  content: "Awesome service";
}
#rate-4:checked ~ form header:before{
  content: "Satisfied service";
}
#rate-5:checked ~ form header:before{
  content: "Excellent service ";
}
.container form{
  display: none;
}
input:checked ~ form{
  display: block;
}
form header{
  width: 100%;
  font-size: 25px;
  color: #fe7;
  font-weight: 500;
  margin: 5px 0 20px 0;
  text-align: center;
  transition: all 0.2s ease;
}
form .textarea{
  height: 100px;
  width: 100%;
  overflow: hidden;
}
form .textarea textarea{
  height: 100%;
  width: 100%;
  outline: none;
  color: #eee;
  border: 1px solid #333;
  background: #222;
```

```
padding: 10px;
font-size: 17px;
resize: none;
}
.textarea textarea:focus{
border-color: #444;
}
form .btn{
height: 45px;
width: 100%;
margin: 15px 0;
}
form .btn button{
height: 100%;
width: 100%;
border: 1px solid #444;
outline: none;
background: #222;
color: #999;
font-size: 17px;
font-weight: 500;
text-transform: uppercase;
cursor: pointer;
transition: all 0.3s ease;
}
form .btn button:hover{
background: #1b1b1b;
}
h1{
background:lightpink;
width:1500px;
margin:none;
color: rgb(99, 22, 23);
padding-top: 20px;
height:100px;
margin-top:none;
border: solid 5px rgb(99, 22, 23);
}
h2{
color:rgb(255, 11, 11);
font-size: 40px;
font-family: 'Times New Roman', Times, serif;
}
h3{
font-size:30px;
color:rgb(246, 187, 11);
```



}

## **CHAPTER - 8**

### **TESTING**

#### **8.1 Test Cases :**

- Check the working condition of **Temperature** sensor
- Verify the working of **Humidity** sensor
- Check the working condition of **soil moisture** sensor
- Check the **Performance of the app** on the different internet networks
- Check whether the application is working for **real-time updates**
- Check **response** when a Login Button is pressed

#### **8.2 User Acceptance Testing :**

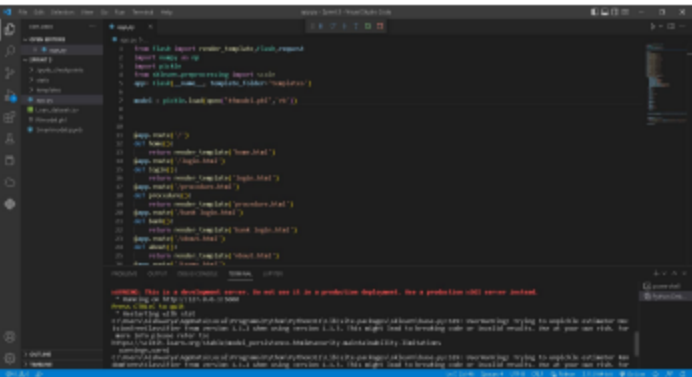
User acceptance testing, also called end-user, user acceptability testing, or beta testing, is the process of testing software by the clients or users to see if the product is acceptable for release or not. The testers are familiar with the software's business requirements, so they can adequately gauge the product's readiness.

## **CHAPTER - 9**

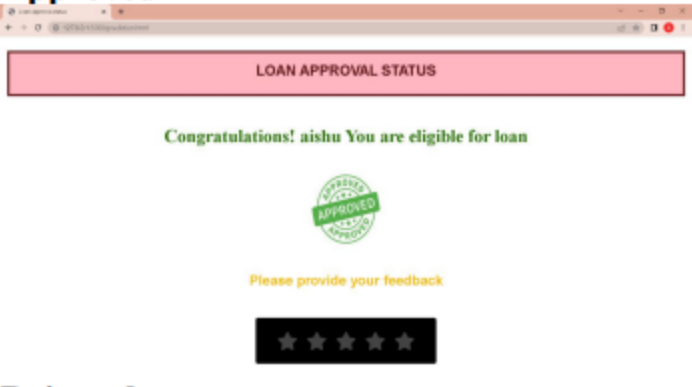
### **RESULTS**

#### **9.1 Performance Metrics :**

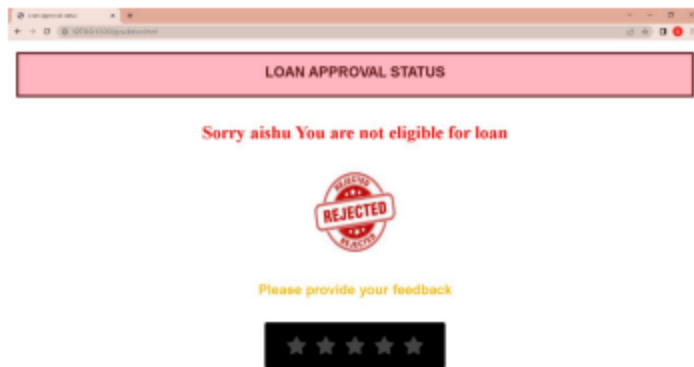
Run the application:



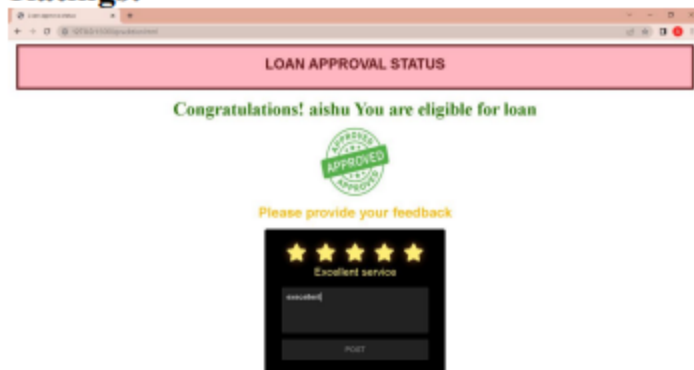
Predicted result:  
Approved



Rejection:



## Ratings:



## CHAPTER - 10 ADVANTAGES AND DISADVANTAGES

### Advantages :

The Loan Prediction System can automatically calculate the weight of each feature taking part in loan processing and on new test data the same features are processed with respect to their associated weight .

## **Disadvantages :**

The disadvantage of this model is that it emphasize different weights to each factor but in real life sometime loan can be approved on the basis of single strong factor only, which is not possible through this system.

## **CHAPTER - 11** **CONCLUSION**

Machine learning algorithms play a significant role in predicting the risks of bank loans and decision support systems. The choice of the algorithm used to make the decision(whether the borrower will default), which is the key to addressing decision management when issuing a loan. In this paper, the performance of machine learning algorithms has been tested and their performance compared to standard measurements used on a dataset that includes 1000 loans and their repayment status. Finally, the results showed the possibility of using the proposed algorithms for this purpose with acceptable accuracy rates and superiority of the neural networks for this purpose.

## **CHAPTER - 12** **FUTURE SCOPE**

The system is trained on old training dataset so future software can be made such that new testing data should also take part in training data after some fixed time.

## **CHAPTER - 13** **APPENDIX**

```
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("Rfmodel.pkl", 'rb'))

@app.route('/')
def home():
    return render_template('home.html')
@app.route('/login.html')
def login():
    return render_template('login.html')
@app.route('/procedure.html')
def procedure():
    return render_template('procedure.html')
@app.route('/bank_login.html')
def bank():
    return render_template('bank_login.html')
@app.route('/About.html')
def about():
    return render_template('About.html')
@app.route('/terms.html')
def terms():
    return render_template('terms.html')
@app.route('/register.html')
def register():
    return render_template('register.html')
@app.route('/contact.html')
def contact():
    return render_template('contact.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('/rating.html')
def rat():
    return render_template('rating.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,CoapplicantI
ncome,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 Link:

<https://github.com/IBM-EPBL/IBM-Project-19332-1659696311>

#### Project Demo Link :

<https://we.tl/t-hDTq7wJErB?src=dnI>