SMART LENDER - APPLICANT CREDIBILITY PREDICTION FOR LOAN APPROVAL

TEAM MEMBERS:

Siddarth M
VarunKrishna G
Jerfin Nishanth S
Sarah Deepthi Sahaya Kingsley

1. INTRODUCTION

1.1 Project Overview:

The primary source of revenue for the banking industry and source of financial risk for banks is a loan. The interest collected on loans disbursed directly accounts for significant amounts of a bank's assets. The ability of the borrower to repay the loan within the allotted period is one of the major hazards associated with the lending of loans. It's known as credit risk. A candidate's credit score served as the basis for determining whether or not to approve a loan on them. Thus, the purpose of this study is to describe the use of various machine learning approaches that effectively identify who to lend money to and assist banks in identifying loan defaulters for significantly lower credit risk.

The fact that our banking system offers a wide range of goods, a bank's credit line is its primary source of income. As a result, they are able to profit from interest on the loans they credit. Lenders always seek to lower their credit risk since commercial loans have historically represented a significant portion of the banking sector. The function that banks play in the modern market economy is substantial. Loans, or whether clients repay or don't return them, significantly impact a bank's profitability. Before granting loans to borrowers, banks must determine if they are good (nondefaulters) or bad (defaulters). The credit worthiness of the borrowers is one of the most critical issues in commercial loan financing.

The probability that borrowers may default on their loan commitments is referred to as credit risk. For any bank or institution, determining whether a borrower will be good or bad is a very difficult process. The banking system employs a manual 4 procedure to determine whether or not a borrower has defaulted. The manual method will undoubtedly be more precise and efficient, but it will not be able to handle a high volume of loan applications at once. When a situation like this arises, it will take a very long time to make decisions and a lot of labour will be needed.

1.2 Purpose:

Loans account for a large portion of bank profits. For financial companies, the loan approval process is crucial. Because loan defaults are occurring more frequently and it is becoming more challenging for banking authorities to properly access loan requests and address the dangers of people defaulting on loans, it is very difficult to forecast if clients will be able to pay back the loan. Numerous scholars have been focusing on loan approval system prediction in recent years. For vast amounts of data, the machine learning technique is highly helpful in predicting outcomes. Four algorithms, including Random Forest, Decision Tree, Naive Bayes, and Logistic Regression, are employed in this study to forecast whether or not clients would be approved for loans.The same information will be analysed for all four methods, and the most accurate algorithm will be chosen to deploy the model. From this point forward, we create a machine learningbased bank loan prediction system that chooses the qualified applicants for loan approval on its own

2. LITERATURE SURVEY

2.1 Existing problem:

Existing frameworks are frequently broken. Computations can be undeniably challenging, particularly if a significant number of the qualities are muddled and/or a considerable lot of the outcomes are connected. In the future, this prediction module can be more improved and integrated. The system is prepared on the previous training data but in the future, it is possible to make changes to software, which can accept new testing data and should also take part in training data and predict accordingly.

2.2 References:

1. PAPER NAME: Loan Approval Prediction using Machine Learning Models
AUTHORS: Ritika Purswani, Sakshi Verma, Yash Jaiswal, Prof. Surekha
DESCRIPTION: Outlier detection and removal, as well as imputation removal processing, were done during the pre-processing stage. To predict the chances of current status regarding the loan approval process, SVM, DT, KNN, and gradient boosting models were used in this method. To divide the dataset into training and testing processes, the 80:20 rule was used. Experimentation concluded that the Decision Tree has significantly higher loan prediction accuracy than the other models.
MODULES: Data collection and pre-processing, applying machine learning models, training, and testing the data.

2. PAPER NAME: Accurate loan approval prediction based on machine learning approach.
AUTHORS: J. Tejaswini1, T. Mohana Kavya, R. Devi Naga Ramya, P. Sai Triveni Venkata Rao Maddumala.
DESCRIPTION: The framework acknowledges or won't have any significant bearing for a loan. Obligation reimbursement is a critical mark of bank funds. It is extremely challenging to foresee the capacity of clients to reimburse a loan. AI methods are extremely valuable in uncovering the aftereffects of many sources. This paper utilizes three AI calculations: Logistic Regression (LR), Decision Tree (DT) and Random Forest (RF) to distinguish client advances. Studies have shown that the respectability of AI calculations is more noteworthy than the backwardness and technique of learning AI.
MODULES: Logistic Regression , Decision Tree and Random Forest.

3.PAPER NAME: Bank Loan Prediction System using Machine Learning.
AUTHORS: Anshika Gupta , Vinay Pant , Sudhanshu Kumar and Pravesh Kumar Bansal.
DESCRIPTION: In this paper, they use a machine learning technique that will predict the person who is reliable for a loan, based on the previous record of the person whom the loan amount is accredited before. This work's primary objective is to predict whether the loan approval to a specific individual is safe or not. . Fraud detection and credit risk applications are particularly well suited to classification technique. This approach frequently employs Decision tree based classification Algorithm. In classification, a training set is used to build the model as the classifier which can classify the data items into its appropriate classes. A testset is used to validate the model.
MODULES: Loan Dataset, Logistic Regression, Random Forest, Django

4.PAPER NAME: Loan Sanctioning Prediction Procedure based on NB approach.
AUTHORS: Kacheria, Shivakumar, Sawkar and Gupta.
DESCRIPTION: The seven parameters considered were income, age, profession, existing loan with its tenure, amount and approval status. The sub-processes include, Pre-processing (handling the missing values with KNN and data refinement using binning algorithm), Classification using NB approach and updating the dataset frequently results in appropriate improvement in the loan prediction process. Experimentation put-forth the conclusion that, integration of KNN and binning algorithm with NB resulted in improved prediction of loan sanctioning process.
MODULES: NB approach integrated with K-Nearest Neighbour (KNN) and binning algorithms.

5.PAPER NAME: Prediction of Modernized Loan Approval System Based on Machine Learning Approach.

AUTHORS: Vaidya, Ashlesha

DESCRIPTION: This model uses logistic regression as a machine learning tool. This paper uses a statistical model (Logistic Regression) to predict whether the loan should be approved or not for a set of records of an applicant. Logistic regression can even work with power terms and nonlinear effect. The historical data of candidates was used to build a machine learning model using different classification algorithms. The main objective of this paper is to predict whether a new applicant granted the loan or not using machine learning models trained on the historical data set.

MODULES: Logistic regression, XGBoost.

2.3 Proposed Methodology:

Since the problem of predicting the approval of a loan application is a classification problem, the model can be trained using classification algorithms like Logistic Regression, Decision Tree, Random Forest Classifier, Support Vector machine. This proposed model will characterize the behaviour of customers on the Basis of their record. These records is taken from the customers, and create a data set. With the help of These data sets and training machine learning model, we predict that the customer’s loan will passed or not.

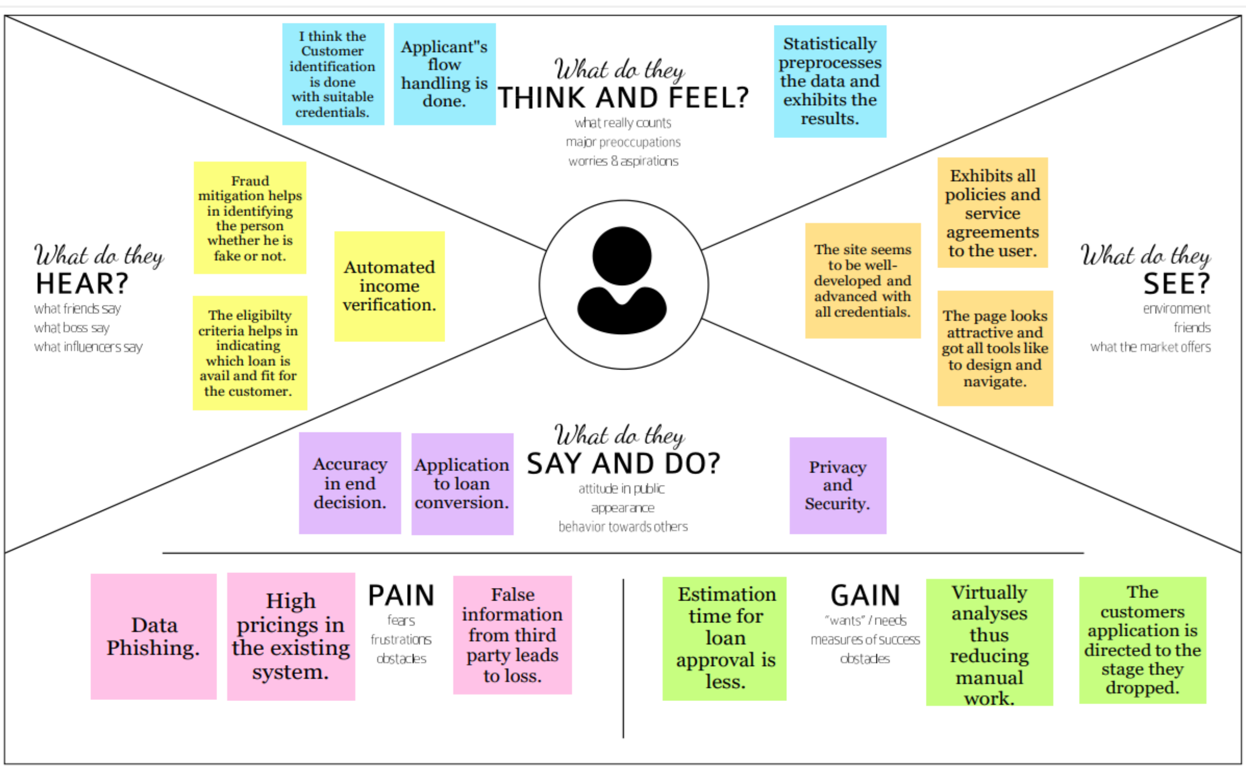
3. IDEATION & PROPOSED SOLUTION

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 helps 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.

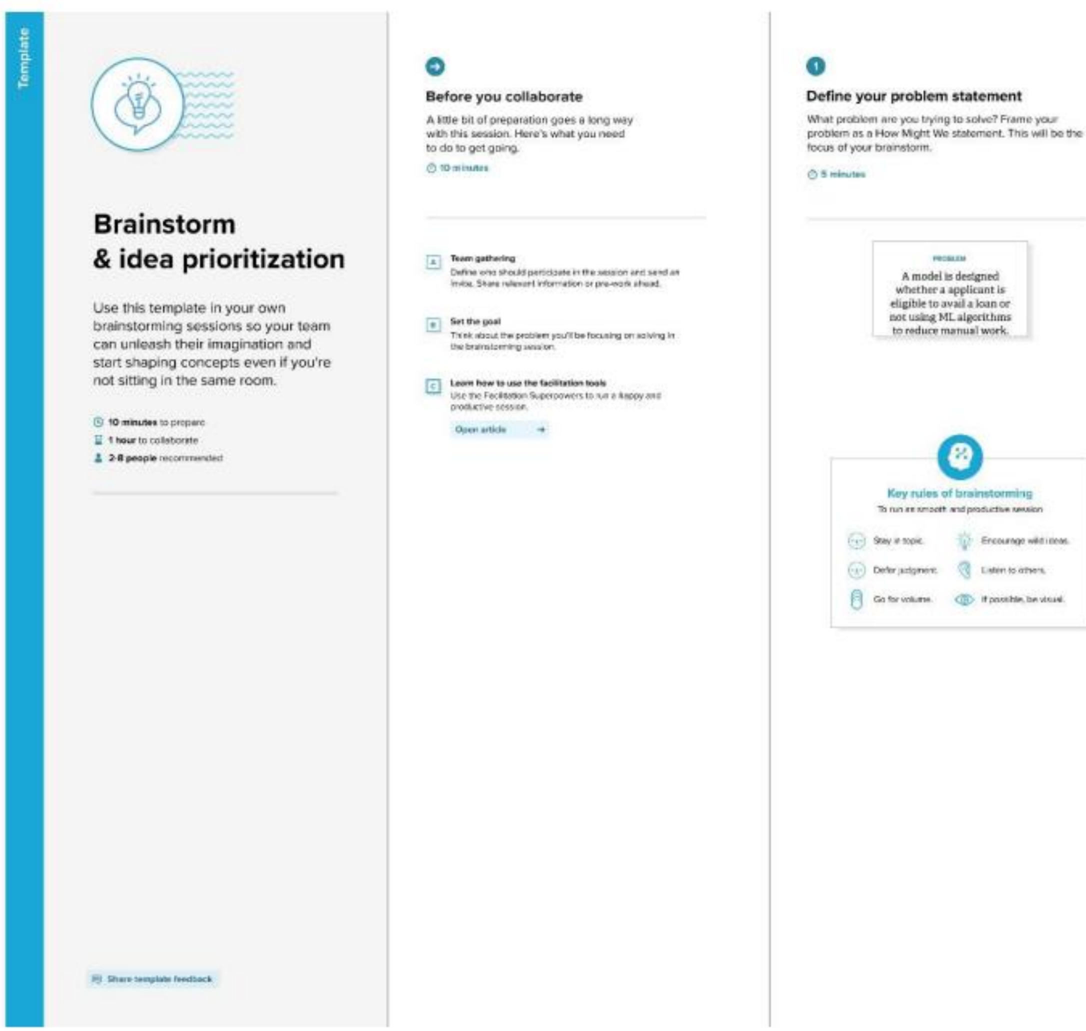
Empathy Map Canvas:



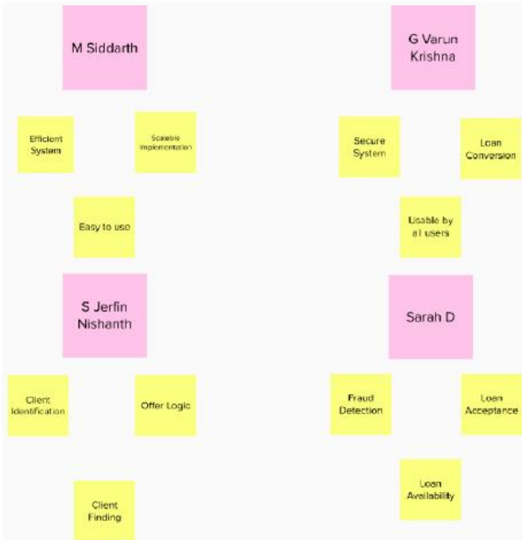
3.2 Ideation & Brainstorming:

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping

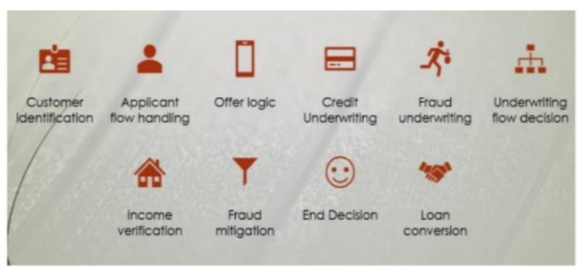
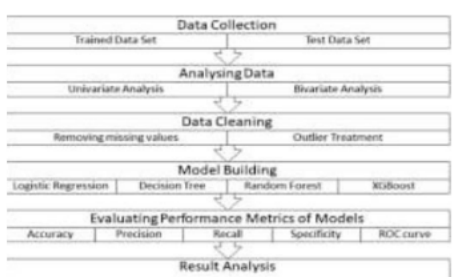


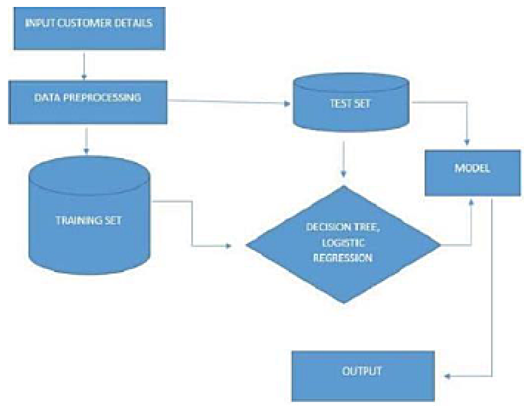
Step-3: Idea Prioritization

3.3 Proposed Solution:

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To reduce the manual work in the banking sector a model is designed to analyse whether an individual is fit enough to avail the loan or not. The main objective is to predict whether a new applicant granted the loan or not using machine learning models trained on the historical data set. The application approved or not approved depends upon the historical data of the candidate by the system. The historical data of candidates was used to build a machine learning model using different classification algorithms.
2.	Idea / Solution description	1. LOGISTIC REGRESSION - LOAN DEFAULTERS: A very important approach in predictive analytics is used to study the problem of predicting loan defaulters using "The Logistic regression model". Here the data is collected from the Kaggle for studying and prediction. The models are compared on the basis of the performance measures such as sensitivity and specificity. 2. RANDOM FOREST - LOAN APPROVAL: To decrease the approval time and the risk associated with the loan many loan prediction models were introduced. Here we are comparing those models and it was found that the Random Forest proved to be the most accurate and fitting where it uses a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression. 3. DECISION TREE – CREDIT RISK ASSESSMENT: Here an effective prediction model is used for the bankers that help them predict the credible customers who have applied for loan. Decision Tree Induction Data Mining Algorithm is applied

		to predict the attributes relevant for credibility. This can be used by the organizations to screen or filter the pool of requests by the customers and it has highest accuracy results.
3.	Novelty / Uniqueness	The novelty of the present study is that the model subtracts the two most pressing issues in the banking sector which is finding out if the borrower is risky and lend the loan to non-risky borrower. The automation of the loan eligibility process acts on the customer details provided while filling online application form. The details are gender, marital status, education, number of dependents, income, loan amount, credit history and others. We are screening the customers through three main factors which is by customer identification, credit underwriting and fraud underwriting. Previous records of applicant is used for better filtering and we direct customers with low interest loans according to their income.
4.	Social Impact / Customer Satisfaction	Since the applicants are approved with low interest loans according to their income and there will be no social impact. The customers will be convenient to pay their interest and no loan defaulters will be identified. This model also helps in concluding that a bank should not only target the rich customers for granting loan but it should assess the other attributes of a customer as well which play a very important part in credit granting decisions and predicting the loan defaulters.
5.	Business Model (Revenue Model)	1. Appliacant Flow Handling:  2. Analyzing or Pre-processing a Dataset: 

		3.Model Results: 
6.	Scalability of the Solution	This model also supports huge set of data and undergoes so many data mining techniques. Those applicants who have high income and demands for lower amount of loan are more likely to get approved and are more likely to pay back their loans. Some other characteristic like gender and marital status seems not to be taken into consideration by the company. But here all the characteristics are considered for approving a loan and not only the rich gets approved for a loan but also a candidate with low income gets to approved for a loan with a low interest. This helps them to pay back their debt without any due.

3.4 Problem Solution fit:

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

- The applicant must be above 20 years
- Loan lenders like banking firms or small financial firms.
- Bank account users.
- Credit/debit card holders.

6. CUSTOMER CONSTRAINTS CC

Not clear in finding out his eligibility criteria for different schemas and There is an increasing rate of loan defaults. Banks identify the loan defaulters for much reduced credit risk as large portions of a bank' s assets directly come from the interest earned on loans given.

5. AVAILABLE SOLUTIONS AS

• Random forest, Logistic regression, Decision tree and Naive bayes algorithm are used
• Using data pre-processing data mining and data filtering
• Algorithms such as naïve bayes, k-nearest neighbors are used.

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS J&P

To find an applicant which can give best interest. Needs to find a loan applicant with good credit score. Accuracy of data should be precise so that it won' t mislead the loan to ineligible user.

9. PROBLEM ROOT CAUSE RC

Identifying the loan defaulters is a difficult task as credit risk evaluation is a very crucial process where it lies as a major factor in the trend of banking sector that affect country' s economy which is credit system handled by banks.

7. BEHAVIOUR BE

The small finance sector that deals with middle class and poor class people seek to find the credibility.
The user can select the loan repayment time and can know the interest rates, credit score and available of loan in nearby location of the consumer.

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TR

The slow process of loan approval is affecting the business of our customer and it also decline the revenue of our customers. Due to the sudden surge in the number of loan defaulters our customers business is highly affected like Financial situation of the user, Credit score rates, Low interest rates.

4. EMOTIONS: BEFORE / AFTER EM

If the data is not secure it shows the insecurity of the user towards the app, it indicates emotion of fear and the vulnerability of the app towards attackers.

10. YOUR SOLUTION SL

• There is an increasing rate of loan defaulters and banks are not able to correctly handle the loan request. To avoid this problem a machine learning algorithm is developed.
• The system automatically selects the credible candidates to approve the loan and it will improve the speed, efficacy, and accuracy of loan approval processes.
• This help the user(Lender) to accurately identify whom to lend the loan and also help the banks to identify the loan defaulter for much-reduced credit risk.
Decision tree, Random forest and logistic regression can be used to detect the credit risk evaluation. We use classification algorithms such as KNN and XGBOOST algorithms that forecast the loan defaulters and predict loan approval.

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE
Customers can easily predict their eligibility through a user interface.
8.2 OFFLINE
- Submission of documents.
- Avail loan manager
- Apply credit/Debit card and also installing the Machine Learning algorithm in their system to work efficiently.

Identify strong TR & EM

4. REQUIREMENT ANALYSIS:

4.1 Functional requirement:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	1. Applicant Credibility description. 2. Information about Credibility details required for loan approval. 3. if new user, REGISTER . 4. if already exists, SIGNIN.
FR-2	User Registration	Enter Email Id ,phone number and other personal information to register in the application
FR-3	User Confirmation	Confirm users via sending OTP to their Email address 0r their phone number.
FR-4	User Login	Enter the user Email Id and the password to login.
FR-5	Loan Approval	Credibility details with their documents have to be submitted for prediction.
FR-6	Result	• If Approved - It displays the credit score and the information about what is done to be next. • If Not Approved - It displays the reason why you are rejected and not eligible for the loan.

4.2 Non-Functional requirement:

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	If the customer is eligible ,he/she should be able to receive the acknowledgement receipt for loan application within 7 days from the bank. If not, then he/she should be intimated with the reason for rejection.
NFR-2	**Security**	Checks if the consumer has any fraudulent history and no data theft to any third party apps.
NFR-3	**Reliability**	Consumer should have good credit scores and stable source of income.
NFR-4	**Performance**	By training the model using different Machine Learning algorithms ,the performance of the system can be increased.

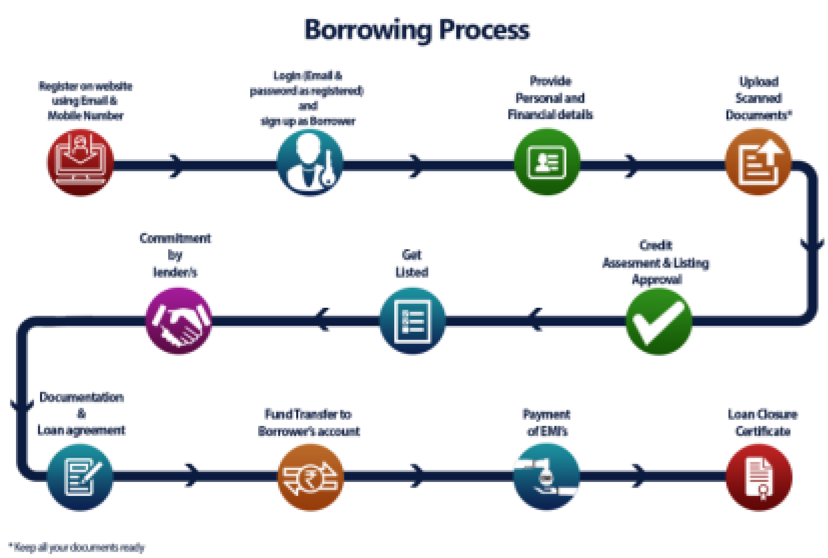
NFR-5	**Availability**	The loan will be easily available for those who have high income and to those who assure to repay the high sum within short period of time.
NFR-6	**Scalability**	The customer should be above the age of 21.And also based on customer's capacity to handle this.

5. PROJECT DESIGN:

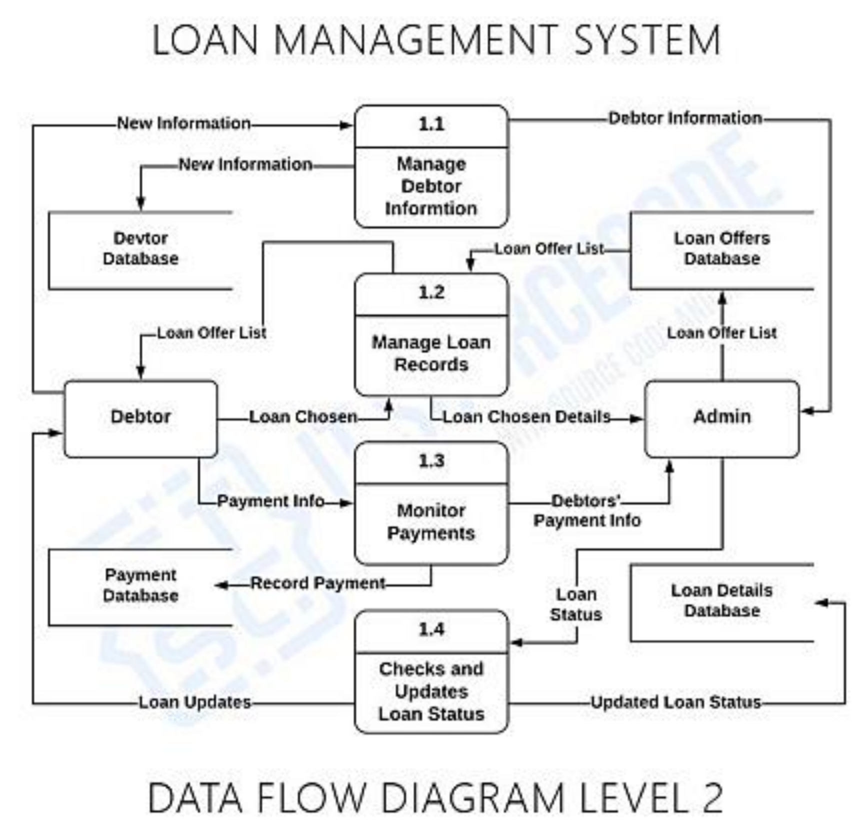
5.1 Data Flow Diagrams:

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.

FLOW: (Simplified)



Example: DFD Level 0 (Industry Standard)



5.2 Solution & Technical Architecture:

Technical Architecture:

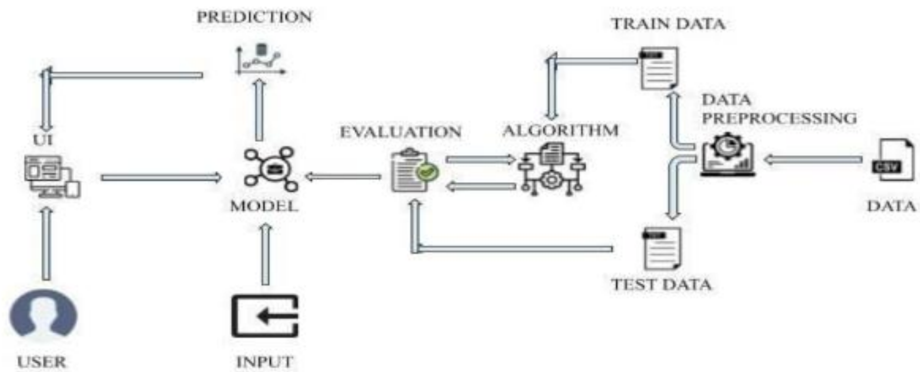


Table-1 : Components & Technologies:

S. No	Component	Description	Technology
1.	User Interface	Users interact with the application with the help of a web UI.	HTML, CSS, JavaScript
2.	Building application	Getting user information from UI and feeding it to ML model	Python Flask
3.	Application Filing	Available to customers only, this screen allows customers to fill details in an online application for loan approval	JavaScript
4.	View Application Status and Manage Applications	View pertinent information relevant to the application and Ability to modify applications asand when needed. Can be accessed by admin only.	JavaScript
5.	Database	Loan Approval dataset.	csv file
6.	Cloud Database	Deploying the model on cloud	IBM cloud
7.	File Storage	Network File System(NFS)	Network File System(NFS)
8.	Visualizing and analysing data	Reading and understanding the data properly with the help of visualization and analysing techniques	Python pandas, numpy, matplotlib,seaborn
9.	Pre-processing or cleaning data	Handling missing values, Handling categorical data, Handling outliers, Scaling Techniques	Python pandas
10.	Machine Learning Model	Using machine learning model for predicting loan approval	Using machine learning model forpredicting loan approval
11.	Infrastructure (Server / Cloud)	Default	Flask

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Open-Source Frameworks Flask is used to host the website. Scikit, numpy and tensorflow are all open source python machine learning frameworks.	Scikit, Numpy
2.	Security Implementations	OpenSSL is a program and library that supports many different cryptographic operations, including: Symmetric key encryption. Public/private key pair generation. Public key encryption. Hash functions.	OpenSSL Encryption
3.	Scalable Architecture	Since the application servers can be deployed on many machines. Also, the database does not make longer connections with every client – it only requires connections from a smaller number of application servers. It improves data integrity.	3 Tier Architecture
4.	Availability	Decentralized storage and distribution along-with web application approach make the service highly available.	IBM Cloud file storage, MySQL Online
5.	Performance	Long term header expiration. Cacheable AJAX Cookie Free Domain Compress zip components.	AJAX, CDN

5.3 User Stories:

Use the below template to list all the user stories for the product.

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

6. PROJECT PLANNING & SCHEDULING:

6.1 Sprint Planning & Estimation:

Use the below template to create product backlog and sprint schedule

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.	3	High	M Siddarth Varun Krishna Jerfin Sarah
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	3	High	M Siddarth Varun Krishna Jerfin Sarah
Sprint-1		USN-3	As a user, I can register for the application through Facebook	1	Low	M Siddarth Varun Krishna Jerfin Sarah
Sprint-1		USN-4	As a user, I can register for the application through Gmail	2	Medium	M Siddarth Varun Krishna Jerfin Sarah
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	3	High	M Siddarth Varun Krishna Jerfin Sarah
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	M Siddarth Varun Krishna Jerfin Sarah
Sprint-2	Register	USN-7	As a loan approval officer, I should be able to register myself as one using unique email and password.	5	Medium	M Siddarth Varun Krishna Jerfin Sarah
Sprint-2	Login	USN-8	As a loan approval officer I should be able to login myself as one using unique email and password.	5	Medium	M Siddarth Varun Krishna Jerfin Sarah
Sprint-3	Automated analysis of credit history	USN-9	As a loan approval officer, I can access the dashboard where I feed application for loan prediction.	10	High	M Siddarth Varun Krishna Jerfin Sarah
Sprint-3		USN-10	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	M Siddarth Varun Krishna Jerfin Sarah
Sprint-4	Register	USN-11	As an admin, I should be able to register myself as one using unique email and password.	2	Medium	M Siddarth Varun Krishna Jerfin Sarah

Sprint-4	Login	USN-12	As an admin I should be able to login myself as one using unique email and password.	2	Medium	M Siddarth Varun Krishna Jerfin Sarah
Sprint-4	Dashboard	USN-13	As an admin, I should be able to access the dashboard with everything I am allowed to use.	2	Medium	M Siddarth Varun Krishna Jerfin Sarah

6.2 Sprint Delivery Schedule:

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	26 Oct 2022	01 Nov 2022	28	01 Nov 2022
Sprint-2	20	6 Days	02 Nov 2022	07 Nov 2022	10	07 Nov 2022
Sprint-3	20	6 Days	08 Nov 2022	13 Nov 2022	25	13 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	6	19 Nov 2022

Velocity:

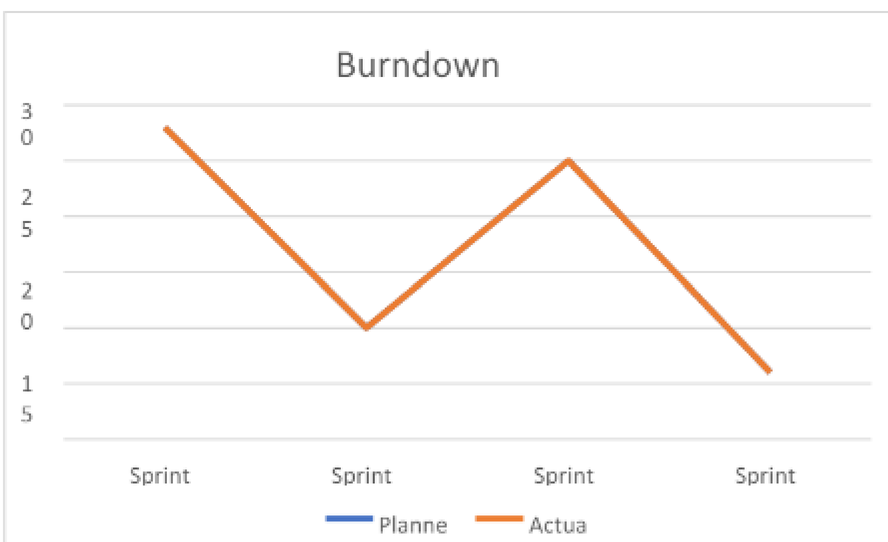
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

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

6.3. Reports from JIRA:

Burndown Chart:

A burn down 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.



7. CODING & SOLUTIONING

7.1. Feature-1:

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

 <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0"
 />
    <title>SMART LENDER</title>
    <style>
       @import
 url("https://fonts.googleapis.com/css2?family=Aref+Ruqaa+Ink:wght@70
 0&display=swap");
       @import
 url("https://fonts.googleapis.com/css2?family=EB+Garamond&display=sw
 ap");
       @import
 url("https://fonts.googleapis.com/css2?family=Antic+Slab&display=swap")
 ;

       html {
          user-select: none;
       }

       body {
          margin-top: 5%;
```

```
    color: white;
}

html {
    background: linear-gradient(rgba(0, 0, 0, 0.5), rgba(0, 0, 0, 0.5)),
url(static/loan.jpeg);
    height: 100%;
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;
    object-fit: cover;
}

h1 {
    font-size: 45px;
    font-family: "Aref Ruqaa Ink", serif;
}

h3 {
    font-size: 20px;
    font-family: "Antic Slab", serif;
}

h6 {
    font-size: 20px;
    font-family: "Antic Slab", serif;
}

/* ~~~~~~~ BUTTON ~~~~~~~ */

.container,
```

```
.container:before,
.container:after {
   box-sizing: border-box;
   padding: 0;
   margin: 0;
   font: 300 1em/1.5 "Open Sans", "Helvetica Neue", Arial, sans-
serif;
   text-decoration: none;
   color: #111;
}

.btn {
   background: rgba(236, 240, 241, 0.425);
}

.container {
   min-width: 500px;
   margin: 5% auto;
   text-align: center;
}

button:hover {
   cursor: pointer;
}

button {
   background: transparent;
   outline: none;
   position: relative;
   border: 3px solid #FCDDB0;
   padding: 15px 50px;
```

```
    overflow: hidden;
}

/*button:before (attr data-hover)*/
button:hover:before {
    opacity: 1;
    transform: translate(0, 0);
}

button:before {
    content: attr(data-hover);
    position: absolute;
    top: 1.1em;
    left: 0;
    width: 100%;
    text-transform: uppercase;
    letter-spacing: 3px;
    font-weight: 800;
    font-size: 0.8em;
    opacity: 0;
    transform: translate(-100%, 0);
    transition: all 0.3s ease-in-out;
}

/*button div (button text before hover)*/
button:hover div {
    opacity: 0;
    transform: translate(100%, 0);
}

button div {
```

```
    text-transform: uppercase;
    letter-spacing: 3px;
    font-weight: 800;
    font-size: 0.8em;
    transition: all 0.3s ease-in-out;
}

/*--- Footer ---*/

.footer {
    margin-top: 10px;
}

.nav-link {
    font-weight: bold;
    font-size: 14px;
    text-transform: uppercase;
    text-decoration: none;
    color: #ffffff;
    padding: 20px 0px;


    display: inline-block;
    position: relative;
    opacity: 0.75;
}

#d {
    margin-top: -40px;
    font-family: "EB Garamond", serif;
    letter-spacing: 0.5px;
```

```
}

#p {
    margin-top: -50px;
    font-family: "EB Garamond", serif;
    letter-spacing: 0.5px;
}

.nav-link:hover {
    opacity: 1;
}

.nav-link::before {
    transition: 300ms;
    height: 3px;
    content: "";
    position: absolute;
    background-color: #FCDDB0;
}

.nav-link-fade-up::before {
    width: 100%;
    bottom: 5px;
    opacity: 0;
}

.nav-link-fade-up:hover::before {
    bottom: 10px;
    opacity: 1;
}
```

```css
p {
    color: white;
    font-family: "Aref Ruqaa Ink", serif;
    letter-spacing: 0.5px;
}

.tooltip {
    position: relative;
    display: inline-block;
    /* If you want dots under the hoverable text */
}

/* Tooltip text */
.tooltip .tooltiptext {
    border-radius: 10px;
    visibility: hidden;
    width: 100px;
    color: #fff;
    right: 28vh;
    /* Position the tooltip text - see examples below! */
    position: absolute;
    z-index: 1;
}



/* Show the tooltip text when you mouse over the tooltip container */
.tooltip:hover .tooltiptext {
    visibility: visible;
}
```

```
.tooltip1 {
   position: relative;
   display: inline-block;
   /* If you want dots under the hoverable text */
}

/* Tooltip text */
.tooltip1 .tooltiptext1 {
   border-radius: 10px;
   visibility: hidden;
   width: 100px;
   color: #fff;
   text-align: center;
   left: 28vh;
   /* Position the tooltip text - see examples below! */
   position: absolute;
   z-index: 1;
}



/* Show the tooltip text when you mouse over the tooltip container */
.tooltip1:hover .tooltiptext1 {
   visibility: visible;
}


@media only screen and (max-width: 600px) {

   html {
      width: 100% !important;
```

```css
}

body {
    margin-top: 110px;
}

h1 {
    font-size: 40px;
}

h3 {
    font-size: 15px;
}

.container {
    min-width: 200px;
}

.btn {
    margin-right: 2vh;
}

#d {
    letter-spacing: 0px;
    font-size: 14px;
}

#p {
    letter-spacing: 0px;
    font-size: 14px;
}
```

```
            .footer {
                margin-top: 15vh;
            }

            .tooltip .tooltiptext {
                display: none;
            }

            .tooltip1 .tooltiptext1 {
                display: none;
            }

        }
    </style>
</head>

<body>
    <main>
        <center>
            <h1>Smart Lender</h1>
            <h3>Get to know your applicant application will get accepted or
not</h3>
            <h6>
                Click the <em><b> Predict </b></em> button and update the
details to
                know the prediction for the applicant.
            </h6>

            <div class="container">
                <a href="predict.html">
```

```
                    <button style="color: #ffffff ;" class="btn" data-hover="Loan
Predictor" onclick="predict.html">
                        <div>Predict</div>
                    </button>
                </a>
            </div>

        </center>
    </main>
</body>

</html>
```

predict.html:

```
<html lang="en">

<head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>SMART LENDER</title>
    <style>
        @import url("https://fonts.googleapis.com/css2?family=Aref+Ruqaa+Ink:wght@700&display=swap");
        @import url("https://fonts.googleapis.com/css2?family=Albert+Sans&display=swap");
        @import url("https://fonts.googleapis.com/css2?family=EB+Garamond&display=swap");

        html {
```

```
    height: 100%;
}

body {
    margin: 0;
    margin-bottom: 50%;
    padding: 0;
    font-family: sans-serif;
    /* background: linear-gradient(#141e30, #243b55);*/
    background-image: linear-gradient(rgba(0, 0, 0, 0.5),
            rgba(0, 0, 0, 0.5)),
        url(static/loan1.jpeg);
    height: 10%;
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;
    background-attachment: fixed;
    object-fit: fill;
}

.login-box {
    position: absolute;
    top: 100%;
    left: 50%;
    width: 400px;
    padding: 40px;
    transform: translate(-50%, -50%);
    background: rgba(0, 0, 0, 0.5);
    box-sizing: border-box;
    box-shadow: 0 15px 25px rgba(0, 0, 0, 0.6);
    border-radius: 10px;
}

::placeholder {
```

```
    color: aliceblue;
}

.login-box h2 {
    margin: 0 0 30px;
    padding: 0;
    color: #fff;
    text-align: center;
}

.fon {
    color: #fff;
    text-align: center;
    font-family: "Albert Sans", sans-serif;
}

.login-box .user-box {
    position: relative;
}

.login-box .user-box input {
    width: 100%;
    padding: 10px 0;
    font-size: 16px;
    color: #fff;
    margin-bottom: 30px;
    border: none;
    border-bottom: 1px solid #fff;
    outline: none;
    background: transparent;
}

.login-box .user-box label {
    position: absolute;
```

```
    top: 0;
    left: 0;
    padding: 10px 0;
    font-size: 16px;
    color: #fff;
    pointer-events: none;
    transition: 0.5s;
}

.login-box .user-box input:focus~label,
.login-box .user-box input:valid~label {
    top: -20px;
    left: 0;
    color: #FCDDB0;
    font-size: 12px;
}

/*--- Button */

.container,
.container:before,
.container:after {
    box-sizing: border-box;
    padding: 0;
    margin: 0;
    font: 300 1em/1.5 "Open Sans", "Helvetica Neue", Arial, sans-serif;
    text-decoration: none;
    color: #111;
}

.btn {
    background: rgba(236, 240, 241, 0.425);
}
```

```
.container {
    min-width: 500px;
    margin: 5% auto;
    text-align: center;
}

button:hover {
    cursor: pointer;
}

button {
    background: transparent;
    outline: none;
    position: relative;
    border: 3px solid #FCDDB0;
    padding: 15px 50px;
    overflow: hidden;
}

/*button:before (attr data-hover)*/
button:hover:before {
    opacity: 1;
    transform: translate(0, 0);
}

button:before {
    content: attr(data-hover);
    position: absolute;
    top: 1.1em;
    left: 0;
    width: 100%;
    text-transform: uppercase;
    letter-spacing: 3px;
    font-weight: 800;
```

```
    font-size: 0.8em;
    opacity: 0;
    transform: translate(-100%, 0);
    transition: all 0.3s ease-in-out;
}

/*button div (button text before hover)*/
button:hover div {
    opacity: 0;
    transform: translate(100%, 0);
}

button div {
    text-transform: uppercase;
    letter-spacing: 3px;
    font-weight: 800;
    font-size: 0.8em;
    transition: all 0.3s ease-in-out;
}

/*--- Footer ---*/

.footer {
    margin-top: 200vh;
    margin-bottom: 10px;
}

.nav-link {
    font-weight: bold;
    font-size: 14px;
    text-transform: uppercase;
    text-decoration: none;
    color: #ffffff;
    padding: 20px 0px;
```

```
    /* margin: 0px 20px;*/

    display: inline-block;
    position: relative;
    opacity: 0.75;
}

#d {
    margin-top: -40px;
    font-family: "EB Garamond", serif;
    letter-spacing: 0.5px;
}

#p {
    margin-top: -50px;
    font-family: "EB Garamond", serif;
    letter-spacing: 0.5px;
}

.nav-link:hover {
    opacity: 1;
}

.nav-link::before {
    transition: 300ms;
    height: 3px;
    content: "";
    position: absolute;
    background-color: #FCDDB0;
}

.nav-link-fade-up::before {
    width: 100%;
    bottom: 5px;
```

```
    opacity: 0;
}

.nav-link-fade-up:hover::before {
    bottom: 10px;
    opacity: 1;
}

p {
    color: white;
    font-family: "Aref Ruqaa Ink", serif;
    letter-spacing: 0.5px;
}

.tooltip {
    position: relative;
    display: inline-block;
    /* If you want dots under the hoverable text */
}

/* Tooltip text */
.tooltip .tooltiptext {
    border-radius: 10px;
    visibility: hidden;
    width: 100px;
    right: 28vh;
    /* Position the tooltip text - see examples below! */
    position: absolute;
    z-index: 1;
}

/* Show the tooltip text when you mouse over the tooltip container */
.tooltip:hover .tooltiptext {
    visibility: visible;
```

```
}

.tooltip1 {
    position: relative;
    display: inline-block;
    /* If you want dots under the hoverable text */
}

/* Tooltip text */
.tooltip1 .tooltiptext1 {
    border-radius: 10px;
    visibility: hidden;
    width: 100px;
    left: 28vh;
    /* Position the tooltip text - see examples below! */
    position: absolute;
    z-index: 1;
}

/* Show the tooltip text when you mouse over the tooltip container */
.tooltip1:hover .tooltiptext1 {
    visibility: visible;
}

@media only screen and (max-width: 600px) {
    .login-box {
        width: 300px;
    }

    .container {
        min-width: 200px;
    }

    .footer {
```

```
            position: sticky;
            margin-top: 198vh;
            font-size: 20px;
        }

        #d {
            letter-spacing: 0px;
            font-size: 14px;
        }

        #p {
            letter-spacing: 0px;
            font-size: 14px;
        }

        .fon {
            font-size: 15px;
        }

        .tooltip .tooltiptext {
            display: none;
        }

        .tooltip1 .tooltiptext1 {
            display: none;
        }
      }
    </style>
</head>

<body>
    <div class="login-box">
      <h2 style="text-transform: uppercase; font-family: 'Aref Ruqaa Ink', serif">
        Smart lender - <br />
```

```
  <span style="font-size: 14px; color: azure">Know your Loan eligibility</span>
</h2>
<p class="fon" style="font-size: 14px">
  Let's begin by entering your deatils below
</p>
<br />
<form action="/submit" method="post">
  <div class="user-box">
    <input type="text" name="" required="" onfocus="this.placeholder='Enter your name'"
      onblur="this.placeholder=''" />
    <label>Name</label>
  </div>
  <div class="user-box">
    <input type="text" name="Loan_ID" required="" onfocus="this.placeholder='Enter your Loan ID'"
      onblur="this.placeholder=''" />
    <label>Loan ID</label>
  </div>
  <div class="user-box">
    <input list="gender" type="data-list" name="Gender" required="" onchange="resetIfInvalid(this);"
      onfocus="this.placeholder='Enter your Gender'" onblur="this.placeholder=''" />
    <label>Gender</label>
    <datalist id="gender" name="gender">
      <option value="Male"></option>
      <option value="female"></option>
    </datalist>
  </div>
  <div class="user-box">
    <input list="married" type="text" name="Married" required="" onchange="resetIfInvalid(this);"
      onfocus="this.placeholder='Enter your Marital Status'" onblur="this.placeholder=''" />
    <label>Married</label>
    <datalist id="married" name="married">
      <option value="yes"></option>
      <option value="no"></option>
    </datalist>
```

```html
</div>
<div class="user-box">
    <input list="dep" type="text" name="Dependents" required="" onchange="resetIfInvalid(this);"
        onfocus="this.placeholder='Enter your Dependents'" onblur="this.placeholder=''" />
    <label>Dependents</label>
    <datalist id="dep" name="dep">
        <option value="0"></option>
        <option value="1"></option>
        <option value="2"></option>
        <option value="3+"></option>
    </datalist>
</div>
<div class="user-box">
    <input list="edu" type="text" name="Education" required="" onchange="resetIfInvalid(this);"
        onfocus="this.placeholder='Enter your Educational Qualification'" onblur="this.placeholder=''" />
    <label>Education</label>
    <datalist name="edu" id="edu">
        <option value="Graduate"></option>
        <option value="Non-Graduate"></option>
    </datalist>
</div>
<div class="user-box">
    <input list="emp" type="text" name="Self_Employes" required="" onchange="resetIfInvalid(this);"
        onfocus="this.placeholder='Are you self employed?'" onblur="this.placeholder=''" />
    <label>Self Employed</label>
    <datalist name="emp" id="emp">
        <option value="yes"></option>
        <option value="no"></option>
    </datalist>
</div>
<div class="user-box">
    <input type="number" name="ApplicantIncome" required=""
        onfocus="this.placeholder='Enter your Income in Dollars'" onblur="this.placeholder=''" />
    <label>Applicant Income</label>
```

```
</div>
<div class="user-box">
    <input type="number" name="CoaaplicantIncome" required=""
        onfocus="this.placeholder='Enter your CO Applicant Income in Dollars'"
        onblur="this.placeholder=''" />
    <label>CO Applicant Income</label>
</div>
<div class="user-box">
    <input type="number" name="LoanAmount" required=""
        onfocus="this.placeholder='Enter your Loan Amount in Dollars'" onblur="this.placeholder=''" />
    <label>Loan Amount</label>
</div>
<div class="user-box">
    <input list="term" type="text" name="Loan_Amount_Term" required="" onchange="resetIfInvalid(this
        onfocus="this.placeholder='Enter the loan amount term'" onblur="this.placeholder=''" />
    <label>Loan Amount Term</label>
    <datalist name="term" id="term">
        <option value="480"></option>
        <option value="360"></option>
        <option value="300"></option>
        <option value="240"></option>
        <option value="180"></option>
        <option value="120"></option>
        <option value="84"></option>
        <option value="60"></option>
        <option value="36"></option>
        <option value="12"></option>
    </datalist>
</div>
<div class="user-box">
    <input list="credit" type="text" name="Credit_History" required="" onchange="resetIfInvalid(this);"
        onfocus="this.placeholder='Enter your Credit History'" onblur="this.placeholder=''" />
    <label>Credit History</label>
    <datalist name="credit" id="credit">
```

```
                <option value="yes"></option>
                <option value="no"></option>
            </datalist>
        </div>
        <div class="user-box">
            <input list="prop" type="text" name="Property_Area" required="" onchange="resetIfInvalid(this);"
                onfocus="this.placeholder='Enter your area of the property'" onblur="this.placeholder=''" />
            <label>Property Area</label>
            <datalist name="prop" id="prop">
                <option value="Urban"></option>
                <option value="Rural"></option>
                <option value="Semi-Rural"></option>
            </datalist>
        </div>

        <div class="container">
            <a href="submit.html">
                <button style="color: #ffffff ;" class="btn" data-hover="PREDICT" onclick="submit.html">
                    <div>SUBMIT</div>
                </button>
            </a>
        </div>
    </form>
  </div>
</body>
<script>
  function resetIfInvalid(el) {
    //just for beeing sure that nothing is done if no value selected
    if (el.value == "") return;
    var options = el.list.options;
    for (var i = 0; i < options.length; i++) {
        if (el.value == options[i].value)
            //option matches: work is done
            return;
```

```
        }
        //no match was found: reset the value
        el.value = "";
    }
</script>

</html>
```

submit.html:

```
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>SMART LENDER</title>
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Aref+Ruqaa+Ink:wght@700&display=swap');
        @import url('https://fonts.googleapis.com/css2?family=EB+Garamond&display=swap');

        body {
            color: white;
            font-family: 'Aref Ruqaa Ink', serif;
            background-image: linear-gradient(rgba(0, 0, 0, 0.5),
                    rgba(0, 0, 0, 0.5)),
                url(static/loan.jpeg);
            height: 10%;
            background-position: center;
            background-repeat: no-repeat;
```

```
    background-size: cover;
    background-attachment: fixed;
    object-fit: fill;
}

.output {
    margin-top: 15%;
}


/*--- Footer ---*/

.footer {
    margin-top: 21vh;
}

.nav-link {
    font-weight: bold;
    font-size: 14px;
    text-transform: uppercase;
    text-decoration: none;
    color: #ffffff;
    padding: 20px 0px;
    /* margin: 0px 20px;*/

    display: inline-block;
    position: relative;
    opacity: 0.75;
}

#d {
    margin-top: -40px;
    font-family: 'EB Garamond', serif;
    letter-spacing: 0.5px;
```

```
}

#p {
    /* margin-top: -50px;*/
    font-family: 'EB Garamond', serif;
    letter-spacing: 0.5px;
}

.nav-link:hover {
    opacity: 1;
}

.nav-link::before {
    transition: 300ms;
    height: 3px;
    content: "";
    position: absolute;
    background-color: #FCDDB0;
}

.nav-link-fade-up::before {
    width: 100%;
    bottom: 5px;
    opacity: 0;
}

.nav-link-fade-up:hover::before {
    bottom: 10px;
    opacity: 1;
}

p {
    color: white;
    font-family: 'Aref Ruqaa Ink', serif;
```

```
    letter-spacing: 0.5px;
}

.tooltip {
    position: relative;
    display: inline-block;
    /* If you want dots under the hoverable text */
}

/* Tooltip text */
.tooltip .tooltiptext {
    border-radius: 10px;
    visibility: hidden;
    width: 100px;
    right: 28vh;
    /* Position the tooltip text - see examples below! */
    position: absolute;
    z-index: 1;
}

/* Show the tooltip text when you mouse over the tooltip container */
.tooltip:hover .tooltiptext {
    visibility: visible;
}

.tooltip1 {
    position: relative;
    display: inline-block;
    /* If you want dots under the hoverable text */
}

/* Tooltip text */
.tooltip1 .tooltiptext1 {
    border-radius: 10px;
```

```
        visibility: hidden;
        width: 100px;
        top: 3vh;
        left: 28vh;
        /* Position the tooltip text - see examples below! */
        position: absolute;
        z-index: 1;
      }

      /* Show the tooltip text when you mouse over the tooltip container */
      .tooltip1:hover .tooltiptext1 {
        visibility: visible;
      }

      @media only screen and (max-width: 600px) {

        body {
          margin-top: 30vh;
        }

        .footer {
          margin-top: 30vh;
        }

        .tooltip .tooltiptext {
          display: none;
        }

        .tooltip1 .tooltiptext1 {
          display: none;
        }
      }
    </style>
  </head>
```

```
<body>
    <main class="output">
        <center>
            <h1>SMART LENDER</h1>
            <h3>{{prediction_text}}</h3>
        </center>
    </main>
</body>

</html>
```

8. Testing

8.1. Test Cases:

For checking the loan application, We have two testcase

• Eligible

• Not Eligible

This is based on the training and testing the model we used in our application. This eligibility can be checked by using the details entered by the users.

This includes the details like

• Gender

• Status

• Dependants

• Education

• Employ

• Income

• Co-income(additional income)

• Loan amount

• Loan amount term(in days)

• Credit history

• Property area(type of location)

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 Smart Lender – Applicant Credibility Prediction for Loan Approval project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	7	3	2	3	11
Duplicate	1	0	3	0	3
External	3	2	0	1	6
Fixed	0	2	4	16	17
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	2	2	1	8
Totals	11	9	13	22	48

3. Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	13	0	0	15
Security	2	0	0	3

9. RESULTS:

9.1. Performance Metrics:

Model Performance Testing:

For our model performance testing, we are using XG-boost for prediction.

S.No.	Parameter	Values	Screenshot
1.	Metrics	**Regression Model:** MAE - , MSE - , RMSE - , R2 score - **Classification Model:** Confusion Matrix - , Accuracy Score- & Classification Report -	FIGURE-1
2.	Tune the Model	Hyper parameter Tuning - Validation Method -	FIGURE-2

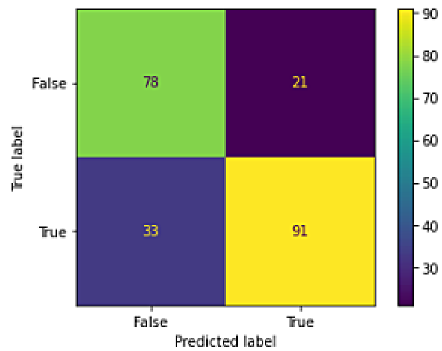
DECISION TREE CLASSIFIER

SOURCE CODE

```
In [33]: from sklearn.tree import DecisionTreeClassifier
import matplotlib.pyplot as plt
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report,f1_score,plot_precision_recall_curve,plot_roc_c
def decisionTreeClassifier(x_train, x_test, y_train, y_test):
    dt = DecisionTreeClassifier()
    dt.fit(x_train,y_train)
    yPred = dt.predict(x_test)
    print("****DecisionTreeClassifier****")
    print("Confusion matrix")
    confusion_matrix = metrics.confusion_matrix(y_test, yPred)
    cm_display = metrics.ConfusionMatrixDisplay(confusion_matrix = confusion_matrix, display_labels = [False, True])
    cm_display.plot()
    plt.show()
    print("Classification report")
    print(classification_report (y_test, yPred))
    y_pred=dt.predict(x_test)
    y_pred1=dt.predict(x_train)
    print('Testing Accuracy : ',accuracy_score(y_test,y_pred))
    print('Training Accuracy : ',accuracy_score(y_train,y_pred1))
    print('AUC Score : ',roc_auc_score(y_test,y_pred))
    plot_roc_curve(dt, x_test, y_test, name = 'Decison Tree Model')
    plot_precision_recall_curve(dt, x_test, y_test, name = 'Decison Tree Model')
decisionTreeClassifier(x_train, x_test, y_train, y_test)
```

OUTPUT

```
****DecisionTreeClassifier****
Confusion matrix
```

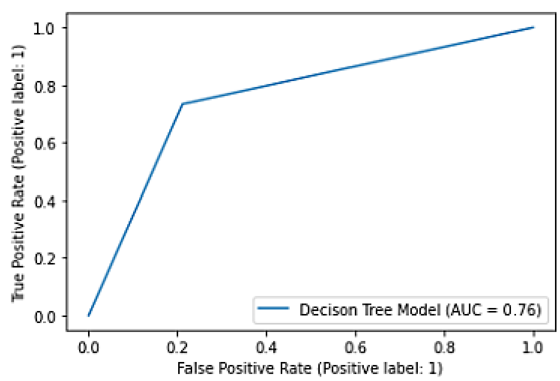


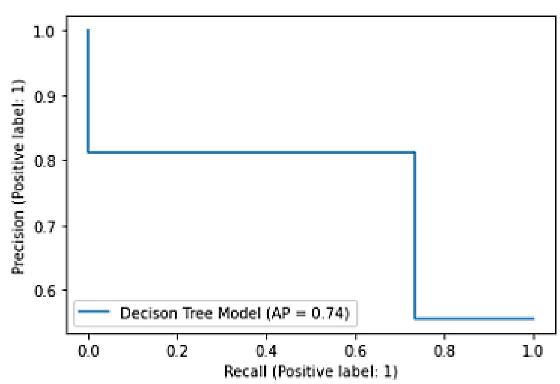

```
Classification report
              precision    recall  f1-score   support

           0       0.70      0.79      0.74        99
           1       0.81      0.73      0.77       124

    accuracy                           0.76       223
   macro avg       0.76      0.76      0.76       223
weighted avg       0.76      0.76      0.76       223

Testing Accuracy :  0.757847533632287
Training Accuracy :  1.0
AUC Score :  0.7608748778103617
```





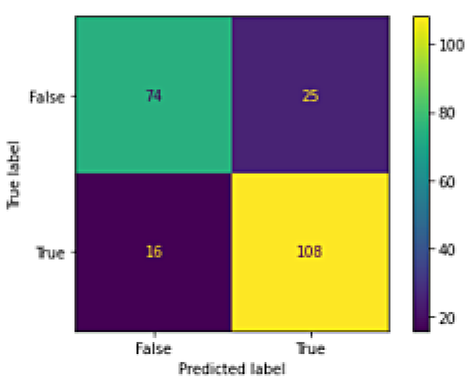
RANDOM FOREST CLASSIFIER

SOURCE CODE

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report,f1_score,plot_precision_recall_curve,plot_roc_c
def randomForestClassifier(x_train, x_test, y_train, y_test):
    rf = RandomForestClassifier()
    rf.fit(x_train,y_train)
    yPred = rf.predict(x_test)
    print("****RandomForestClassifier****")
    print("Confusion matrix")
    confusion_matrix = metrics.confusion_matrix(y_test, yPred)
    cm_display = metrics.ConfusionMatrixDisplay(confusion_matrix = confusion_matrix, display_labels = [False, True])
    cm_display.plot()
    plt.show()
    print("Classification report")
    print(classification_report (y_test, yPred))
    y_pred=rf.predict(x_test)
    y_pred1=rf.predict(x_train)
    print('Testing accuracy: ',accuracy_score(y_test,y_pred))
    print('Training accuracy: ',accuracy_score(y_train,y_pred1))
    print('AUC Score : ',roc_auc_score(y_test,y_pred))
    plot_precision_recall_curve(rf, x_test, y_test, name = 'Random Forest Model')
    plot_roc_curve(rf, x_test, y_test, name = 'Random Forest Model')
randomForestClassifier(x_train, x_test, y_train, y_test)
```

OUTPUT

```
****RandomForestClassifier****
Confusion matrix
```

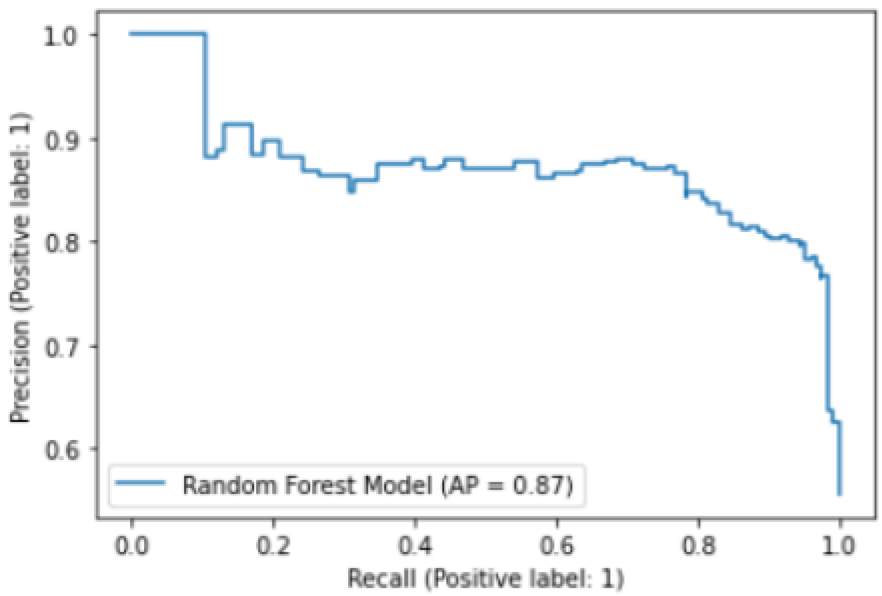


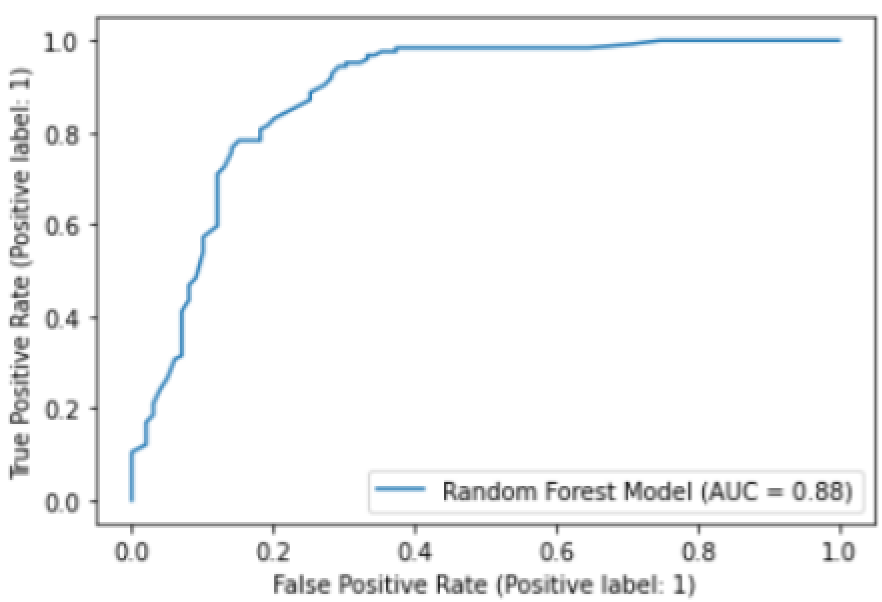

```
Classification report
              precision    recall  f1-score   support

           0       0.82      0.75      0.78        99
           1       0.81      0.87      0.84       124

    accuracy                           0.82       223
   macro avg       0.82      0.81      0.81       223
weighted avg       0.82      0.82      0.81       223

Testing accuracy:  0.8161434977578476
Training accuracy:  1.0
AUC Score :  0.8092212447051157
```


1.0
0.9
0.8
0.7
0.6
Precision (Positive label: 1)
Random Forest Model (AP = 0.87)
0.0
0.2
0.4
0.6
0.8
1.0
Recall (Positive label: 1)


1.0
0.8
0.6
0.4
0.2
0.0
True Positive Rate (Positive label: 1)
Random Forest Model (AUC = 0.88)
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate (Positive label: 1)

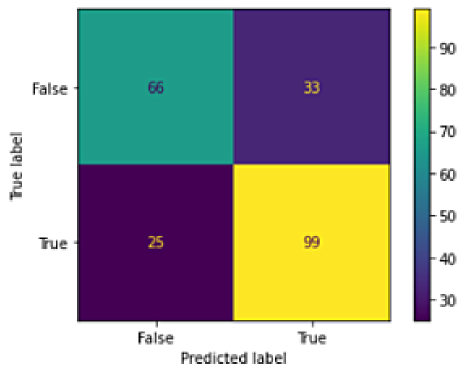
KNN CLASSIFIER

SOURCE CODE

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report,f1_score,plot_precision_recall_curve,plot_roc_c
def kneighborsClassifier(x_train, x_test, y_train, y_test):
    knn = KNeighborsClassifier()
    knn.fit(x_train,y_train)
    yPred = knn.predict(x_test)
    print("****KNeighborsClassifier****")
    print("Confusion matrix")
    confusion_matrix = metrics.confusion_matrix(y_test, yPred)
    cm_display = metrics.ConfusionMatrixDisplay(confusion_matrix = confusion_matrix, display_labels = [False, True])
    cm_display.plot()
    plt.show()
    print("Classification report")
    print(classification_report (y_test, yPred))
    y_pred=knn.predict(x_test)
    y_pred1=knn.predict(x_train)
    print('Testing accuracy: ',accuracy_score(y_test,y_pred))
    print('Training accuracy: ',accuracy_score(y_train,y_pred1))
    print('AUC Score : ',roc_auc_score(y_test,y_pred))
    plot_precision_recall_curve(knn, x_test, y_test, name = 'KNN Model')
    plot_roc_curve(knn, x_test, y_test, name = 'KNN Model')
kneighborsClassifier(x_train, x_test, y_train, y_test)
```

OUTPUT

```
****KNeighborsClassifier****
Confusion matrix
```

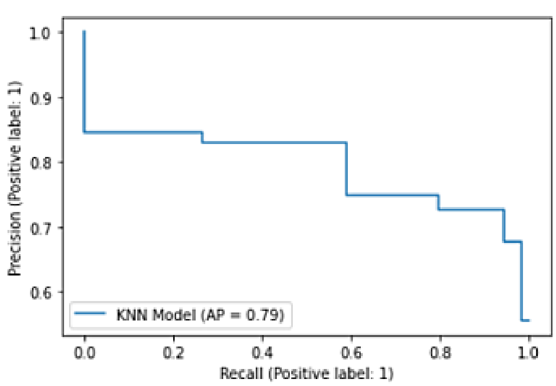


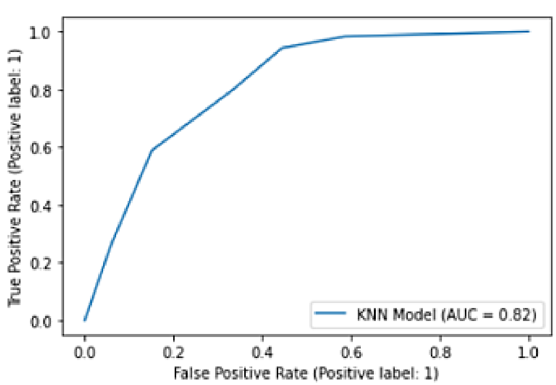

```
Classification report
              precision    recall  f1-score   support

           0       0.73      0.67      0.69        99
           1       0.75      0.80      0.77       124

    accuracy                           0.74       223
   macro avg       0.74      0.73      0.73       223
weighted avg       0.74      0.74      0.74       223

Testing accuracy:  0.7399103139013453
Training accuracy:  0.8333333333333334
AUC Score :  0.7325268817204301
```





GRADIENT BOOSTING CLASSIFIER

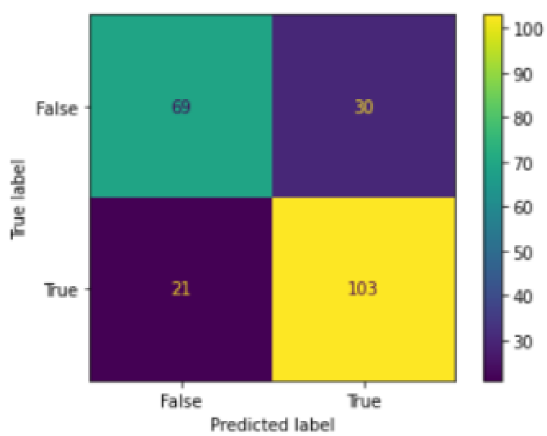
SOURCE CODE

```
In [35]: from sklearn.ensemble import  GradientBoostingClassifier
         from sklearn.metrics import accuracy_score,confusion_matrix,classification_report,f1_score,plot_precision_recall_curve,plot_roc_c
         def xgboost(x_train, x_test, y_train, y_test):
             xg = GradientBoostingClassifier()
             xg.fit(x_train,y_train)
             yPred = xg.predict(x_test)
             print("****Gradient BoostingClassifier****")
             print("Confusion matrix")
             confusion_matrix = metrics.confusion_matrix(y_test, yPred)
             cm_display = metrics.ConfusionMatrixDisplay(confusion_matrix = confusion_matrix, display_labels = [False, True])
             cm_display.plot()
             plt.show()
             print("Classification report")
             print(classification_report (y_test, yPred))
             y_pred=xg.predict(x_test)
             y_pred1=xg.predict(x_train)
             print('Testing accuracy: ',accuracy_score(y_test,y_pred))
             print('Training accuracy: ',accuracy_score(y_train,y_pred1))
             print('AUC Score : ',roc_auc_score(y_test,y_pred))
             plot_precision_recall_curve(xg, x_test, y_test, name = 'XGBoost Model')
             plot_roc_curve(xg, x_test, y_test, name = 'XGBoost Model')

         xgboost(x_train, x_test, y_train, y_test)
```

OUTPUT

```
****Gradient BoostingClassifier****
Confusion matrix
```

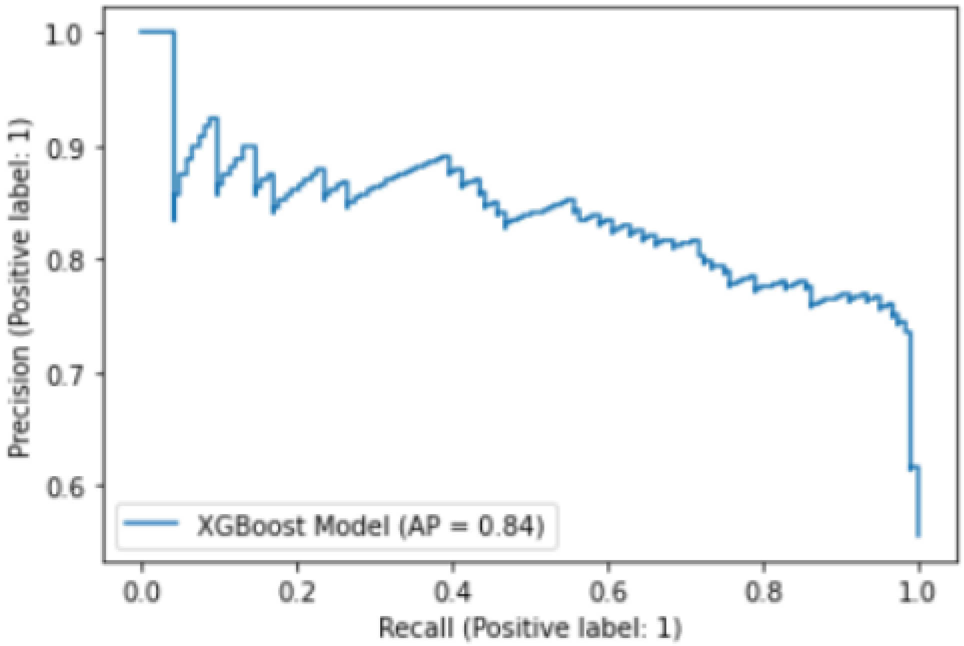


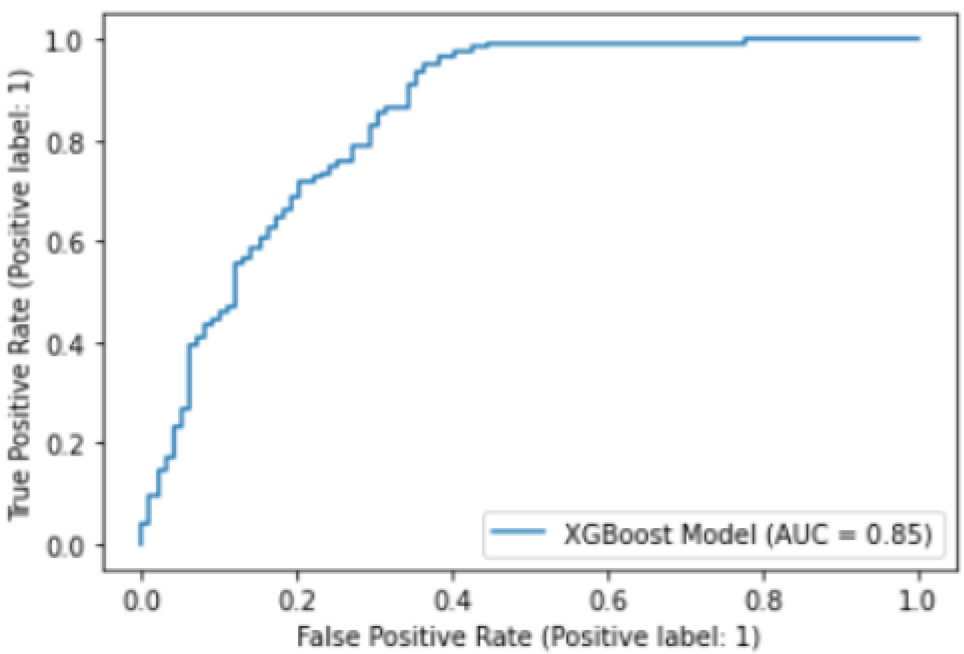

```
Classification report
              precision    recall  f1-score   support

           0       0.77      0.70      0.73        99
           1       0.77      0.83      0.80       124

    accuracy                           0.77       223
   macro avg       0.77      0.76      0.77       223
weighted avg       0.77      0.77      0.77       223

Testing accuracy:  0.7713004484304933
Training accuracy:  0.9444444444444444
AUC Score :  0.7638074291300098
```


1.0
0.9
0.8
0.7
0.6
Precision (Positive label: 1)
XGBoost Model (AP = 0.84)
0.0
0.2
0.4
0.6
0.8
1.0
Recall (Positive label: 1)


1.0
0.8
0.6
0.4
0.2
0.0
True Positive Rate (Positive label: 1)
XGBoost Model (AUC = 0.85)
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate (Positive label: 1)

10.ADVANTAGES & DISADVANTAGES:

ADVANTAGES:

- Fast and highly accurate result
- Easy handling of the problem
- Less risk and more convenient to use
- Reliablity is pretty high
- Better choice for responsive result
- Better user interface

DISADVANTAGES

- Machine Learning model in general is little complex
- Prediction sometime not reliable because the model is build on the old data.
- The prediction result is more depend on the model.

11.CONCLUSION:

Today's fast-growing IT industry needs to discover new technology and update the old technology that helps us to reduce human intervention and increase the efficiency of the work. This model is used for the banking system or anyone who wants to apply for a loan. It will be very helpful in bank management. From the analysis of the data, it is very clear that it reduces all the frauds done at the time of loan approval. Time is also very precious for everyone through this not only the bank but also the waiting time of the applicant will also reduce. As it seems, it will not deal with some special cases when only one parameter is enough for the decision, but it is quite efficient and reliable in some instant.

12.FUTURE SCOPE:

The prediction algorithm can be optimized more to improve o the accuracy. Currently around 91% accuracy is offered by the model but it can be taken ahead and improved to get 100% accuracy by increasing the boosting methods. More parameters can also be used to improve the prediction accuracy and make it more efficient. The biggest flaw in this project is that there is no dynamic updating of the dataset. User inputs can be used to dynamically add data to the dataset. These are the developments that might be done in this project.

13. APPENDIX:

Source Code:

```
import numpy as np
import pickle
import requests

# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
API_KEY = "hmIOFhnjuvRGrJaKtFnyvNKEQTINuL4eRrcnbp6K7c8R"
token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":API_KEY
, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
mltoken = token_response.json()["access_token"]

header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}


app= Flask(__name__, template_folder='templates')

scale = pickle.load(open('scale.pkl','rb'))
```

```
@app.route('/')
def home():
    return render_template('index.html')
@app.route('/predict.html')
def formpg():
    return render_template('predict.html')
@app.route('/submit',methods = ['POST'])
def predict():
    loan_num,gender,married,depend,education,self_emp,applicant_income,co_income,loan_amount,
loan_term,credit_history,property_area = [x for x in request.form.values()]
    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 self_emp == 'Yes':
        self_emp = 1
    else:
        self_emp = 0

    if depend == '3+':
        depend = 3
```

```
    applicant_income = int(applicant_income)
    applicant_income = np.log(applicant_income)
    loan_amount = int(loan_amount)
    loan_amount = np.log(loan_amount)

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

    if property_area == 'Urban':
        property_area = 2

    elif property_area == 'Rural':
        property_area = 0
    else:
        property_area = 1

    features = [[gender,married,depend,education,self_emp,applicant_income,co_income,loan_amount,
loan_term,credit_history,property_area]]
    #con_features = [np.array(features)]
    scale_features = scale.fit_transform(features)
    sf = scale_features.tolist()

    payload_scoring = {"input_data": [{"fields": ['gender','married','depend','education','self_emp','applicant_inco

    response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/
5108313c-f101-4c06-8f87-151aa0d1c3ff/predictions?version=2022-10-26', json=payload_scoring,headers=
{'Authorization': 'Bearer ' + mltoken})
    print("response_scoring")
    prediction = response_scoring.json()
    predict = prediction['predictions'][0]['values'][0][0]
```

```
    #prediction = model.predict(scale_features)
    if predict == 0:
        return render_template('submit.html', prediction_text ='You are eligible for loan')
    else:
        return render_template('submit.html',prediction_text = 'Sorry you are not eligible for loan')


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

GitHub Link:

https://github.com/IBM-EPBL/IBM-Project-5361-1658760261

Project Demo Link:

https://drive.google.com/file/d/13aBC0k5o6SJDpRtAG2qjgaTi-FRdDPS5/view?usp=sharing