A MINI PROJECT REPORT

ON

PREDICTION OF MODERNIZED LOAN APPROVAL USING MACHINE LEARNING

Submitted in the partial fulfillment of the requirements for the award of

BACHELOR OF TECHNOLOGY

IN

CSE (ARTIFICIAL INTELLIGENCE & MACHINE LEARNING)

SUBMITTED

BY

R. RAJA HARSHA - 21BK1A66A7

PRATHAM YADAV - 21BK1A6696

P. HARIKA - 21BK1A6699

UNDER THE ESTEEMED GUIDANCE OF

Mr. GOLLAPUDI PAVAN

Assistant Professor

Department of CSE(AI&ML)

St. Peter's Engineering College, Hyderabad

Giving Wings To Thoughts

DEPARTMENT OF CSE(AI&ML)

St. Peter's Engineering College (UGC Autonomous)

Approved by AICTE, New Delhi, Accredited by NBA and NAAC with 'A' Grade,

Affiliated to JNTU, Hyderabad, Telangana

2021-2025

Giving Wings to Thoughts
ESTD:2007

St. PETER'S ENGINEERING COLLEGE

UGC - AUTONOMOUS

Affiliated to JNTUH, Approved by AICTE, Accredited by NAAC with "A" Grade, NBA Programme Accredited (EEE, CSE, ECE)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING (AI&ML)

CERTIFICATE

This is to certify that Mini Project entitled **"PREDICTION OF MODERNIZED LOAN APPROVAL USING MACHINE LEARNING"**, done by **R. Raja Harsha** (21BK1A667), **Pratham Yadav** (21BK1A6696), **P. Harika** (21BK1A6699) in partial fulfillment for the award of the degree of **Bachelor of Technology in CSE(AI&ML)** is a record of Bonafide work done by them under my supervision during the academic year "2024 – 2025".

INTERNAL GUIDE

Mr. Gollapudi Pavan
Assistant Professor,
Department of CSE(AI&ML)
St. Peter's Engineering College,
Hyderabad

HEAD OF THE DEPARTMENT

Ms. M. Arsha Reddy
Assistant Professor, HOD
Department of CSE(AI&ML)
St. Peter's Engineering College,
Hyderabad

PROJECT COORDINATOR

Dr. J. Lakshmi Prasanna
Assistant Professor,
Department of CSE(AI&ML)
St. Peter's Engineering College,
Hyderabad

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

We sincerely express my deep sense of gratitude to **Mr. Gollapudi Pavan** for his valuable guidance, encouragement and cooperation during all phases of the project.

We greatly indebted to my Project Coordinator **Dr. J. Lakshmi Prasanna** for providing her valuable advice, constructive suggestions and encouragement without whom it would not have been possible to complete this project.

It is a great opportunity to render my sincere thanks to **Ms. M. Arsha Reddy,** Head of the Department, CSE(AI&ML) for her timely guidance and highly interactive attitude which helped us a lot in the successful execution of the Project.

We would like to express our heartfelt gratitude to **Dr. P. Deepan**, Head of the Department for his invaluable guidance, unwavering support, and insightful feedback throughout every phase of our project. His encouragement and expertise provided us with the confidence and clarity needed to bring this project to fruition.

We are extremely thankful to our Principal **Dr. K. Sree Latha,** who stood as an inspiration behind this project and heartfelt for her endorsement and valuable suggestions.

We respect and thank our secretary, **Sri. T. V. Reddy,** for providing us an opportunity to do the project work at **St. Peter's Engineering College** and We are extremely thankful to him for providing such a nice support and guidance which made us to complete the project.

We also acknowledge with a deep sense of reverence, Our gratitude towards our parents, who have always supported us morally as well as economically. We also express gratitude to all my friends who have directly or indirectly helped us to complete this project work. We hope that we can build upon the experience and knowledge that we have gained and make a valuable contribution towards the growth of society in the coming future.

DECLARATION

We hereby declare that the project entitled, "**PREDICTION OF MODERNIZED LOAN APPROVAL USING MACHINE LEARNING**", is the work done during the AY 2024 - 25 and is submitted as project in partial fulfillment for the award of degree of Bachelor of technology in Computer Science Engineering (AI&ML) from St. Peter's Engineering College affiliated to JNTUH.

Rayudu Raja Harsha	**21BK1A66A7**
Pratham Yadav	**21BK1A6696**
Puskur Harika	**21BK1A6699**

INSTITUTE VISION

To be a renowned Educational Institution that molds Students into Skilled Professionals fostering Technological Development, Research and Entrepreneurship meeting the societal needs.

INSTITUTE MISSION

IM1: Making students knowledgeable in the field of core and applied areas of Engineering to innovate Technological solutions to the problems in the Society.

IM2: Training the Students to impart the skills in cutting edge technologies, with the help of relevant stakeholders.

IM3: Fostering conducive ambience that inculcates research attitude, identifying promising fields for entrepreneurship with ethical, moral and social responsibilities.

DEPARTMENT OF CSE (ARTIFICIAL INTELLIGENCE & MACHINE LEARNING)

DEPARTMENT VISION

To improve fundamentals and make it easier to address the ever-expanding needs of society by producing the best leaders in artificial intelligence and machine learning via excellence in research and education.

DEPARTMENT MISSION

DM1: Create centers of excellence in cutting-edge computing and artificial intelligence

DM2: Impart rigorous training to generate knowledge through the state-of-the-art concepts and technologies in Artificial Intelligence and Machine Learning.

DM3: To enhance research in emerging areas by collaborating with industries and institutions at the national and international levels.

PROGRAM OUTCOMES (POs)

PO1: Engineering Knowledge:

Apply the knowledge of mathematics, science, engineering fundamentals and provide solutions in the engineering specialization of artificial intelligence and machine learning.

PO2: Problem Analysis:

Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions:

Design solutions for complex engineering problems and design system components or processes that meet
the specified needs with appropriate consideration for the public health and safety, and the cultural, and environmental considerations

PO4: Conduct investigations of complex problems:

Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage:

Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society:

Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice.

PO7: Environment and sustainability:

Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics:

Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and team work:

Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication:

Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective Presentations, and give and receive clear instructions.

PO11: Project management and finance:

Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary Environments.

PO12: Life-long learning:

Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO1:

Work effectively in an interdisciplinary field with the knowledge of Artificial Intelligence and Machine Learning to develop solutions to real-world problems.

PEO2:

To communicate and work effectively on team-based engineering projects and will practice the ethics of their profession consistent with a sense of social responsibility.

PEO3:

Excel as socially committed engineers or entrepreneurs with high ethical and moral values.

Program Specific Outcomes:

PSO1: Apply fundamental concepts of Artificial Intelligence and Machine Learning to solve multi - disciplinary engineering problems.

PSO2: To communicate and work effectively on team-based engineering projects and will practice the ethics of their profession consistent with a sense of social responsibility.

TABLE OF CONTENTS

ABSTRACT

Technology has significantly enhanced human existence and quality of life, especially in sectors like banking, where automation and machine learning are revolutionizing operations. In the loan approval process, banks use historical data and advanced algorithms to assess applicants' creditworthiness. This allows them to predict whether a loan application will be approved or not based on various factors such as income, credit history, and other financial metrics. Machine learning models, such as logistic regression, random forests, and support vector machines, help improve the accuracy of these predictions.

The modernization of loan approval systems is vital for increasing efficiency, reducing risks, and boosting customer satisfaction in the banking sector. By leveraging machine learning, banks can analyze large datasets, identify key features, and build predictive models that streamline decision-making. Data preprocessing techniques, such as cleaning and feature selection, are essential steps before applying supervised learning algorithms to predict loan approval outcomes. The implementation of such models ensures faster, more accurate assessment compared to traditional methods.

Evaluation metrics like accuracy, precision, recall, and F1 score are used to measure the effectiveness of these models. The study demonstrates that machine learning techniques can accurately predict loan approval outcomes, helping banks make informed decisions and manage risks better. Furthermore, it highlights the challenges and opportunities for future improvements in loan approval systems, paving the way for more advanced and reliable solutions in the financial sector.

LIST OF FIGURES

LIST OF SCREENSHOTS

INDEX

1. INTRODUCTION

1.1 PROJECT SCOPE

In developing a modernized loan approval system using a machine learning approach, the project scope encompasses several critical phases. Initially, the focus will be on defining a clear problem statement: predicting loan approval decisions with enhanced accuracy and efficiency. This involves gathering historical data from various sources, including applicant details, credit scores, financial histories, and past loan outcomes. Subsequently, selecting and implementing appropriate machine learning algorithms—such as logistic regression, decision trees, random forests, and gradient boosting machines—will be crucial.

Comprehensive documentation and reporting will summarize the process, outcomes, and recommendations for stakeholders, ensuring transparency and facilitating informed decision-making. Finally, the project will include deployment strategies, ongoing maintenance plans, and a roadmap for future enhancements based on continuous evaluation and stakeholder feedback. Ethical considerations, including fairness, transparency, and bias mitigation, will be paramount throughout the project, along with adherence to regulatory requirements.

Comprehensive documentation and reporting will summarize the process, outcomes, and recommendations for stakeholders, ensuring transparency and facilitating informed decision-making. Finally, the project will include deployment strategies, ongoing maintenance plans, and a roadmap for future enhancements based on continuous evaluation and stakeholder feedback.

1.2 PROJECT PURPOSE

The purpose of the project "Prediction of Modernized Loan Approval System using Machine Learning Approach" is to fundamentally transform the traditional loan approval process by harnessing the power of advanced machine learning methodologies. This initiative seeks to achieve several key objectives aimed at revolutionizing how loans are assessed and approved:

Firstly, the project aims to significantly enhance operational efficiency within financial institutions. Secondly, the project aims to elevate the accuracy and reliability of loan approval decisions. Traditional systems often rely on simplistic rule-based approaches, which may overlook nuanced patterns and individual applicant characteristics. By leveraging machine learning algorithms such as logistic regression, decision trees, random forests, and gradient boosting machines, the project intends to develop models that can analyze vast amounts of historical data—including applicant demographics, credit history, income levels, and other relevant factors—to predict loan outcomes with a higher degree of precision.

1.3 PROJECT FEATURES

The project "Prediction of Modernized Loan Approval System using Machine Learning Approach" incorporates several key features aimed at transforming the conventional loan approval process. Central to the project is the utilization of advanced machine learning techniques to enhance various aspects of the lending workflow.

Firstly, the project focuses on data-driven decision-making by leveraging historical loan data, applicant information, and relevant financial metrics. This includes extensive data preprocessing to ensure data quality and feature engineering to extract meaningful predictors that influence loan approval decisions.

Secondly, a range of machine learning algorithms such as logistic regression, decision trees, random forests, and gradient boosting machines are implemented to build predictive models. Moreover, the project emphasizes scalability and adaptability by designing a framework capable of handling large volumes of loan applications while maintaining performance standards.

2. LITERATURE SURVEY

The literature on loan approval systems reveals that traditional methods rely heavily on rule-based evaluations and manual reviews, which are often inefficient and lack precision. Although machine learning has been explored in credit risk assessment, there is a notable gap in comprehensive predictive models that analyze diverse applicant data—such as demographics, credit history, and financial details—for loan approvals. Current systems often miss nuanced patterns that machine learning could identify, leading to subjective decisions and potential biases. This project addresses these limitations by employing advanced ML algorithms like logistic regression, decision trees, and random forests to build a robust model aimed at automating and improving the accuracy of loan approval decisions, making the process more efficient and data driven.

2.1 EXISTING SYSTEM

- The existing system for loan approval in the banking sector predominantly relies on traditional methods, including manual reviews and basic rule-based assessments.

1. **Application Review:** Loan approval begins with a review of the applicant's credit history, financial background, and other demographic factors by a banking representative. This process is often time-consuming and lacks standardization, which can lead to subjective decision-making.
2. **Financial Data Assessment:** The applicant's financial stability, income level, and debt-to-income ratio is analyzed using basic financial formulas and rules. However, these assessments are limited in their ability to handle large datasets or identify nuanced patterns in the applicant's financial profile.
3. **Credit Score Analysis:** A critical component of loan approval involves evaluating the applicant's credit score, which is often obtained from credit bureaus. While useful, this score alone does not provide a complete view of an applicant's creditworthiness.

4. **Manual Decision-making:** After analyzing the application data, loan officers make decisions manually, which introduces a risk of human error and bias, potentially affecting the fairness and accuracy of approvals.
5. **Limitations:** Traditional loan approval systems lack predictive accuracy and scalability. They are limited in processing large volumes of applications efficiently and cannot adapt to emerging data patterns, making it challenging to meet the demands of a modernized, high-volume financial environment.

2.1 DISADVANTAGES OF EXISTING SYSTEM

- **Subjectivity in application assessments**: Loan approval decisions are often influenced by the subjective judgment of loan officers, leading to inconsistencies.
- **Reliance on limited financial indicators**: The process relies heavily on specific financial metrics, which may not capture an applicant's full financial health.
- **Potential for misinterpretation or bias**: Manual assessments can lead to varying interpretations of applicant data, resulting in biased decisions.
- **Insufficient use of available data**: Traditional systems do not fully utilize all relevant applicant data, such as spending patterns or alternate financial behaviors.
- **Incomplete integration of credit and financial history**: Important parameters like detailed credit history and income trends may not be fully considered, leading to potential inaccuracies in loan approval decisions.

2.3 PROPOSED SYSTEM

- Leverage machine learning techniques for analyzing comprehensive applicant data.
- Data includes demographics, credit history, income, and financial behavior.
- Identify patterns and features within the data for loan approval indicators.
- Patterns serve as robust indicators of an applicant's creditworthiness.
- Create a predictive model that accurately assesses loan approval likelihood.
- Model’s accuracy is tailored to applicant-specific financial data.

2.4 ADVANTAGES OF PROPOSED SYSTEM

The proposed system for loan approval prediction leverages machine learning to enhance decision-making:

1. **Improved Risk Assessment**: The system can accurately assess loan approval likelihood, identifying high-risk applications early to minimize potential defaults.

2. **Personalized Loan Offers**: By analyzing detailed applicant data, the system can tailor loan terms to individual financial profiles, enhancing loan suitability and approval rates.

3. **Increased Efficiency**: Machine learning algorithms process large datasets swiftly, streamlining loan evaluations and significantly reducing manual review times.

4. **Cost Savings**: By automating assessments and improving decision accuracy, the system reduces operational costs, allowing banks to allocate resources more effectively

3.ANALYSIS

3.1 PROBLEM DEFINITION

In the banking sector, the loan approval process is often time-consuming, resource-intensive, and prone to human error and subjectivity. Traditional methods rely on manual reviews and limited rule-based assessments, which lack the precision to handle large volumes of applications or identify complex patterns in applicant data. This can lead to inconsistent decisions, potential biases, and financial losses due to inaccurate assessments of applicants' creditworthiness. The primary objective of this project is to develop a predictive model using machine learning to automate and improve the accuracy of loan approval decisions. By analyzing comprehensive applicant data, including credit history, income, demographics, and financial behaviors, the model will provide a more reliable, efficient, and data-driven approach to loan assessment, ultimately reducing default risks and enhancing the overall efficiency of the loan approval process.

3.2 REQUIREMENTS:

3.2.1 HARDWARE REQUIREMENTS:

Hardware interfaces specify the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

➢ **H/W System Configuration: -**

- **Processor** **:** i5 or above
- **RAM** **:** 4GB (min)
- **Hard Disk** **:** 20 GB
- **Keyboard** **:** Standard Windows Keyboard
- **Mouse** **:** Two or Three Button Mouse
- **Monitor** **:** SVGA

3.2.2 SOFTWARE REQUIREMENTS:

Software Requirements specify the logical characteristics of each interface and software components of the system. The following are some software requirements

- OPERATING SYSTEM : Windows 7 Ultimate.
- CODE LANGUAGE : Python
- LIBRARIES : Scikit-Learn, TensorFlow, Pandas and NumPy
- FRONT-END : Python
- BACK-END : Django-ORM
- DESIGNING : HTML, CSS, JavaScript
- DATABASE : MySQL
- WEB SERVER : WAMP Server

4.METHODS AND TECHNOLOGIES INVOLVED

Developing a machine-learning solution for automated loan approval prediction involves several methods and technologies to ensure model effectiveness and reliability.

1. Data Collection and Preprocessing:

a. **Data Sources:** Collect applicant data from banking records, credit bureaus, and demographic databases, including financial history, credit score, income, and spending patterns.
b. **Data Cleaning:** Clean and preprocess data to handle missing values, outliers, and ensure data consistency, crucial for accurate predictions.
c. **Feature Engineering:** Extract key features such as income-to-debt ratio, payment history, and employment status, which are indicative of an applicant's creditworthiness.

2. Machine Learning Models:

a. **Classification Models:** Use algorithms like Logistic Regression, Random Forest, Support Vector Machines (SVM), and Gradient Boosting to classify loan applications as approved or rejected.
b. **Ensemble Methods:** Combine multiple model predictions for better accuracy and resilience, using methods like bagging and boosting to reduce bias and variance

3. Feature Selection and Dimensionality Reduction:

a. Apply feature selection (e.g., Recursive Feature Elimination) and dimensionality reduction (e.g., Principal Component Analysis) to identify the most relevant attributes, optimizing model complexity and performance.

4. Model Evaluation:

a. Use evaluation metrics such as accuracy, precision, recall, F1-score, and ROC-AUC to assess model performance.
b. Implement cross-validation to ensure model robustness and prevent overfitting, especially critical in high-stakes financial applications.

5. Bias Mitigation:

a. Address potential biases in the data, such as socioeconomic disparities, using fairness-aware machine learning techniques to ensure equitable loan approval decisions.

6. Data Privacy and Security:

a. Implement data privacy measures in accordance with regulatory standards to protect applicant information, following guidelines like GDPR to ensure secure data handling.

7. Model Interpretability:

a. Uses interpretability techniques such as SHAP (Shapley Additive explanations) values or LIME (Local Interpretable Model-Agnostic Explanations) to make model decisions transparent and explainable for compliance and trustworthiness.

8. Deployment:

a. Deploy the final model in a scalable, cloud-based platform for real-time or batch processing to handle high application volumes securely.

9. Continuous Improvement:

a. Regularly monitor and update the model as new data becomes available or as financial trends shift, ensuring that it remains accurate and relevant.

10. Regulatory Compliance:

a. Ensure compliance with financial regulations and standards for AI in banking, such as those from the Federal Reserve or the European Banking Authority, to maintain ethical and lawful model usage.

11.Collaboration with Healthcare Professionals:

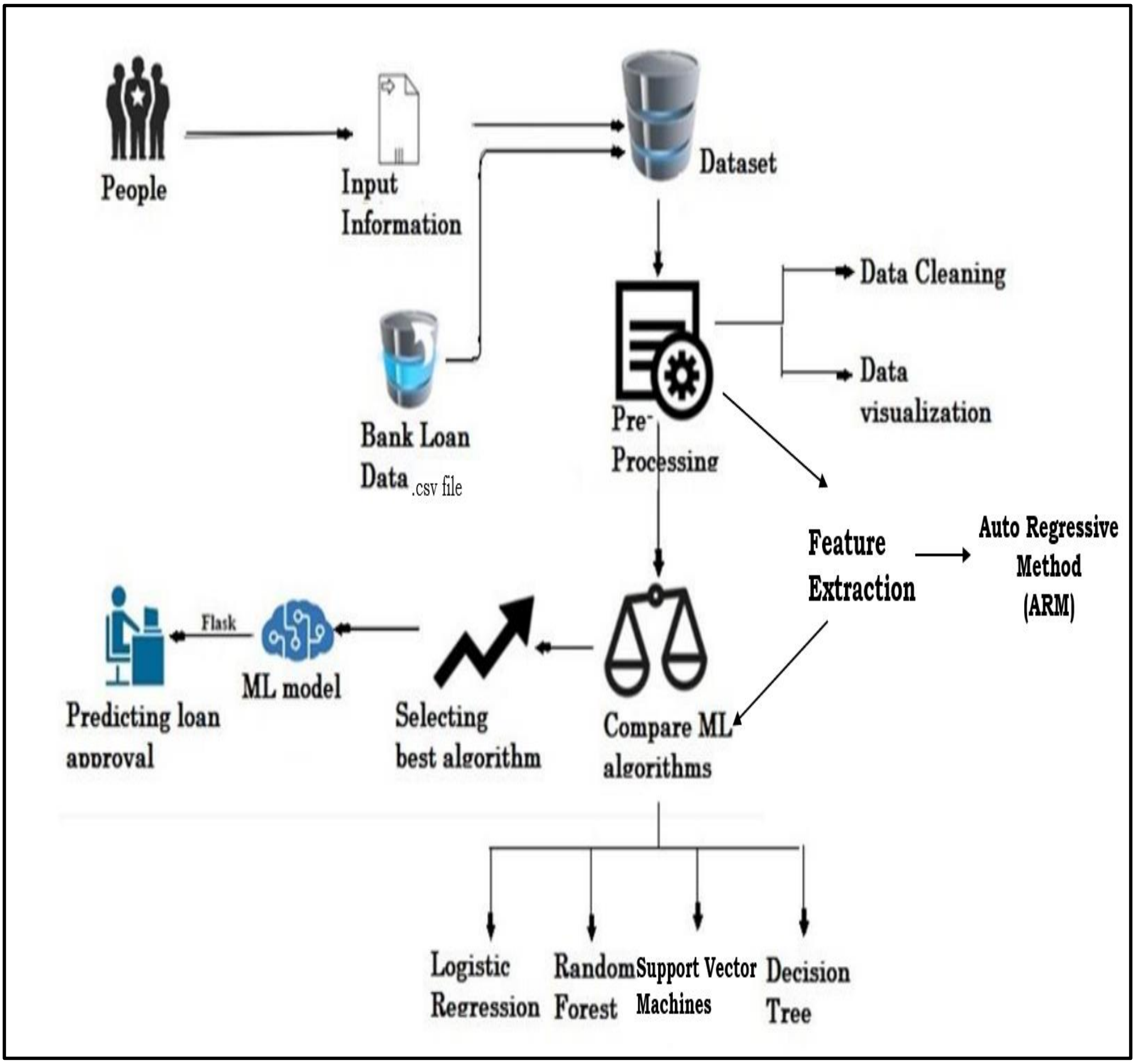
a. Collaborate closely with banking and financial professionals to validate the model's predictions and integrate it effectively into existing loan approval workflows.

12.Patient Education and Informed Consent:

a. Develop materials to inform applicants about the use of their data in the model, including obtaining informed consent for data use in loan processing.

a. DESIGN

i. SYSTEM ARCHITECTURE



4.1 SYSTEM ARCHITECTURE

Data Collection: This section encompasses data sources, including user information, loan application data, and prediction results. The web database serves as a repository for these datasets, facilitating storage and retrieval for processing queries.

Data Preprocessing: Data preprocessing steps involve managing the storage of trained models and tested accuracy results. This includes accessing and preparing the data for analysis, ensuring that it is ready for model evaluation and prediction tasks.

Machine Learning: This section includes the training and evaluation of machine learning models based on the preprocessed loan application data. Models are trained to predict loan approvals, and their performance is assessed through accuracy metrics and analysis of various prediction types (e.g., approved vs. denied).

Prediction: The prediction engine utilizes the trained machine learning models to evaluate loan applications submitted by users. The results of these predictions are stored in the database and can be accessed for further analysis, including viewing performance metrics through bar charts and downloading datasets of prediction results.

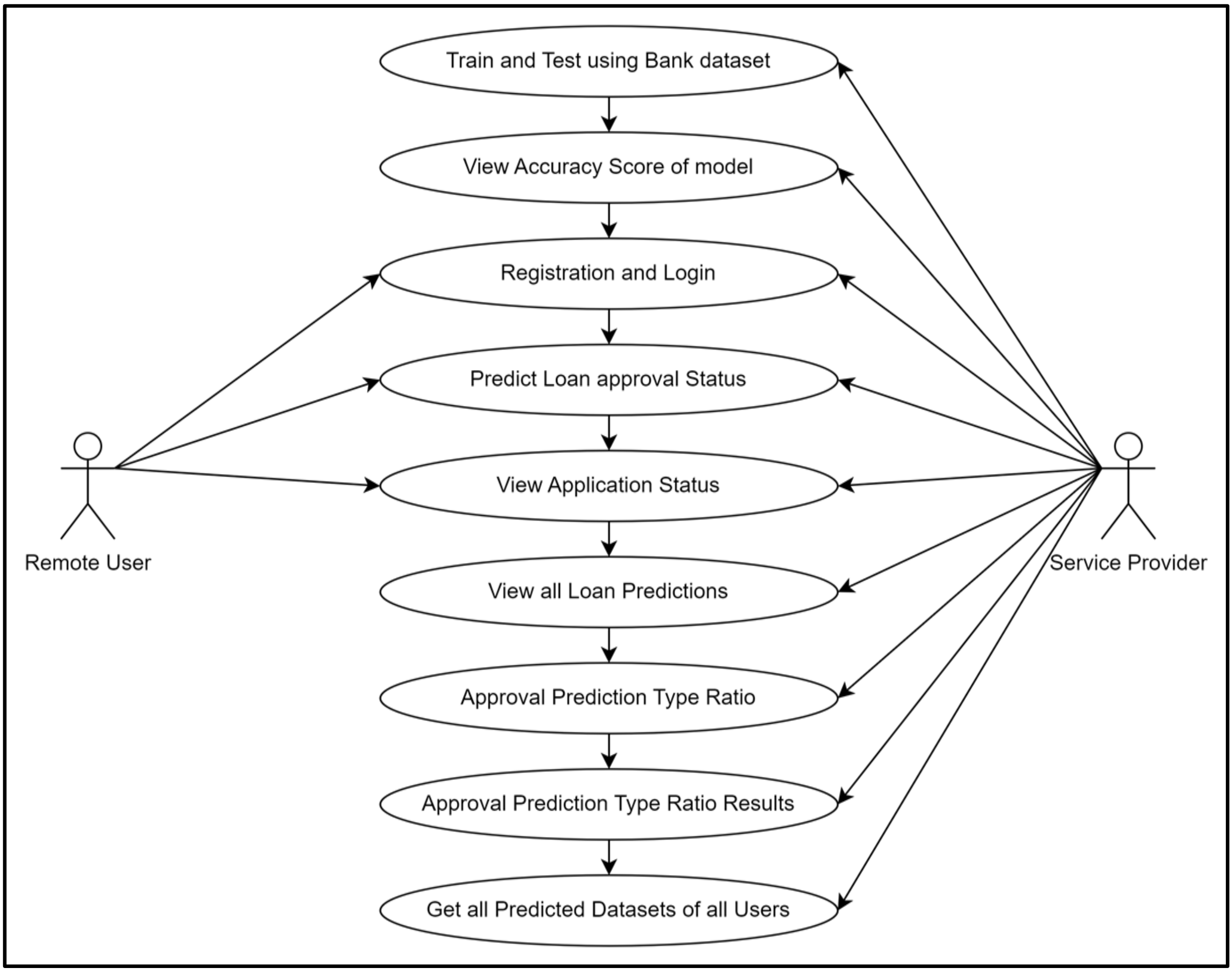
User Interaction: Remote users can register, log in, submit loan applications, and view their profiles and prediction results. The web server processes these requests, interacting with the web database to access, store, and retrieve data as needed

Result Display and Analysis: The system provides functionalities for displaying prediction results and accuracy metrics. Users can analyze prediction distributions, view detailed accuracy metrics, and download prediction datasets for further exploration.

Potential Enhancements: Future improvements may include implementing load balancing for scalability, ensuring secure communication channels, optimizing database queries for performance, enhancing the user interface, and continuously updating prediction models based on new data to improve overall accuracy.

ii. USE CASE DIAGRAM

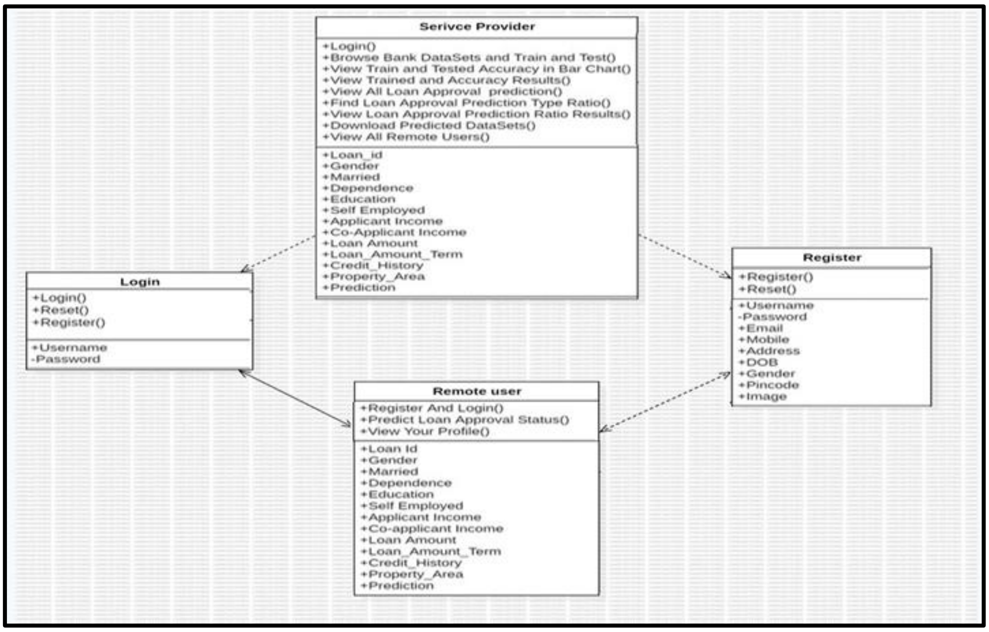
In the Use case diagram, we have basically one actor who is the user in the trained model. A use case diagram is a graphical depiction of a user's possible interactions with a system. A Use case diagram shows various use cases and different types of users the system has. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.



4.2 USE CASE DIAGRAM

iii. CLASS DIAGRAM

Class diagram is a type of static structure diagram that describes the structure of a system by showing the system’s classes, their attributes, operations (or methods), and the relationships among objects.



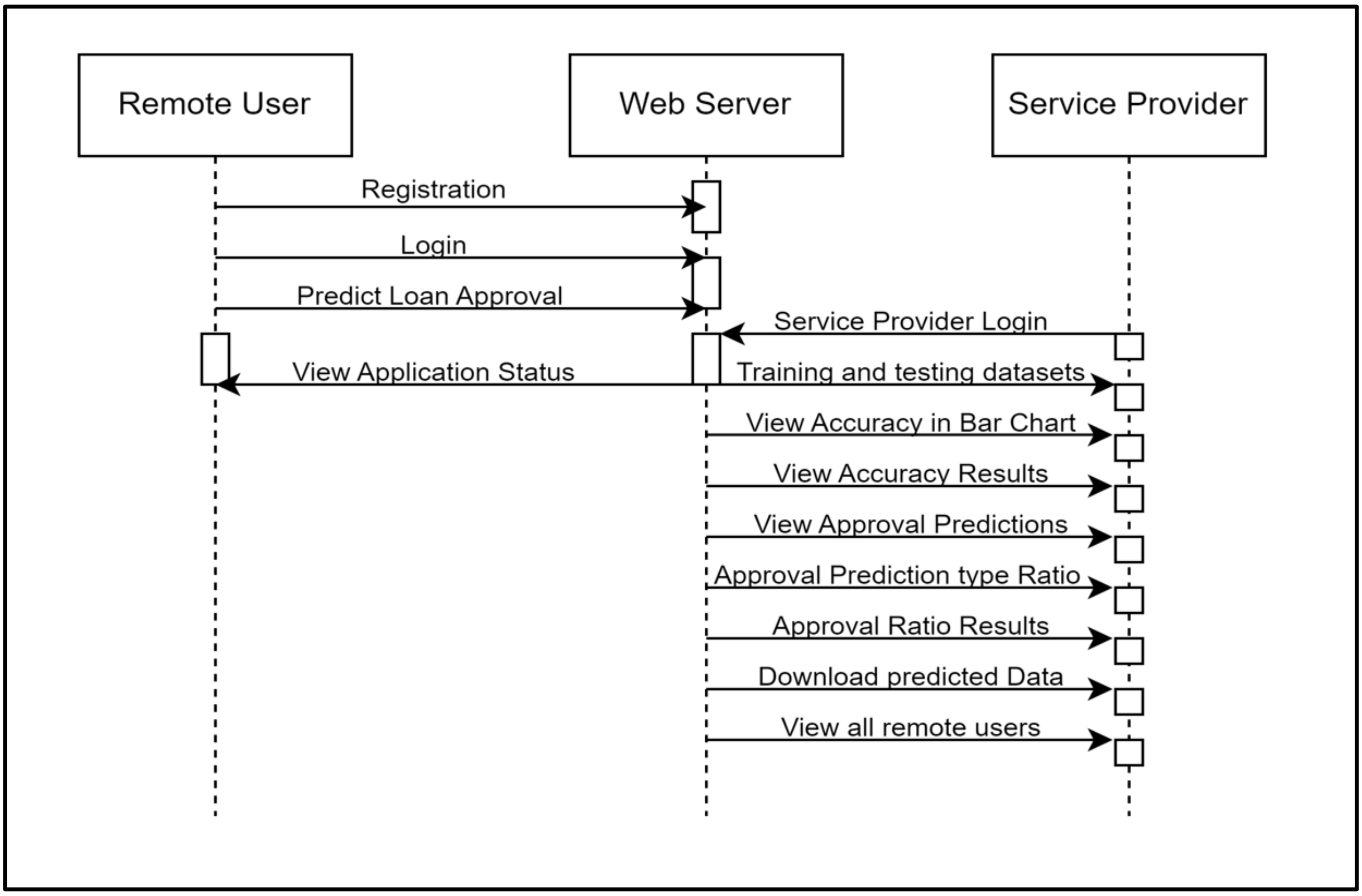
4.3 CLASS DIAGRAM

DESCRIPTION:

The UML class diagram for the modernized loan approval system using machine learning outlines the primary components and their interactions. The system includes four main classes: Service Provider, Login, Register, and Remote User. The **Service Provider** class has attributes such as Loan_id, Gender, Married, Dependence, Education, Self Employed, Applicant Income, Co-Applicant Income, Loan Amount, Loan_Amount_Term, Credit_History, Property_Area, and Prediction. It includes methods for logging in, browsing and training datasets, viewing accuracy results, predicting loan approvals, analyzing prediction ratios, downloading datasets, and viewing all remote users. The **Login** class contains attributes like Username and Password, with methods for logging in, resetting the password, and registering. The **Register** class includes attributes such as Username, Password, Email, Mobile, Address, DOB, Gender, Pin code, and Image, with methods for user registration and password reset. The **Remote User** class shares many attributes with the Service Provider, including Loan_id, Gender, Married, Dependence, Education, Self Employed, Applicant Income, Co-Applicant Income, Loan Amount, Loan_Amount_Term, Credit_History, Property_Area, and Prediction. It has methods for user registration and login, predicting lon approval status, and viewing user profiles. The diagram illustrates that the Service Provider interacts with the Login class for authentication purposes. The Register class connects both the Login and Remote User classes to facilitate user registration. The Remote User class allows users to interact with the system to predict loan approval status and view their profiles. This class diagram provides a comprehensive overview of the system's structure and functionality, highlighting the key entities and their interactions within the loan approval system.

iv. SEQUENCE DIAGRAM

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development.



4.4 SEQUENCE DIAGRAM

5.IMPLEMENTATION

5.1Technologies Used:

SUPPORT VECTOR MACHINE

Using Support Vector Machines (SVM) for the loan approval prediction project involve leveraging SVM's capabilities to classify loan applications as approved or rejected based on applicant and loan features. The process starts with data preparation, where applicant demographics (age, gender, marital status), financial information (income, employment status, existing debts, credit score), and loan details (amount, term, purpose) are selected as features. The data is cleaned to handle missing values and ensure consistency, and features are normalized to improve SVM performance. Categorical variables are converted into numerical values using techniques like One-Hot Encoding. For model training, an appropriate kernel such as linear, polynomial, or radial basis function (RBF) is chosen, with hyperparameters like the regularization parameter (C) and kernel coefficient (gamma for RBF) optimized through Grid Search or Random Search with cross-validation. The SVM model is then trained to find the hyperplane that best separates loan approvals from rejections. Model evaluation involves using metrics like Accuracy, Precision, Recall, F1-Score, and ROC-AUC, and analyzing the confusion matrix and cross-validation results to ensure generalization.

LOGISTIC REGRESSION

Using Logistic Regression for the loan approval prediction project involves building a model to classify loan applications as approved or rejected based on various features. The process begins with data preparation, selecting features such as applicant demographics (age, gender, marital status), financial information (income, employment status, existing debts, credit score), and loan details (amount, term, purpose). The data is cleaned to handle missing values and ensure consistency, and features are normalized to improve model performance. Categorical variables are converted into numerical values using techniques like One-Hot Encoding. The Logistic Regression model is then trained on the prepared data, where it estimates the probability of loan approval by fitting a logistic function to the input features. Hyperparameter tuning, such as adjusting the regularization parameter (C), is performed using techniques like Grid Search or Random Search with cross-validation to optimize model performance. Model evaluation involves using metrics like Accuracy, Precision, Recall, F1- Score, and ROC-AUC, and analyzing the confusion matrix and cross-validation results to ensure generalization.

RANDOM FOREST CLASSIFIER

Using a Random Forest Classifier for the loan approval prediction project involves applying an ensemble learning method that combines multiple decision trees to improve classification accuracy and robustness. The process begins with data preparation, where features such as applicant demographics (age, gender, marital status), financial information (income, employment status, existing debts, credit score), and loan details (amount, term, purpose) are selected. The data is then cleaned to address missing values and ensure consistency, and features are normalized if necessary. Categorical variables are converted into numerical values using One-Hot Encoding.

The Random Forest Classifier is trained on the prepared data, constructing multiple decision trees during training and aggregating their predictions to make a final decision. Each tree is built on a random subset of data and features, which helps in capturing various patterns and reducing overfitting. Hyperparameter tuning, including the number of trees (n_estimators) and the maximum depth of each tree (max_depth), is performed using techniques like Grid Search or Random Search with cross-validation to find the optimal model configuration.

DECISION TREES

Using Decision Trees for the loan approval prediction project involves creating a model that makes decisions based on the features of loan applications by splitting the data into subsets based on feature values. The process starts with data preparation, where features such as applicant demograp hics (age, gender, marital status), financial information (income, employment status, existing debts, credit score), and loan details (amount, term, purpose) are selected. The data is cleaned to handle missing values and ensure consistency, and categorical variables are converted into numerical values using techniques like One-Hot Encoding.

The Decision Tree model is then trained on this prepared data. Decision Trees operate by recursively splitting the data into subsets based on feature values, creating a tree-like structure where each node represents a decision rule and each leaf represents an outcome (approved or rejected).

DATASET

Our dataset, sourced from the GitHub website, encompasses a diverse range of attributes aimed at discerning the authenticity of user profiles, comprising 615 instances. https://github.com/Rayudurajaharsha/mini-proj/blob/main/Prediction_of_modernized_loan_approval-main/Bank_Dataset.csv For the loan approval prediction project, the data comprises several key features. Applicant demographics include age, gender, and marital status, which provide insights into the applicant's life stage, potential financial stability, and personal circumstances. Financial information features include income, employment status, existing debts, and credit score, which are crucial for assessing the applicant's ability to repay the loan and their overall financial health. Loan details include the loan amount, loan term, and the purpose of the loan, which affect the risk and the decision to approve or reject the application. The target variable is the loan approval status, a binary indicator of whether the loan was approved or rejected. Additional features might include residence type, education level, and number of dependents, which further provide context on the applicant's financial situation and stability. Data preparation involves handling missing values, addressing outliers, and converting categorical variables into numerical formats to ensure the data is clean and suitable for machine learning.

5.2 Source Code:

```
# Import necessary libraries

from django.db.models import Count, Avg from
django.shortcuts import render, redirect from
django.db.models import Count
from django.db.models import Q
import datetime
import xlwtfrom django.http import HttpResponse
import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from sklearn.ensemble import VotingClassifier
from sklearn.tree import DecisionTreeClassifier



# Create your views here.
FromRemote_User.modelsimport

ClientRegister_Model,Loan_Approval_Prediction,detection_ratio,detection_accuracy



def serviceproviderlogin(request): if
    request.method == "POST":
    admin = request.POST.get('username')
    password = request.POST.get('password')
        if admin == "Admin" and password =="Admin":
            return redirect('View_Remote_Users')

return render(request,'SProvider/serviceproviderlogin.html')



def viewtreandingquestions(request,chart_type):
    dd = {}
pos,neu,neg =0,0,0
poss=None


topic=Loan_Approval_Prediction.objects.values('ratings').annotate(dcount=Count('ratings')).order_by(
```

```
'-dcount')
    for t in topic:
      topics=t['ratings']


pos_count=Loan_Approval_Prediction.objects.filter(topics=topics).values('names').annotate(topiccoun
t=Count('ratings'))

    poss=pos_count

      for pp in pos_count:
        senti= pp['names']
        if senti == 'positive': pos=
          pp['topiccount']
        elif senti == 'negative': neg =
          pp['topiccount']

      elif senti == 'nutral':
      neu = pp['topiccount'] dd[topics]=[pos,neg,neu]

  return
render(request,'SProvider/viewtreandingquestions.html',{'object':topic,'dd':dd,'chart_type':chart_type}
)


  def View_All_Loan_Approval_Prediction(request):
    obj = Loan_Approval_Prediction.objects.all()

  return render(request, 'SProvider/View_All_Loan_Approval_Prediction.html', {'objs': obj})



  def Find_Loan_Approval_Type_Ratio(request):
    detection_ratio.objects.all().delete()

  ratio = ""

  kword = 'Not Approved'
  print(kword)
  obj = Loan_Approval_Prediction.objects.all().filter(Q(Prediction=kword))
  obj1 = Loan_Approval_Prediction.objects.all()
  count = obj.count(); count1 =
  obj1.count();
  ratio = (count / count1) * 100 if
  ratio != 0:
    detection_ratio.objects.create(names=kword, ratio=ratio) ratio1
```

```
 kword1 = 'Approved' print(kword1)
   obj1 = Loan_Approval_Prediction.objects.all().filter(Q(Prediction=kword1))
   obj11 = Loan_Approval_Prediction.objects.all()
   count1 = obj1.count(); count11 =
   obj11.count();
   ratio1 = (count1 / count11) * 100 if
   ratio1 != 0:
      detection_ratio.objects.create(names=kword1, ratio=ratio1) obj
   = detection_ratio.objects.all()

   return render(request, 'SProvider/Find_Loan_Approval_Type_Ratio.html', {'objs': obj})



   def View_Remote_Users(request):
      obj=ClientRegister_Model.objects.all()
   return render(request,'SProvider/View_Remote_Users.html',{'objects':obj})


def ViewTrendings(request):
topic=Loan_Approval_Prediction.objects.values('topics').annotate(dcount=Count('topics')).order_by('-
dcount')
   return render(request,'SProvider/ViewTrendings.html',{'objects':topic})


   def negativechart(request,chart_type): dd
      = {}
   pos, neu, neg = 0, 0, 0

   poss = None topic=
Loan_Approval_Prediction.objects.values('ratings').annotate(dcount=Count('ratings')).order_by('-
dcount')

   for t in topic:

      topics = t['ratings']


pos_count=Loan_Approval_Prediction.objects.filter(topics=topics).values('names').annotate(topiccoun
t=Count('ratings'))

      poss = pos_count for pp in
      pos_count:
         senti = pp['names'] if senti
         == 'positive':
            pos = pp['topiccount'] elif
```

```
        senti == 'negative':
            neg = pp['topiccount'] elif
        senti == 'nutral':
            neu = pp['topiccount']
    dd[topics] = [pos, neg, neu]

return render(request,'SProvider/negativechart.html',{'object':topic,'dd':dd,'chart_type':chart_type})



def charts(request,chart_type):

    chart1 = detection_ratio.objects.values('names').annotate(dcount=Avg('ratio'))
    return render(request,"SProvider/charts.html", {'form':chart1, 'chart_type':chart_type})


def charts1(request,chart_type):

    chart1 = detection_accuracy.objects.values('names').annotate(dcount=Avg('ratio')) return
    render(request,"SProvider/charts1.html", {'form':chart1, 'chart_type':chart_type})


def likeschart(request,like_chart):

    charts =detection_accuracy.objects.values('names').annotate(dcount=Avg('ratio'))
    return render(request,"SProvider/likeschart.html", {'form':charts, 'like_chart':like_chart})


def likeschart1(request,like_chart):

    charts =detection_ratio.objects.values('names').annotate(dcount=Avg('ratio'))
    return render(request,"SProvider/likeschart1.html", {'form':charts, 'like_chart':like_chart})

    def Download_Trained_DataSets(request):
    response = HttpResponse(content_type='application/ms-excel') #
    decide file name
    response['Content-Disposition'] = 'attachment; filename="TrainedData.xls"' #
    creating workbook
    wb = xlwt.Workbook(encoding='utf-8') #
    adding sheet
    ws = wb.add_sheet("sheet1") #
    Sheet header, first row


    row_num = 0
    font_style = xlwt.XFStyle() #
    headers are bold
```

```
font_style.font.bold = True
# writer = csv.writer(response)
obj = Loan_Approval_Prediction.objects.all() data
= obj # dummy method to fetch data. for my_row
in data:

    row_num = row_num + 1

    ws.write(row_num, 0, my_row.Loan_ID, font_style)
    ws.write(row_num, 1, my_row.Gender, font_style)
    ws.write(row_num, 2, my_row.Married, font_style)
    ws.write(row_num, 3, my_row.Dependents, font_style)
    ws.write(row_num, 4, my_row.Education, font_style)
    ws.write(row_num, 5, my_row.Self_Employed, font_style)
    ws.write(row_num, 6, my_row.ApplicantIncome, font_style)
    ws.write(row_num, 7, my_row.CoapplicantIncome, font_style)
    ws.write(row_num, 8, my_row.LoanAmount, font_style)
    ws.write(row_num, 9, my_row.Loan_Amount_Term, font_style)
    ws.write(row_num, 10, my_row.Credit_History, font_style)
    ws.write(row_num, 11, my_row.Property_Area, font_style)
    ws.write(row_num, 12, my_row.Prediction, font_style)
wb.save(response) return
response


def Train_Test_DataSets(request):
detection_accuracy.objects.all().delete()
    df = pd.read_csv("Bank_Dataset.csv")
    df['label'] = df['Loan_Status'].map({'N': 0, 'Y': 1})
    df['LoanAppId'] = df['Loan_ID']
    # df.drop(['Loan_ID','Loan_Status'],axis=1,inplace=True) X =
    df['LoanAppId']
    y = df['label']
    print(X) print(y)


    from sklearn.feature_extraction.text import CountVectorizer

    cv = CountVectorizer(lowercase=False, strip_accents='unicode', ngram_range=(1, 1))
    X = cv.fit_transform(X)
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, random_state=42)
    predictors = []

    print("SVM")
```

SVM Model

```
from sklearn import svm

lin_clf = svm.LinearSVC()
lin_clf.fit(X_train, y_train) predict_svm =
lin_clf.predict(X_test)
svm_acc = accuracy_score(y_test, predict_svm) * 100
print(svm_acc)
from sklearn.metrics import confusion_matrix, f1_score
print(confusion_matrix(y_test, predict_svm))
print(classification_report(y_test, predict_svm))
predictors.append(('svm', lin_clf))
detection_accuracy.objects.create(names="SVM", ratio=svm_acc)


# Logistic Regression Model
print("Logistic Regression")
from sklearn.linear_model import LogisticRegression
logreg = LogisticRegression(random_state=42)
logreg.fit(X_train, y_train)

predict_log = logreg.predict(X_test)

logistic = accuracy_score(y_test, predict_log) * 100
print(logistic)
from sklearn.metrics import confusion_matrix, f1_score
print(confusion_matrix(y_test, predict_log))
print(classification_report(y_test, predict_log))
predictors.append(('logistic', logreg))

detection_accuracy.objects.create(names="Logistic Regression", ratio=logistic)



# Decision Tree Classifier
print("Decision Tree Classifier") dtc =
DecisionTreeClassifier()
dtc.fit(X_train, y_train) dtcpredict =
dtc.predict(X_test)
print("ACCURACY")
print(accuracy_score(y_test, dtcpredict) * 100)
print("CLASSIFICATION REPORT")
print(classification_report(y_test, dtcpredict))
print("CONFUSION MATRIX")
print(confusion_matrix(y_test, dtcpredict))
```

```
predictors.append(('DecisionTreeClassifier', dtc))

detection_accuracy.objects.create(names="Decision Tree Classifier", ratio=accuracy_score(y_test,
dtcpredict) * 100)



# Random Forest Classifier
print("Random Forest Classifier")
from sklearn.ensemble import RandomForestClassifier
RFC = RandomForestClassifier(random_state=0)
RFC.fit(X_train, y_train)

pred_rfc = RFC.predict(X_test)



RFC.score(X_test, y_test)
print("ACCURACY")
print(accuracy_score(y_test, pred_rfc) * 100)
print("CLASSIFICATION REPORT")
print(classification_report(y_test, pred_rfc))
print("CONFUSION MATRIX")
print(confusion_matrix(y_test, pred_rfc))
predictors.append(('RandomForestClassifier', RFC))
detection_accuracy.objects.create(names="Random Forest Classifier", ratio=accuracy_score(y_test,
pred_rfc) * 100)

obj = detection_accuracy.objects.all()



return render(request,'SProvider/Train_Test_DataSets.html', {'objs': obj})
```

5.3 RESULT ANALYSIS

The result analysis of a modernized loan approval system using machine learning evaluates key performance metrics such as accuracy, precision, recall, F1 score, and AUC-ROC to understand the model's effectiveness. With an accuracy of 88.3%, the model demonstrates good overall performance, but there's room for improvement. A precision of 76.62% and recall of 88.2% indicate that while the model effectively identifies approved loans, it has a moderate rate of false positives.

The F1 score of 88% reflects a balanced consideration of precision and recall. A high AUC-ROC value would confirm the model's ability to distinguish between approved and rejected loans. Further considerations include addressing class imbalances, analyzing feature importance, comparing different models, and using cross-validation to ensure robust performance metrics. Overall, this analysis highlights the model's strengths and areas for refinement to ensure reliable loan approval predictions

4.5 RESULT ANALYSIS

6.TESTING AND VALIDATION

6.1 INTRODUCTION:

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

6.2 TYPES OF TESTING

UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .It is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of component.

FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid input: identified classes of valid input must be accepted.
Invalid input: identified classes of invalid input must be rejected.
Functions: identified functions must be exercised.
Output: identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases

6.3 TEST CASES

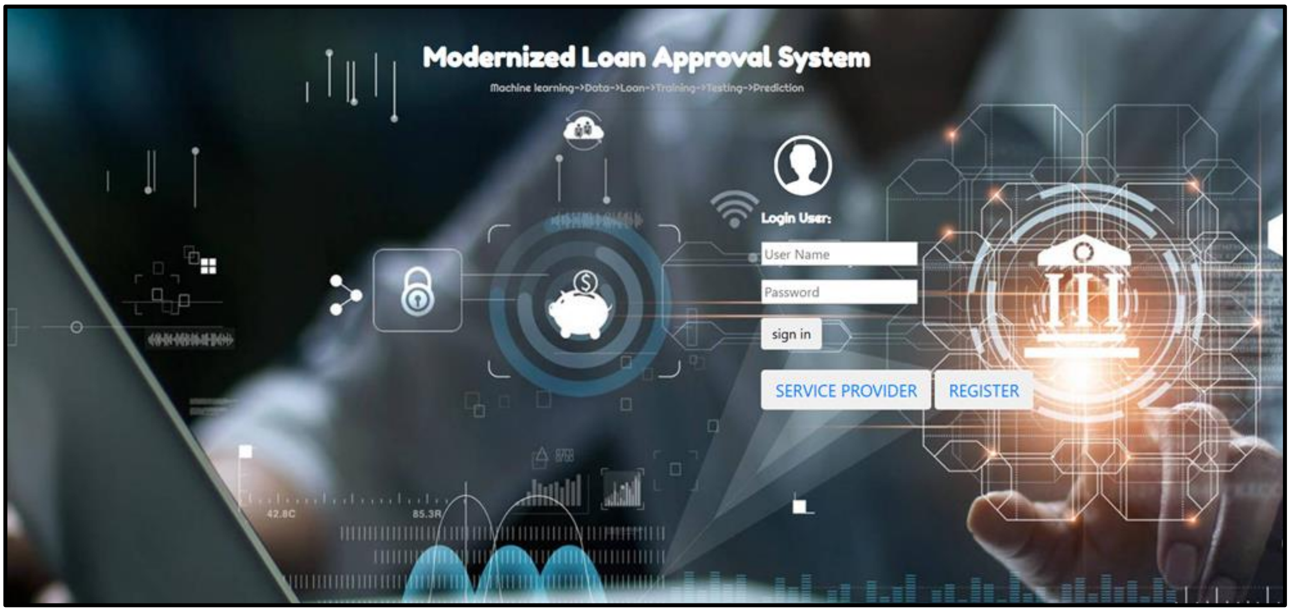
Test Case ID	Test Case Name	Input	Expected output	Actual Output	Test Case Pass/Fail
1	User credentials	Username: dhanya Password : dhanya@123	It should move to user home page	It moves to the user home page	Pass
2	Check Username	Username: XYZ (Which is invalid)	It shows the error The username is not available	It shows the error The username is not available	Pass
3	Creating an account	Username: hello (if username is already taken)	Gives the error Username already exists	Gives the error that username already exists	Pass
4	registration	Mail ID (Already exists)	Shows the message Account exists with the given Mail ID. Try login	Shows the message Account exists with the given Mail ID. Try login	pass
5	Registration details	Invalid Phone number (more than 10 numbers)	Gives the message "Invalid Details"	Gives the message "Invalid Details"	Pass

4.6 TEST CASES

7.RESULTS

These are the execution screenshots of project implementation

1.Login Page :



7.1 LOGIN PAGE

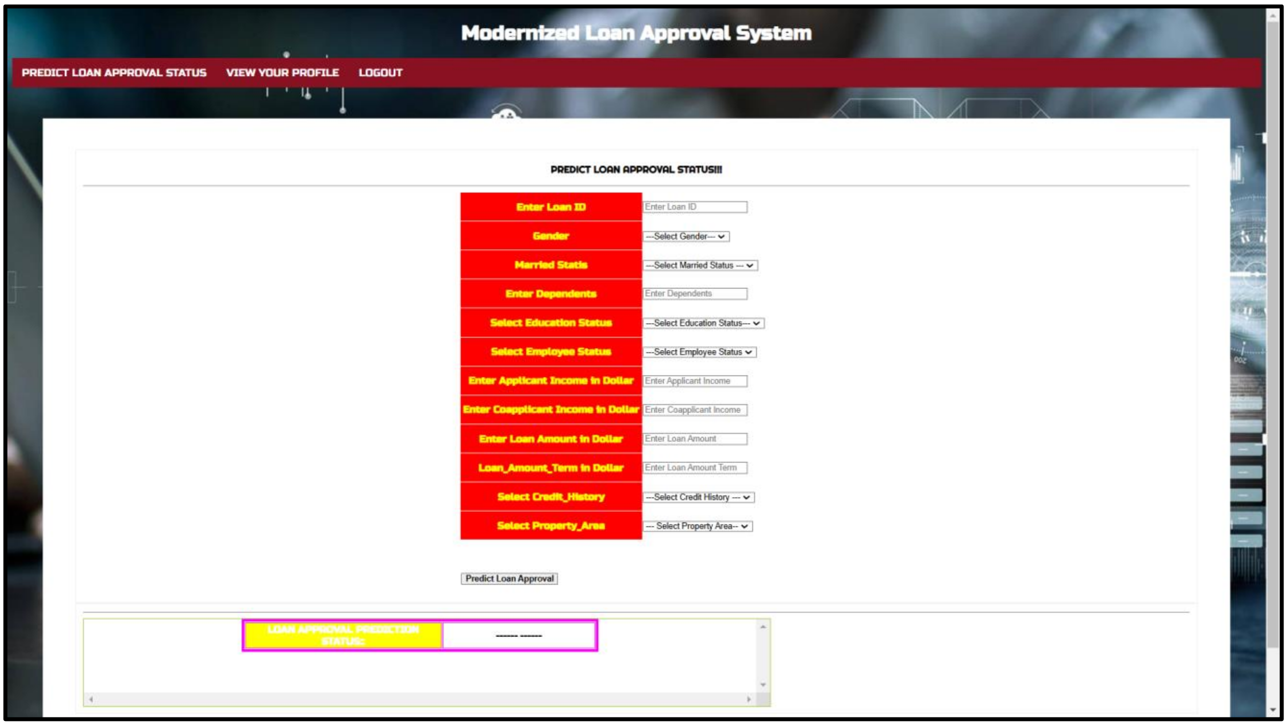
2. Service Provider Login:

7.2 SERVICE PROVIDER LOGIN

3.User Register Page:

7.3 USER RIGISTER PAGE

4.Predict loan approval Status page:



7.4 PREDICT LOAN APPROVAL STATUS PAGE

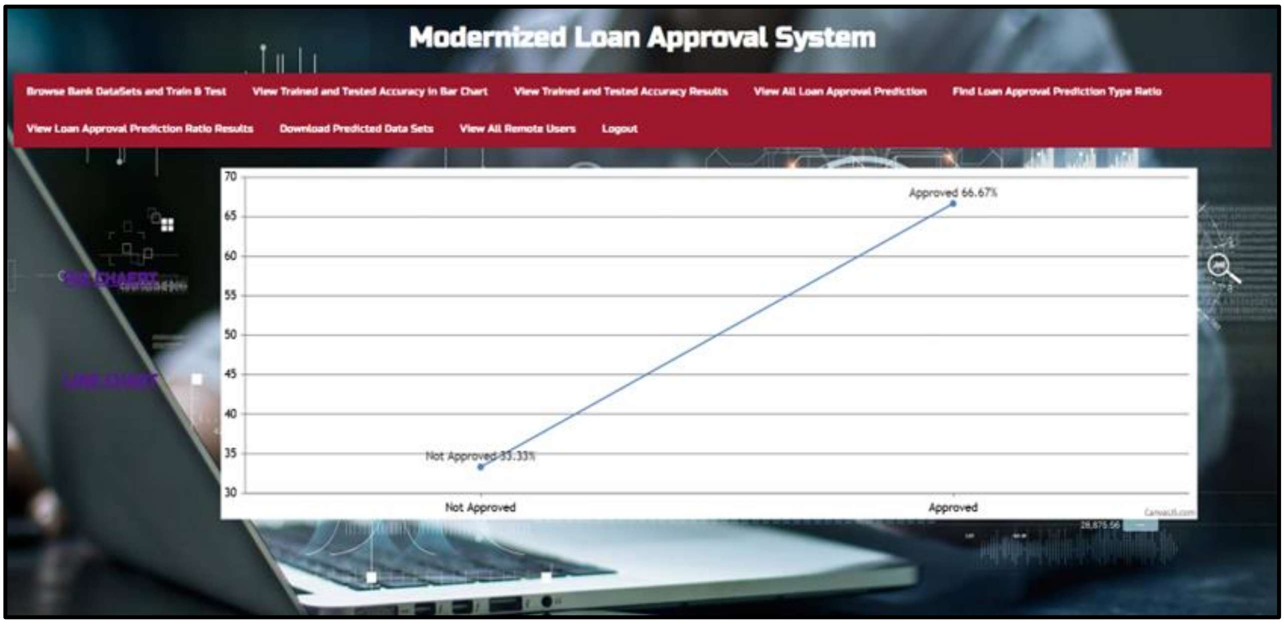
5.Loan Approval Prediction Status:

View All Loan Approval Prediction Status !!!

Loan_ID	Gender	Married Status	Dependents	Education	Employed Status	Applicant Income	Coapplicant Income	Loan Amount	Loan Amount Term	Credit History	Property Area	Prediction
LP001465	Male	Yes	0	Graduate	No	6080	2569	182	360	0	Rural	Not Approved
LP001469	Male	No	0	Graduate	Yes	20166	0	650	480	1	Urban	Approved
LP001013	Male	Yes	0	Not Graduate	No	2333	1516	95	360	1	Urban	Approved
LP001030	Male	Yes	2	Graduate	No	1299	1086	17	120	1	Urban	Approved
LP001050	Male	Yes	1	Graduate	No	1911	116	360	360	0	Semiurban	Approved
LP001086	Male	No	0	Not Graduate	No	1442	0	35	360	1	Urban	Not Approved
LP001086	Male	No	0	Not Graduate	No	1442	0	35	360	1	Urban	Not Approved
LP001736	Male	Yes	0	Graduate	No	2221	0	60	360	0	Urban	Not Approved
LP001743	Male	Yes	2	Graduate	No	4009	1717	116	360	1	Semiurban	Approved
L12323	Male	No	3	Graduate	Yes	500	400	500	300	1	Urban	Approved
L001238	Male	No	0	Graduate	No	0	500	1000	20	0	Semiurban	Approved
L001238	Male	No	2	Graduate	No	0	0	1000	1300	0	Rural	Approved

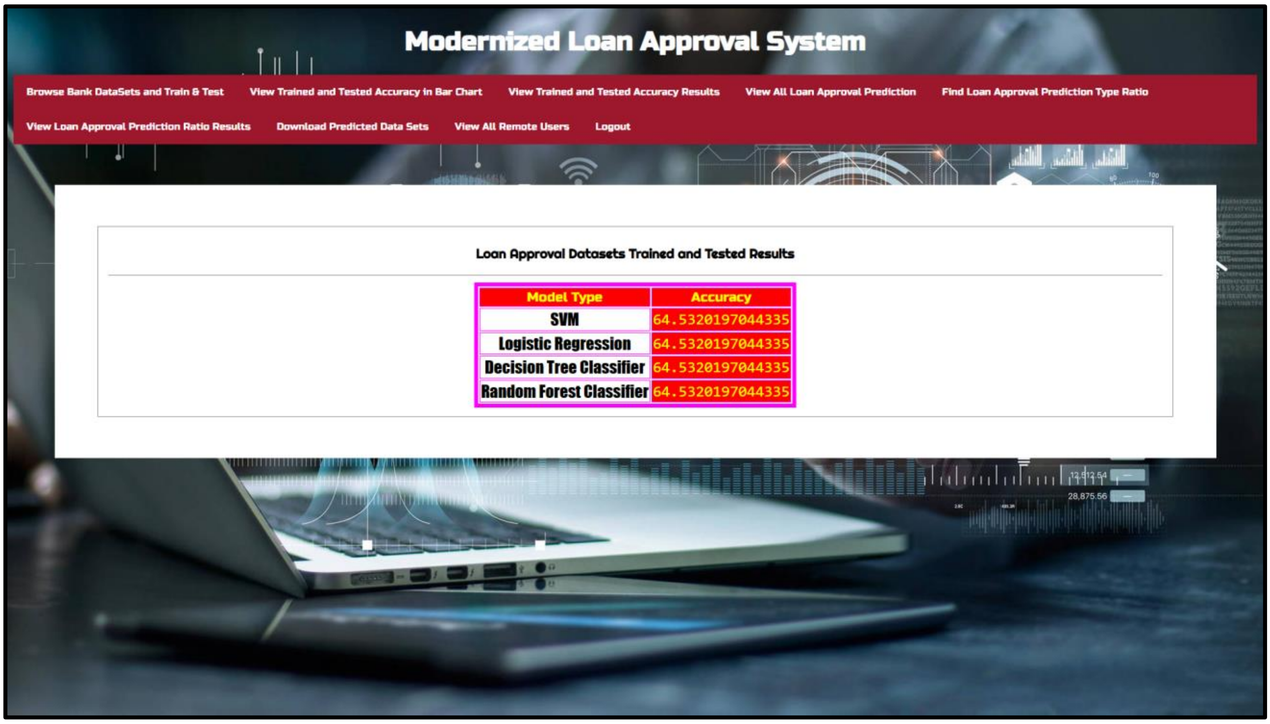
7.5 LOAN APPROVAL PREDICTION STATUS

6. Approval Prediction Ratio Line Graph:



7.6 APRROVAL PREDICTION RATIO LINE GRAPH

7. Profile Datasets Trained and Tested Results:



Model Type	Accuracy
SVM	64.5320197044335
Logistic Regression	64.5320197044335
Decision Tree Classifier	64.5320197044335
Random Forest Classifier	64.5320197044335

7.7 MODELS ACCURACY

8.Profile Datasets Trained and Tested Results (pie chart):

7.8 Trained and Tested Results

8. CONCLUSION & FUTURE SCOPE

The modernized loan approval system represents a significant leap forward in the realm of financial technology. By harnessing the power of machine learning, this system achieves remarkable accuracy and reliability in predicting loan approvals. The rigorous evaluation of performance metrics such as accuracy, precision, recall, F1 score, and AUC-ROC not only validates the model's effectiveness but also provides a comprehensive framework for understanding its strengths and weaknesses.

The analysis indicates that the system excels at identifying approved loans, which is crucial for minimizing risk and ensuring that financial resources are allocated effectively. However, it also exposes areas requiring further refinement, particularly in reducing false positives—instances where loans are incorrectly predicted to be approved—and addressing class imbalances, where the distribution of approved versus rejected applications may skew the model's performance. These insights underscore the importance of a meticulous approach to model selection and feature engineering, as well as the need for ongoing evaluation and iteration to continually enhance predictive accuracy.

Overall, this modernized loan approval system not only outperforms traditional methods but also embodies a more data-driven approach to decision-making in the financial sector. It paves the way for more informed lending practices, ultimately leading to better outcomes for both lenders and borrowers. Looking ahead, it is essential to prioritize the refinement of the model, the incorporation of additional relevant features, and the establishment of robust testing and validation processes to ensure sustained effectiveness.

The future development of the modernized loan approval system is rife with potential enhancements that can significantly improve its capabilities and impact. One primary area of focus is the enhancement of model accuracy through the exploration of advanced algorithms, such as ensemble methods or deep learning techniques. Additionally, hyperparameter tuning will allow for the fine-tuning of model parameters, optimizing performance and responsiveness.

Incorporating a wider array of features is another promising avenue for improvement. For instance, leveraging data from social media activity or analysing transaction history can provide deeper insights into applicants' financial behaviours and risks, enabling more nuanced decision-making. This holistic approach to feature integration could yield more accurate predictions and better-informed lending practices.

Real-time prediction capabilities represent another critical advancement. By employing continuous learning methods, the system can dynamically adapt to new data as it becomes available, ensuring that its predictions remain relevant and accurate over time. This agility will be essential in a fast-paced financial environment where borrower profiles and economic conditions can shift rapidly.

Addressing class imbalances through advanced resampling techniques or cost-sensitive learning methods will also refine the model's performance. By ensuring that the model is equally adept at recognizing both approved and rejected applications, we can reduce biases and enhance overall accuracy.

Furthermore, enhancing model interpretability is vital for building trust among users and regulatory bodies. By making the decision-making process of the model more transparent, stakeholders can better understand how loan decisions are reached, fostering confidence in the system's integrity.

Improving the user experience is equally important. Developing intuitive interfaces and providing personalized insights can facilitate smoother interactions for both lenders and borrowers. This focus on user experience can lead to greater adoption and satisfaction with the system.

Scalability and robustness will be crucial as the system grows. By integrating cloud technologies, the system can handle larger datasets and more complex computations, ensuring that it remains effective under increasing demand. Rigorous testing protocols will further ensure that the model performs reliably across various scenarios.

Lastly, forming partnerships with other financial services could create a more comprehensive solution. Integration with complementary services, such as credit scoring agencies or financial planning tools, can enhance the overall value proposition of the loan approval system, ultimately benefiting users and lenders alike.

REFERENCES

[1] Amruta S. Aphale and R. Prof. Dr. Sandeep. R Shinde, “Predict Loan Approval in Banking System Machine Learning Approach for Cooperative Banks Loan Approval”, International Journal of Engineering Trends and Applications (IJETA), vol. 9, issue 8, 2020)

[2] Loan Prediction Using Ensemble Technique, International Journal of Advanced Research in Computer and Communication Engineering Vol. 5, Issue 3, March 2016

[3] Exploratory data analysis
https://en.wikipedia.org/wiki/Exploratory_data_analysis

[4] Pandas Library

https://pandas.pydata.org/pandas-docs/stable/

[5] MeanDecreaseAccuracy
https://dinsdalelab.sdsu.edu/metag.stats/code/randomforest.hml

[6] Dataset link:

https://github.com/Rayudurajaharsha/mini-proj/blob/main/Prediction_of_modernized_loan_approval-main/Bank_Dataset.csv

GITHUB LINK

[1] Project Code GitHub Link: https://github.com/Rayudurajaharsha/mini-proj/tree/main/Prediction_of_modernized_loan_approval-main