



Predicting Personal Loan Approval Using

Machine Learning

1. **INTRODUCTION**
   1. **OVERVIEW**

Predicting personal loan approval using machine learning (ML) involves developing a model that can accurately classify loan applications as approved or not approved based on various features and factors. The ML model learns from historical loan data to predict whether an applicant is likely to be approved or not.

Predicting personal loan project involves developing a machine learning model that can accurately predict whether a loan application is likely to be approved or rejected based on various factors such as applicant demographics, financial history, credit scores, and loan amount requested. The project typically involves collecting loan application data, cleaning and preprocessing the data, performing feature engineering to create new variables that may be predictive of loan approvals, selecting an appropriate machine learning algorithm such as logistic regression or decision trees, training the model on a portion of the data, evaluating the model's performance on a separate portion of the data, and deploying the model to predict loan approvals for new loan applications. The project can be used by financial institutions to automate the loan approval process, reduce the risk of defaults and losses, and improve the efficiency of their lending operations.

The process of predicting personal loans using ML generally involves the following steps:

**Data Collection:** Collecting relevant loan application data, such as applicant demographics, financial history, employment status, credit scores, loan amount requested, and loan purpose.

**Data Preprocessing:** Cleaning, filtering, and transforming the data to prepare it for analysis. This may include handling missing values, encoding categorical variables, and scaling numeric variables.

**Feature Engineering:** Creating new features from the existing data that may improve the predictive power of the model. For example, creating a debt-to-income ratio variable by dividing the applicant's total debt by their income.

**Model Selection:** Choosing the appropriate ML algorithm to use for the prediction task. This could be a binary classification algorithm such as logistic regression, decision trees, or support vector machines.

**Model Training:** Using a portion of the available data to train the ML model. The model learns to identify patterns in the data that are associated with loan approvals.

**Model Evaluation:** Testing the trained model on a separate portion of the data to measure its accuracy and generalization performance. Various evaluation metrics such as accuracy, precision, recall, and F1 score can be used to evaluate the model's performance.

**Model Deployment:** Deploying the trained ML model into production to predict loan approvals for new loan applications.

* 1. **PURPOSE**

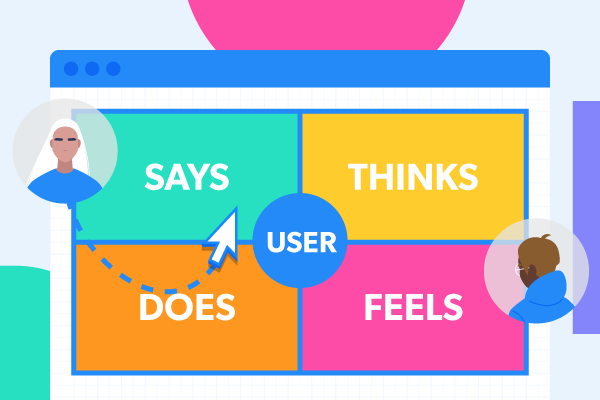
The purpose of predicting personal loan project is to develop a machine learning model that can accurately predict whether a loan application is likely to be approved or rejected based on various factors such as applicant demographics, financial history, credit scores, and loan amount requested.

This project can help financial institutions automate the loan approval process, reduce the risk of defaults and losses, and improve the efficiency of their lending operations. By accurately predicting loan approvals, financial institutions can make better-informed lending decisions, avoid high-risk loans, and ensure that loans are only approved for applicants who are likely to repay them.

This project can also help borrowers by providing them with faster and more efficient loan approval processes, which can help them get access to funds when they need them. Overall, the project can help financial institutions and borrowers alike by improving the accuracy and efficiency of the loan approval process.

**2.PROBLEM DEFINITION AND DESIGN THINKING**

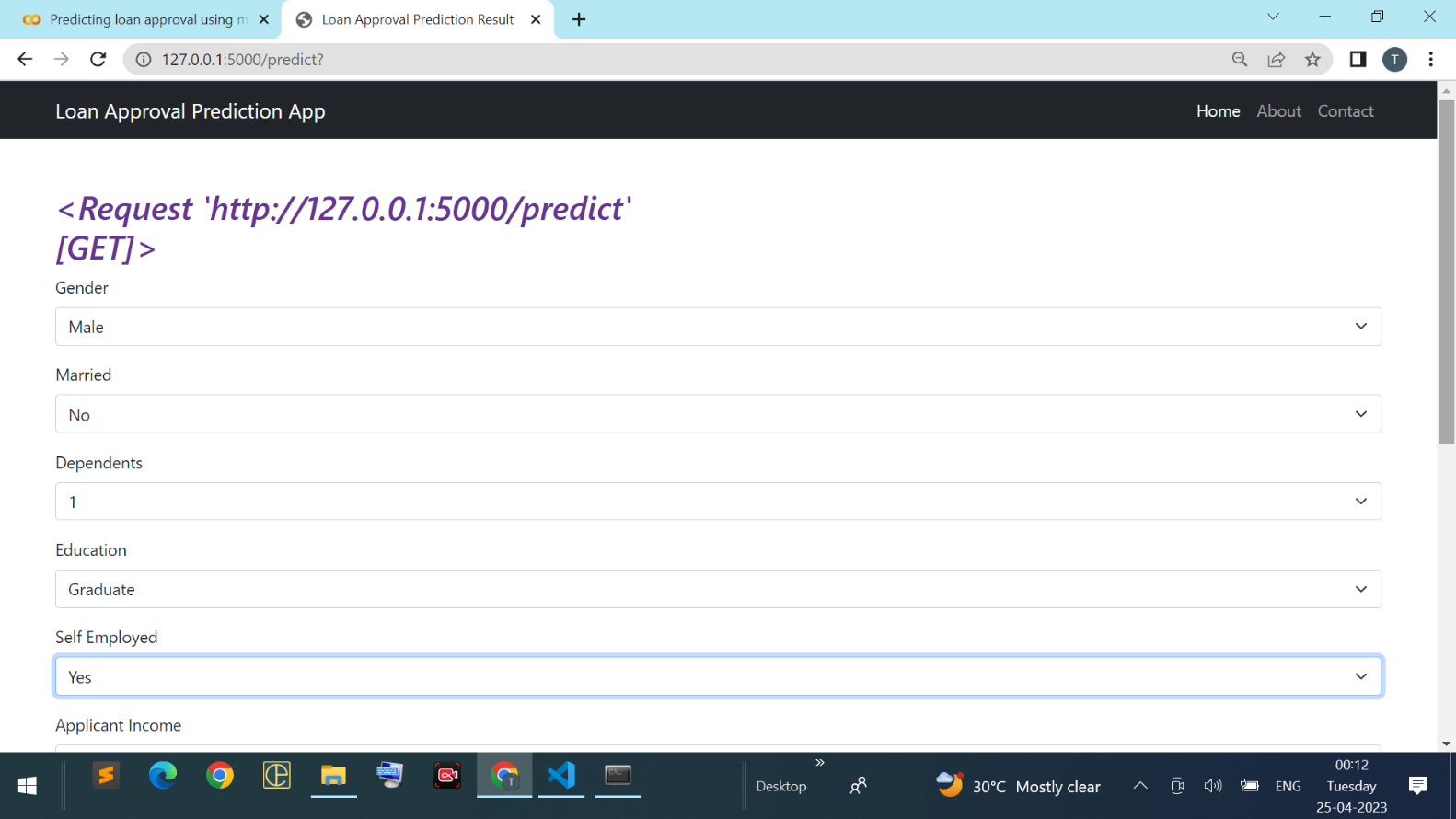
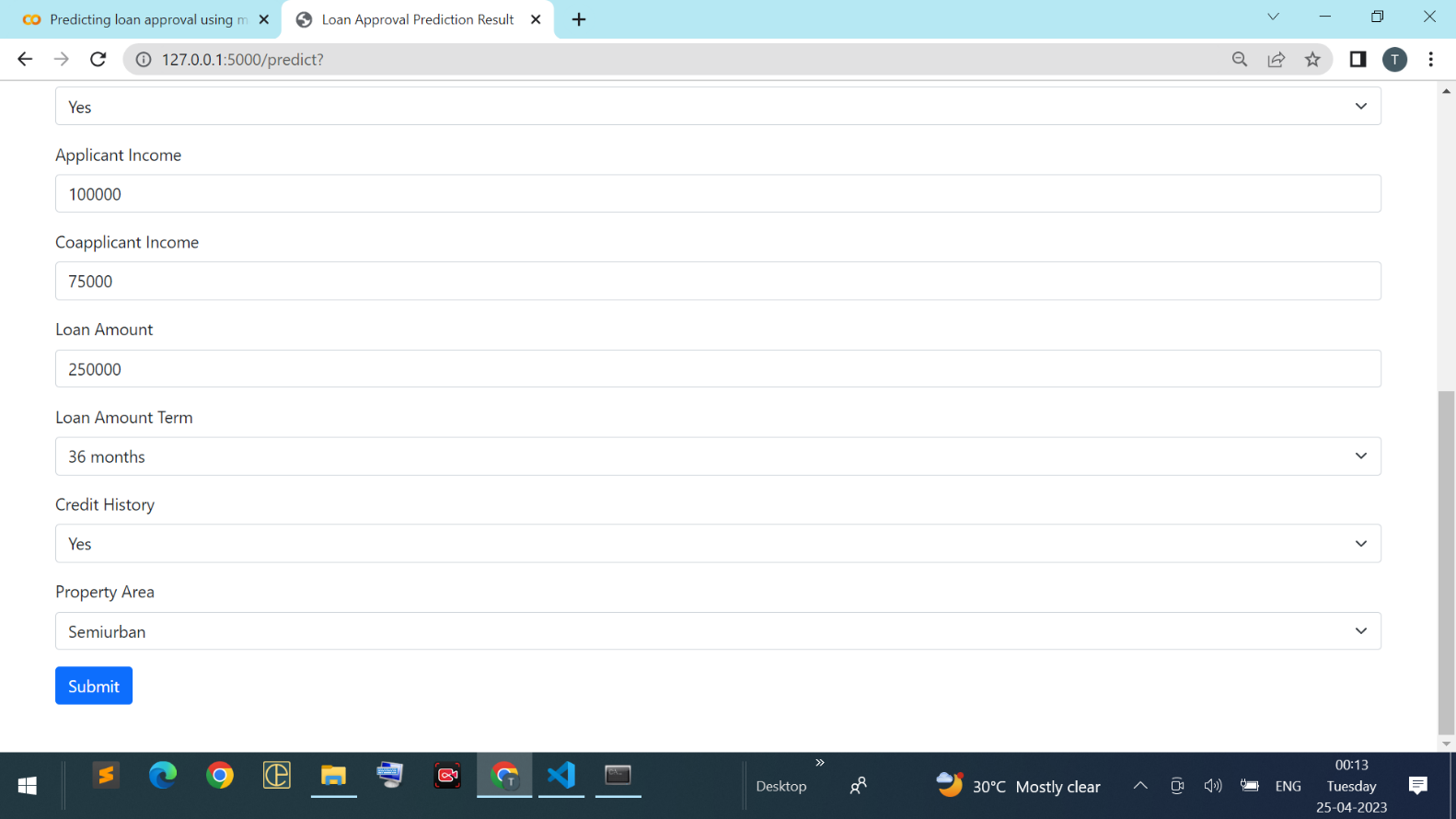
**2.1 EMPATHY MAP**



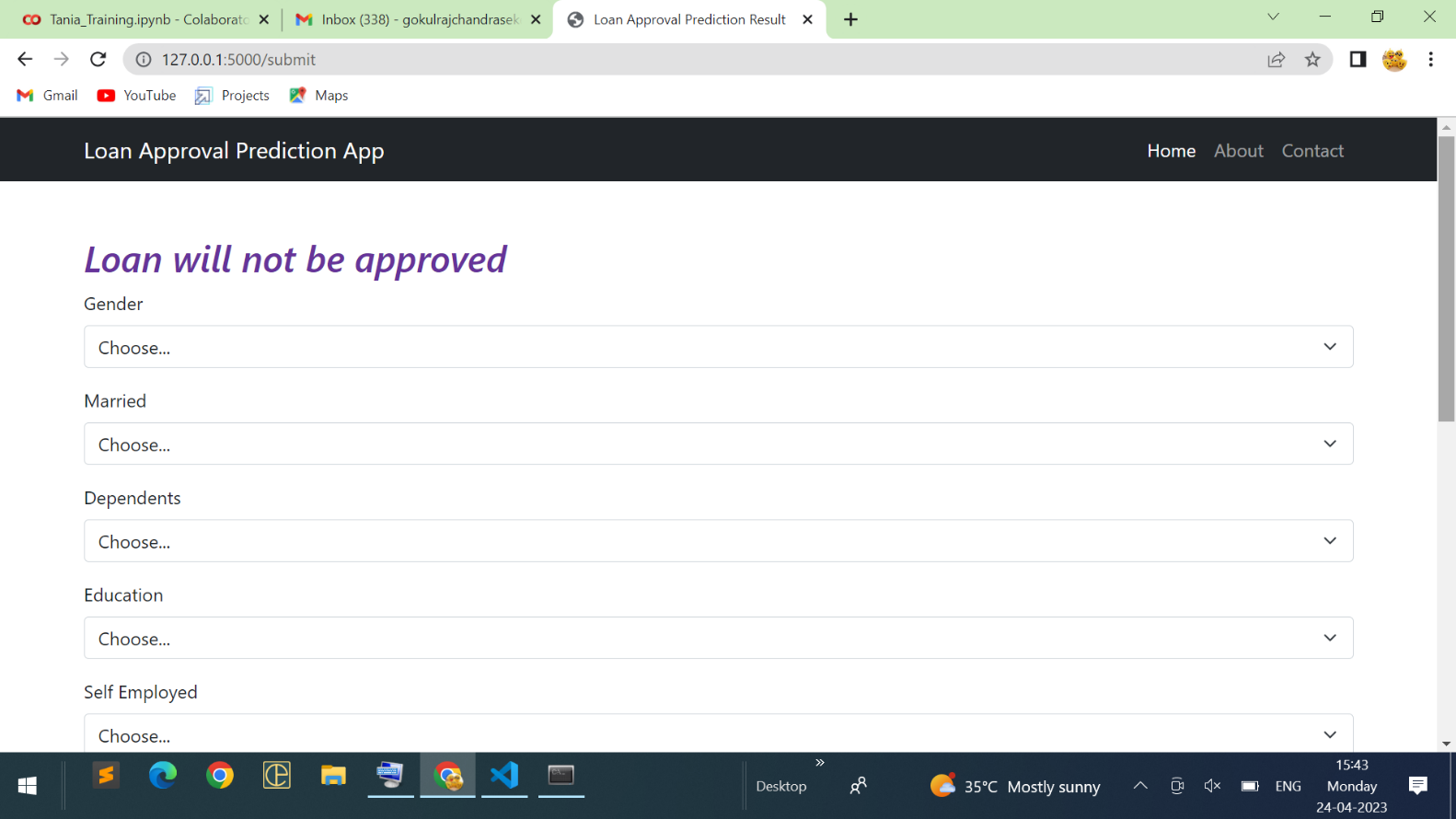
**2.2 IDEATION AND BRAINSTORMING MAP**

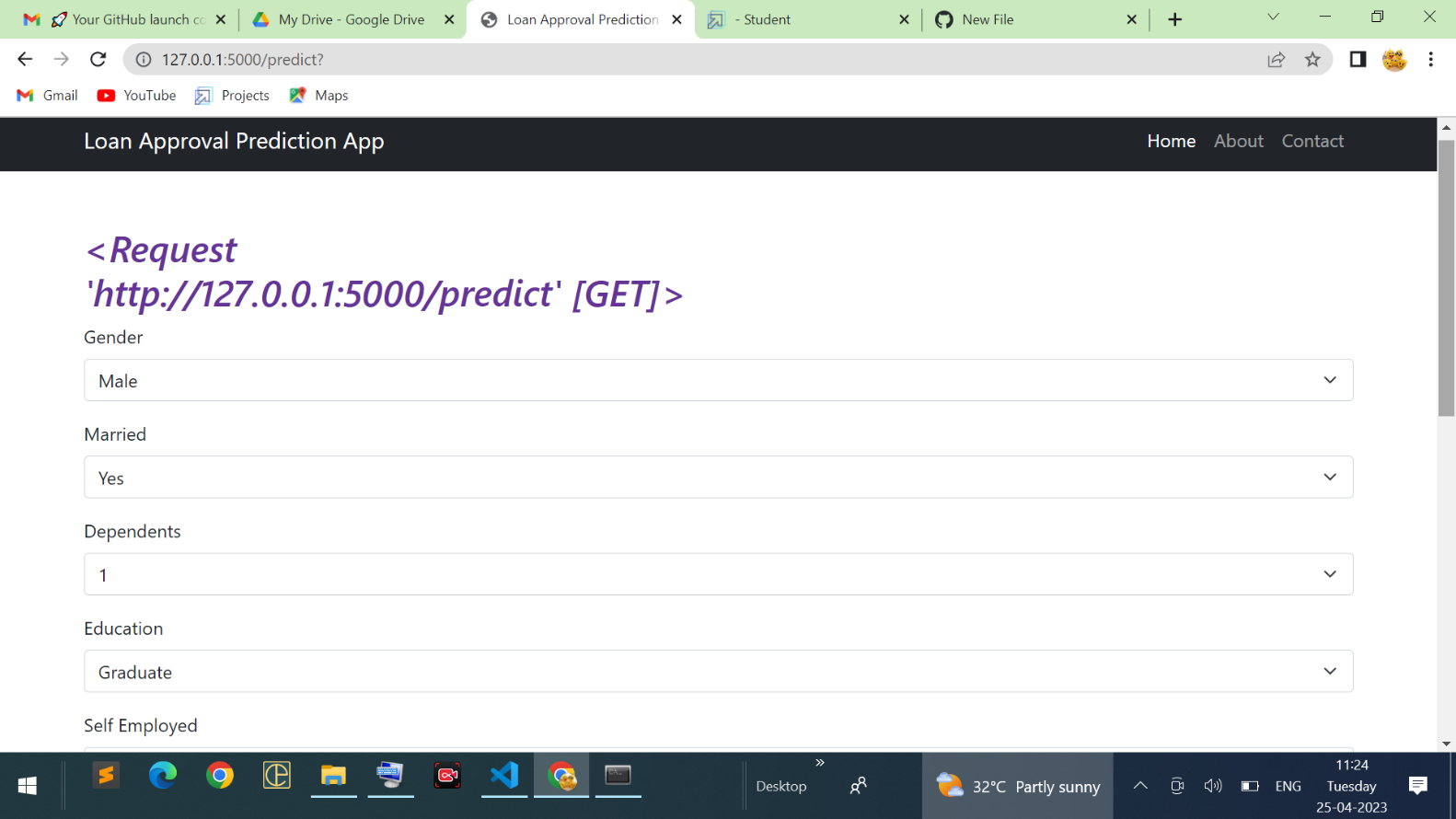
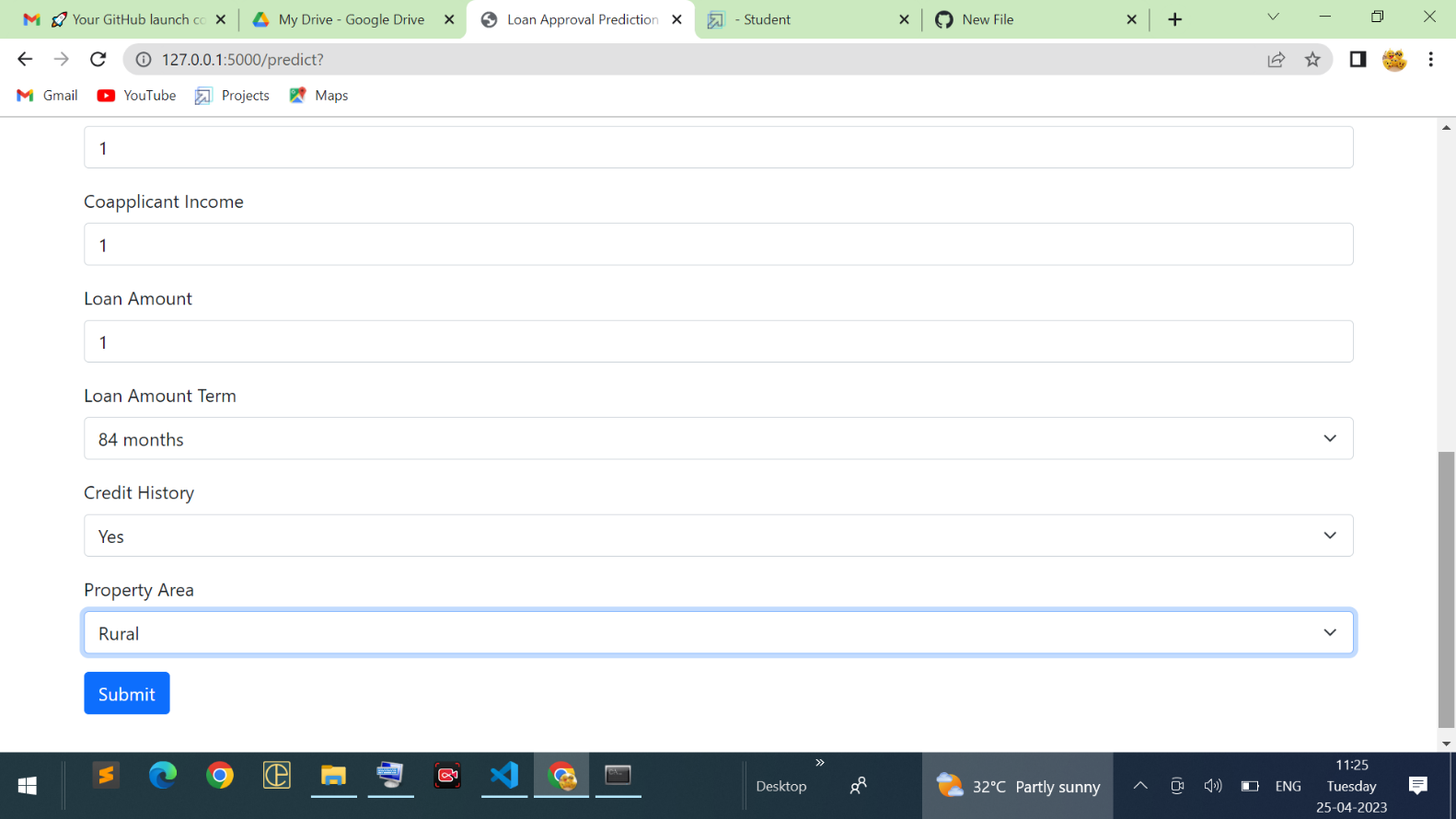
****

**3.RESULT**

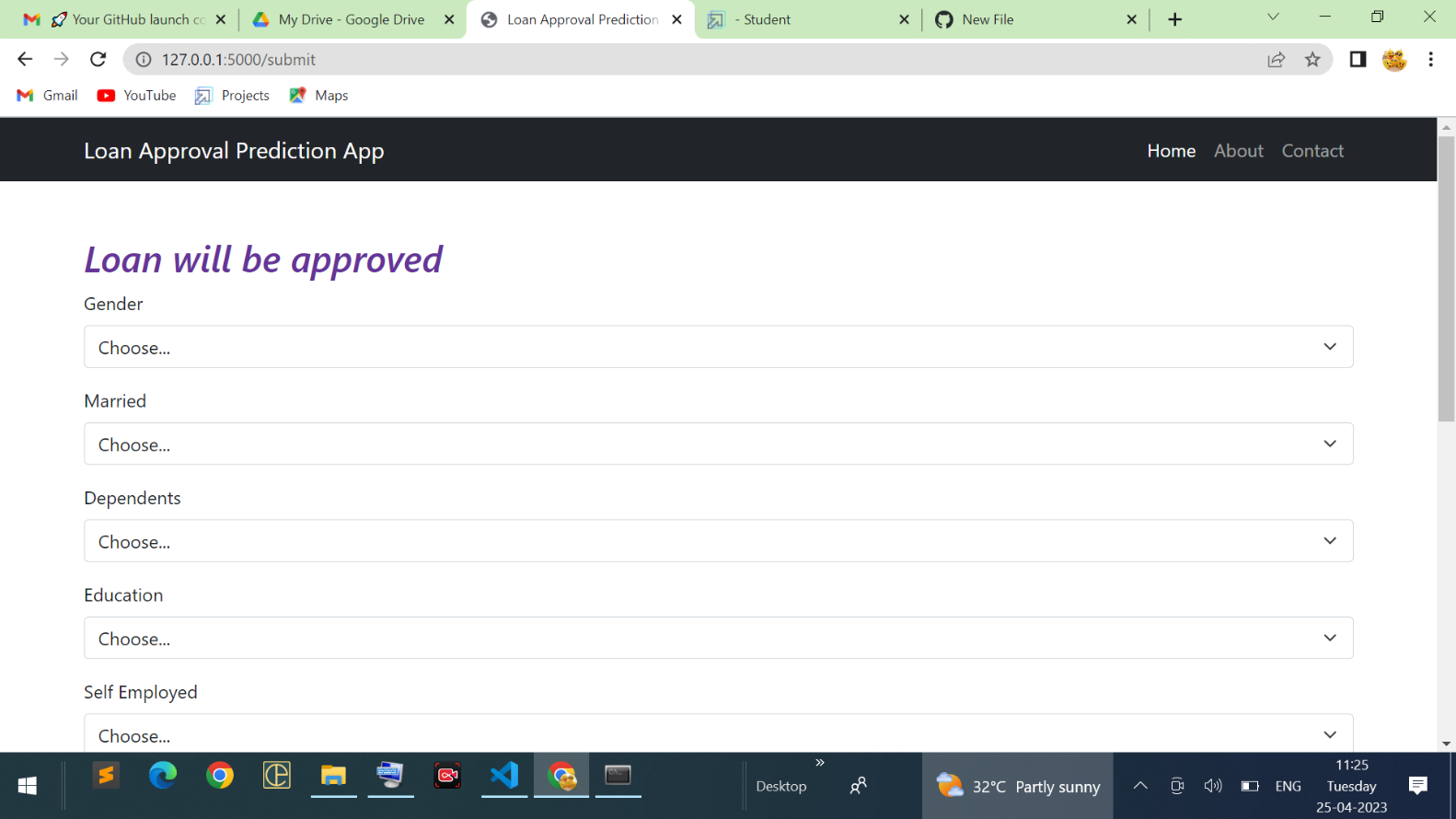
****

Output :





Output:



**4. ADVANTAGES AND DISADVANTAGES**

*ADVANTAGES:*

Improved customer experience: By creating a streamlined and efficient loan application and approval process, financial institutions can provide a better customer experience. This can lead to increased customer satisfaction and loyalty.

Faster loan approvals: A personal loan approval project can help reduce the time it takes to get approved for a loan, providing borrowers with access to funds more quickly.

Reduced risk: By implementing rigorous application and approval processes, financial institutions can reduce the risk of loan defaults and protect their investment.

Increased efficiency: With new technologies and software, financial institutions can automate parts of the loan application and approval process, reducing the need for manual processing and increasing efficiency

*DISADVANTAGES:*

Higher upfront costs: Implementing a personal loan approval project may require financial institutions to invest in new technologies and software, which can be expensive.

Increased complexity: With more rigorous application and approval processes, the loan application process may become more complex, potentially leading to confusion for borrowers.

Dependence on technology: A personal loan approval project relies heavily on technology, which can be prone to glitches and errors. If the technology fails, the loan application and approval process may be delayed.

Limited personal interaction: With more automation in the loan application and approval process, there may be limited opportunities for personal interaction between borrowers and financial institution staff, potentially leading to a less personalized experience.

**5.APPLICATIONS**

Consumer lending: Financial institutions can use a personal loan approval project to streamline the loan application and approval process for individuals seeking personal loans.

Small business lending: Small business owners may also need access to funds, and a personal loan approval project can be adapted to meet their unique needs.

Debt consolidation: Borrowers who have multiple high-interest debts may benefit from consolidating their debts into a single, lower-interest loan. A personal loan approval project can help facilitate this process.

Medical expenses: Unexpected medical expenses can be a financial burden for many individuals. A personal loan approval project can provide a way for people to quickly access the funds they need to cover medical bills.

Home improvements: Homeowners may use a personal loan to finance home improvements or repairs. A personal loan approval project can help make the loan application and approval process faster and more efficient.

Educational expenses: Students may need loans to cover educational expenses such as tuition and textbooks. A personal loan approval project can provide a way for students to quickly access the funds they need to cover these expenses.

Emergency expenses: Unexpected expenses such as car repairs, home repairs, or sudden job loss can put individuals in a financial bind. A personal loan approval project can provide a way for people to access funds quickly in case of emergencies.

Travel expenses: Individuals may need to travel for personal or business reasons, and a personal loan can help cover the expenses. A personal loan approval project can make the loan application and approval process faster and more convenient.

Wedding expenses: Weddings can be expensive, and some individuals may need a loan to cover the costs. A personal loan approval project can provide a way for people to quickly access the funds they need to pay for wedding expenses.

Green energy initiatives: Consumers may want to finance green energy initiatives such as solar panel installations or energy-efficient upgrades. A personal loan approval project can help make these loans more accessible and efficient.

Auto financing: Individuals may need loans to purchase or lease a vehicle. A personal loan approval project can help make the auto financing process faster and more efficient.

Other personal expenses: There are many other personal expenses that individuals may need to finance, such as vacations, electronics, or household appliances. A personal loan approval project can provide a way for people to quickly access the funds they need for these expenses.

In summary, a personal loan approval project can be applied to a wide range of loan types and purposes, depending on the needs of the borrower. By streamlining the loan application and approval process, financial institutions can provide a better customer experience and make loans more accessible to a wider range of individuals.

**6.CONCLUSION**

In conclusion, a personal loan approval project can be a valuable tool for financial institutions looking to streamline their loan application and approval process. By leveraging technology such as machine learning algorithms and data analytics, financial institutions can more accurately assess a borrower's creditworthiness and ability to repay the loan, resulting in faster loan approvals, reduced risk of defaults, and improved customer experience. However, there are also potential disadvantages to consider, such as increased dependence on technology and reduced flexibility in loan terms and conditions. Ultimately, the success of a personal loan approval project depends on careful planning and implementation, as well as ongoing monitoring and adjustment to changing market conditions and borrower needs.

A personal loan approval project can provide financial institutions with a competitive advantage by enabling them to offer faster, more convenient loan processing, which can be an important factor for borrowers when choosing a lender. Moreover, the automation of the loan approval process can lead to cost savings for financial institutions through reduced administrative expenses, more efficient fraud detection, and improved regulatory compliance.

Additionally, a personal loan approval project can provide benefits to borrowers, including quicker access to funds, greater transparency in the loan application process, and more accurate loan assessments that may result in better loan terms for creditworthy borrowers. Furthermore, by using data analytics and machine learning algorithms, financial institutions can tailor loan products to better meet the needs of specific borrower segments, such as small business owners or students.

Overall, a personal loan approval project can benefit both financial institutions and borrowers by improving the loan application and approval process, reducing risk, and enhancing the customer experience. While there are potential challenges to consider, such as technological dependence and reduced flexibility, these can be managed through careful planning and ongoing monitoring and adjustment. In a rapidly evolving financial landscape, a personal loan approval project can provide a valuable tool for financial institutions looking to remain competitive and meet the evolving needs of borrowers.

**7.FUTURE SCOPE**

The future scope of a personal loan approval project is vast, with many opportunities for continued innovation and improvement. Here are some potential areas for future development:

1. Use of alternative data sources: Financial institutions may increasingly look to alternative data sources such as social media, mobile device usage, and other non-traditional data points to supplement credit bureau data and provide a more comprehensive view of a borrower's creditworthiness.
2. Integration with blockchain technology: The use of blockchain technology may enable financial institutions to streamline loan application and approval processes, reduce fraud, and increase transparency and security.
3. Expansion of loan products: Financial institutions may develop new loan products that better meet the needs of specific borrower segments, such as gig economy workers or individuals with non-traditional income sources.
4. Increased personalization: The use of data analytics and machine learning algorithms can help financial institutions personalize loan offers based on a borrower's unique needs and credit history.
5. Implementation of artificial intelligence: The use of AI can enable financial institutions to automate the loan application and approval process further, reduce manual intervention, and increase efficiency.
6. Collaboration with fintech companies: Financial institutions may partner with fintech companies to develop and deploy new loan products and technologies that better meet the needs of borrowers and improve the loan application and approval process.

Overall, the future scope of a personal loan approval project is promising, with many opportunities for continued innovation and improvement through the use of new technologies, data sources, and partnerships. As financial institutions continue to adapt to the changing needs of borrowers and the evolving regulatory landscape, the use of technology and data analytics will likely play an increasingly critical role in the loan application and approval process.

**APPENDIX**

**Source Code:**

**training code:**

import pandas as pd

import numpy as np

import pickle as pl

import matplotlib.pyplot as plt

%matplotlib inline

import seaborn as sns

import sklearn

from sklearn.tree import DecisionTreeClassifier

from sklearn.ensemble import GradientBoostingClassifier, RandomForestClassifier

from sklearn.neighbors import KNeighborsClassifier

from sklearn.model\_selection import RandomizedSearchCV

import imblearn

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler

from sklearn.metrics import accuracy\_score, classification\_report, confusion\_matrix, f1\_score

data=pd.read\_csv(r"train\_u6lujuX\_CVtuZ9i.csv")

data

data.drop(['Loan\_ID'],axis=1,inplace=True)

data.head()

data['Gender']=data['Gender'].map({'Female':1,'Male':0})

data.head()

data['Property\_Area']=data['Property\_Area'].map({'Urban':2,'Semiurban':1,'Rural':0})

data.head()

data['Married']=data['Married'].map({'Yes':1,'No':0})

data.head()

data['Education']=data['Education'].map({'Graduate':1, 'Not Graduate':0})

data.head()

data['Self\_Employed']=data['Self\_Employed'].map({'Yes':1,'No':0})

data.head()

data['Loan\_Status']=data['Loan\_Status'].map({'Y':1, 'N':0})

data.head()

data.isnull().sum()

data['Gender'] = data['Gender'].fillna(data['Gender'].mode()[0])

data['Married'] = data['Married'].fillna(data['Married'].mode()[0])

data['Dependents']=data['Dependents'].str.replace('+','')

data['Dependents'] = data['Dependents'].fillna(data['Dependents'].mode()[0])

data['Self\_Employed'] = data['Self\_Employed'].fillna(data['Self\_Employed'].mode()[0])

data['LoanAmount'] = data['LoanAmount'].fillna(data['LoanAmount'].mode()[0])

data['Loan\_Amount\_Term'] = data['Loan\_Amount\_Term'].fillna(data['Loan\_Amount\_Term'].mode()[0])

data['Credit\_History'] = data['Credit\_History'].fillna(data['Credit\_History'].mode()[0])

data.isnull().sum()

data.info()

data['Gender']=data['Gender'].astype('int64')

data['Married']=data['Married'].astype('int64')

data['Dependents']=data['Dependents'].astype('int64')

data['Self\_Employed']=data['Self\_Employed'].astype('int64')

data['CoapplicantIncome']=data['CoapplicantIncome'].astype('int64')

data['LoanAmount']=data['LoanAmount'].astype('int64')

data['Loan\_Amount\_Term']=data['Loan\_Amount\_Term'].astype('int64')

data['Credit\_History']=data['Credit\_History'].astype('int64')

data.info()

plt.figure(figsize=(12,5))

plt.subplot(121)

sns.distplot(data['ApplicantIncome'], color='r')

plt.subplot(122)

sns.distplot(data['Credit\_History'])

plt.show()

plt.figure(figsize=(18,4))

plt.subplot(1,4,1)

sns.countplot(data['Gender'], color='y')

plt.subplot(1,4,2)

sns.histplot(data['Education'])

plt.show()

plt.figure(figsize=(20,5))

plt.subplot(131)

sns.countplot(x=data['Married'], hue=data['Gender'])

plt.subplot(132)

sns.countplot(x=data['Self\_Employed'], hue=data['Education'])

plt.subplot(133)

sns.countplot(x=data['Property\_Area'], hue=data['Loan\_Amount\_Term'])

plt.show()

pd.crosstab(data['Gender'],[data['Self\_Employed']])

sns.swarmplot(x=data['Gender'], y=data['ApplicantIncome'], hue=data['Loan\_Status'])

from imblearn.combine import SMOTETomek

smote = SMOTETomek()

y = data['Loan\_Status']

x = data.drop(columns=['Loan\_Status'],axis=1)

x.shape

y.shape

x\_bal,y\_bal = smote.fit\_resample(x,y)

print(y.value\_counts())

print(y\_bal.value\_counts())

names = x\_bal.columns

x\_bal.head()

sc=StandardScaler()

x\_bal=sc.fit\_transform(x\_bal)

x\_bal = pd.DataFrame(x\_bal,columns=names)

x\_bal.head()

X\_train, X\_test, y\_train, y\_test = train\_test\_split(x\_bal, y\_bal, test\_size=0.33, random\_state=42)

X\_train.shape

X\_test.shape

y\_train.shape, y\_test.shape

def RandomForest(X\_train,X\_test,y\_train,y\_test):

model = RandomForestClassifier()

model.fit(X\_train,y\_train)

y\_tr = model.predict(X\_train)

print(accuracy\_score(y\_tr,y\_train))

yPred = model.predict(X\_test)

print(accuracy\_score(yPred,y\_test))

RandomForest(X\_train,X\_test,y\_train,y\_test)

def decisionTree(X\_train,X\_test,y\_train,y\_test):

model = DecisionTreeClassifier()

model.fit(X\_train,y\_train)

y\_tr = model.predict(X\_train)

print(accuracy\_score(y\_tr,y\_train))

yPred = model.predict(X\_test)

print(accuracy\_score(yPred,y\_test))

decisionTree(X\_train,X\_test,y\_train,y\_test)

def KNN(X\_train,X\_test,y\_train,y\_test):

model = KNeighborsClassifier()

model.fit(X\_train,y\_train)

y\_tr = model.predict(X\_train)

print(accuracy\_score(y\_tr,y\_train))

yPred = model.predict(X\_test)

print(accuracy\_score(yPred,y\_test))

KNN(X\_train,X\_test,y\_train,y\_test)

def XGB(X\_train,X\_test,y\_train,y\_test):

model = GradientBoostingClassifier()

model.fit(X\_train,y\_train)

y\_tr = model.predict(X\_train)

print(accuracy\_score(y\_tr,y\_train))

yPred = model.predict(X\_test)

print(accuracy\_score(yPred,y\_test))

XGB(X\_train,X\_test,y\_train,y\_test)

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense

classifier = Sequential()

classifier.add(Dense(units=100, activation='relu', input\_dim=11))

classifier.add(Dense(units=50, activation='relu'))

classifier.add(Dense(units=1, activation='sigmoid'))

classifier.compile(optimizer="adam", loss="binary\_crossentropy", metrics=['accuracy'])

classifier.fit(X\_train,y\_train, batch\_size=100, validation\_split=0.2, epochs=100)

y\_pred = classifier.predict(X\_test)

y\_pred

y\_pred = y\_pred.astype(int)

y\_pred

print(accuracy\_score(y\_pred, y\_test))

print("ANN Model")

print("Confusion\_Matrix")

print(confusion\_matrix(y\_test, y\_pred))

print("Classification Report")

print(classification\_report(y\_test, y\_pred))

rf = RandomForestClassifier()

parameters = {

'n\_estimators' : [1,20,30,55,68,74,90,120,115],

'criterion' : ['gini','entropy'],

'max\_features' : ["auto", "sqrt", "log2"],

'max\_depth' : [2,5,8,10], 'verbose' : [1,2,3,4,6,8,9,10]

}

RCV = RandomizedSearchCV(estimator=rf,param\_distributions=parameters,cv=10,n\_iter=4)

RCV.fit(X\_train,y\_train)

bt\_params = RCV.best\_params\_

bt\_score = RCV.best\_score\_

bt\_params

bt\_score

def RandomForest(X\_train,X\_test,y\_train,y\_test):

model = RandomForestClassifier(verbose= 9, n\_estimators= 68, max\_features= 'auto',max\_depth= 8,criterion= 'entropy')

model.fit(X\_train,y\_train)

y\_tr = model.predict(X\_train)

print("Training Accuracy")

print(accuracy\_score(y\_tr,y\_train))

yPred = model.predict(X\_test)

print('Training Accuracy')

print(accuracy\_score(yPred,y\_test))

model = RandomForestClassifier(verbose= 9, n\_estimators= 68, max\_features= 'auto',max\_depth= 8,criterion= 'entropy')

model.fit(X\_train,y\_train)

RandomForest(X\_train,X\_test,y\_train,y\_test)

import pickle

pickle.dump(model,open('rdf.pkl','wb'))

pickle.dump(sc,open('scale.pkl','wb'))

**app.py**

import numpy as np

import pickle

import pandas

import os

from flask import Flask, request, render\_template

app = Flask(\_\_name\_\_)

model = pickle.load(open(r'rdf.pkl', 'rb'))

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

@app.route('/')

def home():

return render\_template('home.html')

@app.route('/predict', methods=["POST","GET"])

def predict():

return render\_template('prediction\_new.html')

@app.route('/submit', methods=["POST","GET"])

def submit():

input\_feature = [int(x) for x in request.form.values()]

input\_feature = [np.array(input\_feature)]

print(input\_feature)

names =['Gender','Married','Dependents','Education','Self\_Employed','ApplicantIncome','CoapplicantIncome','LoanAmount','Loan\_Amount\_Term','Credit\_History','Property\_Area']

data = pandas.DataFrame(input\_feature, columns=names)

print(data)

data\_scaled = scale.fit\_transform(data)

data = pandas.DataFrame(data, columns=names)

prediction = model.predict(data)

print(prediction)

prediction = int(prediction)

print(type(prediction))

if (prediction==0):

return render\_template("prediction\_new.html", request="Loan will not be approved")

else:

return render\_template("prediction\_new.html", request="Loan will be approved")

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=False)

**home.html**

<!doctype html>

<html lang="en">

<head>

<!-- Required meta tags -->

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<!-- Bootstrap CSS -->

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0/dist/css/bootstrap.min.css" rel="stylesheet"

integrity="sha384-wEmeIV1mKuiNpC+IOBjI7aAzPcEZeedi5yW5f2yOq55WWLwNGmvvx4Um1vskeMj0" crossorigin="anonymous">

<!-- Custom CSS -->

<link rel="stylesheet" href="static/style.css">

<title>Banking</title>

</head>

body>

<section id="header">

<div class="container">

<nav class="navbar navbar-expand-lg navbar-dark">

<div class="container-fluid">

<svg xmlns="http://www.w3.org/2000/svg" width="16" height="16" fill="#e6b30e" class="bi bi-house"

viewBox="0 0 16 16">

<path fill-rule="evenodd"

d="M2 13.5V7h1v6.5a.5.5 0 0 0 .5.5h9a.5.5 0 0 0 .5-.5V7h1v6.5a1.5 1.5 0 0 1-1.5 1.5h-9A1.5 1.5 0 0 1 2 13.5zm11-11V6l-2-2V2.5a.5.5 0 0 1 .5-.5h1a.5.5 0 0 1 .5.5z" />

<path fill-rule="evenodd"

d="M7.293 1.5a1 1 0 0 1 1.414 0l6.647 6.646a.5.5 0 0 1-.708.708L8 2.207 1.354 8.854a.5.5 0 1 1-.708-.708L7.293 1.5z" />

</svg> <a class="navbar-brand theme-text">

Predicting Personal Loan</a>

<button class="navbar-toggler" type="button" data-bs-toggle="collapse"

data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"

aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav ms-auto mb-2 mb-lg-0">

<li class="nav-item">

<a class="nav-link act" aria-current="page" href="#home-section">Home</a>

</li>

</li>

<li class="nav-item">

<a class="nav-link" href="#">About Us</a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">Contact Us</a>

</li>

</ul>

</div>

</div>

</nav>

</div>

<br><br>

<h1 style="color: beige; text-align: center;">BANKING SOULTION</h1>

<p style="color: beige; text-align: center;">Loan prediction is a very common real-life problem that each retail bank faces at least onece in its lifetime.

<br>

If done correctly, it can save a lot of man-hours at the end of a retail bank.

</p>

<br><br>

<form action="{{ url\_for('predict') }}" method="get">

<center>

<button type="submit" class="btn" style="text-align: center;">Click Me! to predict</button>

</center>

</form>

<svg class="wave" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 1440 320">

<path fill="#fff" fill-opacity="1"

d="M0,192L60,181.3C120,171,240,149,360,133.3C480,117,600,107,720,106.7C840,107,960,117,1080,122.7C1200,128,1320,128,1380,128L1440,128L1440,320L1380,320C1320,320,1200,320,1080,320C960,320,840,320,720,320C600,320,480,320,360,320C240,320,120,320,60,320L0,320Z">

</path>

</svg>

</section>

<!-- Option 1: Bootstrap Bundle with Popper -->

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0/dist/js/bootstrap.bundle.min.js"

integrity="sha384-p34f1UUtsS3wqzfto5wAAmdvj+osOnFyQFpp4Ua3gs/ZVWx6oOypYoCJhGGScy+8"

crossorigin="anonymous"></script>

</body>

</html>

**Prediction\_new.html**

<!doctype html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Loan Approval Prediction Result</title>

<!-- Bootstrap CSS -->

<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/css/bootstrap.min.css">

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-dark bg-dark">

<div class="container">

<a class="navbar-brand" href="#">Loan Approval Prediction App</a>

<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarNav">

<ul class="navbar-nav ms-auto">

<li class="nav-item">

<a class="nav-link active" aria-current="page" href="#home-section">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">About</a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">Contact</a>

</li>

</ul>

</div>

</div>

</nav>

<div class="container my-5">

{% if request %}

<h2 style="color: rebeccapurple; font-style: italic; margin-right: 50%;">{{ request }}</h2>

{% endif %}

<form action="{{ url\_for('submit') }}" method="post">

<div class="mb-3">

<label for="gender" class="form-label">Gender</label>

<select class="form-select" id="gender" name="Gender">

<option selected disabled>Choose...</option>

<option value="0">Male</option>

<option value="1">Female</option>

</select>

</div>

<div class="mb-3">

<label for="married" class="form-label">Married</label>

<select class="form-select" id="married" name="Married">

<option selected disabled>Choose...</option>

<option value="0">No</option>

<option value="1">Yes</option>

</select>

</div>

<div class="mb-3">

<label for="dependents" class="form-label">Dependents</label>

<select class="form-select" id="dependents" name="Dependents">

<option selected disabled>Choose...</option>

<option value="0">0</option>

<option value="1">1</option>

<option value="2">2</option>

</select>

</div>

<div class="mb-3">

<label for="education" class="form-label">Education</label>

<select class="form-select" id="education" name="Education">

<option selected disabled>Choose...</option>

<option value="0">Not Graduate</option>

<option value="1">Graduate</option>

</select>

</div>

<div class="mb-3">

<label for="self-employed" class="form-label">Self Employed</label>

<select class="form-select" id="self-employed" name="Self\_Employed">

<option selected disabled>Choose...</option>

<option value="0">No</option>

<option value="1">Yes</option>

</select>

</div>

<div class="mb-3">

<label for="applicant-income" class="form-label">Applicant Income</label>

<input type="number" class="form-control" id="applicant-income" name="ApplicantIncome" required>

</div>

<div class="mb-3">

<label for="coapplicant-income" class="form-label">Coapplicant Income</label>

<input type="number" class="form-control" id="coapplicant-income" name="CoapplicantIncome" required>

</div>

<div class="mb-3">

<label for="loan-amount" class="form-label">Loan Amount</label>

<input type="number" class="form-control" id="loan-amount" name="LoanAmount" required>

</div>

<div class="mb-3">

<label for="loan-amount-term" class="form-label">Loan Amount Term</label>

<select class="form-select" id="loan-amount-term" name="Loan\_Amount\_Term">

<option selected disabled>Choose...</option>

<option value="12">12 months</option>

<option value="36">36 months</option>

<option value="60">60 months</option>

<option value="84">84 months</option>

<option value="120">120 months</option>

<option value="180">180 months</option>

<option value="240">240 months</option>

<option value="300">300 months</option>

<option value="360">360 months</option>

</select>

</div>

<div class="mb-3">

<label for="credit-history" class="form-label">Credit History</label>

<select class="form-select" id="credit-history" name="Credit\_History">

<option selected disabled>Choose...</option>

<option value="0">No</option>

<option value="1">Yes</option>

</select>

</div>

<div class="mb-3">

<label for="property-area" class="form-label">Property Area</label>

<select class="form-select" id="property-area" name="Property\_Area">

<option selected disabled>Choose...</option>

<option value="0">Rural</option>

<option value="1">Semiurban</option>

<option value="2">Urban</option>

</select>

</div>

<button type="submit" class="btn btn-primary">Submit</button>

</form>

</div>

</body>

</html>