### **Wireframe Documentation for Credit Card Default Prediction System**

#### **1. Overview**

* **Purpose:** Predict whether a credit card holder will default on their payment.
* **Scope:** The system includes data collection, preprocessing, analysis, model training, evaluation, and deployment.

#### **2. Architecture Diagram**

sql

Copy code

+---------------------------------------------------+

| Data Collection |

|---------------------------------------------------|

| | | |

| External Data Sources Internal Data Sources |

+---------------------------------------------------+

|

V

+---------------------------------------------------+

| Data Preprocessing |

|---------------------------------------------------|

| Data Cleaning | Feature Engineering | Scaling |

+---------------------------------------------------+

|

V

+---------------------------------------------------+

| Exploratory Data Analysis (EDA) |

|---------------------------------------------------|

| Data Visualization | Statistics |

+---------------------------------------------------+

|

V

+---------------------------------------------------+

| Feature Selection |

+---------------------------------------------------+

|

V

+---------------------------------------------------+

| Model Selection |

|---------------------------------------------------|

| Logistic Regression | Decision Trees | Neural Net |

+---------------------------------------------------+

|

V

+---------------------------------------------------+

| Model Training |

+---------------------------------------------------+

|

V

+---------------------------------------------------+

| Model Evaluation |

+---------------------------------------------------+

|

V

+---------------------------------------------------+

| Model Deployment |

|---------------------------------------------------|

| API Creation | Cloud Deployment |

+---------------------------------------------------+

|

V

+---------------------------------------------------+

| Monitoring and Maintenance |

+---------------------------------------------------+

#### **3. Component Details**

##### **3.1 Data Collection**

* **Sources:** Financial institutions, publicly available datasets (e.g., UCI Credit Card Default dataset).
* **Format:** CSV, Excel, JSON, SQL databases.

##### **3.2 Data Preprocessing**

* **Data Cleaning:**
  + Handle missing values using techniques like imputation.
  + Remove or correct outliers and erroneous data.
* **Feature Engineering:**
  + Create new features such as credit utilization ratio, payment history, etc.
  + Encode categorical variables using one-hot encoding or label encoding.
* **Scaling:**
  + Normalize or standardize numerical features to bring them to a similar scale.

##### **3.3 Exploratory Data Analysis (EDA)**

* **Visualizations:**
  + Histograms, bar plots, scatter plots, and box plots using Matplotlib and Seaborn.
  + Correlation heatmaps to understand relationships between features.
* **Statistical Analysis:**
  + Summary statistics (mean, median, standard deviation).
  + Distribution analysis.

##### **3.4 Feature Selection**

* **Techniques:**
  + Correlation analysis to remove highly correlated features.
  + Feature importance using models like Random Forest.
  + Recursive Feature Elimination (RFE).

##### **3.5 Model Selection**

* **Algorithms:**
  + Logistic Regression
  + Decision Trees
  + Random Forest
  + Gradient Boosting
  + Neural Networks
* **Evaluation Metrics:**
  + Accuracy
  + Precision
  + Recall
  + F1-score
  + AUC-ROC

##### **3.6 Model Training**

* **Process:**
  + Split data into training and testing sets (e.g., 80-20 split).
  + Train multiple models.
  + Tune hyperparameters using Grid Search or Random Search.

##### **3.7 Model Evaluation**

* **Validation:**
  + Evaluate models on the testing set.
  + Use cross-validation to ensure robustness.
* **Metrics:**
  + Confusion matrix
  + ROC curve
  + Precision-Recall curve

##### **3.8 Model Deployment**

* **API Creation:**
  + Use Flask or FastAPI to create an API for the model.
* **Deployment:**
  + Deploy the model on cloud platforms like AWS, Google Cloud, or Azure.

##### **3.9 Monitoring and Maintenance**

* **Monitoring:**
  + Continuously monitor the model’s performance using metrics.
  + Set up alerts for significant performance drops.
* **Maintenance:**
  + Periodically retrain the model with new data.
  + Update the model to adapt to changing patterns.

#### **4. User Interface (UI) Wireframes**

##### **4.1 Data Upload Page**

* **Components:**
  + File upload button
  + Data format instructions
  + Example file download link

##### **4.2 Data Preprocessing Page**

* **Components:**
  + Data preview table
  + Options for handling missing values
  + Options for feature engineering and scaling

##### **4.3 EDA Page**

* **Components:**
  + Visualizations (charts, graphs)
  + Summary statistics
  + Correlation heatmap

##### **4.4 Model Training Page**

* **Components:**
  + Select algorithm dropdown
  + Hyperparameter tuning options
  + Train model button

##### **4.5 Model Evaluation Page**

* **Components:**
  + Model performance metrics
  + Confusion matrix
  + ROC and Precision-Recall curves

##### **4.6 Model Deployment Page**

* **Components:**
  + API endpoint information
  + Deployment status
  + Logs and monitoring dashboard

#### **5. API Endpoints**

* **Upload Data:** POST /upload
* **Preprocess Data:** POST /preprocess
* **Train Model:** POST /train
* **Evaluate Model:** POST /evaluate
* **Predict Default:** POST /predict
* **Monitor Model:** GET /monitor

#### **6. Data Flow Diagram**

lua

Copy code

+------------------+ +--------------------+ +----------------+

| | | | | |

| Data Collection +------->+ Data Preprocessing+------->+ Model Training|

| | | | | |

+--------+---------+ +---------+----------+ +-------+--------+

| | |

| | |

V V |

+--------+---------+ +---------+----------+ +-------+--------+

| | | | | |

| EDA +------->+ Feature Selection +------->+ Model Evaluation|

| | | | | |

+--------+---------+ +---------+----------+ +-------+--------+

| | |

| | |

V V |

+--------+---------+ +---------+----------+ +-------+--------+

| | | | | |

| Model Deployment +------->+ API Creation +------->+ Model Monitoring|

| | | | | |

+------------------+ +--------------------+ +----------------+

This wireframe documentation outlines the components, processes, and interactions involved in a credit card default prediction system. It provides a high-level view of the system architecture and the flow of data through the various stages.