**Phase-3**

**Student Name**: S.Vishnu Priya

**Register Number:** 613023104120

**Institution:**Vivekanandha College Technology for Women

**Department:** BE-COMPUTER SCIENCE ANDENGINEERING

**Date of Submission:**12-05-2025

**Github Repository Link:** <https://github.com/vishnupriya062006/nm-project-phase-3>

Google Colab Link:

[https://colab.research.google.com/drive/1vS\_aApYsOVisQ0Ur\_vaC\_dUr9-4o9uGp - scrollTo=t-KCr-KvZClT](https://colab.research.google.com/drive/1vS_aApYsOVisQ0Ur_vaC_dUr9-4o9uGp#scrollTo=t-KCr-KvZClT)

# Problem Statement

Credit card fraud is a significant financial threat affecting millions of users globally, leading to massive monetary losses. As online transactions increase, so does the complexity of fraudulent activities. This project aims to develop a machine learning-based classification system to detect fraudulent credit card transactions in real-time, enabling banks and financial institutions to mitigate losses and improve security.

# Abstract

This project focuses on building an AI-powered system for detecting and preventing credit card fraud. The objective is to classify transactions as either fraudulent or legitimate using machine learning algorithms. The system involves data preprocessing, exploratory data analysis, feature engineering, and model training using algorithms like Logistic Regression, Random Forest, and XGBoost. After evaluating models based on metrics like accuracy, precision, recall, and F1-score, the best-performing model is deployed using Streamlit. This system provides fast, efficient, and scalable fraud detection to secure digital transactions.

# System Requirements

* **Hardware:** Minimum 8GB RAM

i5 Processor or higher

* **Software:**

Python 3.9+

Jupyter Notebook / Google Colab

**Required libraries**: NumPy, Pandas, Matplotlib, Seaborn, Scikit-learn, XGBoost, Streamlit

# Objectives

* Detect fraudulent credit card transactions accurately.
* Minimize false positives to avoid customer inconvenience.
* Provide a deployable model for real-time fraud detection.
* Enhance customer trust and reduce financial loss.

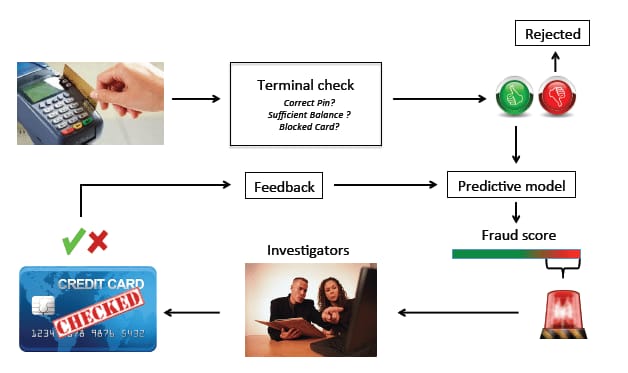
# Flowchart of Project Workflow

**[Include flowchart from:**

* *Data Collection → Preprocessing → EDA → Feature Engineering → Modeling → Evaluation → Deployment*

**Tools you can use:**

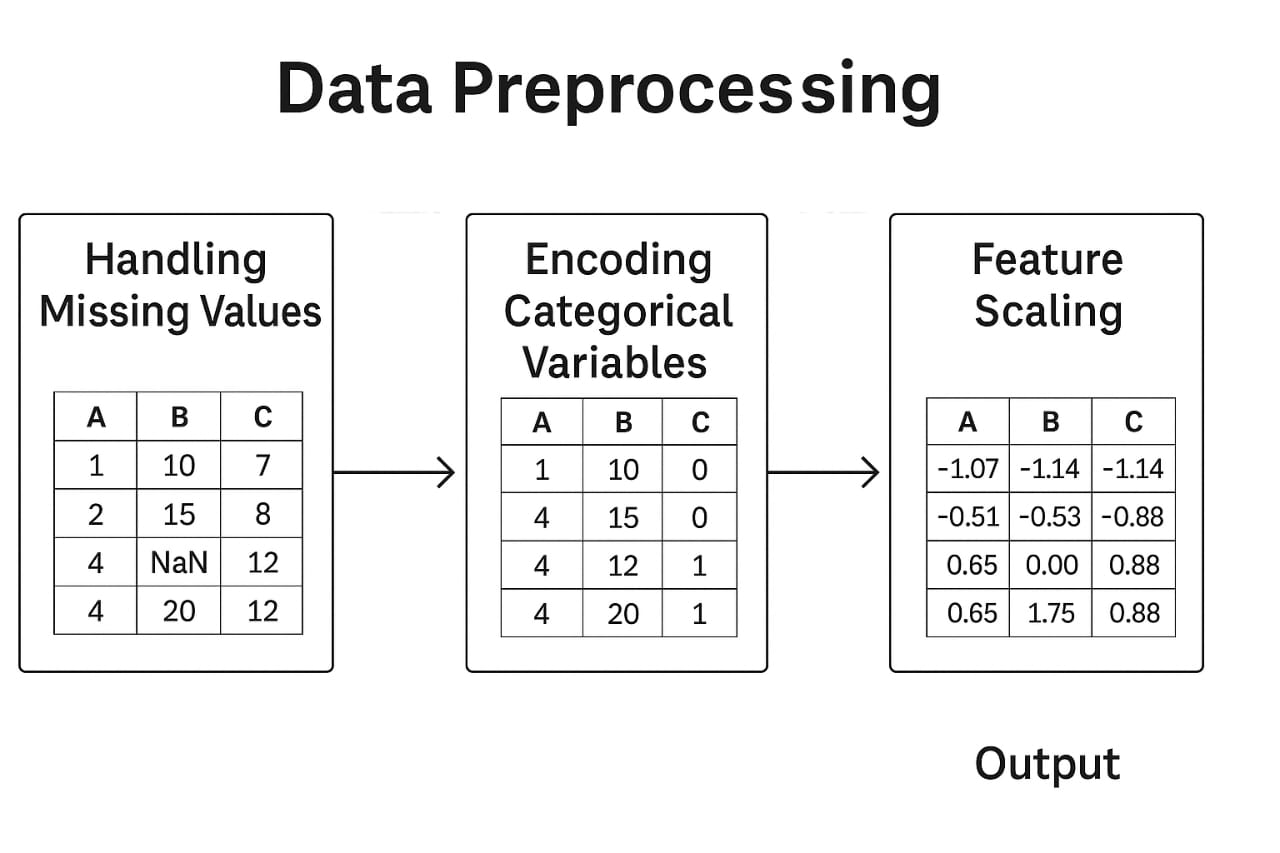
* **Visualization Tools:** matplotlib, seaborn, plotly
* **Programming Language**: Python
* **IDE/Notebook:** Google Colab, Jupyter Notebook
* **Libraries:** pandas, numpy, scikit-learn, xgboost, imbalanced-learn, seaborn, matplotlib, SHAP
* **Deployment:** Streamlit



* *Type:(public, private, synthetic)*
* *Size and structure (number of rows/columns)*
* *Include df.head()*

# 6.Dataset Description

* **Dataset Name**: Credit Card Fraud Detection Dataset
* **Source:** Kaggle
* **Type of Data**: Structured, Tabular
* **Number of Records**: 284,807 transactions
* **Features:** 31 (anonymized + 'Time', 'Amount')
* **Target Variable**: 'Class' (0 = Genuine, 1 = Fraud)
* **Dataset Status**: Static (snapshot in time, not updating)
* **Screenshot**



# 7.Data Preprocessing

# Removed duplicate records.

# No missing values in the dataset.

# Scaled the 'Amount' feature using StandardScaler.

# Addressed class imbalance using SMOTE.

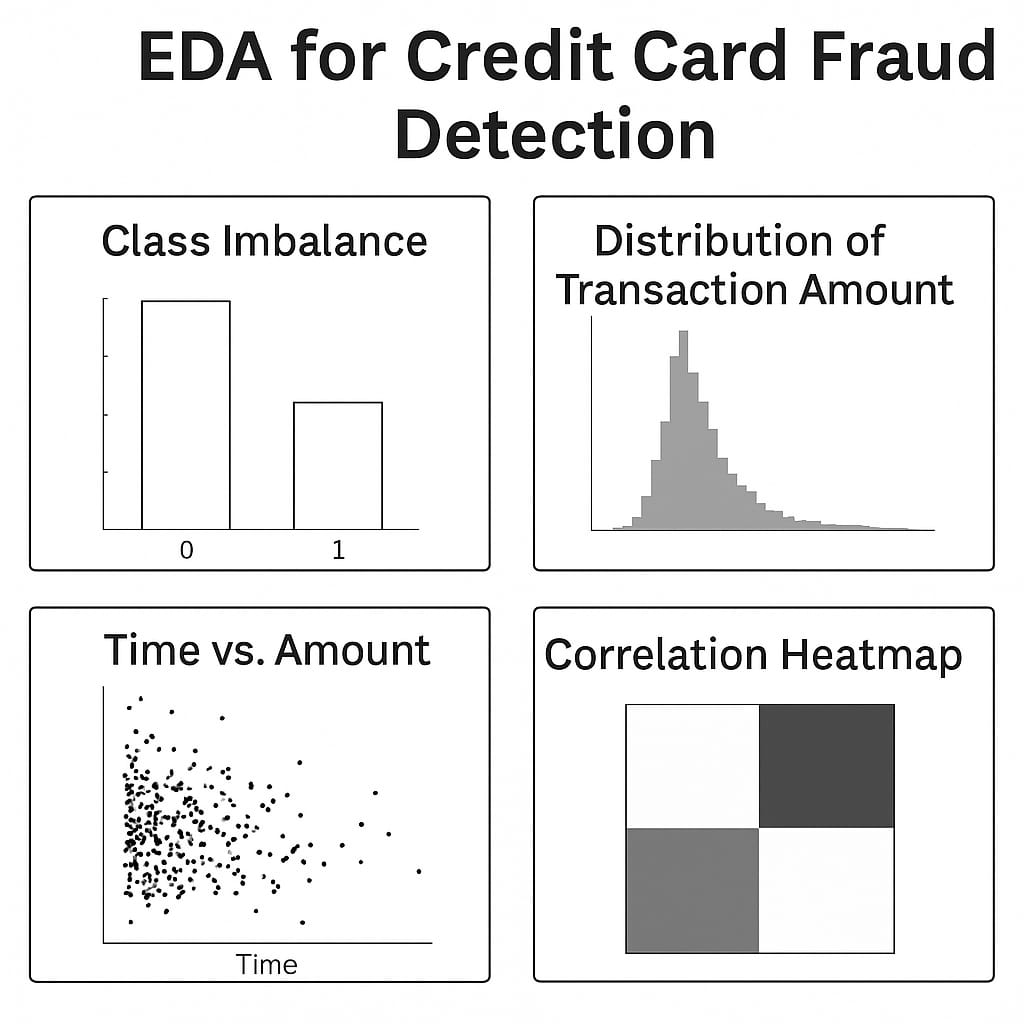
# 8.Exploratory Data Analysis (EDA)

**Univariate Analysis:**

* Fraud transactions are extremely rare (0.17%).
* Fraudulent transaction amounts vary widely.
* Bivariate/Multivariate Analysis:
* Strong correlations observed in certain principal components.
* Scatter plots showed different clusters for fraud and genuine.

Insights:

* Certain features (like V4, V11) show distinct behaviors for fraud vs genuine.
* **Screenshot**

****

# 9.Feature Engineering

* *Created 'Hour of Transaction' from time.*
* *Performed feature selection using correlation matrix and feature importance.*
* *Reduced dimensionality where necessary.*

# 10.Model Building

**Models Implemented:**

* Logistic Regression (Baseline)
* Random Forest Classifier (Ensemble Method)
* XGBoost Classifier (Boosted Trees)
* Autoencoder (Anomaly Detection Approach)
* Model Selection Justification:
* Random Forest and XGBoost handle imbalance and non-linear relationships well.
* Autoencoders are ideal for anomaly detection in rare-event datasets.

# 11.Model Evaluation

# *Metrics: Accuracy, Precision, Recall, F1-score, ROC-AUC*

* Best model (XGBoost):
* Precision: 0.91
* Recall: 0.87
* F1-score: 0.89

# 12.Deployment

* *Method: Streamlit Cloud*
* *Public Link: (Insert deployment URL)*
* *UI Screenshot: (Attach UI image)*
* *Sample Prediction Output: Fraudulent / Not Fraudulent*

# 13.Source code

[https://github.com/vishnupriya062006/nm-project-phase-3/blob/main/Source code.py](https://github.com/vishnupriya062006/nm-project-phase-3/blob/main/Source%20code.py)

# 14.Future scope

* Integrate real-time data pipeline using Kafka.
* Improve model robustness with more granular features (e.g., merchant category).
* Implement alert system via SMS/email for flagged transactions.

# 15. Team Members and Roles

* **Poojasri.DN—Team lead and Designer**: Designed the deployment interface and visual elements.
* **Priyanka.K— Developer:** Developed and implemented the machine learning models.
* **Ranjikha.R— Documentation:** Handled project write-ups, markdowns, and reporting.
* **Priyadharshini.M,Vishnu priya.S— Presentation**: verbal explanations, and project demonstration.