

Phase-2 Submission

Student Name: Nithyashree S

Register Number: 410723104056

Institution: Dhanalakshmi College of Engineering

Department: CSE

Date of Submission: 07.05.2025

GitHub Repository Link: https://github.com/Nithyashree-S-06/NM_NithyashreeS_DS.git

1. Problem Statement

Guarding transaction with AI-powered credit card fraud detection and prevention

Operational inefficiencies:

Manual review processes that are costly and time-consuming Credit card fraud continues to be a growing threat in the digital payments ecosystem, costing financial institutions, merchants, and consumers billions annually. Traditional rule-based fraud detection systems are increasingly inadequate against

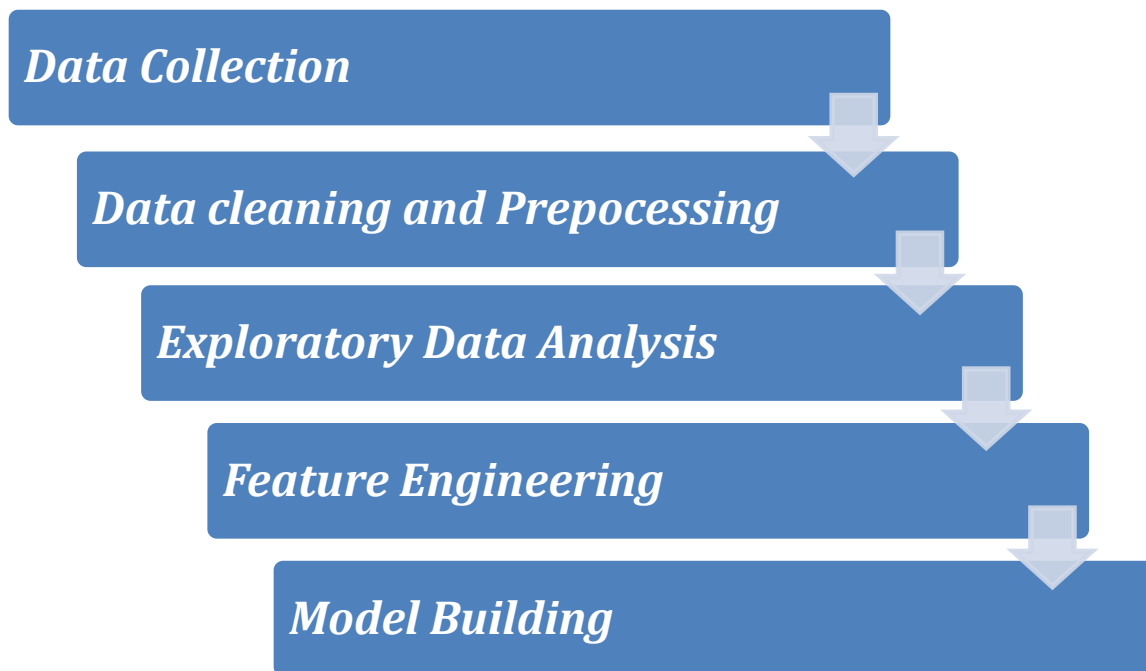
Sophisticated fraud patterns, leading to:

- ***High false positives :***
Legitimate transactions being declined, causing customer frustration
- ***False negatives:***
Fraudulent transactions slipping through, resulting in financial losses
- ***Inability to adapt:***
Static rules can't keep pace with evolving fraud techniques

2. Project Objectives

- *Collect and preprocess transaction data to prepare a clean, balanced dataset for analysis.*
- *Analyze data patterns to identify key indicators and trends related to fraudulent activities.*
- *Develop and train machine learning models to accurately detect and classify fraudulent transactions.*
- *Evaluate model performance using metrics like precision, recall, F1-score, and AUC-ROC to ensure reliability.*
- *Design a practical fraud prevention framework for integrating the model into real-time transaction systems.*

3. Flowchart of the Project Workflow



4. Data Description

- **Dataset Name:** Credit card transaction data set
- **Source:** Kaggle
- **Type of Data:** Structured tabular data
- **Records and Features:** 100001 transactions data set (categorical + numerical data) and 8 features
- **Target Variable :** Is fraud(1= Fraud, 0 = Legitimate)
- **Static or Dynamic :**static data set
- **Attributes Covered :** TransactionID, TransactionDate, Amount, MerchantID, TransactionType , Location and IsFraud

5. Data Preprocessing

- Checked and confirmed no missing values
- Converted Date to datetime format and extracted hour-based features
- Encoded categorical variables like TransactionType and Location
- Normalized Amount using MinMaxScaler
- Handled imbalance using SMOTE to oversample fraudulent cases
- Split dataset into 80% training and 20% test sets

6. Exploratory Data Analysis (EDA)

- **Univariate Analysis:**

- *Distribution of Amount (right-skewed)*
- *Count plot showing class imbalance*
- *Bivariate/Multivariate Analysis:*
 - *Boxplots of Amount by IsFraud*
 - *Bar plots of TransactionType and Location grouped by fraud*
- *Insights Summary:*
 - *Certain merchant IDs and transaction types correlate with fraud*
 - *High-value transactions are more likely to be flagged as fraud*

7. Feature Engineering

- *Extracted hour-of-day and weekday from Date*
- *Created Amount_per_hour = Amount / transaction hour*
- *Used one-hot encoding for categorical fields*
- *Removed low variance or redundant features*

8. Model Building

- *Algorithms Used:*
 - *Logistic Regression (Baseline)*
 - *Random Forest Classifier*
 - *XG Boost Classifier*
 - *Isolation Forest (Anomaly Detection)*
- *Data Split:*

- *Train-test split: 80% training, 20% testing using stratify=y*

➤ ***Metrics Evaluated:***

- *Precision, Recall, F1-Score, ROC-AUC*
- *Confusion Matrix*

9. Visualization of Results & Model Insights

➤ ***Feature Importance:***

- *XG Boost and Random Forest identified V14, V10, V12, and V17 as critical predictors*

➤ ***Performance Summary:***

- *XGBoost achieved the highest F1-score and AUC.*
- *Logistic Regression had good precision but lower recall.*
- *Random Forest balanced both.*

➤ ***Graphs & Charts:***

- *ROC Curves for all models.*
- *Confusion matrix for best-performing model.*
- *Feature importance bar chart.*

10. Tools and Technologies Used

➤ ***Language:*** Python 3

➤ ***Environment:*** Jupyter Notebook / Google Colab

➤ ***Libraries:***

- *pandas, numpy – Data Manipulation*
- *matplotlib, seaborn, plotly – Visualization*
- *scikit-learn, xgboost, imbalanced-learn – ML Models*

- *Gradio– Model Deployment Interface*

11. Team Members and Contributions

<i>S.No</i>	<i>Team members</i>	<i>Roles</i>
<i>1.</i>	<i>Nirosha M</i>	<i>EDA</i>
<i>2.</i>	<i>Nithyashree S</i>	<i>Data cleaning</i>
<i>3.</i>	<i>Poorna Kala G</i>	<i>Feature Engineering</i>
<i>4.</i>	<i>Yalini Nachiyar S</i>	<i>Model development and Documentation and reporting</i>