Phase-3 Submission Template

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

Student Name: [R.KANISH KUMAR]

Register Number: [620123106047]

Institution: [AVS ENGINEERING COLLEGE]

Department: [ELECTRONICS COMMUNICATION AND ENGINEERING]

Date of Submission: [17-05-2025]

Github Repository Link: [https://github.com/kanishkumar/credit-card-fraud-detection]

1. Problem Statement

Credit card fraud is a persistent issue in financial transactions, costing businesses and consumers billions annually.
This project addresses the problem by building an AI-powered system for detecting and preventing fraudulent credit card transactions in real-time.
It is a classification problem where transactions are labeled as fraudulent or legitimate based on input features.
Such systems are crucial for banks and financial institutions to enhance trust and reduce financial losses.

2. Abstract

This project aims to develop an AI-based credit card fraud detection system using machine learning.
The system will analyze historical transaction data to identify patterns indicative of fraud.
The objective is to minimize false positives while accurately flagging fraudulent activities.
Using preprocessing, exploratory data analysis, feature engineering, and model building, we evaluate several algorithms and deploy the best one.
The final model is accessible through a user-friendly interface to demonstrate its practical application.

3. System Requirements

Hardware:
- Minimum 8GB RAM, Intel i5 processor or equivalent

Software:
- Python 3.8+

- Libraries: pandas, numpy, matplotlib, seaborn, scikit-learn, xgboost, -learn
- IDE: Google Colab or Jupyter Notebook

4. Objectives
- Detect fraudulent credit card transactions in real time
- Minimize false positives to avoid inconveniencing genuine users
- Provide insights into patterns of fraudulent behavior
- Enhance the security of digital financial systems

5. Flowchart of Project Workflow
[Insert flowchart image here: Data Collection → Preprocessing → EDA → Feature Engineering → Modeling → Evaluation → Deployment]

6. Dataset Description
- Source: Kaggle (Credit Card Fraud Detection dataset)
- Type: Public
- Size: ~284,807 rows and 31 columns (Time, V1-V28, Amount, Class)

7. Data Preprocessing
- Handled missing values: None found
- Removed duplicates
- Detected and treated outliers using IQR method
- Applied standard scaling to 'Amount' and 'Time'
- Screenshots of before/after transformations are attached

8. Exploratory Data Analysis (EDA)
- Used histograms, boxplots, and heatmaps
- Found significant correlation between certain features and the 'Class' label
- Fraudulent transactions tend to have smaller amounts and appear more clustered in time
- Screenshots of visualizations are included

9. Feature Engineering
- Created interaction features between highly correlated variables
- Used PCA components as features to reduce dimensionality
- Feature selection via recursive elimination and importance ranking

10. Model Building

- Tried Logistic Regression, Random Forest, XGBoost
- Chose XGBoost for best balance between performance and interpretability
- Included screenshots of model training logs and parameter tuning

11. Model Evaluation

- Metrics: Accuracy, Precision, Recall, F1-score, ROC-AUC- XGBoost achieved highest AUC (~0.98)
- Included confusion matrix, ROC curve
- Error analysis showed majority of false negatives were low-amount frauds

12. Deployment

- Deployed using Streamlit on Streamlit Cloud
- Deployment method: Streamlit app with form input for transaction features
- Public link: [Insert Streamlit link]
- UI Screenshot and sample outputs included

13. Source code

[Provide link to the complete source code hosted on GitHub]

14. Future scope

- Incorporate real-time data streaming for live detection
- Use deep learning models for sequential analysis (LSTM)
- Build a feedback loop with user validation for continual model improvement

15. Team Members and Roles

- [R.KANISH KUMAR]: Data Collection and Preprocessing
- [S.HARISH]: EDA and Feature Engineering
- [M.JEEVAN]: Model Training and Evaluation
- [S.JAYAVEL]: Deployment and Documentation