**Phase-1 Submission**

**Student Name:** JAMES ALWIN J

**Register Number:** 510823243016

**Institution:** Ganadipathy Tulsi’s Jain Engineering College

**Department:** Artificial Intelligence and Data Science

**Date of Submission:**

# Project Title: Guarding Transactions With AI-Powered

# Credit Card Fraud Detection and Prevention

# PHASE – 1

# 1.Problem Statement

Credit card fraud is a growing global threat, resulting in billions of dollars in losses annually. Traditional detection systems are often slow, rigid, and inaccurate, failing to keep up with evolving fraud tactics. This project focuses on leveraging AI-powered fraud detection to identify suspicious transactions in real time—enhancing security, reducing false positives, and protecting both users and financial institutions.

# 2.Objectives of the Project

Develop a machine learning model that accurately identifies fraudulent transactions using historical credit card transaction data.

Achieve a balance between high detection rate and low false positive rate to ensure legitimate transactions are not unnecessarily flagged.

Understand patterns and features that differentiate fraudulent transactions from genuine ones.

Explore the feasibility of integrating the model into a real-time transaction monitoring system.

Provide visual insights and an interpretable report for stakeholders to understand model process interactively.

# 3.Scope of the Project

# *Analyze key features such as transaction amount, time, anonymized PCA features, and class labels indicating fraud.*

# *Focus on binary classification to distinguish fraudulent transactions from legitimate ones.*

# *Address class imbalance through resampling techniques like SMOTE or undersampling.*

# 4.Data Sources

**Dataset:** Credit Card Fraud Detection

**Source:** Kaggle - Credit Card Fraud Detection Dataset

**Description:** This dataset contains transactions made by European cardholders in September 2013. It contains 284,807 transactions, among which only 492 are fraudulent (0.172%).

**Type:** Public and static dataset (not updated in real-time). Features are anonymized due to confidentiality concerns and transformed using Principal Component Analysis (PCA).

# 5.High-Level Methodology

* **Data Collection:**

Obtain the dataset from Kaggle as a CSV file and load it into a Jupyter Notebook or Google Colab environment for preprocessing.

* **Data Cleaning:**

Check for and handle missing or null values.

Remove or impute any outliers or anomalies.

Normalize features such as ‘Time’ and ‘Amount’ for better model performance.

Handle class imbalance using oversampling (SMOTE) or under-sampling methods.

* **Exploratory Data Analysis (EDA):**

Visualize class distribution to understand imbalance.

Use boxplots, histograms, and pairplots to identify feature relationships and distributions.

Correlation heatmap to identify the most relevant PCA features.

Time-series plots to observe trends and timing of fraudulent activities.

* **Feature Engineering:**

Scale the ‘Time’ and ‘Amount’ columns using Min-Max or Standard Scaler.

Consider creating new features such as transaction frequency by cardholder, average spend, or time since last transaction.

Reduce dimensionality if needed using PCA (though data is already partially PCA-transformed).

* **Model Building:**

Experiment with various machine learning algorithms:

➢ Logistic Regression

➢ Decision Trees and Random Forest

➢ XGBoost and LightGBM

➢ Support Vector Machines (SVM)

➢ Neural Networks (MLP)

Compare their performance on training and validation sets.

* **Model Evaluation:**

Use metrics appropriate for imbalanced classification:

* + - Precision, Recall, and F1-Score
    - Area Under the ROC Curve (AUC-ROC)
    - Confusion Matrix
    - Apply cross-validation for generalizability.
* **Visualization & Interpretation:**

Present model evaluation results using confusion matrices and ROC curves.

Use SHAP or LIME for model interpretability to identify key contributing features in fraud detection.

Create interactive plots and dashboards for deeper insights.

* **Deployment:**

Simulate deployment using tools like Streamlit or Flask.

Build a web-based interface where users can upload transaction data and get instant fraud predictions.

Visualize model decisions and metrics through an interactive dashboard.

# 6.Tools and Technologies

* **Programming Language:**

Python

* **Notebook/IDE:**

Jupyter Notebook / Google Colab

* **Libraries:**

pandas, numpy, seaborn, matplotlib, plotly scikit-learn, xgboost, lightgbm, imbalanced-learn, tensorflow (if using deep learning) shap, lime for explainability

* **Deployment** **Tools:**

Streamlit, Flask, or Gradio for creating a front-end dashboard

GitHub for version control

Heroku or Render for hosting the app

# 7.Team Members and Roles

Harish B – Development

James Alwin J – Design

Jaya Monisha J – Documentation

Aravindiya N– Communication