**GUARDING TRANSACTION WITH AI-POWERED CREDIT CARD**

**FRAUD DETECTION AND PREVENTION**

**Student Name:** S.Dharshini

**Register Number:** 422223243012

**Institution:** Surya Group of Institution

**Department:** Artificial Intelligence and Data Science

**Date of Submission:** 29-04-2025

**Github Repository Link:** [**https://github.com/dharshini-s13/Dharshinishivani**](https://github.com/dharshini-s13/Dharshinishivani)

**1. Problem Statement**

In the age of digital transactions, credit card fraud is a growing concern leading to massive financial losses and customer distrust. The objective of this project is to build an AI-powered system that can detect and prevent fraudulent credit card transactions in real-time.

Problem Type: Classification Problem

Impact: Saving billions for financial institutions and protecting customers.

**2. Project Objectives**

* Develop a model to classify fraudulent transactions.
* Minimize false positives.
* Ensure real-time detection.
* Handle highly imbalanced data.

**3. Flowchart of the Project Workflow**

Start -> Data Collection -> Data Preprocessing -> EDA -> Feature Engineering -> Model Building -> Model Evaluation -> Deployment Preparation -> End

**4. Data Description**

Dataset Name: Credit Card Fraud Detection Dataset

Source: Kaggle

Type: Structured tabular data

Records and Features: 284,807 transactions, 31 features

Dataset Type: Static

Target Variable: Class (0: Genuine, 1: Fraudulent)

**5. Data Preprocessing**

* No missing values
* Duplicates removed
* Outliers treated
* Amount feature normalized
* SMOTE applied for imbalance handling

**6. Exploratory Data Analysis (EDA)**

Univariate: Rare frauds (~0.17%)

Bivariate: Certain features correlate with fraud

Insights: V14, V12, V10 important

**7. Feature Engineering**

* Created Amount\_log (log transformation)
* Feature scaling
* No dimensionality reduction needed

**8. Model Building**

Models: Logistic Regression, Random Forest

Justification: Interpretability and handling imbalance

Evaluation Metrics: Precision, Recall, F1-Score, ROC-AUC

**9. Visualization of Results & Model Insights**

* Feature Importance: V14, V12, V10High recall priori
* Confusion Matrix
* ROC Curve (AUC = 0.98)
* tized

**10. Tools and Technologies Used**

Programming Language: Python

IDE: Google Colab

Libraries: pandas, numpy, scikit-learn, seaborn, matplotlib, imbalanced-learn

Visualization Tools: matplotlib, seaborn, plotly

**11. Team Members and Contribution**

1. J.Aseena : Data cleaning-EDA
2. U.Dayasri : Feature Engineering
3. S.Dharshini(2005): Model Development
4. S.Dharshini(2006):Documentation