**CREDIT CARD FRAUD DETECTION USING MACHINE LEARNING**

PHASE-1 DOCUMENT SUBMISSION

PROJECT: CREDIT CARD FRAUD DETECTION

**PHASE 1: PROBLEM DEFINITION AND DESIGN THINKING**

**PROBLEM DEFINITION:**

The objective of this project is to develop a machine learning model that creates a solution that can accurately identify fraudulent transactions while minimizing false positives. This project involves data preprocessing, feature engineering, model selection, training, and evaluation to create a robust fraud detection system.

**DESIGN THINKING:**

Detecting and preventing credit card fraud is a critical use case of data science and machine learning. Credit card fraud detection involves identifying fraudulent transactions or activities in credit card transactions to protect both cardholders and financial institutions.Here’s an overview of how data science can be applied to tackle this problem:

1. DATA SOURCE:

Utilize a data set containing transaction data, including features such as transaction amount, time stamp, merchant information, and card details.

**Dataset Link: [https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud](https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud" \t "https://courses.myclass.skillup.online/courses/course-v1:IBM+ADS101+2023_B1/courseware/d8660830b7ec4f2e8158584fd8319a7d/26f511b1c44a4ebca5e36f0c238346c9/[object Object])**

1. DATA PREPROCESSING:

Clean and preprocess the data , handle missing values, correct any inconsistencies in the data and normalize features.

1. FEATURE PREPROCESSING:

Create relevant features that can help the model distinguish between legitimate and fraudulent transactions. Features could include transaction frequency, time of day, location consistency, etc.

1. MODEL SELECTION:

Choose appropriate machine learning algorithms or deep learning architectures for credit card fraud detection. Common choices include logistic regression, decision trees, random forests, gradient boosting, and neural networks.

- Consider ensemble methods to combine the strengths of multiple algorithms

1. MODEL TRAINING:

Train the selected models on the training data using appropriate performance metrics such as precision, recall, F1-score, and area under the ROC curve (AUC-ROC).

- Use techniques like oversampling or undersampling to address class imbalance issues (fraudulent transactions are often rare compared to legitimate ones).

1. EVALUATION:-

Evaluate the model's performance on the validation set and fine-tune hyperparameters if needed.

- Use techniques like cross-validation to ensure robustness.