# Credit Card Fraud Detection Synopsis

## Project Title:

Credit Card Fraud Detection Using Machine Learning

## Project Synopsis:

This project aims to develop a machine learning model to detect credit card fraud by analyzing transaction patterns and identifying anomalies. The model will help financial institutions minimize losses and enhance customer trust by flagging fraudulent transactions in real-time.

## Objective:

- Develop a predictive model for detecting fraudulent transactions.  
- Identify key indicators of fraud and generate actionable insights.  
- Create a scalable and user-friendly system for continuous fraud monitoring.

## Data Sources and Features:

1. Data Sources:  
 - Historical transaction data (e.g., transaction amount, timestamp, location).  
 - Customer demographics (e.g., age, account history).  
 - Merchant data (e.g., merchant category, location).  
2. Features:  
 - Transaction attributes (e.g., amount, frequency, time of transaction).  
 - Customer behavior patterns (e.g., usual spending locations, typical purchase categories).  
 - Merchant risk levels (e.g., high-risk categories, suspicious locations).

## Risk Factors:

- Class Imbalance: Fraudulent transactions represent a small percentage of total transactions.  
- Data Quality: Missing or inconsistent transaction details may affect model accuracy.  
- Real-time Processing: Ensuring minimal latency for real-time fraud detection.

## Data Preprocessing:

1. Handle missing data using imputation techniques.  
2. Normalize transaction amounts and encode categorical variables.  
3. Address class imbalance using methods like SMOTE or undersampling.  
4. Split data into training, validation, and test sets.

## Model Selection:

- Baseline Models: Logistic Regression, Decision Trees.  
- Advanced Models: Random Forest, Gradient Boosting (e.g., XGBoost), Neural Networks.  
  
Reasoning: Advanced models capture complex patterns in transaction data, critical for accurate fraud detection.

## Exploratory Data Analysis (EDA):

1. Analyze transaction patterns by time, location, and merchant type.  
2. Identify correlations between features and fraudulent activity.  
3. Visualize key trends using histograms, scatterplots, and time-series analysis.

## Model Evaluation:

- Metrics: Precision, Recall, F1-Score, AUC-ROC, and Log Loss.  
- Validation: Cross-validation techniques to ensure robustness.

## Model Deployment:

1. Platform: Deploy as an API using Flask or FastAPI.  
2. Interface: Provide a dashboard for real-time transaction monitoring and alerts.  
3. Monitoring: Continuous model performance tracking and retraining as new data becomes available.