Requirement Specifics — Credit Card Fraud Detection Project

1. Introduction

This project aims to build a machine learning model that detects fraudulent credit card transactions with high accuracy while handling imbalanced data. The model will analyze transaction features and classify each transaction as **Fraud (1)** or **Not fraud(0)**.

2. Project Objectives

- Detect fraudulent transactions with high recall (catch as many frauds as possible).
- Handle large, highly imbalanced datasets.
- Compare models (KNN, Logistic Regression, Random Forest, etc.).
- Provide evaluation metrics like **Accuracy, Precision, Recall, F1-score, Confusion Matrix**.

3. Dataset Requirements

- Use the public **Credit Card Fraud Detection dataset (Kaggle)**.
- The dataset must include:
  - 284,807 transactions
  - 492 fraud cases (highly imbalanced)
  - 30 input features:
    - **V1–V28** (PCA-transformed features)
    - **Time**
    - **Amount**
  - Target label: **Class (0 = normal, 1 = fraud)**

4. Data Preprocessing Requirements

✓ Remove missing or corrupted values
✓ Apply **scaling** to numerical features:

- StandardScaler for: Time, Amount
- scaled_Amount

✓ Split dataset into:

- 80% training
- 20% testing

✓ Handle **imbalanced data** using:

- SMOTE
- Undersampling / oversampling
- Class weights

5. Feature Requirements

The model must use relevant features such as:

- **Time**
- **Amount**
- **V1–V28**
- Created features:
  - scaled_Time
  - scaled_Amount

Remove features only if correlation analysis shows no importance.

6. Model Requirements

The system must support training and testing of:

Required Models

- KNN classifier
- Logistic Regression
- Random Forest

Model Behaviors

- Hyperparameter tuning using GridSearchCV
- Train with cross-validation (k=5)
- Compare performance metrics

7. Evaluation Requirements

Model must provide:

- Accuracy
- Precision
- Recall (very important for fraud)
- F1-score
- Confusion matrix
- ROC curve
- AUC score

Fraud detection priority:

- **Recall for Class 1 must be high** (catching fraud)
- Accept lower precision if needed

8. System Requirements

- Python 3.8+
- Libraries:
  - pandas
  - numpy
  - scikit-learn
  - matplotlib
  - seaborn
  - imblearn (SMOTE)

- Jupyter Notebook or Google Colab

9. Output Requirements

The system must output:

- Fraud predictions (0/1)
- Evaluation report
- Visualizations:
  - Correlation heatmap
  - Fraud vs valid counts
  - ROC curve
  - Confusion matrix
  - Feature importance (if supported)

10. Constraints

- Dataset is highly imbalanced → must handle using methods.
- PCA-transformed features → cannot interpret directly.
- Fraud cases are extremely rare → risk of overfitting.

11. Success Criteria

- Minimum Recall (Class 1): **>80%**
- F1-score for fraud: **>0.70**
- AUC score: **>0.90**