Department of Computer Science & Engineering

Meerut Institute of Technology,Meerut

(Affiliated to APJ Abdul Kalam Technical University, Lucknow)

Statement of Problem

Of

Major Project

(UPI Fraud Detection System)

Submitted By

S.No.	Name	Roll No.	Section	Year
1	RICHA KUMARI (Group Leader)	2302921530043	E	2nd Year (IV Sem)
2	NITISH KUMAR	2302921530032	E	2nd Year (IV Sem)
3	VISHAL KUMAR	2302921530060	E	2nd Year (IV Sem)
4	PRITI KUMARI	2302921530036	E	2nd Year (IV Sem)

Submitted to

Mr. Ayush Singhal

(Assistant Professor & Coordinator – Department of CSE)

Table of Content

Introduction

Unified Payments Interface (UPI) is a widely used real-time payment system developed by the National Payments Corporation of India (NPCI). With its rapid adoption, there has also been a surge in fraudulent transactions. This project aims to develop a **UPI Fraud Detection System** that can identify and flag suspicious transactions in real-time using data analytics and machine learning techniques.

Problem Statement

The primary objective is to detect fraudulent UPI transactions by analyzing patterns in the transaction data. This involves building a predictive model capable of classifying transactions as legitimate or fraudulent based on the identified anomalies.

Objectives

- **Data Analysis:** Identify transaction patterns that indicate potential fraud.
- **Predictive Model:** Implement machine learning models to detect fraudulent transactions.
- **User Interface:** Develop a dashboard for monitoring and flagging transactions in real-time.
- **Reporting:** Generate reports and visualizations for fraud trend analysis.

Scope of the Project

- **Data Collection:** Gather UPI transaction data (synthetic or publicly available datasets).
- **Data Preprocessing:** Clean the dataset, handle missing values, and normalize data.
- **Feature Engineering:** Extract critical features such as transaction amount, frequency, time, location, device ID, etc.
- **Model Selection:** Implement and evaluate multiple ML algorithms:
    - Logistic Regression
    - Decision Trees
    - Random Forest
    - XGBoost
    - Neural Networks
- **Evaluation Metrics:** Assess model performance using:
    - Accuracy
    - Precision
    - Recall
    - F1 Score
    - ROC-AUC curve
- **Deployment:** Develop a real-time monitoring dashboard using frameworks like **Streamlit** or **Dash**.
- **Alert System:** Implement notifications via email or SMS for flagged transactions.

Literature Review

- Study existing fraud detection systems in the banking sector.
- Review machine learning models commonly used for fraud detection.
- Understand UPI transaction protocols, data flow, and potential vulnerabilities.

System Architecture

The system architecture is divided into several modules:

- **Data Collection Module:** Fetches transaction data from the UPI network or a simulated dataset.
- **Data Preprocessing Module:** Cleans data and standardizes formats.
- **Feature Engineering Module:** Extracts relevant features for model training.
- **Model Training Module:** Trains ML models using labeled datasets.
- **Prediction Module:** Predicts whether a transaction is fraudulent or legitimate.
- **Alerting Module:** Sends notifications for suspicious transactions.
- **Dashboard Module:** Visualizes transaction data and fraud analysis in real-time.

Technology Stack

- **Programming Language:** Python
- **Data Analysis:** Pandas, NumPy
- **Data Visualization:** Matplotlib, Seaborn
- **Machine Learning:** Scikit-Learn, TensorFlow, PyTorch
- **Database:** MySQL, MongoDB
- **Web Framework:** Flask, Django
- **Dashboard:** Streamlit, Plotly Dash
- **Notification System:** Twilio, Email SMTP

Data Flow Diagram (DFD)

- **Data Collection:** Fetch transaction data.
- **Data Preprocessing:** Clean and standardize data for analysis.
- **Feature Extraction:** Extract key indicators for fraud detection.
- **Model Training:** Apply and train ML algorithms.
- **Prediction:** Classify transactions as fraudulent or legitimate.
- **Alerting:** Notify stakeholders in case of suspicious transactions.
- **Dashboard:** Display transaction data and fraud analysis in real-time.

Implementation Plan

- **Phase 1: Data Collection and Preprocessing**
  - Gather synthetic or publicly available datasets.
  - Clean and preprocess data for model training.
- **Phase 2: Model Development and Training**
  - Implement multiple ML models.
  - Compare model performance and select the best model.
- **Phase 3: System Integration**
  - Integrate the selected model with a web interface.
  - Develop a real-time monitoring dashboard.
- **Phase 4: Testing and Deployment**
  - Test the system with synthetic data.
  - Deploy the system on a local server or cloud.

Evaluation Metrics

- **Confusion Matrix:** Visual representation of true positives, false positives, true negatives, and false negatives.
- **Accuracy Score:** Measures the percentage of correctly classified transactions.
- **Precision, Recall, F1 Score:** Assess the balance between false positives and false negatives.
- **ROC-AUC Curve:** Evaluates the model's ability to differentiate between fraudulent and legitimate transactions.

Expected Outcomes

- A fully functional fraud detection system with real-time transaction monitoring.
- Alert system for notifying users of suspicious transactions.
- Comprehensive reports and visualizations for fraud analysis and pattern recognition.

References

□ **UPI System and Architecture:**
[1] National Payments Corporation of India (NPCI), “Unified Payments Interface (UPI) Overview,” [Online]. Available: https://www.npci.org.in. Accessed: May 18, 2025.

□ **Fraud Detection Techniques in Banking:**
[2] S. R. Bharathi and V. Geetha, "Credit Card Fraud Detection using Machine Learning Algorithms," *IEEE Access*, vol. 7, pp. 184082-184090, 2019. doi: 10.1109/ACCESS.2019.2949295.

□ **Machine Learning Models for Fraud Detection:**
[3] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. Waltham, MA, USA: Morgan Kaufmann, 2012.

□ **Data Analysis and Visualization Libraries:**
[4] W. McKinney, "Data Structures for Statistical Computing in Python," in *Proc. 9th Python in Science Conf.*, Austin, TX, USA, 2010, pp. 56-61. doi: 10.25080/Majora-92bf1922-00a.

□ **Alerting Systems in Fraud Detection:**
[5] Twilio Inc., “Twilio SMS API Documentation,” [Online]. Available: https://www.twilio.com/docs/sms. Accessed: May 18, 2025.

□ **Streamlit for Dashboard Development:**
[6] Streamlit Inc., “Streamlit Documentation,” [Online]. Available: https://docs.streamlit.io. Accessed: May 18, 2025.