

ONLINE PAYMENT FRAUD DETECTION USING MACHINE LEARNING

1. INTRODUCTION

1.1 Project Overview

Project Title:Online Payment Fraud Detection Using Machine Learning

Team ID: LTVIP2026TMIDS63406

Team Members:

- Mannaru Prem Kumar
- Anda Tharun
- Chavva Tarun
- Gaduputi Hima Varshini

This project is a web-based intelligent system that detects fraudulent online transactions using machine learning. The system analyzes transaction data such as amount, transaction type, sender and receiver balances to identify whether a transaction is fraudulent or safe.

1.2 Purpose

The purpose of this project is to:

- Prevent financial fraud in online payment systems
- Help banks and digital payment platforms identify suspicious transactions
- Provide a fast, automated fraud detection solution using machine learning

2. IDEATION PHASE

2.1 Problem Statement

Online transactions are increasing rapidly, and so are online frauds. Manual verification is slow, costly, and inefficient. Fraudulent transactions cause huge financial losses to customers and banks. Therefore, there is a need for an automated system that can detect fraud accurately in real time.

2.2 Empathy Map Canvas

Users	Feel	Think	Do	Pain Points
Bank Customers	Fear of losing money	“Is my money safe?”	Perform online transactions	Fraud risk
Banks	Concerned about losses	“How to detect fraud?”	Monitor transactions	Manual verification
Payment Apps	Want trust	“System must be reliable”	Process transactions	Security breaches

2.3 Brainstorming

Different solutions were considered:

- Rule-based fraud detection
- Manual verification
- AI-based fraud detection

Machine learning was chosen because it can detect hidden patterns and adapt to new fraud techniques.

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

1. User enters transaction details
2. System sends data to ML model
3. Model analyzes transaction
4. System shows fraud or safe result

3.2 Solution Requirement

Functional Requirements

- Accept transaction details
- Predict fraud or safe
- Display result

Non-Functional Requirements

- Fast response
- High accuracy
- Secure system

3.3 Data Flow Diagram

1. User → Web Form
2. Web Form → Flask Server
3. Flask → ML Model
4. Model → Prediction
5. Prediction → User Interface

3.4 Technology Stack

Layer	Technology
Frontend	HTML, CSS
Backend	Python, Flask
ML Model	Scikit-learn
Database	CSV Dataset
Tools	VS Code, GitHub

4. PROJECT DESIGN

4.1 Problem–Solution Fit

Online fraud is increasing. Our system provides an automated fraud detection tool, which fits the problem perfectly by improving security and reducing losses.

4.2 Proposed Solution

A web application integrated with a machine learning model that predicts fraudulent transactions using transaction details.

4.3 Solution Architecture

User → Web Page → Flask Server → ML Model → Result → User

5. PROJECT PLANNING & SCHEDULING

Phase	Duration
Data Collection	1 week
Model Training	2 weeks
UI Development	1 week
Backend Integration	1 week
Testing	1 week

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

- The system predicts results within seconds

- Handles multiple transactions efficiently
- Tested with multiple sample values

7. RESULTS

The model successfully classifies transactions as:

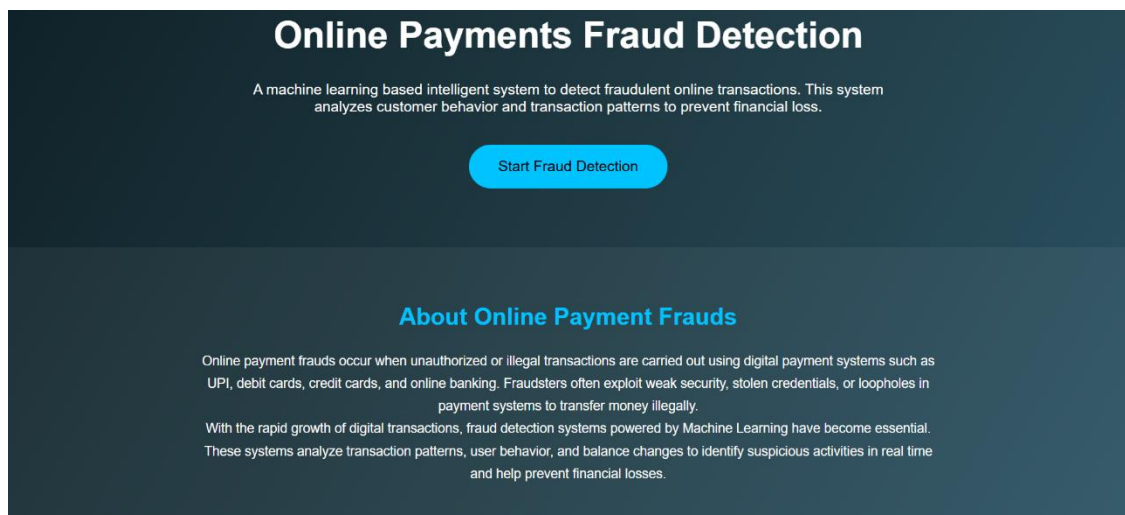
- Fraudulent
- Safe

The web interface shows the prediction clearly.

7.1 Output Screenshots

Screenshots include:

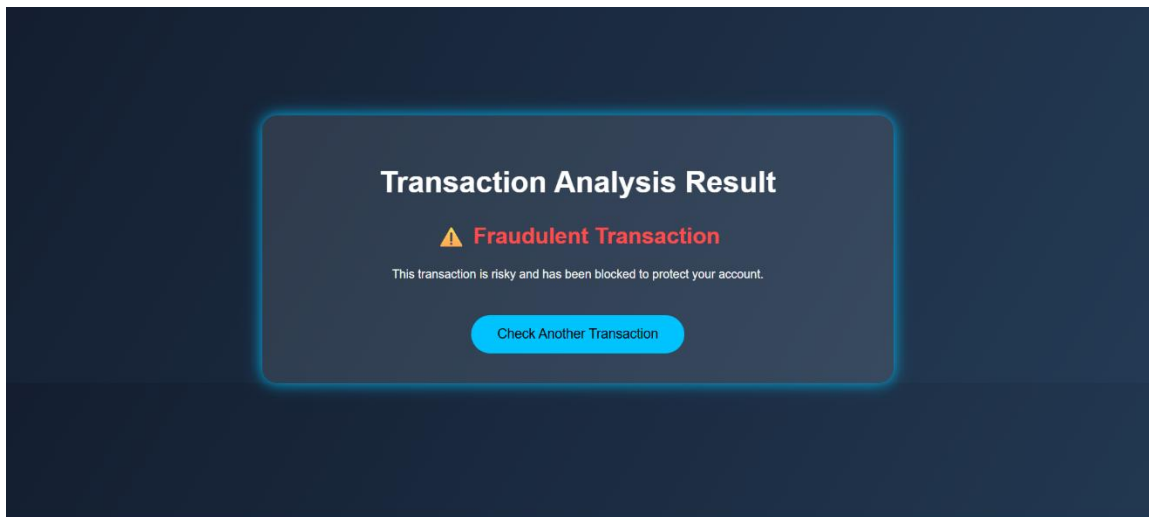
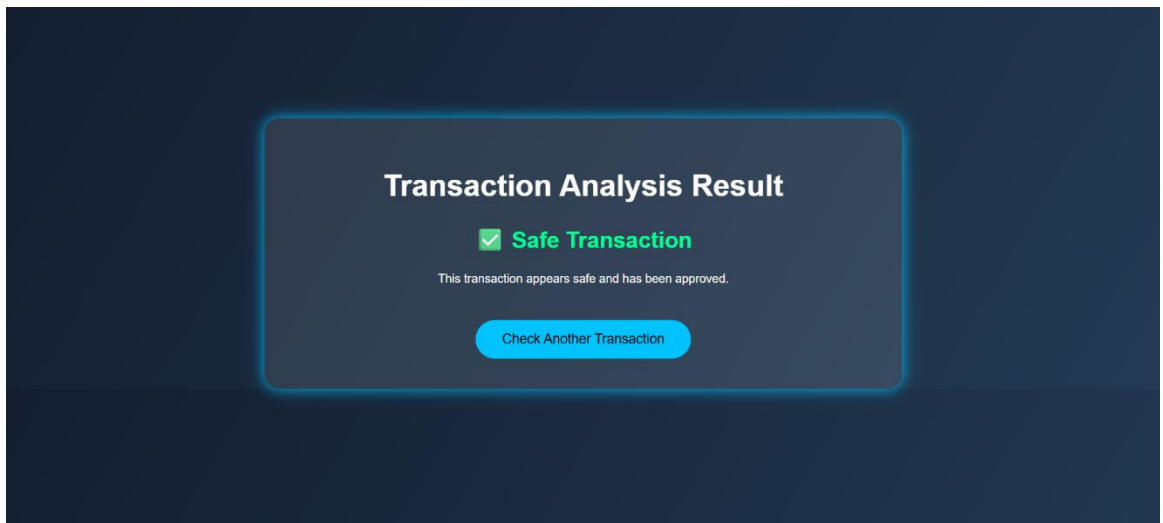
- Home page



- Fraud input page

The screenshot shows the 'Fraud input page' of the 'Online Payments Fraud Detection' system. The page has a light blue background. The title 'Online Payments Fraud Detection' is centered at the top. Below the title, the 'Step' is indicated as '1'. The form contains several input fields: 'Type' (a dropdown menu currently showing 'PAYMENT'), 'Amount' (a text box with '20000'), 'OldbalanceOrg' (a text box with '20000'), 'NewbalanceOrig' (a text box with '40000'), 'OldbalanceDest' (a text box with '30000'), and 'NewbalanceDest' (a text box with '40000'). At the bottom of the form, there is a large blue button labeled 'Check Fraud'.

- Fraud / Safe result page



8. ADVANTAGES & DISADVANTAGES

Advantages

- Fast fraud detection
- Reduces financial loss
- Automated system
- Easy to use

Disadvantages

- Depends on dataset quality
- May give false predictions
- Needs retraining for new fraud patterns
-

9. CONCLUSION

This project provides an efficient solution for detecting online payment fraud. By using machine learning, the system improves transaction security and helps financial institutions reduce fraud losses.

10. FUTURE SCOPE

- Add real-time fraud detection
- Use deep learning models
- Add user login and dashboards
- Mobile app integration
- Cloud deployment

11. APPENDIX

Source Code:

Available in the project folder

Dataset Link:

Kaggle – Online Payment Fraud Dataset

GitHub Repository:

<https://github.com/Prem-0427/online-fraud-detection>

Project Demo:

https://drive.google.com/file/d/1ojFNVaOuzDiyT2H6mQsM3cTnwBZLyHCU/view?usp=drive_link