Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	31 January 3035
Team ID	LTVIP2026TMIDS79606
Project Name	online payments fraud detection using machine learning
Maximum Marks	4 Marks

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

Example: Order processing during pandemics for offline mode

Reference: https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/

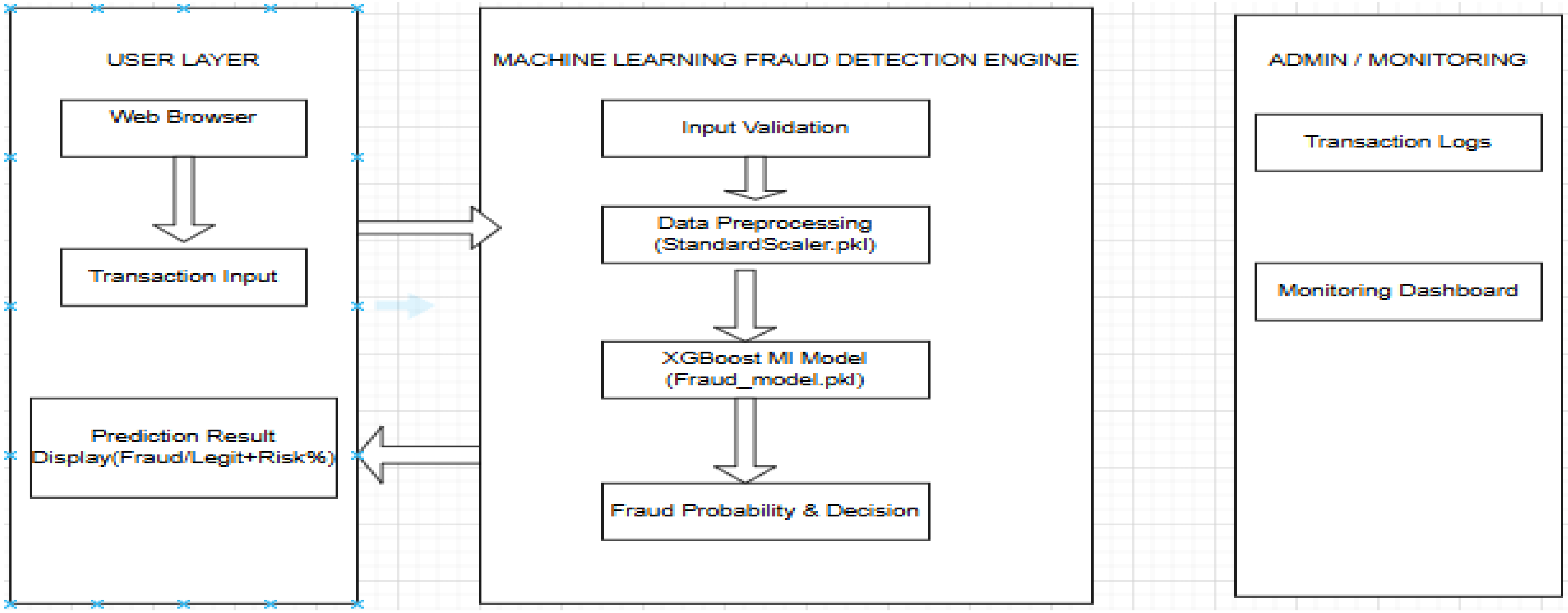
USER LAYER
Web Browser
Transaction Input
Prediction Result Display(Fraud/Legit+Risk%)
MACHINE LEARNING FRAUD DETECTION ENGINE
Input Validation
Data Preprocessing (StandardScaler.pkl)
XGBoost Ml Model (Fraud_model.pkl)
Fraud Probability & Decision
ADMIN / MONITORING
Transaction Logs
Monitoring Dashboard

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web interface used to input transaction details and display prediction results	HTML, CSS
2.	Application Logic-1	Handles routing, request processing, and response generation	Python,Flask
3.	Application Logic-2	Input validation and data formatting before model prediction	Flask Backend (Python)
4.	Application Logic-3	Fraud prediction processing using trained model	XGBoost (Scikit-learn)
5.	Database	Stores transaction logs and prediction results (if implemented)	MySQL / SQLite
6.	Cloud Database	Stores trained ML model and scaler files	Pickle (.pkl files)
7.	File Storage	Scales and transforms input data before prediction	StandardScaler (Scikit-learn)
8.	External API-1	Not applicable in current version (future integration with payment gateways possible)	REST API (Future Scope)
9.	Machine Learning Model	Performs fraud classification and probability prediction	XGBoost Classifier
10.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: http://127.0.0.1:5000	. Localhost (Flask Server) / Cloud (Future Deployment)

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Frameworks used for development of the application	Flask, Scikit-learn, XGBoost
2.	Security Implementations	Input validation, model protection, secure backend processing	Flask Validation, Python Backend Security
3.	Scalable Architecture	System designed to support future integration with real-time payment systems	Layered Architecture (Frontend + Backend + ML)
4.	Availability	Application runs continuously when Flask server is active	Flask Development Server
5.	Performance	Generates fraud prediction within few seconds	XGBoost Optimized Model

References:

https://c4model.com/

https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/

https://www.ibm.com/cloud/architecture

https://aws.amazon.com/architecture

https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d