Project Design Phase-II

Technology Stack (Architecture & Stack)

Date	13 February 2026
Team ID	LTVIP2026TMIDS88814
Project Name	Online Payments Fraud Detection using Machine Learning
Maximum Marks	4 Marks

Technical Architecture

Architecture includes Web UI / Payment Interface, Backend API layer, preprocessing pipeline, Machine Learning fraud detection model, transaction database, and optional integration with payment gateway APIs. Initial deployment can be Local with a cloud-ready scalable design.

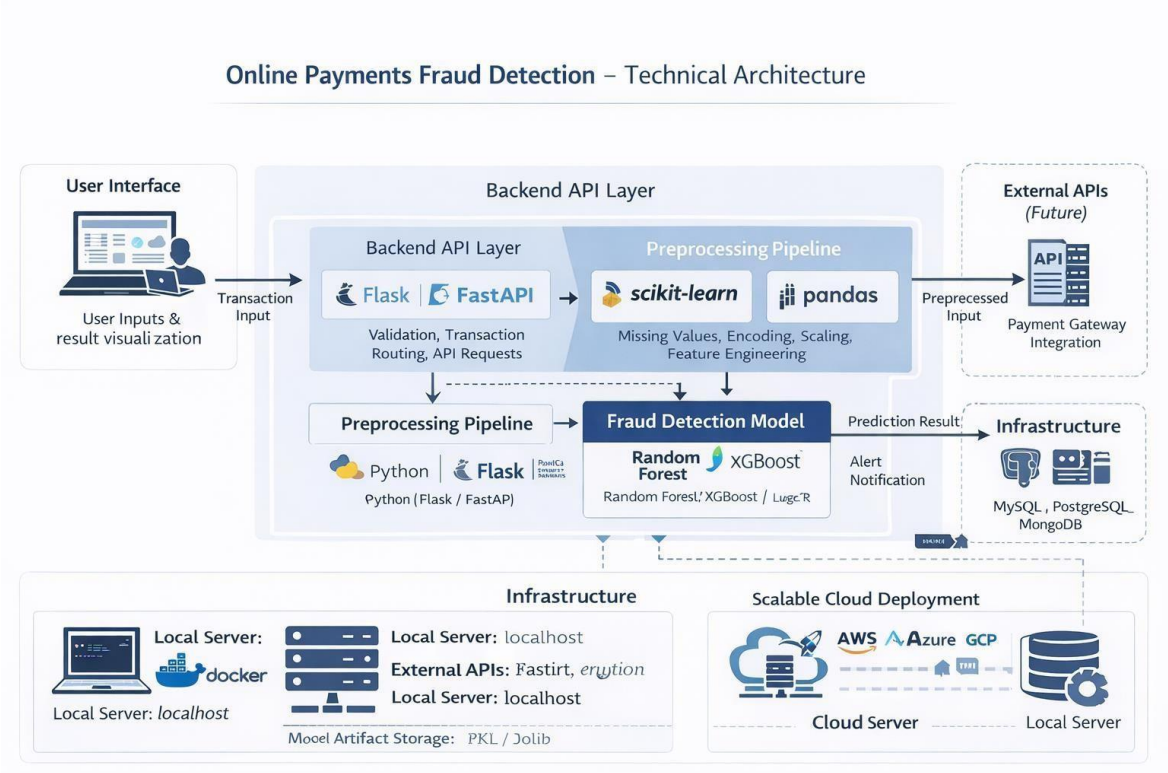

Table-1: Components & Technologies

S.No	Component	Description	Technology
1	User Interface	Transaction input, monitoring dashboard, fraud result display	HTML, CSS, JavaScript / React (Optional)
2	Application Logic	API endpoints, transaction routing,	Python (Flask / FastAPI)

		validation	
3	Preprocessing Pipeline	Missing value handling, encoding, scaling, feature engineering	scikit-learn, pandas, NumPy
4	Machine Learning Model	Fraud detection & probability scoring	Random Forest / XGBoost / Logistic Regression
5	Database / Storage	Transaction records, logs, model artifacts	MySQL / PostgreSQL / MongoDB
6	Model Artifact Storage	Store trained model files (PKL/Joblib)	Local File System
7	External API (Future)	Payment gateway integration	Stripe API / Razorpay API (Optional)
8	Infrastructure	Local deployment; scalable cloud deployment	Localhost, Docker (Future), AWS / Azure / GCP

Table-2: Application Characteristics

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	ML and backend built using open-source stack	Flask/FastAPI, scikit-learn, pandas
2	Security Implementations	Secure transaction handling, encryption, input validation	HTTPS, JWT/Auth, Data Encryption
3	Scalable Architecture	Modular layered design; cloud-ready	3-tier architecture, Docker/Kubernetes (Future)
4	Availability	Continuous fraud monitoring support	Cloud VM, Load Balancer (Future)
5	Performance	Real-time low-latency fraud detection	Optimized inference pipeline, caching (Future)