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

Date	19 February 2026
Team ID	LTVIP2026TMIDS74048
Project Name	Rising Waters: A Machine Learning Approach to Flood Prediction
Maximum Marks	4 Marks

Technical Architecture: Rising Waters

The system architecture for the Rising Waters: Flood Prediction project integrates multi-source data ingestion, temporal deep learning, external meteorological API integration, and a GIS-enabled web application.

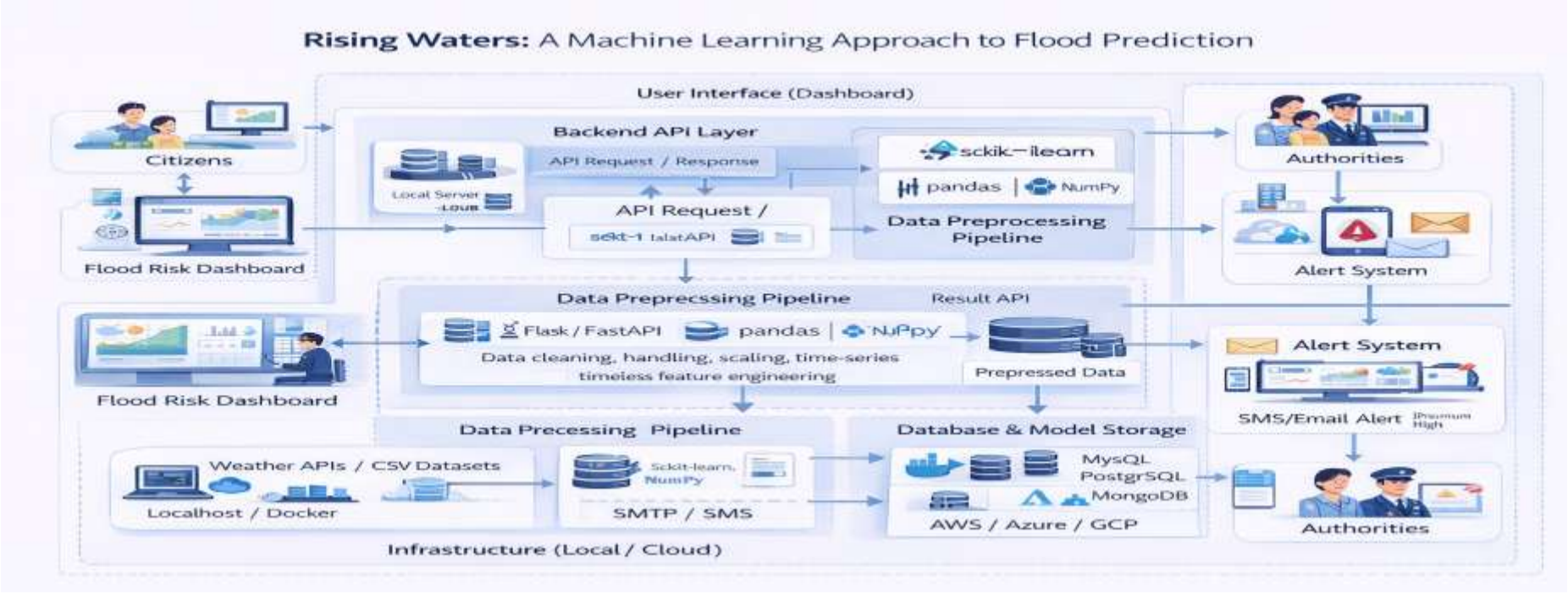

Table-1: Components & Technologies

S. No	Component	Description	Technology
1	User Interface	Flood risk dashboard, district-wise monitoring, alert visualization	HTML, CSS, JavaScript / React (Optional)
2	Application Logic	API endpoints, request handling, prediction routing	Python (Flask / FastAPI)
3	Data Collection Layer	Fetch rainfall, river level, and weather data	Weather APIs / CSV datasets
4	Preprocessing Pipeline	Missing value handling, scaling, time-series feature engineering	scikit-learn, pandas, NumPy
5	Machine Learning Model	Flood risk classification & probability scoring	Random Forest / XGBoost / Logistic Regression / LSTM
6	Database / Storage	Store environmental data, flood risk records, logs	MySQL / PostgreSQL / MongoDB
7	Model Artifact Storage	Store trained model files	PKL / Joblib
8	Alert System	SMS/Email notification for high-risk alerts	SMTP / SMS Gateway API
9	Infrastructure	Local deployment & scalable cloud deployment	Localhost, Docker (Future), AWS/Azure/GCP (Future)

Table-2: Application Characteristics

S. No	Characteristics	Description	Technology
1	Open-Source Frameworks	ML and backend built using open-source tools	Flask/FastAPI, scikit-learn, pandas
2	Security Implementations	Secure API endpoints, role-based access, data protection	HTTPS, JWT/Auth, Data Encryption
3	Scalable Architecture	Modular layered design supporting multiple districts	3-layer architecture, Docker (Future)
4	Availability	System runs continuously during monsoon season	Cloud VM, Load Balancer (Future)
5	Performance	Near real-time flood prediction	Optimized inference pipeline
6	Reliability	Consistent preprocessing using saved model artifacts	Saved scaler & trained model
7	Maintainability	Modular code structure for easy updates & retraining	Layered project structure