

:

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

Date: 22 July 2025

Team ID: LTVIP2025TMID45340

Project Name: Traffic Volume Estimation

Maximum Marks: 4 Marks

◆ Technical Architecture

Architecture Overview:

The system is designed as a 3-tier architecture (Presentation Layer, Application Layer, Data Layer) deployed on cloud infrastructure (Kubernetes / Cloud Foundry).

- **Users** access the system via Web UI or Mobile App.
- **Application layer** handles requests, predictions, and API interactions.
- **Data layer** manages traffic data and model storage in a cloud database.

Architecture Diagram:

(I can generate an infographic for this — let me know!)

◆ Table-1: Components & Technologies

S. No	Component	Description	Technology
-------	-----------	-------------	------------

1	User Interface	Web UI, Mobile App for user interaction	ReactJS, HTML, CSS
2	Application Logic-1	Traffic volume prediction logic	Python (Flask)
3	Application Logic-2	Optional integration of voice assistant	IBM Watson Assistant (if used)
4	Application Logic-3	Alert generation logic	Python
5	Database	Stores user data, traffic data	MySQL / NoSQL (MongoDB)
6	Cloud Database	Managed database service	IBM Cloudant / AWS DynamoDB
7	File Storage	Store model files, logs	IBM Cloud Object Storage / AWS S3
8	External API-1	Weather data for model	IBM Weather API / OpenWeatherMap
9	External API-2	Optional location verification	Google Maps API
10	Machine Learning Model	Predict traffic volume	Random Forest / XGBoost regression model
11	Infrastructure	Cloud deployment	IBM Cloud Kubernetes / AWS Elastic Kubernetes / Cloud Foundry

◆ **Table-2: Application Characteristics**

S. No	Characteristics	Description	Technology
-------	-----------------	-------------	------------

1	Open-Source Frameworks	ReactJS, Flask, Scikit-Learn	ReactJS, Flask, Scikit-Learn
2	Security Implementations	SHA-256 password hashing, HTTPS, IAM roles, OWASP guidelines	SHA-256, SSL/TLS, IAM
3	Scalable Architecture	Microservices, Kubernetes orchestration	Kubernetes, Docker
4	Availability	Load balancer, distributed servers	NGINX Load Balancer, Cloud Foundry
5	Performance	Caching, CDN for UI, model optimized for latency	Redis cache, CloudFront CDN