

# Project Design Phase

---

## Solution Architecture

Date: 27th June 2025

Team ID: LTVIP2025TMID42078

Project Name: TrafficTelligence: Advanced Traffic Volume Estimation with Machine Learning

Maximum Marks: 4 Marks

### Overview:

The solution architecture is structured as a modular and scalable machine learning system that accurately estimates and predicts traffic volume using real-time and historical data. It supports traffic authorities, urban planners, and navigation apps by delivering precise volume insights through a simple web-based interface. The architecture ensures robust data handling, prediction speed, and clear user communication, with support for offline demos and future cloud deployment.

## Architecture Components:

### 1. Frontend (User Interface Layer)

Technology Used: Streamlit / HTML / CSS

Role:

- Allows users to input routes, regions, and timeframes.
- Visualizes predicted traffic volume using charts or maps.
- Provides intuitive design for non-technical users.
- Shows real-time or forecasted traffic results clearly.

### 2. Backend (Application Logic Layer)

Technology Used: Python with Flask or FastAPI

Role:

- Receives frontend inputs and formats data for model inference.
- Manages prediction APIs and user session handling.
- Logs metadata like location, time, and prediction output.
- Handles error messages, performance logs, and input validation.

### 3. Model Layer (AI/ML Engine)

Technology Used: scikit-learn / XGBoost / LSTM / TensorFlow

Role:

- Predicts traffic volume based on multi-source input (weather, events, historical flow).
- Models trained using time-series forecasting or regression techniques.
- Optimized for speed and real-time inference accuracy.
- Optionally stored in-memory or accessed via API.

### Deployment & Usage:

- Offline Use: All required datasets and models can be stored locally for demos or test environments.
- Online/Cloud Option (Future Scope): Deploy using Docker, accessible via REST APIs.
- System Requirements: Python 3.x, GPU (optional), Anaconda or virtual environment.
- Optionally integrated with cloud services like AWS/GCP for scalability.

### Data Flow Summary:

[User Inputs Route/Time/Location]



[Frontend (Streamlit/Web Interface)]



[Backend (API + Preprocessing)]



[Trained ML Model (XGBoost/Regression)]



[Traffic Volume Prediction Output]



[Frontend Displays Results + Logging]