***PROJECT TITLE:***

***TrafficTelligence: Advanced Traffic Volume Estimation with Machine Learning***

**Team ID : LTVIP2025TMID35556**

**Team Members:**

Name 1: Adapa Venkata Sai Kumar

Name 2: A Nishitha

Name 3: A Gnana Manikanta

Name 4: A Tholakari Naga Sri

***TrafficTelligence: Advanced Traffic Volume Estimation with Machine Learning***

**Pre-Requisites:**

To successfully implement and understand the TrafficTelligence project, the following prerequisites are necessary:

Basic programming knowledge in Python

Familiarity with machine learning algorithms

Understanding of data analysis and statistics

Experience with libraries such as Pandas, NumPy, Scikit-learn, and Matplotlib

Basic knowledge of web development frameworks (e.g., Flask or Django)

**Prior Knowledge:**

Before starting this project, it's important to have prior knowledge in the following areas:

Supervised learning techniques (regression and classification)

Data collection methodologies

Data preprocessing (handling missing data, normalization, encoding)

Model evaluation metrics (MAE, RMSE, R² Score)

Use of APIs and integration for real-time data streaming

**Project Objectives:**

The main objectives of the TrafficTelligence project are:

Project Objective:

TrafficTelligence is an intelligent traffic volume estimation system powered by machine learning. It provides accurate, data-driven insights into traffic patterns by analyzing historical traffic data, real-time feeds, weather, and local events. The system is designed to support:

Dynamic traffic management

Urban development planning

Commuter guidance and navigation

To design a machine learning model capable of estimating and predicting traffic volume

To improve real-time traffic management using predictive insights

To support urban planning and infrastructure development with data-driven decisions

To provide commuters with route optimization and travel time forecasting

**Project Flow:**

1. Define problem and objectives

2. Collect and prepare data

3. Preprocess the data

4. Build and train machine learning models

5. Evaluate model performance

6. Develop and integrate the application

7. Deploy and test the system in a real-world scenario

**Project Structure:**

The project is organized into the following major modules:

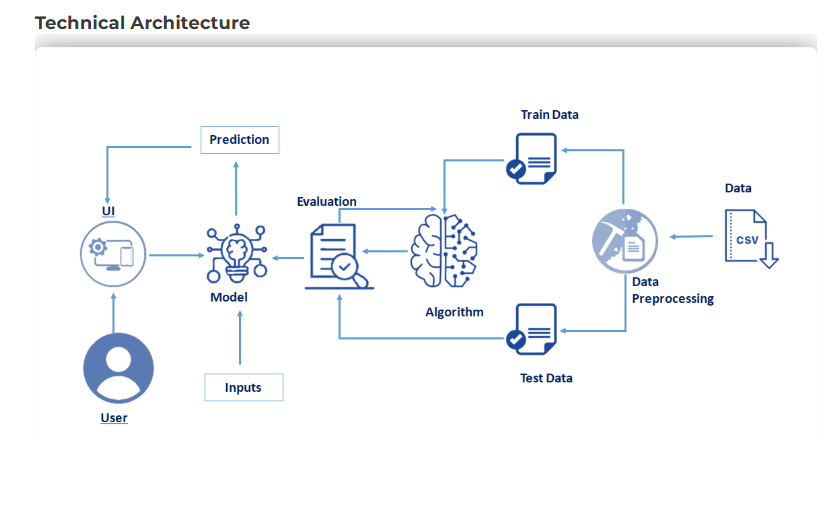
Data Collection

Data Pre-Processing

Model Building

Application Building

Technical Architecture:



**Data Collection:**

Data is collected from various sources including:

Historical traffic datasets

Weather information APIs

Event data (e.g., concerts, parades)

IoT sensors or camera feeds (if applicable)

Real-time traffic APIs (e.g., Google Maps Traffic API)

**Data Pre-Processing:**

Steps involved in preprocessing include:

Handling missing or inconsistent data

Normalizing traffic data for uniformity

Encoding categorical variables (e.g., weather type, day of the week)

Time-series formatting (if applicable)

Feature selection and dimensionality reduction

**Model Building**:

Machine learning models are trained using historical data to predict traffic volume:

Algorithms used: Linear Regression, Random Forest Regressor, Gradient Boosting

Training-test split and cross-validation for performance analysis

Metrics evaluated: MAE, RMSE, R² Score

Hyperparameter tuning for model optimization

**Application Building:**

The final system is built to be accessible and interactive:

Backend: Python with Flask/Django for processing and API integration

Frontend: HTML, CSS, JavaScript for a simple UI

Integration with real-time data sources

Visualization dashboards using Plotly or Dash

Output: Estimated traffic volume with time-based and location-based breakdown

**GITHUB LINK:** [**https://github.com/Manikantaag/traffic-inteligence**](https://github.com/Manikantaag/traffic-inteligence)