**Objective:**  
To develop a machine learning model that accurately predicts traffic volume using historical data and relevant features like weather, holiday, and date components. The system will assist in urban planning, traffic control, and congestion management.

**2. Functional Requirements**

| **ID** | **Requirement** | **Description** |
| --- | --- | --- |
| FR1 | Data Input Module | Allow users to input data (CSV upload or form entry). |
| FR2 | Preprocessing Module | Clean and transform data (handle missing values, encoding, scaling). |
| FR3 | Model Training Module | Train models like Linear Regression, Decision Tree, Random Forest, SVR, and XGBoost. |
| FR4 | Model Evaluation | Evaluate model performance using metrics like RMSE, MAE, R². |
| FR5 | Prediction Interface | Allow users to enter new data and receive predicted traffic volume. |
| FR6 | Visualization Dashboard | Show trends in traffic volume and model insights via graphs. |
| FR7 | Model Deployment | Deploy trained model through Flask app with pickle serialization. |

**3. Non-Functional Requirements**

| **ID** | **Requirement** | **Description** |
| --- | --- | --- |
| NFR1 | Usability | User interface should be simple and intuitive. |
| NFR2 | Performance | Model should respond within 2 seconds for prediction queries. |
| NFR3 | Scalability | Should be able to accommodate larger datasets in future. |
| NFR4 | Portability | System should run on local machines or be hosted online (if extended). |
| NFR5 | Maintainability | Code should be modular and commented for easy updates. |

**4. Data Requirements**

| **Feature** | **Description** |
| --- | --- |
| temp | Temperature data |
| rain | Rainfall amount |
| snow | Snowfall amount |
| holiday | Whether the day is a holiday |
| weather | Weather condition (e.g., Clear, Cloudy, Rainy) |
| day | Day of the month |
| month | Month |
| year | Year |
| traffic\_volume | Target variable to predict |

**5. System Requirements**

| **Component** | **Specification** |
| --- | --- |
| Programming Language | Python |
| Libraries | pandas, scikit-learn, xgboost, matplotlib, seaborn, Flask |
| Environment | Jupyter Notebook / VS Code / Anaconda |
| OS | Windows / Linux |
| Storage | Min. 1GB free space |
| RAM | Min. 4GB (8GB recommended) |

**6. Testing Requirements**

* Unit testing for preprocessing and prediction functions
* Integration testing between frontend and Flask model
* User Acceptance Testing (UAT) using test cases
* Model performance validation using train-test split and cross-validation

**7. Dependencies and Constraints**

* Dataset availability and quality
* User must enter values in correct format
* Flask model depends on pickle file being correctly saved and loaded
* Weather data must be correctly encoded for model to interpret