# Functional Requirement

ID Function Requirement  
INGEST-001 Data Ingestion The system shall support ingestion of traffic data from multiple sources, including traffic cameras, GPS devices, road sensors, and probe vehicles.  
CLEAN-001 Data Cleaning The system shall automate data cleaning tasks, including removing duplicate entries.  
CLEAN-002 Data Cleaning The system shall automate data cleaning tasks, including standardizing date and time formats.  
CLEAN-003 Data Cleaning The system shall automate data cleaning tasks, including flagging missing or invalid data fields.  
RECONCILE-001 Data Reconciliation The system shall reconcile conflicting data from different sources and assign a confidence score to each resolved event to reflect its reliability.  
MAPPING-001 Data Mapping The system shall automatically detect new data sources and suggest mappings to existing data schemas or formats.  
ANALYSIS-001 Traffic Analysis The system shall analyze data to identify traffic trends, including peak hours of traffic.  
ANALYSIS-002 Traffic Analysis The system shall analyze data to identify traffic trends, including recurring bottlenecks.  
ANALYSIS-003 Traffic Analysis The system shall analyze data to identify traffic trends, including unusual traffic spikes or anomalies.  
PREDICT-001 Predictive Insights The system shall provide predictive insights, such as anticipating slow-moving products.  
PREDICT-002 Predictive Insights The system shall provide predictive insights, such as forecasting potential traffic incidents.  
PREDICT-003 Predictive Insights The system shall provide predictive insights, such as suggesting proactive measures based on data patterns.  
FLAG-001 Issue Flagging The system shall flag high-impact issues and suggest actions, such as reducing stock levels for slow-moving products.  
FLAG-002 Issue Flagging The system shall flag high-impact issues and suggest actions, such as adjusting traffic signal timings.  
FLAG-003 Issue Flagging The system shall flag high-impact issues and suggest actions, such as scheduling maintenance.  
REPORT-001 Reporting The system shall generate visual reports with color coding to indicate traffic severity: red for severe congestion or incidents, yellow for moderate traffic issues, and green for normal conditions.  
FILTER-001 Data Filtering The system shall allow users to filter traffic data by time of day.  
FILTER-002 Data Filtering The system shall allow users to filter traffic data by day of the week.  
FILTER-003 Data Filtering The system shall allow users to filter traffic data by geographic zones (e.g., city districts, road segments).  
DASHBOARD-001 Dashboard The system shall provide a centralized dashboard to display real-time traffic data.  
DASHBOARD-002 Dashboard The system shall provide a centralized dashboard to display historical data from public safety, city planning, and weather services.  
DASHBOARD-003 Dashboard The system shall provide a centralized dashboard to display the status of data requests and alerts.  
REQUEST-001 Data Request Tracking The system shall support a formal data request tracking system, including a request form with fields for requestor, description, due date, and status.  
REQUEST-002 Data Request Tracking The system shall support a formal data request tracking system, including status updates and alerts for each request.  
SHARE-001 Data Sharing The system shall facilitate data sharing between departments while ensuring compliance with data governance and security policies.  
VISUALIZATION-001 Data Visualization The system shall enable users to access and visualize real-time data from other departments (e.g., public safety, city planning, and weather services).

# External Description

# 5 Constraints  
  
## 5.1 Regulatory and Legal Constraints  
  
The system shall comply with all applicable local, national, and international regulations regarding data privacy, data protection, and urban traffic management. These include but are not limited to the General Data Protection Regulation (GDPR) where applicable, as well as city-specific data governance policies.  
  
- \*\*Priority\*\*: Must Have   
- \*\*Rationale\*\*: Legal compliance is essential to ensure the system can operate without risk of fines, data misuse, or public backlash.   
- \*\*Source\*\*: SRL-2.4   
- \*\*Acceptance Criteria\*\*: The system shall pass a regulatory compliance audit conducted by an independent third party.  
  
## 5.2 Hardware Constraints  
  
The system shall operate on high-performance servers and storage systems that are capable of handling large volumes of structured and unstructured data. Redundant backup infrastructure is mandatory to ensure data availability and integrity.  
  
- \*\*Priority\*\*: Must Have   
- \*\*Rationale\*\*: The system must process and store real-time and historical data efficiently and reliably.   
- \*\*Source\*\*: SRL-2.5   
- \*\*Acceptance Criteria\*\*: The system shall run on a server with at least 64 GB RAM and 1 TB SSD storage, and the backup infrastructure shall support 99.99% uptime.  
  
## 5.3 Interface Constraints  
  
The system shall interface with external systems and data sources through predefined APIs, including traveler information services, weather APIs, and public safety databases. These interfaces must be secure, scalable, and maintain backward compatibility.  
  
- \*\*Priority\*\*: Should Have   
- \*\*Rationale\*\*: Consistent and secure interfaces ensure seamless data exchange and reduce integration effort.   
- \*\*Source\*\*: SRL-2.6   
- \*\*Acceptance Criteria\*\*: The system shall support integration with at least three external APIs (e.g., OpenWeatherMap, Google Maps, and 911 incident logs) using RESTful or GraphQL protocols with TLS 1.3 encryption.  
  
## 5.4 Design and Implementation Constraints  
  
The system shall be implemented using a modular, microservices-based architecture. It shall be containerized using Docker and orchestrated with Kubernetes to support scalability and deployment flexibility.  
  
- \*\*Priority\*\*: Must Have   
- \*\*Rationale\*\*: A modular and containerized design supports easier maintenance, updates, and deployment in a dynamic environment.   
- \*\*Source\*\*: SRL-2.5   
- \*\*Acceptance Criteria\*\*: The system shall be deployed using Docker containers and managed via Kubernetes, with each service running in its own container and communicating via RESTful APIs.  
  
The system shall be compatible with Linux-based operating systems, specifically Ubuntu and CentOS, and shall not require proprietary or platform-specific software components for core functionality.  
  
- \*\*Priority\*\*: Must Have   
- \*\*Rationale\*\*: Linux compatibility ensures flexibility, cost-effectiveness, and ease of deployment across city infrastructure.   
- \*\*Source\*\*: SRL-2.5   
- \*\*Acceptance Criteria\*\*: The system shall run on both Ubuntu and CentOS without modification, with full functionality in each environment.  
  
The system shall not be dependent on a single vendor’s technology stack for core data processing and visualization capabilities.  
  
- \*\*Priority\*\*: Should Have   
- \*\*Rationale\*\*: Vendor neutrality ensures long-term sustainability, reduces lock-in risks, and supports open standards.   
- \*\*Source\*\*: SRL-2.5   
- \*\*Acceptance Criteria\*\*: The system shall be able to replace any core data processing or visualization component with a third-party equivalent without re-engineering the entire system.  
  
## 5.5 Other Constraints  
  
The system shall not require manual intervention for schema validation, data reconciliation, or baseline comparison during normal operation.  
  
- \*\*Priority\*\*: Must Have   
- \*\*Rationale\*\*: Automation is a key requirement for the system to reduce manual workload and ensure consistent and timely processing.   
- \*\*Source\*\*: SRL-3.4, SRL-4.6   
- \*\*Acceptance Criteria\*\*: The system shall perform schema validation, data reconciliation, and baseline comparison without user input, and log any exceptions for review.  
  
The system shall not store raw traffic data in an unencrypted format at rest.  
  
- \*\*Priority\*\*: Must Have   
- \*\*Rationale\*\*: Data security is critical for maintaining public trust and meeting compliance standards.   
- \*\*Source\*\*: SRL-2.7   
- \*\*Acceptance Criteria\*\*: All raw data shall be encrypted using AES-256 or higher before storage, and decryption shall occur only in secure, authenticated environments.  
  
The system shall not allow unauthorized users to access or modify data or system configurations.  
  
- \*\*Priority\*\*: Must Have   
- \*\*Rationale\*\*: Unauthorized access or modification could compromise data integrity and system reliability.   
- \*\*Source\*\*: SRL-2.7   
- \*\*Acceptance Criteria\*\*: The system shall enforce role-based access control (RBAC) and prevent unauthorized access through multi-factor authentication and audit logging.  
  
The system shall not require more than 10 minutes of downtime for routine updates or maintenance.  
  
- \*\*Priority\*\*: Should Have   
- \*\*Rationale\*\*: Minimal downtime ensures continuous availability of real-time and historical data for decision-making.   
- \*\*Source\*\*: Stakeholder Input (IT Administrator)   
- \*\*Acceptance Criteria\*\*: The system shall support rolling updates and hot-swapping of modules, with no more than 10 minutes of downtime per update cycle.