Functional Requirements

Created	@November 12, 2025 9:51 AM
Tags	

A. FUNCTIONAL REQUIREMENTS

This section outlines the specific functional requirements for each defined Use Case. These requirements describe *what the system must do*.

Use Case 1: Log in to the System

1. **Scope of the Product:** This functionality serves as the entry gateway to the system, ensuring that only authenticated users can access protected features.
2. **Functional and Data Requirements:**
    - **Functional Requirements:**
        - The system **shall** provide a login interface for the user to input credentials.
        - The system **shall** validate the user's credentials (email and password) against the records stored in the database.
        - The system **shall** use a secure hashing mechanism to store and compare passwords.
        - The system **shall** generate and return a JSON Web Token (JWT) to the client upon successful authentication.
        - The system **shall** deny access and display an appropriate error message upon failed authentication.
    - **Data Requirements:**
        - The user's login credentials (email, hashed password) **must** exist in the database.
        - The input email **must** adhere to a valid format.

Use Case 2: View Real-time Monitoring Dashboard

1. **Scope of the Product:** This is the primary workspace for operators, providing an intuitive and immediate overview of the traffic situation.
2. **Functional and Data Requirements:**
    - **Functional Requirements:**
        - The system **shall** establish a persistent WebSocket connection between the frontend and the backend when the user accesses the dashboard.
        - The system **shall** push processed video frames (with rendered bounding boxes) from the backend to the frontend via the WebSocket connection.
        - The system **shall** push new violation events from the backend to the frontend via the WebSocket connection as soon as they are detected.
        - The user interface **shall** render the video stream smoothly.
        - The user interface **shall** display a list of the latest violation alerts in a dedicated section.
    - **Data Requirements:**
        - Video frame data and violation alert data **must** be formatted in a predefined JSON structure before being pushed over the WebSocket.

Use Case 3: Manage System Configuration

1. **Scope of the Product:** This functionality provides the core flexibility of the system, allowing users to customize business logic without modifying the source code.
2. **Functional and Data Requirements:**
    - **Functional Requirements (for Rule Management):**
        - The system **shall** provide a text editor interface for users to create and edit DSL rules.

    - The system **shall** parse and validate the syntax of the DSL string submitted by the user.
    - The system **shall** save the rule to the database if the syntax is valid.
    - The system **shall** return a detailed error message to the user if the syntax is invalid.
    - The system **shall** allow users to delete existing rules.
- **Functional Requirements (for Zone Management):**
    - The system **shall** allow users to select a specific camera for configuration.
    - The system **shall** provide an interface that allows users to draw a polygon by clicking points on the video frame.
    - The system **shall** allow users to assign a unique name to the newly drawn zone.
    - The system **shall** store the zone's name and its normalized coordinates in the database, associated with the corresponding camera.
- **Data Requirements:**
    - A DSL rule **must** conform to the defined grammar.
    - A zone polygon **must** consist of at least three vertices.

Use Case 4: View and Filter Violation Reports

1. **Scope of the Product:** This feature supports post-event analysis, statistics generation, and evidence retrieval.
2. **Functional and Data Requirements:**
    - **Functional Requirements:**
        - The system **shall** display a list of recorded violations, sorted chronologically (most recent first).
        - The system **shall** provide filtering controls for users to search violations by a date range, the name of the rule that was triggered, and

the camera that recorded it.

    - The system **shall** allow users to view detailed information for a single violation, including an image or video clip as evidence.

- **Data Requirements:**
    - Each violation record in the database **must** contain comprehensive information: the rule ID, camera ID, timestamp, and a reference to the evidence file.

Use Case 5: Manage System Infrastructure

1. **Scope of the Product:** These are high-level administrative functions, restricted to Admin roles, for managing the foundational components of the system.
2. **Functional and Data Requirements:**
    - **Functional Requirements (for Camera Management):**
        - The system **shall** allow an Admin to add a new camera source by providing a name and a video stream URL.
        - The system **shall** allow an Admin to update a camera's information or temporarily deactivate it.
        - The system **shall** allow an Admin to remove a camera from the system.
    - **Functional Requirements (for User Management):**
        - The system **shall** allow an Admin to create a new user account with an email, password, and a specified role (User/Admin).
        - The system **shall** allow an Admin to edit the information or change the role of an existing user.
    - **Data Requirements:**
        - A camera's source URL **must** be in a valid format that the system can process (a video file path for the demo, an RTSP URL for production).
        - A new user's email **must** be unique within the system.