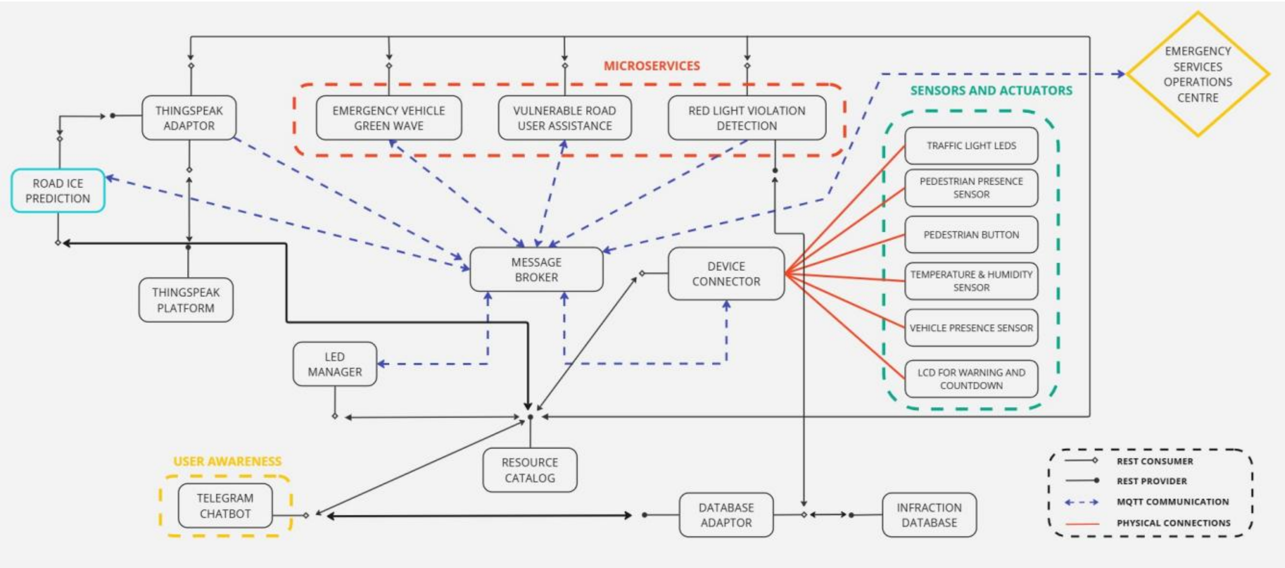
1. Name of Use Case

Name of the Use Case	**IoT platform for Smart Traffic Lights**
Version No.	3.0
Submission Date	01/03/2025
Team Members (with student ids)	Mattia Antonini (s344064), Livio Dadone (s347038), Giovanni Grieco (s346012), Simone Peradotto (s343420)

2. Scope and Objectives of Function

Scope and Objectives of Use Case	
Scope	The platform provides advanced IoT-based traffic light control at intersections.
Objective(s)	- Optimize traffic flow. - Reduce waiting times for pedestrians and cyclists. - Enhance safety for vulnerable users. - Enhance road user safety. - Improve the efficiency and oversight of traffic light systems.
Domain(s)	- Smart control of traffic lights. - Smart traffic management - Sustainable urban mobility. - Road safety improvements.
Stakeholder(s)	Public Administration, Infrastructure Management Companies, Road users
Short description	The Smart traffic light IoT platform's main goal is to optimize urban traffic management. It integrates IoT devices, such as sensors and actuators, to monitor and manage intersections dynamically. The platform provides control strategies to prioritize vulnerable users, such as pedestrians with mobility impairments, and to enable green waves for emergency vehicles, minimizing response times. The system ensures efficient communication via unified interfaces, employing both REST and MQTT protocols. It incorporates environmental monitoring to predict hazardous road conditions, enhancing road safety. Additionally, the platform enables specialized users to access traffic violation data through a Telegram Chatbot interface, supporting law enforcement activities. Overall, this IoT-based solution promotes sustainable urban mobility, reducing congestion and improving the safety and efficiency of urban intersections.

3. Diagram of Use Case



The Smart Traffic Light system is composed of different IoT components that allow it to optimize traffic management, prioritize vulnerable users, and improve urban mobility. The architecture is based on a modular and scalable approach, that uses MQTT and RESTful web services to let the various components communicate. The system provides dynamic traffic light control, real-time detection of different situations, and monitoring of road and weather conditions.

4. Core Components:

4.1. Message Broker:

The fundamental part that enables communication via MQTT between device connectors, services and microservices. It comes into play when it is necessary to transfer information in real time.

4.2. LED Manager:

A service that receives information from other microservices that are responsible for influencing the duration of the traffic light, establishes a priority scale and finally communicates with the device connector directly connected to the traffic light LEDs to turn them on and off based on the information received. The device connector, which acts as the subscriber to specific MQTT topics, receives on/off commands from the message broker.

4.3. Device connector:

The device connector is the device that in our project will be the Raspberry Pi. The various sensors and actuators are connected to it via GPIO ports and the data exchanged with them is shared with the other parts of the system. The data collected by the sensors is stored or processed to make decisions. The actuators instead (LED lights and LCD screen) are activated based on the information that the device connector receives via MQTT from the various services. Each connected sensor or actuator refers to a specific service or microservice:

- *Pedestrian presence sensor* --> Vulnerable road users assistance

- *Pedestrian button* --> Vulnerable road users assistance
- *Temperature and Humidity sensors* --> Environmental Monitoring
- *Vehicle presence sensor* --> Red light violation detection
- *LCD for warning and countdowns* --> LCD display
- *Traffic light LEDs* --> LED Manager

4.4. Microservices:

5. Emergency Vehicle Green Wave:

This microservice aims to decrease the transit time of emergency vehicles in the city during their activity. When a vehicle leaves, the operations center (Emergency services operations center) estimates the route it will take and a transit time for each intersection. The LED manager of each traffic light, after it receives an MQTT request from the emergency vehicle green wave, will set the green light for the transit direction of the emergency vehicle for a time interval (30s) before and after the expected transit time. This time can be dynamically adjusted by the center based on the data received from the vehicle (we do not intend to fully implement this part since it is not part of what is covered in the course).

6. Vulnerable road users assistance:

This microservice is designed to allow people with walking difficulties or who in any case need more time to cross, to have it. Presence sensors facing the sidewalk are used to detect if there is someone waiting to cross, while a button allows to request extra time at the next green light to enable vulnerable users to cross safely. These sensors are connected to the device connector which communicates via MQTT with the vulnerable road users assistance.

7. Red light violation detection:

This microservice is responsible for penalizing users who commit traffic light violations, that is, those who cross the stop line when the red light is on. The system retrieves data from the LED manager by MQTT to understand the status of the traffic light (if the red light is on) and the data that the device manager receives from a presence sensor positioned after the stop line. When the red light is on and a vehicle is detected beyond the stop line, a camera photographs the license plate and saves the data relating to the violation in an SQL database with a REST request to the database adaptor.

7.1. Thingspeak:

Thingspeak is an IoT platform designed for basic data storage and processing. Through the adapter, it is possible to insert information into the platform via MQTT, even though it is designed to work via REST requests. The adapter receives messages mainly from the "Environmental monitoring" microservice. The platform can be consulted through the adapter from the Telegram Chatbot, to obtain statistics and analysis of the data collected by the various control units present on the traffic lights. We decided to use such storage for timestamp data such as weather data.

8. Road ice predictions:

This service performs a linear regression on historical data to create a model, it communicates with thingspeak adaptor with REST. This will then be applied to the data acquired in real time by the weather station (which sends them via MQTT to the service) to generate possible warnings of ice on the road to be shown on the LCD display

9. **Infraction Database**:
It is a SQL database that contains a list of violations committed by road users. A SQL database is the best choice for this type of data, for which a table is the best and most efficient structure. Each violation is associated with a unique ID, the timestamp at which it occurred, the license plate number of the vehicle involved and the photo taken by the camera, in which the license plate was framed. The database can be consulted by authorized users following a login via the telegram bot. It will be possible to retrieve all the violations associated with a license plate number simply by sending it by message to the bot, receiving a list and the photo of the license plate. The bot communicates with the database adaptor via REST requests and infraction data are retrieved from the database by means of SQL queries.

9.1. Resource Catalog:
The resource catalog serves as a service registry system for both devices and all elements of the system. It contains and provides information about the endpoints (REST Web Services and MQTT topics) of all devices, resources and services in the platform. It contains configuration settings for applications and control strategies. Each actor, during startup, will retrieve this information from the resource Catalog using its own REST Web Services.

10. **Telegram Chatbot:**
The Telegram chatbot serves as a user interface. Through the chat, each user (no authentication required) can consult the environmental data collected by the control units on the traffic lights. For specialized users, such as police forces, who authenticate themselves in the Bot, it is possible to have access to the Infraction Database and consult the violations linked to each license plate number that is entered. The bot

communicates with the Infraction Database adaptor via REST requests and with the Thingspeak adaptor via REST requests

10.1. Communication Protocols:

- **MQTT:** Used for real-time communication between sensors, the central system, and actuators like traffic lights.
- **REST:** Used for querying data from the central system, specifically through the Telegram bot interface.
- **SQL:** Used for structured storage and querying of historical data within the database, enabling advanced data analysis and reporting.

Scalability and adaptability are at the base of the design of this project. It is designed to be easily deployed in any urban context and to be able to connect different devices to the same network without difficulty.

11. Desired Hardware components (only among those we can provide)

Device Name	Quantity	Needed for…