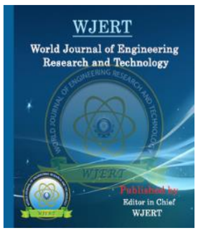
WJERT, 2025, Vol. 11 Issue 12, 277-285 **Original Article** **ISSN 2454-695X**

World Journal of Engineering Research and Technology

www.wjert.org

Impact Factor: 7.029 **Coden USA: WJERA4**



REAL-TIME IOT DASHBOARD FOR SMART LUMINAIRE MANAGEMENT

J. L. Sanchez-Cuevas[1]*, L. J. Mona-Peña[1], M. P. Piña-Villanueva[1], E. Fernandez-Chavez[1], A. A. Dominguez-Martinez[1], A. Flores-Valdes[1], M. Arevalo-Carbajal[1], G. J. Pinal-Ramirez[1]

[1]Tecnologico Nacional de Mexico/Instituto Tecnologico de Saltillo, Departamento de Sistemas Computacionales, Av. Venustiano Carranza 2400, Col. Tecnológico, C.P 25280, Saltillo, Coahuila, México.

Article Received on 21/10/2025 **Article Revised on 11/11/2025** **Article Published on 01/12/2025**

***Corresponding Author**

J. L. Sanchez-Cuevas

Tecnologico Nacional de Mexico/Instituto Tecnologico de Saltillo, Departamento de Sistemas Computacionales, Av. Venustiano Carranza 2400, Col. Tecnológico, C.P 25280, Saltillo, Coahuila, México.

https://doi.org/10.5281/zenodo.17961761

How to cite this Article: J. L. Sanchez-Cuevas1*, L. J. Mona-Peña1, M. P. Piña-Villanueva1, E. Fernandez-Chavez1, A. A. Dominguez-Martinez1, A. Flores-Valdes1, M. Arevalo-Carbajal1, G. J. Pinal-Ramirez1. (2025). REAL-TIME IOT DASHBOARD FOR SMART LUMINAIRE MANAGEMENT. World Journal of Engineering Research and Technology, 11(12), 277–285.

ABSTRACT

Intelligent lighting systems powered by Internet of Things (IoT) technologies offer scalable and energy-efficient solutions for public infrastructure. This study presents the deployment of a smart lighting system at the Tecnológico Nacional de México Campus Saltillo, integrating UbiCell UGU controllers via seven-pin NEMA receptacles and LTE-M (CAT-M1) connectivity. Real-time diagnostics are transmitted to the Ubicquia UbiVu platform through secure APIs, enabling alerts for power loss, disconnection, and driver failure. The system includes a cloud-based dashboard built with Node.js, Express, Nginx, and Bootstrap, connected to a Docker-deployed SQL Server database. Users can manage sensors, roles, and API credentials, visualize luminaires on a geospatial map using Leaflet.js, and access live energy metrics through Ubicquia's public API. Remote control of luminaires is supported via authenticated POST requests. Validation in a real-world campus environment demonstrated improved energy monitoring, reduced manual inspections, and strengthened operational reliability. The system establishes a technical foundation for broader urban applications in smart infrastructure management.

KEYWORDS: IoT, LTE-M, smart lighting, energy monitoring, Ubicquia, Node.js, API integration, SQL Server, Docker, Leaflet.js, remote diagnostics.

1. INTRODUCTION

Urban growth has driven the adoption of smart technologies like Smart Public Lighting (SPL) systems to optimize energy consumption. Recent studies demonstrate their effectiveness: Sánchez Sutil and Cano-Orteg achieved 35% energy savings in Spain, while Manyake and Mathaba confirmed reductions of up to 50% in South Africa. However, limitations persist, such as the lack of real-time interactive visualization or the inability to detect vandalism and technical failures In Latin America, projects like NEC's in Chile showed progress but failed to incorporate predictive analytics.

This research proposes a comprehensive solution: a georeferenced dashboard with dynamic energy consumption charts and an early-warning system for tilt anomalies, electrical faults, and wiring theft. Unlike previous studies, it combines proactive management and real-time decision-making, critical for complex urban environments.

At Tecnológico Nacional de México, Campus Saltillo, the absence of automated monitoring leads to reactive maintenance, high energy costs, and a lack of air quality data. Our hypothesis states that an LTE-M-based IoT system will improve fault detection by 50% while providing real-time environmental metrics. Objectives include designing a remote-control platform, integrating energy-monitoring sensors, and validating the system on campus to promote efficiency and sustainability.

2. MATERIALS AND METHODS

2.1 Materials

To develop the intelligent public lighting system (API), IoT components based on LTE-M (CAT-M1) technology were used, including:

- UbiCell UGU Controller: An industrial-grade device that monitors real-time parameters such as tilt, vibration, power quality, and electrical consumption in luminaires. Compatible with DALI, 0–10V, and D4i protocols, offering 0.5% measurement accuracy and overvoltage protection (20 kV/10 kA).

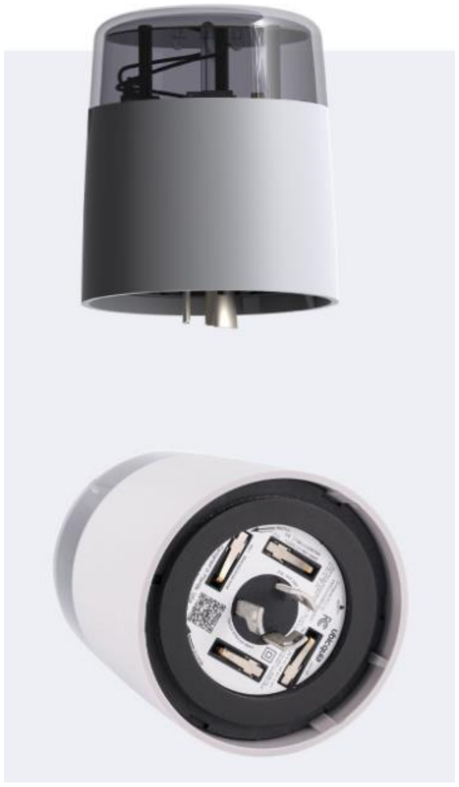


Figure 1: UbiCell UGU Controller.

- **NEMA Connectors:** Standardized interfaces for public lighting, available in 3-, 5-, or 7-pin configurations, enabling dimming control and RF Mesh-based communication (Gong, 2025).

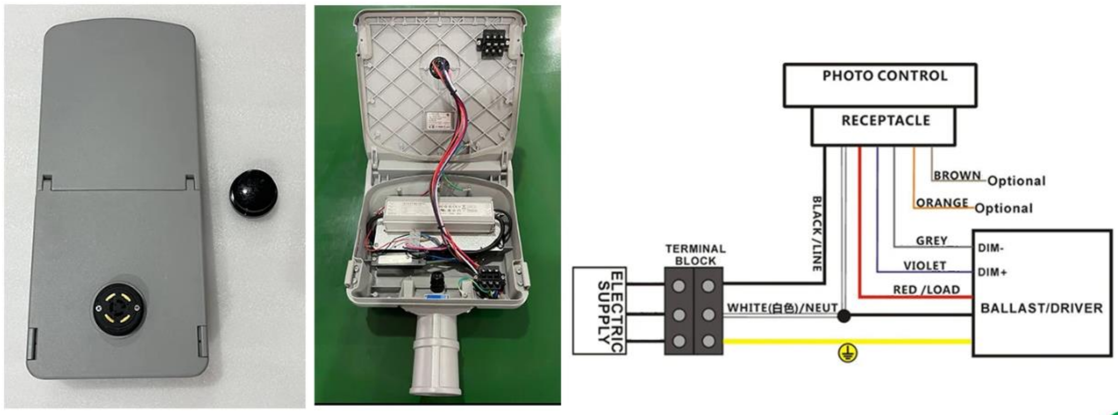


Figure 2: Nema ANSI.

- **UbiVu Platform:** A cloud-based asset management system offering interactive dashboards, predictive analytics, and proactive alerts for technical faults. Certified under ISO/IEC 27001 cybersecurity standards.

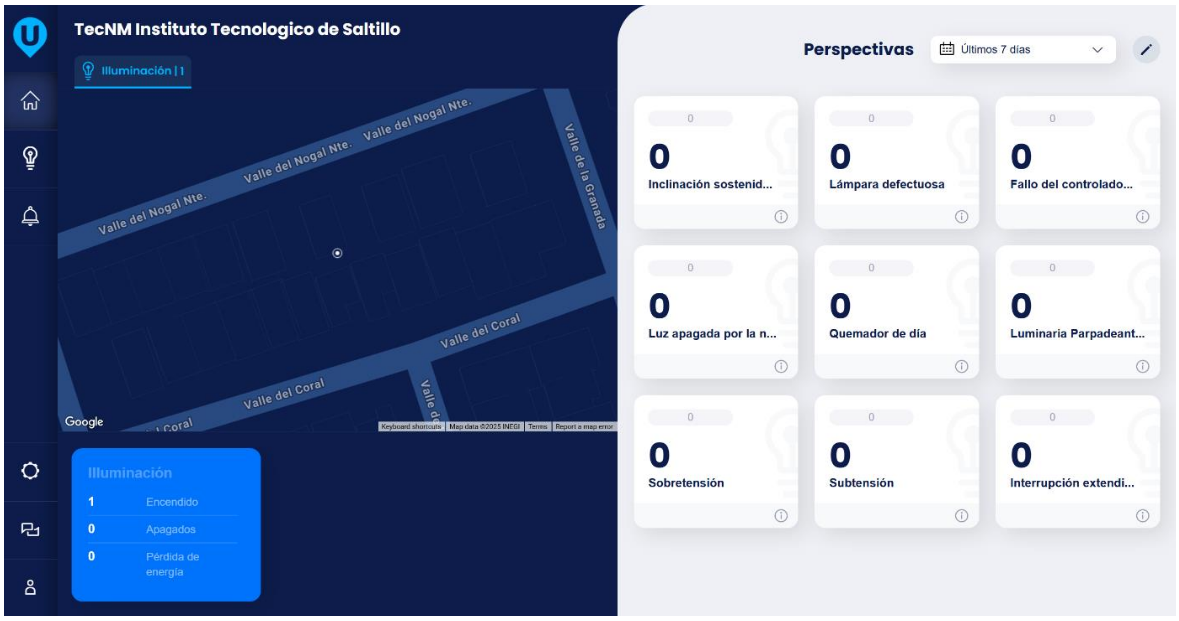


Figure 3: UbiVu Dashboard.

2.1.1 Tecnología LTE-M (CAT-M1)

This LPWAN standard operates within licensed LTE bands, optimizing energy consumption and signal coverage (Chaudhari & Zennaro, 2020). Key features include:

- Low latency (15–30 ms) with strong indoor signal penetration.
- Extended battery life (up to 10 years using AA batteries).
- Scalability: Supports over 100,000 devices per base station.
- Compatibility with 5G-IoT and NB-IoT networks.

Parameter	**LTE-M**	**Standard LTE**
Data Rate (Mbps)	0.375–1	10–326.5
Energy Consumption	Ultra-low	Moderate
Indoor Coverage	High	Medium

2.1.2 Software Infrastructure

- **Backend**: Built using Node.js and Express.js for RESTful APIs, with PM2 process manager for production-grade deployment.
- **Frontend**: Dashboard developed with Bootstrap, enabling real-time data visualization (georeferenced maps, dynamic charts).
- **Database**: Microsoft SQL Server, managed using SQL Server Enterprise Manager.
- **Web Server**: NGINX configured as reverse proxy and load balancer.
- **Security**: HTTPS/TLS protocol with AES-256 encryption and SSL certificates.

2.1.3 Virtualization

- **Oracle VirtualBox**: Used to deploy virtual machines on Ubuntu Server environments, optimized using Vagrant for automated provisioning.
- **Operating System**: Ubuntu Server 22.04 LTS, supporting Docker, Kubernetes, and built-in monitoring tools.

3. RESULTS AND DISCUSSION

3.1 RESULTS

Figure 4 illustrates the system's integration, showcasing a streamlined network architecture composed of a minimal set of components for simplified implementation and operation. The system includes:

- Lighting Controller: Responsible for collecting and transmitting operational data from each luminaire—such as on/off/dimmed status, power consumption, GPS location, and usage hours. It also detects anomalous events such as tilt due to impact, unscheduled switching, power loss, wiring faults, voltage drops, driver malfunctions, and potential cable theft.
- UbiVu Control Panel: A dashboard-style interface that receives data from the controller and supports remote node control via API commands (Ubicquia, n.d.)
- APIs: Serve as integration bridges between the controller and UbiVu panel, enabling real-time transmission of luminaires' status, energy usage, alerts, and location data. They also facilitate remote command execution from the dashboard to the controller for centralized lighting system management (Red Hat, 2023).

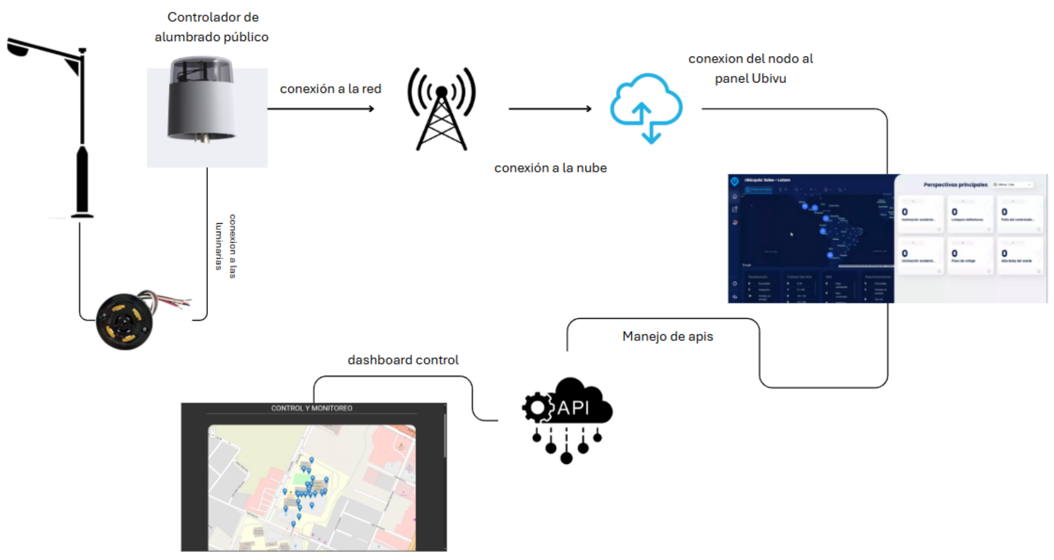

Controlador de alumbrado público
conexión a la red
conexión a la nube
conexion del nodo al panel Ubivu
conexion a las luminarias
Manejo de apis
dashboard control
CONTROL Y MONITOREO
API

Figure 4: System architecture.

After deploying the intelligent lighting system, a network of controllers, dashboards, and communication mechanisms was configured to support remote operation and real-time data analysis.

To simulate and deploy the backend, a virtual machine was configured in VirtualBox with the following specifications:

- Ubuntu Server OS with XFCE graphical environment (installed via xubuntu-desktop) and LightDM session manager.

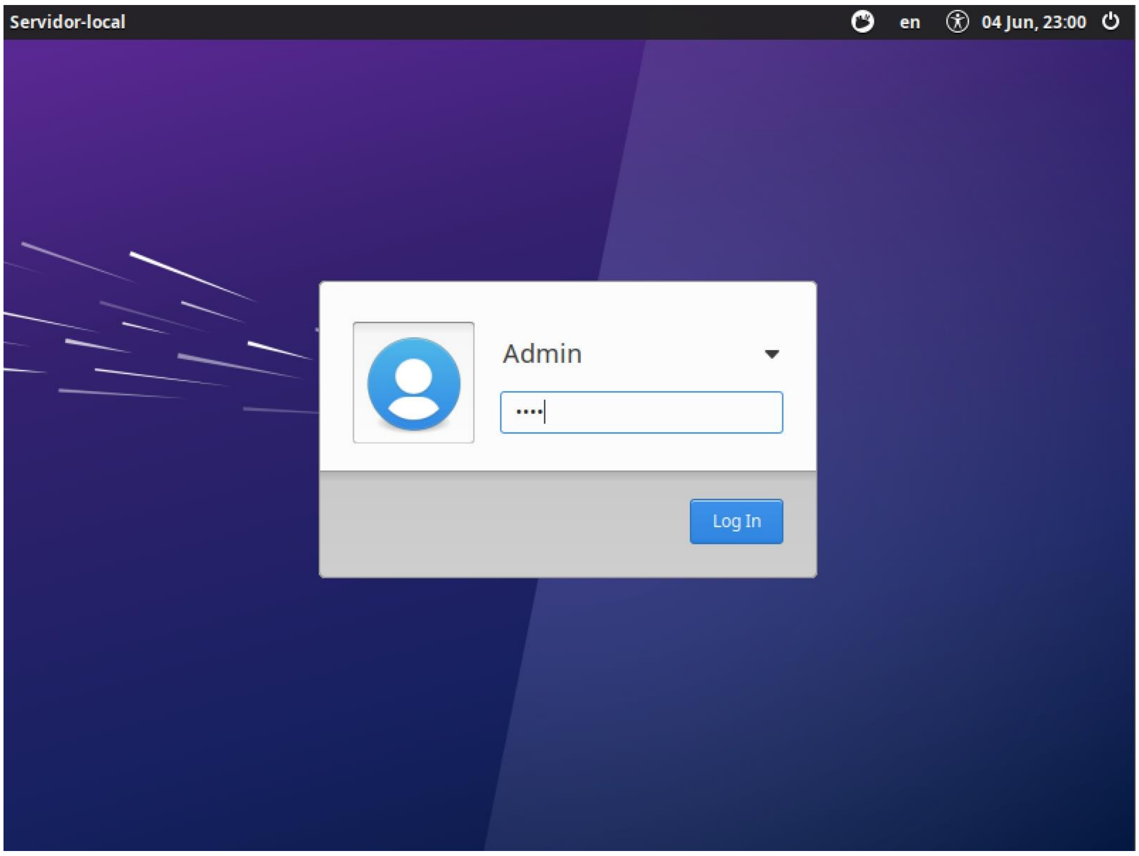


Figure 5: Successful Ubuntu Server execution.

- Dual network interfaces: NAT for internet access and Bridged for local communication with the campus network.

Essential backend tools installed and configured

- **NGINX**: Set as a reverse proxy for Node.js running on port 3000; UFW firewall rules enabled necessary traffic.
- **Node.js & npm**: Used to initialize the backend project and integrate the Express framework.

- **Netplan**: Configured to assign a static IP to the enp0s8 interface for reliable local connectivity.
- **PM2**: Maintains the backend application as a persistent service.
- **Docker & SQL Server 2022**: SQL Server deployed in a container; additional tools (mssql-tools and unixodbc-dev) installed to manage database operations and backend integration.

The system uses a relational database model with core tables including Usuario, Luminarias, Registro_consumo_fechas, Rol, and CONFIG_NODE, all defined with foreign keys to preserve referential integrity. Communication with the database is handled via SQL client queries through dedicated configuration files and API endpoints.

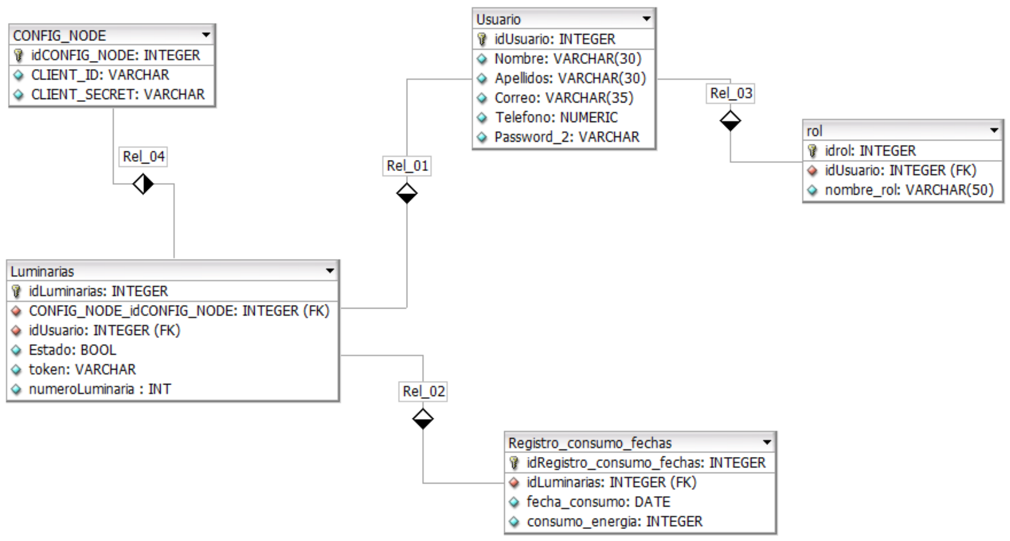


Figure 6: Relational model of the database.

Authentication and User Management

- Login validation using fetch, with local storage of user ID and role.
- Role-based permissions where only administrators can perform create, update, and delete operations.
- Dynamic CRUD interface for managing users via linked API endpoints.
  Credential management is handled through modal forms built with Bootstrap, connected to dedicated endpoints that allow credential validation, updates, and retrieval.

For sensor assignment and management

- Token-to-luminaire linking is implemented with duplicate validation mechanisms

- Token removal is logical—tokens are disabled without deleting their associated records
- Query endpoints are available to verify tokens linked to each luminaire

Geospatial visualization and control are supported via an interface built with Leaflet.js and OpenStreetMap, where each luminaire is represented by an interactive marker displaying live data in a side panel. The backend retrieves stored data via specific endpoints, and also fetches real-time metrics (e.g., consumption, voltage, status, alerts) directly from the node using the Ubicquia API, authenticated through credentials client_id, client_secret, and NODE_ID.

Finally, the system supports remote control of luminaires (on/off) through authenticated POST requests.

This set of results demonstrates the successful integration and operational performance of the system, enabling efficient monitoring, control, and dynamic remote management of smart lighting infrastructure.

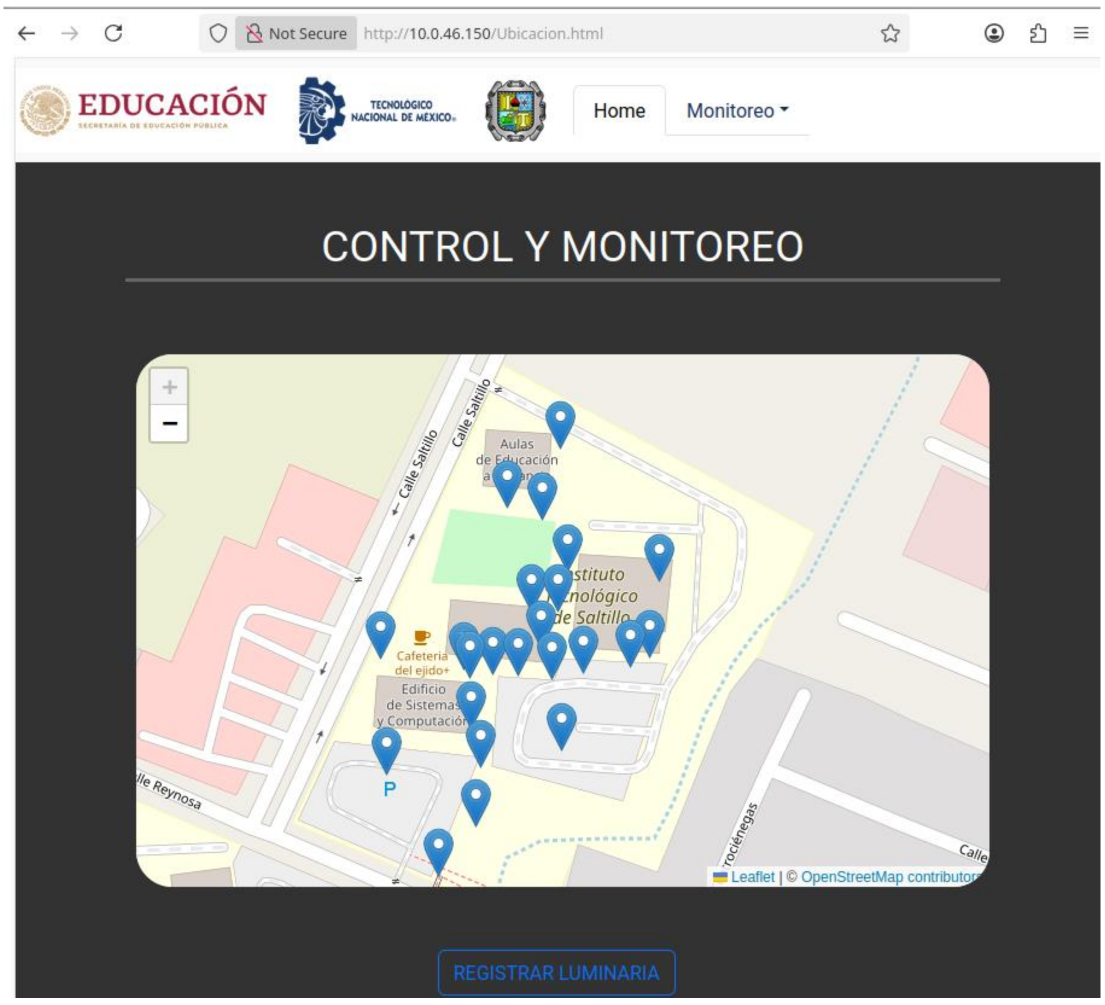

Not Secure
http://10.0.46.150/Ubicacion.html
EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA
TECNOLÓGICO NACIONAL DE MÉXICO
Home
Monitoreo
CONTROL Y MONITOREO
Calle Saltillo
Aulas de Educación
Instituto Tecnológico de Saltillo
Cafeteria del ejido
Edificio de Sistemas y Computación
P
Calle Reynosa
Leaflet | © OpenStreetMap contributors
REGISTRAR LUMINARIA

Figure 7: Luminaire control visualization.

CONCLUSIONS

An intelligent lighting system was developed to provide geospatial location and real-time operational data for luminaires across the Tecnológico de Saltillo campus. Through an interactive web-based map, the system displays status indicators (on/off) and associated metrics for each node. It also enables remote switching control directly from the interface.

Sensor assignment and power-loss conditions are monitored and logged. User access is protected through role-based permissions, limiting administrative actions to authorized users. Although automatic alert notifications for abnormal conditions have not yet been implemented, the system serves as a solid and scalable foundation for future enhancements in smart lighting management.

REFERENCES

1. Chaudhari, B. S., & Zennaro, M. (2020). *LPWAN Technologies for IoT and M2M Aplications.* Academic Press.
2. Gong, T. (7 de abril de 2025). *NEMA vs ZHAGA.* Obtenido de https://es.zgsm-china.com/blog/nema-vs-zhaga.html
3. Red Hat. (20 de Enero de 2023). *¿Qué es una API y cómo funciona?* (Red Hat, Inc.) Recuperado el 28 de mayo de 2025, de https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces.
4. ubicquia. (s.f.). *UbiVu Intelligent*. (Ubicquia) Recuperado el 5 de mayo de 2025, de https://www.ubicquia.com/products/intelligent-asset-management-ubivu