

الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
جامعة أبي بكر بلقايد - تلمسان

Université Abou bakr Belkaïd – Tlemcen –
Faculté de TECHNOLOGIE



MEMOIRE

Présenté pour l'obtention du **diplôme de MASTER**

En : Télécommunications

Spécialité : Télécommunications Optique

Par : OULDALI Yassine

Projet de fin d'études réalisé dans le cadre du diplôme Projet Innovant (1275)

Sujet :

Système IoT de surveillance intelligente de l'éclairage nocturne pour une Smart City

Soutenu publiquement, le 24 /06/2026, devant le jury composé de :

Mr. MOUSSAOUI Djillali	MCA	Univ. Tlemcen	Président
Mr. HADJILA Mourad	PR	Univ. Tlemcen	Examinateur
Mr. BELBACHIR Ahmed	MCA	Univ. Tlemcen	Expert I2E
Mme. OTMANI Amina	MCB	Univ. Tlemcen	Encadrante
Mr. ELGORMA Mourad	Master	Univ. Tlemcen	Co-encadrant
Mme. BOUCHENAK KHELLADI Yassamina	MCB	Univ. Tlemcen	Co-encadrante

Année universitaire : 2025/2026

Résumé

Ce travail présente la conception et la réalisation d'un prototype IoT destiné à la supervision intelligente de l'éclairage public. Le système repose sur des nœuds ESP32 associés à des capteurs LDR pour détecter l'état ON/OFF des lampes, ainsi que sur une unité mobile ESP32–GPS NEO-6M utilisée pour enregistrer les coordonnées géographiques de chaque point lumineux et les associer à un identifiant unique. Les données sont transmises par Wi-Fi via le protocole MQTT vers un broker Mosquitto hébergé sur un Raspberry Pi 4, puis traitées sous Node-RED, stockées dans une base SQLite et exploitées à travers une carte interactive, un tableau d'archives, un historique des pannes et une synthèse globale. Les essais réalisés ont permis de valider l'enregistrement GPS, la communication MQTT, la détection des états lumineux, le stockage des données et le fonctionnement des interfaces de supervision.

Mots clés

Internet des Objets, éclairage public intelligent, ESP32, LDR, GPS NEO-6M, MQTT, Raspberry Pi, Node-RED, SQLite.

Abstract

This work presents the design and implementation of an Internet of Things prototype for the intelligent monitoring of a public lighting network. The system is based on ESP32 nodes connected to LDR sensors for detecting the ON/OFF status of street lamps, together with a mobile ESP32–NEO-6M GPS unit used to register the geographical coordinates of each lamp and associate them with a unique identifier. The collected data are transmitted over Wi-Fi using the MQTT protocol to a Mosquitto broker hosted on a Raspberry Pi 4, then processed in Node-RED, stored in an SQLite database, and displayed through an interactive map, an archive table, a fault history interface, and a global status summary. Experimental tests validated GPS registration, MQTT communication, light-state detection, data storage, and the operation of the monitoring interfaces.

Keywords

Internet of Things, smart public lighting, ESP32, LDR sensor, NEO-6M GPS, MQTT, Raspberry Pi, Node-RED, SQLite.

ملخص

يهدف هذا العمل إلى تصميم وإنجاز نموذج أولي لنظام ذكي قائم على إنترنت الأشياء لمراقبة شبكة الإنارة العمومية، وذلك بالاعتماد على عقد ESP32 مرتبطة بحساسات LDR لتحديد حالة المصابيح ON/OFF ، ووحدة متنقلة مكوّنة من ESP32 و GPS NEO-6M لتسجيل الإحداثيات الجغرافية لكل مصباح وربطها بمعرّف خاص. تُرسل البيانات عبر شبكة Wi-Fi باستعمال بروتوكول MQTT إلى وسيط Mosquitto المثبت على Raspberry Pi 4 ، ثم تتولى منصة-Node RED معالجة الرسائل وتخزينها في قاعدة SQLite وعرضها من خلال خريطة تفاعلية، وجدول أرشيف، وسجل للأعطال، وواجهة تلخيصيه للحالة العامة للشبكة. وقد أظهرت الاختبارات نجاح تسجيل المواقع، واكتشاف حالات التشغيل والإطفاء، ونقل البيانات وتخزينها وعرضها، مما يجعل النموذج قاعدة قابلة للتطوير مستقبلاً عبر تصميم PCB مخصص، واعتماد ESP-WIFI-MESH، وتعزيز أمن الاتصالات.

الكلمات المفتاحية

إنترنت الأشياء، الإنارة العمومية الذكية، ESP32، LDR، GPS NEO-6M، MQTT، Raspberry Pi، Node-RED، SQLite..

Dédicace

Je dédie ce modeste travail,

À ma chère mère, **Zoulikha**, pour son amour inconditionnel, sa tendresse, ses prières et son soutien permanent. Aucun mot ne saurait exprimer toute ma reconnaissance pour les sacrifices qu'elle a consentis afin de me voir réussir.

À mon cher père, **Ahmed**, pour ses encouragements, ses conseils précieux et les efforts qu'il a toujours fournis pour m'accompagner tout au long de mon parcours.

À mes trois chères sœurs :

Sara, Hadjer et Khouloud pour leur affection, leur présence et leur soutien constant.

À mes chers neveux et nièces, **Haythem, Noursine, Joud et Jad**, qui apportent joie, bonheur et douceur à notre famille.

À ma chère, **Newres**, pour son aide précieuse, ses encouragements et sa présence à mes côtés durant la réalisation de ce travail.

À mon cher ami **Fakhr Eddine**, pour son soutien, sa fidélité et les moments partagés tout au long de ce parcours.

À mes **amis d'études**, avec qui j'ai partagé les efforts, les difficultés et les beaux souvenirs de cette aventure universitaire.

À toute **ma famille**, pour son affection, ses encouragements et son soutien.

Enfin, à toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce travail.

Remerciements

Au terme de ce travail, je tiens à exprimer ma profonde gratitude à toutes les personnes qui ont contribué, directement ou indirectement, à la réalisation de ce mémoire.

Je souhaite tout d'abord adresser mes sincères remerciements à Madame **OTMANI Amina**, mon encadrante, pour son accompagnement, ses conseils, sa disponibilité et l'intérêt qu'elle a porté à ce travail tout au long de sa réalisation.

J'exprime également ma profonde reconnaissance à Monsieur **ELGORMA Mourad** pour son aide précieuse, ses orientations techniques et son soutien durant les différentes étapes de ce projet.

Mes sincères remerciements s'adressent aussi à Madame **BOUCHENAK KHALLADI Yassmina** pour son encadrement, ses recommandations et sa contribution à l'amélioration de ce travail.

Je remercie respectueusement les membres du jury, Monsieur **MOUSSAOUI Djillali** et Monsieur Hadjila Mourad, pour l'honneur qu'ils me font en acceptant d'évaluer ce mémoire. Leurs observations et leurs recommandations contribueront à enrichir ce travail.

Je tiens également à remercier Monsieur **BELBACHIR Ahmed**, représentant de centre I2E, ainsi que l'ensemble de l'équipe du centre I2E pour leur accueil, leur accompagnement et leur soutien. Mes remerciements s'adressent aussi au Centre de Développement de l'Entrepreneuriat (CDE) pour les moyens, les orientations et les opportunités mis à la disposition des étudiants.

Enfin, j'adresse mes sincères remerciements à l'ensemble des enseignants de la Faculté de Technologie, particulièrement ceux du Département de Télécommunications, pour la qualité de la formation dispensée, les connaissances transmises et les efforts fournis tout au long de mon parcours universitaire.

Table des matières

Résumé	i
Abstract.....	ii
Dédicace	iv
Remerciements.....	v
Table des matières	vi
Liste des figures	ix
Liste des tableaux.....	xi
Liste des Abréviations	xii
Introduction générale	2
Chapitre I : Fondements de l'Internet des Objets et des Villes Intelligentes	4
I.1 Introduction	5
I.2 Villes intelligentes et Internet of Things	5
I.2.1 Définition de la villes intelligentes.....	5
I.2.2 Définition de l'internet des objets	10
I.3 Surveillance intelligente de l'éclairage nocturne	12
I.3.1 Problématique	12
I.3.2 Solutions existantes.....	12
I.4 Cahier des charges	13
I.4.1 Objectif général.....	13
I.4.2 Matériels nécessaires.....	13
I.5 Conclusion.....	14
Chapitre II : Architecture et conception du système IoT	15
II.1 Introduction.....	16
II.2 Architecture Globale du Système	16
II.2.1 Synoptique Général du Système.....	16
II.2.2 Description des Différents Blocs Fonctionnels.....	17
II.3 Matériel et acquisition des données.....	17
II.3.1 Le Microcontrôleur ESP32.....	18
II.3.2 Capteur LDR	19
II.3.3 Module GPS NEO 6M	22

II.3.4 Raspberry Pi 4 Model B	25
II.3.5 Couche réseau et protocole de communication MQTT	26
II.4 Traitement des données et gestion de la base de données.....	29
II.4.1 Plateforme d'Orchestration : Node-RED	29
II.4.2 Base de données SQLite.....	30
II.5 Conclusion	31
Chapitre III : Implémentation et Développement de la chaîne IoT	32
III.1 Introduction	33
III.2 Présentation du prototype développé	33
III.2.1 Environnement matériel et logiciel.....	33
III.2.2 Architecture réelle du système	34
III.2.3 Scénario opérationnel du système proposé.....	35
III.3 Implémentation et configuration de la chaîne IoT.....	36
III.3.1 Phase de géolocalisation des points d'éclairage.....	36
III.3.2 Acquisition des données lumineuses via ESP32 et LDR.....	39
III.3.3 Publication des données du nœud ESP32 vers le broker MQTT	40
III.3.4 Configuration du broker MQTT Mosquitto sur Raspberry Pi	41
III.3.5 Traitement des données IoT via Node-RED	42
III.3.6 Stockage des données dans une base SQLite	43
III.4 Développement de l'interface de supervision et visualisation des données IoT	44
III.4.1 Interface cartographique de supervision avec Worldmap	44
III.4.2 Interface d'archivage des données.....	45
III.4.3 Interface de consultation de l'historique des pannes.....	45
III.4.4 Interface de synthèse globale du système.....	46
III.5 Conclusion.....	47
Chapitre IV : Expérimentation, validation et évaluation du prototype	48
IV.1 Introduction	49
IV.2 Présentation du prototype expérimental.....	49
IV.2.1 Prototype réel développé.....	49
IV.2.2 Description du montage expérimental	50
IV.2.3 Conditions expérimentales	51

IV.2.4 Scénarios de test et validation fonctionnelle	51
IV.3 Résultats expérimentaux et validation	52
IV.3.1 Validation de l'enregistrement des coordonnées GPS	52
IV.3.2 Validation de la communication MQTT	53
IV.3.3 Validation de la détection de l'état lumineux (LDR)	54
IV.3.4 Validation de la carte interactive de supervisions	55
IV.3.5 Validation de l'historique des pannes	56
IV.3.6 Validation du rapport de synthèse.....	57
IV.4 Discussion des résultats.....	58
IV.4.1 Évaluation des performances du prototype.....	58
IV.4.2 Valeur ajoutée du prototype.....	59
IV.5 Intégration physique du prototype.....	59
IV.5.1 Intégration du prototype dans un boîtier	60
IV.5.2 Stratégie d'alimentation synchronisée avec la ligne d'éclairage	61
IV.6 Conclusion.....	62
Conclusion générale et perspectives	64
Bibliographie	68
Annexes	76

Liste des figures

Figure I.1 Schéma programmatique d'une ville intelligente selon l'agence européenne Dédale consacrée à la culture, aux technologies et à l'innovation sociale [4]	6
Figure I.2 Les critères principaux de la Ville intelligente	9
Figure I.3 Exemple sur l'Internet des objets [25]	10
Figure I.4 Schéma représentatif de notre projet.....	14
Figure II.1 Schéma synoptique général.....	16
Figure II.2 Module Wifi Bluetooth ESP32	18
Figure II.3 Référence des broches de l'ESP32	18
Figure II.4 Capteur photorésistant LDR 7 mm.....	20
Figure II.5 Schéma équivalent de la LDR	20
Figure II.6 Schéma de câblage de LDR avec ESP32	21
Figure II.7 Montage du capteur lumineux a la base d'ESP32	21
Figure II.8 Module GPS Neo 6m.....	22
Figure II.9 Schéma de câblage GPS NEO 6M avec ESP32.....	24
Figure II.10 Montage de GPS NEO 6M avec ESP32.....	24
Figure II.11 Raspberry pi 4 Model B	25
Figure II.12 Architecture et fonctionnement du protocole MQTT dans un système IoT	28
Figure III.1 Les composants matériels utilisés dans le prototype.....	34
Figure III.2 Architecture fonctionnelle du système de supervision IoT.....	35
Figure III.3 Scénario global de fonctionnement du prototype de supervision IoT	36
Figure III.4 Organigramme du processus d'enregistrement des coordonnées GPS des points d'éclairage	37
Figure III.5 Table des lampes avec leurs coordonnées GPS.	38
Figure III.6 Organigramme de l'algorithme d'acquisition de l'état lumineux à l'aide de l'ESP32 et du capteur LDR.....	39
Figure III.7 Illustration du processus de publication des données des nœuds ESP32 vers le broker MQTT	41
Figure III.8 Vérification de l'état du service Mosquitto sur Raspberry Pi.....	41
Figure III.9 Flux Node-RED de traitement des données MQTT.....	43
Figure III.10 Interface cartographique Worldmap pour la supervision des lampes	44

Liste des figures

Figure III.11 Interface Archives des états lumineux enregistrés.....	45
Figure III.12 Interface Historique des pannes des lampes supervisées	46
Figure III.13 Interface de synthèse globale du réseau d'éclairage.....	46
Figure IV.1 Prototype expérimental complet du système de supervision IoT	50
Figure IV.2 Montage expérimental utilisé pour les essais de validation	51
Figure IV.3 Flux Node-RED d'enregistrement des coordonnées GPS dans SQLite.....	53
Figure IV.4 Test de réception des messages MQTT sous Node-RED	54
Figure IV.5 Détection de l'état (a) ON (b) OFF d'une lampe à partir du capteur LDR.....	55
Figure IV.6 Validation de l'affichage cartographique des lampes sous Node-RED Worldmap.....	56
Figure IV.7 Historique des pannes des lampes supervisées	57
Figure IV.8 Validation du rapport de synthèse globale du système	57
Figure IV.9 Boîtiers d'encapsulation choisi pour l'intégration du prototype.....	61
Figure IV.10 Principe d'alimentation synchronisée du nœud IoT avec la ligne d'éclairage public.	62

Liste des tableaux

Tableau II.1 Les caractéristiques de la carte ESP32.....	19
Tableau II.2 Les caractéristiques de Raspberry Pi 4 Model B.....	26
Tableau III.1 Environnement matériel et logiciel utilisé.....	34
Tableau III.2 Structure des tables SQLite utilisées pour le stockage des données.....	43
Tableau IV.1 Évaluation fonctionnelle des performances du prototype.....	59

Liste des Abréviations

- **AC** : Alternating Current — Courant alternatif.
- **ACL** : Access Control List — Liste de contrôle d'accès.
- **ADC** : Analog-to-Digital Converter — Convertisseur analogique-numérique.
- **API** : Application Programming Interface — Interface de programmation d'applications.
- **ARM** : Advanced RISC Machine — Architecture de processeurs à jeu d'instructions réduit.
- **ARPANET** : Advanced Research Projects Agency Network — Réseau précurseur d'Internet.
- **AWS** : Amazon Web Services — Services d'informatique en nuage d'Amazon.
- **BLE** : Bluetooth Low Energy — Bluetooth à basse consommation.
- **CPU** : Central Processing Unit — Unité centrale de traitement.
- **CSI** : Camera Serial Interface — Interface série destinée aux caméras.
- **CSS** : Cascading Style Sheets — Feuilles de style en cascade.
- **DAC** : Digital-to-Analog Converter — Convertisseur numérique-analogique.
- **DC** : Direct Current — Courant continu.
- **DIY** : Do It Yourself — Faire soi-même.
- **DMIPS** : Dhrystone Million Instructions Per Second — Unité de mesure des performances d'un processeur.
- **DSI** : Display Serial Interface — Interface série destinée aux écrans.
- **EEPROM** : Electrically Erasable Programmable Read-Only Memory — Mémoire morte programmable et effaçable électriquement.
- **ES** : Embedded Systems — Systèmes embarqués.
- **FBP** : Flow-Based Programming — Programmation orientée flux.
- **FTP** : File Transfer Protocol — Protocole de transfert de fichiers.
- **GND** : Ground — Masse électrique du circuit.
- **GPGGA** : Global Positioning System Fix Data — Trame NMEA contenant les données de positionnement GPS.
- **GPIO** : General-Purpose Input/Output — Entrée-sortie à usage général.
- **GPRMC** : Recommended Minimum Specific GNSS Data — Trame NMEA contenant les données minimales de navigation.

- **GPS** : Global Positioning System — Système mondial de positionnement.
- **GSM** : Global System for Mobile Communications — Système mondial de communications mobiles.
- **HDMI** : High-Definition Multimedia Interface — Interface multimédia haute définition.
- **HTML** : HyperText Markup Language — Langage de balisage hypertexte.
- **HTTP** : HyperText Transfer Protocol — Protocole de transfert hypertexte.
- **I2C** : Inter-Integrated Circuit — Bus série de communication entre circuits intégrés.
- **I2S** : Inter-IC Sound — Bus série utilisé pour transmettre les données audio numériques.
- **IBM** : International Business Machines.
- **IDE** : Integrated Development Environment — Environnement de développement intégré.
- **IEEE** : Institute of Electrical and Electronics Engineers — Institut des ingénieurs électriciens et électroniciens.
- **IHM** : Interface Homme-Machine.
- **IoT** : Internet of Things — Internet des objets.
- **IP** : Internet Protocol — Protocole Internet.
- **IP65** : Indice de protection contre la poussière et les jets d'eau.
- **IP67** : Indice de protection contre la poussière et l'immersion temporaire dans l'eau.
- **IPv4** : Internet Protocol Version 4 — Protocole Internet version 4.
- **IPv6** : Internet Protocol Version 6 — Protocole Internet version 6.
- **IR** : Infrared — Infrarouge.
- **ISO** : International Organization for Standardization — Organisation internationale de normalisation.
- **IEC** : International Electrotechnical Commission — Commission électrotechnique internationale.
- **JSON** : JavaScript Object Notation — Format textuel d'échange de données.
- **LDR** : Light Dependent Resistor — Photorésistance dépendante de la lumière.
- **LED** : Light-Emitting Diode — Diode électroluminescente.
- **LPDDR4** : Low-Power Double Data Rate 4 — Mémoire vive DDR de quatrième génération à faible consommation.
- **M2M** : Machine-to-Machine — Communication directe entre machines.

- **MaaS** : Mobility as a Service — Mobilité en tant que service.
- **MIPI** : Mobile Industry Processor Interface — Interface destinée aux processeurs des appareils mobiles.
- **MQTT** : Message Queuing Telemetry Transport — Protocole léger de communication basé sur le modèle publication-abonnement.
- **NMEA** : National Marine Electronics Association — Norme de transmission des données de navigation et de géolocalisation.
- **OpenGL** : Open Graphics Library — Bibliothèque graphique ouverte.
- **OS** : Operating System — Système d'exploitation.
- **P2P** : Peer-to-Peer — Communication de pair à pair.
- **PC** : Personal Computer — Ordinateur personnel.
- **PCB** : Printed Circuit Board — Circuit imprimé.
- **PUBACK** : Publish Acknowledgement — Accusé de réception d'un message publié avec MQTT.
- **PWM** : Pulse-Width Modulation — Modulation de largeur d'impulsion.
- **QoS** : Quality of Service — Qualité de service.
- **RAM** : Random Access Memory — Mémoire vive.
- **ROM** : Read-Only Memory — Mémoire morte.
- **RTC** : Real-Time Clock — Horloge temps réel.
- **RX** : Receive ou Receiver — Réception ou récepteur.
- **SAR** : Successive Approximation Register — Registre à approximations successives.
- **SBC** : Single-Board Computer — Ordinateur monocarte.
- **SD** : Secure Digital — Format de carte mémoire.
- **SDIO** : Secure Digital Input/Output — Interface d'entrée-sortie basée sur le standard SD.
- **SDRAM** : Synchronous Dynamic Random Access Memory — Mémoire vive dynamique synchrone.
- **SGBDR** : Système de gestion de base de données relationnelle.
- **SIG** : Système d'information géographique.
- **SoC** : System on a Chip — Système sur puce.
- **SoftAP** : Software Access Point — Point d'accès logiciel.

- **SPI** : Serial Peripheral Interface — Interface série destinée aux périphériques.
- **SRAM** : Static Random Access Memory — Mémoire vive statique.
- **SSH** : Secure Shell — Protocole sécurisé d'accès à distance.
- **SSL** : Secure Sockets Layer — Protocole de sécurisation des communications réseau.
- **TCP** : Transmission Control Protocol — Protocole de contrôle de transmission.
- **TIC** : Technologies de l'information et de la communication.
- **TX** : Transmit ou Transmitter — Transmission ou émetteur.
- **UART** : Universal Asynchronous Receiver-Transmitter — Émetteur-récepteur asynchrone universel.
- **UDP** : User Datagram Protocol — Protocole de datagrammes utilisateur.
- **UI** : User Interface — Interface utilisateur.
- **USB** : Universal Serial Bus — Bus série universel.
- **USB-C** : Universal Serial Bus Type-C — Connecteur USB de type C.
- **UV** : Ultraviolet — Rayonnement ultraviolet.
- **VCC** : Tension positive d'alimentation d'un circuit électronique.
- **Wi-Fi** : Technologie de réseau local sans fil.

Introduction générale

Introduction générale

L'évolution des systèmes urbains vers des infrastructures intelligentes repose de plus en plus sur l'intégration des technologies de l'information, des réseaux de communication et des systèmes embarqués. Dans ce contexte, l'éclairage public constitue une infrastructure essentielle, car il participe à la sécurité des usagers, à l'organisation de l'espace urbain et au confort des citoyens. Cependant, la gestion traditionnelle de ce type de réseau reste souvent basée sur des inspections manuelles ou sur des signalements tardifs, ce qui limite la rapidité de détection des pannes et complique les opérations de maintenance.

L'Internet des Objets offre une approche adaptée pour moderniser la supervision de l'éclairage public. En intégrant des capteurs, des microcontrôleurs et des protocoles de communication légers, il devient possible de collecter les données directement sur le terrain, de les transmettre vers une plateforme centrale et de les exploiter en temps réel. Cette évolution permet de transformer un réseau d'éclairage classique en un système communicant capable d'identifier l'état de chaque lampe, de localiser les points lumineux et de fournir des informations exploitables pour la maintenance [1], [2].

Cette étude s'inscrit dans la conception et la réalisation d'un prototype IoT destiné à la supervision intelligente de l'éclairage public. La solution proposée repose sur des nœuds embarqués à base d'ESP32 associés à des capteurs LDR afin de détecter l'état lumineux des lampes. Un module GPS NEO-6M est utilisé comme unité mobile d'initialisation pour enregistrer les coordonnées géographiques de chaque lampe et les associer à un identifiant unique. Les données collectées sont ensuite transmises via le protocole MQTT vers un Raspberry Pi, qui joue le rôle de plateforme locale de traitement.

La plateforme centrale intègre le broker Mosquitto, l'environnement Node-RED et une base de données SQLite. Mosquitto assure la réception des messages MQTT publiés par les ESP32, tandis que Node-RED permet le traitement des flux, l'association entre les identifiants des lampes et leurs coordonnées GPS, le stockage des données et l'affichage des informations sur des interfaces de supervision. Ces interfaces comprennent une carte interactive, un tableau d'archives, un historique des pannes et une synthèse globale de l'état du réseau.

La problématique principale de ce mémoire peut être formulée comme suit : comment concevoir une architecture IoT locale, légère et fiable permettant de détecter l'état ON/OFF des

lampes d'un réseau d'éclairage public, de les localiser géographiquement et de présenter les informations sous une forme exploitable pour la supervision et la maintenance?

Pour répondre à cette problématique, une démarche expérimentale a été adoptée. Elle repose sur l'acquisition de l'état lumineux par capteur LDR, l'enregistrement initial des positions par GPS, la transmission des données par MQTT, le traitement sous Node-RED, le stockage local dans SQLite et la visualisation en temps réel. Cette chaîne permet de valider la faisabilité d'un système de supervision capable d'identifier les lampes opérationnelles et les lampes présentant une extinction ou une anomalie.

Le présent mémoire s'articule autour de quatre chapitres. Le premier chapitre présente le cadre général des villes intelligentes et le rôle de l'Internet des Objets dans la modernisation des infrastructures urbaines, notamment dans le domaine de l'éclairage public. Le deuxième chapitre décrit l'architecture globale du système proposé ainsi que les principaux composants matériels et logiciels utilisés, tels que l'ESP32, le capteur LDR, le module GPS NEO-6M, le Raspberry Pi, MQTT, Node-RED et SQLite. Le troisième chapitre est consacré à l'implémentation pratique du prototype, au développement de la chaîne IoT, aux interfaces de supervision et à la validation expérimentale des résultats obtenus. Enfin, le quatrième chapitre aborde l'étude de déploiement industriel du système, en traitant les aspects liés à la conception matérielle, à la protection environnementale, à l'alimentation, à la topologie ESP-WiFi-MESH, à la sécurité des communications et au diagnostic avancé.

À travers ce travail, l'objectif est de proposer une solution fonctionnelle, simple et évolutive pour la supervision intelligente de l'éclairage public. Le prototype réalisé constitue une base technique pouvant être améliorée dans le cadre d'un déploiement réel, notamment par l'intégration d'une alimentation autonome ou synchronisée, d'une communication maillée, d'une protection matérielle adaptée aux conditions extérieures et de mécanismes de sécurité renforcés.

Chapitre I : Fondements de l'Internet des Objets et des Villes Intelligentes

I.1 Introduction

L'évolution rapide de la démographie urbaine et l'augmentation constante des activités humaines imposent aujourd'hui un défi majeur aux infrastructures des grandes métropoles. Pour répondre à ces enjeux de résilience et de durabilité, le concept de ville intelligente (ou Smart City) s'est imposé comme un nouveau paradigme de planification urbaine. La ville intelligente représente l'aboutissement d'une évolution continue visant à optimiser les services publics tout en améliorant la qualité de vie des citoyens grâce à l'exploitation intelligente des données.

Ce modèle repose sur l'intégration des Technologies de l'Information et de la Communication (TIC) pour accroître l'efficacité de la gestion urbaine. Son objectif central est de rendre les services plus performants, de réduire les coûts opérationnels et de favoriser un développement économique pérenne tout en préservant les ressources environnementales.

Au cœur de cette transformation se trouve l'Internet des Objets (Internet of Things, IoT), qui fait office de véritable « système nerveux » de la ville intelligente. L'IoT permet d'interconnecter une multitude d'équipements physiques capables de capter, traiter et transmettre des informations en temps réel. Cette interconnexion transforme les infrastructures statiques en systèmes dynamiques et réactifs, permettant par exemple une gestion optimisée de l'éclairage public ou de l'énergie.

À travers l'étude de ces technologies, ce chapitre pose le cadre théorique et technique nécessaire pour comprendre comment le numérique favorise un développement urbain plus écoresponsable et participatif.

I.2 Villes intelligentes et Internet of Things

L'évolution des technologies numériques et des systèmes de communication a profondément transformé la manière de concevoir et de gérer les espaces urbains. Dans ce contexte, les villes intelligentes et l'Internet des objets (IoT) apparaissent comme des approches innovantes visant à améliorer la performance des infrastructures, à optimiser l'exploitation des ressources et à renforcer la qualité des services proposés aux citoyens. Grâce à l'intégration des technologies connectées, ces nouveaux modèles urbains favorisent une gestion plus efficace, durable et adaptée aux exigences croissantes des environnements modernes.

I.2.1 Définition de la villes intelligentes

Le concept des villes intelligentes repose sur un éventail d'innovations technologiques destinées à perfectionner les espaces urbains grâce à l'exploitation et à l'analyse des données. Une agglomération se transforme en ville intelligente dès lors qu'elle parvient à interconnecter et centraliser ses flux d'informations et ses systèmes numériques. L'objectif est de rendre le

fonctionnement de ses services plus performant, tout en favorisant la résilience, le développement économique pérenne et un cadre de vie optimal. Le défi principal consiste à apporter des solutions aux infrastructures des grandes métropoles occidentales, souvent mises à rude épreuve par une démographie et des activités en constante augmentation [1].

Par conséquent, l'expression « ville intelligente » désigne une métropole qui s'appuie sur les TIC afin d'enrichir la qualité des services urbains et d'en diminuer les coûts. Elle intègre également de multiples équipements physiques connectés au réseau, formant ainsi l'Internet des objets (IoT), pour fluidifier les opérations, optimiser les services et renforcer les interactions avec les habitants. De ce fait, une ville intelligente se doit de piloter des infrastructures interconnectées, flexibles et durables, dont l'automatisation vise à accroître le bien-être des citoyens tout en préservant l'environnement [2].

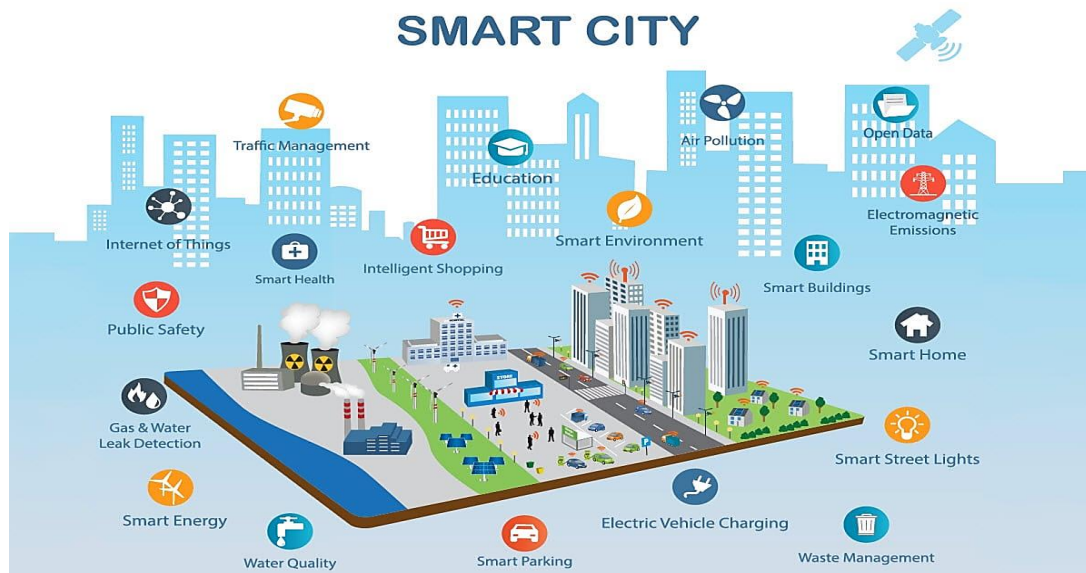


Figure I.1 Schéma programmatique d'une ville intelligente selon l'agence européenne Dédale consacrée à la culture, aux technologies et à l'innovation sociale [4]

En somme, une ville intelligente se caractérise par des investissements ciblés dans le capital humain et social, couplés à la modernisation de ses réseaux de communication. Cette démarche a pour finalité d'élever le niveau de vie de ses résidents et de stimuler l'économie locale, tout en assurant une administration écoresponsable des ressources et en déployant des outils qui encouragent l'implication active des citoyens dans le façonnement de leur propre ville [3].

I.2.1.1 Évolution du Villes intelligents

La ville intelligente est l'aboutissement d'une quête historique visant à optimiser l'urbanisme. Cette évolution s'articule autour de jalons technologiques et conceptuels majeurs :

Héritage pré-industriel (XIIIe - XVe s.) : L'Alhambra de Grenade illustre une forme d'intelligence urbaine avant l'heure, grâce à une architecture bioclimatique, un réseau hydraulique gravitaire et des principes de permaculture [5].

Genèse des télécommunications (1880) : Avec l'invention du « photophone » par Graham Bell, le principe de la transmission par la lumière voit le jour, posant les bases des systèmes optiques modernes, notamment la fibre optique [6].

Cybernétique et vision systémique (années 1950) : L'urbanisme intègre la cybernétique pour concevoir la ville comme un organisme vivant régulé par des boucles de rétroaction, tandis que les premiers Systèmes d'Information Géographique (SIG) émergent [7].

Convergence numérique (1969 - 1990) : La naissance d'ARPANET et l'essor de l'informatique embarquée permettent d'automatiser certaines infrastructures urbaines, marquant le début de la gestion en temps réel de l'éclairage et de la surveillance [8].

Pionnier coréen (2003 - 2006) : La Corée du Sud institutionnalise l'« ubiquité numérique » avec le projet de Songdo, intégrant l'informatique de manière invisible au sein de l'environnement urbain [9].

Impulsion industrielle (2008 - 2009) : Le concept prend une dimension mondiale sous l'impulsion de géants tels qu'IBM, qui identifient dans la gestion des données urbaines (Big Data) un levier économique majeur face aux crises [10].

Maturité et standardisation : La Smart City est désormais un paradigme structuré, évalué mondialement selon six piliers : Smart Economy (économie intelligente), Smart People (citoyens intelligents), Smart Governance (gouvernance intelligente), Smart Mobility (mobilité intelligente), Smart Environment (environnement intelligent) et Smart Living (cadre de vie intelligent) [11].

I.2.1.2 Les objectifs de la ville intelligente

La vocation d'une ville intelligente dépasse largement le simple déploiement d'infrastructures technologiques. Ses objectifs fondamentaux s'articulent autour de l'optimisation rationnelle des coûts d'exploitation, d'une structuration plus efficiente des services urbains et d'une maîtrise globale du territoire. L'ambition ultime de cette démarche est l'amélioration tangible et continue du bien-être, de la sécurité et du confort des habitants. Cette vision se décline à travers plusieurs axes majeurs :

- Un levier de gouvernance numérique et d'optimisation managériale : La ville intelligente s'affirme avant tout comme un puissant instrument de gouvernance, de digitalisation et de gestion. L'intégration des technologies de l'information permet de dématérialiser les processus administratifs (e-administration) et d'adopter une approche décisionnelle basée sur l'analyse des

données urbaines (data-driven decision making). Cela permet aux collectivités de passer d'une gestion réactive à une gestion prédictive et proactive de l'espace urbain [12].

- Dynamisation territoriale et synergie d'un écosystème multiacteurs : La ville intelligente a pour ambition de métamorphoser le tissu urbain en un territoire vivant, résilient et hautement dynamique. Elle vise à repenser l'aménagement de l'espace public et de l'habitat pour répondre aux nouveaux usages démographiques. Plus qu'une simple restructuration physique, elle se veut le catalyseur de l'économie collaborative et de la cohésion sociale sous toutes ses formes. Pour ce faire, la ville intelligente orchestre de nouveaux modèles de synergie et de collaboration transversale entre une multitude d'acteurs autrefois cloisonnés : les collectivités locales, les opérateurs énergétiques, les urbanistes et aménageurs, les bailleurs et promoteurs immobiliers, le secteur privé (entreprises de services et de technologies), le monde académique (universités et centres de recherche) et, bien entendu, les citoyens [13].

- Le défi crucial de la transition énergétique et de la résilience écologique : Face à l'urgence climatique mondiale, la ville intelligente s'impose comme un vecteur indispensable de la transition énergétique. Elle intègre des outils avancés de planification et de régulation (tels que les smart grids, ou réseaux électriques intelligents) pour lutter activement contre le réchauffement climatique et freiner l'épuisement des ressources naturelles. L'enjeu est de concilier développement urbain, sobriété énergétique et gestion durable des déchets, afin de garantir au plus grand nombre l'accès à des conditions de vie saines, équitables et satisfaisantes sur le long terme [14].

- L'autonomisation citoyenne par l'interactivité des données (Civic Tech) : La Smart City déploie des modèles dynamiques dotés d'interfaces numériques accessibles (applications mobiles, portails open data, plateformes participatives). Ces outils démocratisent l'accès à l'information, offrant à chaque citoyen la capacité de consulter, de comprendre, d'analyser et d'influencer en temps réel les données de son environnement (qualité de l'air, état du trafic routier, gestion de l'énergie). Cette interactivité favorise une citoyenneté active et renforce le lien de confiance entre les administrateurs et la population [15].

I.2.1.3 Les caractéristiques de la smart city

La transformation numérique des villes dépasse aujourd'hui la simple modernisation technique. Comme le souligne Rudolf Giffinger, chercheur à l'Université technologique de Vienne, la « ville intelligente » repose sur une approche globale articulée autour de six piliers de performance, devenus une référence internationale [16] :



Figure I.2 Les critères principaux de la Ville intelligente

I.2.1.3.1 Smart Economy (économie intelligente)

L'accent est mis sur la compétitivité et l'innovation. Une ville intelligente favorise un écosystème dynamique pour les start-ups, tout en adaptant son marché du travail à la transition numérique afin de rester compétitive à l'échelle mondiale [17].

I.2.1.3.2 Smart People (citoyens intelligents)

Au-delà de la technologie, ce pilier valorise le capital humain. Il s'agit de favoriser l'éducation tout au long de la vie, l'inclusion numérique, la créativité et l'engagement citoyen pour construire une société ouverte et tolérante [18].

I.2.1.3.3 Smart Governance (gouvernance intelligente)

L'administration se modernise pour devenir plus agile, transparente et collaborative. Grâce à l'Open Data et à l'e-démocratie, les citoyens sont directement impliqués dans les processus de décision, décloisonnant ainsi les services publics [19].

I.2.1.3.4 Smart Mobility (mobilité intelligente)

L'objectif est de fluidifier les déplacements via des infrastructures connectées aux TIC. Ce pilier privilégie l'intermodalité (MaaS), les transports partagés et durables, ainsi qu'une gestion intelligente du trafic en temps réel pour désengorger les centres urbains [20].

I.2.1.3.5 Smart Environment (environnement intelligent)

Face aux enjeux climatiques, la ville intelligente mise sur la résilience écologique. Elle optimise la gestion des ressources (énergie, eau, déchets) grâce aux réseaux intelligents (smart grids) et promeut des espaces urbains sains, verts et à faible impact environnemental [21].

I.2.1.3.6 Smart Living (mode de vie intelligent)

C'est la finalité du modèle : améliorer la qualité de vie des résidents. Cela passe par un accès facilité aux services de santé (télémédecine) et à la culture, une sécurité renforcée, ainsi qu'un habitat connecté (domotique), le tout visant à garantir une cohésion sociale durable [22].

I.2.2 Définition de l'internet des objets

L'Internet des Objets (IoT) est le pilier technologique de la ville intelligente. Il forme un écosystème où des objets embarqués captent et traitent des flux massifs de données en temps réel pour assurer une continuité entre le monde matériel et numérique. Ses applications sont vastes, allant du grand public aux environnements industriels et critiques, comme les réseaux énergétiques ou les télécommunications [23].

L'objectif de l'IoT est de connecter ces « choses » au réseau mondial pour qu'elles interagissent de manière autonome (M2M) ou avec les utilisateurs. Cette montée en puissance repose sur deux piliers :

L'identification unique : chaque objet dispose d'une adresse propre (grâce à l'IPv6, qui lève les limites de l'adressage traditionnel).

L'interopérabilité : l'usage de protocoles légers comme MQTT et de plateformes d'orchestration comme Node-RED permet à des équipements hétérogènes de communiquer entre eux, indépendamment de leur origine ou de leur conception, garantissant ainsi une intégration fluide des services [24].

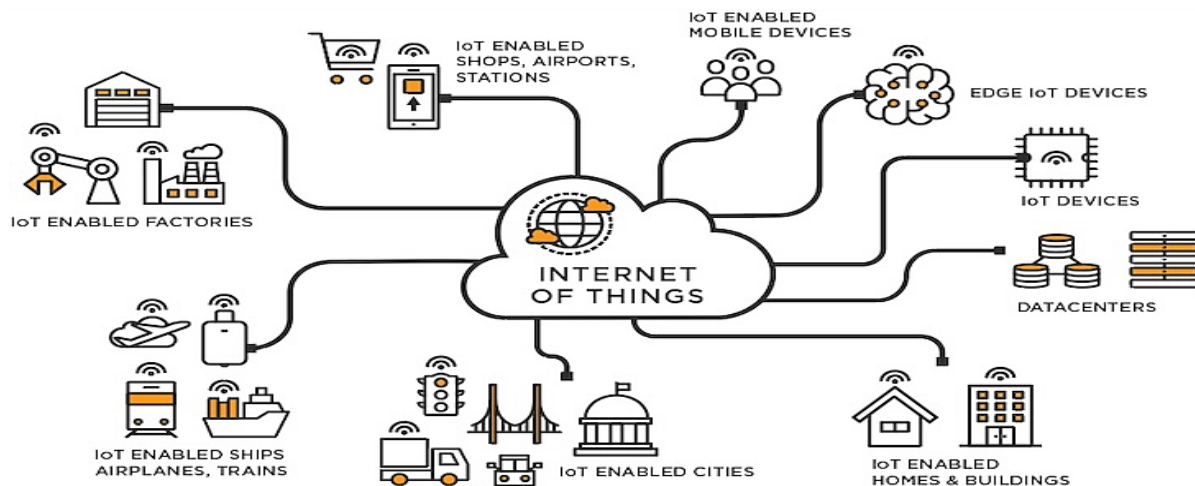


Figure I.3 Exemple sur l'Internet des objets [25]

Le déploiement des villes intelligentes repose sur une multitude de services numériques spécialisés, interconnectés grâce aux architectures IoT et aux réseaux télécoms. Voici les domaines stratégiques :

I.2.3.1 Mobilité et transports

L'objectif est de fluidifier le trafic via la télématique, des feux adaptatifs et la centralisation des données en temps réel, permettant une communication directe entre infrastructures et véhicules [26].

I.2.3.2 Environnement et énergie

Ce pilier optimise la gestion des ressources. L'éclairage public intelligent, par exemple, utilise des capteurs de mouvement et de luminosité pour ajuster l'intensité lumineuse selon la fréquentation réelle, réduisant drastiquement la consommation électrique [27].

I.2.3.3 Smart buildings

À l'aide de microcontrôleurs et de plateformes de supervision, les bâtiments gèrent de manière autonome le chauffage, la climatisation et la sécurité, garantissant une efficacité énergétique et un confort optimal [28].

I.2.3.4 Éducation numérique

L'infrastructure haut débit facilite l'apprentissage à distance, la modélisation en laboratoire et l'accès aux bibliothèques virtuelles, tout en rationalisant les coûts de fonctionnement des institutions académiques [29].

I.2.3.5 Tourisme connecté

L'expérience des visiteurs est enrichie par la réalité augmentée, la géolocalisation interactive et des services personnalisés, rendant le patrimoine culturel plus accessible et immersif [30].

I.2.3.6 Santé et télémédecine

La digitalisation des parcours de soins et l'utilisation d'objets connectés favorisent une médecine préventive, accélérant la prise en charge et optimisant la gestion des établissements de santé [31].

I.2.3.7 Sécurité publique

Le déploiement de caméras intelligentes et l'analyse de données en temps réel permettent aux centres de commandement d'anticiper les incidents et de coordonner les interventions d'urgence avec précision [32].

I.2.3.8 Agriculture intelligente

Des réseaux de capteurs sans fil surveillent l'humidité et la météo. Les données, transmises via des protocoles légers (MQTT), automatisent l'irrigation et améliorent la traçabilité des cultures [33].

I.2.3.9 Gestion municipale

En centralisant tous les flux au sein d'un « hyperviseur urbain », les autorités rationalisent la prise de décision, assurant une gouvernance transparente et une gestion agile des services publics [34].

L'étude des concepts de ville intelligente et d'Internet des objets permet de mieux comprendre les bases technologiques des systèmes urbains intelligents. Dans ce contexte, une analyse de l'existant dans le domaine de l'éclairage public intelligent s'impose afin d'identifier les approches déjà proposées ainsi que leurs limites.

I.3 Surveillance intelligente de l'éclairage nocturne

Les systèmes d'éclairage public ont connu une évolution remarquable, passant par étapes successives d'appareils manuels traditionnels à des solutions automatisées, puis intelligentes, intégrant progressivement les technologies de l'Internet des objets (IoT). Cette transformation s'inscrit dans une tendance plus large de modernisation des infrastructures urbaines et d'optimisation de la consommation d'énergie.

I.3.1 Problématique

Dans les systèmes traditionnels, la gestion de l'éclairage repose principalement sur l'intervention humaine et des horaires fixes. Ce modèle présente plusieurs limitations majeures :

- Pannes de lampes non détectées en temps réel ;
- Interventions de maintenance tardives et coûteuses ;
- Consommation énergétique inutile due à un fonctionnement non optimisé ;
- Absence de système de supervision centralisé.

Ces limitations entraînent une dégradation de la qualité de service ainsi qu'une augmentation des coûts d'exploitation pour les collectivités.

I.3.2 Solutions existantes

L'introduction de l'automatisation, des dispositifs photovoltaïques et des minuteries a partiellement amélioré la gestion de l'éclairage public. Cependant, ces solutions restent limitées par leur incapacité à centraliser la surveillance ou à rendre compte de l'état des lampes [35].

L'émergence des technologies IoT a ouvert la voie à des systèmes d'éclairage intelligent plus avancés. Ces solutions reposent sur l'intégration de capteurs intelligents, de modules de communication sans fil (GSM, Wi-Fi,...) et de plateformes de surveillance capables de traiter les données en temps réel. Ces technologies permettent notamment la détection automatique des pannes, la localisation des anomalies et l'optimisation dynamique de l'intensité lumineuse [36].

Cependant, malgré ces avancées, les solutions actuelles présentent encore plusieurs limitations :

- La dépendance à une infrastructure réseau coûteuse.
- La consommation énergétique parfois élevée des unités de communication.
- La complexité de l'intégration dans des environnements urbains hétérogènes.
- L'absence d'architectures standardisées.

Par conséquent, une analyse des technologies les plus récentes met en évidence la nécessité de solutions plus flexibles et économiques, adaptées aux exigences des villes en développement, grâce à l'intégration de technologies embarquées optimisées et d'une architecture d'Internet des objets (IoT) simplifiée.

I.4 Cahier des charges

Dans le cadre de ce projet de surveillance intelligente de l'éclairage public basé sur l'Internet des Objets (IoT), un cahier des charges est défini afin de formaliser les besoins fonctionnels et non fonctionnels du système. Il constitue une étape essentielle permettant de cadrer le développement de la solution et d'assurer la cohérence entre les objectifs du projet et les fonctionnalités attendues.

I.4.1 Objectif général

L'objectif principal de ce projet s'inscrit pleinement dans le paradigme de l'Internet des Objets (IoT), un domaine qui révolutionne l'interaction entre les infrastructures physiques et les systèmes d'information. Notre démarche vise à concevoir et à déployer un système de monitoring embarqué et intelligent, spécifiquement dédié à la supervision technique continue des réseaux d'éclairage. L'idée centrale est de transformer un équipement matériel traditionnellement passif en un nœud de communication actif, capable de diagnostiquer et de transmettre son propre état opérationnel.

Le système doit permettre les fonctionnalités suivantes :

- Détecter l'état de fonctionnement des lampadaires (allumé, éteint ou en panne) ;
- Collecter les données issues des capteurs installés sur les points lumineux ;
- Transmettre les informations vers une plateforme centrale de supervision ;
- Afficher l'état des lampadaires sur une interface utilisateur ;
- Signaler automatiquement les anomalies ou dysfonctionnements.

I.4.2 Matériels nécessaires

Pour la réalisation de cette solution technique, nous avons utilisé les composants matériels et logiciels suivants :

- Microcontrôleur ESP32.

- Capteur de luminosité (LDR).
- Module GPS NEO-6M.
- Raspberry Pi 4 Model B.
- Interface de supervision basée sur Node-RED

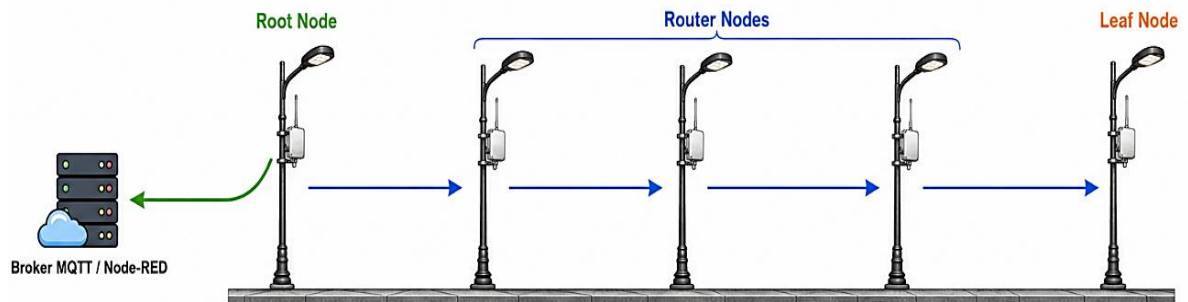


Figure I.4 Schéma représentatif de notre projet

Une description approfondie de ces différents composants, ainsi que leur intégration dans l'architecture du système proposé, sera développée dans le chapitre suivant.

I.5 Conclusion

En guise de conclusion pour ce premier chapitre, nous avons établi le cadre théorique des villes intelligentes (Smart Cities) et démontré le rôle central de l'Internet des Objets (IoT) dans la modernisation des infrastructures urbaines. L'étude a mis en évidence la nécessité de passer d'une gestion réactive à une approche automatisée, particulièrement pour optimiser la supervision de l'éclairage public. À travers la définition de l'objectif de notre projet, nous avons souligné l'importance de transformer les équipements statiques en nœuds actifs capables de diagnostiquer et de transmettre leur état en temps réel (Edge Computing) afin d'isoler numériquement et immédiatement les défaillances matérielles.

Maintenant que le contexte fondamental et l'architecture globale de notre système de monitoring sont clairement définis, il convient de passer à la phase de concrétisation technique. Ainsi, le chapitre suivant sera entièrement consacré à la réalisation pratique : nous y détaillerons les spécificités des composants matériels sélectionnés (microcontrôleur ESP32, capteur LDR, module GPS NEO 6-M et Raspberry Pi 4), ainsi que l'architecture électronique et les schémas de câblage nécessaires à la mise en œuvre de notre solution.

Chapitre II : Architecture et conception du système IoT

II.1 Introduction

Après avoir présenté le cadre théorique de l'Internet des objets et son rôle dans la gestion intelligente des infrastructures urbaines, ce deuxième chapitre est consacré à la conception et à la réalisation pratique du système proposé. Il présente l'architecture globale de la solution, organisée autour de l'acquisition des données, de la communication réseau, du traitement des informations et de l'interface de supervision.

Ce chapitre décrit également les principaux composants matériels utilisés, notamment l'ESP32, le capteur de luminosité LDR, le module GPS NEO-6M et le Raspberry Pi 4 Model B. Enfin, il met en évidence le rôle du protocole MQTT dans la transmission légère et fiable des données, nécessaire au bon fonctionnement du système de surveillance intelligente de l'éclairage nocturne.

II.2 Architecture Globale du Système

L'architecture globale de notre système de supervision est basée sur une approche IoT (Internet des objets) structurée en plusieurs blocs fonctionnels, allant de l'acquisition des données sur le terrain jusqu'à la visualisation finale sur l'interface utilisateur.

II.2.1 Synoptique Général du Système

Le schéma synoptique général (Figure II.1) illustre le flux de données au sein du système. Le nœud capteur installé au niveau du lampadaire collecte les données de luminosité et les coordonnées géographiques, puis les transmet via le protocole MQTT vers le serveur Node-RED.

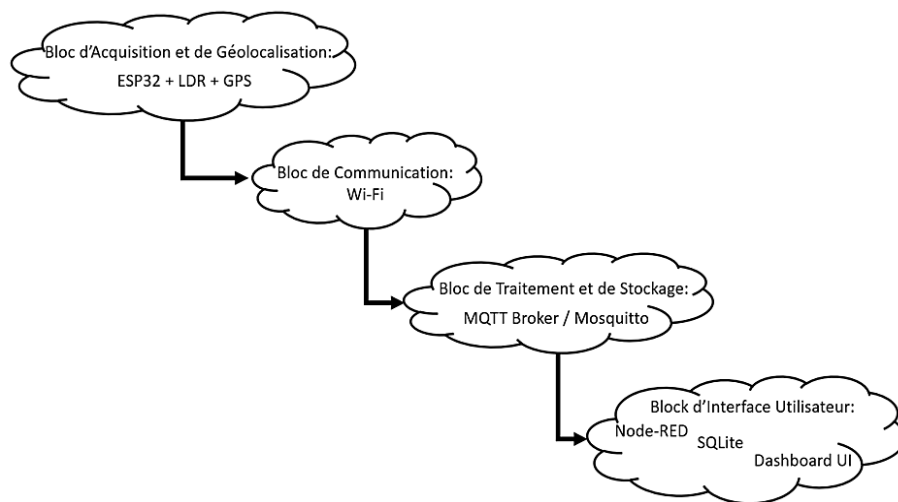


Figure II.1 Schéma synoptique général

Ce dernier traite les informations, les stocke dans la base de données SQLite et met à jour l'interface de supervision en temps réel.

II.2.2 Description des Différents Blocs Fonctionnels

Nous détaillons l'architecture matérielle et logicielle de notre système de monitoring en décomposant le montage en plusieurs sous-ensembles. L'objectif est de présenter le rôle spécifique de chaque bloc fonctionnel, allant de l'acquisition des données physiques par les capteurs jusqu'au traitement et à la transmission des informations par l'unité de contrôle centrale.

II.2.2.1 Bloc d'Acquisition et de Géolocalisation (Nœud Capteur)

Ce bloc est situé sur le terrain (au niveau du candélabre). Il est composé du microcontrôleur ESP32, un capteur LDR pour mesurer l'intensité lumineuse et un module GPS NEO-6M destiné à déterminer la position géographique exacte de la lampe.

II.2.2.2 Bloc de Communication (Passerelle Réseau)

Il assure le transfert des données entre le nœud capteur et le serveur. Il utilise la connectivité Wi-Fi locale et s'appuie sur le protocole de communication léger MQTT, idéal pour les applications IoT, via un broker Mosquitto.

II.2.2.3 Bloc de Traitement et de Stockage (Serveur/Backend)

Le cœur logique central du système. Il comprend la plateforme Node-RED qui orchestre les flux de données, filtre les messages, et gère l'insertion des informations dans la base de données relationnelle SQLite (smart_street.db) pour assurer l'archivage historique.

II.2.2.4 Bloc d'Interface Utilisateur (Dashboard Supervision)

L'interface homme-machine (IHM) développée sous Node-RED Dashboard. Elle permet à l'opérateur de visualiser l'état du réseau en temps réel à travers un tableau de bord comportant plusieurs indicateurs, un historique détaillé des pannes, et un rapport statistique global des performances du système.

II.3 Matériel et acquisition des données

Dans cette section, nous détaillons les choix technologiques et les composants matériels utilisés pour la réalisation de notre nœud de supervision. L'objectif est de mettre en place un système embarqué capable de surveiller l'état de l'éclairage public, de mesurer la luminosité ambiante et de déterminer sa position géographique exacte de manière autonome.

II.3.1 Le Microcontrôleur ESP32

L'ESP32 est bien plus qu'une simple carte de prototypage, c'est un véritable système sur puce (SoC - *System on a Chip*) ultra-intégré, conçu spécifiquement pour répondre aux exigences complexes d'IoT. Équipé nativement de modules de communication Wi-Fi et Bluetooth (incluant le Bluetooth Low Energy - BLE) (voir figure II.2), il se distingue par sa compacité, sa faible consommation énergétique et sa flexibilité d'utilisation.

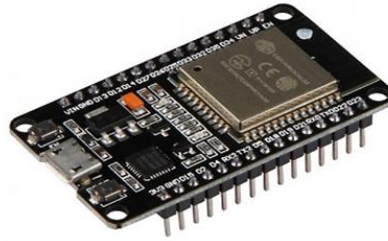
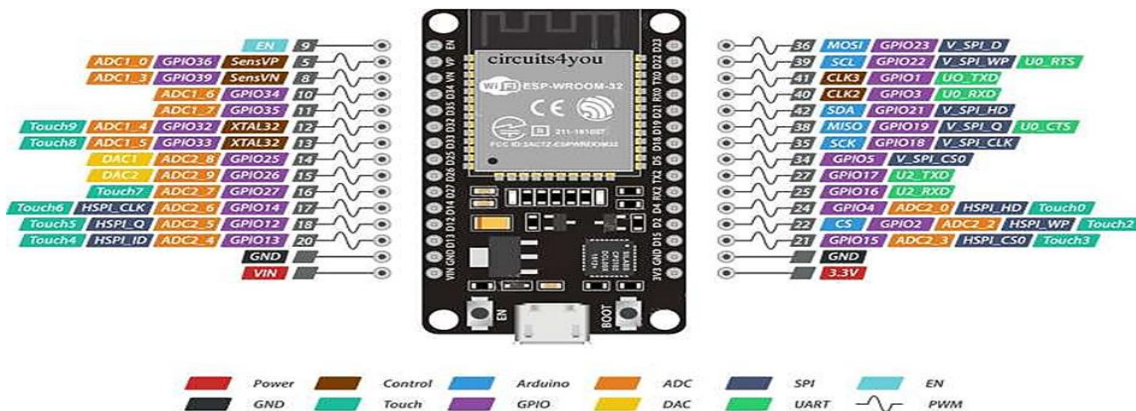


Figure II.2 Module Wifi Bluetooth ESP32

Contrairement aux cartes traditionnelles de type Arduino, l'ESP32 dispose d'une capacité mémoire largement supérieure (SRAM et mémoire Flash) ainsi que d'une puissance de calcul plus élevée, ce qui en fait une plateforme matérielle adaptée aussi bien à l'apprentissage des systèmes embarqués qu'au déploiement de serveurs locaux ou au développement d'objets connectés industriels [37].

L'ESP32 s'impose comme une solution incontournable pour les objets connectés grâce à sa polyvalence réseau et à ses performances matérielles. Architecture d'entrées/sorties (GPIO) polyvalente : Avec 25 broches disponibles (sur 48 au total), l'ESP32 excelle par son haut niveau de multiplexage (figure II.3).



Le microcontrôleur ESP32 représente l'unité de traitement et de communication embarquée au niveau de chaque nœud du réseau. Choisi pour sa puissance de calcul et sa connectivité Wi-Fi intégrée, il est chargé d'acquérir les données des capteurs (état de la lampe et géolocalisation) et de les transmettre de manière autonome vers le serveur central.

Le tableau suivant récapitule ses caractéristiques techniques principales :

Tableau II.1 Les caractéristiques de la carte ESP32

Caractéristique	ESP32
CPU	Xtensa Dual-Core 32-bit LX6 avec 600 DMIPS
Vitesse du Wi-Fi	802.11n jusqu'à 150 Mbps
Protocole Wi-Fi	802.11 b/g/n (2.4 GHz)
GPIO	36
Bluetooth	OUI
CAD (DAC)	Deux canaux DAC 8 bits
ADC	SAR 12 bits (indiqué "RAS" sur l'image)
Canaux ADC	8 canaux
Référence ADC V	1100mV
SPI/I2C/I2S/UART	4/2/2/3
Modes Wi-Fi	Station/SoftAP/SoftAP+Station/P2P
Capteur tactile	OUI (8-Channel)
Capteur de température	OUI
Capteur à effet Hall	OUI
SRAM	520 KB (8 KB de SRAM dans RTC)
FLASH (externe)	4Mbytes (également disponible plus haut)
ROM	448 KB ROM pour le démarrage et les fonctions de base
Protocoles de réseau	IPv4, IPv6, SSL, TCP/UDP/HTTP/FTP/MQTT
Interface périphérique	UART/SDIO/SPI/I2C/I2S/IR Télécommande GPIO/ADC/DAC/Touch/PWM/LED
Plage de température de fonctionnement	-40°C ~ +85°C
Tension de fonctionnement	2,5V ~ 3,6V
Courant de fonctionnement	Moyenne : 80 mA

II.3.2 Capteur LDR

La photorésistance, appelée LDR (*Light Dependent Resistor*), est un composant semi-conducteur passif, généralement à base de sulfure ou de séléniure de cadmium, dont la valeur ohmique fluctue en fonction de l'intensité lumineuse reçue.

Contrairement aux capteurs actifs comme les photopiles au silicium, qui s'avèrent onéreuses et limitées par un faible courant de sortie, la LDR offre une solution optimale grâce à son coût très bas, sa taille compacte de 5 mm et une plage de mesure nettement plus étendue.

Son comportement électrique est régi par la loi de puissance : $R = \frac{A}{\phi^\alpha}$ où les constantes A et α (respectivement de l'ordre de $340 \cdot 10^3$ et 0,85) lient la résistance à l'éclairement lumineux ϕ . Cette polyvalence la rend indispensable dans les systèmes de détection, d'alarme et de comptage, bien qu'une utilisation en extérieur puisse nécessiter l'adjonction d'un filtre type vitre teintée pour stabiliser les mesures [38].

II.3.2.1 Principe de fonctionnement du LDR

Le fonctionnement de la photorésistance repose sur les propriétés physiques des semi-conducteurs, dont la conductivité varie en fonction de l'apport d'énergie externe. À basse température, le cristal se comporte comme un isolant car ses électrons sont piégés dans des liaisons covalentes, mais une augmentation thermique ou, plus spécifiquement, une exposition à une radiation lumineuse modifie cet état.

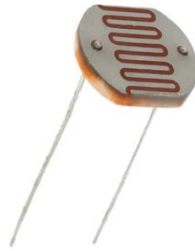


Figure II.4 Capteur photorésistant LDR 7 mm

Lorsque les photons atteignent le matériau, leur énergie peut libérer des électrons de ces liaisons, les transformant en porteurs de charge mobiles. Ce phénomène de création de paires électron-trou augmente la conductivité du cristal proportionnellement à l'intensité du flux lumineux reçu, ce qui explique pourquoi la résistance de la LDR est inversement proportionnelle à l'éclairement.

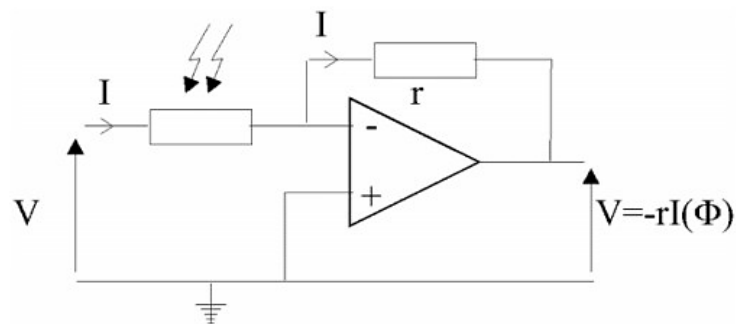


Figure II.5 Schéma équivalent de la LDR

Enfin, cette réactivité n'est pas uniforme, car la sensibilité du composant varie également selon la fréquence ou la longueur d'onde de la lumière incidente [39].

II.3.2.2 Câblage du capteur LDR

Pour évaluer le niveau d'éclairement naturel, nous avons intégré une photorésistance (LDR - Light Dependent Resistor). Montée en diviseur de tension, la résistance de la LDR varie en fonction de la lumière. La tension analogique résultante est lue par le convertisseur analogique-numérique (ADC) de l'ESP32, puis convertie en un pourcentage représentant l'intensité lumineuse de l'environnement.

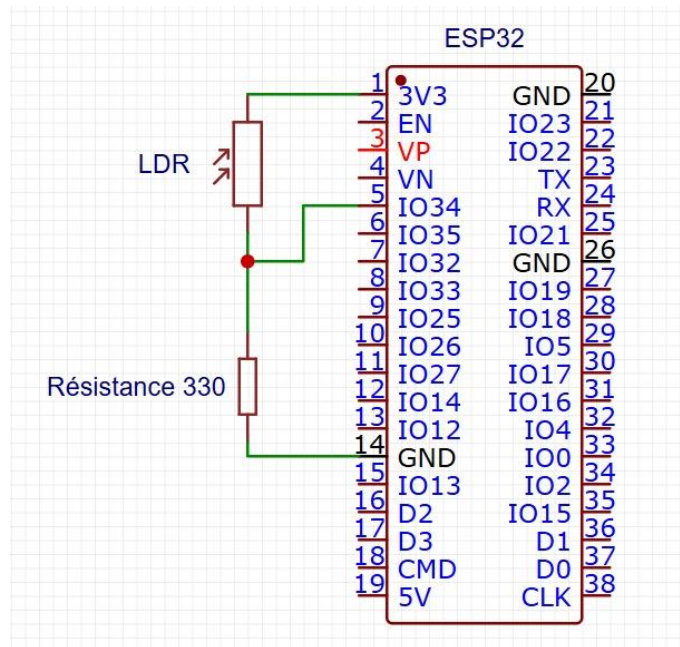


Figure II.6 Schéma de câblage de LDR avec ESP32

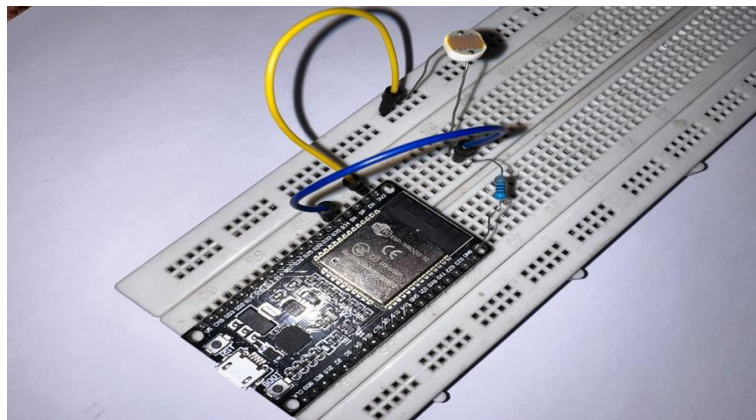


Figure II.7 Montage du capteur lumineux à la base d'ESP32

II.3.3 Module GPS NEO 6M

Le u-blox NEO-6M (Figure II.8) est un récepteur de positionnement par satellite de haute performance, appartenant à la famille des modules de navigation autonome NEO-6 développée par la firme suisse u-blox. Véritable référence dans le domaine de l'électronique embarquée et de IoT, ce module se distingue par une architecture compacte et une intégration poussée, permettant de fournir des données de géolocalisation d'une grande précision tout en conservant une consommation énergétique extrêmement maîtrisée.

Il est spécifiquement conçu pour offrir une sensibilité de réception exceptionnelle, capable de détecter et de traiter les signaux GPS même dans des conditions de réception difficiles, comme les environnements urbains denses ou les zones partiellement couvertes [40].

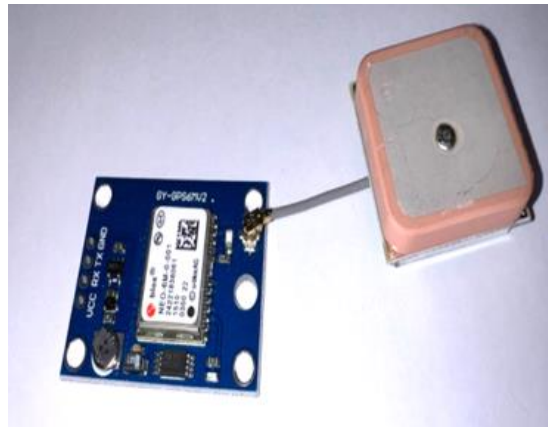


Figure II.8 Module GPS Neo 6m

Au-delà de sa fonction primaire de récepteur, le NEO-6M est une solution matérielle complète qui intègre un moteur de positionnement à 50 canaux, une mémoire EEPROM pour le stockage des données de configuration et une batterie de secours intégrée pour la rétention des informations orbitales.

Cette robustesse technique en fait un composant privilégié pour les applications exigeant une fiabilité constante, telles que le suivi de flottes logistiques, les systèmes de navigation pour drones, la synchronisation temporelle de précision ou encore les stations météorologiques connectées.

Sa conception modulaire facilite son couplage avec diverses antennes céramiques actives, garantissant ainsi une flexibilité d'installation optimale pour les ingénieurs et les concepteurs de systèmes. Économique et largement documenté, il constitue aujourd'hui un standard industriel pour

quiconque souhaite doter un projet électronique de capacités de localisation géographique et temporelle globales [41].

II.3.3.1 Principe de fonctionnement du GPS NEO 6m

Le fonctionnement du module GPS NEO-6M repose sur une communication constante entre le récepteur au sol et la constellation de satellites GPS en orbite.

Le module commence par capter les signaux radiofréquences émis par plusieurs satellites afin de déterminer sa position géographique. Pour obtenir une position tridimensionnelle précise (latitude, longitude et altitude), il exploite généralement les signaux d'au moins quatre satellites.

Il utilise pour cela le principe de la trilatération, calculant la distance qui le sépare de chaque satellite en mesurant le temps mis par le signal pour voyager à la vitesse de la lumière. Grâce à son moteur de positionnement à 50 canaux, il peut suivre simultanément un grand nombre de satellites, ce qui améliore la stabilité du signal et la précision des données, même dans des environnements dits "urbains" où les réflexions sur les bâtiments peuvent perturber la réception [42].

Une fois les calculs internes effectués par le processeur u-blox 6, le module traduit ces coordonnées complexes en un format compréhensible par l'utilisateur : le protocole NMEA 0183. Il transmet alors via ses broches de sortie (TX/RX) une série de lignes de texte appelées "phrases NMEA", telles que \$GPRMC ou \$GPGGA.

Ces phrases contiennent non seulement les coordonnées géographiques, mais aussi l'heure exacte synchronisée sur les horloges atomiques des satellites, la date, la vitesse de déplacement et le nombre de satellites actuellement captés [43].

Pour optimiser ses performances, le NEO-6M utilise plusieurs modes de démarrage. Lors d'un "Cold Start" (démarrage à froid), le module doit télécharger l'almanach complet des satellites, ce qui peut prendre environ 27 secondes. Cependant, grâce à sa batterie de secours intégrée et sa mémoire EEPROM, il peut effectuer un "Hot Start" (démarrage à chaud) en moins d'une seconde, car il conserve en mémoire la dernière position connue et les données orbitales des satellites, évitant ainsi de recalculer l'intégralité des paramètres à chaque mise sous tension [42].

II.3.3.2 Câblage du GPS NEO-6M

Le raccordement du module GPS NEO-6M au microcontrôleur ESP32 s'effectue via une liaison série asynchrone utilisant le protocole UART (Universal Asynchronous Receiver Transmitter). Sur le plan électrique, l'alimentation du module est assurée en connectant sa broche VCC à la broche 3.3 V de l'ESP32, et sa broche GND à la masse commune (GND). Pour la

transmission des données de géolocalisation (trames NMEA), le câblage croisé des broches de communication est nécessaire :

- 1- La broche TX (Transmission) du module GPS est reliée à la broche RX (Réception) de l'ESP32 (configurée sur le port GPIO 16 / RX2).
- 2- La broche RX (Réception) du module GPS est connectée à la broche TX (Transmission) de l'ESP32 (le port GPIO 17 / TX2).

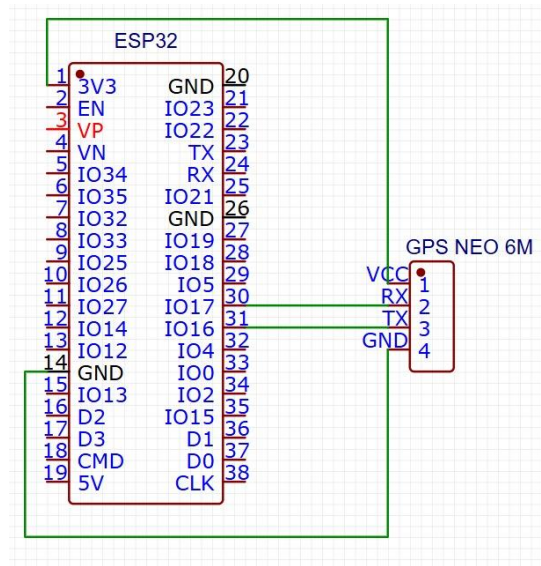


Figure II.9 Schéma de câblage GPS NEO 6M avec ESP32

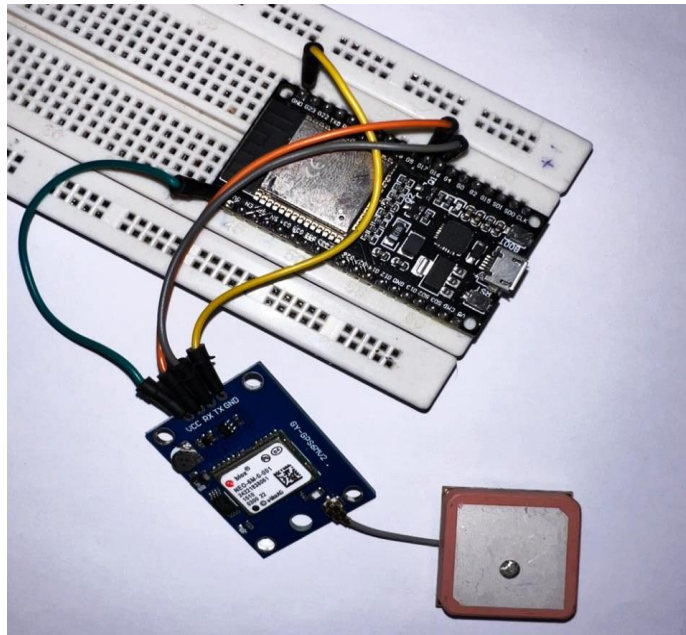


Figure II.10 Montage de GPS NEO 6M avec ESP32

Cette configuration permet au microcontrôleur de recevoir et d'interpréter les informations de position transmises par le module GPS afin de les intégrer au système de supervision.

II.3.4 Raspberry Pi 4 Model B

Le Raspberry Pi 4 Model B (Figure II.11) marque une évolution majeure pour les ordinateurs mono cartes (Single Board Computer – SBC), franchissant le cap du simple outil d'apprentissage pour devenir un véritable mini-ordinateur. Propulsé par un processeur Broadcom BCM2711 quadricœur (Cortex-A72) à 1,5 GHz, il offre une puissance de calcul largement supérieure à ses prédécesseurs. Sa grande nouveauté réside dans sa flexibilité matérielle : il propose pour la première fois plusieurs options de mémoire vive (LPDDR4, de 2 Go à 8 Go). Cela lui permet de s'adapter précisément aux exigences de chaque projet, de la simple domotique à l'hébergement de serveurs gourmands en ressources.

Ses nombreux atouts en font une plateforme incontournable aussi bien pour les passionnés que pour les professionnels. Outre son format ultra-compact, il intègre une connectique moderne avec deux ports USB 3.0 pour des transferts rapides et deux ports USB 2.0. Sa capacité d'affichage est remarquable, supportant le double écran 4K via deux ports micro-HDMI. Sur le plan réseau, il s'enrichit d'un véritable Gigabit Ethernet, d'un Wi-Fi double bande et du Bluetooth 5.0 pour une communication optimale. Alimenté désormais par USB-C pour soutenir ces nouveaux composants, il reste fidèle à l'esprit "DIY" avec ses 40 broches GPIO. Il réussit ainsi le pari de rivaliser avec des PC d'entrée de gamme, tout en conservant un prix très abordable et le soutien d'une immense communauté [44].

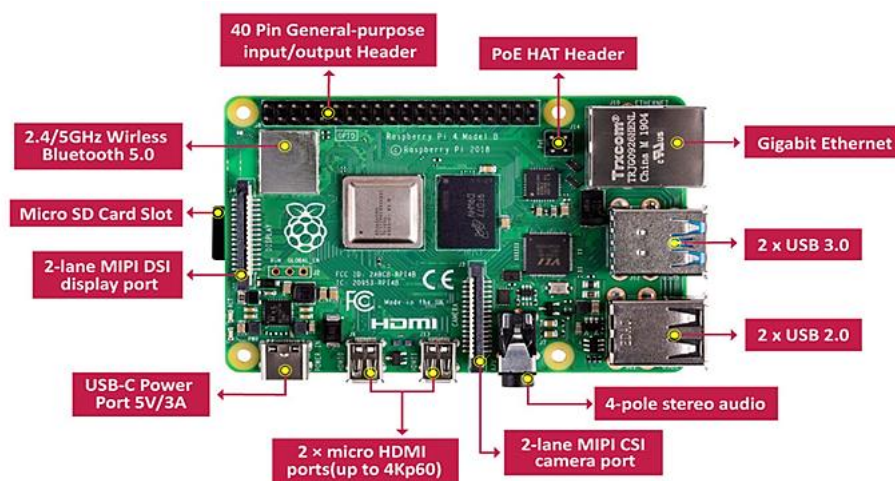


Figure II.11 Raspberry pi 4 Model B

Le Raspberry Pi 4 Model B constitue le cœur de notre système de supervision en jouant le rôle de serveur central (Broker MQTT) et d'unité de traitement. Grâce à ses performances matérielles robustes et à sa connectivité étendue, il assure la réception continue des données transmises par les nœuds ESP32 et héberge l'interface de supervision développée sous Node-RED. Le tableau ci-dessous résume ses principales caractéristiques techniques :

Tableau II.2 Les caractéristiques de Raspberry Pi 4 Model B

Composant / Caractéristique	Spécifications Détails
Processeur (CPU)	Broadcom BCM2711, Quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
Mémoire Vive (RAM)	2Go, 4 Go ou 8 Go LPDDR4-3200 SDRAM (selon le model)
Connectivité Sans Fil	Wi-Fi double bande 2.4 GHz et 5.0 GHz (IEEE 802.11ac)
Bluetooth	Bluetooth 5.0, BLE
Réseau Filaire	Gigabit Ethernet (débit réel non bridé)
Ports USB	2 ports USB 3.0 + 2 ports USB 2.0
GPIO	En-tête standard à 40 broches (entièrement rétrocompatible)
Vidéo et Audio	2 ports micro-HDMI (support jusqu'à 4K @ 60fps)
Multimédia	H.265 (décodage 4Kp60), H264 (décodage 1080p60, encodage 1080p30)
Graphismes	OpenGL ES 3.1, Vulkan 1.0
Stockage	Port pour carte Micro-SD (chargement de l'OS et stockage)
Affichage et Caméra	Port MIPI DSI (écran) et Port MIPI CSI (caméra)
Sortie Audio/Vidéo	Prise jack 3,5 mm (audio stéréo et vidéo composite)
Alimentation	5 V DC via connecteur USB-C (minimum 3 A) ou via GPIO
Température de service	0°C à 50°C (température ambiante)

II.3.5 Couche réseau et protocole de communication MQTT

Dans une architecture IoT urbaine dédiée aux villes intelligentes, la robustesse de la couche réseau est aussi critique que la précision des capteurs. Cette section détaille l'ingénierie des communications de notre système, les protocoles choisis pour optimiser la bande passante, ainsi

que l'analyse approfondie des défis techniques rencontrés et des solutions algorithmiques développées pour garantir une résilience à toute épreuve.

II.3.5.1 Le Protocole MQTT

Contrairement aux architectures traditionnelles basées sur le protocole HTTP, qui reposent généralement sur un modèle de communication de type requête/réponse, nous avons opté pour le protocole MQTT (Message Queuing Telemetry Transport).

Ce protocole léger et asynchrone est spécifiquement conçu pour les environnements où les ressources réseau sont limitées et où une faible latence est recherchée. MQTT est particulièrement adapté aux systèmes IoT grâce à son faible surcoût de communication et à son efficacité dans l'échange de messages entre équipements connectés.

L'architecture MQTT repose sur le découplage total entre l'émetteur (le nœud ESP32) et le récepteur (le serveur Node-RED), géré par un nœud central appelé "Broker". Ce découplage permet au microcontrôleur de publier ses données et de retourner immédiatement à son cycle de traitement sans attendre de réponse de la plateforme de supervision [45].

II.3.5.2 Présentation du protocole MQTT

Créé en 1999, le protocole MQTT (Message Queuing Telemetry Transport) est un protocole de communication Machine-to-Machine (M2M) basé sur le modèle publication/abonnement. Conçu pour être léger, économe en bande passante et capable d'assurer différents niveaux de qualité de service (Quality of Service – QoS), il s'est imposé comme l'un des protocoles les plus utilisés dans le domaine de l'Internet des objets.

Son évolution a abouti à sa standardisation internationale en 2016 sous la norme ISO/IEC 20922 [46].

Techniquement, MQTT définit un format d'échange binaire optimisé entre des dispositifs connectés (clients) et un serveur central (Broker). Cette architecture assure une parfaite interopérabilité, même pour des équipements aux ressources matérielles limitées.

Face à l'essor massif de l'IoT, ce protocole est devenu un standard industriel incontournable, adopté par des géants technologiques tels que Facebook, pour optimiser sa messagerie, et Amazon Web Services (AWS), pour la diffusion fluide et ciblée de flux de données [47].

II.3.5.3 Principe de fonctionnement du protocole MQTT

Le fonctionnement du protocole MQTT repose fondamentalement sur une architecture de publication et de souscription (Publish/Subscribe) qui découple totalement les dispositifs émetteurs des récepteurs. Contrairement aux modèles de communication traditionnels, les clients (capteurs, interfaces) ne communiquent jamais directement entre eux, ils interagissent exclusivement via un serveur central appelé broker. Ce dernier agit comme intermédiaire en réceptionnant les messages puis en les redistribuant aux destinataires concernés [48].

Ce mécanisme garantit une communication asynchrone et une grande flexibilité temporelle : le broker est capable de mémoriser le dernier message d'un flux pour le transmettre aux clients dès leur connexion, ce qui rend cette architecture parfaitement adaptée aux environnements IoT où la connectivité peut être intermittente [49].

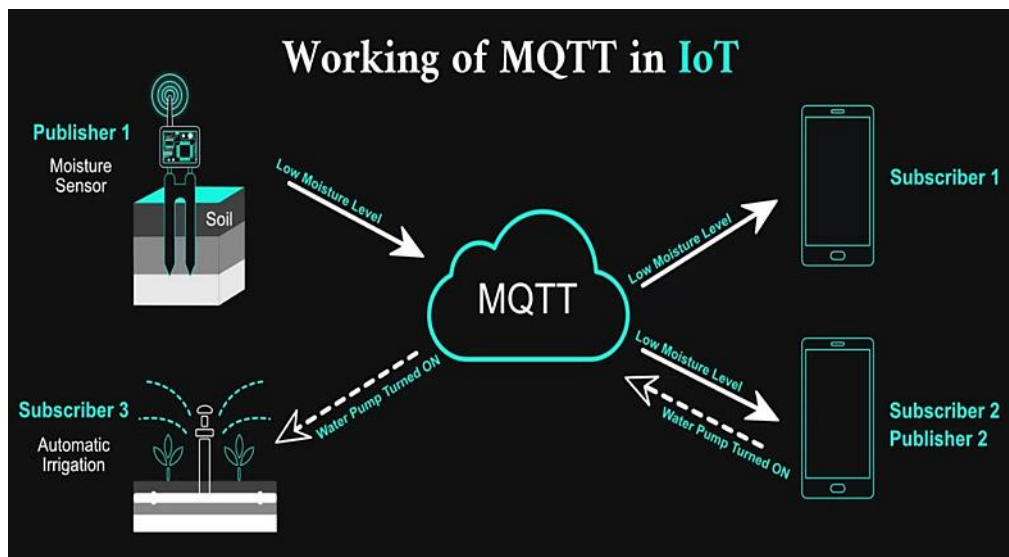


Figure II.12 Architecture et fonctionnement du protocole MQTT dans un système IoT

L'organisation et le routage de ces flux de données s'articulent autour de canaux thématiques hiérarchisés appelés *topics* (sujets). Représentés sous forme arborescente (par exemple : /eclairage/zone1/etat), ces topics permettent une distribution granulaire de l'information. La dynamique du réseau est régie par trois commandes essentielles : PUBLISH (émission d'un message avec son topic de destination vers le broker), SUBSCRIBE (abonnement d'un client auprès du broker pour recevoir les futures publications d'un topic) et UNSUBSCRIBE (désengagement d'un flux). Pour faciliter la supervision d'un vaste parc de dispositifs sans multiplier les requêtes individuelles, le protocole autorise des abonnements groupés via des caractères génériques (*wildcards*). Le symbole multi-niveaux (#) permet de capter l'intégralité des sous-niveaux d'une

arborescence (ex. : /eclairage/zone1/#), tandis que le symbole mono-niveau (+) permet de regrouper un paramètre spécifique sur l'ensemble du réseau (ex. : /eclairage/+ /etat) [50].

Pour s'adapter aux contraintes réseau et à l'importance des données transmises, MQTT propose trois niveaux de Qualité de Service (Quality of Service – QoS) :

QoS 0 (At most once) : le message est transmis une seule fois sans confirmation de réception.

QoS 1 (At least once) : le message est retransmis jusqu'à réception d'un accusé (PUBACK), avec possibilité de doublons.

QoS 2 (Exactly once) : le message est délivré une seule fois grâce à un mécanisme d'échange renforcé garantissant l'absence de duplication [51].

L'intégrité de ce fonctionnement repose sur une sécurité déployée à travers trois piliers : l'identification (nommage unique de chaque entité), l'authentification (vérification de l'identité) et l'autorisation (droits d'accès aux topics gérés par le broker). L'authentification n'étant pas cryptée nativement, l'encapsulation du flux MQTT dans un tunnel sécurisé de type SSH (utilisant des clés publiques/privées ou des mots de passe) est couramment employée pour garantir la confidentialité absolue des échanges, une méthode particulièrement robuste et recommandée sur des systèmes embarqués basés sur Linux, tels que le Raspberry Pi [52].

II.4 Traitement des données et gestion de la base de données

La réussite d'un système de supervision IoT ne repose pas uniquement sur la fiabilité des nœuds capteurs, mais également sur l'efficacité de l'infrastructure logicielle (Backend) chargée de la collecte, du traitement, de l'orchestration et du stockage des données. Dans cette section, nous détaillons l'architecture serveur mise en place, s'articulant autour de Node-RED pour le flux logique et de SQLite pour la persistance relationnelle.

II.4.1 Plateforme d'Orchestration : Node-RED

Node-RED est un outil de développement open source, créé par IBM en 2013, basé sur une approche de programmation visuelle inspirée du paradigme Flow-Based Programming (FBP) proposé par Paul J. Morrison. Il permet de connecter des périphériques matériels, des API (Application Programming Interface) ainsi que des services en ligne dans le cadre des applications de l'Internet des objets [53].

Node-Red utilise le JavaScript qu'est un langage de programmation de scripts principalement employé dans les pages web interactives et à ce titre est une partie essentielle des applications web. Avec les langages HTML et CSS [54].

Il est imposé comme une plateforme incontournable de l'IoT et de l'automatisation grâce à son approche innovante :

Développement orienté flux (Low-Code) : Son interface visuelle permet de modéliser des logiques complexes par simple « glisser-déposer » de nœuds. Cette méthode simplifie le développement en se concentrant sur l'architecture fonctionnelle plutôt que sur la syntaxe complexe du code [55].

Interopérabilité et extensibilité : Sa force réside dans la convergence de protocoles variés (MQTT, HTTP, WebSockets). Grâce à une immense bibliothèque de nœuds communautaires, l'intégration de matériels divers (ESP32, Raspberry Pi) et de services cloud devient fluide[56].

Agilité et prototypage rapide : La plateforme favorise un cycle de développement itératif. Les flux peuvent être modifiés et déployés en temps réel, garantissant une grande réactivité lors des phases d'expérimentation et d'optimisation [57].

Efficience et portabilité : Basé sur Node.js, Node-RED est léger et performant. Son architecture non bloquante lui permet de fonctionner aussi bien sur du matériel aux ressources limitées que sur des serveurs cloud robustes [58].

Interface de supervision (Dashboard) : Il propose des outils intégrés pour créer rapidement des tableaux de bord graphiques. Cette fonctionnalité offre une visualisation immédiate des données et permet un pilotage intuitif des équipements à distance [59].

Node-RED représente une solution d'orchestration robuste qui concilie simplicité d'usage et puissance technique, répondant ainsi aux exigences de flexibilité et de rapidité d'exécution du secteur technologique actuel.

II.4.2 Base de données SQLite

Dans un système de supervision IoT, le stockage des données constitue une étape essentielle afin d'assurer la conservation des mesures collectées ainsi que leur exploitation ultérieure. Dans notre architecture, nous avons utilisé SQLite comme solution de persistance pour les données générées par le système de surveillance [60].

II.4.2.1 Présentation de SQLite

SQLite est un système de gestion de base de données relationnelle (SGBDR) léger, open source et autonome. Contrairement aux solutions client-serveur traditionnelles, SQLite fonctionne sous forme d'une bibliothèque logicielle intégrée directement à l'application, sans nécessiter de serveur de base de données dédié [61].

Cette base de données est particulièrement adaptée aux systèmes embarqués et aux applications IoT grâce à sa faible consommation de ressources, sa simplicité d'intégration et ses bonnes performances d'exécution. Les données sont stockées dans un fichier unique, ce qui facilite leur sauvegarde, leur transfert ainsi que leur maintenance [62].

II.4.2.2 Rôle de SQLite dans le système proposé

Dans notre projet, SQLite est utilisée comme couche de stockage locale afin d'enregistrer les données collectées puis traitées par la plateforme Node-RED [63].

Plus précisément, cette base de données permet de :

- Stocker les informations relatives à l'état de fonctionnement des lampadaires.
- Conserver l'historique des données collectées.
- Faciliter l'affichage des informations sur l'interface de supervision.
- Assurer la disponibilité des données pour le suivi et l'analyse du système.

L'intégration de SQLite au sein de l'architecture proposée contribue ainsi à améliorer l'organisation et la gestion des données tout en garantissant une solution légère, adaptée aux contraintes d'un système IoT embarqué [64].

II.5 Conclusion

Ce chapitre a permis de présenter l'architecture matérielle et réseau du système de supervision IoT proposé. Les principaux composants utilisés, notamment l'ESP32, le capteur LDR, le module GPS NEO-6M et le Raspberry Pi 4, ont été décrits afin de justifier leur rôle dans l'acquisition, la transmission et le traitement des données.

L'intégration du protocole MQTT et de la plateforme Node-RED a également permis de mettre en place une architecture de communication légère, fiable et adaptée aux exigences du système de surveillance intelligente de l'éclairage public.

Ainsi, ce chapitre a posé les fondements techniques nécessaires à la réalisation du prototype. Le chapitre suivant sera consacré à l'implémentation de la chaîne IoT, au développement des différentes interfaces de supervision ainsi qu'à la validation fonctionnelle du système proposé.

Chapitre III : Implémentation et Développement de la chaîne IoT

III.1 Introduction

Ce chapitre constitue le cœur opérationnel de ce mémoire. Après avoir établi, dans les chapitres précédents, les fondements théoriques des villes intelligentes et justifié l'architecture matérielle et protocolaire de la solution proposée, nous abordons à présent la phase de concrétisation technique : l'implémentation effective du système. Ce chapitre poursuit trois objectifs principaux. Tout d'abord, nous présentons la topologie réelle du système déployé, en décrivant l'ensemble des composants matériels mis en œuvre ainsi que leurs interactions au sein de l'architecture IoT retenue. En second lieu, nous détaillons le développement et la configuration du système, en exposant la logique algorithmique embarquée sur les nœuds capteurs (ESP32), la configuration du broker de communication (Mosquitto) et l'ingénierie des flux de données sur la plateforme de supervision (Node-RED). En dernier lieu, nous présentons le développement de l'interface de supervision permettant la visualisation et l'exploitation des données collectées. À travers cette démarche d'ingénierie intégrée, ce chapitre vise à démontrer la mise en œuvre concrète et la cohérence technique de notre prototype dans le contexte des infrastructures urbaines intelligentes.

III.2 Présentation du prototype développé

Le prototype réalisé consiste en un système IoT de supervision de l'éclairage public basé sur une architecture locale centralisée autour d'un Raspberry Pi. Le système permet d'identifier les lampes, d'enregistrer leurs coordonnées géographiques de surveiller leur état lumineux et d'afficher les informations sur des interfaces développées sous Node-RED. La solution repose sur deux parties complémentaires. La première concerne l'enregistrement initial des positions à l'aide d'une unité mobile composée d'un ESP32 et d'un module GPS NEO-6M. La seconde concerne la supervision continue des lampes à l'aide de nœuds ESP32 associés à des capteurs LDR. Les données sont transmises par MQTT vers le broker Mosquitto installé sur Raspberry Pi, puis traitées sous Node-RED et stockées dans une base de données SQLite.

III.2.1 Environnement matériel et logiciel

La réalisation du prototype repose sur un ensemble de composants matériels et d'outils logiciels permettant l'acquisition, la transmission, le traitement, le stockage et la visualisation des données. La partie matérielle est basée sur des microcontrôleurs ESP32, un capteur LDR pour la détection de l'état lumineux, un module GPS NEO-6M pour l'enregistrement initial des coordonnées géographiques et un Raspberry Pi 4 utilisé comme serveur local.

La partie logicielle comprend l'environnement Arduino IDE pour la programmation des ESP32, le broker Mosquitto pour la communication MQTT, Node-RED pour le traitement des flux et la conception des interfaces de supervision, ainsi que SQLite pour le stockage local des données et des historiques.

Le tableau III.1 présente la liste des matériels utilisés dans la conception du système, tandis que la figure III.1 illustre le système réel développé ainsi que son organisation matérielle.

Tableau III.1 Environnement matériel et logiciel utilisé

Élément	Rôle dans le prototype
ESP32	Acquisition des données et publication MQTT
Capteur LDR	Détection du niveau lumineux de la lampe
GPS NEO-6M	Enregistrement initial des coordonnées GPS
Raspberry Pi 4	Serveur local du système
Mosquitto MQTT	Broker de communication entre ESP32 et Node-RED
Node-RED	Traitement des données et conception des interfaces de supervision
SQLite	Stockage des positions et historiques

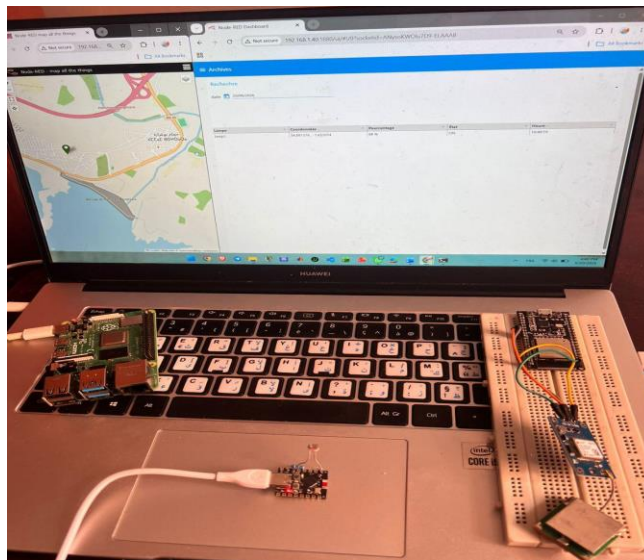


Figure III.1 Les composants matériels utilisés dans le prototype

III.2.2 Architecture réelle du système

L'architecture réelle du prototype est organisée autour d'une plateforme centrale basée sur un Raspberry Pi 4. Ce dernier héberge le broker MQTT Mosquitto, la plateforme Node-RED et la base de données SQLite. Il assure ainsi la réception des messages, le traitement des flux, le stockage des données et l'affichage des interfaces de supervision.

Le système comporte deux types d'unités embarquées. La première est une unité mobile ESP32 associée au module GPS NEO-6M, utilisée uniquement pendant la phase d'initialisation afin d'enregistrer les coordonnées géographiques des lampes. La seconde est constituée de nœuds ESP32 équipés de capteurs LDR, installés au niveau des lampes pour mesurer le niveau lumineux et transmettre l'état ON/OFF via MQTT.

Les données de localisation sont publiées sur le topic `street/lamp`, tandis que les états lumineux sont publiés sur les topics `street/lampX`. Node-RED associe ensuite chaque identifiant de lampe aux coordonnées enregistrées dans SQLite, puis affiche l'état correspondant sur la carte interactive et les tableaux de supervision (figure III.2).

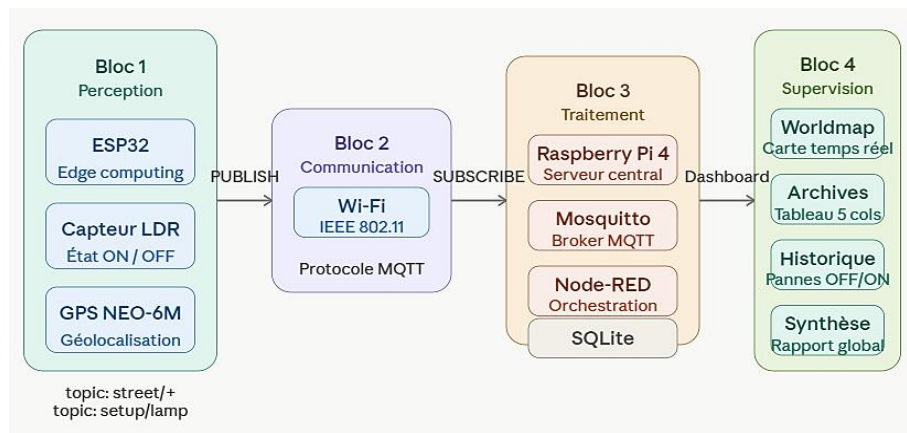


Figure III.2 Architecture fonctionnelle du système de supervision IoT

La figure III.2 illustre l'architecture fonctionnelle globale du système, organisée en quatre blocs fonctionnels complémentaires.

III.2.3 Scénario opérationnel du système proposé

Le fonctionnement global du prototype se déroule en deux phases complémentaires. La première phase concerne l'enregistrement initial des coordonnées GPS des lampes. Une unité mobile composée d'un ESP32 et d'un module GPS NEO-6M est placée successivement à proximité de chaque lampe afin d'acquérir sa position géographique. Les coordonnées obtenues sont associées à un identifiant unique, puis envoyées vers le Raspberry Pi via MQTT sur le topic `setup/lamp`, avant d'être stockées dans la table `lamps_location` de la base SQLite (figure III.3).

La seconde phase correspond à la supervision continue de l'état lumineux. Chaque nœud ESP32-LDR mesure le niveau de luminosité de la lampe, détermine son état ON/OFF, puis publie les données sur le topic `street/lampX`. Node-RED reçoit les messages MQTT, récupère les coordonnées correspondantes à partir de l'identifiant de la lampe, stocke l'historique dans SQLite

et met à jour les interfaces de supervision : Worldmap, Archives, Historique des pannes et Synthèse globale (figure III.3).

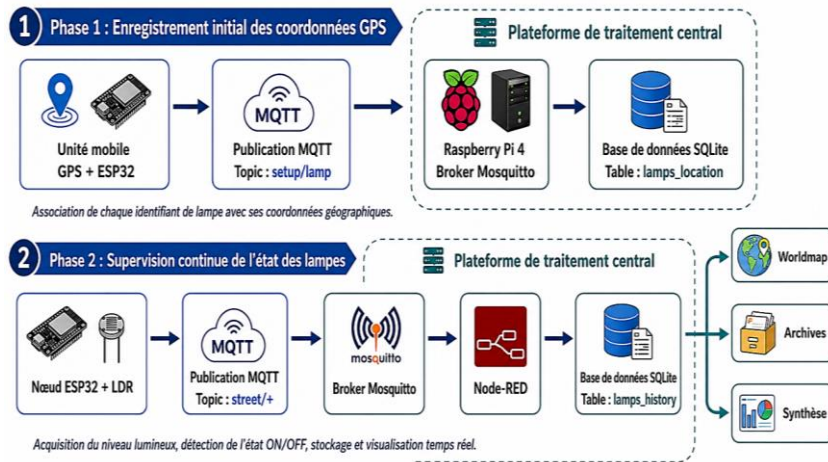


Figure III.3 Scénario global de fonctionnement du prototype de supervision IoT

Ce scénario permet donc d'assurer la continuité entre l'acquisition locale, la transmission MQTT, le traitement centralisé, le stockage et la visualisation des données.

La figure ci-dessus présente la séparation entre la phase d'enregistrement initial des coordonnées GPS et la phase de supervision continue de l'état lumineux des lampes.

III.3 Implémentation et configuration de la chaîne IoT

Après la définition de l'architecture matérielle et réseau du prototype, cette section présente l'implémentation pratique de la chaîne IoT adoptée pour la supervision intelligente de l'éclairage public. Cette chaîne assure la continuité entre l'acquisition locale des données par les nœuds ESP32, la publication des messages via le protocole MQTT, la réception par le broker Mosquitto installé sur Raspberry Pi, le traitement des flux sous Node-RED, le stockage dans la base de données SQLite et la visualisation des états à travers les interfaces de supervision.

Les sous-sections suivantes détaillent séparément chaque bloc fonctionnel de cette chaîne, depuis l'enregistrement initial des coordonnées GPS jusqu'à l'affichage cartographique, l'archivage et l'analyse statistique des lampes.

III.3.1 Phase de géolocalisation des points d'éclairage

La première étape de l'implémentation consiste à enregistrer les coordonnées géographiques des lampes à superviser. Dans le prototype réalisé, le module GPS NEO-6M n'est pas intégré de manière permanente dans chaque nœud lumineux. Il est utilisé comme une unité

mobile d'initialisation, associée à un microcontrôleur ESP32, afin de relever successivement la position de chaque lampe et de l'envoyer vers la plateforme centrale.

Cette approche permet de réduire la complexité matérielle du système, car un seul module GPS est utilisé pour localiser plusieurs points lumineux. Lors de la phase d'enregistrement, l'unité GPS-ESP32 est placée à proximité de la première lampe. Après la réception d'une position GPS valide, le microcontrôleur associe les coordonnées obtenues à un identifiant logique, par exemple lamp1. Ensuite, les données sont publiées via le réseau Wi-Fi local vers le Raspberry Pi en utilisant le protocole MQTT (figure III.4).

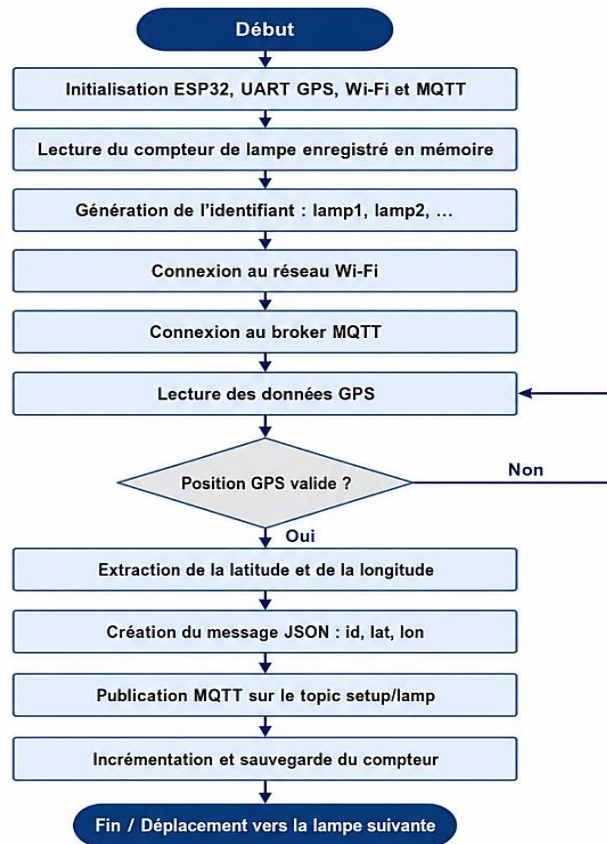


Figure III.4 Organigramme du processus d'enregistrement des coordonnées GPS des points d'éclairage

Le message transmis contient principalement trois informations : l'identifiant de la lampe, la latitude et la longitude. La structure logique du message peut être représentée sous la forme suivante :

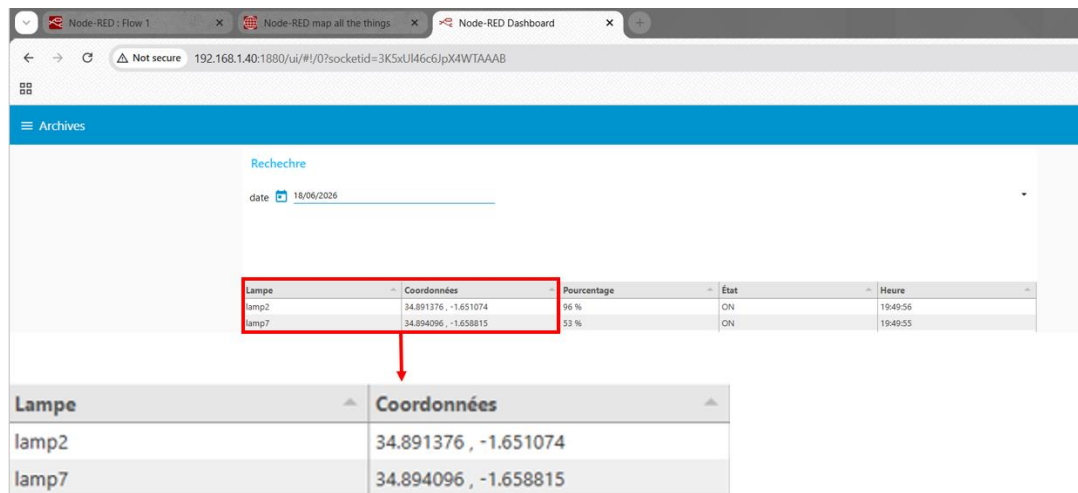
```

</> JSON

{
  "id": "lamp1",
  "lat": 34.xxxxxx,
  "lon": -1.xxxxxx
}

```

Comme le montre l'organigramme, l'ESP32 commence par initialiser les interfaces nécessaires, notamment la liaison série utilisée pour la réception des trames GPS, la connexion Wi-Fi et le client MQTT. Après l'établissement de la communication avec le broker, le programme attend l'obtention d'une position GPS valide. Lorsque les coordonnées sont disponibles, elles sont associées à l'identifiant courant de la lampe, puis envoyées sous forme de message JSON vers le topic `setup/lamp`. À la fin de l'opération, l'identifiant est incrémenté afin de préparer l'enregistrement de la lampe suivante.



Lampe	Coordonnées	Pourcentage	État	Heure
lamp2	34.891376, -1.651074	96 %	ON	19:49:56
lamp7	34.894096, -1.658815	53 %	ON	19:49:55

Lampe	Coordonnées
lamp2	34.891376, -1.651074
lamp7	34.894096, -1.658815

Figure III.5 Table des lampes avec leurs coordonnées GPS.

Après l'enregistrement de la première lampe, l'unité GPS-ESP32 peut être déplacée vers une autre position afin d'enregistrer la lampe suivante. Le même processus est répété pour lamp2, lamp3 et les autres points lumineux du prototype. À la fin de cette phase, la base de données contient une table de correspondance entre chaque identifiant de lampe et ses coordonnées géographiques.

Cette phase d'enregistrement constitue une étape essentielle pour l'affichage cartographique. En effet, lors de la supervision normale, les nœuds LDR n'ont plus besoin d'envoyer leurs coordonnées. Ils transmettent uniquement l'identifiant de la lampe, son état et le pourcentage de luminosité mesuré. La plateforme Node-RED utilise ensuite l'identifiant reçu pour

rechercher les coordonnées correspondantes dans la table `lamps_location`, ce qui permet d'afficher automatiquement la lampe sur la carte interactive avec sa position exacte.

III.3.2 Acquisition des données lumineuses via ESP32 et LDR

L'acquisition de l'état lumineux constitue la fonction principale du nœud de supervision installé au niveau de chaque lampe. Dans cette partie du prototype, le microcontrôleur ESP32 est associé à une photorésistance LDR afin de mesurer le niveau de luminosité produit par la lampe surveillée. Le capteur LDR est monté sous forme de diviseur de tension, ce qui permet de générer une tension analogique variable en fonction de l'intensité lumineuse reçue.

Le signal analogique issu du diviseur de tension est appliqué à une entrée analogique de l'ESP32. Le convertisseur analogique-numérique intégré au microcontrôleur transforme cette tension en une valeur numérique exploitable par le programme embarqué. Cette valeur brute est ensuite convertie en pourcentage afin de faciliter l'interprétation de l'état lumineux. Ainsi, une valeur élevée indique la présence d'un éclairage suffisant, tandis qu'une valeur faible traduit une extinction ou une défaillance potentielle de la lampe.

Dans le programme, chaque nœud est associé à un identifiant unique, par exemple `lamp1`, `lamp2` ou `lampX`. Cet identifiant permet de relier la mesure locale à la lampe correspondante dans la plateforme centrale. Après l'acquisition de la valeur lumineuse, l'ESP32 compare le pourcentage mesuré avec un seuil prédéfini. Si le pourcentage est supérieur au seuil, l'état de la lampe est considéré comme ON. Dans le cas contraire, la lampe est déclarée OFF (figure III.6).

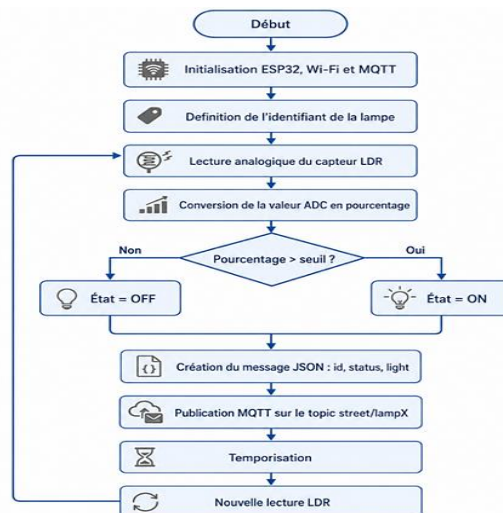


Figure III.6 Organigramme de l'algorithme d'acquisition de l'état lumineux à l'aide de l'ESP32 et du capteur LDR

Les données préparées par l'ESP32 sont ensuite structurées sous forme d'un message JSON contenant l'identifiant de la lampe, son état logique et le pourcentage de luminosité mesuré. La structure utilisée peut être représentée comme suit :

```
<> JSON
{
  "id": "lamp2",
  "status": "ON",
  "light": 12
}
```

Cette étape permet de transformer une grandeur physique analogique, représentée par le niveau lumineux détecté par le LDR, en une information numérique directement exploitable par le système de supervision. L'état obtenu sera ensuite publié vers le broker MQTT afin d'être traité par Node-RED, stocké dans SQLite et affiché sur les interfaces de supervision.

III.3.3 Publication des données du nœud ESP32 vers le broker MQTT

Après l'acquisition des données par les nœuds ESP32, les informations sont publiées vers la plateforme centrale à travers le protocole MQTT. Chaque ESP32 joue le rôle d'un publisher qui envoie périodiquement un message structuré au broker Mosquitto installé sur le Raspberry Pi. Ce choix permet d'assurer une transmission légère, asynchrone et adaptée aux contraintes des systèmes IoT.

Deux types de publication sont utilisés dans le prototype. Le topic setup/lamp est réservé à l'enregistrement initial des coordonnées GPS des lampes. Il transporte l'identifiant de la lampe ainsi que ses coordonnées géographiques. Le topic street/lampX est utilisé pendant la supervision normale pour transmettre l'état lumineux détecté par le LDR. Le message publié contient l'identifiant de la lampe, son état ON/OFF et le pourcentage de luminosité mesuré.

Cette organisation par topics permet de séparer clairement les données de configuration GPS et les données de supervision. Elle facilite également le traitement côté Node-RED, car chaque message reçu peut être orienté automatiquement vers le flux correspondant selon son topic MQTT.

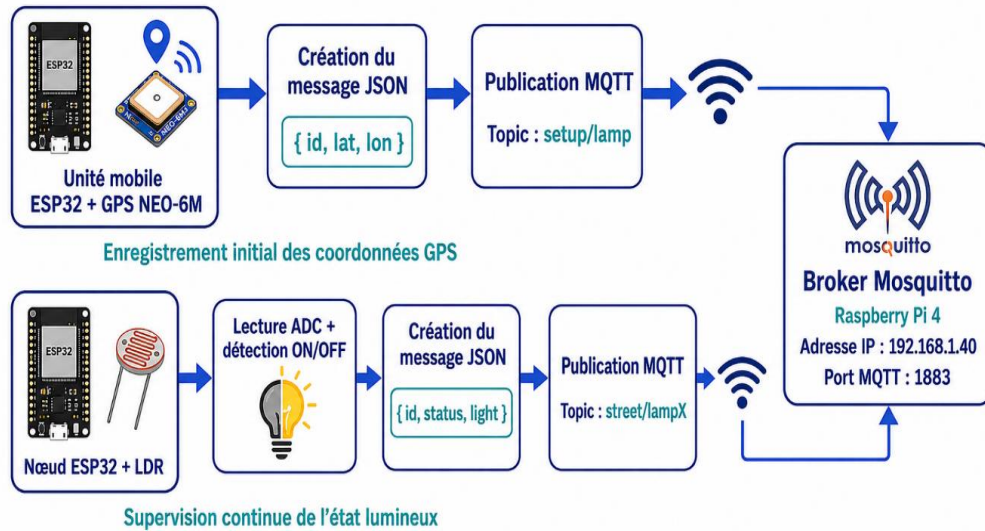


Figure III.7 Illustration du processus de publication des données des nœuds ESP32 vers le broker MQTT

III.3.4 Configuration du broker MQTT Mosquitto sur Raspberry Pi

Le Raspberry Pi 4 est utilisé comme serveur local pour assurer la réception et la distribution des messages MQTT. Le broker Mosquitto est installé afin de jouer le rôle d'intermédiaire entre les nœuds ESP32 publishers et les flux de traitement développés sous Node-RED. Dans cette configuration, les ESP32 publient leurs données vers l'adresse IP du Raspberry Pi, tandis que Node-RED s'abonne aux topics correspondants afin de récupérer les informations transmises.

```
Linux yassine 6.12.47+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.12.47-1+rpt1 (2025-09-16) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat May 16 22:53:02 2026 from 192.168.1.14
yassine12@yassine:~$ sudo systemctl status mosquitto
● mosquitto.service - Mosquitto MQTT Broker
   Loaded: loaded (/usr/lib/systemd/system/mosquitto.service; enabled; preset: enabled)
   Active: active (running) since Sun 2026-05-31 00:47:20 CET; 2 days ago
     Invocation: 14718f4e76964b398a206e1f4876897a
       Docs: man:mosquitto.conf(5)
            man:mosquitto(8)
   Process: 797 ExecStartPre=/bin/mkdir -m 740 -p /var/log/mosquitto (code=exited, status=0/SUCCESS)
   Process: 802 ExecStartPre=/bin/chown mosquitto:mosquitto /var/log/mosquitto (code=exited, status=0/SUCCESS)
   Process: 805 ExecStartPre=/bin/mkdir -m 740 -p /run/mosquitto (code=exited, status=0/SUCCESS)
   Process: 807 ExecStartPre=/bin/chown mosquitto:mosquitto /run/mosquitto (code=exited, status=0/SUCCESS)
  Main PID: 809 (mosquitto)
    Tasks: 1 (limit: 4458)
      CPU: 108ms
    CGroup: /system.slice/mosquitto.service
            └─809 /usr/sbin/mosquitto -c /etc/mosquitto/mosquitto.conf

May 31 00:47:19 yassine systemd[1]: Starting mosquitto.service - Mosquitto MQTT Broker...
May 31 00:47:20 yassine systemd[1]: Started mosquitto.service - Mosquitto MQTT Broker.
```

Figure III.8 Vérification de l'état du service Mosquitto sur Raspberry Pi

Le broker fonctionne sur le port MQTT standard 1883. Deux catégories de topics sont exploitées : setup/lamp pour l'enregistrement initial des coordonnées GPS, et street/+ pour la supervision continue de l'état lumineux des lampes. L'utilisation du caractère générique + permet à Node-RED de recevoir les messages provenant de plusieurs lampes sans créer un abonnement séparé pour chaque identifiant.

La figure III.8 montre que le service Mosquitto est correctement installé, activé et en cours d'exécution sur le Raspberry Pi. Cet état confirme que le broker MQTT est prêt à recevoir les publications des nœuds ESP32 et à les transmettre aux flux Node-RED abonnés aux topics du système.

Cette configuration centralisée simplifie l'architecture de communication du prototype. Elle permet au Raspberry Pi de regrouper les fonctions de broker MQTT, de traitement Node-RED et de stockage SQLite dans une seule plateforme locale adaptée à une application IoT de supervision urbaine.

III.3.5 Traitement des données IoT via Node-RED

Les messages MQTT provenant des nœuds ESP32 sont reçus par Node-RED, convertis au format JSON, puis oriente chaque flux selon sa fonction. Les messages publiés sur le topic setup/lamp sont utilisés pour l'enregistrement des coordonnées GPS des lampes, tandis que les messages reçus sur le topic street/+ sont exploités pour la supervision continue de l'état lumineux.

Après la réception des données, Node-RED extrait l'identifiant de la lampe, son état et le pourcentage de luminosité. L'identifiant permet ensuite de rechercher les coordonnées correspondantes dans la base SQLite. Cette association entre les données de supervision et les coordonnées enregistrées permet d'afficher chaque lampe sur la carte interactive (Figure 7) avec un code couleur indiquant son état : vert pour une lampe opérationnelle et rouge pour une lampe éteinte ou en panne.

Node-RED assure également l'insertion des mesures dans la table d'historique, ce qui permet d'alimenter les interfaces Archives, Historique des pannes et Synthèse. Ainsi, la plateforme joue un rôle central dans le décodage, le filtrage, le stockage et la visualisation des données du système.

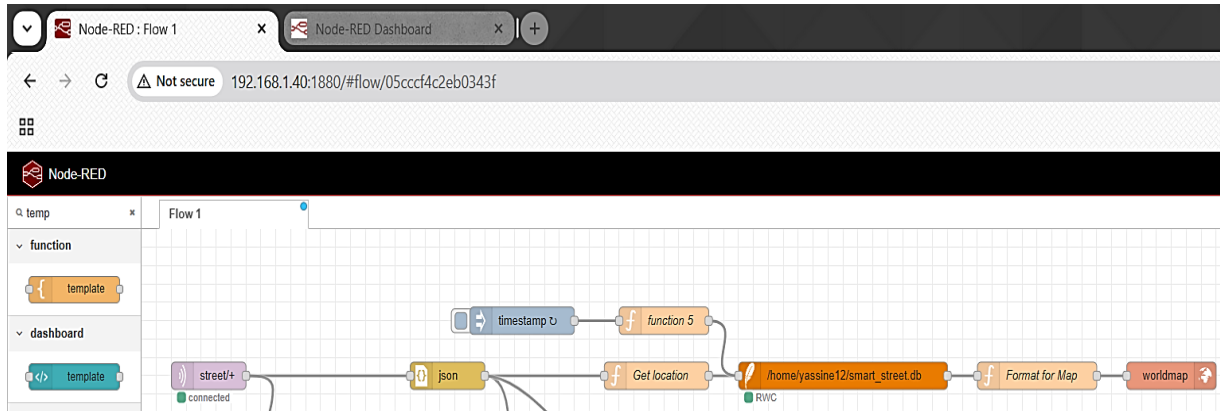


Figure III.9 Flux Node-RED de traitement des données MQTT

Ce flux assure la réception des messages MQTT, l'extraction des données, l'association avec les coordonnées stockées et l'envoi des informations vers la carte interactive et les interfaces de supervision.

III.3.6 Stockage des données dans une base SQLite

Le stockage des données est assuré par une base SQLite hébergée localement sur le Raspberry Pi. Cette base permet de conserver les informations nécessaires au fonctionnement du système, notamment les coordonnées géographiques des lampes et l'historique de leurs états lumineux.

Deux tables principales sont utilisées. La table `lamps_location` contient l'identifiant de chaque lampe ainsi que ses coordonnées GPS, à savoir la latitude et la longitude. Elle est alimentée pendant la phase d'enregistrement initial à partir des messages reçus sur le topic `setup/lamp`. La table `lamps_history` stocke les mesures envoyées par les nœuds ESP32, notamment l'identifiant de la lampe, son état ON/OFF, le pourcentage de luminosité et l'horodatage de la mesure.

Cette organisation permet à Node-RED de relier chaque état reçu à la position géographique correspondante. Elle permet également de générer les tableaux d'archives, l'historique des pannes et les rapports de synthèse affichés dans le Dashboard.

Tableau III.2 Structure des tables SQLite utilisées pour le stockage des données

Table	Rôle	Champs principaux
<code>lamps_location</code>	Stockage des positions GPS des lampes.	id, lat, lon
<code>lamps_history</code>	Stockage de l'historique des états lumineux.	id, status, light, timestamp

III.4 Développement de l'interface de supervision et visualisation des données IoT

Les interfaces de supervision ont été développées sous Node-RED afin de permettre l'exploitation visuelle des données reçues depuis les nœuds ESP32. Après le traitement des messages MQTT et leur stockage dans la base de données SQLite, les informations sont affichées sous plusieurs formes complémentaires : une carte interactive, un tableau d'archivage, un historique des pannes et une synthèse globale du système.

Ces interfaces permettent à l'utilisateur de localiser les lampes, de suivre leur état ON/OFF en temps réel, de consulter les mesures enregistrées et d'évaluer rapidement le niveau de fonctionnement du réseau d'éclairage. Elles constituent ainsi l'interface homme-machine (IHM) du prototype et assurent la transformation des données collectées en informations exploitables pour la supervision et l'aide à la décision.

III.4.1 Interface cartographique de supervision avec Worldmap

L'interface cartographique est développée à l'aide du nœud Worldmap de Node-RED. Elle permet d'afficher les lampes supervisées selon leurs coordonnées GPS enregistrées dans la base de données SQLite. Lorsqu'un message MQTT est reçu depuis un nœud ESP32, Node-RED récupère l'identifiant de la lampe, recherche sa latitude et sa longitude, puis met à jour sa position sur la carte.

L'état de chaque lampe est représenté par un code couleur. Un marqueur vert indique une lampe en fonctionnement normal, tandis qu'un marqueur rouge indique une lampe éteinte ou en panne. Cette interface permet ainsi une localisation rapide des anomalies et facilite les interventions techniques sur le terrain.

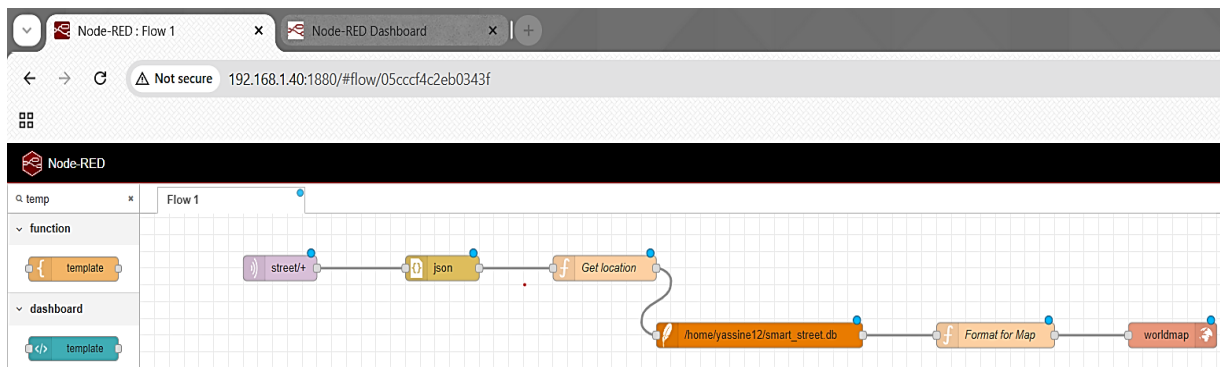


Figure III.10 Interface cartographique Worldmap pour la supervision des lampes

Les marqueurs affichés sur la carte (figure III.10) indiquent la position des lampes et leur état de fonctionnement selon un code couleur verre=ON rouge=OFF.

III.4.2 Interface d'archivage des données

L'interface Archives permet de consulter les données enregistrées dans la base de données SQLite selon une date sélectionnée par l'utilisateur. Après la sélection de la date, Node-RED interroge la table `lamps_history` et l'associe à la table `lamps_location` afin d'afficher les informations principales de chaque lampe.

Le tableau généré présente l'identifiant de la lampe, ses coordonnées GPS, le pourcentage de luminosité mesuré, son état ON/OFF et l'heure de la dernière mesure. Cette interface facilite le suivi historique du réseau d'éclairage et permet de vérifier l'évolution de l'état des lampes au cours d'une journée donnée.

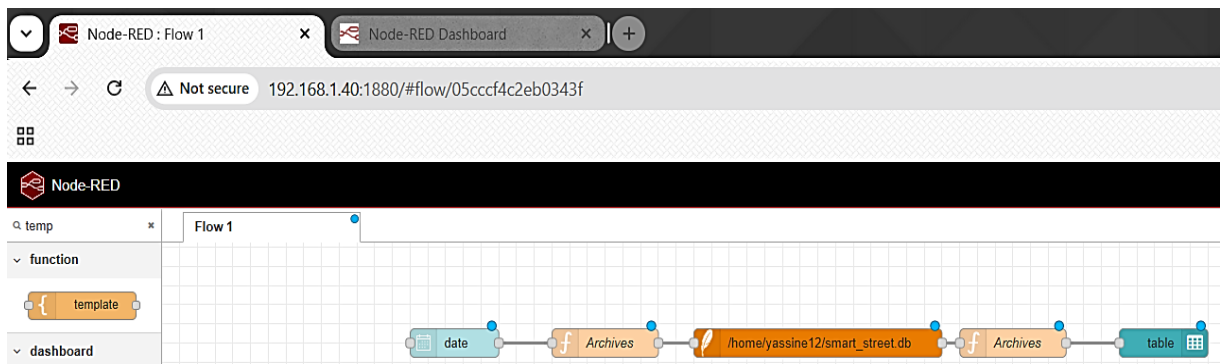


Figure III.11 Interface Archives des états lumineux enregistrés

La figure III.11 affiche les mesures enregistrées selon la date sélectionnée, avec l'identifiant de la lampe, ses coordonnées, son pourcentage lumineux, son état et l'heure de mesure.

III.4.3 Interface de consultation de l'historique des pannes

L'interface Historique des pannes permet d'analyser l'évolution de l'état des lampes au cours d'une journée donnée. Après la sélection de la date, Node-RED récupère les enregistrements stockés dans la table `lamps_history` et les classe chronologiquement afin d'identifier les périodes d'extinction.

Lorsqu'une lampe passe à l'état OFF, le système enregistre l'heure de début de la panne. Lorsque la même lampe repasse à l'état ON, l'interface affiche l'heure d'extinction et l'heure de remise en service. Si la lampe reste éteinte, elle est signalée comme toujours éteinte. Cette interface facilite donc le diagnostic des pannes et permet de suivre la durée d'indisponibilité des points lumineux.

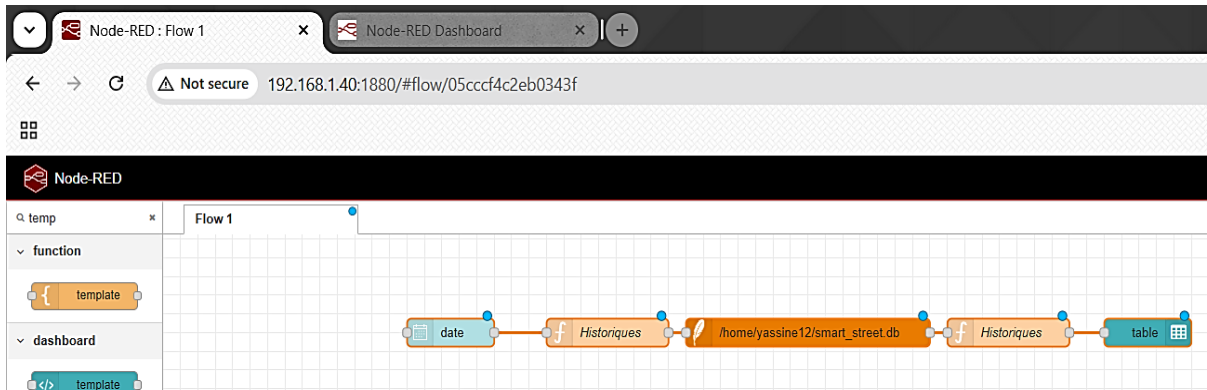


Figure III.12 Interface Historique des pannes des lampes supervisées

Cette interface (figure III.12) affiche les périodes d'extinction détectées pour chaque lampe, ainsi que l'heure de retour à l'état ON lorsque la panne est résolue.

III.4.4 Interface de synthèse globale du système

L'interface Synthèse globale permet d'obtenir une vue rapide sur l'état général du réseau d'éclairage supervisé. À partir de la date sélectionnée, Node-RED récupère la dernière mesure enregistrée pour chaque lampe, puis calcule automatiquement le nombre de lampes opérationnelles et le nombre de lampes en panne.

Les résultats sont présentés sous forme d'un rapport indiquant le pourcentage des lampes en état ON, le pourcentage des lampes en état OFF ainsi que la liste des lampes défaillantes. Cette interface facilite l'évaluation globale du système et permet à l'utilisateur d'identifier rapidement le niveau de disponibilité du réseau d'éclairage.

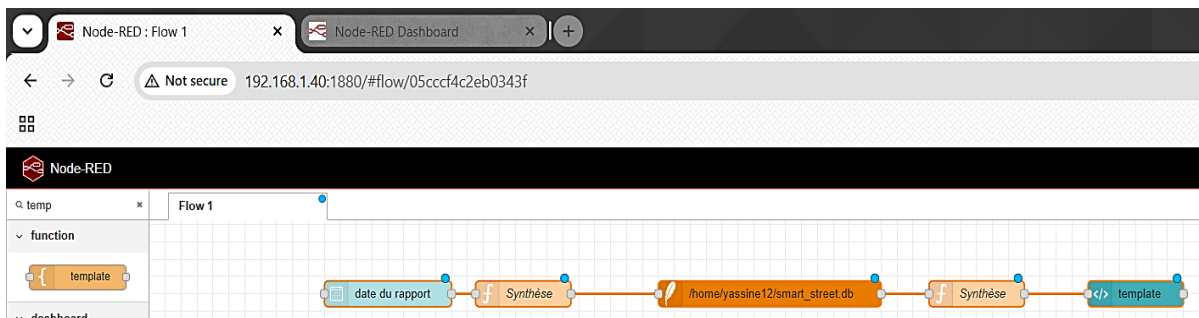


Figure III.13 Interface de synthèse globale du réseau d'éclairage

Cette interface (figure III.13) présente le taux de lampes opérationnelles, le taux de lampes en panne et la liste des points lumineux nécessitant une intervention.

III.5 Conclusion

Dans ce chapitre, nous avons présenté l'implémentation pratique du prototype de supervision intelligente de l'éclairage public. La chaîne IoT développée repose sur l'utilisation de nœuds ESP32, d'un capteur LDR pour la détection de l'état lumineux, d'un module GPS NEO-6M pour l'enregistrement initial des coordonnées, du protocole MQTT pour la transmission des données et du Raspberry Pi comme plateforme centrale.

Ainsi, ce chapitre a permis de concrétiser l'architecture proposée et de valider le fonctionnement global de la chaîne IoT mise en œuvre. Sur la base de cette implémentation, le chapitre suivant sera consacré à l'exploitation des résultats obtenus, à leur analyse, ainsi qu'à l'évaluation des performances et des limites du prototype développé.

Chapitre IV : Expérimentation, validation et évaluation du prototype

IV.1 Introduction

Après avoir présenté l'architecture du système et détaillé l'implémentation de la chaîne IoT dans les chapitres précédents, ce quatrième chapitre est consacré à la validation expérimentale du prototype ainsi qu'à l'analyse des résultats obtenus.

Dans un premier temps, nous présentons les conditions expérimentales mises en place afin de vérifier le bon fonctionnement des différentes composantes du système développé. Les essais réalisés permettent d'évaluer le comportement global de la chaîne IoT dans un contexte de supervision de l'éclairage public.

Dans un second temps, nous exposons les tests effectués afin de valider l'enregistrement des coordonnées GPS, la communication MQTT, la détection de l'état ON/OFF par le capteur LDR ainsi que le fonctionnement des interfaces développées sous Node-RED. La validation porte notamment sur l'affichage cartographique, l'archivage des données, le suivi de l'historique des pannes et la génération d'un rapport de synthèse.

Enfin, les résultats obtenus sont analysés selon plusieurs critères fonctionnels liés à la communication, au stockage, à la détection et à la visualisation des données. Les performances du prototype sont évaluées, ses limites sont identifiées et des pistes d'amélioration sont proposées afin d'envisager des évolutions futures du système.

IV.2 Présentation du prototype expérimental

Cette section présente l'environnement expérimental mis en place afin de valider le fonctionnement du prototype de supervision intelligente de l'éclairage public. Après avoir décrit l'architecture et l'implémentation de la chaîne IoT dans le chapitre précédent, l'objectif est ici de mettre en évidence la réalisation physique du système ainsi que les conditions expérimentales utilisées lors des essais.

Le dispositif expérimental regroupe l'ensemble des équipements matériels et logiciels nécessaires à l'acquisition, la transmission, le traitement et la visualisation des données, afin d'évaluer le comportement global du système dans un contexte contrôlé.

IV.2.1 Prototype réel développé

Le prototype expérimental réalisé est composé d'un serveur local basé sur un Raspberry Pi 4 et de plusieurs unités embarquées ESP32 assurant les fonctions de géolocalisation et de supervision lumineuse.

Une unité mobile ESP32–GPS NEO-6M est utilisée lors de la phase initiale pour l'enregistrement des coordonnées géographiques des points d'éclairage. Une seconde catégorie de nœuds, basée sur ESP32 associés à des capteurs LDR, assure la surveillance continue de l'état lumineux des lampes.

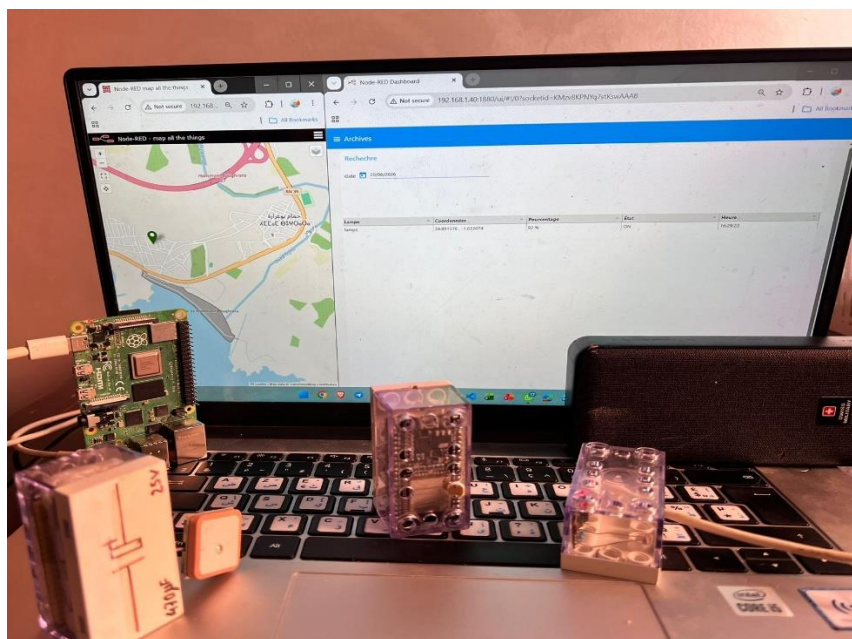


Figure IV.1 Prototype expérimental complet du système de supervision IoT

Les données collectées sont transmises via le protocole MQTT vers le Raspberry Pi, où elles sont traitées sous Node-RED, puis stockées dans la base de données SQLite avant leur visualisation sur les interfaces de supervision.

IV.2.2 Description du montage expérimental

Le montage expérimental a été conçu afin de reproduire les conditions de fonctionnement du système dans un environnement de test contrôlé. Des sources lumineuses de faible puissance ont été utilisées pour simuler les points d'éclairage public et générer différents états (ON/OFF).

Chaque capteur LDR est positionné de manière à capter directement le flux lumineux de la source simulée. Les données acquises par les nœuds ESP32 sont ensuite transmises vers le Raspberry Pi via le réseau Wi-Fi local.

L'ensemble du dispositif permet de reproduire les principales étapes du système : acquisition locale, transmission MQTT, traitement sous Node-RED, stockage dans SQLite et visualisation sur les interfaces de supervision.

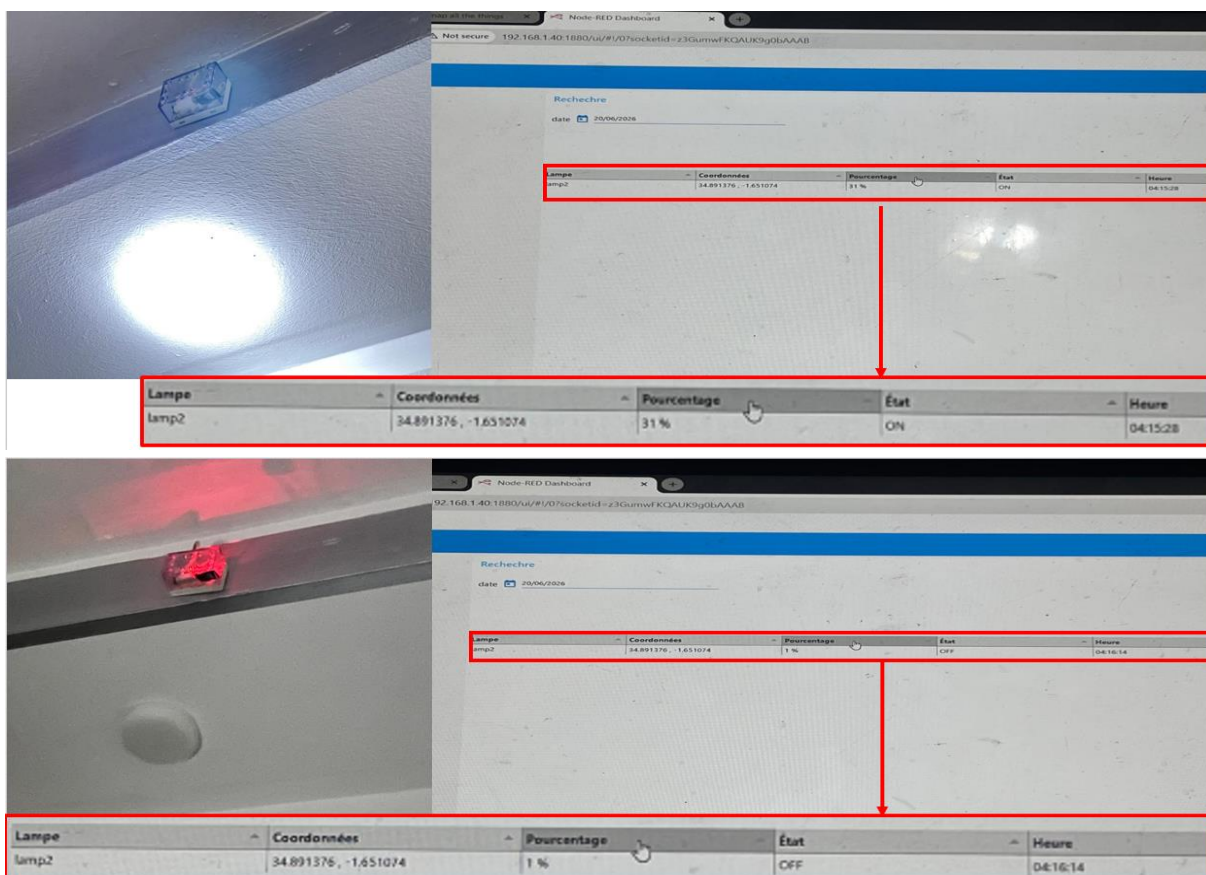


Figure IV.2 Montage expérimental utilisé pour les essais de validation

IV.2.3 Conditions expérimentales

Les essais de validation ont été réalisés dans un environnement local simulant un système de supervision d'éclairage public. L'objectif est d'évaluer le comportement global du prototype dans des conditions proches d'un déploiement réel.

La plateforme expérimentale repose sur un Raspberry Pi 4 configuré comme serveur central, plusieurs nœuds ESP32 dédiés à l'acquisition des données, ainsi qu'un réseau Wi-Fi local assurant la communication via MQTT.

Cette configuration permet de reproduire les principales situations opérationnelles : variations d'éclairage, transmission des données et mise à jour des interfaces de supervision.

IV.2.4 Scénarios de test et validation fonctionnelle

Afin de vérifier le bon fonctionnement du prototype, plusieurs scénarios de test ont été définis et exécutés progressivement. Chaque test vise à valider une fonctionnalité spécifique de la chaîne IoT, sans revenir sur l'architecture déjà détaillée dans les chapitres précédents.

Les essais portent d'abord sur la vérification de l'acquisition des coordonnées GPS, en s'assurant de la bonne réception et du stockage des positions dans la base de données.

Ensuite, la détection de l'état lumineux est évaluée en appliquant différentes intensités lumineuses sur le capteur LDR afin de valider la distinction entre les états ON et OFF.

La communication MQTT est également testée afin de confirmer la transmission correcte des messages entre les nœuds ESP32 et le serveur central via le broker Mosquitto.

Enfin, les interfaces de supervision sont validées à travers l'affichage cartographique, les tableaux d'archives et l'historique des pannes.

Ces tests confirment la cohérence fonctionnelle du système et la conformité des résultats avec les objectifs du projet.

IV.3 Résultats expérimentaux et validation

Les résultats obtenus après l'implémentation complète de la chaîne IoT du prototype. La validation porte sur les principales fonctions du système : la transmission des messages MQTT, l'enregistrement des données dans SQLite, l'affichage cartographique des lampes, la consultation des archives et la génération de la synthèse globale.

Les tests réalisés permettent de vérifier la continuité entre les nœuds ESP32, le broker Mosquitto, la plateforme Node-RED et les interfaces de supervision. L'objectif est de confirmer que chaque lampe peut être identifiée, localisée et suivie selon son état lumineux ON/OFF en temps réel.

IV.3.1 Validation de l'enregistrement des coordonnées GPS

Le premier test porte sur l'enregistrement des coordonnées géographiques des lampes. L'unité mobile composée de l'ESP32 et du module GPS NEO-6M a été placée successivement à proximité des points lumineux à identifier. Après l'obtention d'une position GPS valide, l'ESP32 publie un message MQTT contenant l'identifiant de la lampe, la latitude et la longitude sur le topic setup/lamp.

Les messages reçus sont traités par Node-RED puis insérés dans la table lamps_location de la base SQLite. Cette étape permet d'établir une correspondance unique entre chaque identifiant de lampe et sa position géographique réelle.

Ce mécanisme d'initialisation garantit une séparation claire entre les données de localisation (phase d'enregistrement) et les données de supervision (phase de fonctionnement)

Ainsi, lors de la supervision continue, les nœuds LDR transmettent uniquement l'identifiant et l'état de la lampe, tandis que Node-RED récupère les coordonnées déjà stockées dans la base de données.

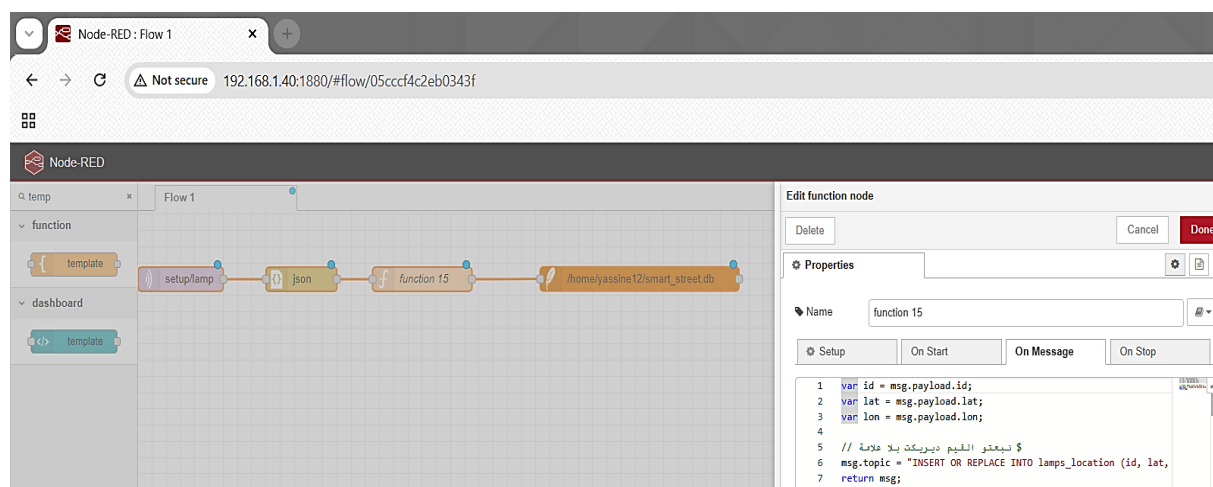


Figure IV.3 Flux Node-RED d'enregistrement des coordonnées GPS dans SQLite

Cette figure IV.3 montre le flux Node-RED dédié à l'enregistrement initial des coordonnées GPS. Les données reçues sur le topic setup/lamp sont converties en format JSON puis insérées dans la table lamps_location à l'aide de l'identifiant de la lampe, de la latitude et de la longitude.

IV.3.2 Validation de la communication MQTT

La validation de communication MQTT a été réalisée afin de vérifier l'échange des données entre les nœuds ESP32 et le Raspberry Pi. Les ESP32 publient leurs messages vers le broker Mosquitto en utilisant deux topics principaux : setup/lamp pour l'enregistrement des coordonnées GPS et street/+ pour la transmission de l'état lumineux des lampes.

Les résultats expérimentaux confirment la bonne réception des messages par la plateforme Node-RED, sans perte ni désynchronisation observable lors des tests. Après réception, les données sont converties au format JSON puis orientées vers les blocs de traitement correspondants.

Cette validation met en évidence la robustesse du modèle publish/subscribe implémenté, ainsi que la bonne intégration entre les nœuds embarqués, le broker Mosquitto et l'environnement de traitement Node-RED.

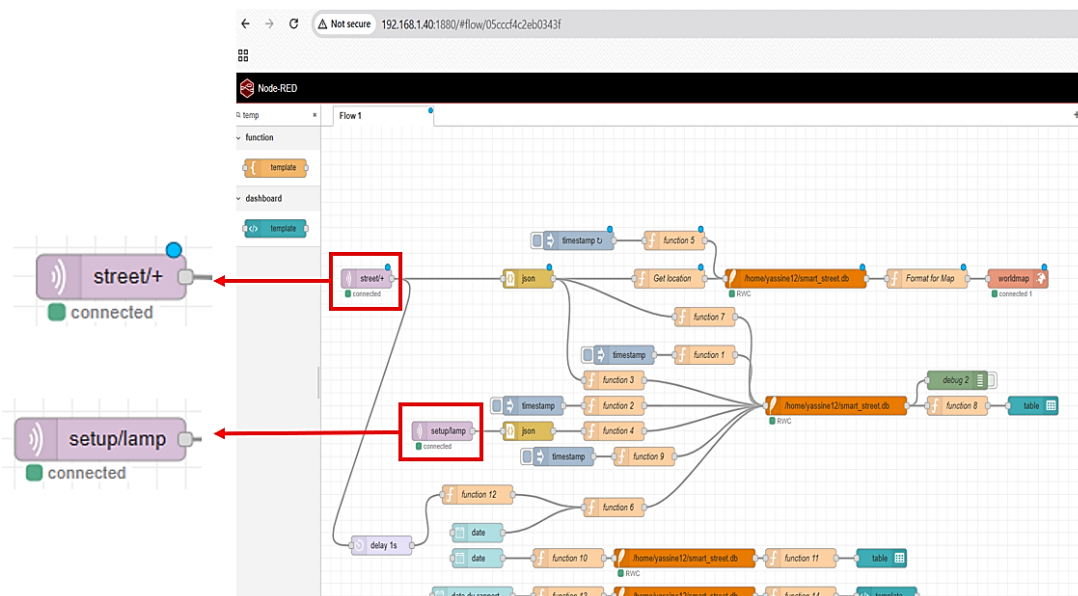


Figure IV.4 Test de réception des messages MQTT sous Node-RED

La figure IV.4 illustre la réception des messages MQTT publiés par les nœuds ESP32 sur les topics `setup/lamp` et `street/+`.

IV.3.3 Validation de la détection de l'état lumineux (LDR)

Ce test a pour objectif de vérifier la capacité du capteur LDR à détecter l'état lumineux de la lampe. Le capteur est connecté à une entrée analogique de l'ESP32 selon une configuration en diviseur de tension, permettant la conversion des variations de luminosité en signal électrique exploitable.

Lorsqu'une lampe est allumée, l'intensité lumineuse mesurée augmente, entraînant une valeur analogique plus élevée. Dans ce cas, l'état logique transmis est *ON*. À l'inverse, lorsque la luminosité diminue ou devient nulle, la valeur mesurée chute et l'état est interprété comme *OFF*.

Les mesures sont ensuite normalisées sous forme de pourcentage et comparées à un seuil défini dans le programme embarqué afin d'assurer une classification robuste des états.

Les résultats issus de l'interface de supervision confirment une distinction correcte entre les deux états. Une valeur lumineuse élevée correspond systématiquement à un état *ON*, tandis qu'une valeur proche de 0 % correspond à un état *OFF*.

Cette validation confirme la fiabilité de la chaîne d'acquisition locale basée sur ESP32 et LDR avant la transmission des données vers le broker MQTT et leur exploitation dans Node-RED.

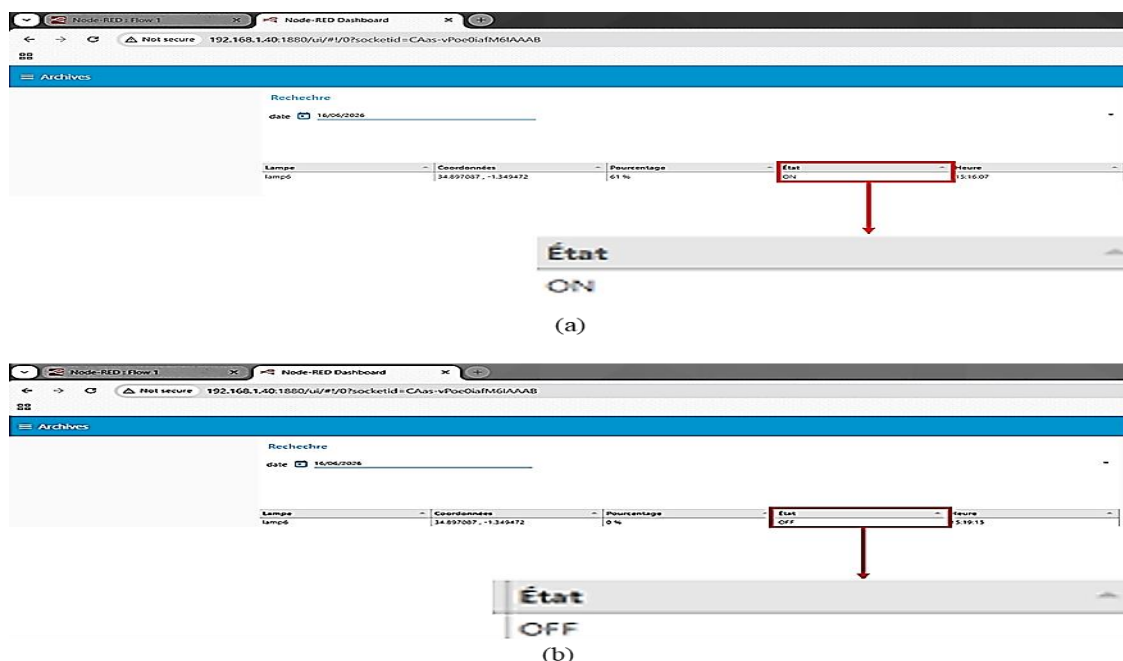


Figure IV.5 Détection de l'état (a) ON (b) OFF d'une lampe à partir du capteur LDR

La figure IV.5 illustre la validation expérimentale du système de détection lumineuse. Dans le premier cas, la lampe est identifiée à l'état ON avec une luminosité mesurée de 61 %. Dans le second cas, la valeur mesurée est nulle, correspondant à l'état OFF.

IV.3.4 Validation de la carte interactive de supervisions

La validation de la carte interactive a été réalisée afin de vérifier l'affichage géographique des lampes supervisées. Après la réception d'un message MQTT, Node-RED récupère l'identifiant de la lampe, interroge la base SQLite pour obtenir ses coordonnées GPS, puis transmet les données au nœud Worldmap.

Les résultats expérimentaux confirment que chaque point lumineux est correctement positionné sur la carte à son emplacement réel. L'état de fonctionnement est représenté par un code couleur : un marqueur vert correspond à une lampe en état ON, tandis qu'un marqueur rouge indique une lampe en état OFF.

Cette représentation cartographique permet une identification rapide des anomalies et facilite la localisation des interventions sur le terrain.

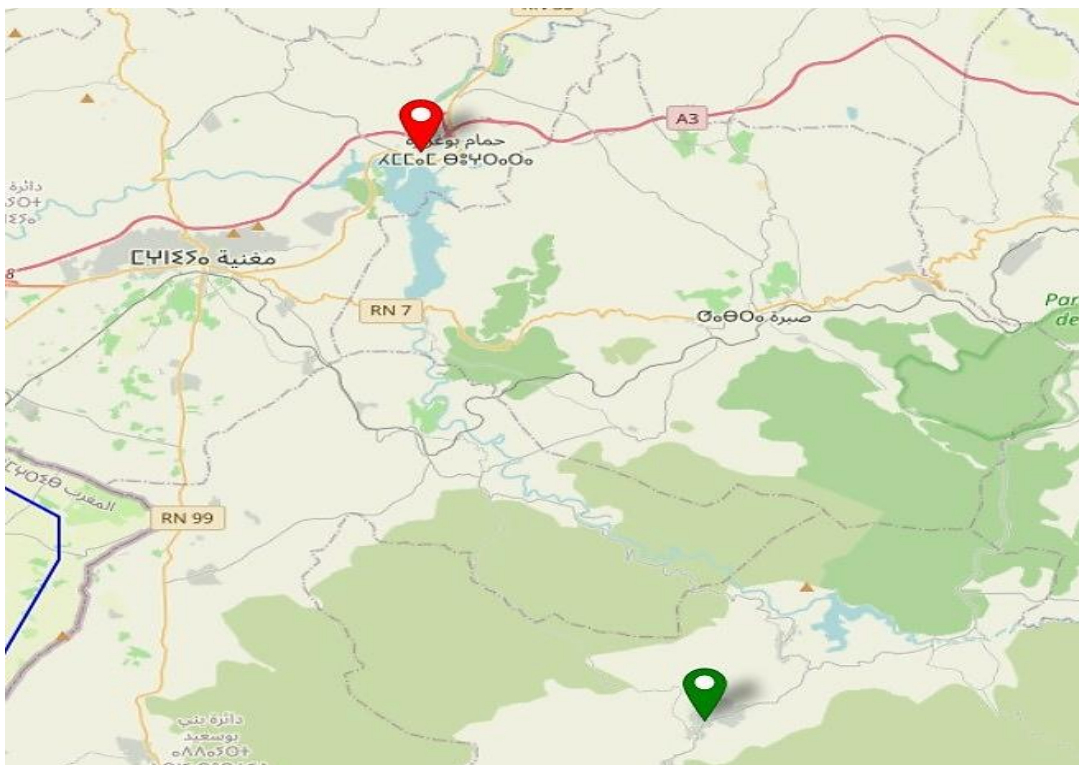


Figure IV.6 Validation de l’affichage cartographique des lampes sous Node-RED Worldmap

Les marqueurs affichés sur la carte (figure IV.6) indiquent la position des lampes et leur état de fonctionnement. Le vert correspond à l’état ON, tandis que le rouge correspond à l’état OFF.

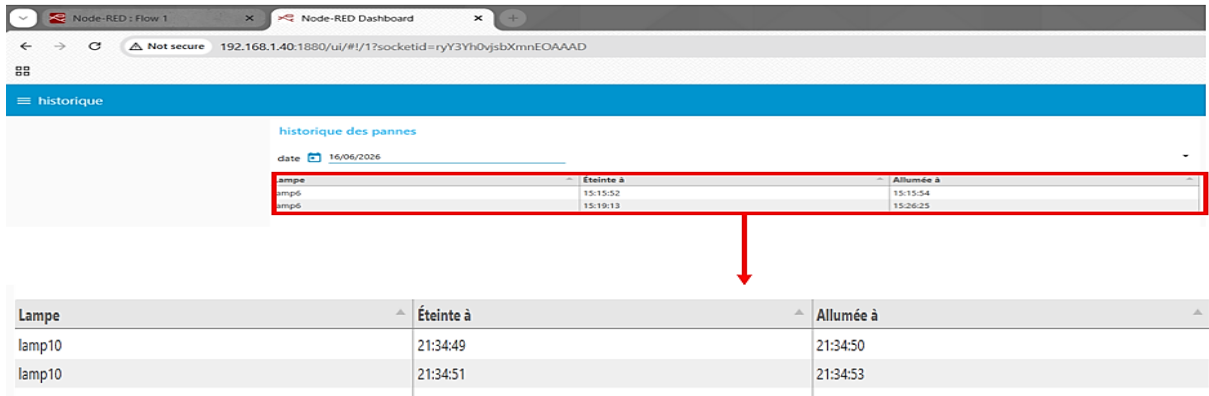
IV.3.5 Validation de l’historique des pannes

La validation de l’historique des pannes consiste à vérifier la capacité du système à suivre les changements d’état des lampes au cours du temps. L’objectif est de vérifier que les transitions entre les états *ON* et *OFF* sont correctement détectées, enregistrées et exploitées par la plateforme de supervision.

Lorsqu’une lampe passe de l’état *ON* à l’état *OFF*, Node-RED enregistre l’instant d’extinction à partir des données stockées dans la table *lamps_history*. Lors du retour à l’état *ON*, le système associe automatiquement l’heure de rallumage à l’épisode de panne correspondant.

Dans le cas où une lampe demeure à l’état *OFF*, elle est considérée comme en panne persistante et signalée comme telle dans l’interface.

Ce mécanisme confirme que le prototype ne se limite pas à une observation instantanée des états, mais intègre également une dimension temporelle permettant le suivi des interruptions de service et l’analyse des défaillances.



historique des pannes

date 16/06/2026

lampe	Éteinte à	Allumée à
lampe6	15:15:52	15:19:54
lampe6	15:19:13	15:26:25

Lampe	Éteinte à	Allumée à
lampe10	21:34:49	21:34:50
lampe10	21:34:51	21:34:53

Figure IV.7 Historique des pannes des lampes supervisées

L'interface présentée en figure IV.7 met en évidence les périodes d'extinction détectées pour chaque lampe, ainsi que les instants de rétablissement lorsque l'état *ON* est rétabli.

IV.3.6 Validation du rapport de synthèse

La validation du rapport de synthèse a pour objectif d'évaluer la capacité du système à fournir une vue globale et automatisée de l'état du réseau d'éclairage. À partir des dernières mesures enregistrées pour chaque lampe, Node-RED détermine le nombre de lampes en état *ON* et en état *OFF*, puis calcule les pourcentages de disponibilité associés.

Les résultats expérimentaux montrent que l'interface de synthèse permet une évaluation rapide et cohérente de l'état global du système. Lorsque toutes les lampes sont opérationnelles, le taux de fonctionnement atteint 100 %. Lorsqu'une lampe est détectée en panne, son identifiant est affiché dans la liste des lampes défectueuses, ce qui facilite l'identification rapide du point lumineux nécessitant une intervention.



Résumé

Rapport Global du Système

✓ Opérationnel: 50.0% (1)

✗ En panne: 50.0% (1)

⚠ Lampes en panne: lampe2

date du rapport 16/05/2026

Figure IV.8 Validation du rapport de synthèse globale du système

L'interface de synthèse présentée dans la figure IV.8 ci-dessus affiche le taux de lampes opérationnelles, le taux de lampes en panne ainsi que les identifiants des lampes nécessitant une intervention.

Ce mécanisme de synthèse constitue un indicateur décisionnel essentiel, permettant une supervision rapide et une analyse globale de la performance du réseau d'éclairage.

IV.4 Discussion des résultats

Les résultats obtenus montrent que le prototype assure correctement les principales fonctions attendues d'un système IoT de supervision de l'éclairage public. L'enregistrement initial des coordonnées GPS permet d'associer chaque lampe à une position géographique fixe, tandis que les nœuds ESP32-LDR assurent la détection locale de l'état lumineux. La transmission MQTT entre les ESP32 et le Raspberry Pi permet une communication légère et adaptée à un fonctionnement en temps réel du système.

L'intégration de Node-RED et de SQLite permet de centraliser le traitement, le stockage et l'exploitation des données. Les interfaces développées présentent clairement l'état des lampes sous plusieurs formes : affichage cartographique, tableau d'archives, historique des pannes et synthèse globale. Cette organisation facilite l'identification rapide des lampes défectueuses et améliore le suivi technique du réseau.

Cependant, les performances du prototype restent liées à certains paramètres, notamment la stabilité du réseau Wi-Fi, la précision du module GPS lors de la phase d'enregistrement et le choix du seuil de détection lumineux dans le programme de l'ESP32. Malgré ces limites, les résultats confirment la faisabilité de la solution proposée et montrent son intérêt pour une supervision numérique, locale et automatisée de l'éclairage public.

IV.4.1 Évaluation des performances du prototype

L'évaluation des performances du prototype a été réalisée selon plusieurs critères fonctionnels : la fiabilité de la communication MQTT, la rapidité de mise à jour des interfaces, la cohérence du stockage dans SQLite et la précision de la détection ON/OFF par le capteur LDR.

Les tests montrent que le système assure une transmission correcte des messages entre les nœuds ESP32 et le Raspberry Pi. Les données publiées sont reçues par Node-RED, stockées dans la base SQLite, puis affichées sur les interfaces de supervision. La carte interactive permet une visualisation rapide de l'état des lampes, tandis que les interfaces Archives, Historique des pannes et Synthèse permettent un suivi structuré des événements.

Les performances du prototype sont satisfaisantes pour une application locale de supervision. Cependant, elles dépendent principalement de la stabilité du réseau Wi-Fi, de la qualité du signal GPS lors de l'enregistrement initial et du seuil choisi pour la détection lumineuse. Une amélioration future pourrait intégrer des tests de latence, une alimentation autonome et une sécurisation avancée des échanges MQTT.

Tableau IV.1 Évaluation fonctionnelle des performances du prototype

Critère évalué	Résultat observé
Communication MQTT	Réception correcte des messages ESP32
Détection LDR	Distinction des états ON/OFF
Stockage SQLite	Enregistrement des positions et historiques
Worldmap	Affichage des lampes selon leur état
Dashboard	Synthèse et archives exploitables

IV.4.2 Valeur ajoutée du prototype

Le prototype réalisé apporte une valeur ajoutée importante dans la supervision de l'éclairage public, car il permet de passer d'un contrôle manuel et réactif à une supervision numérique, automatisée et centralisée. Grâce à l'utilisation des nœuds ESP32, du protocole MQTT et de la plateforme Node-RED, l'état des lampes peut être suivi en temps réel sans intervention directe sur le terrain.

L'intégration de la localisation GPS, du stockage SQLite et de la carte interactive permet d'identifier rapidement la lampe concernée par une panne ainsi que sa position géographique. De plus, les interfaces Archives, Historique des pannes et Synthèse facilitent l'analyse des événements, le suivi des défauts et la prise de décision pour les opérations de maintenance.

Ainsi, le prototype constitue une solution légère, évolutive et adaptée aux applications IoT urbaines. Il peut être amélioré ultérieurement par l'ajout d'une alimentation autonome, d'une communication longue portée et de mécanismes de sécurité afin de se rapprocher d'un déploiement réel à grande échelle.

IV.5 Intégration physique du prototype

L'approche adoptée s'inscrit dans une logique de protection physique et de projection vers un déploiement en conditions réelles.

IV.5.1 Intégration du prototype dans un boîtier

L'intégration du prototype dans un boîtier constitue une étape essentielle visant à assurer la protection mécanique et la stabilité fonctionnelle du nœud IoT en vue d'un déploiement en environnement réel. Cette phase consiste à regrouper les différents composants électroniques (ESP32, capteur LDR et éléments de câblage) dans une structure compacte et organisée.

Le boîtier choisi permet de réduire l'exposition des circuits aux contraintes extérieures telles que la poussière, l'humidité et les variations thermiques, tout en garantissant la cohérence des connexions électriques et la facilité de maintenance. Une attention particulière est accordée à l'orientation du capteur LDR, qui nécessite une ouverture ou une zone transparente afin de préserver la précision des mesures lumineuses.

IV.5.1.1 Assemblage et intégration du système

L'assemblage du prototype a été réalisé sous forme de deux unités distinctes. La première est un nœud fixe constitué d'un ESP32 associé à un capteur LDR monté en diviseur de tension. Ce nœud assure l'acquisition du niveau lumineux, la détermination de l'état ON/OFF de la lampe et la publication des données via MQTT.

La seconde unité associe un ESP32 au module GPS NEO-6M. Elle est utilisée de manière mobile lors de la phase d'initialisation afin d'enregistrer successivement les coordonnées géographiques de chaque lampe.

Les deux unités communiquent via Wi-Fi avec le Raspberry Pi 4, qui centralise le broker Mosquitto, le traitement Node-RED et le stockage SQLite.

L'intégration modulaire retenue facilite les essais, la maintenance ainsi que le remplacement des différents blocs fonctionnels. Dans une perspective de déploiement réel, le montage expérimental sur plaque d'essai peut être remplacé par un circuit imprimé dédié, afin d'améliorer la stabilité électrique et la robustesse mécanique du système.

IV.5.1.2 Choix du boîtier d'encapsulation

Le choix du boîtier d'encapsulation constitue une étape importante pour protéger les composants électroniques du nœud IoT contre les contraintes extérieures, notamment la poussière, l'humidité, les variations de température et les chocs mécaniques. Le boîtier doit également permettre la réception du signal Wi-Fi de l'ESP32 et l'exposition du capteur LDR à la lumière de la lampe.

Dans le cadre de ce projet, deux boîtiers disponibles au laboratoire ont été sélectionnés afin d'étudier leur adéquation avec l'intégration des différents composants du système. Ce choix a permis de vérifier la possibilité de regrouper l'ESP32, le capteur LDR et les connexions électriques dans une structure compacte, tout en facilitant la fixation et la maintenance.

Une ouverture ou une zone transparente est prévue au niveau du capteur LDR afin de garantir une mesure fiable de l'intensité lumineuse.

La figure suivante présente les boîtiers choisis et utilisés pour l'intégration expérimentale du prototype. Pour un déploiement réel en environnement extérieur, ces boîtiers devront être améliorés par l'utilisation de matériaux résistants aux rayonnements UV, de joints d'étanchéité et d'un indice de protection adapté, tel que IP65 ou IP67.



Figure IV.9 Boîtiers d'encapsulation choisis pour l'intégration du prototype

IV.5.2 Stratégie d'alimentation synchronisée avec la ligne d'éclairage

Dans une perspective de déploiement réel, l'alimentation du nœud IoT est envisagée directement à partir de la ligne d'éclairage public. Un convertisseur AC/DC isolé permet d'abaisser la tension du réseau, de 230 V AC vers une tension continue stabilisée de 5 V, adaptée à l'alimentation de l'ESP32 et des capteurs associés.

Le module d'alimentation est raccordé en parallèle et en amont de la lampe. Ainsi, lorsque le réseau d'éclairage est activé pendant la nuit, le nœud IoT est automatiquement mis sous tension et assure la supervision. En cas de défaillance de la lampe, l'ESP32 reste alimenté, détecte l'absence de luminosité à l'aide du capteur LDR et transmet l'information au serveur via MQTT.

Cette stratégie assure une synchronisation automatique entre le fonctionnement du nœud et celui du réseau d'éclairage. Elle permet également de réduire la consommation inutile en période diurne et d'améliorer la détection des pannes physiques des lampes.

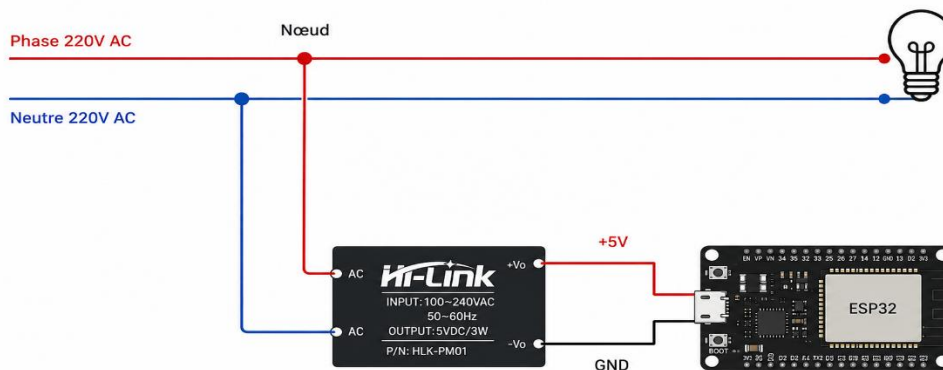


Figure IV.10 Principe d'alimentation synchronisée du nœud IoT avec la ligne d'éclairage public

IV.6 Conclusion

Ce chapitre a permis de valider expérimentalement le prototype de supervision intelligente de l'éclairage public et d'évaluer son comportement dans un environnement de test contrôlé. Les essais réalisés ont confirmé le bon fonctionnement des différentes briques de la chaîne IoT, depuis l'acquisition des données par les nœuds ESP32 jusqu'à leur visualisation sur les interfaces de supervision.

Les résultats obtenus montrent une transmission fiable des messages via MQTT, un stockage cohérent dans la base SQLite et une exploitation efficace des données au niveau de Node-RED. L'ensemble des modules développés — carte interactive, archivage, historique des pannes et synthèse globale — assure une visualisation claire et structurée de l'état du réseau d'éclairage.

L'intégration matérielle du prototype, notamment à travers les boîtiers et la stratégie d'alimentation synchronisée avec la ligne d'éclairage, constitue une étape clé vers un déploiement réel. Elle met en évidence les contraintes liées à la robustesse, à la protection et à l'alimentation des nœuds IoT.

Enfin, cette phase expérimentale confirme la viabilité technique de la solution proposée tout en ouvrant la voie à des améliorations futures en matière de robustesse matérielle, de couverture réseau et de sécurité des communications.

Conclusion générale et perspectives

Conclusion générale et perspectives

Ce mémoire a été consacré à la conception et à la réalisation d'un prototype IoT destiné à la supervision intelligente de l'éclairage public. L'objectif principal était de proposer une solution locale capable de détecter l'état lumineux des lampes, d'enregistrer leur position géographique et de présenter les informations collectées sous une forme exploitable pour le suivi technique, la maintenance et l'aide à la décision.

Dans un premier temps, l'étude théorique a permis de situer le travail dans le contexte des villes intelligentes et de mettre en évidence l'importance de l'Internet des objets dans la modernisation des infrastructures urbaines. L'éclairage public constitue, dans ce cadre, un domaine d'application pertinent, car il nécessite une surveillance continue, une détection rapide des anomalies et une meilleure organisation des interventions de maintenance.

La solution proposée repose sur une architecture IoT composée de nœuds ESP32, de capteurs LDR pour la détection de l'état lumineux, d'un module GPS NEO-6M pour l'enregistrement des coordonnées géographiques, ainsi que d'un Raspberry Pi 4 utilisé comme plateforme centrale. Le protocole MQTT a été adopté pour assurer une communication légère et adaptée aux applications IoT, tandis que Node-RED et SQLite ont permis le traitement, la visualisation et le stockage local des données.

La phase expérimentale a permis de valider les principales fonctions du prototype, notamment la communication entre les nœuds ESP32 et le Raspberry Pi, l'enregistrement des coordonnées GPS, la détection de l'état ON/OFF des lampes et l'affichage des résultats à travers des interfaces de supervision. Les interfaces développées, telles que la carte interactive, les archives, l'historique des pannes et la synthèse globale, ont montré la capacité du système à fournir une vision claire et exploitable de l'état du réseau d'éclairage.

Par ailleurs, l'étude du passage vers un déploiement plus avancé a permis d'identifier plusieurs éléments nécessaires à l'industrialisation du système, notamment la conception d'un circuit imprimé, l'intégration dans un boîtier de protection adapté, l'optimisation de l'alimentation, l'extension de la communication par une topologie ESP-WiFi-Mesh, ainsi que le renforcement de la sécurité et des mécanismes de diagnostic. Ces aspects constituent des leviers importants pour améliorer la fiabilité, la résilience et l'évolutivité de la solution proposée.

Les résultats obtenus montrent que le prototype réalisé est fonctionnel et répond aux objectifs fixés à l'échelle expérimentale. Il permet l'acquisition locale des données, leur transmission

via MQTT, leur stockage dans une base SQLite et leur exploitation à travers des interfaces de supervision. Le système facilite ainsi l'identification des lampes éteintes ou défaillantes et constitue une base technique intéressante pour une gestion plus intelligente de l'éclairage public.

Toutefois, certaines limites doivent être prises en considération. Le fonctionnement du prototype dépend notamment de la stabilité du réseau Wi-Fi, de la précision de la mesure lumineuse par LDR, de la qualité du signal GPS lors de l'enregistrement initial et du choix du seuil de détection. De plus, les aspects liés à la sécurité avancée, à la communication maillée et au déploiement en conditions réelles nécessitent des validations supplémentaires.

En perspective, ce travail pourrait être amélioré par l'intégration d'une alimentation autonome, l'utilisation de technologies de communication longue portée ou maillée, le renforcement des mécanismes de cybersécurité et l'exploitation des données historiques pour la maintenance prédictive. L'intégration d'algorithmes d'analyse ou d'intelligence artificielle pourrait également permettre d'anticiper les pannes, d'optimiser la consommation énergétique et d'améliorer la gestion globale du réseau d'éclairage.

En conclusion, le prototype développé constitue une base fonctionnelle, évolutive et adaptée aux besoins de supervision intelligente de l'éclairage public. Il démontre l'intérêt de l'intégration des systèmes embarqués, des communications sans fil, du traitement local des données et des interfaces de supervision dans le cadre des applications IoT dédiées aux villes intelligentes.

Perspectives

En raison des contraintes de temps, certaines améliorations n'ont pas été implémentées ni validées expérimentalement. Elles constituent néanmoins des axes d'évolution du prototype :

- Industrialisation du prototype : conception d'un circuit imprimé (PCB) intégrant l'ESP32, les capteurs et l'alimentation afin d'améliorer la compacité et la fiabilité du système.
- Architecture réseau évoluée : adoption d'une topologie ESP-WIFI-MESH pour étendre la couverture du réseau et renforcer sa résilience.
- Sécurisation du système IoT : mise en place de l'authentification MQTT, de listes de contrôle d'accès (ACL) et du chiffrement des échanges.
- Optimisation énergétique : intégration de modes basse consommation et gestion intelligente de l'alimentation des nœuds.
- Amélioration de la précision : calibration avancée des seuils du capteur LDR et amélioration de la fiabilité des mesures en environnement extérieur.

- Déploiement cloud hybride : extension vers une architecture cloud afin de permettre la supervision multi-sites et l'analyse à grande échelle.

Bibliographie

Bibliographie

- [1] A. Caragliu, C. Del Bo, and P. Nijkamp, “Smart cities in Europe,” *Journal of Urban Technology*, vol. 18, no. 2, pp. 65–82, 2011, doi: 10.1080/10630732.2011.601117.
- [2] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, “Internet of Things for Smart Cities,” *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, Feb. 2014, doi: 10.1109/JIOT.2014.2306328.
- [3] A. Caragliu and C. Del Bo, “Smartness and European urban performance: Assessing the local impacts of smart urban attributes,” *Innovation: The European Journal of Social Science Research*, vol. 25, no. 2, pp. 97–113, 2012, doi: 10.1080/13511610.2012.660323.
- [4] “Smart City, IoT et choix du réseau (1/2) : l’IoT dans la stratégie smart city,” *Les Smart Grids*. Consulté le : 13 juin 2026. [En ligne]. Disponible sur : <https://les-smartgrids.fr/smart-city-iot-choix-reseau-1-2/>
- [5] A. G. Bell, “On the production and reproduction of sound by light,” *American Journal of Science*, vol. s3-20, no. 118, pp. 305–324, Oct. 1880, doi: 10.2475/ajs.s3-20.118.305.
- [6] N. Wiener, *Cybernetics: Or Control and Communication in the Animal and the Machine*. Cambridge, MA, USA: MIT Press, 1948.
- [7] B. McLoughlin, *Urban and Regional Planning: A Systems Approach*. London, U.K.: Faber and Faber, 1969.
- [8] J. Abbate, *Inventing the Internet*. Cambridge, MA, USA: MIT Press, 1999.
- [9] L. Atidel, C. Larabi, A. Beghdadi, E. Viennet, and A. Bouridane, “Knowledge-based taxonomic scheme for full-reference objective image quality measurement models,” *Electronic Imaging*, vol. 2017, no. 12, pp. 64–78, Jan. 2017, doi: 10.2352/ISSN.2470-1173.2017.12.IQSP-228.
- [10] M. Dieuzeide and C. Coulée, “Smart cities : quelles sont les principales menaces ?,” 2018. Consulté le : 6 mai 2026. [En ligne]. Disponible sur : <https://hdl.handle.net/2078.2/5400>
- [11] R. Giffinger, C. Fertner, H. Kramar, R. Kalasek, N. Pichler-Milanović, and E. Meijers, *Smart Cities: Ranking of European Medium-Sized Cities*. Vienna, Austria: Centre of Regional Science, Vienna University of Technology, 2007.
- [12] “Stratégies et projets Smart City en Belgique (2018),” *Université de Liège*. Consulté le : 7 mai 2026. [En ligne]. Disponible sur : https://www.smart-city.uliege.be/cms/c_5688723/fr/strategies-et-projets-smart-city-en-belgique-2018
- [13] T. Nam and T. A. Pardo, “Conceptualizing smart city with dimensions of technology, people, and institutions,” in *Proc. 12th Annual International Digital Government Research Conference: Digital*

Government Innovation in Challenging Times, College Park, MD, USA, 2011, pp. 282–291, doi: 10.1145/2037556.2037602.

[14] “Smart cities : un modèle européen plus respectueux et transparent,” *Techniques de l’Ingénieur*. Consulté le : 7 mai 2026. [En ligne]. Disponible sur : <https://www.techniques-ingenieur.fr/actualite/articles/smart-cities-un-modele-europeen-plus-respectueux-et-transparent-121203/>

[15] “Développement d’un système d’IoT (Internet of Things) pour le Smart Lighting sous la plateforme IBM.” Consulté le : 7 mai 2026. [En ligne]. Disponible sur : <https://theses-algerie.com/2815143823569508/memoire-de-master/universite-larbi-ben-mhidi-omelbouaghi/developpement-dun-systeme-diot-internet-of-things-pour-le-smart-lighting-sous-la-plateforme-ibm>

[16] B. Ra and J. D. S, “La ville intelligente : origine, définitions, forces et limites d’une expression polysémique.”

[17] M. Thite, “Smart cities: Implications of urban planning for human resource development,” *Human Resource Development International*, vol. 14, no. 5, pp. 623–631, 2011, doi: 10.1080/13678868.2011.618349.

[18] J. V. Winters, “Why are smart cities growing? Who moves and who stays,” *Journal of Regional Science*, vol. 51, no. 2, pp. 253–270, May 2011, doi: 10.1111/j.1467-9787.2010.00693.x.

[19] A. Meijer and M. P. Rodríguez Bolívar, “Governing the smart city: A review of the literature on smart urban governance,” *International Review of Administrative Sciences*, vol. 82, no. 2, pp. 392–408, 2016, doi: 10.1177/0020852314564308.

[20] C. Benevolo, R. P. Dameri, and B. D’Auria, “Smart mobility in smart city: Action taxonomy, ICT intensity and public benefits,” in *Empowering Organizations*, T. Torre, A. M. Braccini, and R. Spinelli, Eds. Cham, Switzerland: Springer, 2016, pp. 13–28, doi: 10.1007/978-3-319-23784-8_2.

[21] A. Kramers, M. Höjer, N. Lövehagen, and J. Wangel, “Smart sustainable cities: Exploring ICT solutions for reduced energy use in cities,” *Environmental Modelling & Software*, vol. 56, pp. 52–62, Jun. 2014, doi: 10.1016/j.envsoft.2013.12.019.

[22] J. M. Shapiro, “Smart cities: Quality of life, productivity, and the growth effects of human capital,” *The Review of Economics and Statistics*, vol. 88, no. 2, pp. 324–335, May 2006, doi: 10.1162/rest.88.2.324.

[23] “Système IoT pour Smart Lighting,” *Scribd*. Consulté le : 7 mai 2026. [En ligne]. Disponible sur : <https://fr.scribd.com/document/511415688/memoire-finale-iot>

[24] “Internet des objets-Part 2-Les protocoles,” *ResearchGate*. Consulté le : 7 mai 2026. [En ligne]. Disponible sur : https://www.researchgate.net/publication/311844730_Internet_des_objets-Part_2-Les_protocoles

- [25] “Internet of things Applications [7] | Download Scientific Diagram,” *ResearchGate*. Consulté le : 13 juin 2026. [En ligne]. Disponible sur : https://www.researchgate.net/figure/Internet-of-things-Applications-7_fig1_370153008
- [26] N. Farhi, “Mobilité urbaine : Systèmes de transport intelligents,” *Séminaire Mathématique de Béjaïa*, vol. 19, no. 1, p. 11, Dec. 2019. Consulté le : 7 mai 2026. [En ligne]. Disponible sur : <https://asjp.cerist.dz/en/article/232089>
- [27] G. Dimitrakopoulos and P. Demestichas, “Intelligent transportation systems,” *IEEE Vehicular Technology Magazine*, vol. 5, no. 1, pp. 77–84, Mar. 2010, doi: 10.1109/MVT.2009.935537.
- [28] “Smart Building : les tendances tech des bâtiments en 2026.” Consulté le : 7 mai 2026. [En ligne]. Disponible sur : <https://www.constructionbtp.com/batiment/article/2026/01/06/154685/smart-building-les-tendances-tech-des-batiments-2026>
- [29] “Modelling the smart city performance,” *Innovation: The European Journal of Social Science Research*, vol. 25, no. 2. Consulté le : 13 juin 2026. [En ligne]. Disponible sur : <https://www.tandfonline.com/doi/abs/10.1080/13511610.2012.660325>
- [30] B. Benbba, F. Ismaili, and Y. El Archi, “Smart tourism destination et développement durable : Quels apports à l’expérience touristique ?,” *Alternatives Managériales Économiques*, pp. 1–18, Mar. 2023, doi: 10.48374/IMIST.PRSM/ame-v1i0.36926.
- [31] “E-santé : télésanté, santé numérique ou santé connectée,” 2025.
- [32] “Introduction à l’Internet des Objets,” *Scribd*. Consulté le : 8 mai 2026. [En ligne]. Disponible sur : <https://www.scribd.com/document/611733235/cours-extensif-IntobjT>
- [33] A. Tzounis, N. Katsoulas, T. Bartzanas, and C. Kittas, “Internet of Things in agriculture, recent advances and future challenges,” *Biosystems Engineering*, vol. 164, pp. 31–48, Dec. 2017, doi: 10.1016/j.biosystemseng.2017.09.007.
- [34] R. Kitchin, “The real-time city? Big data and smart urbanism,” *GeoJournal*, vol. 79, no. 1, pp. 1–14, Feb. 2014, doi: 10.1007/s10708-013-9516-8.
- [35] Z. Mumtaz, S. Ullah, Z. Ilyas, N. Aslam, S. Iqbal, S. Liu, J. A. Meo, and H. A. Madni, “An automation system for controlling streetlights and monitoring objects using Arduino,” *Sensors*, vol. 18, no. 10, Art. no. 3178, Oct. 2018, doi: 10.3390/s18103178.
- [36] L. Qin, L.-L. Dong, W.-H. Xu, L.-D. Zhang, and A. Leon, “An intelligent luminance control method for tunnel lighting based on traffic volume,” *Sustainability*, vol. 9, no. 12, Art. no. 2208, Nov. 2017, doi: 10.3390/su9122208.
- [37] N. Kolban, *Kolban’s Book on ESP32*. 2017. Consulté le : 8 mai 2026. [En ligne]. Disponible sur : <http://archive.org/details/kolban-ESP32>

- [38] “Aide-mémoire des composants électroniques,” *Scribd*. Consulté le : 14 juin 2026. [En ligne]. Disponible sur : <https://www.scribd.com/document/731974496/Feuilletage>
- [39] “LDR Sensor Module Datasheet,” *Scribd*. Consulté le : 8 mai 2026. [En ligne]. Disponible sur : <https://www.scribd.com/doc/128939598/LDR-Datasheet>
- [40] “Applications du GPS NEO-6M et MPU6050,” *Scribd*. Consulté le : 14 juin 2026. [En ligne]. Disponible sur : <https://www.scribd.com/presentation/862324516/NEO-6-U-blox-6-GPS-Modules-1>
- [41] “Comment utiliser le GPS Neo-6M V2 Arduino : guide complet pour débutants,” *Moussasoft*. Consulté le : 8 mai 2026. [En ligne]. Disponible sur : <https://www.moussasoft.com/comment-utiliser-gps-neo-6m-v2-arduino-guide/>
- [42] “NEO 6 - DataSheet - (GPS.G6 HW 09005),” *Scribd*. Consulté le : 14 juin 2026. [En ligne]. Disponible sur : <https://www.scribd.com/document/220594275/NEO-6-DataSheet-GPS-G6-HW-09005-pdf>
- [43] “GPS Neo-6M V2 Module RS232 TTL Arduino Raspberry Pi,” *Arduino Maroc*. Consulté le : 8 mai 2026. [En ligne]. Disponible sur : <https://arduinomaroc.com/gps-neo-6m-v2-module-rs232-ttl-arduino-raspberry-pi/>
- [44] S. Monk, *Raspberry Pi Cookbook: Software and Hardware Problems and Solutions*. Sebastopol, CA, USA: O'Reilly Media, 2022.
- [45] N. Naik, “Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP,” in *Proc. 2017 IEEE International Systems Engineering Symposium (ISSE)*, Vienna, Austria, 2017, pp. 1–7, doi: 10.1109/SysEng.2017.8088251.
- [46] H. Megnafi, N. Boukli-Hacene, N. Ravue, and H. Baudrand, “The use of harmonic balance in wave concept iterative method for nonlinear radio frequency circuit simulation,” *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 3, pp. 303–310, 2018, doi: 10.14569/IJACSA.2018.090343.
- [47] H. Megnafi, “Frequency plan optimization based on genetic algorithms for cellular networks,” *Journal of Communications Software and Systems*, vol. 16, no. 3, pp. 217–223, 2020, doi: 10.24138/jcomss.v16i3.1012.
- [48] H. Megnafi, A. A. Chellal, and A. Benhanifia, “Flexible and automated watering system using solar energy,” in *Proc. International Conference in Artificial Intelligence in Renewable Energetic Systems*, 2020, pp. 747–755.
- [49] “MQTT in depth.” Consulté le : 8 mai 2026. [En ligne]. Disponible sur : <http://programmingwithreason.com/article-mqtt-in-depth.html>
- [50] “mqtt-v3 1 1,” *Scribd*. Consulté le : 14 juin 2026. [En ligne]. Disponible sur : <https://www.scribd.com/document/388218290/mqtt-v3-1-1-doc>

- [51] G. Abdellaoui, H. Megnafi, and F. T. Bendimerad, “A novel model using Reo for IoT self-configuration systems,” in *Proc. 2020 1st International Conference on Communications, Control Systems and Signal Processing (CCSSP)*, El Oued, Algeria, 2020, pp. 1–5, doi: 10.1109/CCSSP49278.2020.9151679.
- [52] A. A. Chellal, J. Lima, J. Gonçalves, and H. Megnafi, “Battery management system for mobile robots based on an extended Kalman filter approach,” in *Proc. 2021 29th Mediterranean Conference on Control and Automation (MED)*, 2021, pp. 1131–1136.
- [53] Guillaume, “Découverte de Node-RED, un puissant logiciel d’automatisation,” *DomoPi*, Apr. 4, 2021. Consulté le : 8 mai 2026. [En ligne]. Disponible sur : <https://www.google.com/search?q=Guillaume%2C+%C2%ABdecouverte-de-node-red-un-puissant-logiciel-d->
- [54] “Practical Node-RED Programming: Learn powerful visual programming techniques and best practices for the web and IoT,” *dokumen.pub*. Consulté le : 14 juin 2026. [En ligne]. Disponible sur : <https://dokumen.pub/practical-node-red-programming-learn-powerful-visual-programming-techniques-and-best-practices-for-the-web-and-iot-1nbsped-9781800201590.html>
- [55] “Node-RED,” *Wikipedia*. Consulté le : 8 mai 2026. [En ligne]. Disponible sur : <https://www.google.com/search?q=%C2%ABNode-RED%2C%C2%BB+%5BEn+ligne%5D.+Available%3A+https%3A%2F%2Fen.wikipedia.org%2Fwiki%2FNode->
- [56] M. Lekić and G. Gardašević, “IoT sensor integration to Node-RED platform,” in *Proc. 2018 17th International Symposium INFOTEH-JAHORINA (INFOTEH)*, East Sarajevo, Bosnia and Herzegovina, 2018, pp. 1–5, doi: 10.1109/INFOTEH.2018.8345544.
- [57] “Node-RED: Introduction,” *Stackhero*. Consulté le : 8 mai 2026. [En ligne]. Disponible sur : <https://www.google.com/search?q=%C2%ABNode->
- [58] “Node-RED Programming Essentials Guide,” *Scribd*. Consulté le : 14 juin 2026. [En ligne]. Disponible sur : <https://www.scribd.com/document/696053509/Node-Red-Book>
- [59] Automation Sense, “Node-red français,” *Automation Sense*, May 15, 2020. Consulté le : 8 mai 2026. [En ligne]. Disponible sur : https://www.google.com/search?q=+automationsense%2C+%C2%ABNode-red+fran%C3%A7ais%2C%C2%BB+Automation+Sense%2C+15+05+2020.+&sca_esv
- [60] “SQLite Documentation.” Consulté le : 20 juin 2026. [En ligne]. Disponible sur : <https://sqlite.org/docs.html>
- [61] “SQLite Documentation (Official),” *Scribd*. Consulté le : 20 juin 2026. [En ligne]. Disponible sur : <https://www.scribd.com/document/343868490/SQLite-Documentation-Official>
- [62] “Appropriate Uses for SQLite.” Consulté le : 20 juin 2026. [En ligne]. Disponible sur : <https://sqlite.org/whentouse.html>

[63] “node-red-node-sqlite,” *Node-RED*. Consulté le : 20 juin 2026. [En ligne]. Disponible sur : <http://flows.nodered.org/node/node-red-node-sqlite>

[64] “Documentation - SQL Notebook.” Consulté le : 20 juin 2026. [En ligne]. Disponible sur : <https://sqlnotebook.com/doc.html>

Annexes



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Abou Bekr Belkaid Tlemcen

Business Model Canvas

BMC

Date de dépôt : 24/06/2026

N° de projet : FT-018

Faculté/Institut : Technologie

Département : Télécommunication

Nom du projet : Système IOT de surveillance intelligence d'éclairage nocturne pour une Smart City

Encadrant 1 : Otmani Amina

Co-encadrant 1 : Elgorma Mourad

Co-encadrant 2 : Bouchenak Khelladi Yassamina

Etudiants : Ouldali Yassine

Année universitaire : 2025/2026

1- Proposition de valeur

a. Quels problèmes résolvons-nous pour nos clients ?

Notre projet propose une solution radicale aux principaux défis de la gestion de l'éclairage public urbain, actuellement confronté à un problème de maintenance à l'aveugle. À l'heure actuelle, les municipalités s'appuient sur des patrouilles nocturnes aléatoires ou sur les signalements des citoyens pour identifier les lampadaires défectueux, ce qui engendre un gaspillage considérable de temps, de carburant et de ressources humaines. Grâce à un système de surveillance en temps réel qui suit l'état de fonctionnement de chaque lampe (allumée/éteinte), nous permettons un ciblage très précis des équipes de maintenance directement depuis un panneau de contrôle centralisé (Node-RED).

De plus, contrairement aux solutions traditionnelles qui imposent des abonnements de communication coûteux (une carte SIM 4G par antenne-relais), notre architecture décentralisée basée sur la technologie ESP-WIFI-MESH offre un réseau autonome, stable et entièrement gratuit. Enfin, notre approche de fabrication et d'assemblage locaux (dans notre propre laboratoire) garantit une maintenance rapide et un équipement extrêmement résistant aux intempéries (indice de protection IP65), éliminant ainsi la dépendance à des équipements importés difficiles, voire impossibles à réparer.

b. Quels besoins de nos clients satisfont nos produits ou services ?

1-Nécessité d'un guidage précis pour les équipes de maintenance (Intervention Ciblée) :

Le besoin : Le personnel de maintenance municipal peine à envoyer des équipes en patrouille nocturne à la recherche de poteaux endommagés sans objectif précis.

Notre solution : Le système leur fournit un panneau de contrôle central (Node-RED) qui leur indique instantanément la position exacte du poteau endommagé, permettant ainsi au camion de maintenance (Nacelle) de se diriger directement vers la cible, sans approximation.

2-Nécessité de réduire les dépenses logistiques :

Le besoin : Le budget de maintenance est fortement impacté par la consommation de carburant des camions (diesel) et le temps perdu par les techniciens à rechercher les pannes de manière aléatoire.

Notre solution : En supprimant complètement la maintenance préventive, les ressources humaines et logistiques ne sont mobilisées que lorsqu'une panne réelle et confirmée est identifiée sur la carte, ce qui permet à la municipalité de réaliser d'importantes économies.

3-Le besoin d'un réseau libre et indépendant (connectivité sans abonnement) :

Le besoin : Les organisations refusent les systèmes intelligents qui les obligent à acheter des cartes SIM 4G pour chaque poteau et à payer des abonnements mensuels aux opérateurs télécoms.

Notre solution : Grâce à l'intégration de la technologie ESP-WIFI-MESH, les poteaux créent un réseau sans fil gratuit qui relaie les données d'un poteau à l'autre (de saut en saut), répondant ainsi au besoin du client d'un système fiable et connecté sans frais d'exploitation supplémentaires.

4-Le besoin d'équipements durables avec maintenance locale rapide :

Le besoin : Les clients hésitent à investir dans des équipements importés susceptibles de tomber rapidement en panne en raison des conditions climatiques et pour lesquels les pièces détachées ne sont pas facilement disponibles localement.

Notre solution : Nous assemblons et conditionnons les circuits localement selon la norme de protection IP65, répondant ainsi au besoin du client d'un produit robuste capable de résister au climat algérien, avec un service après-vente garanti et une intervention rapide et fiable.

c. En quoi notre offre est-elle différente de celle de nos concurrents ?

Notre offre se distingue de la concurrence par son caractère pratique, indépendant et économique, répondant aux besoins de notre clientèle diversifiée. Alors que les systèmes concurrents intègrent des cartes SIM 4G à chaque poteau, engendrant des abonnements mensuels onéreux, notre système repose sur une infrastructure entièrement décentralisée et gratuite utilisant la technologie ESP-WIFI-MESH qui relaie automatiquement les données entre les poteaux. De plus, nous évitons de commercialiser des appareils importés, souvent difficiles à réparer ; nous proposons un produit fabriqué et assemblé localement, certifié IP65, qui surveille avec précision l'état (marche/arrêt) de l'éclairage et envoie des alertes en temps réel via la plateforme Node-RED. Cette combinaison d'indépendance vis-à-vis du réseau, de fabrication locale et de conseils précis pour les équipes de maintenance fait de notre système l'option la plus économique, fiable et durable du marché.

d. Quelles est notre proposition unique de valeur ?

Un système de surveillance intelligent et indépendant des réseaux d'éclairage élimine totalement les coûts liés à la maintenance à l'aveugle et aux interventions aléatoires. Nous fournissons à nos clients de divers secteurs un diagnostic précis et en temps réel de l'état des lampes (allumées/éteintes) via un panneau de commande central. Ce système s'appuie sur un réseau sans fil gratuit (ESP-WIFI-MESH) sans abonnement, et sur un équipement robuste (IP65) fabriqué et pris en charge localement.

2- Segments de clients

a. Quels sont nos clients principaux ?

- Municipalités (APC) et sociétés de gestion de l'éclairage public.
- Directions des travaux publics (DTP) et nouvelles autorités de gestion urbaine.
- Administrations universitaires et résidences étudiantes.
- Administrations de zones industrielles et de grands complexes industriels.
- Promoteurs immobiliers (propriétaires de complexes résidentiels sécurisés).
- Hôpitaux, aéroports et infrastructures critiques.
- Sociétés de gestion des autoroutes.
- Sociétés de gestion des stades.
- Résidences de luxe, hôtels et grandes exploitations agricoles.
- Entreprises et usines.
- Usines de fabrication de lampes LED.

b. Quels sont les différents segments de clients que nous visons ?

- 1-Secteur public et collectivités territoriales (B2G) :
 - Municipalités (APC) et gestionnaires de l'éclairage public.
 - Directions des travaux publics (DTP) et autorités de gestion des villes nouvelles.
- 2-Secteur privé et institutions économiques (B2B) :
 - Administrations régionales et parcs industriels.
 - Promoteurs immobiliers (propriétaires de complexes résidentiels).

- Entreprises et usines.
 - Usines de fabrication de lampes LED.
 - Résidences de luxe, hôtels et grandes exploitations agricoles.
- 3-Grands équipements et infrastructures critiques :
- Administrations universitaires et résidences étudiantes.
 - Grands hôpitaux et aéroports.
 - Sociétés de gestion des autoroutes.
 - Sociétés de gestion des stades.

c. Quels sont les besoins spécifiques de chaque segment de clients ?

1. Secteur public et collectivités territoriales (municipalités et directions des travaux publics) :

-Suppression complète de la maintenance à l'aveugle et réduction des coûts de déplacement des véhicules d'intervention (carburant et temps).

-Guidage précis et direct des équipes d'intervention vers le lieu de la panne grâce à la carte du panneau de contrôle (Node-RED).

-Réseau de communication maillé gratuit et indépendant, évitant ainsi les coûts d'abonnement mensuels auprès des opérateurs de télécommunications (carte SIM 4G).

2. Secteur privé et institutions économiques (usines et promoteurs immobiliers) :

-Gestion et surveillance simplifiées des pannes dans les grands espaces clos depuis un point central (service de sécurité ou de maintenance).

-Intégration de technologies intelligentes aux projets immobiliers pour accroître leur valeur marchande.

-Accès à des équipements robustes (IP65) avec garantie d'un service après-vente local et réactif pour prévenir les arrêts de production.

3. Catégorie Infrastructures et équipements majeurs (Universités, Hôpitaux, Autoroutes) :

-Détection des pannes et intervention immédiate pour garantir un éclairage continu, pour des raisons de sécurité et d'exploitation.

-Nécessité d'un réseau de communication fiable et ininterrompu, capable de rétablir automatiquement la transmission des données (auto-réparation) en cas de chute d'un poteau.

-Consolidation de la surveillance sur un seul écran pour des zones géographiques très étendues ou linéaires (comme les autoroutes).

d. Comment pouvons-nous catégoriser nos clients en groupes distincts ?

1. Classification par secteur :

-Secteur public (B2G) : Municipalités, Directions des travaux publics.

-Secteur privé (B2B) : Zones industrielles, Promoteurs immobiliers.

-Grands équipements publics : Universités, Hôpitaux, Aéroports.

2. Classification par taille et structure du réseau :

-Réseaux ouverts et étendus : Voirie municipale, Autoroutes (Objectif principal : Localiser précisément les pannes pour une intervention optimale et une consommation de carburant réduite).

-Réseaux fermés et limités : Complexes résidentiels, Usines, Campus -universitaires (Objectif principal : Permettre une surveillance centralisée par le service de sécurité pour une intervention rapide).

3. Classification par sensibilité et rapidité d'intervention (niveau d'urgence) :

-Établissements à haute priorité (intervention immédiate) : Hôpitaux et aéroports (où une panne d'éclairage représente un risque pour la sécurité ou le fonctionnement et nécessite une intervention immédiate dès l'apparition de l'alerte dans Node-RED).

-Zones à priorité moyenne (intervention planifiée) : Rues et zones résidentielles (les pannes sont regroupées dans le panneau de contrôle et l'équipe de maintenance n'est dépêchée qu'une seule fois afin de minimiser les coûts de déplacement).

3- Relation avec les clients

a. Quel type de relation chaque segment de clients attend il de nous ?

1. Secteur public et collectivités territoriales (municipalités et directions des travaux publics) :

-Relation contractuelle à long terme (contrats de maintenance annuels et mises à niveau du système).

-Transparence totale grâce à un accès direct au panneau de contrôle (Node-RED) pour le dépannage.

-Assistance technique continue et garantie depuis le siège local de l'entreprise (SARL).

2. Secteur privé et institutions économiques (usines et promoteurs immobiliers) :

-Fourniture de solutions clés en main, pratiques et faciles à installer et à exploiter pour le client.

-Une relation fondée sur une efficacité opérationnelle éprouvée et une réduction de la charge de gestion quotidienne pour les chefs de projet et le personnel de sécurité.

3. Grands établissements et infrastructures critiques (hôpitaux, universités, aéroports) :

-Assistance technique immédiate et réactive en cas de panne du réseau (maillé).

-Formation et perfectionnement sur site pour leurs équipes de maintenance internes afin de garantir le bon fonctionnement du système.

b. Comment entretenons-nous actuellement les relations avec nos clients ?

-Réception directe et rapports périodiques via la direction.

-Service après-vente local et réactif grâce à une maîtrise totale du montage et du conditionnement.

-Assistance technique continue et diagnostic à distance des pannes via la plateforme Node-RED.

-Mises à jour régulières du micrologiciel système et de l'interface du panneau de commande.

-Mise en place de contrats de maintenance et d'assistance technique à long terme.

c. Comment pouvons-nous améliorer ou personnaliser nos interactions avec nos clients ?

-Personnalisation de l'interface de la plateforme de contrôle Node-RED avec le logo et l'identité visuelle de chaque organisation ou municipalité, ainsi que des rapports personnalisés adaptés à leurs besoins.

-Attribution d'un interlocuteur dédié à chaque client stratégique, basé au sein du service administratif et d'accueil, afin de fluidifier les échanges.

-Mise en place d'une ligne directe avec le service de programmation pour les clients disposant d'installations sensibles (comme les hôpitaux) afin de garantir une intervention immédiate.

-Organisation de visites proactives sur site par les techniciens de montage et de maintenance pour vérifier la conformité des équipements (IP65) avant tout dysfonctionnement.

-Proposition de formations personnalisées sur site pour former les équipes techniques et de sécurité du client à la surveillance en temps réel.

4-Canaux de distribution

a- Par quels canaux nos clients veulent-ils être atteints ?

-Séminaires : A destination du secteur public (collectivités territoriales, directions des travaux publics) et des grands gestionnaires d'infrastructures (hôpitaux, universités).

-Expositions : Pour permettre aux gestionnaires de zones industrielles et aux promoteurs immobiliers de découvrir le système.

-Publicité : A destination des entreprises privées et des entrepreneurs.

-Boutiques d'appareils intelligents : Pour les petits ateliers de réparation et les entrepreneurs indépendants à la recherche d'appareils prêts à l'emploi.

-Panneaux d'affichage : Pour cibler et capter l'attention des décideurs (maires et directeurs d'institutions importantes) dans des zones stratégiques.

b- Quels canaux sont les plus efficaces pour atteindre chaque segment de clients ?

-Pour le secteur public et les collectivités territoriales (municipalités et directions des travaux publics) : les séminaires constituent le canal le plus efficace, car ils offrent un cadre formel et crédible pour convaincre les commissions d'achat et présenter le système comme une solution économique réduisant les coûts de maintenance.

-Pour le secteur privé et les zones industrielles : les salons professionnels sont le canal le plus efficace, car les directeurs d'usine et les responsables de la sécurité privilégient une expérience pratique et une observation directe de l'interface Node-RED et des équipements IP65 avant de prendre une décision d'investissement.

-Pour les promoteurs immobiliers et les entreprises de construction : les boutiques d'objets connectés et la publicité sont les canaux les plus efficaces, car ils recherchent constamment sur ces plateformes des objets connectés prêts à être intégrés et à valoriser leurs projets résidentiels.

-Pour un accès stratégique aux décideurs (maires et directeurs généraux) : le canal le plus efficace est l'affichage publicitaire, surtout lorsqu'il est placé à proximité des centres administratifs pour renforcer la crédibilité et la notoriété de l'entreprise.

c- Comment pouvons-nous intégrer différents canaux pour améliorer l'expérience clients ?

Nous pouvons intégrer ces canaux pour créer un parcours client fluide et cohérent. Nous commençons par capter l'attention des décideurs grâce à des panneaux d'affichage et des publicités ciblées conçues pour éveiller leur curiosité. Ensuite, nous les invitons à des séminaires professionnels afin de leur présenter concrètement les avantages économiques du système de surveillance. Puis, nous les orientons vers des showrooms où ils peuvent découvrir le panneau

de commande et les équipements. Enfin, nous facilitons l'accès à des solutions clés en main pour les professionnels et les particuliers en les proposant directement dans les boutiques d'appareils connectés, garantissant ainsi une transition en douceur de la découverte à l'achat.

5-Partenaires clés

a. Qui sont nos partenaires clés ?

-Fournisseurs de composants et d'équipements électroniques : Fourniture de pièces essentielles (ESP32, circuits imprimés et boîtiers IP65) pour garantir un assemblage continu dans vos ateliers et bureaux.

-Entreprises de construction et d'électricité : En tant que partenaires de terrain, intégration de votre système de surveillance à leurs projets (installation de nouveaux candélabres ou modernisation des candélabres existants).

-Universités et incubateurs d'entreprises : Assistance technique et administrative dans le cadre légal des projets innovants.

-Fournisseurs de services et d'hébergement : Hébergement local sécurisé et stable du panneau de contrôle central (Node-RED), accessible aux clients 24 h/24 et 7 j/7.

-Bureaux d'études : En tant que partenaires stratégiques, intégration de votre technologie comme exigence fondamentale dans les cahiers des charges des nouveaux projets et ensembles résidentiels.

b. Quels sont les partenariats qui nous aident à réduire les coûts, à accéder à de nouvelles ressources ou à améliorer notre proposition de valeur ?

-Partenariats pour réduire les coûts : Nous concluons des contrats directs avec des importateurs de composants électroniques et des fabricants de boîtiers de protection afin de vous fournir des matières premières à prix de gros pour votre atelier d'assemblage, ce qui réduit considérablement vos coûts de production.

-Partenariats pour accéder à de nouvelles ressources : Nous tirons parti des partenariats universitaires dans le cadre des dispositifs ministériels de soutien à l'entrepreneuriat (tels que le Projet 1275 dans le domaine de l'environnement universitaire) pour accéder gratuitement à des laboratoires de recherche, à des équipements de test de pointe et à un accompagnement technique.

-Partenariats pour optimiser notre proposition de valeur : Nous nouons des alliances avec des entreprises d'électricité et de travaux publics afin de proposer notre système comme solution clé en main pour leurs projets, et nous collaborons avec des fournisseurs d'hébergement locaux pour garantir une stabilité et une rapidité maximales du panneau de contrôle.

c. Comment pouvons-nous aligner nos intérêts avec ceux de nos partenaires ?

-Aux entreprises d'électricité et de construction : nous leur offrons un avantage concurrentiel pour remporter des appels d'offres en leur fournissant une solution d'éclairage intelligente et indépendante, en échange de la commercialisation et de l'installation de notre système auprès de leurs clients.

-Aux fournisseurs de composants électroniques : nous leur garantissons un volume de ventes stable et des commandes régulières, en échange d'un approvisionnement prioritaire et de tarifs de gros préférentiels pour votre SARL, permettant de réduire les coûts de production.

-Aux bureaux d'études : nous leur fournissons des solutions innovantes et économiques pour améliorer la qualité de leurs conceptions, en échange de l'intégration de votre technologie comme critère essentiel dans les cahiers des charges des nouveaux projets.

-Aux hébergeurs locaux : nous leur proposons des contrats d'abonnement à long terme, en échange de leur engagement à assurer la stabilité et la disponibilité des serveurs Node-RED et du panneau de contrôle 24h/24 et 7j/7.

6-Activités clés

a. Quelles sont les actions principales que nous devons entreprendre pour livrer notre proposition de valeur ?

1. Activités de recherche, développement et programmation :

-Développement et maintenance du firmware ESP32 pour garantir la stabilité du réseau Wi-Fi mesh.

-Gestion et mise à jour de la plateforme de contrôle centrale (Node-RED) pour fournir des rapports précis et une interface utilisateur conviviale.

-Réalisation de tests réguliers sur le terrain pour garantir l'efficacité des alertes en temps réel et la détection rapide des pannes.

2. Activités de fabrication et d'assemblage (Aspects techniques) :

-Assemblage et installation des composants électroniques dans des boîtiers de protection IP65 au sein de l'atelier de l'entreprise.

-Réalisation de contrôles et de tests de qualité pour chaque appareil avant expédition au client afin d'en garantir la fiabilité.

-Gestion des stocks de composants électroniques pour assurer la disponibilité des matières premières et la continuité de la production.

3. Activités de marketing et de relations stratégiques :

-Fournir une assistance technique sur site et après installation aux clients afin de garantir la continuité du service.

-Participer activement à des salons professionnels et des séminaires pour présenter aux clients les avantages du système.

-Établir et développer des relations durables avec les bureaux d'études et les entreprises de construction afin d'intégrer le système dans les nouveaux projets.

b. Quelles sont les opérations essentielles pour notre entreprise ?

1-Opérations administratives et commerciales (Administration et accueil) :

-Accueil des clients (représentants municipaux, directeurs d'usine et distributeurs) et organisation de réunions d'affaires.

-Traitement des contrats, gestion des factures et suivi des appels d'offres et des transactions.

2-Programmation et opérations systèmes (Bureau de développement) :

-Programmation des puces ESP32 et configuration des réseaux Wi-Fi maillés.

-Conception, personnalisation et maintenance des panneaux de contrôle centraux Node-RED des clients.

3-Opérations de fabrication (Assemblage et contrôle qualité) :

-Assemblage manuel et local de composants électroniques dans des boîtiers de protection (IP65).

-Réalisation de tests techniques rigoureux sur chaque appareil avant sa mise sur le marché.

4-Logistique et service après-vente (entrepôt et assistance technique) :

-Gestion des stocks de matières premières (puces, câbles, boîtiers) et stockage des équipements prêts à être expédiés.

-Suivi à distance des réclamations clients et coordination des interventions des équipes de maintenance sur site.

c.Quelles sont les activités qui créent le plus de valeur pour nos clients ?

-Personnalisation et développement de l'interface Node-RED : Fournir des diagnostics géographiques précis en temps réel, permettant ainsi aux clients d'économiser les coûts liés à la maintenance à l'aveugle (carburant et main-d'œuvre).

-Maintenance et mise à niveau du réseau ESP-WIFI-MESH : Garantir une transmission de données continue, gratuite et fiable (auto-réparation), et supprimer les frais d'abonnement mensuels auprès des opérateurs de télécommunications.

-Assemblage rigoureux des équipements en local : Réaliser des tests de qualité IP65 afin de garantir une durabilité et une fiabilité maximales quelles que soient les conditions climatiques, offrant ainsi une alternative pratique aux équipements importés, souvent difficiles à réparer.

-Fourniture d'un service après-vente local et rapide : Proposer des formations sur site aux équipes de sécurité et de maintenance du client (en usine ou à l'hôpital) afin de garantir une maîtrise optimale du système.

7- Ressources clés

a.Quels sont nos actifs matériels, immatériels et humains essentiels ?

1.Actifs matériels :

-Siège social (40-50 m²) équipé et divisé en quatre bureaux fonctionnels (accueil, développement, assemblage et entrepôt).

-Équipements d'atelier d'assemblage et de mesure (machines à souder, ordinateurs et appareils de test).

-Stocks de matières premières et de composants électroniques (puces ESP32, cartes de circuits imprimés, LDR, GPS NEO 6-M et boîtiers IP65).

2. Ressources humaines :

-L'équipe de direction, composée des associés fondateurs : le directeur général (PDG) pour les opérations et le directeur technique (CTO) pour le développement.

-Techniciens et ouvriers qualifiés chargés des opérations d'assemblage en atelier et de l'installation sur site.

3. Actifs immatériels et intellectuels :

-Expertise technique et logicielle en ingénierie de réseaux (ESP-WIFI-MESH) et personnalisation de panneaux de contrôle (Node-RED).

-Statut juridique de la société (SARL) et crédibilité qu'il confère pour traiter avec les grandes institutions et participer aux appels d'offres publics.

-Un réseau de relations et de partenariats stratégiques (avec des entreprises d'électricité, des bureaux d'études et des fournisseurs).

b. Quels sont les outils, les technologies ou les partenariats dont nous avons besoin pour réussir ?

1. Technologies et outils logiciels :

-Environnements de développement intégrés (tels que l'IDE Arduino) pour la programmation et la configuration des puces ESP32.

-Protocoles de communication légers (tels que MQTT) et serveurs Mosquitto pour garantir une transmission de données rapide et stable.

-Plateforme Node-RED pour la conception et la personnalisation de panneaux de contrôle interactifs.

2. Matériel et équipements :

-Équipements de soudage et de mesure de précision pour l'assemblage des circuits dans l'atelier de l'entreprise.

-Composants IoT essentiels (ESP32, capteurs de lumière LDR, modules GPS Raspberry Pi).

-Boîtiers conformes à la norme IP65.

3. Partenariats stratégiques :

-Les incubateurs universitaires et la communauté académique bénéficieront d'un soutien continu, de laboratoires de recherche et d'infrastructures administratives pour leurs projets innovants.

-Les fournisseurs de composants électroniques (importateurs et grossistes) garantiront un approvisionnement continu et réduiront les coûts de production.

-Les entreprises de travaux publics et d'électricité intégreront le système sur le terrain et élargiront leur clientèle.

c. Quels sont les principaux avantages concurrentiels de nos ressources ?

-Maîtrise technologique indépendante (Ressources intellectuelles et humaines) : L'expertise technique pointue de l'équipe dirigeante en matière de maîtrise des logiciels ESP32 et MQTT et de développement de Node-RED vous permet de vous affranchir des importations de solutions complexes et difficiles à modifier. Vous avez ainsi la possibilité de personnaliser le logiciel système selon les besoins précis de chaque client.

-Flexibilité de production et qualité de service (Ressources matérielles) : L'organisation fonctionnelle rigoureuse du siège social (sur une surface de 40 à 50 m²), comprenant un bureau dédié à l'assemblage et au contrôle qualité, garantit une maîtrise totale de la norme de protection IP65 et assure un service après-vente local, immédiat et garanti, surpassant ainsi les performances des produits importés.

-Fiabilité et crédibilité institutionnelles (Ressources juridiques) : S'appuyer sur une structure organisationnelle solide, telle qu'une SARL, confère un poids et un professionnalisme considérables à votre portefeuille d'activités, vous offrant ainsi l'avantage concurrentiel nécessaire pour remporter des appels d'offres publics et convaincre les dirigeants des grandes institutions.

-Accès aux environnements de recherche et d'innovation (Ressources de partenariat) : Tirer parti de la participation aux programmes d'innovation universitaires permet à l'entreprise d'accéder stratégiquement à des laboratoires de recherche, des environnements expérimentaux et un encadrement académique de pointe pour le développement de produits, sans engager de dépenses importantes en R&D.

8- Charges et coûts

a. Quels sont les coûts fixes et variables associés à notre modèle économique ?

1. Charges fixes :

- Loyer des locaux (40 à 50 m²)
- Matériel : mobilier de bureau, établis, ordinateurs, matériel de soudage et instruments de mesure de précision.
- Charges courantes (internet haut débit, électricité et eau) pour le fonctionnement de l'atelier et des bureaux.
- Véhicule de service
- Frais administratifs, frais de comptabilité et charges sociales liées au statut juridique de la société (SARL).

Coûts variables :

- Coût d'acquisition des matières premières et des composants électroniques (puces ESP32, capteurs, câbles et boîtiers de protection IP65).
- Salaires des techniciens et ouvriers chargés de l'assemblage et de l'installation sur site.
- Frais d'expédition, de transport et de logistique pour la livraison des équipements finis chez les clients.
- Budget de la campagne marketing.
- Coûts des interventions sur site pour le service après-vente.

b. Quels sont les coûts les plus importants pour notre entreprise ?

Les coûts les plus importants et les plus sensibles de notre entreprise concernent l'acquisition de composants électroniques et de matières premières essentielles à la production, comme les puces ESP32 et les boîtiers de protection IP65. À cela s'ajoutent les coûts de main-d'œuvre et d'expertise technique, notamment les salaires des équipes d'assemblage et de gestion, indispensables pour garantir la qualité et la fiabilité de nos systèmes.

Par ailleurs, nous devons couvrir les frais fixes de fonctionnement et d'équipement de notre siège social, dont la superficie est limitée (40 à 50 mètres carrés), afin d'assurer l'efficacité opérationnelle de nos quatre bureaux. Enfin, le marketing et l'acquisition de clients par le biais de salons, de séminaires et de panneaux publicitaires représentent une part importante de nos dépenses, compte tenu de leur rôle crucial pour convaincre les décideurs et les grandes institutions.

c. Comment pouvons-nous réduire les coûts ou améliorer l'efficacité de nos opérations ?

Pour réduire efficacement les coûts, privilégiez l'achat en gros de composants électroniques (comme les puces ESP32) afin de bénéficier de prix avantageux. Optimisez l'utilisation de l'espace de l'entreprise (40 à 50 m²) en adoptant un stockage vertical pour éviter la location d'entrepôts supplémentaires. De plus, le recours aux laboratoires universitaires pour les tests de qualité et une approche marketing ciblée auprès des décideurs, plutôt qu'une publicité coûteuse et non ciblée, permettront de réduire significativement les budgets d'exploitation.

Pour améliorer l'efficacité opérationnelle, standardisez les étapes d'assemblage en atelier afin d'accélérer le travail et de minimiser les erreurs techniques. Côté logiciel, la création de

modèles prédéfinis pour la plateforme Node-RED facilitera la personnalisation des interfaces pour les nouveaux clients sans nécessiter de reprogrammation systématique.

9- Revenus

a. Quels produits ou services nos clients sont-ils prêts à payer ?

-Les clients sont principalement disposés à payer pour le matériel et les équipements, notamment les modules d'éclairage intelligents assemblés localement (puces ESP32 dans des boîtiers IP65). Ce produit tangible représente le premier investissement réalisé par un client (qu'il s'agisse d'une municipalité ou d'une usine) pour moderniser son infrastructure.

-De plus, les clients paient pour les logiciels et les services d'installation, qui comprennent l'intégration du système aux réseaux d'éclairage existants ou nouveaux et la fourniture d'une interface de contrôle centrale personnalisée (Node-RED). La valeur ajoutée réside dans la fourniture d'un système autonome basé sur un réseau Wi-Fi, libérant ainsi le client des contraintes liées aux abonnements mensuels auprès des opérateurs de télécommunications.

-Les contrats de maintenance et de service après-vente (SAV) constituent un service essentiel et très rentable. Les grands clients sont disposés à payer des frais périodiques pour des contrats étendus garantissant une intervention rapide sur site, des diagnostics à distance et la formation de leurs équipes techniques et de sécurité afin d'assurer le fonctionnement efficace et durable de leurs systèmes.

b. Quels sont les différents moyens par lesquels nous pouvons générer des revenus ?

-Vente directe d'appareils : par la fourniture de modules d'éclairage intelligents entièrement équipés et prêts à l'emploi.

-Frais d'installation et de mise en service : paiement unique couvrant l'intervention sur site pour la programmation du réseau de communication indépendant et la personnalisation du panneau de commande central pour chaque client.

-Contrats de maintenance périodiques : garantissant à l'entreprise un flux de trésorerie continu et stable en contrepartie de la fourniture de services après-vente, d'une maintenance proactive et d'une intervention technique rapide.

c. Quel est notre modèle de tarification ?

Notre modèle de tarification repose sur un modèle hybride, conçu pour encourager l'investissement initial tout en garantissant la pérennité des revenus. Nous proposons d'abord un prix fixe pour la vente des équipements et les services d'installation (un paiement unique), déterminé par la taille du projet et le nombre de points optiques à installer. Ensuite, nous proposons une tarification contractuelle via des contrats de maintenance annuels incluant le support technique et les mises à jour des panneaux de contrôle. Notre principal avantage concurrentiel réside dans l'absence totale de frais mensuels auprès des opérateurs télécoms, grâce à notre réseau indépendant.

YassloT

Partenaires clés

- Fabricants des cartes PCB
- Fabricants des boîtiers IP65
- Fournisseurs des composants (ESP32, GPS, ct)
- Fabricants des lampes
- Universités de Abou bakr Belkaïd– Tlemcen – (Labo de recherche STIC) et incubateurs d'entreprises

Activités clés

- Fabrication
- Vente
- Après-vente.

Ressources clés

- Auto financement
- 2 travailleurs
- Matériels : 2 ordinateurs, fer à souder (2), multimètre (3), ESP32, LDR, Résistance, Raspberry Pi, GPS NEO 6M

Proposition de valeur

- Un système de surveillance technique et en temps réels des lampes
- Un système d'inspection des lampes défectueuses
- Un système qui repérez les lampes défectueuses
- Un système qui permet une faciliter à l'accès à la réparation des lampes cassées

Relation clients

- Services après ventes
- Service d'entretien
- Garantie d'un an
- Fournir des mises à jour aux clients

Canaux de distribution

- Publicités
- Expositions
- Magasins d'appareils intelligents
- Séminaire

Segment client

- Municipalités
- Directions des travaux publics
- Administrations universitaires
- Administrations de zones industrielles
- Promoteurs immobiliers.
- Hôpitaux, aéroports et infrastructures critiques.
- Sociétés de gestion des autoroutes.
- Sociétés de gestion des stades.
- Résidences de luxe, hôtels et grandes exploitations agricoles.
- Usines de fabrication de lampes LED.

Coûts

L'équipement : Ordinateur (*2) =80,000DA, Fer à souder (*2) =22,000DA, Multimètre (*3) =9000DA, Travailleur (*2) =720,000DA/ans **Total** = 831,000DA
Matière première : ESP32=260DA, LDR=5DA, Résistance=5DA, Raspberry Pi=1,5000DA, GPS NEO 6M=400DA **Total** = 1,5670DA

Revenus

Ventes : Prix une pièce 700DA, Prix de fabrication 300DA
6 mois = 1000 pièces = 70,000DA – Décharge = nulle
1 ans = 4000 pièces =2,800,000DA – Décharge = 1,000,000DA
3ans = 8000 pièces =5,600,000DA - Décharge = 2,820,000DA
Installation = pour le projet 150,000DA
Maintenance=50,000 par 6 mois pour un seul projet

رقم: FT-018 / الحاضنة / 2026

شهادة توظيف / تحضين مشروع مبتكر

أنا الممضي أسفله، السيد: سليمان سفيان

مدير حاضنة الأعمال: جامعة أبي بكر بلقايد تلمسان

المقر الاجتماعي / العنوان: 22، شارع أبي عباد عبد الكريم، نهج باستور، ص.ب 119، تلمسان

رقم علامة الحاضنة: 0809213030

تاريخ تسليم العلامة: 2021 / 09 / 09

أشهد أن حامل (ة) المشروع:

الاسم واللقب	تاريخ ومكان الإزدياد	رقم تسجيل الطالب	الطور الدراسي	التخصص	الكلية
ولد علي ياسين	2002/12/27 بمغنية	202037019214	ماستر 2	اتصالات عبر الألياف البصرية	تكنولوجيا

تحت إشراف الأساتذة:

الاسم واللقب	الرتبة	التخصص	الكلية
عثماني أمينة	أستاذ محاضر ب	اتصالات السلكية و اللاسلكية	تكنولوجيا
بوشناق خلادي ياسمين	أستاذ محاضر ب	اتصالات سلكية و اللاسلكية	تكنولوجيا
القرمة مراد	دكتور	شبكات و أنظمة الاتصالات السلكية و اللاسلكية	تكنولوجيا

تم إحتضانه (ا) على مستوى حاضنة الأعمال لجامعة أبي بكر بلقايد تلمسان، بمشروع تحت إسم:

Système IoT de surveillance intelligence de l'éclairage nocturne pour une smart City (YassIoT)

سُلمت هذه الشهادة بطلب من المعني للإدلاء بها في حدود ما يسمح به القانون.

حرر في تلمسان بتاريخ 14 جوان 2026

مدير الحاضنة



043.41.11.89

incubator@univ-tlemcen.dz

univ-tlemcen.dz

Fg Pasteur B.P 119 13000, Tlemcen, Algérie

