Universidade do Minho
Escola de Engenharia

Smart City

Master in Industrial Eletronics and Computers Engineering

Embedded Systems

Authors:
Diogo Fernandes PG47150
José Tomás Abreu PG47386

Supervisors:
Prof. Dr. Tiago Gomes
Prof. Ricardo Roriz
Prof. Sérgio Pereira

December 5, 2021

Contents

List of Figures

List of Tables

Acronyms

API Application Programming Interface

CAGR Compound Annual Growth Rate

CPS Cyber-Physical System

CSI Camera Serial Interface

GPIO General Purpose Input/ Output

GPS Global Positioning System

GUI Graphical User Interface

IEEE Institute of Electrical and Electronics Engineers

IoT Internet of Things

ISM Industrial Scientific and Medical

LDR Light Dependant Resistor

LED Light-Emitting Diode

LPWA Low Power Wide Area

LPWAN Low Power Wide Area Network

LTE-M Long Term Evolution for Machines

NB-IoT Narrow-Band IoT

PIR Passive Infrared

PWM Pulse Width Modulation

RF Radio Frequency

SPI Serial Peripheral Interface

UNB Ultra-Narrow Band

Chapter 1

Introduction

1.1 Problem Statement

Nowadays, the energy crisis is a constant theme because of the inflated energy prices [2]. Furthermore, huge energy consumption is a burden to the environment, as not all means of energy production are non-polluting. According to "Our World in Data"[3], in 2019, 63,3 % of eletrical energy production comes from fossil fuels. It is known that generally, street lamps are continuously switched on at night, most of the time unnecessarily glowing with its full intensity, in the absence of any activities in the street, leading to a great waste of energy. Furthermore, it is in cities where the consequences of using cars are most noticeable. An example of this is the search for a parking space. According to the RAC Foundation [4], in England, an average car is parked 95 % of the time, which explains how hard it can get sometimes when trying to find a parking spot. This struggle leads to an increase in carbon dioxide production as well as fuel and energy consumption.

With that in mind, this project aims the implementation of applications for a Smart City, regarding Smart Lighting and Smart Parking, in order to decrease the energy consumption in public streets, while improving the lives of citizens around the world. The solution will embrace a centralized system, composed by smart street lights capable of turning on only when they detect movement in the surroundings, at night time, and also, capable of detecting available parking spaces in the street post vicinity.

1.2 Problem Statement Analysis

This solution provides a network of street lamp posts, each implementing Smart Street Lighting and Smart Parking Detection, using Raspberry Pi 4B [5] has a controller. A gateway is needed to gather all the information from the street lamp posts, and store that in a remote system, needed to provide a way for a responsible entity manage the network.

When there is no activity detected in the area, the lamp post is at a predefined minimum light level, whereas when a car or pedestrian is noticed in the area, the light automatically activates at full brightness. To allow to dynamically turn on the lights of the following poles, each the street lamp post communicates with the neighbor lamp posts, indirectly, through the gateway. To detect movement in the vicinity of the pole, a motion detector is used. Since the lamppost will only light up during the night time, the motion detector will also only work during that period. To ensure this, a luminosity sensor is used, determining the ambient light conditions. In order to facilitate the maintenance of the pole, a system that determines the operating conditions of the lamp is also implemented. When this system verifies that the lamp is not in good working conditions, in other words, that it is broken or burnt, this information is transmitted to the entity responsible for the network of lamp posts, through a mobile app. This is also used by the person in charge, to manage all information on the pole network, such as the location and working conditions of each pole.

In order to detect empty parking spots, this system should only be used in an area where there are parking spaces nearby. For this, the lamp post has a camera, turned on all day, and, after Raspberry Pi processes the acquired information, it will be available on a website, so that a user, a car driver, can know where there are empty parking spaces.

Chapter 2

Market Research

2.1 Market Definition

As figure 2.1 shows, there are various applications of IoT technology for smart cities. In this project it will be created a solution that comprises Smart Lighting management and Smart Parking.

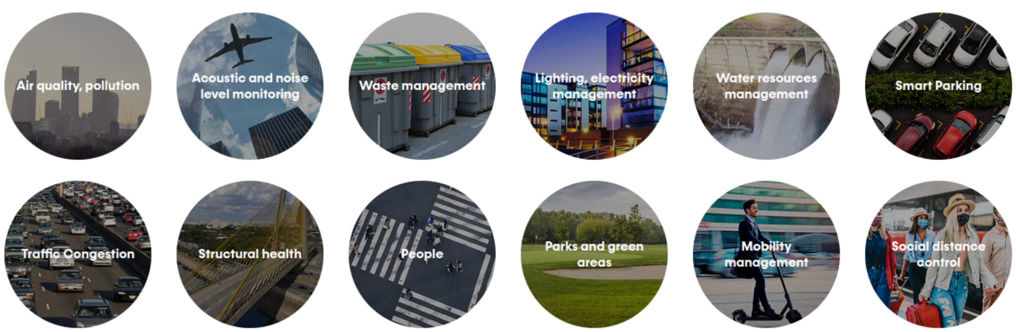

Figure 2.1: Applications of IoT technology for Smart Cities. [1]

The IoT starts with connectivity, but since it is a widely diverse and multifaceted realm, one certainly cannot find a one-size-fits-all communication solution. Next, one can identify these common types of IoT wireless technologies, through mesh and star topologies:

- **LoRaWAN** (LoRa from "long range") is a Low Power Wide Area Network (LPWAN) specification that targets key requirements of IoT, such as secure bi-directional communication. LoRaWAN network architecture is deployed in a star-of-stars topology in which gateways relay messages between end-devices and a central network server. The gateways are connected to the network server via standard IP connections and act as a transparent bridge, simply converting Radio Frequency (RF) packets to IP packets and vice versa. The wireless communication takes advantage of the Long Range characteristics of the LoRa physical layer, allowing a single-hop link between the end-device and one or many gateways. [6]

- **Wi-SUN** (Institute of Electrical and Electronics Engineers (IEEE) standard 802.15.4g) is a RF mesh communication technology, which enables large-scale outdoor IoT networks including applications such as asset management, environmental monitoring, agriculture, structural health monitoring and much more. [7] Using the same IEEE standard, there is also **ZigBee**, a short-range, low-power, commonly deployed in mesh topology to extend coverage by relaying sensor data over multiple sensor nodes. [8]

- **Sigfox** is a cellular style communication technology that provides low power, low data rate and low communication costs for IoT applications. Sigfox employs Ultra-Narrow Band (UNB) technology, which enables very low transmitter power levels to be used while still being able to maintain a robust data connection, using unlicensed Industrial Scientific and Medical (ISM) radio bands. The simple and easy to roll-out star-based cell infrastructure has encouraged its current extended worldwide availability. [9]

- **Narrow-Band IoT (NB-IoT)** is a carrier-grade RF, narrowband communication technology, specially designed for the IoT. It connects devices more simply and effciently on already established mobile networks, and handles small amounts of infrequent 2-way data, securely and reliably. The special focus of this standard is on very low power consumption, excellent penetration coverage and lower component costs, deployed in GSM and LTE regulated frequencies. [10]

- **Long Term Evolution for Machines (LTE-M)** is a Low Power Wide Area (LPWA) technology standard published by 3GPP. It supports IoT through lower device complexity and extended coverage, while allowing the reuse of the LTE installed base. Supported by all major mobile equipment, chipset and module manufacturers, LTE-M networks will co-exist with 2G, 3G, and 4G mobile networks and benefit from all the security and privacy features of carrier-grade networks. [11]

2.1.1 Smart Lighting

Smart Street lighting is a rapidly growing lighting market, with an expected Compound Annual Growth Rate (CAGR) of 20.4 % until 2026 [12], implementing a smart management of public lighting to optimize energy consumption according to lighting needs. This is boosted by regulatory policies that encourage energy efficiency, IoT convergence and the drop of Light-Emitting Diode (LED) prices. This new concept of smart light post is also growing, implementing not only the smart management of street lights, but also features that go from basic LED replacement control, to traffic and video monitoring, environmental monitoring, and others.

2.1.1.1 FLASHNET - inteliLIGHT

FLASHNET is a company focused on developing intelligent systems for smarter cities and better infrastructures and have created a solution that provides the right amount of light where and when needed to lighten the streets, the inteliLIGHT. [13] Using the existing infrastructure, this solution saves money and transforms the existing distribution level network into an intelligent infrastructure of the future.

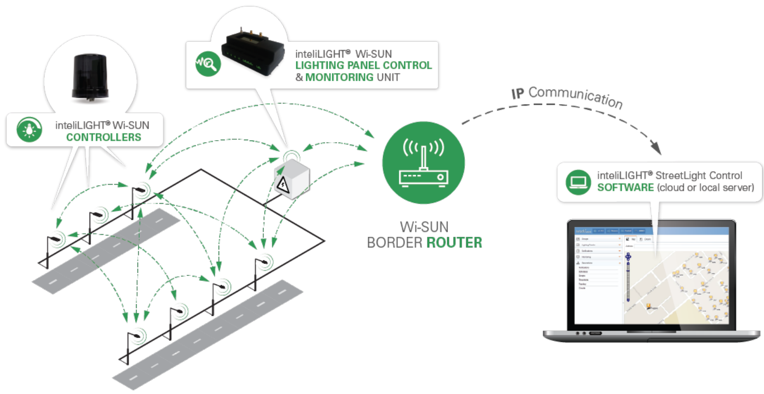

Figure 2.2: inteliLIGHT Communication Technology.

In figure 2.2 it is presented one of the many communication technologies that inteliLIGHT can provide in their smart street light solution. In this case, it is shown the use of Wi-SUN, a RF mesh street lighting communication technology. Furthermore, the system is integrated with major IoT platforms and provides Application Programming Interface (API) connectivity with City Management applications, ensuring compatibility with existing smart lighting and smart city initiatives.

2.1.1.2 Telensa - PLANet

Nowadays, Telensa is the market share leader in smart street lighting with more than ten years of experience.[14] PLANet is connected street lighting system that consists of wireless control nodes, an UNB wireless network and a Central Management System, as seen in figure 2.3. This system reduces energy and maintenance costs associated with street lighting and also improves quality of maintenance through automatic fault reporting.

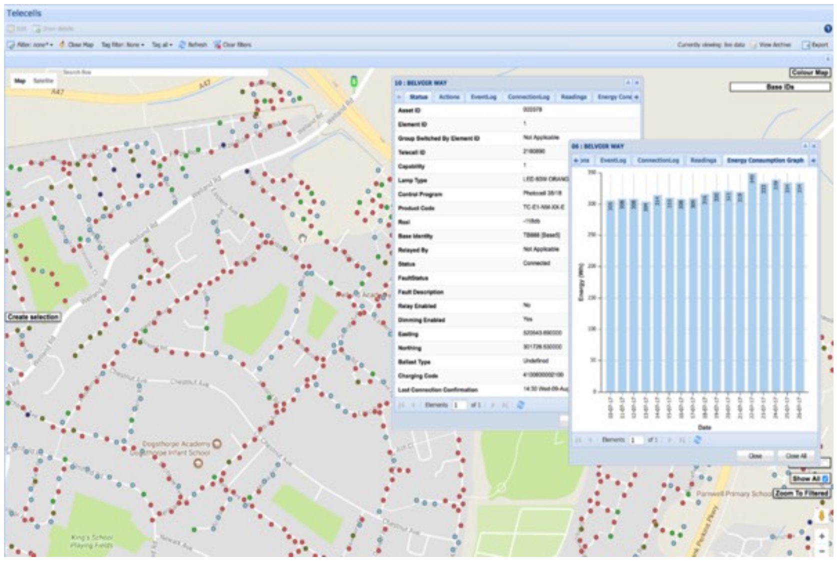

Figure 2.3: Telecells - PLANet's Central Management System.

2.1.2 Smart Parking

Smart parking, through the monitoring of parking spaces availability in the city, is also a growing market, expected to grow with a CAGR of 17.85% in the forecast period of 2021 to 2028.[15] The rise in investment in building driverless vehicles and an increase in the government's initiative in building smart cities across the globe, along with the demand and adoption of IoT technology, are the main driving factors for the growth of smart parking market.

2.1.2.1 intuVision - intuVision VA Parking

Regarding only to the detection of available parking spaces, there is a solution, by intuVision, named intuVision VA Parking, which provides parking lot analytics to determine vehicle count and security, and monitor parking space availability at all times, both for cities and for private parking lots, as one can see in the figure 2.4.[16]

Figure 2.4: intuVision Parking Lot Demonstration.

2.2 Why choose our product

This product aims to decrease power consumption associated with the traditional street light network, and also, using that infrastructure, contribute to the development of a smart city, detecting available parking spaces in the streets. This street lighting solution can be used in residential areas, public spaces or a large outdoor parking lot, feasible of being installed in existent lamp posts, requiring minimum changes to the original infrastructure. Although in this project it is not implemented, aside the parking spaces availability detection, this product can have the ability to monitor and to process various areas of interest using the camera built in, like for example, security purposes.

Chapter 3

System

3.1 Network Architecture

To define the network architecture of the solution to be created, some aspects must be remembered. One is that there are various communication technologies that may be used, as presented previously in Market Research. Other important aspect to keep in mind is that this solution implements both Smart Street Lighting and Smart Parking, through the use of street lampposts. So, these must have parking spots nearby, in order to allow full use of the Smart Parking feature. This lack of flexibility demands a creation of a network with nodes that may be far apart. Besides that, the data stream in the network will be very low since each lamppost will only communicate notifications on it's state. That is, if the lamp is light up, if it was detected a malfunction with the lamp, if it was detected an available parking space.

In figure 3.1, one can see that LoRa is ideal for applications that transmit small chunks of data with low bit rates. Data can be transmitted at a longer range compared to technologies like WiFi, Bluetooth, ZigBee or cellular communication technologies like Sigfox or LTE-M, as presented previously. These features make LoRa well suited for sensors and actuators that operate in low power mode. LoRa can be operated on the ISM license free sub-gigahertz bands, for example, 915 MHz, 868 MHz, and 433 MHz. It also can be operated on 2,4 GHz to achieve higher data rates compared to sub-gigahertz bands, at the cost of range. [?]

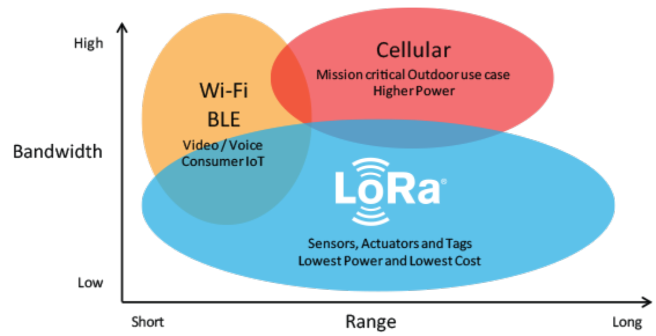

Figure 3.1: Communication technologies range vs bandwidth.

With that in mind, one can identify LoRa as a proper communication technology to use in this network.

In figure 3.2 one can see the network architecture diagram. This is a star topology, in which the gateway relay messages between each local system (lamppost) and a central network server, the remote system. This wireless communication takes advantage of the Long Range characteristics of the LoRa physical layer, allowing a single-hop link between the local system and the gateway. All communication modes are capable of bi-directional communication, and there is support for multicast addressing groups. The gateway is connected to the internet in order to store new information about the network in the remote system.

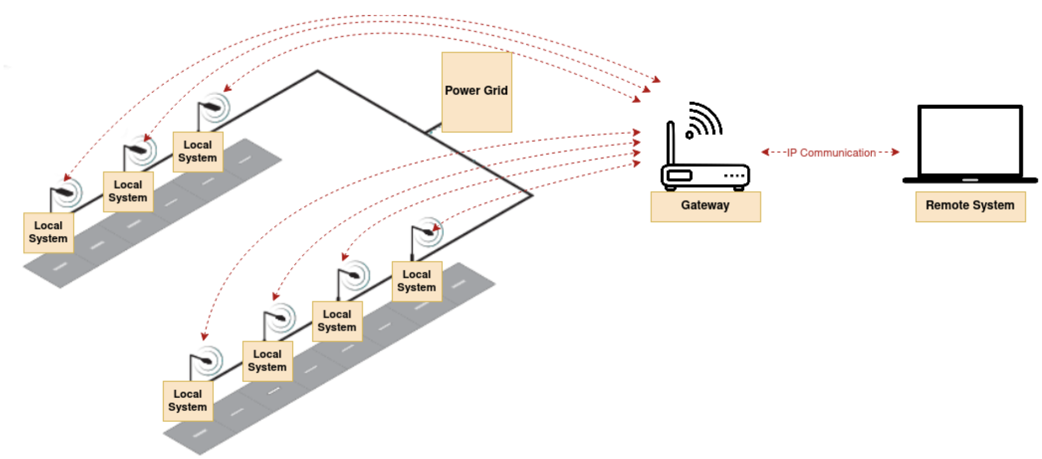

Figure 3.2: Network architecture.

LoRa end-devices serve different applications and have different requirements, that's why there are device classes, as one can see in figure 3.3.

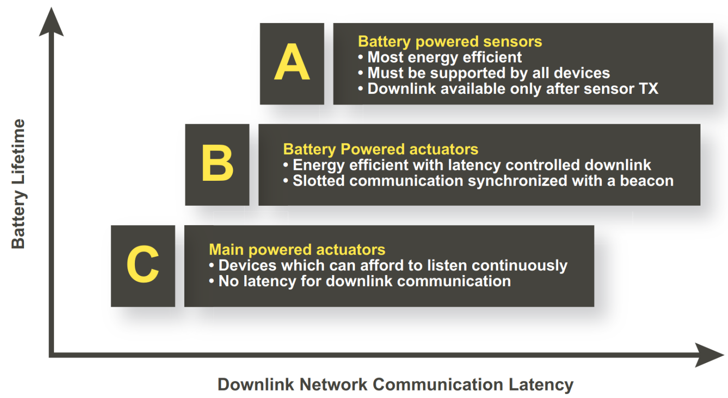

Figure 3.3: LoRa device classes.

Knowing that street lampposts have main power, from the power grid, one can classify the network nodes as class C. End-devices of class C are bi-directional end-devices with maximal receive slots, has they have almost continuously open receive windows, only closed when transmitting. That way, the downlink communication latency may be very low. [**?**]

LoRa provides long range communication, as LoRaWAN gateways can transmit and receive signals over a distance of over 10 kilometers in rural areas and up to 3 kilometers in dense urban areas. It uses license free spectrum, so one doesn't have to pay expensive frequency spectrum license fees to deploy a LoRaWAN network. It is low cost, since it is a minimal infrastructure, low-cost end nodes and open source software.

To determine the maximum number of nodes that can be connected to a single gateway, one needs to evaluate gateway specifications, more specifically, the number of packets it can support. For instance, if there is a gateway supporting 1 Million packets per day, and if the application sends 10 packet per hour, or 240 packets per day, then, more than 4000 nodes can be handled by that gateway.

3.2 System Overview

Through the system overview diagram, in figure 3.4, it is possible to identify the main modules of the system to be developed, and how they interact. We can divide the system into three subsystems: the local system, which represents a lamppost, the gateway system, device connected to the lampposts network and to the remote server, and the remote system, that stores information about the lamppost network and allows interaction with the system users by remote client applications.

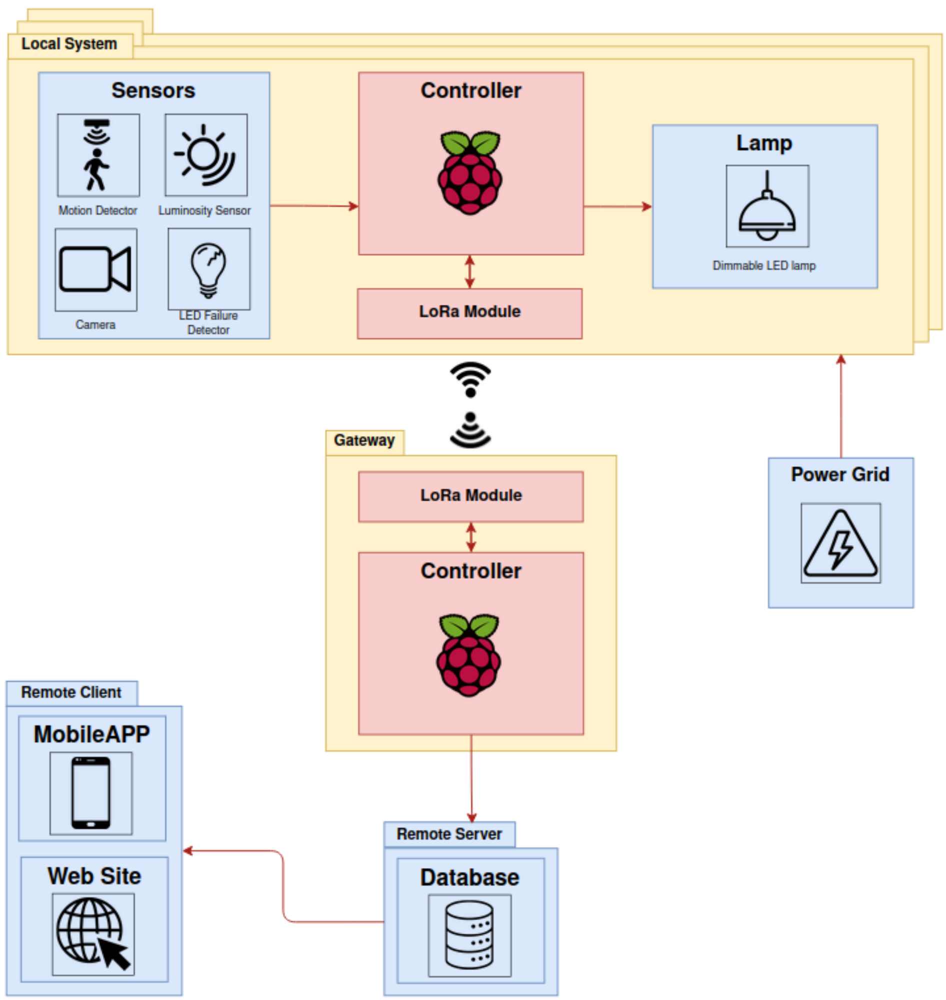

Figure 3.4: System Overview Diagram.

The local system is composed of sensors, a controller, a lamp and a wireless communication module. Regarding the sensors, there will be a motion detector, to allow the detection of movement in the vicinity of the pole, a luminosity sensor, to detect the light conditions of the pole's surroundings, a lamp failure detector to know if the lamp is working and a camera to detect empty parking spots in the lamppost vicinity. The controller of the local system is a Raspberry Pi, that use the sensors information and communicate wirelessly with the gateway using a LoRa communication module, so that the lights are dinamically turned on. The gateway controller also communicates through internet with a remote server.

The remote system is composed by the remote server and the remote client. The remote server consists of a database that stores all information about each lamp post location and operating status. This information can be accessed through a mobile application by the operator in order to carry out the necessary maintenance of the lamp of each pole. Furthermore, the operator, when installing a new lamppost, can add its location to the database, using the mobile application. In addition, the database stores information on available parking spaces detected by the camera. When a user, a car driver, wants to know where there are empty parking places, he can access a website that informs him of the location of the empty parking spaces.

Knowing that the public lighting network is directly related to the electrical network, this will be used to power each local system.

3.3 System Requirements and Constraints

In order for the system to have the desired performance, these requirements and constraints must be respected:

3.3.0.1 Functional Requirements

- Sensors data acquisition
- Motion detection
- Control of a street lamp
- Control a network of street poles
- Wireless communication between local systems and base station

- Access system information through a mobile application
- Empty parking spots detection
- Manage system information through a mobile application
- Add lamp post location through a mobile application
- Access available parking spots location through a web site

3.3.0.2 Non-Functional Requirements

- User friendly mobile application and web site
- Ambient luminosity sensing
- Lower power consumption than actual street lights
- Soft Real-Time Embedded System

3.3.0.3 Technical Constraints

- Buildroot
- C and C++
- Device Drivers
- Linux
- Raspberry Pi
- Cyber-Physical System (CPS)
- Makefiles
- Pthreads

3.3.0.4 Non-Technical Constraints

- Two members team
- Project deadline at the end of the semester
- Low budget

3.4 System Architecture

Using the system overview diagram information, one can describe the system in two different architectures. Hardware architecture, as how the hardware modules interfaces with itself, what are the physical components of the system, and software architecture, which details how the information is processed among different software layers. In this section, it will only be referenced the architectures of the base station, since the local system architecture is similar, as seen previously.

3.4.1 Hardware Architecture

3.4.1.1 Local System

In figure 3.5, one can see the diagram that represents the physical connections of the system. The Raspberry Pi is the main component in the system, processing all the information given by the sensors, via GPIO pins and Camera Serial Interface (CSI) for the camera. The communication between the Raspberry Pi and the LoRa module is done by Serial Peripheral Interface (SPI) protocol. The power of all system components comes from the power grid and, through an AC/DC converter, will power the Raspberry Pi and its associated sensors. In order to power the lamp and at the same time control its brightness, a driver is used, taking the controller output, a Pulse Width Modulation (PWM) signal, and system power as inputs.

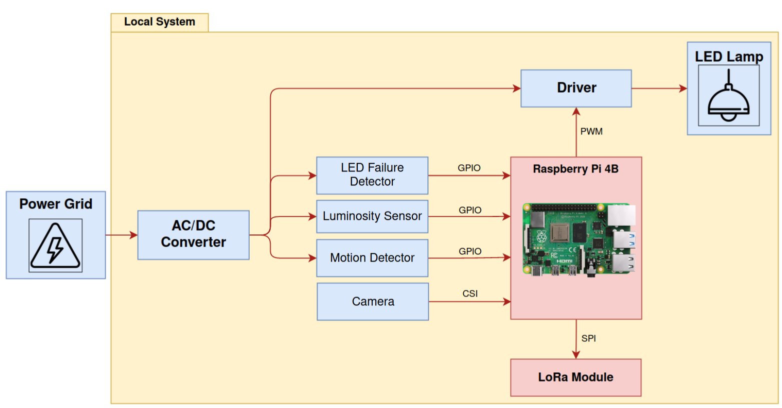

Figure 3.5: Local System Hardware Architecture Diagram.

3.4.1.2 Gateway

The hardware achitecture of the gateway is shown in figure 3.6. The purpose of this device is communicate with the local systems and control a network of street lampposts, so the hardware needed to complete these tasks is only the LoRa communication module that uses SPI to interface with the Raspberry Pi, the gateway controller.

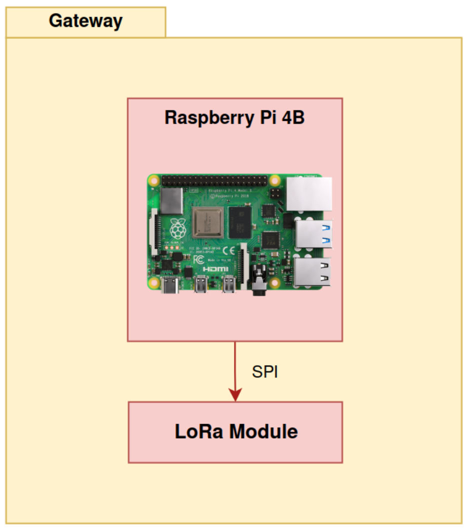

Figure 3.6: Gateway Hardware Architecture Diagram.

3.4.2 Software Architecture

The software architecture is divided into three layers:

- The **Operating System** layer, which is composed by the Operating System drivers and Board Support Packages;
- The **Middleware** layer, which includes software for abstracting the lower level layer packages. It works as a pipe since it links two applications, in different layers, so that data can be easily transmitted;
- The **Application** layer, where the core functionality of the program is built, with a resource for the API's in the lower level layers.

3.4.2.1 Local System

As shown in figure 3.7, the operating system layer is composed by the sensor drivers, such as the LED Failure Sensor, the Luminosity Sensor, the Motion Detector which uses GPIO drivers, the camera, that uses CSI drivers and also the LoRa Communication driver. In the middleware layer are the tools needed to process the images from the camera, to multitasking, using PThreads execution model, to acquire data from sensors and to communicate via LoRa protocol with the gateway. The application layer manages the communication with the gateway.

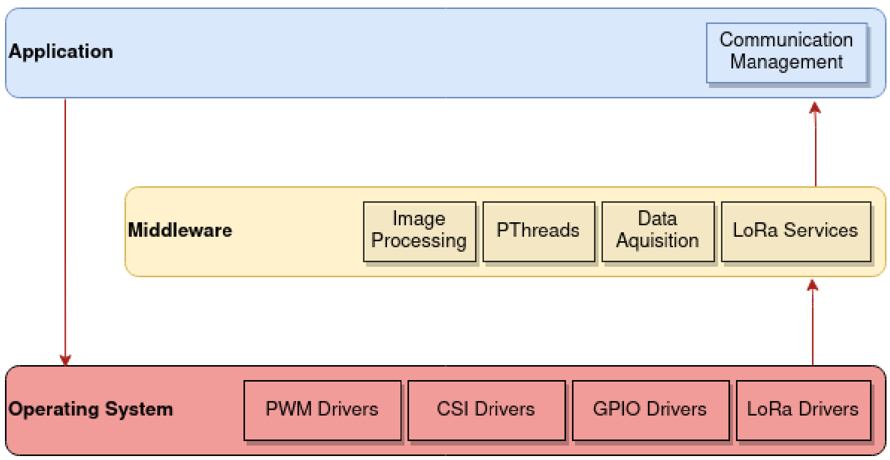

Figure 3.7: Local System Software Architecture Diagram.

3.4.2.2 Gateway

In figure 3.8 is shown the software architecture of gateway device. The operating system layer is composed by the LoRa communication device driver, whose interface is done using SPI protocol. The middleware layer deals with the PThreads execution model, in order to have multitasking, and the LoRa protocol services. Finally, the application layer manages the system database, as well as the Graphical User Interface (GUI), that is the mobile application and the web site, and also all communications with the neighbor street poles.

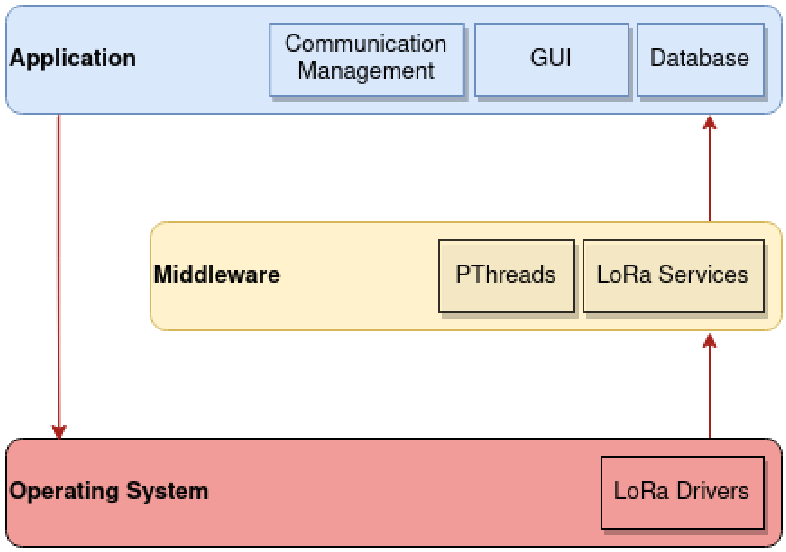

Figure 3.8: Gateway Software Architecture Diagram.

3.4.3 Database E-R Diagram

In the database will be stored all the information about each lamppost, its operators and parking spots. In the figure 3.9 one can see the Entity Relationship Diagram (E-R Diagram) that displays the relationships between the entities, that is the things of interest in this specific domain of knowledge. This database has five entities: lamppost, location, region, operator and parking_space. A light pole have a unique identification, a Global Positioning System (GPS) coordinates and the status of the lamp (light dimmed/ light fully ON/ OFF/ broken). The location has the GPS coordinates, the post code associated and the street name. A region has multiple locations associated, that are defined by the post code. This entity has also information about parish, county, district of the specified region and the operator responsible for the lampposts in that region. The entity operator has the operator identification and the operator name. A parking spot is defined by the entity parking_space and has the attributes identification of the parking space, its GPS coordinates, the parking space type (normal park/ park for disabled people/ restrict park) and the park status (taken/ not taken).

3.4.3.1 Relational Model

lamppost(pole_id, gps, pole_status)
location(gps, post_code, street_name)
region(post_code, operator_id, parish, county, district)
operator(operator_id, operator_name)
parking_space(park_id, gps, park_type, park_status)

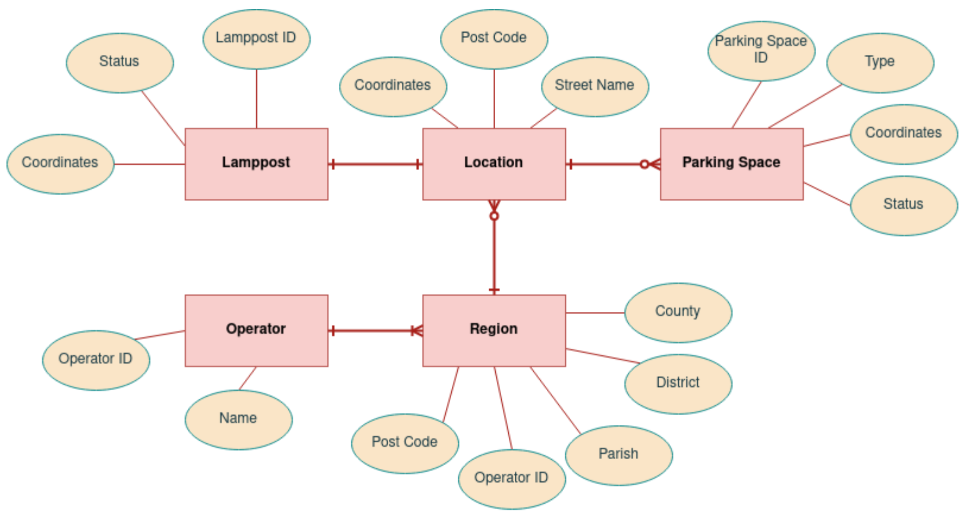

Figure 3.9: Database E-R Diagram.

Chapter 4

System Analysis

4.1 Local System

4.1.1 Events

To better understand the local system behavior, it is necessary to be aware of the events that may occur, defining how the system will respond to each one of them, as shown in table 4.2. The asynchronous events are generally triggered from external sources. In contrast, the synchronous events are triggered periodically.

Event	**System Response**	**Source**	**Type**
Luminosity detector OFF	Power the lamp	Environment	Asynchronous
LED failure detector ON	Notify remote system	Base station	Asynchronous
Motion detected	Turn on the lamp	User	Asynchronous
Requested to turn on the lamp	Turn on the lamp	Local system	Asynchronous
Camera sample	Image processing	Timer	Synchronous
Sensors data acquisition	Sample sensor values	Timer	Synchronous
Update system information	Send data to remote system	Local system	Asynchronous

Table 4.1: Local system events.

4.1.2 Use Cases

The base station use cases are presented in figure 4.1. A street passerby, a car or a pedestrian, can interact with the base station by moving in the vicinity of the lamppost, triggering it's motion detector, or by clearing a parking space.

When movement is detected, the base station lights up the lamp and requests to the neighbor lampposts to turn on their lamps. The opposite can also happen, when a neighbor local system, with the lamp already on, requests the base station to turn on its lamp. At the same time, in both situations, the information that the lamppost was activated is sent to the remote server. Since the base station is the "primary" station of the network, a third scenario can be put, when there is a local system requesting the base station to turn on the lamp of the local system's neighbor lampposts.

Moreover, the base station is frequently doing image processing through the capture of camera frames. So when the street passerby clears a parking space, the system will detect that, sending that information to the remote system.

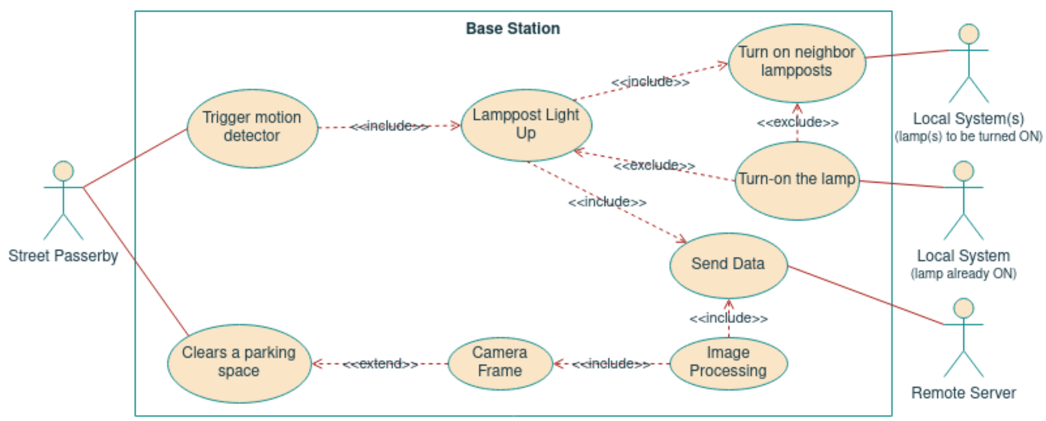

Figure 4.1: Base station use cases.

4.1.3 State Chart

In figure 4.2 its represented the state chart of the base station. It initiates with the system configuration, initializing all subsystems inside the base station, as the Wi-Fi communications management, sensors data acquisition, image processing. After that, the system enters an idle state.

To do the sensors data acquisition, it is used a sample period, that periodically triggers the execution of the function "SampleSensors", detailed in figure 4.3. When the base station is requested to turn on its lamp, the lamp is turned on and a timeout is started, named “turn off time” on the diagram. This timeout makes sure that the lamp stays on for a predefined period of time, after being triggered for being on. To do the image processing, it is also used a sample period to get image frames through the camera. If there is an available parking space detected, that information is sent to the remote server, as well as when the lamp is turned on or off, or when a LED failure is detected.

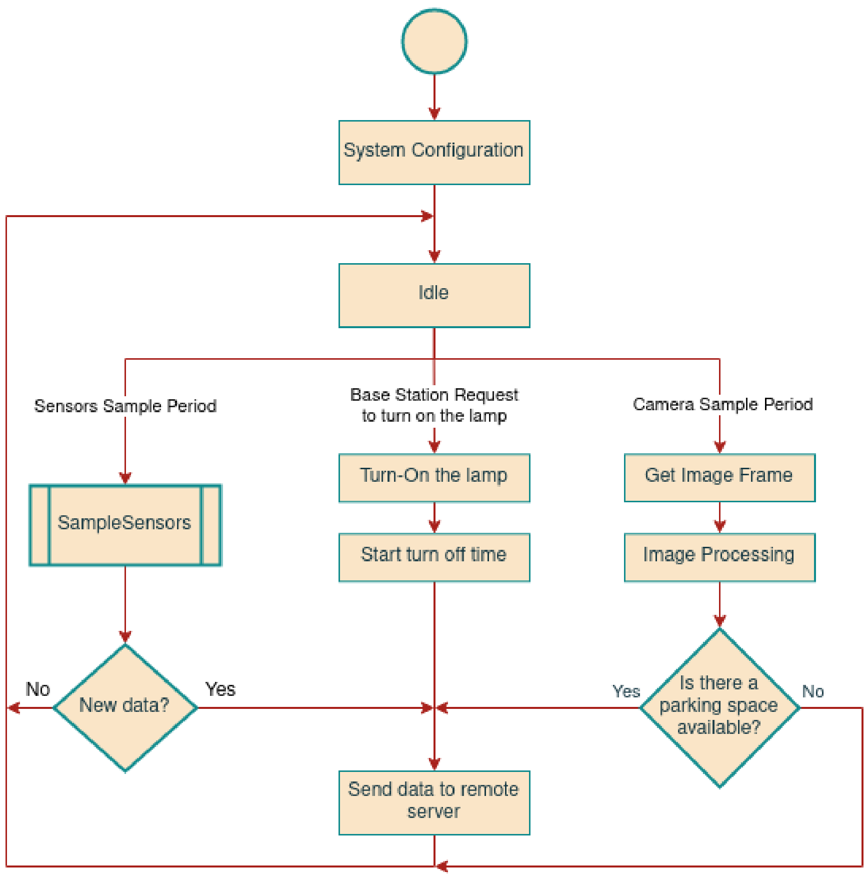

Figure 4.2: Base station state chart.

4.1.3.1 Sample Sensors

The sampling of the sensors is represented through the state chart detailed in figure 4.3.

Firstly, the LED failure detector is checked. If it is on, means that is detecting that the lamp has a failure, if not, the other sensors can be checked.

When low luminosity conditions are detected, through the luminosity sensor, the lamp is powered on, putting the lamp at a predefined minimum bright level. If motion is detected, the lamp is turned on to its maximum bright level, and a timeout is started, as explained before. Besides that, the base station requests the neighbor lampposts to turn their lamps on. When motion is not detected, it is checked if the turn off time has already ended. If that is true, the lamp is turned off.

Note that, regarding the lamp control, one will use “turn on” to represent the lamp bright transition from minimum bright level to maximum bright level, and use “turn off” to represent the opposite, the transition from maximum bright level to minimum bright level. The lamp is only off when low luminosity conditions is not verified, i.e, during the day.

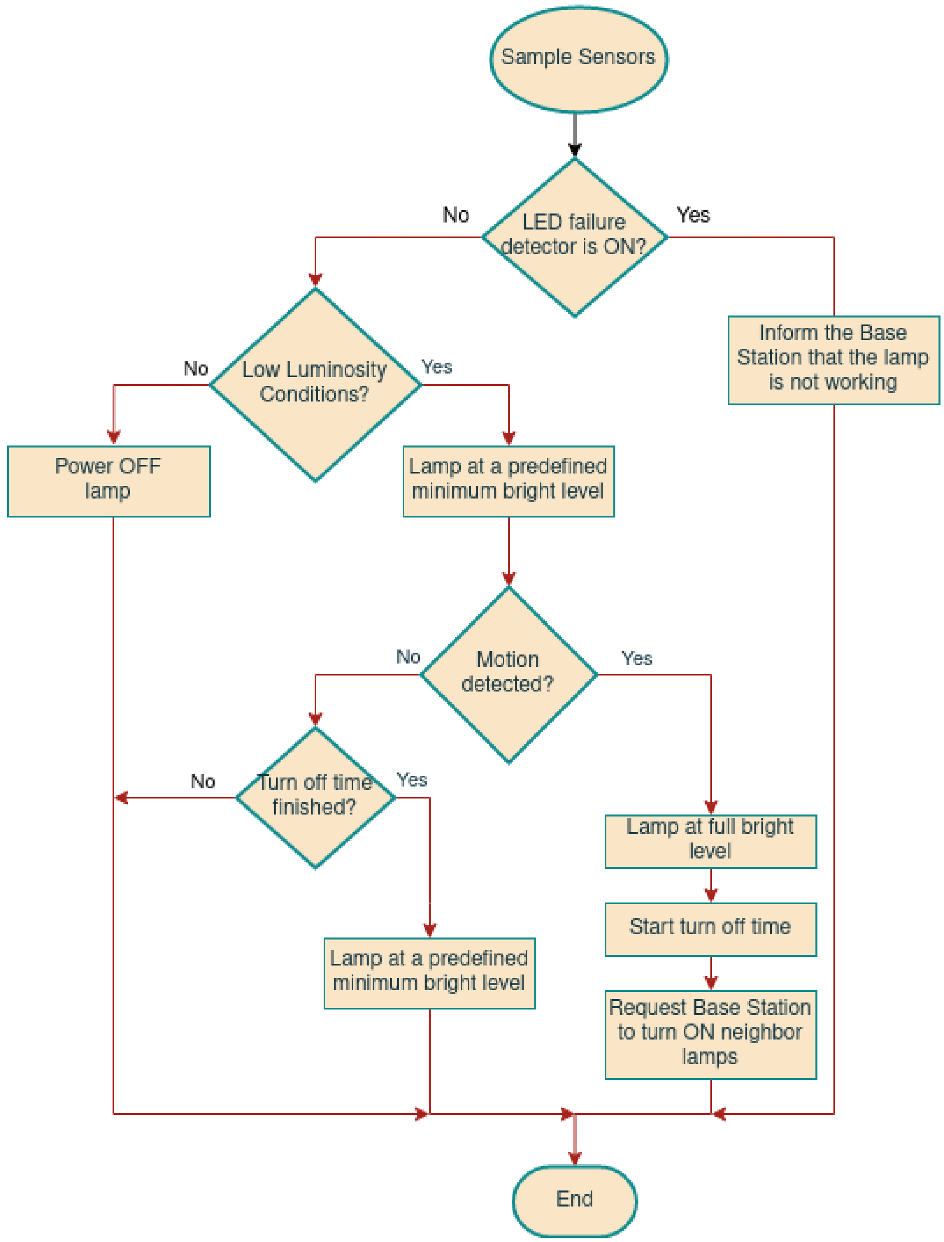

Figure 4.3: Sampling of the sensors state chart.

4.1.4 Sequence Diagram

In figure 4.4 it is shown the base station sequence diagram. When a street passerby triggers the motion detector, the base station turns on its lamp, and, at that moment, that information is updated in the remote server. After that, the communication management of the base station is free to communicate with the neighbor local systems to turn on their lamps. If, however, no more movement is detected, the lamp turns off after a predefined time (turn off time), and again, the lamp status is updated in the remote server.

An alternative of an interaction with the base station is when a local system requests the base station to turn on its lamp, this being processed in a similar way to the previous example.

Finally, the base station can also be requested to turn on the neighbor local systems of the local system that is requesting that.

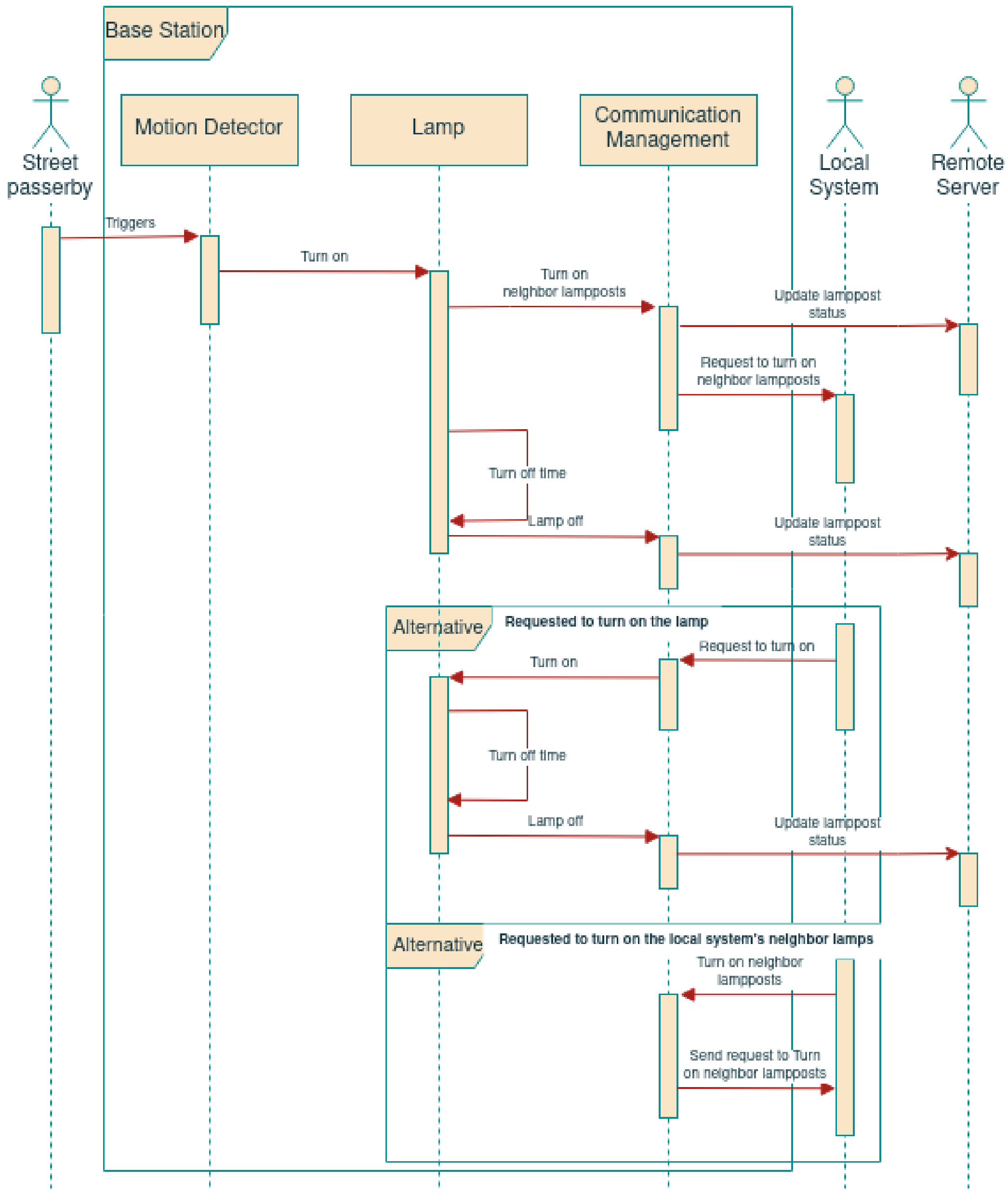

Figure 4.4: Base station sequence diagram.

4.2 Base Station

The local system's analysis is very similar to the base station analysis. Comparing to the base station, this system doesn't implement the parking spaces detection, since it doesn't have a camera, and only communicates with the base station. As in the base station, one can say that this is a passive system since it is most of the time waiting for something to happen.

4.2.1 Events

Listed in the table below, table 4.2, are the main events that will occur in the local system and will affect its behavior.

Event	System Response	Source	Type
Luminosity detector OFF	Power the lamp	Environment	Asynchronous
LED failure detector ON	Notify remote system	Local system	Asynchronous
Motion detected	Turn on the lamp	User	Asynchronous
Requested to turn on the lamp	Turn on the lamp	Base station	Asynchronous
Sensors data acquisition	Sample sensor values	Timer	Synchronous
Update system information	Send data to remote system	Base station	Asynchronous

Table 4.2: Local system events.

4.2.2 Use Cases

The local system use cases are represented on figure 4.5. Like the base station, a street passerby, can interact with the local system by moving in the vicinity of the lamppost, triggering it's motion detector.

When movement is detected, the lamp is turned on and requests the base station to turn on the local system's neighbor lampposts. At the same time, the information that the lamppost was activated is sent to the base station, for this to be sent to the remote system.

Also, the local system can be asked, by the base station, to turn on its lamp.

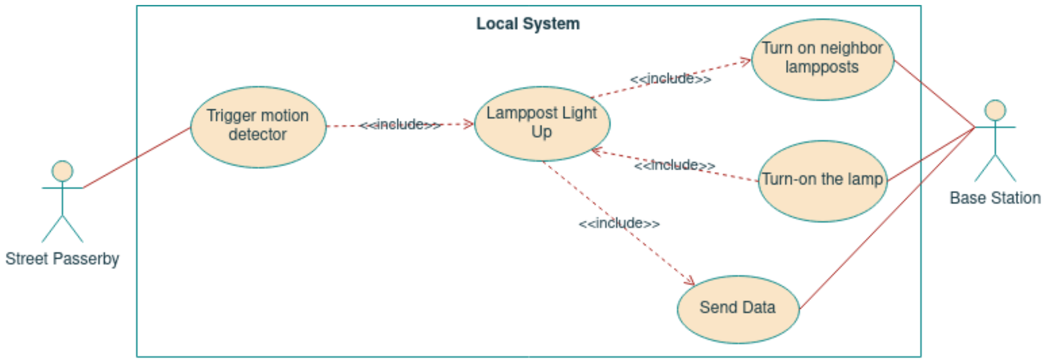

Figure 4.5: Local system use cases.

4.2.3 State Chart

In figure 4.6 is represented the state chart of the local system. After the system configuration, which initializes the Wi-Fi communication management, sensors data acquisition, the system enters an idle state. To do the sensors data acquisition, like in the base station, it is used a sample period, that periodically triggers the execution of the function "SampleSensors", presented previously, in figure 4.3. When the local system is requested by the base station to turn on its lamp, the lamp is turned-on and a timeout is started (turn off time, as mentioned before). The lamppost status is then sent to the base station, in order to send it to the remote server.

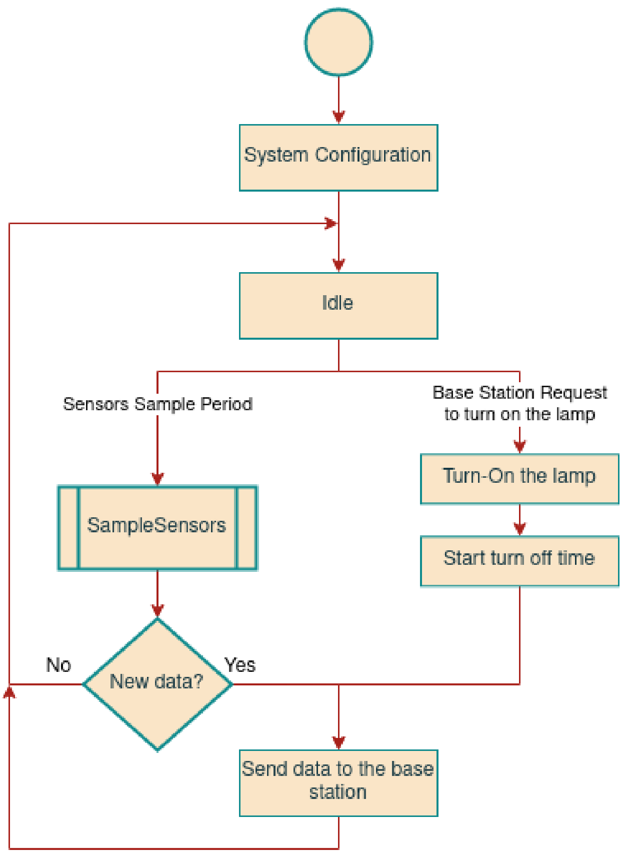

Figure 4.6: Local system state chart.

4.2.4 Sequence Diagram

The sequence diagram for the local system is represented in figure 4.7. Again, this diagram is analogous to the one shown on the base station (figure 4.4), having the differences that the local system only communicates with the base station, so any request that the local system wants to make to another local system has to go through the base station, and, due to that, the local system doesn't have the power to request another local system to turn its lamp on.

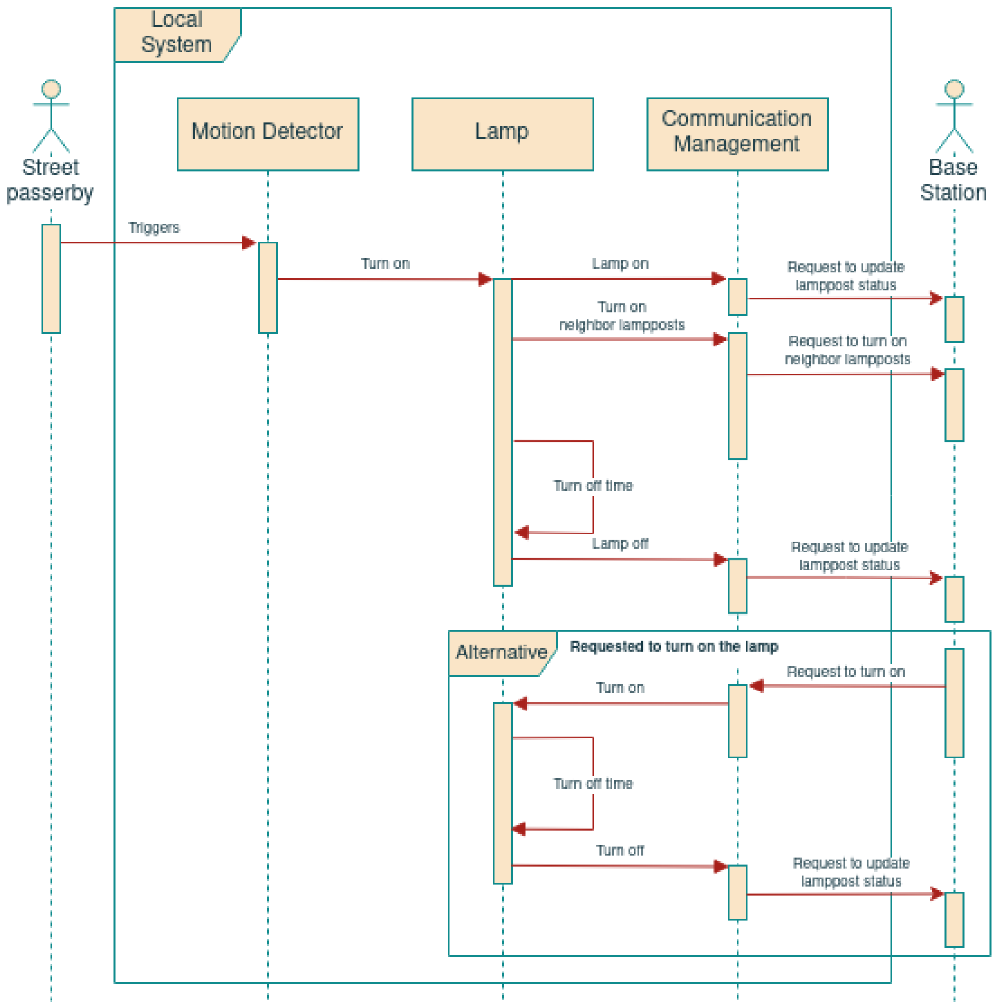

Figure 4.7: Local system sequence diagram.

4.3 Remote System

The remote system is composed by the remote server, that is a database, being connected to the local system in order to store its information. The remote server can be acessed by a remote client, which can be a web site or a mobile application. In this section is shown the system events, state charts and sequence diagrams of the remote client interface.

4.3.1 Events

In order to better understand the system, it is necessary to identify the events that may occur, how the system will respond to the event, what caused the event and what type of event it is. For the remote system, the events that may occur are presented in table 4.4.

Event	System Response	Source	Type
Login	Show application main screen if successful	Operator	Asynchronous
Obtain geolocation	Request device geolocation	Mobile device	Asynchronous
App notification	Notifies the operator about the lamppost status	Remote Server	Asynchronous
Register operator	Add operator information to databas	Operator	Asynchronous
Modify lamppost	Update lamppost information to database	Operator	Asynchronous
Register lamppost	Add lamppost information to database	Operator	Asynchronous
Insert location	Show parking spots	User	Asynchronous
Obtain geolocation	Request device geolocation	Mobile Device	Asynchronous

Table 4.3: Remote system events.

4.3.2 Use Cases

4.3.2.1 Mobile Application

The mobile application use cases diagram is represented in figure 4.8, showing that the main actors in the system are the operator, the database and the mobile device. The operator can perform operations such as login, logout, register, modify data about a lamppost and register newly installed posts that are installed, using, for this, the database (to obtain and record data) and the mobile device (to obtain the device location when registering a new post).

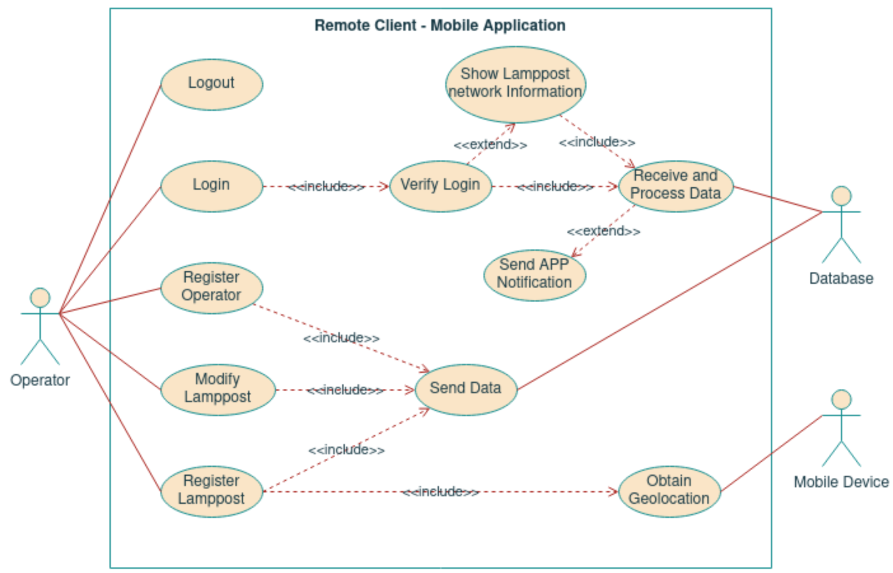

Figure 4.8: Remote System – Mobile Application Use Cases.

4.3.2.2 Web Site

The web site use cases diagram is shown in figure 4.9. The main actors are a user, the database and a mobile device. In order to know if there are free parking spaces in a certain location, the user can enter a location (through the street name, for example) or, as in the application, use his mobile device to obtain the location automatically. The database lets the user know where there are empty parking spaces.

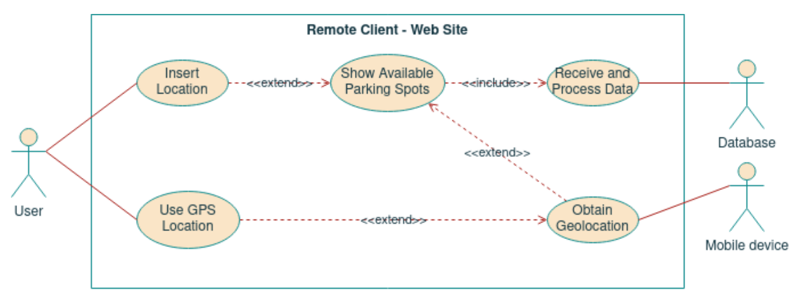

Figure 4.9: Remote System – Web Site Use Cases.

4.3.3 State Chart

4.3.3.1 Mobile Application

In the figure 4.10 is represented the state chart of the mobile application. It initiates with the system configuration, showing a home screen that allows the operator, the application user, to log into the system or register himself, if he doesn't have login credentials. After a successful login, the system will show information about the lampposts associated to the logged in operator and he can do operations like register lampposts, modify lampposts information and logout of the system.

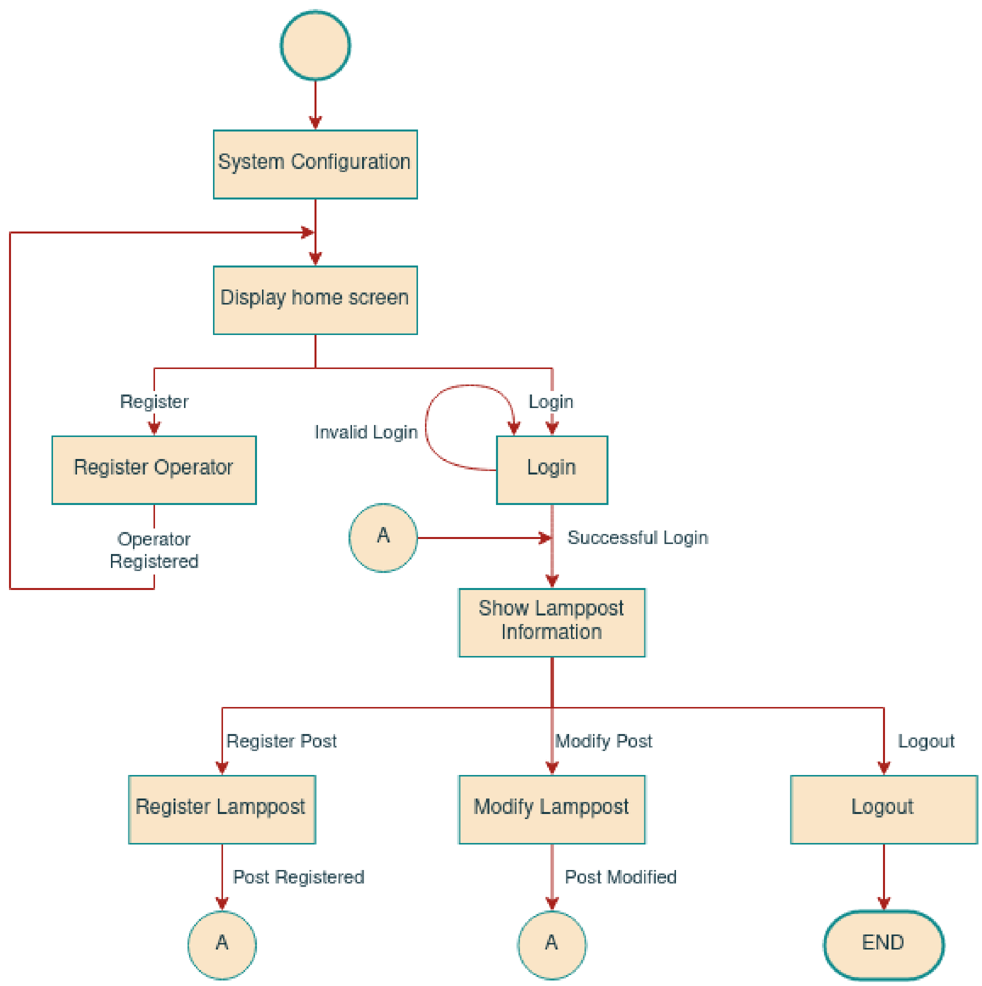

Figure 4.10: Remote System – Mobile Application State Chart.

4.3.3.2 Web Site

In figure 4.11, one can see the web site state chart. The system initiates with the system configuration. Then it asks the user if he wants to use his location (through the mobile phone's GPS tracking system) or type manually the location. If the user enters the location manually (the street name, for example), it will be checked and, if valid, the free parking spaces will be displayed.

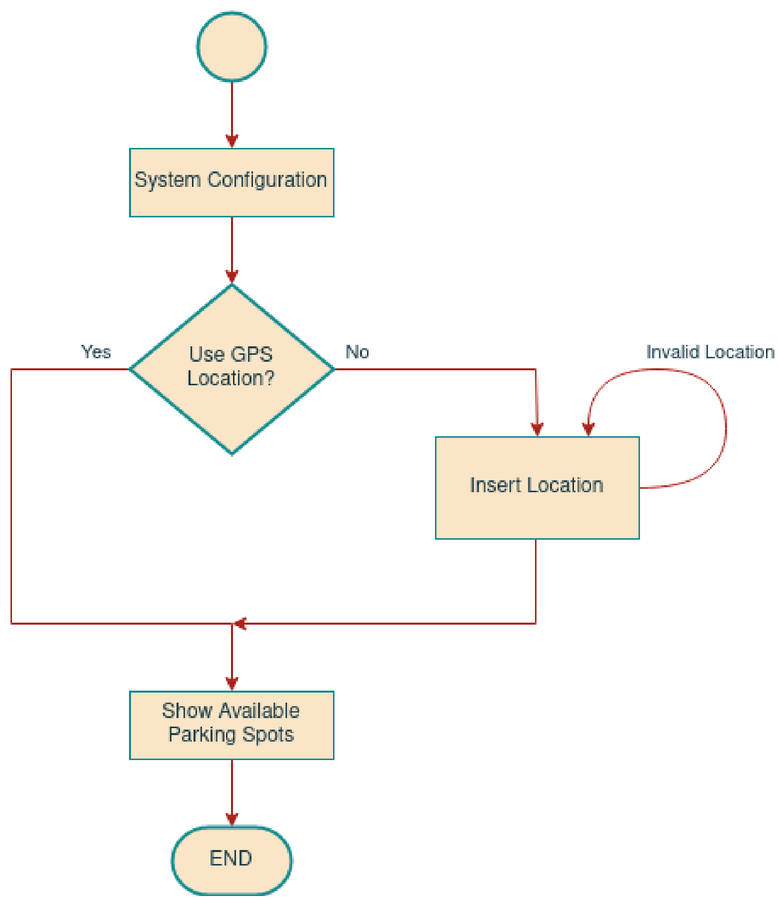

Figure 4.11: Remote System – Web Site State Chart.

4.3.4 Sequence Diagram

4.3.4.1 Mobile Application

In figure 4.13 is represented the sequence diagram of the mobile application. Most of the actions are triggered by the operator, starting with the registration or login operations in the application. If the login is valid, information about the network of lampposts will be shown and, depending on the interaction with the operator, the application may have different execution flows: registration of a lamppost; changing information about a lamppost; verification of the existence of damaged posts, notifying the operator if so. If the login is invalid, the application returns an error to the operator.

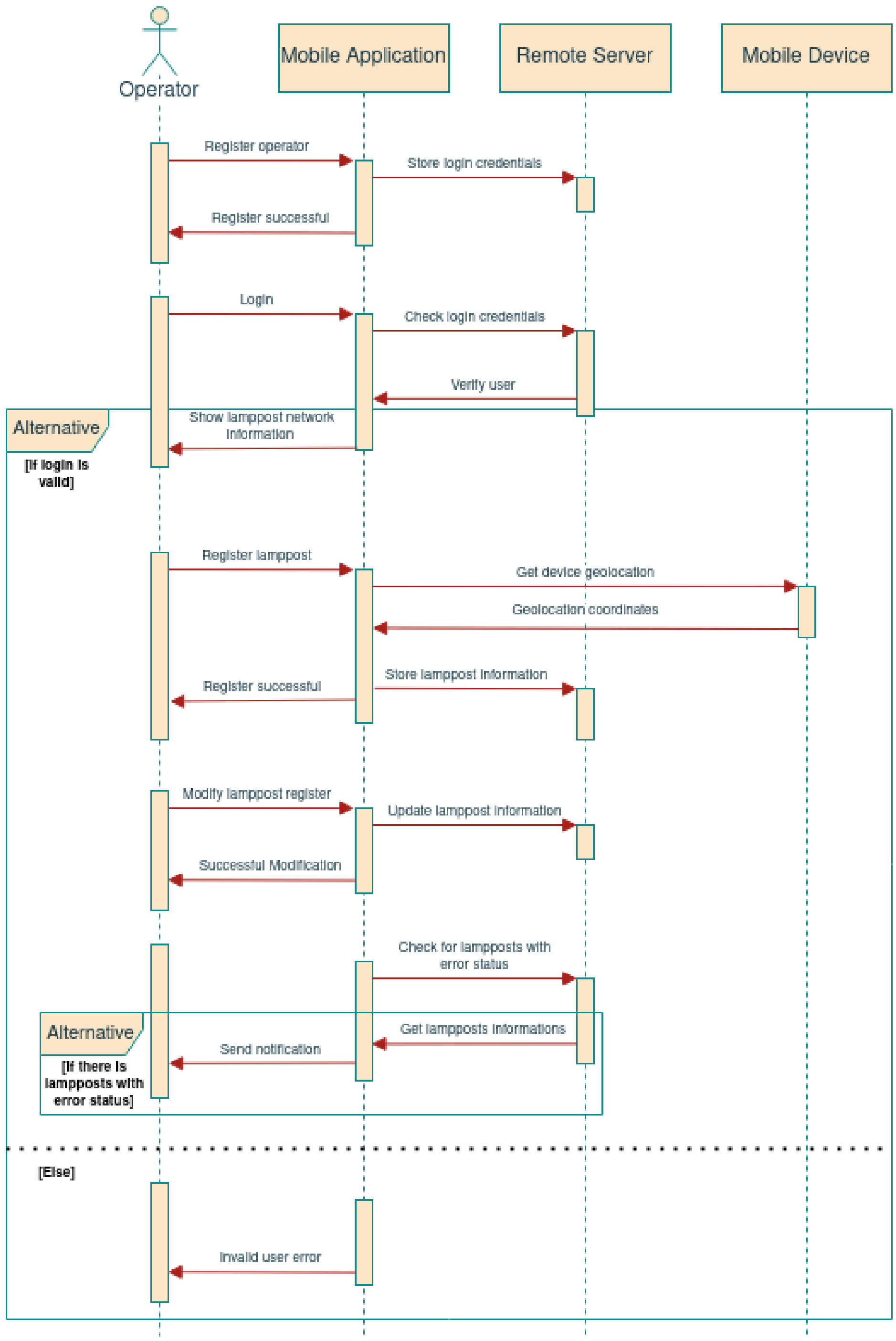

Figure 4.12: Remote System – Mobile Application Sequence Diagram.

4.3.4.2 Web Site

The sequence diagram of the web site is shown in the figure 4.13. To know where there are empty parking spots, the user can insert manually a location or use his GPS location. In both cases the web site asks the remote server (database) if there are available parking spots near the location and displays them to the user. However, in the case of using the GPS location, the web site has to get the GPS coordinates from the mobile device running the web site.

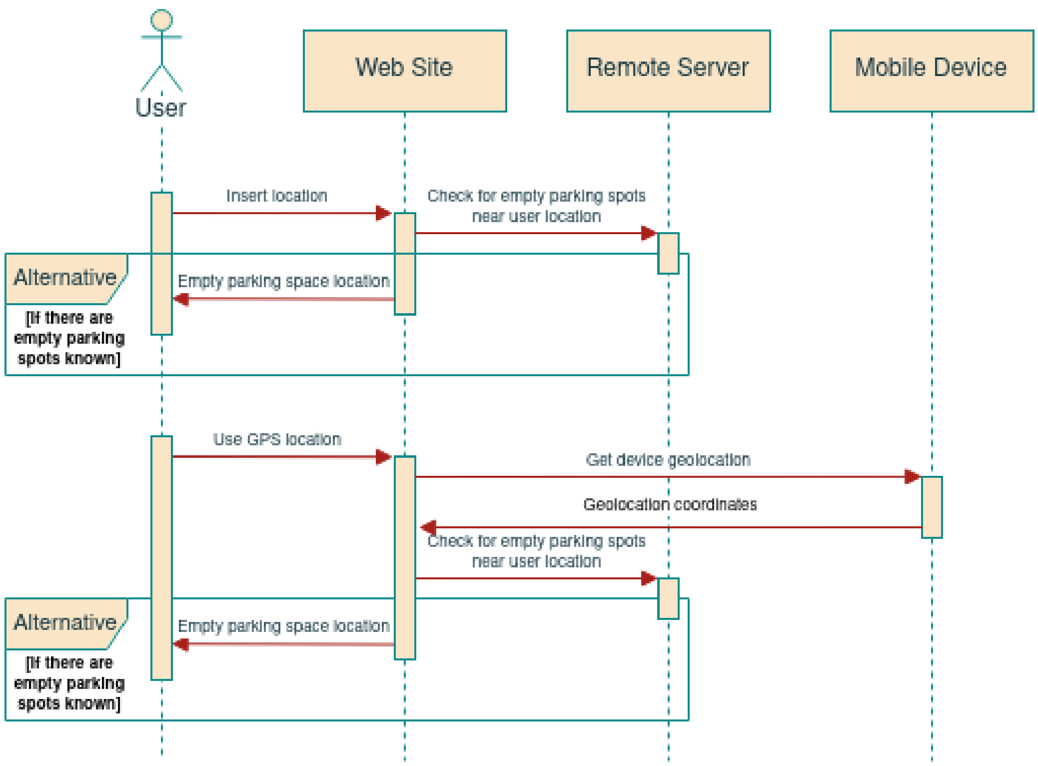

Figure 4.13: Remote System – Web Site Sequence Diagram.

4.4 Estimated Budget

In table 4.4 is shown the estimated budget of a base station, excluding the external casing of this system.

Product	**Price(€)**
Raspberry Pi 4B	63,50
Industrial power supply 12 V	5,00
Video camera	8,86
Motion detector	4,60
Luminosity sensor	1,69
LED lamp 12 V	3,63
Driver (MOSFET)	1,00
Basic Eletronic Components	5,00
Total	**93,28**

Table 4.4: Estimated budget.

4.5 Task Division and Gantt Chart

In figure 4.14, is represented the Smart Street Lighting project schedule in form of a Gantt chart.

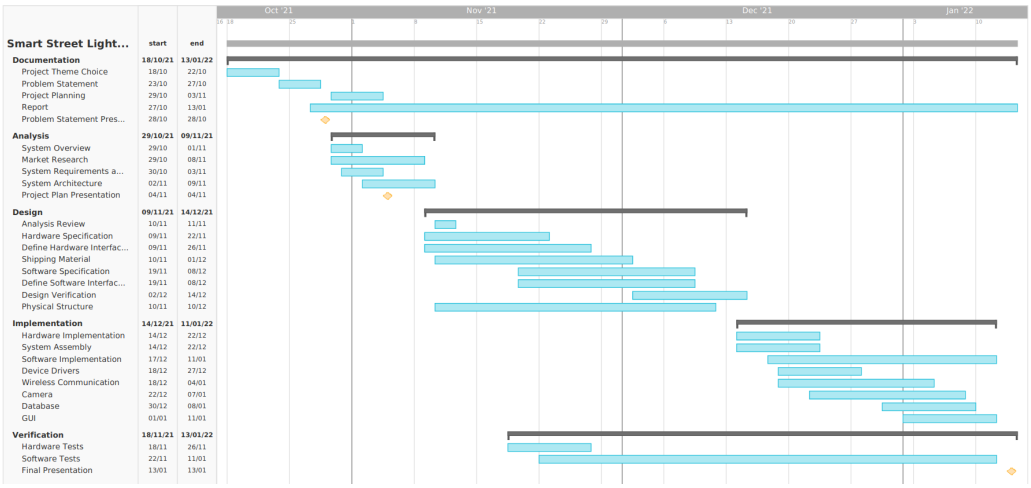

Figure 4.14: Gantt chart.

Chapter 5

System Design

5.1 Hardware Specification

5.1.1 Development Board

The development board for this project is the Raspberry Pi 4 Model B 5.1, considering it is one of the constraints identified in the analysis phase (3.3). This board includes a 64-bit quad-core ARM processor, the BCM2711, multimedia and connection features, ressembling to a computer-like board that serves multiple applications. The following list show the Raspberry Pi 4 Model B main features:

- 2GB LPDDR4-3200 SDRAM;
- 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE;
- Raspberry Pi standard 40 pin GPIO header;
- 2 USB 3.0 ports and 2 USB 2.0 ports;
- 2 micro-HDMI ports;
- 1 display port (2-lane MIPI DSI);
- 1 camera port (2-lane MIPI CSI);
- 1 jack 3,5 mm port (4-pole stereo audio and composite video port);
- graphic support (OpenGL ES 3.1, Vulkan 1.0);

- Micro-SD card slot.

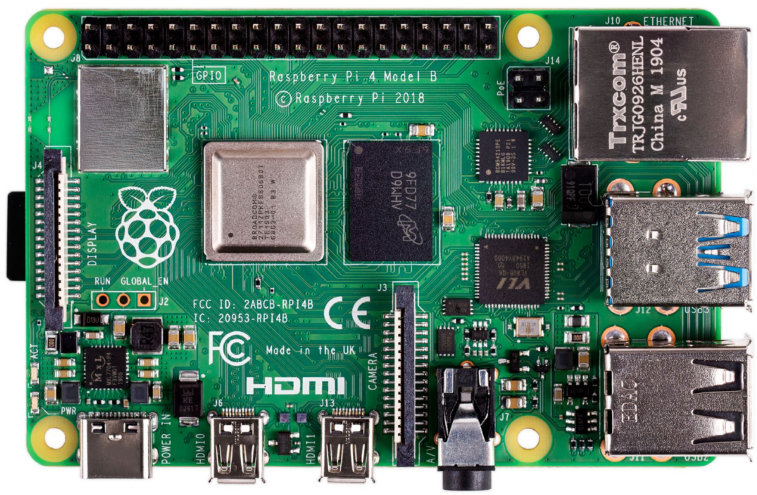

Figure 5.1: Raspberry Pi 4 Model B.

5.1.1.1 GPIO

The Raspberry Pi 4 Model B board comes with a standard 40 pin GPIO header, that allows to interface with external peripherals. This GPIO also provides some interface technologys, like UART, I2C or SPI. The GPIO pinout of this board is shown in figure 5.2 [19].

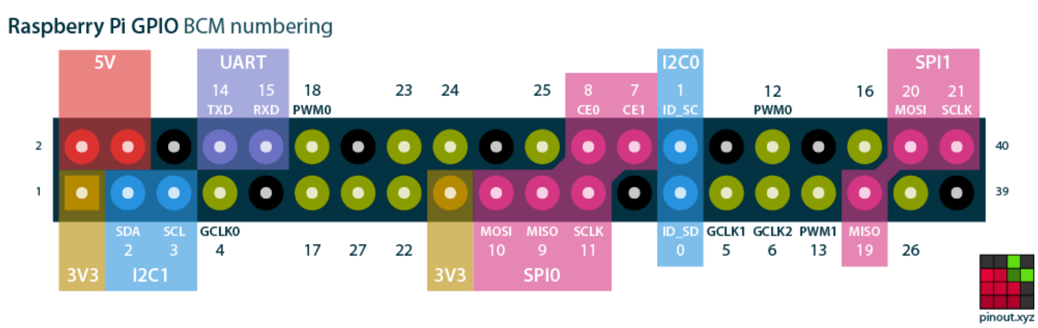

Figure 5.2: Raspberry Pi 4 Model B GPIO Pinout.

5.1.2 LED Lamp

In order to light the streets efficiently, we must use a LED lamp.

5.1.3 Light Sensor

In order to know when is night time, that is when the light conditions are low, one needs to determine the ambient light conditions, using a module composedby a Light Dependant Resistor (LDR) sensor and a LM393 comparator, represented in figure 5.3. The LDR determines the environmental light conditions varying its resistivity. In the table 5.1 is shown the LDR Module interface pins. When ambient light intensity does not reach the threshold value (defined by a potentiometer), the module's digital output (DO) is high, and when the ambient light level exceeds the threshold, the module's digital output terminal outputs low.

Figure 5.3: LDR Module.

LDR Module Pin	Board Pin
VCC	5 V
DO	Pin 17
GND	GND

Table 5.1: LDR Module Interface Pins.

5.1.4 Movement Sensor

To know when to turn on the light, it is necessary to detect movement in the streets. For this project it is used a movement sensor, more specifically a Passive Infrared (PIR) sensor. The chosen sensor for this purpose was the PIR HC-SR501 that have a supply voltage range of 4,5 - 20 V and a detection range of 7 meters with a 110 degrees angle. In the table 5.2 is shown the PIR HC-SR501 interface pins, being the output signal the pin SIGNAL. When no movement is detected, the sensor output is low, and when movement is detected, the sensor SIGNAL is high. The sensor has also two potenciometers to adjust the trigger sensitivity and the delay of the trigger signal, between 0,3 seconds and 5 minutes.

Figure 5.4: PIR Sensor Module.

LDR Module Pin	**Board Pin**
VCC	5 V
SIGNAL	Pin 27
GND	GND

Table 5.2: PIR Module Interface Pins.

Bibliography

[1] libelium, "Iot solutions smart cities," https://www.libelium.com/iot-solutions/smart-cities/ [Accessed on 17/11/2021].

[2] C. Helman, "Energy crisis 2021: How bad is it, and how long will it last?" Forbes, 2021, https://www.forbes.com/sites/christopherhelman/2021/10/19/energy-crisis-2021-how-bad-is-it-and-how-long-will-it-last/?sh=6a5feff14c63 [Accessed on 28/10/2021].

[3] H. Ritchie and M. Roser, "Energy," Our World in Data, 2020, https://ourworldindata.org/energy [Accessed on 28/10/2021].

[4] "Cars parked 23 hours a day," https://www.racfoundation.org/media-centre/cars-parked-23-hours-a-day [Accessed on 8/11/2021].

[5] R. Pi, "Raspberry pi 4," https://www.raspberrypi.com/products/raspberry-pi-4-model-b/ [Accessed on 22/11/2021].

[6] L. Alliance, "What is lorawan specification," https://lora-alliance.org/about-lorawan/ [Accessed on 3/12/2021].

[7] W.-S. Alliance, "Iot networks," https://wi-sun.org/iot-networks/ [Accessed on 3/12/2021].

[8] Z. Alliance, "Zigbee," https://zigbeealliance.org/solution/zigbee/ [Accessed on 3/12/2021].

[9] Sigfox, "Sigfox technology," https://www.sigfox.com/en/what-sigfox/technology [Accessed on 3/12/2021].

[10] GSMA, "Narrowband – internet of things," https://www.gsma.com/iot/narrow-band-internet-of-things-nb-iot/ [Accessed on 3/12/2021].

[11] ——, "Long term evolution for machines," https://www.gsma.com/iot/long-term-evolution-machine-type-communication-lte-mtc-cat-m1/ [Accessed on 3/12/2021].

[12] M. Intelligence, "Global connected street lighting market - growth, trends, covid-19 impact, and forecasts (2021 - 2026)," https://www.mordorintelligence.com/industry-reports/connected-street-lights-market [Accessed on 8/11/2021].

[13] FLASHNET, "intelilight smart street lighting control," https://www.flashnet.ro/project/intelilight/ [Accessed on 1/11/2021].

[14] Telensa, "Smart city solutions," https://www.telensa.com/solutions/ [Accessed on 1/11/2021].

[15] D. B. M. Research, "Global smart parking market," https://www.databridgemarketresearch.com/reports/global-smart-parking-market [Accessed on 22/11/2021].

[16] "intuvision parking," https://www.intuvisiontech.com/intuvisionVA_solutions/intuvisionVA_parking [Accessed on 8/11/2021].

[17] "What is the range of a typical wi-fi network?" 2020, https://www.lifewire.com/range-of-typical-wifi-network-816564 [Accessed on 22/11/2021].

[18] "802.11n wi-fi in computer networking," 2021, https://www.lifewire.com/802-11n-wireless-network-818281 [Accessed on 22/11/2021].

[19] R. P. Pinout, "The raspberry pi gpio pinout guide," https://pinout.xyz/# [Accessed on 1/12/2021].