UNISINOS

Programa de Pós-Graduação em

Computação Aplicada

Mestrado Acadêmico

Bianca dos Santos Brand

Sapientia - A Smart Campus Model That Promotes Flexibility

São Leopoldo, 2020

UNIVERSIDADE DO VALE DO RIO DOS SINOS - UNISINOS
UNIDADE ACADÊMICA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA
NÍVEL MESTRADO

BIANCA DOS SANTOS BRAND

SAPIENTIA - A SMART CAMPUS MODEL THAT PROMOTES FLEXIBILITY

São Leopoldo
2020

BIANCA DOS SANTOS BRAND

SAPIENTIA - A SMART CAMPUS MODEL THAT PROMOTES FLEXIBILITY

Dissertação apresentada como requisito parcial para obtenção do título de Mestre pelo Programa de Pós-Graduação em Computação Aplicada da Universidade do Vale do Rio dos Sinos - UNISINOS

Advisor: Prof. Dr. Sandro José Rigo

Coadvisor: Prof. Dr. Rodrigo Marques de Figueiredo

São Leopoldo

2020

B817s Brand, Bianca dos Santos.
Sapientia : a smart campus model that promotes flexibility / Bianca dos Santos Brand. – 2020.
129 f. ; 30 cm.

Dissertação (mestrado) – Universidade do Vale do Rio dos Sinos, Programa de Pós-Graduação em Computação Aplicada, 2020.
“Orientador: Prof. Dr. Sandro José Rigo ; coorientador: prof. Dr. Rodrigo Marques de Figueiredo”.

1. Smart campus. 2. Smart campus model. 3. Flexibilidade. 4. Acessibilidade. 5. Internet-das-coisas. 6. Inteligência artificial. I. Título.

CDU 004

Dados Internacionais de Catalogação na Publicação (CIP)
(Bibliotecário: Flávio Nunes – CRB 10/1298)

BIANCA DOS SANTOS BRAND

SAPIENTIA - A SMART CAMPUS MODEL THAT PROMOTES FLEXIBILITY

Dissertação apresentada como requisito parcial para a obtenção do título de Mestre pelo Programa de Pós-Graduação em Computação Aplicada da Universidade do Vale do Rio dos Sinos - UNISINOS

Aprovado em 19/03/2020

BANCA EXAMINADORA

Prof. Dr. Sandro José Rigo

Orientador – Universidade do Vale do Rio dos Sinos (UNISINOS)

Prof. Dr Rodrigo Marques de Figueiredo

Coorientador – Universidade do Vale do Rio dos Sinos (UNISINOS)

Prof. Dr. José Pallazo Moreira de Oliveira

Avaliador Externo – Universidade Federal do Rio Grande do Sul (UFRGS)

Prof. Dr. Jorge Luis Victoria Barbosa

Avaliador PPGCA – Universidade do Vale do Rio dos Sinos (UNISINOS)

ACKNOWLEDGEMENTS TO CAPES

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 88887.179214/2018-00.

For my family.

ACKNOWLEDGEMENTS

First, I would like to thank my family, for their support, understanding and encouragement along this process.

I would also like to thank my advisors, Prof. Phd Sandro J. Rigo and Prof. Phd Rodrigo M. de Figueiredo, for their patience, teachings, understanding, guidance and advice. Also, I would like to thank Prof. Phd Lúcio R. Prade and my classmate Everton Nedel, for their help, guidance, teachings and advice in the development of the LoRa prototype.

I would also like to thank my managers Alexandre Junges, Marco Schwertner and José Turchiello for their understanding and support, allowing me the necessary time to dedicate to this research.

ABSTRACT

Nowadays, the expansion of IoT and ICT technologies is observed, together with the growing interest in the solutions offered by the Intelligent Concepts applied to the context of cities and other spaces. However, it was identified in the literature the lack of flexible models that also included accessibility aspects, besides being able to incorporate existing hardware and software solutions, which motivates this research to address such issues and challenges. Thus we propose the Sapientia Smart Campus model, which promotes flexibility, also including accessibility aspects as well, with the independence of contexts and being able to incorporate existing solutions.

Such features are possible due to the architecture of the model, composed of layers that facilitate the management and update of the technologies used, besides facilitating the insertion of accessibility aspects and the integration of new and existing applications. The model implementation evidences this, involving hardware and software infrastructure, by allowing the same technology to have different applications and new technologies to be easily incorporated. Experiments were carried out with a mobile application that incorporated accessibility aspects of the model and the collection of user behavior information on campus. Also, a data analysis unit was developed, containing Artificial Intelligence resources that perform clustering and time series analysis in the collected data. In addition, it was incorporated existing applications to the developed infrastructure, thus demonstrating the flexibility of the model.

Key-Words: Smart Campus. Smart Campus Model. Flexibility. Accessibility. Internet-of-Things. Artificial Intelligence.

RESUMO

Atualmente observa-se a expansão das tecnologias IoT e ICT, aliadas ao crescente interesse pelas soluções oferecidas pelos Conceitos Inteligentes aplicados ao contexto de cidades e outros espaços. Entretanto, identifica-se na literatura a falta de modelos flexíveis e que também suportem soluções acessíveis, além de ser capaz de incorporar soluções existentes de hardware e software, o que motiva esta pesquisa a abordar tais questões e desafios. Desta forma é proposto o modelo Sapientia Smart Campus, o qual promove flexibilidade, e também inclui aspectos de acessibilidade, com independência de contextos e sendo capaz de incorporar soluções existentes.

Tais características são possíveis devido à arquitetura do modelo, composto por camadas que facilitam o gerenciamento e a atualização das tecnologias utilizadas, além de facilitar a inserção de aspectos de acessibilidade e a integração de aplicações novas e existentes. Isso é evidenciado pela implementação realizada, envolvendo a infraestrutura de hardware e de software, que permite que a mesma tecnologia tenha diferentes aplicações e que novas tecnologias sejam facilmente incorporadas. Experimentos foram conduzidos com um aplicativo móvel, o qual incorporou aspectos de acessibilidade do modelo e realizou a coleta de informações de comportamento do usuário no campus. Além disto, uma unidade de análise de dados foi desenvolvida, a qual contém recursos de Inteligência Artificial que realizam agrupamentos e análises de séries temporais nos dados coletados. Além disto, foram incorporadas aplicações existentes na infraestrutura desenvolvida, assim demonstrando a flexibilidade do modelo.

Palavras-Chave: Campus Inteligente. Modelo Campus Inteligente. Flexibilidade. Acessibilidade. Internet-das-Coisas. Inteligência Artificial.

LIST OF EQUATIONS

LIST OF FIGURES

LIST OF GRAPHICS

LIST OF TABLES

LIST OF ACRONYMS

3GPP	3rd Generation Partnership Project
ABP	Activation by Personalization
ADC	Analog/Digital Converter
AI	Artificial Intelligence
AP	Access Point
API	Application Programming Interface
ARP	Address Resolution Protocol
BLE	Bluetooth Low Energy
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
GPIO	General Purpose Input/Output
HDBSCAN	Hierarchical Density-Based Spatial Clustering of Applications with Noise
HTTP	HyperText Transfer Protocol
I2C	Inter-Integrated Circuit
IBGE	Instituto Brasileiro de Geografia e Estatística
ICT	Information and Communication Technology
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet-of-Things
IP	Internet Protocol
IT	Information Technology
LAN	Local Area Network
LoRa	Long Range
LoRaWAN	Long Range Wide Area Network
LTE	Long Term Evolution
M2M	Machine-to-Machine
MAC	Medium Access Control
MQTT	Message Queuing Telemetry Transport
MQTT-SN	Message Queuing Telemetry Transport – Sensor Network
NLP	Natural Language Processing
ONU	Organização das Nações Unidas
QoS	Quality-Of-Service
RSSI	Received Signal Strength Indication

SoC	System-On-Chip
SPI	Serial Peripheral Interface
TCP	Transmission Control Protocol
TTN	The Things Network
UART	Universal Asynchronous Receiver/Transmitter
UDP	User Datagram Protocol
W3C	World Wide Web Consortium
WAI	Web Accessibility Initiative
WCGA	Web Content Accessibility Guidelines
Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Networks

CONTENTS

1 INTRODUCTION

With the constant increase of population in urban regions, allied with the immigration to populated urban areas, the concept of Smart City is being increasingly considered as fundamental for society, since this aims to assist government and society to solve problems originated from this population growth, such as infrastructure and mobility problems. Besides, this solution can offer new opportunities to population, promoting quality of life, competitiveness, economic growth and sustainability, besides making use of natural resources in an intelligent way, based on the application of new technological resources in an efficient way (Tanda *et al*., 2017). In order to achieve this, an ICT (Information and Communication Technology) and similar technologies are utilized, aiming to enhance the quality of services and operations offered to its citizens, increasing the city's intelligence (Silva *et al*, 2018).

Thereby, other concepts emerged with the expansion of the Smart City concept, like the Smart Campus and the Smart Region, where the former adapts its concept and many of its requirements from the Smart City concept due to its similarities (Muhamad *et al*., 2017), and the latter, which also inherits its concept from the Smart City, is defined by O'Brolchain *et al*. (2018) as,

> A high-tech intensive and advanced approach connecting people, information, governance and policies using new technologies to create an efficient, clean, energy secure, sustainable, eco-friendly, competitive and innovative region with an enhanced quality of life.

This approach must be supported through information technology tools, which are responsible for making the exchange of information between the cities or regions (Mannaro *et al*., 2017). Meanwhile, the IoT (Internet-of-Things) technology is in expansion, allowing the interaction and integration between objects and environments through the Internet, and consequently, connecting environments and people. Such devices have the capacity to communicate and connect mutually through wireless technologies, which results in an optimization of processes and an easier management of environments and systems (Zanella *et al*., 2014).

As stated by Abou-Zahra, Brewer, Cooper (2017), "Internet of Things (IoT) has the potential to disproportionally benefit people with disabilities, and allow unprecedented access to the physical world". Also, they cite the use of Beacon devices as a mean to support the indoor and outdoor navigation, which can enable people with disabilities to access and navigate independently and efficiently such places. Also, the IoT can enable the smart transportation and mobility, providing accessible routes and allowing journey planning in real-time (Abou-Zahra, Brewer, Cooper, 2017).

Abou-Zahra, Brewer, Cooper (2017), also highlight one of the current issues of the IoT technology, the interoperability, which does not support access to information in an accessible way, particularly in data and API levels, mainly because such technologies are based on proprietary specifications. This happens in consequence of the fear of many vendors of losing customers and businesses (Abou-Zahra, Brewer, Cooper, 2017).

Besides, according to IBGE (2013), the South region of Brazil has the highest indicator of people with visual disability among the other regions in the country, totalizing 5,9%, in addition to the global estimate of 285 million people that suffer from visual impairment (Pascolini and Mariotti, 2012). Aside from these facts, the solutions for accessibility in Smart Cities and Smart Campus are generally inflexible and applicable only in specific situations.

Therefore, based on the issues aforementioned, this research aims to identify the requirements, challenges and definitions of the Smart Campus concept, besides analyzing the accessibility adhesion and inclusion on the existing models and solutions, in order to propose a flexible model for a Smart Campus that promotes flexibility and includes accessibility aspects on its infrastructure, making it possible through IoT and ICT technology.

However, with the expansion of the IoT technology allied with the growing interest about Smart Concepts solutions, and the identified challenges for people with disability, one of the many factors to motivate this research was the possibility of such technologies in addressing mobility problems and offer better services to the users, improving their experience and promoting inclusion, besides of incorporating the existing solutions.

1.1 OBJECTIVES AND RESEARCH QUESTION

Based on the context as mentioned earlier, the research question for this work is defined as: How to implement a Smart Campus model that promotes the device's flexibility, and also includes accessibility aspects and can incorporate existing hardware and software solutions?

Considering the research question and the general context in this area, the main objective of this research is to propose and implement a Smart Campus model that promotes flexibility and includes accessibility aspects on its infrastructure.

To support the achievement of this main objective, the following specific objectives were defined: a) to use the model to develop prototypes of applications to encourage inclusion and offer better services to the campus users; b) explore the models capabilities for incorporating other existing solutions; c) gather data originated from the model components to provide a better understanding of the campus usage and its user's behavior; d) apply different analysis approaches over the collected data, to support better management of resources and decision-making to optimize the resources usage; e) make available the infrastructure implemented to instantiate the model as a guide for future projects and applications.

1.2 METHODOLOGY AND TEXT STRUCTURE

This research started with a systematic literature review, identifying through it the research challenges, understanding the existing models and the basic concepts necessary for its understanding. Next, using the requirements, concepts and challenges identified in the review, the model was proposed and developed, aiming to incorporate features such as device flexibility, including accessibility aspects and present a structure that allows to incorporate existing applications and solutions. Then, a prototype was implemented, based on the proposed model, and divided in four structures, in order to facilitate the tests on it, whose results are presented later on this text.

The following section presents the theoretical background, defining the basic concepts studied in this research such as Smart Concepts, Accessibility, IoT Architectures, as well as the localization techniques used for the mobile application implemented, and the AI (Artificial Intelligence) techniques utilized for the analysis of the data obtained. Section 3 presents related works, and the analysis obtained through the systematic literature review conducted. Section 4 presents a detailed description of the proposed model. The implementation of the model, describing in detail the IoT infrastructure, the mobile application, integration application and data analysis application is described in section 5. The experiments conducted are described in section 6. At last, section 7 presents the conclusions, contributions and future work for this research.

2 CONCEPTUAL FUNDAMENTATIONS

This section will define and detail the basic concepts used for the development of the proposed model. Firstly, the accessibility concept is detailed, as well as the existing standards and laws. This section will also detail how accessibility is inserted in a smart environment. Secondly, the smart concepts are detailed, such as Smart City, Smart Campus, Smart Region and Smart Environment. Also, the available Smart Models are presented. Thirdly, the Internet-of-Things (IoT) concept is detailed, along with its basic architecture. In addition, the hardware used for the prototype developed is presented, explaining its functionality and the possibilities. Next, the localization techniques studied are presented and detailed, focusing on the chosen technique, the trilateration. Lastly, the Artificial Intelligence techniques utilized for the analysis of the gathered data are presented.

2.1 ACCESSIBILITY

This section will define the accessibility concept, as well as present the existing standards and laws, besides detailing the accessibility concept in relation to Smart Environments.

2.1.1 Definition

As stated by Van Wee (2016), accessibility has many definitions, but most include either destinations or activities, and travel resistance. The generic definition that best describes accessibility, in my opinion, is defined by Mora *et al*. (2016) as:

> The access for people with disabilities to be on an equal basis with others, to the physical environment, transportation, information and communications technology and systems and other facilities and services.

Most studies about accessibility focus on urban mobility and navigation, which is one of the biggest problems for cities nowadays. Although, as stated in the definition by Mora *et al*. (2016), currently there is another big issue for accessibility, which are information and data accessibility. With the expansion of big data and artificial intelligence, it is indispensable that the data produced via IoT also presents accessibility. For instance, as stated by Abou-Zahra *et al*. (2017), sensors that produce integers values, should also present the same value in a text variable, which in association with NLP (Natural Language Processing) techniques, make this data more accessible for the users.

Moreover, applications that adheres accessibility should also give the user a choice or option, like for example, to choose between different routes and contexts. Suppose it’s a raining day, the navigation application should allow the user to choose for a route, being this a "drier" route or the suggested route, not basing its choice only in a singular parameter alone, like the distance. Accessibility should also allow the user to adapt to a context, adapting its configurations and services in order to better meet the user’s disabilities or preferences.

2.1.2 Standards

The World Wide Web Consortium (W3C, 2019) develops guidelines and resources in order to help stakeholders to develop web applications and mobile applications that are accessible. For web content, the W3C developed the Web Content Accessibility Guidelines (WCAG), which “explains how to make web content more accessible to people with disabilities” (WCAG, 2019).

Currently, there’s no specific guideline for mobile applications, which utilizes the W3C Web Accessibility Initiative (WAI) standards and guidelines, although, a more specific guideline is under development, which will also update the requirements (WAI, 2019). Aside from these, each government has its own legislation in relation to accessibility. In Brazil, the government has the law nº 10.098 which:

> Estabelece normas gerais e critérios básicos para a promoção da acessibilidade das pessoas portadoras de deficiência ou com mobilidade reduzida, e dá outras providências (BRASIL, 2000).

In 2015, the government approved the Lei Brasileira de Inclusão (Brazilian Inclusion Law) (BRASIL, 2015), which completes the law nº 10.098 and is inspired by the UN Convention Protocol about People with Disabilities Rights, whose main objective was to ensure that people with disability have the same rights as the rest of the population (LDA, 2019).

2.1.3 Internet-of-Things and Accessibility

As previously mentioned, the IoT has the potential to benefit people with disability. It is already supporting many solutions, one of them being the Smart Homes (Abou-Zahra, Brewer, Cooper, 2017), which are a Smart Environment. Such technologies can help people with disabilities to be more independent, offering combined solutions with IoT devices and artificial intelligence systems (Abou-Zahra, Brewer, Cooper, 2017).

Although the IoT seems promising, if not combined with other solutions that also adhere to accessibility aspects, it can be more harmful than helpful, as stated by Abou-Zahra, Brewer, Cooper (2017). There are many solutions that use the IoT as a means to offer accessibility solutions, aiming to help several disabilities, such as visual impairment, mobility, among others. Although, these are neither generic nor flexible, offering architectures, systems and models that focus on a specific solution, which, although being efficient in the proposed solution, are dependent of a certain technology.

2.2 SMART CONCEPTS

This section will introduce the Smart Concepts definitions and the existing models for a Smart Campus.

2.2.1 Smart Concepts Definitions

As stated by Nam, Pardo (2011), there are many definitions for a Smart City, which has not a common definition or a single template as yet (Medina *et al*., 2017), which results in different models and applications. However, Silva, Khan, Han (2018) define a Smart City as:

> an advanced modern city that utilizes ICT and other technologies to improve quality of life (QoL), competitiveness, operational efficacy of urban services, while ensuring the resource availability for present and future generations in terms of social, economic, and environmental aspects.

From the Smart City concept, and the ascension and expansion of the IoT technology, other Smart Concepts were born. One of these is the Smart Region concept, which is an adaptation from the Smart City concept (Muhamad *et al*., 2017), which O'Brolchain *et al*. (2018) defined as:

> A high-tech intensive and advanced approach connecting people, information, governance and policies using new technologies to create an efficient, clean, energy secure, sustainable, eco-friendly, competitive and innovative region with an enhanced quality of life.

Like the Smart Region, another rising concept is the Smart Homes, which is considered a Smart Environment, that Crowley, Curry, Breslin, (2016) defined being:

> physical worlds interwoven with sensors, actuators, displays, and computational elements, embedded seamlessly into everyday objects and connected through a continuous network.

Caţă *et al*., (2015), also defined Smart Environments as a" small world where sensor-enabled and networked devices work continuously and collaboratively to make its inhabitants lives more comfortable".

Finally, like the concepts aforementioned, from the Smart City concept also emerged the Smart Campus concept, which, as affirmed by Muhamad *et al*. (2017) also does not have a common definition as yet, but can be adapted from a Smart City concept, as this has similarities of a city (Muhamad *et al*., 2017). Alghami *et al*. (2016) affirms that the objective of a Smart Campus "is to develop a campus that utilizes the resources efficiently, deliver high quality services to the campus community, while the operational cost gets reduced". It also states that a Smart Campus can bring many benefits, such as:

> provide an interactive and creative environment for students and faculty, promote smart energy management, bring effective surveillance system and real-time incidents warnings, automate maintenance and business processes, maintain efficient parking and access control management, and provide secure payments and transparent voting systems (Alghami *et al*., 2016).

All of these concepts have two aspects in common. The first is using the IoT and ICT technologies as support for the implementation of smart solutions. The second is applying the smartness aspects to its solutions. Nam, Pardo (2011) define smartness in different contexts, helping to understand the meaning and importance of this term usage in the concepts. They stated that "in marketing language, smartness is centered on a *user perspective*” (Nam, Pardo, 2011). This is an important feature, seeing that the “Smart City is required to adapt itself to the user needs and to provide customized interfaces” (Nam, Pardo, 2011). Also, they state that smartness, from a government perspective, means “distinguish their new policies, strategies, and programs for targeting sustainable development, sound economic growth, and better quality of life for their citizens” (Nam, Pardo, 2011). Although these definitions refer to a Smart City, all the other concepts can also apply the smartness to its own context.

The Smart Campus concept has yet to be formally defined, but based on the several definitions, models and architectures, and focusing on one of the required aspects of smartness, which is focusing in the users perspective, a Smart Campus model should offer flexibility, in order to adapt, update and change IoT and ICT technologies, and manage its applications, which combined with other technologies, such as Artificial Intelligence techniques, will offer better services and quality of life to its users.

This was one of the main concerns of this research, which aimed to incorporate the several aspects already defined, but adding aspects that could make the proposed model flexible, and also including accessibility aspects, this way providing better services for the users, a better understanding of the campus usage and behavior, promoting inclusion and facilitating to add, manage and share resources and data among the incorporated applications.

2.3 SMART CAMPUS MODELS

As an adaptation from a Smart City concept, the Smart Campus model can have similar components, such as energy, management, environment, among other resources. Most models focus on proposing solutions to a certain component, such as (Khatoun and Zeadally, 2016), which proposed a software platform that improves the learning and social experience on campus. Atif, Y., & Mathew, S. (2013) also proposed a learning model, which used pervasive learning as a solution.

Others, like Adamkó *et al*. (2015), proposed crowdsourcing solutions, intending to offer recommendations for the user, or gather proposed services from them. Some, proposed models to better manage campus resources like Bandara *et al*. (2016), which proposed a Smart Parking sensor network. Several proposed solutions for energy management using Smart Grids, like Labella *et al*. (2017) and Bracco *et al*. (2015).

Finally, some proposed mobility solutions, like Torres-Sospedra *et al*. (2015), enhancing the indoor, outdoor mobility on the campus. Some solutions offered accessibility features, like Tavares *et al*. (2016), which proposed an intelligent system applied to ubiquitous accessibility. None of the proposed models studied offered a flexible solution that also included accessibility aspects. Those that did propose solutions that included accessibility aspects offered specific solutions that were applied to a particular model, which was not flexible, not being capable of incorporating different solutions on the same model.

2.4 IOT HARDWARE STRUCTURE

In this section the IoT concept will be defined, also presenting its basic architecture. Moreover, the hardware utilized in the prototype developed will be introduced, explaining its functionality and application possibilities.

2.4.1 IoT Concepts

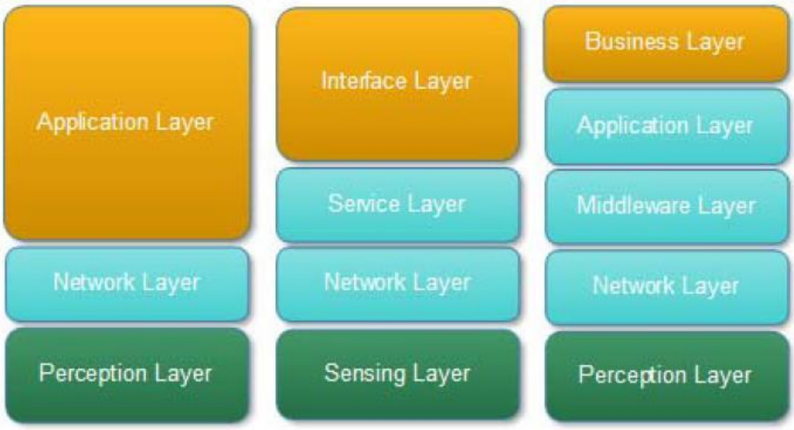
Similar to the Smart Concepts, the Internet-of-Things also has many definitions, but Ray *et al*. (2018) defined IoT as:

> a dynamic global network infrastructure with self-configuring capabilities based on standard and interoperable communication protocols where physical and virtual 'Things' have identities, physical attributes, and virtual personalities and use intelligent interfaces, and are seamlessly integrated into the information network.

2.4.1.1 IoT Architecture

The IoT allows different technologies and several applications domains, which results in a great diversity of solutions (Di Martino *et al*., 2018), and different architectures for each. The solutions can be specific or generic, resulting in different architectures for each case. Giri *et al*. (2017), demonstrate the proposed architectures for the IoT, being three of general application and one of specific application. Figure 1 demonstrates the three architectures of general application, which are disposed in layers, varying from three to five layers.

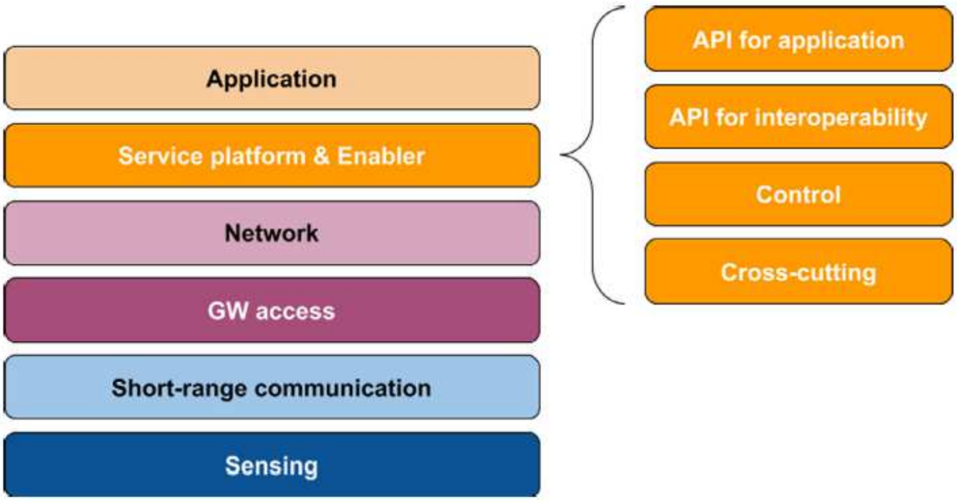
Figure 1 – IoT architectures of general application.



Source: Giri *et al*. (2017).

Borgia *et al*., (2014) also proposed an architecture based on layers (Figure 2), which as Di Martino *et al*., (2018) stated “is very helpful in outlining the interoperability and security issues typical of the IoT environment”.

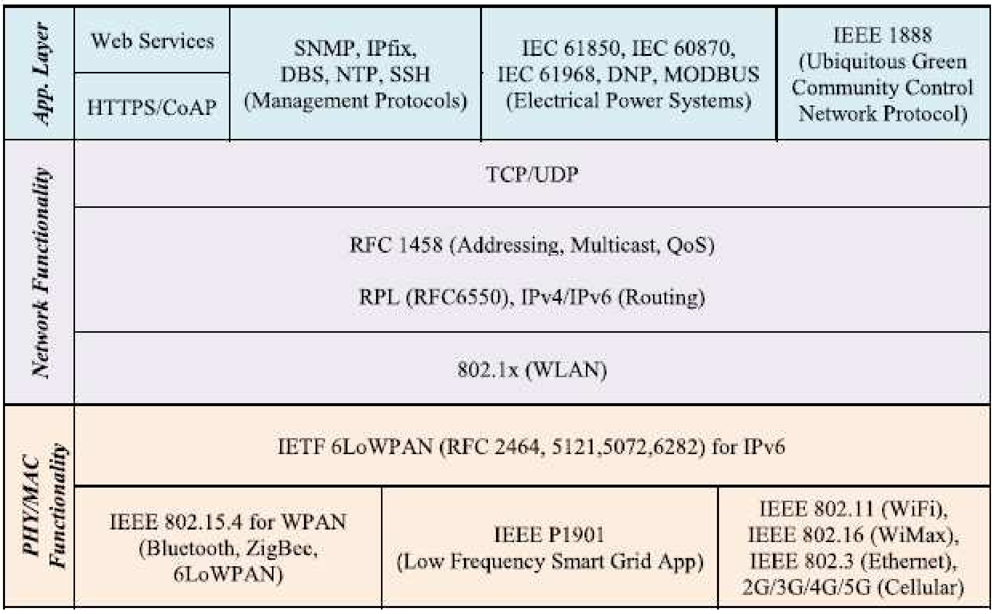
Figure 2 – Borgia *et al*, proposed layers for IoT architecture.



Source: Di Martino *et al*., (2018).

Each layer allows for several communication protocols as demonstrated by Nour *et al*. (2019) in Figure 3.

Figure 3 – IoT stacks and protocols.



Source: Nour *et al*. (2019).

The IoT layered stack protocols demonstrated by Nour *et al*. (2019) do not include the LoRa technology, which is implemented in this research. For clarification, the LoRa protocol belongs to PHY/MAC functionality stack, and the LoRaWAN protocol to the Network functionality stack.

As demonstrate by the researches previously mentioned, in order to propose an IoT architecture of general application, it should be disposed in a layered distribution, allowing independence between the layers and facilitating its management. Also, this distribution can assist in the interoperability and security issues, as mentioned by Di Martino *et al*., (2018).

2.4.2 Hardware

This section will present and detail the hardware components used for the prototype developed. Aiming to prove the model's flexibility and versatility, the prototype used two devices for the perception layer. The first device is the Beacon, which was applied in two different functions, and is presented in the first section. The second device is the centralizer, which also presented two different functions, and is detailed in the second section.

2.4.2.1 Beacons

With the expansion of the IoT technology, new protocols and devices were developed, aiming to address the low energy consume and wireless communication requirements. One of them is the Bluetooth Low Energy (BLE).

As stated by Jeon *et al*. (2018), Beacons are wireless BLE devices that are connectionless and broadcast their signal periodically. They advertise a signal that has a packet with a small data payload, "which may include the packet header, MAC address, device's unique identifier, and a small headroom for manufacturer-specific data" (Jeon *et al*., 2018).

Due to their low energy consume and small data payload, they can operate on small batteries for years, depending on their advertising configuration. Nowadays, as stated by Kolias *et al*. (2017), they are gathering attention due to their possible applications, which include "high quality object identification, proximity estimation features and their potential for location based services" (Kolias *et al*., 2017).

Currently, there are many manufacturers that configured their Beacons with their own profiles, such as iBeacon from Apple and Eddystone from Google, but aside from these, the protocol allows for developers to implement their own profile, using developing boards such as the ESP32, which is System-On-Chip (SoC) that integrates Wi-Fi (IEEE 802.11) and BLE protocols (ESP32, 2019).

Such feature allows the developer to implement a wide range of personalized applications, even associating sensors to the Beacon device. Even more, it is possible to configure the Beacon to operate as a central or peripheral device. It can be associated as client/server relationship, where the central device would be the client, which receives data from the server; and the peripheral device would be the server, repassing the data gathered to the client (Beacon ESP32, 2019).

If configured to operate as a peripheral device (server), it is possible to create a service, which can offer four properties as the characteristics of the data sending. The properties are: read, write, notify and indicate (GATT, 2019). Such features allow the device to function in two different applications, proving its versatility. In the first application, the Temperature Beacon, it was implemented using an ESP32 board that receives temperature data from a DHT11 (DHT11 Sensor, 2019) sensor via serial mode. Then, this data is sent via BLE to a centralizer device, which will be detailed in the next section. In order to send this data, the service implemented must allow the notify property in the characteristics of the service created, which then will enable that the data that was sent be received for any of the clients that connects to the service being broadcasted. In the second application, iBeacon devices were used for a localization application. A mobile application was developed, which received the RSSI signal broadcasted by the iBeacon devices and then estimated the distance in relation to the devices.

2.4.2.2 Centralizers

As a means to gather the data from the Temperature Beacons distributed across the campus, detailed in the previous subsection, it was necessary a module that both received the data via BLE protocol and repassed the received data via LoRa protocol. In order to achieve this, the RM186 module from Laird (RM1xx Module, 2019) was chosen. This module has many features and allows many configurations, having 14 GPIO (General Purpose Input/Output) ports, 10 ADC (Analog/Digital Converter) channels and the SPI, UART and I2C interface protocols. Besides, it is programmed with the *smartBASIC* language, which “enables customers to develop a complete embedded application” (RM1xx Module, 2019).

Due to its versatility, this module allowed two different applications to be implemented. In the first, the module acted as a bridge, receiving the data sent by the Temperature Beacon via BLE, and sending this data via LoRa to the gateway. In the second, the module itself collected temperature data via serial mode of a LM35 (LM35 Sensor, 2019) temperature sensor. This data was then sent to the gateway via LoRa protocol.

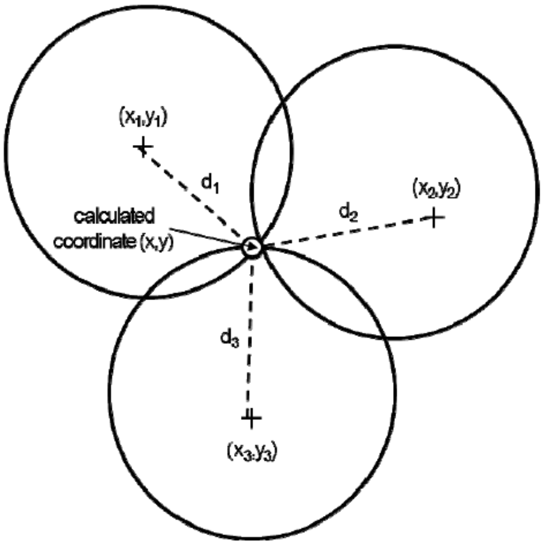
2.5 LOCALIZATION TECHNIQUES

There are many techniques to determine the user’s indoor location, but based on the research conducted, the most applied techniques for localization are Fingerprinting, Trilateration and Triangulation. In this section, the chosen technique will be explained: the trilateration technique. The Fingerprinting and the Triangulation techniques will be explained in the Appendix A.

2.5.1 Trilateration Technique

The trilateration technique utilizes at least three devices in order to determine the user's location, measuring the distance between the devices and the user's smartphone, as shown in Figure 4.

Figure 4 - Trilateration technique example.



Source: https://www.researchgate.net/figure/Trilateration-technique_fig3_267324735

To determine the location, firstly, it is necessary to calculate the distances from each device, using the equation demonstrated in Equation 1 (Gorovyi *et al.*, 2017)(Huh *et al.*, 2017).

Equation 1 - Distance to device equation.

$$d = 10^{\left(Tx-RSSI/_{10n}\right)}$$

Source: Elaborated by the author.

Variables Equation 1:

- Tx: Is the value determined by the manufacturer, representing the received signal strength indication (RSSI) of the device at the distance of one meter;

- RSSI: Is the received signal strength indication (RSSI) being transmitted from the device;
- n: Is the attenuation constant of the environment;
- d: Distance from the target device.

Equation 2 presents the equation to determine the attenuation constant of the environment (Huh *et al.*, 2017).

Equation 2 - Attenuation constant equation.

$$N = \frac{RSSI - Tx}{-10\log_{10} d}$$

Source: Elaborated by the author.

After, the distances from each device are applied in Equation 3, as means to determine the user's location in relation to the devices (Huh *et al.*, 2017).

Equation 3 - Trilateration equation.

$$A: (x - B1x)^2 + (y - B1y)^2 = dB1^2$$
$$B: (x - B2x)^2 + (y - B2y)^2 = dB2^2$$
$$C: (x - B3x)^2 + (y - B3y)^2 = dB3^2$$
$$D: A - B$$
$$E: B - C$$

Source: Elaborated by the author.

Variables Equation 3:

- B1: First Beacon device;
- B2: Second Beacon device;
- B3: Third Beacon device;

- x, y: Coordinates of the user's location;
- dB1: Distance from first Beacon device;
- dB2: Distance from second Beacon device;
- dB3: Distance from third Beacon device.

After resolving the equation, the user's location is determined. This technique can be applied using more devices, this way improving the accuracy of the user's location (Huh *et al*., 2017).

2.6 DATA ANALYSIS TECHNIQUES

Aiming to analyze the gathered data, a data analysis unit was incorporated in the model, which will be further explained in section 4, which will present the proposed model.

This unit will use both Artificial Intelligence techniques and data analysis techniques to analyze the gathered data. For this research, clustering techniques, which are artificial intelligence techniques, were applied, using the K-Means and the HDBSCAN techniques to analyze the location data gathered and organize it in clusters representing aspects of the user's mobility in the campus. Both techniques will be presented in the following subsections. Aside from these, other data analysis techniques were applied to the data, organizing both the location data and the temperature data obtained in a time series. Also, for the location data will be presented the hourly frequency of a cluster, which is demonstrated in section 6.

2.6.1 Clusters

As stated by Tan, Pang-Ning, Steinbach, and Kumar (2013), "Cluster analysis divides data into groups (clusters) that are meaningful, useful, or both", capturing the natural structure of the data. Also, it can be the starting point for other analysis, summarizing the data (Tan, Pang-Ning, Steinbach, and Kumar*,* 2013). Besides, according to Xu, Rui, Wunsch (2005), clusters are an unsupervised classification system, which divides a group in homogenous subgroups based on a determined measure of similarity.

Clustering techniques can be classified as hierarchical and partitional. Hierarchical clustering divides the group with a sequence of partitions, forming a structured tree, while partitional clustering divides the group into a pre-defined number of clusters (Xu, Rui, Wunsch, 2005). This research applied two clustering techniques, the K-Means technique, which is classified as partitional clustering, and the HDBSCAN, this being a variation of the DBSCAN technique, which is a density-based algorithm, both being classified as hierarchical clustering.

2.6.1.1 K-Means Technique

According to Tan, Pang-Ning, Steinbach, and Kumar, (2013), K-Means is a partitional clustering technique which uses pre-stablished numbers (K) introduced by the user in order to find the clusters. Each cluster has a centroid, which is the mean of a group of points (Tan, Pang-Ning, Steinbach, and Kumar, 2013).

The algorithm starts by choosing a number of centroids, as aforementioned this number of centroids (clusters) are pre-established. After, each point is assigned to the closest centroid, and the group of points assigned to a centroid is a cluster. After, the centroids are updated, based on the group of points assigned to its cluster. Figure 5 presents the basic K-Means algorithm.

Figure 5 – K-Means algorithm.

Algorithm 8.1 Basic K-means algorithm.

```
Select K points as initial centroids.
repeat
   Form K clusters by assigning each point to its closest centroid.
   Recompute the centroid of each cluster.
until Centroids do not change.
```

Source: Tan, Pang-Ning, Steinbach, and Kumar, (2013).

2.6.1.2 HDBSCAN Technique

According to HDBSCAN (2019), the HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise) is a variation from the DBSCAN clustering algorithm, converting it into a hierarchical clustering algorithm, and then extracting a flat clustering. The DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a partitional clustering, which determines the number of clusters automatically, besides classifying points in low-regions as noise and omitting them, which is the reason it does not produces a complete clustering (Tan, Pang-Ning, Steinbach, and Kumar, 2013). As stated by Tan, Pang-Ning, Steinbach, and Kumar, (2013), "density-based clustering locates regions of high-density that are separated from one another by regions of low density". Based on this, the algorithm estimates the density of a certain point by counting the number of points within a pre-established radius (*Eps*). A point can be classified as a core point, a border point or as noise/background point. This classification is based on the density of the point (Tan, Pang-Ning, Steinbach, and Kumar, 2013). Figure 6 presents the algorithm.

Figure 6 – DBSCAN Algorithm.

Algorithm 8.4 DBSCAN algorithm.

1: Label all points as core, border, or noise points.
2: Eliminate noise points.
3: Put an edge between all core points that are within *Eps* of each other.
4: Make each group of connected core points into a separate cluster.
5: Assign each border point to one of the clusters of its associated core points.

Source: Tan, Pang-Ning, Steinbach, and Kumar, (2013).

As prior mentioned, the HDBSCAN algorithm is a variation from the DBSCAN algorithm, where the main difference between them is that the HDBSCAN allows varying density clusters. It uses a parameter that defines the minimum cluster size, which instead of using the Eps parameter to classify a point, it uses the minimum cluster size parameter to condense the dendrogram (Tan, Pang-Ning, Steinbach, and Kumar, 2013). Figure 7 presents the HDBSCAN algorithm.

Figure 7 – HDBSCAN Algorithm.

1. Transform the space according to the density/sparsity.
2. Build the minimum spanning tree of the distance weighted graph.
3. Construct a cluster hierarchy of connected components.
4. Condense the cluster hierarchy based on minimum cluster size.
5. Extract the stable clusters from the condensed tree.

Source: https://hdbscan.readthedocs.io/en/latest/how_hdbscan_works.html

3 RELATED WORKS

This section will present the related works selected for this research. In order to identify the related works that best relate to the research question defined in section 1.1, a systematic literature review was conducted. The research method started by identifying and elaborating the most relevant questions pertaining to the research theme. After, the selection criteria were designed. Then, the research in the data bases was made, which posteriorly, based the articles selection on the criteria previously defined. After the selection, it was then extracted the relevant data from these, and later on, this data was synthetized, as it will be presented in section 3.5.

3.1 RESEARCH QUESTIONS

Intending to answer the research question, which involved different domains, the systematic literature review research questions were divided in two classes, the first involving the Smart Concepts, which included the Smart City, Smart Region and Smart Campus concepts, and the second being the challenges presented for IoT devices and architectures when applied in the Smart Concepts. The systematic literature review research questions are presented in Table 1.

Table 1 – Research questions table.

Reference	**Question**
Smart Concepts	
RQ1	Is there a specific model being applied to a Smart Campus?
RQ2	Which definition better describes the smart concept?
RQ3	Which are the basic requirements to consider a campus a Smart Campus?
RQ4	What is the best architecture/model to apply into a Smart Campus?

RQ5	Is there a model that integrates accessibility aspects?
IoT	
RQ6	What are the challenges encountered when using IoT devices?

Source: Elaborated by the author.

3.2 RESEARCH CRITERIA

The selection criteria stated that the articles selected should respect the following rules:

- Have more than four pages;
- Have to be in English or Portuguese;
- Have to be peer reviewed;
- Published in journals or conferences between 2015 and 2019;
- Offer open access.

The search bases selected were the IEEEXplore, the ACM Digital Library, Science Direct and Google Scholar.

3.3 RESEARCH STRINGS

The search strings used as well as the respective results is shown in Table 2.

Table 2 – Research strings table.

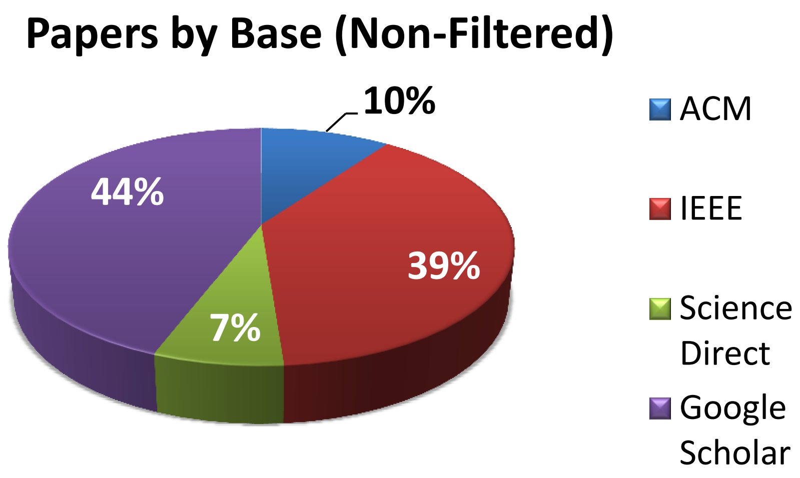
Query	Search Strings	IEEE	ACM	Science Direct	Google Scholar
1	((((smart campus) OR smart university) AND model) AND accessibility)	0	0	4	0
2	(((smart campus) OR smart university) AND model)	7	0	5	4
3	(((smart campus) OR smart university) AND concept)	2	0	0	1
4	(((smart campus) OR smart university) AND survey)	2	1	1	1
5	(((smart campus) OR smart university) AND internet-of-things)	15	2	0	13
6	(((smart campus) OR smart university) AND review)	3	0	0	8

7	((smart city) AND model) AND accessibility)	1	2	1	0
8	((smart city) AND model)	123	49	26	80
9	((smart city) AND survey)	23	7	5	22
10	((((smart campus) AND model) AND visual impairment)	0	0	0	0
11	(((IoT) AND survey) AND challenges)	9	2	0	30
12	((IoT) AND survey)	109	12	11	175
	TOTAL	**294**	**75**	**53**	**334**

Source: Elaborated by the author.

The papers distribution by base is shown Figure 8.

Figure 8 - Papers distribution by base (non-filtered).



Source: Elaborated by the author.

As shown in Figure 8, the bases that presented more papers, without being applied the papers exclusion criteria, were the IEEEXplore and the Google Scholar.

3.4 ARTICLES SELECTION

After applying the research strings in the four bases and collecting the results for each query, the selection criteria was applied, resulting in the selection displayed in Table 3.

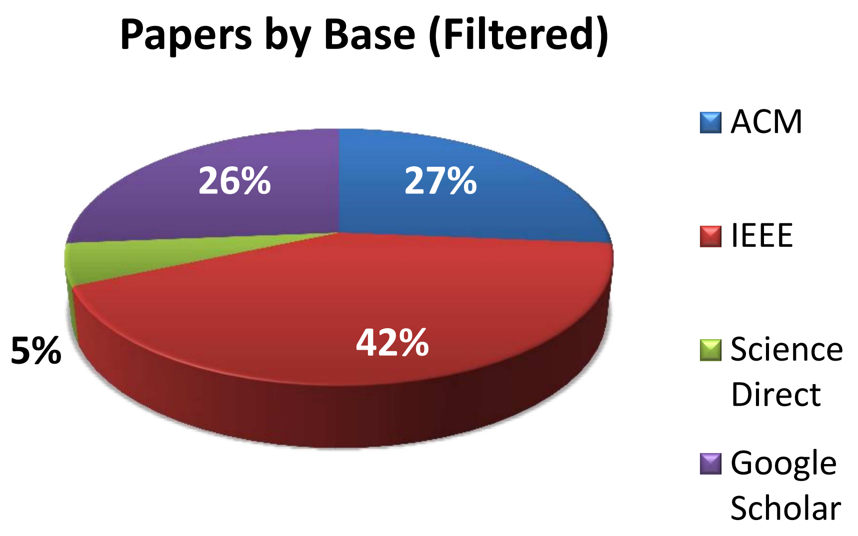
Table 3 – Papers selection by base.

Search Bases	ACM	IEEE	Science Direct	Google Scholar	Total
Papers Search Result	75	294	53	334	756
Papers Selected	5	8	1	5	**19**

Source: Elaborated by the author.

Figure 9 presents the distribution by base of the selected papers.

Figure 9 - Papers distribution by base (Filtered).



Source: Elaborated by the author.

As shown in Figure 9, the bases that presented more papers, after applying the papers exclusion criteria, were the IEEEXplore and the ACM.

3.5 RESEARCH RESULT

This section presents the research results of the papers previously selected, seeking to answer the questions listed in section 3.1, as well as the result analysis.

3.5.1 Smart Campus Model

The research resulted in articles describing models for different purposes and applications. Some asserted that there is not a common definition of a Smart Campus (Muhamad *et al*., 2017; Mattoni *et al*., 2016; Ferreira and Araújo, 2018; Caţă, 2015; Alghamdi and Shetty, 2016), which results in many interpretations, based on different concepts and definitions, or by applying adaptations from Smart City concepts. Some papers proposed models that are developed for a specific domain, like a Smart Grid, focusing on proposing specific solutions to a specific problem; few proposed models for a Smart Campus and none integrated accessibility aspects on its model. Table 4 presents the comparison of the selected papers to answer the RQ1 question. The only paper that proposed a solution which integrated accessibility aspects was the intelligent system solution proposed by Tavares *et al* (2016) (sixtieth item in Table 4).

As demonstrated in Table 4, that are not many papers proposing Smart Campus models, which was the reason why papers that presented a review, features and concepts of a Smart Campus were chosen, seeking to understand which the Smart Campus features, concepts and basic requirements were, which will serve as a base knowledge in the development of the model being proposed.

Table 4 - Smart Campus Model – Papers Comparison.

	Smart Campus Papers									
	Paper Presentation			**Paper Content Summary**	**Features**				
	Title	**Authors**	**Year**		**Presents Model/ Architecture**	**Has Accessibility Aspects**	**Presents a Review**	**Presents a Methodoly**	**Presents an Intelligent System**
1	Smart Campus Features, Technologies And Applications: A Systematic Literature Review	Wardani Muhamad, Novianto Budi Kurniawan, Setiadi Yazid	2017	Specifies six intelligence domains that a Smart Campus must have: learning, governance, green, health, social, management. Also presents the applications for each domain, detailing each application features	No	No	Yes	No	No
2	A matrix approach to identify and choose efficient strategies to develop the Smart Campus	Bendetta Mattoni, Francesca Pagliaro, Giulio Corona, Vito Ponzo, Fabio Bisegna, Franco Gugliermetti, Margarito Quintero-Núñez	2016	Define a methodology to develop a Smart Campus by defining the better strategy that will attend the Campus requirements using a matrix that evaluates several factors. Defines five fields of action: People & Living, and Mobility,Economy, Energy, Environment	No	No	No	Yes	No
3	Campus Inteligentes: Conceitos, aplicações, tecnologias e desafios	Ferreira; Araújo	2018	Presents the necessary requirements for the construction of a Smart Campus. Also presents its concepts and definitions.	No	No	Yes	No	No
4	Smart university, a new concept in the Internet of Things	Marian Cață	2015	Presents an online platform in which the users are able to report irregularities on campus. Presents the advantages of implementing a Smart University.	Yes	No	No	No	No
5	Survey toward a smart campus using the internet of things	Abdullah Alghamdi, Sachin Shetty	2016	Presents the challenges of implementing a IoT network. Presents the advantages of implementing a Smart Campus and also its concept. Presents solutions that can be applied to a Smart Campus.	No	No	Yes	No	No
6	Hefestos: an intelligent system applied to ubiquitous accessibility	João Tavares, Jorge Barbosa, Ismael Cardoso, Cristiano Costa, Adenauer Yamin, Rodrigo Real	2016	Proposes an intelligent system applied to ubiquitous accessibility. The proposed model uses ubiquitous computing concepts to manage accessibility resources for people with disabilities, testing the developed system in a Campus	No	Yes	No	No	Yes

Source: Elaborated by the author

The articles describe a model using the ICT (Information and Communication Technology) as means to develop the models, allying the IoT (Internet-of-Things) technology with software and internet infrastructure, this way helping on the campus management. The articles also propose solutions for the campus resources, where each solution presented had their own system/model and the IoT device that best met the requirements of this resource.

Some articles focused on improving just certain aspects of a Smart Campus, like developing a software platform that improves just one domain in a campus, like learning and social experience, while others focused on how to manage the campus many resources, integrating each as a part of the system, (Muhamad *et al.*, 2017), where each of its resources were turned smart, this way optimizing their operation. Some of these systems were water and waste management, energy grid, and common resources from a university, like the library, the buildings that contain the classrooms and the parking lot, among others. Other articles focused on just one of these systems, like the energy grid. The research couldn’t find an article that included accessibility aspects in a Smart Campus model, but only papers that proposed software solutions, like the solution proposed by Tavares *et al.* (2016), which presents an intelligent system which treated an accessibility problem, finding accessibility aspects included in models only when making queries related to Smart City solutions. As aforementioned, the Smart Campus concept does not have many models proposed. For this reason a broader search was conducted, including Smart City models which is presented in Table 5.

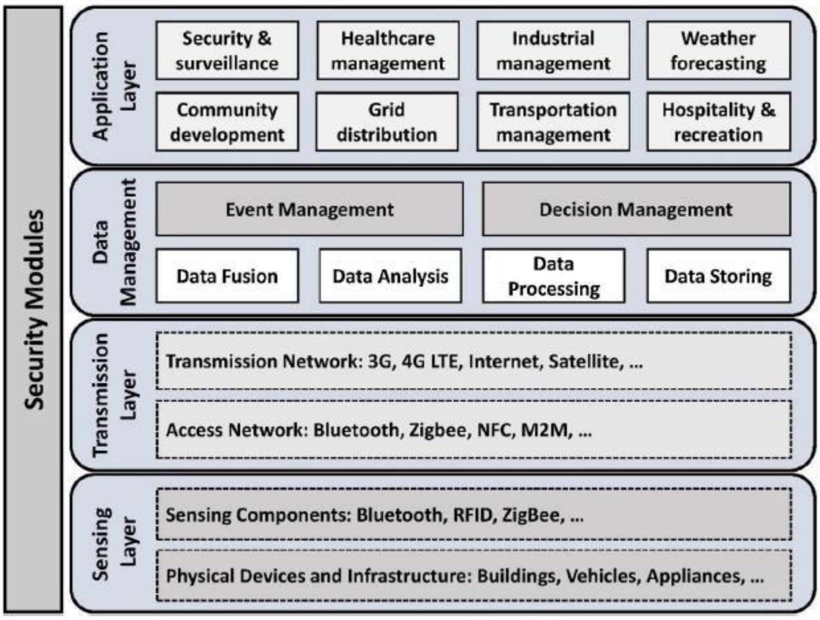
Table 5 - Smart City Model – Papers Comparison.

	Smart City Papers						
	Paper Presentation				**Features**		
	Title	**Authors**	**Year**	**Paper Content Summary**	**Presents Model/ Architecture**	**Has Accessibility Aspects**	**Presents a Review**
1	Towards sustainable smart cities: A review of trends, architectures,components, and open challenges in smart cities	Bhagya Nathali Silvaa, Murad Khanb, Kijun Hana	2018	States that a Smart City is an urban environment which uses ICT, and other technologies, to improve the performance and efficiency of a regular city, also improving the quality of the services provided to its citizens. It also presents the Smart City definitions and the future challenges for the cities, based on estimates, showing the necessity for the regular cities in a Smart City. It presents an architecture, composed of four layers: *sensing*, *transmission*, *management* e *application layer*, presenting the challenges for each layer and describing several sectors of a Smart City. States that developing a smart city architecture is theoretically feasible, but in practice becomes unrealistic due to the divergent characteristics of these	Yes	No	Yes
2	Making Smart and Accessible Cities: An Urban Model based on the Design of Intelligent Environments	Raquel Pérez-delHoyo, Clara García-Mayor, Higinio Mora-Mora, Virgilio Gilart-Iglesias, María Dolores Andújar-Montoya	2016	Presents an implementation model of accessibility, identifying which disability and the limitations generated by it, and through this, proposes measures to be taken by the urban environment so that the disabled can have accessibility. Presents possible technologies to apply. It also states that Smart cities should look for ways to identify a person with a disability in an urban environment, and adapt the environment for that person, showing routes, among other benefits. It further states that smart urban environments are those that adapt to people with disabilities.	Yes	Yes	No
3	A Model for Adaptive Accessibility of Everyday Objects in Smart Cities	Ilaria Torre, Ilknur Celik	2016	It uses virtualization of physical objects and context adaptation as a means of accessibility, applying RWO tags to objects and performing actions that facilitate their use by the user. The interface is adapted to its deficiency by storing user information on a server. Objects can also have sensors added, making the environment smart and making it easier for the user to adapt.	Yes	Yes	No
4	Smart City implementation and discourses: An integrated conceptual model.The case of Vienna	Victoria Fernandez-Aneza, José Miguel Fernández-Güellb, Rudolf Giffingerc	2018	The paper presents several models described in the literature and presents the problems identified in the presented models, besides proposing a conceptual model and its evaluation methodology.	Yes	No	No
5	IoT Paradigm into the Smart City Vision: A Survey	Camilo Alejandro Medina, Manuel Ricardo Pérez, Luis Carlos Trujillo	2017	It states that there is no clear definition of the Smart City concept. It gives you an overview of Smart City and how IoT technologies can be the catalyst for this process. Features a Smart City model with the inclusion of IoT technology. Describes about various IoT devices and protocols and how they can operate within a Smart City. Features several cases.	Yes	No	Yes
6	Security and privacy challenges in Smart Cities	Trevor Braun, Benjamin C.M.Fung, Farkhund Iqbal, Babar Shah	2018	Discusses how security and privacy should be taken into account when implementing a Smart City. It emphasizes that the adhesion of the population is fundamental for the development of a Smart City.	No	No	Yes
7	Conceptualizing Smart City with Dimensions of Technology, People, and Institutions	Taewoo Nam, Theresa A. Pardo	2011	It features a variety of Smart City settings, and features that make it smart. It also presents the current difficulties of cities and the reasons why they need to become intelligent. It states that the systems of a Smart City operate together and are treated as a network.	No	No	Yes
8	Smart cities: concepts, architectures, research opportunities.	Rida Khatoun, Sherali Zeadally.	2016	It presents definitions for Smart City and some features and benefits of it, in addition to its essential components. It also presents the ISO standards being developed to evaluate the performance of a Smart City, and the challenges in its development, being one of these Smart Citizens.	No	No	Yes

Source: Elaborated by the author

The papers that proposed models, presented also architectures that were structured in layers, varying mostly the layers responsibilities and its technologies. For example, Silva, B. N., Khan, M., & Han, K. (2018), proposed the structure presented in Figure 10, which divided its architecture in four layers: sensing layer, transmission layer, data management layer and application layer.

Figure 10 – Smart City Architecture Example



Source: Silva, B. N., Khan, M., & Han, K. (2018)

As the Smart Campus, many of the proposed Smart City models also presented architectures that were for specific domains, and few presented solutions that included accessibility. However, many papers proposed theoretical solutions, but few tested their solutions, which could be difficult due to the complexity of implementing the proposed architectures on a large scale. As the Smart Campus, the Smart City also does not have a specific model.

3.5.2 Smart Concept Definition

As mentioned before, there's not a common definition about the Smart Campus, but there are many concepts about it and allied to that, definitions about the Smart City concept, which can be adapted to a Smart Campus, as it has similarities of a city (Muhamad *et al*., 2017). Another smart concept associated with Smart Campus is the Smart Environment, which is defined as a "small world where sensor-enabled and networked devices work continuously and collaboratively to make its inhabitants lives more comfortable."(Caţă *et al*., 2015).

As stated by Muhamad *et al*. (2017), there’s not a technical committee that shades the Smart Campus concept, which results in not having a standard, this way hampering the process of development and implementation of it. Despite this, there are many concepts, but to consider a campus a Smart Campus, it is not enough to just add technology to it, turning it into a digital campus (Abuarqoub *et al*., 2017), but to ally the concepts of smartness with the technology, this way providing and delivering high quality services in real time, reducing operational costs, improving performance and control over the campus resources and facilities, this way facilitating campus management, which in turn promotes inclusion, improves safety, security and mobility inside the campus, offering a better experience to campus’s users.

Like the Smart Campus concept, the Smart City concept does not have a common definition (Medina *et al*., 2017). Allied to that, the common denominator among the varied definitions is the use of ICT technologies and IoT to develop and support a Smart City. As previously stated, a smart entity is one that merges smartness concepts and technology, using it to facilitate its management, to improve the control over its resources, as well as the quality of life of its users, also being capable to address and adapt to challenges. Besides, it has to promote competitiveness, operational efficacy of urban services, sustainability and availability (Silva *et al*., 2018; Nam and Pardo, 2011).

3.5.3 Smart Concept Requirements

Most of the Smart Campus concepts inherited some of its requirements from the Smart City concept, which usually is adapted to a Smart Campus. Both concepts claim that to be considered smart, the entity must have interoperability across all system, ensuring the security from its data and devices, ensuring the safety and security of the users, protecting their identity and privacy, manage the large volume of data and support scalability (Santana *et al*., 2017). It also must have standards that protect the system against cybercrimes and security threats, besides making use of the energy resources in an efficient way (Mattoni *et al*., 2016; Swaroopa and Chaitra, 2016).

Besides that, (Muhamad *et al*., 2017; Ferreira and Araújo, 2018) affirms that the campus has to use the ICT in a pervasive way and apply the technology in order to offer the users a better quality of life on campus. The system has to respond and adapt quickly and on-demand, responding to requests of different origins and contexts, provide availability of services and information, have procedural knowledge for a better decision making, provide integration and monitor campus evolution. Also, the system must be interconnected, control and monitor diverse systems, such as energy consume, parking, among others. Furthermore, it has to be capable of acquiring and delivering data about the learning process, in order to boost the analysis and improve the learning environment. Moreover, the campus must deliver high quality services to the campus community, besides reducing the operational costs (Abuarqoub *et al*., 2017; Mattoni *et al*., 2016).

3.5.4 Smart Campus Architecture

The articles focused on proposing platforms and software solutions to be applied in a Smart Campus, using a model or architecture, intending to improve and facilitate the management of the campus resources, the campus system and the mobility inside of it or offering solutions that would help evolve and improve the learning process and methods.

Although the Smart Campus can be an adaptation from a Smart City model, there is not a generic model that offers a solution that contemplates all the items aforementioned together, integrating accessibility aspects. The best model to be applied should be generic and should include the requirements described in section 3.5.3, together with the aspects specified in the definition, described in section 3.5.2, this way covering all the challenges from both concepts, allowing a generic implementation, and promoting a feasible solution, which could be applied in any context.

3.5.5 Smart Campus Model and Accessibility

As aforementioned, none of the articles researched about Smart Campus proposed a model which included accessibility aspects, this way, a research on models that integrated accessibility solutions for Smart Cities was conducted. Some of the resulting articles proposed accessibility solutions for Smart Cities, suggesting adaptation of context according to the disability (Pérez-delHoyo *et al*., 2016), virtualization of physical objects and context adaptation (Torre and Celik, 2016), or ubiquitous solutions and intelligent systems (Tavares *et al*., 2016; Telles *et al*., 2016). As shown in Table 2, the articles that proposed solutions for a Smart Campus are few and do not present a model that integrates accessibility as a requirement. The models and solutions that did treat accessibility challenges were not flexible and mostly applicable to specific contexts and conditions.

3.5.6 IoT Challenges

Many articles emphasized the challenges of implementing IoT solutions, demonstrating that the complexity is proportional to the IoT network size, which escalates the challenges along with the number of devices connected, generating problems such as insecurity, complex interoperability and integration, which originated from a common feature of the devices being developed currently, where every device has its own protocol stacks, data format and architecture (Farhan *et al*., 2017), and no standardization (Alaba *et al*., 2017). In order to connect heterogeneous devices and treat the communication problem, it is indicated a flexible layer, which would support the devices (Giri *et al*., 2017) or to provide a communication that is adaptive to allow dynamic interconnectivity and support decentralized nature (Farhan *et al*., 2017).

As already mentioned, one of the biggest challenges is the interoperability issue, which consists of identifying the device in the network, and making the “translation” from different protocols, both in syntax and semantic. To manage it, it is suggested to use gateways (Farhan *et al*., 2017), APIs or ontologies (Giri *et al*., 2017).

Another challenge encountered is the privacy and security issues, where the data can be accessed and compromised in many ways, being it through the devices, altering their configuration; through the communication, altering the data being transferred; through the network layer, creating a false node and feeding false information, among others (Alaba *et al*., 2017). In order to treat or prevent that, many solutions are provided, such as treating the security issue on the internet, assuring the safety of the data stored on the web (Farhan *et al*., 2017), or treating it in the application layer, applying several techniques to ensure and validate that the data being used is correct (Alaba *et al*., 2017). Besides these solutions, it is recommended to utilize communication protocols that provide encryption, so as to secure the communication (Alaba *et al*., 2017).

Furthermore, the IoT should provide access control, in order to ensure the data access only to authorized users, avoiding privacy and security issues, like data tampering (Farhan *et al*., 2017). In addition, another challenge is the system scalability issue, which could affect the quality-of-service (QoS) of the system, this way affecting the quality of the service provided, as well as making overuse of the resources, increasing the cost of the system (Giri *et al*., 2017). This leads to another key issue for IoT systems, the energy consume. Due to devices constrained resources, most devices have a low energy consume, and consequently, low energy resources (Ali *et al*., 2017; Alaba *et al*., 2017), which could result in many issues, from security and privacy to functionality, if not handled correctly.

Moreover, with the increase in the use of IoT devices and the expansion of the networks, this including several devices, the chance of a fault happening has increased as well, resulting in the necessity to adhere fault tolerance in the system (Farhan *et al*., 2017). This would assure the correct functionality of the system, by identifying, mitigating and treating the fault, this way not compromising the system functionality, although this could add an overhead to the system, which increases both resources and costs, which may turn the system unfeasible.

Also, the increase in use of IoT devices produced more data that is transmitted at a high speed, which results in a large volume and variety of data. This categorizes the 5V (volume, velocity, value, variety, veracity) challenge for big data (Farhan *et al*., 2017). This is being treated by utilizing intelligent algorithms, capable of processing this large volume.

Moreover, another issue is the lack of standards, which makes it difficult the development and implementation of systems, since there is not a defined data protocol, communication protocol and architecture, this way resulting in no interoperability as aforementioned, besides resulting in systems that are less secure and more expansive.

3.5.7 Analysis

As mentioned in many studied articles, the Smart Campus concept presents many challenges, from infrastructure to people. Besides not having a standard and a definition, the challenges to conquer are many, initiating from the IoT and ICT technologies, where there's not a standard data type, which results in no interoperability, making it difficult to integrate the data generated from the diverse devices. Besides that, the system presents other challenges, such as: scalability, supporting the load increase on the network without affecting the quality of service; connect heterogenic resources; guarantee data privacy and security, from cloud to devices; have the necessary infrastructure to function; gauge the model acceptance from users; analyze the consequences and influences generated.

Aside from this, another challenge for the smart concepts, being it Smart Campus or Smart City, is the population. Due to different cultures, educations and economic contexts, the model adhesion can be problematic, and for the model to succeed, it needs the collaboration from population. Another challenge is the infrastructure. It also depends heavily on the economic context, being this one of the main problems, seeing that is the base for the information and communication technologies, which could compromise the whole system if it's not working correctly.

However, the articles also presented the basic requirements of a smart model, such as being user-centric, focusing in how to best adapt the application and its functionality for the user, offering better services and improving their experience and quality of life; be highly integrated and highly interconnected; cost-efficient; trustable and collaborative. Finally, the articles also presented a deficiency of models when addressing accessibility aspects, producing solutions that are not flexible and generally applicable in few contexts.

4 SAPIENTIA MODEL

In this section will be presented the proposed model for a Smart Campus which aims to promote flexibility, include accessibility aspects and be capable of incorporating existing solutions. This model was developed with the objective of attending the challenges and issues previously listed and identified in section 3. The proposed model was named Sapientia that in Latin means wisdom, which reflects one of the many principles necessary when designing a Smart Campus solution, which requires knowledge, experience, intelligence and good judgement, in order to promote inclusion and offer better experiences and services, which are supported through IoT and ICT technologies, resulting in a better quality of life for its users.

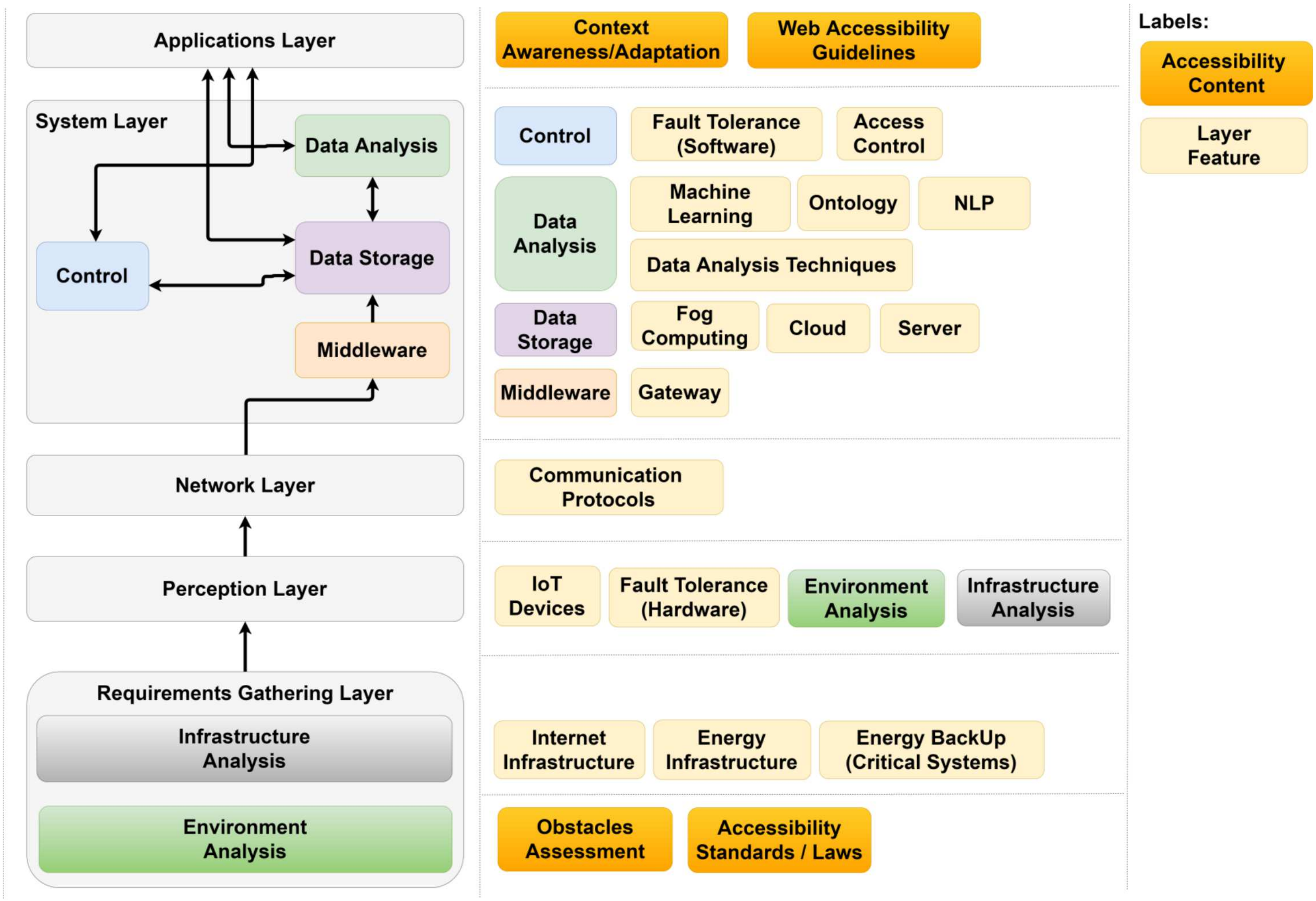
This section starts by presenting the proposed model architecture, followed by the detailing of each layer. Next, in section 4.2, the model's differentials are presented.

4.1 ARCHITECTURE

The model aims to offer a flexible solution that will include accessibility aspects and be able to incorporate existing hardware and software solutions in a Smart Campus. The model's infrastructure allows for existing projects and applications to be easily incorporated due to its layered infrastructure, which facilitates the integration and which will benefit both applications and projects by allowing them access to new information and data, by sharing the resources, such as the data storage unit or the devices and technologies integrated in the structure. The model's structure also supports accessibility aspects, helping to guide new solutions to develop features that support and offer accessibility aspects, which can be mobility or information access to its users. Also, the model's structure helps the solution stakeholders to better manage their solution, reducing the solution development time and the project costs by analyzing and planning the possible challenges and problems beforehand.

Figure 11 demonstrates the proposed model architecture.

Figure 11 – Proposed model.



Source: Elaborated by the author.

The model was structured in a layered form, following the tendency demonstrated in section 3.5.1, where the existing models for Smart Cities and Smart Campus were presented. This structure makes the layers independent from each other, this way facilitating the management, updating and exchange of the technologies being used in it, besides defining each layer responsibility. Although the model proposes the possible technologies to be applied for each layer, it can also serve as a guide in the development or the incorporation of a solution, allowing the addition of other technologies, or even simplifying the responsibility of a layer, this being defined by the requirements presented by the solution. The following sections will detail the layers and its features.

4.1.1 Requirements Gathering Layer

The main objective of this layer is to define the necessity of a process to gather the requirements for the solution. It aims to analyze the environment and the available resources in it, in order to determine the impact that the solution will generate on it and to identify the challenges that it will generate for the solution. As a Smart Campus, one of the many requirements existing is for it to make a better use of the available resources, in a sustainable way. Besides, as one of the proposed differentials for this model is to be able to incorporate existing hardware and software solutions, it must consider that the available resources will vary with the different cultures, economies, legislations and infrastructures. Also, this process will help identify the possible challenges and difficulties for the application being developed, facilitating its planning and management, aiming to minimize the costs and the development time. The two following sections will present the processes responsible for this analysis.

4.1.1.1 Environment Analysis

This process aims to analyze the environment for accessibility obstacles and problems that could jeopardize the functionality of the devices that will be installed for the Perception Layer. An accessibility specialist should determine such obstacles and possible dangers in the environment, and an IT specialist should determine the problems and challenges that the environment can present for the devices functionality.

In addition, in this process is verified if the accessibilities standards and laws defined by the country legislation are respected and followed. In case they are not, the necessary adaptations should be done. Based on both assessments, solutions for the identified challenges should be proposed and planned, aiming to solve the challenges listed, aiming to optimize the time of development, which will result in lower costs for the application. Aside from this, the proposed solutions should be careful in not generating new challenges in the environment, like for example, adapting the environment for a determine disability which could generate problems for another.

4.1.1.2 Infrastructure Analysis

This process will analyze the campus infrastructure condition, verifying if the current infrastructure will be able to support the requirements of the application being developed, as well as the changes that will be applied to it, if necessary.

Concepts such as energy and networking will be analyzed, verifying aspects such as the following examples: if the network will be capable of sustain and support the new technologies and devices; if the devices will need a permanent font of energy, and if so, if the infrastructure is able to supply it; if the devices will harvest their own energy, and what adaptations will be needed for it; if the application will need a critical system; if it will need an energy backup system, among others.

Also, it needs to be analyzed if the current infrastructure will be able to support the devices installation, and if not, which are the necessary adaptations for it to do so and what impacts this installation will generate on the environment.

4.1.2 Perception Layer

In this layer will be treated the challenges encountered regarding the IoT devices installation and their functionality. The environment analysis will be an input in this layer, demonstrating the possible obstacles and challenges encountered in the environment. Furthermore, the infrastructure analysis will also be an input in this layer, demonstrating which infrastructure problems could affect the system, and making possible to adapt the system to current resources or improve the infrastructure in order to support the system requirements.

Another challenge treated in this layer is the arrangement of the devices in the environment, which should not interfere with the environment mobility, besides being aware of the communication coverage and the energy consumption, aiming to be as sustainable as possible without impairing the device functionality and assuring the quality of service. To achieve it, the devices arrangement has to take into account the results in both the infrastructure and the environment analysis, ensuring that the desired quality of service will be provided.

Another item analyzed in this layer is the fault tolerance, which is recommended for critical systems that aims to identify, mitigate and treat the faults occurred in the devices. When applied it should be analyzed which systems will really need it, ensuring that the resources overhead added will not affect the functionality of the system and the project budget, seeing that it is an expensive resource, both on costs and in resources usage.

4.1.3 Network Layer

This layer is responsible for the communication between the Perception Layer and the System Layer, acting as a bridge between the two layers, delivering the data acquired from the IoT devices in the Perception Layer, to be analyzed and processed in the System Layer. This is achieved by the communication protocols, which can function by wireless or cable, depending on the application being developed and the protocol that best attend to the application's requirements. In order to ensure the data security, it is recommended to use communication protocols that offer encryption.

4.1.4 System Layer

This layer is responsible for filtering, storing, processing, analyzing and controlling the access to the data that was acquired by IoT devices on the Perception Layer. It is divided in four units: Middleware, Data Storage, Data Analysis and Control Unit.

The first unit is the Middleware Unit, which is responsible for the data interoperability and for filtering the data received from the devices. It is recommended using a data filter, in order to avoid data redundancy, besides a network bandwidth overload, achieving it by not sending redundant data to the Cloud. For this unit, it is proposed to use a gateway device, in order to assure the interoperability between the devices, this way ensuring the correct communication between protocols.

The second unit is the Data Storage Unit, which will store the data acquired by the Perception Layer. For this, it is recommended to use a server, the Cloud, or a hybrid solution using both. When using the Cloud, the Fog Computing can also be added, to avoid the network overload by processing some of the information locally, not being necessary to send all of it to the Cloud to be analyzed. By using the Cloud, the solution can make use of possibilities such as Big Data analysis, assuring data privacy and scalability, among other services. Besides, this unit allows other applications integrated in the Smart Campus model structure to access the data stored, which amplifies the application’s possibilities by making use of several other data that it would not have previously at its disposal, which can help the solution to offer better services.

The third unit is the Data Analysis Unit which is responsible for analyzing and processing the data stored in the Data Storage Unit. In this unit, data analysis techniques are applied, using both techniques of Artificial Intelligence (AI), which provides data analysis to the applications by using features such as machine learning, deep learning, ontologies, NLP, among others, besides using other data analysis techniques, which can extract information such as time series, data distribution, among others. This unit will be able to provide new information and data associations based on the techniques applied and the data that is stored in the Data Storage Unit, which will expand the application's possibilities by making both the analyzed and the gathered data available for other applications integrated in the model’s structure.

The fourth unit is the Control Unit, which offers fault tolerance via software and manages the access control to the Data Storage Unit, controlling the access to features like the devices configuration, data manipulation and data access, ensuring that only authorized users will have access to the data acquired and devices configuration.

By sharing the data storage, allied with the available AI techniques and their features, the applications will have a wide range of possibilities, gathering more data together than if dedicated, allowing for better data analysis and the creation of new data associations, which will provide enriched data for the applications integrated in the structure.

4.1.5 Application Layer

One of the proposed contributions of the model is to integrate accessibility aspects, which are integrated both in the Perception Layer, through the Environment Analysis process, and the Application Layer, by integrating aspects such as context awareness and adaptation, whose objective is to adapt the interface to the user's disability, adjusting it in order to facilitate the interface between the user and the system.

Besides, aside from the context adaptation, in order to integrate accessibility aspects the application should also offer personalization, and allow the user to choose between options, instead of applying a standard one. For example, an application which contains navigation and identifies that the user is on foot or using a bicycle, should offer different routes to the same destination, verifying other information to determine a route, such as the weather, with the intention of offering optional routes for the user to choose. Also, if the application uses the web, it should follow the Web Accessibility Guidelines, which specifies how to make the web content more accessible to users with disabilities.

4.2 MODEL DIFFERENTIALS

Based on the challenges and issues identified and detailed previously, the model aimed to propose a solution that offered flexibility, offering abstraction from an application point of view. Also, it proposes a solution that can adapt to new technologies without interfering with its basic structure, making the layers independent of each other, of easy management, organized and secure.

Also, the proposed architecture is able to be applied to different solutions, being capable of integrating existing solutions, which will also be able to share the data storage, expanding the application's possibilities in order to offer better services.

Another differential is the capability of integrating accessibility aspects, which is integrated in different layers (Perception Layer, Application Layer), offering a better coverage of the several accessibility issues, assuring that partial solutions (applications only) and complete solutions (complete structure) will both integrate accessibility aspects.

The model also seeks to promote the Smart Concepts requirements, encouraging practices that are sustainable, and making analysis of the impact generated by the solutions, improving the quality-of-life brought by it and improving the quality-of-service provided.

5 IMPLEMENTATION

This section will present the prototype developed in order to validate the proposed model structure, which was divided in two structures and four applications. The structure consists of the temperature prototype and the localization prototype. The applications consist of the localization support mobile application, as well as the integration application, a web API and the data analysis application, which were all developed based on the proposed model architecture.

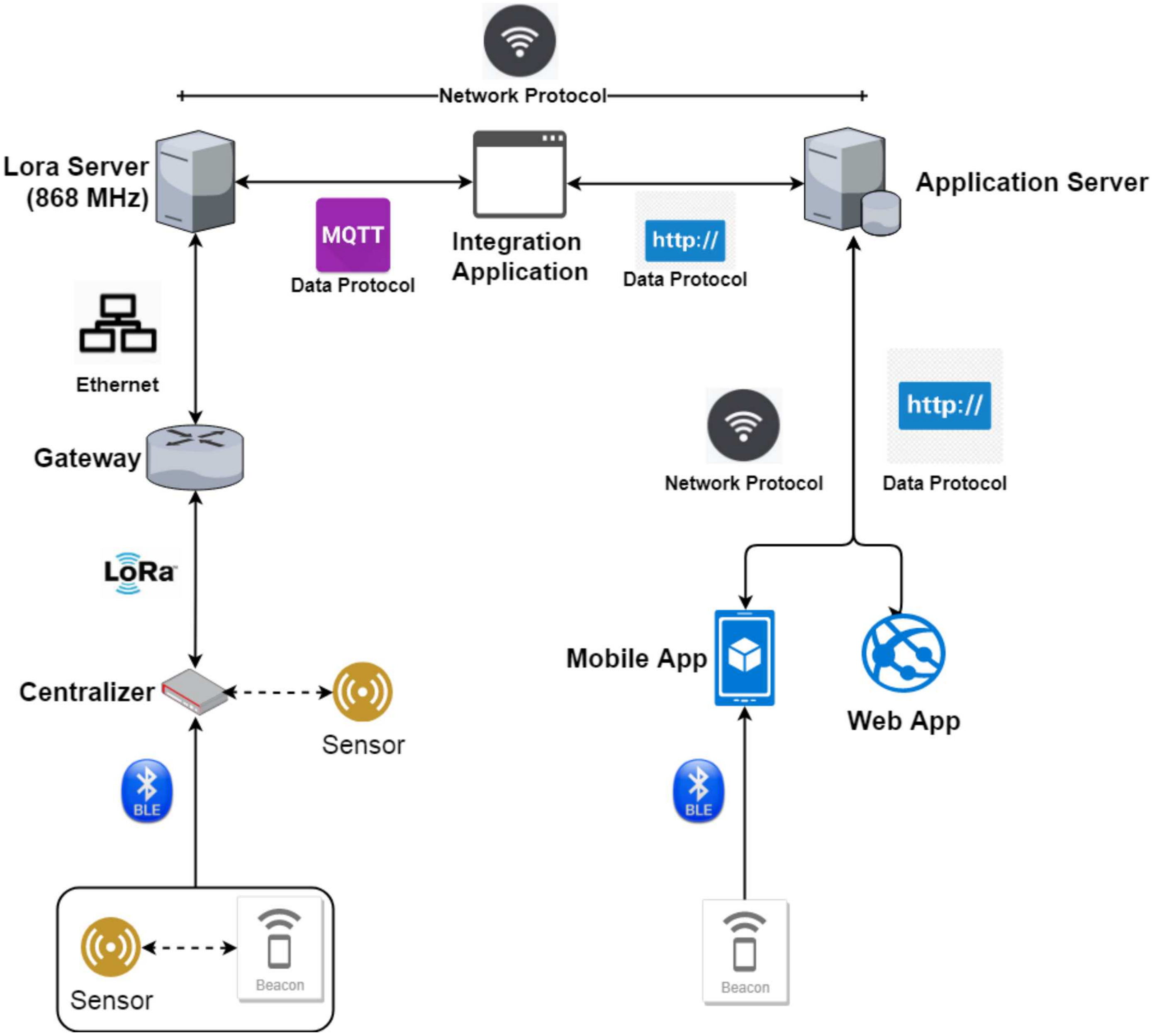
First, an overview of the system functionality will be presented. Secondly, the hardware infrastructure, which consists of a temperature measuring prototype, will be detailed. Thirdly, the localization support mobile prototype will be presented, as well as the mobile application developed and its functionality, besides demonstrating the screens and the web services implemented. Fourthly, the web API developed will be detailed. Fifthly, the integration application developed is presented, and finally, the data analysis application is presented.

5.1 OVERVIEW

The first structure developed was the temperature measuring prototype, which gathered the temperature data in different locations on campus and storage it in database through the use of a LoRa server. The second structure developed was the localization structure, which data was gathered by the mobile application. The structure consists of Beacon devices distributed in a determined route on campus, which signal is captured by the mobile application, which after processing the information, was able to help the user’s mobility, as well as gathering information about the route's traffic and user’s behavior.

The third structure consists of an integration application, which pulled the stored data from the LoRa server and stored it in the application server. Finally, the data analysis application gathered the stored data in the application server and analyzed it. The structure containing the temperature and localization structure, as well as the integration application is shown in Figure 12.

Figure 12 – Prototype structure overview.

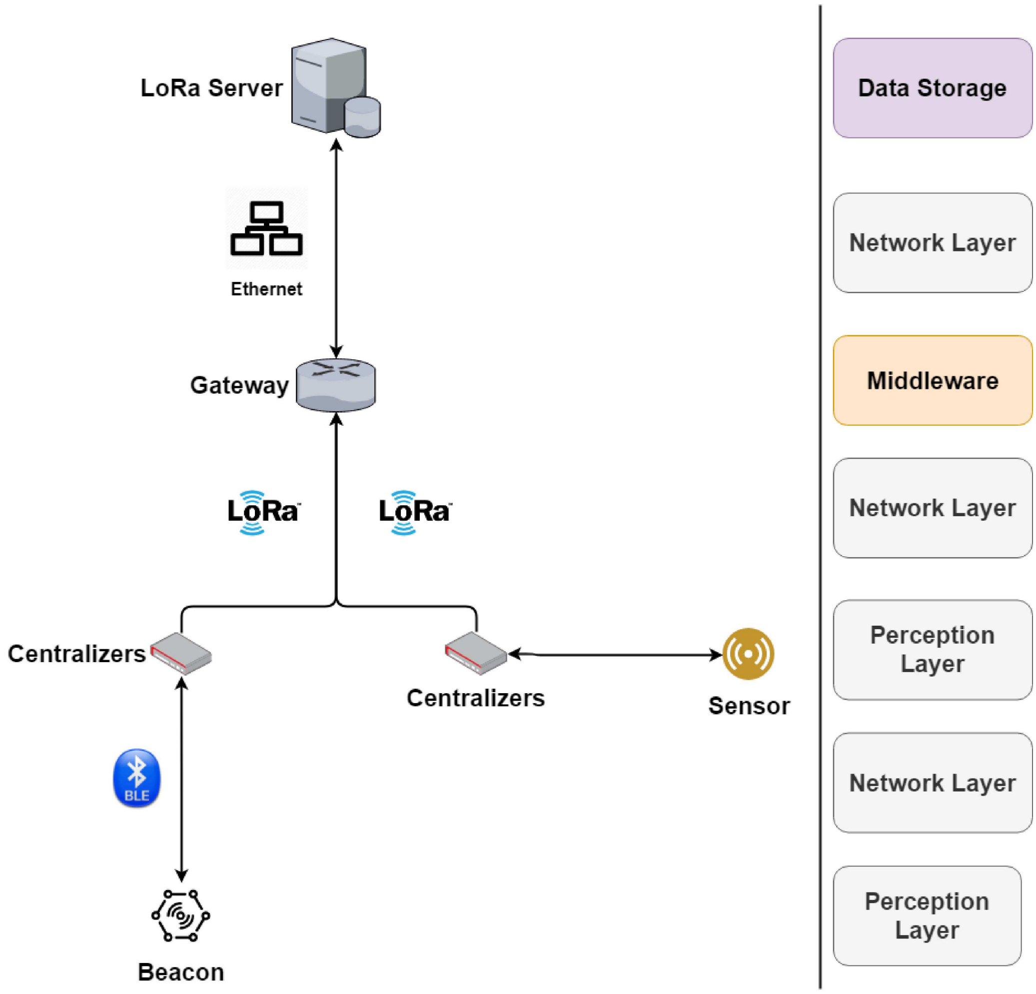


Source: Elaborated by the author.

5.2 TEMPERATURE INFRASTRUCTURE

This infrastructure measured the temperature in three different locations on campus using Temperature Beacon devices, and then sent the data via BLE protocol to the centralizers, which when configured via bridge mode, sent the data received via BLE to the Gateway via LoRa protocol. When the centralizer was configured in perception mode, it gathered the data from the sensor via serial protocol and sent it to the Gateway via LoRa protocol. The Gateway then sends via uplink mode and through the Ethernet protocol, the data receive to the LoRa Server, which implemented the LoRaWAN network protocol. The structure as well as the corresponding model layer is shown in Figure 13.

Figure 13 - Temperature measuring prototype structure.



Source: Elaborated by the author.

5.2.1 Hardware Components

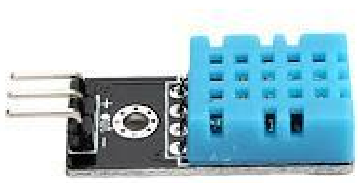
The next sections will detail each device implemented in the structure, as well as its functionality. The first device presented is the Temperature Beacon; the second presented is be the centralizer; the third presented is the Gateway and the lastly presented is the LoRa server.

5.2.1.1 Temperature Beacons

For the implementation of the Temperature Beacon, an ESP32 development board (Figure 15) was used. In order for it to receive the temperature data collected by the sensor, it was connected via serial protocol to a DHT11 (Figure 14) sensor, which is a humidity and temperature sensor, although for this research just the temperature data was used. The schematic for the assembling is displayed in the Figure 16, and Figure 17 shows the final assembling.

As for its configuration, the Beacon was configured in server mode (peripheral), and its service characteristics configured to enable the notify property, which sends to the central device (client) when connected, the data gathered in an interval of 1 second. In terms of energy supply, this assembling offers two options. The first option is by attaching a 3V3 (3.3 Volts) battery to the board, making it of mobile installation, but requiring of maintenance more often. The second option is through a fixed 5V (5 Volts) power supply, less mobile but which requires maintenance less often.

Figure 14 – DHT11 sensor.

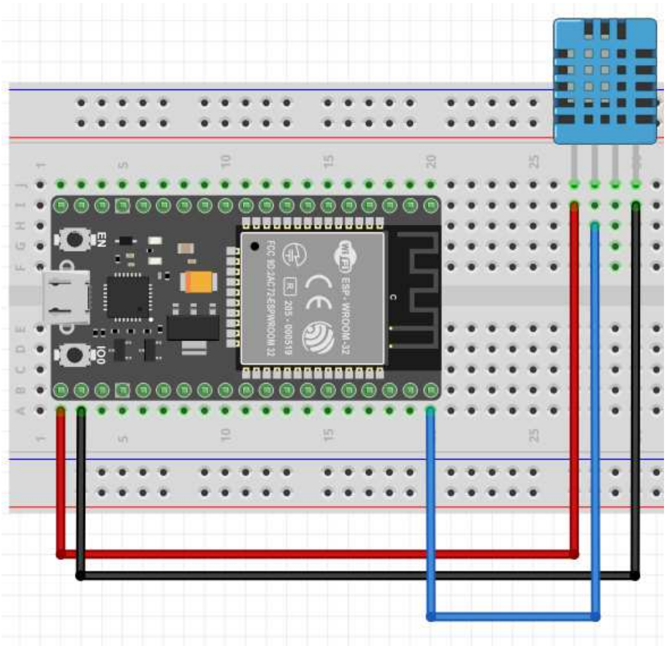


Source: https://1.bp.blogspot.com/-mOlz1vacrro/UZqxKuLNcVI/AAAAAAAAAek/CL9ZZ91hmkU/s1600/unibus-digital-humiture-sensor-module-dht11-for-arduino_grsstl1348482540285.jpg

Figure 15 – ESP32 development board.

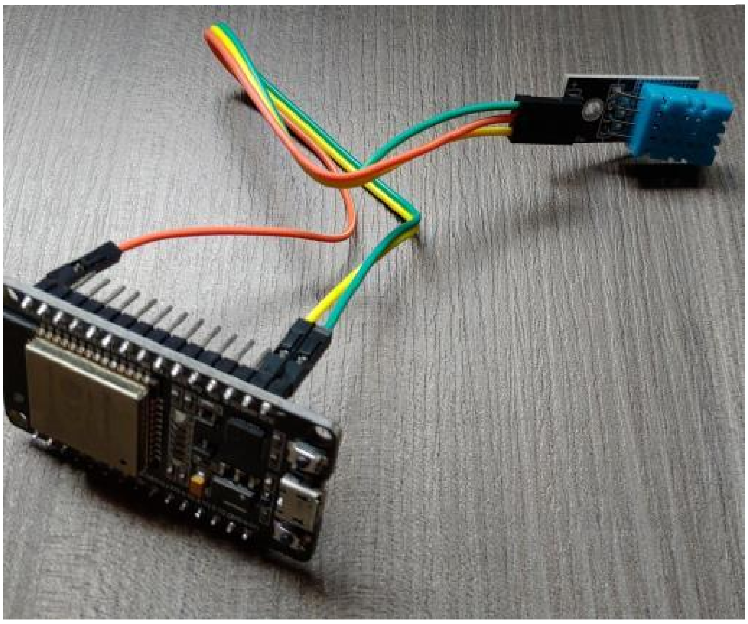
Source:https://www.espressif.com/sites/default/files/dev-board/ESP32-DevKitC-32D-F_1_1.png

Figure 16 – Beacon device - schematic assembling.



Source: Elaborated by the author.

Figure 17 – Beacon device – final assembling.



Source: Elaborated by the author.

The next section will explain how the centralizers implemented the Beacon in a different manner, in order to validate the flexibility of the model, using the same device in two different configurations.

5.2.1.2 Centralizers

For the implementation of the centralizer device, a RM186 (RM1xx Module, 2019) board from Laird was configured, programmed and implemented. It was chosen because of its feature and versatility, which combined both BLE and LoRa protocols in the same development board, displayed in Figure 18.

Figure 18 – Laird RM186 board.

Source: https://br.mouser.com/images/lairdtechnologies/lrg/RM186_SM_SPL.jpg

For this prototype, this module was used in two different applications, being the first in bridge mode, and the second as a perception device. If programmed to operate as a bridge, it will receive the temperature data collected by the Beacon device via BLE protocol and send it via LoRa protocol to the Gateway. If programmed as a perception device, it allows several sensors to be attached to it, collect data, and then sent it via LoRa protocol to the Gateway. Both applications were programmed to collect the information, being from the sensors (perception) or the Beacon (bridge), at a specific time interval, and then sent this to the Gateway.

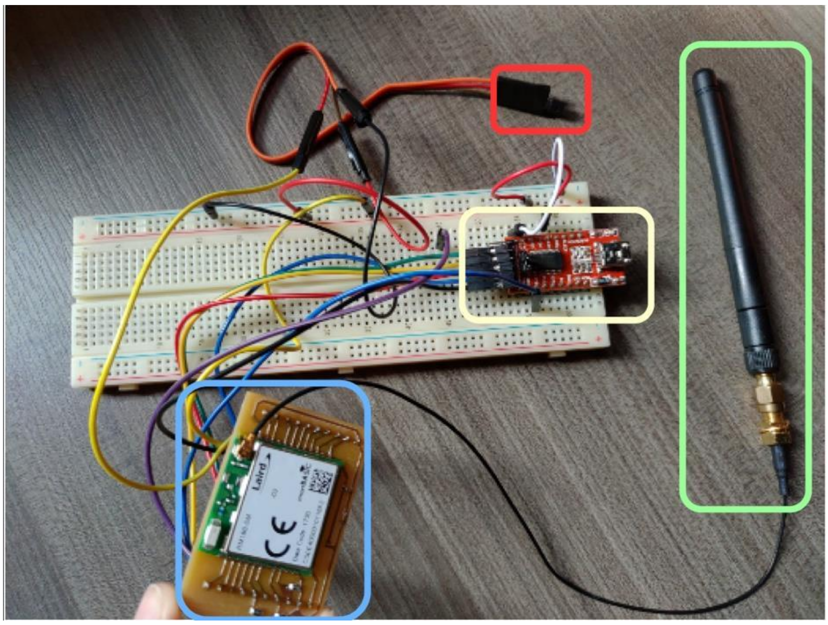
As shown in Figure 18, the module acquired was sold as a stand-alone, which required it to be weld in a board with ports, in order for it to be programmed, configured and used. The board assembling is shown in Figure 19.

Figure 20 shows the final assembly of the board, where the rectangle highlighted in red shows the LM35 temperature sensor, which is connected to the module and used when this is configured as a perception device, collecting the temperature data through the sensor. The rectangle highlighted in yellow represents the FTDI232 board, used to access the module in order to configure and program the scripts on it. The rectangle highlighted in green represents the external dipole antenna, used for the LoRa communication. Finally, the rectangle highlighted in blue represents the RM186 module.

Figure 19 – Laird RM186 module – board assembly.

Source: Elaborated by the author.

Figure 20 – RM186 board – prototype assembly.



Source: Elaborated by the author.

As aforementioned, this implementation together with the previous one, validated that the model is capable of supporting two different applications utilizing the same device.

5.2.1.3 Gateway

In order to implement the Gateway device, a Raspberry Pi 2 model B board was used in association with a LoRaWAN Gateway module RHF0M301, and a bridge board RHF4T002, which is a physical adapter used to connect the Gateway module to the Raspberry Pi. The Raspberry Pi board runs on the Raspbian Stretch Lite operational system. After the assembling, in order for the Gateway to work properly, two software components were installed. The first was the LoRaWAN software and the second, The Things Network (TTN) packet forwarder software, which forwards the received packages to the TTN server. Then the Raspberry Pi configuration file was installed, being used the EU-868 configuration file. This was chosen in order for it to receive the packages sent by the centralizer module, which also is configured for the EU-868 configuration.

5.2.1.4 LoRa Server

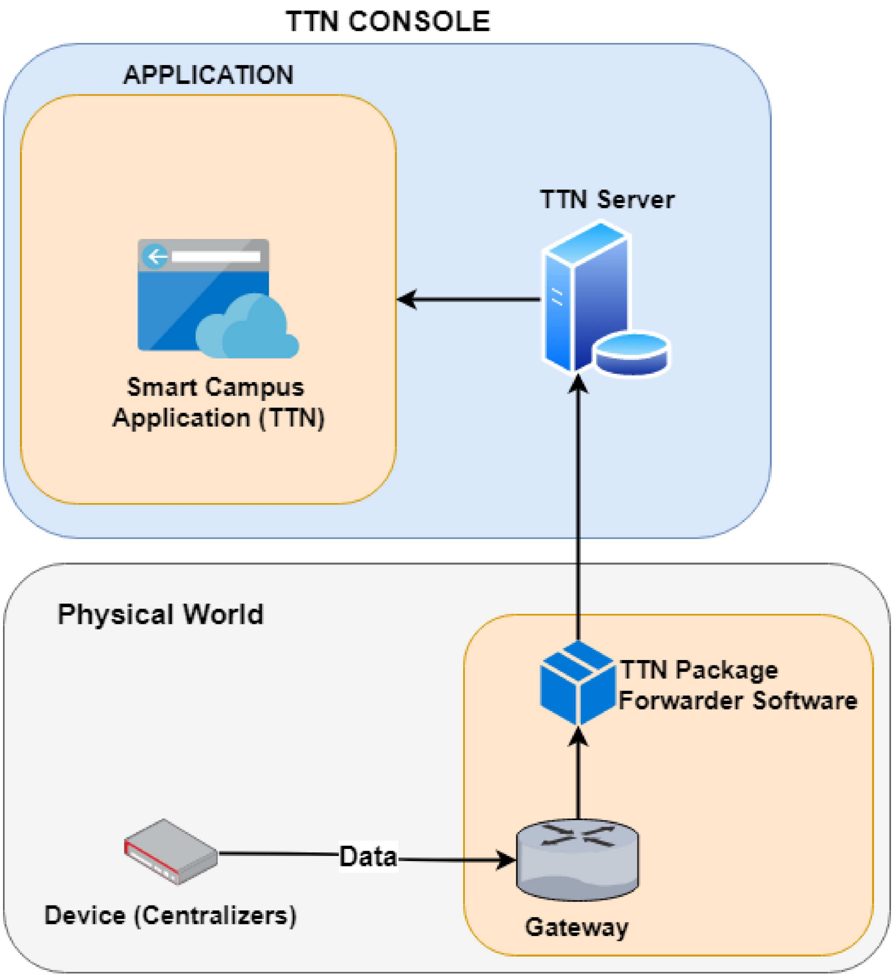
The LoRa server was implemented through the The Things Network architecture (TTN, 2019), which provides a LoRaWAN network server. In order to configure the server, firstly, the Gateway developed previously was registered in the main console, configured to act as UDP Packet Forwarder bridge, and registered the Gateway's EUI identifier.

After, a new application was created on the main console, which was identified as Smart Campus, and on its management console, a new device added, which is the centralizer, previously described. It was configured to operate via ABP (Activation by Personalization) activation mode, which does not require the join procedure to communicate with the application. Due to this, it was required that the centralizer be configured using the App Session Key, Network Session Key and Device Address designated by the TTN application.

As shown in Figure 21, the workflow of the data starts by the device, which sends a LoRa package to the Gateway, which will forward the received package to

The Things Network server. Section 5.5 will detail how this data is extracted from the TTN server and stored in the application server.

Figure 21 – TTN Console package workflow.



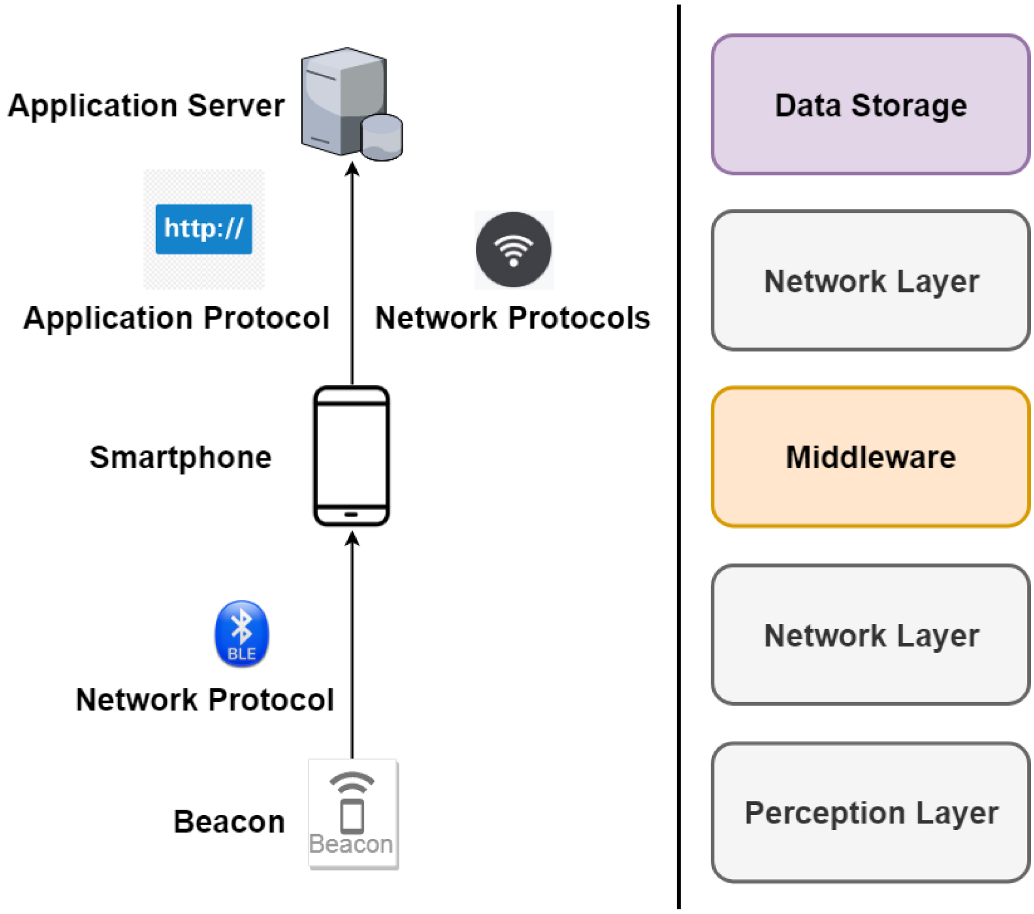
Source: Elaborated by the author.

5.3 LOCALIZATION INFRASTRUCTURE AND MOBILE APPLICATION

The mobile application developed has the purpose of guiding the user through a defined route inside the campus, enabling ten possible locations. Also, it registers the user's behavior on campus, storing the user's coordinates on a server, for future analysis on campus's usage. As for its functionality, after being installed on a smartphone, it receives the RSSI signal from at least three different Beacon devices, which are installed on the environment, using this signal's value to determine the user's location through the distances determined by the trilateration calculus. After, the user's coordinates are sent to the application server, using the HTTP protocol. For this, a web API is used, which is detailed on section 5.4.

The presented structure will work for both indoor as well as outdoor environments, being necessary to configure the attenuation constant in the trilateration calculus (Equation 2) in order for it to work properly. The application architecture as well as its corresponding model layer is displayed on Figure 22.

Figure 22 - Mobile application prototype overview.



Source: Elaborated by the author.

The application will use Beacon devices in order to determine the user's location, determining such through a trilateration calculus, as mentioned previously, which uses the received distance in relation to the three nearer devices to determine the smartphone location.

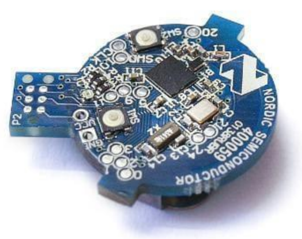
5.3.1 Hardware Components

The next sections will detail each device implemented in the structure, as well as its functionality. The first device presented is the Beacon; the second presented is be the smartphone and its requirements; the third presented is the application server.

5.3.1.1 Beacon

Although there are other localization devices that provide better accuracy, Beacon devices were chosen due to their functionality, low energy usage, low cost, and for providing other features that can be explored in the future, such as triggering an action by proximity, passing an URL by proximity, among others. They also are passive devices, meaning that they don't have access to the user's device; they will just trigger actions on it. However, as any other IoT (Internet-of-Things) device, it needs some measures to make the devices and its connections secure, such as adding encryption to the communication and requesting authentication to configure the devices. The Beacon devices chosen for this implementation were the Nordic Semiconductor nRF51822-Beacon, shown in Figure 23.

Figure 23 – Nordic Semiconductor nRF51822-Beacon



Source: https://br.mouser.com/images/nordicsemiconductor/lrg/nRF51822-Beacon.jpg

This device was chosen for presenting a plug-and-play solution, commercializing a smart kit with the device and a CR1632 coin-cell battery. The device consists of the hardware, firmware and a mobile configuration application, which allowed for minimal configuration, such as changing the UUID, minor and major values and choosing an action when a device connected to it, being this displaying a page, a figure, among others.

5.3.1.2 Smartphone

The project was developed as cross platform, which allowed for it to be deployed to both IOS and Android systems, although, for this implementation, the Android system was chosen, which required the smartphone used to present the following basic requirements: run on the Android Marshmallow (version 6.0) operational system; include connectivity options such as the Bluetooth 4.0 (BLE), Wi-Fi 802.11 b/g/n, 3G or 4G, and GPS.

Based on these requirements, the application was tested on three smartphones. The first was the ASUS Zenfone 5Z, which runs on the Android Pie (version 9.0) operational system and has as connectivity options the Bluetooth 4.2, Wi-Fi 802.11 b/g/n, 4G and GPS. The second was the Samsung Galaxy J5 Pro, which runs on the Android Oreo (version 8.0) operational system and has as connectivity options the Bluetooth 4.1, Wi-Fi 802.11 b/g/n, 4G and GPS. These were the ones used to test the full functionality of the application, although, in order to test the application compatibility with other Android systems, it was also tested on another smartphone, the Motorola Moto G4, which runs on the Android Nougat (version 7.0).

5.3.1.3 Application Server

For the application server, the PostgreSQL was chosen. It is an open source object-relational database system, and the SQL version 10 was used. For its management, the pgAdmin software, version 4.2, which is an open source administration and development platform for the PostgreSQL was chosen. It was configured as a localhost, installed on Windows 8.1 64bit operational system. The data tables stored in the database will be detailed on section 5.4.

5.3.2 Mobile Application Features

The next sections will present the mobile application requirements, configurations and functionalities.

5.3.2.1 Requirements and Configurations

The mobile application was developed through the Visual Studio 2017, version 15.9.11, which is the integrated development environment (IDE) from Microsoft. In order to develop the application, the Xamarin package, version 3.4.0.1008975 was used. It was developed using the cross-platform mode, which allows the same code base to deploy native code to different systems, such as IOS and Android, and lowers the costs for developing, besides facilitating the management of the code. The application was configured to run on the following system versions:

- System version base: Android Marshmallow (version 6.0).
- System version destiny: versions superior to Android Marshmallow (version 6.0).

Also, the following package dependencies were added in the project:

- MathNet.Symbolics, version 0.20.0: used for solving the trilateration calculus equations;
- Microsoft.AspNet.WebApi.Client, version 5.2.7: used for the communication with the web API developed;
- Microsoft.Net.Http, version 2.2.29: used for the HTTP protocol communication;
- Newtonsoft.Json, version 12.0.1: used for serializing and deserializing json objects;
- Plugin.BLE, version 1.3.0: used for the BLE scanning;
- Plugin.Permissions, version 3.0.0.12: used for the locations permission request;
- Xam.Plugin.Connectivity, version 3.2.0: used for the Wi-Fi protocol connection;
- Xam.Plugin.Geolocator, version 4.5.0.6: used for the determination of the users location using the GPS;

- Xamarin.Forms.GoogleMaps, version 3.0.4: used for the maps features.

5.3.2.2 Functionality

The mobile application starts by asking the user to log in, which requires the user's registration number, in this case the user's ID number in the University, and a password. In case the user is not registered yet, is necessary to register in the application. The application also allows for password changing, being necessary to enter the old password for security reasons.

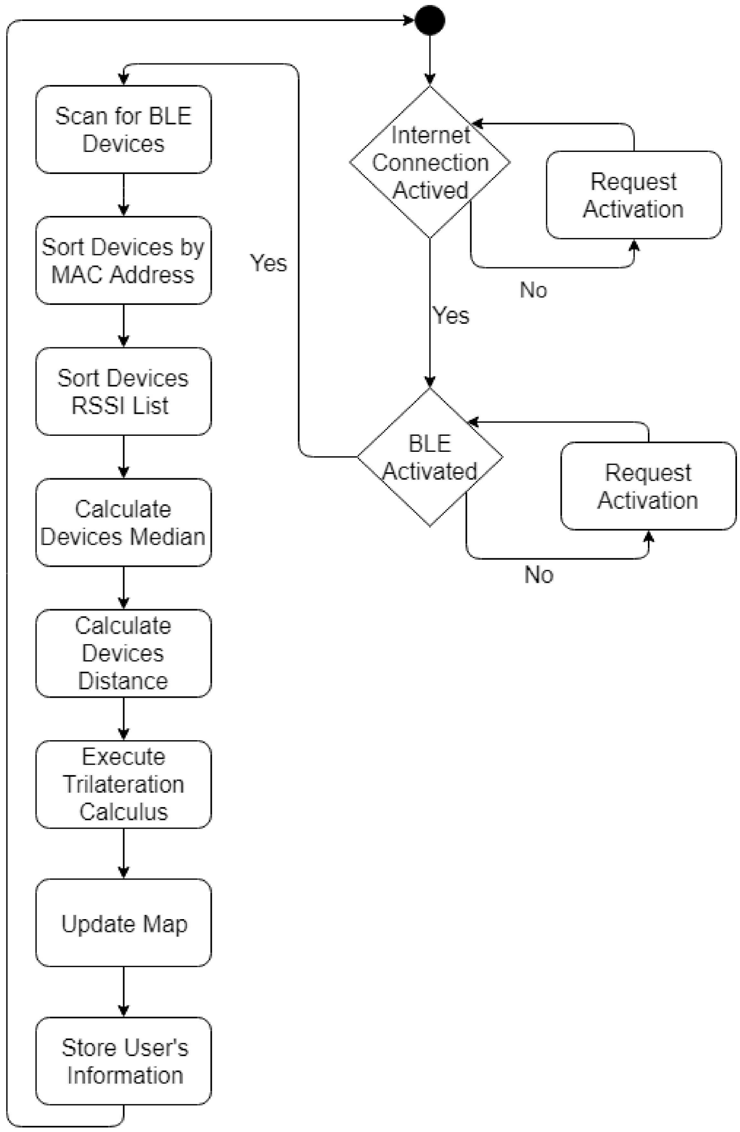
After the log in, the user must choose a route, choosing among ten available locations. The allowed locations are the Access 02 (*Acesso 02*), the buildings B01, B02, BO3, B04, B05, B06, B07, the community center (*A01*) and the library (*A02*). The locations as well as the complete route (orange circles) are shown in Figure 24.

Figure 24 – Mobile application complete route.

Source: Elaborated by the author.

Next, the application will load the chosen route, and will update the application's map every three seconds, updating the user's location on it. Also, after updating the map, the application will send to the application server the user's location on it, storing its behavior on campus. This will allow for future studies regarding campus usage in terms of resources and locations. The flow chart for the application's main algorithm is shown in Figure 25.

Figure 25 – Mobile application algorithm flow chart.



Source: Elaborated by the author.

The algorithm starts by checking the internet connection as well as the BLE connection, if any of them are not connected, the connection request is made. After, the algorithm will scan for BLE devices (Beacons) for one second, storing the received advertising packet's RSSI value in a list. Then, this list's devices will be validated, only selecting the advertising packets from devices that contain their MAC address registered in the system. After, the devices advertising packets list will be sorted by the device MAC address, where each device will have a list containing their RSSI values. Next, each device's list of RSSI values will be sorted. Then, this list is used for the median value determination, this way minimizing the error in the localization estimation of the device. After this, the distance calculation using Equation 1 is made in each device on the list. Then, the three nearest devices on the list are selected for the trilateration calculus (Equation 3), which will determine the user’s position in relation to each of these devices.

After the determination of the user’s location, the map displayed is updated, showing the chosen route, the starting and ending point, and user’s location in the map. Finally, the user’s position is stored in the application server. This process occurs every three seconds, being one second reserved for the scanning and two seconds reserved for the data processing and communication with the server.

5.3.3 Mobile Application Screens

This section will present the mobile application screens, and its functionalities.

5.3.3.1 Login Page

In this page is made the user log in the system. It consists of two data entries and three buttons as shown in Figure 26. The data entries are for the registration number and the password. The buttons are responsible for the login data validation (*Login*), the registration in the application (*Cadastrar*) and for password changing (*Trocar senha*). Also, if the internet connection is not enabled a message indicating it is displayed, which is then requested its activation.

Figure 26 – Login Page layout.

Source: Elaborated by the author.

If the user tries to log in with invalid information (empty fields; wrong password length), a message of error is displayed, indicating for the user that the log information is invalid. If the log information is correct, a message indicating it is displayed and the user is then redirected to the Set Route Page.

5.3.3.2 Registration Page

In this page is made the user registration in the system. It consists of three data entries and two buttons as shown in Figure 27. The data entries are for the registration number, the password and for the confirmation of the password. The buttons are responsible for the registration validation (*Cadastrar*), and for cancelling the registration (*Cancelar*), which when selected, will return the application to the Login page. Similar to the situations described for the Login page, if the user entered invalid data in the entries, alert messages are fired.

Figure 28 – Valid data entries example.

Figure 27 – Registration Page layout.

Source: Elaborated by the author.

If the user entered valid information but the passwords don't match, an alert message is fired. Figure 28 displays an example of valid data entries. If the user registration is successful an alert message is generated, informing the user of the successful registration, and it is then redirected to the Set Route Page.

5.3.3.3 Change Password Page

This page makes the users password changing in the system. It consists of four data entries and two buttons as shown in Figure 29. The data entries are for the registration number, the password, the confirmation of the password and for the old password. The buttons are responsible for saving the password change (*Salvar*), and for cancelling the password change (*Cancelar*), which when selected, will return the application to the Login page.

Similar to the situations described previously in the other pages, if the user enters invalid data in the entries, alert messages will be fired.

Figure 29 – Change password page layout.

Source: Elaborated by the author.

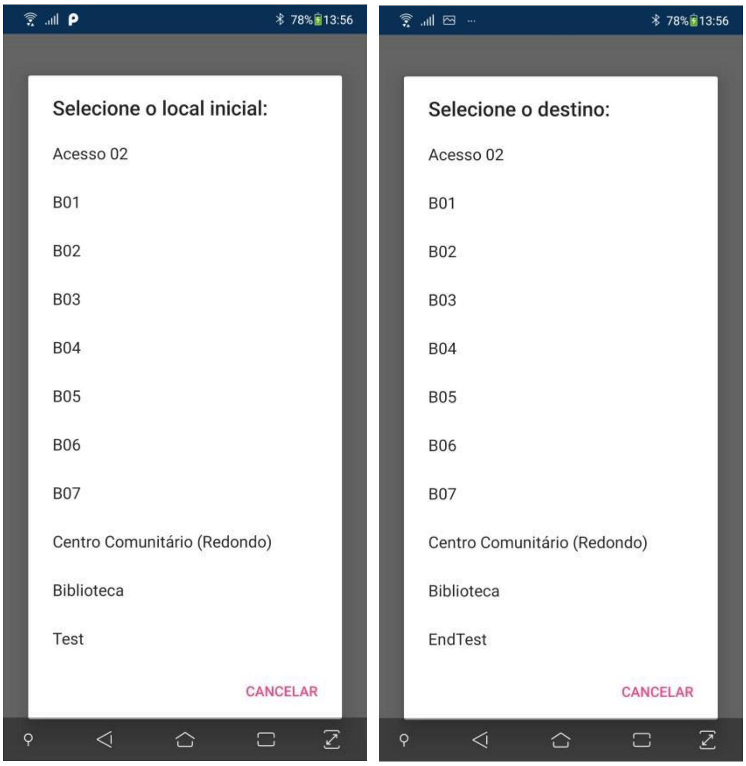
5.3.3.4 Set Route Page

In this page will be setup the user route. It consists of two pickers and a button, as shown in Figure 31. The pickers will select the locations and the button (*Traçar Rota*) will save these and send this information to the map page, in order for it to process the route and draw the map.

Figure 30 demonstrates the enabled locations for selection, and Figure 32 demonstrates two examples of route configurations, which will be demonstrated on the map in the next section.

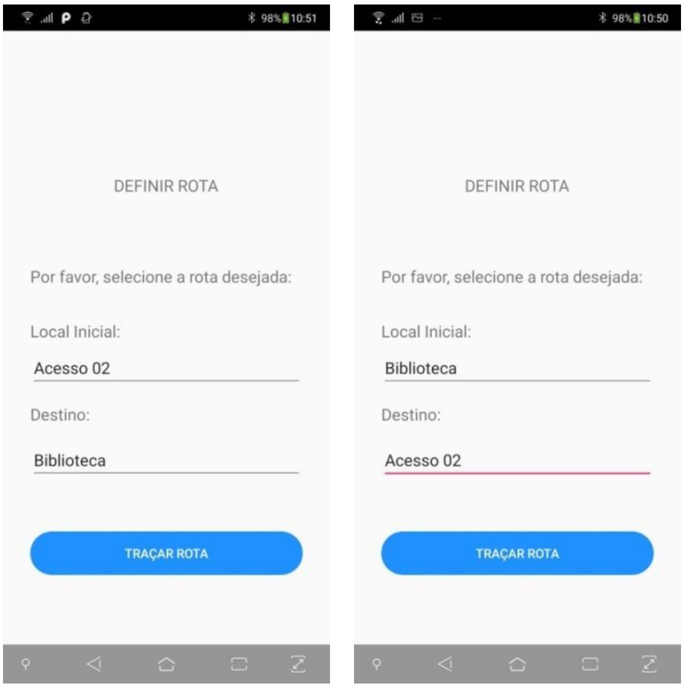
Figure 31 – Set Route page layout.

Figure 30 – Locations pickers - setup examples.



Source: Elaborated by the author.

Figure 32 - Locations selection examples – a) setup 1; b) setup 2.



Source: Elaborated by the author.

5.3.3.5 Map Page

In this page will be displayed the map and its resources. It has one button (*Nova Rota* – bottom left corner of the screen), which will enable the user to input a new route if desired. Among its resources, will be displayed the route between the two chosen locations and three markers. The purple one highlights the initial location; the red one highlights the end location; the green one highlights the user location.

In case the Bluetooth is not connected, a message is displayed as shown in Figure 33, requesting the activation. If it's connected, and the route information inputted is valid, the loading map message (Figure 34) is displayed, while the application process the information and load the map with it.

Figure 33 – BLE activation – message alert.

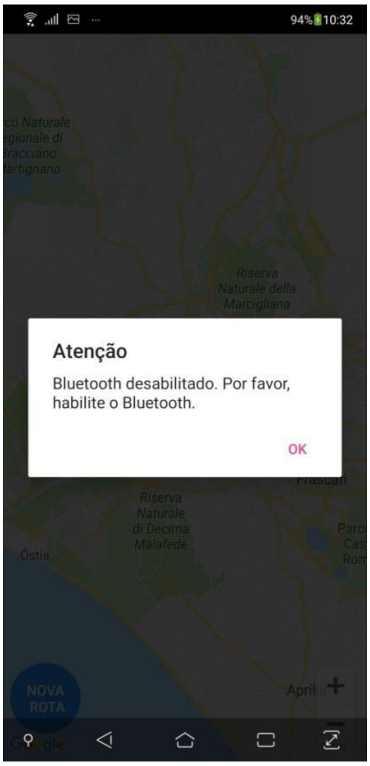
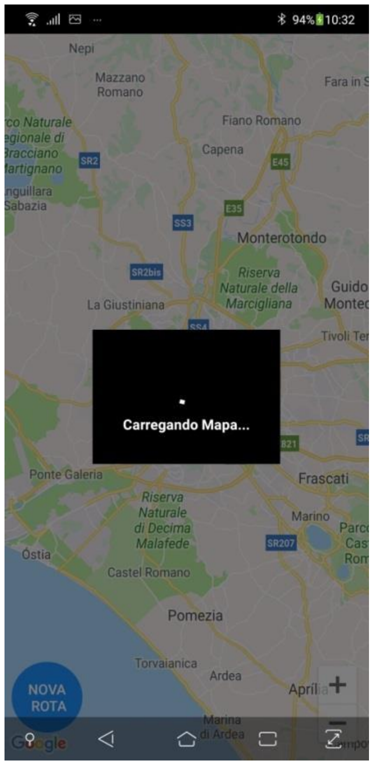


Figure 34 – Map loading screen.



Source: Elaborated by the author.

Figure 35 and Figure 36 displays the map configurations generated for the configurations made in the Set Route page examples, both displayed in Figure 32.

Figure 35 - Map page – setup 2.

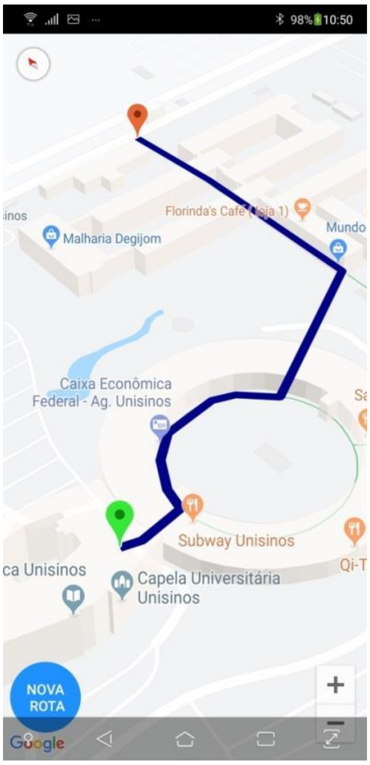


Figure 36 – Map page – setup 1.

Source: Elaborated by the author.

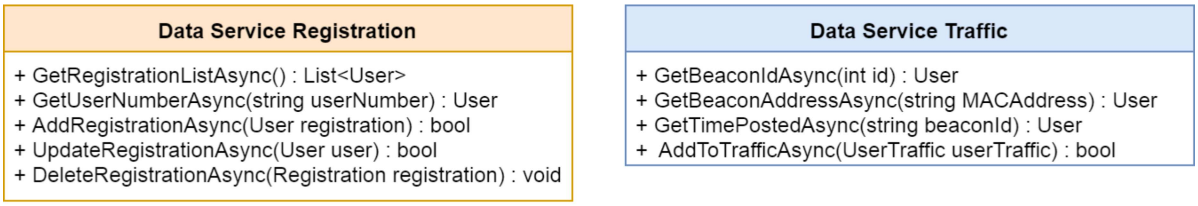
5.3.4 Web Services

For the web services, two classes were implemented, displayed in Figure 37. The first was for the user registration, which validates the data inserted and manage the registrations. The second was for the traffic control, where it manages the user location information that is sent to the application server.

The Data Service Registration class implemented the four HTTP methods (GET, POST, PUT and DELETE), while the Data Service Traffic class implemented only two HTTP methods (GET and POST).

In order for the web services to work properly and because the server was hosted as a localhost on the same computer as the application is running, an extension package was installed on Visual Studio 2017, the Conveyor by Keyoti, which allows that other machines and devices access the web applications hosted on the computer.

Figure 37 – Web servisse classes.



Source: Elaborated by the author.

5.3.5 Mobile Application - Accessibility Feature

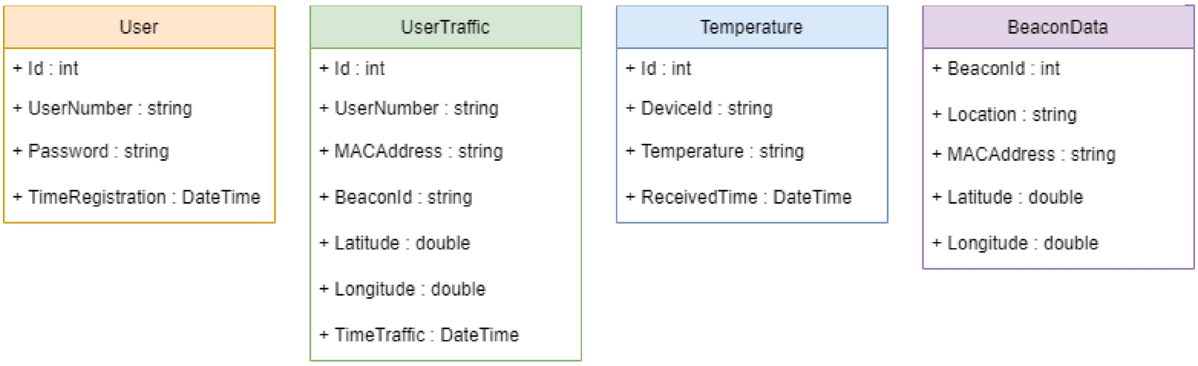
In order to assist the users in their travel to the routes and also help people with disabilities, a voice assistant feature can be added to the mobile application, using NLP (Natural Language Processing) techniques, being able to both receive commands given by the users, as well as give instructions or ask questions to the application, as demonstrated by Ghidini *et al*. (2016), Debatin *et al*. (2017) and Chen *et al*. (2018), which papers proposed solutions that included the voice assistance applied in a mobile application. With this feature, the user could be guided through the campus routes, as well as receive notifications about the campus accessible features.

5.4 WEB API

The web API was developed using the Microsoft Visual Studio 2017 IDE, version 15.9.11, and it updated and managed the PostgreSQL database via Npgsql Entity Framework Core PostgreSQL package, version 2.1.0, which allows that the changes made on the model, be propagated to the database. The API has four controllers, one for each table on the database: Temperature, BeaconData, User and UserTraffic.

Temperature will store the data acquired through the integration with the LoRa server, storing the temperature collected by the devices. BeaconData will store information about the devices installed on campus, like the MAC address, location, among others. User will store the user's information, like their registration number and password. UserTraffic will store the user traffic information, like the locations that he went through on campus. The database tables are shown in Figure 38.

Figure 38 – Database tables.

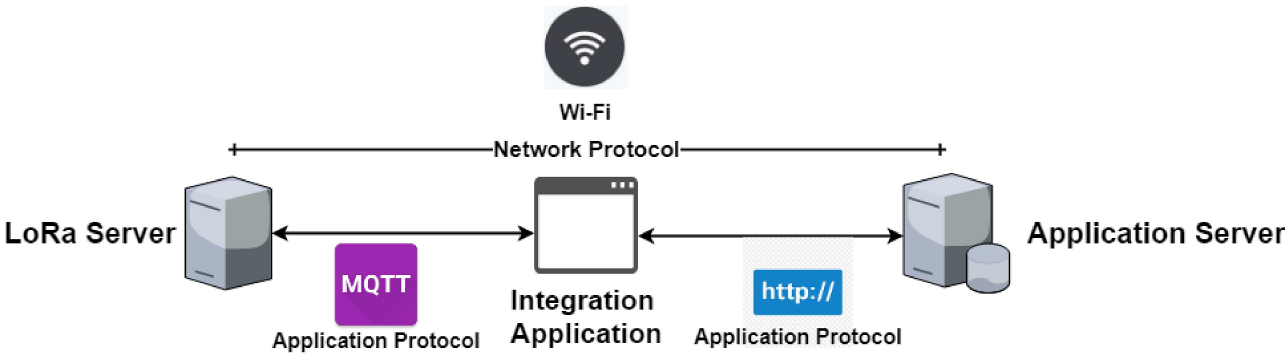


Source: Elaborated by the author.

5.5 INTEGRATION APPLICATION

This application is responsible for extracting the data stored in the LoRa server and storing it in the application server. In order to do this, the MQTT and the HTTP protocols were chosen. This application was developed using the Microsoft Visual Studio 2017 IDE, version 15.9.11. The Figure 39 shows the integration application architecture.

Figure 39 – Integration application overview.



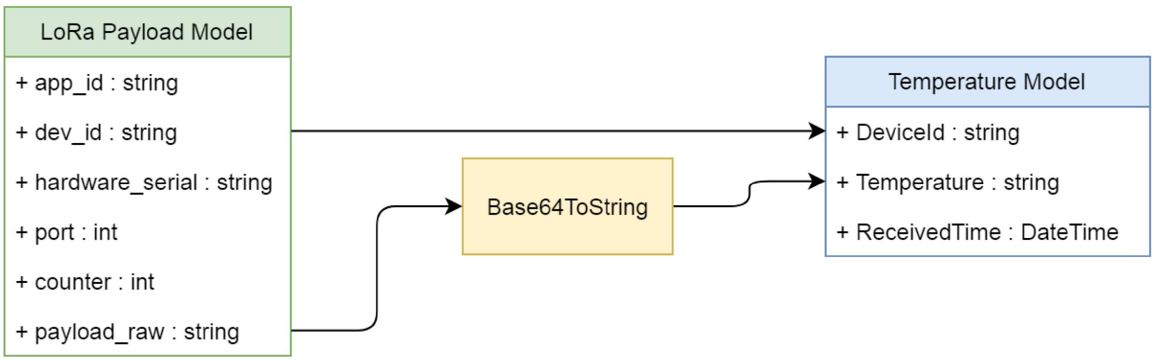
Source: Elaborated by the author.

The application starts by connecting with to The Things Network server through the MQTT protocol. For this, the TTN Application ID and the Application Access Key are necessary. After the connection, the application subscribes the topic in the TTN server that stores the application data. For this project the topic subscribed was this:

smart_campus/devices/centralizador/up.

In this example, *Smart_campus* is the application; *centralizador (centralizer)* is the device which is collecting the information; *up* is the uplink to server. In the subscribe method, an MQTT notify method is called, which will notify the application each time a new package is received by the TTN server. After this, the package message payload is converted to the data model, as shown in Figure 40, and the application then, sends this data via the HTTP POST method to the web API, which stores the received information in the database, on the Temperature table, displayed in Figure 38.

Figure 40 – LoRa payload conversion to data model.



Source: Elaborated by the author.

5.6 DATA ANALYSIS

This unit is responsible for analyzing the data stored in the Data Storage Unit, using both data analysis techniques and Artificial Intelligence techniques in order to analyze the data. For this prototype, two AI techniques were implemented, aiming to classify the data in groups (clusters), which will serve as the basis for other analysis.

Firstly, the methodology used to simulate the data will be presented. Secondly, the application that implemented the techniques used to analyze the data will be explained. Thirdly, the data structure gathered from other existing applications will be presented.

5.6.1 Data Simulation

In order to test the application that analyzed the data, a simulated dataset was created, using the data structure previously presented on Figure 38. As previously presented, the prototype implemented is divided in two structures and three applications; the first structure gathers localization data; the second structure gathers temperature data; one of the applications, the integration application, extracts the temperature data stored on the LoRa server and stores it on the application data server.

For the localization data, three parameters were generated: the location points (latitude, longitude) and the time in which the data was collected. It was stablished that the speed of a person on foot was 15 meters/minute. After, it was stablished the distance of each route, and based on both parameters the time-to-travel the routes were calculated. Next, using the times calculated previously, and using the same updating interval of the mobile application, which was of 2 seconds, the number of location points was calculated, by dividing the estimated time-to-travel the route by the interval time. Also, in order to generate a more realistic dataset, the TimeTraffic parameter, which stores the time in which the data was collected, was randomly generated, between the interval of 7:00 and 22:59. After establishing the parameters and intervals, 30 routes were selected randomly, and the location points for each generated. Figure 41 presents an example of the generated dataset for the location data.

Figure 41 – Example of generated dataset for the location data.

	User Number	Initial Local	Final Local	Date	Latitude	Longitude
0	1668	Biblioteca	B02	09/19/2019, 15:02:12	-29.792754972387648	-51.15438405136586
1	1668	Biblioteca	B02	09/19/2019, 15:02:14	-29.792784020910148	-51.15439989673288
2	1668	Biblioteca	B02	09/19/2019, 15:02:16	-29.792805874060885	-51.154462715917774
3	1668	Biblioteca	B02	09/19/2019, 15:02:18	-29.792713046488398	-51.154344055364305
4	1668	Biblioteca	B02	09/19/2019, 15:02:20	-29.792865938835845	-51.15443042065415
5	1668	Biblioteca	B02	09/19/2019, 15:02:22	-29.792747368282903	-51.15444077866837
6	1668	Biblioteca	B02	09/19/2019, 15:02:24	-29.792824004059003	-51.15432534946437
7	1668	Biblioteca	B02	09/19/2019, 15:02:26	-29.792770764494048	-51.15434855248134
8	1668	Biblioteca	B02	09/19/2019, 15:02:28	-29.792762857402046	-51.1544414925554
9	1668	Biblioteca	B02	09/19/2019, 15:02:30	-29.792797605072508	-51.154436346827914
10	1668	Biblioteca	B02	09/19/2019, 15:02:32	-29.792727037524347	-51.154487269464624

Source: Elaborated by the author.

For the temperature data, two parameters were generated: the time in which the data was collected and the temperature. The temperature data was randomly selected based on an interval, which was stablished following the temperature reference displayed on Climatempo (2019). The time parameter was generated randomly, not being stablished an interval. Figure 42 presents an example of the generated dataset for the temperature data.

Figure 42 - Example of generated dataset for the temperature data.

	Device Name	Device Location	Device Latitude	Device Longitude	Temperature	Date
0	centalizador_2	B07	-29,793612	-51,154242	25	11/03/2019, 09:46:12
1	centalizador	C02	-29,792934	-51,152332	24	11/12/2019, 15:32:23
2	centalizador_3	ittChip	-29,792712	-51,149865	30	11/01/2019, 12:57:47
3	centalizador	C02	-29,792934	-51,152332	28	10/19/2019, 09:10:38
4	centalizador_3	ittChip	-29,792712	-51,149865	30	10/09/2019, 19:10:55
5	centalizador_3	ittChip	-29,792712	-51,149865	22	10/12/2019, 19:31:44
6	centalizador_2	B07	-29,793612	-51,154242	22	11/19/2019, 11:39:50
7	centalizador_3	ittChip	-29,792712	-51,149865	23	11/15/2019, 08:07:44
8	centalizador	C02	-29,792934	-51,152332	29	11/18/2019, 07:55:26
9	centalizador	C02	-29,792934	-51,152332	27	11/07/2019, 09:03:52
10	centalizador_2	B07	-29,793612	-51,154242	17	10/07/2019, 09:41:43

Source: Elaborated by the author.

Both datasets will be used in the data analysis unit, which will present the results on the simulated data in section 6.4.

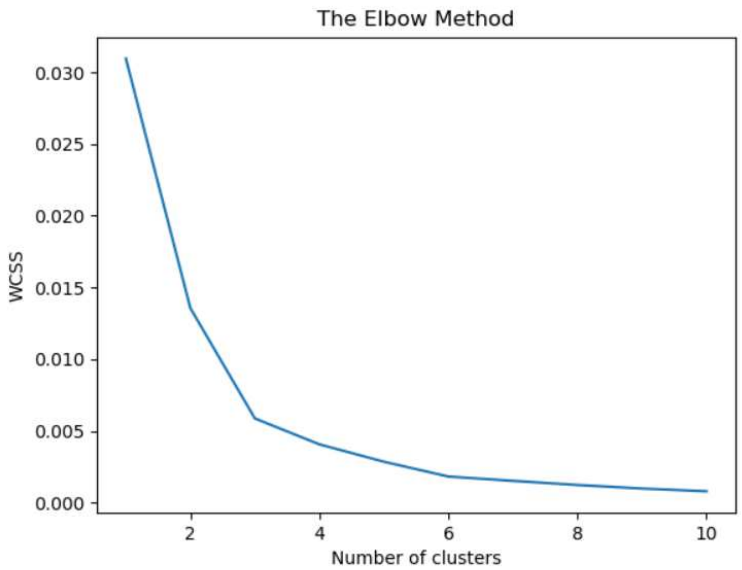
5.6.2 Data Analysis Application

In order to analyze the data simulated previously, a data analysis application was developed. It was implemented using the Python language, and for the localization data, two techniques of clustering were applied, the K-Means and the HDBSCAN, in order to classify the localization data. For the K-Means algorithm, the scikit-learn library was used and for the HDBSCAN algorithm, the hdbscan library was used. After classifying the clusters, a time series analysis was applied to one of the classified clusters, as well as demonstrated its frequency. For the temperature data, a time series was also applied to the data, as well as the frequency.

5.6.2.1 Localization Data Analysis

The application started by defining the best number of clusters for the data collected. To determined it, it was applied the Elbow Method, which helps to determine the best number of clusters for the dataset in analysis. It was necessary due to the K-Means algorithm functionality, which specifies that the number of clusters must be determined previously to its processing. Figure 43 presents the resulting Elbow Method.

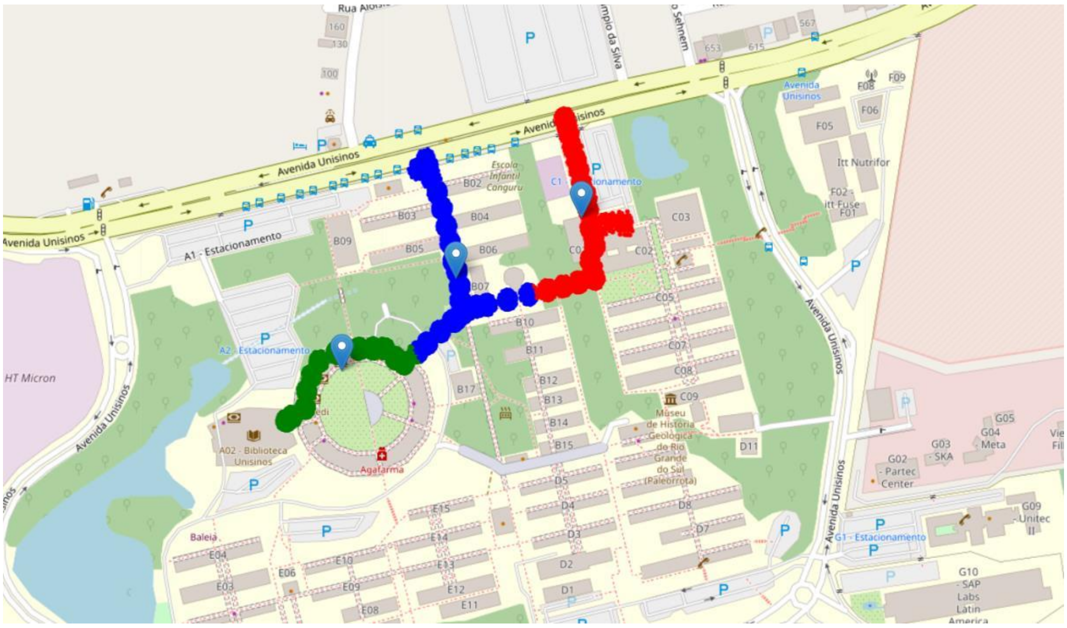
Figure 43 – Elbow Method result.



Source: Elaborated by the author.

As demonstrated by the Elbow Method, the best number of clusters for the dataset simulated was 3. After, the K-Means algorithm was executed and the map (Figure 44) with the clusters classification was generated. As demonstrated on Figure 44, it classified the three clusters and defined the centroid point for each.

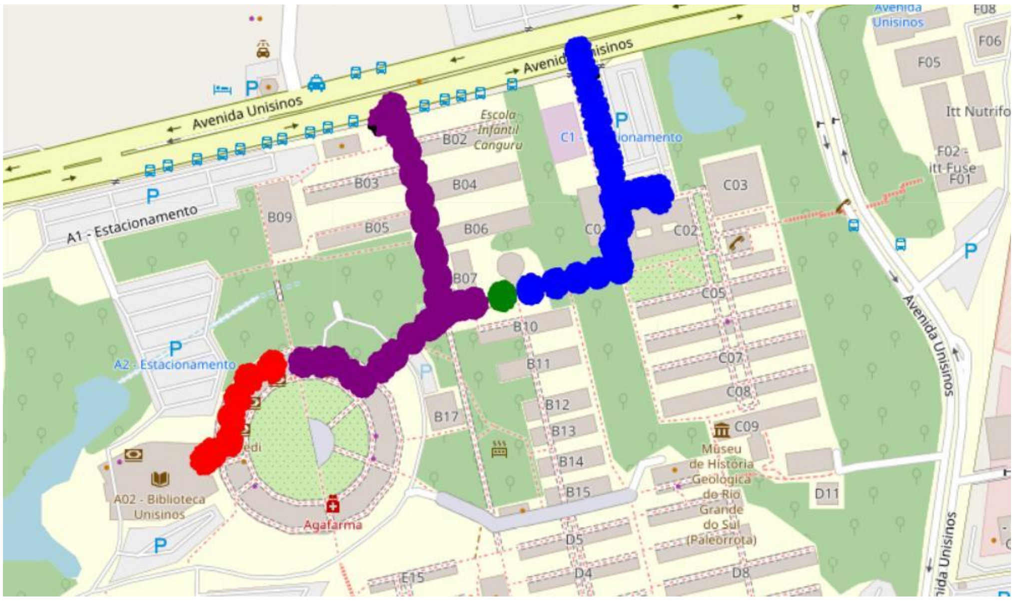
Figure 44 – K-Means classification map.



Source: Elaborated by the author.

Next, the HDBSCAN algorithm was executed, which does not required that a number of cluster be specified previously to its processing due to its functionality, which algorithm automatically specifies the best number of clusters for the analyzed dataset. Figure 45 presents the resulting map for the classification.

Figure 45 – HDBSCAN classification map.



Source: Elaborated by the author.

The analysis of the results of both classifications will be discussed in section 6.4, which will also present the time series for one of the classified clusters as well as its frequency.

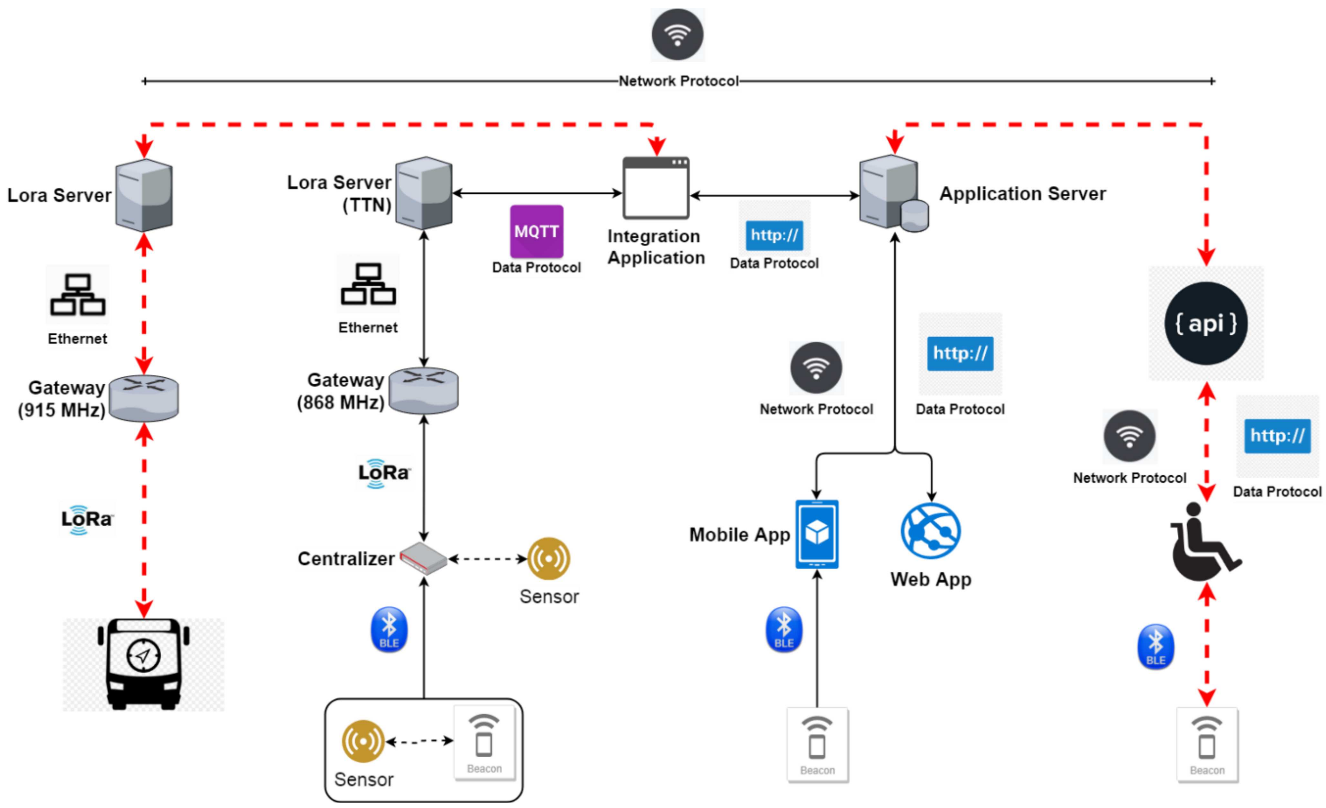
5.6.2.2 Temperature Data Analysis

For the temperature data, a time series analysis was made for each device, analyzing the distribution of the temperature and its frequency. The results of this analysis will be presented in section 6.4.2.

5.6.3 Integrated Projects Data

In order to validate the flexibility of the proposed model, verifying if it's able to incorporate existing solutions into its infrastructure, some projects were incorporated in the prototype structure, which utilized the structure and the devices, for different applications.

Figure 46 – Prototype structure – Existing projects integration



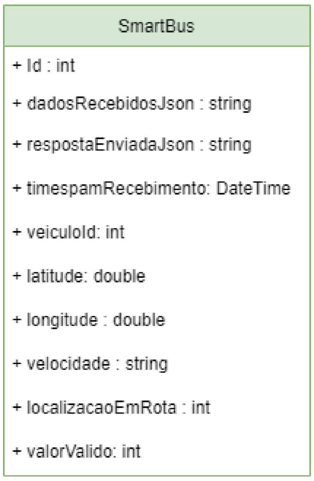
Source: Elaborated by the author.

5.6.3.1 Smart Bus

The Smart Bus application (highlighted on the left of Figure 46) used a similar structure developed for the temperature prototype to gather the localization data of a bus, verifying its route in order to determine if it was following the programmed schedule or if it was late. In order to determine the localization of the bus, it was installed a device on the bus, which sent the location data information to a LoRa gateway, which then sent the data package to the LoRa server, where it was stored.

To incorporate the existing application, it is necessary to fetch the localization data stored in the LoRa server, using the Integration Application of the developed structure, and stored it in the application server, which will then enable that other projects have access to the information. Figure 47 present the data structure used for the project.

Figure 47 – Smart Bus Application – Data structure.



Source: Elaborated by the author.

5.6.3.2 MobiLab Projects

The MobiLab (Laboratório de Computação Móvel - UNISINOS) has several projects that focus on assistive technologies, which are making use of the Beacon devices, used for the localization prototype, in order to determine a location. Among these projects is the Hefestos (Tavares *et. al*, 2016), which is an intelligent system which aims to help people with disabilities and elderly to manage accessibility resources. The application will use Beacons to determine a location, using the

Beacon as marking point, signalizing the room in which the user is present. In order to incorporate this project into the structure, it is necessary to gather the localization data, doing this by sending the data to the existing API, which will receive the data and stored it in the application server (highlighted on the right of Figure 46).

Besides the Hefestos, the MobiLab also has other projects that will also use the Beacons in order to determine a localization, which are also applying the same localization technique used for this research, the trilateration.

6 EXPERIMENTAL RESULTS

This section will present the experimental results obtained through experiments conducted on the structures presented in section 5. Firstly will be presented the experiments conducted on the localization structure, using the mobile application, also presenting an experiment conducted previously which measured the distance accuracy from the Beacon devices. Secondly will be presented the experiments conducted on the temperature structure, demonstrating its functionality through the screens obtained and detailing the tests. Thirdly will be presented the experiments conducted on the integration application, demonstrating the results obtained as well as the screens. Fourthly will be presented the results of the data analysis applied to the simulated dataset described previously, on section 5.6. At the end of the section, a general analysis of the results is presented, dedicated to highlighting the model contributions, as it can be seen in the experiments conducted.

6.1 LOCALIZATION STRUCTURE AND MOBILE APPLICATION EXPERIMENTS

For the mobile application, two experiments verifying the accuracy of the localization were conducted. The first measured the distance accuracy. The second tested the trilateration technique accuracy. The first following section will present the distance accuracy tests, and the second section, the trilateration technique accuracy tests.

6.1.1 Distance Accuracy Tests

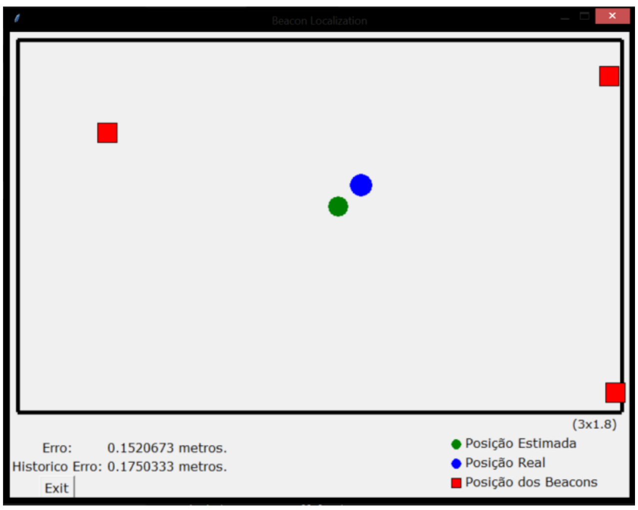
The first test conducted for the mobile application started by developing a Graphical User Interface (GUI) application which demonstrated the distance accuracy and the error from the central device in relation to the Beacon devices. In this prototype, the Arduino IDE was used to program all the devices, being used for the Beacon devices, three HM-10 (CC4125 chipset) devices, displayed in Figure 48, and for the central device, the ESP32 development board (Figure 15).

Figure 48 – HM-10 (CC4125 chipset) device.

Source:https://cdn.awsli.com.br/600x700/101/101615/produto/34081693/648a3b9e16.jpg

As mentioned previously, this prototype showed via GUI, the location of the Beacons, as well as the expected central device position and the real position, which resulted from the trilateration calculus, besides presenting the error in distance. In order to get the data needed for the calculations, the ESP32 board sent via serial communication protocol, the data collected (RSSI value) from the Beacon devices to a Python script, which calculated the distances, executed the trilateration calculus and update the GUI, showing the positions as well as the error obtained. An example is shown Figure 49, where the red squares represented the Beacon devices position in the environment, the green circle represents the central device expected position in relation to the devices and the blue circle represents the central device real position.

Figure 49 – Beacon distance accuracy experiment – GUI prototype.



Source: Elaborated by the author.

To develop this prototype, the software used was the Visual Studio Code version 1.24.1, architecture x64, configured with the Python 3.6 extension. This prototype results respected the expected error range in distance, which varies from 1 to 1.5 meters. All results presented errors of less than one meter.

Based on this, a mobile application was developed, using the same requirements and configurations as presented in section 5.3, now using the ESP32 board configured as a Beacon, as previously detailed in the section 5.2.1.1. This measured the distance accuracy using the smartphone as a central device, where Figure 50 demonstrates the device's RSSI value log screen and Figure 51 demonstrates the screen for the device distance, which was obtained through the Equation 1.

Figure 51 – Device distance screen.

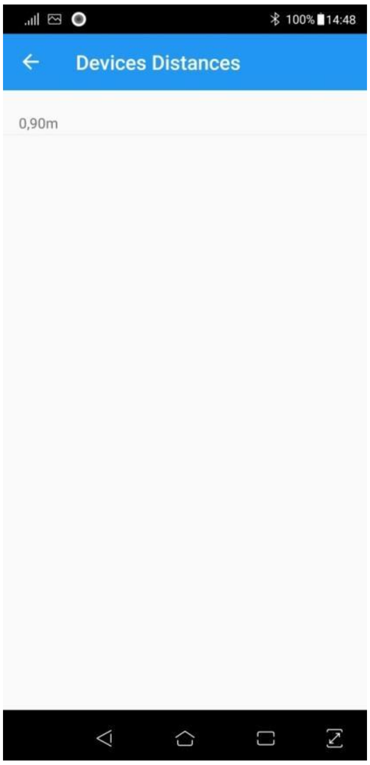
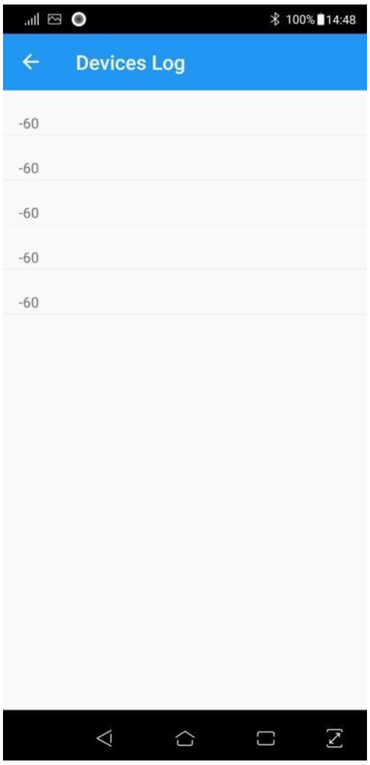


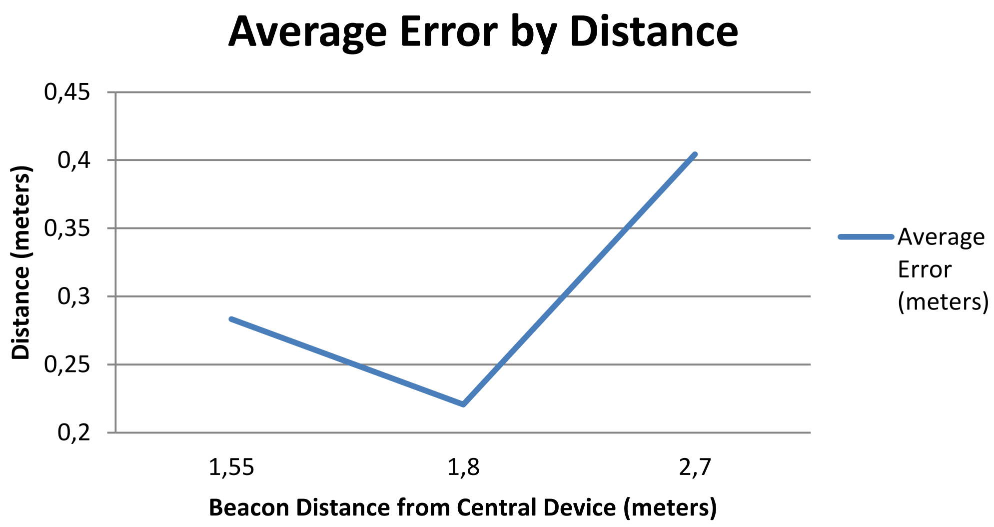
Figure 50 – Device RSSI log screen.



Source: Elaborated by the author.

The test initiated by positioning the iBeacon (ESP32 board) at a distance of 1.55, 1.8 and 2.7 meters from the smartphone, and 30 readings were made for each. After, the standard deviation was calculated, showing the average error for each distance in Graphic 1.

Graphic 1 - Average error by distance



Source: Elaborated by the author.

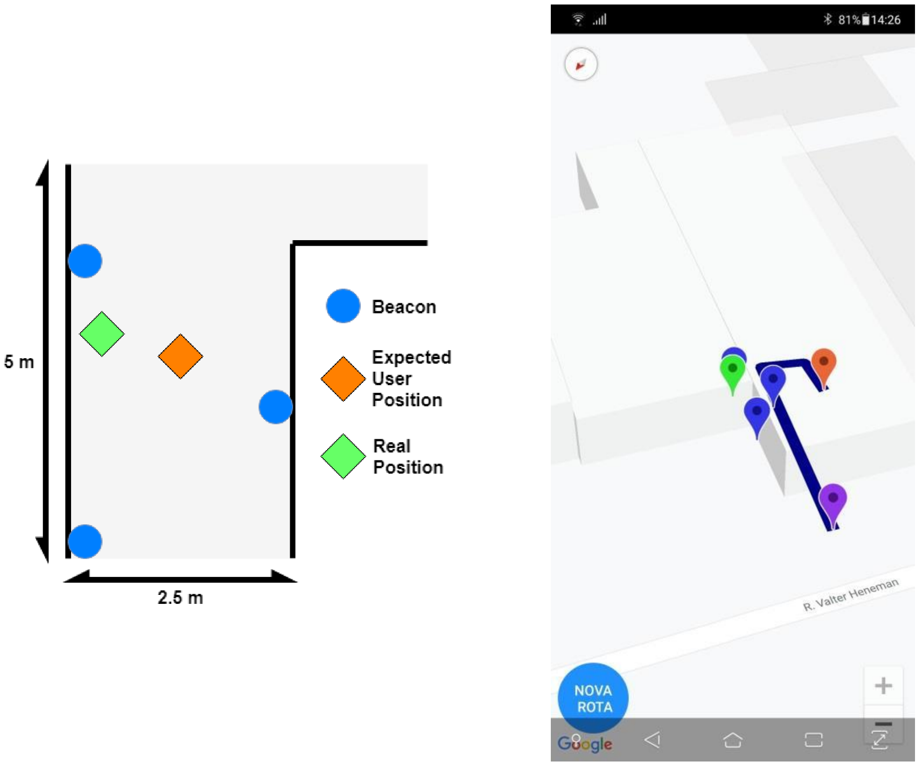
As demonstrated, all the estimations presented errors that were smaller than the average margin of 1 to 1.5 meters error.

6.1.2 Trilateration Technique Accuracy Tests

This experiment tested the features and the hardware presented in the section 5.3, testing the accuracy of the position displayed on map in relation to the real position on the environment. For this, a test route was created, and three Beacon devices installed on the environment, both in indoors and outdoors configuration.

The first test consists of an outdoor configuration, installing three Beacon devices on a corridor. Figure 52 demonstrates the outdoor configuration, where the purple marker highlights the initial location; the red marker the final location; the blue markers the Beacon devices locations and the green marker the user position in relation to three devices.

Figure 52 – Trilateration accuracy test – outdoor configuration.



Source: Elaborated by the author.

The first experiment presented an error smaller than the expected margin of 1 to 1.5 meters. Also, it presented the expected behavior, updating the map each three seconds, although, sometimes it presents a delay updating the location, due to the BLE signal instability, which occurred due to its propagation problems.

The second experiment also consists of an outdoor configuration, installing the Beacon devices in an open space environment. The disposition of the Beacons installed on the environment is demonstrated on Figure 53. For this experiment, it was chosen a location on Unisinos University, on sector C.

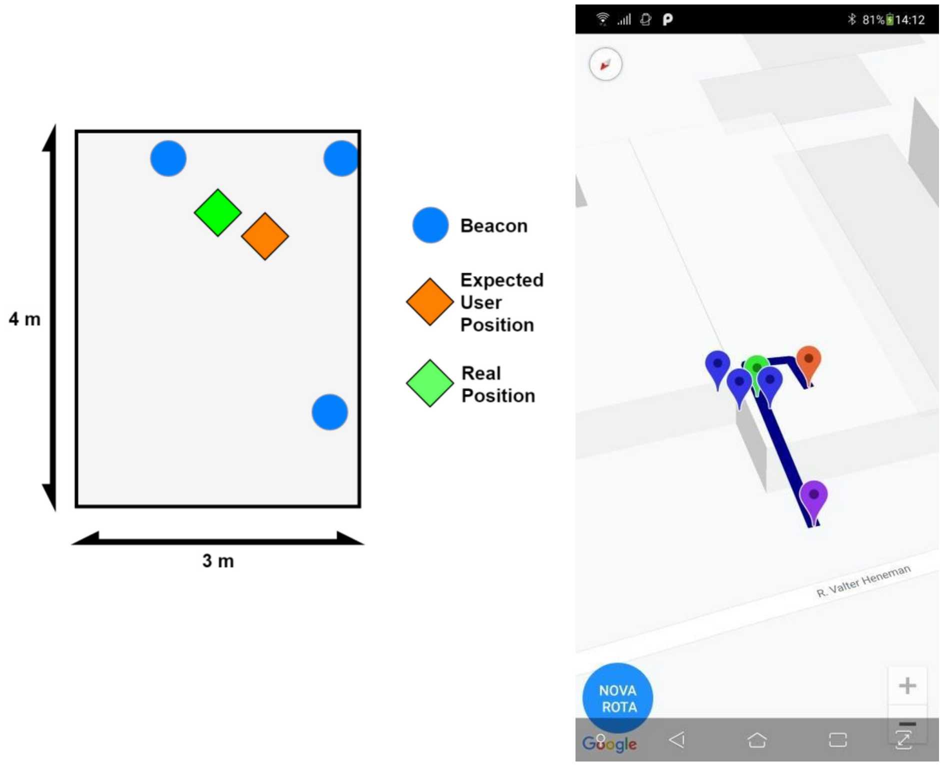
Figure 53 - Trilateration accuracy test – outdoor configuration.

Source: Elaborated by the author.

This experiment was not able to determine the correct location in the environment, presenting locations within the circle highlighted in orange, and presenting errors above the expected margin. There are many reasons that could cause this behavior. The first one is due to signal dispersion. This could have occurred due to the environment characteristics where the test happened, which is a corridor. It also could have occurred due to instability of the BLE signal, which can occur due to aspects such as multipath, multi propagation, environmental characteristics (temperature, humidity, other signals on the same frequency, among others) and the materials present in the environment. These are all hypotheses because no further tests were conducted to identify the cause of this behavior.

The third test consists of an indoor configuration, installing three Beacon devices on a room. Figure 54 demonstrates the indoor configuration, where the purple marker highlights the initial location; the red marker the final location; the blue markers the Beacon devices locations and the green marker the user position in relation to three devices.

Figure 54 - Trilateration accuracy test – indoor configuration



Source: Elaborated by the author.

As well as the first experiment, the second also presented the expected results, presenting an error smaller than the range margin of 1 to 1.5 meters. Also, it presented the expected behavior, updating the map each three seconds.

Besides the trilateration and the distance accuracy tests, the application functionality was also tested, being tested the pages navigation, the data entries, the routes generation as well as the alert messages, which were displayed in section 5.3.3, being this successful, where all of them presented the expected behavior.

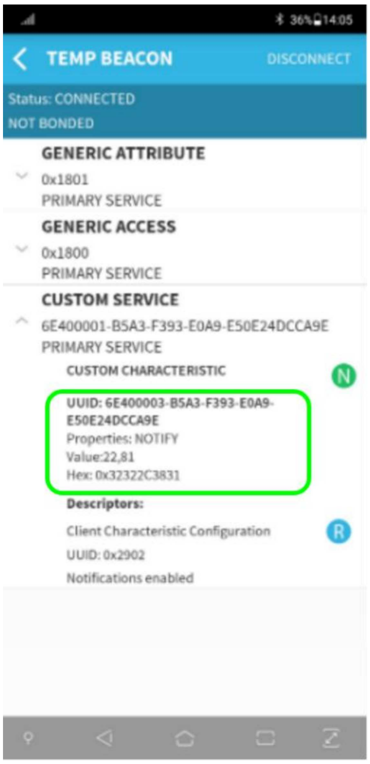
6.2 TEMPERATURE INFRASTRUCTURE EXPERIMENTS

For the temperature infrastructure, three experiments were conducted. The first tested the Temperature Beacon functionality. The second tested the centralizer functionality, testing the device on bridge mode and on perception mode. The third tested both the Gateway and the TTN server. The first following section will present the Temperature Beacon tests; the second will present the centralizer tests and the third the Gateway and TTN server tests.

6.2.1 Temperature Beacon Test

The first device tested in the temperature infrastructure was the Temperature Beacon (section 5.2.1.1), verifying if the notify property sent the temperature information when a central device connected to it. For this, the BLE Scanner application by Bluepixel Technologies was installed on an ASUS Zenfone 5Z smartphone, and then connected to the Beacon device through the BLE protocol. Figure 55 demonstrates the resulting screen after the smartphone connected to the Beacon. The green rectangle highlights the information received, where the value property shows the values 22,81, where 22 is the temperature in Celsius degree and the 81 is the humidity percentage.

Figure 55 – Temperature Beacon – notify property test.



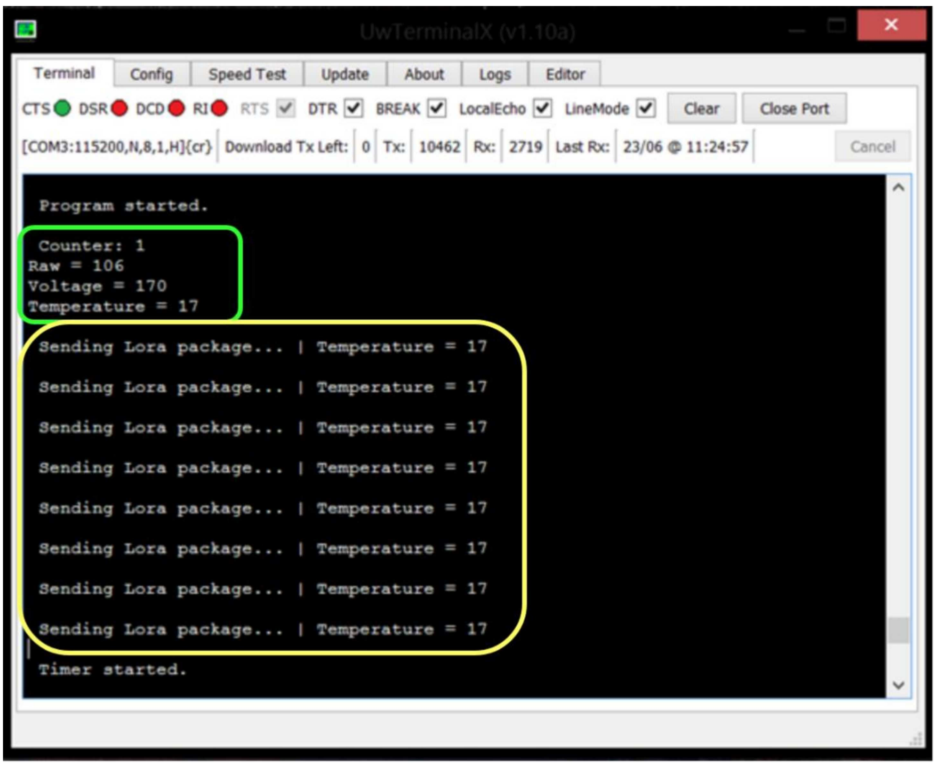
Source: Elaborated by the author.

This proved that the Temperature Beacon worked as expected and presented the predicted results.

6.2.2 Centralizer Tests

The second device tested in the temperature infrastructure was the centralizer (section 5.2.1.2), testing its functionality when in the two available modes, bridge and perception. The first mode tested was the perception mode, where the device received the temperature value through the sensor connected to it via serial protocol. For this, the UwTerminalX software was used, which is a cross-platform utility for communicating and downloading applications onto Laird's modules. After programming the developed script onto the module, the script was executed, resulting in the screen displayed in Figure 56. Highlighted in green is the information collected by the sensor, showing the raw information collected and the voltage and the temperature, which were converted based on the raw information obtained. Highlighted in yellow are the messages displayed when the data packages are sent to the Gateway via LoRa protocol.

Figure 56 – Centralizer test – perception mode.



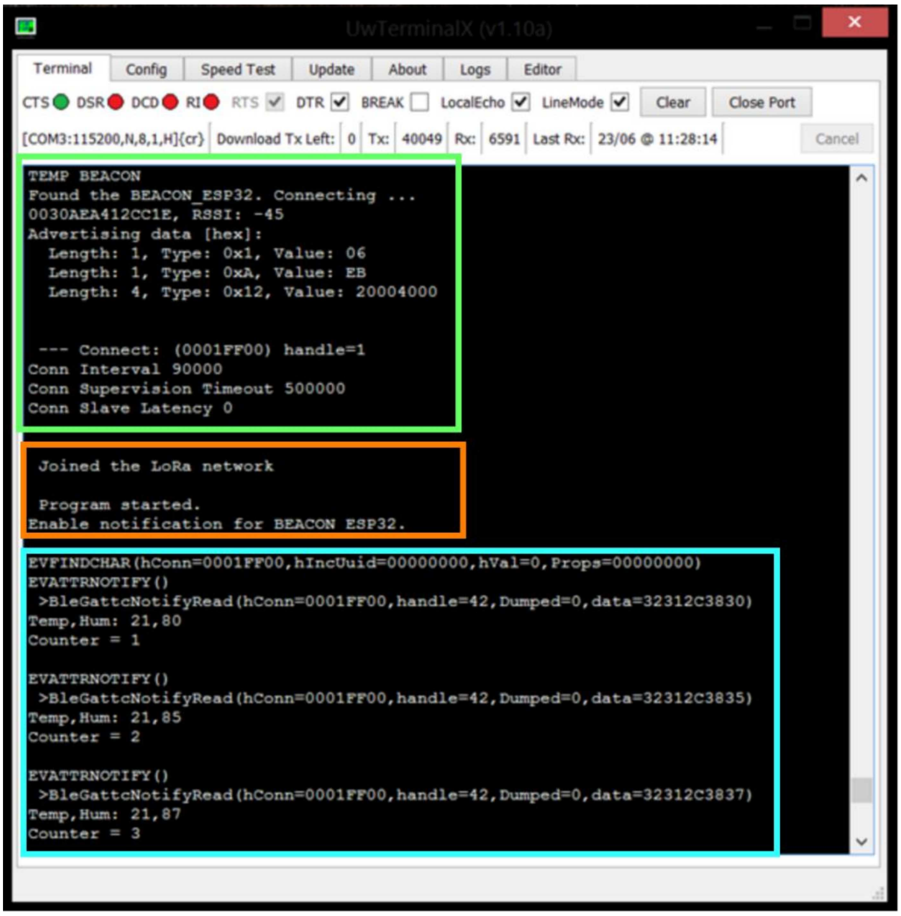
Source: Elaborated by the author.

This test was successful, presenting the predicted results. The only issue presented was by the temperature sensor accuracy, which presented an error of about 2 to 3 degrees on its readings.

The second mode tested was the bridge mode, where the device received the temperature value through its connection with the Temperature Beacon. Again, the UwTerminalX software was used for the module's programming. After its programming, the script was executed, resulting in the screen displayed in Figure 57.

Highlighted in green are the information about the Beacon device discovered as well as the connection message displayed when the module connects with the device. Highlighted in orange is the message displayed when the module joins the LoRa network. Highlighted in blue are the messages displayed when a data package is received by the module from the Beacon, displaying the data both in hexadecimal and decimal forms.

Figure 57 - Centralizer test – bridge mode



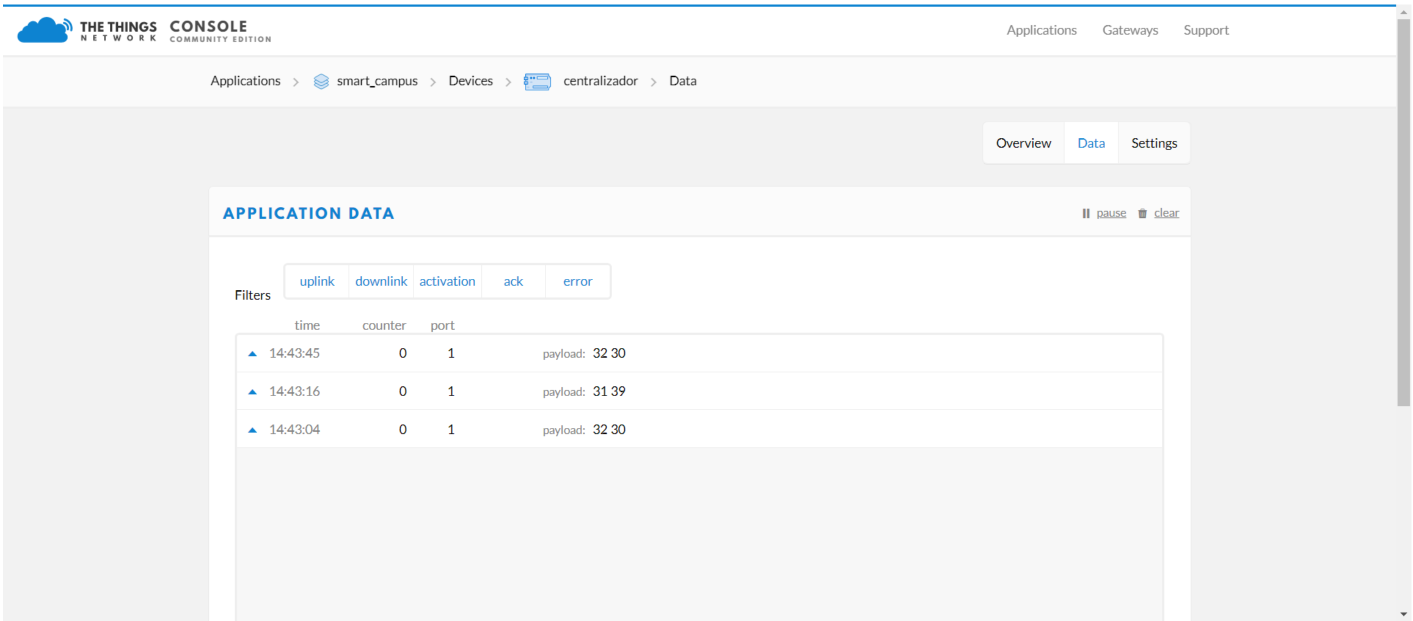
Source: Elaborated by the author.

This test was also successful, presenting the expected results. It received the correct temperature value from the Beacon, and fulfilled its function by sending this received package to the Gateway via LoRa protocol.

6.2.3 TTN Server Tests

The TTN server was tested in two ways. The first was by the Gateway which forwarded the received packages to the TTN server and this displayed the received data in the application console on its website. The second was by simulating the uplink to the server, which also displayed the received data in the application console on its website. Figure 58 displays the application data console, which demonstrates the packages received and its information, such as time and message payload.

Figure 58 – TTN server test – application data console.



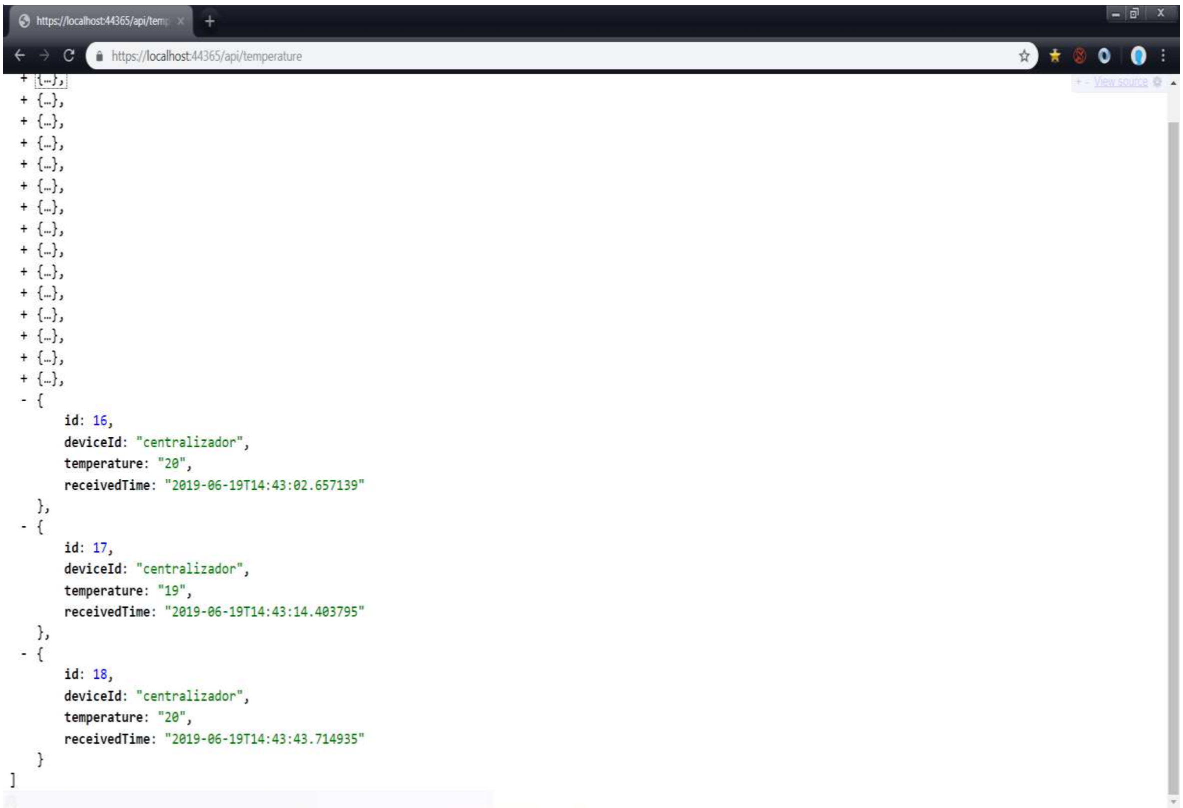
Source: Elaborated by the author.

As displayed in the console, the TTN server received the messages both simulated and sent by the Gateway, proving its functionality and presenting the expected results.

6.3 INTEGRATION APPLICATION EXPERIMENTS

For the integration application, one experiment was conducted. This tested if the application worked as expected, pulling the information stored in the TTN server and storing it in the application server. The script started by connecting with the TTN server and subscribing to its information through the MQTT protocol, as was explained in section 5.5. Then the extracted data was sent to the web API developed (section 5.4) via HTTP protocol, which stored this information on the Temperature table of the database, as shown in Figure 59.

Figure 59 – Web API displaying the information that was stored in the TTN server.



Source: Elaborated by the author.

The test was successful, where the application worked as expected, pulling the information stored in the TTN server in a determined interval and storing this on the application server database, as displayed in Figure 59. This displays the same information that was stored in the TTN server, displayed in Figure 58.

6.4 DATA ANALYSIS RESULTS

In order to validate the results obtained applying both the clustering techniques and the data analysis techniques implemented in the data analysis unit, two analyses were made. The first analysis aimed to validate the two clustering techniques applied, identifying which one of them presents the best results for the dataset applied, therefore using internal criteria to validate the results. For this quality validation, three indexes were used. The CH Score (Calinski Harabasz Score), the DB Score (Davies-Bouldin Score) and the Silhouette Coefficient. The second analysis applied two data analysis techniques, time series and frequency, in order to extract valid information from the data stored.

The following sections will demonstrate the results obtained for the localization data, presenting the results for the indexes used to validate the clustering techniques applied, as well as presenting the results of a time series and the frequency obtained using the data of one of the clusters classified by the K-Means clustering.

The dataset used in the experiments is a dataset created specifically for the experiments, as described previously (section 5.6). The location dataset consists of 16.697 lines, mapping thirty routes and containing information such as the user identification number, the initial location of the route, the final location of the route, the date and time of the location registration, the location information (latitude and longitude) and which cluster the location was classified to. The temperature dataset consists of 503 lines, containing information about the devices, the temperature collected and the date and time which the information was collected.

6.4.1 Localization Data

In order to analyze the classification quality of the clustering techniques applied, three indexes were chosen, the CH Score (Calinski Harabasz Score), the DB Score (Davies-Bouldin Score) and the Silhouette Coefficient. The first difference between the classifications is the number of clusters, which for the K-Means were three clusters, as determine as the best number of clusters for the classification through the Elbow method analysis; and for the HDBSCAN were four clusters, determined automatically by its algorithm.

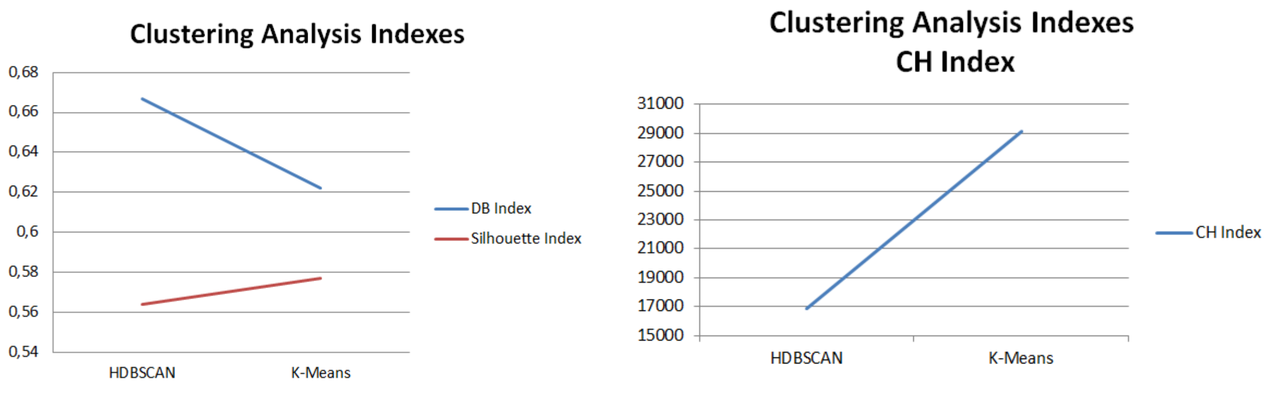
The Calinski Harabasz Score determines the ratio between within-cluster dispersion and between-cluster dispersion, where a bigger ratio represents a better classification. Furthermore, the Davies-Bouldin Score measures the similarity between the clusters, also using a ratio to determine it. The ratio is defined by the relation of within-cluster distances to between-cluster distances, where lower values indicate a better classification. Finally, the Silhouette Coefficient validates the consistency of the classification of each cluster by analyzing the distance between them. The coefficient varies in an interval between -1 to 1, where the higher the value of the coefficient, better the classification. The results obtained for each are presented in Table 6 and demonstrated in Figure 60.

Table 6 - Comparison of Clustering analysis indexes

	HDBSCAN	K-Means
CH Index	16905,41	29124,64
DB Index	0,6667	0,622
Silhouette Index	0,564	0,577

Source: Elaborated by the author.

Figure 60 – Clustering indexes results.

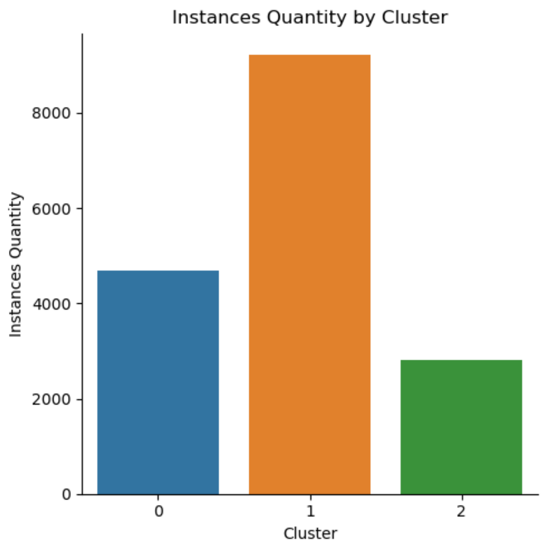


Source: Elaborated by the author.

As demonstrated, the classification made by the K-Means algorithm presented the best results in all indexes. These results could be associated to the number of clusters defined for the classification, which for the K-Means was predetermined as three. It also could be associated to the dataset structure and the data type, which in the K-Means algorithm used the Euclidean distance for the classification of the locations, while in the HDBSCAN it was necessary a transformation to radians in order to execute the classification.

After the classification, it was executed a time series analysis (Figure 62) in one of the classified clusters by the K-Means algorithm, as well as a frequency analysis (Figure 61) on the resulting clusters.

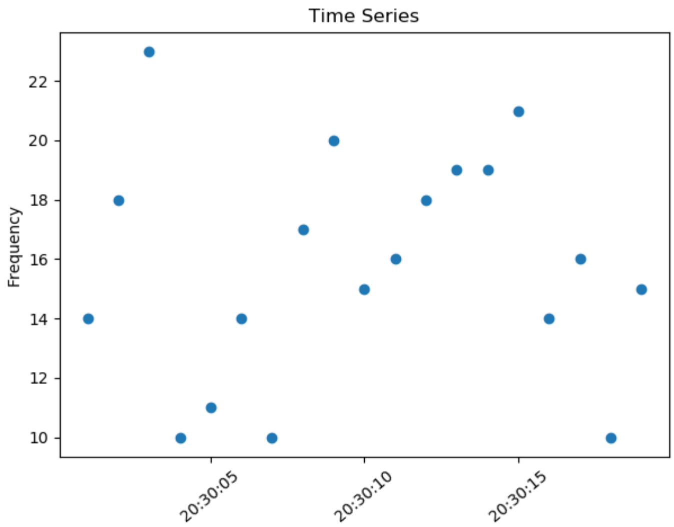
Figure 61 – Frequency of each cluster classified by the K-Means algorithm.



Source: Elaborated by the author.

The time series analyzed the frequency in which a location was visited on campus, as displayed in Figure 62 which extracted an interval from the dataset of a cluster classified by the K-Means algorithm.

Figure 62 – Time series of a cluster classified by the K-Means algorithm.



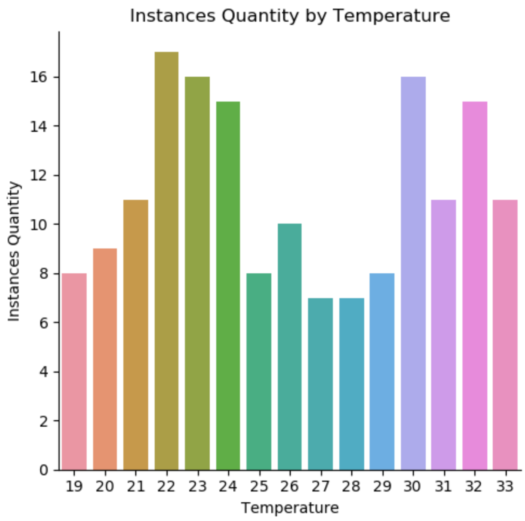
Source: Elaborated by the author.

These analyses together with the cluster classification displayed in Figure 44, demonstrate the capability of the system in helping the stakeholders in decision-making and in the planning for the improvement on campus resources and services. It is able to support other analysis, being capable of generate new and valuable data from the data gathered. Such feature is demonstrated by the data analyzed which presents the usage of the campus, highlighting important regions in the form of the clusters classified, as well as its frequency and time series, which highlights which region is more visited and which intervals a determined route is more traveled.

6.4.2 Temperature Data

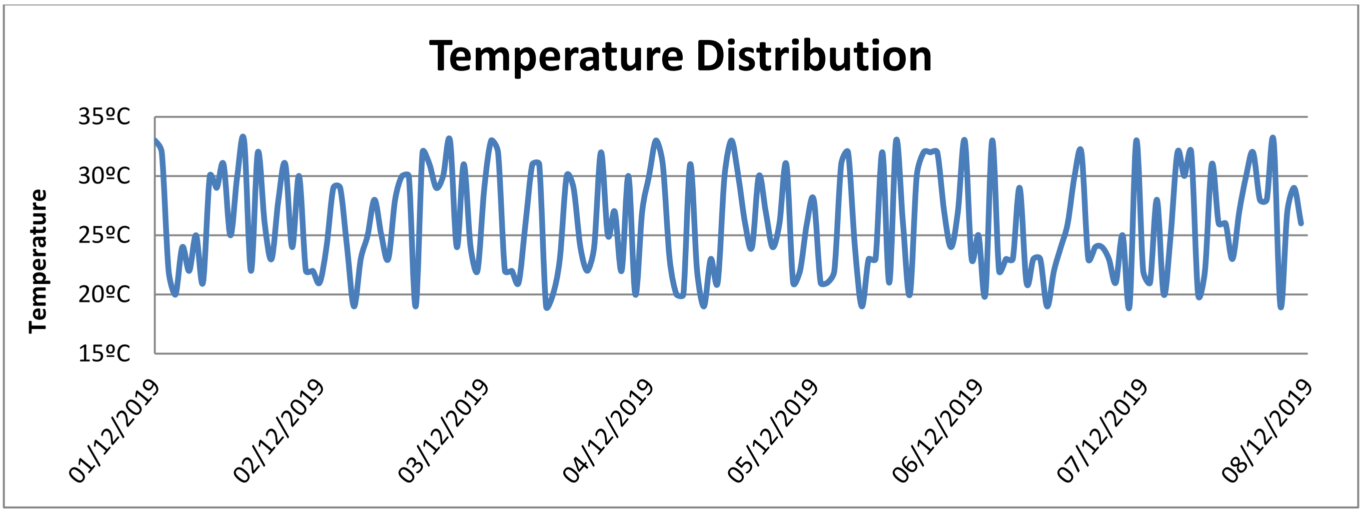
For the temperature data, a time series analysis was made for each device, analyzing the distribution of the temperature and its frequency, highlighting the temperature distribution on a determined area on campus. Figure 63 demonstrates the temperature frequency and Figure 64 demonstrates the time series for the device Centralizador 1.

Figure 63 – Temperature frequency – Centralizador 1.



Source: Elaborated by the author.

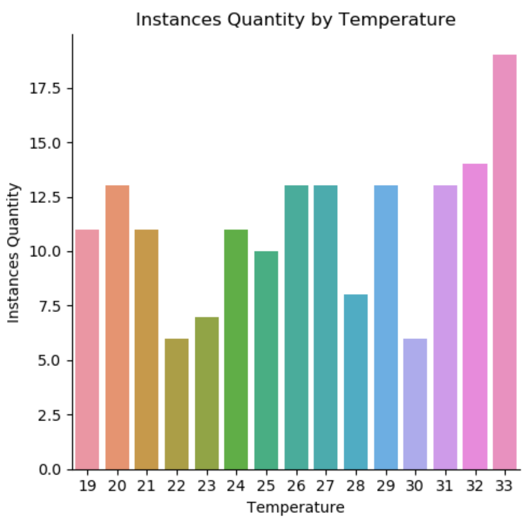
Figure 64 – Time Series – Centralizador 1



Source: Elaborated by the author

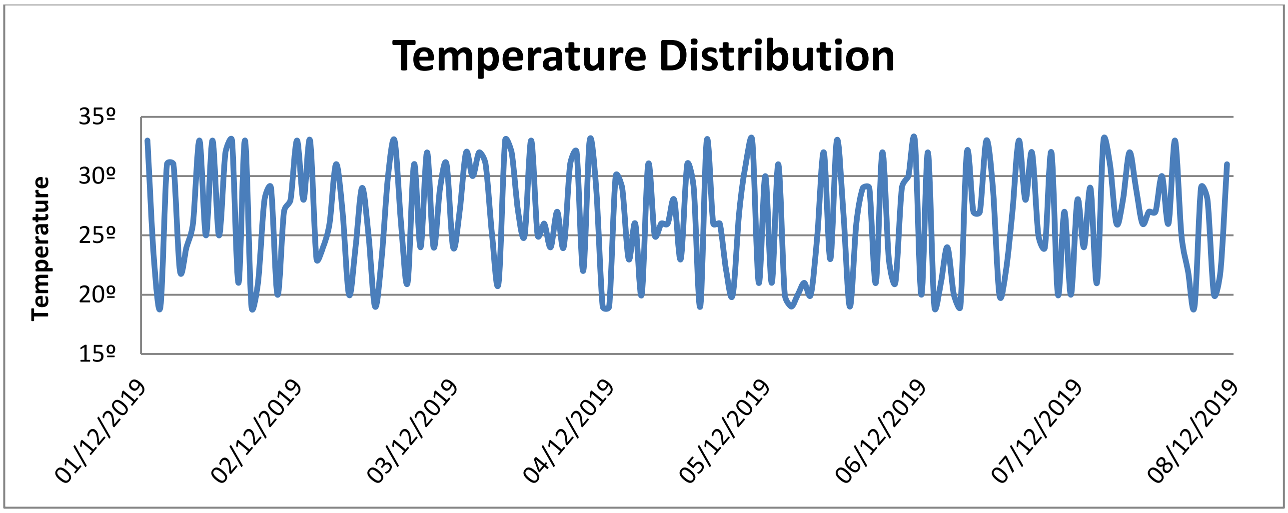
Figure 65 demonstrates the temperature frequency and Figure 66 demonstrates the time series for the device Centralizador 2.

Figure 65 - Temperature frequency – Centralizador 2.



Source: Elaborated by the author.

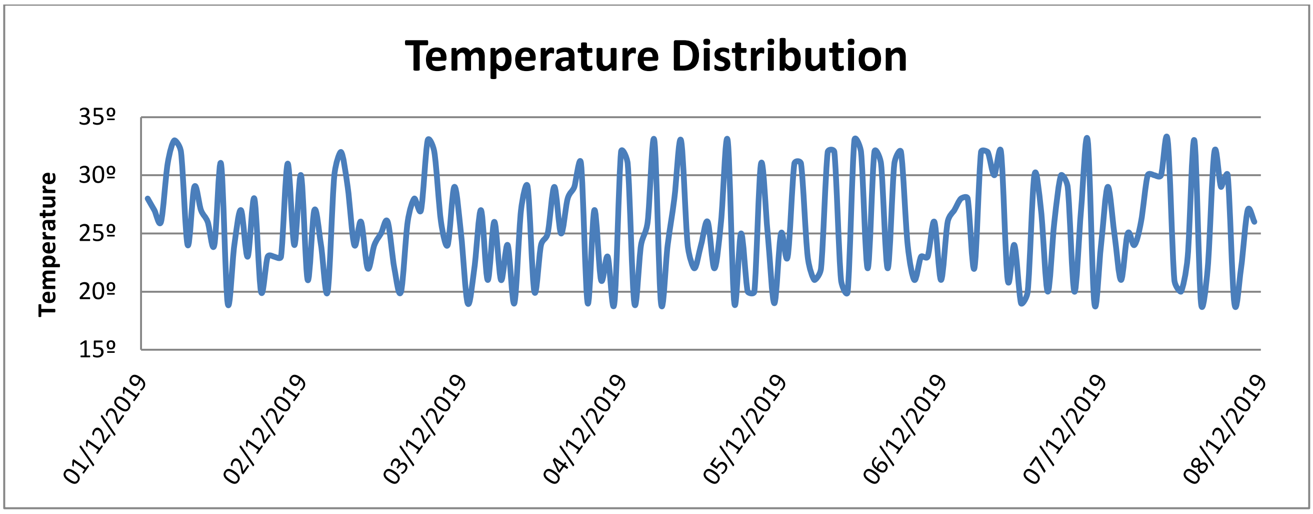
Figure 66 - Time Series – Centralizador 2.



Source: Elaborated by the author.

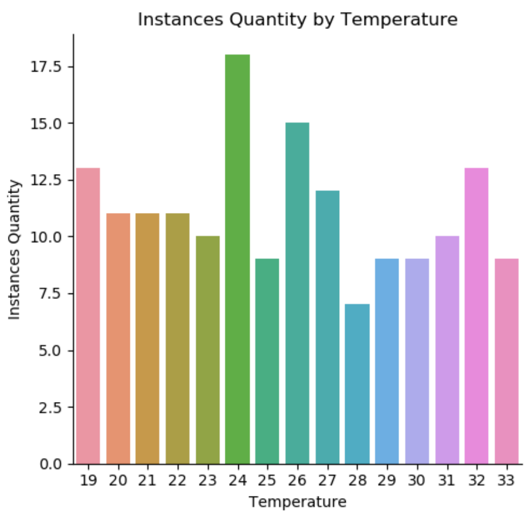
Figure 68 demonstrates the temperature frequency and Figure 67 demonstrates the time series for the device Centralizador 3.

Figure 67 - Time Series – Centralizador 3.



Source: Elaborated by the author.

Figure 68 - Temperature frequency – Centralizador 3.



Source: Elaborated by the author.

Other sensors could be added to the existing infrastructure, generating other data which could help understand the campus behavior, and as previously mentioned, could help the stakeholders in decision-making and in the planning for improvements on campus.

6.5 LIMITATIONS OBSERVED

Along the research and development of the prototype structures some limitations were identified. Firstly, the research focused on the data analysis techniques used for the analysis of the data gathered, which were the user localization and temperature data. Secondly, the research was not able to gather the necessary data size for the data analysis due to a time constraint, simulating them instead.

6.6 RESULTS ANALYSIS

In order to validate the Sapientia model's structure, an infrastructure was developed, consisting of four structures: localization, temperature measuring, integration application, and data analysis.

The localization structure allowed us to test several layers of the model and how they operate together. It starts with the Perception layer, with the installation of the Beacon devices on the environment, which served as a reference for the mobile application developed (located in the Application layer). The smartphone used for the mobile application worked as middleware (located in the Middleware layer), receiving the signal from the Beacons, filtering it and processing it, and sending the location data to the application server (located in the Data Storage layer), in order for it to be analyzed posteriorly by the Data Analysis Unit, the fourth structure.

The second structure was the temperature measuring structure, which tested mainly the first two layers (Perception and Network layers) of the model and its versatility. It started by the implementation of two different applications that gathered the same information (temperature), where one of them used the same device applied for the localization structure (Beacon), together with a centralizer device (Perception layer), which can operate as a bridge, sending the received information to the server and also as perception device, gathering the data itself and sending the information to the server (Data Storage layer). On both applications, the second module of the model was tested, using communication protocols in order to gather the data and transfer it to the server (Data Storage layer). As the previous structure, this also sent the gathered data to the server, in order for it to be analyzed posteriorly by the Data Analysis Unit (System layer). Also, this structure worked together with the third structure, which consists of an integration application (Application layer) aimed to extract the data stored in the LoRa server (Data Storage layer), apply the necessary conversion on the data and store it in the application server (Data Storage layer).

Finally, all the previous structures were crucial in order to gather the data used for the validation of the fourth structure, the Data Analysis Unit. This unit processed and analyzed the data stored in the Data Storage layer, which together with the Artificial Intelligence techniques as well as the data analysis techniques, are able to supply and generate new valuable information, which can be used by several applications, generating new and meaningful information. Besides, the data stored in the Data Storage Unit can also be accessed and used by other applications.

With the performed tests, it was possible to validate the model’s structure as well as its purpose, being flexible and able to incorporate accessibility aspects and existing solutions, besides offering support for both partial (using some of the modules) and complete (complete structure) solutions.

7 CONCLUSION

This text presented the Sapientia Smart Campus model, aiming to promote flexibility, to include accessibility aspects, and to be able to incorporate existing hardware and software solutions. The model is distributed in layers, which makes the management, updating, and addition of new technologies more accessible. Also, it promotes flexibility, allowing the same device to work in different applications with different usages, besides being able to incorporate existing solutions. It also includes accessibility aspects in different layers, encouraging inclusion, and helping to offer better services.

A prototype divided into four structures was developed. The first structure consists of a mobile application that allied with Beacon devices helps the user to navigate through campus and also stores the collected information on a server. The second structure used both Beacons allied with sensors and the centralizer device to measure the campus temperature, storing this on a LoRa server. The third structure consists of an integration application, which pulled the temperature data stored in the second structure server (LoRa server) and stored it in the first structure server, the application server. The fourth structure consists of a data analysis unit, which analyzed the stored data on the application server.

Seeking to answer the research question detailed in section 1.1, were implemented the aforementioned structures. The mobile application helped the user to navigate through campus, and the temperature monitoring application acquires this kind of data. In both contexts, the mobile application structure, and the temperature measuring structure, the same Beacon device was used, therefore supporting the claim for the model’s flexibility, exemplifying the use of the same device for different applications and configurations. The prototype, composed of four structures, as well as the existing external solutions incorporated, evidences that the model is capable of incorporating different solutions on the same infrastructure, supporting different communication protocols and applications, being able to incorporate distinct technologies.

The original prototype was expanded with experiments for the integration of some different applications. In this text, we mentioned two examples: a bus position tracking application throw Lora, and a semi-automatic well-chair for people with disabilities. The first application measured position information received from Buses, using a similar infrastructure from the temperature measuring structure (section 5.2), by sending the collected location data to a LoRa server. Using the implemented integration application (section 5.5), is able to extract the information stored in the LoRa server and store it on the application server in order to process and analyze the bus location information using the Data Analysis Unit implemented (section 5.6). The second application integrated one application aimed to support people with disabilities and the elderly by guiding them using an intelligent system, the Hefestos. This application used the Beacon device jointly, although in different manners, in order to help guide in the traveling of an indoor environment, acting as a reference point for the application. These could also communicate with the application server by using the API developed, storing the data in the server for posterior processing and analysis by the Data Analysis Unit. In both experiments, we aimed to show evidence about the easiness provided from the Sapientia model for the incorporation of the existing application into the infrastructure, besides integrating accessibility aspects in different layers, which will be able to support both complete (complete structure) and partial (only applications) solutions.

Besides the previously detailed features, the model also allows the incorporation of Artificial Intelligence and data analysis techniques to process and manipulate the stored data. This generates new valuable information and supporting several useful analyses, thus helping in decision-making and the management and improvement of the campus resources and services.

7.1 CONTRIBUTION

In this section we highlight and discuss the most important contributions obtained in this research.

According to Table 4 and Table 5, generated from the related works studied, we can observe a gap in the works, regarding the models and its flexibility and support for accessibility. We addressed this gap proposing a model which presents flexibility, allows the inclusion of accessibility aspects and has the capability of incorporating existing hardware and software solutions. Besides, the model offer support and facilitates for solutions to manage some IoT challenges and issues, offering an architecture that allows for easy management and technology updating. It also incorporates a Data Analysis Unit, consisting of both Artificial Intelligence and data analysis techniques, which allows to analyze the information stored, generating new data that can be shared by other applications, which can culminate in better services for the campus users and can help the campus managers in decision-making and in the improvement and management of the campus resources and services. The architecture allows for applications to be incorporated in an easy way, seeking to facilitate the adhesion of existing applications and solutions.

The systematic literature review is another contribution, gathering information about Smart Campus models and its requirements and basic features, as well as demonstrating the lack of models that integrate accessibility aspects in the model's infrastructure. It also highlighted possible IoT challenges faced when implementing solutions for Smart Concepts.

Another contribution was the hardware infrastructure, which mixed different communication protocols and devices, demonstrating the many possibilities and versatility of IoT technologies, such as the Beacon device. The devices developed and adapted during this research are also considered a contribution, which will be available for other projects, and by working with other solutions can lower the costs of the project and also encourage sustainability. Likewise, the mobile application also contributed by demonstrating the many possibilities when combining hardware and software solutions, which can offer better services for the campus users, like helping users to move inside the university.

The research also generated contributions in the form of two papers submitted, one consisting of the literature review, and another consisting of the model, experiments and results. It also contributed with the infrastructure developed, consisting of the four structures previously explained, creating a living lab and serving as a base infrastructure for other projects, being it as an expansion of the existing structures or for new solutions which can be added to the existing infrastructure, besides the generation of real data which can be analyzed posteriorly by the Data Analysis Unit.

The collaborations made with both the Smart Bus and the MobiLab projects are also a contribution, by encouraging the interaction and integration of different researches and knowledge areas, which can share the resources used, making the projects more sustainable and can generate new perspectives, improving the researches knowledge.

The applications developed are also considered a contribution, where the Integration application is able to gather information from different sources (servers), being of easy adaptation, configuration and integration, which can easily be used in other projects. The Data Analysis Unit can process and generate new valuable information, supporting both existing and new solutions by sharing the data analyzed, which can help improve and offer better services, besides sharing the data simulated that was used for the tests and validation, which can help other applications in their validation and testing. Besides, the mobile application developed will be registered, which in the future is intended to be made available for the campus users.

The research question was answered through the contributions previously mentioned, mainly regarding the Sapientia Model, this being flexible due to its structure, which enables the device's versatility, besides being able to incorporate new and existing solutions by proposing a layered model, which can easily integrate new hardware and software elements, also including accessibility aspects on different layers of the model, which will offer accessibility support for both complete and partial solutions and integrations.

7.2 FUTURE WORK

For future work, firstly, the devices used in this research should be installed permanently on the campus. In addition, the structure of the prototype developed could be expanded, adding other sensors and devices, as well as implementing other communication protocols. Also, new solutions could be added as well as incorporating the existing ones into the infrastructure. The mobile application could also be expanded, offering new features. In addition, the Data Analysis Unit should process and analyze real data, collected on a large scale and through a bigger time frame, testing its capability and performance, and comparing the results obtained through the simulated experiment and the real experiment.

The routes collection in the mobile application could also be expanded, adding the remaining possible routes into the system, amplifying the coverage area of the application for the whole campus. It also could incorporate other Artificial Intelligence features, like ontologies, which could map the accessibility resources on campus, and expand the NPL techniques applied, which would offer new accessibility features to the users.

Another feature to be worked on is the hardware developed and adapted for the research. The Beacon devices offer many opportunities for both hardware and software. For the hardware, new sensors could be added to it, or a filter could be added, which would improve and stabilize its signal, which is its bigger weakness. For the software, a Neural Network could be trained in order to help stabilize its signal, or new filter techniques could be used in order to improve its distance accuracy estimation.

For the centralizer, as the Beacon, new sensors could be added to the device, as well as new features and applications added to it, which due to using the LoRa communication protocol, offers many opportunities for long range applications.

A dashboard displaying information about the campus resources and usage could be developed in order to present the data to the campus stakeholders better, helping them in the decision-making, improvement and management of the campus. Using information such as the time series and the map displaying the localization data classification implemented, it can be determined the date and time in which the campus is most busier and which are the most interesting locations. This could help to determine the best locations to place a determined resource, or which locations will need more maintenance, like bathrooms, for example.

REFERENCES

____. IBGE. **Pesquisa Nacional de Saúde**, 2013. Available on: <http://www.ibge.gov.br>. Access on: 20/12/2017.

Abou-Zahra, S., Brewer, J., & Cooper, M. (2017). Web standards to enable an accessible and inclusive internet of things (IoT).

Abuarqoub, Abdelrahman, *et al*. "A Survey on Internet of Thing Enabled Smart Campus Applications." *Proceedings of the International Conference on Future Networks and Distributed Systems*. ACM, 2017.

Adamkó, A., Kãdek, T., & Kósa, M. (2014, November). Intelligent and adaptive services for a smart campus. In *2014 5th IEEE Conference on Cognitive Infocommunications (CogInfoCom)* (pp. 505-509). IEEE.

Alaba, Fadele Ayotunde, *et al*. "Internet of things security: A survey." *Journal of Network and Computer Applications* 88 (2017): 10-28.

Alghamdi, Abdullah, and Sachin Shetty. "Survey toward a smart campus using the internet of things." *Future Internet of Things and Cloud (FiCloud), 2016 IEEE 4th International Conference on.* IEEE, 2016.

Ali, Anum, *et al*. "Technologies and challenges in developing machine-to-machine applications: A survey." *Journal of Network and Computer Applications* 83 (2017): 124-139.

Atif, Y., & Mathew, S. (2013, August). A social web of things approach to a smart campus model. In *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing* (pp. 349-354). IEEE.

Bandara, H. M. A. P. K., Jayalath, J. D. C., Rodrigo, A. R. S. P., Bandaranayake, A. U., Maraikar, Z., & Ragel, R. G. (2016, October). Smart campus phase one: Smart parking sensor network. In *2016 Manufacturing & Industrial Engineering Symposium (MIES)* (pp. 1-6). IEEE.

Beacon ESP32. ESP32 SoC Beacon Application, 2019. Available in: < https://www.fernandok.com/2018/08/esp32-bluetooth-low-energy.html >. Access on 04/06/2019.

Bracco, S., Brignone, M., Delfino, F., & Procopio, R. (2015). An energy management system for the savona campus smart polygeneration microgrid. *IEEE Systems Journal*, *11*(3), 1799-1809.

BRASIL. Lei Nº 10.098, de 19 de dezembro de 2000. Acessibilidade, Brasília, DF. Available in: <http://www.planalto.gov.br/ccivil_03/leis/l10098.htm>. Acess on: 30/05/2019.

BRASIL. Lei Nº 13.146, de 6 de julho de 2015. Acessibilidade, Brasília, DF. Available in: <http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2015/lei/l13146.htm>. Acess on: 30/05/2019.

Braun, Trevor, *et al*. "Security and privacy challenges in smart cities." *Sustainable cities and society* 39 (2018): 499-507.

Caţă, Marian. "Smart university, a new concept in the Internet of Things." *RoEduNet International Conference-Networking in Education and Research (RoEduNet NER)*, 2015 14th. IEEE, 2015.

Chen, Runze, *et al*. "Construction of a voice driven life assistant system for visually impaired people." *2018 International Conference on Artificial Intelligence and Big Data (ICAIBD)*. IEEE, 2018.

Climatempo. Metereologia – Porto Alegre, 2019. Available in: <https://www.climatempo.com.br/climatologia/363/portoalegre-rs>. Access on 28/12/2019.

Crowley, D., Curry, E., & Breslin, J. G. (2016). Citizen actuation for smart environments: Evaluating how humans can play a part in smart environments. *IEEE Consumer Electronics Magazine*, *5*(3), 90-94.

De Blas, A., & López-de-Ipiña, D. (2017, July). Improving trilateration for indoors localization using BLE beacons. In *2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech)* (pp. 1-6). IEEE.

Debatin, Lucas, *et al*. "Implementação de API para Reconhecimento e Sintetização de Voz em um Aplicativo Móvel." *Anais do XIII Simpósio Brasileiro de Sistemas de Informação*. SBC, 2017.

DHT11 Sensor. DHT11 Temperature & Humidity Sensor features, 2019. Available in: <https://www.mouser.com/ds/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>. Access on 04/06/2019.

Di Martino, B., Rak, M., Ficco, M., Esposito, A., Maisto, S. A., & Nacchia, S. (2018). Internet of things reference architectures, security and interoperability: A survey. *Internet of Things*, *1*, 99-112.

E. Borgia, The internet of things vision: key features, applications and open issues, Comput. Commun. 54 (2014) 1–31, doi: 10.1016/j.comcom.2014.09. 008.

Electronic Notes. Long Term Evolution (LTE) Overview, 2019. Available In: <https://www.electronics-notes.com/articles/connectivity/4g-lte-long-term-evolution/what-is-lte-basics-tutorial-overview.php>. Access on 05/06/2019.

ESP32. ESP32 SoC Development Board, 2019. Available in: <https://www.espressif.com/en/support/download/documents>. Access on 04/06/2019.

Ethernet Operation. Ethernet Protocol, 2019. Available in: <http://teachweb.milin.cc/datacommunicatie/tcp_osi_model/data_link_layer/ethernet_protocol.htm>. Access on 23/06/2019.

Farhan, Laith, *et al*. "A Survey on the Challenges and Opportunities of the Internet of Things (IoT)." *Sensing Technology (ICST), 2017 Eleventh International Conference on*. IEEE, 2017.

Fernandez-Anez, Victoria, José Miguel Fernández-Güell, and Rudolf Giffinger. "Smart City implementation and discourses: An integrated conceptual model. The case of Vienna." *Cities* 78 (2018): 4-16.

Ferreira, Francisco Henrique Cerdeira, and Renata Mendes Araújo. "Campus Inteligentes: Conceitos, aplicações, tecnologias e desafios." RelaTe-DIA 11.1 (2018).

GATT. GATT Services and Characteristics, 2019. Available in: <https://www.oreilly.com/library/view/getting-started-with/9781491900550/ch04.html>. Access on 04/06/2019.

Ghidini, Eduardo, *et al*. "Developing apps for visually impaired people: Lessons learned from practice." *2016 49th Hawaii International Conference on System Sciences (HICSS)*. IEEE, 2016.

Giri, Arindam, *et al*. "Internet of things (IoT): a survey on architecture, enabling technologies, applications and challenges." *Proceedings of the 1st International Conference on Internet of Things and Machine Learning*. ACM, 2017.

Global TAG BLE. Bluetooth Low Energy (BLE) Overview, 2019. Available in: < https://www.global-tag.com/bluetooth-low-energy-ble-overview/>. Access on 05/06/2019.

Gorovyi, I., Roenko, A., Pitertsev, A., Chervonyak, I., & Vovk, V. (2017, May). Real-time system for indoor user localization and navigation using bluetooth beacons. In *2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)* (pp. 1025-1030). IEEE.

Gündoğan, C., Kietzmann, P., Schmidt, T. C., Lenders, M., Petersen, H., & Wählisch, M. (2018). NDN, CoAP, and MQTT: A Comparative Measurement Study in the IoT. *arXiv preprint arXiv:1806.01444*.

H. Mora, V. Gilart-Iglesias, R. Pérez-Delhoyo, M.D. Andújar-Montoya, and H.J. Compañ Gabucio. 2016. Interactive cloud system for the analysis of accessibility in smart cities. Int. J. Des. Nat. Ecodynamics 11, 3 (jul 2016), 447–458. https: //doi.org/10.2495/DNE-V11-N3-447-458.

HDBSCAN. The HDBSCAN Clustering Library, 2019. Available in: < https://hdbscan.readthedocs.io/en/latest/index.html/>. Access on 20/12/2019.

HTTP. HTTP Overview, 2019. Available in: <https://www.webnms.com/cagent/help/technology_used/c_http_overview.html>. Access on 05/06/2019.

Huh, J. H., & Seo, K. (2017). An indoor location-based control system using bluetooth beacons for IoT systems. *Sensors*, *17*(12), 2917.

Jeon, K. E., She, J., Soonsawad, P., & Ng, P. C. (2018). Ble beacons for internet of things applications: Survey, challenges, and opportunities. *IEEE Internet of Things Journal*, *5*(2), 811-828.

Kaushik, S. (2012). An overview of technical aspect for WiFi networks technology. *International Journal of Electronics and Computer Science Engineering (IJECSE, ISSN: 2277-1956)*, *1*(01), 28-34.

Khatoun, Rida, and Sherali Zeadally. "Smart cities: concepts, architectures, research opportunities." *Communications of the* ACM 59.8 (2016): 46-57.

Khomp. Long Term Evolution (LTE), 2019. Available In: <https://www.khomp.com/pt/rede-lte/>. Access on 05/06/2019.

Kolias, C., Copi, L., Zhang, F., & Stavrou, A. (2017, April). Breaking BLE beacons for fun but mostly profit. In *Proceedings of the 10th European Workshop on Systems Security* (p. 4). ACM.

Labella, A., Mestriner, D., Procopio, R., & Delfino, F. (2017, June). A simplified first harmonic model for the Savona Campus Smart Polygeneration Microgrid. In *2017 IEEE International Conference on Environment and Electrical Engineering and 2017 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)* (pp. 1-6). IEEE.

Lavric, A., & Popa, V. (2017, July). Internet of things and LoRa™ low-power wide-area networks: a survey. In *2017 International Symposium on Signals, Circuits and Systems (ISSCS)* (pp. 1-5). IEEE.

LDA. Leis de Acessibilidade, 2019. Available in: < http://blog.handtalk.me/leis-de-acessibilidade/ >. Access on 30/05/2019.

Liando, J. C., Gamage, A., Tengourtius, A. W., & Li, M. (2019). Known and Unknown Facts of LoRa: Experiences from a Large-scale Measurement Study. *ACM Transactions on Sensor Networks (TOSN)*, *15*(2), 16.

LM35 Sensor. LM35 Sensor Specifications, 2019. Available in: < http://www.ti.com/product/LM35>. Access on 04/06/2019.

LoRaWAN TTN. The Things Network, LoRaWAN Overview, 2019. Available in: < https://www.thethingsnetwork.org/docs/lorawan/>. Access on 04/06/2019.

Mannaro, Katiuscia, *et al*. "Towards a smart region: The case study of a teledermatology platform in sardinian region (Italy)." *2017 IEEE 13th International*

Conference on Wireless and Mobile Computing, Networking and Communications (WiMob). IEEE, 2017.

Mattoni, Bendetta, *et al*. "A matrix approach to identify and choose efficient strategies to develop the Smart Campus." *Environment and Electrical Engineering (EEEIC), 2016 IEEE 16th International Conference on*. IEEE, 2016.

Medina, Camilo Alejandro, Manuel Ricardo Pérez, and Luis Carlos Trujillo. "IoT Paradigm into the Smart City Vision: A Survey." *Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2017 IEEE International Conference on.* IEEE, 2017.

Muhamad, Wardani, Novianto Budi Kurniawan, and Setiadi Yazid. "Smart campus features, technologies, and applications: A systematic literature review." *Information Technology Systems and Innovation (ICITSI), 2017 International Conference on*. IEEE, 2017.

Nam, T., & Pardo, T. A. (2011, June). Conceptualizing smart city with dimensions of technology, people, and institutions. In *Proceedings of the 12th annual international digital government research conference: digital government innovation in challenging times* (pp. 282-291). ACM.

Nour, B., Sharif, K., Li, F., Biswas, S., Moungla, H., Guizani, M., & Wang, Y. (2019). A survey of Internet of Things communication using ICN: A use case perspective. *Computer Communications*.

O'Brolchain, Niall & Ojo, Adegboyega & Porwol, Lukasz & Minton, David & Barry, Clodagh. “Examining the feasibility of a Smart Region approach in the North West Atlantic and Borders Region of Ireland”. *ICEGOV 2018*. 2018.

Pascolini, Donatella, and Silvio Paolo Mariotti. "Global estimates of visual impairment: 2010." *British Journal of Ophthalmology* 96.5 (2012): 614-618.

Pérez-delHoyo, Raquel, *et al*. "Making smart and accessible cities: An urban model based on the design of intelligent environments." *Smart Cities and Green ICT Systems (SMARTGREENS), 2016 5th International Conference on*. IEEE, 2016.

Pothuganti, K., & Chitneni, A. (2014). A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi. *Advance in Electronic and Electric Engineering*, *4*(6), 655-662.

Ray, P. P. (2018). A survey on Internet of Things architectures. *Journal of King Saud University-Computer and Information Sciences*, *30*(3), 291-319.

RM1xx Module. RM1xx Modules Specifications, 2019. Available in: <https://connectivity-staging.s3.us-east-2.amazonaws.com/2019-02/CS-DS-RM1xx%20v1_9.pdf>. Access on 04/06/2019.

Santana, Eduardo Felipe Zambom, *et al*. "Software platforms for smart cities: Concepts, requirements, challenges, and a unified reference architecture." *ACM Computing Surveys (CSUR)* 50.6 (2017): 78.

Silva, B. N., Khan, M., & Han, K. (2018). Towards sustainable smart cities: A review of trends, architectures, components, and open challenges in smart cities. *Sustainable Cities and Society*, *38*, 697-713.

Stapko, T. (2011). Practical embedded security: building secure resource-constrained systems. Elsevier. Pages 23-48.

Swaroopa, P. T., and Mrs Chaitra HK. "Internet of Things: Smart College." *International Conference on Computer Science and Technology Allies in Research-March.* 2016.

Tan, Pang-Ning, Michael Steinbach, and Vipin Kumar. "Data mining cluster analysis: basic concepts and algorithms." *Introduction to data mining* (2013): 487-533.

Tanda, Adriano, Alberto De Marco, and Marta Rosso. "Evaluating the Impact of Smart City Initiatives." (2017).

Tavares, J., Barbosa, J., Cardoso, I., Costa, C., Yamin, A., & Real, R. (2016). Hefestos: an intelligent system applied to ubiquitous accessibility. *Universal Access in the Information Society*, *15*(4), 589-607.

Techopedia. Long Term Evolution (LTE), 2019. Available in: <https://www.techopedia.com/definition/8149/long-term-evolution-lte>. Access on 05/06/2019.

Telles, Marcelo Josué, Jorge Luis Victória Barbosa, and Rodrigo da Rosa Righi. "Um Modelo Computacional para Acessibilidade em Cidades Inteligentes." *XII Brazilian Symposium on Information Systems.* 2016.

Torre, Ilaria, and Ilknur Celik. "A model for adaptive accessibility of everyday objects in smart cities." *Personal, Indoor, and Mobile Radio Communications (PIMRC), 2016 IEEE 27th Annual International Symposium on.* IEEE, 2016.

Torres-Sospedra, J., Avariento, J., Rambla, D., Montoliu, R., Casteleyn, S., Benedito-Bordonau, M., ... & Huerta, J. (2015). Enhancing integrated indoor/outdoor mobility in a smart campus. *International Journal of Geographical Information Science*, *29*(11), 1955-1968.

TTN. The Things Network, 2019. Available in: <https://www.thethingsnetwork.org/docs/network/>. Access on 14/06/2019.

Van Wee, B. (2016). Accessible accessibility research challenges. Journal of transport geography, 51, 9-16.

Vaščák, J., & Savko, I. (2018, August). Radio Beacons in Indoor Navigation. In *2018 World Symposium on Digital Intelligence for Systems and Machines (DISA)* (pp. 283-288). IEEE.

W3C. The World Wide Web Consortium, 2019. Available in: <https://www.w3.org/>. Access on 28/05/2019.

WAI. Web Accessibility Initiative, 2019. Available in: <https://www.w3.org/WAI/standards-guidelines/mobile/>. Access on 28/05/2019.

WCAG. Web Content Accessibility Guidelines, 2019. Available in: <https://www.w3.org/wai/standards-guidelines/wcag>. Access on 28/05/2019.

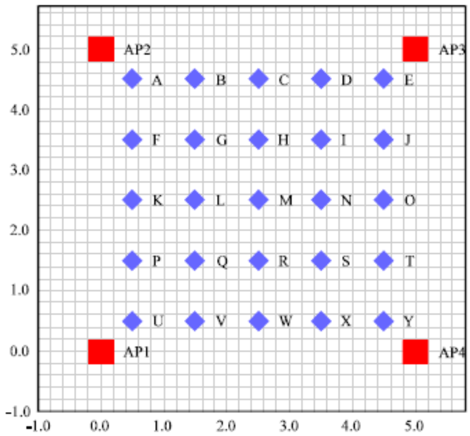
Xu, Rui, and Donald C. Wunsch. "Survey of clustering algorithms." (2005).

Zanella, A., Bui, N., Castellani, A., Vangelista, L., & Zorzi, M. (2014). Internet of things for smart cities. IEEE Internet of Things journal, 1(1), 22-32.

APPENDIX A – LOCALIZATION TECHNIQUES

The Fingerprinting technique needs "fingerprints" of the environment in order to determine the location of the user. To achieve this, it is necessary to record the locations measurements previously, creating a pattern that will latter, be used to compare the user's location with the known patterns (Vaščák *et al*., 2018). Figure 69 demonstrates an example of the Fingerprinting technique.

Figure 69 - Fingerprinting technique example.

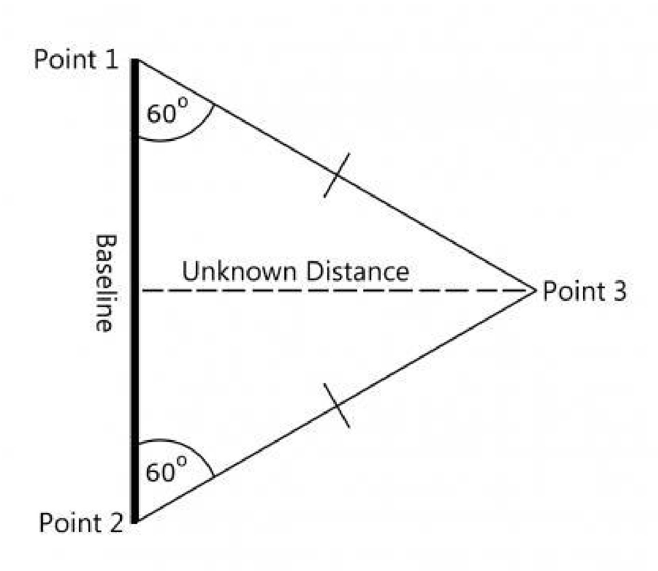


Source: https://scialert.net/fulltext/?doi=rjit.2012.155.165

This technique requires a lot of time in order to record the locations pattern in the environment, which depending on the environment size, it's not doable, which was the factor why this technique was not chosen to be applied in this study.

The Triangulation technique is similar to the trilateration technique, as shown in Figure 70.

Figure 70 - Triangulation technique example.



Source: https://gisgeography.com/trilateration-triangulation-gps/

The difference between them is that for triangulation is necessary to determine the location using trigonometric angles instead of distances (De Blas *et al*., 2017).