Smart Lab: An IoT-centric Approach for Indoor Environment Automation

João Gabriel Azevedo de Carvalho, Arielli Ajudarte da Conceição, Laura Pivoto Ambrósio, Fernando Fernandes Ramborger, Eduardo Henrique Teixeira, Guilherme Pedro Aquino and Evandro César Vilas Boas

***Abstract*—This work presents an Internet of Things (IoT)-centric approach to automation in indoor environments (e.g., laboratories, offices, and houses). The platform allows for Internet-connecting electronic and non-electronic devices for low-cost and fast automation control rather than replacing them as market solutions propose. The IoT solution comprises adaptive hardware modules, a mobile application, a database, and a Message Queuing Telemetry Transport (MQTT)-based broker server deployed in a real scenario for proof of concept. The mobile application functions manage the hardware modules to remote control lamps, air conditioners, TVs, and curtains. The commands are set through predefined touchscreen buttons or voice commands interpreted by a virtual assistant based on natural language processing (NLP) and natural language understanding (NLU) machine learning algorithms embedded in a framework called Wit.ai. The database handles user subscriptions by providing administrative functions. Finally, a real-time broker server manages the entities' communication based on the MQTT message exchange protocol, including a security layer based on the Transport Layer Security (TLS) protocol.**

***Index Terms*—Internet of Things, MQTT protocol, natural language processing, natural language understanding, smart environment, smart lab, TLS protocol, virtual assistant.**

I. Introduction

TECHNOLOGICAL advances boost the development of several standard communications that connect devices through heterogeneous telecommunication systems and networks. Accordingly, new services and applications emerge as solutions to support the digital transformation of different markets. Among these up-to-date technologies, the Internet of Things (IoT) is a suitable approach for Internet-connect daily live objects to automate processes and decision-making through data analytics in different scenarios [1]. Therefore, IoT solutions afford real-time data monitoring based on sensing capabilities while remotely controlling different processes. As a result, IoT technologies contribute to equipment manufacturers, Internet service providers, and application developers, yielding standard services and applications into innovative solutions to reduce costs and enhance performance. This plethora of applications and services is mainly classified as smart indoor environments (homes, laboratories, and offices), smart cities, smart farms, e-health, and Industry 4.0 solutions [1]–[4].

Smart cities IoT solutions improve citizens' quality of life by offering mobility solutions, public surveillance, and efficient resources (e.g., energy distribution, water treatment, selective waste collection, etc.). Hence, smart devices are strategically distributed over urban areas for sensing and monitoring or available based on mobile applications to supply the citizens' needs [2], [5]–[8]. As a result, these services and applications enhance the habitant's social well-being while enabling a circular economy and sustainable development. Likewise, e-health solutions boost service quality in hospital environments by providing accurate medical diagnoses and treatments, reducing costs, and increasing patient safety [9].

Besides, rural areas are equipped with sensors and actuators for crop and livestock monitoring, resulting in the automation of several processes and high productivity and quality rates. Indeed, some control factors are highlighted, such as climate data collection, crop growth monitoring, precise disease detection, reducing waste due to effective harvesting, herds behavior monitoring, and tracking along the life cycle [3], [10], [11]. Furthermore, Industry 4.0 IoT solutions have enabled production sites to collect and analyze data generated by connected objects and automate specific tasks, affording cost reduction with efficient and profitable production [4]. Connected sensors allow real-time accurate analysis and control of production factors for automatically handling changes in the industrial environment.

Regarding smart indoor environment scenarios, the IoT solutions comprise automating daily live housekeepers, remote control of environmental factors (e.g., temperature and lighting), and improving surveillance systems or access control [12]–[14]. Smart indoor environments are accomplished based on two main approaches: automation or IoT device-equipped places. Automated smart environments are realized by including smart process controllers at the building stage, i.e., the IoT devices are embedded in the architecture project, ranging from electrical, hydraulic, and safety systems to daily or personal applications. Meanwhile, the latter approach introduces IoT

João G. A. Carvalho, Arielli A. da Conceição, Laura P. Ambrósio, Fernando F. Ramborger, Eduardo H. Teixeira, Guilherme P. Aquino and Evandro C. Vilas Boas are with Cyber Security and Internet of Things Laboratory (CS&I Lab.) and Cyber Security Center (Centro de Segurança Cibernética do Inatel - CxSC Telecom), Departament of Telecommunication, National Institute of Telecommunications (Instituto Nacional de Telecomunicações - Inatel), João de Camargo Avenue 510 P.O. Box 05, 37540-000, Santa Rita do Sapucaí, MG, Brazil. (e-mails: joao.jg@ges.inatel.br, arielli.a@get.inatel.br, pivoto.laura@ges.inatel.br, fernandoramborger@gec.inatel.br, eduardot@gea.inatel.br, guilhermeaquino@inatel.br, evandro.cesar@inatel.br), ORCID: 0000-0002-7619-891X, 0000-0002-5510-7716, 0000-0003-3876-7672, 0000-0002-2221-1410, 0000-0001-8449-7541, 0000-0003-1734-5534, 0000-0002-7225-7783.

This work was financial support from the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG), and National Institute of Telecommunications (Instituto Nacional de Telecomunicações, Inatel).

Digital Object Identifier: 10.14209/jcis.2024.8

devices to render them smart in typical environments. Conventional electronic or electro-domestic systems lacking IP communication capabilities can be connected to the Internet through adaptive modules. This approach allows for a cost-effective and easy-to-deploy solution since an IoT framework is adopted to standardize the smart environment implementation.

Based on the alternative remote control devices approach, this work presents the development of a device management platform for indoor environment automation, including homes, educational places, and offices. The main goal is to provide an IoT-centric smart environment solution that automates manual processes by controlling systems that ensure human comfort. The proposal is applied to the Cyber Security and IoT Laboratory at the National Institute of Telecommunication (CS&I Lab.) due to the freedom in applying the approach, practically demonstrating its functionalities in rendering conventional environments into intelligent ones. Therefore, the services have been mapped as internal and external light control, temperature control, TV control, and laboratory maintenance requests. The real environment brings a comprehensive understanding of human interaction with the proposed architecture. Consequently, it allows for refining the implementation framework since the work aims for a user-centric application regarding functionalities. Furthermore, the alternative remote control approach offers a low-cost implementation since non-connected legacy electrical and electronic devices are turned smart using the hardware modules.

The proposed structure comprises a mobile application capable of controlling all devices, a broker server responsible for intermediating all data and control communication, a database that stores users' information, and the final device controllers, which are developed based on the NodeMCU microcontroller and circuits responsible for turning these devices on and off or switching them among different functions. The data exchange among these entities is handled by the Message Queuing Telemetry Transport (MQTT) protocol responsible for controlling information. In addition, the mobile application allows for setting commands based on predefined touchscreen buttons or through a virtual assistant using Natural Language Processing (NLP) and natural language understanding (NLU) machine learning algorithms model. Also, the data exchange is encapsulated by the Transport Layer Security (TLS) protocol to afford end-to-end secure communication.

The platform integrates different control functions, focusing on providing support for an IoT structure with control via a mobile application. In addition, the installed system does not interfere with the everyday use of these devices. The system allows for automatic control of these devices while keeping manual control of lighting, temperature, ventilation, and other devices that may come to integrate the IoT framework. Shortly, building a system that provides connectivity and control at a low cost and easy implementation with minimal environmental intervention, preserving the already laboratory functional structure, is experimentally demonstrated.

The paper's structure is organized as follows. Section II discusses the related works, highlighting our contribution compared to previous work. The IoT five-layer vertical-based architecture model is discussed in Section III and used as a design guideline to deploy the Smart Lab Platform. Section IV explores this IoT architecture and lists the enabling technologies and project functionalities aiming at implementing it in a laboratory. Section V addresses the system's deployment regarding hardware modules, a mobile application, a database, and a publish/subscribe server. In addition, the security layer implemented based on TLS protocol is analyzed to show the encrypted data exchange. Finally, Section VI concludes this work with some final considerations and outlines future research using the proposed framework.

II. Related Works

The IoT device landscape has grown significantly in recent years. Subsequently, this technology is easily found in residential scenarios based on several smart devices accessible and customizable by users according to their needs. Therefore, this section discusses some works with a similar objective to the proposal for designing and deploying IoT technologies to increase their control and connectivity in small and medium-sized scenarios. The related works can be classified into homemade-smart projects, market product solutions, and practical environmental integrated framework projects.

Concerning the market product solutions, the Echo Dot, Nest Mini, and Home Pod, among others, are voice-controlled personal assistants that offer various activities. In addition, these devices allow for integrating IoT solutions to achieve indoor environment control, such as intelligent lamps and plugs, air conditioner control, smart locks, and curtains [15]. These IoT device products can be controlled via a dedicated mobile application or virtual assistant based on voice commands. However, although these devices promote smart environments, they are sold individually with different user interfaces without customer installation support, resulting in low adhesion.

Additionally, homemade smart house projects are designed based on the many online tutorials and project examples supported by easy access to development hardware kits. Although this project brightens the concept of the indoor environmental application, it leaks from practical implementation. For instance, some mock-ups of an IoT concept were proposed to demonstrate an intelligent home control system [16]–[21]. These systems simplify human activities inside the house, such as turning on and off the lights, opening and closing the house door, car garage, and water tap, monitoring the state of the house, and maintaining home security. Indeed, an Android-based mobile application allows for remote control of an Arduino Uno or NodeMCU microcontrollers using WiFi, Bluetooth, Zig-Bee, GSM, and other protocols to communicate. The prototype tests have shown that any registered smartphone with the Android operating system can control domestic electronic devices attached to the proposed IoT platform. On the other hand, some prototypes were implemented based on innovative panels, whereas the electronic components and microcontrollers were joined for testing and demonstration purposes [22]–[29].

Regarding practical environmental integrated framework projects, an IoT-centric approach is used to leverage the

TABLE I
COMPARISON OF THE PROPOSED SOLUTION WITH PREVIOUS WORK.

Reference	This Work	[17]	[18]	[20]	[21]	[22]	[23]	[28]	[29]	[32]
Perception Capabilities	No	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Actuation Capabilities	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Microcontroler Hardware	ESP 8266	Arduino	ESP 8266 Xbee Raspberry	Arduino	Arduino	Arduino	Arduino	Wemos D1	ESP32 Arduino Raspberry	Raspberry Pi
Communication Protocol	WiFi	WiFi	GSM Bluetooth	WiFi, ZigBee	Bluetooth	WiFi	WiFi	WiFi	WiFi	WiFi
Database Technology	MySQL	MySQL	-	-	SQLite	SQLite	-	NoSQL	MySQL	MySQL
Messaging Protocol	MQTT	-	-	MQTT	-	-	-	HTTP MQTT	HTTP	MQTT
Automation-User-centric	U/C	U/C	U/C	A/C	A/C	A/C	A/C	A/C	A/C	A/C
User interface Type	Mobile	Mobile	Mobile	Mobile	Mobile	Mobile	Mobile	Mobile	Web	Web
NPL Capabilities	Yes	No	No	No	No	No	No	Yes	No	Yes
Multi-factor Authentication	Yes	No	No	No	No	No	No	No	Yes	No
Security Comm. Protocol	Yes	No	No	No	No	No	No	No	No	No
User data handling life cycle	Yes	No	No	No	No	No	No	No	No	No
Prototype (P) or Environment implemented (EI)	EI	P	P	P	P	P	P	P	EI	EI

hardware control based on a universal user interface. For instance, a case study was built as a framework that supports the connection of applications, servers, databases, devices, and APIs, capable of providing scalable IoT architecture [30]. The proposal used the MQTT protocol to exchange information and commands. Subsequently, the authors presented an infrastructure validation methodology regarding scalability, modularity, and interoperability. Moreover, IoT devices have been integrated into the scenario of a Smart Campus in a university [31]. The architecture has a storage system capable of recording indicators in real time. The data collected from the indicators encourage critical and continuous analysis, emergency action plans, and assertive decision-making. The Smart Campus has been operated for six years, going through different stages of implementation. The project began with defining the lines of action, followed by its implementation.

The authors in [33] created an improved learning and teaching environment through a research laboratory, where it was possible to carry out tests that simulated real conditions of intelligent environments. The architecture focused on processing data from different sensor sources in a correlated manner, providing statistical models. The Smart City Lab solution converts the processed data into automated systems. A practical demonstration was conducted in an experimental environment to collect data on air quality, waste management, lighting, and noise pollution. A gateway and a server with a database were implemented to allow applications to access information from these sensors. Lastly, a smart lab approach has been proposed based on sensors, Raspberry Pi, and a camera, with interactions through voice, text, and visual dashboards [32]. The project aimed to automate an environmental laboratory to support researchers during daily activities, including laboratory and equipment monitoring conditions and systems interaction.

Furthermore, the proposed Smart Lab platform has evolved from a previous design [14]. The initial version was integrated into a real environment for prototype evaluation. However, the virtual assistant and a security protocol layer were not initially implemented. Although voice commands were provided, the operation was based on predefined phrase commands without embedding machine learning algorithms. Besides, the mobile application was simple, without a means to manage user access through an administrator profile. Neither support to maintenance request. Concerning the initial test, the mobile application has been reprogrammed to include new functionalities, while some hardware modules have been redesigned and reinstalled. In addition, some security aspects and user data handling life cycles were added to elevate the platform's security.

Tab. I compares the proposed solution with previous work. The analysis shows that sensing the environment is a functionality most related to automatic-centric IoT solution, where the platform automatically controls the environment variable, and the collected data is only displayed to the user. On the other hand, user-centric solutions allow users to interact with the environment to set it at will, as these approaches are adopted in this work. Since smart environments are equipped with WiFi networks, the communication protocols of the proposed solutions are recurrent. The authors frequently employ mobile applications to facilitate user access, while web pages are

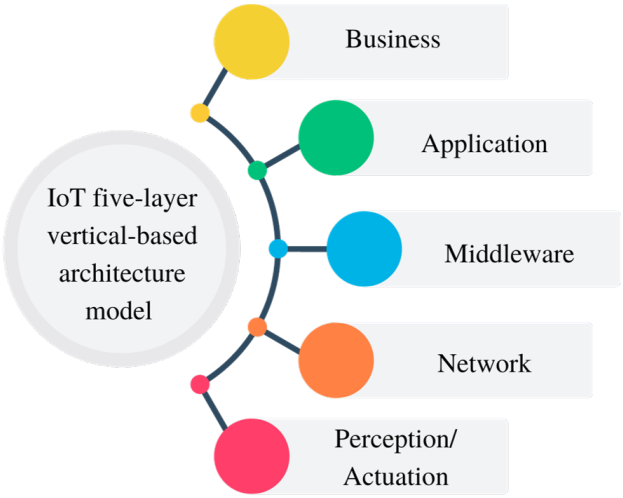


Fig. 1. IoT standardized five-layer vertical-based architecture model.

primarily for automatic-centric platforms. The voice assistant has been only implemented in recent works using market products such as Alexa or other NLP/NLU. The software development does not include security aspects like multi-factor authentication (MFA) and security protocol communication, which are included in the current work along with a user data handling life cycle (data registration, storage, and deletion). From a practical perspective, most works propose a prototype that limits the solution test, while our work has been environment implemented.

III. IoT Five-layer Vertical-based Architecture

Typically, IoT solutions comprise sensors and actuators integrated into a board and a microcontroller that communicates with a database and mobile/web application based on a communication protocol. Therefore, a standardized five-layer vertical-based architecture model provides a comprehensive overview of each proposed solution, as seen in Fig. 1 [1], [34], [35]. Concerning a bottom-up discussion, the perception/actuation layer comprises the sensors, actuators, hardware modules, and microcontrollers performing specific tasks. For instance, the sensors capture the different environments and human or animal stimuli and convert them into electrical signals digitized by the microcontrollers. On the other hand, the actuators alter the surroundings by changing them according to the microcontroller commands. Several sensors and actuators can be employed at this stage according to the IoT solution goals. Generally, the microcontrollers include computing process capabilities and are integrated into a board with other electronic components to allow easy manipulation and system integration.

The data collected by the microcontroller is sent to the upper layer to supply data analytics and driven decision-making. A suitable protocol is selected herein to afford the IoT communication constraints, such as bandwidth, coverage extension, quality of service (QoS), cost, energy consumption, battery life, etc [1]–[4]. Standard wireless technologies such as WiFi, mobile networks, and dedicated IoT communication protocols (e.g., Bluetooth, ZigBee, Z-Wave, Lora, and so forth) arise as candidates to Internet-connect the perception layer. The middleware layer follows by storing and processing the data to extract the information while including a server solution to handle multiple device accesses. A database established on available relational or non-relational solutions, such as MySQL and NoSQL, enables storing and managing data. Device communication might use different approaches, such as a client-server (e.g., HTTP protocol) or a publish/subscribe (e.g., MQTT protocol) architecture. The fourth layer focuses on the user by offering a friendly interface to access and display the IoT process information and/or allowing the user to interact with the application. Finally, the business layer comprises statistic analysis to create strategies for driven investment opportunities and future services and application design.

Concerning the above discussion, every IoT application does not fully implement this standardized IoT vertical-based architecture. Consequently, some IoT projects include specific layers to achieve their functionalities. For instance, several IoT-mobile-based applications eliminate the perception layer to afford a cost-effective solution [7], [36]. Meanwhile, other solutions exploit the environmental hardware to introduce essential services, such as ridesharing mobile applications [37], [38]. On the other hand, network and middleware layers are expected in many IoT projects due to their essential functions, such as connectivity, data storage, and management. The business layer has been restricted to IoT services and applications focusing on a market platform for profit rather than a user-centric approach. This work relies on this IoT architecture to build up a Smart Lab platform with a user-centric approach, which does not encompass a business layer.

IV. Smart Lab plataform

The Smart Lab platform comprises hardware control modules, a database, a publish/subscribe broker server, and a mobile application that follows the four-layer architecture presented in Fig. 2, which is based on the IoT standardized five-layer vertical-based architecture model presented in Fig. 1. The perception layer has actuators handled by microcontrollers to set the environment intervention. The actuators and microcontrollers are embedded in a hardware module solution according to their functions. Therefore, these modules control the air conditioner, television, and external and internal lighting. These capabilities were mapped according to the aimed environment, which different scenarios can expand. In addition, we have chosen only actuators to set the environmental conditions based on a user-centric approach, i.e., the user can interact with the environment to set its needs throughout the platform. Hence, sensors would not have practical application, gathering useless data. These aspects lead the proposal to fall into one of the representative use cases of IoT defined as remote control [see ITU-T Rec. Y.2066 [1]], which includes home automation, manufacturing, and intelligent transport systems (ITS). This definition stands for an IoT application that requires the capability of the user to control devices remotely.

[1]https://www.itu.int/rec/T-REC-Y.2066/en

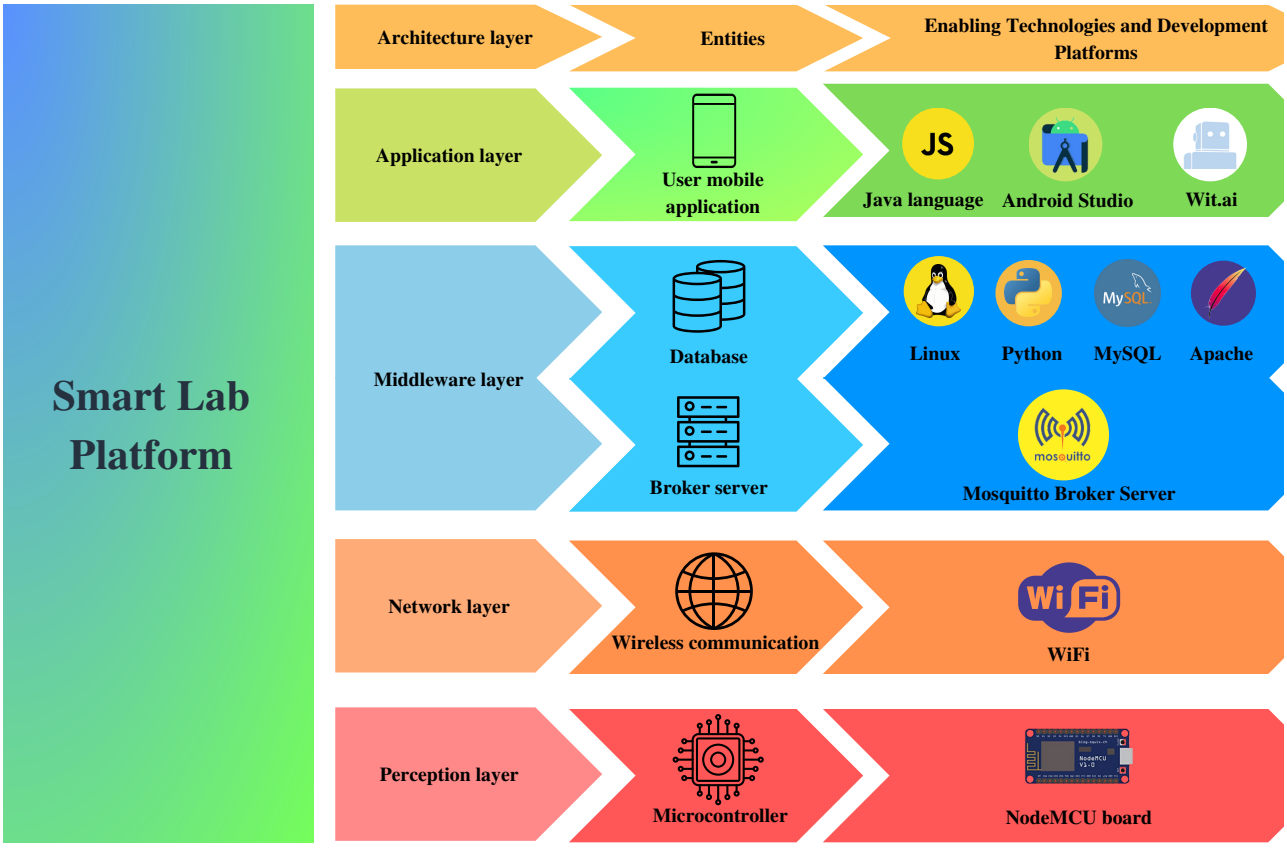


Fig. 2. Smart Lab platform architecture and enabling technologies.

Next, the network layer encompasses the communication protocol that enables the different entities to interact with each other. Consequently, the WiFi communication protocol has been selected since the smart platform aims at an indoor environment, which offers a pre-established WiFi network or means to deploy one efficiently. The middleware provides the resource for storing and processing the incoming data from the mobile application for multiple user access and administration functionalities based on the MySQL relational database. The application layer is represented by the mobile application that allows user interaction with the platform to control the environment. In addition, the commands sent by users throughout the mobile application are managed by the publish/subscribe broker server based on the MQTT protocol.

The Smart Lab platform architecture is built upon different enabling technologies. Also, it provides different functionalities comprising both hardware modules and mobile applications. Indeed, we discuss the enabling technologies and main functionalities before moving forward with the implementation itself.

A. Enabling technologies

The Smart Lab platform architecture is built upon different enabling technologies. The perception layer includes a microcontroller with a WiFi native connection afforded by the NodeMCU microcontroller. Therefore, the hardware module connection is easily managed by the NodeMCU based on a set of commands included in the WiFi library. The database relies on the open-source LAMP platform, which combines the Linux operational systems with the Apache web server, the MySQL database, and Python language. The mobile application is programmed using Java language through the Android Studio integrated development environment. Finally, the network layer applies the well-utilized WiFi communication protocol, standardized by the Institute of Electrical and Electronics Engineers (IEEE).

The publish/subscribe broker server operates through the MQTT protocol and acts as an information centralizer, managing the sending and receiving messages over the network [14], [25], [39], [40]. This protocol is built upon the TCP/IP stack layers, affording asynchronous communication among clients. Since it transmits only requested data, the MQTT protocol offers low processing, memory, and bandwidth consumption compared with other protocols, such as the Hypertext Transfer Protocol (HTTP). Its operation relies on the publish/subscribe architecture model. For instance, a client connected to the broker server can publish some information in a hierarchical topic structure. Further, this client can also subscribe to a topic of interest to access its information. The broker server manages the publishing and subscribing requests by receiving the topic messages and forwarding them only to the respective subscribed client. This communication model is accomplished

based on predefined messages between the broker server and the client. Hence, messages are devoted to connection, establishing, publishing, and subscribing, as discussed in [14].

Moreover, the communication process is encapsulated by the TLS protocol [41], [42]. It is implemented at the transport layer to encrypt the messages between clients and servers, offering confidentiality, integrity, and authentication in the communication process. Encryption algorithms accomplish the former, while integrity check codes guarantee integrity, mainly through Hash. The latter requires digital certificates exchanged based on an optional server-only or mutual authentication [41]. Furthermore, the TLS build up the security channel based on a handshake process encompassing predefined messages to exchange a set of parameters to afford confidentiality, integrity, and authentication. Hence, this protocol has been implemented into the entity's programming to support end-to-end security.

Finally, the mobile application also offers voice commands through a virtual assistant using NLP and NLU machine learning algorithms model embedded in a bot framework called Wit.ai. The NLP and NLU are computational techniques for analyzing and automatically representing and understanding human language. It allows computers to perform various tasks related to natural language at all levels, from syntactic analysis and classification of speech's parts to automatic translation and dialogue systems [43]. Henceforward, machine learning algorithms encompassing NLP and NLU arise, which is the case with the model used [44]. The Wit.ai platform used to implement the NLP and NLU functions supports about 50 languages. It allows the user to configure the machine learning model to recognize a series of instructions [45].

Concerning the application, the Wit.ai embedded machine learning algorithms are trained based on a specific environment and thus achieve better performance metrics. Even though the model has been trained for a specific scenario, new devices can be added, as well as new control functionalities. Once the training is finished, the model can analyze a speech and convert it into text. This text has been transformed into a structured data block, where entities will be searched [46]. In addition to entity analysis, the algorithms aim to find the parameters that will be controlled, looking for new values and states sent via voice command. Accordingly, this data is converted into commands by the application and sent to the devices.

B. Functionalities

The smart platform has been implemented to automate the identified process in the chosen scenario, the CS&I Laboratory at the National Institute of Telecommunication. Henceforth, the IoT solution provides the following functionalities that have been mapped based on users' needs:

- Internal light control
- External light control
- Temperature control
- TV control
- Voice command through a virtual assistant
- User management access through administration profile
- Laboratory maintenance request

The internal light control comprises turning the lamps on and off, while the external light entrance is controlled by setting the window's curtains up or down. The temperature control is related to the air conditioner configuration based on the equipment functionalities. Likewise, the TV is also controlled based on its native commands. These controls are centric on a mobile application based on touchscreen buttons or voice commands through a virtual assistant. Besides, the mobile application also introduces functionalities beyond environmental control. For example, it allows sending maintenance requests to the university responsible sector. The request is accomplished by allowing the user to fill out a quick request and send it directly to the service desk maintenance sector.

C. Smart Lab proposal and Recommendation ITU-T Y.2068

The Recommendation ITU-T Y.2068 [2] introduces the functional framework and capabilities of the IoT, which is based on the functional, implementation, and deployment view. Herein, we provide a discussion comparing our proposal with the IoT functional framework in functional view, as seen in Fig. 3. This framework describes the IoT capabilities at the functional level by defining seven groups: connectivity, communication, data management, service provision, application support, management, and security and privacy protection. The management, security, and privacy protection are multi-domain groups. The connectivity group is related to the device layer. It considers the capability of establishing connectivity between a thing and the other IoT entities, accomplished by the microcontroller (hardware modules side) and the smartphone (user side) WiFi connection in our proposal. The WiFi protocol allows the connectivity group to provide services to the data management and communication group while the management group deals with the maintenance of the link.

The communication group encompasses functional requirements for message exchange among the proposal IoT actors. Therefore, the project implements full-duplex communication from the user side, while the hardware modules communicate in a unidirectional way by receiving commands. The MQTT protocol also allows for different communication modes based on its management capabilities. The data management group benefits from the services the connectivity and communication groups offer to introduce data storage, processing, access control, exchange, and validation. These aspects are provided by the MySQL database alongside the Mosquitto Broker server. Regarding the service provision group, the Smart Lab platform is limited to service mobility aspects since the solution functionalities are defined at the project begins based on the environment implementation.

Besides, the Smart Lab platform includes functional requirements that meet the application support by considering group management (IoT users can be included or excluded from the platform through the administration user) and user management (including creation, authentication, authorization, and accounting of IoT users). Finally, the security and privacy protection relates to the project based on communication security using the TLS protocol to data exchange,

[2]https://www.itu.int/rec/T-REC-Y.4401/en

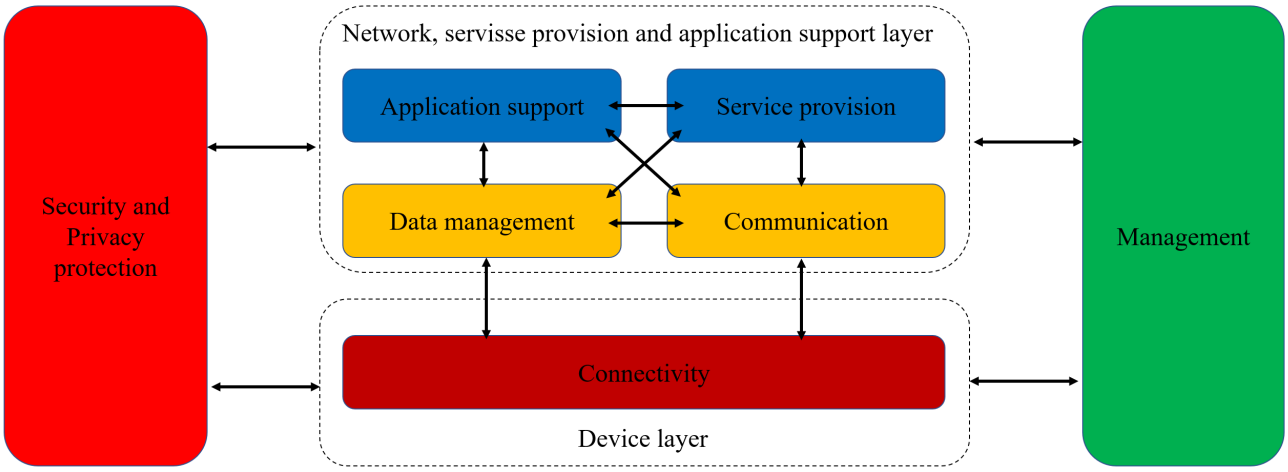


Fig. 3. The IoT functional framework in functional view proposed in Recommendation ITU-T Y.2068.

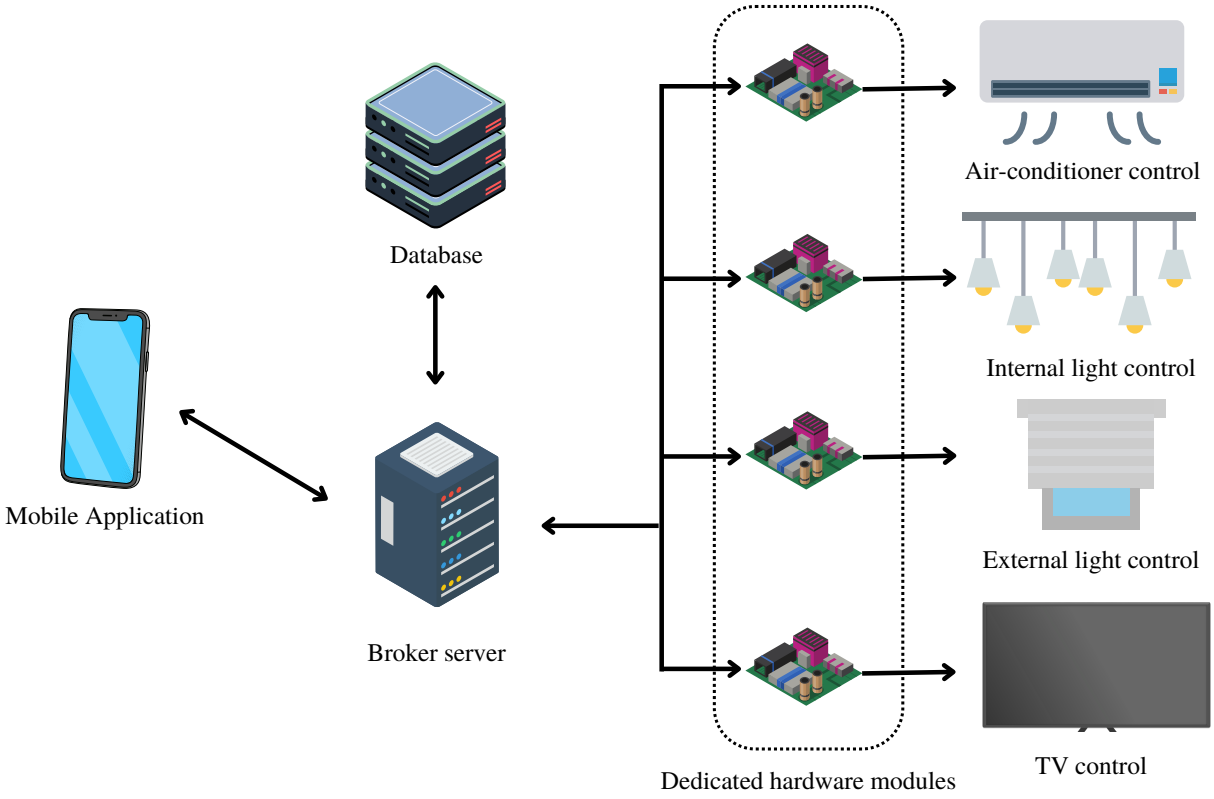


Fig. 4. Communication architecture between devices using the MQTT-based server Broker as an information centralizer.

data management security by managing the database access and prohibiting unauthorized access, and authentication and authorization schemes are included to verify the users are allowed to access the mobile application.

V. Smart Lab Implementation

The proposed IoT solution has been developed by integrating the enabling technologies discussed in Section III to offer the functionalities listed. Fig. 4 shows the communication architecture between devices using the MQTT broker server as a central element. Henceforth, we discuss the project implementation based on the architecture layer presented in Fig. 2. The proposed Smart Lab solution was deployed following some steps:

- Step 01: Map the environment and list the already placed electronics and non-electronics that could be remotely

Fig. 5. Hardware modules implementation in the laboratory environment.

controlled through a mobile application.

- Step 02: Define the commands that allow for remote control of each electronics and non-electronics in step 01, aiming at bringing the environment smart.
- Step 03: Map the commands into the server broker and functionalities for the mobile application.
- Step 04: Develop the hardware modules to connect the mapped electronics and non-electronics through the Internet. Integrate the microcontrollers with the server broker.
- Step 05: Develop the mobile application, define the database needs, and integrate the solution with the server broker, conducting tests to guarantee the integration.

A. Dedicated hardware modules implementation

The project aims to implement a low-cost solution to render the laboratory environment into a Smart Lab. However, objects and electronics can only communicate with the Internet by integrating resources that allow them to connect. Some devices already come with built-in technologies or external means that offer some way of acquiring these characteristics. The laboratory devices rely on the second group, demanding external hardware to connect them to the system via the Internet, as seen in Fig. 5. Therefore, it is enough to transform ordinary devices into intelligent ones using a device capable of providing the connection and processing essential for the IoT scenario, such as the NodeMCU microcontrollers. In addition, to make the non-invasive equipment implantation condition possible, the dedicated hardware modules were positioned in a power distribution board or close to the infrared devices, thus avoiding exchanging already installed devices or the outlets that supply energy for them.

Concerning the proposed IoT framework, the perception layer includes four hardware modules to control the air-conditioner, internal light, curtains (external light), and TV. These modules were installed into the laboratory infrastructure by adapting them to the environment and avoiding disturbing the architectural aesthetics, as seen in Fig. 5. The four hardware modules comprise a NodeMCU microcontroller as the key technology to bridge the perception and network layer of the proposed IoT architecture. This microcontroller offers low cost, reduced size, and low energy consumption, in addition to integrated support for WiFi networks. Henceforward, WiFi technology connects all devices to the smartphone application.

The air-conditioner and the TV are devices controlled through infrared signal emitters implemented into remote controls. Due to their functionalities, each control brings a range of buttons to set up these devices during their daily usage. Therefore, the hardware modules for Internet-connect and remote control through the mobile application have been designed based on a NodeMCU microcontroller with an infrared emission circuit, as seen in Fig. 5. The NodeMCU stores and processes the commands to allow control of the air conditioner and the TV. Consequently, an initial stage comprised mapping all the commands each device remote control offered using a photoreceptor model IR TSOP4838 and a NodeMCU to process the received sequences. These commands were labeled to implement the mobile control touchscreen buttons in the future and recorded in the NodeMCU memory. The infrared emission circuit comprises an infrared emitting diode and a resistor (220 Ω) responsible for converting the electric sequence commands sent by the NodeMCU in photoreceptor frequency-compatible infrared light.

The internal lighting control hardware module has been designed based on the laboratory infrastructure light modular system, which relies on eight switches to turn groups of light on and off individually. Hence, eight multichannel relays were installed parallel to the switches so that each channel could supply or interrupt energy flow in just one set of lamps. This modularization is adequate, as it corresponds to the real behavior of any environment, where users can turn the lights on/off in the most convenient way. Regarding the hardware module implementation, the relays were embedded in a board with a NodeMCU, as seen in Fig. 5. The 127 V electrical network is also connected to the board as the energy supply system for the lamps. At the same time, the relays and the NodeMCU are power supplied by 5 V and 3 V sources, respectively. The NodeMCU controls the relays based on the user commands sent throughout the mobile application. As discussed later, turning on/off groups of lamps or all of them is possible.

Eventually, the external light control is related to the curtains opening/closing management. This environmental aspect control was the most demanding adaptation due to the connectionless and non-electronic features. Subsequently, the curtains type and the installed place infer the control system implementation. The laboratory curtains are opened/closed using a string attached to an axis fixed on the wall above the windows. Accordingly, the proposed hardware module was designed based on two possible implementation processes: control of the string up and down or the axis rotation. These options aim to expand the solution to other laboratories with different curtains installed scenarios. The latter option was selected regarding the chosen laboratory, as seen in Fig. 5. The hardware module comprises a H-bridge circuit, a motor, and a NodeMCU. The microcontroller set up the H-bridge circuit to supply the 12 V motor and rotate it in the clockwise or counterclockwise direction. The opening and closing commands are positioned-related to accomplish complete closing, opening, and partial opening by rotating the curtain axis. Hence, a rotor was printed to adapt the motor axis to the curtain. The H-bridge circuit was designed based on two 1N4007 diodes as protect relays, two-pin connectors for the supply source, two relays to rotate the motor, resistors to limit currents, and two BC548 transistors as drivers.

B. Mobile application programming

The Smart Lab mobile application was developed for the Android operating system based on Java Script Language through the Android Studio Integrated Development Environment (IDE). The mobile application comprises a different process allowing laboratory users to access the IoT framework functionalities of the application. These procedures have been mapped out in Fig. 6.

The current Smart Lab software version has a user system that demands request registration. However, based on app security concerns, device control access is only allowed when new user information is verified and approved by the administrator. A new account demands the following data: name, user name, e-mail, and password. The system verifies this data in real time to validate a new user profile, avoiding multiple accounts with the same data. Hereafter, the new user must verify his e-mail using a two-factor authentication process, informing a token code sent to this e-mail address. The new user's registration success is recorded in the system and sent to the administrator profile for future evaluation. The approval implies that a notification message is sent to the registered e-mail. This process has been designed to prevent unauthorized users from accessing the IoT platform.

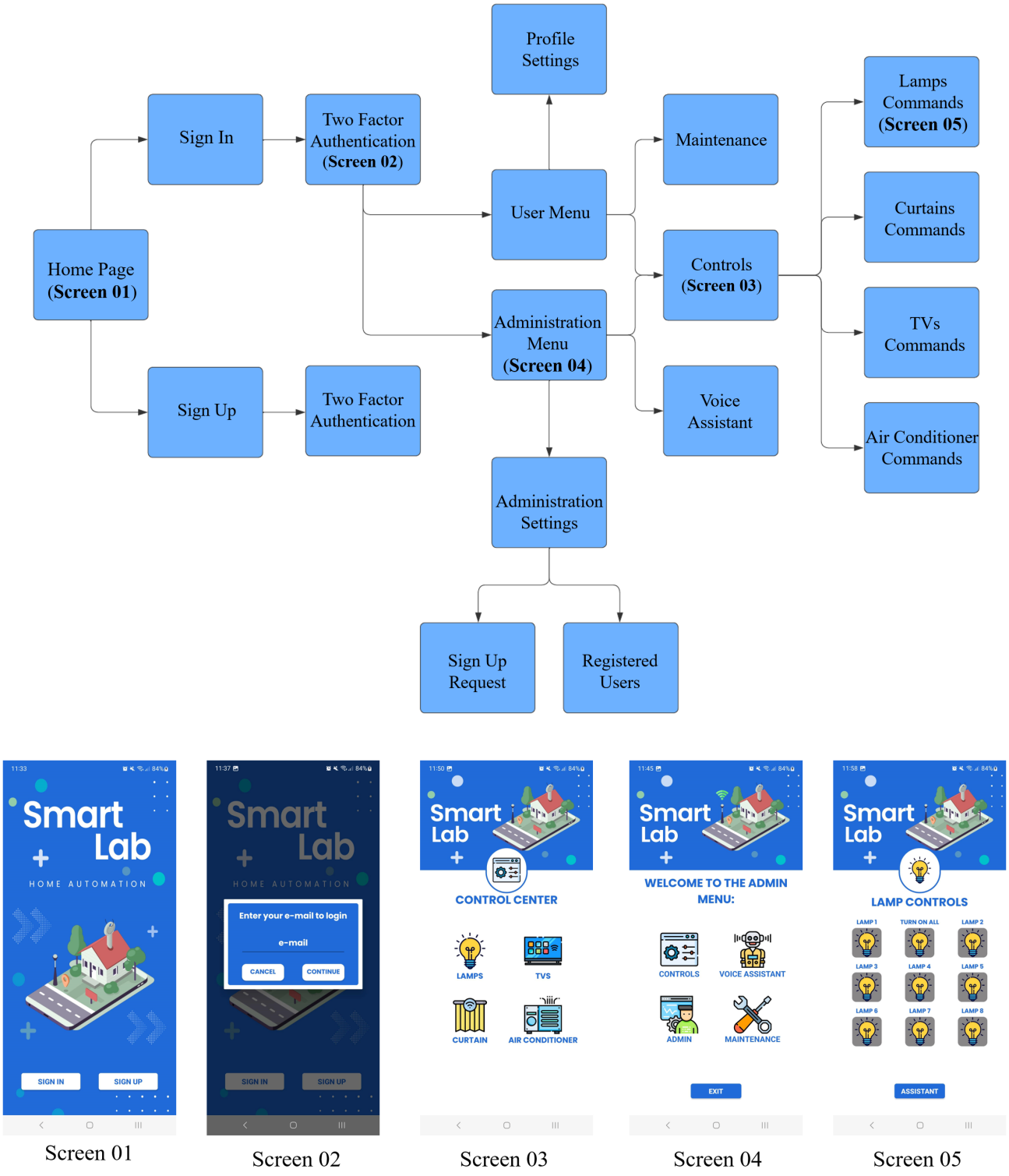


Fig. 6. Mobile application screens as the user interface, comprising the functionalities mapped out.

The registered users can access the mobile application functionalities based on a two-factor authentication login process: password and token code. Afterward, it accesses the main page menu, whereas the mobile application functionalities are disposed of through touchscreen buttons, as seen in Fig. 6. The first button gives access to the control panel, whereas a second group of buttons is labeled according to the hardware modules' control. By selecting one or another, the user can

access the command buttons for interior light, air-conditioner, TV, and curtains control.

Each subsequent screen has a shortcut button leading the user to the virtual assistant central that automatically captures voice commands to control the hardware modules. The virtual assistant also includes a button allowing users to access the command buttons directly. In the proposed structure, the user can use voice commands to control the lighting, air conditioning, ventilation, and TV. In some cases, in addition to controlling the on and off devices, it is also possible to control the intensity. This process uses artificial intelligence and machine learning models to process voice inputs and identify user commands. As a result, the virtual assistant creates a more intuitive experience based on users' voice interaction.

The third function offered by the main menu is the configuration area. Herein, the user can access and edit the account data, change authentication settings, and delete his profile account from the system. The latter procedure removes all user data from the database and logs the user out of the system, leading him to the first screen. The above procedures allow the mobile application to comply with the user data handling cycle, which encompasses the collecting, reception, utilization, access, distribution, process, storing, modification, and deletion. Finally, the users can fill out a maintenance request form when a repair on the laboratory infrastructure is needed. The form is linked to the e-mail address of the service desk maintenance and sent automatically by the IoT system. Since the lamps, air-conditioner, TV, and curtains are already mapped to be controlled, the form has a predefined message related to them. However, another kind of repair might arise during the daily activities, which were implemented by allowing reporting incidents by text.

In addition to the abovementioned functionalities, the mobile application has exclusive functions when connected to the administrator account. Accordingly, the administrator profile has access to registered users and new user access registration requests. Viewing the registered user, the administrator can remove access at any time by deleting the related account. This procedure has been implemented in case of misusage or when a collaborator is laid off from the laboratory. Furthermore, the request for registration approval guarantees that only laboratory students and collaborators are granted access to the system.

C. Broker server and database implementation

The database was implemented using the open-source LAMP platform to add storage capabilities to the proposed Smart Lab application. Hence, this entity comprises a unified user profile database that stores their data during registration. Furthermore, this database is used by the mobile application for user authentication by confirming their profile based on user name and password and confirming the token code. The database and the mobile application work with a label section, which affords multiple users access to the system without collision.

The broker server was locally implemented using the open-source message broker Mosquitto based on the MQTT protocol. Therefore, the publish/subscribe model allows the server broker to manage the device's exchange messages during the system operation. Regarding the proposed IoT framework, the mobile application works as a publisher and a subscriber during the implemented procedures. Likewise, the database also publishes some topics while subscribing to others during the register and login process, establishing a dialogue with the mobile application. Meanwhile, the hardware modules only subscribe to their related topic to receive the commands and further process them into environment intervention through the actuators.

D. Cyber security aspect

Devices connected to the Internet are always subject to attacks that aim to bypass security protections. Consequently, IoT applications are vulnerable to several cyber-attacks, including smart homes and other cloud-connected IoT devices. Most of these devices use MQTT protocol to communicate, which has limited authentication capabilities and does not support encryption unless a secure tunnel such as TLS is used to route traffic [47]–[49]. Thus, the current version of the MQTT-based broker server has the TLS encryption protocol, which increases the level of security in the communication between client and server. It uses a standard encrypted certificate between both, making it possible to only communicate and receive data with the server if the certificate is authentic and matches the server's [41], [42]. TLS encryption ensures that only the server can access the information sent over the network by encrypting the network traffic. In addition to this TLS security step, the application also has an authentication system via Token, requiring the insertion and validation of a set of seven digits sent to the user's email for steps that require security. Currently, the authentication steps are registration, login, and data change. Another essential feature is the user handle data cycle that guarantees that the user retains the right to modify or delete account details from the systems completely.

The Paho Java Client MQTT package was used to implement the TLS protocol during communication to provide a secure connection between the application, database, and broker. The MQTT connection was encapsulated in an Android-Service that runs in the background of the mobile application, switching between different activities. Using the TLS protocol, MQTT needs a certificate validated through the mobile device's chain of trust. Therefore, the paths for storing the certificate are explicit in the code during access to achieve a successful connection. Moreover, the WiFi Client Secure and MQTT libraries are used to implement security in the connections of the NodeMCU microcontrollers with the broker since they support the TLS protocol. Finally, the authentication certificate was installed in the microcontrollers. A ".h" file was included in the code, containing the leading TLS access parameters, such as the public key and fingerprint. For demonstration, the Wireshark tool was used to monitor the data traffic of a project's version without using the TLS protocol and a second one, which implements it. Fig. 7 (a) shows the information traffic in clear text, exposing the commands exchanged between the project entities. On the other

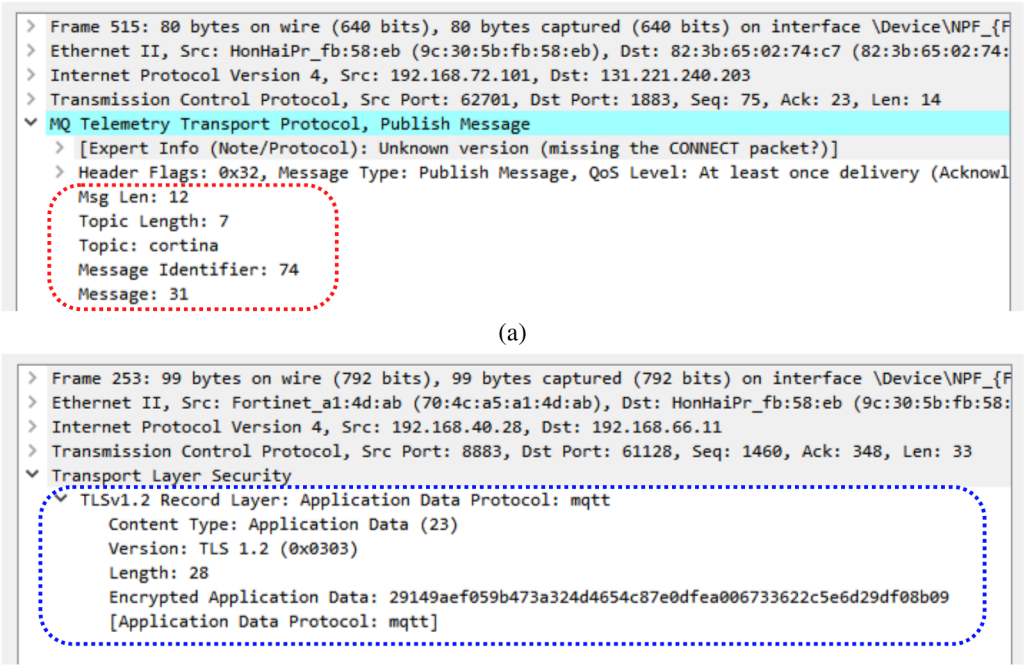


Fig. 7. Analysis of message traffic on the network (a) without using TLS protocol and (b) with TLS protocol.

hand, Fig. 7 (b) depicts the intercepted data traffic of the TLS protocol version in cipher text. Concerning the second scenario, the TLS protocol allows for establishing a secure end-to-end communication channel between entities and the broker server.

E. Technical and User Experience Discussion

The Smart Lab Platform is running and available for users integrating the CS&I Lab. Therefore, we conducted an informal technical and user experience discussion based on the approach in [32]. The former regards monitoring the Smart Lab functionalities for its first version [see [14]] deployed back in 2021 until the third version comprising this paper. During this time, the solution was improved based on user-mapped experience. Meanwhile, the user experience was informally assessed by free interviewing up to 09 users about their interaction with the platform during these versions' implementation. Between the first and third versions of the project, the CS&I Lab. was moved to a new facility, which was turned smart by applying the approach described in this work. This aspect proves the reuse of the proposed methodology in new environments. Fig. 8 compares the hardware-implemented at the first environment (first and second versions) and hardware modules and the new facility (third version).

The Smart Lab Platform was monitored to qualify tracking systems' potential operation problems. During the evaluation, the mobile application worked as expected, allowing the users to interact with the environment without service interruption. The hardware modules have been operated as designed, responding to the platform commands mentioned earlier. The WiFi connection handling routine was tested by turning the network off and on, proving the capability of the hardware modules to reconnect to the platform to process the commands automatically. Furthermore, we have found an architecture problem related to the local running of the Broker and database. This aspect makes the platform unavailable when the electric network interrupts the supply for a long time. Consequently, the machine that runs the MQTT-based server and database must be restarted to allow the platform to set up the communication. Future improvements regard implementing a cloud solution to run the database and server entities, avoiding maintenance under these scenarios.

Regarding the user experience, the report constraints are based on two main aspects: the MFA functionalities and the voice assistant. The MFA has been seen as an access stage delay that discourages users' frequent usage of the platform services. The user claimed accessing the email was laborious in getting the token code. This aspect faces the user's lack of awareness of cybersecurity mechanisms. However, the user experience can be improved by implementing fast Token code sending by allowing the user to configure the mechanism to send to the Token, such as email, SMS, or WhatsApp. Concerning the voice assistant, the typical issue includes misunderstanding some commands, which can be related to the training process. Furthermore, the voice assistant can be improved to provide a broad experience. Finally, the functionalities were not critical in the evaluation since we considered them in the design process.

VI. Conclusion and Future Work

This work presented the development of an IoT-centric platform solution to render typical indoor environments (homes, laboratories, offices, etc.) into smart ones. Thus, different enabling technologies have been integrated to leverage a

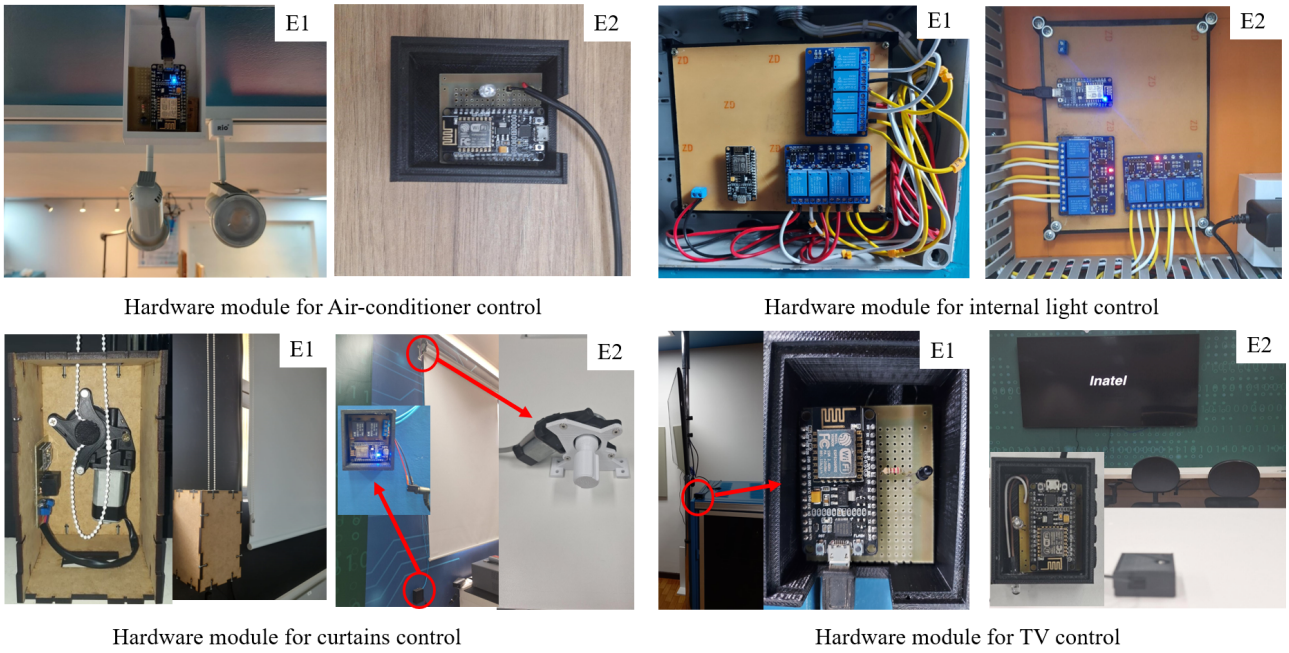


Fig. 8. Hardware modules deployed at first version (Environment 01, E1) and second version (Environment 02, E2).

practical real-scale IoT platform to control the Cyber Security and IoT Laboratory at the National Institute of Telecommunication. The system comprises dedicated hardware modules, a mobile application, a database, and a MQTT-based broker server. The hardware modules were designed on demand based on the environmental aspect aimed to be remotely controlled. Subsequently, these modules have provided non-invasive equipment and low-cost implementation by rendering non-connected electronic and ordinary furniture into Internet-connected devices with automated capabilities. Thus, different actuators were integrated with NodeMCU microcontrollers to bring connectivity and automation processes. On the other hand, a mobile application was programmed based on Java Script Language for Android operating systems to afford user interaction with the controllable installed hardware for setting up internal lights, air-conditioners, TVs, and curtains. Furthermore, the mobile application is backed by the database for user access control, while the broker server manages communication among these entities.

From an implementation perspective, the practical solution has shown an appropriate strategy to bring the IoT concept to different indoor environments with cost-effective solutions and a centric system for user operation simplicity. Evaluating this structure through a controlled environment allowed the ideas to be validated on a small scale. The next step would be to use this already functional infrastructure for tests with the expansion of these technologies, allowing the development of solutions more focused on Smart Cities. Future work regards exploit patterns and correlations between the input data of the system to implement a second control module based on automatic decision models. The proposal is to build a model trained from a specific pattern of user commands. For example, it can suggest a climate control, ventilation, and lighting configuration consistent with the user's historical pattern. This new implementation would further increase the application's intelligence level. Finally, the mobile application must be evaluated based on software engineering methodologies related to usability, such as SUS (System Usability Scale), SUMI (Software Usability Measurement Inventory), SUPR-Q (Standardized User Experience Percentile Rank Questionnaire) or QUIS (Questionnaire for User Interaction Satisfaction).

REFERENCES

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Commun. Surv.*, vol. 17, no. 4, pp. 2347–2376, 2015, doi: 10.1109/COMST.2015.2444095.

[2] A. Kirimtat, O. Krejcar, A. Kertesz, and M. F. Tasgetiren, "Future Trends and Current State of Smart City Concepts: A Survey," *IEEE Access*, vol. 8, pp. 86 448–86 467, 2020, doi: 10.1109/ACCESS.2020.2992441.

[3] M. S. Farooq, S. Riaz, A. Abid, K. Abid, and M. A. Naeem, "A Survey on the Role of IoT in Agriculture for the Implementation of Smart Farming," *IEEE Access*, vol. 7, pp. 156 237–156 271, 2019, doi: 10.1109/ACCESS.2019.2949703.

[4] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The Future of Industrial Communication: Automation Networks in the Era of the Internet of Things and Industry 4.0," *IEEE Ind. Electron. Mag.*, vol. 11, no. 1, pp. 17–27, 2017, doi: 10.1109/MIE.2017.2649104.

[5] R. Sánchez-Corcuera, A. Nuñez-Marcos, J. Sesma-Solance, A. Bilbao-Jayo, R. Mulero, U. Zulaika, G. Azkune, and A. Almeida, "Smart Cities Survey: Technologies, Application Domains and Challenges for the Cities of the Future," *Int. J. Distrib. Sens. Netw.*, vol. 15, no. 6, pp. 1–36, 2019, doi: 10.1177/1550147719853984.

[6] L. Ambrosio, P. L. S. Paulino, J. Antiquera, G. P. Aquino, and E. C. Vilas Boas, "EcoWaste: A Smart Waste Platform Enabling Circular Economy," in *2021 IEEE 19th Student Conference on Research and Development (SCOReD)*, 2021, pp. 411–415, doi: 10.1109/SCOReD53 546.2021.9 652 721.

[7] L. P. Ambrosio, E. C. d. C. Silva, G. P. Aquino, and E. C. Vilas Boas, "Recycling as a Service: A Mobile Application for Circular Economy," in *2022 IEEE International Conference on Computing (ICOCO)*, 2022, pp. 210–214, doi: 10.1109/ICOCO56 118.2022.10 031 977.

[8] J. G. Carvalho, D. M. Rosa, M. E. Camargo, L. F. Dias, G. P. Aquino, and E. C. V. Boas, "Vá de Bike: Plataforma IoT para Aluguel de Bicicletas," in *XLI Simpósio Brasileiro de Telecomunicações e Processamento de Sinais (SBrT 2023)*, 2023, pp. 01–05, doi: 10.14 209/sbrt.2023.1 570 916 139.

[9] M. H. da Fonseca, F. Kovaleski, C. T. Picinin, B. Pedroso, and P. Rubbo, "E-Health Practices and Technologies: A Systematic Review from 2014 To 2019," in *Healthcare*, vol. 9, no. 9. MDPI, 2021, pp. 1192, doi: 10.3390/healthcare9 091 192.

[10] G. Idoje, T. Dagiuklas, and M. Iqbal, "Survey for Smart Farming Technologies: Challenges and Issues," *Comput. Electr. Eng.*, vol. 92, p. 107104, 2021, doi: 10.1016/j.compeleceng.2021.107104.

[11] M. S. Farooq, S. Riaz, A. Abid, K. Abid, and M. A. Naeem, "A Survey on the Role of IoT in Agriculture for the Implementation of Smart Farming," *IEEE Access*, vol. 7, pp. 156 237–156 271, 2019, doi: 10.1109/ACCESS.2019.2949703.

[12] E. K. Hansen, "Home, Smart Home," *IEEE Spectr.*, vol. 47, no. 8, pp. 34–38, 2010, doi: 10.1109/MSPEC.2010.5520626.

[13] M. R. Alam, M. B. I. Reaz, and M. A. M. Ali, "A Review of Smart Homes—Past, Present, and Future," *IEEE Trans. Syst. Man Cybern., Part C*, vol. 42, no. 6, pp. 1190–1203, 2012, doi: 10.1109/TSMCC.2012.2189204.

[14] A. A. da Conceição, L. P. Ambrosio, T. R. Leme, A. C. S. Rosa, F. F. Ramborger, G. P. Aquino, and E. C. Vilas Boas, "Internet of Things Environment Automation: A Smart Lab Practical Approach," in *2022 2nd International Conference on Information Technology and Education (ICIT&E)*, 2022, pp. 01–06, doi: 10.1109/ICITE54 466.2022.9 759 899.

[15] J. Li and Y. Lin, "IoT Home Automation – Smart homes and Internet of Things," in *2021 3rd International Academic Exchange Conference on Science and Technology Innovation (IAECST)*, 2021, pp. 294–298, doi: 10.1109/IAECST54 258.2021.9 695 788.

[16] S. Sawidin, D. S. Pongoh, and A. A. S. Ramschie, "Design of Smart Home Control System Based on Android," in *2018 International Conference on Applied Science and Technology (iCAST)*, 2018, pp. 165–170, doi: 10.1109/iCAST1.2018.8 751 226.

[17] M. Fongbedji, N. Krami, and M. Bouya, "Mobile Application and Wi-Fi Modules for Smart Home Control," in *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, 2020, pp. 1–4, doi: 10.1109/ICECOCS50 124.2020.9 314 475.

[18] A. Shinde, S. Kanade, N. Jugale, A. Gurav, R. A. Vatti, and M. M. Patwardhan, "Smart Home Automation System using IR, Bluetooth, GSM and Android," in *2017 Fourth International Conference on Image Information Processing (ICIIP)*, 2017, pp. 1–6, doi: 10.1109/ICIIP.2017.8 313 770.

[19] Z. Mamatnabiyev and R. Suliyev, "Development of House Automation System Controlled Using IoT Technologies," in *2018 14th International Conference on Electronics Computer and Computation (ICECCO)*, 2018, pp. 206–212, doi: 10.1109/ICECCO.2018.8 634 734.

[20] I. Froiz-Míguez, T. M. Fernández-Caramés, P. Fraga-Lamas, and L. Castedo, "Design, Implementation and Practical Evaluation of an IoT Home Automation System for Fog Computing Applications based on MQTT and ZigBee-WiFi Sensor Nodes," *Sensors*, vol. 18, no. 8, p. 2660, 2018, doi: 10.3390/s18082660.

[21] P. U. Okorie, A. Abdu Ibraim, and D. Auwal, "Design and Implementation of an Arduino Based Smart Home," in *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, 2020, pp. 1–6, doi: 10.1109/HORA49 412.2020.9 152 922.

[22] P. S. Nagendra Reddy, K. T. Kumar Reddy, P. A. Kumar Reddy, G. N. Kodanda Ramaiah, and S. N. Kishor, "An IoT based Home Automation using Android Application," in *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, 2016, pp. 285–290, doi: 10.1109/SCOPES.2016.7 955 836.

[23] F. Alsuhaym, T. Al-Hadhrami, F. Saeed, and K. Awuson-David, "Toward Home Automation: An IoT Based Home Automation System Control and Security," in *2021 International Congress of Advanced Technology and Engineering (ICOTEN)*, 2021, pp. 1–11, doi: 10.1109/ICOTEN52 080.2021.9 493 464.

[24] S. Somani, P. Solunke, S. Oke, P. Medhi, and P. Laturkar, "IoT Based Smart Security and Home Automation," in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, 2018, pp. 1–4, doi: 10.1109/ICCUBEA.2018.8 697 610.

[25] R. K. Kodali and S. Soratkal, "MQTT based Home Automation System using ESP8266," in *2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, 2016, pp. 1–5, doi: 10.1109/R10–HTC.2016.7 906 845.

[26] M. Elkhodr, S. Shahrestani, and H. Cheung, "A smart home application based on the internet of things management platform," in *2015 IEEE International Conference on Data Science and Data Intensive Systems*, 2015, pp. 491–496, doi: 10.1109/DSDIS.2015.23.

[27] H. Heta Utari, "Ubiquitous smart home system using android application," *International Journal of Computer Networks & Communications*, vol. 6, no. 1, 2014, doi: 10.48550/arXiv.1402.2114.

[28] G. Alexakis, S. Panagiotakis, A. Fragkakis, E. Markakis, and K. Vassilakis, "Control of smart home operations using natural language processing, voice recognition and iot technologies in a multi-tier architecture," *Designs*, vol. 3, no. 3, p. 32, 2019, doi: 10.3390/designs3030032.

[29] L. Lagsaiar, I. Shahrour, A. Aljer, and A. Soulhi, "Modular software architecture for local smart building servers," *Sensors*, vol. 21, no. 17, p. 5810, 2021, doi: 10.3390/s21175810.

[30] O. B. Mora-Sánchez, E. López-Neri, E. J. Cedillo-Elias, E. Aceves-Martínez, and V. M. Larios, "Validation of IoT Infrastructure for the Construction of Smart Cities Solutions on Living Lab Platform," *IEEE Trans. Eng. Manag.*, vol. 68, no. 3, pp. 899–908, 2021, doi: 10.1109/TEM.2020.3002250.

[31] I. Negreiros, A. C. C. Francisco, F. H. Fengler, G. Faria, L. G. P. Pinto, M. Tolotto, R. B. Rogoschewski, R. R. Romano, and R. S. Netto, "Smart Campus® as a Living Lab on Sustainability Indicators Monitoring," in *2020 IEEE International Smart Cities Conference (ISC2)*, 2020, pp. 1–5, doi: 10.1109/ISC251 055.2020.9 239 017.

[32] N. J. Knight, S. Kanza, D. Cruickshank, W. S. Brocklesby, and J. G. Frey, "Talk2Lab: The Smart Lab of the Future," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8631–8640, 2020, doi: 10.1109/JIOT.2020.2995323.

[33] M. Kadar, "Smart Learning Environment for the Development of Smart City Applications," in *2016 IEEE 8th International Conference on Intelligent Systems (IS)*, 2016, pp. 59–64, doi: 10.1109/IS.2016.7 737 500.

[34] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A Survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, 2010, doi: 10.1016/j.comnet.2010.05.010.

[35] G. Fortino, C. Savaglio, C. E. Palau, J. S. de Puga, M. Ganzha, M. Paprzycki, M. Montesinos, A. Liotta, and M. Llop, "Towards multi-layer interoperability of heterogeneous iot platforms: The inter-iot approach," *Integration, Interconnection, and Interoperability of IoT Syst.*, pp. 199–232, 2018, doi: 10.1007/978-3-319-61300-010.

[36] A. Kamilaris and A. Pitsillides, "Mobile Phone Computing and the Internet of Things: A Survey," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 885–898, 2016, doi: 10.1109/JIOT.2016.2600569.

[37] A. K. Tripathy, P. K. Tripathy, A. G. Mohapatra, N. K. Ray, and S. P. Mohanty, "WeDoShare: A Ridesharing Framework in Transportation Cyber-Physical System for Sustainable Mobility in Smart Cities," *IEEE Consum. Electron. Mag.*, vol. 9, no. 4, pp. 41–48, 2020, doi: 10.1109/MCE.2020.2978373.

[38] S. A. Renu and B. G. Banik, "Implementation of a Secure Ridesharing DApp using Smart Contracts on Ethereum Blockchain," *Int. J. Saf. Secur. Eng.*, vol. 11, no. 2, pp. 167–173, 2021, doi: 10.18280/ijsse.110205.

[39] A. Rayes and S. Salem, *Internet of Things from Hype to Reality: The Road to Digitization*. Springer, 2019.

[40] B. Tripathy and J. Anuradha, *Internet of Things (IoT): Technologies, Applications, Challenges and Solutions*. CRC press, 2017.

[41] D. Díaz-Sánchez, A. Marín-Lopez, F. A. Mendoza, P. A. Cabarcos, and R. S. Sherratt, "TLS/PKI Challenges and Certificate Pinning Techniques for IoT and M2M Secure Communications," *IEEE Commun. Surv.*, vol. 21, no. 4, pp. 3502–3531, 2019, doi: 10.1109/COMST.2019.2914453.

[42] J. Li, R. Chen, J. Su, X. Huang, and X. Wang, "ME-TLS: Middlebox-Enhanced TLS for Internet-of-Things Devices," *IEEE Internet Things J.*, vol. 7, no. 2, pp. 1216–1229, 2020, doi: 10.1109/JIOT.2019.2953715.

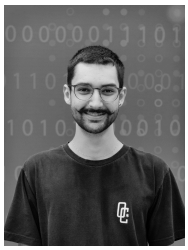
[43] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent Trends in Deep Learning Based Natural Language Processing," *IEEE Comput. Intell. Mag.*, vol. 13, no. 3, pp. 55–75, 2018, doi: 10.1109/MCI.2018.2840738.

[44] P. Danenas and T. Skersys, "Exploring Natural Language Processing in Model-To-Model Transformations," *IEEE Access*, vol. 10, pp. 116 942–116 958, 2022, doi: 10.1109/ACCESS.2022.3219455.

[45] A. A. Qaffas, "Improvement of Chatbots Semantics using Wit.ai and Word Sequence Kernel: Education Chatbot as a Case Study," *Int. J. Mod. Educ. Comput. Sci.*, vol. 11, no. 3, p. 16, 2019, doi: 10.5815/ijmecs.2019.03.03.

[46] E. Handoyo, M. Arfan, Y. A. A. Soetrisno, M. Somantri, A. Sofwan, and E. W. Sinuraya, "Ticketing Chatbot Service using Serverless NLP Technology," in *2018 5th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, 2018, pp. 325–330, doi: 10.1109/ICITACEE.2018.8 576 921.

[47] S. P. Mathews and R. R. Gondkar, "Protocol Recommendation for Message Encryption in MQTT," in *2019 International Conference on Data Science and Communication (IconDSC)*, 2019, pp. 1–5, doi: 10.1109/IconDSC.2019.8 817 043.
[48] M. Michaelides, C. Sengul, and P. Patras, "An Experimental Evaluation of MQTT Authentication and Authorization in IoT," in *Proceedings of the 15th ACM Workshop on Wireless Network Testbeds, Experimental evaluation & CHaracterization*, 2022, pp. 69–76, doi: 10.1145/3 477 086.3 480 838.
[49] M. A. Merzoug, A. Mostefaoui, G. Gianini, and E. Damiani, "Smart Connected Parking Lots based on Secured Multimedia IoT Devices," *Computing*, vol. 103, no. 6, pp. 1143–1164, 2021, doi: 10.1007/s00607-021-00921-1.



João Gabriel Azevedo de Carvalho was born in Santa Rita do Sapucaí, Minas Gerais, Brazil, in 2003. He is currently an undergraduate student in software engineering at the National Institute of Telecommunications (Inatel), Santa Rita do Sapucaí, Minas Gerais, Brazil. He has worked at the Cyber Security and Internet of Things Laboratory (CS&I Lab.)of Inatel since 2021 in IoT projects for smart homes, smart cities, and smart farms.

Arielli Ajudarte da Conceição was born in São Paulo, São Paulo, Brazil, in 2001. She received her electronics technician degree from the Escola Técnica de Eletrônica Sinha Moreira (ETE-FMC) in 2019. She is currently an undergraduate in telecommunication engineering at the National Institute of Telecommunications (Inatel), Santa Rita do Sapucaí, Minas Gerais, Brazil. She has worked at the Cyber Security and Internet of Things Laboratory (CS&I Lab.)of Inatel since 2019, designing and implementing IoT projects and CubeSats for university competitions. She also works at the Nouvenn company developing IoT market solutions.

Laura Pivoto Ambrósio was born in Santa Rita do Sapucaí, Minas Gerais, Brazil, in 2002. She received her electronics technician degree from the Escola Técnica de Eletronica Sinha Moreira (ETE-FMC) in 2019 and is currently an undergraduate in software engineering at the National Institute of Telecommunications (Inatel), Santa Rita do Sapucaí, Minas Gerais, Brazil. She has worked at the Cyber Security and Internet of Things Laboratory (CS&I Lab.)of Inatel since 2020, designing and implementing IoT projects for smart homes and smart city solutions.

Fernando Fernandes Ramborger was born in Três Corações, Minas Gerais, Brazil, in 2001. He is currently an undergraduate student in computer engineering at the National Institute of Telecommunications (Inatel), Santa Rita do Sapucaí, Minas Gerais, Brazil. He has worked at the Cyber Security and Internet of Things Laboratory (CS&I Lab.)of Inatel since 2020 in IoT projects for smart homes and smart cities.

Eduardo Henrique Teixeira was born in Caldas, MG, in 1996. He received a B.Sc. degree in Control and Automation Engineering from the National Institute of Telecommunications (Inatel), MG-Brazil, in 2018, and the M.Sc. degree in Telecommunications in 2021. He is currently working toward the Ph.D. degree in Telecommunications, also from Inatel. His research interests include Computer Vision, Artificial Intelligence, Internet of Things and Cybersecurity. Since 2016 he works at Inatel in disciplines such as Analog Electronics, Control of Dynamic Systems, Telemedicine Systems and Embedded Systems. He has already been a student in disciplines at the Stricto Sensu Graduate programs at the State University of Campinas (Unicamp) and at the Federal University of Itajubá (Unifei), both in the M.Sc and Ph.D program in Electrical Engineering. He is currently a member of the Brazilian Telecommunications Society (SBrT), where he has published papers, as well as in other National and International Congresses and Periodicals.

Guilherme Pedro Aquino has a Ph.D. in Electrical Engineering from the Federal University of Itajubá, UNIFEI. Master in telecommunications engineering from the National Telecommunications Institute (Inatel). He is currently a professor at Inatel in telecommunications networks and coordinator of Inatel's cybersecurity center. He worked as a researcher at the Radiocommunications Reference Center (CRR), researching non-orthogonal multiple access and efficient cooperative spectrum sensing strategies.

Evandro César Vilas Boas was born in Conceição das Pedras, Minas Gerais, Brazil, on 1993. He received the B.SC. (2016) and the M.Sc. degree (2019) in Telecommunication Engineering from the National Institute of Telecommunications (Inatel). He is currently pursuing a Ph.D. degree in Telecommunications at Inatel, researching flexible reconfigurable intelligent surfaces. He has joined Inatel as an Auxiliary Professor in the Telecommunications Engineering Course and teaches at Inatel Post Graduate Program. He also integrates the Telecommunication Coordinating Course, Cyber Security Center (CxSC Telecom), and the Cyber Security and Internet of Things Laboratory (CS&I Lab.). He supervises around 30 undergraduate students in research projects related to the Internet of Things (IoT), software-defined radio (SDR), artificial intelligence applied to Telecommunications, small satellites (CubeSat and CanSat), electromagnetism, and cyber security. He worked at the Inatel Competence Center (2017-2019) on R&D projects for designing passive radio frequency (RF) devices, such as antennas and waveguides. He was part of the research team at the Wireless and Optical Convergent Access Laboratory (2017-2022), where he developed research on antennas for the microwave and millimeter wave frequency range for 5G applications.