الجمهورية الجزائرية الديمقراطية الشعبية

People’s Democratic Republic of Algeria

وزارة التعليم العالي و البحث العلمي

Ministry of Higher Education and Scientific Research

جامعة الدكتور الطاهر مولاي – سعيدة -

University of SAIDA Dr.MOULAY TAHAR

Faculty of TECHNOLOGY

MEMORY

Research project presented for obtaining the MASTER'S degree

In: Telecommunications

Speciality: Networks and Telecommunications

By: **BENDIMIA Abdelkader**

Subject:

Design and simulation of an IoT network for a smart home using Cisco Packet Tracer

Publicly defended in 12th June 2023, in front of the jury composed of:

Mr. HARZELLAOUI Abdelkader	Univ. Saida	President
Mr. GUENDOUZ Mohamed	Univ. Saida	Rapporteur
Mr. CHETIOUI Mohammed	Univ. Saida	Examiner

Academic year 2022/2023

Acknowledgment

First and foremost, I would like to thank **ALLAH Almighty** who give me the strength and the courage to achieve this modest work.

I would like to thank my supervisor, Dr. GENDOUZ Mohamed, for their unwavering support, expertise, and invaluable guidance. Their insights and constructive feedback have been instrumental in shaping the direction of this project.

I would also like to extend my heartfelt thanks to the faculty members of University D.MOULAY Taher of SAIDA, faculty of Technology, for their mentorship and valuable contributions.

I am deeply indebted to my friends and classmates. Lastly, I would like to extend my deepest appreciation to my family for their unwavering support, understanding, and patience.

Thank you all once again for your invaluable contributions, guidance, and encouragement.

BENDIMIA Abdelkade

Dedication

I dedicate this project of study to all the individuals who have inspired, supported, and influenced me along this academic journey.

To my parents, to my family, for their patience, understanding, and continuous support throughout this endeavor.

To my supervisor, whose guidance, expertise, and unwavering commitment to my development have been invaluable.

To my friends and classmates, for their camaraderie, intellectual discussions, and unwavering support. Your presence has made this academic journey more enjoyable, and I am grateful for the memories we have shared.

To the faculty members and mentors, for their dedication, knowledge, and guidance.

This project is dedicated to you, as a testament to the power of collaboration, mentorship, and the unwavering support of loved ones. Your impact on my life and academic pursuits will forever be cherished.

Thank you for being a part of this journey.

BENDIMIA Abdelkader

Abstract

The IoT-based Smart Home system is an innovative and integrated solution aimed at improving the quality of life. This system utilizes IoT applications to provide a user-friendly home automation platform, enabling seamless control and operation.

This project serves as a comprehensive demonstration of the system's capabilities, showcasing its versatility in areas such as home security, lighting control, flame detection, smart heating, motion sensor and door control, among others. By incorporating these features, the system ensures homeowners' comfort, security, energy efficiency, and convenience. Users can effortlessly monitor and control household appliances, conveniently turning devices on or off as required.

The project implements a smart home concept by integrating sensors, actuators, and a Wi-Fi communication protocol connected to a Gateway. This connectivity enables remote control of the home through a 3G/4G client using an IoT server account. The system's functionality was simulated and tested using Cisco Packet Tracer, a reliable simulation tool.

In summary, this project represents the convergence of IoT, networking, smart home automation, and Cisco Packet Tracer simulation. It provides individuals with a robust framework to create an intelligent, connected, and efficient living environment.

The keywords associated with this study are: Internet of Things, Networking, Smart Home, Automation, and Cisco Packet Tracer.

الملخص :

يعد نظام المنزل الذكي القائم على إنترنت الأشياء حلاً مبتكرًا ومتكاملًا يهدف إلى تحسين جودة الحياة. يستخدم هذا النظام تطبيقات إنترنت الأشياء لتوفير منصة أتمتة منزلية سهلة الاستخدام ، مما يتيح التحكم والتشغيل السلس.

يعتبر هذا المشروع بمثابة عرض شامل لقدرات النظام ، حيث يعرض تنوعه في مجالات مثل أمن المنزل ، والتحكم في الإضاءة ، واكتشاف اللهب ، والتدفئة الذكية ، ومستشعر الحركة ، والتحكم في الباب ، من بين أمور أخرى. من خلال دمج هذه الميزات ، يضمن النظام راحة أصحاب المنازل والأمان وكفاءة الطاقة والراحة. يمكن للمستخدمين مراقبة الأجهزة المنزلية والتحكم فيها بسهولة ، وتشغيل أو إيقاف تشغيل الأجهزة بسهولة كما هو مطلوب.

يطبق المشروع مفهوم المنزل الذكي من خلال دمج أجهزة الاستشعار والمشغلات وبروتوكول اتصال Wi-Fi المتصل ببوابة. يتيح هذا الاتصال التحكم عن بعد في المنزل من خلال عميل3G/4G باستخدام حساب خادم IoT تمت محاكاة وظائف النظام واختبارها باستخدام Cisco Packet Tracer ، وهي أداة محاكاة موثوقة.

باختصار ، يمثل هذا المشروع التقارب بين إنترنت الأشياء والشبكات وأتمتة المنزل الذكي ومحاكاة Cisco Packet Tracer. يوفر للأفراد إطار عمل قويًا لإنشاء بيئة معيشية ذكية ومتصلة وفعالة.

الكلمات الرئيسية المرتبطة بهذه الدراسة هي: إنترنت الأشياء ، والشبكات ، والمنزل الذكي ، والأتمتة ، و Cisco Packet Tracer.

Résumé

Le système Smart Home basé sur l'IoT est une solution innovante et intégrée visant à améliorer la qualité de vie. Ce système utilise des applications IoT pour fournir une plate-forme domotique conviviale, permettant un contrôle et un fonctionnement transparents.

Ce projet sert de démonstration complète des capacités du système, mettant en valeur sa polyvalence dans des domaines tels que la sécurité domestique, le contrôle de l'éclairage, la détection de flamme, le chauffage intelligent, le détecteur de mouvement et le contrôle des portes, entre autres. En incorporant ces caractéristiques, le système assure le confort, la sécurité, l'efficacité énergétique et la commodité des propriétaires. Les utilisateurs peuvent surveiller et contrôler sans effort les appareils électroménagers, en allumant ou en éteignant facilement les appareils selon les besoins.

Le projet met en œuvre un concept de maison intelligente en intégrant des capteurs, des actionneurs et un protocole de communication Wi-Fi connectés à une passerelle. Cette connectivité permet le contrôle à distance de la maison via un client 3G/4G utilisant un compte de serveur IoT. La fonctionnalité du système a été simulée et testée à l'aide de Cisco Packet Tracer, un outil de simulation fiable.

En résumé, ce projet représente la convergence de l'IoT, de la mise en réseau, de la domotique intelligente et de la simulation Cisco Packet Tracer. Il fournit aux individus un cadre robuste pour créer un environnement de vie intelligent, connecté et efficace.

Les mots-clés associés à cette étude sont : Internet des objets, réseautage, maison intelligente, automatisation et Cisco Packet Tracer.

Table of contents

CHAPTER 2: SMART HOME

CHAPTER 3: NETWORK DESIGN

CHAPTER 4: SIMULATION AND RESULTS

List of figures:

Liste of tables:

Abbreviations:

3GPP™	Third Generation Partnership Project
6LoWPAN	IPv6 over a low-power personal area network
A	
ACL	Asynchronous Connection-Less
AFH	Adaptive Frequency Hopping
ALG	application layer gateway
AMP	Alternative MAC/PHY
AMQP	Advanced Message Queuing Protocol
AoA	Angle of Arrival
AoD	Angle of Departure
API	Application Programming Interface
AWS	Amazon Web Services
B	
BLE	Bluetooth Low Energy
BPSK	binary phase shift key modulation
BR/EDR	Bluetooth Basic Rate and Enhanced Data Rate
C	
CLI	command line interface
CoAP	Constrained Application Layer Protocol
CoRE	Constrained RESTful Environments
CSMA/CA	Carrier sense multiple access/collision avoidance
DHCP	Dynamic Host Configuration Protocol
D	
DIY	Do-It-Yourself
DL	Downlink
DNS	Domain Name System
DTLS	Datagram Transport Layer Security
E	
EDR	Bluetooth Enhanced Data Rate
ETSI	The European Telecommunications Standards Institute
F	
FFD	Full Function Devices
FHSS	frequency hopping spread spectrum
G	
GFSK	Gaussian Frequency Shift Keying
gpio	General Purpose Input/Output
GPRS	General Packet Radio Services
GSM	Global System for Mobile
GUI	Graphical User Interface
H	
HS	Bluetooth High Speed
HTTP	Hypertext Transfer Protocol
I	
IaaS	Infrastructure as a Service
IaaS	Infrastructure as a Service
ICT	Information and Communication Technology
IEEE	Institute of Electrical and Electronics Engineers
IEEE-SA	Institute of Electrical and Electronics Engineers-Standards Association
IETF	The Internet Engineering Task Force
IFTTT	If This Then That
IIoT	the Industrial Internet of Things
IoT	Internet of Things
IP	Internet Protocol
IPSO	The Internet Protocol for Smart Objects
IPSO	Internet Protocol for Smart Object
IPv6	Internet Protocol Version 6
IPv4	Internet Protocol Version 4
ISA	International Society of Automation
ISM	Industrial, Scientific and Medical
ISP	Internet Service Provider
L	
LAN	local area network
LoRaWAN	Long Range Wide Area Network
LPWAN	Low Power Wide Area Network
LR-WPAN	Low-Rate Wireless Personal Area Network
LTE	Long-Term Evolution
M	
MAC	Media Access Control

MCU	Microcontrollers
MCU	Microcontroller Unit
MIT	Massachusetts Institute of Technology
MQTT	Message queue telemetry transport
N	
NaaS	Networks as a Service
NIC	network interface card
O	
OQPSK	offset quadrature phase shift keying modulation
OSI	Open Systems Interconnection
P	
PaaS	Platform as a Service
PAN	personal area network
PC	personal computer
Q	
QoS	Quality of Service
R	
RFC	the Request for Comments
RFD	Reduced Function Devices
RFID	Radio Frequency Identification
RFID	Radio Frequency Identification
ROLL	Routing Over Low energy and Lossy networks
S	
SaaS	Software as a Service
SaaS	Software as a Service
SCO	Synchronous Connection-Oriented
SDK	Software Development Kit
SF	spread factor
SIG	Bluetooth Special Interest Group
SP	The service provider
SSID	Service Set Identifier
U	
UDP	User Datagram Protocol
UL	uplink
UMTS	Universal Mobile Telecommunications Service
URL	Uniform Resource Locator
UWB	Ultra-Wide Band
W	
WAN	Wide Area Network
WBAN	Wireless Body Area Networks
WSN	wireless sensor networks
X	
XMPP	Extensible Messaging and Presence Protocol

General introduction

The advancement of technology has greatly influenced the development of the Internet of Things (IoT) and its impact on our lives. The IoT, which involves connecting devices and objects through the internet to collect and exchange data, has been made possible by technological advancements such as wireless communication, sensor miniaturization, and increased computing power. This symbiotic relationship between technology and the IoT has paved the way for transformative applications like the smart home. A smart home is a residence equipped with internet-connected devices and sensors that enable automation and remote control of various household functions. These devices, ranging from appliances to security systems, have revolutionized how we interact with our homes, enhancing convenience, energy efficiency, and safety. However, this raises questions about the implications of interconnected devices and the technologies used, as well as the extent to which smart homes truly improve our daily lives.

The emergence of the IoT and smart homes prompts questions about the technology behind them. How do wireless communication, sensor miniaturization, and increased computing power contribute to the growth of the IoT? What are the potential risks and challenges associated with the interconnection of devices and the exchange of data? Furthermore, the concept of the smart home raises inquiries about its impact on our daily lives. How do smart thermostats, security systems, and lighting systems truly enhance our comfort, safety, and energy efficiency? Are there limitations or drawbacks to relying on automated and remote-controlled devices in our homes? Exploring these questions will shed light on the intricacies and implications of the IoT and smart homes, helping us navigate the rapidly evolving technological landscape and make informed decisions about their integration into our lives.

In this study, titled "The Design and Simulation of an IoT Network for a Smart Home using Cisco Packet Tracer," we aim to explore the complexity of creating an efficient and robust IoT network for a smart home.

This study is divided into four chapters, each focusing on specific aspects of the IoT network design and simulation for a smart home. In the first chapter, we explore the core elements of IoT, including its history, architecture, standards, protocols, and applications, providing a vital understanding for designing an efficient smart home network

In the second chapter, we delve into the intricacies of smart homes, covering their definition, historical progression, technological advancements, and architectural considerations that influence their functionality. This exploration provides valuable insights into the evolution of smart homes and the possibilities for future innovations.

In the third chapter, we shift our attention to designing the network infrastructure for a smart home, emphasizing the vital requirements and considerations for seamlessly connecting devices and sensors. We conduct a thorough analysis of the devices and components involved, clarifying their roles in establishing a cohesive and interconnected network.

In the fourth chapter, we share the outcomes of the network test conducted on the IoT network designed for the smart home, revealing insights into its reliability and effectiveness. These findings validate the design approach and serve as a foundation for future enhancements and optimizations.

In conclusion, this study aims to explore the design and simulation of an IoT network for a smart home using Cisco Packet Tracer. Through a comprehensive examination of the IoT and smart home concepts, network design considerations, and device functionalities, we seek to contribute to the advancements in smart home technology. By evaluating the performance of the designed network, we aim to provide insights into its effectiveness and uncover potential areas for enhancement. The findings of this study hold significant implications for the future of smart homes, paving the way for smarter, more efficient, and interconnected living environments.

CHAPTER 1

STATE OF ART ON IOT

Introduction

The Internet of Things (IoT) is one of the cornerstones of the fourth industrial revolution, as it represents the highest level of integration between humans and the devices and machines that surround them. The Internet of Things envisions a world where billions of objects with built-in intelligence, communication media, and sensing and actuation capabilities are connected over an Internet Protocol (IP) network. Our current Internet has undergone a fundamental shift from a hardware-driven (computer, fiber optics, and Ethernet cable) to a market-driven capability (Facebook, Amazon).

In this chapter we will briefly discuss about what IOT is, about its architecture, how IOT enables different protocols and applications.

1. Definitions:

The IoT has been defined by different authors in many different ways. Let us look at some of the most popular definitions:

- The IoT is divided on two words: Internet and things, connecting things to the current Internet and product of information via the interaction between the physical world and the virtual are the principal idea of the IoT [1].
- Making everything like cities, roads, appliances, cars and machines connected to the internet through intelligent behavior keeping in mind the autonomy and privacy that is what the Internet of Things is all about. The main concept of the Internet of Things is to enhance communication between anything from anywhere at any time [2].
- In simpler terms, the IoT describes billions of physical devices around the world that are connected to the Internet thanks to the ubiquitous spread of wireless networks. Adding sensors and processors to these devices adds a level of digital intelligence, enabling them to communicate and share data with other devices and systems over the Internet without the need for human-computer interaction [2].
- IoT is transforming the everyday physical objects in the surrounding environment into ecosystem information that enrich people’s lives. IoT is bridging the gap between the physical and the digital or virtual worlds, facilitating the convergence of advances in miniaturization, wireless connectivity, increased data-storage capacity and batteries. IoT is a set of key enabling technologies for digital businesses and one of the main drivers contributing to transforming the Internet and improving decision-making capacity via its augmented intelligence [3].

- The IoT is the ability to transfer data between a system of interconnected computing devices, objects, people or animals with a unique identifier without a human-to-human interaction communication is required or computer interaction. IoT refers to a type of network to connect anything with the Internet based on stipulated protocols through information sensing equipment's to conduct information exchange and communications in order to achieve smart recognitions, positioning, tracing, monitoring, and administration [4].

These definitions can be combined to say that the Internet of Things is the network of devices (machines, animals, human devices, household appliances, cities ...) that allows data to be connected and exchanged with other things over an IP network and certain protocols.

Figure 1: Internet of Things

2. History:

The concept of networked devices themselves dates back to 1832 when the first electromagnetic telegraph was developed [24]. The telegraph enabled direct communication between two machines by transmitting electrical signals. However, the actual history of the Internet of Things began with the invention of the Internet by the ARPANET[5] in the late 1960s, which then developed rapidly over the next few decades.

The 1980s: the first idea of IoT came with Coca-Cola vending machines at Carnegie Melon University, students in the computer science department installing microswitches in the vending machines to see if the coolers are keeping the drinks cold enough and if the Coke cans are available are in the machine in the internet.[6]

The 1990s: In 1990 John Romkey created the first Internet ‘device’, a toaster that could be turned on and off over the Internet.[7] The year 1999 was easily one of the most significant for the IoT history as the IoT concept was coined by Kevin Ashton [8] executive director of the Auto-IDCentre, MIT (Massachusetts Institute of Technology). They also invented a global RFID (Radio Frequency Identification)-based item identification system in the same year.[7]

"I could be wrong, but I'm fairly sure the phrase "Internet of Things" started life as the title of a presentation I made at Procter & Gamble (P&G) in 1999. Linking the new idea of RFID in P&G's supply chain to the then-red-hot topic of the Internet was more than just a good way to get executive attention. It summed up an important insight which is still often misunderstood." Ashton (2009). [9-10]

The 2000s: At the beginning of the 21st century, the term “internet of things” came into widespread use by the media.In 2000, a smart refrigerator announced by electronics giant LG can define itself if whether or not food items stocked in it are recharged [7]. The year 2003 witnessed the spread of RFID all over the world by Walmart in all its stores. In 2008, IPSO (The Internet Protocol for Smart Objects) alliance was founded by a group of companies (members from technology, communications and energy companies). The alliance aims to promote the use of IP (Internet Protocol) in networks of smart objects and to enable the IoT [7-13].

The 2010s: Connected devices have become widespread and common in our daily lives, thanks to the launch of IPv6 during the year 2011, which led to tremendous growth and great interests in this field[7]. In this period, Global tech giants like Apple, Samsung, Google, Cisco, and General Motors focused their efforts on the production of IoT sensors and devices-from interconnected thermostats and smart glasses to self-driving cars.

3. Architectures :

The Internet of Things is able to connect billions or trillion from heterogeneous things via the Internet, for this reason the need for a flexible layer structure comes. However, despite this need, there is no consensus on one structure and the increasing number of proposed structures is not close to a reference model [11]. Different architectures have been proposed by different researchers [12].

3.1 Three- and Five-Layer Architecture:

The basic architecture of IoT as identified by researchers is Three-layers [11-12-13] as shown in figure.

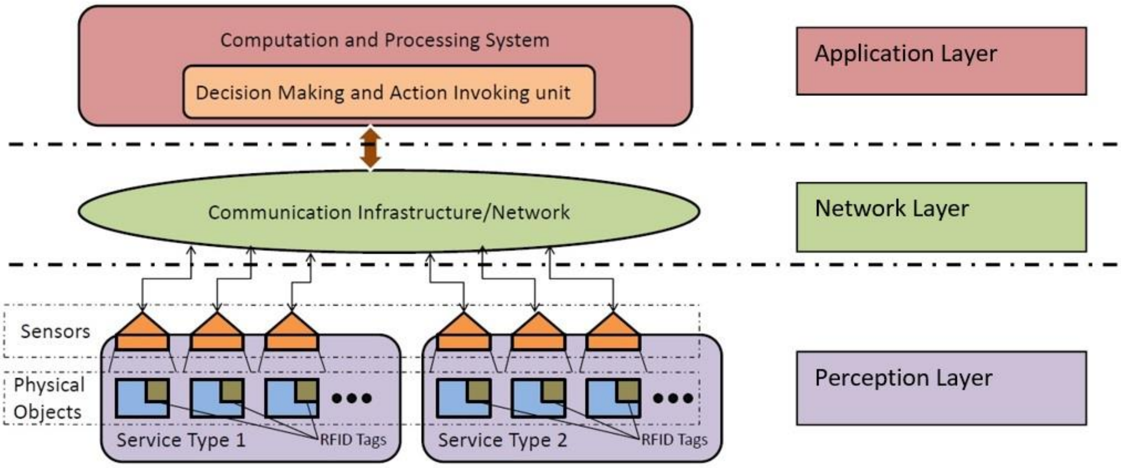


Figure 2: IoT basic architecture [12-14]

It has three layers, namely, the perception, network and application layers [12].

3.1.1 The perception Layer:

The Perception layer is also known as ‘Device Layer [14], Objects Layer [11] and Sensor Layer [8]’, which has sensors that can be sensing and interconnecting of the physical and digital world and gathering information in real-time about environment. The main function of these sensors is to querying location, temperature, weight, motion, vibration, acceleration, humidity, etc. Moreover, the perception layer digitizes and transfers data (information collected) to the network layer through secure channels. The big data created by the IoT are initiated at this layer [14-11].

Most sensors require connection to sensor gateways. This can take the form of a local area network (LAN) such as Ethernet and Wi-Fi connections, or a personal area network (PAN) such as ZigBee, Bluetooth, and Ultra-Wide Band (UWB). For sensors that do not require connectivity to sensor aggregators, their connectivity to backend servers/applications can be provided over a Wide Area Network (WAN) such as GSM, GPRS and LTE. Sensors using low data rate, low power connectivity typically form networks commonly referred to as wireless sensor networks (WSN). WSNs are becoming more popular as they can host many more sensor nodes while maintaining decent battery life and covering large areas [8].

3.1.2 The network Layer:

The network Layer (also be named “Transmission Layer” [14]): This layer can be transporting the information whether the transmission is wired or wireless from sensor devices to the information processing system using different technologies either 3G, UMTS, Wi-Fi, Bluetooth, infrared, ZigBee, etc. depending upon the sensor devices. In addition, this layer responsible for connecting the smart devices to network devices and servers [14].

3.1.3 The application Layer:

The application LayerThis layer is responsible for providing specific application services to the user and provides general management of the application based on information about the objects processed in the network layer [14]. It defines various applications in which the Internet of Things can be implemented, such as smart homes, smart cities, smart agriculture, smart transportation, and smart health[12].

Although the basic idea of the IoT architecture is defined by three layers, it is not sufficient for IoT research, as research often focuses on the finer ones of IoT, which is why multi-layered architectures are proposed by researchers, like five-layer architecture, which goes further includes the processing and business. The five layers are Perception, Transport, Middleware, Application and Business (see Figure 3). The role of the perception, network (transport) and application layer are the same as in the three-layer architecture [12]. The functionality of the other layers is described as follows:

3.1.4 The middleware Layer:

The middleware Layeris also known as the processing layer [12] or the service management layer [15]. This layer is responsible for managing services that store, analyze, and process vast amounts of data from the transport layer. Each device connects and communicates only with other devices that provide the same type of service. However, many IoT devices implement different types of services. In this case, this layer can manage heterogeneous services and connect them to the database [14]. It makes use of numerous technology which include cloud computing, big data processing modules and databases [12].

3.1.5 The Business Layer:

This layer is responsible for management and monitoring the other layers. Based on the information received from the application layer, the business layer builds graphs and models which the success of the IoT technology is depended on it. In addition, it helps to determine strategies of business and the future actions. As well as, to maintain users' privacy and enhance services, the business layer compares output of each layer with the predict output. Moreover, it is the responsible for the management of the whole IoT system [11-14].

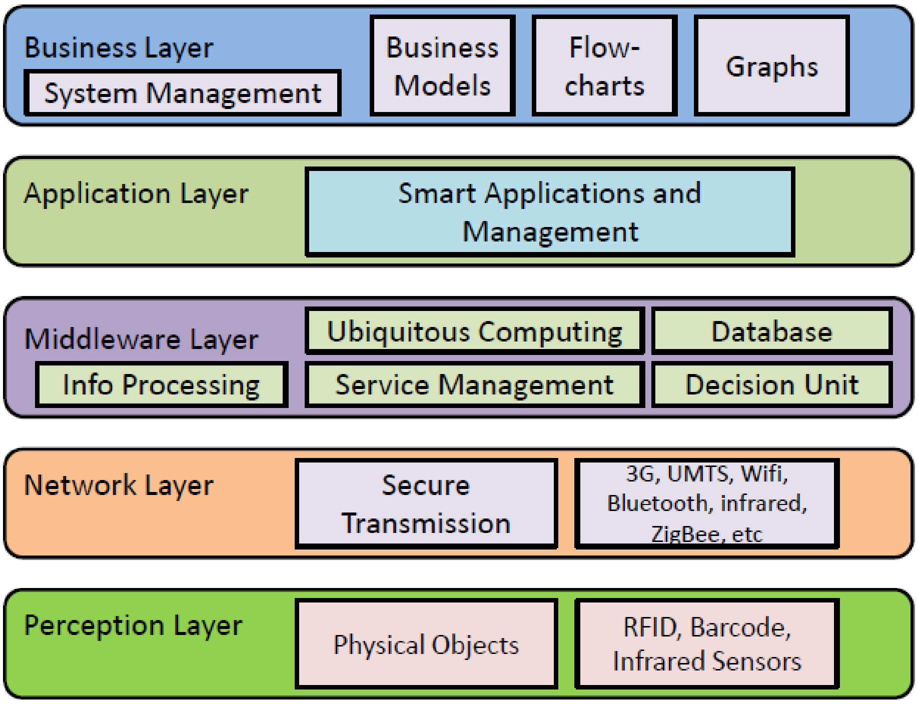


Figure 3: The IoT architecture (5 layers) [14]

3.2 Cloud and Fog Based Architectures:

The Internet of Things (IoT) has grown exponentially in a very brief duration of time. Since 2011, number of devices has already exceeded the number of people of Earth [16]. According to IoT Analytics, connected devices have reach 14.4 billion in 2022 and are expected to grow more rapidly and reach 27 billion in 2025 (IoT Analytics, May 2022) which means the IoT employs many embedded devices, like sensors and actuators that generate big data which in turn requires complex computations to extract knowledge. Therefore, the storage and computing property of the cloud present the terrific preference for the IoT to preserve and process large data [11]. Cloud computing allows to rent infrastructure, runtime environments and services in a pay-per-use basis. It can offer different types of solutions based on user's requirements, such as, scaling an enterprise infrastructure on demand and sizing it according to the business needs. In case of end users, having their data available anytime, anywhere from any device that has a connection to the internet. Cloud computing is an extremely flexible environment for building new systems and application, and even integrating additional capacity or new features into existing systems [17].

Cloud computing presents 4 classes of services, namely: Software as a Service (SaaS), Platform as a Service (PaaS), Networks as a Service (NaaS), and Infrastructure as a Service (IaaS).

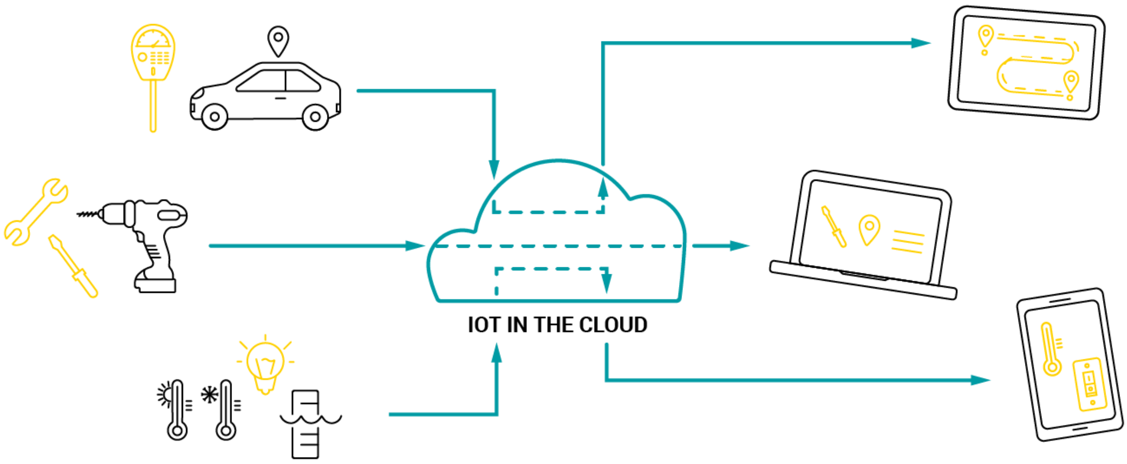


Figure 4: IoT in the Cloud

In spite of the importance of the Cloud computing, there is a move towards another system architecture namely Fog computing (a.k.a. cloudlets or edge computing [11-12]),where a part of the data processing and analytics are doing by the sensors and network gateways. this architecture presents six layers as shown in Figure 5, which contents: the physical, monitoring, preprocessing, storage, security and transport layer [12].

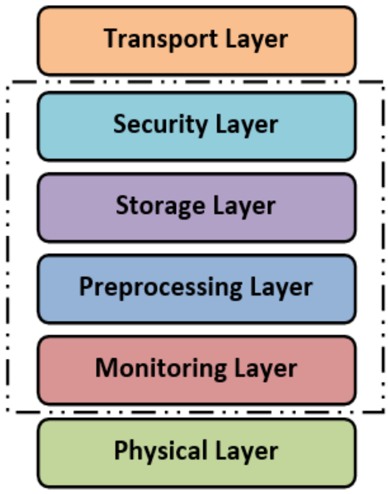


Figure 5: Fog architecture of a smart IoT gateway [12]

The services, responses, resources and power are all keep under observation the monitoring layer. Sensor data is filtered, processed, and analyzed by the preprocessing layer. Storage functions such as copying, distributing, and storing data are provided by the temporary storage layer. Finally, the security layer performs encryption/decryption and ensures data integrity and privacy [12].

With fog computing, it is possible to extend cloud computing services to the edge devices of the network. Thanks to its proximity to the end users compared to cloud data centers, fog computing can offer services with better latency performance. It should be emphasized here that there is usually a significant difference in size between fog and cloud, so the cloud has tremendous processing, storage and communication capabilities compared to fog. Therefore, fog computing has the potential to increase the overall performance of IoT applications while attempting to run some of the high-level services offered by the cloud on local resources. Figure 6 illustrates the roles that the cloud data-centers and the fog computing play to deliver IoT services to end-users [11].

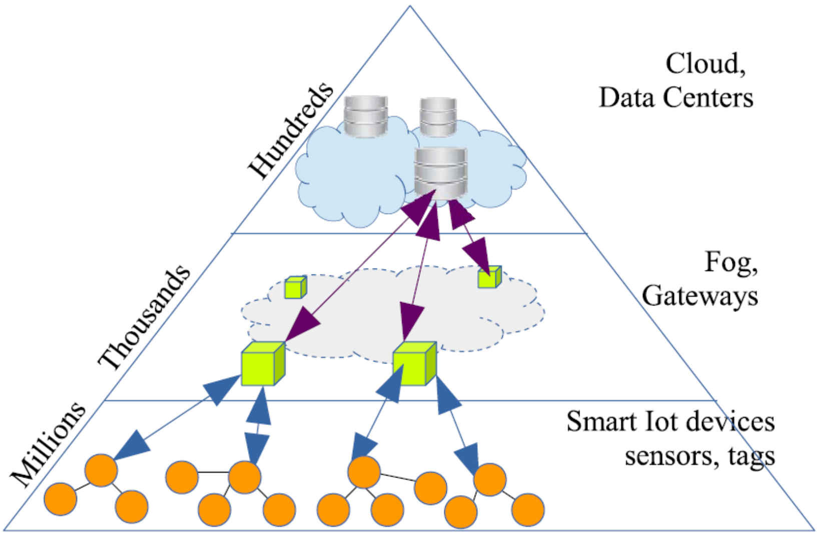


Figure 6: the roles that the cloud data-centers and the fog computing play to deliver IoT services to end-users [11].

4. Standards of Internet of Things:

> ***"Standards are not only a technical question. They determine the technology that will implement the Information Society, and consequently the way in which industry, users, consumers, and administrations will benefit from it." (EC, 1996).***

This quotation presents two crucial observations that are frequently ignored. The initial observation is that Information and Communication Technology (ICT) systems cannot operate without foundational standards. The second observation emphasizes that the ICT standards in use today will evolve into the technology of tomorrow [18].

Countless standards are utilized in all ICT systems, ranging from the most intricate global infrastructure to the simple personal computer found in a household. These standards encompass various areas such as operating systems, programming languages, communication protocols, user interfaces, cables, connectors, and disk drives [18].

Nowadays, the IoT is a fast-developing area that links diverse devices and systems to the internet, allowing them to communicate and share information. Similar to other fast-growing fields, it is necessary to establish standards to guarantee that various IoT devices and systems can work together efficiently.

To start our discussion on IoT standards, let us first give an introduction to various standardization bodies that are focused on creating advanced protocols and standards for IoT.

4.1 Standardization bodies :

One factor which has led to interoperability issues across different application domains in the IoT environment is the diversity of heterogeneous devices from various vendors and technologies.to begin to tackle this problem, several standardization bodies have emerged that accelerated the creation of low-power wireless standards and technologies. These standards incorporate specific improvements in the media access control, physical layer, network, and application layer to meet the requirements of low power consumption, low latency, high reliability, scalability, and security for IoT applications [20].

In the following, we will describe some of the most notable standardization organizations that are actively involved in developing IoT protocols.

4.1.1 IEEE Standards Association (IEEE-SA):

The IEEE Standards Association (IEEE-SA) is widely known as the most organization for standardization. since 2017, IEEE-SA have over 1100 active standards, as well as more than 600 standards currently under development. the lower layers of the model OSI protocol stack is the primary focus of IEEE-SA especially the MAC and PHY layers. The IEEE 802.15.4 standard provides as the basis for many important technologies, including ZigBee, Thread, WirelessHART, and ISA100.11a [18-20].

4.1.2 The Internet Engineering Task Force (IETF):

The Internet Engineering Task Force (IETF) is one of the non-income organizations that developed IoT requirements which includes 6LoWPAN or Routing Over Low energy and Lossy networks (ROLL). IEFT aims to give solutions to solve challenges such as adaptation of IPv6, scalability and timeliness for low-power wireless networks, hence it encourages developers to participate and contact on new ideas to improve the use of IoT in various domains like smart cities, healthcare and other. In addition, the developers and researchers can work together to develop and build standards via a voluntary platform which provided by IETF such as the Request for Comments (RFC)s, for instance in 2007, (RFC4944) become advanced and up to date via way of means of the drafts (RFC6282), (RFC6775), (RFC8025), (RFC8066) up till 2017 for growing 6LoWPAN [18-20].

4.1.3 The European Telecommunications Standards Institute (ETSI):

Similar to the IETF, the European Telecommunications Standards Institute (ETSI) is non-profit organization that works for the standards organizations in Europe. It develops global standards for Internet communication technologies such as fixed, mobile, broadcast and short-distance technologies. An example of an ETSI standard is the Third Generation Partnership Project (*3GPP*^{TM}). ETSI currently has 800 member organizations from 67 countries.

4.1.4 International Society of Automation (ISA):

the Industrial Internet of Things (IoT) and its different applications, such as business integration, security and batch control as one of the goals of the International Society of Automation (ISA). the standard ISA100. 11a is developed by ISA as widely-adopted standard. ISA is being developed by about 140 committees and sub-committees in the form of working groups, and ISA has more than 40,000 members.

4.1.5 Internet Protocol for Smart Object (IPSO) alliance:

Internet Protocol for Smart Object (IPSO) alliance is an active organization for enabling Internet Protocol (IP) connectivity for smart object communication in IoT. IPSO was founded in 2008 and targets IoT applications, such as smart cities, home automation, healthcare, and energy management.

4.1.6 Bluetooth Special Interest Group (SIG):

In September 1998, the Bluetooth Special Interest Group (SIG) was established as a non-profit organization with the aim of developing and licensing Bluetooth technology. The latter also publishes basic specifications for a different version of Bluetooth. Currently, more than 30,000 member companies support SIG. Despite it owns the Bluetooth word mark and the combination mark that make up the Bluetooth trademarks, SIG does not sell any Bluetooth products.

4.2 Standards :

Here are some common IoT standards and their explanations:

4.2.1 The IEEE 802.15.4 standard:

The IEEE 802.15.4 standard establishes the Physical and Media Access Control (MAC) layers for Low-Rate Wireless Personal Area Network (LR-WPAN) (Figure 7) [19-22], which are networks designed to connect low-power, low-data-rate devices over short distances like sensors and control systems in home automation and industry. It offers a low-power, low-data-rate communication protocol, which allows devices to communicate at data rates up from 40 kbps to 250 kbps [20-21], with provisions for security and support for various network topologies, including star, tree, and mesh. Because of its attractive characteristics such as being low-power, low-cost, and having a moderate data rate, it has become the most commonly utilized standard for a variety of applications, including home automation, industrial automation, smart cities, and Wireless Body Area Networks (WBANs). The majority of current standards, including Zig-Bee, WirelessHART, and ISA100.11a, utilize IEEE 802.15.4 as the PHY layer technology, with some modifications made to the upper layers [20].

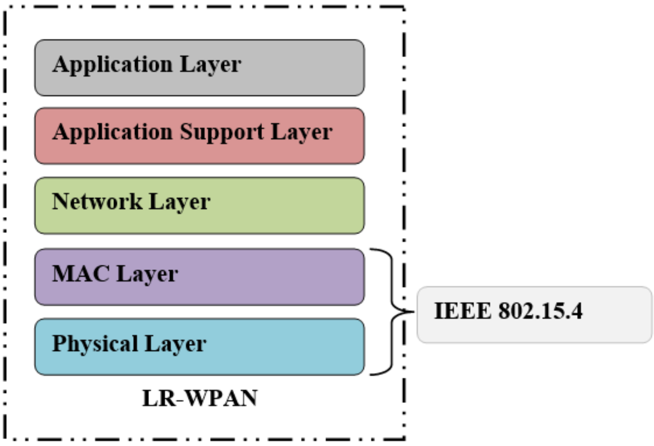


Figure 7: Architecture of IEEE 802.15.4.[19]

The standard also provides reliable data delivery using retransmission and acknowledgments, ensuring data delivery with high reliability in noisy or congested environments. Overall, the IEEE 802.15. 4 standard is a robust and flexible platform for LR-WPANs, facilitating many low-power, low-data-rate wireless applications like smart homes, industrial automation, and healthcare monitoring, among others.

The physical layer has several responsibilities including turning on and off the radio transceiver, sending and receiving data, and choosing a channel to listen on. There are two different frequencies supported at this layer: the low-band (868/915 MHz) which uses binary phase shift key modulation (BPSK), and the high-band (2.4 GHz) which uses offset quadrature phase shift keying modulation (OQPSK). There are a total of 27 channels available, with 11 in the low-band and 16 in the high-band. The low-band has a raw bit rate of 20-40kbps, while the high-band has a raw bit rate of 250kbps. To ensure robust and noise-resistant transmissions, different modulations and spreading techniques are utilized [23].

At the MAC layer, there is a feature called CSMA/CA, which works alongside a time slot structure and optional security measures. Before a device sends data, the PHY layer checks if the medium is being used. If it is occupied, the transmission is delayed for a certain period of time [23].

	Europe	*America*	*Worldwide*
Frequency (MHz)	868-868.6	902.928	2400-2483.5
No. of channels	1	10	16
Channel Bandwidth	600 KHz	2 MHz	5 MHz
Data Rate	20 kbps	40 kbps	250 kbps
Modulation	BPSK	BPSK	OQPSK

Table 1: Frequency bands of the IEEE 802.15.4 standard [20].

The network is comprised of two categories of devices: Full Function Devices (FFD) and Reduced Function Devices (RFD). FFDs have the ability to perform a range of activities such as network coordination, routing, and sensing. On the other hand, RFDs are limited nodes that can only act as end-devices to carry out sensing tasks. They can also communicate with other devices using one of following topologies: star or peer-to-peer [19-20-22-23].

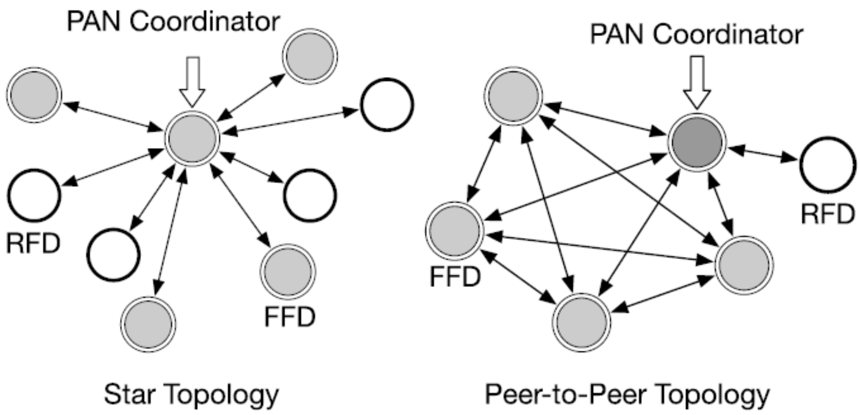


Figure 8: The star and peer-to-peer network topologies supported by the IEEE 802.15.4 standard [20].

4.2.2 Bluetooth:

Bluetooth was designed to enabling electronic devices to communicate wirelessly via replaced cables with the wireless link [20-25]. Bluetooth is a low-power low-cost wireless technology [20-22] for establish Personal Area Network (PAN) over short distances, typically less then 100m [22-26]. Nowadays, most of the sensors, actuators, controllers and critical wireless infrastructures are use a Bluetooth technology as a means to connect each of others, for instance about 30% of devices are connect to the Internet are using Bluetooth, that according to the predict of Allied Business Intelligence [20].to decreasing of the interferences Bluetooth uses frequency hopping spread spectrum (FHSS) with 2.4 GHz [22].

The Bluetooth technology contain of two big categories: First, the Bluetooth Basic Rate and Enhanced Data Rate (BR/EDR) which who represents the earlier versions of Bluetooth mainly designed for audio streaming and file transmission. Second category is Bluetooth Low Energy (BLE) refers to the recent versions of Bluetooth targeting low-power consumption for IoT applications [20].

There are two modes of devices, the first is called single-mode device, that means this device have only implemented BLE and cannot communicate with other device that only implement classic Bluetooth. Second mode, both BLE and BR/EDR are equipped in the devices like smartphones and laptops, this mode called dual-mode devices and can seamlessly transition between the BR/EDR and BLE protocol stacks as required [25]. The second mode is desired by consumers because these devices that have low power consumption and can handle high data transfer rates [20].

Bluetooth Basic Rate and Enhanced Data Rate (BR/EDR):

BR/EDR (also known as a Bluetooth Classic) has gone through several versions, which are mentioned as follows:

The first version aims to share files via an Asynchronous Connection-Less (ACL) connection. The type of this link is point-to-multiple point that allows for the broadcasting of data and can support both asymmetrical and symmetrical connection. After that, in order to add audio streaming capability to BR/EDR another link was used namely Synchronous Connection-Oriented (SCO) link in v1.2. This new connection gives up to three symmetrical point-to-point links which keep time slots to ensure timely transmission. Correspondingly, SCO packets do not transmit ACKs to avoid delay in voice transmission. To progress voice transmission reliability, eSCO is released by SIG to enhance the connection in 2003 year. Additionally, this version uses Adaptive Frequency Hopping (AFH) to improve cohabitation with other wireless protocols. Bluetooth versions up to v1.2 or as known as Bluetooth Basic Rate (BR) provide a maximum throughput of 721.2 kbps [20].

Because there was a high demand for transmitting large files and audio, BR is incapable to meet the required throughput. As a solution, in 2004 SIG introduced Bluetooth v2.0 which was named Bluetooth Enhanced Data Rate(EDR), this version increased the throughput to 2.1 Mbps. Although attempting improve the data rate, EDR does not satisfy the entire user requirements. Therefore, in April 2009 SIG launched Bluetooth v3.0 also known as Bluetooth High Speed (HS). Bluetooth HS can provide up to a 24 Mbps data rate by using the 802.11 radio. The radio is changed from BR/EDR to 802.11 and vice versa by the Alternative MAC/PHY (AMP) controller, which enables the 802.11 radio when a higher throughput is required, and then switches back to the primary BR/EDR radio to save power [20].

In BR/EDR, the radio operates on the 2.4 MHz Industrial, Scientific and Medical (ISM) band at frequencies between 2402 to 2480MHz. It uses Gaussian Frequency Shift Keying (GFSK) modulation with three output power classes:

- Class 1: it is designed for high-range devices with achieving communication range 100 m and maximum output power 100mW (20 dBm).
- Class 2: for normal PCs and portable, the communication range achieves almost 10 m and output power 2.4-2.5mW (4 dBm).

- Class 3: for low power devices with range less than 1 m and output power 1mw (0 dBm).

The frequency band is divided into 79 channels, and each channel is 1MHz wide [20-26].

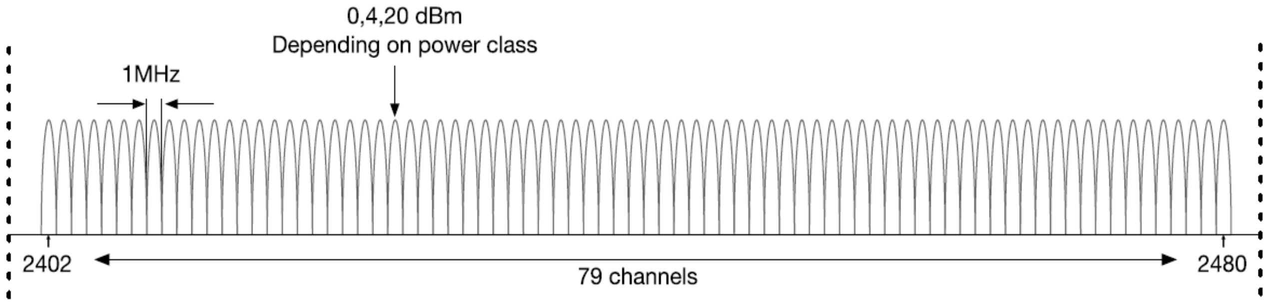


Figure 9: The channel assignment of BR/EDR.

Bluetooth Low Energy (BLE):

Bluetooth Low Energy, also known as Bluetooth Smart, was introduced in 2010 [27] by SIG as a wireless technology suitable for low power applications that did not require high data rates [20]. Its purpose was to expand the use of Bluetooth technology to power-constrained devices such as wireless sensors and controls [27]. BLE was included as part of the Bluetooth 4.0 specification, and it uses short-range radio with minimal power consumption to operate for extended periods. Its coverage range is approximately 100 meters, which is approximately ten times more than traditional Bluetooth, and it has 15 times less latency than traditional Bluetooth. With a power range between 0.01mW and 10mW, BLE is an ideal protocol for use by IoT devices due to its unique features [19]. In order to reduce power usage, the link layer, PHY layer, and packet formats were modified. In addition, whereas BR/EDR was intended for bi-directional communication, BLE devices can function solely as a transmitter or a receiver. For instance, a remote control only needs to send a brief command and does not require a response. On the receiving side, a TV or lamp only operates as a receiver and waits for commands to act upon [20].

After a year, SIG introduced significant modifications and released version 4.2 of Bluetooth. The update involved an increase in the actual payload size from 27 Bytes to 251 Bytes, while retaining the same packet format, which led to a 2.6 times increase in throughput. This enhancement in packet size also improved the feasibility of supporting IPv6 [20].

In 2016, the Bluetooth Special Interest Group (SIG) released a document outlining the features of Bluetooth 5, which aims to address the limitations of older Bluetooth versions. Bluetooth 5 improves range, increases the size of the advertisement payload, and boosts throughput [20]. This update significantly enhances the performance of applications such as smart homes, smart parking, e-governance, enterprise, industrial markets, and others. The introduction of long-range capabilities, higher symbol rates, and advertising extensions marks a significant advancement in Bluetooth technology, providing better area coverage, improved spectral efficiency, and support for next-generation beacons. With its low power consumption, Bluetooth 5 has a real impact on the Internet of Things (IoT) [28].

Theoretically, version 5 supports output power up to 20 dBm. Similar to the classic version, the PHY layer in BLE operates in the 2.4 GHz frequency band. In contrast to BR/EDR, it has 40 channels, and each channel is 2MHz wide. Among these channels, there are three special channels called advertisement channels. As shown in Figure 10, these channels are placed in the center frequency of 2402 MHz, 2426 MHz, and 2480MHz and are specified with a gray color [20].

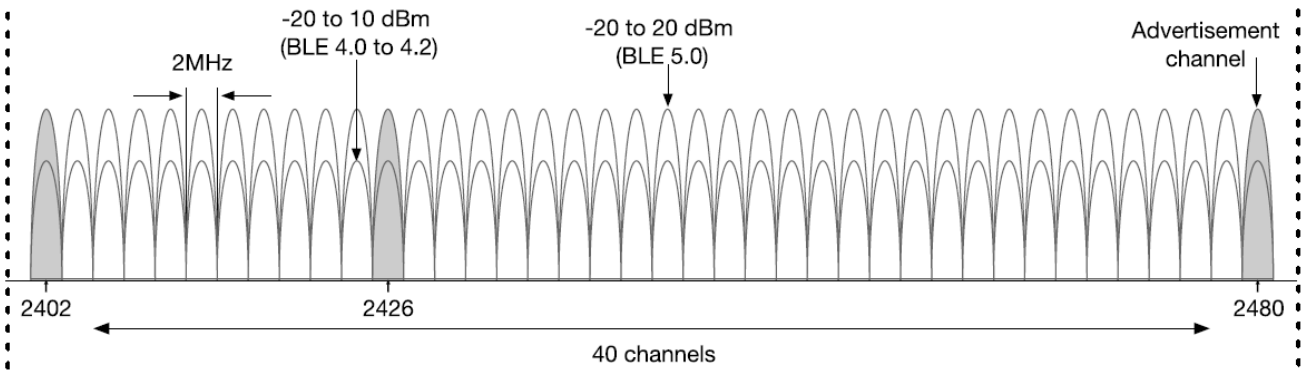


Figure 10: The channel assignment of Bluetooth 5.

The Bluetooth Core Specification version 5.1 has been updated with new features for finding the direction of Bluetooth signal transmission. These features were introduced by the Bluetooth SIG in 2019 and are called direction-finding. There are two methods for determining the angle from which a Bluetooth signal is transmitted with high accuracy, which are called Angle of Arrival (AoA) and Angle of Departure (AoD). Each method requires one of the two communicating devices to have an array of multiple antennae, with the antenna array being in the receiving device for the AoA method and in the transmitting device for the AoD method.With Bluetooth 5.1, the BLE controller in the receiving device can now generate data that can be used to calculate the directional angle to the transmitting device. The ultimate goal is to enhance Bluetooth location services with these new direction-finding features, and when associated profiles are released, developers will be able to create high-accuracy and interoperable positioning systems such as real-time locating systems and indoor positioning systems. These direction-finding systems also improve device proximity by enabling directional item finding and point of information interest. In 2020 Bluetooth sig launched a new version is Bluetooth 5.2, this version contains a series of updates to the Bluetooth core specification with the next generation Bluetooth audio called LE Audio [28].

Specification	***Bluetooth Classic***	***Bluetooth Low Energy (BLE)***
Frequency Band	2.4GHz ISM Band (2.402 – 2.480 GHz Utilized)	2.4GHz ISM Band (2.402 – 2.480 GHz Utilized)
Channels	79 channels with 1 MHz spacing	40 channels with 2 MHz spacing (3 advertising + 37 data channels)
Channel Using	Frequency-Hopping Spread Spectrum (FHSS)	Frequency-Hopping Spread Spectrum (FHSS)
Modulations	GFSK, π/4 DQPSK, 8DPSK	GFSK
Network Topologies	Point-to-Point (including piconet)	Point-to-Point (including piconet) Broadcast or Mesh
Range	100 m	<100 m
Application Throughput	0.7–2.1 Mbps	0.27-1.37 Mbps
Latency	Typically, 100 ms	6 ms
Primary use cases	Mobile phones, gaming, headsets, audio streaming, smart homes, wearables, automotive, PCs, security, proximity, healthcare, etc.	Mobile phones, gaming, smart homes, wearables, automotive, PCs, security, proximity, healthcare, sports, industrial, etc.

Table 2: Difference between Bluetooth Classic and Bluetooth Low Energy [28].

4.2.3 ZigBee:

The ZigBee Alliance has developed ZigBee as the initial universal standard for various applications. This alliance includes corporations such as Honeywell, Mitsubishi Electric, Motorola, Philips, and Invensys [30].

ZigBee is a specification standard for a suite of high-level communication protocols that are designed for creating Personal Area Networks (PAN) from small, digital radios with low power consumption [29]. ZigBee devices, despite having low power, are capable of transmitting data over greater distances by leveraging intermediate devices to pass the data along, thereby establishing a mesh network. These devices are suitable for applications that require secure networking, long battery life, and low data rates [29].

ZigBee is based on an IEEE 802.15.4 standard which describes PHY and MAC layer. In addition, the data transmission rate can reach 250 Kbps [31-32] with distance range is from 10 to 100 meters [32]. Basically, ZigBee operates on three frequency bands: 868 MHz for European applications, 902-928 (or 915) MHz for North American applications and 2.4 GHz all over world [30-32]. It supports three types of network topologies; star topology, mesh topology, and combined topology.

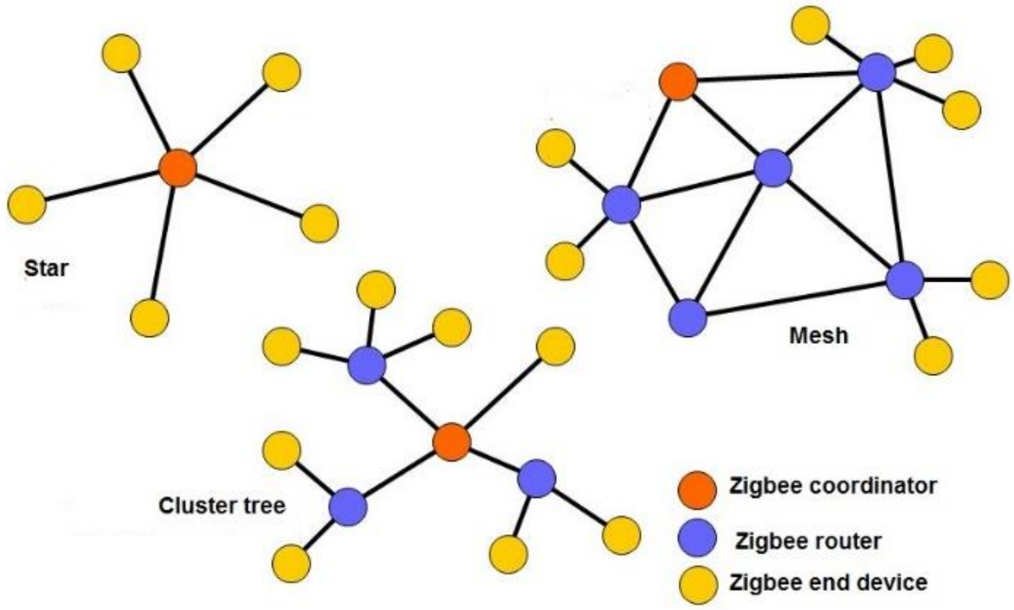


Figure 11: Zigbee network model.

ZigBee's utilization of a low data-rate and its ability to conserve battery power make it highly versatile for monitoring and control functions. hence, it is widely used in several domains such as smart homes, personal health care, building automation, industrial control and commercial control.ZigBee is also appropriate for Wireless Sensor Networks (WSN) which used in homes or agriculture fields. Lately, ZigBee devices are getting more attention as it has varied applications in the domain of Internet of Things (IoT) [32-33].

4.2.4 Long Range Wide Area Network (LoRaWAN):

The LoRa Alliance introduced a layer called LoRaWAN that defines communication roles and is placed on top of LoRa technology, which knows as a one of the most adopted, low-power wireless technologies for low data rate and long-range connectivity [20]. it is one of the Low Power Wide Area Network (LPWAN) technologies that have received significant attention by the research community in the recent years [34].Compared to cellular technologies, LoRaWAN has the advantage of being more suitable for low-power applications because cellular technologies consume a significant amount of battery power. For example, a soil moisture sensor in agricultural monitoring needs to sense the humidity level every few days, and the battery has to last for a few years [20].

The functionality of the network was expanded by LoRaWAN through the introduction of a network server and the specification of unique device classes for various applications. With LoRaWAN, the network server is responsible for managing the communication, filtering the multiple copies from one packet received by different gateways and scheduling the communication.For instance, the network server decides the gateway that should communicate with certain end nodes and when to change the data rate [20].

There are three various modes in which LoRaWAN end-node devices can uses (figure 12), which depend on the communication tool used: Class A (which is the default mode), Class B and Class C, these modes determine the way in which end node device can connect to a wireless network [20-36].

- Class A is the default class and uses pure ALOHA to access the link for uplink (UL) [20-35]. In this class, the end node wakes up periodically and sends a packet. it then remains in a waiting state to receive a downlink (DL) in the same channel and the same spread factor (SF). the end node waits for the second packet with a different SF and in different frequency when it does not receive the first packet. The end node enters sleep mode once it has received a response from the gateway. The device on the gateway side has a limitation and can only send information once, and this can only happen immediately after receiving a packet from the end node.Although this class is not suitable for real-time applications, it has the benefit of having the lowest power consumption due to the extended sleep period. Generally, noncritical sensors with long battery life requirements are the ones that mostly utilize this class [20-35-36].

- Class B is intended for applications that require more frequent connectivity than Class A. It is a hybrid of Classes A and C, where end nodes wake up in response to a beacon transmitted by the gateway, which schedules the wake-up time for the devices. This frequent access ensures that connections can be established when needed, but the overhead caused by the beacons and frequent wake-up times can reduce battery life [20-35-36].
- Class C is typically used for devices that are connected to an AC power source, where energy consumption is not a significant concern. In this class, devices are constantly monitoring the channel except when transmitting data, making them suitable for real-time applications. Examples of Class C devices include street lights, which can also serve as gateways [20-35-36].

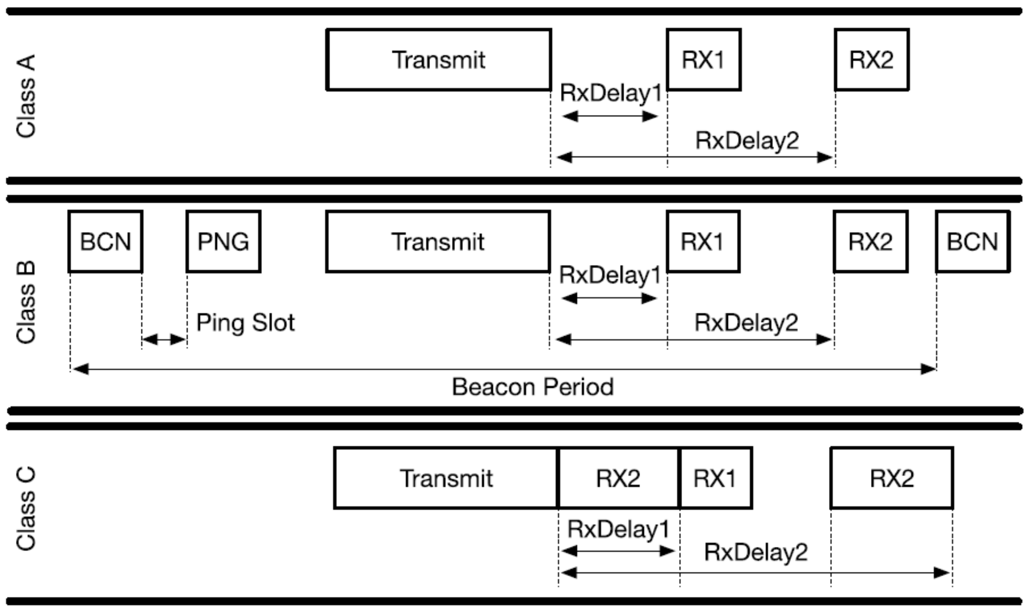


Figure 12: LoRaWAN communication classes [20].

5. Protocols:

In IoT, there are various different devices that need to communicate with each other, and these devices may have different hardware, software, and network capabilities. Protocols help to ensure that these devices can communicate with each other in a consistent and reliable way, regardless of their differences.So, it can be said that protocols in IoT refer to the standardized methods and rules that devices use to communicate with each other and with the cloud. These protocols define the format, structure, and content of the messages that are exchanged between devices, as well as the rules for how devices should handle these messages.

By using standard protocols, IoT devices can exchange data seamlessly, enabling them to work together to achieve common goals. For example, a smart home system may use different devices from different manufacturers, but if they all use the same protocol, they can all communicate with each other and work together to provide a seamless user experience

5.1 IOT Protocol Stack

The IoT protocol stack that fully confirms the effectiveness of the wireless communication stack is depicted below. [37].

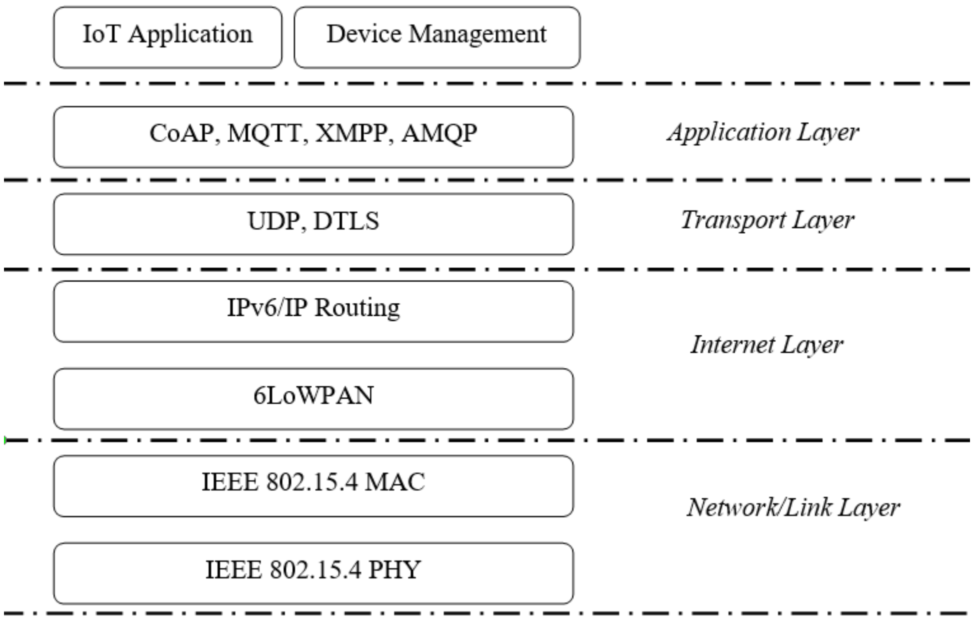


Figure 13: IoT Protocol Stack [38].

5.2 Constrained Application Layer Protocol (CoAP):

The IETF's CoRE (Constrained RESTful Environments) working group created the Constrained Application Protocol (CoAP) to handle data communication, applications, and services between low-power, constrained nodes on a lossy, constrained RF network. CoAP is designed to fulfill a role comparable to HTTP's role on the traditional Internet [38]. CoAP is a lightweight web transfer protocol designed for use with constrained devices, such as low-power microcontrollers.CoAP clients are capable of achieving throughput rates of up to 100 kbps, which is adequate for many IoT applications. CoAP includes mechanisms for reliable message delivery and congestion control to mitigate the impact of packet loss on communication reliability [38].

CoAP relies on the client/server (Request/Response) architecture to facilitate communication between devices. The CoAP client transmits a request message to the CoAP server via one of the four CoAP methods (GET, POST, PUT, DELETE). The CoAP server processes the request and sends a response message back to the CoAP client. In addition, CoAP also employs a variant of the publish/subscribe (Resource/Observe) model, where the client can subscribe to resources and receive updates whenever a change occurs [11-38].

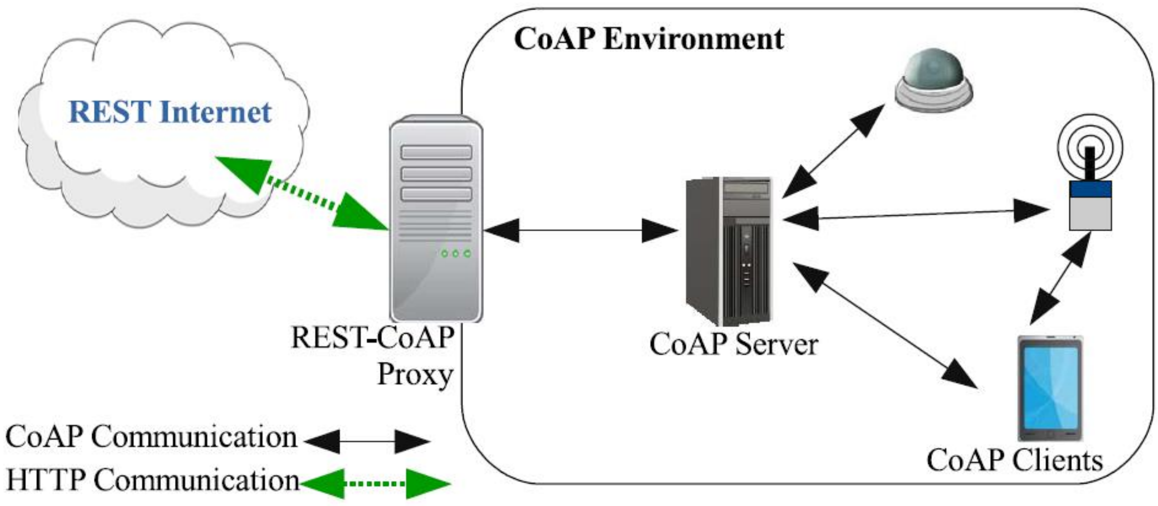


Figure 14: CoAP functionality [11]

CoAP uses a lightweight version of the User Datagram Protocol (UDP) as the transport layer, which enables low-overhead communication [11-12-38]. Additionally , CoAP includes security mechanisms, such as Datagram Transport Layer Security (DTLS), to ensure secure communication between devices [12-38].

5.2.1 Messaging Model

The CoAP communication system operates by transmitting messages between endpoints over UDP. The message format is shared by requests and responses, each message contains a Message ID used to detect duplicates and for optional reliability [39]. There are four types of messages in CoAP: CON (confirmable), NON (non-confirmable), ACK (Acknowledgement), RST (Reset) [12-38].

To ensure reliability, a message can be designated as Confirmable (CON). If a message is marked as Confirmable, it is repeatedly sent at timed intervals until the recipient acknowledges receipt with an Acknowledgement message (ACK) bearing the same Message ID as the original message (figure 15.a) from the corresponding endpoint. If the recipient is unable to process a Confirmable message, it will instead respond with a Reset message (RST) instead of an Acknowledgement (ACK) [39].A message that does not require reliable transmission can be sent as a Non-confirmable message (NON). These are not acknowledged, but still have a Message ID for duplicate detection (Figure15.b). When a recipient is not able to process a Non-confirmable message, it may reply with a Reset message (RST).

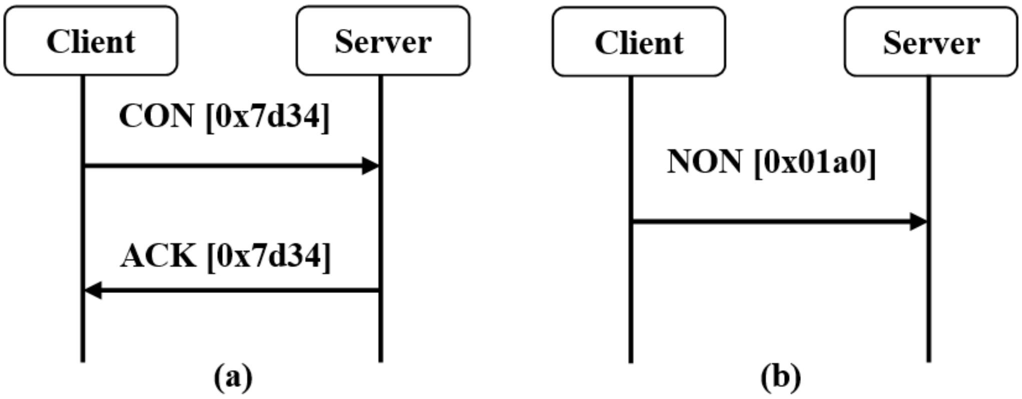


Figure 15: (a). Reliable Message Transmission. (b). Unreliable Message Transmission

5.2.2 Request/Response Model

The structure and meaning of CoAP requests and responses are carried out through CoAP messages, which include Method Codes and Response Codes to indicate the type of request and the status of the response, respectively [39].

A request is transferred in a Confirmable (CON) or Non-confirmable (NON) message, and the answer to a request carried in a Confirmable message is carried in the following Acknowledgement (ACK) message if it is instantly accessible. This is referred to as a piggybacked answer. (There is no need for separately acknowledging a piggybacked response, as the client will retransmit the request if the Acknowledgement message carrying the piggybacked response is lost.) Figure 16 shows two instances of a simple GET request with piggybacked answer, one successful and one resulting in a 4.04 (Not Found) response [39].

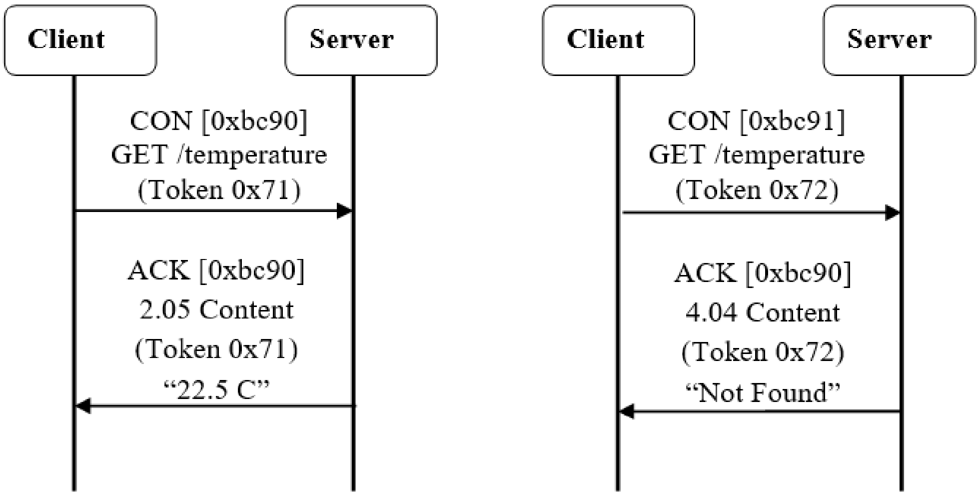


Figure 16: Two GET Requests with Piggybacked Responses [39].

If a server cannot respond immediately to a request received in a Confirmable message, it will transmit an Empty Acknowledgement message as a response, indicating to the client to stop sending further requests. Later, when the server has the response ready, it will send it to the client in a new Confirmable message. The client must acknowledge this message. This type of response is known as a "separate response", as shown in the illustration (Figure 16) [39].

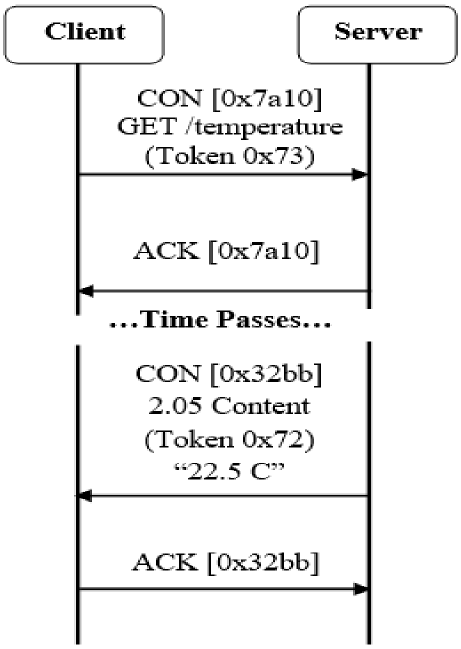


Figure 17: A GET Request with a Separate Response [39].

If a request is sent in a Non-confirmable message, then the responseis sent using a new Non-confirmable message, although the server mayinstead send a Confirmable message. This type of exchange isillustrated in Figure 18.

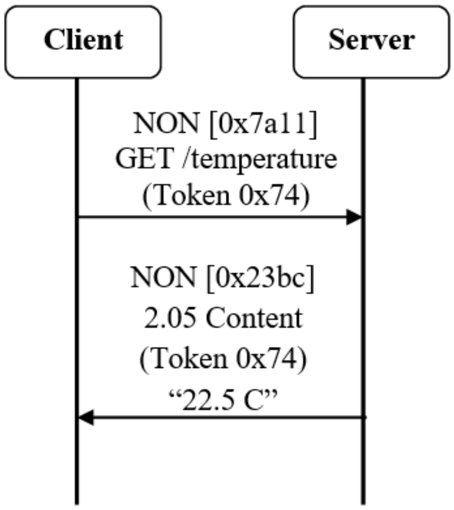


Figure 18: A Request and a Response Carried in Non-confirmable Messages

5.2.3 Message Format:

The general structure of a CoAP message is shown in Figure 19:

<table>
<tr><td colspan="5">0 1 2 3</td></tr>
<tr><td colspan="5">0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1</td></tr>
<tr><td>Ver</td><td>T</td><td>TKL</td><td>Code</td><td>Message ID</td></tr>
<tr><td colspan="5">Token (if any, TKL bytes) ...</td></tr>
<tr><td colspan="5">Options (if any) ...</td></tr>
<tr><td colspan="3">1 1 1 1 1 1 1 1</td><td colspan="2">Payload (if any) ...</td></tr>
</table>

Figure 19: CoAP Message Format [39].

The fields in the header are defined as follows [1-39]:

Version (Ver): 2-bit. Indicates the CoAP versionnumber.Other values are reserved for future versions.Messages with unknown version numbers must be ignored.

Type (T): 2-bit. Indicates the type of message.The following values are defined:

0 = CON is used for confirmable messages.

1 = NON is used for non-confirmable messages.

2 = ACK is used for acknowledgment messages.

3 = RST is used for reset messages.

Token Length (TKL): 4-bit. Indicates the length ofthe variable-length Token field (0-8 bytes). Lengths 9-15 arereserved, must not be sent, and must be processed as a messageformat error.

Code: The code is divided into a 3-bit *class* and a 5-bit *meaning*. The following values are defined for the code:

• Class 0 (000) identifies a request message.

• Class 2 (010) identifies a "Success" response.

• Class 4 (100) identifies a "Client error" response (the client shouldnot repeat the request as it will fail again).

• Class 5 (101) identifies a "Server error" response (the request faileddue to a server error; the clientmight repeat the request, dependingon the failure reason reported by the server).

For requests, the detail is used to indicate the method of the request:

1 = GET 2 = POST 3 = PUT 4 = DELETE.

For responses, the detail is used to give additional informationrelated to the response. Some examples of response codes are:

• Success responses: 2.01 Created, 2.02 Deleted, 2.04 Changed, 2.05Content

• Client error responses: 4.00 Bad Request, 4.02 Bad Option, 4.04NotFound, 4.05 Method Not Allowed, …

• Server error responses: 5.00 Internal Server Error, 5.01 Not Implemented, …

Message ID: 16-bit. Used todetect message duplication and to match messages of typeAcknowledgement/Reset to messages of type Confirmable/Nonconfirmable.

Token (optional): Token (TKL bits) is the message token. It is used to match responses to requests independently from the underlying messages. The length in bytes is specified in the TKL field.This field is optional. If not used, TKL is 0.

CoAP options: CoAP request and response semantics are carried in CoAP messages. Required and optional information for all messages, such as the URI and payload media type, are carried asCoAP. Options have the same role as in HTTP headers.

5.3 Message queue telemetry transport (MQTT):

Message queue telemetry transport (MQTT) was originally developed and released by IBM [19-38] but is now part of OASIS open standards [38]. MQTT is apublish/subscribe protocol that runs over TCP [12-38], it also designed for lightweight Machine-to-Machine IoT communication with reliable di-directional message delivery, with an open source that allows the community to participate and contribute to its development [1-38]. Connecting networks with applications and middleware and embedded devices which are in an environment constrained is the aim of the design of MQTT [11-19].

MQTT consists of three components (Figure 20): the broker, the subscriber, and the publisher. Both the subscriber and the publisher are clients to the broker, as explained [40]:

1. The Broker. The broker acts as a gateway; it is responsible for filtering messages that received through using topics as filters and determining who is subscribed to each message, and sending the message to these subscribed clients. note that the brokers sometimes referred as servers [40].
2. TheSubscriber.The subscriber announces its topics of interest to the broker, and the broker sends messages published to those topics.
3. The Publisher. the publisher considers as a generator of informations, after that sends these informations to the other subscribers by means of the broker.

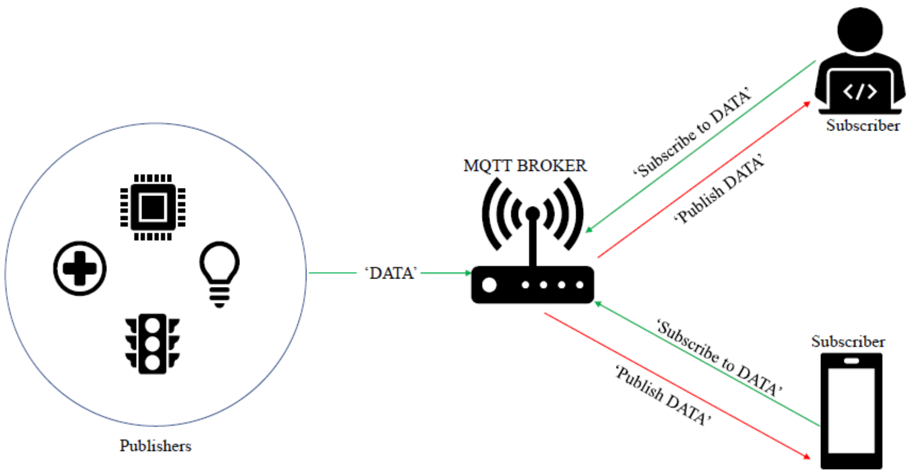


Figure 20: MQTT Protocol and Architecture [38].

5.3.1 MQTT Quality of Service (QoS):

The Quality of Service (QoS) level is an agreement between the sender of a message and the receiver of a message that defines the guarantee of delivery for a specific message. There are 3 QoS levels in MQTT:At most once (0), At least once (1), Exactly once (2).

QoS 0 - at most once: The broker/client will deliver the message with no acknowledgment and guarantee. The message is not stored and re-transmitted by the sender. QoS level 0 is often called “fire and forget”.

Figure 21: MQTT Quality of Service level 0: delivery at most once

QoS 1 - at least once: QoS level 1 guarantees that a message is delivered at least one time to the receiver. The sender stocks the message before it gets a PUBACK packet from the receiver that acknowledges receipt of the message. it can be sent the message or delivered multiple times.

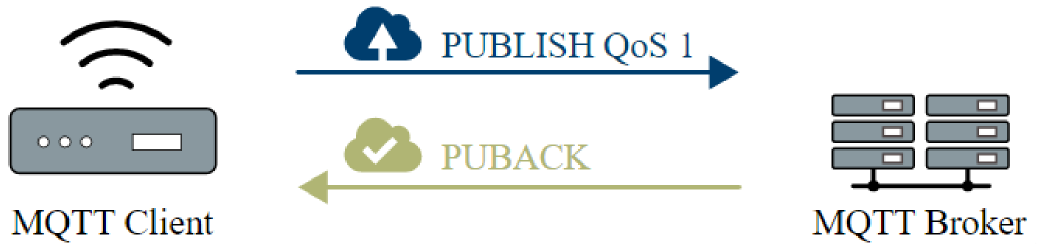


Figure 22: MQTT Quality of Service level 1: delivery at least once

QoS2- exactly once: The broker/client will distribute the message exactly once with 4 step handshakes (The guarantee is provided by at least two request/response flows), QoS 2 is the safest and slowest quality of service level.

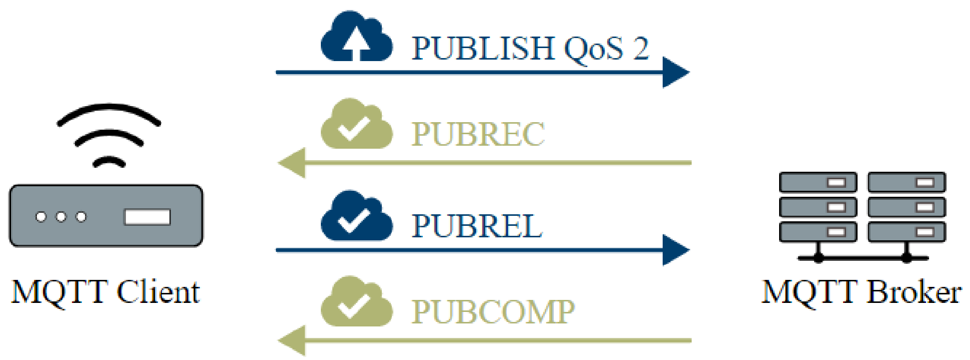


Figure 23: MQTT Quality of Service level 2: delivery exactly once

Following table mentions MQTT protocol messages exchanged between client and Broker

MQTT message	Description with direction of flow
CONNECT	Client to server (i.e. Broker), the client requests to connect to server.
CONNACK	Server to client, connect acknowledgment
PUBLISH	Client to server OR server to client Publish message
PUBACK	Client to server OR server to client Publish acknowledgment
PUBREC	Client to server OR server to client Publish received
PUBREL	Client to server OR server to client Publish release
PUBCOMP	Client to server OR server to client Publish complete
SUBSCRIBE	Client to server Client subscribe request
SUBACK	Server to client Subscribe acknowledgment
UNSUBSCRIBE	Client to server Unsubscribe request
UNSUBACK	Server to client Unsubscribe acknowledgment
DISCONNECT	Client to server, Client is disconnecting

Table 3: MQTT protocol messages exchanged between client and Broker

5.3.2 MQTT message format:

The general structure of a CoAP message is shown in Figure 24. The MQTT protocol is divided of three parts: Fixed Header, Variable Header and Payload. the Fixed Header (2 bytes) is the part that each MQTT message must contain. The other two parts are optional [42].The first Bytecontains: the message type, the flags fields (i.e., Duplicate delivery "DUP"), QoS level, and RETAIN flag "RETAIN", while Byte two consists of the remaining length field which contains a variable header and a payload [41].

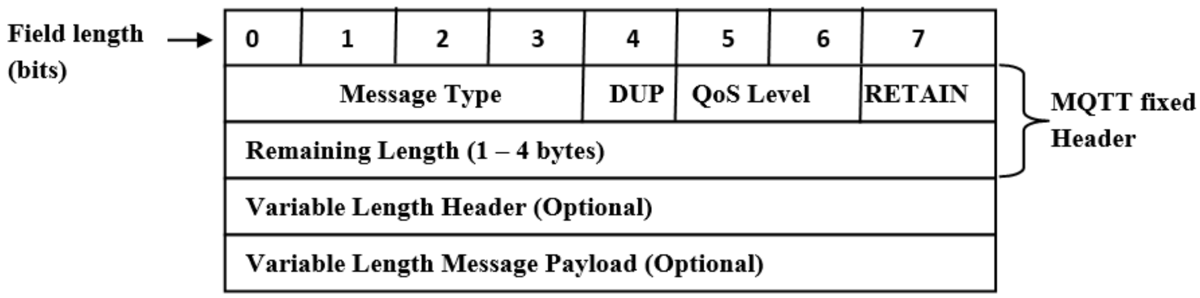


Figure 24: MQTT Message Format [41]

5.4 Advanced Message Queuing Protocol (AMQP):

AMQP was launched in 2003 by JPMorgan Chase in London, UK. AMQP is a lightweight protocol designed to communicate between Machine-to-Machine. It is a corporate messaging protocol designed for reliability, security, provisioning and interoperability[43].AMQP uses message delivery guarantees such as at-most-once, at-least-once, and exactly once delivery to ensure reliable communication. For message exchange, a reliable transport protocol such as TCP is required in AMQP and it uses TLS/SSL and SASL for security. One of the core characteristics of AMQP is reliability and it provides two preliminary levels of Quality of Service (QoS) for delivery of messages: Unsettle Format (not reliable) and Settle Format (reliable) [43].TheFigure 25 illustrates the two main components that handle communications, which are exchanges and message queues [11].

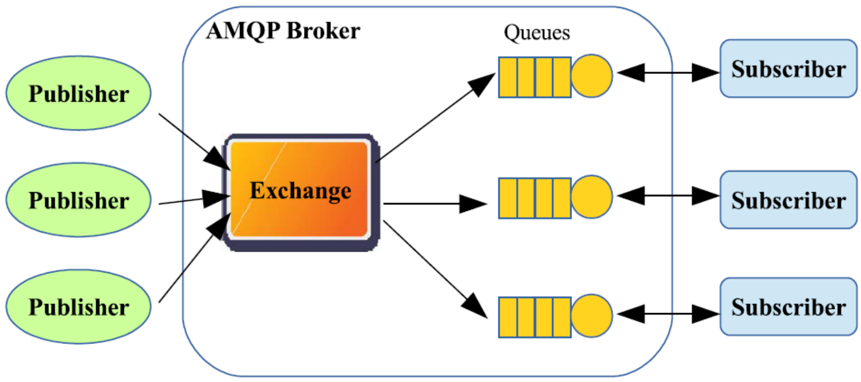


Figure 25: Publish/subscribe mechanism of AMQP [11].

Exchanges are used to direct messages to the appropriate queues, based on pre-defined rules and conditions. Messages are stored in message queues and can be subsequently delivered to their intended recipients [11]. AMQP supports both request/response and publish/subscribe architecture [43]. In AMQP there are two types of messages: bare massages that are provided by the sender and annotated messages that are seen at the receiver [11], these two messages are illustrated in the figure 26.

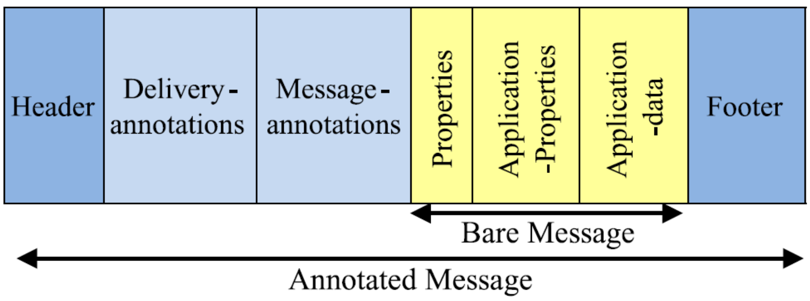


Figure 26: AMQP message format [11].

The structure of AMQP frames is illustrated in Figure 27. The first four bytes show the frame size. DOFF (Data Offset) gives the position of the body inside the frame. The Type field indicates the format and purpose of the frame. For example, 0x00 is used to show that the frame is an AMQP frame or type code 0x01 represents a SASL frame.

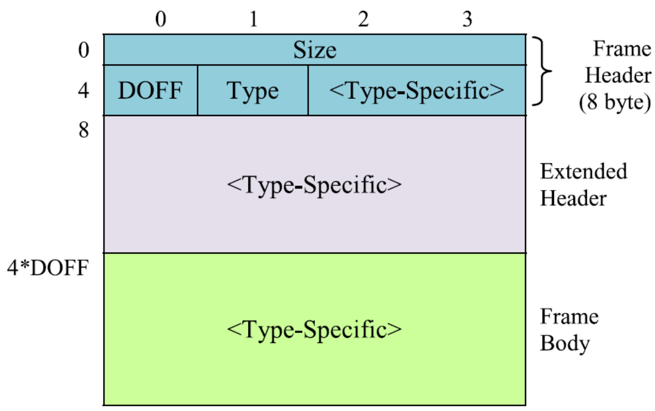


Figure 27: AMQP frame format.

5.5 IPv6 over a low-power personal area network (6LoWPAN)

IPv6 over a low-power personal area network (6LoWPAN) is an interesting protocol that supports the implementation of IoT in resource constrained environments. In addition, Small IoT devices as well as sensors can communicate securely and safely with 6LoWPAN wireless modules. Hence 6LoWPAN represent one of the most significant protocols in IoT, A working group of the IETF developed LoWPAN for interconnected networks and embedded devices by IEEE 802.15.4 (in RFC4944) [44]. It enables communication using IPv6 via the IEEE 802.15.4 protocol. An adaptation layer between the 802.15.4 link layer and the transport layer is determined by this standard. 6LoWPAN devices can connect with other IP based devices on the Internet. 6LoWPAN networks communicate to the Internet through a gateway (WiFi or Ethernet), that also has protocol support for conversion between IPv4 and IPv6 as today's deployed Internet is mostly IPv4 [12]. 6LoWPAN then developed encapsulation and compression to enable wireless IPv6 data transmissions.The first two bits in a 6LoWPAN packet determine its type. The remaining structure depends on type and the next 6 bits. Four types of headers are used by 6LoWPAN frames based on 2 bits: If they are 00x, then this is not a 6LoWPAN frame and are used if case of protocols co-existence. 010 means Uncompressed or HC1 compressed and the type of the address can be determined by the last 5 bits. 10x means that the next header is a mesh header and the next 5 bits are used for routing. 11x is a fragmentation header [12-44].

6. Applications:

IoT applications promise to bring immense value into our lives. With newer wireless networks, superior sensors and revolutionary computing capabilities, the Internet of Things could be the next frontier in the race for its share of the wallet. IoT applications are expected to equip billions of everyday objects with connectivity and intelligence. It is already being deployed extensively, few applications of IoT:

6.1 Smart environment

Nowadays, some people prefer to be in a smart environment, that connects computers, smartphones, and the internet with our daily activities, reflecting the automated society we live in. The Internet of Things (IoT) enables this smart environment to become a reality. Smart City, in general, refers to the urbanization and infrastructure development of a high standard. However, in the context of IoT, Smart City means the automation of an entire city, with managed everything via the internet this contains controlling traffic signals, monitoring pollution levels, and many other applications. In this status, for IoT deployment in a Smart City, it needs the sensors which considerate as an important and crucial device. Such a city would rely on major automation systems that support its economy, including vehicle parking tracking, monitoring seismic vibrations in buildings, pollution and radiation level tracking, traffic management, disaster recovery, waste management, and supply chain management, among many others [7].

6.2 Smart homes:

Different from a traditional home that features appliances that are operated locally and manually (e.g., magnetic switches and push buttons), the smart connected home incorporates Internet-connected appliances and devices that can interact “intelligently” with the house holders. for instance, toasting a bread via commands from a phone, turning ON the air conditioner before reaching home to keep the room cool, remotely operating all the electronic devices and electrical devices at home via a smart phone. Logging details of room temperature, dust level, noise level, even oxygen level could help users to optimize living conditions [7].

6.3 Smart agriculture:

Many activities will be easy thanks to Internet of Things such as control climate conditions that allows to maximize the production of fruits and vegetables and its quality. Also monitoring of humidity and temperature levels to prevent fungus ad other microbial contaminants and tracking of animals which grazing in open pastures or location in big stables, Study of ventilation and air quality in farms and detection of harmful gases from excrement, Control the offspring in animal farms to ensure its survival and health, and field Monitoring accurate ongoing data obtaining, and management of the agriculture fields to reduce spoilage and crop waste, and control of watering and electricity [8].

6.4 Smart health:

Patients Surveillance: Monitoring of conditions of patients inside hospitals and in old people’s home, Medical Fridges: Control of conditions inside freezers storing vaccines, medicines and organic elements, Fall Detection: Assistance for elderly or disabled people living independent, Dental: Bluetooth connected toothbrush with Smartphone app analyzes the brushing uses and gives information on the brushing habits on the Smartphone for private information or for showing statistics to the dentist, Physical Activity Monitoring: Wireless sensors placed across the mattress sensing small motions, like breathing and heart rate and large motions caused by tossing and turning during sleep, providing data available through an app on the Smartphone [8].

Conclusion:

In conclusion, the Internet of Things (IoT) has transformed the way we interact with technology, and its benefits are numerous. IoT has enabled us to connect and control devices remotely, resulting in greater convenience, efficiency, and productivity in our daily lives. IoT has also improved various industries, including healthcare, transportation, agriculture, and manufacturing, Additionally, IoT has the potential to create new business models, increase revenue, and reduce costs for companies. Overall, IoT is a powerful and transformative technology that will continue to shape our lives in the years to come. This chapter provides a brief overview of Internet of things (IoT) and its reference architecture with mains standards technologies developed especially for embedded devices and constrained environments and some of application protocols, and also its applications that menkind needs.

CHAPTER 2

SMART HOME

Introduction:

In today's world, the use of smartphones and connectivity is widespread in many regions. This alongside technological innovations, availability, and affordability of smart devices which has given rise to the Internet of Things (IoT). A smart connected home is an instance of this paradigm, as it utilizes the connectivity of various devices. Therefore, the notion of a smart home has gained significant attention in recent years due to its potential to revolutionize the way we live and how utilizes various technologies to automate and streamline daily tasks, enhance security, and improve energy efficiency. This technology has become increasingly important as it offers numerous benefits such as convenience, comfort, and cost savings. The chapter on smart homes will show the history of this technology, explore the various definitions, discuss the different types of technologies used, and examine the architecture that enables smart home functionality.

1. Definition:

There is no common definition of Smart Home. There are rather lots also lots of different names for it:

Smart Homes, also known as automated homes, intelligent buildings, integrated home systems or domotics, are a recent design development. Smart homes incorporate common devices that control features of the home [45].

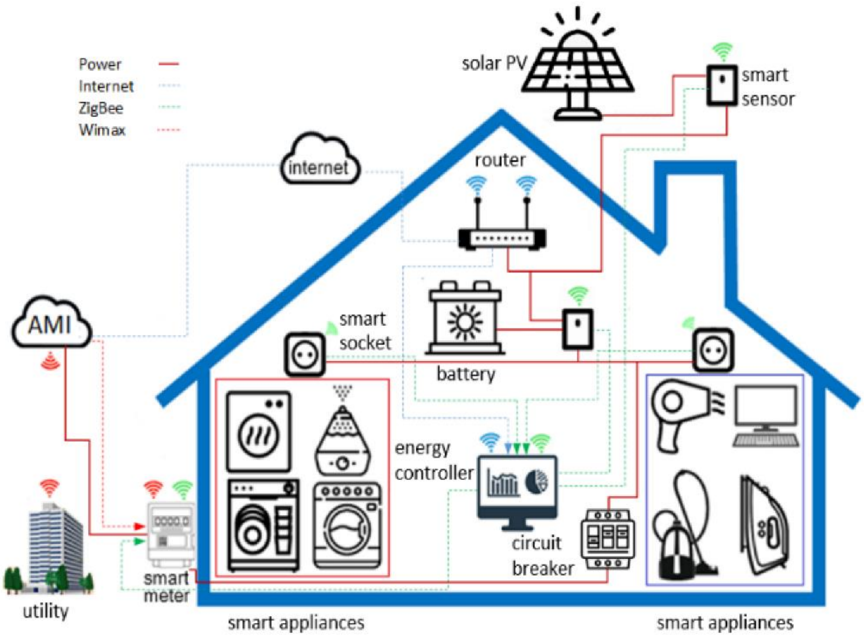


Figure II 1: Smart Home

Smart Home is a main component of the "Internet of Things," where everything has an assigned IP address, and can be supervised, controlled, and reached remotely from anywhere at any time. It is the method that the home appliances and other devices are connected to control over all aspects of smart home [56].

A smart connected home is a residence that can be remotely accessed, controlled, and monitored typically via the Internet, with information and communication technologies and sensors, systems, and devices. There are three types of homes: smart home, connected home, and a smart connected home. The relation between these types is illustrated in Figure 2 [46].

- **A smart home** refers to a collection of systems that enable homeowners to control their household appliances from within their house. Typically, these systems are not connected to the internet and use wireline-based standards like KNX. Smart homes mainly focus on automating lighting, windows, and in-house entertainment, and are commonly associated with building automation [46].
- **A connected home** is a type of home where homeowners can control and manage their appliances remotely, usually through IP-based networks such as the internet. This type of home also offers services such as security, healthcare, and energy management. It usually has a central hub or gateway that allows the entire system to be controlled, and a user interface that can be accessed through a smartphone [46].
- **A smart connected home** combines the features of both smart and connected homes and also allows for communication and service-exchange between related areas such as the grid, electric vehicle, and on-site microgeneration like rooftop solar panels. It incorporates functionalities such as learning, prediction, and response to the occupants' needs and lifestyle preferences in their home environment. The system usually includes intelligent devices that use machine learning algorithms for activity recognition and to perform actions automatically [46].

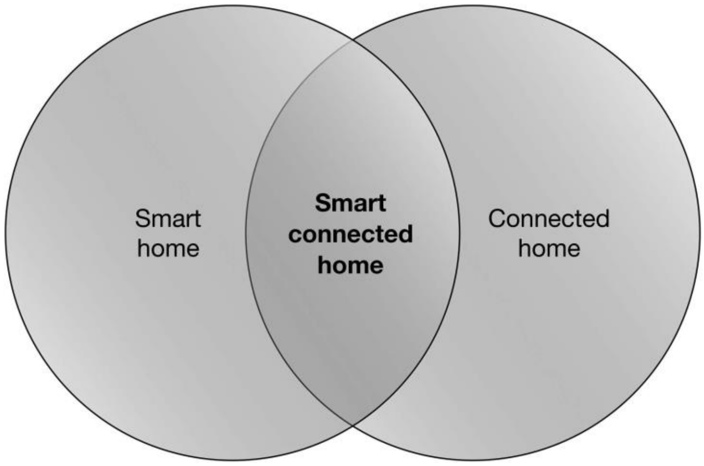


Figure II 2: A classification of the three types of smart homes [46].

2. History

The concept of a "smart home" has been around for several decades which originated in the early 1900s, and the idea of using technology to automate and control various aspects of the home has evolved over time and has had recent success with advances in IoT. Here are some key milestones in the history of smart home technology:

In the 1910's, the first emergence of technology in the household was the introduction of the machines such as vacuum cleaners, food processors, and sewing machines marked. Since then, advancements in technology have allowed more and more devices to enter the home space. Now, it is typical for homes in developed countries to have washing machines, toasters, dishwashers, and other similar appliances [47].

In the 1970s, the X10 communication protocol was developed as a way to allow electronic devices to communicate with each other over power lines. This paved the way for the development of home automation systems that could control lighting, appliances, and other devices [11-48-49].

In the 1980s and 1990s, several companies developed proprietary home automation systems, such as the C-Bus system from Clipsal and the LONWORKS system from Echelon Corporation [11-48-49].

In the early 2000s, wireless communication technologies such as Wi-Fi and Zigbee became more prevalent, allowing for greater flexibility in smart home systems. The introduction of the

Amazon Echo in 2014 and the subsequent rise of voice-activated assistants has also been a significant development in the smart home industry [11-48-49].

Today's Smart Homes: The modern smart homes focus on enhancing security measures and promoting environmentally-friendly living. They are designed to conserve energy and prevent unnecessary waste. The smart home technology also includes features that can alert homeowners of intruders regardless of their location. The latest home automation trends involve remote control via mobile devices, automated lighting and temperature control, scheduling appliances, receiving notifications through email or text messages, and accessing video surveillance remotely [47].

3. Technologies

A smart home is a complex system that consists of many devices connected to the Internet that serve different applications. The heterogeneity of these devices, with differing hardware, software, and communication technologies, poses a challenge for seamless integration .To address this, gateways act as a central hub to translate communication protocols, enabling devices to communicate with each other and external services. Cloud services provide additional computing power, storage, and advanced features like machine learning, optimizing and automating smart home systems. Gateways and cloud services are crucial components to enable smart home systems to provide a more streamlined user experience.The most relevant technologies for smart homes are grouped according to the following four categories [46]:

3.1 Sensors and Actuators:

Sensors and actuators are essential components of a smart home system. They work together to monitor and control various aspects of the home environment.The sensors are the fundamental source for information of end user activity. Through the artificial system, the information is collected by sensors are used to get a picture of the user current activity. In the last decades sensors has developed, and nowadays, there are small embedded low-power sensors exist, there are many different sensor types exist ranging from environmental sensors to biologic and medical sensors [50].

In smart homes, Sensors are devices that detect changes in the environment and send signals to the smart home system to trigger specific actions. Examples of sensors commonly used in smart homes include motion sensors, temperature sensors, humidity sensors, light sensors, and proximity sensors.

In general, when sensors are activated, their state changes. However, if a sensor only has a transmitter and no receiver, it is not possible to establish a communication link handshake. Therefore, the communication link must address this issue.

One way that deals with this is the event driven, where the sensor sends an event whenever it is activated. Such an approach uses power only when some change in the sensor state is discovered. However, if the event is lost, the network consumer is not notified again.

Another approach, which uses more power, is transmitting sensor status periodically e.g., every second. Such an approach increases the likelihood that a sensor state change will be detected, for the cost of consumed power and increased network load.

A third approach is using a polling approach where the consumer asks the sensor if some changes have been detected. In this approach events are only transmitted when the consumer needs it. But this requires that the sensor node has a receiver and not only a transmitter [50].

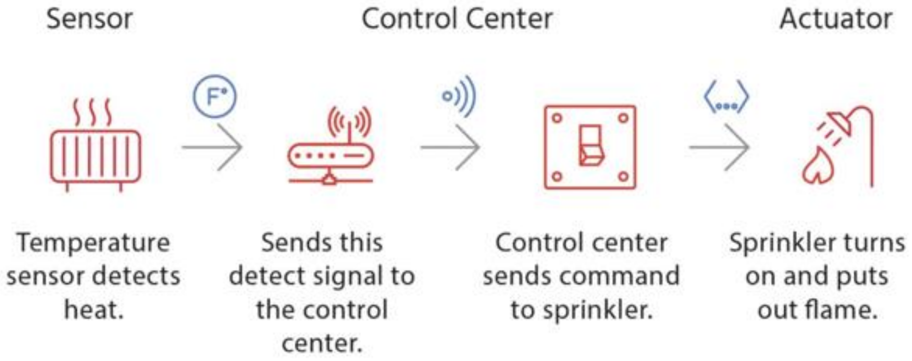


Figure II 3: Sensor to Actuator flow [Source: Bridgera.com]

In the other hand, actuators are devices that can carry out actions in the physical world based on the signals received from the sensors. They are responsible for managing different aspects of the smart home environment, such as controlling lighting, temperature, air conditioning, ventilation, and security systems. In essence, they enable the smart home to execute actions based on the data collected by the sensors. Some examples of actuators used in smart homes include smart locks, smart thermostats, smart cameras, and smart lighting [50].

3.2 Gateways:

Among the various devices included in a smart connected home, there are gateways which are used to gather, transform, and transmit data received from sensors, sensor hubs, and devices, some of which are intelligent. These gateways are responsible for facilitating the transfer of data to and from the Internet and the internal network, serving as the element accountable for linking with the external environment [46].

Usually, the smart connected home gateway is linked to the household's broadband router, which is then usually linked to the Internet via a wired connection such as an Ethernet cable to a cable or DSL modem. However, it is also feasible to integrate the gateway with a wireless/cellular radio, in which case the device can communicate directly with the Internet or a cellular carrier[46].

Modern gateways would typically offer the following functions:

3.2.1 Notifications *:*

Most gateways have a notification system in place that facilitates the routing of notifications. The aim of these notifications is to inform the end-user about the messages and alerts received from connected devices. Typically, these alerts can be prompts or pop-ups displayed through a mobile application installed on an end-user device, such as a smartphone, or through emails or SMS messages

3.2.2 Automation:

Some gateways support an automation or scheduling system that allows for the creation of different rules for the connected devices. For example, these systems can create rules based on time, such as sunrise or sunset rules, and other rules based on events and conditions. One such rule could be to turn on the lights automatically when movement is detected in the home.

3.2.3 Local Control:

Gateways often come equipped with a feature known as "local control." This feature enables manual control and execution of rules even when there is no internet or cloud service available. Typically, the interface for local control is provided in the form of a desktop or mobile application, or a local web browser. However, some gateways may also have a built-in display panel.

3.2.4 Cloud Service:

Most major gateways use cloud services, which provide remote device management capabilities. In addition to device management, the cloud is also used for a variety of other purposes such as storing home data, serving the user interface, and communicating with external systems like weather services, push messages, and SMS gateways.

3.2.5 Third-Party Integration:

as the number of smart home devices and standards continue to increase, it's becoming increasingly important to be able to connect devices that use different protocols. Modern gateways can help facilitate this by integrating with third-party services like Amazon Echo, Apple HomeKit, and "If This Then That" (IFTTT). Amazon Echo is a cloud-based, voice-activated platform that enables users to control their connected devices with their voice. Apple HomeKit is a platform designed to facilitate communication between different devices, often with the help of compatible bridges or hubs. IFTTT is a web-based service that allows users to create "recipes" that enable multiple services to work together, program devices to run routines, react to triggers, or send commands to other devices in the home.

3.2.6 API/SDK:

Most hubs usually come equipped with an Application Programming Interface (API) and/or a Software Development Kit (SDK). These tools allow developers to interface with the hub and build custom functionality. For instance, Amazon's Alexa smart home ecosystem features a Smart Home Skill API, which can be used to create capabilities such as the ability to control cloud-connected devices, play music, answer questions, and provide weather forecasts. Another example is a Smart TV App Store that includes an SDK, enabling users to control the screen, manage apps, and download and upload content via the network.

3.3 End User Client Devices:

End user client devices play an important role in the technology ecosystem of a smart home. Smart homes use a network of devices and sensors to automate and control various systems and appliances within the home, such as lighting, heating and cooling, security, entertainment, and more.

End user client devices are the primary means by which users interact with smart home systems, also end-users can control, monitor, and access the functionality of the smart connected home. Moreover, residents can take advantage of mobile devices to check the main tasks and schedule of the smart home system, access system information, and transmit instructions for immediate execution. As an example, end-users could use their smartphones, tablets, desktop computers, or specialized smart remote-control systems to issue commands such as unlocking their home doors or remotely monitoring their children through available applications. Additionally, these devices can recognize user input through speech, hand gestures, and user tracking.

End user client devices also play a key role in providing feedback and monitoring for smart home systems. For example, a smart thermostat can send temperature readings to a user's smartphone, allowing them to monitor and adjust the temperature remotely. Smart security cameras can also send alerts to a user's smartphone when motion is detected or when someone enters the home.

Generally, end user client devices are an important technology within smart homes, providing a convenient and intuitive interface for users to control and monitor various systems and appliances within the home.

3.4 Cloud Services:

Cloud-based solutions for smart homes utilize cloud computing technology to facilitate remote management and access of various devices and services within a household. These systems can be implemented through private, public, or hybrid backend infrastructures (clouds) and may include service delivery or application platforms that are managed and operated by device manufacturers and service providers [51].

Cloud services in smart homes offer additional features such as data storage, analytics, and integration with third-party applications. Homeowners can leverage these services to access and manage their smart home devices from anywhere with an internet connection, using a smartphone or other internet-enabled device.

Commonly used cloud services for smart homes include (Figure 4):

3.4.1 Infrastructure as a Service (IaaS):

This cloud service provides virtualized computing resources, such as servers, storage, and networking, which can be used to store and process data from smart home devices. IaaS enables smart home owners to adjust their computing resources as required, without investing in expensive hardware. IaaS storage not only involves storing data with the IaaS provider, but it also allows for global accessibility of the data via the internet. Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform are examples of IaaS providers [16-52].

3.4.2 Platform as a Service (PaaS):

PaaS offers a platform for developers to build, test, and deploy smart home device applications. PaaS providers typically offer a range of pre-built tools and frameworks that developers can use to create and customize their applications. Heroku, Google App Engine, and Microsoft Azure are examples of PaaS providers [16-52].

3.4.3 Software as a Service (SaaS):

SaaS provides access to software applications that enable the management and control of smart home devices, such as smartphone apps and web-based dashboards. in addition, the user is not required to store, install, or manage the application. Rather, they only need an internet connection to reach the rented service offered by the SaaS provider on the cloud.

SaaS providers usually charge a subscription fee on a monthly or yearly basis, and users can access the software from anywhere with an internet connection. Nest, Ring, and Philips Hue are examples of SaaS providers for smart homes [16-52].

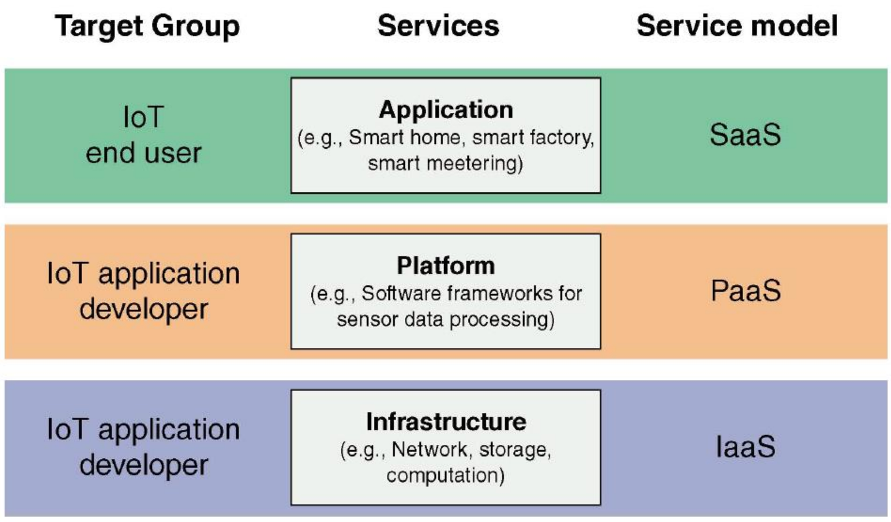


Figure II 4: Cloud computing service models.

The selection of cloud services for a smart home depends on the specific needs and requirements of the homeowner or developer. Different cloud services offer varying benefits and features, and smart home owners may decide to use a combination of services to fulfill their needs.

3.5 Integration frameworks (platforms):

Integration frameworks or systems are technologies that simplify and enhance compatibility and cooperation among the diverse intelligent connected home environment. Instances of current frameworks that are created and maintained by businesses include Apple HomeKit, Google Brillo/Weave, Allseen AllJoyn, Samsung SmartThings, and Amazon Alexa. These smart home systems also provide a programming framework for developers to design applications that deliver smart home benefits. Furthermore, there are also free home automation systems constructed on open protocols and under open-source license. Three such systems are openHAB, OpenMotics, and Domoticz. Generally, these connected home systems are favored among the do-it-yourself (DIY) communities. DIY users install and set up the intelligent home devices themselves rather than relying on professionals [46].

Smart home integration platforms can help homeowners save energy and reduce utility bills by providing insights into their energy consumption patterns. With the help of smart meters and sensors, the platform can track energy usage in real-time and provide suggestions on how to optimize consumption

3.6 Communication Protocols and Models:

Smart connected homes rely heavily on the Internet protocol (IP) as it is the foundation of the IoT. The IP's easy interoperability, widespread adoption, pervasive nature, and recent research into lightweight interfaces make it essential for the success of smart connected homes. Nonetheless, it's worth noting that non-IP-based devices can also be integrated into a smart home system.

Smart homes that are connected to the internet use a variety of communication protocols, including wired options like X10, HomePlug, and LonWorks, and wireless options such as Wi-Fi, Z-Wave, and ZigBee. These homes may rely on either open standard like ZigBee or proprietary protocols like Z-Wave. Additionally, smart homes may use short-range communication protocols like NFC, RFID, and 6LowPAN or long-range protocols such as Sigfox, NWave, and LoRaWAN.

Differentiating between low-level and high-level protocols is crucial in understanding how devices communicate with each other. high-level protocols, such as MQTT, CoAP, and XMPP, are used to transmit sensor data or receive remote commands through the home gateway. correspondingly, Low-level protocols, such as ZigBee, are mainly used for device-to-device networking within the house and, for long-range protocols like 4G, to communicate with service providers. Furthermore, there are domain-specific protocols at the high-level, like the Smart Energy Profile 2.0, that connect home energy devices to the smart grid, making it possible to manage and optimize energy usage.

In the other hand, in a smart connected home, various devices able to connect with each other via heterogeneous communication protocols and by using different models. There are usually three communication models used in IoT environments, including the smart connected home: device-to-device, device-to-cloud, and device-to-gateway.

3.6.1 Device-to-device:

Device-to-device communication (Figure 5) covers two or more devices which are connected directly and communicate with each other using protocols like Bluetooth, Z-Wave, and ZigBee. and without an intermediary application server. Many home automation systems use this communication model (device-to-device) to transfer small amounts of data between devices at alow data rate. Devices like light bulbs, thermostats, and door locks can use this communication method to send and receive small pieces of information. As an example of device-to-device model is wearable IoT devices like a heart monitor paired to a smartwatch, where data doesn't necessarily have to be shared with multiple people[53-54].

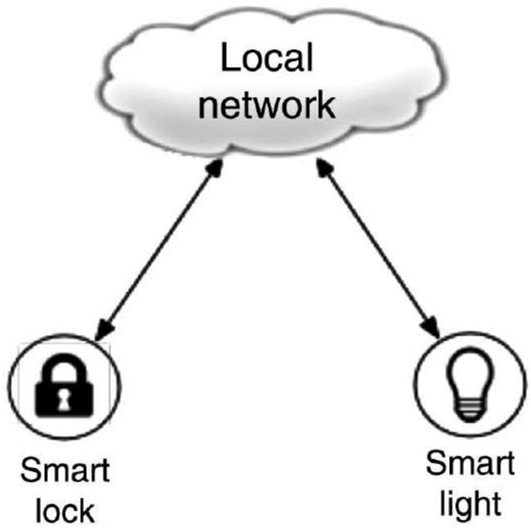


Figure II 5: Device-to-device model

3.6.2 Device-to-cloud:

In a communication model where devices connect to the cloud (Figure 6), the IoT gadget directly links to an internet-based cloud solution, such as an application service supplier, to exchange information and manage message traffic. This method often leverages existing modes of communication, such as conventional wired Ethernet or Wi-Fi connections, to establish a bond between the device and the IP network, which finally links up with the cloud service.This conversation version is utilized by numerous famous purchaser IoT gadgets just like the Nest Labs Learning Thermostat. In the case of the Nest Learning Thermostat, the device sends data to a cloud database where the data can be used to analyze home energy consumption.Further, this cloud connection permits the person to reap faraway get admission to their thermostat via a smartphone or web interface, and it additionally helps software program updates to the thermostat [53-54].

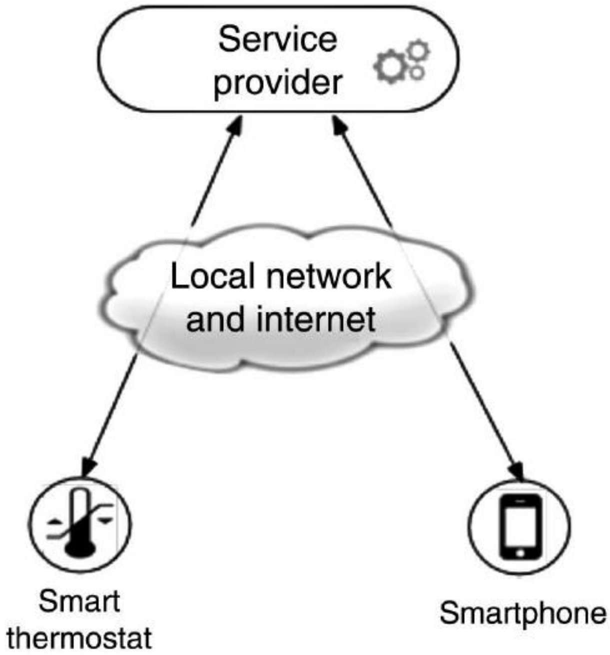


Figure II 6: Device-to-cloud model.

3.6.3 Device-to-Gateway:

The device to gateway(Figure 7)also known as application layer gateway (ALG) model. The ALG service consider as a channel which allow the IoT device to connect via it to reach a cloud service. In simple terms, in the local gateway device there is an application software which function like an intermediary between the cloud service and the device. In addition, acts as a provider of security and other functions like a translator of protocol or data.There is other model of device to gateway in home automation application is the emergence of “hub” devices which are devices that works as a local gateway between a cloud service and individual IoT devices. However, they can bridge the interoperability gap between devices themselves.An instance of a stand-alone gateway device that utilizes both Z-Wave and Zigbee transceivers to communicate with their respective devices is the Smart Things hub. This hub subsequently links up with the Smart Things cloud service, which enables users to control and access their devices via a smartphone application and an internet connection.

According to an IETF Journal article, the device-to-gateway approach is particularly beneficial from a technical standpoint when it comes to ensuring interoperability with non-IP devices. This model is often implemented for integrating IPv6-only devices into systems that have legacy IPv4-only devices and services. Essentially, the device-to-gateway communication model is frequently utilized to assimilate new smart gadgets into an existing system that lacks native inter-operability with these devices. However, a potential drawback of this approach is that the development of application-layer gateway software and system adds complexity and cost to the overall system[53-54].

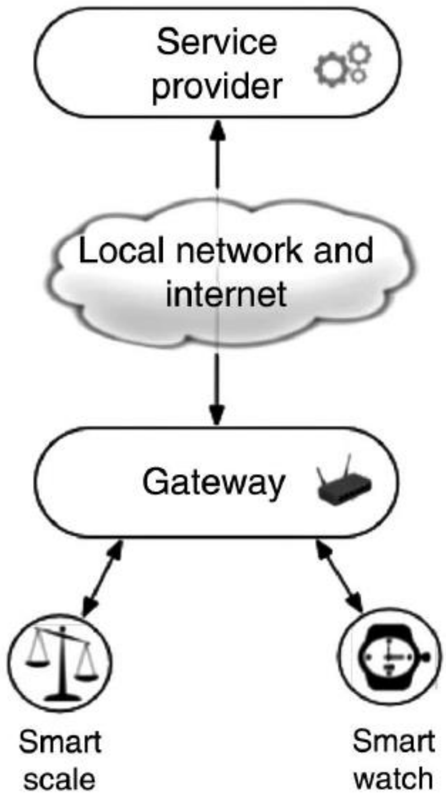


Figure II 7: Device-to-gateway model.

4. Smart Home Architectures:

There are two primary architectural styles of smart connected homes: centralized and distributed. This differentiation is based on spatial organization, that is, the physical placement of intelligence, service provisioning, and cooperation between devices to achieve a common goal [46].

4.1 Centralized:

The centralized architecture is a way of managing data in which all information is accessed through a single central entity. This entity can be a local gateway that stores data and communicates with IoT devices, or the cloud can serve as the central hub to collect data from smart devices within the network. End-users connect to the central entity via the internet to access services like cloud-based voice services from Amazon Echo, Google, and Apple.

However, implementing effective privacy measures is a challenge with the centralized architecture. Since the central entity is where the "intelligence" is located, edge devices have limited control over the data they generate and process. The central entity decides which data elements to process and share, making it difficult to implement flexible privacy measures.

Another issue is that the system is vulnerable to single points of failure since all data flows through the central entity. A failure in the central entity can lead to downtime and a loss of services for end-users.

Despite these challenges, the centralized architecture is still a popular choice for managing data in IoT devices due to its simplicity and ease of implementation. However, there is a growing trend towards decentralization to address these challenges and provide greater flexibility, privacy, and resilience in managing data in IoT devices.

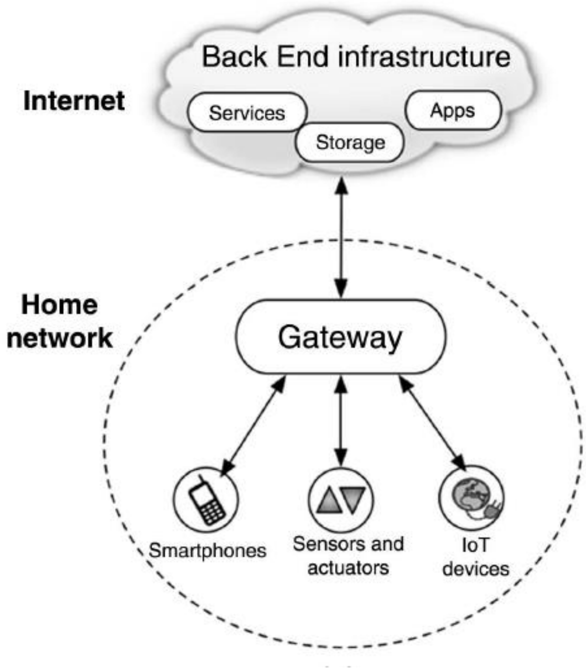


Figure II 8: Centralized architectural models

4.2 Distributed:

The distributed approach to managing data in IoT devices involves each device being self-sufficient and capable of handling all aspects of data processing, retrieval, assimilation, and sharing with other entities. Unlike the centralized approach where data flows through a single point, the distributed model is more like a peer-to-peer system where information is only exchanged when needed. In this model, connected devices are considered smart on their own and make decisions locally, without relying heavily on the Internet as is the case with the centralized architecture. However, the Internet may still be needed for coordination and analysis purposes.

For example, a connected oven may detect the presence of connected light bulbs and flash them when it's done cooking. However, due to market fragmentation, this approach can be challenging to implement in practice. Distributed architectures can leverage a service-oriented or a mobile agent approach, such as the Jini platform used by ProSyst Software company to provide products for controlling household devices.

One of the challenges of the distributed model is implementing network security, which is more complex than in the centralized approach. Any smart device can connect with any node at any time, without prior knowledge of each other. Additionally, some devices may be heavily constrained, making key management a significant challenge.

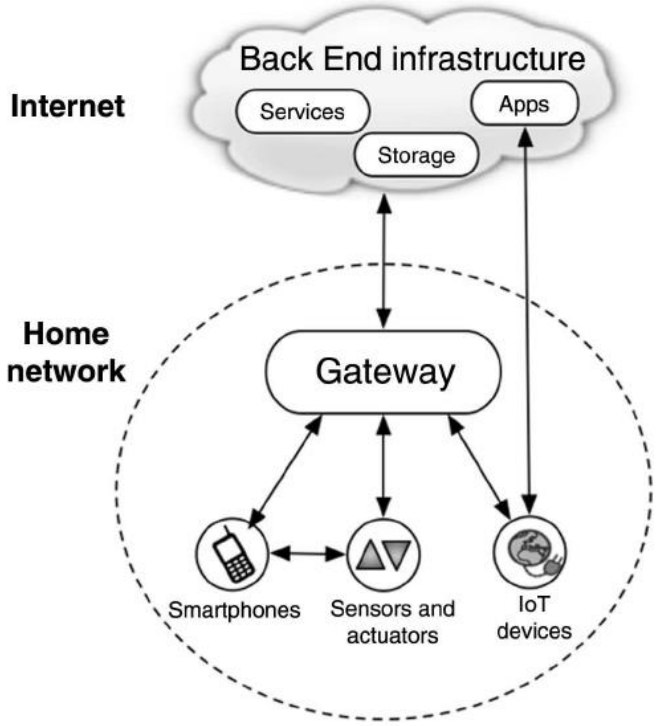


Figure II 9: Distributed architectural models.

In reality, it is also possible to have hybrid or decentralized architectures combining the characteristics of both [55].

Conclusion :

In conclusion, this chapter highlighted on smart homes as one of the most benefit of IoT in human life where provided a comprehensive overview of the definitions, history, and technologies used in smart homes. The chapter discussed also how smart homes have developed over time to become a fundamental aspect of the Internet of Things (IoT), transforming traditional homes into intelligent and interconnected living spaces. Furthermore, the chapter focused various technologies used in smart homes, including sensors, actuators, and communication protocols that enable different smart devices to interact with one another. and also covered different types of smart home architectures, including centralized, decentralized, and hybrid architectures, each with its unique benefits and drawbacks.

In the end, the chapter demonstrated that smart homes offering important benefits for homeowners, including increased comfort, convenience, and control. As technology continues to advance, it is evident that smart homes will become more prevalent, transforming the way we live, work and interact with our environment.

CHAPTER 3

NETWORK DESIGN

Introduction:

In recent years, the concept of a smart home has captivated homeowners and technology enthusiasts. As discussed in the previous chapter, smart homes utilize advanced technology to enhance our lives by making them more convenient, efficient, and secure. It's like having a home that can think and respond to our needs. A smart home is composed of various devices that form its foundation. These devices include microcontrollers, which are small computers, sensors that can detect motion or temperature, and actuators that can control the movement or on/off status of specific items.

These devices collaborate to create a network that connects different systems within the home. This connectivity enables us to effortlessly manage aspects such as lighting, temperature, and security using our smartphones or voice commands.

In this chapter, our main focus is on creating a smart home network using Cisco Packet Tracer 8.1, a powerful simulation tool. We will take advantage of this tool to thoroughly test the interactions between different devices in the network. Our goal is to design and configure a cohesive ecosystem of devices

1. Cisco Packet Tracer:

1.1 What is a CISCO Packet Tracer?

Cisco Packet Tracer is a free software that enables users to practice and enhance their network configuration and troubleshooting skills on desktop or laptop computers. It serves as a network simulator, allowing users to simulate networks and gain hands-on experience without physical equipment. Apart from networking, Packet Tracer also supports learning and practicing Internet of Things (IoT) and Cybersecurity skills. Users can choose to build networks from scratch, use pre-built sample networks, or complete lab assignments. While it does not replace physical routers, switches, firewalls, or servers, it offers numerous benefits [59].

Developed by Cisco, Packet Tracer is a versatile tool that provides simulation capabilities for both simple and complex networks. Its primary purpose is to help students gain practical networking experience and acquire specific skills related to Cisco technology. The tool incorporates built-in protocols and networking support, making it useful for various technical concepts, such as CCENT and CCNA.

Packet Tracer serves multiple purposes, allowing students to work on assignments individually or in teams. It also assists engineers in testing applications before implementation. Network support engineers can deploy and test changes using Packet Tracer, ensuring their effectiveness before moving to production. Additionally, engineers can easily add or remove simulated network devices using the intuitive drag-and-drop user interface or the command line interface.

Overall, Cisco Packet Tracer provides a convenient and effective platform for learning and practicing networking skills, IoT concepts, and cybersecurity. It offers simulation capabilities, empowers students and engineers to test and validate network changes, and facilitates various technical concepts and networking systems [60].

1.2 GUI and CLI Configuration in Cisco Packet Tracer [59].

Packet Tracer also provides a variety of tabs for device configuration including the following:

- Physical
- Config
- CLI
- Desktop
- Services

The tabs that are shown depend on the device you are currently configuring.

1.2.1 Physical Tab

The Physical tab provides an interface for interacting with the device including powering it on or off or installing different modules, such as a wireless network interface card (NIC).

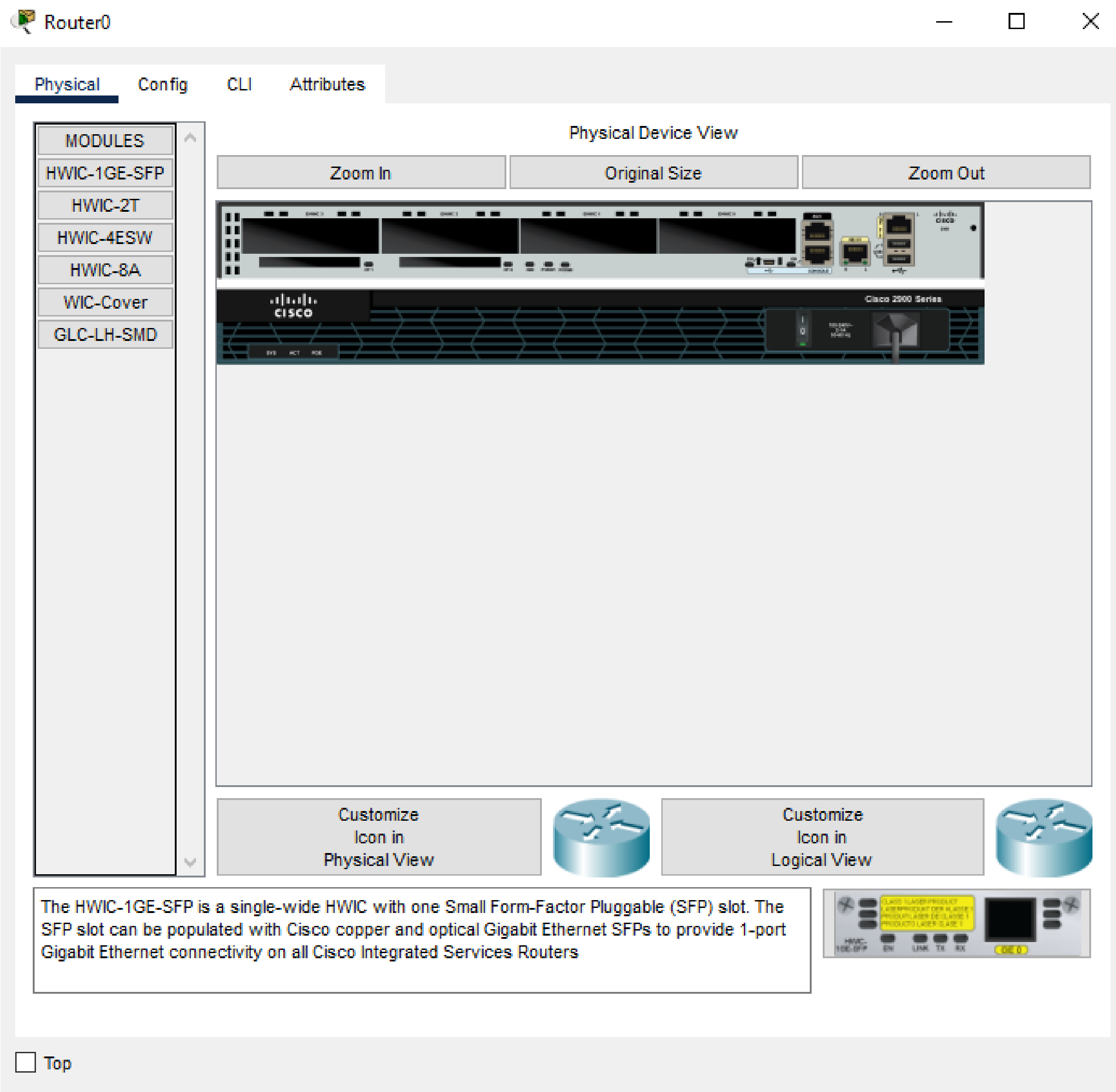


Figure III 1: Physical Tab.

1.2.2 Config Tab

For intermediate devices such as routers and switches, there are two ways to access device configurations. Configurations can be accessed via a Config tab, which is a Graphical User Interface (GUI). Configurations can also be accessed using a command line interface (CLI).

The Config tab does not simulate the functionality of a device. This tab is unique to Packet Tracer. If you don’t know how to use the command line interface, this tab provides a way to use a Packet Tracer-only GUI to configure basic settings. As settings are changed in the GUI, the equivalent CLI commands appear in the **Equivalent IOS Commands** window. This helps you to learn the CLI commands and the Cisco Internetwork Operating System (IOS) while you are using the Config tab.

For example, in the figure, the user has configured **MyRouter** as the name of the device. The Equivalent IOS Commands window shows the IOS command that achieves the same results in the CLI.

In addition, device configuration files can be saved, loaded, erased, and exported here.

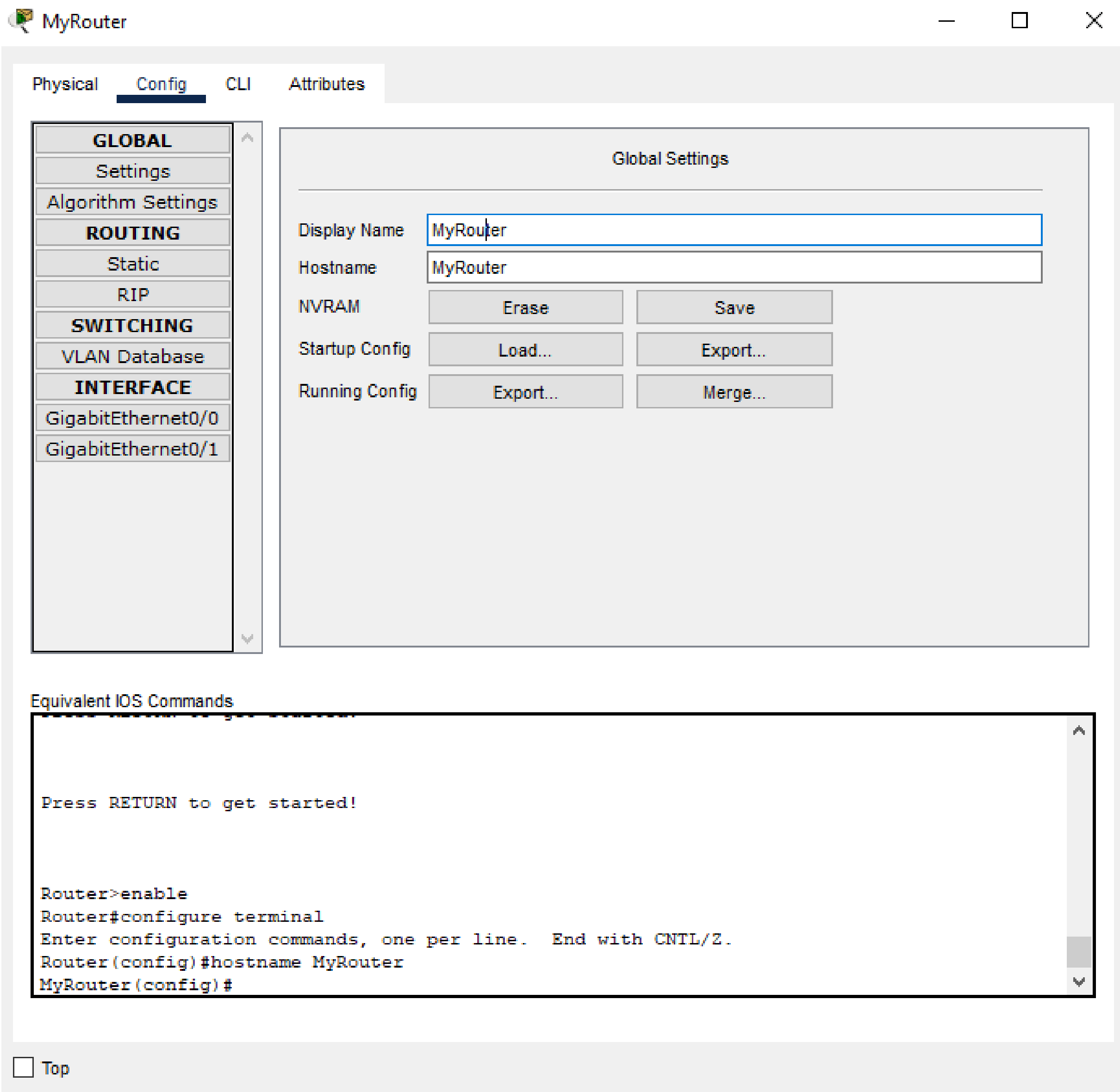


Figure III 2: Config Tab

1.2.3 CLI Tab

The CLI tab provides access to the command line interface of a Cisco device. Using the CLI tab requires knowledge of device configuration with IOS. Here, you can practice configuring Cisco devices at the command line. CLI configuration is a necessary skill for more advanced networking implementations.

Note: Any commands that were entered from the Config tab are also shown in the CLI tab.

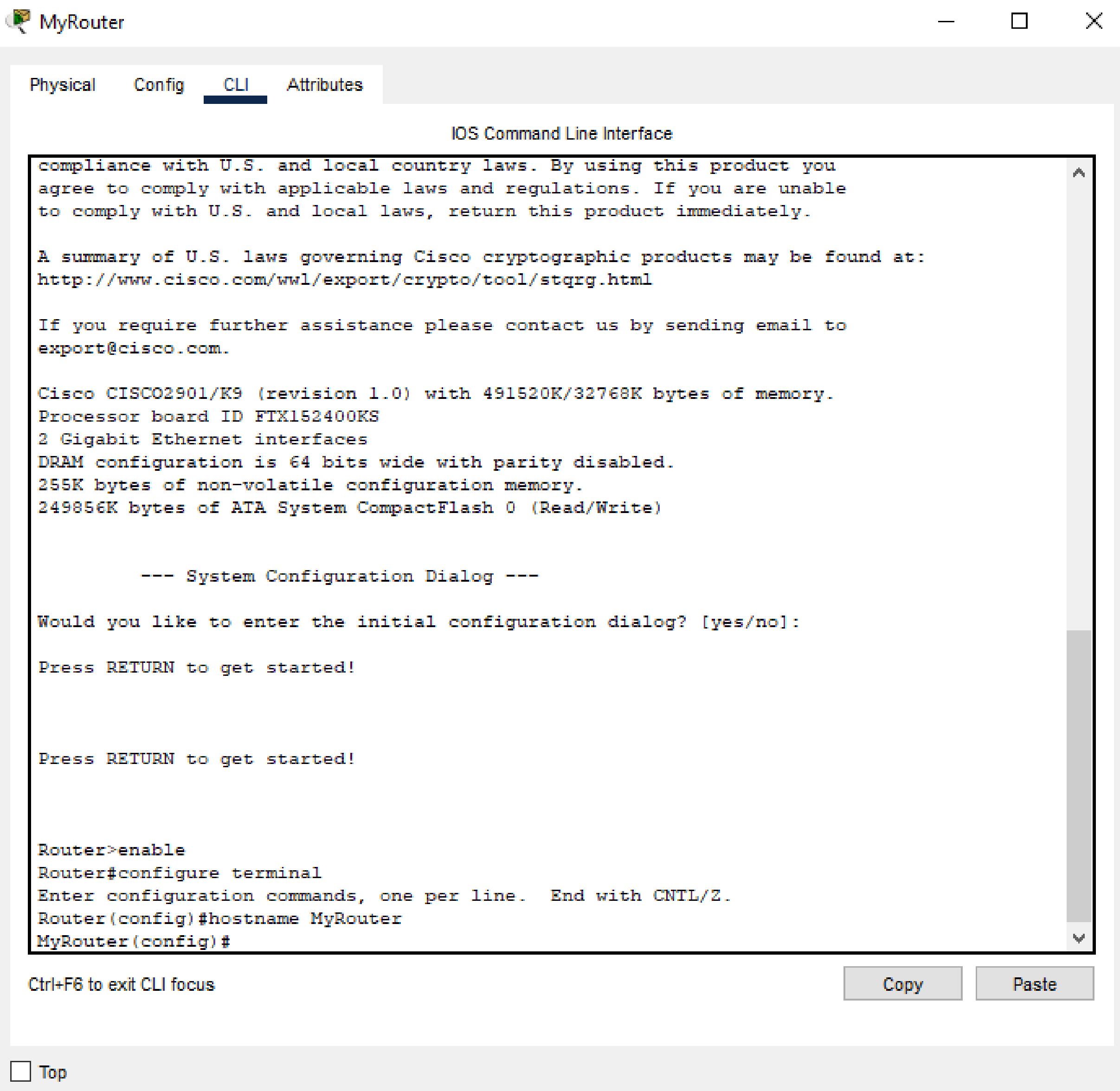


Figure III 3: CLI Tab

1.2.4 Desktop Tab

For some end devices, such as PCs and laptops, Packet Tracer provides a desktop interface that gives you access to IP configuration, wireless configuration, a command prompt, a web browser, and other applications.

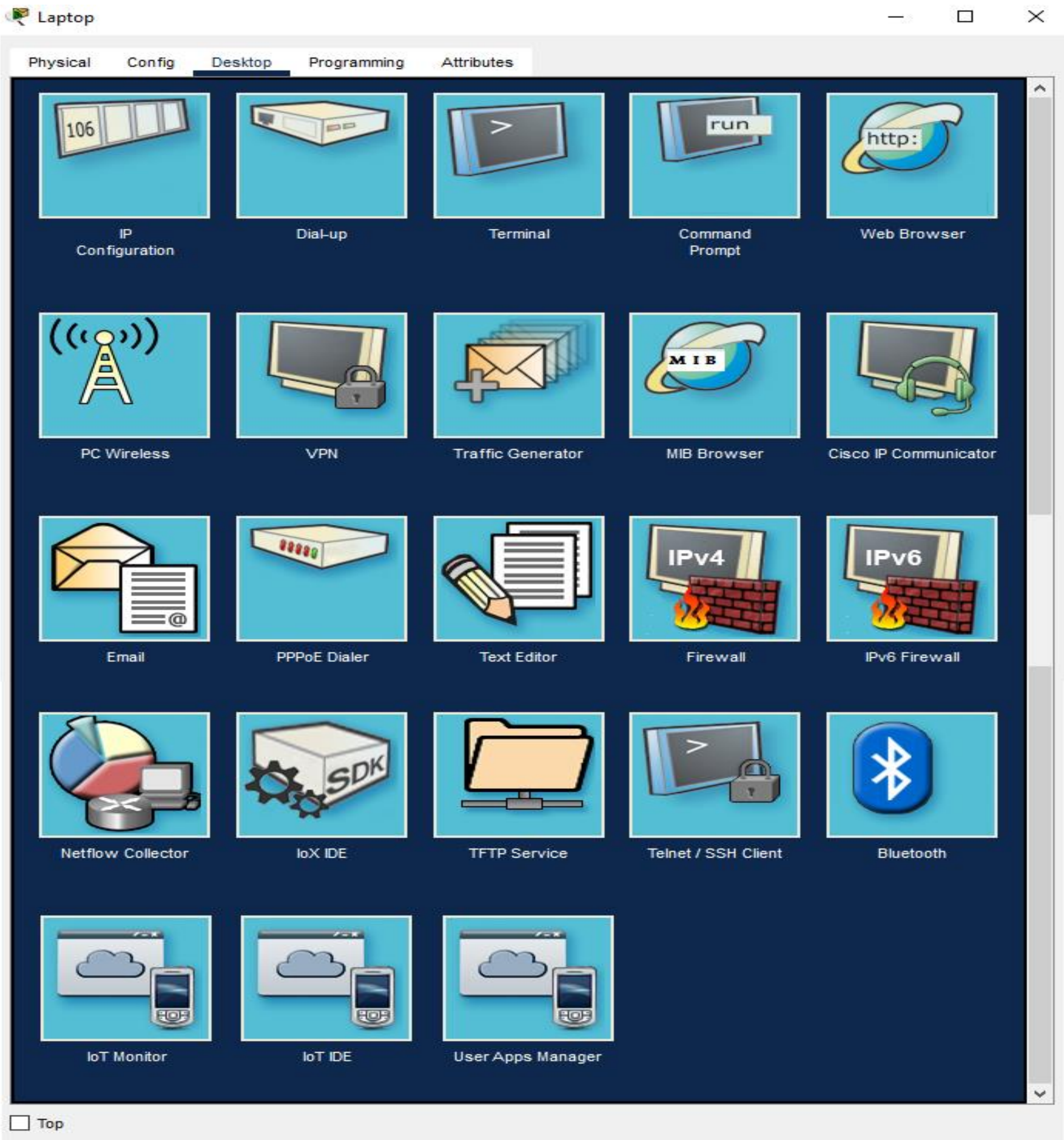


Figure III 4: Desktop Tab

1.2.5 Services Tab

A server has all of the functions of a host with the addition of one more tab, the Services tab. This tab allows a server to be configured with common server processes such as HTTP, DHCP, DNS, or other services, as shown in the figure.

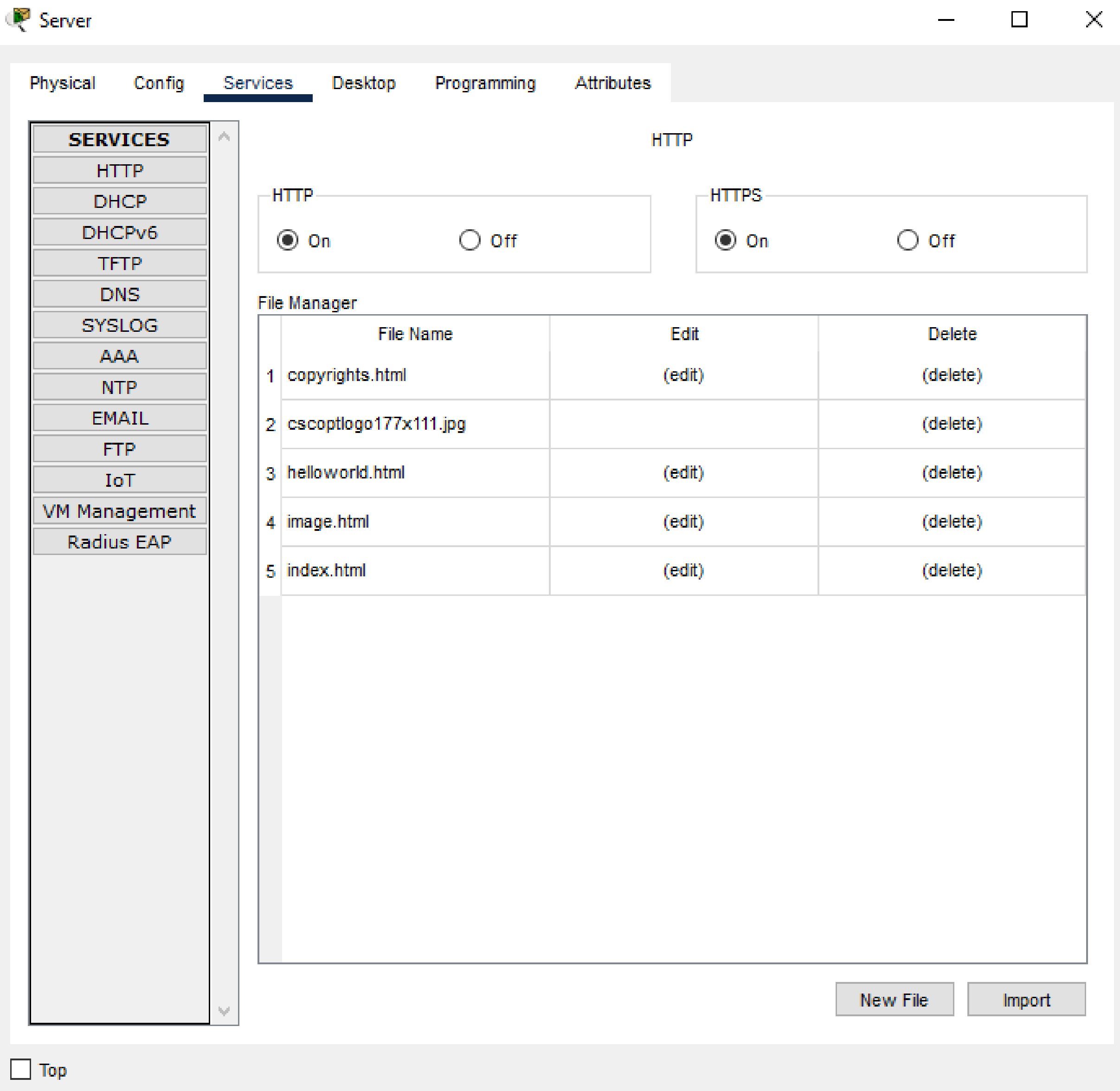


Figure III 5: Services Tab

2. Methodology of the design:

In order to design a network for an Internet of Things (IoT) smart home, we will create a topology consisting of four main components:

2.1 The Smart Home Network:

This network will connect various devices wirelessly through a HomeGateway, enabling communication between devices with different protocols. It will also include small computers like Microcontrollers (MCUs), which will facilitate monitoring and control of sensors and actuators. These MCUs can be programmed using languages such as Python, JavaScript, or Visual. Additionally, a PC, tablet, smartphone, or laptop will be required to control the different smart devices over the internet.

2.2 Internet Cloud (WAN):

The Internet Cloud represents the segment of the network that connects to the home gateway via a cable modem. Its purpose is to provide internet connectivity to the devices within the Internet of Things (IoT) ecosystem.

2.3 The service provider (SP) network:

The service provider (SP) network component focuses on the IoT server responsible for registering all connected devices, enabling enhanced functionalities for the Internet of Things. It also serves as the host for the IoT homepage, allowing home users to connect. Additionally, a DNS server is necessary to translate the URL of the IoT homepage into the respective IP address of the IoT server.

2.4 The 3G/4G provider network:

The 3G/4G provider network constitutes the final segment of the topology, enabling smartphones to connect to cell towers for internet access. This connection allows users to remotely access their devices.

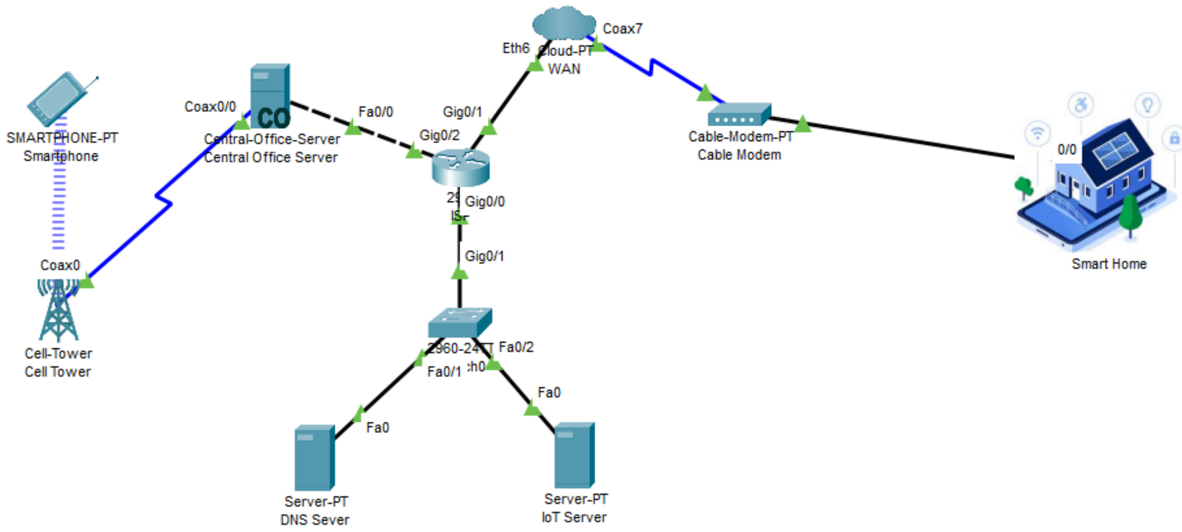


Figure III 6: smart home network topology.

3. Configuration the parts of the network:

Configuring the various parts of the network involves setting up and optimizing each component to ensure proper functionality. Here is a breakdown of the configuration tasks for the different network parts mentioned:

3.1 The service provider (SP) network:

3.1.1 Configuration of Router ISP:

The router is connected to the service provider network through a switch using the GigabiteEthernet interface **Gig0/0**. On the other hand, it is connected to the Central-Office Center via the **Gig0/2** interface. Additionally, it is connected to the Wide Area Network (WAN) through the **Gig0/1** interface. The CLI commands tab on the router contains the following configuration:

3.1.2 Assigning hostname and Ip address for ISP router:

```
Router>enable
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname ISP
ISP(config)#int g0/2
ISP(config-if)#ip address 200.165.14.225 255.255.255.224
ISP(config-if)#no shutdown

ISP(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/2, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/2, changed state to up

ISP(config-if)#int g0/1
ISP(config-if)#ip address 200.165.13.225 255.255.255.224
ISP(config-if)#no shutdown

ISP(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

ISP(config-if)#int g0/0
ISP(config-if)#ip address 10.20.30.1 255.255.255.0
ISP(config-if)#no shutdown

ISP(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

ISP(config-if)#
```

Figure III 7: Cofiguration of Router ISP

3.1.3 Configurating DHCP server for cell and WAN:

```
ISP(config-if)#
ISP(config-if)#
ISP(config-if)#exit
ISP(config)#ip dhcp excluded-address 200.165.14.225 200.165.14.229
ISP(config)#ip dhcp pool cell
ISP(dhcp-config)#network 200.165.14.224 255.255.255.224
ISP(dhcp-config)#default-router 200.165.14.225
ISP(dhcp-config)#dns-server 10.20.30.254
ISP(dhcp-config)#exit
ISP(config)#ip dhcp excluded-address 200.165.13.225 200.165.13.229
ISP(config)#ip dhcp pool WAN
ISP(dhcp-config)#network 200.165.13.224 255.255.255.224
ISP(dhcp-config)#default-router 200.165.13.225
ISP(dhcp-config)#dns-server 10.20.30.254
ISP(dhcp-config)#exit
ISP(config)#
```

Figure III 8: Configuration DHCP

3.2 The Switch:

The switch is self-configured; it is used just to connect theGig0/1 interface from Router ISP to multiple devices (DNS Server and IoT Server).

3.3 IoT Server:

The IoT Server serves as a central hub for remotely connecting and accessing IoT devices through a web interface using a computer or smartphone. It allows users to control and manage all registered smart objects via the hosted web interface.

To ensure consistent connectivity, the IoT server is configured with a static IP address. This allows all smart devices to connect to the server using the same IP address.

During the registration process of devices on the IoT server, it is necessary to provide the server's IP address along with a pre-existing username and password. These credentials are used to authenticate and authorize access to the devices on the IoT server. By using the specified username and password, users can securely interact with their registered devices through the web interface hosted on the IoT server.

3.3.1 Network configuration

For interface FastEthernet0/2, it is connected to the network “10.20.30.0” with the subnet mask 255.255.255.0. The default gateway address is “10.20.30.1”. We give it “10.20.30.253” as an address IP. We go to settings in the same TAB and add the DNS and gateway IP addresses.

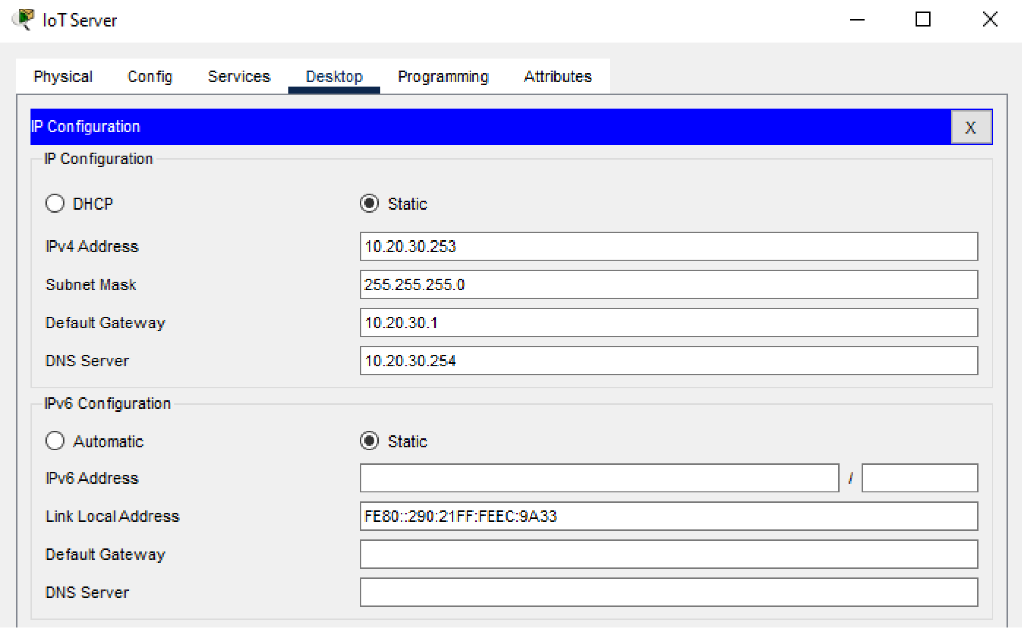


Figure III 9: IoT service network configuration.

3.3.2 Setting up the IoT service

To make this server work as an IoT server, the IoT service has to be enabled from services TAB. Then, an account must be created in the server, by starting a web browser from the desktop of the server itself and access the IoT server address “10.20.30.253” or the domain name “iotserver.dz”. Since we do not already have an account, we need to click “sign up now”.

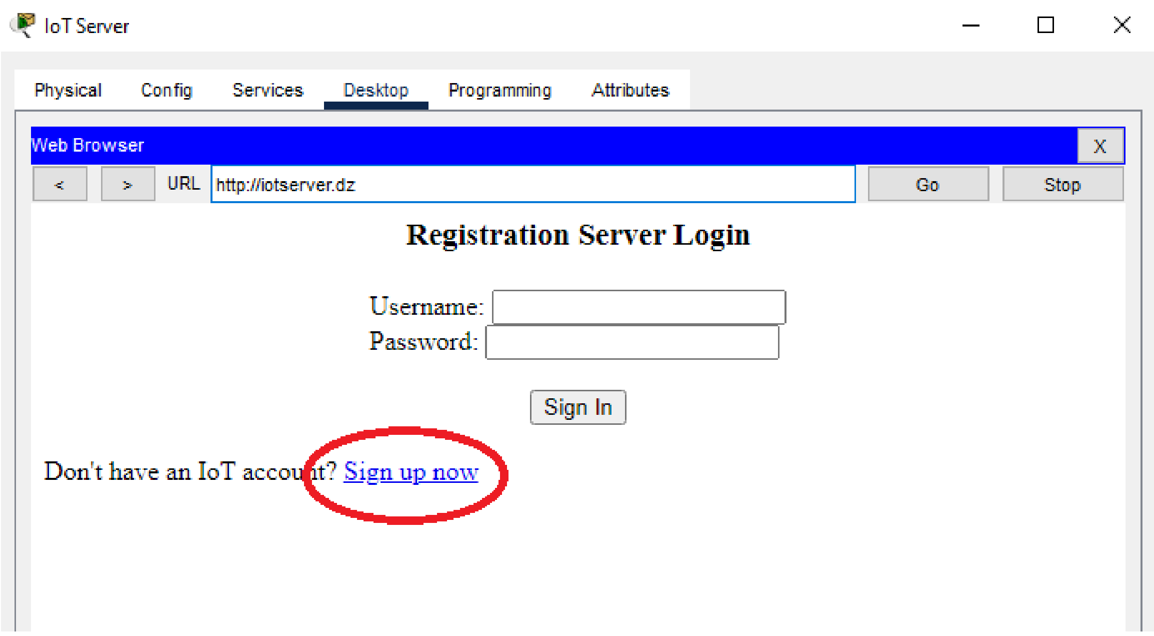


Figure III 10: Setting up the IoT service

After clicking on "Sign up Now," you will be prompted to enter your desired username and password. For our project, the username and password are both set as "aek"

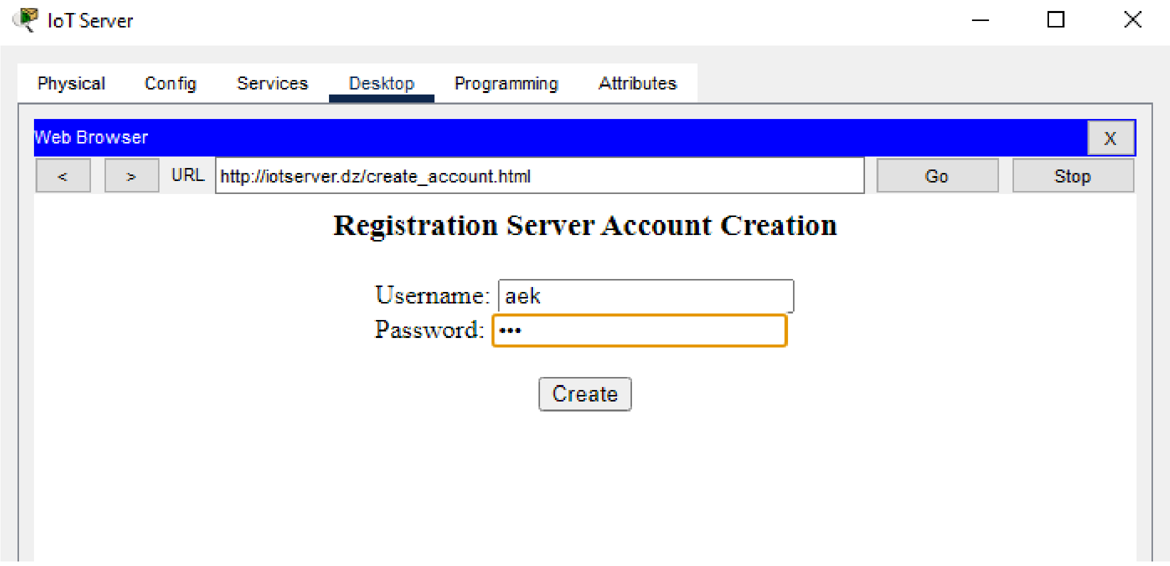


Figure III 11: Creation a server account

You can also verify the username and password by navigating to the "Services" tab and clicking on "IoT" as indicated in the figure below.

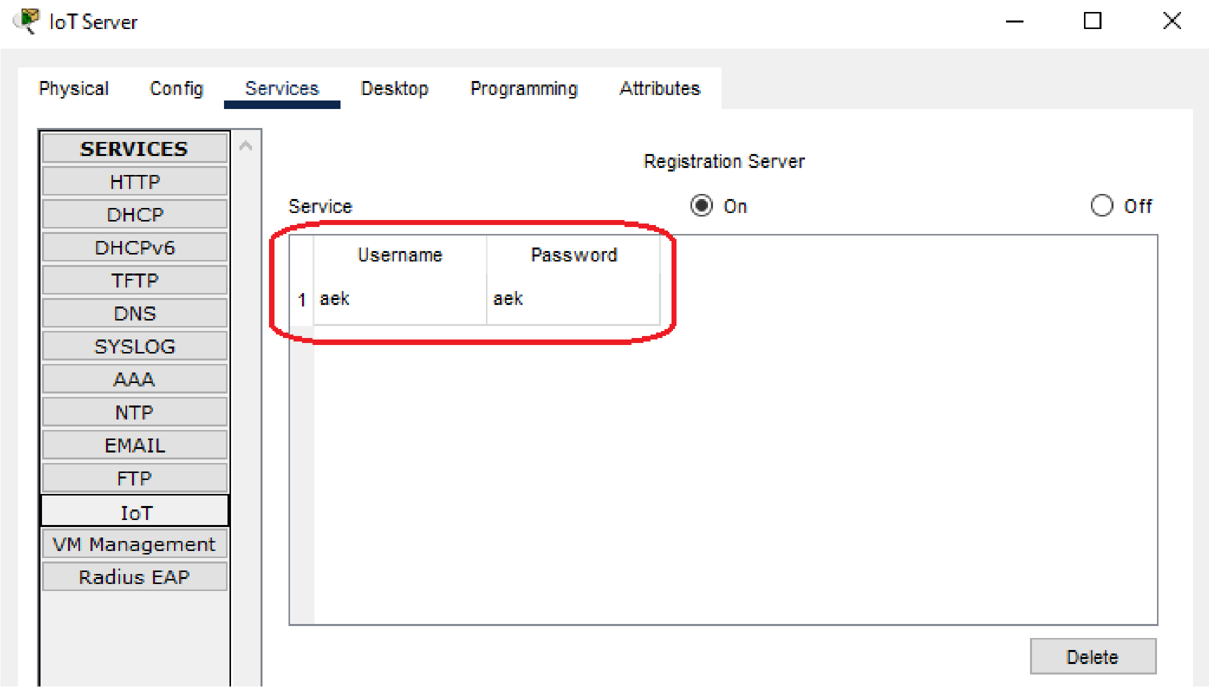


Figure III 12: Registration server

To access our IoT account, follow these steps:

1. Open a web browser on any device connected to the internet.
2. Enter either the IP address of the server, "10.20.30.253," or the domain name, "iotserver.dz," in the browser's address bar.
3. Enter the appropriate username and password associated with your IoT account.
4. Click on the "Sign In" button to log into the IoT account.

3.4 DNS server:

DNS servers play a crucial role in bridging the gap between domain names and IP addresses, ensuring seamless internet connectivity. They are considered an essential component of the internet infrastructure. Their main function is to translate IP addresses into domain names, allowing users to access websites, send emails, and engage in internet activities using user-friendly domain names instead of complex IP addresses. DNS servers maintain a database containing domain names and their corresponding IP addresses, facilitating the resolution process. This database enables efficient and reliable mapping between domain names and IP addresses, enhancing the overall functioning of the internet.

In order to configure the DNS server, we must assign a static IPv4 address to the server. The server is connected to the network "10.20.30.0" with a subnet mask of 255.255.255.0, using the FastEthernet0 interface. The default gateway address is "10.20.30.1". Consequently, we set the IP address of the server as "10.20.30.254". Next, we navigate to the settings within the same tab and input the DNS and gateway IP addresses.

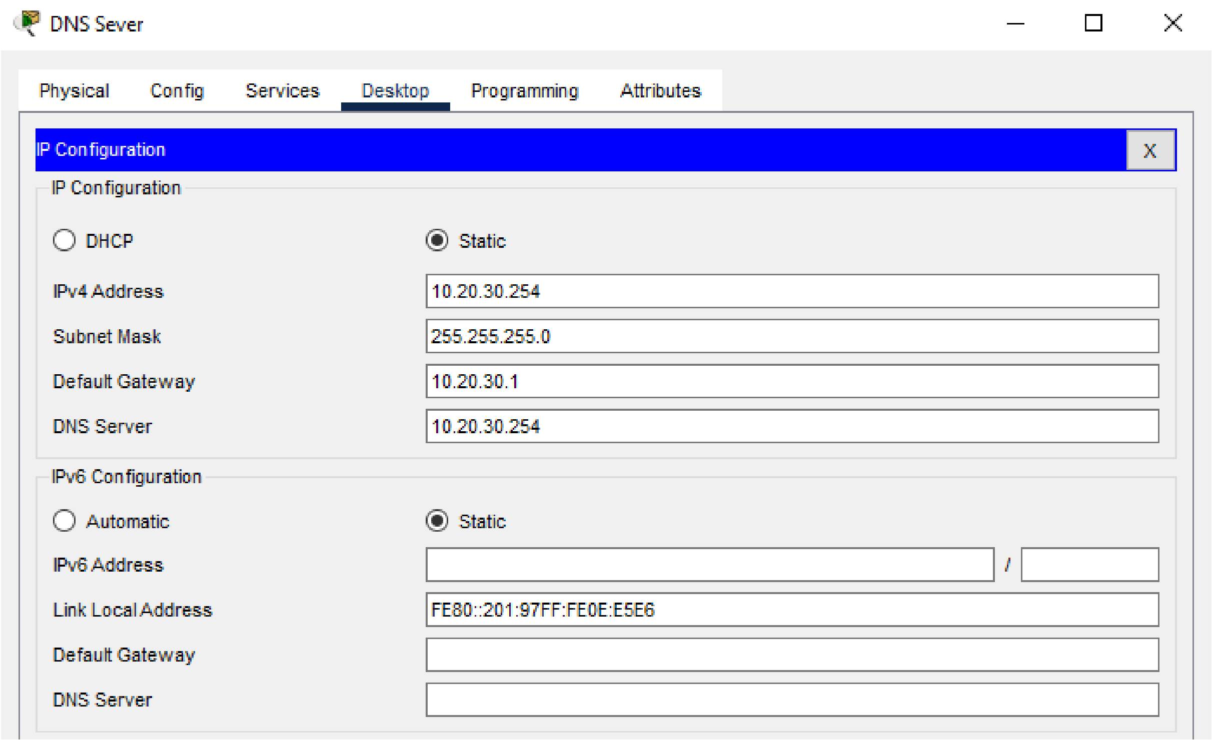


Figure III 13: DNS network configuration.

To associate the IP address "10.20.30.253" of the IoT server with the domain name "iotserver.dz", we navigate to the server's Services TAB. Within the DNS settings, we enable the DNS service. Following that, we enter the domain name and the corresponding IP address, as depicted in the provided figure. This configuration establishes the translation between the specified IP address and domain name, enabling the server to be accessed using the user domain name "iotserver.dz".

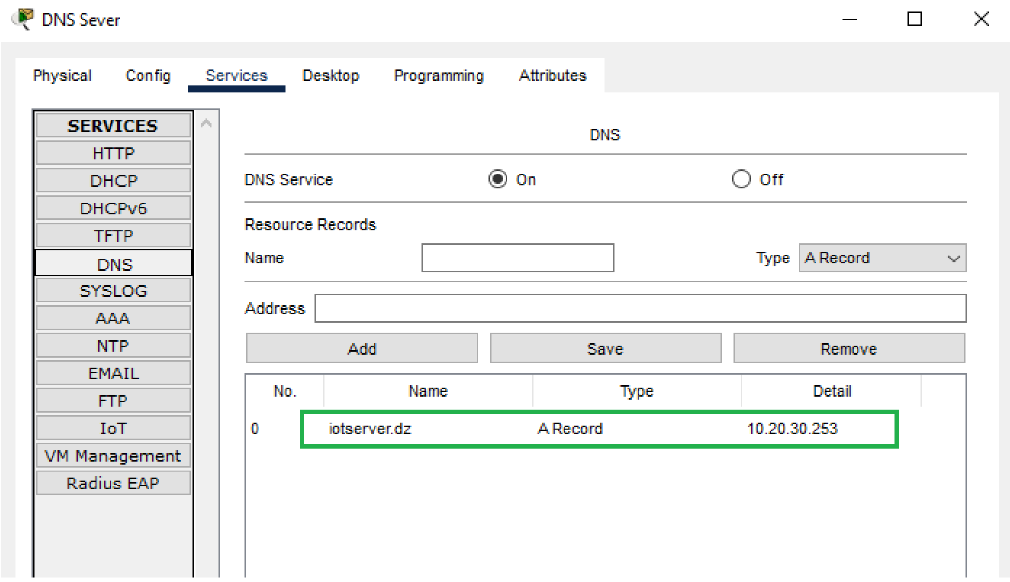


Figure III 14: Domain name translation.

3.5 3G/4G provider network:

3.5.1 Central Office Server:

A central office server plays a crucial role in connecting a cell tower to an ISP router and facilitating the transfer of information between them. Once the DHCP server and DNS server are configured on the ISP router, the central office server automatically receives all the necessary IP information from the ISP. It is equipped with two interfaces: a backbone interface that connects the server to the WAN (Wide Area Network) and a coaxial interface that establishes a connection between the server and the cell tower. This configuration enables seamless communication and data exchange between the cell tower and the ISP router.

Backbone interface: this interface is connected to the internet provider. So, it will get a dynamic address once we enable the DHCP protocol as shown in figure15.

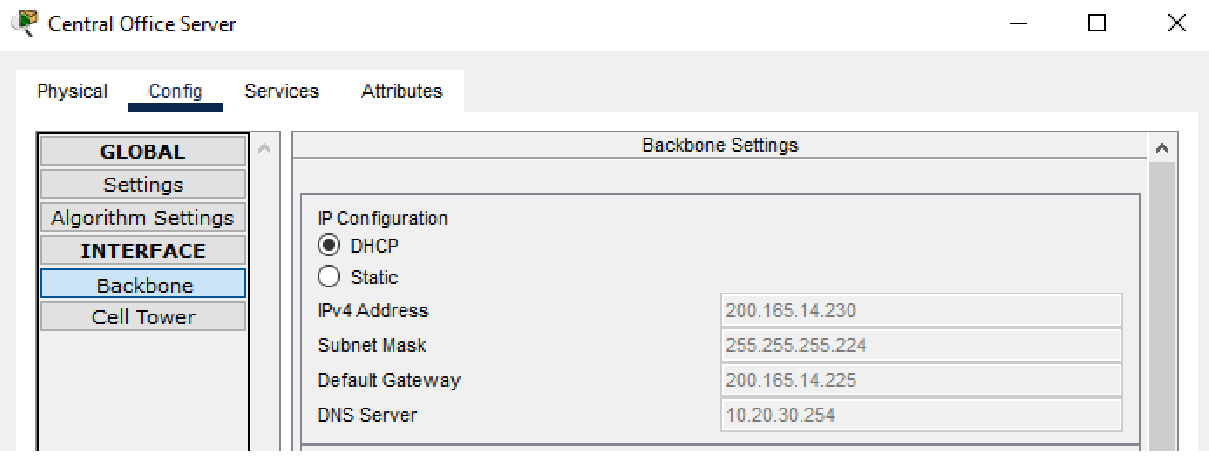


Figure III 15: Backbone interface DHCP enabling

Cell Tower interface: The central office server serves as a DHCP server for all devices connected to the Cell Tower interface. In the depicted configuration shown in Figure 16, the central office server is set up to provide IP addresses ranging from 172.16.1.2 to 172.16.1.254 to the devices connected to the Cell Tower. This ensures that each device connected to the Cell Tower receives a unique IP address within that specified range.

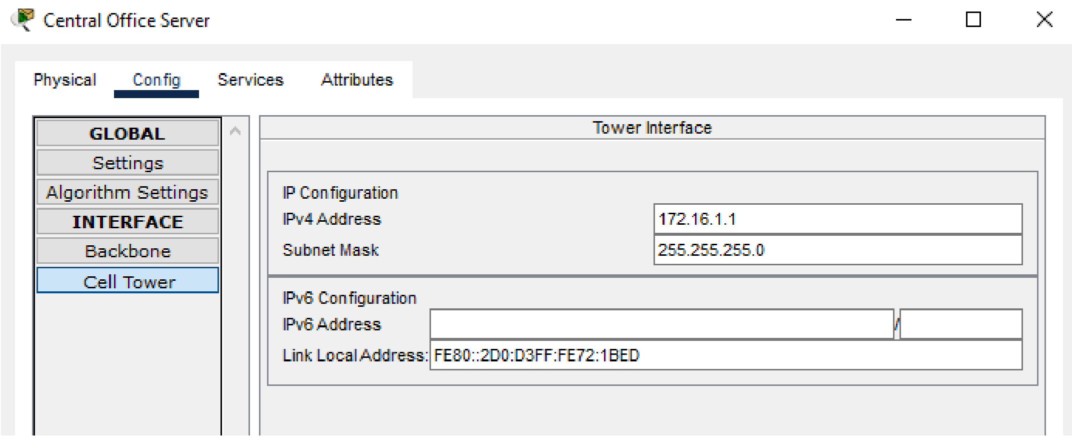


Figure III 16: Cell Tower interface configuration.

3.5.2 3G/4G Client

By typing the name of the 3G/4G provider "ptcellular", the phone will be connected to the tower. Hence, the Central Office server will assign an address to the Client from the network 172.16.1.0/24, in this case the address IP is 172.16.1.100 as illustrated in Figure 17 below.

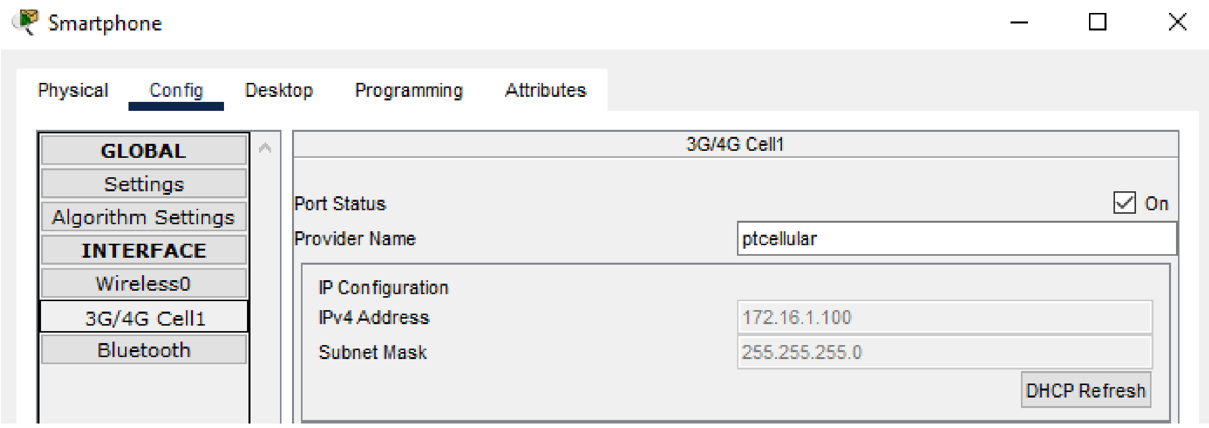


Figure III 17: 3G/4G client configuration.

The default gateway will be "172.16.1.1" and the DNS server provided by the service provider is "10.20.30.254". The settings can be verified from the Config TAB of the Client.

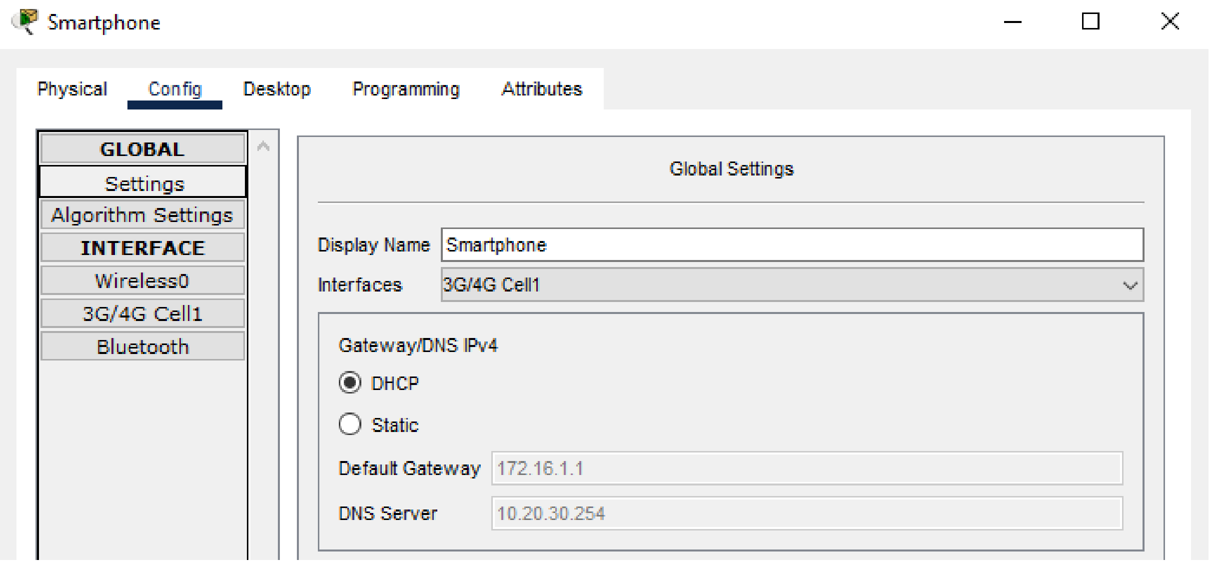


Figure III 18: Default Gateway and DNS server

By following the aforementioned steps, the 3G/4G client will establish a connection to the internet and obtain authorization to access the IoT server. Provided that the client possesses the correct username and password, it will be granted access to the associated account. Additionally, with the appropriate credentials, the client will have the capability to make modifications or perform other permitted actions within the account.

3.6 Internet Cloud (WAN):

The IoT cloud, which operates over the Wide Area Network (WAN), facilitates the transfer of data collected by smart devices from homes to the IoT server for storage. The smart devices acquire their IP addresses from the home gateway through the cloud. Configuring the WAN involves establishing a connection between the Ethernet interface on the router and the coaxial interface on the cable modem, as illustrated in Figure 19. This setup allows for the seamless transmission of data from the smart devices to the IoT server via the WAN infrastructure.

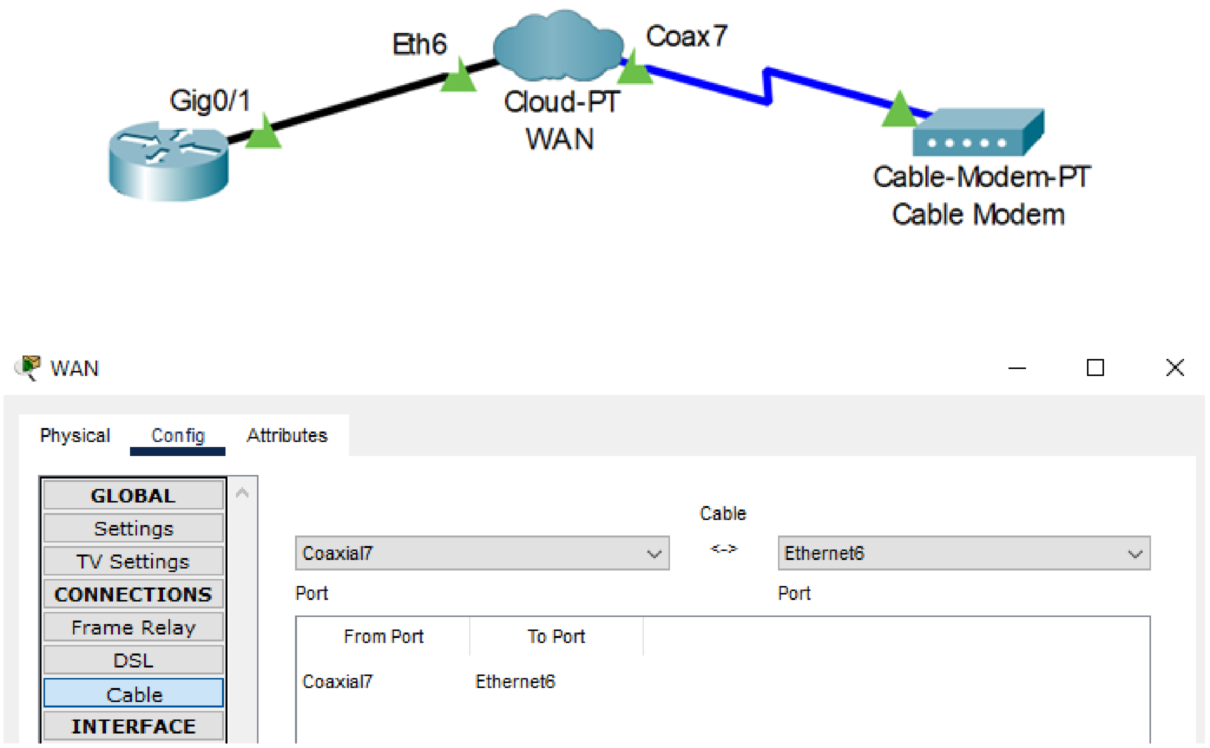


Figure III 19: Internet provider setting.

3.6.1 Modem:

After connecting the Ethernet cable and coaxial cable, the modem will begin functioning without requiring any additional configuration.

3.7 The Smart Home Network:

Our home comprises six main areas: the garage, bedroom, living room, main room, kitchen, and garden.We will explain how these areas are interconnected and the devices involved in the network setup later.

3.7.1 Home gateway:

The home gateway is equipped with four Ethernet ports and a wireless access point. The wireless access point is configured with the Service Set Identifier (SSID) "**SmartHome**". Each device in the network is connected to the gateway by accessing the configuration menu on the respective device and entering the name of the gateway. This process can be repeated for all sets of devices that need to be controlled through the home gateway [57].

By accessing the IP configuration menu on the home gateway, a list of connected devices can be displayed. The control of all these devices can be managed through a smartphone, tablet, PC, or laptop. To establish a secure wireless connection, authentication and validation can be configured on the home gateway “**smart123**” using protocols such as WEP, WPA-PSK, or WPA2.

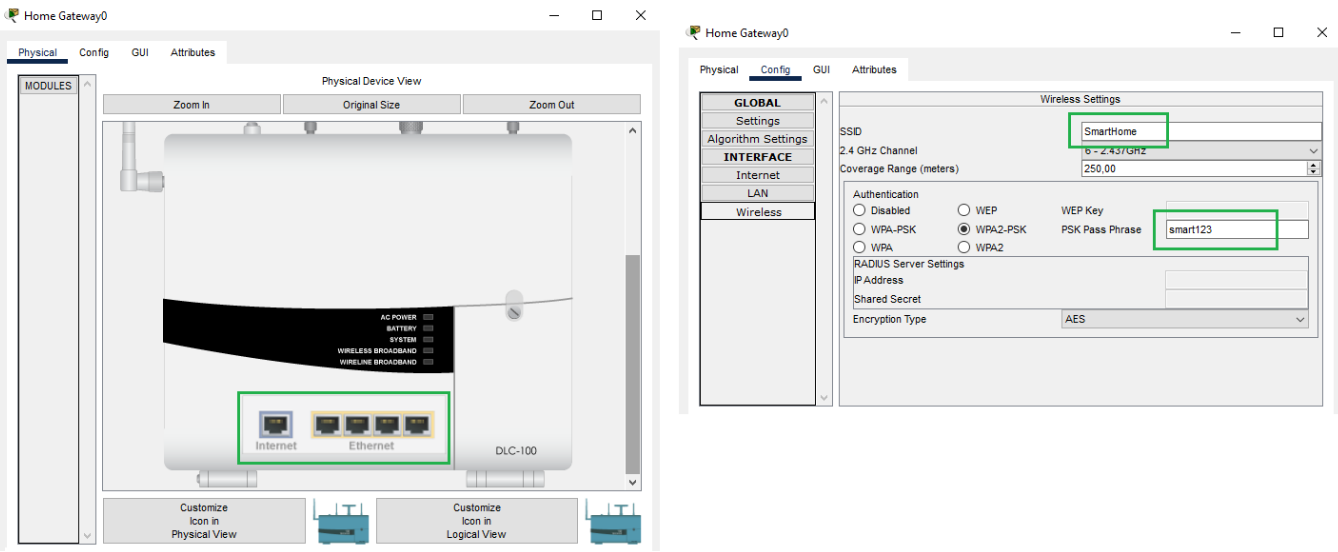


Figure III 20: Home gateway Internet and Ethernet port.

3.7.2 Laptop:

When "**SmartHome**" is entered in the SSID field and "**smart123**" is entered in the Authentication field (WPS2-PSK) within the wireless interface configuration tab, the laptop establishes a physical connection with the home gateway. If DHCP is selected, the home gateway assigns the laptop a dynamic IP address of "192.168.25.116," as depicted in Figure 21.

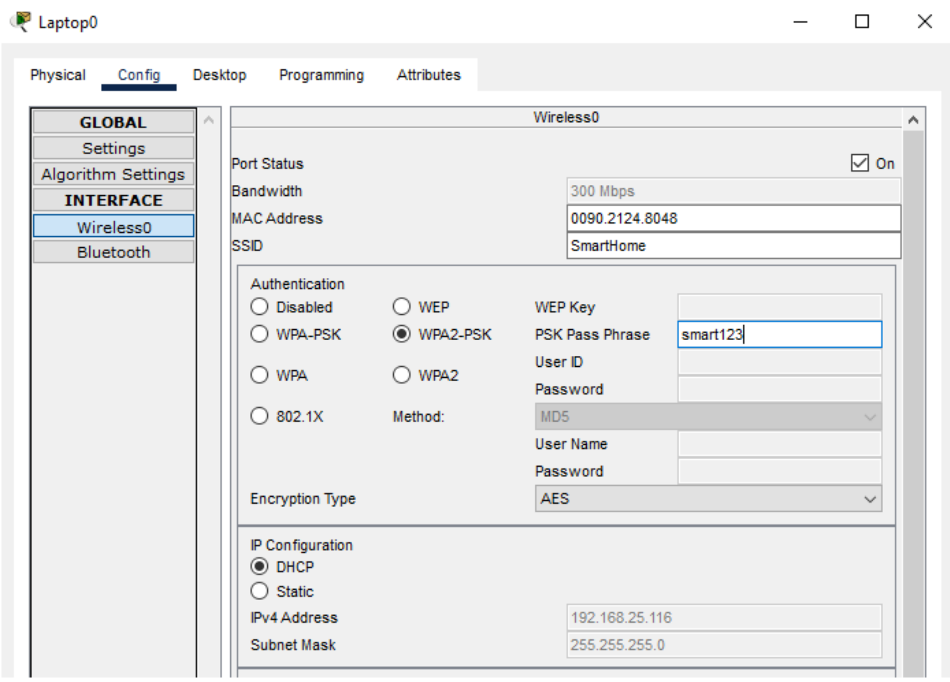


Figure III 21: Laptop Wireless configuration.

Ultimately, the laptop successfully connects to the internet. It receives the DNS server address from the home gateway and gains the necessary permissions to access the local IoT server for management purposes, as illustrated in Figure 22.

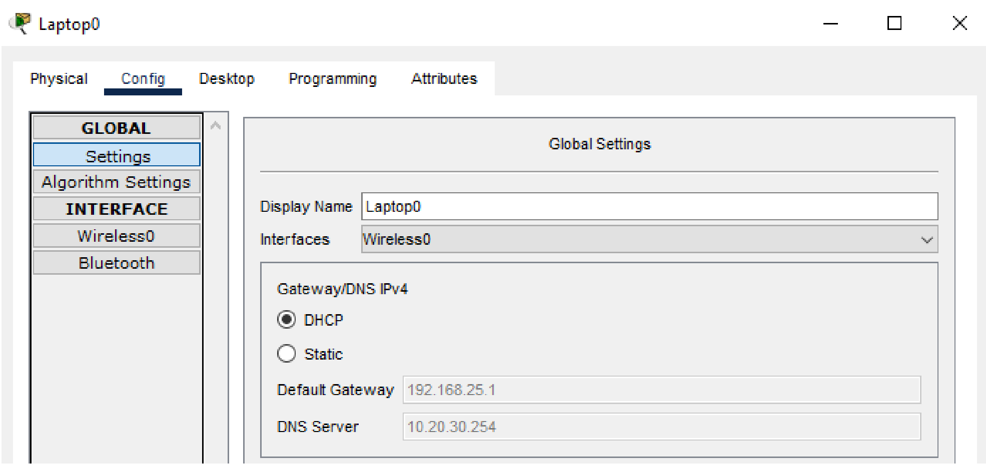


Figure III 22: Laptop Gateway and DNS configurations.

3.7.3 The MCU-PT (Microcontroller) board

The MCU-PT (Microcontroller) board serves as an interface for connecting various smart objects and offers a programming environment that supports multiple languages, including JavaScript, Python, and Visual Basic. These programming languages enable the control and management of the interconnected smart objects.

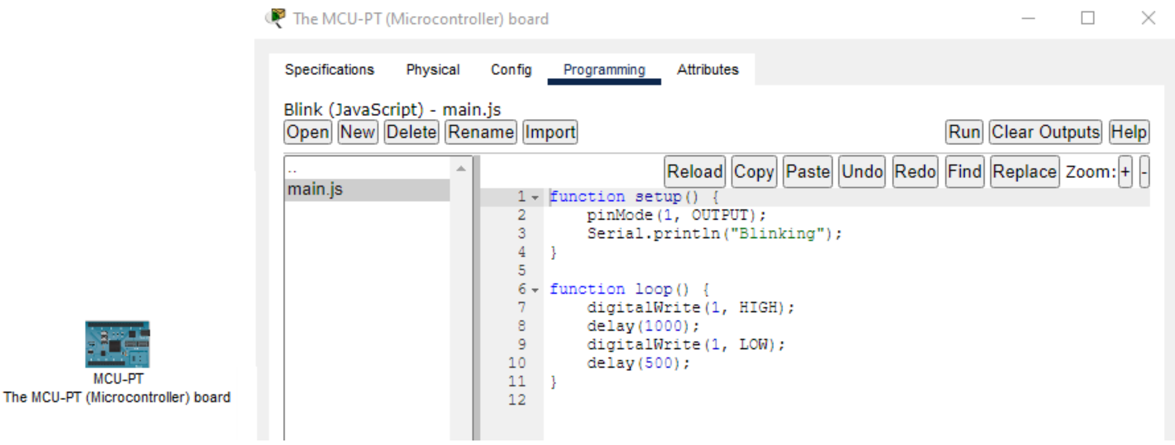


Figure III 23: MCU programming environment.

3.7.4 Configuring the IoT devices

In Cisco Packet Tracer 8.1, the IoT devices have wireless connectivity, but there are some devices that lack a wireless interface. In order to create a wireless interface for these devices, you need to click on the "**advanced**" button, then in the I/O config tab. Within the network adapter settings, the configuration should be set to PT-IOT-NW-1W. Figure 24 displays the advanced options in the I/O config settings for each smart object.

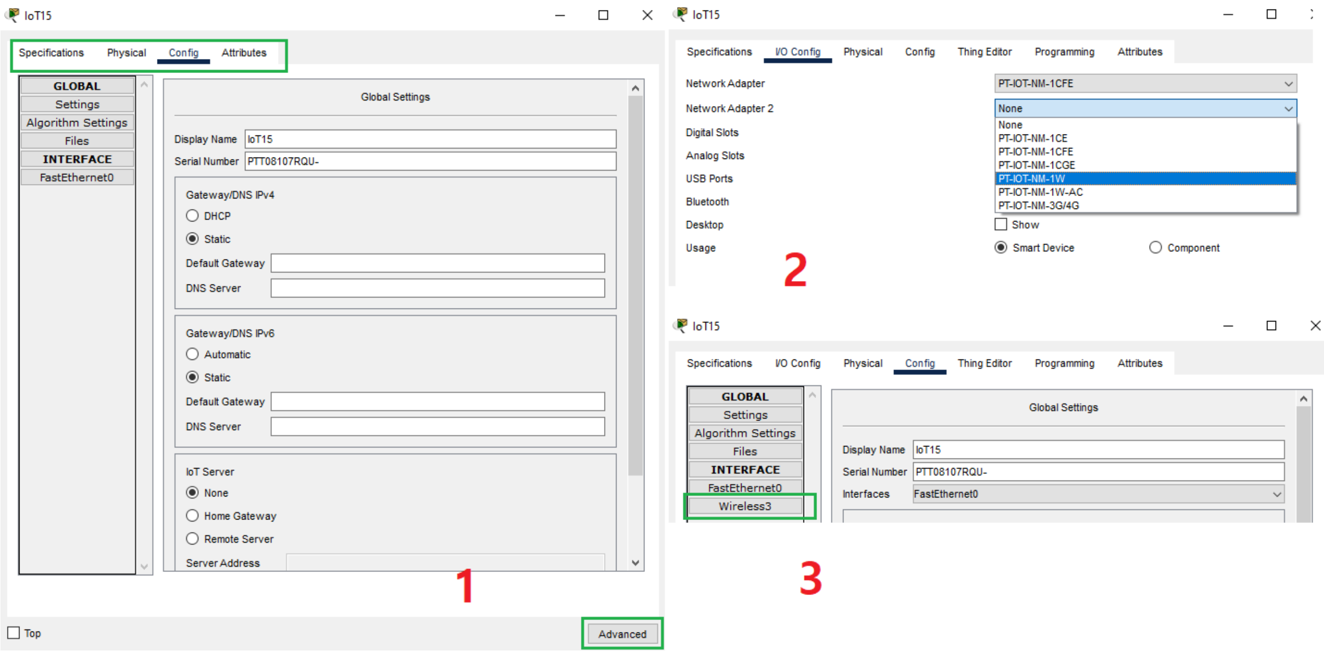


Figure III 24: IoT device setting.

Once this step is completed, we can establish a connection between the IoT device and the Home Gateway by entering the SSID and password.

3.7.5 Device Registration Server:

In order to maintain a record of all IoT devices installed in a home, a registration server is utilized. By default, IoT devices are assigned names starting with "IOT" followed by a number. However, to ensure accuracy and prevent errors, it is necessary to assign clear and distinct names to the devices.

To register a device on the server, the following information is required: the server's address, account username, and password. All devices must utilize the same IoT credentials, which are also used by the homeowner to authenticate themselves when connecting to the IoT Server via a web browser.

To initiate the registration process, access the device's configuration tab and select "Remote Server" in the IoT server field (refer to Figure 25). Then, enter the necessary information and click on the "Connect" button."

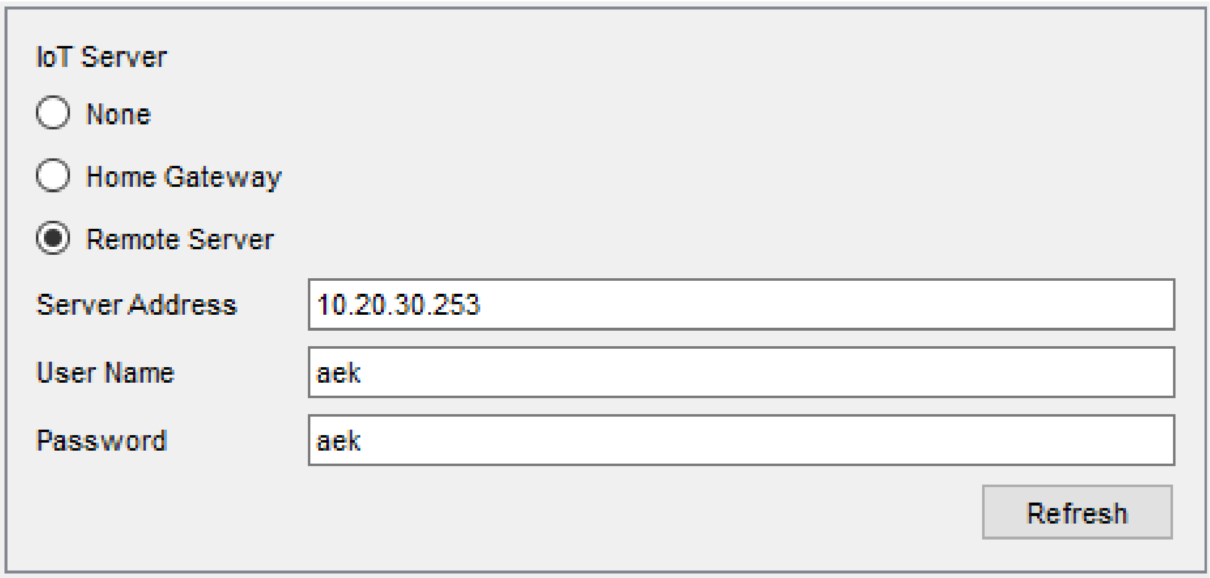


Figure III 25: Registration server.

3.8 Interaction between Devices:

As previously mentioned, our home consists of six primary areas: the garage, bedroom, living room, main room, kitchen, and garden. Now, let's examine each area individually for a detailed analysis.

3.8.1 The garage:

In the depicted system illustrated in Figure 26, both the garage and the door are managed through the utilization of RFID (Radio Frequency Identification) technology, facilitated by an MCU (Microcontroller Unit) board. The system incorporates two RFID cards, each representing a car. One of the RFID cards possesses an authorized ID, while the other card carries an unauthorized ID.

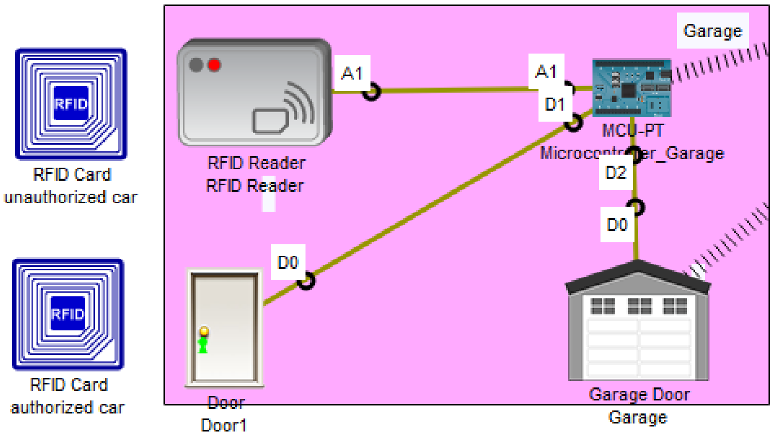


Figure III 26: Garage IoT devices.

This RFID-based control system provides enhanced security and access control for the garage and door. By utilizing RFID technology, the system can distinguish between authorized and unauthorized vehicles based on the unique identification codes stored on the RFID cards.

When the authorized car, equipped with the corresponding RFID card, approaches the RFID reader, the system detects the presence of the authorized ID. As a result, the system triggers the automatic opening of both the garage and the door, granting seamless access to the vehicle owner.

On the other hand, if an unauthorized car, equipped with an RFID card that carries an unrecognized or unauthorized ID, comes into range of the RFID reader, the system recognizes the invalid ID and denies access. In this scenario, the garage and the door remain securely closed, effectively preventing unauthorized entry.

This RFID-based control mechanism ensures that only vehicles with the approved RFID cards are granted access to the garage and the premises beyond. By employing this system, homeowners can enhance the security of their property, restrict unauthorized access, and maintain better control over their vehicle and home protection.

The system's code is implemented using the Python programming language, as indicated in Figure 27.

Figure III 27: Garage & Door Security python code.

3.8.2 The bedroom:

In the specific area mentioned, we have installed two distinct devices: a Smart Window and a Smart Lamp1 (refer to Figure 28). These devices are wirelessly connected to a Home Gateway, which serves as a central communication hub. To efficiently monitor and control these devices, users can access the IoT server's interface using their laptop or smartphone. By navigating to the Desktop tab, users can locate and access the IoT Monitor section, where they can enter the IP address "**10.20.30.253**" or the domain name "**iotserver.dz**" along with their corresponding username and password for the IoT server. Figure 29 provides an illustrative view of the laptop's interface, presenting a comprehensive list of various devices. This interface empowers users to conveniently control and manage the operations of these devices remotely, enhancing their overall experience.

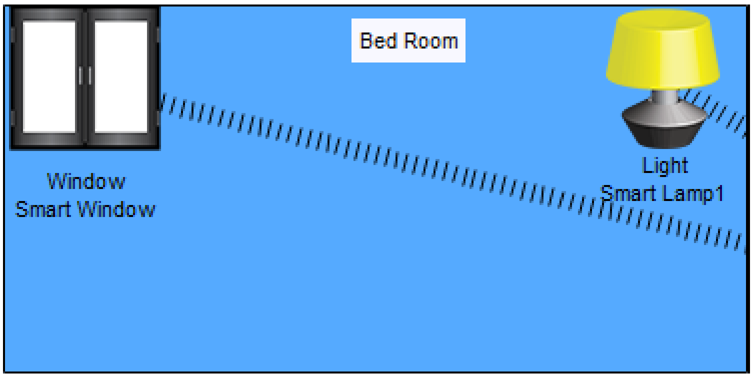


Figure III 28: devices in Bedroom

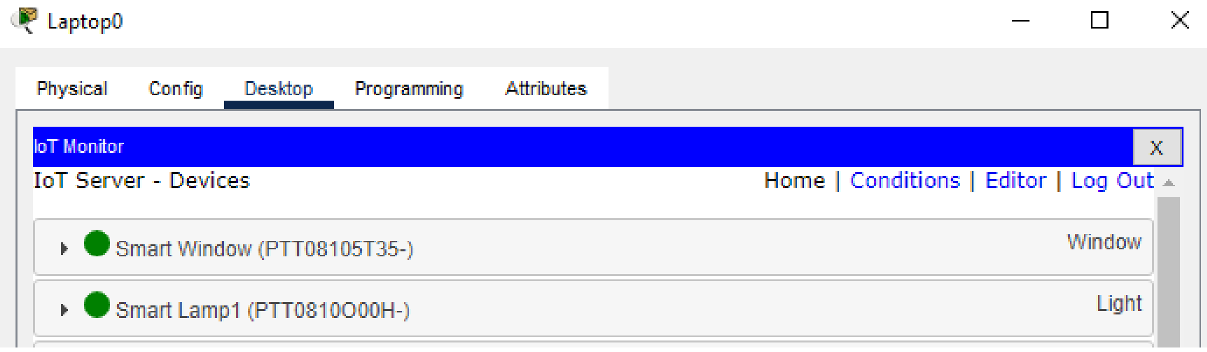


Figure III 29: list of devices in Bedroom

3.8.3 Living room:

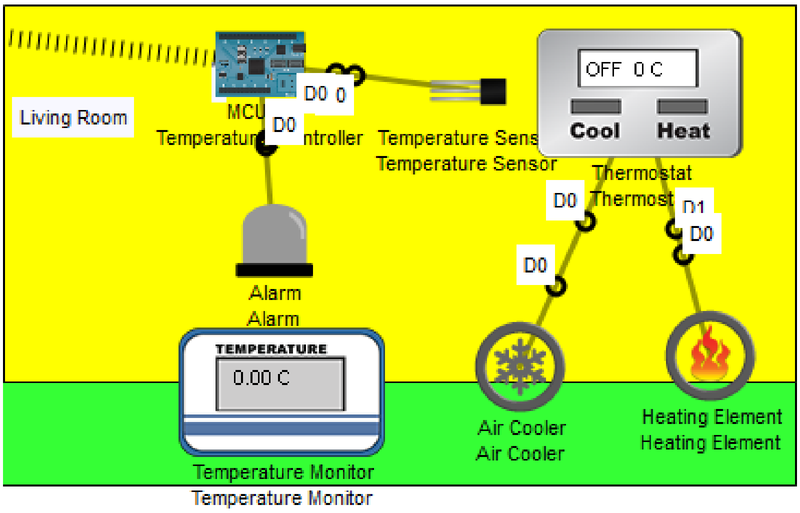


Figure III 30: devices of Living room.

The living room is equipped with a sophisticated setup focused on ensuring comfort and safety. At the core of this system is a microcontroller unit (MCU) serving as the central control unit. It processes data, makes decisions, and coordinates various components within the living room. A temperature sensor connected to the MCU allows real-time monitoring of the ambient temperature, enabling precise responses to temperature changes. In terms of safety, an integrated alarm system is triggered when the temperature exceeds a predefined threshold, prompting the MCU to send email notifications to the user's smartphone or laptop through the IoT server.

To regulate the temperature, a thermostat communicates with the MCU and controls the heating and cooling systems. Connected to the thermostat are a heating element and a chiller, providing warmth and cooling as needed. The thermostat closely monitors the temperature and communicates with the MCU to ensure precise control. Additionally, a dedicated temperature monitor displays the current temperature in a user-friendly manner, allowing users to stay informed about the room's temperature conditions.

This comprehensive living room setup, with its MCU, temperature sensor, alarm system, thermostat, heating element, chiller, and temperature monitor, creates an environment of optimal comfort and safety. It effectively manages temperature and provides immediate alerts in case of any anomalies or potential risks, contributing to an enhanced living experience.

Python Code for Sending Email Notifications to Users:

```
from email import *
from time import *
from gpio import *

def onEmailReceive(sender, subject, body):
    print("Received from: " + sender)
    print("Subject: " + subject)
    print("Body: " + body)

def onEmailSend(status):
    print("send status: " + str(status))

def main():
    EmailClient.setup(
        "mcu@iotserver.dz",
        "iotserver.dz",
        "mcu",
        "password"
    )

    EmailClient.onReceive(onEmailReceive)
    EmailClient.onSend(onEmailSend)

    while True:
        temp=digitalRead(0);
        print("Temperature", temp);
        if (temp>=550):
            EmailClient.send("pc@iotserver.dz", "Temperature ALARM", "Temperture is <<<..HIGH...>>>")
            print("Temperature is HIGH");
            digitalWrite(1,HIGH);
        else:
            print("Temperature is LOW");
            digitalWrite(1,LOW);
        sleep(2);

    # check email once a while
    while True:
        EmailClient.receive()
        sleep(5)

if __name__ == "__main__":
    main()
```

Figure III 31: Python Code for Sending Email

This code demonstrates a Python program for sending email notifications to users. It utilizes the "email" library for email-related functionalities, the "time" library for time-related operations, and the "gpio" library for handling input/output operations.

The main function sets up the email client by providing the necessary configuration details, such as the sender's email address, the server address, the username, and the password. It then registers the onEmailReceive and onEmailSend callbacks.

The code then enters a loop where it continuously reads the value of a digital input (presumably a temperature sensor connected to pin 0). It checks if the temperature reading is above or equal to 550 (assumed threshold for a high temperature). If the temperature is high, it sends an email to the specified recipient address with a predefined subject and body message indicating the high temperature. It also sets pin 1 to HIGH (presumably controlling an output device such as an alarm or warning indicator) and prints a corresponding message. If the temperature is below the threshold, it sets pin 1 to LOW and prints a message indicating that the temperature is low. The program then waits for 2 seconds before repeating the temperature check.

Additionally, the code includes another loop dedicated to checking for new email messages. It calls the receive function of the email client to check for any incoming emails every 5 seconds.

In the end, this code sets up an email client, monitors the temperature, sends email notifications in case of high temperature, and checks for incoming emails periodically. It demonstrates a basic implementation for sending email alerts based on temperature readings using an email library and GPIO operations.

3.8.4 Main room:

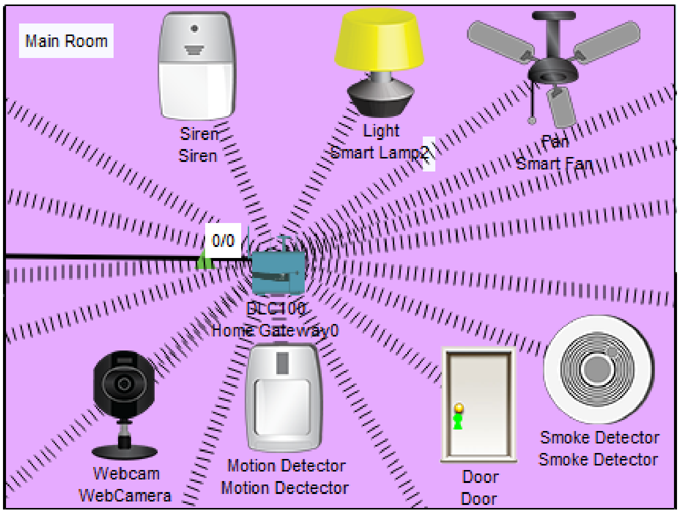


Figure III 32: Main room devices.

The main room is equipped with a comprehensive set of seven devices to enhance functionality and security. These devices include a Smart Light, Smart Fan, Siren, Door, Smoke Detector, Motion Detector, and Webcam. All of these devices are interconnected and communicate with each other through a connection established with the Home Gateway. This enables seamless integration and synchronized operations within the room.

When the Motion Detector detects any movement out the room, it triggers an automatic response (see table 4). In this case, the Door is automatically locked to ensure the safety and privacy of the homeowners. Simultaneously, the Webcam initiates live video streaming, capturing the scene in real-time. The live video feed is then securely transmitted to the user, allowing them to remotely monitor the room from their preferred device, such as a smartphone or laptop. This enables users to stay updated on any activities or potential security concerns in the room, even when they are not physically present.

On the other hand, to ensure comprehensive safety measures, a Smoke Detector is installed within the room. When the Smoke Detector detects the presence of smoke, it immediately triggers an alarm message to the Siren. The Siren emits a loud and attention-grabbing sound, alerting about the potential fire hazard. As part of a security procedure, the Smart Fan is activated to help mitigate the smoke and improve air circulation within the room. Additionally, both the bedroom and kitchen windows automatically open to facilitate the ventilation process and allow the smoke to dissipate effectively. Once the smoke level reduces to an acceptable and safe limit, the Siren and Smart Fan are automatically stopped. This prevents any unnecessary noise or disturbance. Furthermore, the windows in the bedroom and kitchen are securely closed, restoring the room to its normal state and maintaining a controlled environment.

Name	Condition	Actions
Door_ON	Motion Dectector On is true	Set Door Lock to Lock Set WebCamera On to true
Door_OFF	Motion Dectector On is false	Set Door Lock to Unlock Set WebCamera On to false
Siren_ON	Smoke Detector Level >= 0.15	Set Siren On to true Set Smart Window On to true Set Smart Window1 On to true Set Smart Fan Status to High
Siren_OFF	Smoke Detector Level <= 0.11	Set Smart Window On to false Set Smart Window1 On to false Set Smart Fan Status to Off Set Siren On to false

Table 4: system operations in main room.

3.8.5 Kitchen:

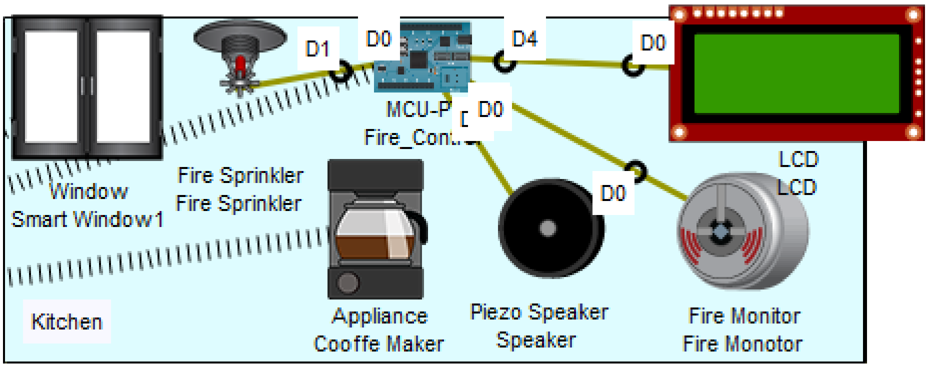


Figure III 33: Kitchen IoT devices.

The kitchen area is equipped with several devices aimed at enhancing functionality and ensuring safety. Among these devices are the Coffee Maker, Window, and a comprehensive fire monitoring system consisting of a Fire Detector, Fire Sprinkler, Speaker, and LCD Monitor. These devices are interconnected and controlled through a Microcontroller Unit (MCU) board, which runs a Python code specifically designed to manage their operations.

In the event of a fire, the Fire Detector promptly detects the presence of flames or excessive heat. Upon detection, the Speaker immediately emits a loud and distinctive alarm sound. Simultaneously, the LCD Monitor prominently displays the alarming message "Fire Detected," providing clear visual indication of the hazardous situation. This immediate response ensures that individuals within the vicinity are promptly notified and can take necessary action.

To suppress the fire, the Fire Sprinkler system is activated. The fire sprinkler heads disperse water throughout the kitchen area, effectively reducing the spread of the fire and preventing further damage. The coordinated effort between the fire monitoring system and the sprinkler system ensures swift and efficient firefighting capabilities, enhancing the safety of the kitchen.

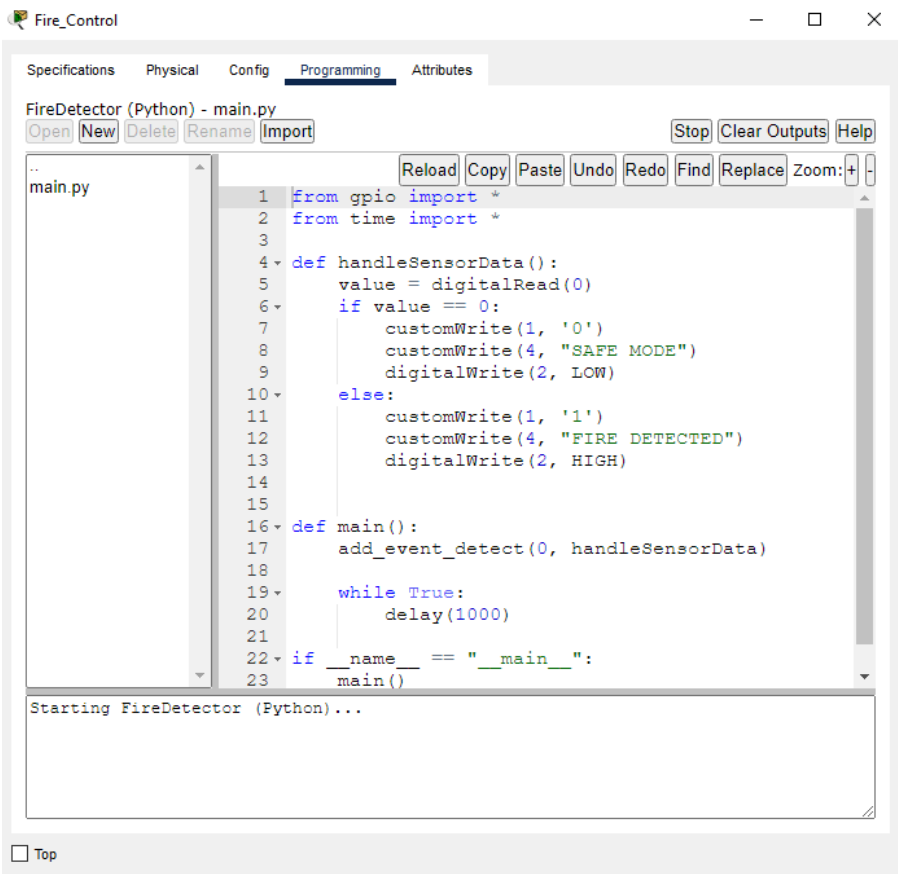


Figure III 34: Code Python for detection of fire.

The Window in the kitchen is also integrated into the overall safety measures. Similar to the previous description regarding the Smoke Detector, the Window operation is controlled by the Smoke Detector system. In the event of smoke detection, the Window automatically opens to facilitate ventilation and allow smoke to dissipate, contributing to improved air quality and visibility within the kitchen. This feature aids in creating a safer environment and assists in the mitigation of fire-related risks.

Moreover, users have the convenience of monitoring and controlling the Coffee Maker through their laptop or smartphone. Using the IoT server and appropriate software applications, individuals can remotely access the Coffee Maker's interface. This allows them to conveniently turn the Coffee Maker on or off, ensuring a freshly brewed cup of coffee at their preferred time. The ability to monitor and control the Coffee Maker remotely adds convenience and efficiency to the kitchen area.

3.8.6 Garden:

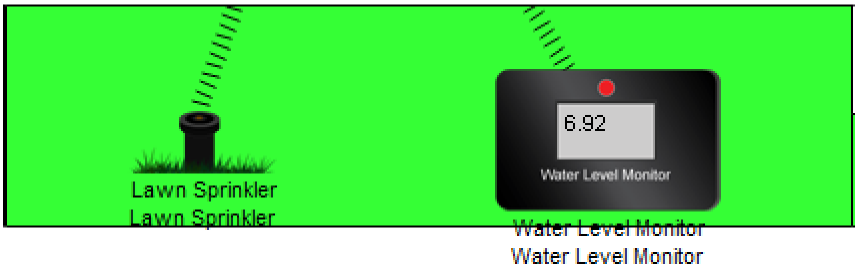


Figure III 35: Garden IoT devices

The garden features two primary devices: a water level monitor and a lawn sprinkler. The system operates in the following manner:

When the water level monitor detects that the water level is less than 5.0 cm, it signifies that the water level is low. In response to this condition, the system activates the lawn sprinkler by setting the "Lawn Sprinkler Status" to true. This action ensures that the sprinkler turns on and provides water to the lawn, maintaining the desired level of moisture.

Conversely, if the water level monitor indicates that the water level is greater than 7.0 cm, it implies that the water level is high. In this case, the system deactivates the lawn sprinkler by setting the "Lawn Sprinkler Status" to false. This action prevents overwatering, as the sprinkler is turned off when the water level exceeds the specified threshold.

By implementing these rules, the lawn sprinkler system can automatically adjust its operation based on the monitored water level, ensuring an optimal balance of watering the lawn without wasting excessive water.

Name	Condition	Actions
Lawn_Sprinkler_ON	Water Level Monitor Water Level < 5.0 cm	Set Lawn Sprinkler Status to true
Lawn_Sprinkler_OFF	Water Level Monitor Water Level > 7.0 cm	Set Lawn Sprinkler Status to false

Table 5: Garden system operations.

Conclusion:

In this chapter, we have successfully constructed an integrated network comprising four fundamental components. Each component has been meticulously explained, and we have provided detailed insights into their interconnections. By comprehending the intricate workings of these components, we have laid a solid foundation for understanding the system as a whole.

Furthermore, our exploration expanded to encompass the realm of smart home networks. We delved into the six distinct regions that make up the smart home network, unraveling their intricacies and shedding light on the inner workings of the systems within each region. Additionally, we explored the diverse range of smart devices employed in these systems, showing off their capabilities and functionality.

The upcoming fourth chapter promises to delve further into the experiential aspect of these devices. Through analysis and first-hand experiences, we will delve into the comprehensive practical usage scenarios, user interactions, and the overall benefits and implications of these smart devices within the broader context of the integrated network and smart home ecosystem.

CHAPTER 4

SIMULATION AND RESULTS

Introduction:

After we have designed and created the smart home network, we will test each device by providing our project with the Internet, allowing us to monitor and control them wirelessly from anywhere in the world. Throughout the simulation, we will explore real-world scenarios and assess the reliability, responsiveness, and security of our network. We will evaluate the effectiveness of our design, connectivity between devices, and overall user experience, while also examining the implemented security measures and analyzing the efficiency of remote monitoring and control.

Overall, this chapter provides a deep dive into the world of smart home networks and their capabilities. By utilizing Cisco Packet Tracer 8.1, we are able to create a simulated environment where we can design, configure, and test the functionalities of a variety of devices. We explore the potential of IoT and its role in enabling seamless connectivity and control. Through extensive analysis and evaluation, we gain insights into the performance and effectiveness of our network design, paving the way for a deeper understanding of the opportunities and challenges presented by IoT-based systems in smart homes.

1. Testing and Results of the project:

The provided image illustrates the comprehensive details of our project.

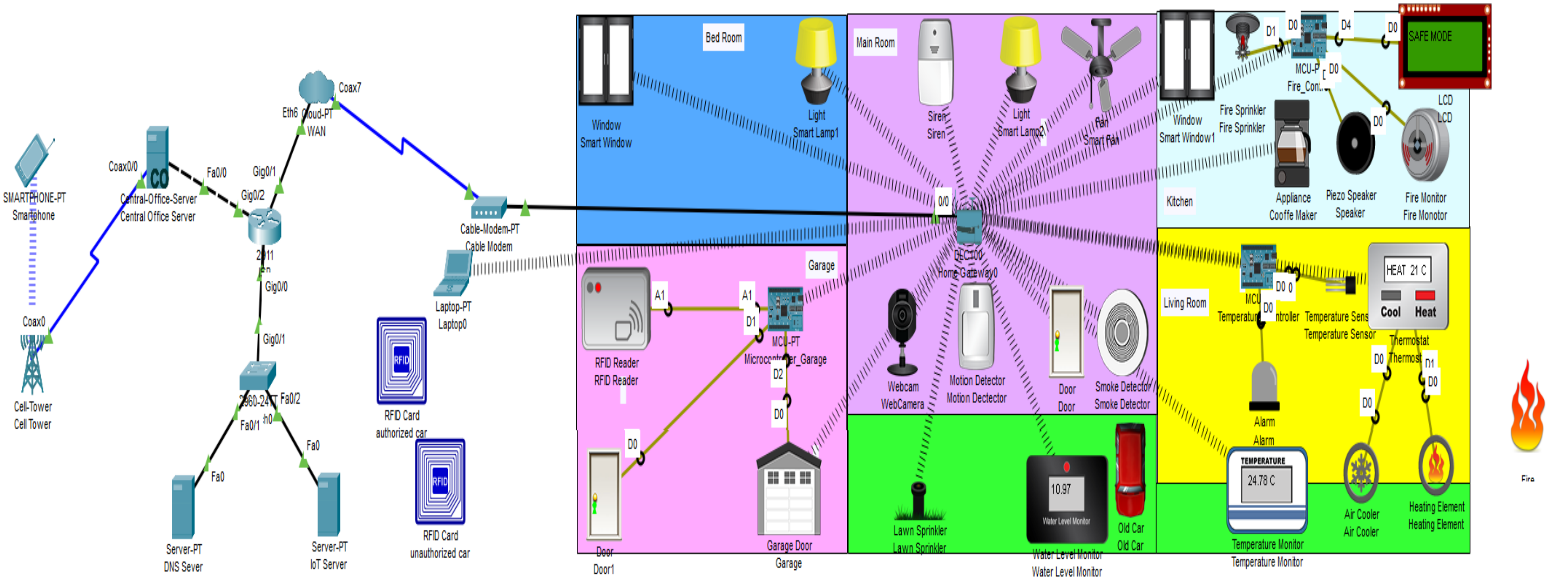
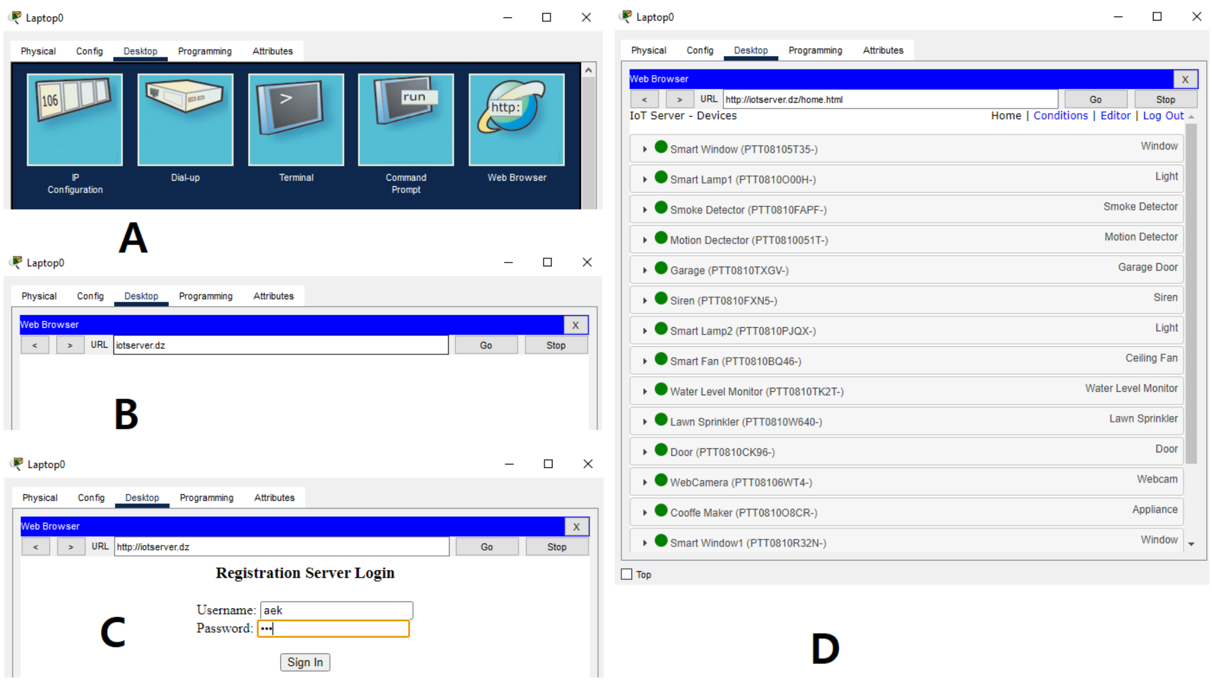


Figure IV 1: Smart Home Network.

1.1 Evaluation of Manual Usage for IoT Devices Testing:

Before proceeding with the testing of various devices in our project, it is essential to ensure that both the IoT server and DNS server are functioning correctly. We can accomplish this by conducting tests using a laptop and a smartphone.

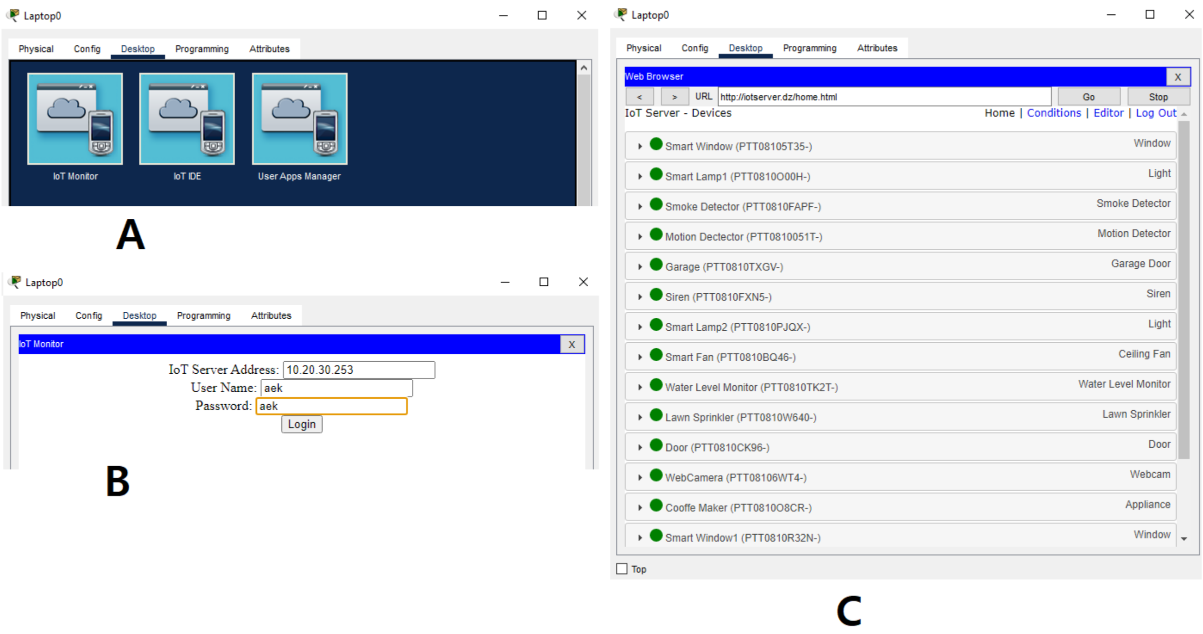
To verify the functionality of these servers, we have two options within the desktop tab of the laptop or smartphone. The first option involves using a web browser. In the web browser, we will type the domain name "**iotserver.dz**" (see Figure 2: A, B, C, and D). If this domain name translates to the IP address of the IoT server, which is "**10.20.30.253**", it indicates that the DNS server is working. Additionally, if we can successfully access the IoT server and are presented with fields to enter the username and password “**aek**”, it confirms that the IoT server is operational.



(A): click on web browser, **(B):** write the domain name, **(C):** enter the username & password, **(D):** list of IoT devices.

Figure IV 2: Verifying DNS and IoT Server Functionality - First way

The second option is to utilize the "IoT monitor". We can enter the address of the IoT server in the IoT monitor and provide the appropriate username and password. This will grant us access to the list of IoT devices associated with the server (see Figure 3).



(A): click on IoT Monitor, **(B):** enter the address, username, and password, **(C):** list of IoT devices

Figure IV 3: verification the IoT server – second way

By performing these tests, we can ensure that both the DNS server and IoT server are functioning properly before proceeding with further device testing in our project.

As mentioned in the previous chapter, our smart home is divided into six distinct areas. To proceed with the implementation and testing, let's focus on each device within these areas individually. This approach ensures thorough testing of the smart home network.

1.1.1 The garage door:

As depicted in the figure, the garage is set to remain locked by default at all times.

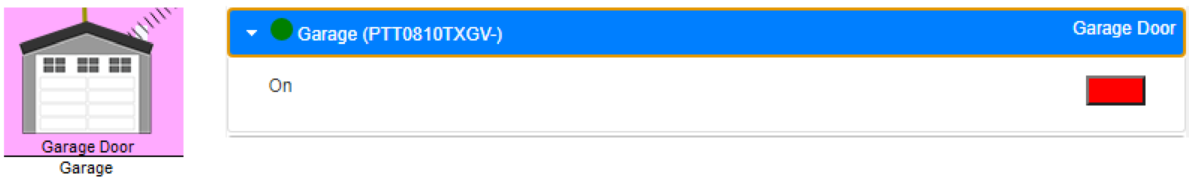


Figure IV 4 :Locked Status of the Garage

Upon clicking the red button, it undergoes a transition and changes to green, indicating that the garage has been successfully opened (see the Figure 5). When controlling IoT devices wirelessly, a series of commands or instructions are sent from a controlling device, such as a smartphone or computer, to the target IoT devices over a wireless connection. Once the commands reach the IoT devices, they are processed and executed accordingly. These commands are typically transmitted through a network, such as Wi-Fi or cellular data.

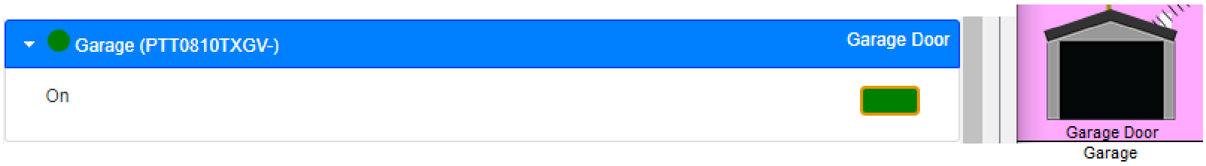


Figure IV 5: Opened Status of the Garage

1.1.2 The door:

The door can be described by two statuses. The first status pertains to whether the door is locked or unlocked. Currently, the door is always in the unlocked state, indicated by the keyhole being displayed in green color. However, if the keyhole appears red, it signifies that the door is locked.

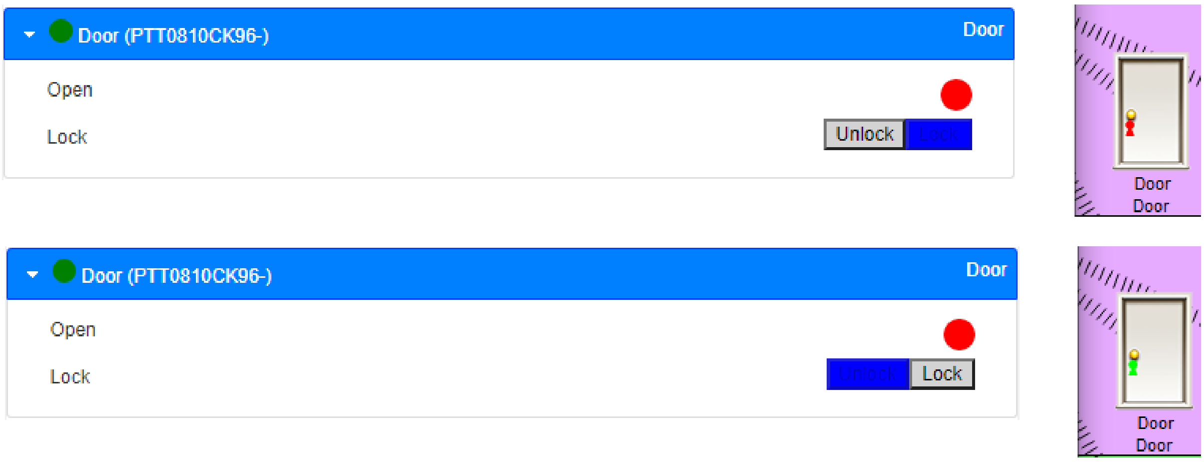


Figure IV 6: Lock/Unlock status of the door

The second status relates to whether the door is open or closed. It is important to note that this aspect of the door's state is not controllable remotely.

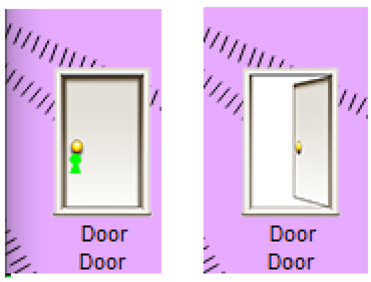


Figure IV 7: Open/Close status of the door

For direct control, the following actions can be performed:

- To lock or unlock the door, press and hold the ALT key, then click on the keyhole.
- To open or close the door, press and hold the ALT key, then click on the door.

Note:The door1 is opened only when the RFID reader sends a validation message to the MCU board.

1.1.3 The window:

The window also, by default, is locked as illustrated in figure 11.To control the window, it is necessary to disable certain conditions or restrictions.

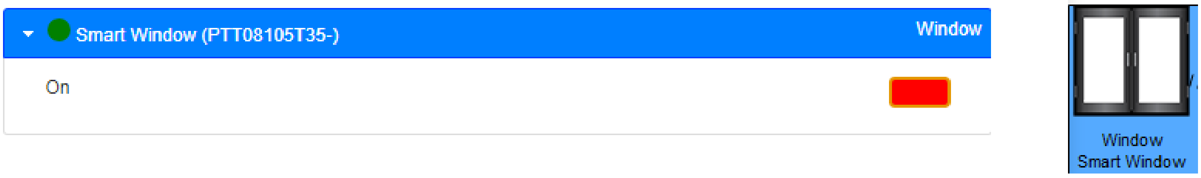


Figure IV 8: Window in Closed Position.

After clicking the “unlock” bottom, it unlocks the door. The door is opened manually by clicking the door itself (ALT+click).

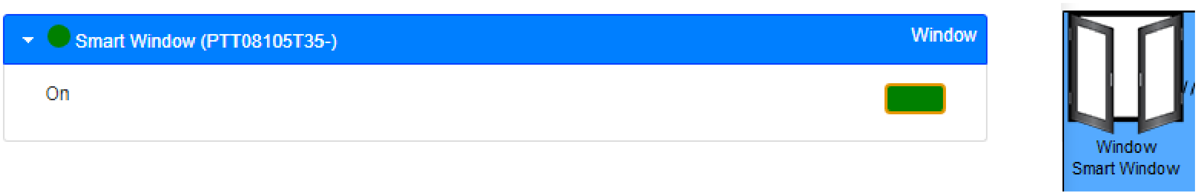


Figure IV 9: Window in Opened Position.

1.1.4 The lamp:

A Controllable Lamp with Three Statuses:

The lamp offers three distinct statuses: Off, Dim, and On. It can emit light to illuminate the surrounding environment.

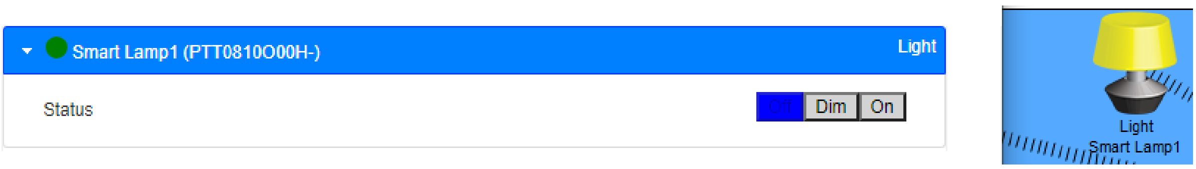


Figure IV 10: Lamp in "Off" Status.

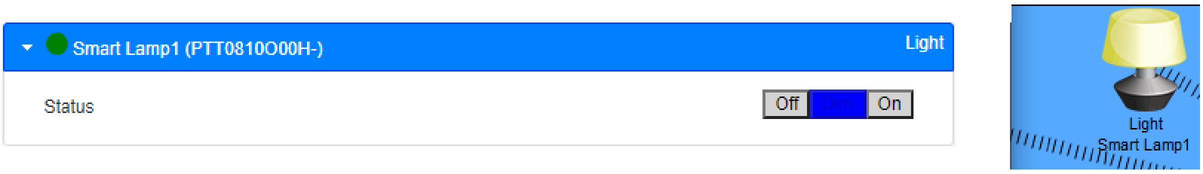


Figure IV 11: Lamp in "Dim" Status.

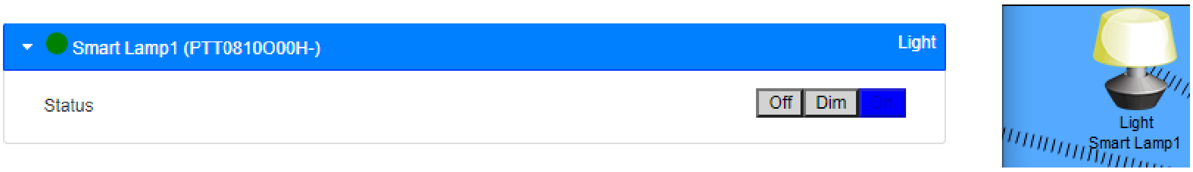


Figure IV 12: Lamp in "On" Status.

1.1.5 The fan:

The fan provides three distinct states: off, low speed, and high speed. By clicking on the buttons displayed on the front of the fan control page, as depicted in Figures 13, 14, and 15, you can transition seamlessly between these different modes.

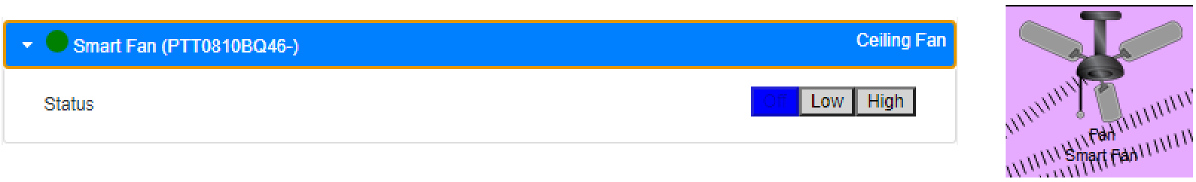


Figure IV 13: Fan in “Off” status.

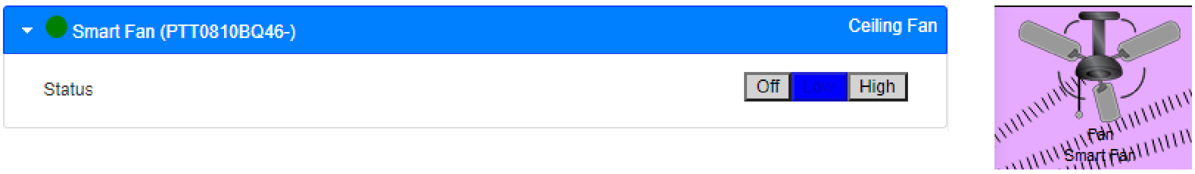


Figure IV 14: Fan in “Low speed” status.

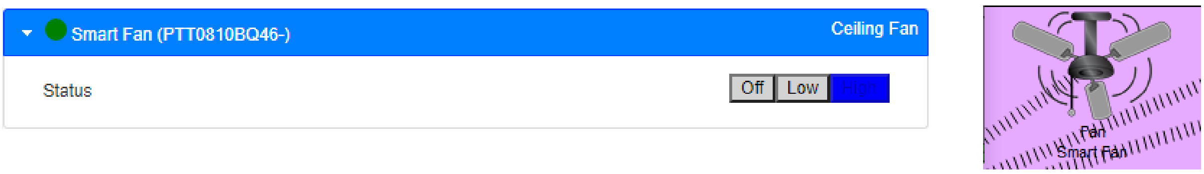


Figure IV 15: Fan in “High speed” status.

1.1.6 Webcam:

Camera device used for recording and transmitting data is commonly employed to enhance home security. It acts as a surveillance tool, monitoring activities and detecting potential intrusions. The camera provides several features, including the ability to switch it off for privacy or power conservation. Turning it on enables real-time video capture for remote monitoring. Additionally, the camera supports video recording, allowing users to store footage for later review or evidentiary purposes. Users can conveniently interact with the camera through direct control, accessible by ALT-clicking to access settings and perform specific actions. Advancements in technology have also made it possible to control the webcam remotely using smartphones or laptops, providing flexibility and accessibility. Figure 16 depicts the webcam in the "Off" status, indicating its deactivation, while Figure 17 represents the "On" status, signifying its readiness for real-time video capture.

Figure IV 16: Webcam in the “Off” status.

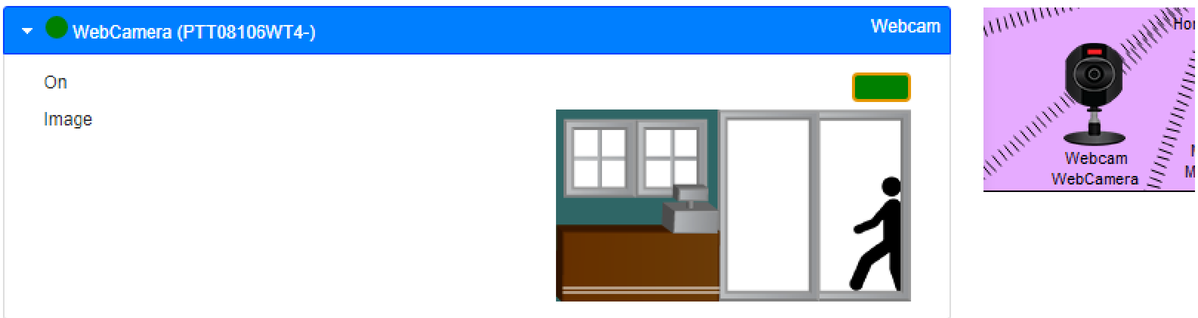


Figure IV 17: Webcam in the “On” status.

1.1.7 Coffee maker:

A coffee maker is a household instrument/device used for making coffee. Like many other devices, the coffee maker can be controlled in two ways. The first method involves remote control using a smartphone or other compatible devices. This allows users to operate the coffee maker from a distance, providing convenience and flexibility. The second method of control is by holding down the ALT key and clicking on the coffee maker.

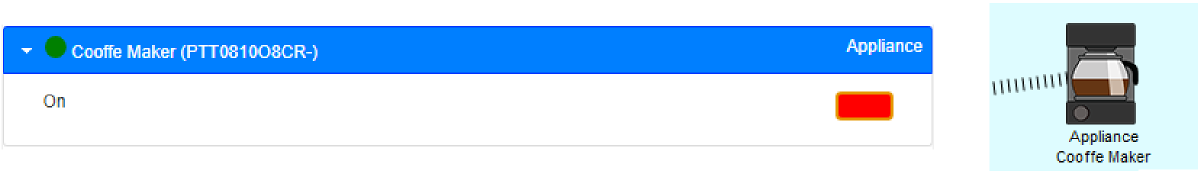


Figure IV 18: Coffee maker in “Off” status.

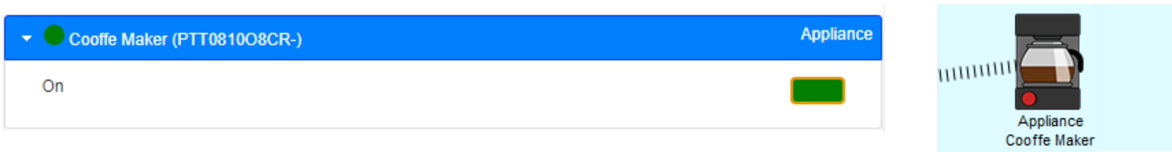


Figure IV 19: Coffee maker in “On” status.

1.1.8 Motion detector:

The motion detector plays a vital role in the smart home setup by providing seamless and reliable motion detection based on any body movement. Its automatic deactivation feature further enhances the smart home system's overall energy efficiency, making it an intelligent choice for homeowners.

Here are the features of the motion detector:

Motion Detection: The motion detector is equipped to detect motion based on mouse movement. providing real-time input for various applications and interactions.

Auto Deactivation: To optimize power consumption and ensure efficiency, the motion detector is designed to automatically deactivate itself after 5 seconds of detecting no mouse movement.

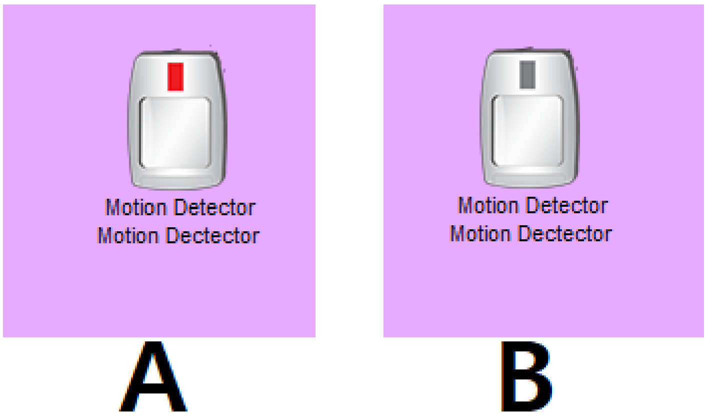


Figure IV 20: Motion detector: (A): Motion detection, (B): Auto Deactivation

1.1.9 Siren:

The same concept applies to the siren, which can be remotely controlled using a smartphone or laptop. Alternatively, you can directly control it by holding down the ALT key and clicking with the mouse.

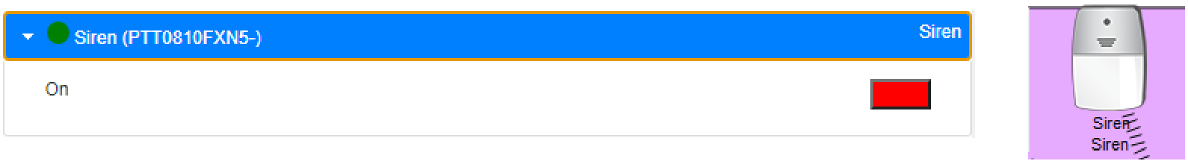


Figure IV 21: Siren in "Off" status.

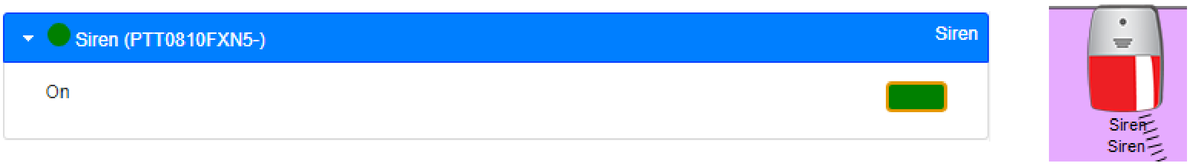


Figure IV 22: Siren in "ON" status.

1.1.10 Fire Sprinkler:

The fire sprinkler is a device designed to extinguish fires. It possesses the following features: it can raise the water level to combat the fire effectively. When it comes to direct control, you can interact with it by simply holding down the ALT key and clicking with the mouse. Additionally, it can be connected to a fire sensor, enhancing its functionality and enabling it to respond swiftly to detected fire hazards.

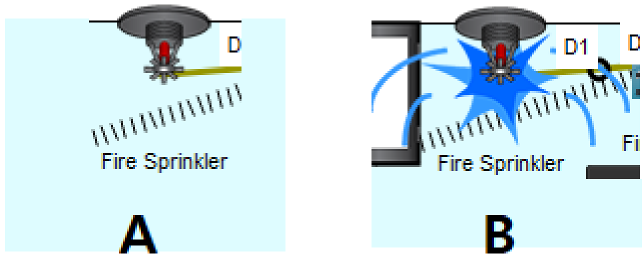


Figure IV 23: Fire Sprinkler: (A): in “OFF” status, (B): in “ON” status.

1.1.11 Lawn Sprinkler:

The lawn sprinkler is specifically designed for watering lawns. It operates by raising the water level to ensure proper irrigation. When it comes to direct control, you can easily interact with the lawn sprinkler by holding down the ALT key and clicking with the mouse, allowing for convenient adjustments and customization of its watering patterns.

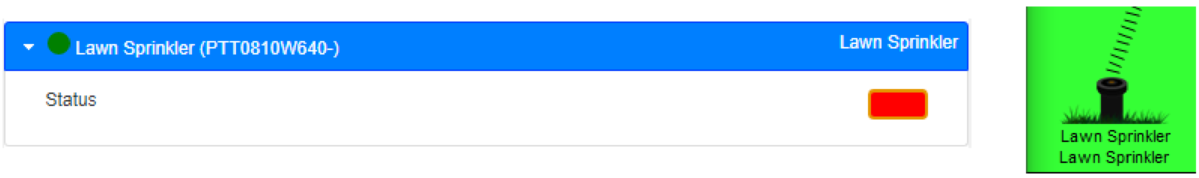


Figure IV 24: Inactive Lawn Sprinkler.

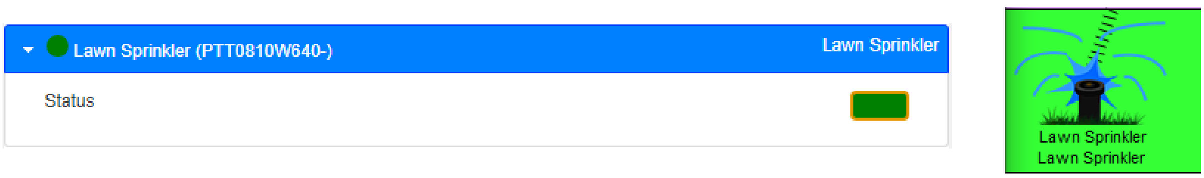


Figure IV 25: Active Lawn Sprinkler

1.1.12 Thermostat:

The thermostat offers various features including Off Mode, Cooling Mode, Heating Mode, and Auto Mode. we can control it remotely with a smartphone or laptop. Also, to interact with it directly, simply hold down the ALT key and click. For example, it effectively utilizes a temperature sensor to detect and display changes, seamlessly transitioning between heating and cooling modes based on the desired temperature settings.

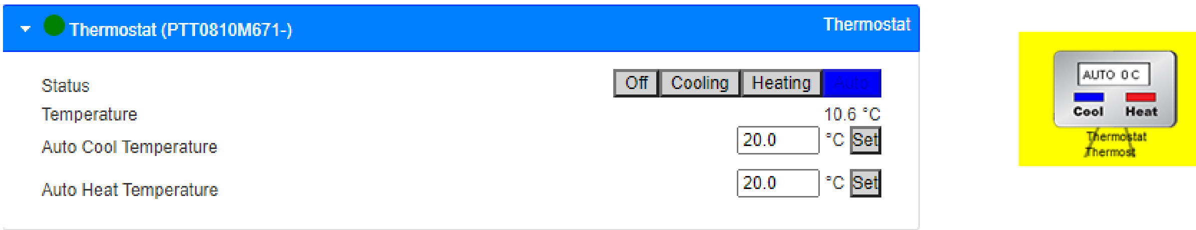


Figure IV 26: Thermostat on Auto Mode

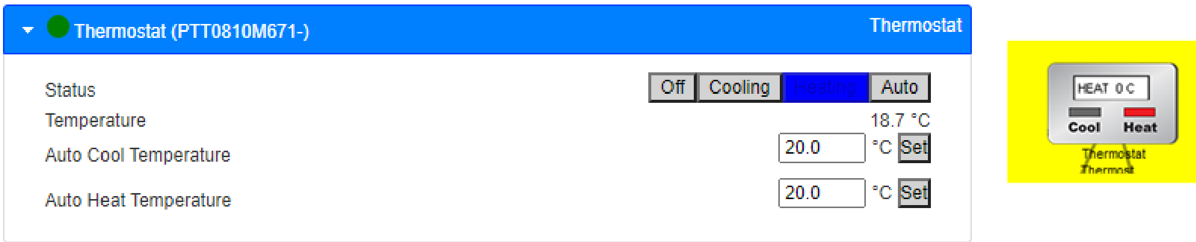


Figure IV 27: Thermostat on Heating Mode

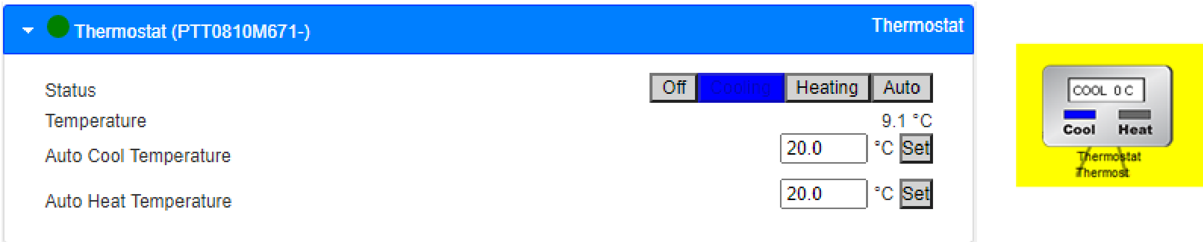


Figure IV 28: Thermostat on Cooling Mode

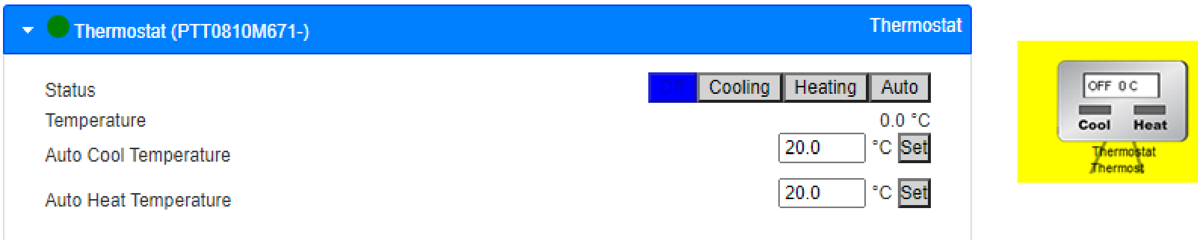


Figure IV 29: Thermostat on Off Mode

Note:

During our project, we encountered various devices like LCDs, speakers, fire monitors, and RFID readers that couldn't be controlled remotely. Furthermore, there were other devices within our project that had the potential for remote control. However, we made a deliberate choice not to establish a connection between these devices and the server for remote control purposes. Instead, we opted to directly link them to microcontroller boards.

1.2 Validation of the IoT device automation:

After thoroughly examining the manual control capabilities of the devices, our focus now shifts towards testing their automation features. These devices will operate collaboratively within an interconnected system, enabling seamless interactions and enhanced functionality. By conducting comprehensive tests in this area, we aim to ensure a smooth and efficient user experience while leveraging the full potential of the Internet of Things (IoT) technology.

1.2.1 The garage door:

In the previous section, we explored a system comprising two devices connected to an MCU board. This configuration facilitated the control of garage and door operations by utilizing an RFID reader to receive specific messages. Let us now delve into two distinct scenarios to evaluate the system's performance.

Scenario 1: To initiate the test, we approached the RFID reader with an authorized card (representing our car). As we brought the authorized card in close proximity, we observed the system's response. Due to the presence of the car near the garage, the motion detector was triggered, indicating movement. Consequently, the RFID reader transmitted a validation message to the MCU board, which granted permission to open the smart garage and the door. Conversely, when the authorized car moved away, the MCU board received an invalid message from the RFID reader, prompting the closure of both the garage and the door. This successful validation confirms that the system operates as intended in Scenario 1.

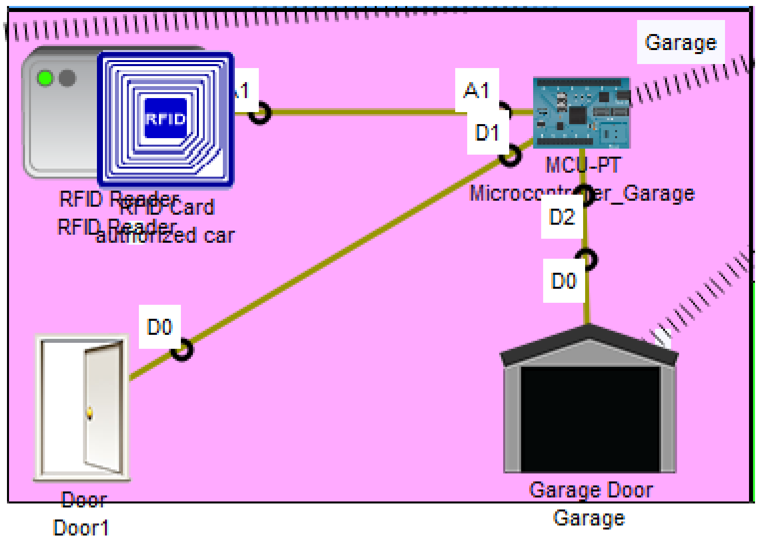


Figure IV 30: Proximity of the Authorized Car to the RFID Reader

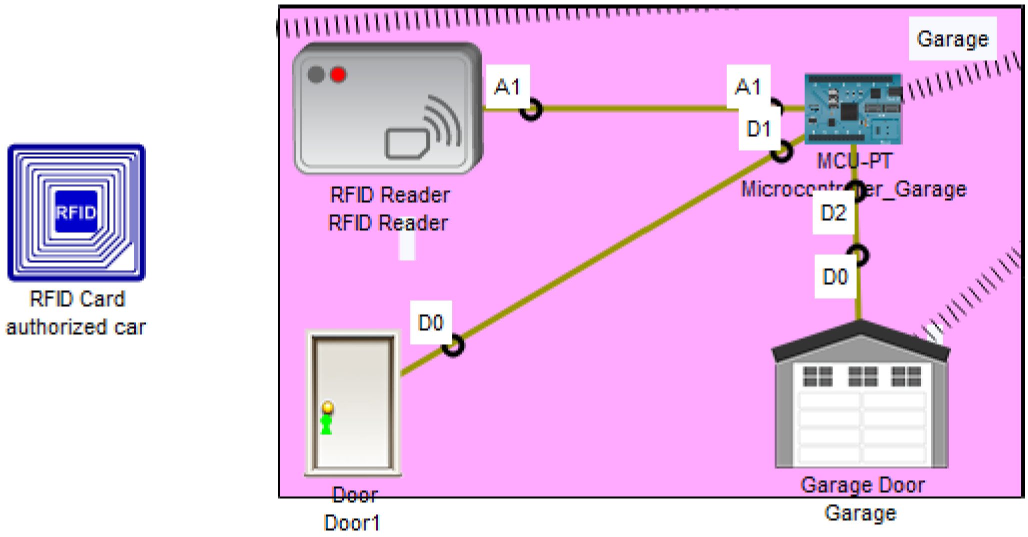


Figure IV 31: Distance between the Authorized Car and the RFID Reader

Scenario 2: In this test, we repeated the process using an unauthorized card near the RFID reader while simultaneously monitoring the garage and motion sensor. As expected, the garage and the door did not open. The system appropriately recognized the unauthorized card and maintained the closed state of both the garage and the door. This outcome aligns with our expectations, confirming the successful execution of the second test.

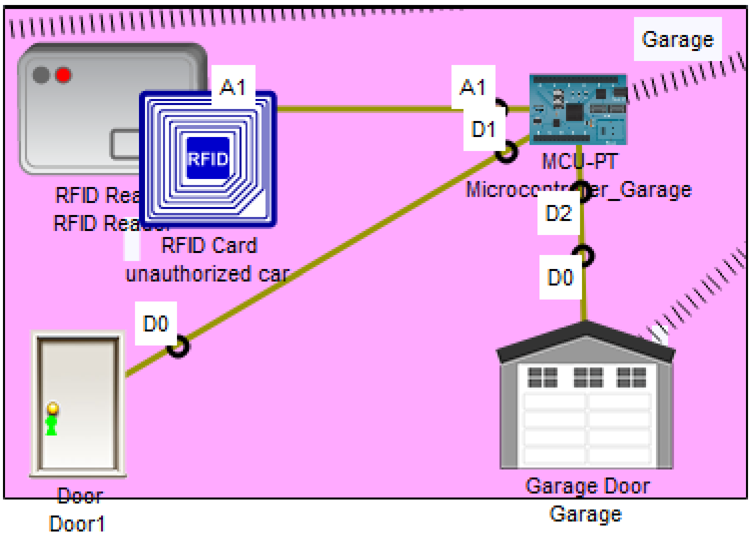


Figure IV 32: Proximity of the Unauthorized Car to the RFID Reader.

By meticulously evaluating both scenarios, we have validated the system's ability to respond appropriately to authorized and unauthorized access attempts, ensuring the security and efficient operation of the garage and the door.

1.2.2 The System for Smoke Detector Monitoring:

In this experiment, our objective is to monitor the smoke level within a home environment. To achieve this, we utilize an old car that emits smoke. By following a specific procedure, we can observe the behavior of the smoke detector and its interaction with other devices in the system.

To initiate the experiment, we press and hold the ALT key while clicking the mouse on the old car. This action triggers the car to emit smoke continuously until the smoke level reaches or exceeds 0.15. Once the smoke detector detects this increased smoke level, it promptly sends a message to the siren, activating an alarm to alert the homeowners. Furthermore, as part of the safety protocol, all windows of the home automatically open, and the smart fan switches to the "on" status, as depicted in Figure 33. The successful execution of this experiment demonstrates the system's ability to detect elevated smoke levels and initiate appropriate responses.

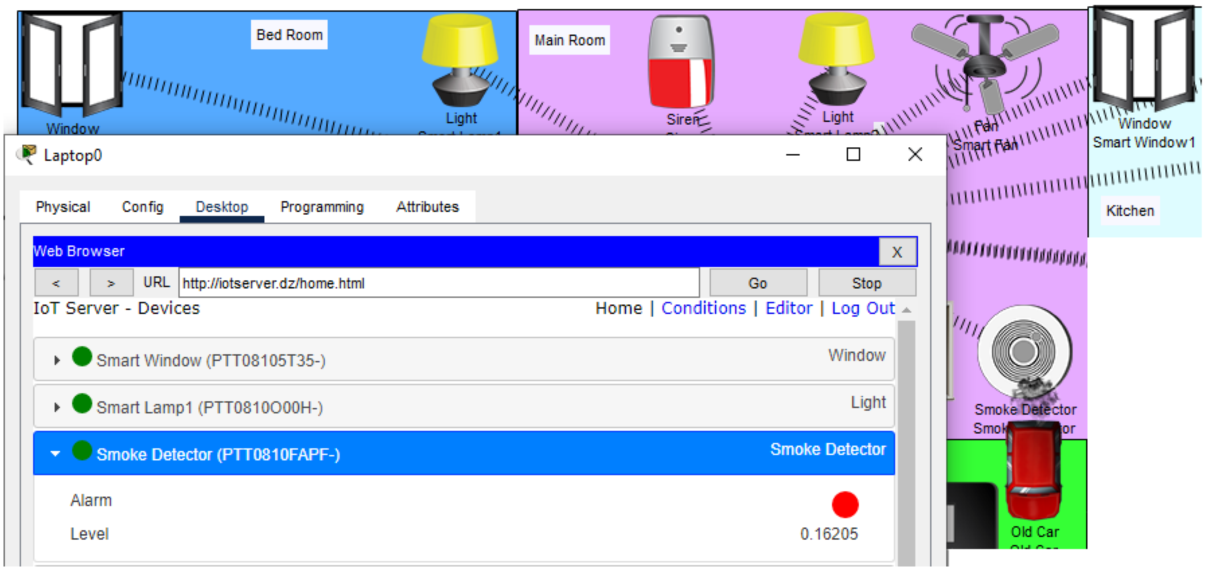


Figure IV 33: Detection of elevated smoke levels and activation of response mechanisms.

Moreover, the interaction between the devices and the system allows for the gradual reduction of the smoke level until it reaches 0.11 and falls below. Consequently, all devices within the smoke level monitoring system automatically turn off. This functionality ensures that the system responds effectively to decreasing smoke levels, providing a safe and controlled environment (see Figure 34).

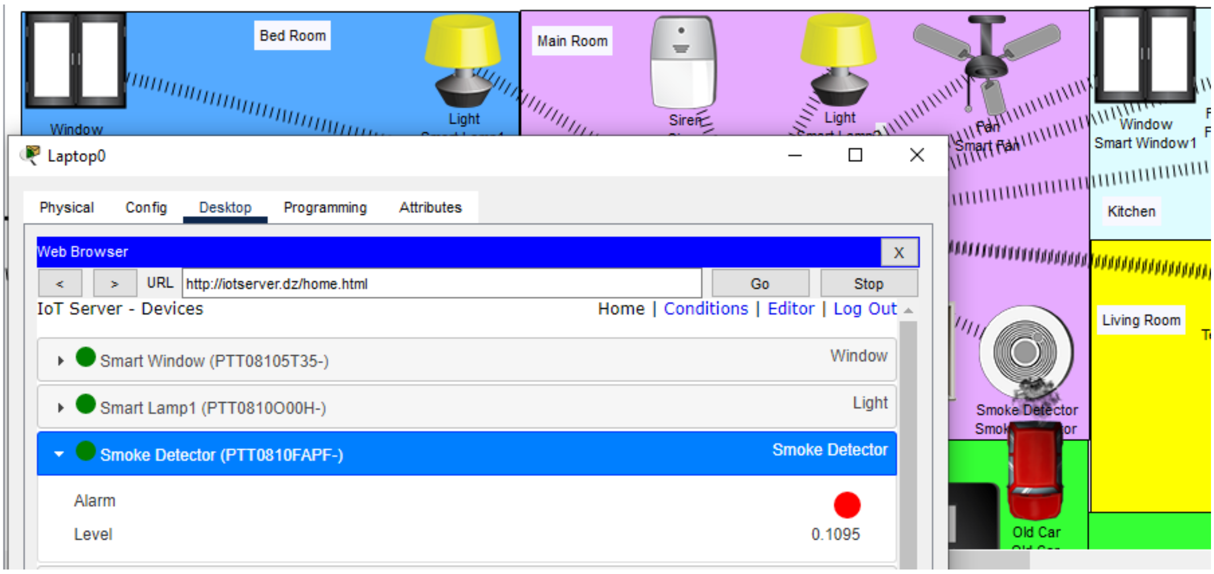


Figure IV 34: Reduction of smoke level and automatic devices shutdown.

Through these comprehensive experiments, we have validated the system's ability to detect and respond to varying smoke levels, safeguarding the homeowners and facilitating efficient smoke control within the monitored area.

1.2.3 The Fire Detection System:

In this system, our primary objective is to detect fires using a fire detector. To effectively test the system's capabilities, we need to simulate fire in two distinct scenarios.

Scenario 1: In this test, we bring the fire source in close proximity to the fire detector. As a result, the system promptly responds by activating the piezo speaker, which emits an audible alarm to alert the homeowners of the fire. Simultaneously, the LCD monitor displays the message "Fire Detected," providing a visual indication of the emergency. Additionally, the fire sprinkler system activates, releasing water to suppress and extinguish the fire in the affected area. The successful execution of this test demonstrates the system's ability to detect fires accurately and initiate appropriate response mechanisms (see figure 35).

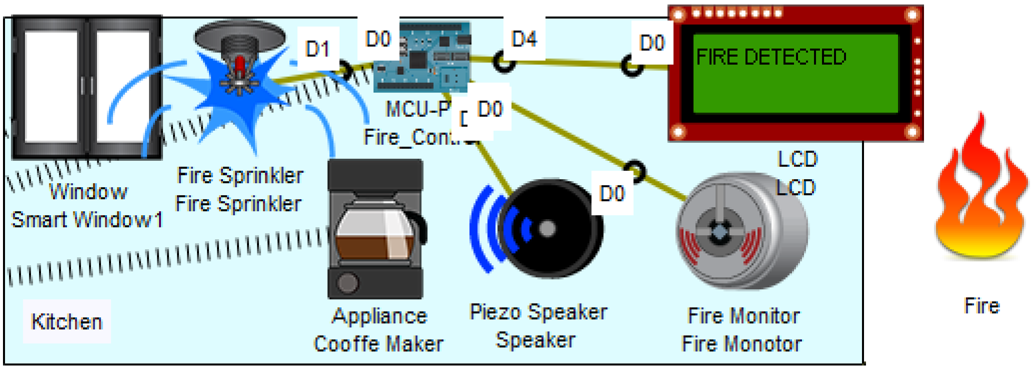


Figure IV 35: The system in "Fire Detection" mode.

Scenario 2: In this scenario, we move the fire source away from the fire detector. As the fire source becomes distant, the system recognizes the reduced threat level. Consequently, the piezo speaker is deactivated, ensuring there is no unnecessary alarm sound. The LCD monitor transitions to display the message "Safe Mode," indicating that the system is in a non-emergency state. Moreover, the fire sprinkler system ceases to release water since there is no fire detected. This outcome confirms the system's ability to adapt its response based on the absence of a fire.

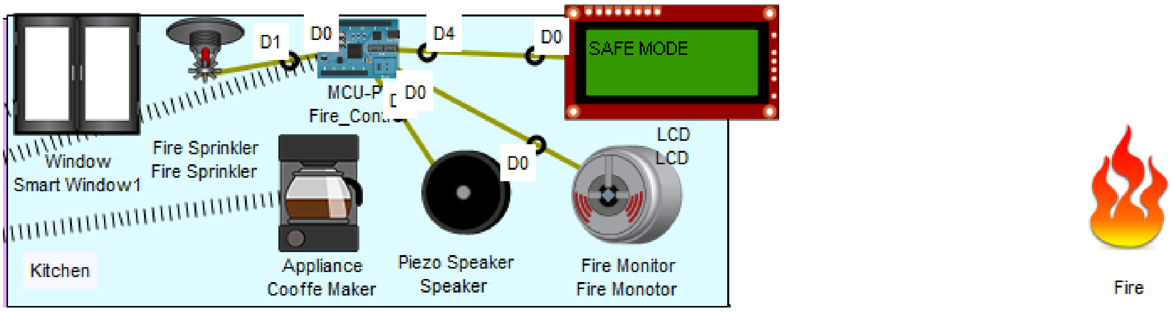


Figure IV 36: The System in "Safe Mode" mode.

By meticulously testing the system in both scenarios, we have verified its efficiency in detecting fires accurately and triggering appropriate responses. The integration of the piezo speaker, LCD monitor, and fire sprinkler system ensures effective communication with homeowners and rapid deployment of fire suppression measures when needed.

1.2.4 The Motion Detection System:

The Motion Detection System is an advanced security system designed to detect any movement in front of a door. Let's explore in detail what happens when something or someone moves within its range.

To begin with, the motion detector is securely connected to the home's door and a webcam through the home gateway, as mentioned in Chapter 3. This integration allows for seamless communication and coordinated actions.

When the motion detector identifies any movement (hold on ALT + passing the mouse on the detector), it activates a button on the detector, which illuminates in a distinct red color. This serves as a visual indication that the system has detected motion and is responding accordingly. in this case, the door automatically locks to prevent unauthorized entry. Simultaneously, the webcam strategically positioned in front of the home begins recording a video of the surrounding area. This video is then streamed live online and sent directly to the homeowner's smartphone or laptop.

Once the system determines that the detected movement is no longer a cause for concern, such as when the intruder has moved away from the motion detector's range, the door automatically unlocks. Consequently, the webcam ceases video recording, conserving resources and ensuring that only relevant footage is captured.

Figure IV 37: The Motion Detection System

The success of this system has been proven through practical implementation and testing. It effectively alerts homeowners and provides them with essential visual information during potential security threats.

1.2.5 The smart lawn irrigation system:

The Advanced Lawn Sprinkler System is a state-of-the-art irrigation system designed to efficiently water your lawn and ensure its optimal maintenance. Let's explore the intricate workings of this system in detail.

At the heart of the system lies a strategically placed lawn sprinkler, connected to a home gateway. This home gateway, in turn, is integrated with a water level monitor. This innovative feature enables seamless communication between components to ensure precise water management.

The water level monitor continuously monitors the water level in your lawn's irrigation system. When the water level falls below the specified value of 5 cm, the monitor sends a message to activate the lawn sprinkler. As a result, the sprinkler system is turned on, ensuring a consistent and even distribution of water throughout your garden. This ensures that every inch of your lawn receives the optimal amount of water required for healthy growth. To illustrate the effectiveness of this system, refer to the figure 38, which demonstrates the successful results achieved through this irrigation experience.

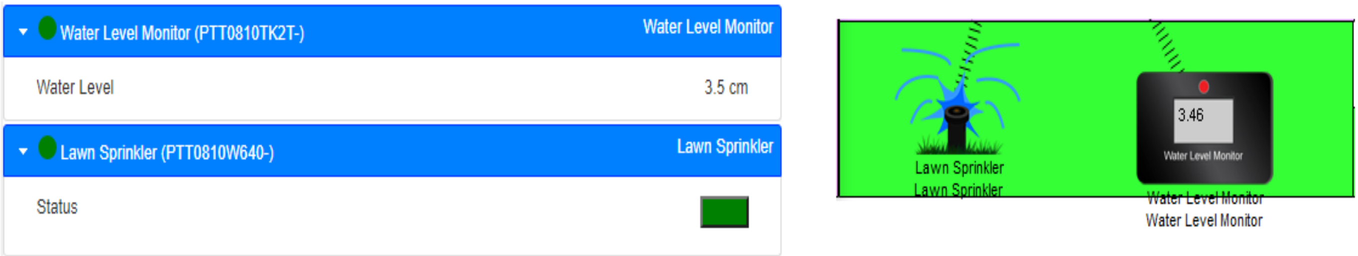


Figure IV 38: The lawn sprinkler system in ON state.

Conversely, when the water level increases and reaches or exceeds the value of 7 cm, the water level monitor promptly sends another message to deactivate the lawn sprinkler. This proactive approach prevents overwatering and ensures that your lawn is not unnecessarily saturated, promoting healthy root development and preventing water wastage. The intelligent integration of the water level monitor, home gateway, and lawn sprinkler guarantees efficient water usage and precise irrigation control.

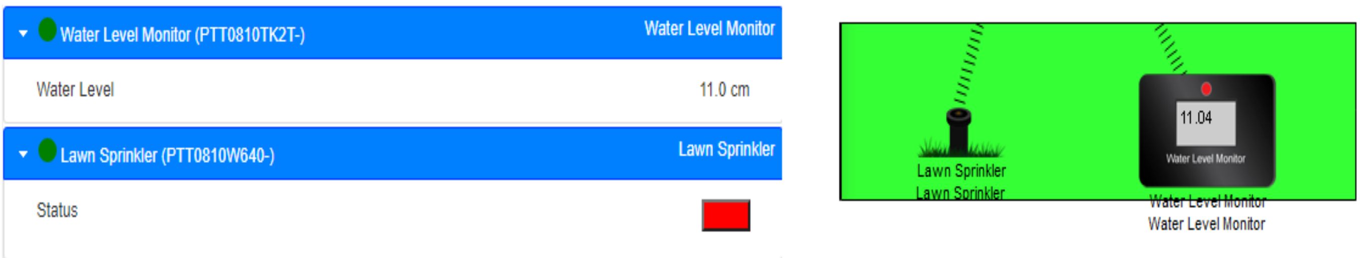


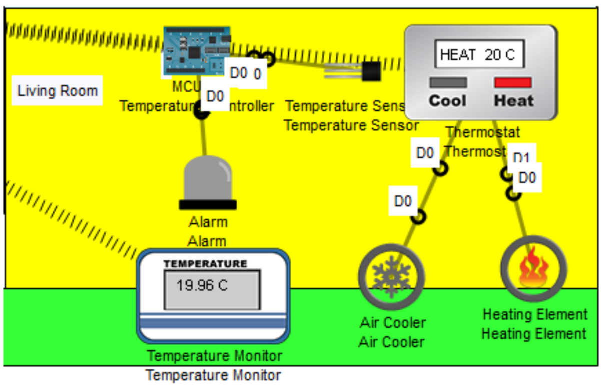
Figure IV 39: The lawn sprinkler system in OFF state

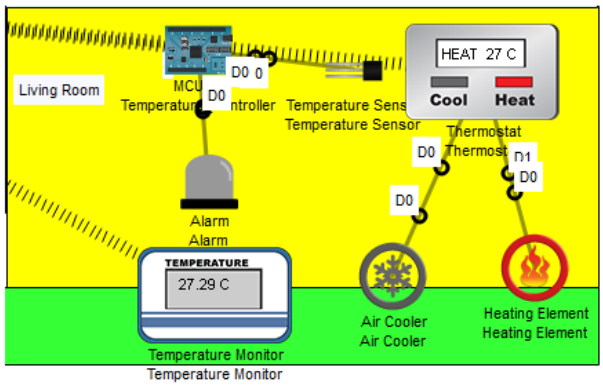
1.2.6 The intelligent temperature monitoring system:

The Smart Temperature Monitoring System represents a crucial component in our project, operating with remarkable efficiency and effectiveness. This system revolves around a thermostat seamlessly connected to two main elements: a heating element responsible for elevating the temperature and an air cooler that operates in direct opposition to the heating element, effectively lowering the temperature. Additionally, the system includes a highly precise temperature sensor and a dedicated temperature monitor device.

The functioning of the system begins when the temperature value reaches approximately 28°C. At this threshold, the temperature sensor promptly detects the change and sends two distinct messages. The first message triggers the alarm device, causing it to activate through the MCU board. Simultaneously, the second message is sent to the homeowner via email is " **temperatur is <<<..HIGH..>>>**" , providing a timely notification about the temperature increase.

Upon receiving the email notification, the homeowner gains remote control capabilities over the thermostat. They can promptly adjust its settings to halt the temperature increase. However, if the homeowner fails to take immediate action, the temperature continues to rise until it reaches 35°C. At this critical point, the thermostat seamlessly transitions from heating mode to cooling mode, thanks to the temperature monitor device. With the cooling mode activated, the air cooler component is engaged, diligently working to reduce the temperature. This process continues until the temperature reaches a comfortable 20°C and the system repeats the aforementioned cycle, maintaining a harmonious balance between heating and cooling to sustain an ideal temperature range. in the end we can say that this experience was also success.





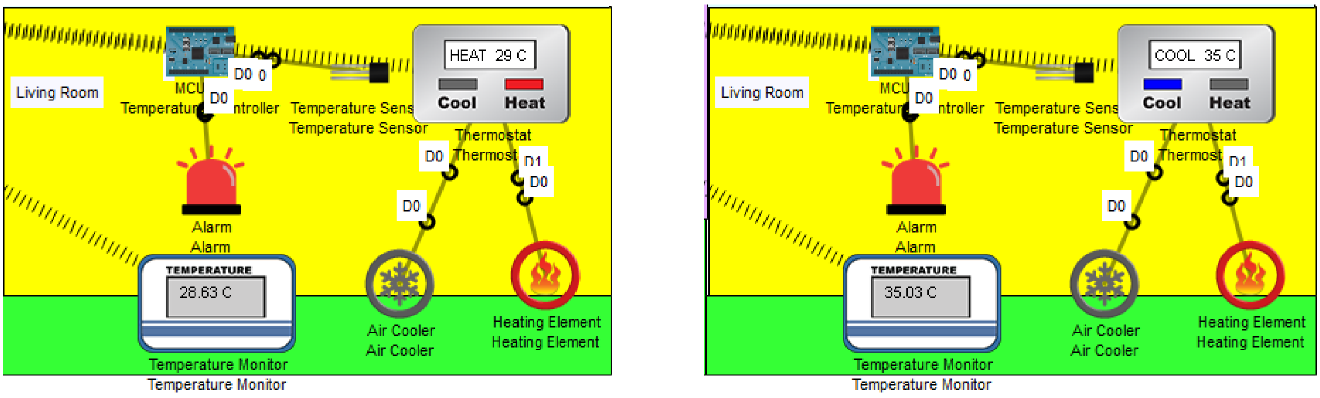


Figure IV 40: the different states of temperature monitor system.

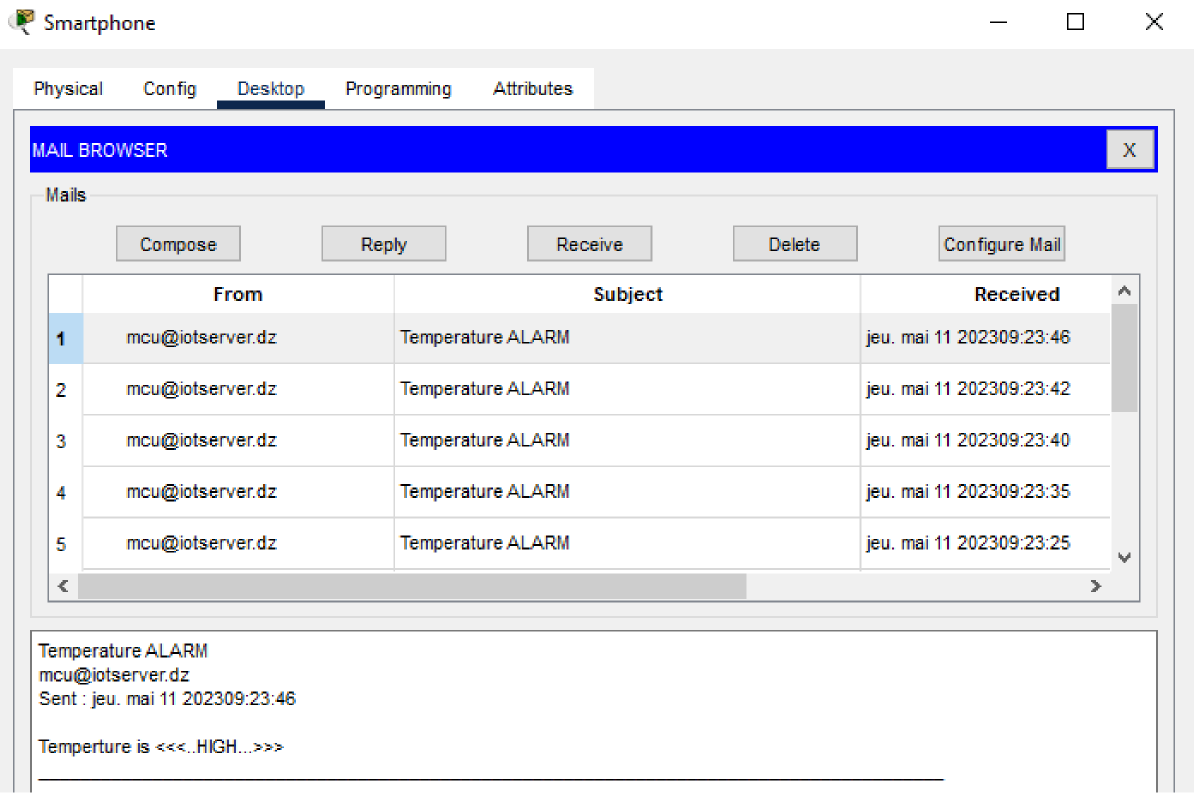


Figure IV 41: Email notification from the smart temperature monitoring system.

In summary, the smart temperature monitoring system revolutionizes climate control through its intelligent and adaptive features. By seamlessly transitioning between heating and cooling modes based on temperature fluctuations, this system guarantees optimal comfort and eliminates the need for constant manual adjustments. With this advanced system in place, homeowners can experience a consistently comfortable indoor environment throughout the year.

Conclusion:

In conclusion, the exploration of smart home systems in this chapter has underscored their transformative impact on our lives. Through practical testing and firsthand experience, we have witnessed how these systems not only simplify but also significantly enhance various aspects of our daily routines. The integration of intelligent devices, automation, and seamless connectivity has ushered in a new era of convenience, efficiency, and safety.

Our journey into the realm of smart homes commenced with a diverse array of devices and systems that we created. From the garage opening and closing system to the smoke and fire detection system, our focus has always been on prioritizing safety. Additionally, the motion detection system has provided a sense of security by promptly alerting us to potential threats and offering real-time video monitoring. The automated lawn sprinkler system, which relies on precise water level measurements, has proven its ability to streamline irrigation processes. Furthermore, the temperature monitoring system has brought enhanced comfort and climate control into our homes. We have also explored other remotely controllable devices such as the coffee maker, further exemplifying the convenience and empowerment offered by these intelligent solutions.

As we conclude this chapter, it is evident that smart home systems have revolutionized our living spaces. They have paved the way for a future where technology seamlessly integrates with our everyday lives, making them safer, more comfortable, and more efficient. With continued advancements and innovations, the potential for smart home systems to enhance our lives and contribute to a sustainable future is immense.

General conclusion:

In conclusion, the study on designing and simulating an IoT network for a smart home using Cisco Packet Tracer has provided valuable insights into creating an efficient and robust network infrastructure. The research covered various aspects including definitions, histories, technologies, and architectures of IoT and smart homes. It also discussed the requirements, considerations, devices, and components for designing a smart home network. Network tests were conducted to evaluate the performance of the IoT network.

Based on the findings, several recommendations are proposed for future research:

- Conduct a comprehensive analysis of IoT architectures and protocols, exploring different models and protocols for their advantages and disadvantages.
- Integrate emerging technologies like edge computing, artificial intelligence, and blockchain to enhance smart home capabilities and security.
- Address security and privacy concerns by developing robust mechanisms, authentication protocols, and encryption techniques.
- Validate the designed network in real-world settings to assess scalability, performance, and compatibility with various devices and user scenarios.
- Focus on energy efficiency and sustainability by optimizing energy consumption, implementing energy-efficient protocols, device power management strategies, and integrating renewable energy sources.

In conclusion, this study provides a strong foundation for further research in the field of smart home IoT networks. By addressing the mentioned recommendations, researchers can contribute to the advancement of smarter, more secure, and sustainable living environments.

Refrences

[1]: S. Cirani, G. Ferrari, M. Picone and L. Veltri., (2019)., Internet of Things Architectures, Protocols and Standards., First edition., Hoboken, NJ: Wiley.

[2]: Fatma AZNAG, Keira TAHANOUT., (2022)., Internet of Things (IoT) technology and the future of payments (Case of Amazon-Go)., Administrative And Financial Sciences Review., Volume: 06/ N° : 01, pp: 466-480.

[3]: Ovidiu Vermesan, Markus Eisenhauer, Harald Sundmaeker, Pateick Geuillemin, Martin Serrano, Elias Z. Tragos, Javier Valino, Arthur van derWees, Alex Glunhak and Roy Bahr., (2022)., Internet of Things cognitive transformation technology research trands and applications, in Cognitive Hyperconnected Digital Transformation., River Publishers: Gistrup, Denmark, pp.17-95.

[4]:H. Mohan and L. C. Manikandan, (2020), A Study Paper on Vision of IOT, International Journal of Scientific Research in Computer Science, Engineering and Information Technology, vol 6, pp.328-335.

[5]: Jorge E. Ibarra-Esquer., Félix F. González-Navarro., Brenda L. Flores-Rios ., Larysa Burtseva and María A. Astorga-Vargas 1., (2017)., Tracking the Evolution of the Internet of Things Concept Across Different Application Domains., MDPI, *Sensors* 2017, *17*(6), 1379; https://doi.org/10.3390/s17061379

[6]: Congressional Research Service (CRS) (2020) Internet of Things (Iot): An introduction, Available at: crsreports.congress.gov (Accessed: Accessed: 12.02.2020).

[7]: P. Suresh, J. Vijay Daniel, V.Parthasarathy, R.H. Aswathy.,(2014)., A state of the art review on the Internet of Things (IoT) History, Technology and fields of deployment., International Conference on Science, Engineering and Management Research (*ICSEMR* 2014).

[8]: Keyur K Patel and Sunil M Patel.,(2016)., Internet of Things-IOT: Definition, Characteristics, Architecture, Enabling Technologies, Application & Future Challenges., International Journal of Engineering Science and Computing., Volume 6 Issue No. 5., pp 6123-6131.

[9]: Kevin Ashton., (2009)., That ‘The Internet of Things’ thing., RFID JOUTNAL.

[10]: Dan-Radu BERTE., (2018)., defining the IoT., Proceedings of the 12th International Conference on Business Excellence 2018., pp. 118-128.

[11]: Ala Al-Fuqaha, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aledhari and Moussa Ayyash., (2015)., Internet of Things: A Survey on Enabling Technologies, Protocols, and

Applications., IEEE Communication Surveys & Tutorials, VOL. 17, NO. 4, Fourth Quarter 2015., pp 2347-2376.

[12]: Pallavi Sethi, Smruti R. Sarangi., (2017)., Internet of Things: Architectures, Protocols, and Applications., Hindawi Journal of Electrical and Computer Engineering Volume 2017, 25 pages https://doi.org/10.1155/2017/9324035

[13]: Zainab.H.Ali, Hesham.A.Ali and Mahmoud M. Badawy., (2015)., Internet of Things (IoT): Definitions, Challenges and Recent Research Directions., International Journal of Computer Applications., Volume 128 – No.1., pp:0975 – 8887.

[14]: Khan, R., Khan, S. U., Zaheer, R., & Khan, S. (2012). Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges. In 2012 10th International Conference on Frontiers of Information Technology (FIT): Proceedings (pp. 257-260). Institute of Electrical and Electronics Engineers Inc. DOI: 10.1109/FIT.2012.53

[15]: Sabah M. Alturfi, Haydar Abdulameer Marhoon and Bahaa Al-Musawi., (2020)., Internet of Things security techniques: A Survey., The 8th International Conference on Applied Science and Technology (ICAST 2020) AIP Conf. Proc. 2290, 040016-1–040016-10; https://doi.org/10.1063/5.0027358.

[16]: Mohammad Aazam, Imran Khan, Aymen Abdullah Alsaffar and Eui-Nam Huh., (2014)., Cloud of Things: Integrating Internet of Things and Cloud Computing and the Issues Involved., 11th International Bhurban Conference on Applied Sciences & Technology (IBCAST) Islamabad, Pakistan, pp 414-419.

[17]: C.Dores, L.P.Reis, N.V.Lopes., (2014)., Internet of Things and Cloud Computing., 9th Iberian Conference on Information Systems and Technologies (CISTI)., DOI: 10.1109/CISTI.2014.6877071.

[18]:K. Jakobs., (2018)., “On Standardizing the Internet of Things and Its Applications” in Qusay F. Hassan., Internet of Things A to Z Technologies and Applications., The Institute of Electrical and Electronics Engineers., Published by John Wiley & Sons, Inc., Hoboken, New Jersey.

[19]: P. Raj and A. C. Raman., (2017)., Title: The Internet of things: enabling technologies, platforms, and use cases., Taylor & Francis, CRC Press., Florida.

[20]: A. Nikoukar, S. Raza, A. Poole, M. Güneş and B. Dezfouli., (2018)., Low-Power Wireless for the Internet of Things: Standards and Applications., in IEEE Access., vol. 6., pp. 67893-67926., doi: 10.1109/ACCESS.2018.2879189.

[21]: O.Vermesan and J.Bacquet., (2017)., Cognitive Hyperconnected Digital Transformation Internet of Things Intelligence Evolution., River Publishers., Delft the Netherlands.

[22]: H. Shariatmadari, S. Iraji, R. Jantti., (2018)., “From Machine-to-Machine Communications to Internet of Things: Enabling Communication Technologies”., in H. Sun, C. Wang, B. I. Ahmad., From Internet of Things to Smart Cities Enabling Technologies., Taylor & Francis, CRC Press., Florida., pp.03-30.

[23]: I. Florea, R. Rughinis, L. Ruse and D. Dragomir., (2017)., Survey of Standardized Protocols for the Internet of Things., 21st International Conference on Control Systems and Computer Science., University Politehnica of Bucharest Bucharest, Romania., DOI 10.1109/CSCS.2017.33.

[24]: D. Galipeau., (2015)., A Brief History of the IoT., United Nations Social Enterprise Facility (UNSEF)., Boracay Island, the Philippines.

[25]: J. C. Haartsen., (2000)., ``The bluetooth radio system,'' *IEEE Pers. Commun.*, vol. 7, no. 1, pp. 28_36, **DOI:** 10.1109/98.824570.

[26]:H. Labiod, H. Afifi, C. De Santis., (2007)., Wi-Fi, Bluetooth, Zig Bee and Wi-max., Springer., Dordrecht, The Netherlands.

[27]: Kuor-Hsin Chang., (2014)., Bluetooth: A Viable Solution for IoT?., IEEE Wireless Communications., pp: 6-7.

[28]: S. D. Padiya and V. S. Gulhane., (2022)., “Analysis of Bluetooth Versions (4.0, 4.2, 5, 5.1, and 5.2) for IoT Applications”., in C. Bhatt, N. Kumar, Ali K. Bashir and M. Alazab., Implementing Data Analytics and Architectures for Next Generation Wireless Communications., IGI Global., United States of America., pp: 153-178., DOI: 10.4018/978-1-7998-6988-7.ch010.

[29]:V. Bhuvaneswari and R. Porkodi., (2014)., The Internet of Things (IoT) Applications and Communication Enabling Technology Standards: An Overview., International Conference on Intelligent Computing Applications., pp:324-329., DOI 10.1109/ICICA.2014.73.

[30]: S. Safaric and K. Malaric., (2006)., ZigBee wireless standard., 48th International Symposium ELMAR-2006, Zadar, Croatia., pp: 259-262.

[31]: Firdaus, Eko Nugroho, Alvin Sahroni., (2014)., ZigBee and Wifi Network Interface on Wireless Sensor Networks., Makassar International Conference on Electrical Engineering and Infonnatics (MICEEI)., South Sulawesi, Indonesia., pp: 54-58.

[32]: T. Elarabi, V. Deep and C. K. Rai., (2015)., Design and Simulation of State-of-Art ZigBee Transmitter for IoT Wireless Devices., IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)., pp: 297-300.

[33]: Vishal Deep and T. Elarabi., (2017)., Efficient IEEE 802.15.4 ZigBee Standard Hardware Design for IoT Applications., International Conference on Signals and Systems (ICSigSys)., pp: 261-265.

[34]: Haxhibeqiri, J., De Poorter, E., Moerman, I., & Hoebeke, J. (2018). A Survey of LoRaWAN for IoT: From Technology to Application. *Sensors*, *18*(11), 3995. https://doi.org/10.3390/s18113995.

[35]:F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melià-Seguí, and T. Watteyne., (2017)., Understanding the Limits of LoRaWAN., IEEE Communications Magazine., pp: 34-40., Digital Object Identifier: 10.1109/MCOM.2017.1600613.

[36]: D.M. Ibrahim., (2019)., Internet of Things Technology based on LoRaWAN Revolution., 10th International Conference on Information and Communication Systems (ICICS)., pp: 234-237.

[37]: C. Sharma and N. K. Gondhi., (2018)., Communication Protocol Stack for Constrained IoT Systems., 2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU), Bhimtal, India, pp. 1-6, doi: 10.1109/IoT-SIU.2018.8519904.

[38]: A. Thantharate, C. Beard and P. Kankariya., (2019)., CoAP and MQTT Based Models to Deliver Software and Security Updates to IoT Devices over the Air," 2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Atlanta, GA, USA, pp. 1065-1070, doi: 10.1109/iThings/GreenCom/CPSCom/SmartData.2019.00183.

[39]: Z. Shelby, K. Hartke and C. Bormann., (2014)., The Constrained Application Protocol (CoAP)., Internet Engineering Task Force (IETF), Request for Comments: 7252 ARM, Category: Standards Track., Universitaet Bremen TZI.

[40]:M. Meruje, M.G. Samaila,V.N.L. Franqueira, M.M. Freire,and P.R. MoraisInácio., (2018)., “A Tutorial Introduction to IoT Design and Prototyping with Examples” in Qusay F. Hassan., Internet of Things A to Z Technologies and Applications., The Institute of Electrical and Electronics Engineers., Published by John Wiley & Sons, Inc., Hoboken, New Jersey.

[41]: M. B. Yassein, M. Q. Shatnawi, S. Aljwarneh and R. Al-Hatmi, (2017), "Internet of Things: Survey and open issues of MQTT protocol," 2017 International Conference on Engineering & MIS (ICEMIS), Monastir, Tunisia, pp. 1-6, doi: 10.1109/ICEMIS.2017.8273112.

[42]: M. Yue, Y. Ruiyang, S. Jianwei and Y. Kaifeng, (2017), "A MQTT Protocol Message Push Server Based on RocketMQ," 2017 10th International Conference on Intelligent Computation Technology and Automation (ICICTA), Changsha, China, pp. 295-298, doi: 10.1109/ICICTA.2017.72.

[43]: N. Naik, (2017), "Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP," 2017 IEEE International Systems Engineering Symposium (ISSE), Vienna, Austria, pp. 1-7, doi: 10.1109/SysEng.2017.8088251.

[44]: K. K. Vaigandla, R. K. Karne, and Rao, A. S. (2021). A Study on IoT Technologies, Standards and Protocols. IBMRD's Journal of Management & Research, 10(2), 7-14.

[45]: V. Ricquebourg, D. Menga, D. Durand, B. Marhic, L. Delahoche and C. Loge, (2006), The Smart Home Concept: our immediate future, 2006 1ST IEEE International Conference on E-Learning in Industrial Electronics, Hammamet, Tunisia, pp. 23-28, doi: 10.1109/ICELIE.2006.347206.

[46]: J. Bugeja, A. Jacobsson, and P. Davidsson., (2018)., “Smart Connected Homes” in Qusay F. Hassan., Internet of Things A to Z Technologies and Applications., The Institute of Electrical and Electronics Engineers., Published by John Wiley & Sons, Inc., Hoboken, New Jersey.

[47]: Justin Jo., (2018)., The Rise of the Smart Home., ECE Senior Capstone Project.,

[48] : Hong, J., Kim, H., & Kim, S. (2015). Smart home energy management system including renewable energy based on ZigBee and PLC. Renewable Energy, 78, 620-627. doi: 10.1016/j.renene.2015.01.023

[49]: Prasad, N., & Gupta, R. (2018). A review on home automation and security systems. Journal of Ambient Intelligence and Humanized Computing, 9(1), 111-125. doi: 10.1007/s12652-017-0532-8.

[50]: Lynggaard, P. (2014). Artificial intelligence and Internet of Things in a “smart home” context: A Distributed System Architecture. (1 ed.) Department of Electronic Systems, Aalborg University.

[51]: Hassan, Q. (2011). Smart homes for the elderly dementia sufferers: Identification and prediction of abnormal behaviour. In Proceedings of the Fifth International Conference on Software, Knowledge, Information Management and Applications (pp. 1-6).

[52]: Daniel Happ., (2018)., “Cloud and Fog Computing in the Internet of Things” in Qusay F. Hassan., Internet of Things A to Z Technologies and Applications., The Institute of Electrical and Electronics Engineers., Published by John Wiley & Sons, Inc., Hoboken, New Jersey.

[53]: Chris., (2016)., The Four Internet of Things Connectivity Models Explained., https://www.channelfutures.com/best-practices/the-four-internet-of-things-connectivity-models-explained.

[54]: I. Zain, A. Rehan, J. Ashraf., (2016)., Internet of Things., International Journal of Engineering Development and Research IJEDR., Volume 4, Issue 3., pp: 1088-1092.

[55]: Joseph Bugeja., (2018)., Smart Connected Homes: Concepts, Risks, and Challenges., studies in computer science no 7, licentiate thesis., Malmö University.

[56]: Jabbar, W. A., Alsibai, M. H., Amran, N. S. S., & Mahayadin, S. K. (2018, June). Design and implementation of IoT-based automation system for smart home. In 2018 International Symposium on Networks, Computers and Communications (ISNCC) (pp. 1-6). IEEE.

[57]: Shemsi, I. (2018). Implementing smart home using cisco packet tracer simulator. International Journal of Engineering Science Invention Research & Development, 4(VII).

[59]: https://www.netacad.com/courses/packet-tracer.

[60]: https://www.javatpoint.com/what-is-a-cisco-packet-tracer.