



Mohamed Khider University  
of Biskra Faculty of Sciences  
and Technology Department  
of Electrical engineering

# MASTER'S THESIS

Electrical Engineering  
Telecommunications  
Networks and Telecommunications

---

Submitted and Defended by:

**Ziadi Echikha**

On: 02 October 2020

# Web-Based smart home system

---

## Board of Examiners:

<b>Mr.</b> Dhiabi Fathi	<b>MCB</b>	University of Biskra	<b>President</b>
<b>Mr.</b> Ouamane Abdelmalik	<b>MCA</b>	University of Biskra	<b>Examiner</b>
<b>Mr.</b> Bekhouche Khaled	<b>MCA</b>	University of Biskra	<b>Supervisor</b>

Academic Year: 2019 - 2020

الجمهورية الجزائرية الديمقراطية الشعبية  
People's Democratic Republic of Algeria  
وزارة التعليم العالي و البحث العلمي  
Ministry Of Higher Education and Scientific Research



**Mohamed Khider Biskra**  
**University Faculty of Sciences**  
**and Technology Electrical**  
**Engineering Department Field:**  
**Telecommunication**  
**Option: Networks and Telecommunication**  
**A Dissertation for the Fulfillment of the Requirement of a**

## **Master's Degree**

# **Web-Based smart home system**

**Presented by:**

Ziadi Echikha

**Favorable opinion of the supervisor:**

Mr. Bekhouche Khaled

**Favorable opinion of the Jury President**

Mr. Dhiabi Fathi

**Stamp and signature**



**Mohamed Khider Biskra**  
**University Faculty of**  
**Sciences and Technology**  
**Electrical Engineering**  
**Department Field:**  
**Telecommunication**

**Option: Networks and Telecommunication**

## ***Theme:***

# **Web-Based smart home system**

**Proposed by:** Mr. Bekhouche Khaled

**Directed by:** Mr. Bekhouche Khaled

## **ABSTRACT**

The goal of our project is to control a smart home with the technology of internet of thing (IoT), and that is through the web server in Raspberry Pi which is the IoT device used. This control can be from anywhere in the world and is not limited to a certain distance.

We accomplished the project as follows: install a webcam server, install Apache local server and finally a remote server Remote.it. It is considered one of the low cost and best solutions for controlling the RPi everywhere and safely because without port forwarding. Finally making sure that controlling the Raspberry Pi is not only in LAN but also in WAN.

**Keyword:** IoT, Raspberry Pi, Apache server, Remote.it.

## **المخلص**

الهدف من مشروعنا هو كيفية التحكم في المنزل الذكي بتقنية انترنت الاشياء (IoT) وذلك من خلال خادم الويب في Raspberry Pi وهو جهاز انترنت الاشياء المستعمل. يمكن أن يكون هذا التحكم من اي مكان في العالم ولا يقتصر على مسافة معينة.

أتمنا هذا المشروع كما يلي: تثبيت خادم كاميرا الويب، تثبيت خادم Apache واخيرا تطبيق Remote.it. يعتبر احد الحلول للتحكم في RPi في كل مكان و بأمان لأنه بدون اعادة توجيه المنفذ وكذلك غير مكلف. أخيرًا ، نتأكد من أن التحكم في Raspberry Pi ليس فقط في LAN ولكن أيضًا في WAN.

**الكلمات المفتاحية:** انترنت الاشياء, جهاز Raspberry Pi, خادم apache, Remote.it.

## **DEDICATION**

I dedicate this modest work

To my dear parents

To my brothers and sisters and all my family

To all my teachers

To all my friends

To people who helped me

And for all who be a part of my life.

## ACKNOWLEDGEMENT

First and foremost thanks to the god, for giving me all the patience, courage, and motivation, for his blessings throughout my research work to complete the research.

Every challenging work needs self-efforts as well as guidance of elders especially those who were very close to our heart and who helped us.

My thanks to my supervisor Dr.Bekhouche Khaled in this research for his sincerity, motivation and guidance to me. Without forgetting to thank all my teachers and jury members.

I would also like to show my gratitude to my Mother a strong and gentle soul who taught me to trust in Allah, believe in hard work and that so much could be done with little. My Father who provided all my needs and for encouragement to reach success and for he be proud of me. And my Uncle Salah Ziadi for being my guardian during my educational career and the second father.

My special thanks go to all people who supported me: brothers Mohamed lamin and Aniss, sister Aya, cousins Maroua, souad, Abdallah and Ferhat, Abdelnour my uncle's wife M.Aldjia, my aunt I.Rachida, uncles Mohammed and Said.

My Big thanks to all persons who halped me: Mario Pojoy, Qays ahmed Qaysaneah, Ismail Ben khalfa and Donia Refas.

I must acknowledge as well the many friends: Rym, Nour El Houda, Maroua, Bouthaina, Chaima, Ibtissem and all my colleagues.

## TABLES LIST

<b>Table 1.1:</b> Comparison of IoT Technologies .....	10
<b>Table 2.1:</b> Characteristics of Different Raspberry Pi's .....	33
<b>Table 3.1:</b> List tags with Description .....	41
<b>Table 3.2:</b> CSS Basic Properties .....	44
<b>Table 3.3:</b> Commonly encountered CSS length/size, keyword, and color units .....	46

## FIGURES LIST

<b>Figure 1.1:</b> RFID Scenario .....	06
<b>Figure 1.2:</b> Architecture of WSN .....	07
<b>Figure 1.3:</b> Cloud connection .....	08
<b>Figure 1.4:</b> IOT Layers .....	16
<b>Figure 1.5:</b> System Configuration using HTTP .....	18
<b>Figure 1.6:</b> System Configuration using MQTT .....	18
<b>Figure 1.7:</b> Protocols Stack of IOT System .....	20
<b>Figure 2.1:</b> Raspberry Pi 3 model B .....	23
<b>Figure 2.2:</b> The Raspberry Pi's Wi-Fi and Bluetooth antenna .....	24
<b>Figure 2.3:</b> The Raspberry Pi's USB ports .....	24
<b>Figure 2.4:</b> The Raspberry Pi's Ethernet port .....	25
<b>Figure 2.5:</b> The Raspberry Pi's 3.5 mm AV jack .....	25
<b>Figure 2.6:</b> The Raspberry Pi's camera connector + RPi camera module .....	25
<b>Figure 2.7:</b> The Raspberry Pi's HDMI port .....	26
<b>Figure 2.8:</b> The Raspberry Pi's micro USB power port .....	26
<b>Figure 2.9:</b> The Raspberry Pi's display connector .....	26
<b>Figure 2.10:</b> The Raspberry Pi's GPIO .....	27
<b>Figure 2.11:</b> The Raspberry Pi's microSD card connector .....	27
<b>Figure 2.12:</b> Logo of Arch Linux .....	28
<b>Figure 2.13:</b> Logo of Pidora .....	29
<b>Figure 2.14:</b> Logo of Operating system Raspbian and Windows 10 IoT .....	29
<b>Figure 2.15:</b> VNC interface .....	31
<b>Figure 2.16:</b> PuTTY Interface .....	32
<b>Figure 2.17:</b> Home automation with Raspberry Pi .....	34
<b>Figure 2.18:</b> Robot car with Raspberry Pi .....	35

<b>Figure 3.1:</b> HTML Element and tag .....	38
<b>Figure 3.2:</b> Logo of PHP .....	47
<b>Figure 4.1:</b> Global schematic of our project .....	51
<b>Figure 4.2:</b> Architecture of IoT .....	52
<b>Figure 4.3:</b> LED .....	53
<b>Figure 4.4:</b> Webcam .....	53
<b>Figure 4.5:</b> Jumper wires .....	54
<b>Figure 4.6:</b> Breadboard .....	54
<b>Figure 4.7:</b> Raspberry Pi Imager .....	56
<b>Figure 4.8:</b> Memory card slot .....	56
<b>Figure 4.9:</b> Raspberry Pi with accessories .....	57
<b>Figure 4.10:</b> Installation of OS on RPi .....	57
<b>Figure 4.11:</b> Raspberry Pi password .....	58
<b>Figure 4.12:</b> Raspberry Pi Configuration interfaces .....	59
<b>Figure 4.13:</b> Network Address .....	60
<b>Figure 4.14:</b> Static WLAN IP .....	61
<b>Figure 4.15:</b> PuTTY Configuration .....	62
<b>Figure 4.16:</b> Remote Desktop .....	63
<b>Figure 4.17:</b> Apache server .....	64
<b>Figure 4.18:</b> Motion Configuration .....	66



<b>Figure 4.19:</b> Live video steaming .....	67
<b>Figure 4.20:</b> LED Control Code .....	68
<b>Figure 4.21:</b> Organogram of LED code .....	69
<b>Figure 4.22:</b> LED control from web page .....	70
<b>Figure 4.23:</b> Apache index .....	70
<b>Figure4.24:</b> Remote.it account .....	71
<b>Figure 4.25:</b> installation of Remote.it package on RPi .....	72
<b>Figure 4.26:</b> Remote.it device services .....	72
<b>Figure 4.27:</b> Proxy URL with port .....	73
<b>Figure 4.28:</b> Remote.it application from smartphone .....	74

## ABBREVIATIONS LIST

**IoT:** Internet of Thing

**RFID:** Radio Frequency Identification.

**P2P:** Point to Point.

**WSN:** Wireless sensor Network.

**LPWAN:** Low Power Wide Area Network.

**BLE:** Bluetooth Low Energy.

**IEEE:** Institute of Electrical and Electronics Engineers.

**Wi-Fi:** Wireless Fidelity.

**LoRa:** Long Range Wide Area.

**GPS:** Global Position System.

**IP:** internet Protocol.

**GSM:** Global System for Mobile.

**GPRS:** General Packet Radio Service.

**WiMAX:** Worldwide Interoperability for Microwave access.

**HTTP:** Hypertext Transfer Protocol.

**HTTPS:** Hypertext Transfer Protocol secure.

**SMTP:** Simple Mail Transfer Protocol.

**FTP:** File Transfer Protocol.

**CoAP:** Constrained application protocol.

**URI:** Uniform Resource Identifier.

**URL:** Uniform Resource Locator.

**TCP:** Transmission Control Protocol.

**UDP:** User Datagram Protocol.

**MQTT:** Message Queue Telemetry Transport.

**M2M:** Machine-to-Machine.

**AMQP:** Advanced Message Queuing Protocol.

**REST:** Representational State Transfer.

**XMPP:** Extensible Messaging and Presence Protocol.

**XML:** Extensible Markup language.

**6LOWPAN:** IPv6 over Low Power WPAN.

**IETF:** Internet Engineering Task Force.

**IPv6:** Internet Protocol version 6.

**WPAN:** Wireless personal Area Network.

**ARM:** Advanced RISC Machine.

**CPU:** Central Processing Unit.

**SoC:** System on Chip.

**GPU:** Graphics Processing Unit.

**RAM:** Random Access Memory.

**USB:** Universal Serial Bus.

**AV:** Audio Visual.

**TRRS:** Tip-Ring-Ring-Sleeve.

**RPI:** Raspberry Pi.

**HDMI:** High Definition Multimedia Interface.

**DSI:** Display Serial Interface.

**SD:** Secure Digital.

**GPIO:** General Purpose Input Output pins.

**OS:** Opening System.

**NOBS:** New Out Of Box Software.

**VNC:** Virtual Network Computing.

**SSH:** Secure Shell.

**PC:** Personal Computer.

**VPN:** Virtual Private Network.

**ADC:** Analog to Digital Converter.

**RTC:** Real-Time Clock.

**BIOS:** Basic Input Output System.

**HTML:** Heading Text markup language.

**CSS:** Cascading style sheets.

**PHP:** PHP Hypertext Preprocessor

**Id:** Identifier.

**Ol:** Ordered lists.

**Ul:** Unordered lists.

**Di:** Definition lists.

**RGB:** Red Green Blue.

**GUI:** Graphical User Interface.

**PIR:** Passive InfraRed sensor.

**DHCP:** Dynamic Host Configuration Protocol.

**RFB:** Remote Frame Buffer.

**URL:** Uniform Resource Locator.

**RTP:** Real Time Protocol.

**RTSP:** Real Time Streaming Protocol.

**RTCP:** Real-time Transport Control Protocol.

# TABLE OF CONTENTS

**DIDICATION**

**ACKNOWLEDGEMENT**

**TABLES LIST .....I**

**FIGURES LIST .....II**

**ABBREVIATIONS LIST .....V**

**TABLE OF CONTENTS .....VIII**

**GENERAL INTRODUCTION .....01**

**Chapter 01: An overview of internet of thing .....04**

1.1 Introduction .....05

1.2 Definition of IoT .....05

1.3 Technologies of IoT .....05

    1.3.1 RFID .....06

    1.3.2 WSN .....06

    1.3.3 Cloud Computing .....07

    1.3.4 LPWAN .....08

    1.3.5 Cellular Technology .....08

    1.3.6 ZigBee .....09

    1.3.7 Bluetooth and BLE .....09

    1.3.8 WiFi .....09

1.4 Comparison of IoT Technologies .....10

1.5 Application of IoT .....10

    1.5.1 Industrial application .....10

1.5.2 Smart agriculture .....	11
1.5.3 Healthcare .....	11
1.5.4 Smart Home .....	12
1.5.4.1 History .....	12
1.5.4.2 Definition of Smart Home .....	12
1.5.4.3 System of Home automation .....	13
1.5.4.4 Smart Home Technologies .....	13
1.5.4.5 Smart Home Components .....	14
1.6 IoT Architecture .....	15
1.6.1 IoT Layers .....	15
1.6.2 IoT Protocols .....	17
1.7 Conclusion .....	21
<b>Chapter 02: Raspberry Pi Board .....</b>	<b>22</b>
2.1 Introduction .....	23
2.2 Definition of Raspberry Pi .....	23
2.3 Components of Raspberry pi .....	23
2.3.1 The Raspberry Pi's system-on-chip (SoC) .....	24
2.3.2 Wi-Fi and Bluetooth antenna .....	24
2.3.3 The Raspberry Pi's ports .....	24
2.4 Accessories of Raspberry Pi .....	27
2.5 Operating System .....	28
2.5.1 Arch Linux .....	28
2.5.2 Pidora .....	29

2.5.3 Raspbian .....	29
2.6 Opening the Raspberry Pi .....	30
2.6.1 Download NOOBS .....	30
2.6.2 Install SD Card .....	30
2.6.3 Connect the Cable .....	30
2.6.4 Install Raspbian .....	30
2.7 Remote Access .....	31
2.7.1 VNC .....	31
2.7.2 SSH .....	32
2.8 Characteristics of Raspberry Pi Types .....	33
2.9 The uses of Raspberry Pi .....	34
2.9.1 Low-Cost Desktop PC .....	34
2.9.2 Home Automation System .....	34
2.9.3 Web server .....	35
2.9.4 Robot .....	35
2.9.5 VPN .....	35
2.10 Raspberry Pi's Performances and constraints .....	35
2.11 Conclusion .....	36
<b>Chapter 03: Programming Language .....</b>	<b>37</b>
3.1 Introduction .....	38
3.2 HTML .....	38
3.2.1 Definition of HTML .....	38
3.2.2 Tags and Elements .....	38

3.2.3 Basic HTML .....	39
3.2.3.1 HTML Document .....	39
3.2.3.2 HTML Headings .....	39
3.2.3.3 HTML Paragraphs .....	39
3.2.3.4 HTML Body .....	39
3.2.4 HTML Attributes .....	39
3.2.5 HTML Links .....	40
3.2.6 HTML Images .....	40
3.2.7 HTML Lists .....	40
3.2.8 HTML Table .....	41
3.3 CSS .....	41
3.3.1 Definition of CSS .....	41
3.3.2 Ways to Insert CSS .....	42
3.3.3 CSS Syntax .....	43
3.3.4 CSS selector .....	43
3.3.5 CSS Basic Properties .....	44
3.3.6 CSS Units .....	45
3.4 PHP .....	46
3.4.1 Definition of PHP .....	46
3.4.2 PHP Syntax .....	47
3.4.3 PHP comment .....	47
3.4.4 PHP Variables .....	48
3.4.5 PHP Types of data .....	48



3.5 Conclusion .....	49
<b>Chapter 04: Implementation part .....</b>	<b>50</b>
4.1 Introduction .....	51
4.2 Global schematic of the project .....	51
4.3 Hardware section .....	52
4.3.1 Raspberry Pi .....	52
4.3.2 Sensor .....	52
4.3.3 Jumper wire .....	53
4.3.4 Breadboard .....	54
4.4 Software section .....	54
4.4.1 Raspbian OS .....	54
4.4.2 PuTTY .....	55
4.4.3 VNC .....	55
4.4.4 Apache server .....	55
4.4.5 Remote.it application .....	55
4.5 Implementation section .....	55
4.5.1 Set up of Raspberry Pi 3 Model B .....	55
4.5.1.1 Operating System Installation .....	55
4.5.1.2 Running the Raspberry Pi .....	56
4.5.2 Static IP .....	58
4.5.3 SSH Remote access .....	61
4.5.4 VNC Remote Desktop .....	62
4.5.5 Apache server .....	63

4.5.6 Webcam server using RPi .....	65
4.5.7 LED Control .....	67
4.5.8 Access from anywhere with Remote.IT .....	71
4.6 Conclusion .....	75
<b>GENERAL CONCLUSION .....</b>	<b>76</b>
<b>BIBLIOGRAPHY .....</b>	<b>79</b>

**GENERAL  
INTRODUCTION**

## GENERAL INTRODUCTION

The internet of things (IoT) is a catch-all term for the growing number of electronics that are smart devices, but are connected to the internet to send data, receive instructions or both. And one of the best IoT device to use is the Raspberry Pi. That is a computer far more powerful than many IoT devices, is a good candidate for IoT processing.

Technology is advancing with every passing day and owning smart home is becoming a necessity. In the past, smart homes were considered to be part of a luxurious life style but today they have become an important part of our lives.

There are many ways to control the smart home, the most important of which is via a web page, and there are different technologies used by the smart home, such as WiFi, Zigbee and Bluetooth but now we are in permanent search to use technology that covers a greater distance and safely and this is to control the smart home from any place in the world.

### **Memory objective:**

The goal of this project is:

- Studying the concept and basics of internet of things (with smart home basics).
- Studying of basics Raspberry Pi.
- Creating a server web page.
- Connection and control the IoT device from anywhere in the world.

### **Memory outline:**

We split the project into four chapters:

First chapter:

It consists of an IoT overview and more information about technologies, architecture, protocols and some applications including smart home. Some of these applications are dedicated to our project.

Second chapter:

We explained in details Raspberry Pi which is considered as an internet of things device. Its characteristics and uses in the world of technology are also shown.

Third chapter:

In this chapter we have put the basics of webpage programming languages such as HTML, PHP, etc.

Last chapter:

It is the most important chapter and it is the practice of many information explained in the previous chapters. This is split out into four sections:

- Overall system diagram produced
- Hardware part
- Software part
- And the implementation section.

# **Chapter 1:**

**An overview of Internet of thing**

## 1.1 Introduction:

Internet of things “IoT” is an increasing number of connected smart devices, which have their identity and ability to develop advanced communication, such as phones, home appliances, smart cars...etc. that link to many sensors and actuators that allow to measure the environment for remote control.

Everyone dreams about automation of many technologies at home (lighting, video surveillance ...) with this home automation that use IoT technology.

In this chapter we will give an overview about internet of thing, we'll see some of its technologies passing through its applications such as “ smart home”, then the architecture of Iot with protocols that manage it.

## 1.2 Definition of Internet of Things:

The Internet of Things, or IoT, refers to the billions of physical devices around the world that are now connected to the internet, collecting and sharing data. Thanks to cheap processors and wireless networks, it's possible to turn anything, from a pill to an aeroplane to a self-driving car into part of the IoT. This adds a level of digital intelligence to devices that would be otherwise dumb, enabling them to communicate real-time data without a human being involved, effectively merging the digital and physical worlds. [1]

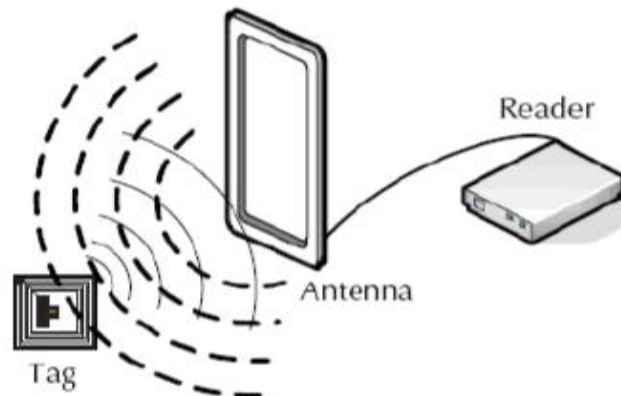
Internet of things is an idea from computer science: connecting ordinary things like lights and doors to a computer network to make them "intelligent". An embedded system or a computer connects each thing together in a network and to the internet. Some technologies used for the internet of things are: RFID and meshnets. The connections allow each thing to collect and exchange data, and we can control them remotely or by setting rules or chains of actions. IOT improves the ease of life of humans and their daily activities. Experts estimate that the IOT will consist of almost 50 billion objects by 2020. [2]

## 1.3 Technologies of IoT:

### 1.3.1 RFID Technology:

RFID is not new and it was popular in the early 20<sup>th</sup> century. Initially, it was based on radio waves and later radio waves combined with radar signals. They can be used to provide P2P connection between objects. RFID consists of three main components such as a

transponder or tag to carry data, which is located on the object to be identified, an interrogator or reader, which reads the transmitted data, and Middleware, which forward the data to another system, such as a database, a PC or robot control system. [3]



**Figure 1.1:** RFID Scenario [4]

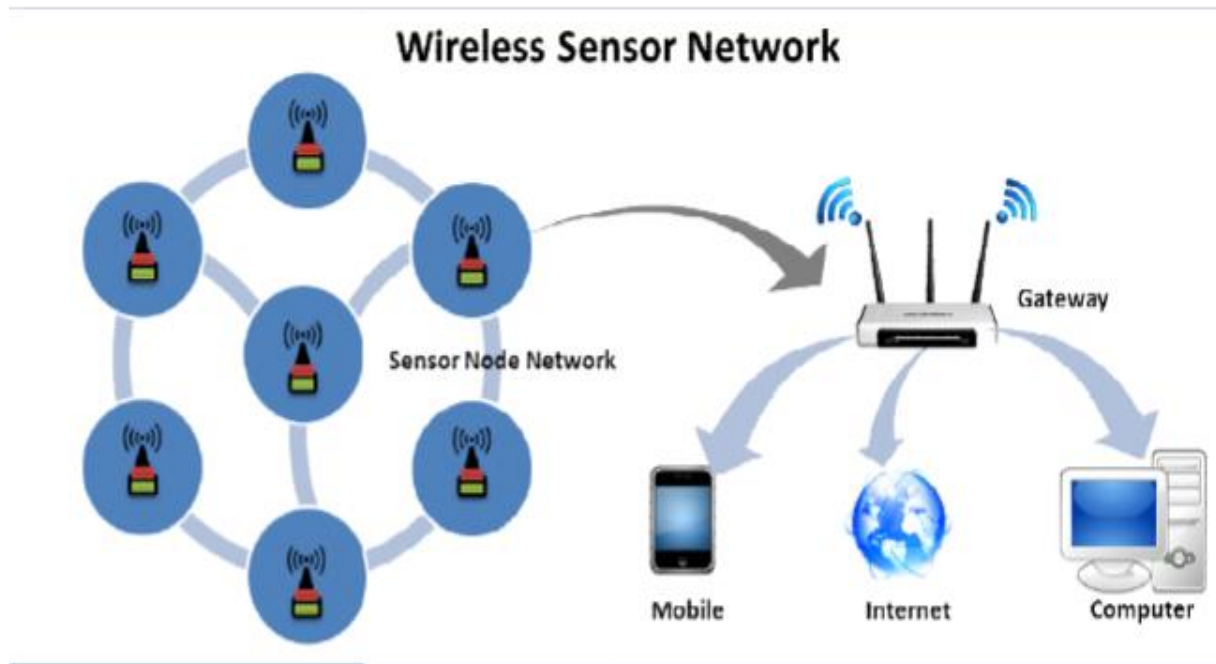
### 1.3.2 Wireless Sensor Network (WSN):

(WSN) refers to a group of spatially dispersed and dedicated sensors for monitoring and recording the physical conditions of the environment and organizing the collected data at a central location. WSNs measure environmental conditions like temperature, sound, pollution levels, humidity, wind, and so on. [5]

WSN are heterogeneous in natures, and some time contain thousands of tiny sensor nodes that are distinguish in their characteristics. A single node normally has very low processing, storage and transmitting capability .With these limited physiognomies still, these nodes are being used in many commercial and military applications. In fact, development of WSNs, was originally encouraged by military applications for surveillance in battle zones. [6]

(WSN) will play a fundamental role in the future Internet of Things (IoT), with millions of devices actively exchanging confidential information with one another in a multi-hop manner. Ensuring secure end-to-end communication channels is crucial to the success of innovative IoT applications, as they are essential to limit attacks' impacts and avoid exposure of information. [7]





**Figure 1.2:** Architecture of WSN [8]

### 1.3.3 Cloud Computing:

Cloud computing is the delivery of different services through the Internet. These resources include tools and applications like data storage, servers, databases, networking, and software. Rather than keeping files on a proprietary hard drive or local storage device, cloud-based storage makes it possible to save them to a remote database. As long as an electronic device has access to the web, it has access to the data and the software programs to run it. Cloud computing is a popular option for people and businesses for a number of reasons including cost savings, increased productivity, speed and efficiency, performance, and security. [9] The transfer of extensive data to and from cloud computers presents a challenge due to limited bandwidth. [10]



**Figure 1.3:** Cloud connection.

#### **1.3.4 Low Power Wide Area Networks (LPWAN):**

LPWANs are the new phenomenon in IoT. By providing long-range communication on small, inexpensive batteries that last for years, this family of technologies is purpose-built to support large-scale IoT networks sprawling over vast industrial and commercial campuses. [11]

LPWANs achieve that feat by having their IoT devices send only small packets of information periodically or even infrequently—status updates, reports, etc.—upon waking from an external trigger or at a preprogrammed interval. However, with the advent of the cellular LPWAN, there is now more flexibility in the definition of “low-power” and “wide-area”. [12]

#### **1.3.5 Cellular Technology:**

Cellular IoT connects physical devices (such as sensors) to the internet using essentially the same technology as your smartphone. Instead of needing to create a new, private network to operate your IoT devices, they can piggyback on existing mobile networks. [13]

Cellular IoT is designed to meet the requirements of such low-power, long-range applications. It takes existing technology that we already use every day for our smartphones and scales it back to meet the needs of low-power devices. [14]

Cellular next-gen 5G with high-speed mobility support and ultra-low latency is positioned to be the future of autonomous vehicles and augmented reality. 5G is also expected to enable

real-time video surveillance for public safety, real-time mobile delivery of medical data sets for connected health, and several time-sensitive industrial automation applications in the future. [11]

### **1.3.6 Zigbee:**

Zigbee wireless communication technology is a kind of newly arisen wireless network technology; the characteristic is short distance communication, low speed, low power dissipation, and low cost. It, application of Zigbee wireless communication technology, makes that inconvenient wire repeat can be avoided in the area of home, factory, hospital, etc Zigbee will play an important role in wireless sensor network. [15]

ZigBee is an IEEE 802.15.4-based specification for a suite of high-level communication protocols used to create personal area networks with small, low-power digital radios, such as for home automation, medical device data collection [16]

### **1.3.7 Bluetooth and BLE:**

Bluetooth is a short-range wireless communication technology that allows devices such as mobile phones, computers, and peripherals to transmit data or voice wirelessly over a short distance. [17] The IEEE standardized Bluetooth as IEEE 802.15.1. [18]

Bluetooth Low Energy (BLE) is an emerging low-power wireless technology developed for short-range control and monitoring applications that is expected to be incorporated into billions of devices in the next few years.

In contrast with previous Bluetooth flavors, BLE has been designed as a low power solution for control and monitoring applications. BLE is the distinctive feature of the Bluetooth 4.0 specification. [19]

In summary, Bluetooth and Bluetooth Low Energy are used for very different purposes. Bluetooth can handle a lot of data, but consumes battery life quickly and costs a lot more. BLE is used for applications that do not need to exchange large amounts of data, and can therefore run on battery power for years at a cheaper cost. It all depends on what you're trying to accomplish. [20]

### **1.3.8 Wireless Fidelity (Wi-Fi):**

Wi-Fi is the term used to describe a class of certified wireless networking products conforming to an industry standard designated by IEEE as "802.11b", it is a new

technology that is preferred for wireless local area networking in both business and home environments. [21]

#### 1.4 Comparison of IoT technologies:

There are a difference between the technologies of internet of thing as shown on **Table 1.1**:

<b>Connectivity standard</b>	<b>Data rate (Bandwidth)</b>	<b>Power consumption</b>	<b>Range</b>	<b>Costs</b>
<b>WiFi</b>	Very high (100+Mbps)	Medium	Low (25-50 meters)	Low
<b>Bluetooth</b>	High (25 Mbps)	Low	Medium (50-100 meters)	Very low
<b>Cellular</b>	Medium (3G : 4 Mbps, 4G : 15 Mbps)	High	High (1-5 kilometers)	Medium
<b>LPWAN (Sigfox, LoRa)</b>	Very low (LoRa 50 Kbps Sigfox 0.1 Kbps)	Low	High (LoRa: 5-20 Km Sigfox: 10-40km)	Low
<b>RFID</b>	Very low (100 Kbps)	Low	Low (1-10 meters)	Very low

**Table 1.1:** Comparison of IoT technologies

#### 1.5 Application of IoT:

There are many applications of IoT to mention like:

##### 1.5.1 Industrial application:

An Industrial Internet of Things (IIoT) is emerging as a commercially viable embodiment of the IoT where physical sensors gather data readings from the field and deliver the traffic to

the Internet. The collected real-time “big data”, in turn, allows optimizing entire industry verticals with enormous return of investments. [22]

In recent years a wide range of industrial IoT applications have been developed and deployed. Evolution of this starts from RFID technology, which allows microchips to transmit the identification information to a reader through wireless communication. By using RFID readers, people can identify, track, and monitor any objects attached with RFID tags automatically. Another technology is the wireless sensor networks (WSNs), which mainly use interconnected intelligent sensors to sense and monitoring. Its applications include environmental monitoring, industrial monitoring, traffic monitoring. Both RFID and WSN are used to develop IoT. [23]

### **1.5.2 Smart Agriculture:**

IoT and smart agriculture using automation. Monitoring environmental factors is the major factor to improve the yield of the efficient crops. [24] To change this situation and promote fast development of agricultural information, it is necessary to use cloud computing and visualization technology to construct “agricultural information cloud”, combine IOT technology and RFID technology, so as to realize smart agriculture. [25]

Smart agriculture is a broad term that collects food production practices powered by Internet of Things, big data and advanced analytics technology. The most common IoT applications in smart agriculture are:

1. Sensor-based systems for monitoring crops, soil, fields, livestock, storage facilities, or basically any important factor that influences the production.
2. Smart agriculture vehicles, drones, autonomous robots and actuators.
3. Connected agriculture spaces such as smart greenhouses or hydroponics.
4. Data analytics, visualization and management systems. [26]

### **1.5.3 Healthcare:**

IoT promises market potential in the field of e-health services and the telecommunications industry. IoT can enhance business intelligence in hospitals and ease serving the patients in health sectors. The Internet of Things can improve the health grounds and prevent diseases by providing ongoing monitoring activities to ordinary people or prone patients. The usages of IoT can develop some platforms which provide smart and innovative services to patients and people in need of medical attention. Furthermore, improves their health, Security, ease of

access to emergency medical care, continuing care, and quick support also improving the quality of life. [27]

#### **1.5.4 Smart Home:**

##### **1.5.4.1 History:**

Early home automation began with labour-saving machines. Self-contained electric or gas powered home appliances became viable in the 1900s with the introduction of electric power distribution and led to the introduction of washing machines (1904), water heaters (1889), refrigerators, sewing machines, dishwashers, and clothes dryers.

In 1975, the first general purpose home automation network technology, X10, was developed. It is a communication protocol for electronic devices. It primarily uses electric power transmission wiring for signalling and control, where the signals involve brief radio frequency bursts of digital data, and remains the most widely available. By 1978, X10 products included a 16 channel command console, a lamp module, and an appliance module. Soon after came the wall switch module and the first X10 timer.

By 2012, in the United States, according to ABI Research, 1.5 million home automation systems were installed. [28]

##### **1.5.4.2 Definition of Smart Home:**

A smart home can be defined as a home equipped with sensors, middleware system, and communication interfaces, which anticipate and respond to the needs and requirements of the inhabitants, working to promote their luxury, convenience, security, and entertainment through the supervision of technology within home .A smart home can provide a variety of services and automated tasks as room-temperature control and smart air conditioner to complex tasks as analysis or prediction of the location of a resident, to behaviour or health status recognition of an occupant living at home. [29]

Smart Home is the term commonly used to define a residence that uses a Home Controller to integrate the residence's various home automation systems. The most popular Home Controllers are those that are connected to a Windows based PC during programming only, and are then left to perform the home control duties on a standalone basis. Integrating the home systems allows them to communicate with one another through the home controller, thereby enabling single button and voice control of the various home systems simultaneously, in preprogrammed scenarios or operating modes. [30]

### 1.5.4.3 Systems of Home Automation:

There are 3 main network types of home automation systems that can be installed. These systems are categorized as power line systems, wired systems and wireless systems.

- **Power Line Systems:** This is called the most affordable system. Power line systems rely on the existing power lines to transfer things such as security camera feeds and lighting information to a common control interface. Usually, these are X10 technology based systems.
- **Wired Systems:** Wired systems use cables to communicate information. It is easier to install. Hardwiring all of the systems together through cables to a common junction point, the system is developed.
- **Wireless Systems:** For this system no need to use traditional cables. Wireless systems integrate with Wi-Fi networks, meaning that they are easily compatible with any existing home networks, such as the ones formed by computers. However, some wireless systems use a different radio frequency for making them incompatible with open networks. [31]

### 1.5.4.4 Smart Home Technologies:

There are many of technologies like: Bluetooth, Wi-Fi, Zigbee, X10, Z-wave ...

- **X10:**

X10 is one of the methods I'll cover that allows you to remotely control the power of any device plugged into the standard ring main in your home. The lights, electric teakettle, and toaster are all examples of existing devices in this category. Additionally, I'll cover devices that were originally invented to be controlled by X10 such as motorized curtain rails. X10 achieved its market penetration by being fairly cheap and very easy to install. [32]

- **Z-Wave:**

Z-Wave is the world's leading wireless Smart Home automation technology. Smart solutions based on Z-Wave technology offer a great convenience for homeowners, allowing them to save time and money. [33]

Z-Wave devices communicate with each other through an uninterrupted, low-band radio frequency. Devices with a built-in Z-Wave chip are capable of transforming ordinary

appliances into “smart” appliances. Using a Z-Wave gateway or smartphone, you can control, monitor and automate almost any electrical appliance around your house. From operating your lights to opening and closing blinds, Z-Wave gives you complete control over your home and you can easily install more Z-Wave enabled devices whenever you choose to. [33]

#### **1.5.4.5 Smart home Components:**

- **Sensors/Devices:**

Sensors or devices help in collecting data from the surrounding environment. The device may have more than one sensor that can be bundled together to do more than just sensing things. For example, our phone is a device that has many sensors such as GPS, accelerometer, camera but our phone simply does not understand things. [34]

- **Connectivity:**

That collected data is sent to a cloud infrastructure but it needs a medium for transport. The sensors can be connected to the cloud through various mediums of communication and transports such as cellular networks, satellite networks, Wi-Fi, Bluetooth, WAN, low power wide area network and many more. Every option we choose has some specifications and trade-offs between power consumption, range, and bandwidth. So, choosing the best connectivity option in the IOT system is important. [35]

- **Data Processing:**

Once the data is collected and it gets to the cloud, the software performs processing on the acquired data. This can range from something very simple, such as checking that the temperature reading on devices such as AC or heaters is within an acceptable range. It can sometimes also be very complex, such as identifying objects (such as intruders in your house) using computer vision on video. [35]

- **User Interface:**

The information is made useful to the end-user in some way. This could be via an alert to the user (email, text, notification, etc). For example, a text alert when the temperature is too high in the company’s cold storage. Also, a user might have an interface that allows them to proactively check in on the system. For example, a user might want to check the video feeds in their house via a phone app or a web browser. However, it’s not always a one-way street.



Depending on the IoT application, the user may also be able to perform an action and affect the system. For example, the user might remotely adjust the temperature in the cold storage via an app on their phone. And some actions are performed automatically. Rather than waiting for you to adjust the temperature, the system could do it automatically via predefined rules. And rather than just call you to alert you of an intruder, the IoT system could also automatically notify relevant authorities. [36]

## **1.6 IOT Architecture:**

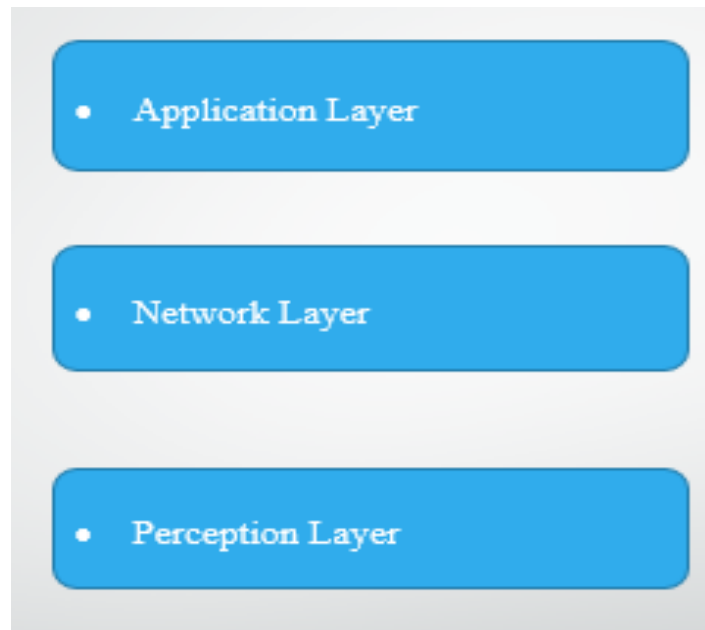
### **1.6.1 IOT layers:**

There is no single consensus on architecture for IoT, which is agreed universally. Different architectures have been proposed by different researchers. [37]

The Internet of Things is considered as the third wave of the World Wide Web after static web pages and social networking's based web. The IoT is a worldwide network that connects different type of objects at anytime and anywhere via a popular internet protocol named Internet Protocol (IP). According to most of the researcher's opinions about conventional IoT architecture, it is considered as three layers: -

- Perception Layer
- Network Layer
- Application Layer

In other aspects, some researchers analyzed one more layer which is also included in IoT's latest architecture that is a support layer that lies between the application layer and network layer. The support layer consists of fog computing and cloud computing. The cloud computing is also the hottest topic today in research. The mostly finalized researchers' view about IoT architecture is shown in **Figure 1.4**, in given below. [38]



**Figure 1.4:** IOT layers [38]

### •Perception Layer:

The main task of the perception layer is to perceive the physical properties of things around us that are part of the IoT. This process of perception is based on several sensing technologies (e.g. RFID, WSN, GPS, etc.). In addition, this layer is in charge of converting the information to digital signals, which are more convenient for network transmission. However, some objects might not be perceived directly. Thus, microships will be appended to these objects to enhance them with sensing and even processing capabilities. Indeed, nanotechnologies and embedded intelligence will play a key role in the perception layer. The first one will make chips small enough to be implanted into the objects used in our everyday life. The second one will enhance them with processing capabilities that are required by any future applications. [39]

### •Network Layer:

This layer, also called transport layer, including access network and core network, provides transparent data transmission capability. By the existing mobile communication network, radio access network, wireless sensor network (WSN) and other communications equipment, such as GSM, GPRS, Wimax, Wi-Fi, Ethernet, etc., the information from perception layer can be sent to the upper layer. At the same time, this layer provides an

efficient, reliable, trusted network infrastructure platform to upper layer and large scale industry application. [40]

### •Application Layer:

The application layer is responsible for delivering application specific services to the user. It defines various applications in which the Internet of Things can be deployed, for example, smart homes, smart cities, and smart health. [37]

It is the interface between the end devices and the network and it is implemented through a dedicated application at the device end. Like for a computer, application layer is implemented by the browser. It is the browser which implements application layer protocols like HTTP, HTTPS, SMTP and FTP. Some Way, there are application layer protocols specified in context to IOT as well.

This layer is responsible for data formatting and presentation. The application layer in the internet is typically based on HTTP Protocol. However, HTTP is not suitable in resource constrained environment because it is extremely heavyweight and thus incurs a large parsing overhead. [40]

#### 1.6.2 IoT Protocol:

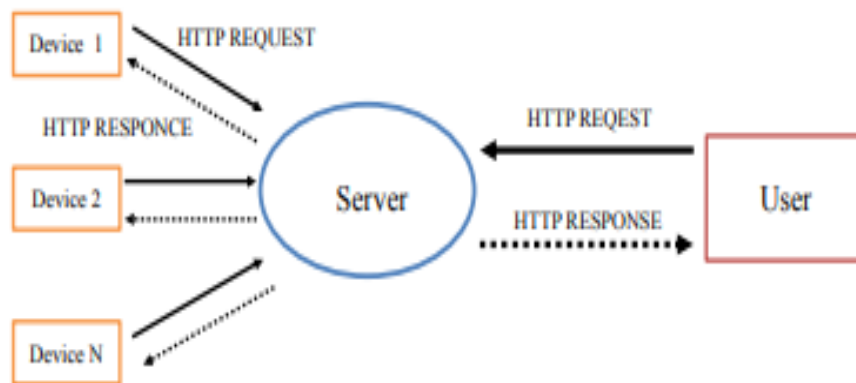
- **HTTP:**

HTTP is predominantly a web messaging protocol, which was originally developed by Tim Berners-Lee. And first published as a standard protocol in 1997 [13]. HTTP supports request/response RESTful Web architecture. Analogous to CoAP, HTTP uses URI instead of topics. Server sends data through the URI and client receives data through particular URI. HTTP is a text-based protocol and it does not define the size of header and message payloads rather it depend on the web server or the programming technology. [41]

Regarding communication for IoT, functions of IP/TCP/UDP are minimized. Then, lightweight protocols for IoT are required over TCP/UDP. Meanwhile, for conventional communication, legacy protocols are invoked over TCP/UDP. HTTP is one of the promising protocols for Internet access. It is assumed that HTTP is applied to communication for IoT.

HTTP must transfer a large number of tiny packets. Protocol overhead of HTTP causes serious problems, such as consumption of network resources and large delays.

Communication using HTTP is configured as shown in **Figure 1.5**. [42]

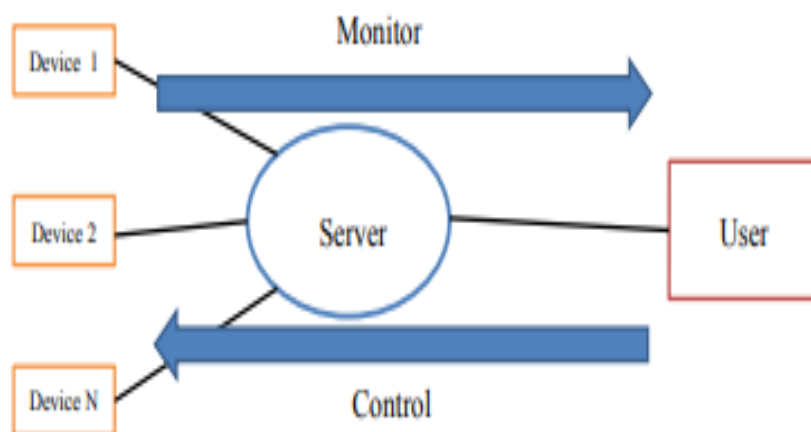


**Figure 1.5:** System configuration using HTTP [42]

- **MQTT:**

MQTT is one of the most common application layer protocols used for sharing information in IoT applications. MQTT protocol does not have any inbuilt mechanism to secure the information that is communicated between the two devices. MQTT protocol depends on lower layers for the security of information that is communicated. [43]

is an IoT-oriented protocol with publish/subscribe semantics. The protocol is designed for low overhead and is agnostic to the data payload. MQTT provides three levels of quality of service: at most once provides best-effort service, at least once assures delivery but may incur duplicates, and exactly once ensures the message is delivered without duplication. [44]



**Figure 1.6:** System configuration using MQTT. [42]

- **CoAP:**

Constrained application protocol (CoAP) was designed by using a subset of HTTP methods. It uses UDP instead of TCP to keep things lightweight. This protocol has been specially made to be used on constrained devices having low processing power and little RAM. It follows request-response model for interaction between client and server. CoAP fulfills most of the requirements required for M2M communication such as simplicity, small header packet, and multicast. It also provides asynchronous message transfers. [45]

- **AMQP:**

AMQP is an open protocols for asynchronous message queuing which has been developed and matured over several years. AMQP is an open standard, binary application layer protocol designed for message-oriented middleware i.e., AMQP standardises messaging using Producers, Brokers and Consumers and messaging increases loose coupling and scalability. [46]

- **REST:**

REST service is an architecture which uses HTTP as application layer protocol. TCP is used by HTTP as transport layer protocol. It provides various web services for data exchange and communication. It is a client server approach where client can access and use these resources provided by the server. REST uses methods defined by HTTP such as GET, PUT, POST, DELETE etc. GET and POST are used to retrieve and send data to server. PUT is used to update and do changes to resource data. REST uses simple request/ response model for messaging. The advantage of using REST is it is widely used architecture style and it is supported by all commercial cloud platform which support M2M. It can be easily implemented in mobile or any embedded device in any language as it uses HTTP library which is available for all Operating System Distributions and almost all programming languages. [45]

- **XMPP:**

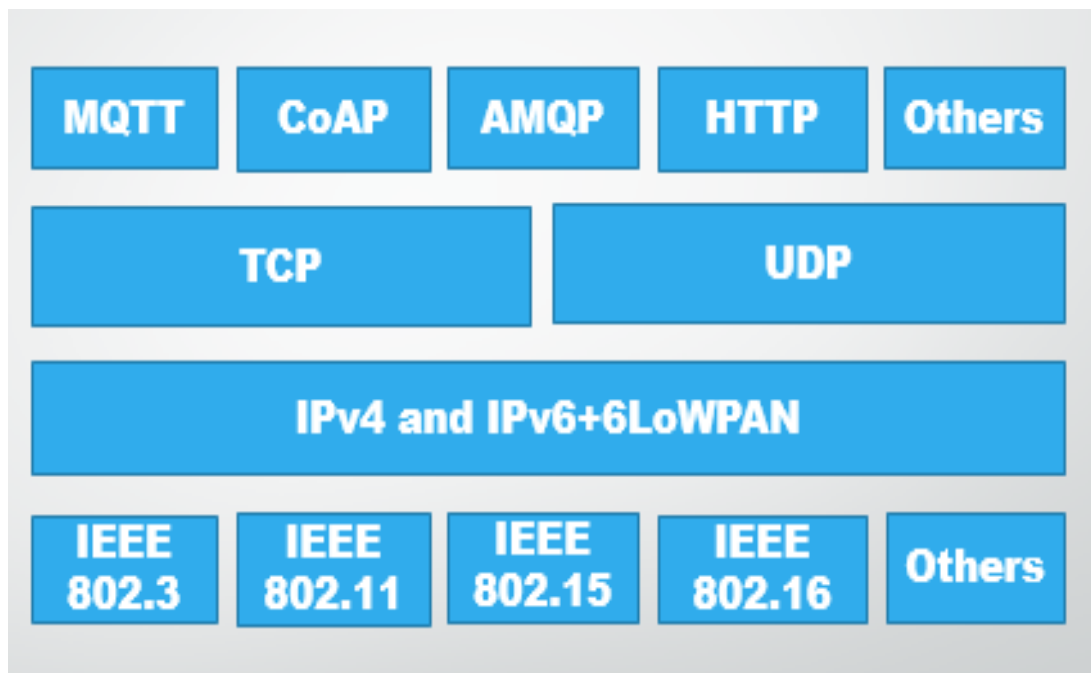
XMPP is an open set of rules for streaming XML elements in order to swap messages and presence information in close to real-time. The XMPP protocol is based on the typical client server architecture, in which the XMPP client uses the XMPP server with the TCP socket. XMPP provides a general framework for messaging across a network, offering a multitude of

applications beyond traditional instant messaging (IM) and the distribution of presence data. It enables the discovery of services residing locally or across a network, as well as finding out about the availability of these services. [47]

XMPP or the Extensible Messaging and Presence Protocol, formerly known as Jabber, is a communications protocol based on XML aimed at message-oriented middleware and applications such as near-real-time instant messaging and presence information. XMPP is designed to be extensible, and has been used for publish-subscribe systems, file transfer, and communication in embedded Internet-of-Things networks. [48]

- **6LoWPAN:**

Since the beginning of the IETF research on IoT related technologies, IPv6 has been selected as the only choice to enable wireless communication. Its key features, such as universality, extensibility, and stability, have attracted a lot of attention and may become the de facto solution for future Internet technology. In order to enable IP connectivity in resource constrained sensor networks, the IPv6 over Low Power WPAN (6LoWPAN) Working Group has been established and works on protocol optimization of IPv6 over networks using IEEE 802.15.4. [49]



**Figure 1.7:** Protocol Stack for IoT Systems. [41]

**1.7 Conclusion:**

In this chapter I mentioned the basic elements of the internet of things, such as its concept and most important technologies that it uses through its applications, especially smart home with its components, in addition its basic layers and some protocols.

Without a doubt, we knew through this that technology has made a great challenge in the world of technology in various fields.

# **Chapter 2:**

# **Raspberry Pi Board**



## 2.1 Introduction:

We will use Raspberry Pi board in our project which is a small computer and run by the Linux operating system.

In this chapter we will give the definition of this board, mention its components and operating systems and remote access to see the desktop or the command interface. After that we will go through the characteristics and types of Raspberry Pi and present Raspberry Pi applications, advantage and constraints.

## 2.2 Definition of Raspberry Pi:

The Raspberry Pi is a low cost, computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It's capable of doing everything you'd expect a desktop computer to do. And this microcomputer operated by systems Linux open source. [1] Raspberry Pi is using the Advanced Reduced Instruction Set Computing Machine (ARM) technology. ARM technology is used on the board which reduces cost, heat and power consumption. It is energy effective multi core CPU implemented as SoC weighing 50gm and operates on 5V, 700mA power rating. This board is available in three models named A, B, B+. [2]

## 2.3 Components of Raspberry pi:

This is Raspberry pi 3 model B shown in **Figure 2.1**



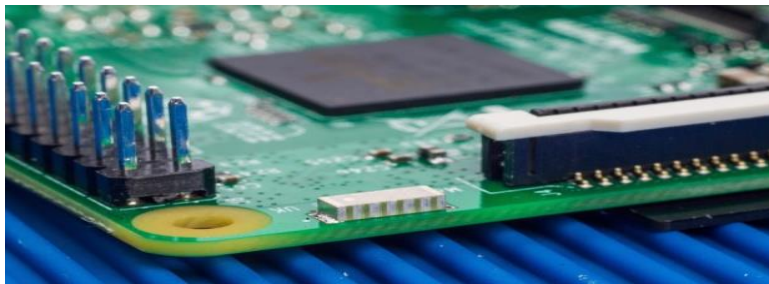
**Figure 2.1** Raspberry Pi 3 model B.

### 2.3.1 The Raspberry Pi's system-on-chip (SoC):

The name system-on-chip is a great indicator of what you would find if you prised the metal cover off: a silicon chip, known as an integrated circuit, which contains the bulk of the Raspberry Pi's system. This includes the CPU, commonly thought of as the 'brain' of a computer, and the GPU, which handles the visual side of things, and Pi's RAM. [3]

### 2.3.2 Wi-Fi and Bluetooth antenna:

This is the antenna of two technologies Wi-Fi and Bluetooth as shown in **figure 2.2**.



**Figure 2.2:** The Raspberry Pi's Wi-Fi and Bluetooth antenna. [4]

### 2.3.3 The Raspberry Pi's ports:

- **The Raspberry Pi's USB ports:**

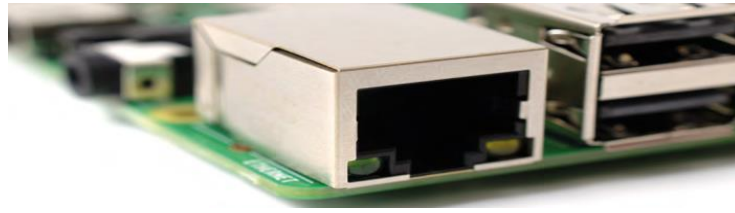
It used to connect any USB device such as flash disk, wireless network cards, keyboard or mouse.



**Figure 2.3:** The Raspberry Pi's USB ports. [3]

- **The Raspberry Pi's Ethernet port:**

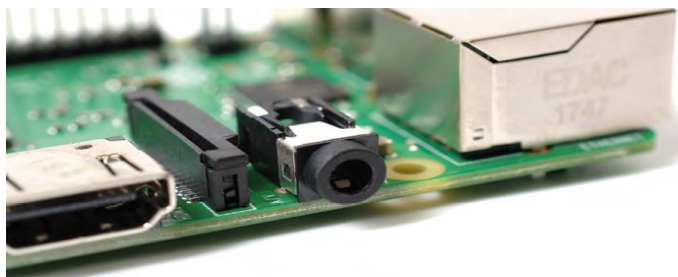
The port used to connect Raspberry pi to internet network.



**Figure 2.4:** The Raspberry Pi's Ethernet port. [3]

- **The Raspberry Pi's 3.5 mm AV jack:**

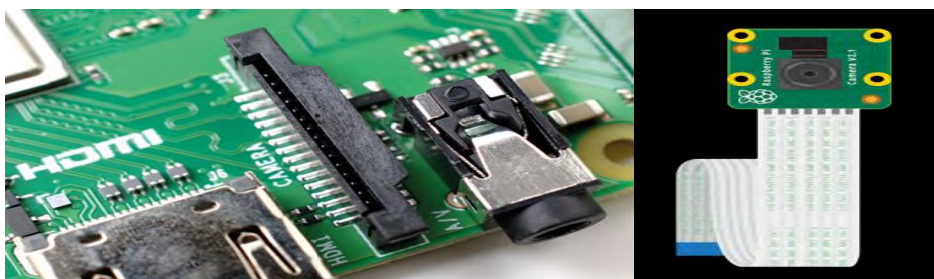
AV jack this is known as the headphone jack, and it can be used for that exact purpose though you'll get better sound connecting it to amplified speakers rather than headphones. It has a hidden, extra feature, though: as well as audio, the 3.5 mm AV jack carries a video signal which can be connected to TVs, projectors, and other displays that support a composite video signal using a special cable known as a TRRS adapter. [3]



**Figure 2.5:** The Raspberry Pi's 3.5 mm AV jack. [3]

- **The Raspberry Pi's camera connector:**

The Port to connect the camera high precision and custom made for Raspberry pi.



**Figure 2.6:** The Raspberry Pi's camera connector + RPi camera module. [3]

- **The Raspberry Pi's HDMI port:**

This connector allows you to hook up a high-definition television or other compatible device using an HDMI cable. [5]



**Figure 2.7:** Raspberry Pi's HDMI port. [3]

- **The Raspberry Pi's micro USB power port:**

This is a 5v Micro USB power connector into which you can plug your compatible power supply. [5]



**Figure 2.8:** The Raspberry Pi's micro USB power port. [3]

- **The Raspberry Pi's display connector (DSI):**

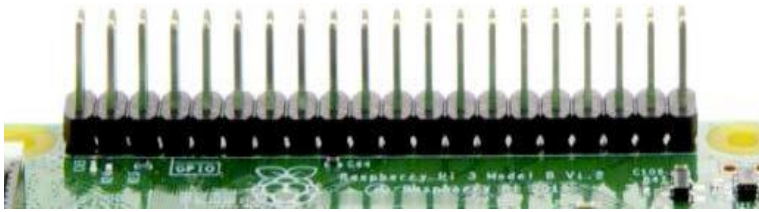
It used to connecting screen touch like used in smartphones.



**Figure 2.9:** The Raspberry Pi's display connector. [3]

- **The Raspberry Pi's GPIO header:**

A group of electronic control ports, which are ports that enable to control various electronic components. It makes Raspberry pi capable of performing microcontroller functions.



**Figure 2.10:** The Raspberry Pi's GPIO.

- **The Raspberry Pi's microSD card connector:**

The place for installation a memory card holds the operating system and all files for the Raspberry pi.



**Figure 2.11:** The Raspberry Pi's microSD card connector. [3]

## 2.4 Accessories of Raspberry Pi:

- **Power supply:**

We need a 5V, 2.5 Amp micro USB power adapter for powering our Raspberry Pi 3 Model B and we can use many of the good Android phone charger to power it. [6]

**Micro SD card:**

The SD card holds all the information to run a Raspberry Pi from the operating system to applications.

- **Hdmi cable:**

We can connect directly to a monitor or TV with an HDMI cable on Raspberry Pi that has an HDMI port.

- **Camera module (optional):**

The Raspberry Pi Camera Module is an official product from the Raspberry Pi Foundation. The original 5-megapixel model was released in 2013, and an 8-megapixel Camera Module v2 was released in 2016. [7]

- **Case (optional):**

Since Raspberry pi is naked, so needs a case to protect it while handling it.

- **Keyboard / Mouse:**

When we have Raspberry Pi powered, booted and connected to a screen we definitely need mouse and keyboard to interact.

- **Raspberry Pi Touch Display (optional):**

Raspberry Pi has another very important add on accessory called Pi Touch Display. It gives raspberry Pi users ability to create all-in-one projects such as making tablets, photo frames, gaming consoles, and other embedded projects, and it Connect via DSI port. [8]

## 2.5 Operating System:

Most systems that run on Raspberry Pi are versions of the system Linux operating system. Because Linux is open source, developers can adopt it for specific purposes. [9]

And the different Linux distributions for the Raspberry Pi are:

### 2.5.1 Arch Linux:

The famous distribution for the Raspberry Pi. It is reserved for experienced users of Linux. It does not offer any graphic office. [10]



**Figure 2.12:** Logo of Arch Linux. [10]

### 2.5.2 Pidora:

Pidora is a Linux software distribution for the Raspberry Pi computer. It contains software packages from the Fedora project. [11]



**Figure 2.13:** Logo of Pidora. [11]

### 2.5.3 Raspbian:

The recommended distribution is Raspbian. It is a free operating system based on Debian optimized for Raspberry hardware Pi. However Raspbian provides more than just an operating system: it comes with more than 35,000 packages, precompiled software that facilitates our development soft.

There is another operating system created by Microsoft especially for the Windows 10 IoT Core IoT domain. [12]



**Figure 2.14:** Logo of Operating system Raspbian and Windows 10 IoT. [9]

## 2.6 Opening the raspberry pi:

### 2.6.1 Download NOOBS:

The Raspberry Pi comes with many operating systems you can use, which you could manually install yourself if you wish. There's a much easier way to install these OSes, though, and that's via the New Out Of Box Software, or NOOBS. It holds all the latest versions of the Raspberry Pi operating systems and you can grab it from the download page at: [raspberrypi.org/downloads](http://raspberrypi.org/downloads). We prefer to use the full version of NOOBS, as it comes with Raspbian already downloaded, making the process slightly faster than with NOOBS Lite. However, all the other operating systems will be downloaded as they install, on both versions.

### 2.6.2 Install SD Card:

While that's downloading, you'll need to get your SD card ready to work on your Raspberry Pi. This will require you to format it, so if there are any files on the card you want to keep, now's the time to take them off. You'll need to install the SD Card Formatter 4.0 tool to prepare the card, once you've formatted your SD card, extract the files from the NOOBS zip folder and put them all on the card. That's it: NOOBS is installed to your SD card and ready to use. [12]

### 2.6.3 Connect the Cable:

Take the SD card adaptor out, retrieve the micro SD card, and slot it into the Raspberry Pi; this is very important, as the Raspberry Pi won't be able to turn on properly otherwise. To start with, you'll need to plug in an HDMI cable between the Raspberry Pi and your screen, an Ethernet cable for your router (or a USB Wi-Fi dongle), along with a mouse and keyboard. [12]

### 2.6.4 Install Raspbian:

The Raspberry Pi will turn on and display some text on the screen you can ignore this until it gets to a menu which lists all the available operating systems. It allows you to select multiple OSes at once, but right now we just want to use the one that's called Raspbian. This is the main operating system for the Raspberry Pi, with all the official apps, software, and learning documents. Upon selecting Raspbian, click on Install and it will begin the Raspbian installation process, which may take a little while to complete. [12]



## 2.7 Remote Access:

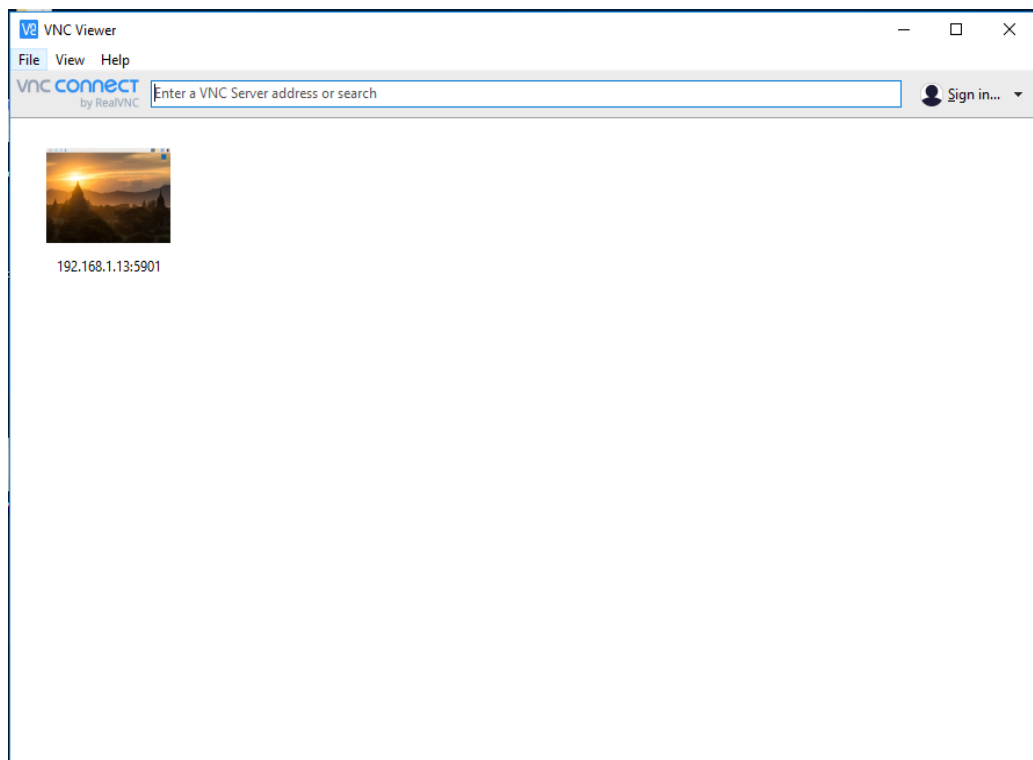
Sometimes we need to access a Raspberry Pi without connecting it to a monitor so there is solution to access remotely on the same network:

### 2.7.1 VNC:

VNC is a graphical desktop sharing system that allows you to remotely control the desktop interface of one computer (running VNC Server) from another computer or mobile device (running VNC Viewer). VNC Viewer transmits the keyboard and either mouse or touch events to VNC Server, and receives updates to the screen in return.

We will see the desktop of the Raspberry Pi inside a window on your computer or mobile device. You'll be able to control it as though you were working on the Raspberry Pi itself.

[13]



**Figure 2.15:** VNC interface.

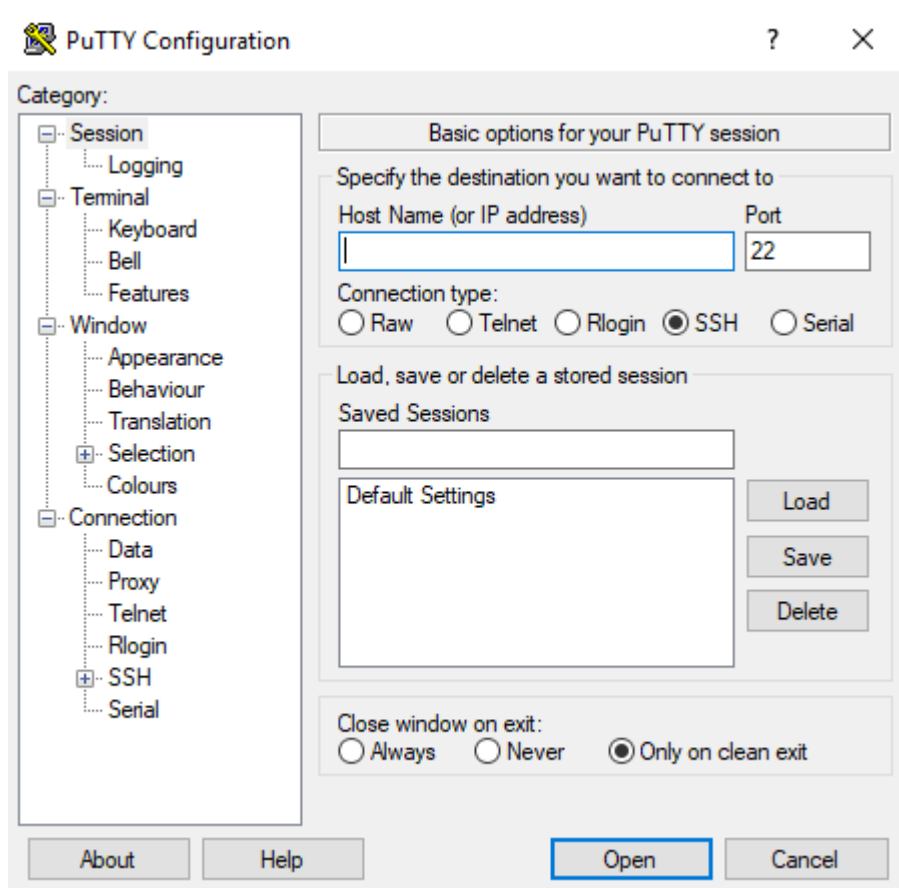
### 2.7.2 SSH:

SSH is an encrypted networking technology that enables you to manage computers from the command line over a network. [14]

We can access the command line of a Raspberry Pi remotely from another computer or device on the same network by using SSH. [13]

We don't need to install any software to start using Secure Shell. Linux, and Windows 10 have the SSH command-line application installed by default. On older Windows you will need download an SSH client; the most commonly used one is called PuTTY. [14]

And the PuTTY interface as shown on **Figure 2.16**:



**Figure 2.16:** PuTTY Interface.

## 2.8 Characteristics of Raspberry Pi Types:

We have different types of Raspberry pi like Pi Zero, Pi 1 Model A+, Pi 1 Model B+, Pi 2 Model B and Pi 3 Model B this is what we used in our project so the **Figure 2.1** show us the characteristics of every type and the difference between them.

Pi Zero	Pi 1 Model A+	Pi 1 Model B+	Pi 2 Model B	Pi 3 Model B
1 GHz 32-bit Single Core processor	700 MHz 32-bit Single Core processor	700 MHz 32-bit Single Core processor	900 MHz 32-bit Quad Core processor	1.2 GHz 64-bit Quad Core Processor
512 MB RAM	512 MB RAM	512 MB RAM	1 GB RAM	1 GB RAM
Broadcom VideoCore IV GPU	Broadcom VideoCore IV GPU	Broadcom VideoCore IV GPU	Broadcom VideoCore IV GPU	Broadcom VideoCore IV GPU
2 micro USB ports	1 micro USB port	1 micro USB port	1 micro USB port	1 micro USB port
No USB ports	1 USB port	4 USB ports	4 USB ports	4 USB ports
1 mini HDMI port, no HDMI	1 HDMI port, no mini HDMI	1 HDMI port, no mini HDMI	1 HDMI port, no mini HDMI	1 HDMI port, no mini HDM
1 microSD slot	1 SD/MMC slot	1 microSD slot	1 microSD slot	1 microSD slot
No dedicated audio port	3.5mm audio out port	3.5mm audio out port	3.5mm audio out port	3.5mm audio out port
No Wi-Fi, No Ethernet	No Wi-Fi, No Ethernet	No Wi-Fi, Ethernet via USB Adapter	No Wi-Fi, Ethernet via USB Adapter	Onboard Wi-Fi, Ethernet port
No Bluetooth	No Bluetooth	No Bluetooth	No Bluetooth	Bluetooth 4.1
65x30 mm (Half of Standard Pi Size)	85.60×56.5 mm (Standard Pi Size)	85.60×56.5 mm (Standard Pi Size)	85.60×56.5 mm (Standard Pi Size)	85.60×56.5 mm (Standard Pi Size)
<b>\$5</b>	<b>\$20</b>	<b>\$25</b>	<b>\$35</b>	<b>\$35</b>

**Table 2.1:** Characteristics of different Raspberry Pi's. [15]

## 2.9 The uses of Raspberry Pi:

There are many uses for Raspberry pi board we mention some of it:

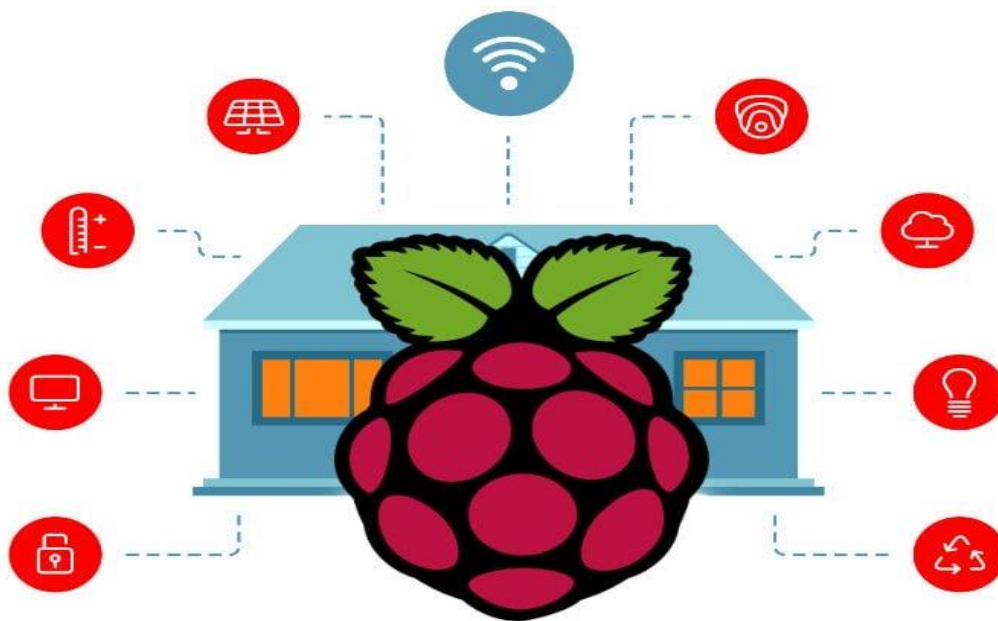
### 2.9.1 Low-Cost Desktop PC:

The Raspberry Pi can make a useful desktop computer if set up properly. To turn your raspberry pi into a desktop PC you'll need other gadgets than RPi itself such as a screen, a mouse, a keyboard and if you want, an extra storage device.

You also need to install an operating system like Raspbian or any other operating systems available for RPi. Some apps such as email and web browsing apps are included and many other are available for RPi. [16]

### 2.9.2 Home Automation System:

The Raspberry Pi is capable of hosting a powerful home automation application. You can attach sensors, a camera, relays, etc. And you can monitor and control your house remotely. [16]



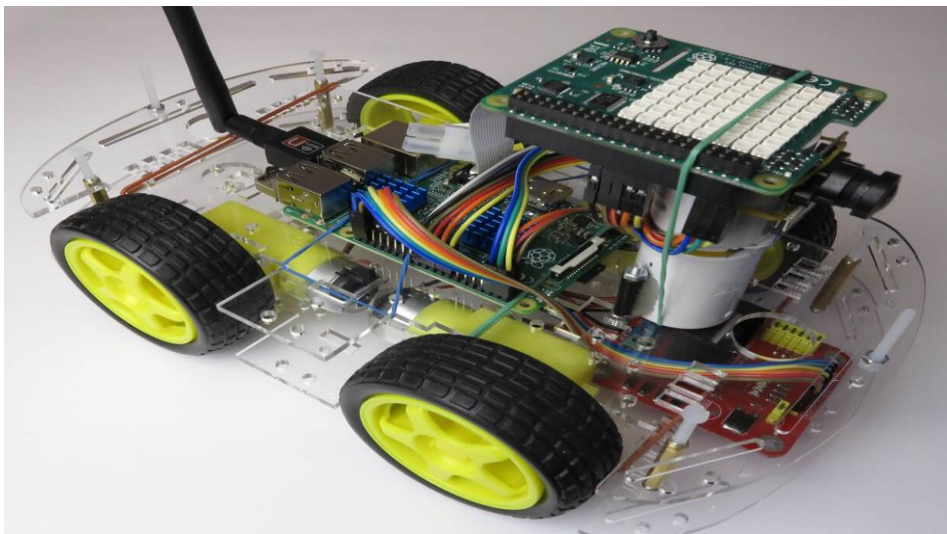
**Figure 2.17:** Home automation with Raspberry Pi. [17]

### 2.9.3 Web server:

Another fantastic use for a Raspberry Pi is to set it up as a web server. This basically means that it can be configured to host a website. Several methods can be used like Apache server. [18]

### 2.9.4 Robot:

There are so many robot-controller Raspberry Pi projects that it's difficult to settle on a single example. You might, for instance, rely on a dedicated robotics package for your Pi, with the device battery powered and used to communicate and control your robot. [18]



**Figure 2.18:** Robot car with Raspberry Pi. [19]

### 2.9.5 VPN:

Free, unencrypted wireless is everywhere, but you shouldn't be checking your accounts on it unless you don't mind somebody else snooping. The solution A VPN. A VPN extends your own private network into public places, so even if you're using Starbucks' Wi-Fi connection, your Internet browsing stays encrypted and secure. [16]

## 2.10 Raspberry Pi's Performances and constraints:

Raspberry Pi advantages can be summarized as follows:

- You can install a fully fledged Operating System (eg Raspbian which is Linux-based or Windows IOT Core which is Windows based OS) and use it as a day to day computer

- The presence of GPIO is what distinguishes a RPi from traditional computers. You can connect these pins to sensors and external components and interact with them programmatically using a language such as Python. This allows you to build and prototype Internet of Things devices that can sense the real world
- Newer models like the RPi has Wi-Fi and Bluetooth built in. This allows you to take projects into wireless mode easily. [20]
- C, Python or object oriented languages such as C++ and Java can be used for programming of Raspberry Pi.
- It can be powered form battery or sollar cell.
- It can be run in server mode.
- A various web server can be installed and running on Raspberry Pi. [21]

Also lacks some useful features. The main disadvantages of Raspberry Pi are:

- There is not any fuse protection on the RPi, so if you connect pins incorrectly, you can damage the board.
- It is not as fast in terms of CPU processing speed nor does it have as much memory as traditional PC or laptops. But for \$35, it packs quite a punch
- There is no built in analog to digital conversion on the GPIO pins like there is in Arduino. So you need to have an ADC chip to work with analog signals when you need more accuracy. [20]
- It does not have a RTC with a backup battery but it can easily work around the missing clock using a network time server, and most operating systems do this automatically.
- It does not have a Basic Input Output System (BIOS) so it always boots from an SD card. [21]

## 2.11 Conclusion:

The performances of Raspberry pi is compared to the various hardware platforms on a basis of size, cost, communication capabilities and programming languages. On the basis of above end user can easily program Raspberry pi according to his need. Raspberry pi has wide domain of applicability.

# **Chapter 3:**

## **Web Programming Language**

### 3.1 Introduction:

There are many of Programming Languages in the world, in this project we have used HTML, CSS, and PHP for the web page.

In this chapter we discuss the basics, syntax and properties of the above listed languages.

### 3.2 HTML:

#### 3.2.1 Definition of HTML:

HTML is computer language specifically designed for creating websites. Such websites can be visited by anyone regardless of their location or time zone. As long as you have an internet connection, you can visit websites. [1]

HTML is not a programming language; it is a markup language, which means it is a system for identifying and describing the various components of a document such as headings, paragraphs, and lists. [2]

#### 3.2.2 Tags and Elements:

Each element reference in a document is contained within one or two tags tokens enclosed by angle brackets (< and >) containing the name of the element being used. Opening tags always begin with < immediately followed by the element name, and reference all attributes and values associated with the element; closing tags are featureless apart from the element name, which is preceded by a forward slash. [3]

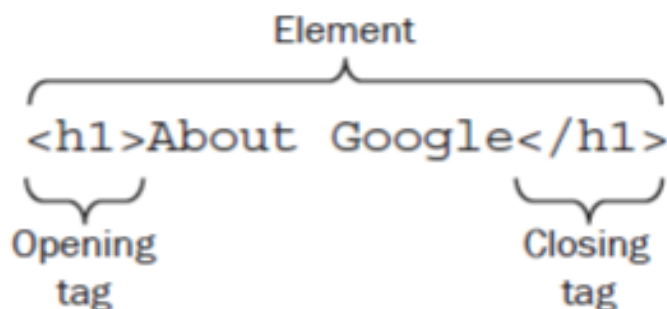


Figure 3.1: HTML Element and tag.



### 3.2.3 Basic HTML:

In any web page there are some basic items must-have:

#### 3.2.3.1 HTML Document:

- All HTML documents must start with a document type declaration: `<!DOCTYPE html>`.
- The HTML document itself begins with `<html>` and ends with `</html>`.
- The visible part of the HTML document is between `<body>` and `</body>`. [4]

#### 3.2.3.2 HTML Headings:

- HTML headings are defined with the `<h1>` to `<h6>` tags.
- `<h1>` defines the most important heading. `<h6>` defines the least important heading. [4]

#### 3.2.3.3 HTML Paragraphs:

- HTML paragraphs are defined with the `<p>` tag.
- The HTML paragraphs itself begins with `<p>` and ends with `</p>`, Like this `<p>` This is the paragraphs `</p>`. [4]

#### 3.2.3.4 HTML Body:

- The body element of the page contains almost everything like: headings, paragraphs, images, any navigation that's required, and footers that sit at the bottom of the web page. The HTML Body itself begins with `<body>` and ends with `</body>`. [5]

### 3.2.4 HTML Attributes:

There are some attributes, such as `id`, `class`, `title`, `style` etc. that you can use on the majority of HTML elements. The following section describes their usages.

- **The id Attribute:**

Is used to give a unique name or identifier to an element within a document. This makes it easier to select the element using CSS or JavaScript. [6]

- **The class Attribute:**

Like id attribute, the class attribute is also used to identify elements. But unlike id, the class attribute does not have to be unique in the document. This means you can apply the same class to multiple elements in a document. [6]

- **The style Attribute:**

The style attribute is a new HTML attribute. It introduces the use of CSS in HTML. The purpose of the style attribute is to provide a common way to style all HTML elements. [7]

- **The Title Attribute:**

The title attribute is used to specify extra information about the element. When the mouse moves over the element then it shows the information. [8]

### 3.2.5 HTML Links:

- The tags used to produce links are the `<a>` and `</a>`.
- The `<a>` tells where the link should start and the `</a>` indicates where the link ends.
- Everything between these two will work as a link.
- The target of the link is added to the `<a>` tag using the `href="http://www.whateverpage.com"` setting.
- The example below shows how to make the word here work as a link to yahoo. Click `<a href="http://www.yahoo.com">here</a>` to go to yahoo. [9]

### 3.2.6 HTML Images:

- Images enhance visual appearance of the web pages by making them more interesting and colorful.
- The `<img>` tag is used to insert images in the HTML documents. It is an empty element and contains attributes only. The syntax of the `<img>` tag can be given with:

```
 [6]
```

### 3.2.7 HTML Lists:

In general, there are three ways on how you can specify your list of information. Keep in mind that every list has to contain least one element. Must be given the tag `<li>` That defines list item in every way.

A list can contain the Following:

- `<ol>` You use it for ordered lists because it makes use of various schemes in order to list the items.
- `<ul>` You use it for unordered lists because it lists the items with the use of bullets.
- `<dl>` You use it for definition lists because it organizes the items as if they were in a dictionary. [1]

### 3.2.8 HTML Table:

HTML table allows you to arrange data into rows and columns. They are commonly used to display tabular data like product listings, customer's details, financial reports, and so on.

You can create a table using the `<table>` element. Inside the `<table>` element, you can use the `<tr>` elements to create rows, and to create columns inside a row you can use the `<td>` elements. You can also define a cell as a header for a group of table cells using the `<th>` element. [6]

This table summarizes each tag with a description:

TAG	DESCRIPTION
<code>&lt;table&gt;</code>	Defines a table
<code>&lt;th&gt;</code>	Defines a table header
<code>&lt;tr&gt;</code>	Defines a table row
<code>&lt;td&gt;</code>	Defines a table cell

**Table 3.1:** List tags with Description. [7]

## 3.3 CSS:

### 3.3.1 Definition of CSS:

Cascading style sheets, more commonly referred to a CSS, is a simple design language used to make developing websites easier. It is basically used to define the styles of web pages,

including their layout, design, variations in display for various devices, and sizes of screens.

[1]

### 3.3.2 Ways to Insert CSS:

There are three ways to insert CSS:

- **External CSS:**

An external style sheet allows you to change the look of the entire site by changing a single file. This is ideal when a particular style you want to modify is applied to multiple or all of the pages, [10]

When using an external stylesheet you must reference the stylesheet in the HTML page that is using it. an External stylesheet looks like:

```
<link rel="stylesheet" type="text/css" href="style.css" /> [11]
```

- **Internal CSS:**

An internal stylesheet holds the CSS code for the webpage in the head section of the particular file. This makes it easy to apply styles like classes or id's in order to reuse the code. The downside of using an internal stylesheet is that changes to the internal stylesheet only effect the page the code is inserted into. an Internal stylesheet looks like:

```
<head> <style type="text/css"> h1 {color:blue;} h2 {color:red;} p {color:green;} </style>  
</head> [12]
```

- **Inline CSS:**

Inline CSS coding is not recommended as it mixes the content and presentation thus losing the many advantages of style sheets, to use inline style, use the style attribute inside each relevant tag. By using the style attribute you will be able to define any CSS property that you want. [10]

an Inline stylesheet looks like:

```
<tag style="property: value"> </tag> [11]
```

### 3.3.3 CSS Syntax:

CSS syntax is easy. A stylesheet contains styles; a style contains **selectors** and **rules**; and a rule contains a **property** and a **value**. The following is the design pattern for a style:

```
SELECTORS { RULES }
```

The following is the design pattern for a rule:

```
PROPERTY:VALUE; [13]
```

### 3.3.4 CSS selector:

CSS selectors are patterns used to select the element(s) you want to style. Within CSS, selectors are followed with curly brackets, {}, which encompass the styles to be applied to the selected element. There are a types of CSS selectors:

- **Matching Elements by Name:**

The easiest selector to understand is the plain element selector, as in the following example:

```
h1 { color: red; }
```

Using the actual element name (h1) as the selector causes all those tags to be formatted with the attributes of the definition (color: red). You can also attach multiple selectors to the same definition by listing them all in the selector area, separated by commas. For example, this definition will affect all heading tags in the document: h1, h2, h3, 4h, h5, h6 { color: red; }. [14]

- **Matching Using the Universal Selector:**

The universal selector can be used to match any element in the document. The universal selector is an asterisk (\*). As an extreme example, you can use the universal selector to match every tag in a document: \* { color: red; } [14]

- **Matching Elements by Class:**

Class selectors allow us to select an element based on the element's class attribute value. Class selectors are a little more specific than type selectors, as they select a particular group of elements rather than all elements of one type.

Class selectors allow us to apply the same styles to different elements at once by using the same class attribute value across multiple elements. Within CSS, classes are denoted by a leading period, . , followed by the class attribute value. [15]

- **Matching Elements by Identifier:**

ID selectors are even more precise than class selectors, as they target only one unique element at a time. Just as class selectors use an element's class attribute value as the selector, ID selectors use an element's id attribute value as a selector. Regardless of which type of element they appear on, id attribute values can only be used once per page. If used they should be reserved for significant elements. Within CSS, ID selectors are denoted by a leading hash sign, #, followed by the id attribute value. [15]

### 3.3.5 CSS Basic Properties:

In this Table some CSS basic Properties will be illustrated with description and values:

Property	Description	Values
Color	Sets the color of a text	RGB, hex, keyword
text-align	Aligns the text in an element	left, right, center, justify
list-style	Sets all the properties for a list in one declaration	list-style-type, list-style-position, list-style-image, inherit
list-style-image	Specifies an image as the list-item marker	URL, none, inherit
list-style-position	Specifies where to place the list-item marker	inside, outside, inherit

Border	Sets all the border properties in one declaration	border-width, border-style, border-color
border-bottom	Sets all the bottom border properties in one declaration	border-bottom-width, border-bottom-style, border-bottom-color
border-color	Sets the color of the four borders	color_name, hex_number, rgb_number, transparent, inherit
font-style	Specifies the font style for text	normal, italic, oblique, inherit
Background	Background properties are used to define the background effects for elements.	background-color background-image background-attachment background-position
Padding	Padding properties are used to generate space around an element's content, inside of any defined borders.	padding-top padding-right padding-left

**Table 3.2:** CSS Basic Properties. [16]

### 3.3.6 CSS Units:

While CSS supports a seemingly endless list of properties, the scheme for setting values is fairly predictable. **Table 3.3** Describes the most frequently used values:

Unit	Type	Example
px (pixels)	Length	Width: 744px;
em (ems)	Length	Margin-left: 1.25em;
% (percent)	Length	Left: 34%;
pt (points)	Length	Font-size: 12pt;
in (inches)	Length	Margin-top: .75in;
cm (centimeters)	Length	Margin-top: 1.905cm;
xx-small ... xx-large	Font-size	Font-size: large;
Rgb ( r,g,b)	Color (decimal)	Background-color: rgb(221, 204,187)
#rrggbb	Color(hexadecimal)	Background-color: #ddccbb
#rgb	Color(hexadecimal, reduced depth)	Background-color: #dcb

**Table 3.3:** Commonly encountered CSS length/size, keyword, and color units. [3]

## 3.4 PHP:

### 3.4.1 Definition of PHP:

PHP is a programming language designed to generate web pages interactively on the computer serving them, which is called a web server. Unlike HTML, where the web browser uses tags and markup to generate a page, PHP code runs between the requested page and the web server, adding to and changing the basic HTML output. [17]

PHP is a powerful scripting language that can be run by itself in the command line of any computer with PHP installed. However, PHP alone isn't sufficient for building dynamic web sites. To use PHP on a web site, you need a server that can process PHP scripts. Apache is a free web server that, once installed on a computer, allows developers to test PHP scripts



locally; this makes it an invaluable piece of your local development environment.

Additionally, web sites developed with PHP often rely on information stored in a database, so it can be modified quickly and easily. [18]



**Figure 3.2:** Logo of PHP.

### 3.4.2 PHP Syntax:

PHP's syntax and semantics are similar to most other programming languages with the addition that all PHP code is contained with a tag, starts with `<?php` and ends with `?>` as the following:

```
<?php
```

```
// PHP code goes here
```

```
?>
```

The default file extension for PHP files is ".php". [19] [20]

### 3.4.3 PHP comment:

A comment in PHP code is a line that is not executed as a part of the program. Its only purpose is to be read by someone who is looking at the code.

Comments can be used to:

- Let others understand your code

Remind yourself of what you did - Most programmers have experienced coming back to their own work a year or two later and having to re-figure out what they did. Comments can remind you of what you were thinking when you wrote the code. [19]

There are two types of comments used in PHP:

- **Single line comments:**

Single line comment used for short explanations. Declaration of Single line comment are two types: Either Begin with (#) or backslash (//).

- **Multi-lines comments:**

Multi lines comments used to comment multiple lines. Here we can give comments in bulk the bulk comments are enclose within (/\*.....\*/). [21]

### 3.4.4 PHP Variables:

Variables are used to store values like strings numbers etc. When a variable is declared, it can be used again and again in your script. Symbol used to represent PHP variables is “\$”.

Syntax:

```
$var _ name = value; [22]
```

### 3.4.5 PHP Types of data:

There is a set of PHP types data that we list it:

#### **PHP Integers:**

An integer is a number which is not a fraction; a whole number (... , -2, -1, 0, 1, 2, 3 ...), an integer can either be a negative or a positive number.

Four ways to specify an integer in following notations:

- Specifying Integers in Decimal - base 10.
- Specifying Integers in Hexadecimal - base 16.
- Specifying Integers in Octal - base 8.

- Specifying Integers in Binary - base 2. [23]

**PHP Strings:**

A string is any finite sequence of characters (letters, numerals, symbols, punctuation marks, etc.). [23]

**PHP Booleans:**

A Boolean is a variable that can have one of two possible values, true or false. [23]

**PHP Arrays:**

An array stores multiple values in one single variable. [19]

**PHP Objects:**

An object is a data type that not only allows storing data but also information on, how to process that data. An object is a specific instance of a class which serve as templates for objects. Objects are created based on this template via the new keyword.

Every object has properties and methods corresponding to those of its parent class. Every object instance is completely independent, with its own properties and methods, and can thus be manipulated independently of other objects of the same class. [24]

**3.5 Conclusion:**

After presenting all of programming languages used in our project (HTML, CSS, PHP). We found that HTML is a markup language, CSS is a design language and both of them are a good tool for static web sites whereas PHP is used to build dynamic web sites.

# **Chapter 4:**

# **Implementation Part**

## 4.1 Introduction:

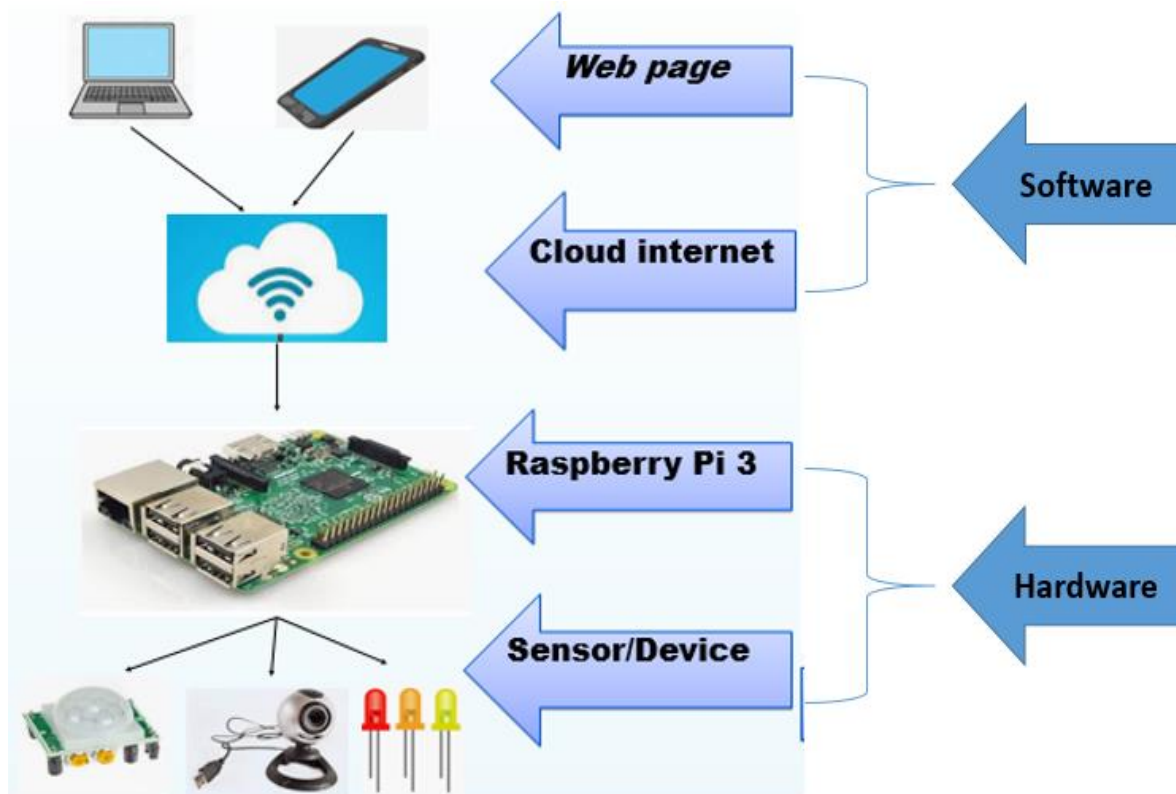
The world of home automation and the smart home is changing and converging. As the Internet of Things (IoT) is increasingly being used for smart home use.

Hence we have made our project based on Raspberry Pi 3 model B as IoT device. Furthermore the programs, we need to control this board through a server in a safe way, are studied and employed.

In this chapter, we laid down the general outline of our modest project, which is divided into two main aspects, software and hardware and explaining them through this project, step by step.

## 4.2 Global schematic of the project:

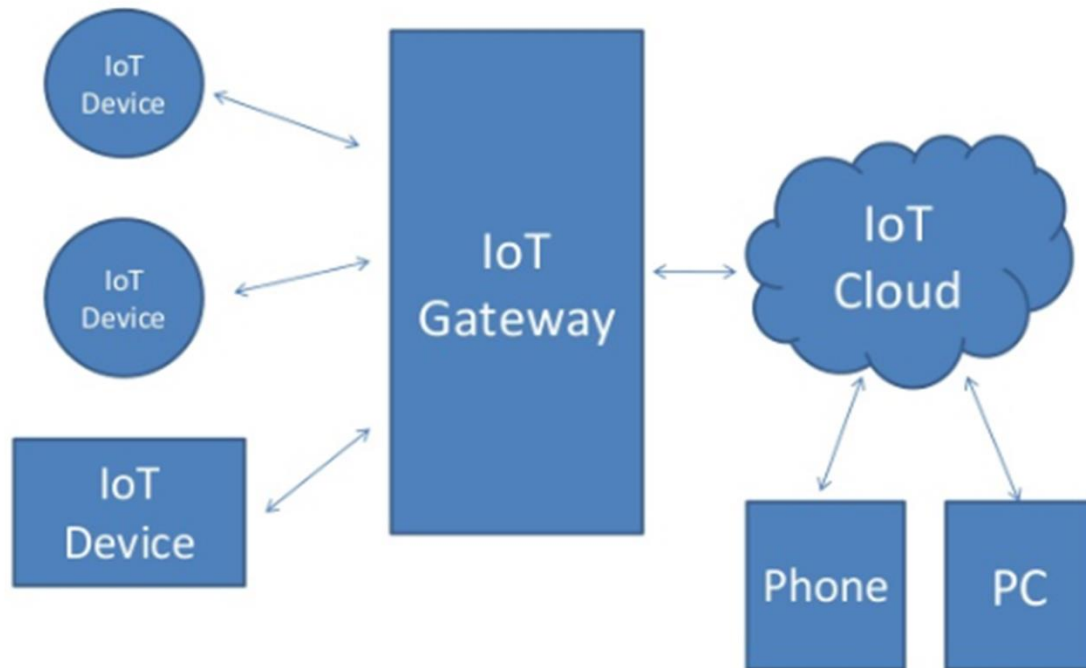
This is the global schematic of our project and it's based on IoT architecture:



**Figure 4.1:** Global schematic of our project.

The IoT Architecture basics are summarized in the following diagram. It consists of: IoT Gateway, IoT device, IoT cloud, PC/Phone.

As the main part is the Raspberry Pi board (IoT Gateway), it is the connection point between the Web page on our device (PC/Phone) and the IoT devices (Sensors and actuators) to be controlled. This control via cloud internet.



**Figure 4.2:** Architecture of IoT.

### 4.3 Hardware section:

#### 4.3.1 Raspberry Pi 3 Model B:

Raspberry Pi 3 Model B is the third generation Raspberry Pi. 10x faster than the first generation, additionally it adds wireless LAN & Bluetooth connectivity making it the ideal solution for powerful connected designs.

And we choose this device on our project for its distinctive features and RPi uses many programming languages unlike Arduino, which is only limited to the C language.

#### 4.3.2 Sensor:

A sensor is a device that detects and responds to some type of input from the physical environment. The specific input could be light, heat, motion, moisture, pressure, or any one of a great number of other environmental phenomena. The output is generally a signal that is converted to human-readable display at the sensor location or transmitted electronically over a network for reading or further processing. [1]

Listed below are some sensors we have used in our project:

- **LED:**

A light-emitting diode (LED) is a semiconductor light source that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons. [2]



**Figure 4.3:** LED.

- **Webcam:**

A webcam is a video camera that feeds or streams an image or video in real time to or through a computer to a computer network, such as the Internet. Webcams are typically small cameras that sit on a desk, attach to a user's monitor, or are built into the hardware. Webcam software enables users to record a video or stream the video on the Internet. As video streaming over the Internet requires much bandwidth, such streams usually use compressed formats. The maximum resolution of a webcam is also lower than most handheld video cameras, as higher resolutions would be reduced during transmission. The lower resolution enables webcams to be relatively inexpensive compared to most video cameras, but the effect is adequate for video chat sessions. [3]

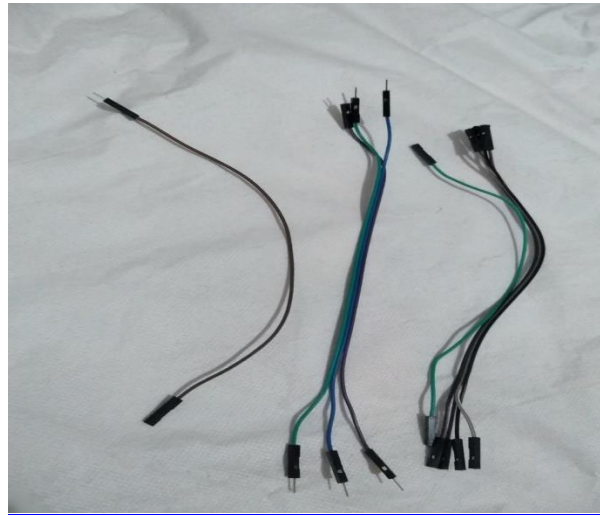


**Figure 4.4:** Webcam.

### 4.3.3 Jumper wire:

Jumper wires are simply wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering. Jumper wires are typically used with breadboards and other prototyping tools in order to make it easy to change a circuit as needed.

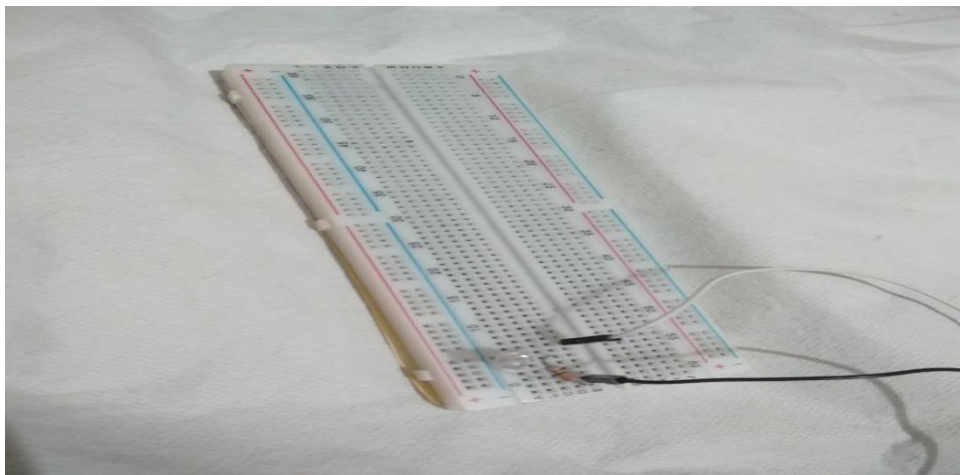
Jumper wires typically come in three versions: male-to-male, male-to-female and female-to-female. The difference between each is in the end point of the wire. Male ends have a pin protruding and can plug into things, while female ends do not and are used to plug things into. [4]



**Figure 4.5:** Jumper wires.

#### 4.3.4 Breadboard:

A breadboard is a solderless device for temporary prototype with electronics and test circuit designs. Most electronic components in electronic circuits can be interconnected by inserting their leads or terminals into the holes and then making connections through wires where appropriate. The breadboard has strips of metal underneath the board and connect the holes on the top of the board. Note that the top and bottom rows of holes are connected horizontally and split in the middle while the remaining holes are connected vertically. [5]



**Figure 4.6:** Breadboard.

### 4.4 Software section:

#### 4.4.1 Raspbian OS:

It is one of the famous versions of Linux Based on Debian, Raspberry Pi made a special version called Raspbian, this is the basic Operating system for RPi and there are other operating systems that can be used but in our project we used Raspbian.



#### **4.4.2 PuTTY:**

PuTTY terminal window on Windows can use SSH or Telnet protocols for remote access but we only use SSH in our project and we find that we prefer working on our Raspberry Pi via SSH because its more secure, encrypted communications protocol designed to ensure the password and data are maximally protected.

#### **4.4.3 VNC:**

VNC is a graphical desktop-sharing system, allows to remotely access a computer and use its desktop over the Internet. And its works in a client/server model and uses a specialized network protocol called remote frame buffer (RFB).

#### **4.4.4 Apache server:**

Apache Web Server is an open-source web server creation, deployment and management software. Initially developed by a group of software programmers, it is now maintained by the Apache Software Foundation. [6]

#### **4.4.5 Remote.it application:**

Remote.it virtual private internet (VPI) is a secure communication and networking solution that provides the benefits of a VPN without the need for VPN appliances or complicated security management. Simply install the small software on our Raspberry Pi, register it on our own VPI, and close off any ports that could be attacked by hackers, including 22 (SSH), 80 (HTTP) and 443 (HTTPS). [7]

### **4.5 Implementation section:**

This Project is divided into two parts, the first section is hardware which includes all components and their connections. The second section is software that is all codes, commands and libraries that this project needs.

#### **4.5.1 Set up of Raspberry Pi 3 Model B:**

To set up raspberry Pi we need:

HDMI cable.

Micro SD card (16GB, Class10).

Wire Keyboard (or wireless).

Wire Mouse (or wireless).

Raspberry Pi case (Optional).

#### 4.5.1.1 Operating System Installation:

At this point we put the SD card in the reader and in raspberry pi OS imager we choose the name of the SD card and the OS type we want to download it

This imager program will formatting and writing the OS on the card as shown on the

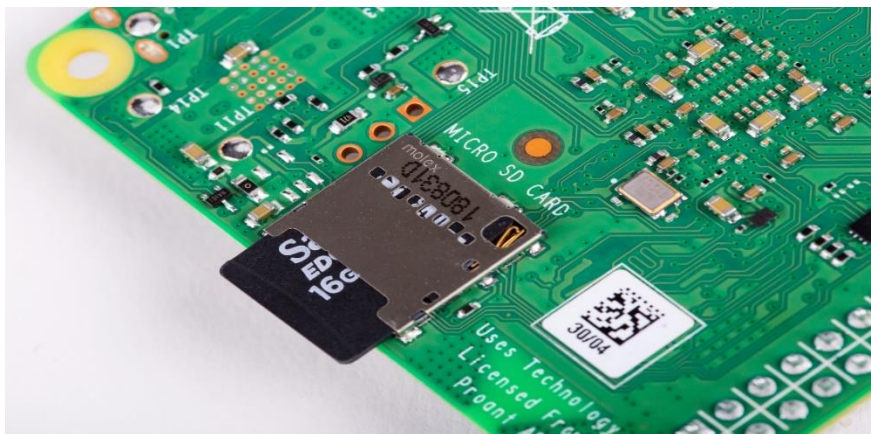
**Figure 4.7:**



**Figure 4.7:** Raspberry Pi Imager.

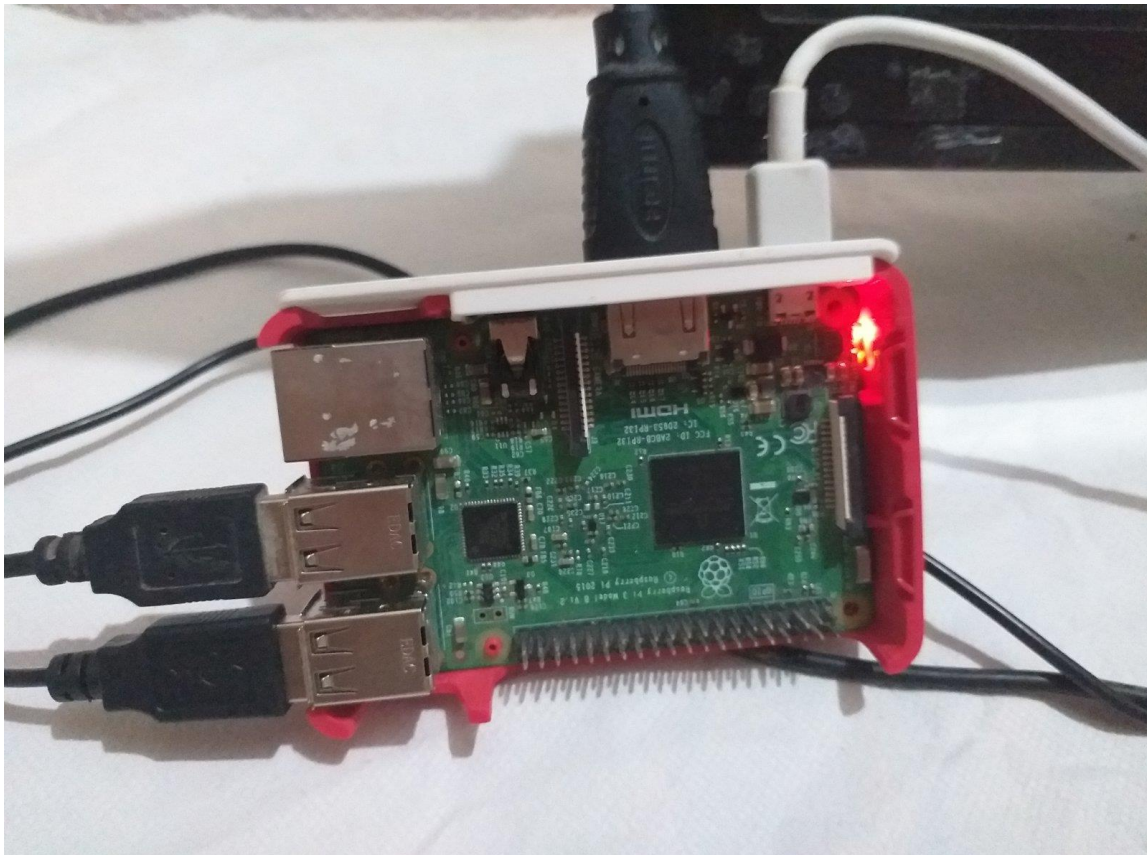
#### 4.5.1.2 Running the Raspberry Pi:

We insert the memory card containing the operating system in the micro SD card slot on the Raspberry Pi as shown in **Figure 4.8:**



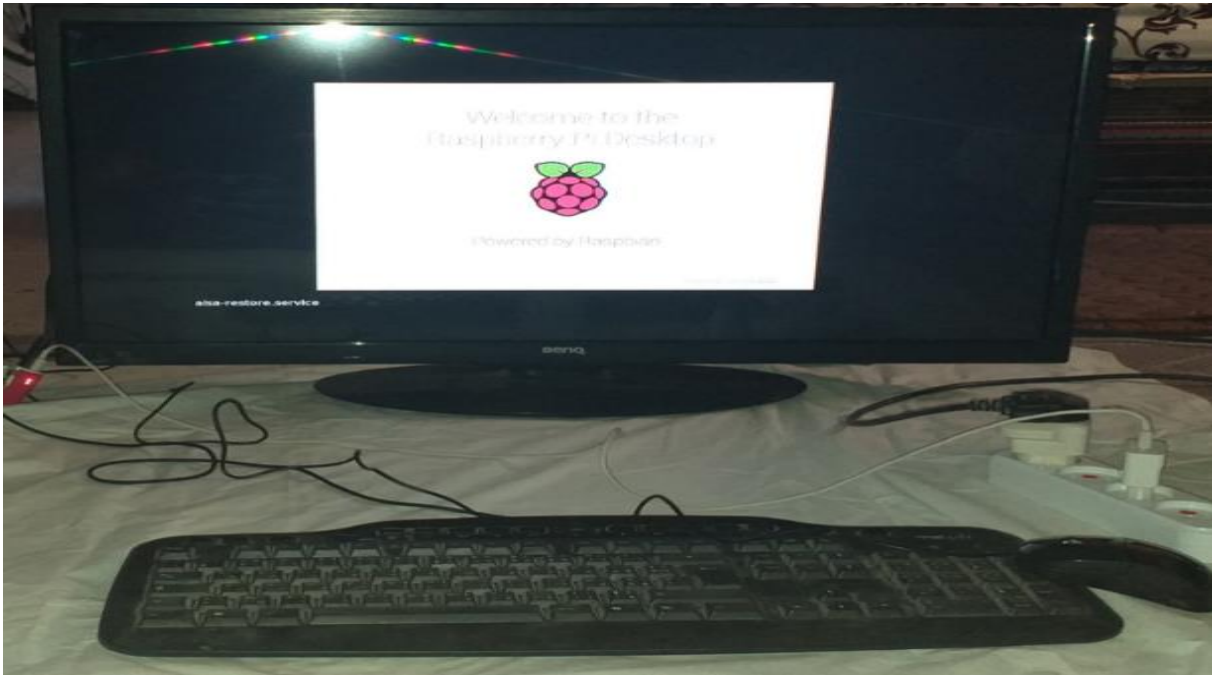
**Figure 4.8:** Memory card slot.

We connect the Raspberry Pi device to HDMI cable, Keyboard, Mouse and Power (5V, 2.5A for Raspberry Pi 3 Model B).



**Figure 4.9:** Raspberry Pi with accessories.

Then we access the screen to Raspberry Pi with HDMI cable after that connect it to electricity with charger to start the installation using the installation interface. We choose confirm and then ok then it appears on the screen welcome to Raspberry Pi and wait for few seconds to see install successfully.



**Figure 4.10:** Installation of OS on RPi.

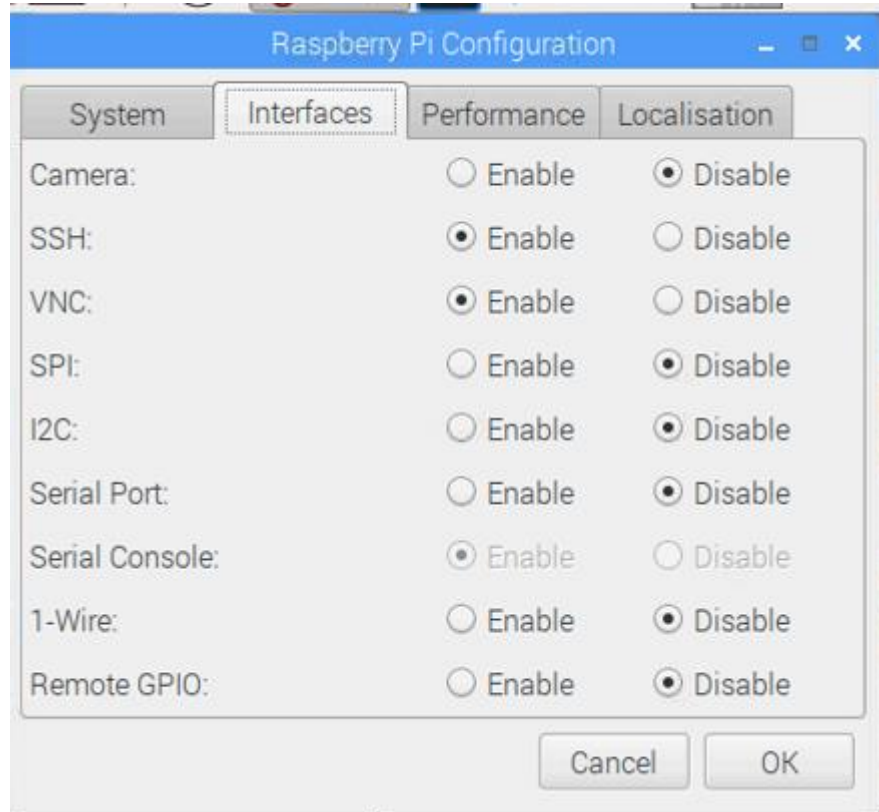
It is also important to set regional and security parameters such as the country and password.



**Figure 4.11:** Setting Raspberry Pi password.

### 4.5.2 Static IP

First we go to the details then Raspberry Pi Configuration then interfaces and in the field of SSH and VNC we put the enable option as shown in the **figure 4.12**, then we shut down the raspberry pi and restart it.



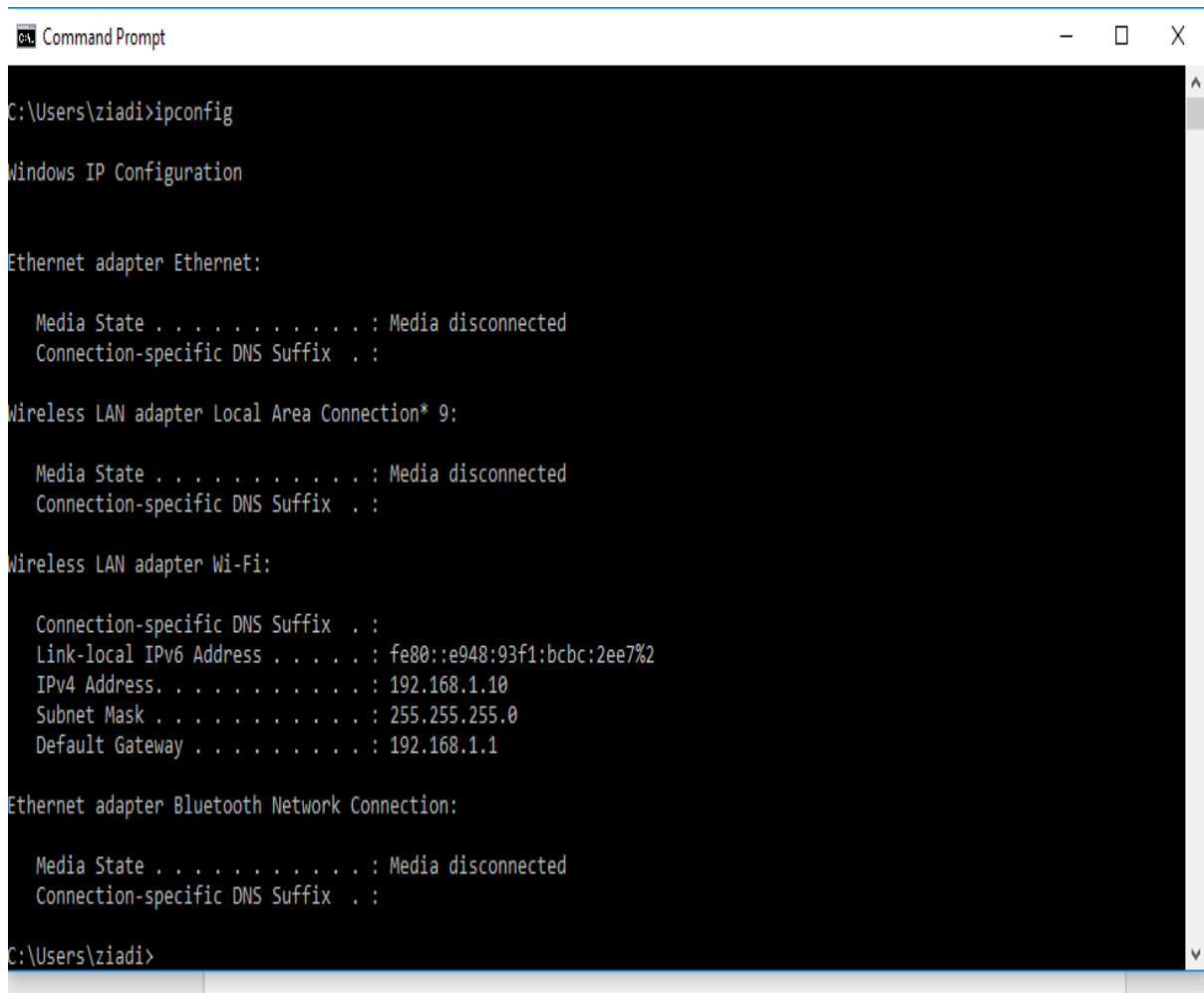
**Figure 4.12:** Raspberry Pi Configuration interface.

We enter the network to Raspberry Pi. After we turn the IP to static because by default the router gives to raspberry Pi dynamic IP with DHCP service. It mean that automatically gives on its own the IP to the RPi, so when we turn off the device and turn it on it gives a new IP.

In this case, each time we must know the IP of the Raspberry Pi, and the solution is to change the IP from dynamic to static meaning IP is fixed.

We choose the IP for my network's Raspberry Pi. In order to know the network address, we go to the terminal of the PC and we write: `>ipconfig`

Through this command we find that the network address (Default Gateway) is: 192.168.1.1



```
Command Prompt
C:\Users\ziadi>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 9:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::e948:93f1:bcbc:2ee7%2
    IPv4 Address. . . . . : 192.168.1.10
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.1

Ethernet adapter Bluetooth Network Connection:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

C:\Users\ziadi>
```

**Figure 4.13:** Network Address.

Now that we know the network address, we write the following command in the terminal:

```
Sudo nano /etc/dhcpd.conf
```

This command allow the configuration of DHCP.

After the above command is complete, we write:

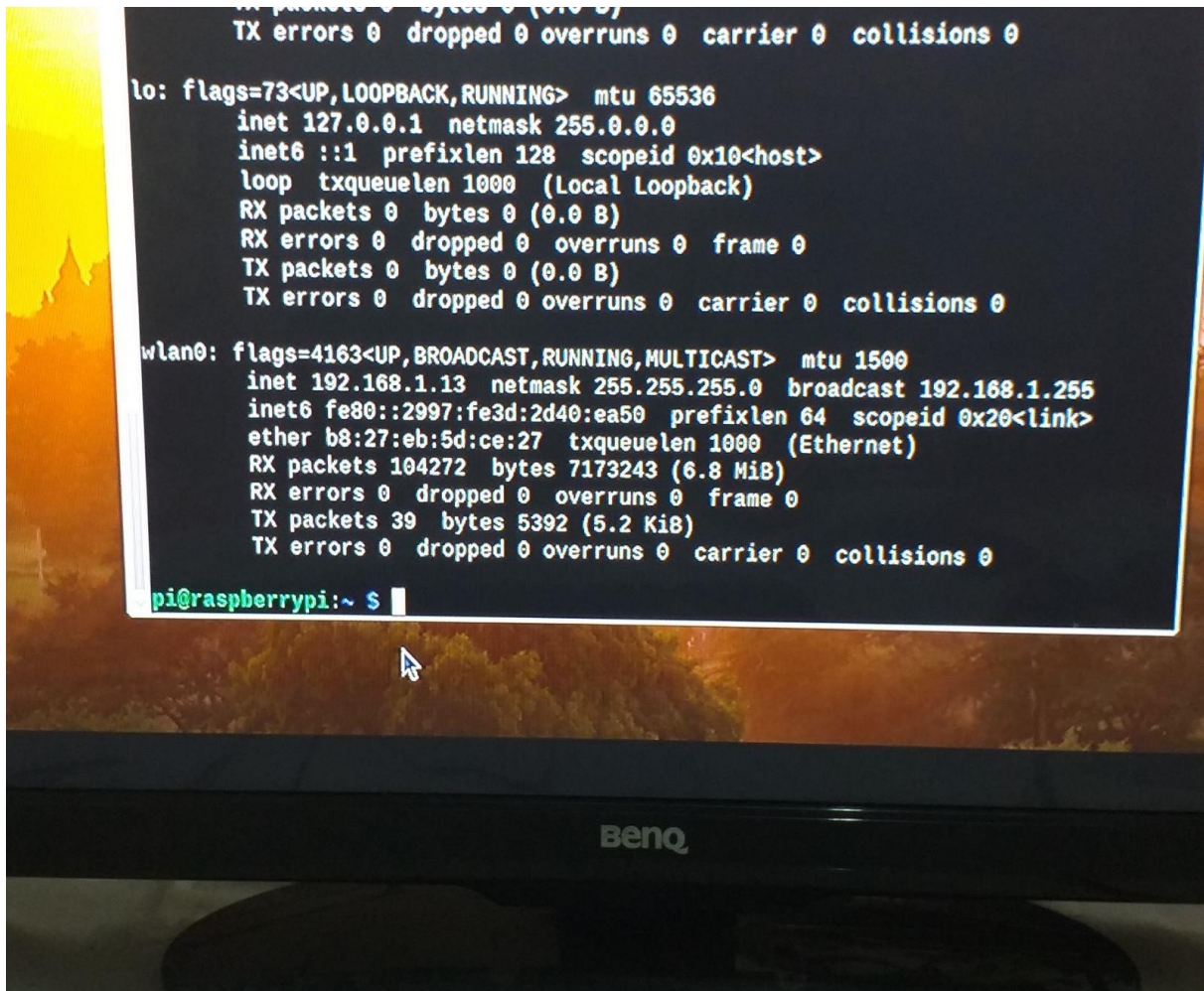
```
Interface WLAN0
```

```
Static ip address = 192.168.1.13
```

```
Static routers = 192.168.1.1
```

```
Static domain_name_servers = 192.168.1.1
```

Now the IP address of our device Raspberry Pi is fixed.

A terminal window on a Raspberry Pi showing network configuration. The terminal output includes details for the loopback interface 'lo' and the wireless LAN interface 'wlan0'. The 'wlan0' interface is configured with a static IP address of 192.168.1.13, a netmask of 255.255.255.0, and a broadcast address of 192.168.1.255. It also shows the MAC address b8:27:eb:5d:ce:27 and the interface is currently up and running.

```
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
  inet 127.0.0.1 netmask 255.0.0.0
  inet6 ::1 prefixlen 128 scopeid 0x10<host>
  loop txqueuelen 1000 (Local Loopback)
  RX packets 0 bytes 0 (0.0 B)
  RX errors 0 dropped 0 overruns 0 frame 0
  TX packets 0 bytes 0 (0.0 B)
  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
  inet 192.168.1.13 netmask 255.255.255.0 broadcast 192.168.1.255
  inet6 fe80::2997:fe3d:2d40:ea50 prefixlen 64 scopeid 0x20<link>
  ether b8:27:eb:5d:ce:27 txqueuelen 1000 (Ethernet)
  RX packets 104272 bytes 7173243 (6.8 MiB)
  RX errors 0 dropped 0 overruns 0 frame 0
  TX packets 39 bytes 5392 (5.2 KiB)
  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
pi@raspberrypi:~$
```

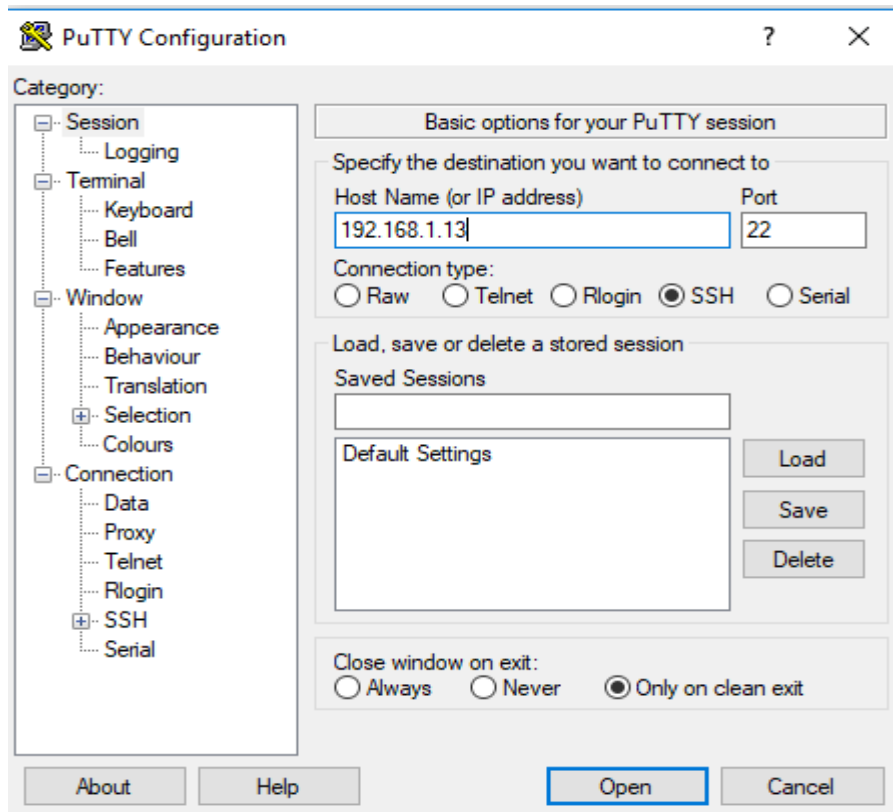
**Figure 4.14:** Static WLAN IP.

#### 4.5.3 SSH Remote access:

On a device that uses mac or Linux, it has SSH enabled automatically in the terminal but for windows operating system as it is in my computer we need to download PuTTY.

Putty is a tool to enter to the terminal of Raspberry Pi device, by entering the RPi address and port of SSH and click open. Then we write the name and password.

Now we have a remote access by SSH, from the PC we can access the Raspberry Pi terminal.



**Figure 4.15:** PuTTY Configuration.

This way we don't need independent keyboard, mouse, screen and HDMI cable, we can enter to the Raspberry Pi via computer using SSH or PuTTY but only in a LAN network.

#### 4.5.4 VNC Remote Desktop:

We will see the desktop of the Raspberry Pi inside a window on our computer (PC), so without connect the RPi with screen HDMI, mouse, and keyboard this is the best solution for Remote Desktop access.

We loaded VNC on Raspberry Pi with the following command on the terminal and we write:

```
Sudo apt-get install tightvncserver
```

Then we choose: Y

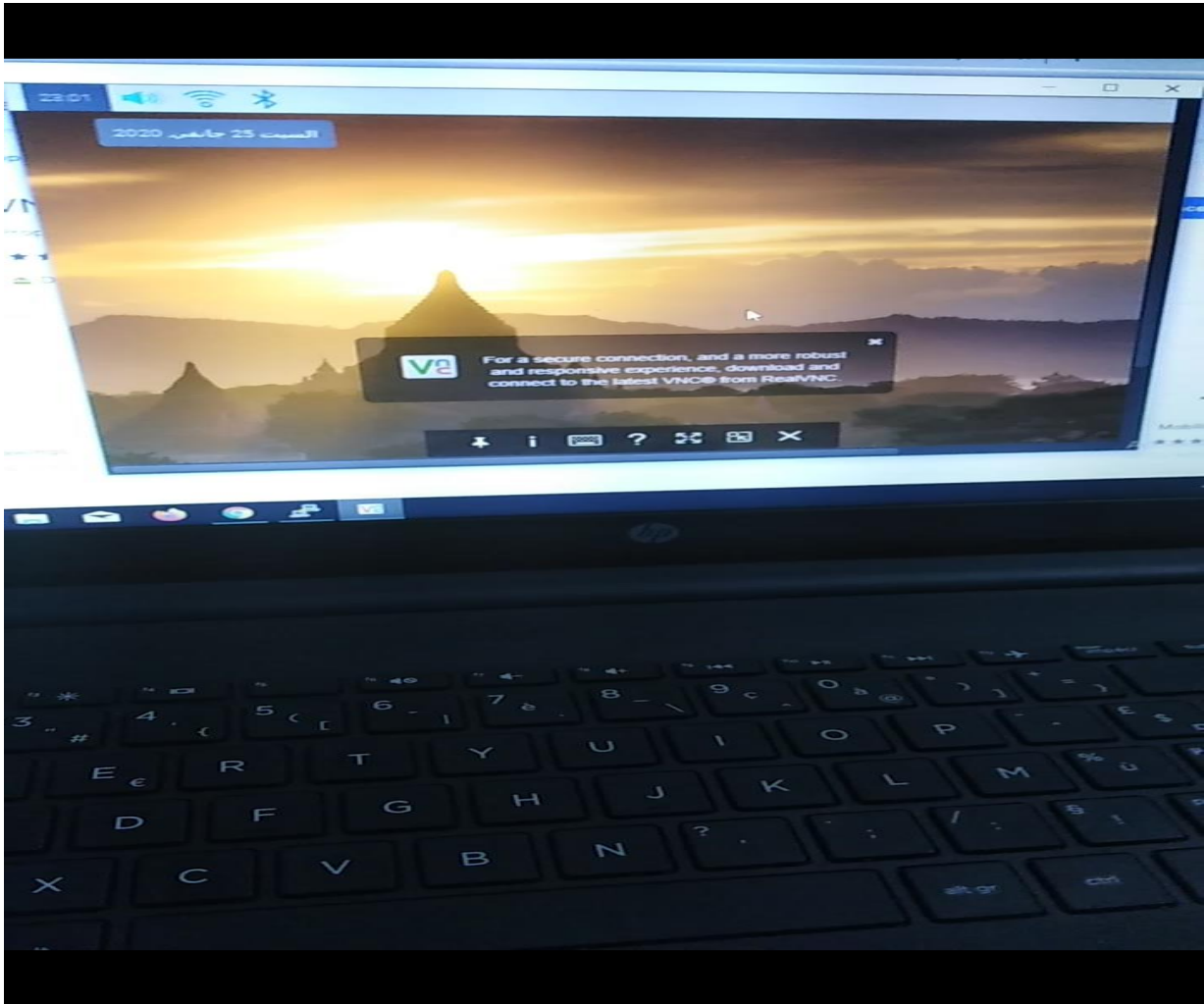
This is how VNC was loaded after that we write in the terminal:

```
vncserver
```

Then we enter at VNC viewer our password we set and IP address with port.

Now we can see the desktop of our Raspberry Pi from PC with VNC viewer as shown on the **Figure 4.16:**





**Figure 4.16:** Remote Desktop.

The VNC viewer program from the PC allows the access to RPi, and the Desktop of RPi displayed to us in the PC via it.

#### **4.5.5 Apache server:**

A server is a device with a set of programs providing services requested by client. And one of types of servers the web server, we use apache web server in our project

- **Installation of Apache server:**

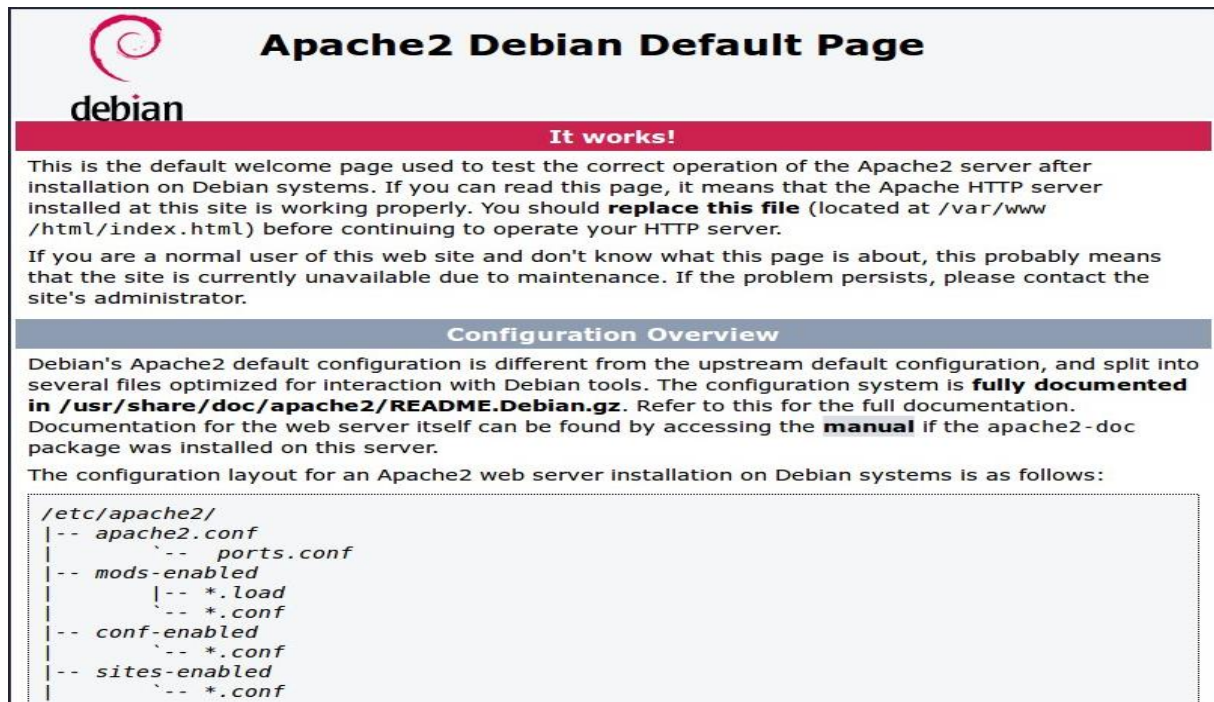
To install Apache server, we use the following command at the terminal on the Raspberry Pi:

```
Sudo apt-get update
```

```
Sudo apt-get install apache2 -y
```

Then we go to the browser in Raspberry and write localhost or the address of RPi 192.168.1.13 we find the apache web server is installed.

We write Raspberry Pi address in any browser or localhost on any device connected to the same network we find the apache page.



**Figure 4.17:** Apache server.

- **Installation of PHP:**

We need PHP in the server page so we installed it on RPi and we write on the terminal the following:

```
Sudo apt-get install php libapache2-mod-php -y
```

- **Creating a Database:**

We Downloaded MySQL Database and we write at the terminal of RPi the commands:

```
Sudo apt-get install mysql-server
```

Then we wrote Y that mean yes.

```
Sudo mysql_secure_installation
```

Then it asked for a password we write it.

To enter to the database installed, we write:

```
Sudo mysql -u root -p
```

- **Installation of PHPmyadmin:**

To install PHP myadmin by typing the following command on the terminal for Raspberry Pi:

Sudo apt-get install phpmyadmin

- **Connecting apache PHPmyadmin:**

The connection is done through these commands:

```
Sudo nano /etc/apache2/apache2.conf
```

Then we went down to the last line in configuration and wrote the following:

```
Include /etc/phpmyadmin/apache.conf
```

Then ctrl x then y then enter to save the apache configuration file.

Here we will restart apache with this command:

```
Sudo /etc/init.d/apache2 restart
```

- **Connecting PHP myadmin with www file:**

All apache files are located in www file exactly in the html folder and for linking www file with PHP myadmin we wrote this command:

```
Sudo ln -s /usr/share/phpmyadmin /var/www/html
```

Then we create a user to log in to PHP myadmin.

To enter the php myadmin page and we write in the browser: 192.168.1.13/phpmyadmin.

## 4.5.6 Webcam server using RPi:

Motion is free and open source software for webcam server.

### Installation of webcam software on RPi:

On command prompt of RPi we write:

```
Sudo apt-get update
```

```
Sudo apt-get install motion
```

After that we connect our cam with RPi and we go to file configuration of motion software and write:

```
Sudo nano etc/motion/motion.conf
```

Now we change at the motion configuration some options:

```

GNU nano 3.2 /etc/motion/motion.conf
# Rename this distribution example file to motion.conf
#
# This config file was generated by motion 4.1.1
# Documentation: /usr/share/doc/motion/motion_guide.html
#####
# Daemon
#####
# Start in daemon (background) mode and release terminal (default: off)
daemon on

# File to store the process ID, also called pid file. (default: not defined)
process_id_file /var/run/motion/motion.pid

#####
# Basic Setup Mode
#####
[ Read 737 lines ]
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line

```

**Figure 4.18:** Motion Configuration.

Daemon on

Framerate 1500

Stream port 8081

Stream\_localhost off

Webcontrol\_html\_output off

Quality 100

Width 640

Height 480

Brightness 20

Post\_capture 5

To save we press Ctrl x → Yes → Enter.

Then we go to this file:

Sudo nano etc/default/motion

And change the option “state\_motion\_daemon” from no to yes and save with Ctrl x → Yes

→ Enter.

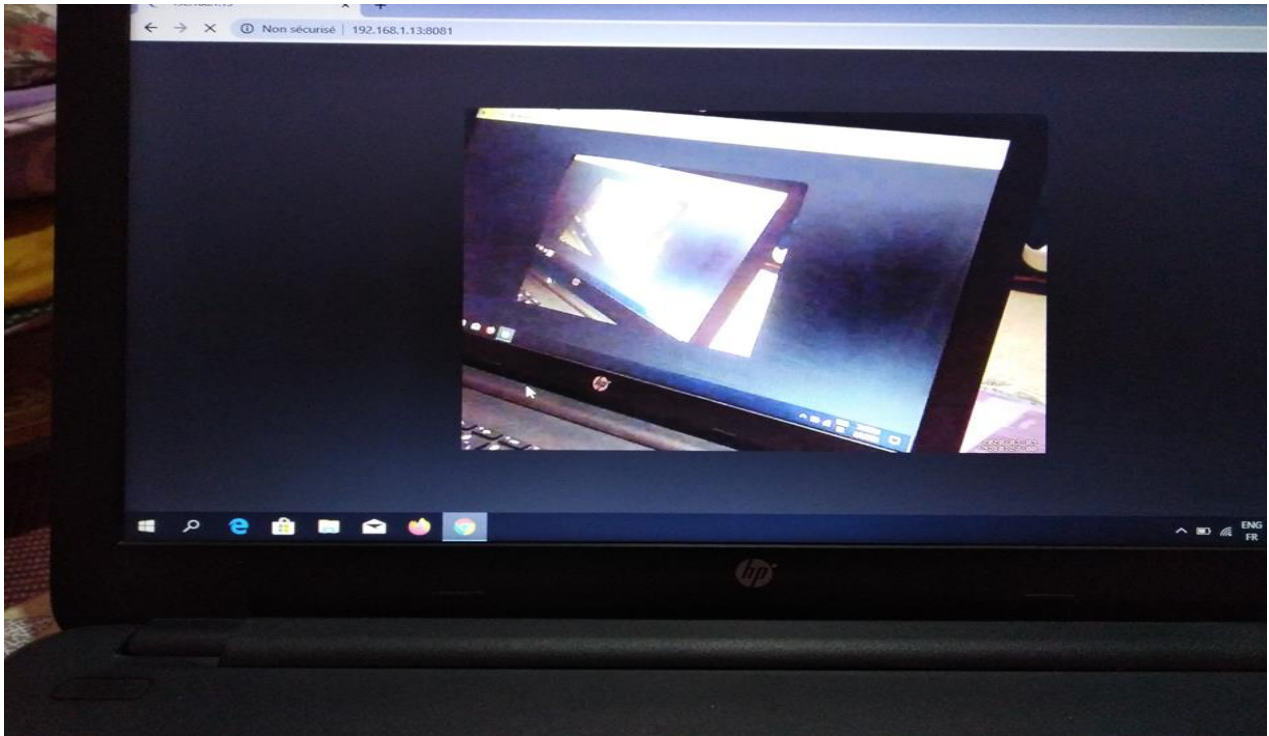
Then we restart and start the webcam server software “motion” by:

```
Sudo service motion restart
```

```
Sudo motion
```

Now we can see by this webcam live video at the real time with the TCP port 8081.

We write in the browser 192.168.1.13:8081



**Figure 4.19:** Live video steaming.

#### 4.5.7 LED Control:

We must install the wiringPi library. That is a library written in C used to access GPIO pins on Raspberry and we are using WiringPi library for Raspberry Pi GPIO access.

The WiringPi Library is installed with this commands:

```
Sudo apt-get install git-core
```

```
Cd Desktop
```

```
Git clone git://.drogon.net/wiringPi
```

```
Cd wiringPi
```

```
Ls
```

```
./build
```

Gpio -v

Gpio readall

Cd /var/www

Ls

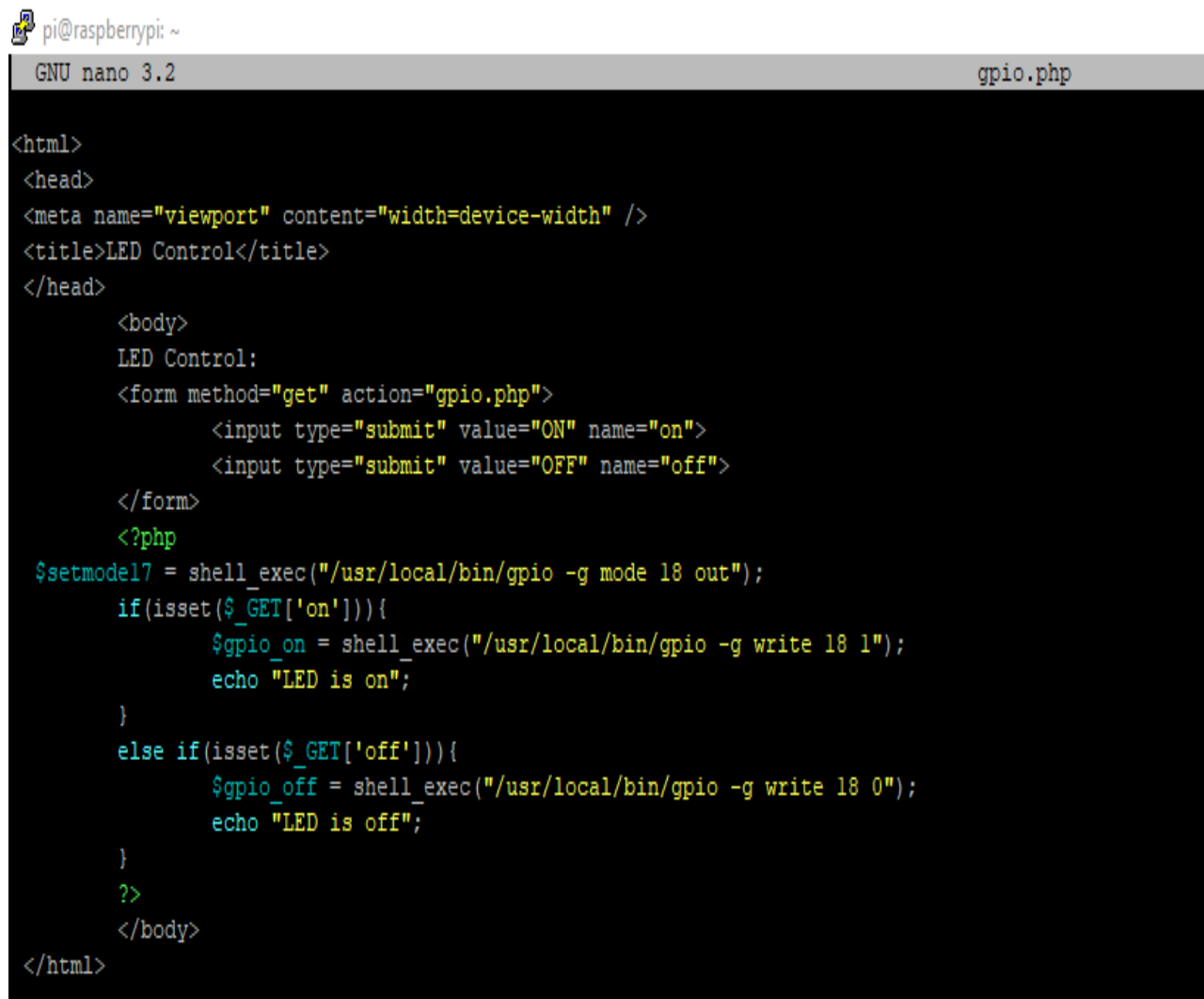
Cd html

Ls

Now for open the new file we write:

Sudo nano gpio.php

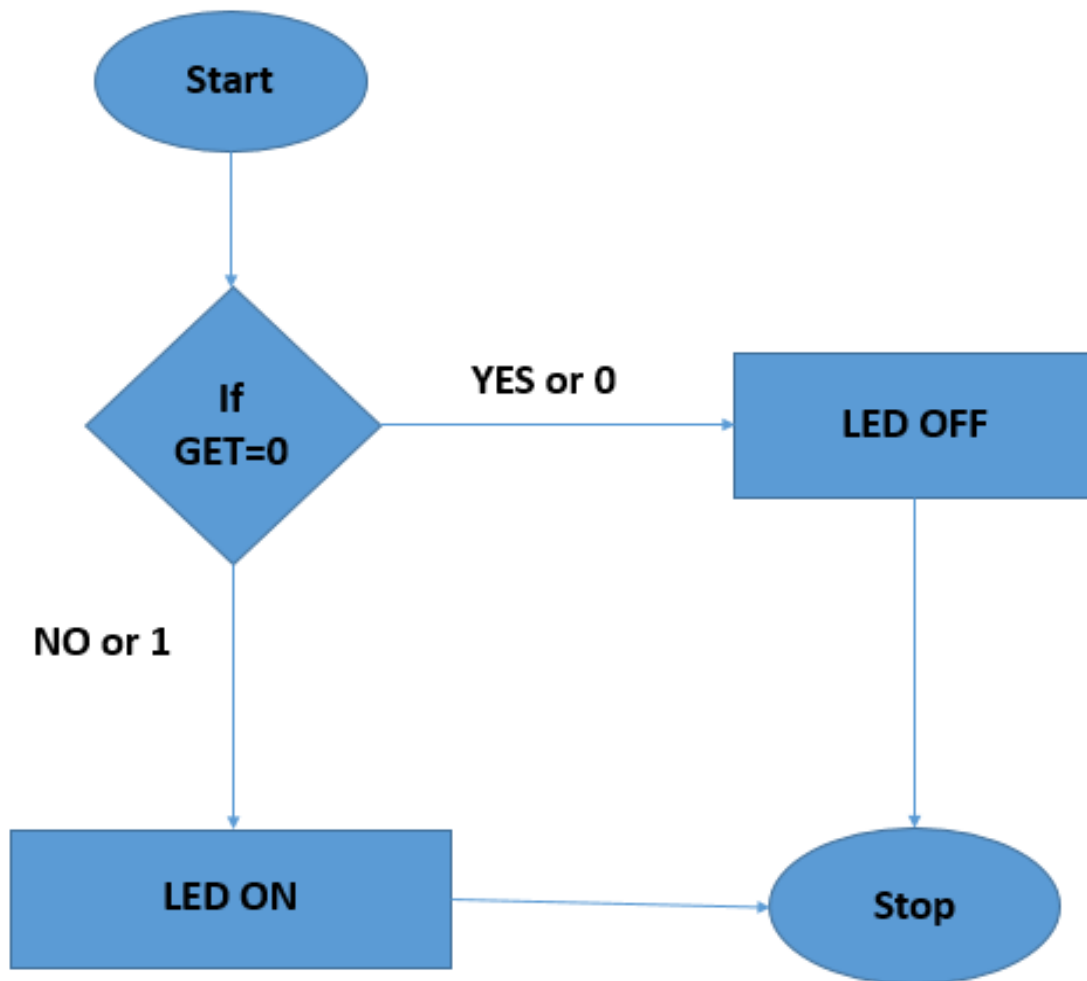
Then we put the code of PHP to control the LED:



```
pi@raspberrypi: ~
GNU nano 3.2                               gpio.php
<html>
<head>
<meta name="viewport" content="width=device-width" />
<title>LED Control</title>
</head>
  <body>
    LED Control:
    <form method="get" action="gpio.php">
      <input type="submit" value="ON" name="on">
      <input type="submit" value="OFF" name="off">
    </form>
    <?php
    $setmode17 = shell_exec("/usr/local/bin/gpio -g mode 18 out");
    if(isset($_GET['on'])){
      $gpio_on = shell_exec("/usr/local/bin/gpio -g write 18 1");
      echo "LED is on";
    }
    else if(isset($_GET['off'])){
      $gpio_off = shell_exec("/usr/local/bin/gpio -g write 18 0");
      echo "LED is off";
    }
    ?>
  </body>
</html>
```

Figure 4.20: LED Control Code.

**Explication of code:** (Organigram)



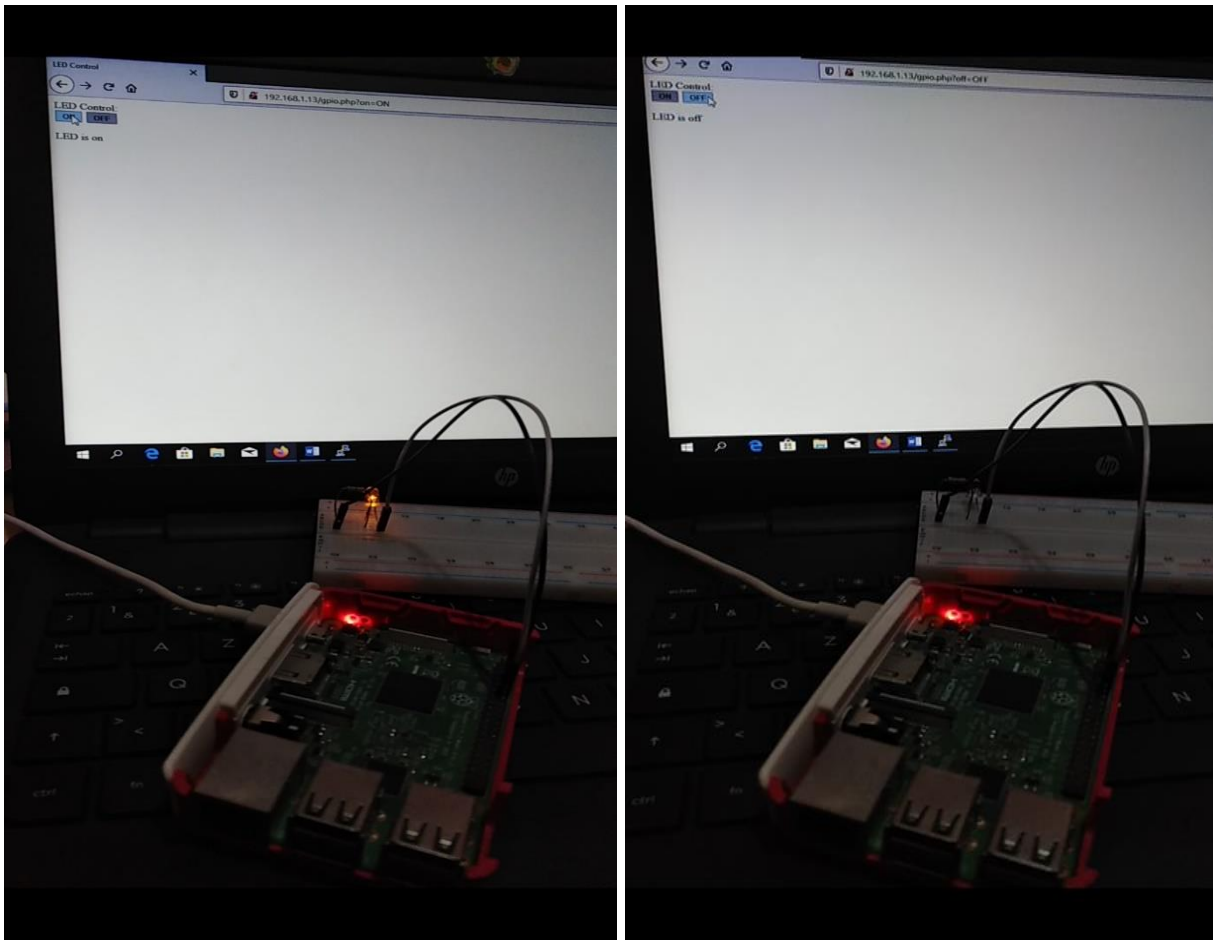
**Figure 4.21:** Organigram of LED code.

A resistor is used in series with Led to limit the current. The negative side accessed to GND (PIN 6), and the positive side accessed to BCM18 (PIN 12) of Raspberry Pi.

In order to directly control the LED we write at the browser:

192.168.43.13/gpio.php

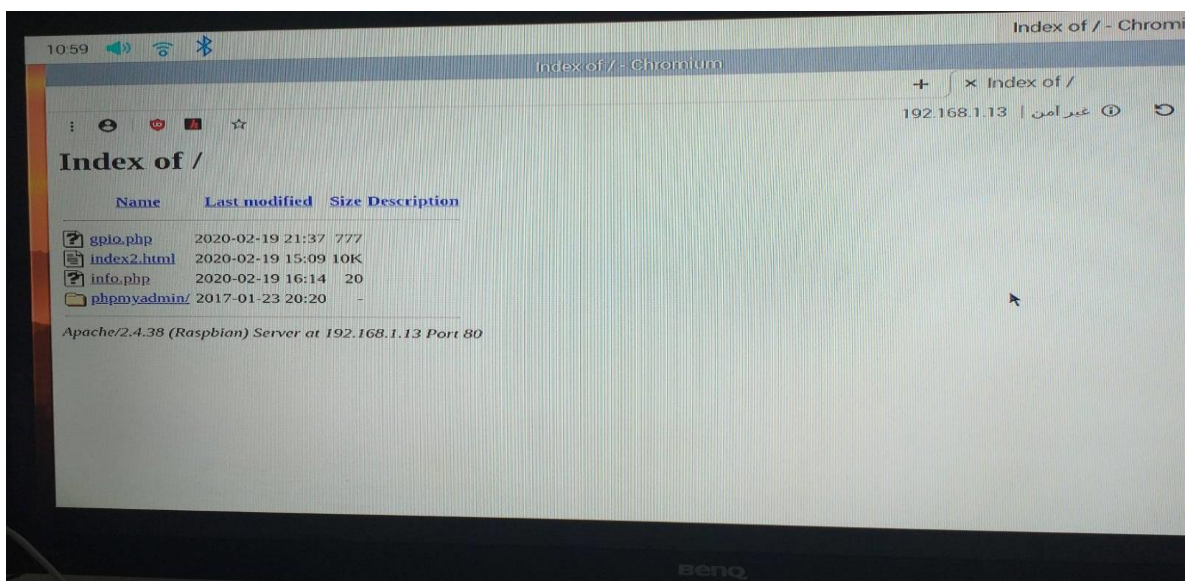
As shown in the figure the LED ON or OFF:



**Figure 4.22:** LED control from web page.

**Remark:**

Now when we write the ip address of Raspberry Pi or Localhost in the any browser on the same network it show all files we create it of apache server as shown in the **Figure 4.23:**



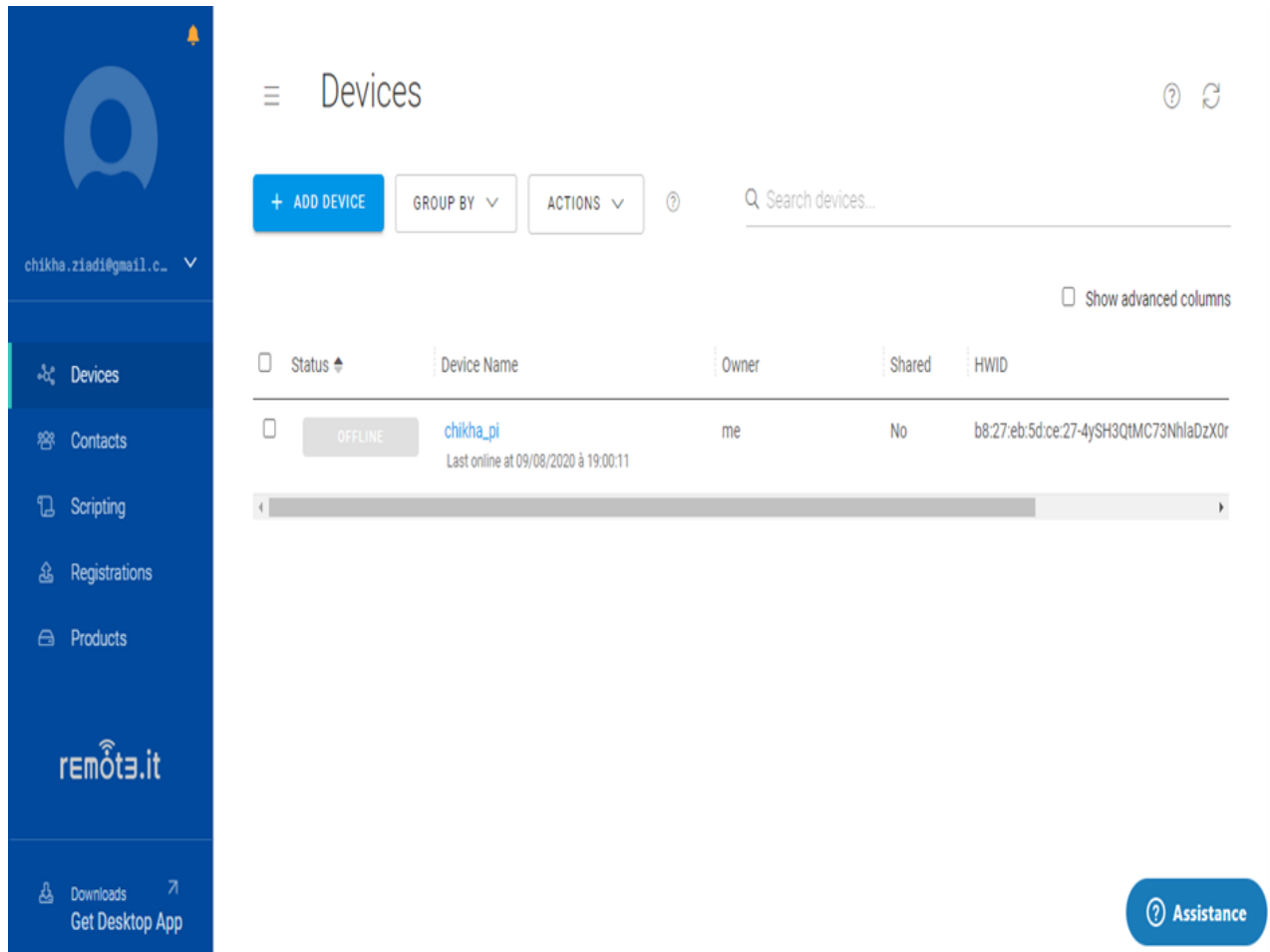
**Figure 4.23:** Apache index.



### 4.5.8 Access from anywhere with Remote.IT:

We can Access Raspberry Pi ports from anywhere in the world without port forwarding.

We create account of remote.it with email and this is my account on the **Figure 4.24**:



**Figure 4.24:** Remote.it account.

We install the remote.it package in Raspberry pi:

Sudo apt-get update

Sudo apt install connectd

Sudo connectd\_installer (this command for add ports we choose it)

```

pi@raspberrypi ~
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Oct 27 14:21:31 2019 from 192.168.0.100
pi@raspberrypi:~$ sudo apt-get update
Hit:1 http://archive.raspberrypi.org/debian buster InRelease
Hit:2 http://raspbian.raspberrypi.org/raspbian buster InRelease
Reading package lists... Done
pi@raspberrypi:~$ sudo apt install connectd
Reading package lists... Done
Building dependency tree
Reading state information... Done
connectd is already the newest version (2.3.17).
0 upgraded, 0 newly installed, 0 to remove and 52 not upgraded.
pi@raspberrypi:~$ sudo connectd_installer

remote.it connection installer Version: 2.3.17 lib_v2.1.17
Modified: April 15, 2019 (library) April 17, 2019
Build date: Thu Apr 18 18:21:59 PDT 2019
curl with SSL support installed.

Checking the daemon for compatibility...

Using architecture arm-linux-gnueabihf...

Checking your network for compatibility...

Network connection OK...
Your network is compatible with remote.it services.

***** Sign In Menu *****

1) Sign in to your existing remote.it account
2) Request a code for a new remote.it account
3) Enter a verification code received in e-mail
4) Exit

```

Figure 4.25: Installation of Remote.it package on RPi.

The services that we add it to Remote.it application:

The following Services are available on Device **chikha\_pi**.

Status	Service	Application	Port	Service ID
⊖	chikha_pi	Bulk Service		
⊖	chikha_ssh	SSH	22	80:00:00:00:01:03:DD:E8
⊖	chikha_tcp_camera	Generic TCP	8081	80:00:00:00:01:03:DD:E9
⊖	chikha_web_pi	HTTP	80	80:00:00:00:01:03:DD:EA
⊖	camera-RTSP-pi	Generic TCP	554	80:00:00:00:01:03:E6:15

**Note on connections:** From the remote.it portal, connections will either be IP restricted or reverse proxy (HTTP/Secure Web).

**Reverse proxy:** For security purposes, this will be a randomly generated text string value for each connection, making it highly unlikely that anyone will discover the link. Once the connection closes, the link will no longer be valid, even to you.

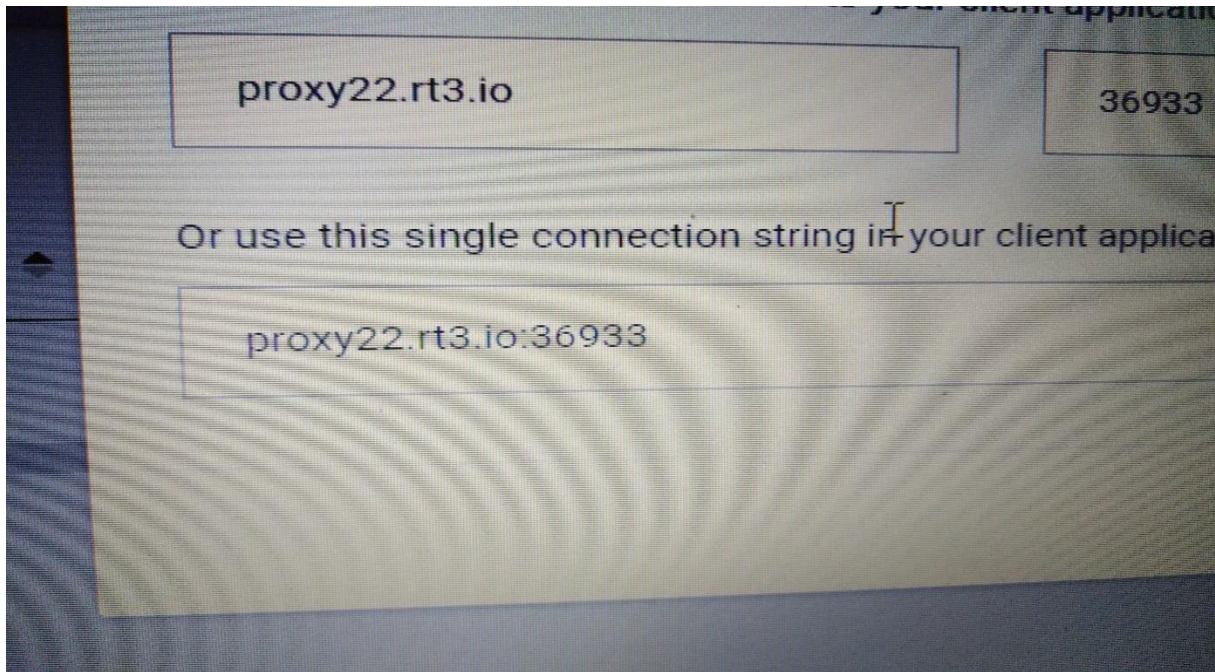
Figure 4.26: Remote.it device services.

**Explanation:**

After adding the ports to be controlled in Raspberry Pi, we can control SSH, HTTP and TCP ports from any other networks.

When we click on our services we want to control as shown in **Figure 4.26**, it gives us a Proxy URL with port as shown on the **Figure 4.27** and this is what we copy/past it in any browser and not from the same network where Raspberry Pi is connected to.

The Remote.it Proxy URLs are temporary and expire after 8 hours ensure protection. Thus, access to our device is done in a safe way.

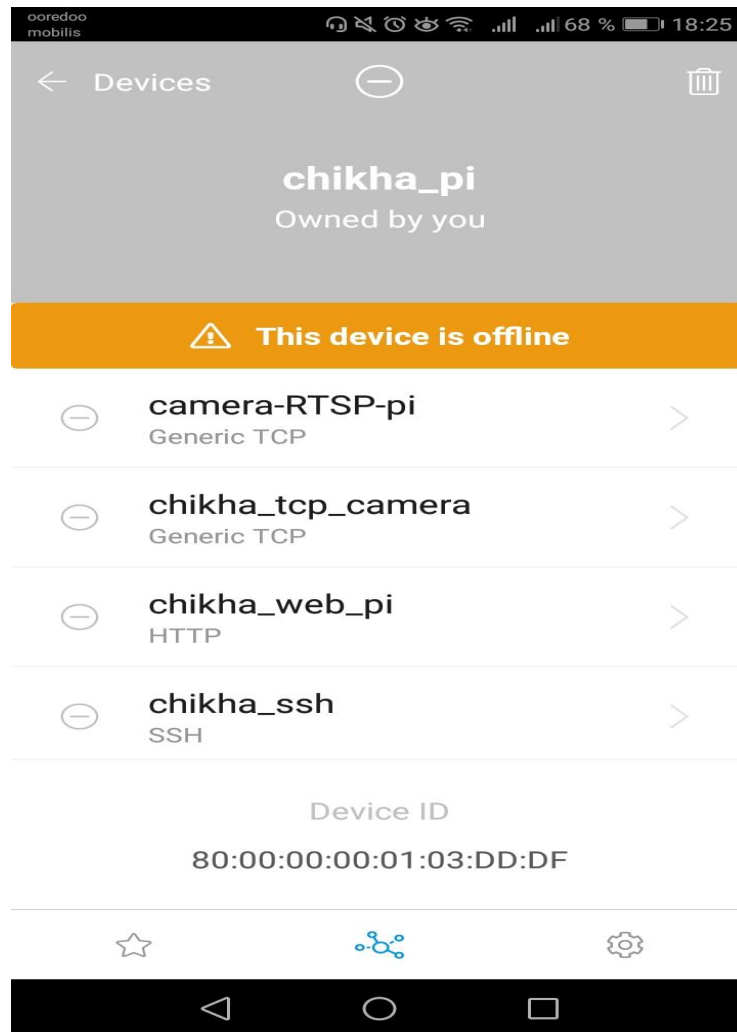


**Figure 4.27:** Proxy URL with port.

**Remark:**

We can download the Remote.it application from play store for android phones too.

And we login to our account from this app.



**Figure 4.28:** Remote.it application from smartphone.

#### About Remote.it solution:

Hosted network provider remote.it expands Virtual Private Internet solution to protect devices, cloud, web and edge application services from unauthorized TCP/ IP access.

Hosted network provider remote.it today announced the global commercial availability of its enterprise Virtual Private Internet (VPI) solution. remote.it enterprise VPI is a new way to create completely private network overlays, enabling secure connectivity for routers, connected devices, as well as cloud, container, web and edge applications and devices.

Central to the remote.it solution is the company's core patented networking technology. The software allows the TCP/IP stack's localhost addresses to then be used as doorways to secure peer-to-peer connections between other remote.it enabled network stacks. remote.it's hosted network solution functions like a private domain name service (DNS), where each node in a network can make a secure direct connection to any other node. Host devices, cloud port, or web and edge services can then assume a 'drop state' rendering them invisible and impenetrable to malicious hackers or programs seeking to exploit IP addresses or open ports, effectively cloaking them from discovery. [7]

## 4.5 Conclusion:

In this practical chapter, all the project phases are presented phase by phase. Firstly: we have drawn up a general schematic explaining our project, secondly: we defined the hardware and software components used. In addition, we have put all the steps for installing a web server, webcam server as well as LED Control.

Finally, we installed and create an account on the Remote.it app and this is for control the Raspberry Pi ports without port forwarding from any network in the world and this is the Best solution because the remote.it is faster to set up, simpler to manage, less costly, and close off any ports that could be attacked by hackers and safer than other solution.

# **GENERAL COCLUSION**

## GENERAL CONCLUSION

As human beings, we've always had deeply personal with our homes. After all, a home is an oasis from the chaos and bustle of the real world. We seek to make our homes as comfortable and personal as possible. From early cave paintings to today's modern technology, the way we live and the way we interact with our home has evolved significantly over the years. And one of the most prominent development that have emerged in the world of technology is the internet of thing. IoT is just getting started. In the coming years, new smart home devices will skyrocket in number and variety, all connected to the internet and controlled with smart home apps and cell phones, in 2020 20 billion devices will be connected to internet.

The goal of this humble project is: control by a web server in the Raspberry Pi IoT device, control by a webcam server to Access to live video streaming via web cam that connected to the Raspberry Pi device, Control Raspberry Pi from any network in the world (WAN) in a safe way as no exposure to hackers attack.

### Results:

- LED Control via apache web server with http (port: 80).
- Access to live video streaming with TCP (port: 8081).
- Control a Raspberry Pi from any network other than the network connected to RPi by applying Remote.it solution without resorting to port forwarding solution.

### Constraints:

- The lack of some electronic devices in the laboratory.
- Corona pandemic, which disrupted work, especially the practice part.
- Unstable internet, which disrupted the installation of software.

### Perspectives:

For the next students who will work this project we suggest:

- Use high speed SD card such as class 10 or more to ensure the speed of writing to the memory, and 32 GB for that you don't need to use another SD card, in case the size is not enough for the work.

- Never turn off the Raspberry Pi directly from power supply because it causes problems at work.
- The Raspberry Pi needs a cooling fan otherwise the RPi will be damaged.
- There are several solutions for controlling RPi on any network and without port forwarding. This is an outdated and insecure solution that allows hackers to easily access a network because the ports are open, in our project we use Remote.it and the advantages of this solution: Low cost, Secure, Simple to use.
- We have reached live video streaming by the TCP and with this protocol there is a delay in the arrival of data, that is why I suggest improving the video because it is a live broadcast that needs a real-time information access without delay so it needs UDP. One of the most important protocols needed for live video streaming (RTP, RTSP, and RTCP).



# **Bibliography**

**BIBLIOGRAPHY****Chapter 1**

[1] <https://www.zdnet.com/article/what-is-the-internet-of-things-everything-you-need-to-know-about-the-iot-right-now/>

[2] [https://simple.wikipedia.org/wiki/Internet\\_of\\_things](https://simple.wikipedia.org/wiki/Internet_of_things)

[3] Dr. V. Bhuvaneshwari<sup>1</sup> Dr. R Porkodi<sup>2</sup>. (March 2014). The Internet of Things (IoT) Applications and Communication Enabling Technology Standards: An Overview.

[4] M.U. Farooq, Muhammad Waseem, Sadia Mazhar, Anjum Khairi, Talha Kamal. (March 2015). A Review on Internet of Things (IoT)

[5] [https://en.wikipedia.org/wiki/Wireless\\_sensor\\_network](https://en.wikipedia.org/wiki/Wireless_sensor_network)

[6] Umer Farooq. (February 2019). Wireless Sensor Network Challenges and Solutions.

[7] <https://www.sciencedirect.com/topics/computer-science/wireless-sensor-networks>

[8] BOUCETTA ABDEL HAKIM. (2019). Development of a smart mobile garden robot for supervising greenhouses.

[9] <https://www.investopedia.com/terms/c/cloud-computing.asp>

[10] IOT-OPEN.EU: Introduction to the IoT Course book in English, <http://iot-open.eu/download/io1-introduction-to-the-iot/>

[11] <https://behrtech.com/blog/6-leading-types-of-iot-wireless-tech-and-their-best-use-cases/>

[12] <https://www.iotforall.com/what-is-lpwan-lorawan/>

[13] <https://www.embedded-computing.com/guest-blogs/how-cellular-iot-is-taking-over-the-world>

[14] What is Cellular IoT? By Peder Rand, April 11, 2018.  
<https://blog.nordicsemi.com/getconnected/what-is-cellular-iot>

[15] Wei Wang, Guangyu He, Junli Wan. (Sept 2011) Research on Zigbee wireless communication technology.

[16] <https://en.wikipedia.org/wiki/Zigbee>

[17] <https://www.lifewire.com/what-is-bluetooth-2377412> .By Melanie Pinola, November 2019

[18] <https://en.wikipedia.org/wiki/Bluetooth>

[19] Carles Gomez, Joaquim Oller, Josep Paradells. (August 2012). Overview and Evaluation of Bluetooth Low Energy: An Emerging Low-Power Wireless Technology.

[20] <https://www.link-labs.com/blog/bluetooth-vs-bluetooth-low-energy>

[21] Dr. Adel Ismail Al-Alawi. (2006). WiFi Technology: Future Market Challenges and Opportunities.

[22] Maria Rita Palattella, Nicola Accettura, Luigi Alfredo Grieco, Gennaro Boggia Mischa Dohler, Thomas Engel. (May 29, 2013). On Optimal Scheduling in Duty-Cycled Industrial IoT Applications using IEEE802.15.4e TSCH

[23] Ashwini Deshpande, Prajakta Pitale, Sangita Sanap. (February 2016). Industrial Automation using Internet of Things (IOT).

[24] S. R. Prathibha, Anupama Hongal, M. P. Jyothi. (16-17 March 2017). IOT Based Monitoring System in Smart Agriculture).

[25] Fan TongKe. (Jan 2013). Smart Agriculture Based on Cloud Computing and IOT.

- [26] <https://www.digiteum.com/iot-agriculture>
- [27] Zainab Alansari, Safeeullah Soomro, Mohammad Riyaz Belgaum, Shahaboddin Shamshirband. (December 2016). The Rise of Internet of Things (IoT) in Big Healthcare Data: Review and Open Research Issues.
- [28] [https://en.wikipedia.org/wiki/Home\\_automation#History](https://en.wikipedia.org/wiki/Home_automation#History)
- [29] N. OUKRICH. (24Jul2019). Daily Human Activity Recognition in Smart Home based on Feature Selection ,Neural Network and Load Signature of Appliances.
- [30] Rosslin John Robles1, Tai-hoon Kim1. (February 2010). Applications, Systems and Methods in Smart Home Technology: A Review.
- [31] DR. MOHAMMED BELAL HOSSAIN BHUIAN. (26 APRIL, 2015). WEB BASED HOME AUTOMATION.
- [32] Steven Goodwin. (Copyright © 2010). Smart Home Automation with Linux.
- [33] <https://www.digitalhomesystems.com.au/about-smart-home/z-wave-technology>
- [34] <https://www.quora.com/What-are-the-main-parts-of-IoT-systems>
- [35] <https://data-flair.training/blogs/how-iot-works/>
- [36] <https://medium.com/iotforall/iot-explained-how-does-an-iot-system-actually-work-e90e2c435fe7>
- [37] Pallavi Sethi, Smruti R, Sarangi. (26 January 2017). Internet of Things: Architectures, Protocols, and Applications.
- [38] Muhammad Bilal. (17 Jun 2017). A Review of Internet of Things Architecture, Technologies and Analysis Smartphone-based Attacks Against 3D printers.

- [39] Mohammed Riyadh Abdmeziem<sup>1</sup>, Djamel Tandjaoui<sup>2</sup>, Imed Romdhani<sup>3</sup>. (July 2015). Architecting the Internet of Things: State of the Art.
- [40] Gherbia Amel. (6 July 2019). The implementation of security and privacy in IOT device.
- [41] Nitin Naik. (©2017 IEEE). Choice of Effective Messaging Protocols for IoT Systems: MQTT, CoAP, AMQP and HTTP.
- [42] Tetsuya Yokotani, Yuya Sasaki. (2016). IEEE 3rd World Forum on Internet of Things (WF-IoT) , Transfer Protocols of Tiny Data Blocks in IoT and their Performance Evaluation.
- [43] Atul Oak, R.D. Daruwala. (©2018 IEEE). Assessment of Message Queue Telemetry and Transport (MQTT) protocol with Symmetric Encryption.
- [44] Dimitrios Serpanos, Marilyn Wolf. (Springer International Publishing AG 2018). Internet-of-Things (IoT) Systems (Architectures, Algorithms, Methodologies.
- [45] Upendra Tandale, Dr. Bashirahamad Momin, Deva P. Seetharam. (©2017 IEEE). An Empirical Study of Application Layer Protocols for IoT.
- [46] <https://iotboys.com/what-is-amqp-how-amqp-works-for-internet-of-things/>
- [47] <https://opensourceforu.com/2019/10/xmpp-a-communication-protocol-for-the-iot/>
- [48] <https://lx-group.com.au/xmpp-extensible-messaging-protocol-iot/>
- [49] Zhengguo Shen, Shusen Yang, Yifan Yu, Athanasios V. Vasilakos, Julie A. Mccann, Kin K. Leung. (December 2013). A SURVEY ON THE IETF PROTOCOL SUITE FOR THE INTERNET OF THINGS: STANDARDS, CHALLENGES, AND OPPORTUNITIES.

## Chapter 2

- [1] <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>

[2] Vamsikrishna Patchava<sup>1</sup>, Hari Babu Kandala<sup>2</sup>, P Ravi Babu<sup>3</sup>. (2015). A Smart Home Automation Technique with Raspberry Pi using IoT.

[3] Gareth Halfacree. (First published in 2018). THE OFFICIAL RASPBERRY PI BEGINNER'S GUIDE.

[4] <https://makezine.com/2016/02/28/meet-the-new-raspberry-pi-3/>

[5] <https://computer.howstuffworks.com/raspberry-pi2.htm>

[6] Shahriar Shovon. [https://linuxhint.com/raspberry\\_pi\\_3\\_power\\_requirements/](https://linuxhint.com/raspberry_pi_3_power_requirements/)

[7] <https://www.raspberrypi.org/documentation/hardware/camera/>

[8] Anand Nayyar, Vikram Puri. (December- 2015). Raspberry Pi- A Small, Powerful, Cost Effective and Efficient Form Factor Computer: A Review.

[9] Mr: BOUHARAOUA Abderrahim, Mr: BOUKLI HACENE Mohammed Imad. (11 / 06 /2017). Automatisation d'une maison intelligente via une application Android.

[10] BELKACEM ANES, DIDOUH MOHAMED. (Le 12/06/2017). Commande à distance d'un robot mobile via un smartphone Android.

[11] <https://distrowatch.com/table.php?distribution=pidora>

[12] THE Official RASPBERRY PI PROJECTS BOOK, [https://www.raspberrypi.org/magpi-issues/Projects\\_Book\\_v1.pdf](https://www.raspberrypi.org/magpi-issues/Projects_Book_v1.pdf)

[13] <https://www.raspberrypi.org/documentation/remote-access/>

[14] <https://magpi.raspberrypi.org/articles/ssh-remote-control-raspberry-pi>

[15] <https://ewall.com.pk/blog/7-things-youll-always-want-to-explore-about-raspberry-pi>

[16] <https://randomnerdtutorials.com/11-clever-uses-for-your-raspberry-pi/>

[17] Julian Horsey. (April 30, 2019). <https://www.geeky-gadgets.com/raspberry-pi-home-automation-system-30-04-2019/>

[18] Christian Cawley. (Updated December 10/2019). <https://www.makeuseof.com/tag/different-uses-raspberry-pi/>

[19] TechSpot Deals. (September 13, 2017). Build your own robots with this Raspberry Pi 3 board and training bundle. <https://www.techspot.com/news/70738-build-own-robots-raspberry-pi-3-board-training.html>

[20] Lee Assam. (Apr 17, 2018). <https://www.quora.com/What-are-the-advantages-and-disadvantages-of-Raspberry-pi#>

[21] Mirjana Maksimović, Vladimir Vujović, Nikola Davidović, Vladimir Milošević, Branko Perišić. (June 2014). Raspberry Pi as Internet of Things hardware: Performances and Constraints.

### Chapter 3

[1] Micheal Knapp. (Copyright 2017). HTML&CSS .learn the fundamentals in 7 days.

[2] Jennifer Niederst. (Copyright © 2018). Robbins LEARNING WEB DESIGN “Fifth Edition”.

[3] Ben Henick. (Copyright © 2010). HTML & CSS: The Good Parts.

[4] THE WORLD'S LARGEST WEB DEVELOPER SITE, HTML Tutorial [https://www.w3schools.com/html/html\\_basic.asp](https://www.w3schools.com/html/html_basic.asp)

[5] BY IAN LLOYD. (Copyright © 2008). BUILD YOUR OWN WEB SITE THE RIGHT WAY USING HTML & CSS .2ND EDITION.

[6] HTML Tutorial <https://www.tutorialrepublic.com/>

[7] Hege Refsnes, Ståle Refsnes, Kai Jim Refsnes, Jan Egil Refsnes, C. Michael Woodward. (Copyright © 2010). Learn HTML and CSS.

[8] <https://www.geeksforgeeks.org/html-title-attribute/>

[9] <http://www.echoecho.com/htmllinks01.htm>

[10] <https://www.hiddenwebgenius.com/graphic-design/how-to-insert-css-files-into-your-web-pages/>

[11] <https://htmlcheatsheet.com/css/>

[12] <https://www.inmotionhosting.com/support/website/linking-your-css-to-your-website/>

[13] Michael Bowers. (Copyright © 2007). Pro CSS and HTML Design Patterns.

[14] Brian Pfaffenberger, Steven M. Schafer, Charles White, Bill Karow. (Copyright 2004). HTML, XHTML, and CSS Bible 3rd Edition.

[15] Shay Howe. (Copyright © 2014). Learn to Code HTML & CSS Develop & Style Websites.

[16] <http://web.simmons.edu/~grabiner/comm244/weekthree/css-basic-properties.html>

[17] Michele E. Davis, Jon A. Phillips. (Copyright © 2007). Learning PHP and MySQL.

[18] Thomas Blom Hansen and Jason Lengstorf. (Copyright © 2014). PHP for Absolute Beginners.

[19] PHP Tutorial. <https://www.w3schools.com/php/default.asp>

[20] PHP Tutorial. <http://www.tizag.com/phpT/>

[21] PHP Tutor <https://www.phptpoint.com/php-tutorial/>



[22] LEARNING PHP AND MYSQL. <http://index-of.co.uk/Programming/Learning%20PHP%20and%20MySQL%20by%20Knowledge%20flow%20-%202015.pdf>.

[23] PHP tutorial <https://tutorials.supunkavinda.blog/php/introduction>

[24] PHP Tutorial. <https://www.tutorialrepublic.com/>

## Chapter 4

[1] <https://whatis.techtarget.com/definition/sensor>

[2] [https://en.wikipedia.org/wiki/Light-emitting\\_diode](https://en.wikipedia.org/wiki/Light-emitting_diode)

[3] <https://en.wikipedia.org/wiki/Webcam>

[4] <http://blog.sparkfuneducation.com/what-is-jumper-wire>

[5] <http://wiring.org.co/learning/tutorials/breadboard/>

[6] <https://www.techopedia.com/definition/4851/apache-web-server>

[7] <https://docs.remote.it/>