International Journal of Computer Science Trends and Technology (IJCST) – Volume 11 Issue 3, May-Jun 2023

RESEARCH ARTICLE OPEN ACCESS

Modeling and simulation of the automatic opening of a door by recognition Facial with deep learning using a Raspberry Pi nanocomputer and cloud services

Binele Abana Alphonse, Abou Loume Gautier, Djimeli Dtiabou Berline, Bavoua Kenfack Patrick Dany, Tonye Emmanuel

Enspy/Uyi

19 October 2022

ABSTRACT

The objective of this work is to simulate in Packet Tracer a video supervision system using connected object technology. This device, based on facial recognition, uses a Raspberry Pi nano computer to which a camera is connected and uses cloud services to store the faces of people with authorization to enter the company. This system, thanks to the motion detector, makes it possible to detect the presence of a person around the door of the company. Indeed, the motion detector lights up to signal a human presence, which triggers the activation of the camera which films the scene in real time. If after 6 seconds the detector has turned off then the siren is triggered because this means that the face has not been recognized. From this moment, an email containing a message and the image of the scene is sent to the administrator to report the presence of an intruder. The latter, through his control interface in his telephone, has the possibility of activating the equipment (opening/closing of the door, activation/stopping of the siren, etc.). If, on the other hand, after 6 seconds, the detector has not turned off then this means that the face has been recognized (this is manifested by the activation of the LED) then the door opens. The administrator receives an email containing the image of the scene in addition to the opening message.

Key words: Internet of Things (IoT), Raspberry Pi, Cloud Services, Deep Learning, Closed circuit television system (CCTV).

I. INTRODUCTION

Facial recognition is a means of identification that is the subject of much research. Among these works, the use of the Internet of Things (IoT), which relies on connected objects [1], has shown its effectiveness. For example, in ultra-modern systems, we use biometric facial recognition technology for physical access control to a door (automatic opening or closing).

In our previous works, on a practical level, we realized a facial recognition device for the authorization of the opening of a door based on deep learning. The models (Deep Learning, convolutional neural networks: CNN) embedded in a raspberry pi nanocomputer make it possible to automate a chain of actions, from detecting a person's movement, taking their photo, recognizing this person to authorize the opening of a door and finally the sending of notifications (SMS and email) to authorized persons [2].

This work was published in October 2022 in a scientific article with the title "Facial Recognition in the Opening of a Door using Deep Learning and a Cloud Service" [3].

Figure 1 below shows the architecture of this device.

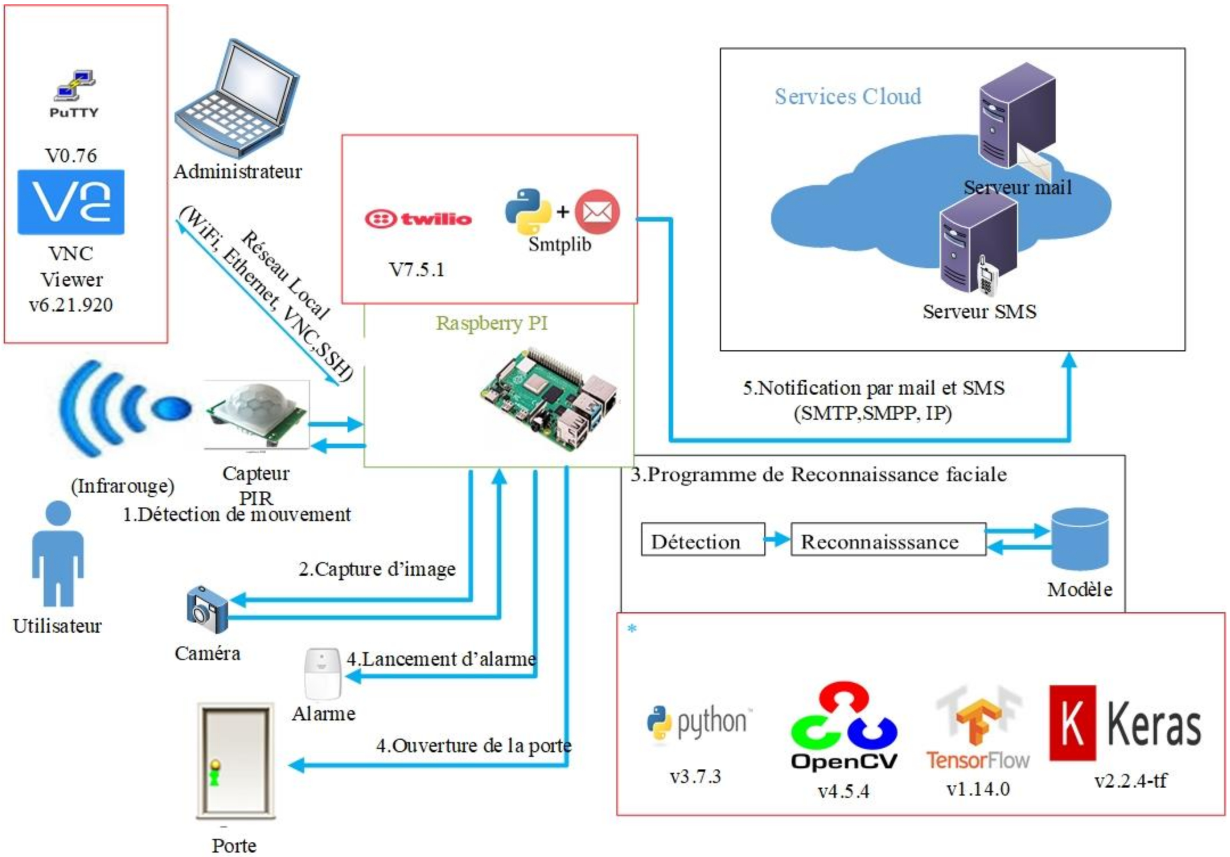

***Figure 1**: Architecture of the device [2] [3]*

In this new work, we want to simulate with Packet Tracer, for educational purposes, the automatic access control system to a door by recognition using Deep Learning and a Cloud notification service (SMS and mail) [4]. We will put into perspective aspects such as the comparison of our educational tool with our practical device on the one hand, and with other devices on the other hand.

II. METHODOLOGIE

2.1. CCTV system

Today, CCTV systems have become an essential part of our infrastructure. These systems play an essential role in our lives due to their enormous benefits, such as the security of public and private places. [5]

CCTV consists of placing cameras in an environment, keeping track of people observed and detecting suspicious behaviour by trained operators. CCTV systems are typically used to monitor high security areas. [5] As the number of cameras increases, this mission becomes very complex and sometimes impossible. It is for this reason that a particular aspect of artificial intelligence has emerged. This is the development of computer vision algorithms. These algorithms make it possible to process visual data and provide observations similar to those of a human being, making this system more intelligent. [5]

2.1.1. Technological Evolution

The technological evolution of video surveillance is correlated with the improvement of cameras. We have moved from 1GSS generation systems to 3GSS and 2GSS [6] [7]. Thus, three (03) types of innovation that punctuate the video surveillance revolution. It is:

- The first generation (1GSS 1960-1980) : all is analogue [8] [9] [10] ;
- The second generation (2GSS 1980-2000) : hybrid system [11] [12] ;
- The third generation (3GSS 2000-nowadays): all is digital [11].

2.1.2. New generation of video surveillance system: the Internet of Video Objects

Over the past decades, a large amount of data has been generated by multi-camera surveillance networks. [13]. this rapid evolution of camera-generated data poses many significant challenges to conventional video surveillance systems.

Faced with this evolution, a new generation of video surveillance systems called the Internet of Video Things (IoVT) is emerging to increase flexibility and address the challenges of conventional systems in terms of network, architecture, optimization, and real-time operation. [14].

The Internet of Video Things IoVT - (in French: L'Internet des objets vidéo IdOV -) network of distributed visual sensors or intelligent cameras, which are unambiguously uniquely identified, operating in an IoT (Internet of Things) environment. These cameras can interact and communicate with each other and/or with other IoT and human objects using information and communication technologies (ICT) for distributed processing, data exchange/sharing and increased system autonomy. [15]

The Internet of Video Things (IoVT) infrastructure is constituted of the following elements [5]:

- Smart cameras: These smart devices consist of cameras and microcontrollers to acquire, store, process information, and even communicate with each other and/or with other objects in the IoVT environment.
- The network: The communication networks that connect the smart cameras to each other or to others and transmit the data collected through various recent technologies (such as BLE, Wifi, ZigBee, etc.).
- Applications: The link between the IoVT interface and end users (people or systems), this is a layer for processing, storing and analysing the large volumes of data received from smart cameras.
- Computing paradigms: Modern IoT paradigms such as cloud computing, which allows large amounts of data to be stored and analysed thanks to their significant storage capacity and high computing power. Fog computing which effectively manages and controls a set of smart cameras located in its geographical area.

The Internet of Video Things (IoVT) infrastructure is illustrated in the figure below.

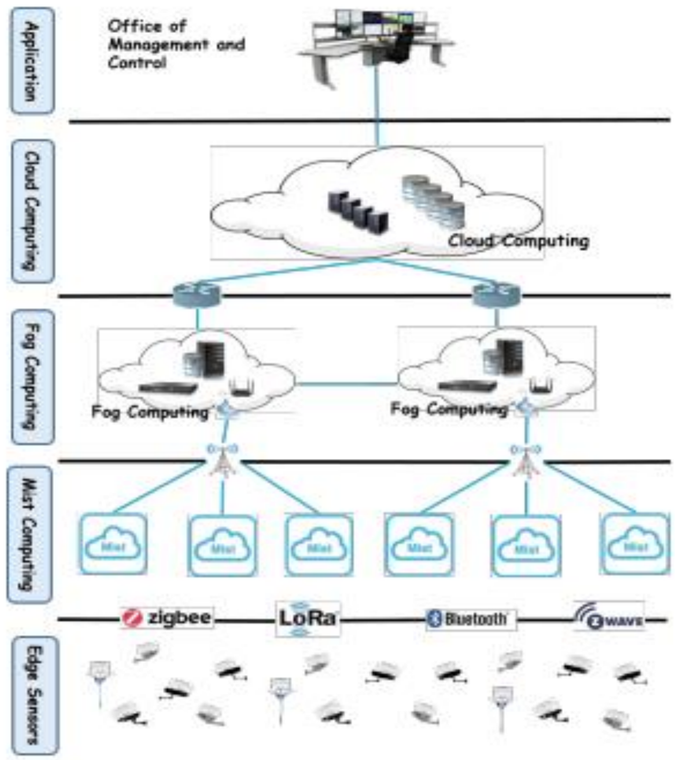

Figure 2: *Infrastructure of the IoVT [15]*

Model of Artificial Intelligence: The Deep Learning

Deep Learning is a set of machine learning methods attempting to model data with a high level of abstraction through articulated architectures of different nonlinear transformations. These techniques have fostered rapid advances in the fields of sound or visual signal analysis and in particular facial recognition, voice recognition, automated language processing, etc. [4]

Deep Learning techniques constitute a class of machine learning algorithm whose characteristics are as follows:

- They use different layers of nonlinear processing unit for feature extraction and transformation. Each layer takes as input the output of the previous one. Algorithms can be supervised or unsupervised and their applications include pattern recognition or statistical classification.
- They work with learning at several levels of detail or data representations. Through the different layers we pass from low level parameters to higher level parameters.
- These different levels correspond to different levels of data abstraction.

- This new field of study aims to advance further towards artificial intelligence capabilities. Its architectures now make it possible to give meaning to data in the form of an image, sound or text. [4]
- A Deep Learning system is based on artificial neural networks, which consists of a set of hidden layers, the word deep (deep learning) comes from the large number of layers and also neurons. This type of algorithm requires significant computing capacity, for this it is necessary to use GPUs in order to be able to perform complex operations in a short time.

In the literature, Deep Learning models for face detection and recognition are a combination of different architectures that have been developed for other uses (most often for image classification and pattern recognition) but which are equally valid for faces. Thus, we distinguish for example some approaches such as:

- Facenet: a neural network made up of 22 deep layers [17].
- GoogleNet : a neural network made up of 22 deep layers.[18]
- DeepFace: a neural network made up of 9 deep layers. [19]
- Resnet50: a neural network made up of 50 deep layers: 49 convolution layers and 1 layer fully connected. [20]
- VGG-19 : a neural network made up of 9 deep layers: 8 convolution layers and 3 layers fully connected.[21]
- VGG-16 (VGG Face-16): a neural network made up of 19 deep layers: 13 convolution layers and 3 fully connected layers, 5 pooling layers (sub-sampling), a classification layer that uses the SoftMax function (SoftMax layer). It takes as input an image ("feature map") of size 224 x 224 pixels and its output is a classifier of size 1000 (vector of facial features). [22]
- VGG-19 : is a neural network made up of 19 deep layers: 16 convolutional layers and 3 fully connected layers.[21]

In our previous work [3] [2], we used for face recognition a VGG-16 convolutional neural network pre-trained using the Transfer Learning technique. In this case, we have replaced the classification layer with a classifier of size 7. It consists of 7 classes (initially 1000) which represent the total number of individuals present in our dataset. [2] The three-dimensional representation of the architecture of VGG-16 used in our experimental device is given by figure 3. It presents the extraction of the characteristics of the photo (in colour: three channels) taken of the face of size 224×224 pixels.

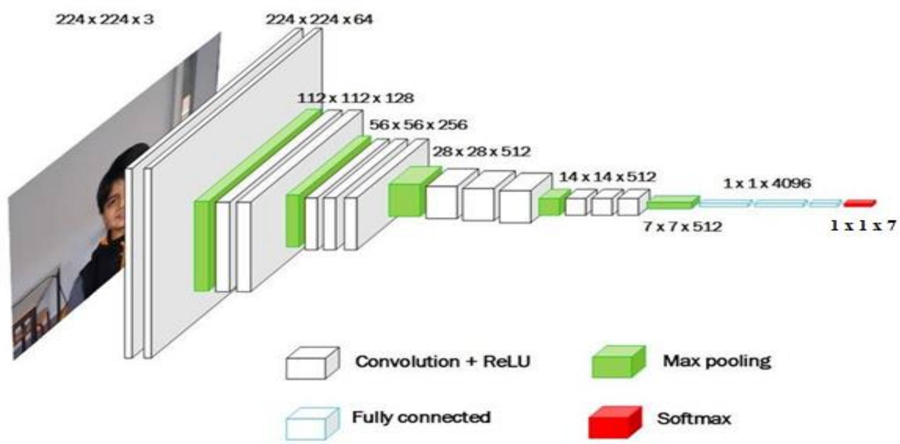

Figure 3*: 3D representation of the architecture of the neural network used [3]*

Figure 4 presents an overview of the two-dimensional architecture of the model used.

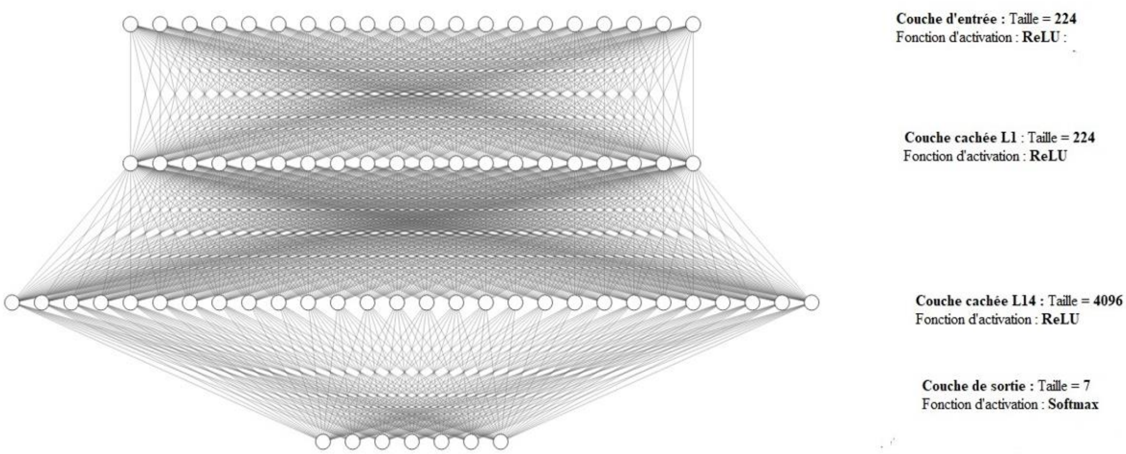

Figure 4*: Overview of the 2D architecture of the neural network used [3]*

Layer 1 is able to extract features of lower level of abstraction than Layer 2, while Layer 3 has higher quality. From these characteristics the system is able to recognize faces with a small error rate [23].

The source codes of the different algorithms used in our previous work [3]: facial recognition (model training, face detection and identification, real-time model testing), SMS and email notifications and door unlocking/closing are presented in appendix B.

2.2. Modeling of the system

2.2.1. Architecture of the system

The architecture of our tool is shown in the figure below.

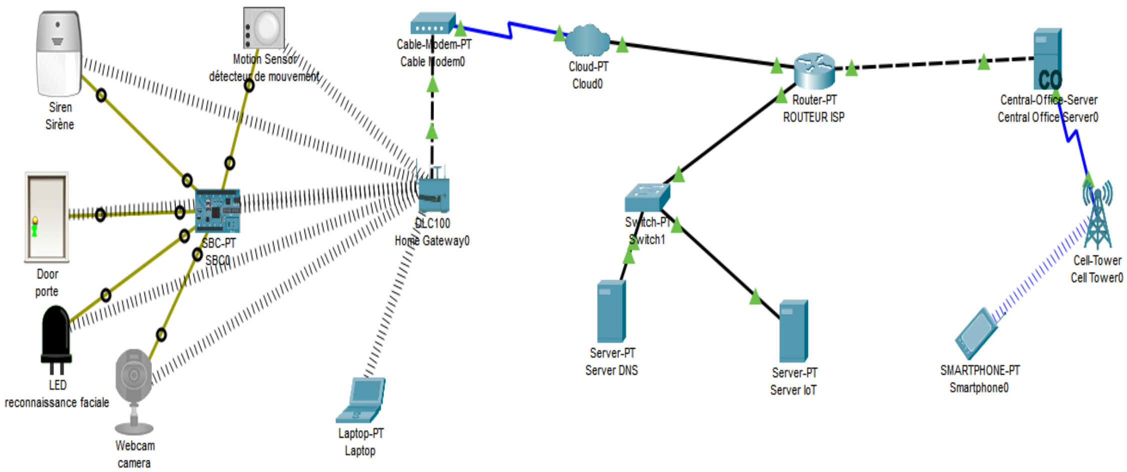

Figure 5*: Architecture of the tool [4]*

When a person arrives, there is:

(1) Detection of the movement of the person using the motion detector;
(2) Shooting of the face of the person who wishes the door to be opened using a connected camera. This image is transmitted to the Raspberry Pi nano computer;
Automatic door opening upon face recognition;
(3) Notification by Email through servers hosted in the Cloud.
(4) Alert in case of non-recognition of the face.

A phone connected to the same local network (WiFi, Ethernet) as the Raspberry Pi, allows you to administer, monitor, control or maintain the tool remotely.

The scenario can be summarized as follows:

- **First case: face detected and recognized**. When the face is recognized by the Raspberry Pi (this materializes by the activation of the LED), the door opens. An email is sent to the local administrator, using Cloud services to notify of the presence of an individual. The email contains a message indicating either the opening of the door.
- **Second case: face detected and not recognized. W**hen the face is not recognized, the Raspberry PI sends an email to the owner (administrator) of the infrastructure to report the presence of an unknown person. The email only contains a message indicating the presence of an unknown person. The administrator of the device also has the possibility of authorizing the opening of the door to this one or even recording it.

2.2.2. Organizational chart of the methodological steps

The scientific approach adopted for the realization of our tool is illustrated in Figure 5.

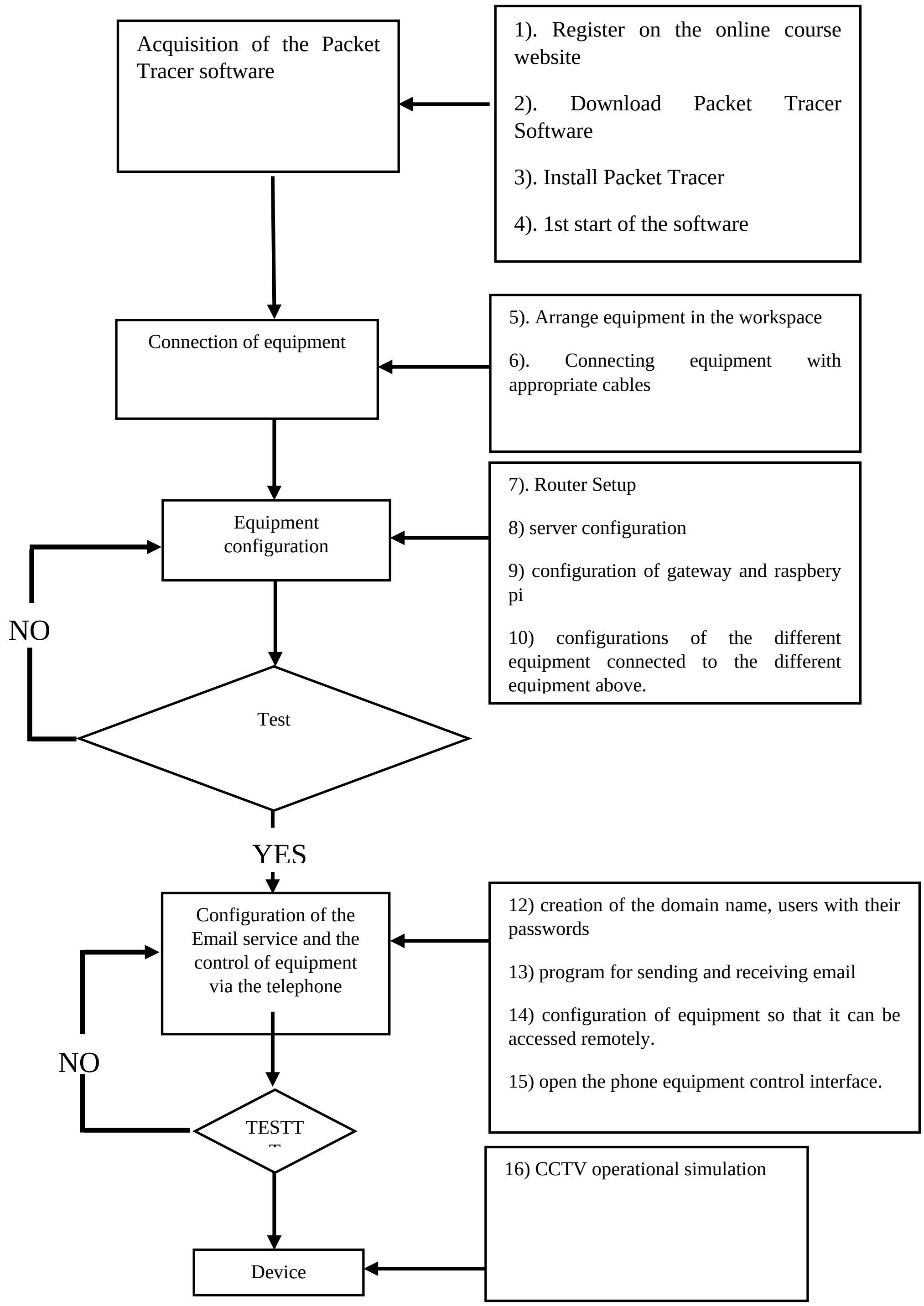

***Figure 6**: Organizational chart of the methodological steps [4]*

III. RESULTS

4.2.1. The router

We did the configuration on the router as following:

The first configuration was the one for the port GigabiEthernet 0 /7

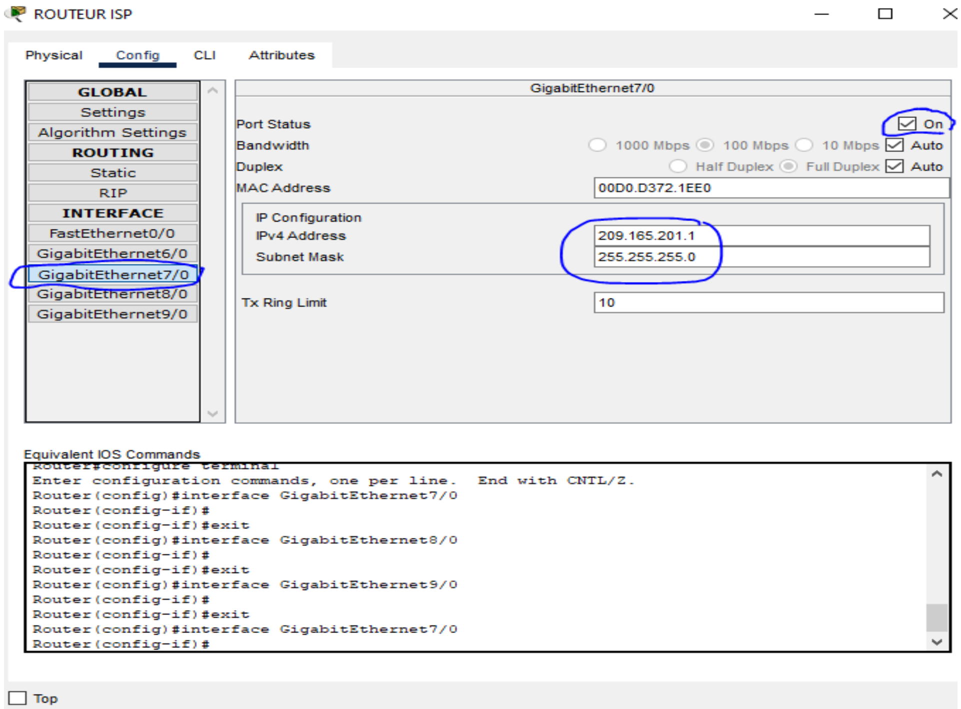

Figure 7: configuration of the port GigabutEthernet 0/7

Subsequently, port 0/8 configuration followed

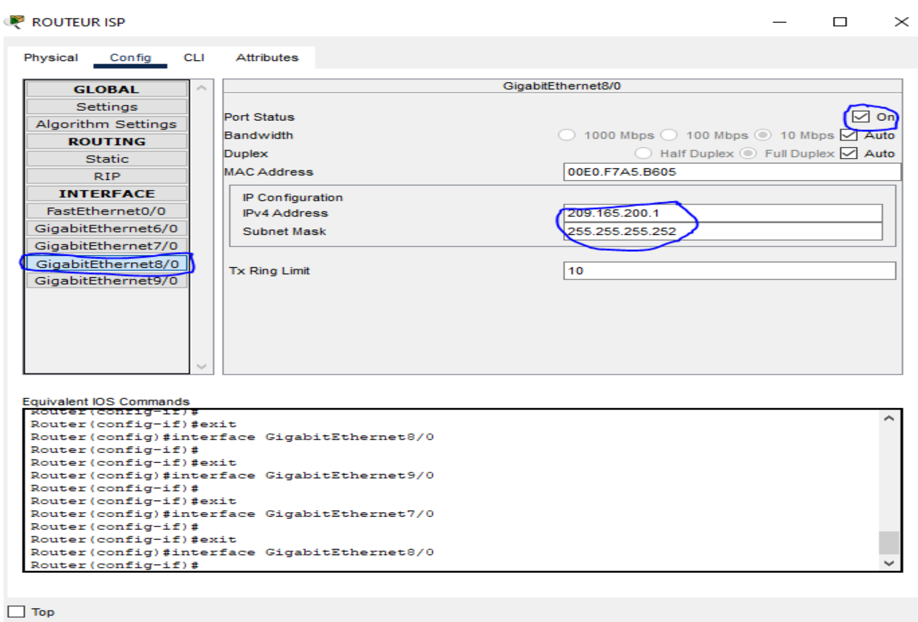

Figure 8: configuration du port gigabitEthernet 0/8

Finally, the configuration of the third port is done as follows:

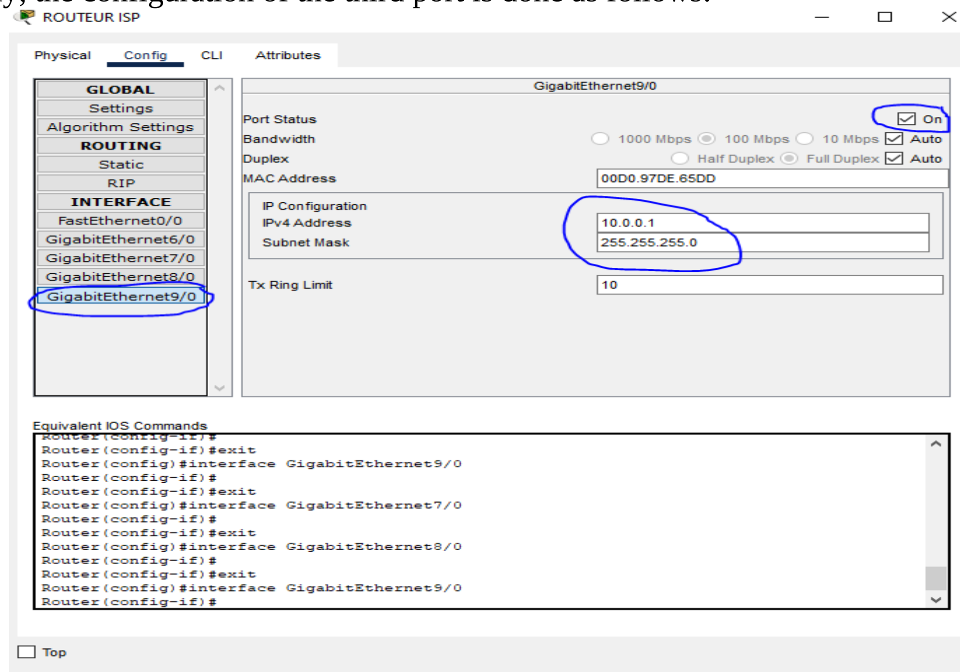

Figure 9: configuration of the port gigabitEthernet 0/9

This phase being finished, we moved on to the configuration of the routes:

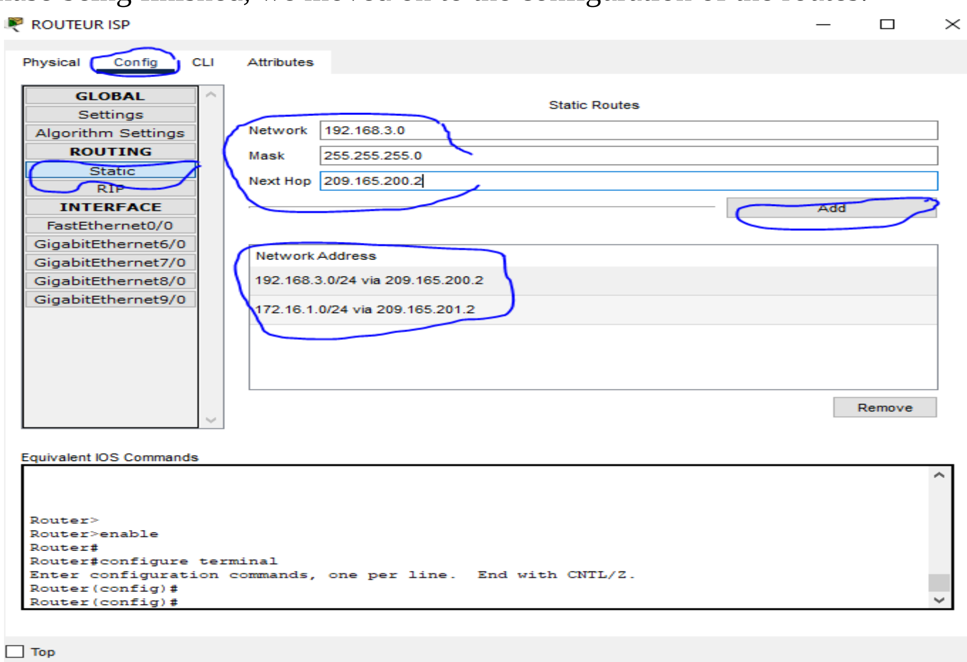

Figure 10: configuration of routes

Finally we configured the DHCP service on the router via the CLI tab. To do this, we typed the following commands: enable

```
configure terminal
ip dhcp pool isp
network 10.0.0.0 255.255.255.0
default-router 10.0.0.1
dns-server 10.0.0.10
exit
exit
ip dhcp centralofficeserver
ip dhcp pool centralofficeserver
network 209.165.201.0 255.255.255.0
default-router 209.165.201.1
dns-server 10.0.0.10
exit
exit
copy run start
sh run
```

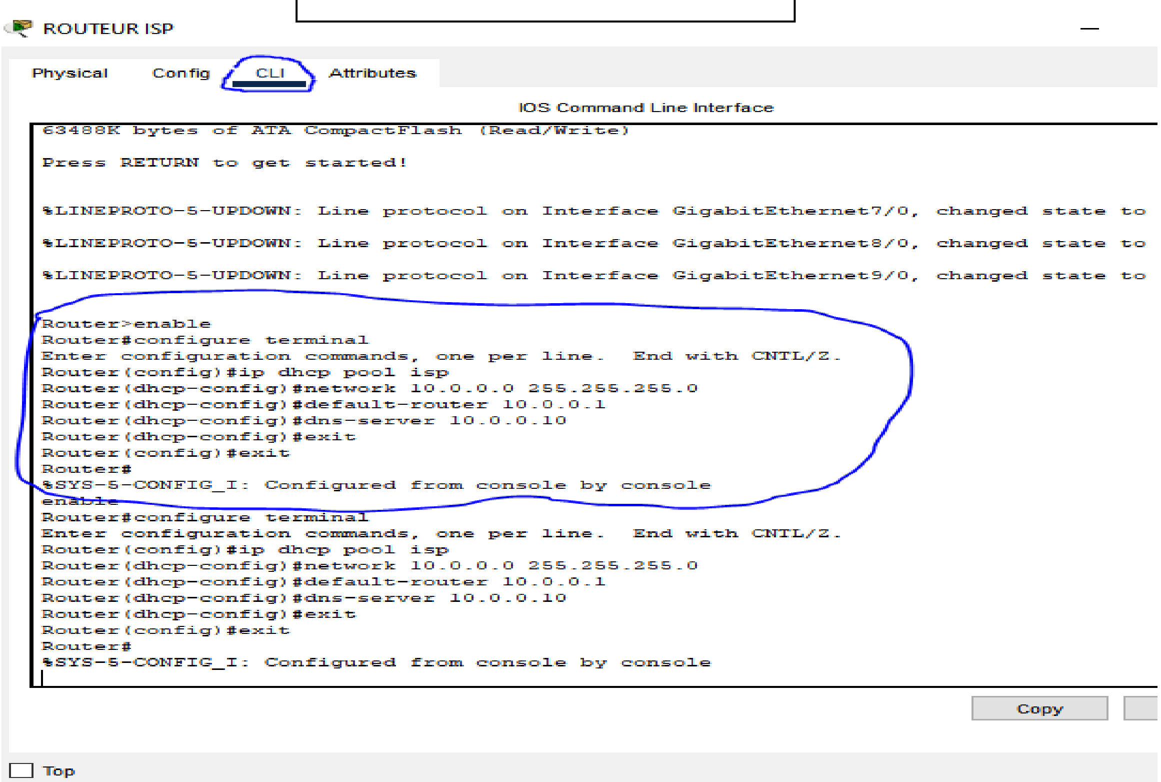

Figure 11: configuration of the DHCP in the global configuration mode

This phase shows the end of the configuration of the router.

4.2.2. Configuration of the central server

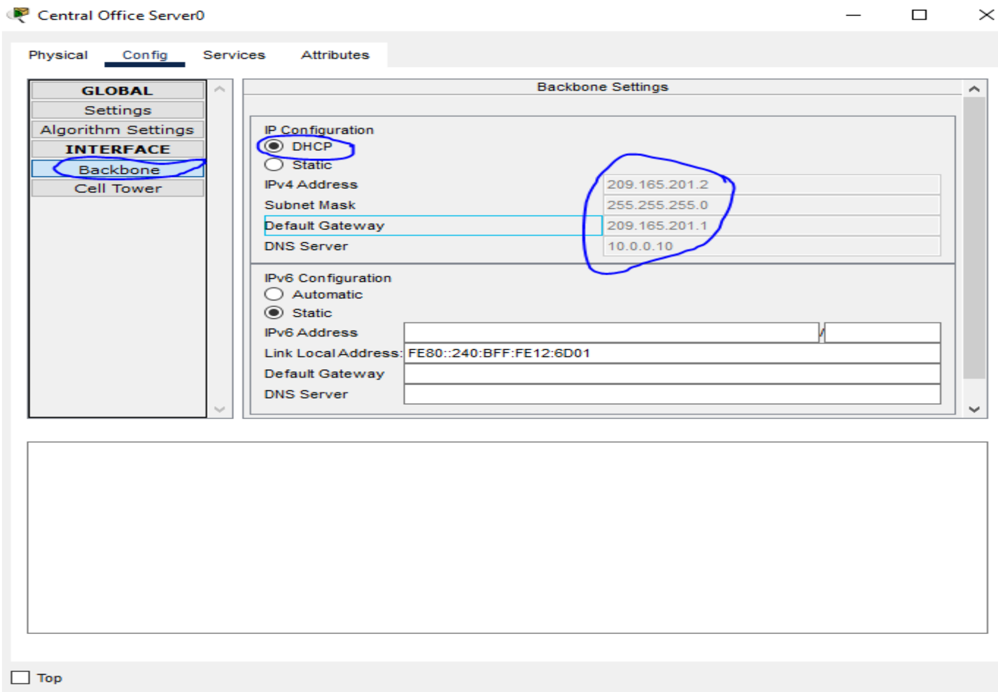

Figure 12: configuration of the central server

4.2.3. Configuration of the Cloud

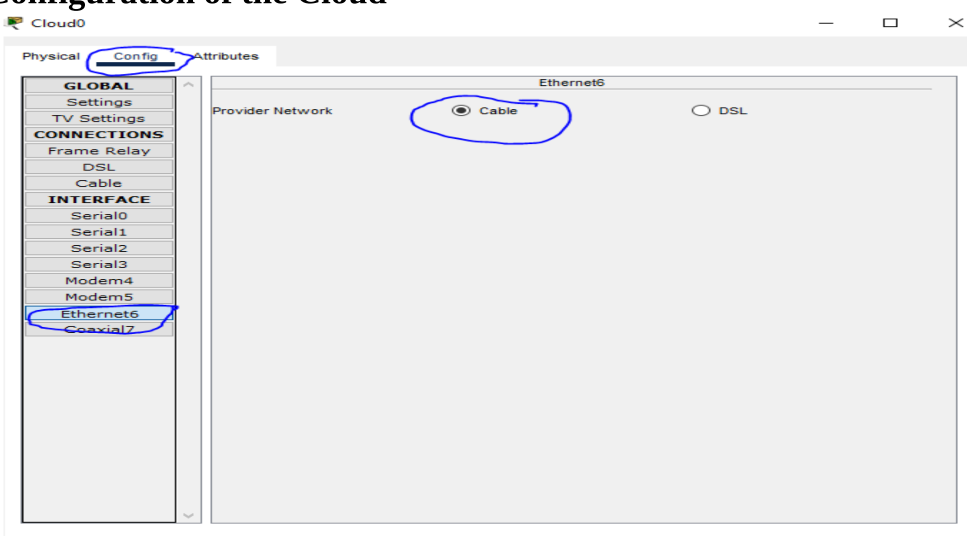

Figure 1: configuration of the Ethernet6 interface

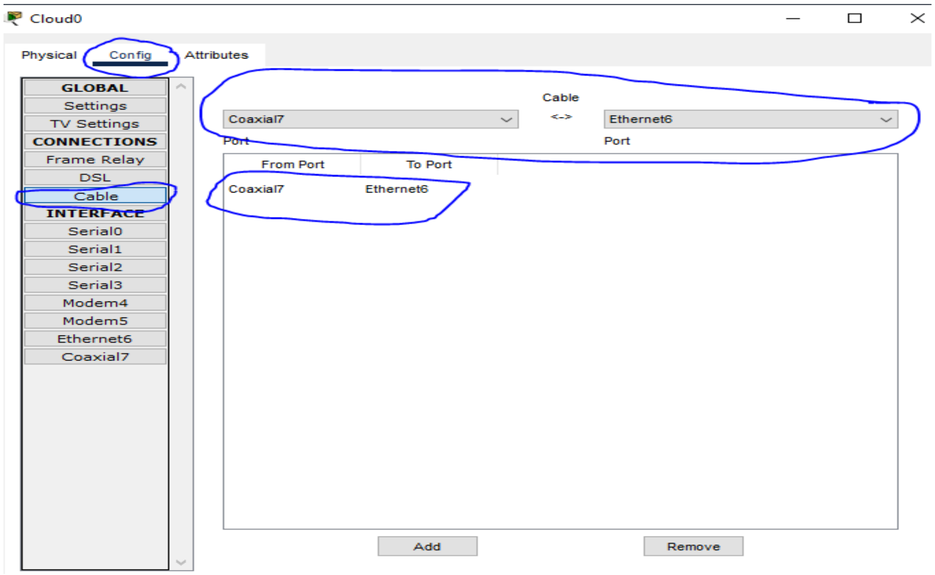

Figure 13: configuration of the type of connection

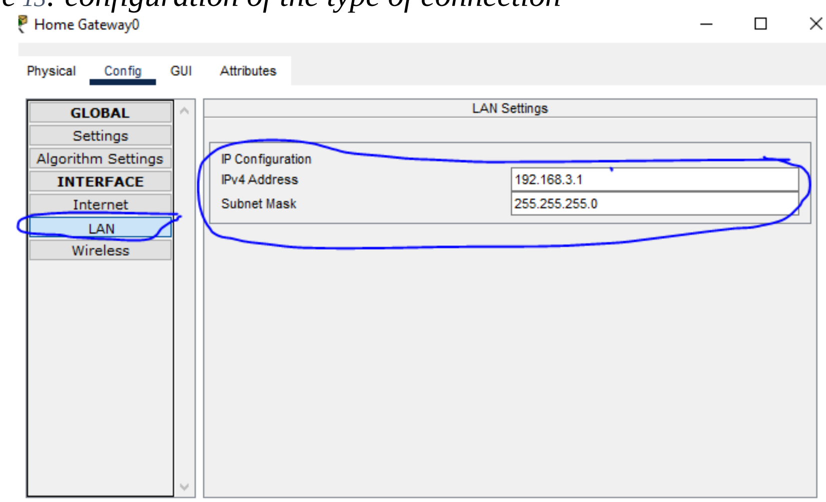

Figure 14: gateway (a)

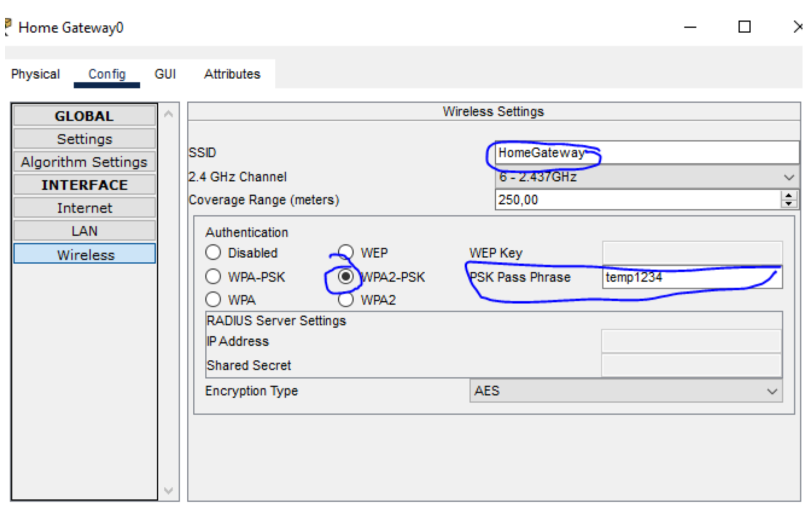

Figure 15: configuration of the gateway (b)

4.2.4. Configuration of the camera

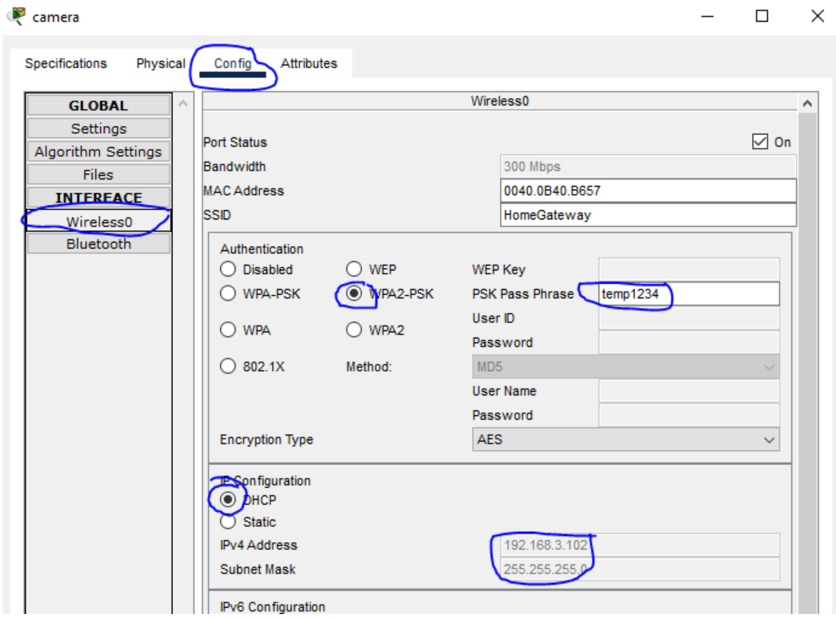

Figure 16: configuration of the connection of the camera to the gateway as well as the dynamic allocation of the IP address

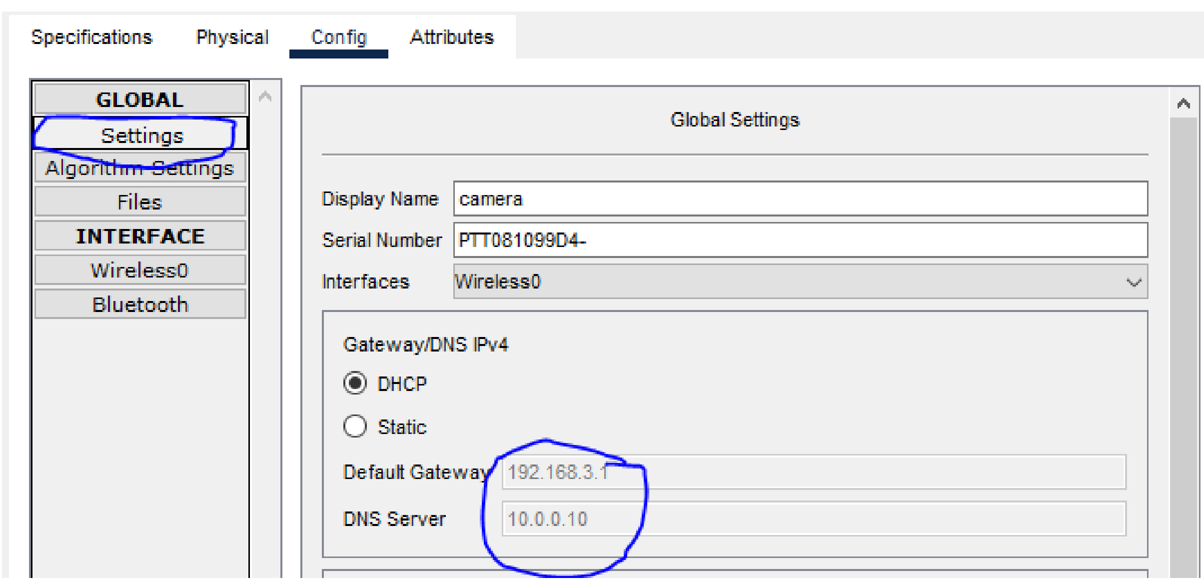

Figure 2: DHCP assignment of gateway and DNS server addresses

4.2.4. Configuration of the door

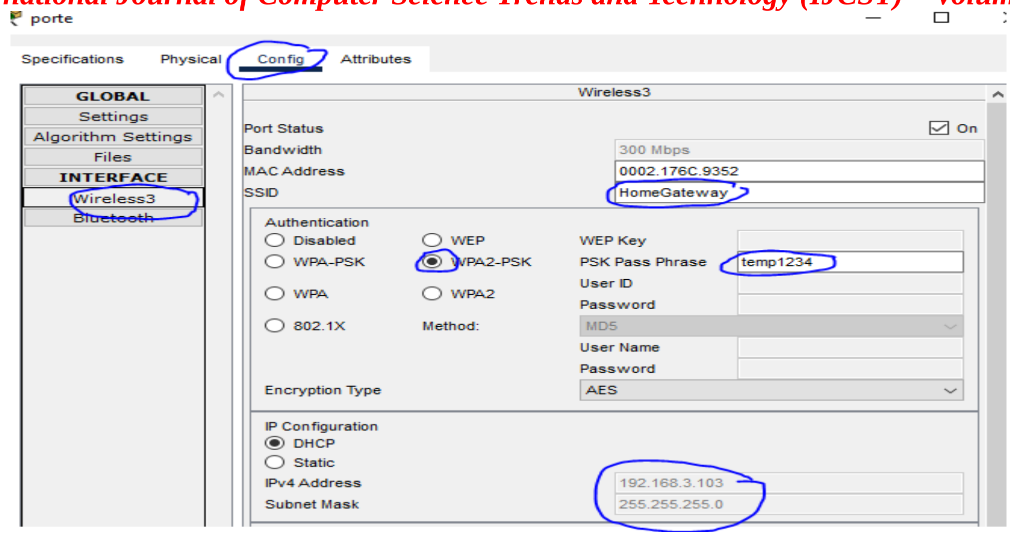

Figure 3: connecting the door of the door to the gateway and assigning the IP address

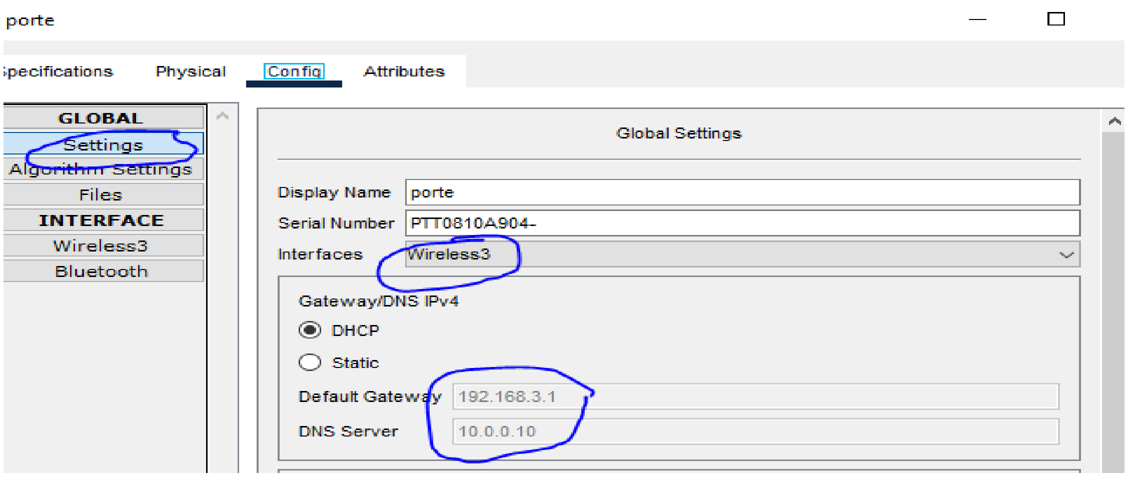

Figure 4 : dynamic address assignment to gateway and DNS server

4 .2 .5. Configuration of the alarm

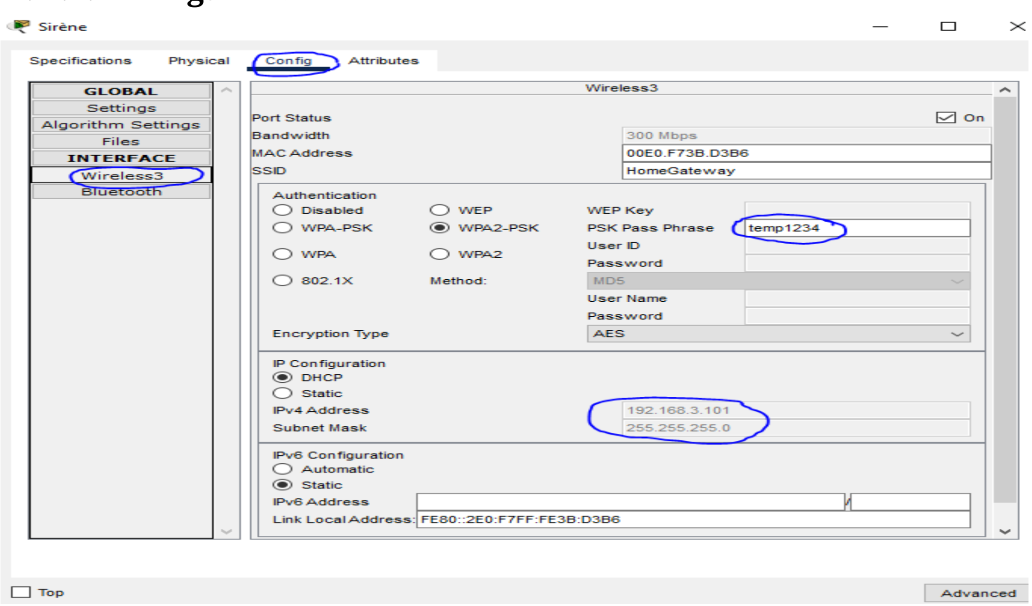

Figure 5: connection of the alarm to the gateway and assignment of the IP address

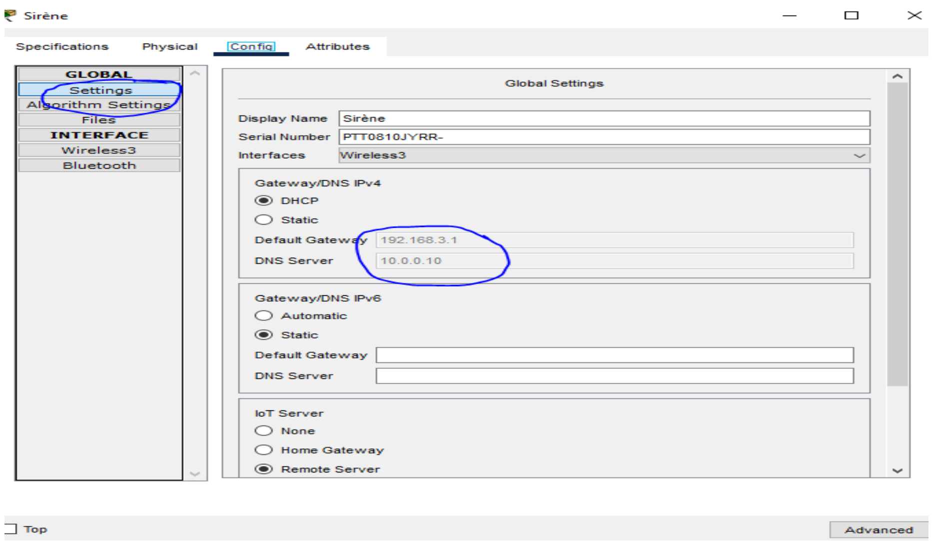

Figure 6: dynamic IP address assignment to gateway and DNS server

4.2.6. Configuration of the motion sensor

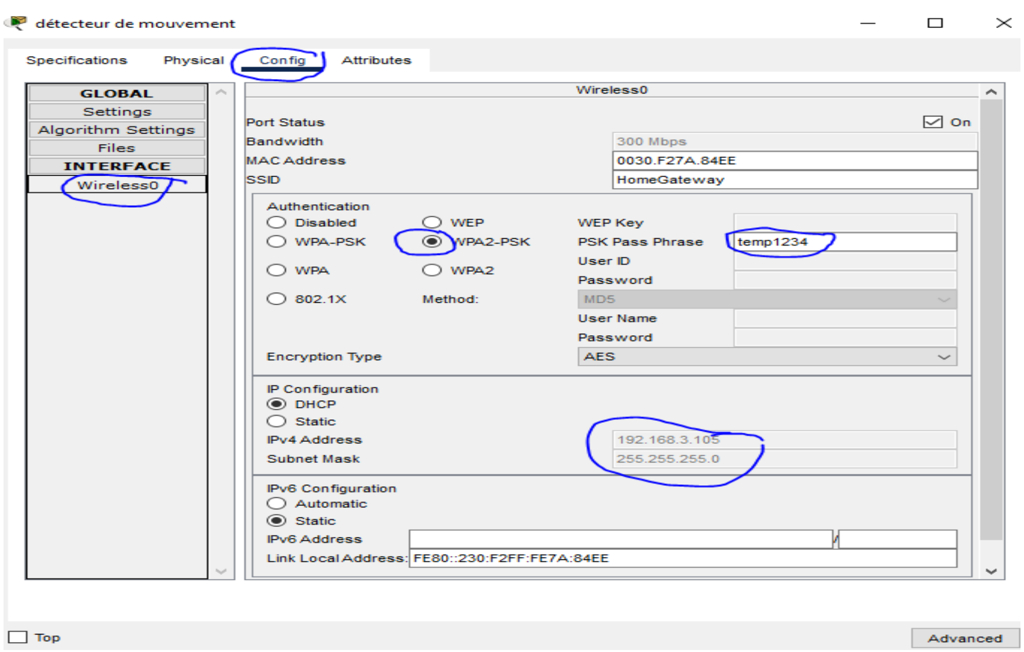

Figure 7: gateway connection and dynamic IP address assignment

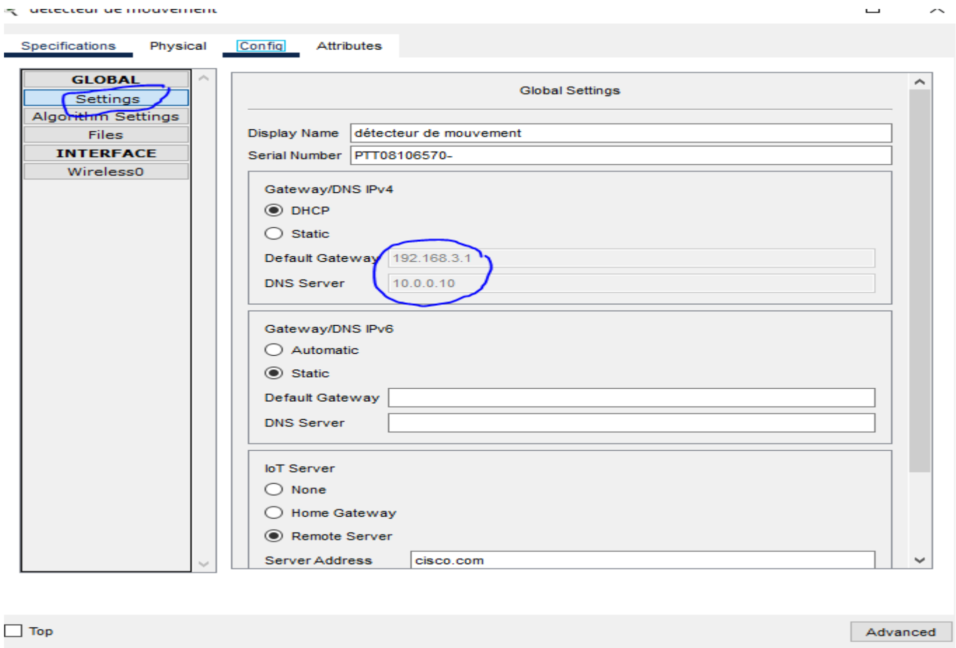

Figure 8: assignment of IP address to the gateway and the motion detector

2.1. Configuration of the DNS server

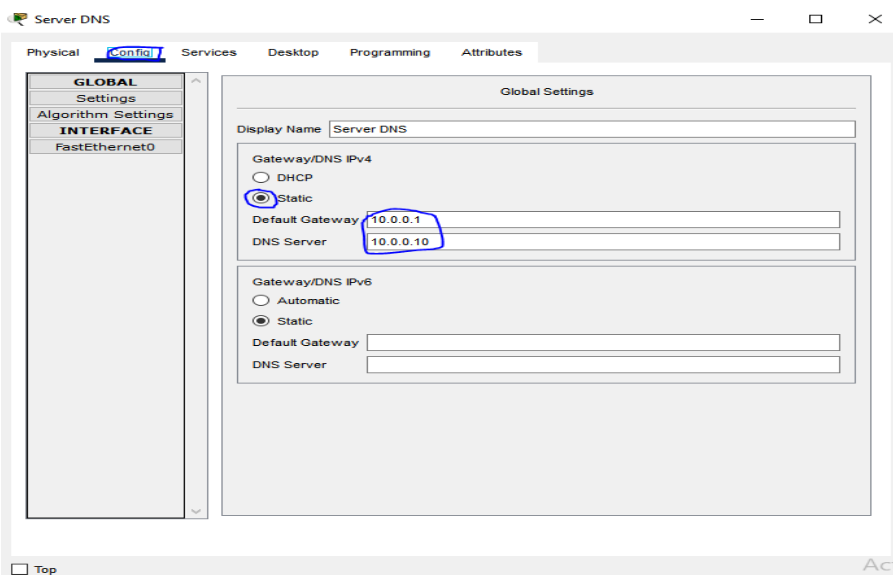

Figure 9: definition of static server addresses

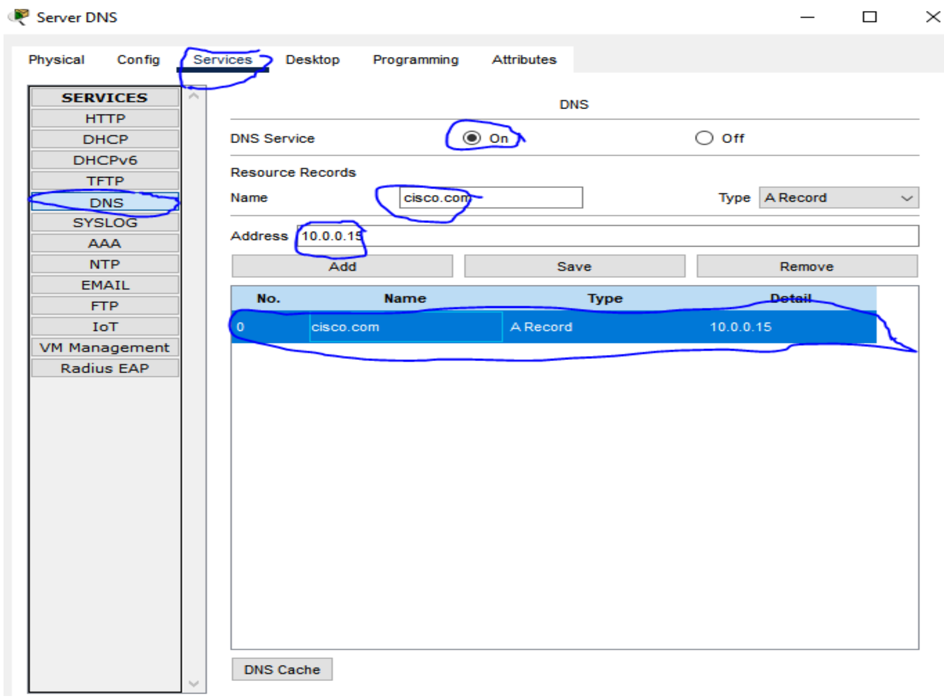

Figure 10: domain name definition

2.2. Configuration of the IoT server

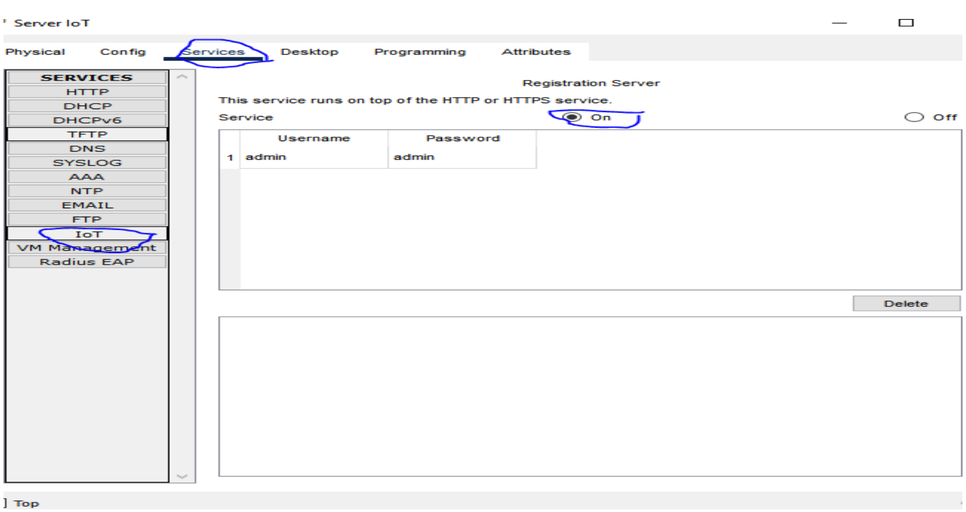

Figure 11: Activation of the IoT server

2.2.1. Creation of a user

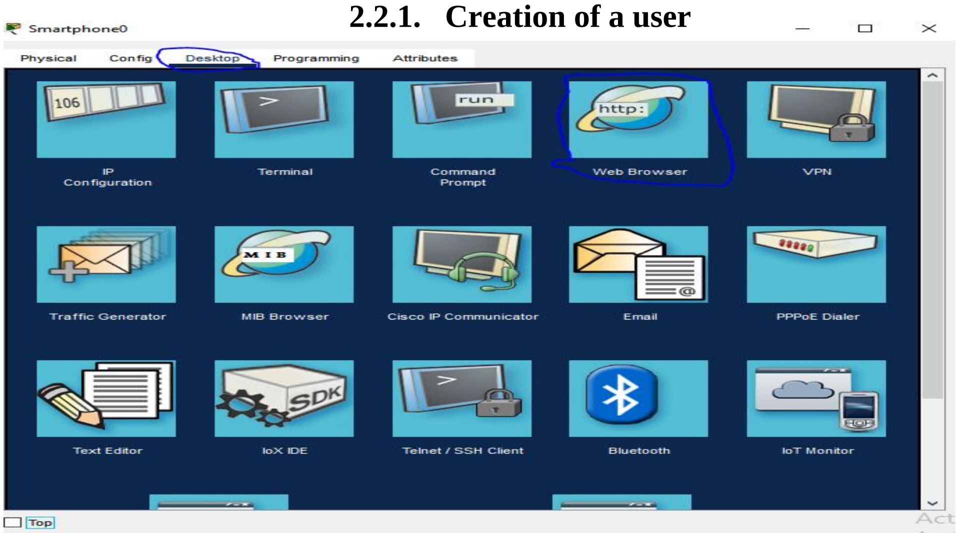

Figure 12: Opening of the web browser interface

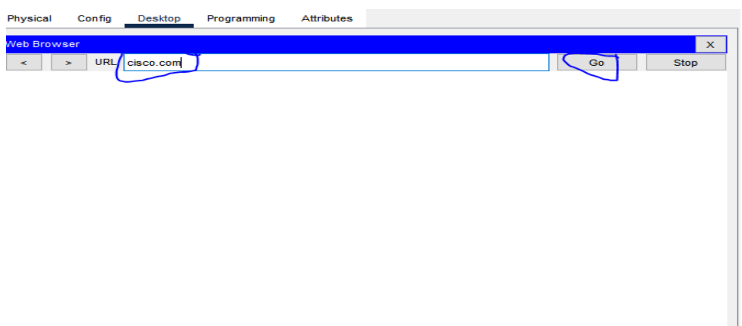

Figure 13: address of the target server

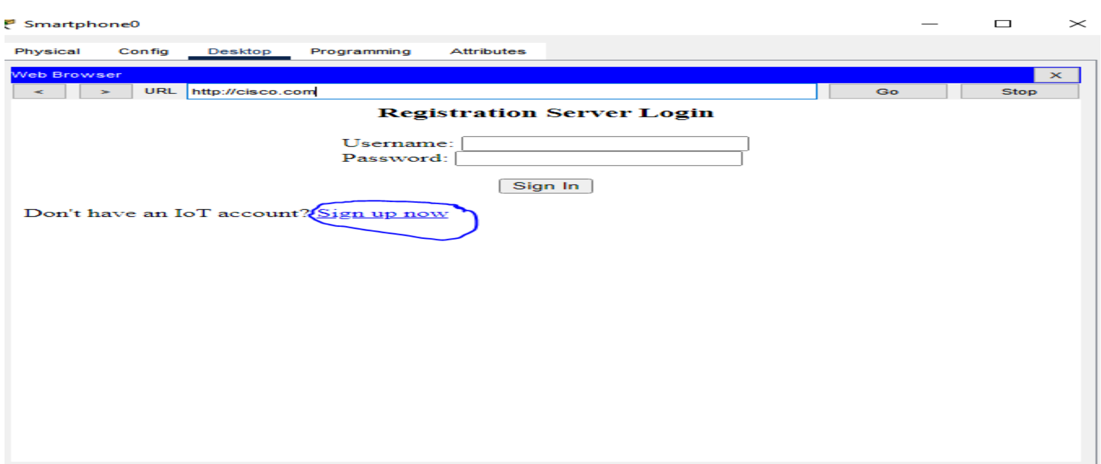

Figure 14: create a new user

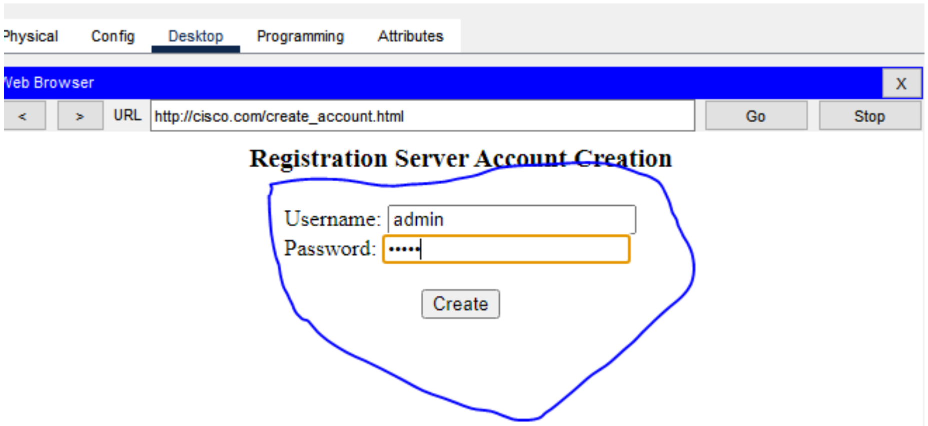

Figure 15: creation of a user

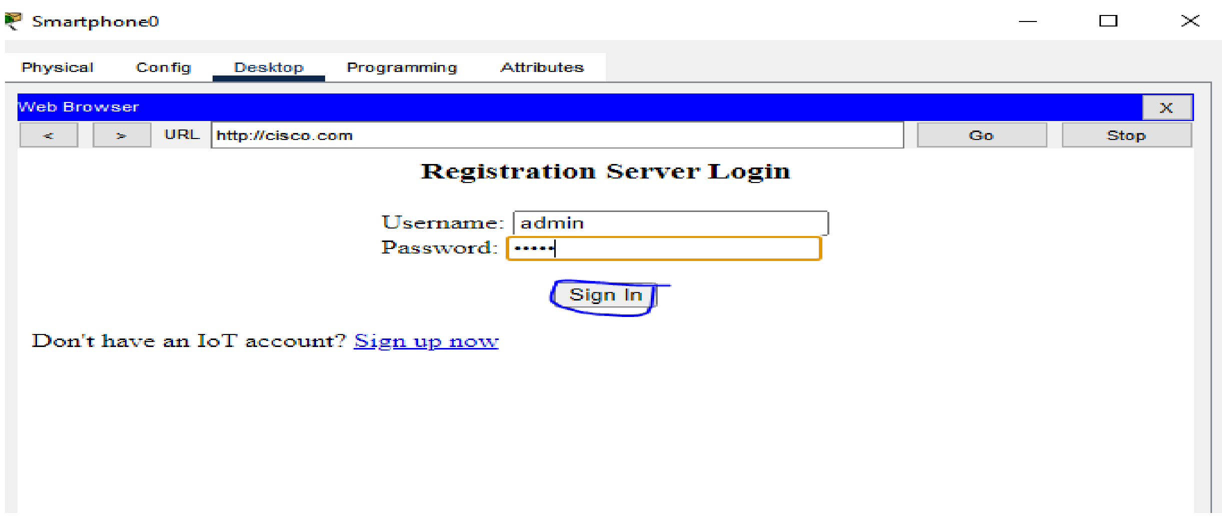

Figure 16: coordinate of the previously created user

Figure 17: no device connected yet

2.2.2. The administrator machine

Here, you must first connect the machine to the gateway as shown in Figure 52, then activate the DHCP service on the machine.

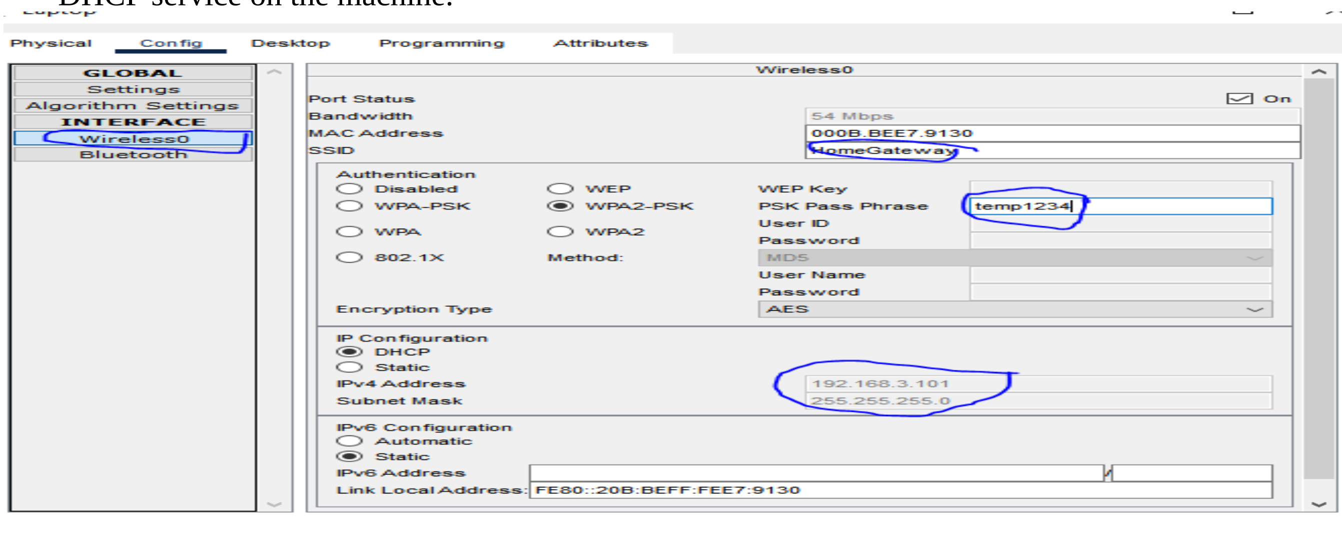

Figure 18: configuration of the administrator machine

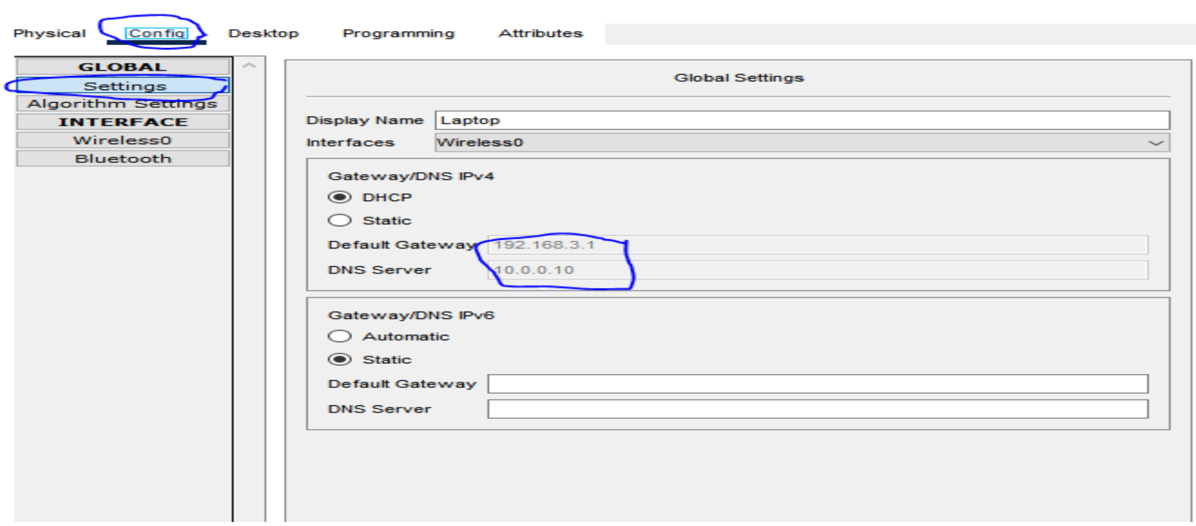

Figure 19: dynamic addressing

2.2.3. The gateway

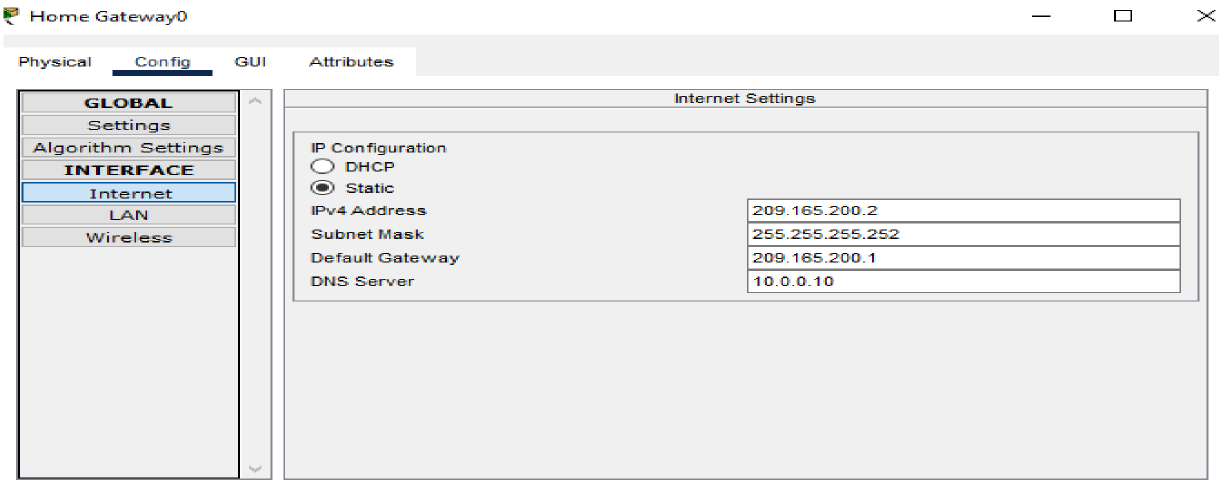

Figure 20: configuration of the internet interface of the gateway

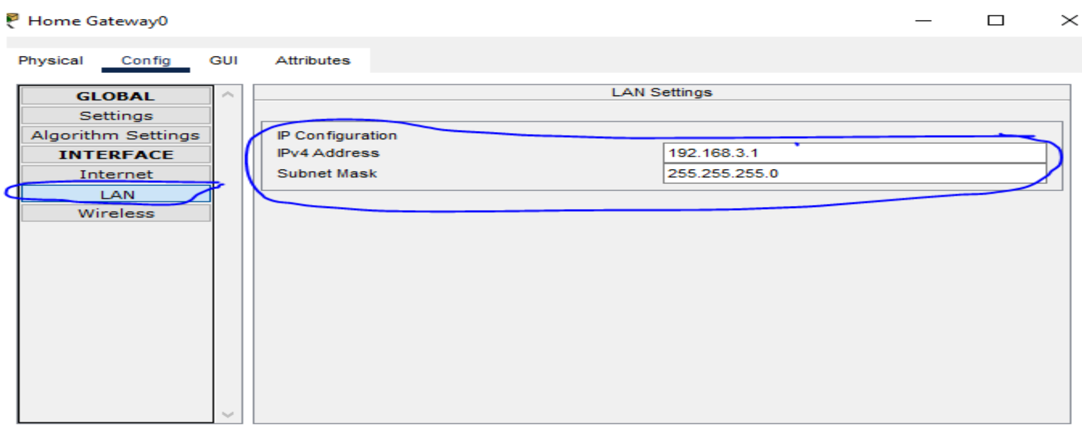

Figure 21: configuration of the LAN i, nterface of the gateway

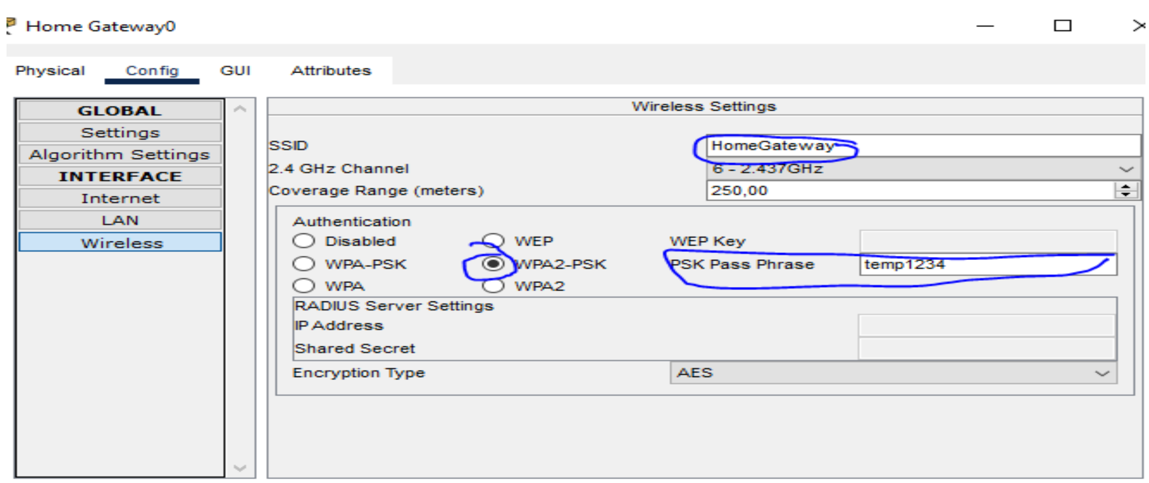

Figure 22: configuration of the wireless interface

2.2.4. Raspberry pi

To connect the raspberry to the gateway, simply click (left click) on the raspberry, go to "config", then to "wireless3", set the gateway password.

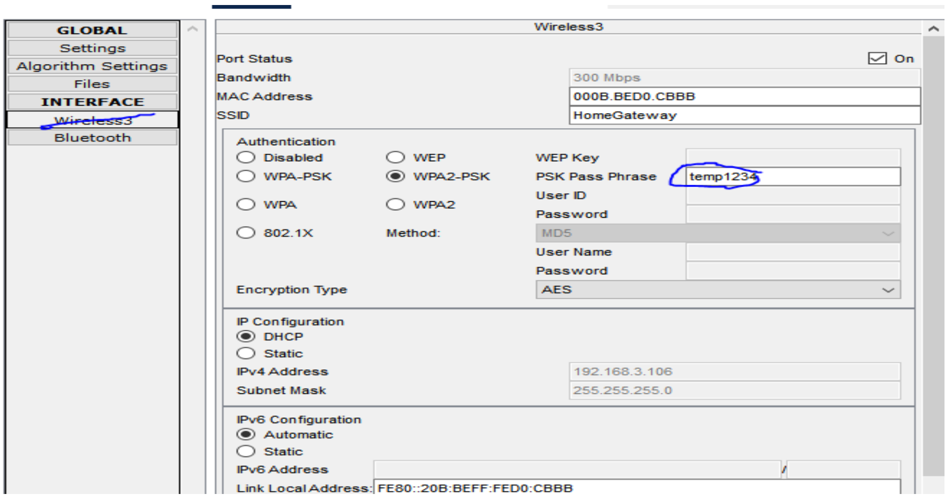

Figure 23: configuration of the raspberry pi wireless interface and dynamic IP address assignment

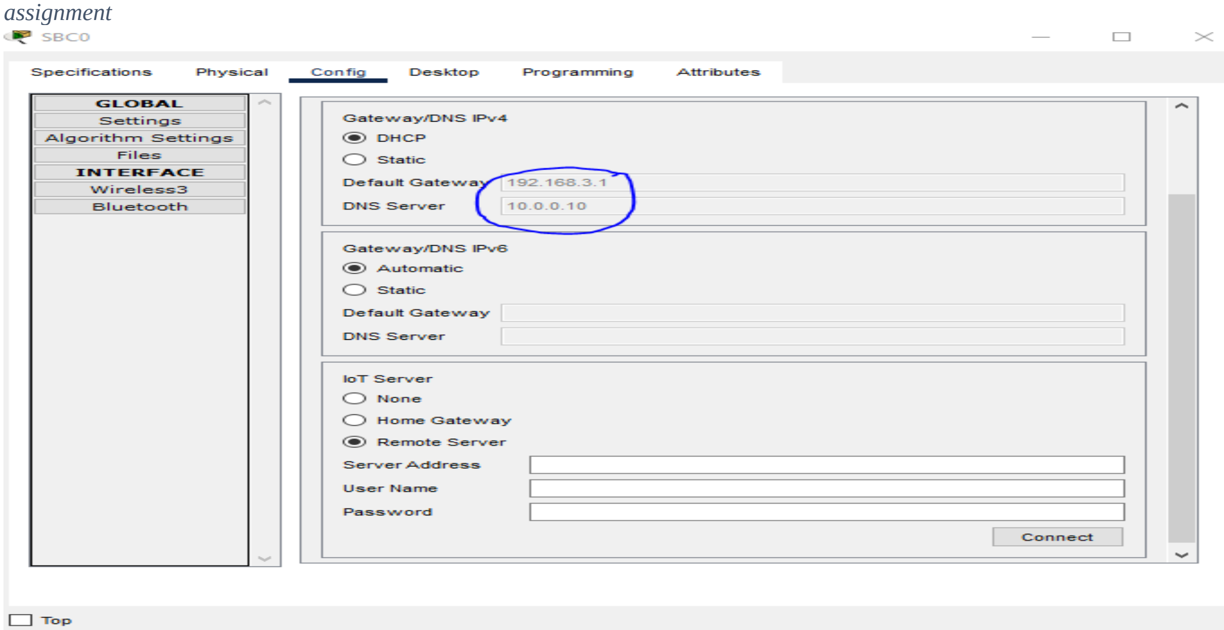

Figure 24: dynamic ip address assignment to raspberry default gateway and dns server

2.3. Configuration for the reception of emails

2.3.1. Configuration of the Raspberry pi

To configure the raspberry so that it can receive messages, you must:

-click on the raspberry

-choose “programming”

-define the name of the program and choose the language (we chose "python" later "email python"

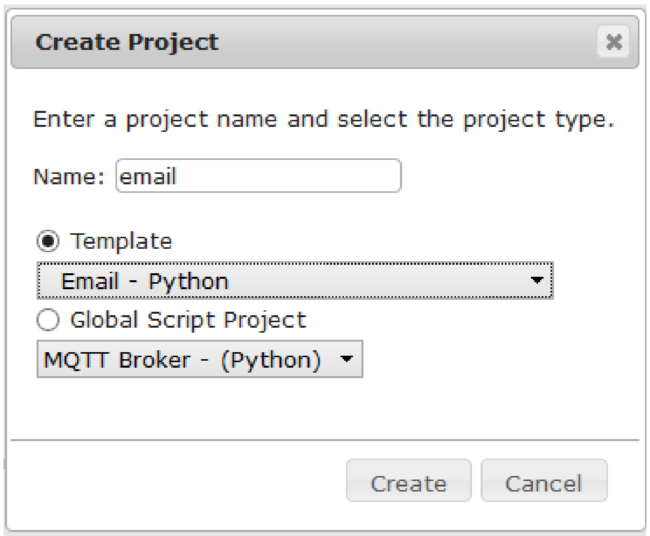

Figure 25: Creation of 'email

Finally, click on “create”.

After that we go to the interface and click on “main”. We will therefore have a basic program for sending emails that we are going to modify.

2.3.2. Configuration of the server

After doing this work, we return to the server for a new configuration

For this, we will follow the following procedure:

- click on the server,

- choose "email"

-define users and passwords. (We have defined 3 users so one sender and two recipients).

All this after the email services are activated and also that the domain name is correctly defined.

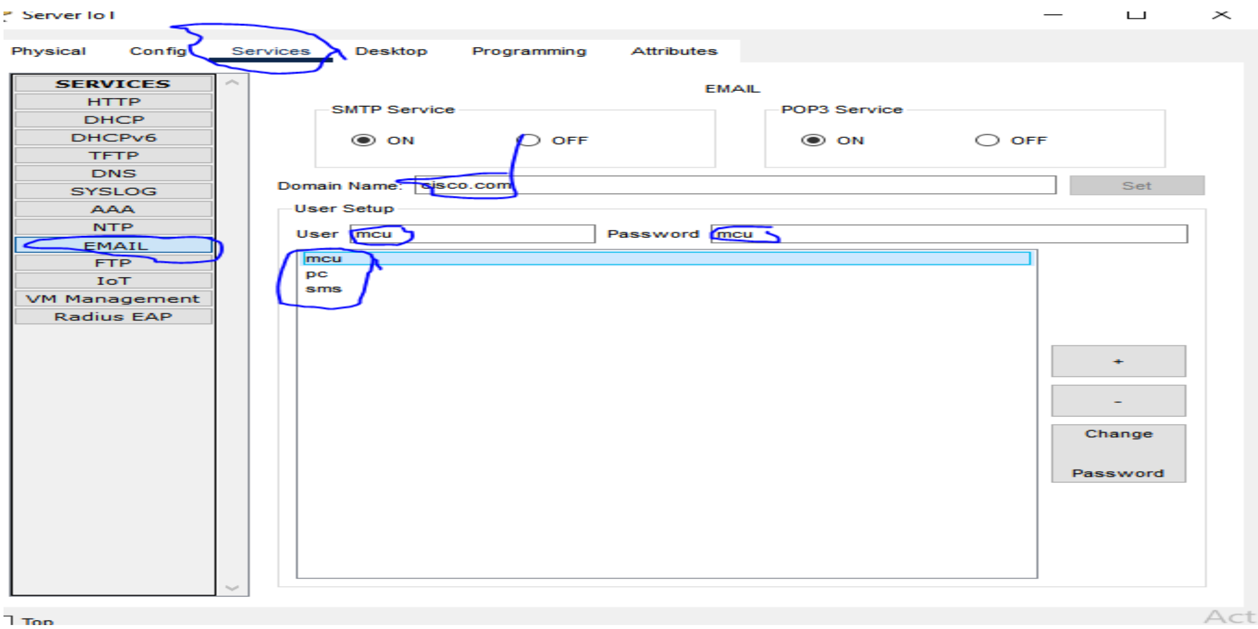

Figure 26: email service (creation of logins and passwords)

2.3.3. Configuration of the computer

Click on pc, choose "Desktop" then "email" and "email configuration".

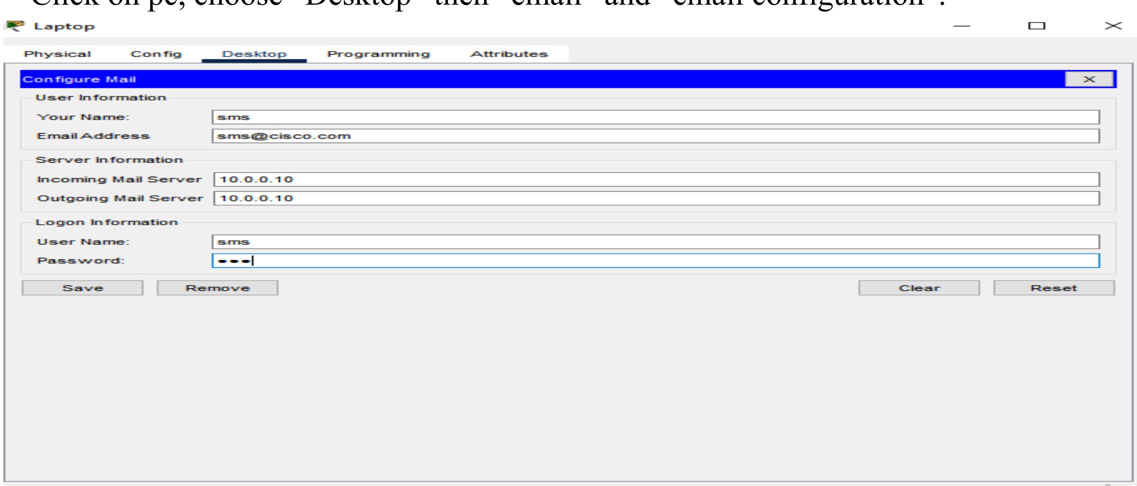

Figure 27: Configuration of email

RESULTS

After configuring the equipment and writing the algorithm, we will compile our program by clicking on “run”.

After configuring our system equipment (router, central server, cloud server, IoT server, DNS server, gateway, door, siren, motion detector, Raspberry pi, administrator machine, user) [4], and l writing the algorithm, we can then run our program.

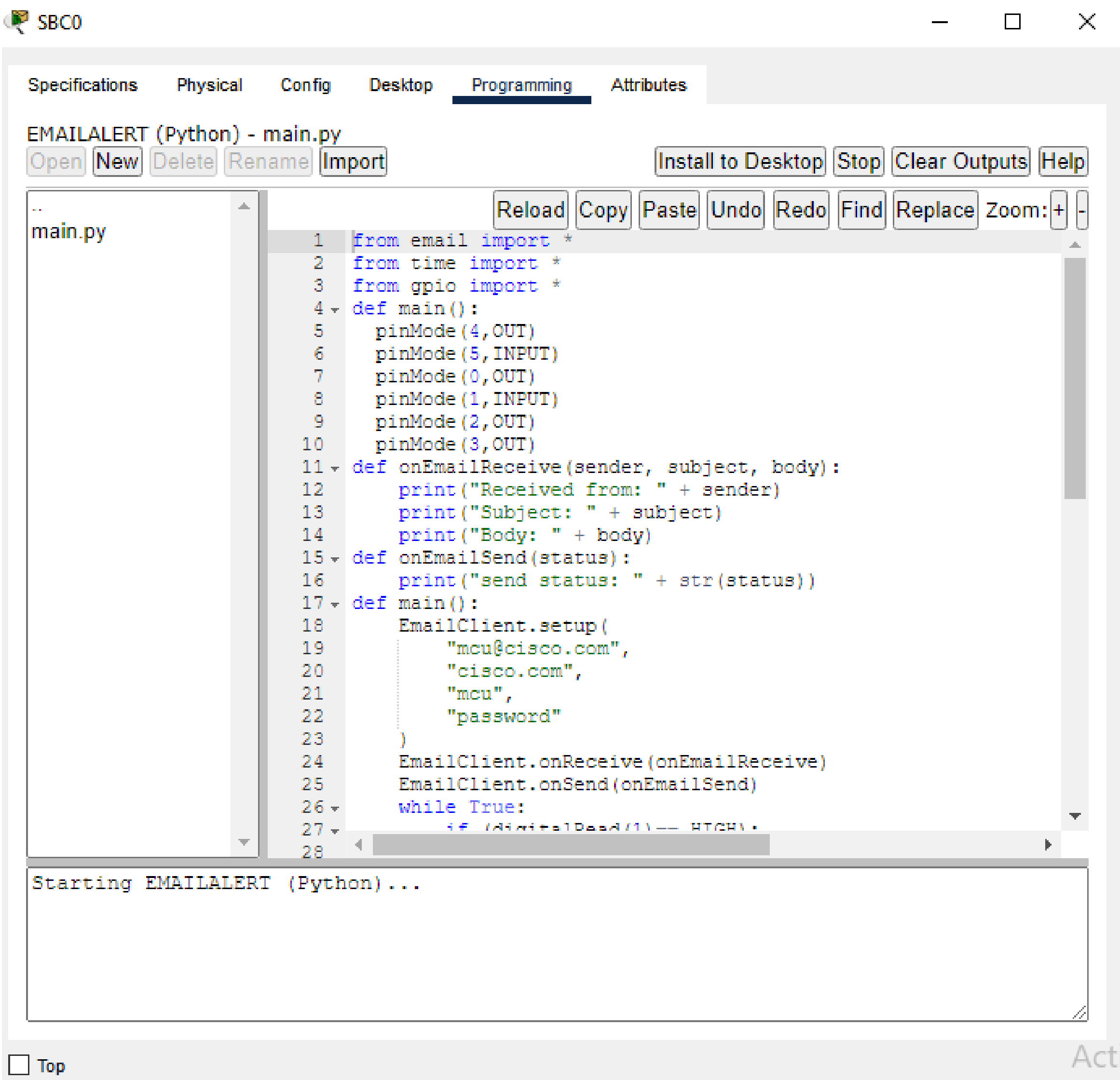

***Figure 42**: Execution of the program [4]*

Appendix A presents the source code of the program used to simulate our door control system by facial recognition [4] on Pocket Tracer.

3.1. **First case:** Face detected and recognized

There is facial recognition, materialized by the activation of the LED, the door opens.

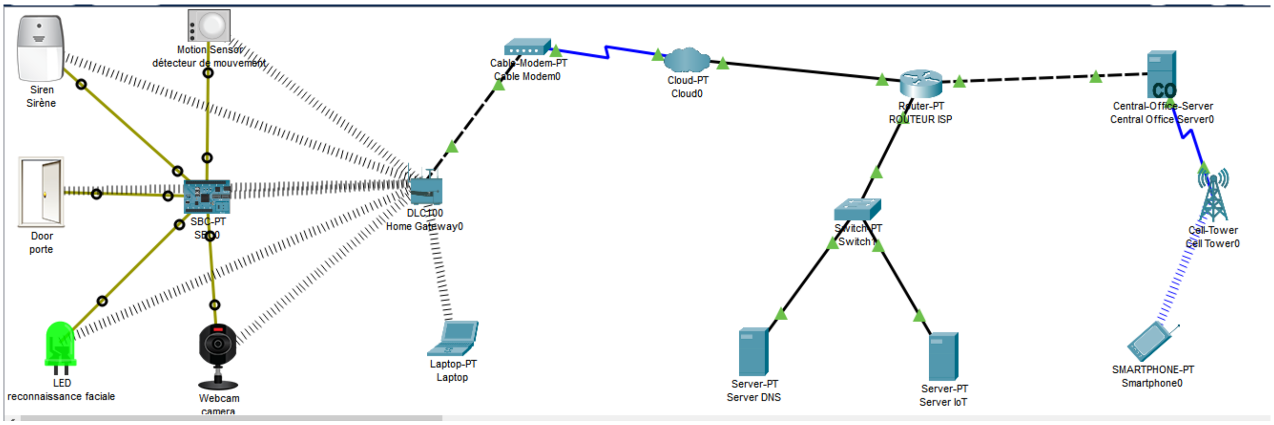

Figure 43: Opening of the door [4]

An email is sent to the administrator to report the opening of the door.

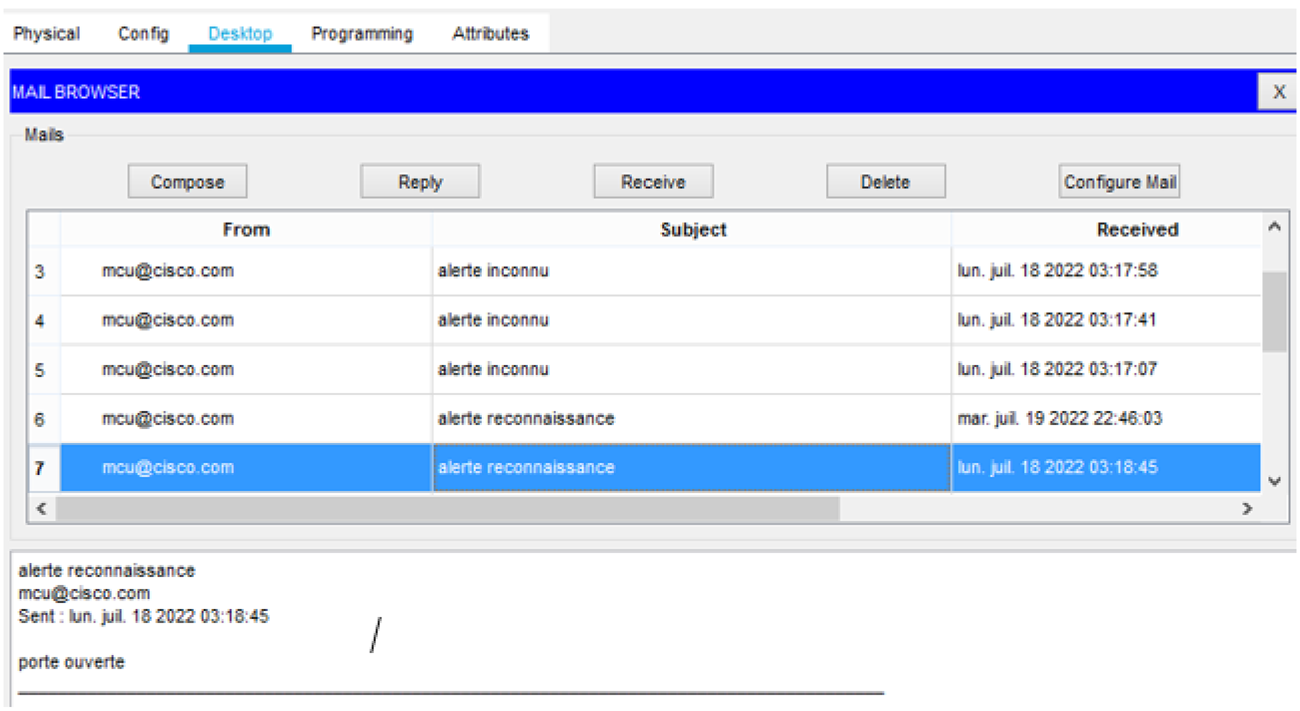

***Figure 44:** mail for the door opening [4]*

3.2. **Second case**: Face detected and not recognized

There is no facial recognition and the alarm activates itself.

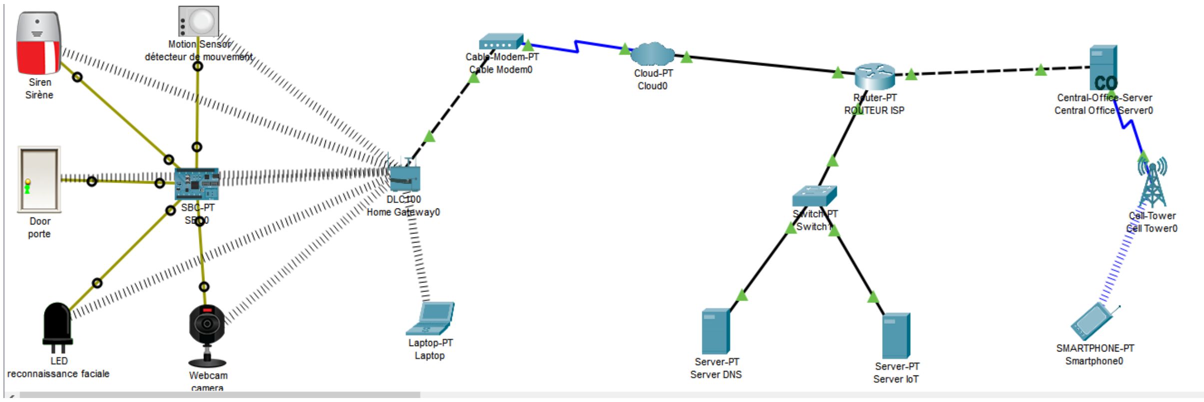

***Figure 45**: Activation of the alarm [4]*

An email is also sent to the administrator in this case.

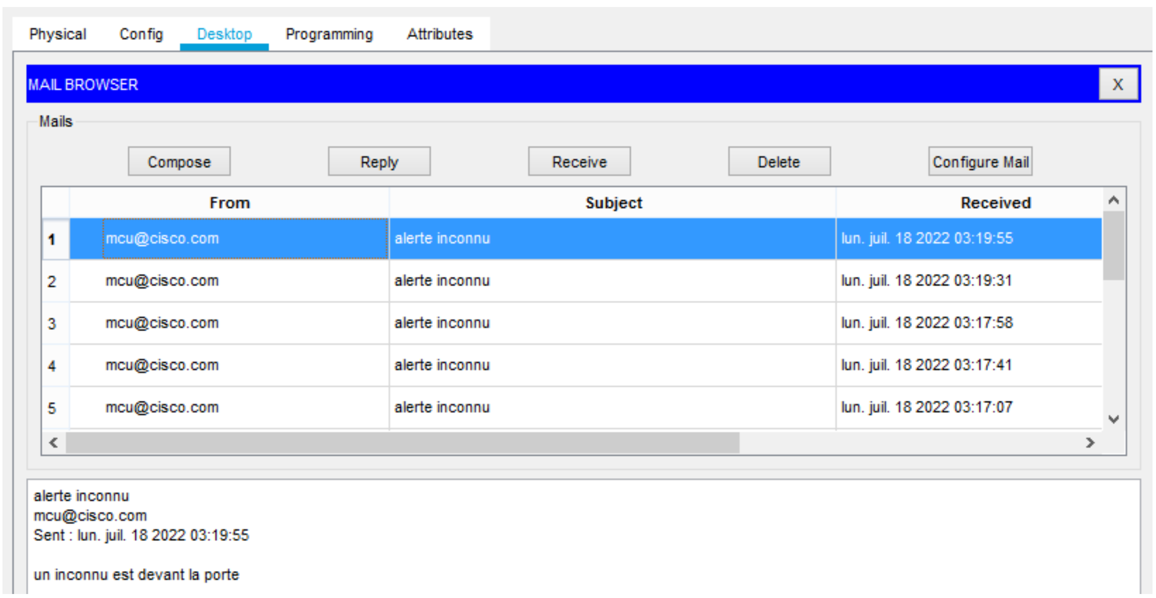

***Figure 46**: Receival of the message*

3.3. Monitoring Interface

Il est possible de contrôler les équipements de notre système à travers le téléphone. En effet, l'administrateur peut, à travers son téléphone, activer ou éteindre la caméra, ouvrir ou fermer la porte ou encore permettre ou éteindre la sirène. Pour cela tous ces équipements doivent être connectés au serveur IoT en mode « remote ».

La figure ci-dessous présente l'interface de commandes des différents équipements à travers le smartphone.

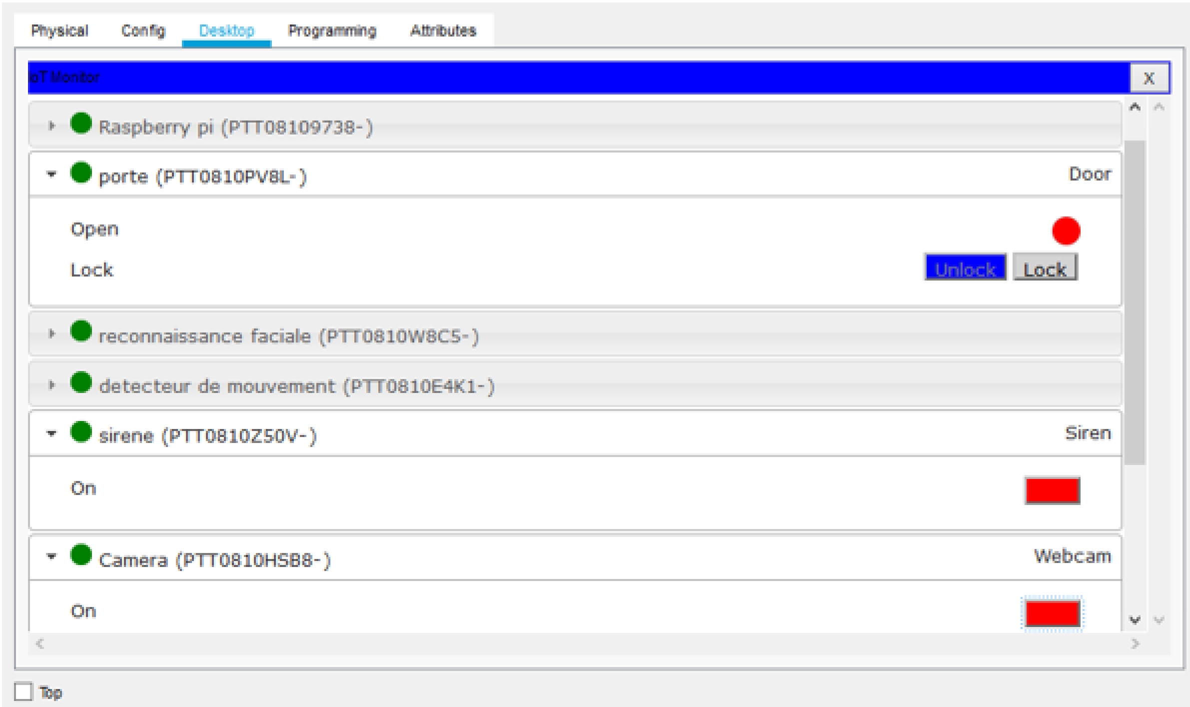

***Figure 47**: Administration of equipment via smartphone [4]*

IV. CONCLUSION

This work highlights the design and simulation of a video surveillance system with facial recognition on the Packet Tracer network simulator. This system is an Internet of Things (IoT) application that uses a Raspberry Pi nanocomputer on which a camera is connected, and exploits cloud services to store the faces of people with authorization to enter the company. . The solution we propose is a didactic tool. It can therefore be used by teachers for students for educational purposes.

BIBLIOGRAPHIC REFERENCES

[1] Eclipse IoT Working Group , "The Three Software Stacks Required for IoT Architectures", Eclipse, 2016.

[2] Gautier ABOU LOUME, "IoT applied to facial recognition leading to the opening of a door using the raspberry pi and a cloud service", Master's thesis in Telecommunications Engineering, University of Yaoundé I / Ecole Nationale Supérieure Polytechnique de Yaoundé, 2022 .

[3] Gautier ABOU LOUME, Alphonse BINELE ABANA, Emmanuel TONYE, and Yvan KABIENA, "Facial Recognition in the Opening of a Door using Deep Learning and a Cloud Service", *International Journal of Intelligent System Application Engineering (IJISAE),* vol. 10, no. 3, pp. 40-45, Oct. 2022

[4] Berline DJIMELI DTIABOU, "Modeling and simulation of the automatic opening of a door by facial recognition with deep learning using a Raspberry Pi nanocomputer and Cloud services", Professional Master's thesis in Information Systems Security and Communication (MASSICO), University of Yaoundé I / National Polytechnic School of Yaoundé, 2022.

[5] Allele Imane , "A new indexing structure for data from the object tracking process in the Internet of Video Objects", Master's dissertation in Computer Science, University of May 8, 1945-Guelma, September 2021.

[6] Siyuan Gao, "An intelligent video surveillance system", In : *2010 International Conference on E-Product E-Service and E-Entertainment*, IEEE, pp. 1-4, 2010.

[7] Brahim FAROU, "Multimedia mining Pattern recognition in a video", Thèse de doct, Université Badji Mokhtar-Annaba, 2016

[8] Sergio A Velastin et Paolo Remagnino. *Intelligent distributed video surveillance systems*. T. 5. IET, 2006.

[9] Aleksandar Milosavljević, et al. "Integration of GIS and video surveillance". In: *International Journal of Geographical Information Science*, vol. 30, no. 10, pp. 2089-2107, 2016.

[10] Chang Wen Chen, "Internet of Video Things : Next-Generation IdO With Visual Sensors", In : *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6676-6685, 2020.

[11] Maria Valera, et Sergio A Velastin. "Intelligent distributed surveillance systems: a review". In: *IEE Proceedings-Vision, Image and Signal Processing*, vol. 152, no. 2, pp. 192-204, 2005.

[12] Niki Martinel. "On a Distributed Video Surveillance System to Track Persons in Camera Networks". In: *ELCVIA Electronic Letters on Computer Vision and Image Analysis* , vol. 14, no. 3, pp. 39-41, 2015.

[13] Paolo Remagnino, AI Shihab et Graeme A Jones. "Distributed intelligence for multicamera visual surveillance". In: *Pattern recognition* vol. 37, no. 4, pp. 675-689, 2004.

[14] Mohamed Amine Ferrag et al. "Big IdO Data Indexing: Architecture, Techniques and Open Research Challenges". In: 2019 *International Conference on Networking and Advanced Systems (ICNAS)*. IEEE. pp. 1-6, 2019.

[15] Ala-Edine BENRAZEK. *Internet of Video Things (IoVT) : Next generation of video surveillance systems.* Déc. 2020. URL : https://benrazealaeddine.medium.com/itecturewhat-is-internet-of-video-things-iovt-1f2323e02a0b

[16] Medium – Deep Learning for Computer Vision for the average person, [On line] available on: https://medium.com/diaryofawannapreneur/deeplearning-for-computer-vision-for-the-average-person-861661d8aa61, publié par: Aashay Sachdeva, le: 06/03/2017 (consulted on 26/10/2022).

[17] Florian SCHROFF, Dimitry KALENICHENKO, et James PHILBIN . "A unified embedding for face recognition and clustering". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2015.

[18] Anand RAJU, T. SHANTI, et al. "Face Recognition and classification Using GoogleNet Architecture". In: *Soft Computing for Problem Solving*. pp. 261-269. 2015.

[19] Maheen ZULFIQAR, Fatima Syed, et al. "Deep Face Recognition for Biometric Authentication". In: *IEEE International Conference on Electrical, Communication and Computer Engineering*. Pakistan. 2022.

[20] Ivan Gruber, Miroslav Hlavac, et al. "Facing Face Recognition with ResNet : Round One". In: *International Conference on Interactive Collaborative Robotics*. pp. 67-74 2017.

[21] Rita Goel, Irfan Mehmood et Hassan Ugail. “A Study of Deep Learning-Based Face Recognition Models for Sibling Identification”. In: *Sensor*. vol. 21. 2021.

[22] Ja Hyung Koo, Se Woon Cho, et al. “CNN-Based Multimodal Human Recognition in Surveillance Environments”. In: Sensor. vol. 18. 2017.

[23] Clubic – Deeplearning : quand l'intelligence artificielle tente d'imiter le cerveau humain, [On line] available on: https://www.clubic.com/technologies-d-avenir/intelligence-artificielle/deeplearning/article-783038-2-promesses-deep-learning.html, (consulted on 6/10/2022)

[24] Satish, Karuturi S R V, and M Swamy Das. "Multi-Tier Authentication Scheme to Enhance Security in Cloud Computing." IJRAR (International Journal of Research and Analytical Reviews) 6, no. 2 (2019): 1-8, 2019.