



KATHMANDU UNIVERSITY
SCHOOL OF ENGINEERING
MTech in ARTIFICIAL INTELLIGENCE
PANCHKHAL, KAVRE

An Assignment
For
Simulating Smart Security Camera System
Using Cisco Packet Tracer

SUBMITTED TO

Prof. Sudan Jha

SUBMITTED BY

Prabesh Regmi
MTech AI 230008

August, 2024

Abstract

As our lives become increasingly connected, the idea of a smart home, one that can be controlled automatically or remotely through devices like smartphones and laptops, has gained significant traction. Building and managing these smart homes effectively requires platforms that support the integration of various smart devices and components. The Internet of Things (IoT) plays a crucial role in this innovation, enabling us to control and simulate smart home environments. IoT technology has applications that extend beyond homes to areas like healthcare, education, and business. This study focuses on developing a home security monitoring system using IoT. The system incorporates devices like a PIR motion sensor, a microcontroller, a webcam, an alarm, and a server, among others. The main objective is to simulate a smart home security setup where a camera and alarm are triggered if someone enters the house, demonstrating the concept of a smart home security system. Using Cisco Packet Tracer, we simulated a smart security camera system and monitored various IoT devices. The results show that these smart devices can be effectively connected and monitored, indicating that such a system could be implemented in real-world scenarios.

Keywords - IOT technology, Cisco Packet Tracer, Home gateway, IoT server, IoT monitor, Microcontroller, Personal computers, Webcam

Introduction

The expression “Internet of Things” (IoT), coined back in 1999 by Kevin Ashton, the British technology pioneer who co-founded the Auto-ID Center at the Massachusetts Institute of Technology, is becoming more and more mainstream [1]. The Internet of Things (IoT) and Internet of Services (IoS) ideas are key components of the broader Industry 4.0 technology. These interconnected things and services allow modern smart factories and integrated value chains to run smoothly [2]. The Internet of Things (IoT) is the networked interconnection of physical or virtual objects embedded with sensors and technologies that communicate and share data with other devices via the Internet. This comprises a wide range of things, such as machinery, gadgets, appliances, and actuators. Each object has a distinct identity and is linked to the internet [3].

Cisco Packet Tracer is a comprehensive, networking technology teaching and learning program that offers a unique combination of realistic simulation and visualization experiences, assessment and activity authoring capabilities, and opportunities for multi-user collaboration and competition. The innovative features of Packet Tracer help students and teachers collaborate, solve problems, and learn concepts in an engaging and dynamic social environment. Some of the major features of the Cisco Packet Tracer are:

- Simulates network devices and protocols
- Supports multiple protocols and technologies
- Includes a range of network devices and topologies
- Provides a comprehensive set of networking tools
- Provides a safe and virtual environment for network experimentation
- Reduces the need for physical hardware
- Allows for quick and efficient network configuration and troubleshooting



Figure 1 Cisco Packet Tracer Interface

Methodology

This study focuses on the short development life cycle of an IoT project using microcontrollers, sensors, home devices, and network connections. The freshly released Cisco version was picked since, in addition to networking functionality, it covers microcontrollers, sensors, and home devices and provides a programming environment. Visual Basic, JavaScript, and Python are among the programming languages supported. Cisco Packet Tracer was used for developing the IoT platform because it offers a variety of network components that simulate a real network.

Cisco packet tracer enables developers to view the flow of data packets and carry out analysis on the data. A humidity sensor is used to measure the humidity level and the humidifier is set to ON/OFF according to the condition stored on the server.

IoT System Development

Router-PT has four Ethernet ports and can route between different networks. Its first Ethernet port Fa0/0 is connected to the switch to which the server and computer are connected. Its second Ethernet port Fa0/1 is connected to the home gateway. The home gateway has four Ethernet ports and a wireless access point configured with the "Home Wi-Fi" service set identifier (SSID) with WEP password 0123456789. Microcontrollers, Webcam, Alarm, PC, and Smartphone are connected to the home gateway via Ethernet and Wi-Fi. The PIR motion sensor, LED light, and LCD display are connected to the microcontroller and a small JavaScript code is written in the microcontroller to check if motion is detected from the PIR motion sensor and display in LCD display along with turning on a small LED connected to the microcontroller. It also shows if motion is detected via API in the IoT server. Server is used to store the rules and control the webcam and alarm according to the motion status read from the PIR motion sensor. Fig 2 shows the entire simulation of the smart home security camera system.

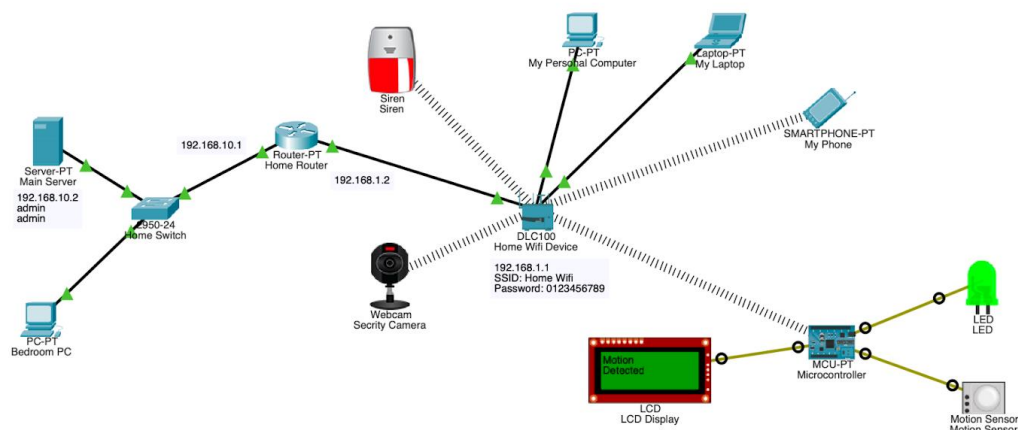


Figure 2 Smart Home Security Camera System

The Cisco packet tracer interface provides inbuilt devices to be added to the network. The first step is to select a router from the network devices and connect the switch to its Ethernet port Fa1/0. The second step is to connect devices to the switch. In this simulation, two wired devices were attached to the switch. The devices configured included a server and a PC. The router's Ethernet port Fa1/0 was given static IP 192.168.10.1 and turned on and the server was given

static IP address 192.168.1.2 with gateway 192.168.1.1, i.e. IP address of Fa1/0 port of a router. Fig 3 shows the IP configuration for the first Fa1/0 port of the router and Fig 4 shows the IP configuration for the server. The user was registered to the main server via a web browser in the desktop tab in the server configuration. Then the IoT server was turned on in the config of the webcam, microcontroller, and alarm where the remote server was set to the IP address of the main server with the username and password used while creating the user in the main server.

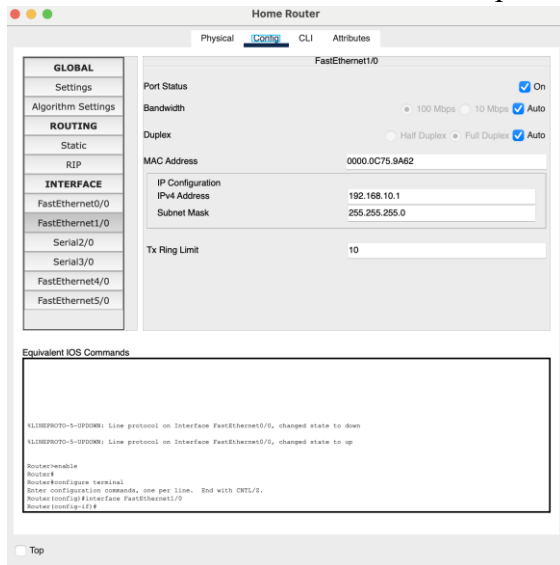


Fig 3 Router Fa1/0 config

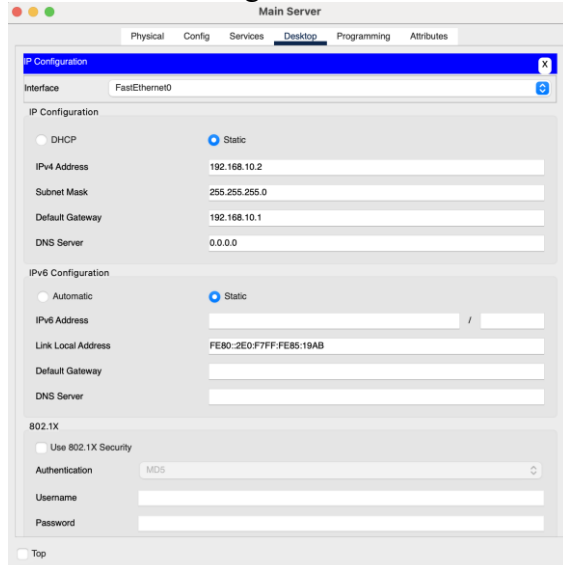


Fig 4 Server config

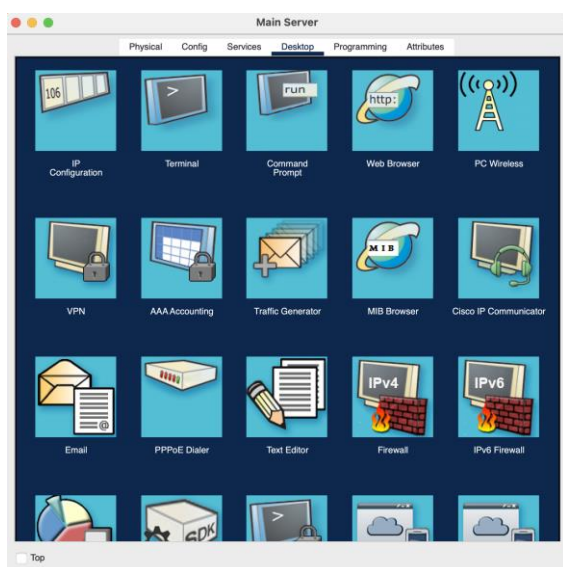


Fig 5 Main server desktop tab

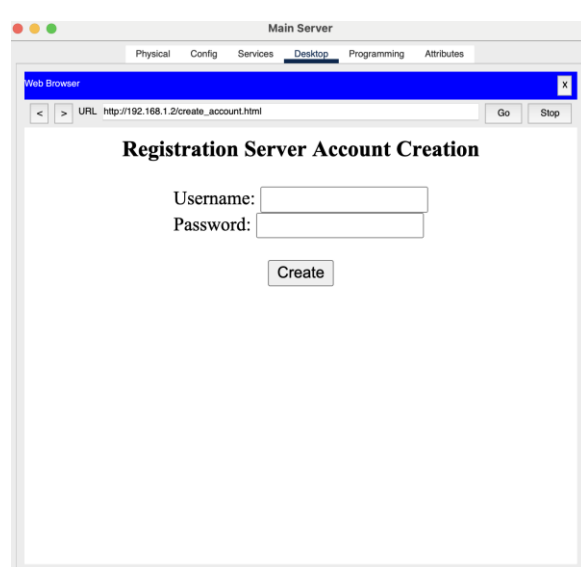


Fig 6 User registration interface on the main server

Now, the home gateway is connected to the router's second Ethernet port Fa0/0 with static IP 192.168.1.2 and the gateway's IP was set to 192.168.1.1. The gateway was then configured for a wireless network to have wireless SSID along with a WEP password. Microcontroller, Siren, Webcam, Smartphones, and different kinds of devices are connected to the gateway both wired and wirelessly. The different devices connected to the gateway are given dynamic IP addresses.

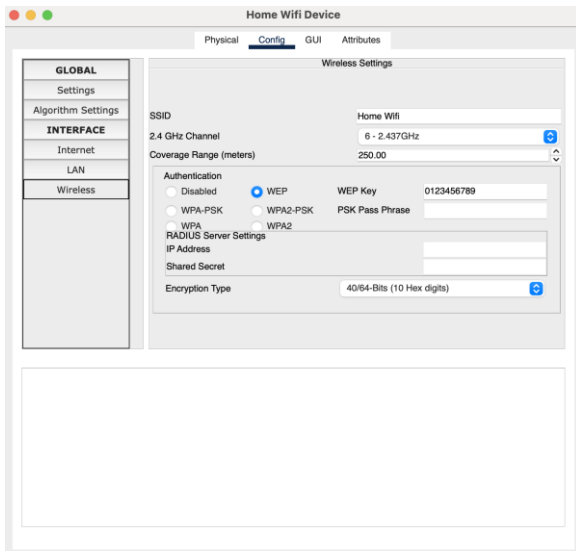


Fig 7: Home gateway Wi-Fi configuration

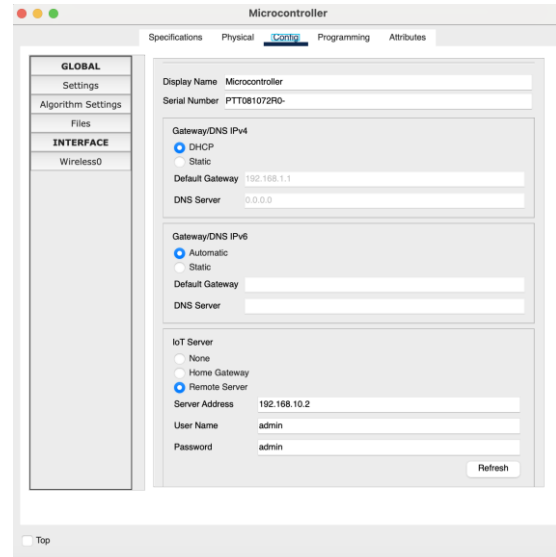


Fig 8: Microcontroller Wi-Fi configuration

In the next step microcontroller's D0 pin was connected to the PIR motion sensor's pin D0, D1 pin to LED's D0 pin and finally D2 pin was connected to the LCD's D0 pin. A small JavaScript code was added in the main.js file in the programming tab of the microcontroller where the microcontroller takes the PIR motion sensor's reading from its D0 pin and custom writes in its D1 pin and D2 pin for LED and LCD display respectively. Microcontrollers also serve the data from the motion sensor to the IoT server. The microcontroller serves as true if there is a reading of HIGH from the PIR motion sensor which means the motion is detected and as false if there is a reading of LOW from the PIR sensor which means no motion is detected. Each sensor's specification can be found in the device's specification tab. Fig 9 shows the PIR motion sensor's specification.

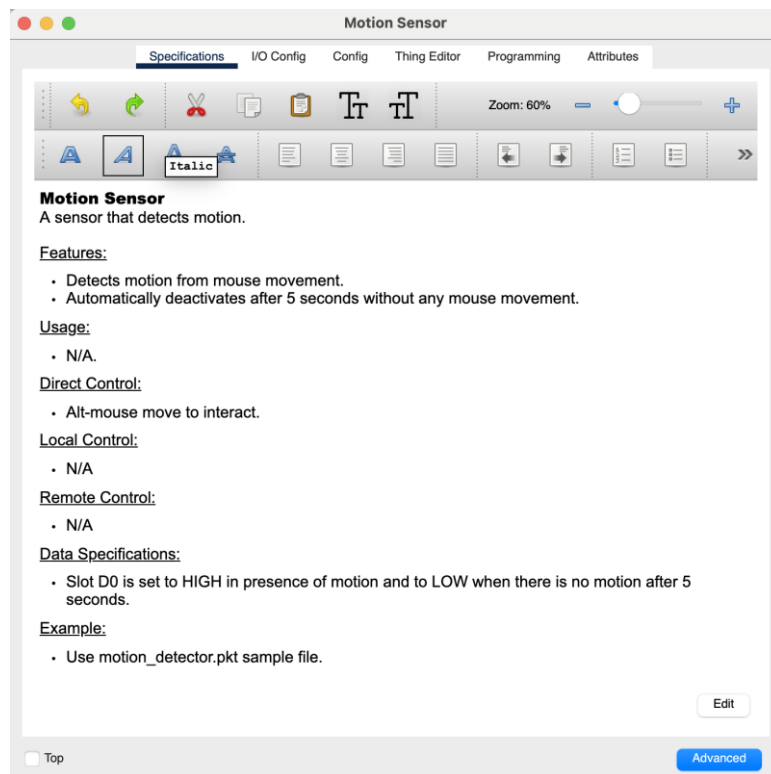


Fig 9: PIR motion sensor's specification

Here's the sample code for the microcontroller:

```
// Define pins for motion sensor, LED, and LCD
var motionSensorPin = 0;
var ledPIN = 1;
var lcdPIN = 2;
// Function that runs once when the program starts
function setup() {
  // Set motionSensorPin as an input pin (for the motion detector)
  pinMode(motionSensorPin, INPUT);
  // Setup IoEClient with device information
  IoEClient.setup ({
    type: "MicroController", // Define the device type as MicroController
    states: [
      {
        name: "Motion Detector", // Name the state as "Motion Detector"
        type: "bool", // Set the state type as boolean (true/false)
      }
    ]
  });
}
// Function that runs continuously in a loop
function loop() {
  // Read the value from the motion detector connected to motionSensorPin
  var motion = digitalRead(motionSensorPin);
  // Print the motion value to the Serial monitor
  Serial.println("Motion Detected: " + motion);
  // Check if the motion value is HIGH (indicating motion is detected)
  if(motion == HIGH) {
    digitalWrite(ledPIN, HIGH); // Turn on the LED connected to ledPIN
    customWrite(lcdPIN, "Motion \nDetected"); // Display a message on the LCD indicating
motion is detected
  } else {
    digitalWrite(ledPIN, LOW); // Turn off the LED connected to ledPIN
    customWrite(lcdPIN, "Motion \nNot Detected"); // Display a message on the LCD
indicating no motion is detected
  }
  // Report the current state of the motion detector to IoEClient
  IoEClient.reportStates([motion]);
}
```

Then the server's IP was visited from any of the computers or smartphones connected and logged in with a created user account. Then it is verified that the microcontroller is sending the PIR motion sensor's reading correctly and the alarm and webcam can also be seen in the IoT server. Next, conditions were added in the IoT server that the webcam and the alarm turn on and start recording automatically if the PIR motion sensor detects the motion and turns off if no motion is detected.

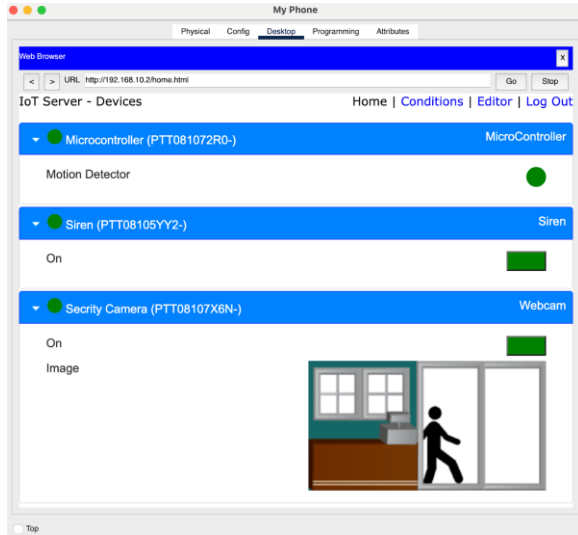


Fig 10: IoT devices registered in the main server

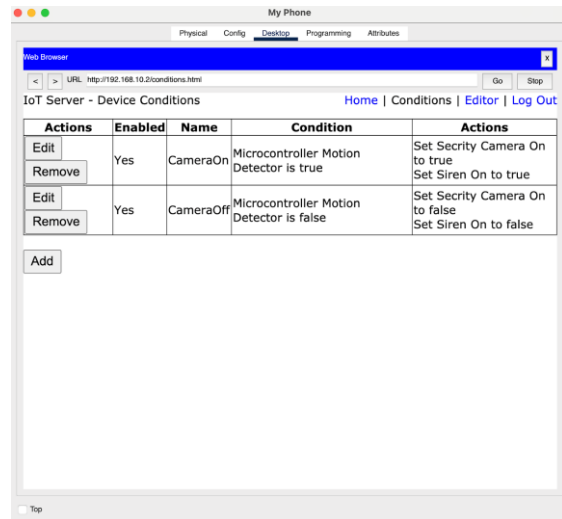


Fig 11: Conditions in the main server

Results

After completing all the main procedures of designing a smart security camera system, the system was tested, Fig 12 shows that the webcam and alarm are set to ON automatically when motion is detected. Fig 13 shows the webcam and alarm set to OFF when no motion is detected.

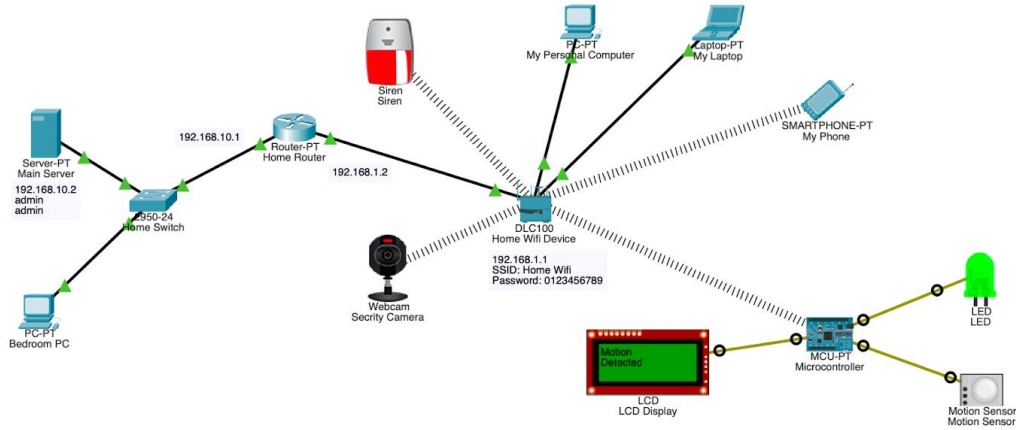


Fig 12 Webcam and alarm status is shown ON when motion is detected

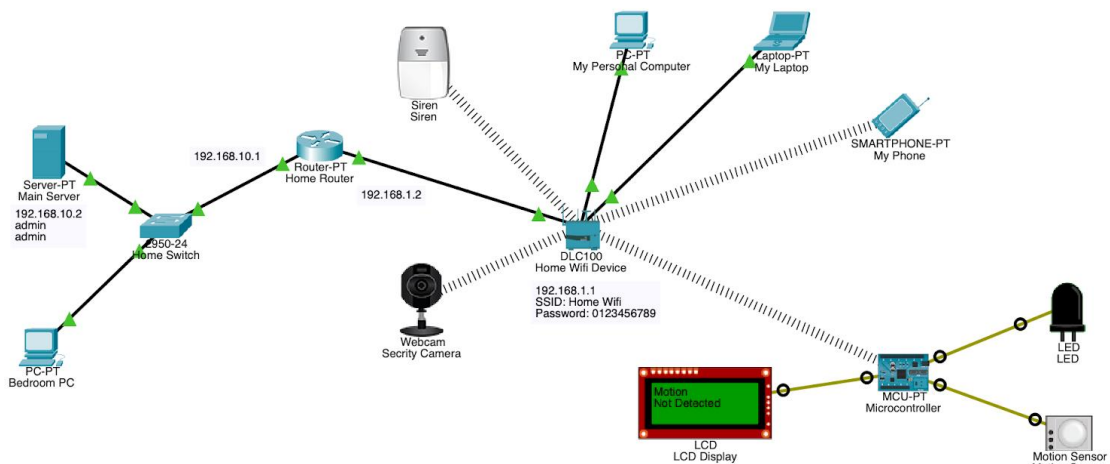


Fig 13 Webcam and alarm status is shown OFF when motion is not detected

In the end, a message was passed between different devices randomly in the simulation and it was found to be successful.

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
●	Successful	Main Server	My Phone	ICMP	■	0.000	N	0	(edit)	(delete)
●	Successful	Security Camera	Microcontroller	ICMP	■	0.000	N	1	(edit)	(delete)
●	Successful	Microcontroller	Security Camera	ICMP	■	0.000	N	2	(edit)	(delete)
●	Successful	Main Server	Microcontroller	ICMP	■	0.000	N	3	(edit)	(delete)
●	Successful	Siren	Main Server	ICMP	■	0.000	N	4	(edit)	(delete)

Fig 14 Message passing between different devices

Conclusion

This research aimed to simulate a smart security camera system. The technology innovation and the hike in automated system usage is the motivation behind this work. Such security measures are very critical and IoT is providing a new and excellent concept to make our surroundings smarter. In this study, the Cisco packet tracer was utilized to design and simulate the smart home security system. Cisco public tutorials and previous research were very beneficial in the implementation process. The result showed that the devices can be controlled and monitored using end-user devices. Cisco packet tracer gives various facilities that make simulation easy. The results proved that there is an opportunity to apply this model in real life. The IoT concept can be applied in various domains.

References

- [1] T. Kramp, R. van Kranenburg, and S. Lange, "Introduction to the internet of things," in *Enabling Things to Talk: Designing IoT Solutions with the IoT Architectural Reference Model*, Springer Berlin Heidelberg, 2013, pp. 1–10. doi: 10.1007/978-3-642-40403-0_1.
- [2] N. Gwangwava and T. B. Mubvirwi, "Design and Simulation of IoT Systems Using the Cisco Packet Tracer," *Advances in Internet of Things*, vol. 11, no. 02, pp. 59–76, 2021, doi: 10.4236/ait.2021.112005.
- [3] G. Alfarsi, J. Jabbar, R. M. Tawafak, S. I. Malik, A. Alsidiri, and M. Alsinani, "Using Cisco Packet Tracer to simulate Smart Home." [Online]. Available: www.ijert.org