LONDON METROPOLITAN UNIVERSITY

islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title
CS5053NI/CC5068NI– Cloud Computing & IoT

Smart Garage Door System

Assessment Weightage & Type
50% Group Coursework

Semester
2023 Spring/Autumn

Group members

London Met ID	Student Name
22067520	Dikshya Sharma
22068750	Shuvam Rajbhandari
22067529	Ikshit Maharjan
22068086	Anushka Bhandari

Assignment Due Date: 12 January 2024
Assignment Submission Date: 11 January 2024
Submitted to: Mr. Sugat Man Shakya
Word Count: 3232

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Acknowledgement

We would like to express our sincere gratitude to our module leader, lecturer, and tutors for their unwavering help and support throughout our coursework. We are so thankful to our Module Leader as well as our tutor Mr. Sugat Man Shakya, for his important direction and support in enabling us to carry out an Internet of Things project which have been invaluable in successful completion of our project.

Likewise, we are more than thankful to our other tutors Mr. Shishir Subedi, Mr. Ayush Bhakta Pradhanang and Mr. Suryansh Mathema for guiding us in all possible ways whether it was in person or through hangouts. Whether it was problem on IDE or problem on hardware components, our tutors were always there to help us and clear our query regarding the project.

Furthermore, we are really grateful to the university and all the members of Islington College for assigning this coursework and also the resources needed to complete the project. Thanks to everyone who directly or indirectly helped us by sharing their knowledge and insights regarding the Internet of Things.

Abstract

This report provides a documentation of a to prototype build of **Smart Garage Door System.** This task includes analysing, developing, and implementing an agreed-upon prototype with our module tutor. The primary objective of this coursework is to cultivate a practical understanding of the challenges associated with developing IoT application that depend on diverse hardware and software components.

Handling garage doors in a domestic setting can be unsafe and time-consuming. A solution is provided by a web-app-based garage door opener system. In addition to improved security features like automated shutting and objection detection alert, users can open and close their garage doors. This technology combines homeowner convenience, control, and security to streamline garage door administration.

This Report covers multiple sections focusing on the **Smart Garage Door System**. In the introduction part scenario of the project and problem statement is outlined. Following that, the Background and the Development section provides an overview of the system and it's development with Resources used. It concludes with the Testing section, detailing the examination and evaluation of the project with future works that can be performed and conclusion.

Table of Contents

Table of Figures

Table of Tables

1. Introduction

Technology has revolutionized people's lives, reshaping the way people connect, work and live. From early computers to smart gadgets today, it is evolving at a rate unlike anything before. One fascinating advancement of technology is Internet of Things (IoT). IoT refers to the interconnection of physical devices like sensors and other objects for exchanging data over the internet (S.Gillis, 2023).

IoT stands out for its versatility and can potentially use to create creative solutions, automate operations, and increase efficiency in a variety of fields and daily life contexts. Considering the area of home automation, using a portable smart device, one can control several features of a linked home remotely. A prime illustration is the **Smart Garage Door System**.

Smart Garage Door System is a IoT device which can be used remotely to open and close the garage door. This innovative IoT is designed to revolutionize traditional garage door, allowing user to operate the garage door from a distance using Web Application specially build for operating the garage door with smartphone until one have an Internet connection.

This system not only eases the hassle of opening and closing the door but also significantly enhance the home security. In the modern world where security concerns are increasing rapidly, with just a button click on a dedicated web app, one can operate the door which not only brings ease to daily life but also the adds an extra layer of protection to the house.

1.1. Current Scenario

For a lot of people smart garage door system is a now a reality since the growth of home automation. People are considering investing on smart garage door system as a wise decision because of its convenience, security and added home value. Due to the covid pandemic, many people are switching their lifestyle to contactless systems and hence the demand of smart garage door system that is accessed remotely are increasing.

The hassle-free nature of this system has led to widespread among people which not only adds convenience to their life but also ensures the security of their belongings by operating remotely through Internet. These features are driving the popularity of smart garage door system making them crucial part of home automation.

1.2. Problem Statement and Project as a Solution

Manual handling of opening and closing the garage door has become more than just an inconvenience in people's life. It has given rise to series of consequences, from unexpected door closure causing accidents and injuries to the risk of theft due to outdated security measures. It's even estimated that 30,000 people are injured every year due to traditional approach of garage door (Schlesinger, 2020).

In the same way, the break-ins to the house through garage door also possess a significant problem due to lack of security measures resulting in loss of properties, belongings, and even human life. According to the alarming statistics, 9% of the burglars use garage door as their access point to the house (The Zebra , 2023).

Therefore, the installation of modern garage door in today's lifestyle is undeniable which not only provides convenience to operate garage door remotely but also provide security. To tackle these challenges, we present the "**Smart Garage Door System**", an innovative solution to operate and monitor garage door seamlessly using a smartphone through a dedicated web application by only accessed users.

Figure 1: Garage Door Accidents (Blog Master, 2015).

Figure 2: Garage Theft((Salisbury Maryland, 2023)

1.3. Aims and Objectives

1.3.1. Aims

The main aim of this project is to gain the ability to control the status of garage door from anywhere by smartphones through dedicated web application on internet.

1.3.2. Objectives

The objectives of this "**Smart Garage Door System"** are listed below:

• The system is expected to open and close garage door using smartphone via dedicated web application.

• The system is expected to enhance safety and security measures.

• The system is expected to utilize cost-effective components like nodeMCU, IR sensors and servo motors.

• The system is expected to enhance user experience through easy monitoring.

2. Background

The background of the system includes following steps:-

2.1. System Overview

This project is about creating a **Smart Garage Door System** that can be opened and closed using a smartphone through web application remotely. It uses Node MCU (ESP8266) a microcontroller to connect to the internet. This allows you to control the garage door from your device as it processes the collected data.

There's a motor that moves the door and a sensor that checks if something is in the way when the door is closing to prevent accidents. The system uses simple web page with buttons for opening and closing the door. It's a modern way to operate your garage door using IOT components easily and safely.

In this project, we used 4 primary components. Node MCU (ESP8266) was used to connect with the breadboard, IR sensor and a servo motor. Connections were made using Jumper wires. IR sensor is used to detect any object near the garage door to prevent the garage door from closing to avoid accidents. A servo motor was used to open and close the garage door.

2.2. Design Diagrams

This report involves several design diagrams to make it further easier to visualize the system.

2.2.1. Hardware Architecture

System architecture is a conceptual model that explains the arrangement and operation of various hardware, software, network, and subsystem components (GeeksOfGeeks, 2023). Below is the system architecture of our system: -

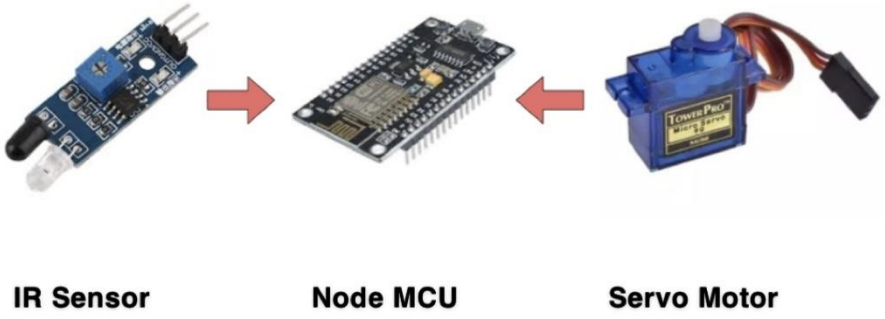

Figure 3: System Architecture of a System

2.2.2. Block Diagram

A block diagram is a graphical representation of a system that provides a visual representation of how different parts of a system work together. It tracks the inputs of a system by creating blocks and using connections to show how those blocks produce a result (Indeed Editorial Team, 2022).

The block diagram of our system is present below: -

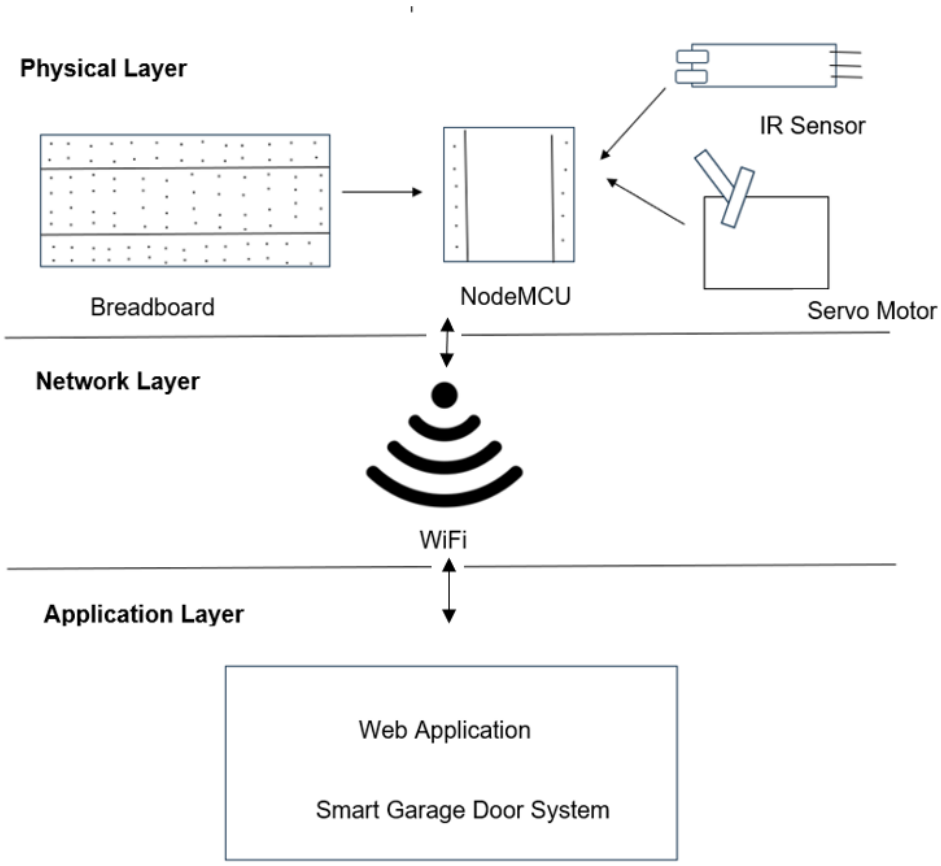

Figure 5: Block Diagram of a system.

2.2.3. Circuit Diagram

A circuit diagram visually represents an electrical circuit, and it can be presented in two main styles: pictorial or schematic. The pictorial version uses simple, easily understandable images of components, suitable for a general audience without technical expertise and a schematic diagram employs standardized symbols familiar to professionals like electricians.

This type of diagram provides a more technical and detailed view of an electrical circuit, catering to those with specific electrical knowledge. The choice between these two styles depends on who is intended to view and use the diagram. (SmartDraw, 2024).

The Circuit diagram of our system is present below: -

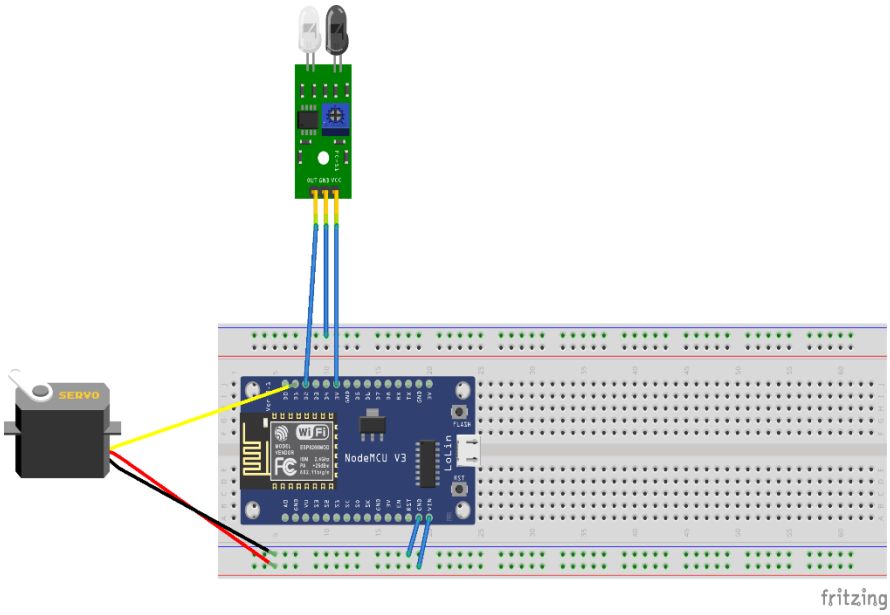

Figure 6: Circuit Diagram of a System.

2.2.4. Schematic Diagram

A schematic diagram is like a roadmap for electronic circuits. It's a simple, two-dimensional representation that shows how different electrical components are connected and work together. For PCB (Printed Circuit Board) designers, understanding schematic symbols is crucial (protoexpress, 2024) .

The schematic diagram of our system is present below: -

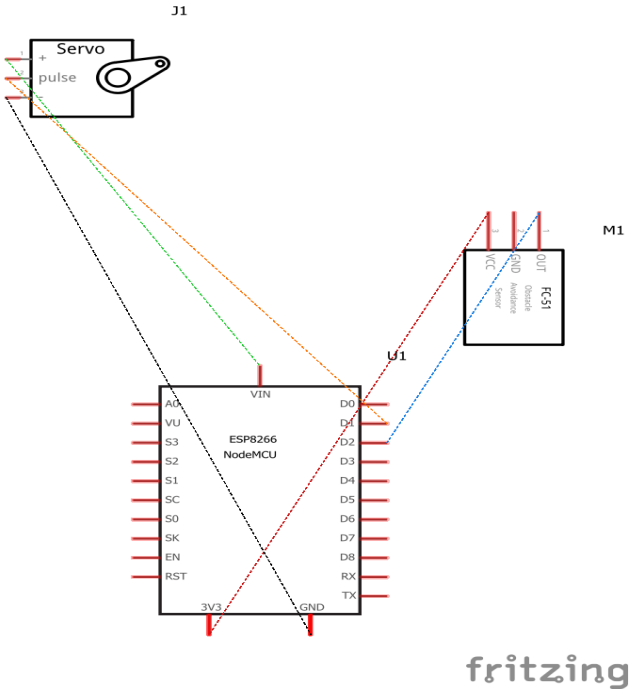

Figure 7: Schematic Diagram of a System.

2.2.5. Flowchart

Flowcharts are graphical representation of the data or the algorithm for a better understanding of the code visually. It displays step-by-step solutions to a problem, algorithm, or process (geeksforgeeks, 2024).

Flowchart of our system is present below:-

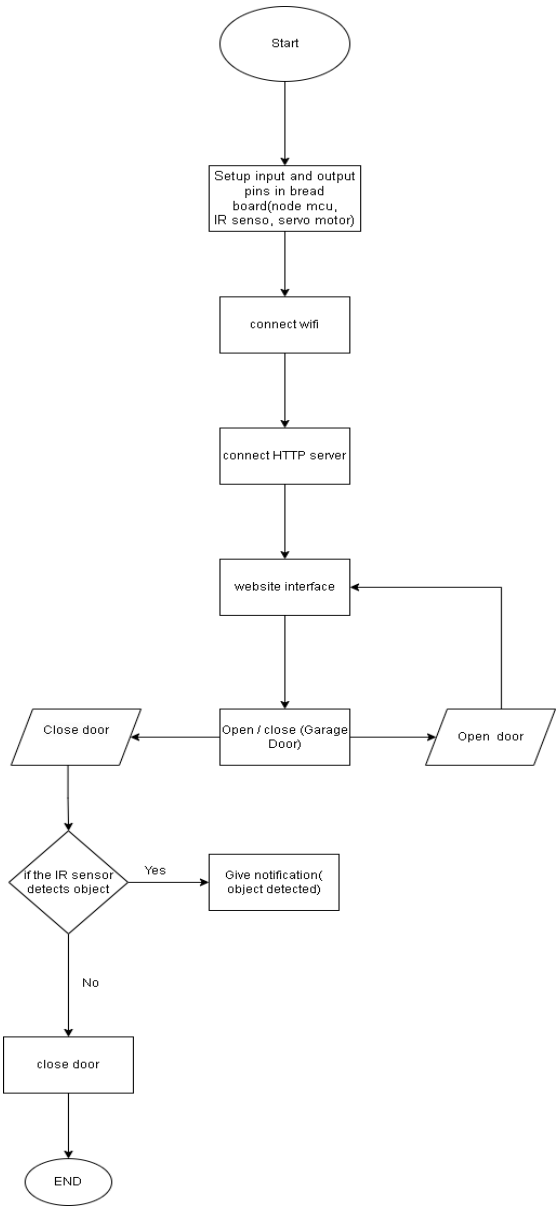

Figure 8: Flowchart of a System.

2.3. Requirement Analysis

To complete the project successfully and to document this report, several components were used that are explained below.

2.3.1. Hardware Components

The development of this project required of several hardware components. They are the physical elements that were used to develop the project. Primary hardware components used for this project are listed below.

- **ESP8266(Node MCU)**

The NodeMCU (Node Microcontroller Unit) is an open-source platform for software and hardware development.

Figure 9: ESP8266 Node MCU

- **Digital IR Sensor**

An Infrared or IR sensor is a **active sensor** since it emits its own infrared signals and detects the reflection from objects to determine their presence.

Figure 10: Digital IR Sensor

- **Jumper Wires**

  Jumper wires are wires with connector pins at both ends, enabling the connection of two points without the need for soldering.

Figure 11: Jumper Wires

- **Servo Motor**

  A Servo Motor is a **rotary actuator** which converts electrical energy into mechanical motion.

Figure 12: Servo Motor

- **Breadboard**

The Breadboard is a rectangular board with embedded holes designed to insert electronic components.

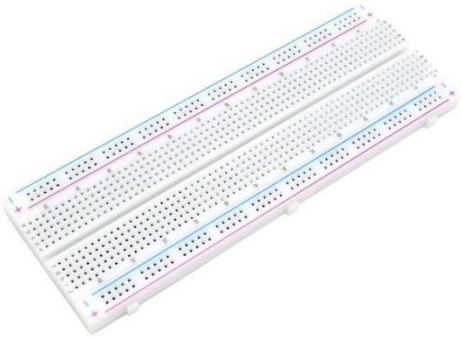

Figure 13: Breadboard

2.3.2. Software Components

The development of this project required of several software components. Primary software components used for this project are listed below.

- **Arduino IDE**

An Arduino IDE is used to connect to Arduino devices to upload programs and communication.

Figure 14: Arduino IDE((MrSottong, 2023)).

- **Fritzing**

Fritzing is used to create different circuit diagrams by joining Pins precisely. It is also used in creating schematic diagram.

Figure 15: Fritzing (Perera, 2021).

- **Draw.io**

  Draw.io is proprietary software for making diagrams and charts and is used as tool for creating Flowchart for our system.

Figure 16: draw.io (Edirisinghe, 2023)

- **Ms-Word**

  Here, Ms- Word is used for documenting reports. The system architecture diagram of our system is also made using Ms-Word.

Figure 17: MS Word((aadi3, 2023)).

3. Development

This segment of the report explains about the development of this IOT project.

3.1. Planning and Design

To initiate the project, our team convened meetings to select a topic and upon reaching a decision, we planned to assign individual tasks to make sure all the things go as planned. We chose **Smart Garage Door System** as it was interesting, and the components were also easily available in the resource department of our college.

The objective of this system is to enable remote access to your garage door which is successfully created with the help of different software and hardware components. As according to the requirement for prototype, we included one microcontroller to process the collected data, one sensor and one actuator to get our task done.

Initially, our plan was to go with Arduino for the project. But due to our project requiring Wi-Fi, we decided to go for NodeMCU which comes with a built-in Wi-Fi chip. We then collected every resource needed for this system and begin to build this system by dividing the task to each individual and later beautified it.

3.2. Resource Collection

To successfully execute this IoT project, many types of hardware devices and tools were needed. Almost every component was available in the resource department of Islington college. However, the IR sensor didn't function as it should have prompting us to acquire a replacement from the market. The hardware components needed for (smart garage door opener) are: -

- NodeMCU
- Jumper Wire
- Breadboard
- Servo Mortar
- Micro USB Data Cable
- IR Sensor

3.3. System Development

The project we have developed is an IoT based Smart Garage Door System which is a smart way of opening and closing your garage door with a simple push of a button on your smartphone. We used four primary components, a Node MCU which was used for the overall software development to control the program, a servo motor for the door open and close mechanism, an IR sensor to detect objects and finally a breadboard for connections.

- **Step 1: Connecting the Node MCU on the breadboard.**

The first step for the system development phase of this project was to connect the Node MCU to the breadboard and the PC using a USB cable to store the code and power the Node MCU.

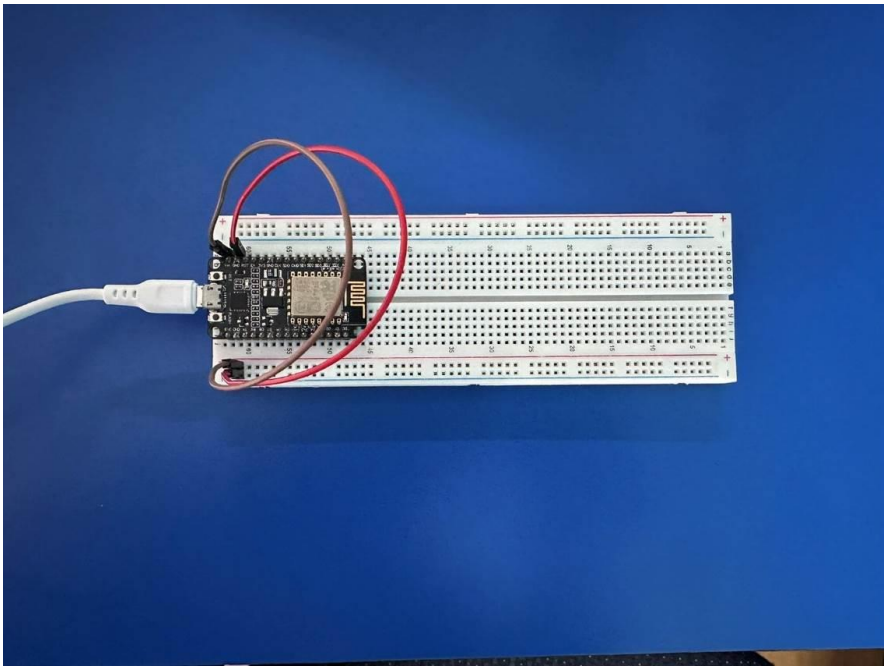

Figure 18: Connecting NodeMCU on breadboard.

Node MCU Pins	Breadboard Pins
GND	**Negative (-ve)**
Vin	**Positive(+ve)**

Table 1: NodeMCU Pins Connections on Breadboard.

- **Step 2: Connecting the IR Sensor**

The next step after connecting the Node MCU to the computer was to connect the IR sensor. The IR sensor was connected to the microcontroller using jumper cables.

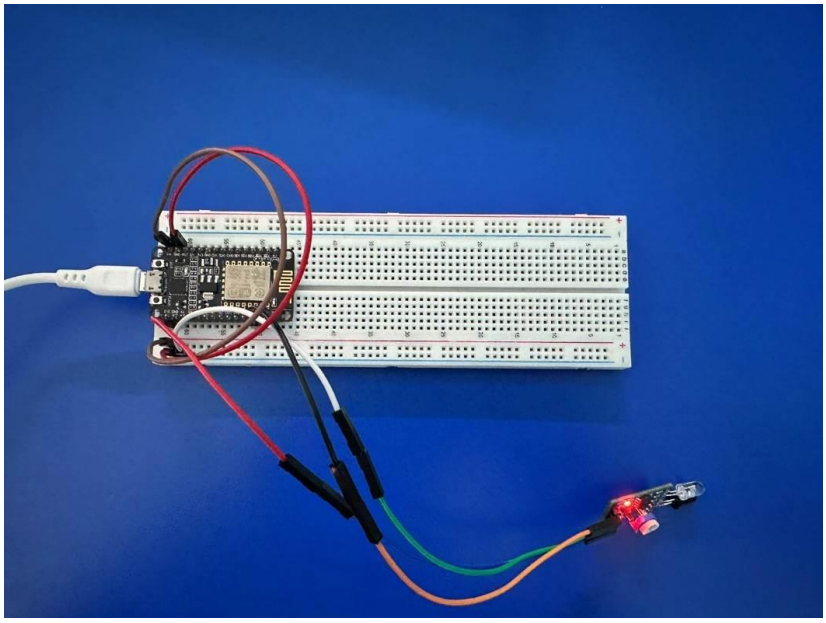

Figure 19: Connecting IR Sensor on Breadboard.

Sensor Pins	Node MCU Pins
VCC	3V3
GND	Negative(-ve)
OUT	D2

Table 2: IR sensors Pins Connections on Breadboard.

- **Step 3: Connecting the Servo Motor**

The final connection was the Servo Motor. Connections made to the microcontroller are listed below.

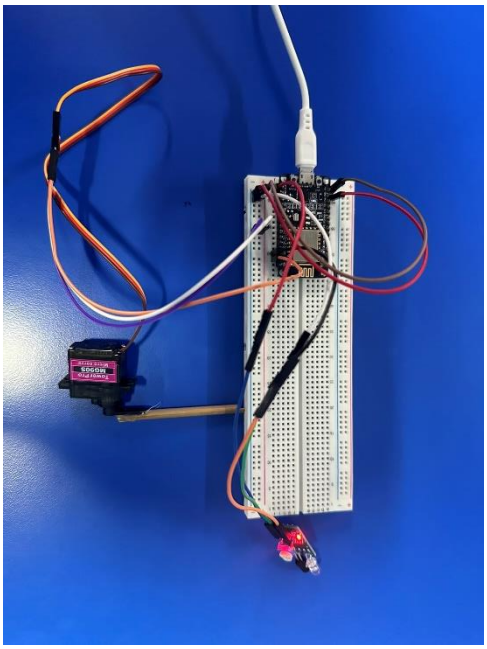

Figure 20: Connecting Servo Motor on Breadboard.

Motor Pins	Node MCU Pins
5V	Positive(+ve)
GND	Negative(-ve)
PWM	D1

Table 3:Servo Motor Pins Connections on Breadboard.

- **Step 4: Beautification**

  After all the sensors and motors were connected, the code was uploaded to the microcontroller. After successfully testing the working mechanism of the system, the beautification of the project was done using cardboards, glues, and papers.

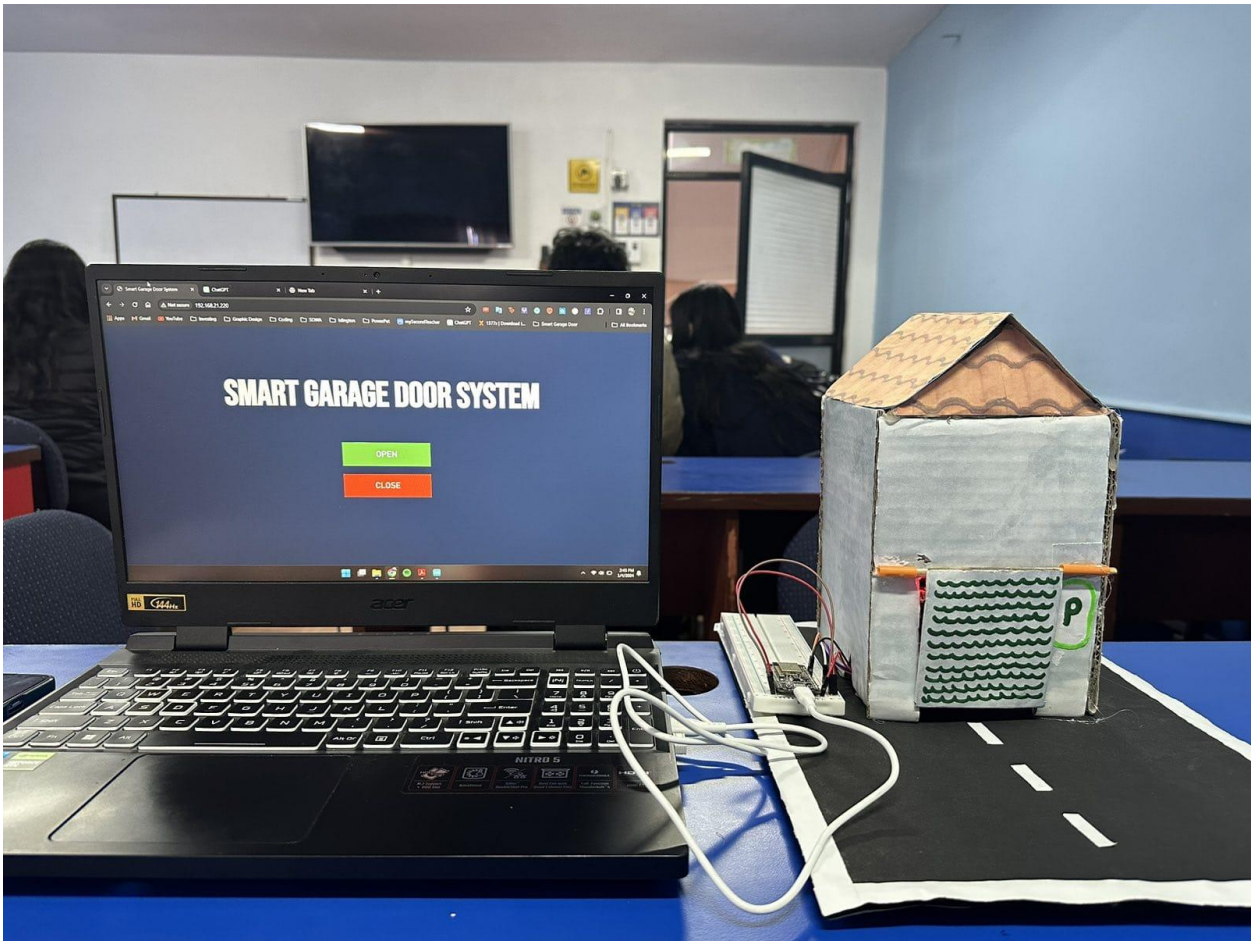

Figure 21: Beautified Smart Garage Door System.

4. Results and Findings

4.1. Results

Our **Smart Garage Door System** is designed to provide seamless and secure solutions for users to conveniently operate and monitor garage door remotely. It provides user remote access to open and close garage door from any distance using smartphone through dedicated web application on Internet.

As our system incorporates cost-effective components like nodeMCU, IR sensor and servo motor, it helps to open and close garage door seamlessly. When user presses open button the garage door is opened and when pressed close it gets closed detecting object on the garage door.

It not only provides convenient to the user but also security as the access of web application is provided only to the users. Our system also prioritizes safety as it sends an alert through web application if any obstacle is detected while closing the door hence reducing the risks of accidents incorporating safety measures.

The user interface of the dedicated web application is also designed for easy monitoring with simple user interface. Therefore, our system not only meets the practical needs of users but also sets a standard in enhancing the safety and convenience of garage door operations.

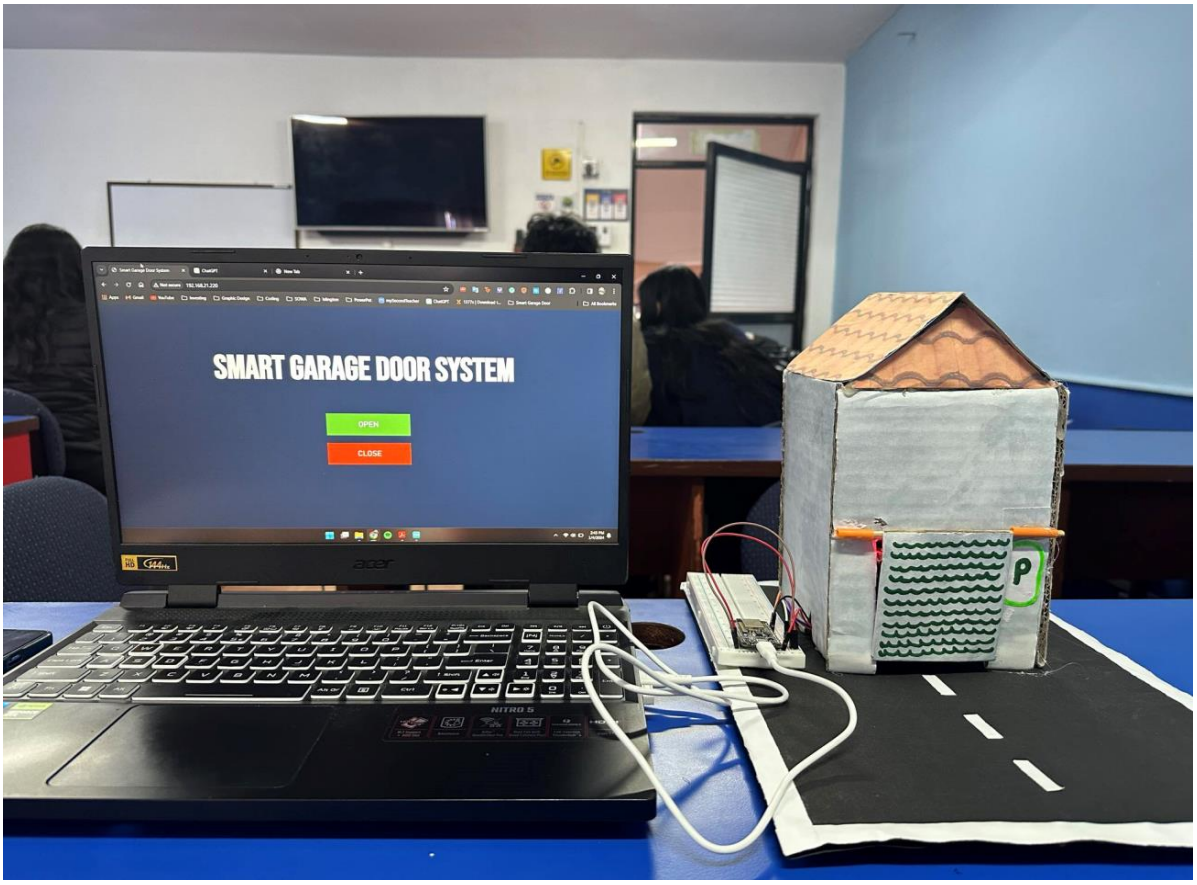

Figure 22: Smart Garage Door System.

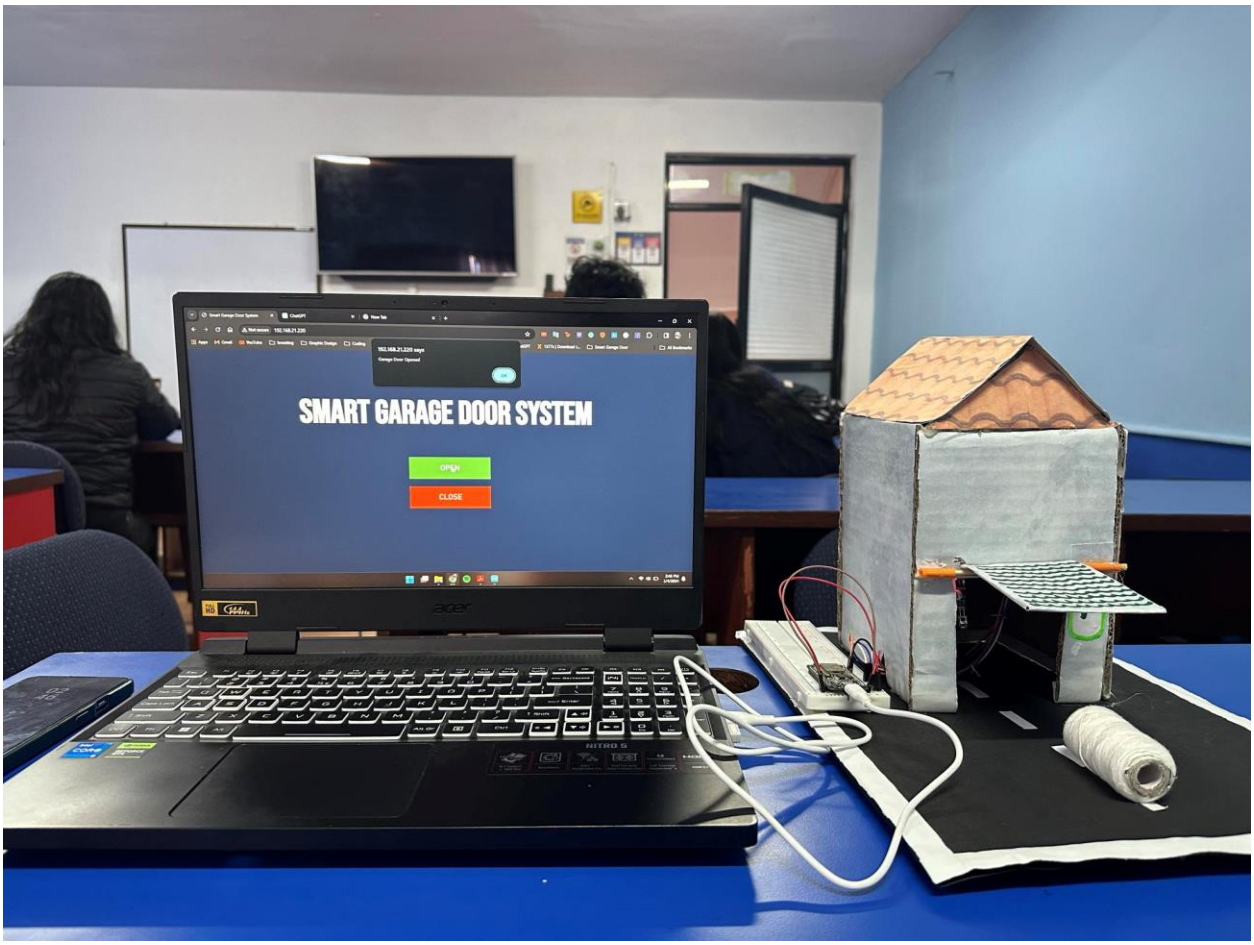

Figure 23: Smart Garage Door System 1.0

4.2. Findings

In this section, we present various test cases that are executed to validate successful completion of project. Each test cases were done to access different aspects of the smart garage door system ensuring its functionality.

4.2.1. Test 1

Test	1
OBJECTIVE	To test whether the sensors detect an object.
ACTION	• The sensor was incorporated into the system and configured correctly. • Finger was placed in front of IR as an object.
Expected Result	• The IR sensor was expected to detect an object and should emit Infrared light.
Actual Result	• The IR sensor detected an object, and it emitted Infrared light.
Conclusion	The test was successful.

Table 4: Testing IR sensor.

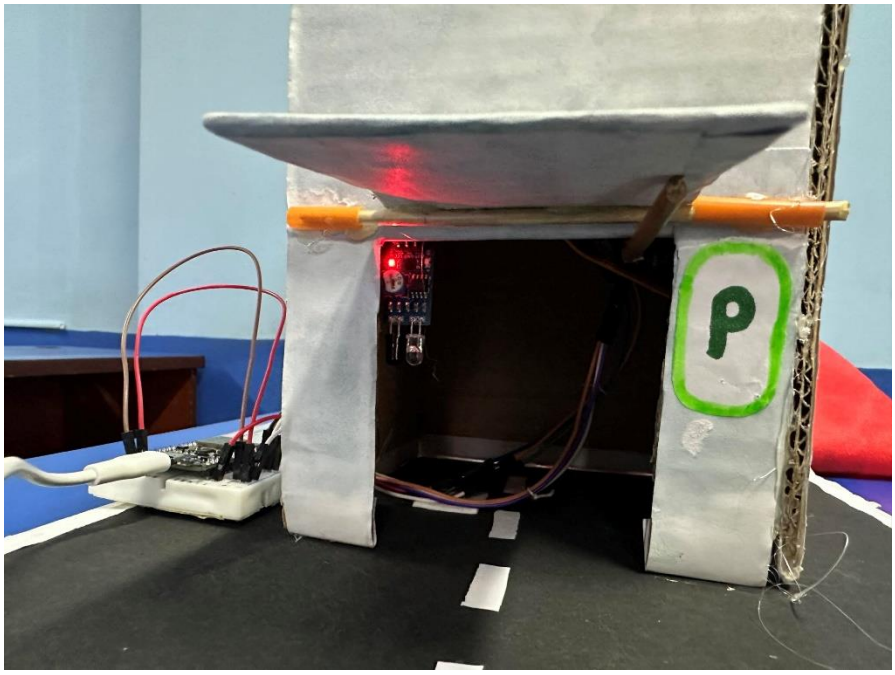

Figure 24: IR Sensor

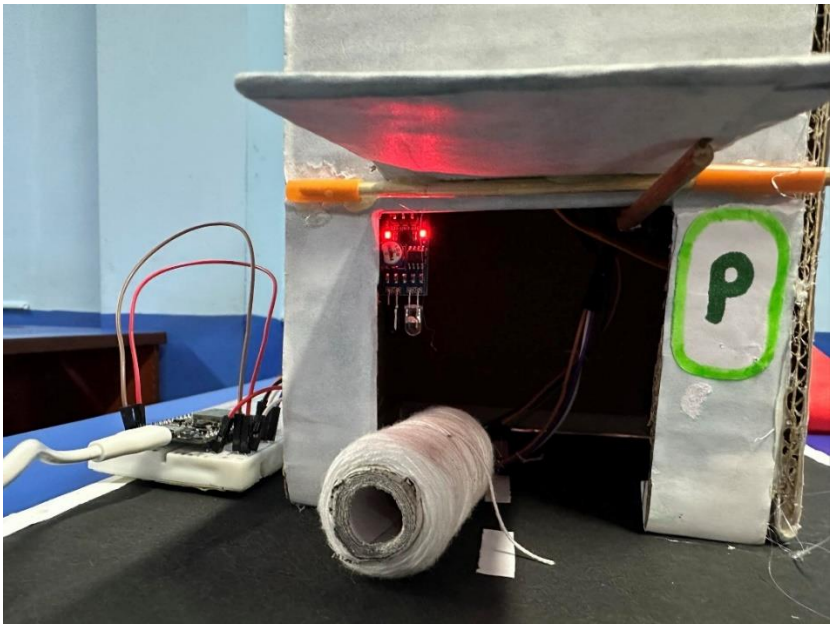

Figure 25: IR Sensor Detecting an object.

Here, an object was kept near IR sensor and therefore it detected it and emitted light.

4.2.2. Test 2

Test	2
OBJECTIVE	To detect whether the door opens and close through servo motor on button click.
ACTION	• The servo motor was incorporated into the system and configured correctly. • Web Application was opened on the smartphone. • Open button was pressed. • Close button was pressed.
Expected Result	The door was expected to open when open button was clicked and close when close button was clicked.
Actual Result	The door was opened when open button was clicked, and the door was closed when close button was closed.
Conclusion	The test was successful.

Table 5:Testing Opening and Closing of Garage Door

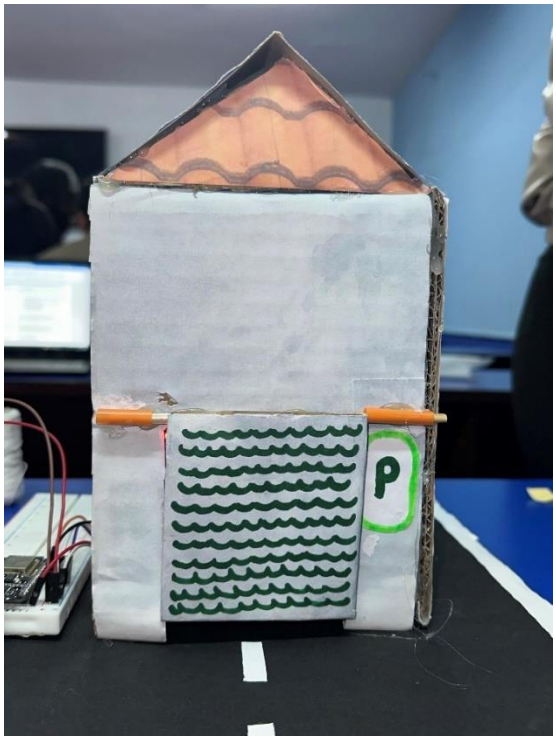

Figure 26: Closed Garage Door.

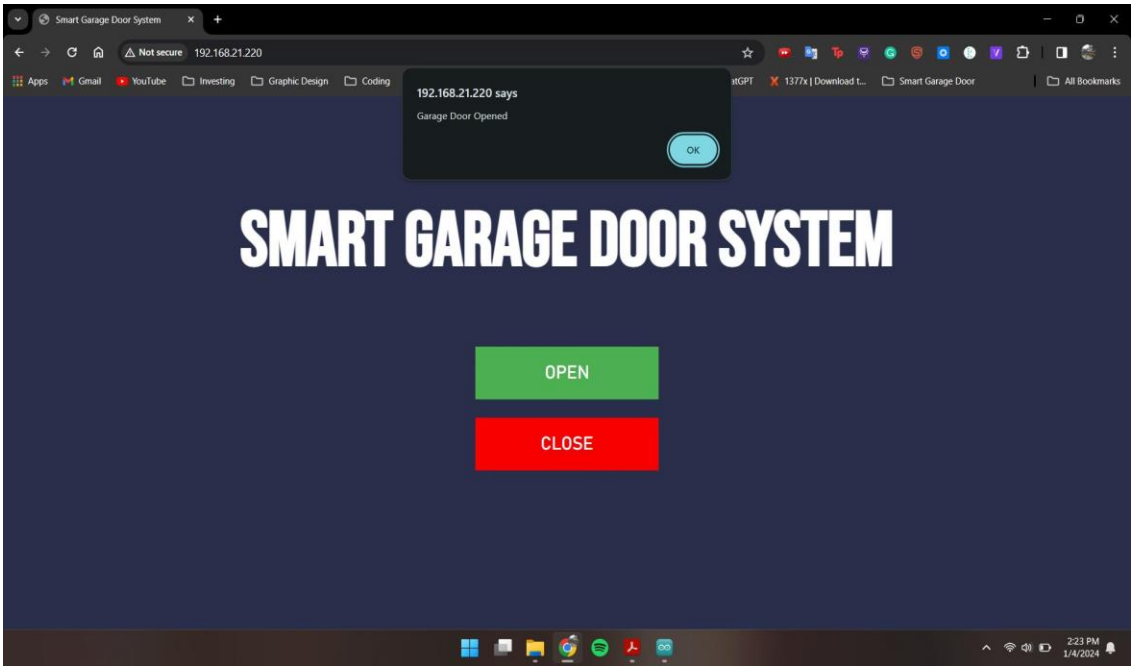

Figure 27:Garage door Opened Prompt

Here, First the door was closed but on entering open button it sent a prompt on web application and opened the door with the help of servo motor.

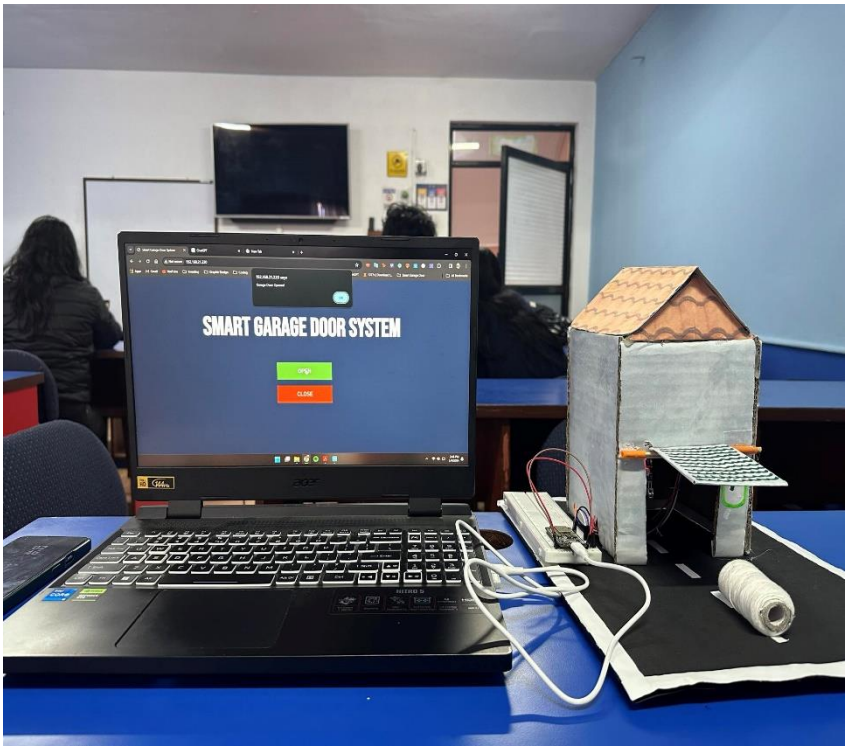

Figure 28: Opened Garage Door.

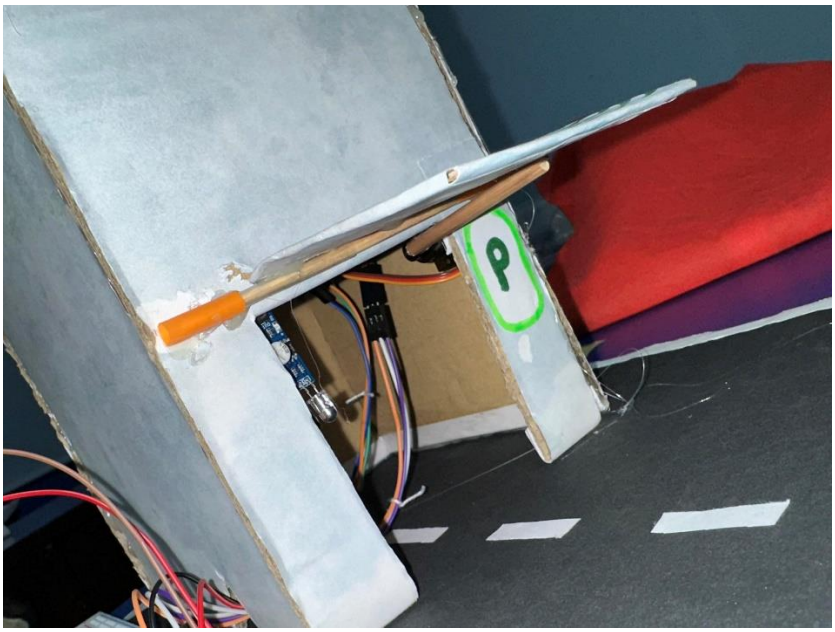

Figure 29: Closure look of opened garage door.

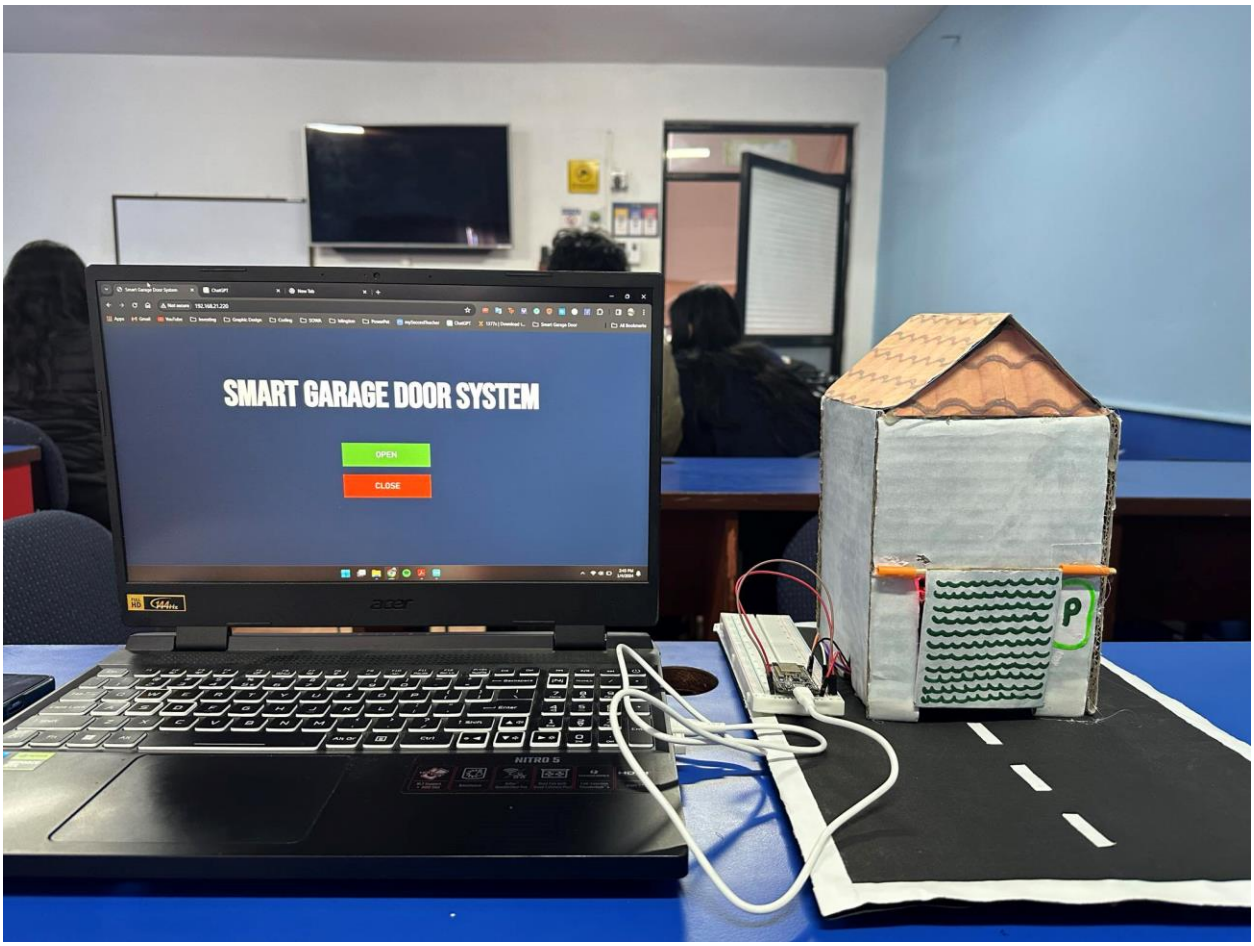

Figure 30: Closed Garage Door.

Likewise, on clicking close button from web application, our garage door got closed.

4.2.3. Test 3

Test	3
OBJECTIVE	To test whether sensors and actuator runs on Arduino IDE.
ACTION	• Code was written, verified, and uploaded on the nodeMCU by the help of Arduino IDE. • The sensor and relay module were incorporated into the system and setup correctly.
Expected Result	The code was expected to compile without any error and the functionality of sensor and relay module was expected to get displayed on Arduino IDE.
Actual Result	The code was compiled without any error and the functionality of sensor and relay module was displayed on Arduino IDE.
Conclusion	The test was successful.

Table 6: Testing Sensor and Actuator on Arduino IDE.

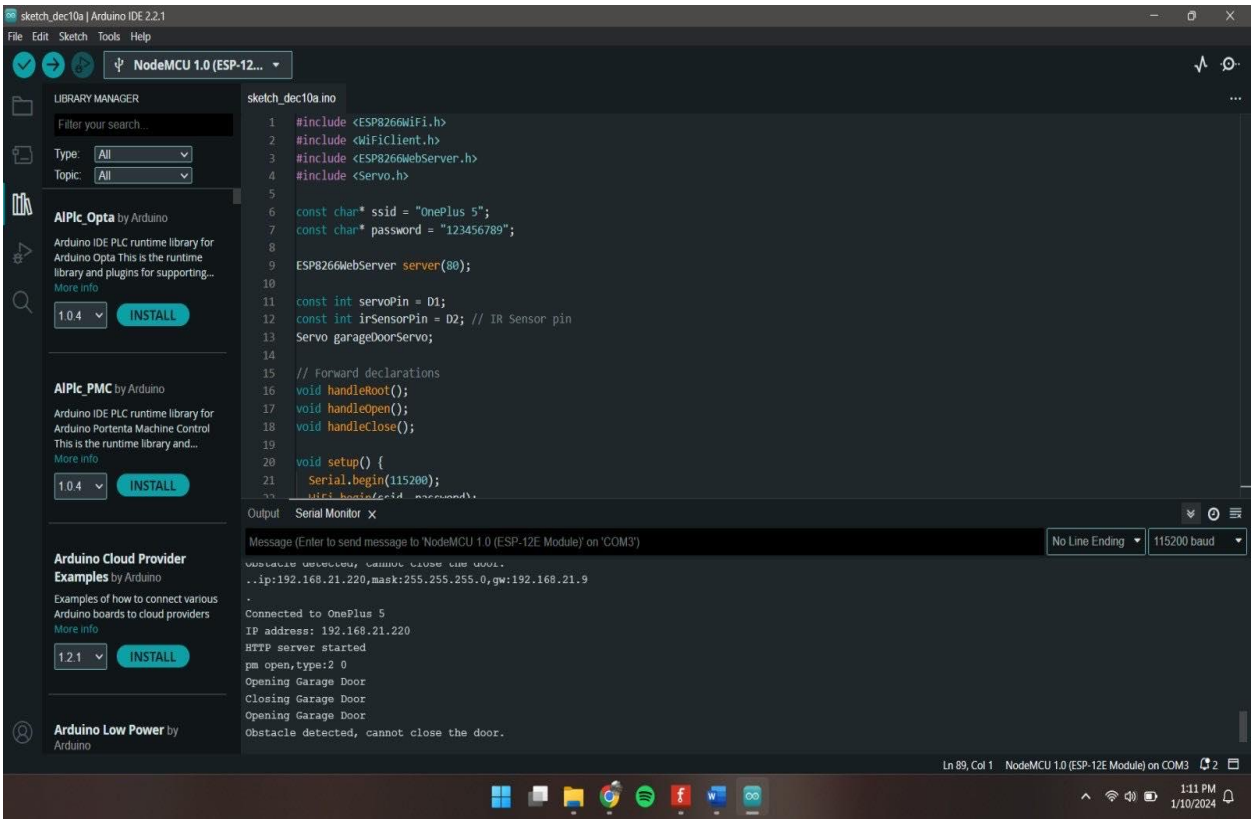

Figure 31: Working of Sensor and Actuator on Arduino IDE.

For this testing we opened Arduino IDE and navigated to serial monitor tool which clearly showed the working of IR sensor and our servo motor.

4.2.4. Test 4

Test	4
OBJECTIVE	To test whether the system sends an alert if sensor detects an obstacle.
ACTION	• Code was written, verified, and uploaded on the nodeMCU by the help of Arduino IDE. • Object was kept near IR sensor.
Expected Result	• Web app was expected to send an alert after IR sensor detects an object.
Actual Result	• Web app sent an alert after IR sensor detected an object.
Conclusion	The test was successful.

Table 7: Testing if system sends an alert on web application on detecting an object while closing the door.

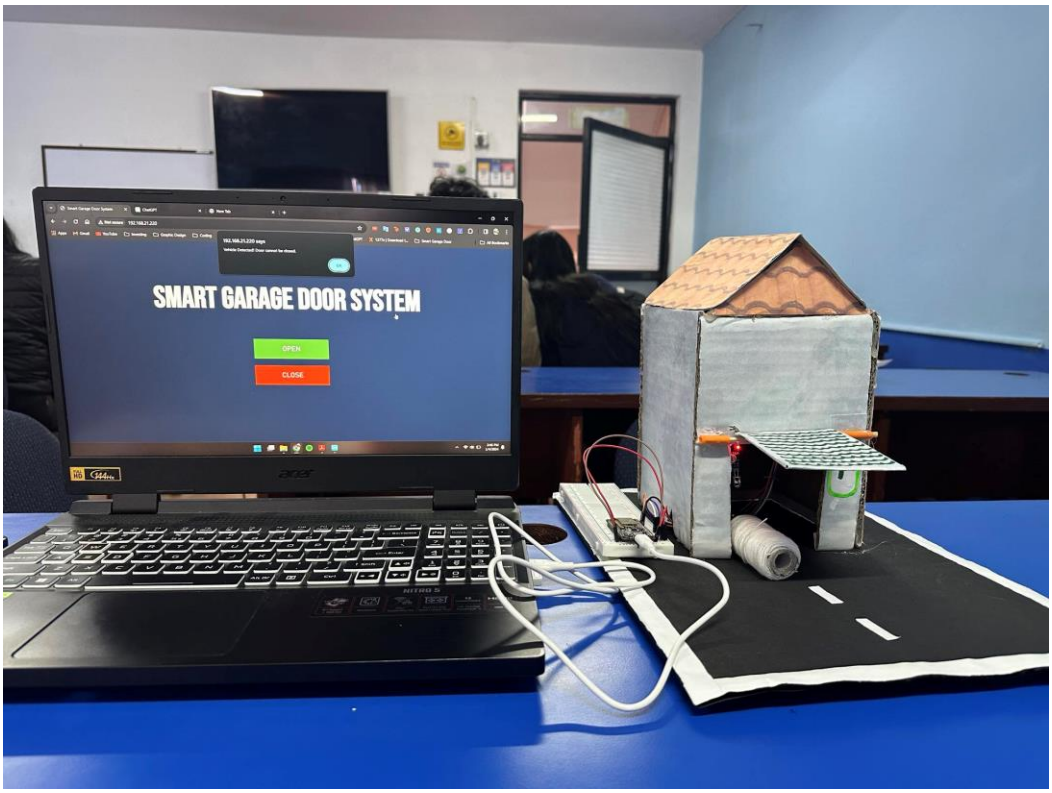

Figure 32: Object just below the garage door

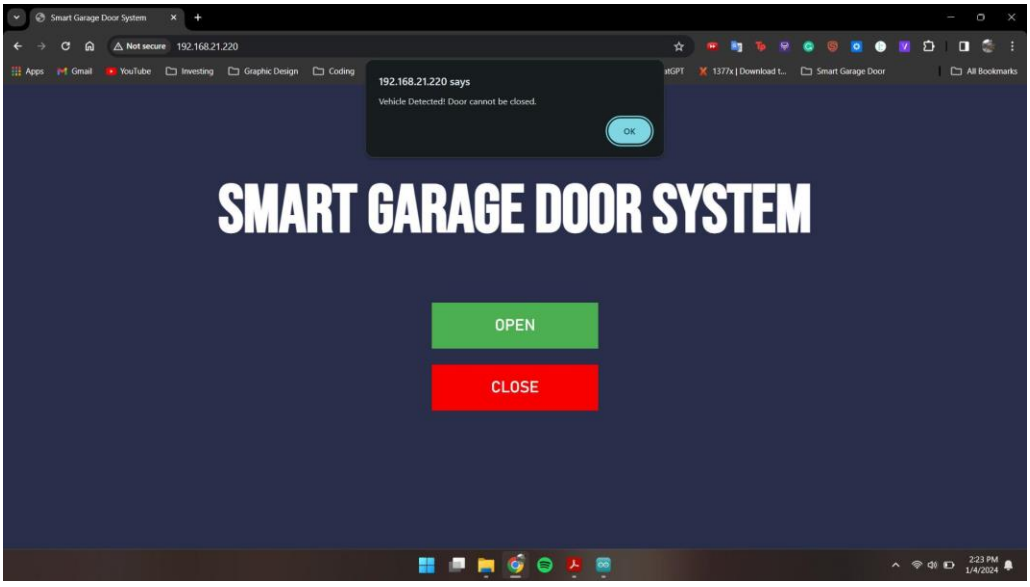

Figure 33: Prompt on Web App on detecting an object.

On this testing, we kept the object near IR sensor and checked if sends an alert on our web app while closing the door. And, it did send the alert detecting an obstacle.

4.2.5. Test 5

Test	5
OBJECTIVE	To test the Servo Motor.
ACTION	• The Servo Motor was incorporated into the system and configured correctly. • Open Button was pressed from Web Application to open the garage door.
Expected Result	The servo motor was expected to open the door.
Actual Result	The servo motor did not open the door.
Conclusion	The test was unsuccessful.

Table 8: To Test the Servo Motor.

Figure 34: Opened Prompt on Web Application but closed door.

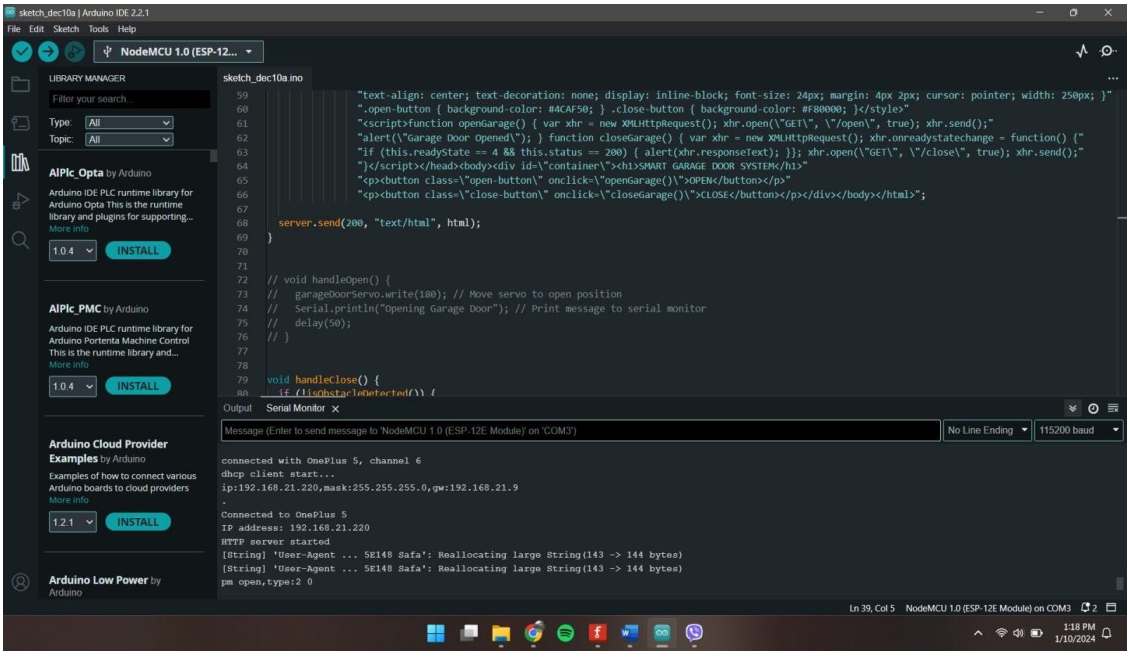

Figure 35: Problem detected on Servo Motor.

Here, due to some error Servo motor didn't work as it was supposed to. We clicked on the open button and the prompt saying "door opened" was also visible on web app but the door didn't open and also some problem was seen on serial monitor on our IDE.

5. Future Works

While our current system serves as a prototype, there are numerous possibilities for future enhancements and refinements. Potential enhancements include expanding sensor integration for a broader monitoring such as environmental and climate monitoring sensors. Similarly, further improvements may include integrating a dedicated mobile application for more user-friendly control.

Implementation of voice command technology can be other aspects of enhancements. Additionally, enabling cloud connectivity for a data storage and machine learning integration can make the system better by allowing the system to adapt and optimize its responses based on usage patterns. Likewise, implementation of energy efficient features, such as power saving modes when the system is in standby can be another great feature to on the system.

Similarly, incorporating an alert or messaging feature can aware users with real-time notifications and alerts keeping them informed about the status of their garage door and other events making this system dynamic and essential in-home automation and security.

6. Conclusion

In conclusion, our Cloud Computing, and IoT Module project, focused on developing an innovative IoT-based Smart Garage Door Opener, has been a significant learning experience for our team. The project involved utilizing a mobile web application to remotely control a garage door, enhancing convenience, and introducing a modern solution to a common everyday task.

This technological implementation showcases the vast potential of IoT systems in simplifying daily activities and solving practical problems. This project provides a hassle-free way to operate your garage door straight from your smartphone by just clicking open and close button. The possibilities for this technology are enormous.

The project, though challenging, was a valuable learning experience, which significantly enhanced our knowledge of IoT devices and their applications as we hand on experience with IoT tools. We are grateful to our team members for their cooperation and collective efforts, which were crucial in the successful completion of this innovative project.

7. References

Blog Master, 2015. *Why Kids and Garage Doors Don't Mix.* [Online] Available at: https://www.overheaddoorco.com/blog/blog/2015/06/15/why-kids-and-garage-doors-dont-mix
[Accessed 9 January 2023].

aadi3, 2023. *How to Open MS Word | 3 easy way to Start Microsoft Word? | by aadi3 | Medium.* [Online] Available at: https://medium.com/@seoteamaadiarun/how-to-open-ms-word-3-easy-way-to-start-microsoft-word-39b07458285a
[Accessed 9 January 2024].

Computer Hope, 2024 . *What is Draw.io?.* [Online] Available at: https://www.computerhope.com/jargon/d/drawio.htm
[Accessed 10 january 2024].

Edirisinghe, H., 2023. *User Guide to draw.io. Introduction | by Hasali Edirisinghe | Medium.* [Online] Available at: https://medium.com/@hasaliedirisinghe/user-guide-to-draw-io-6a1a4d7b5d33
[Accessed 9 January 2024].

fritzing, 2023. *Welcome to Fritzing.* [Online] Available at: https://fritzing.org/
[Accessed 10 january 2023].

geeksforgeeks, 2024. *What is a Flowchart and its Types?.* [Online] Available at: https://www.geeksforgeeks.org/what-is-a-flowchart-and-its-types/
[Accessed 10 january 2024].

GeeksOfGeeks, 2023. *GeeksOfGeeks.* [Online] Available at: https://www.geeksforgeeks.org/difference-between-system-architecture-and-software-architecture/
[Accessed 11 01 2024].

Indeed Editorial Team, 2022. *What Is a Block Diagram? Definition, Uses and Types.* [Online]
Available at: https://www.indeed.com/career-advice/career-development/what-is-block-diagram
[Accessed 10 January 2024].

MrSottong, 2023. *How to Install the Arduino IDE Software on Windows 10 #Arduino_1 : 8 Steps - Instructables.* [Online]
Available at: https://www.instructables.com/How-to-Install-the-Arduino-IDE-Application-on-Wind/
[Accessed 10 January 2024].

Perera, A., 2021. *Fritzing, all you need to know - Automatismos_Xl_Mundo.* [Online]
Available at: https://automatismosmundo.com/en/fritzing-all-you-need-to-know/
[Accessed 9 January 2024].

protoexpress, 2024. *What Is the Meaning of Schematic Diagram?.* [Online]
Available at: https://www.protoexpress.com/blog/whats-the-meaning-of-schematic-diagram/
[Accessed 10 january 2024].

S.Gillis, A., 2023. *internet of things (IoT).* [Online]
Available at: https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT
[Accessed 30 December 2023].

Salisbury Maryland, 2023. *BURGLARY PREVENTION.* [Online]
Available at: https://salisbury.md/departments/police/community-connection/burglary-prevention
[Accessed 1 January 2023].

Schlesinger, M., 2020. *An estimated 30k people are injured by garage doors each year -- don't let that be you!.* [Online]
Available at: https://www.cbs58.com/news/an-estimated-30k-people-are-injured-by-garage-doors-each-year-dont-let-that-be-you
[Accessed 1 January 2024].

The Zebra , 2023. *Burglary Statistics.* [Online] Available at: https://www.thezebra.com/resources/research/burglary-statistics/ [Accessed 1 January 2023].

8. Appendix

Hardware Components

ESP8266(NodeMCU)

ESP8266 is centered around the affordable ESP8266 System-on-a-Chip (SoC) created by Espressif Systems. This chip includes essential components such as CPU, RAM, Wi-Fi connectivity, and a modern Operating system with a Software Development Kit (Make-It.ca, 2023).

Digital IR Sensor

An infrared or IR sensor is an electronic device designed to detect and capture infrared radiation present in its environment. Infrared radiation belongs to the electromagnetic spectrum, ranging from the red end to microwaves, and is not visible to the human eye (Exertherm, 2023).

Jumper Wires

These wires are commonly used with breadboards and other prototyping tools to facilitate convenient circuit modifications (Hemmings, 2018).

Servo Motor

A servo motor is an electric motor that, in response to controller's input signal, can rotate or move with precision to a specific position, speed, or torque. Modern servo motors excel in delivering high performance and accuracy, making them suitable as primary drives in a wide range of applications (Electrical4U, 2023).

Breadboard

It is widely utilized in electronics projects, the breadboard serves as a prototype and functions as a foundational platform for constructing electronic circuits (Javatpoint, 2023).

Software Components

Arduino IDE

The Arduino Integrated Development Environment (IDE) includes a code editor, a message section, a text console, a toolbar with frequently used functions, and various menus.

Fritzing

Fritzing is an open-source hardware initiative that makes electronics accessible as a creative material for anyone. It offers many software tool, a community website and services in the spirit of Processing and Arduino and many more (fritzing, 2023).

Draw.io

Draw.io was designed by Seibert Media and is proprietary software for making diagrams and charts. They offer hundreds of visual elements and a wide variety of shapes to create a unique diagram or chart (Computer Hope, 2024).

Ms-Word

Microsoft is a word processor software used for documenting reports, documents etc. Here, Ms- Word is used for documenting reports. The system architecture diagram of our system is also made using Ms-Word.

Individual Contribution Plan

Below is the individual contribution of our team member who contributed to make our draft and ideas to this successful prototype.

Team Members	Roles	Contribution
Dikshya Sharma	**Report Preparation**: Being the leader, prepared the Introduction, compiled the Results & Findings including Testing also mentioned the Future Works, and created the Block Diagram. **Presentation**: Objectives, Result and Findings and Future Work. **Implementation**: Connecting Physical components to the prototype and performing final testing.	25%
Anushka Bhandari	**Report Preparation**: Prepared the System Architecture, worked with the Acknowledgment, Abstract and Conclusion and helped with the Individual Contribution Plan. **Presentation**: Software Requirement Analysis, Overview of the System and Conclusion. **Implementation**: Beautifying the System.	25%

Shuvam Rajbhandari	**Report Preparation**: Prepared the Requirement Analysis, created various Diagrams including the Circuit Diagram, also contributed to the development of the project and provided an Overview of the project. **Presentation**: Introduction and Development of a system. **Implementation**: Setting up Arduino IDE and writing code for proper functioning of the system.	25%
Ikshit Maharjan	**Report Preparation**: Prepared the Flowchart, contributed to the Development processes, and also worked with the Schematic Diagram and Software components. **Presentation**: Hardware Requirements and Planning and Design. **Implementation**: Connecting Physical Components and looking after the components and tracking it's working.	25%

Source Code

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
#include <Servo.h>

const char* ssid = "OnePlus 5";
const char* password = "123456789";

ESP8266WebServer server(80);

const int servoPin = D1;
const int irSensorPin = D2; // IR Sensor pin
Servo garageDoorServo;

// Forward declarations
void handleRoot();
void handleOpen();
void handleClose();

void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
```

```
  }
  Serial.println("");
  Serial.print("Connected to ");
  Serial.println(ssid);
  Serial.print("IP address: ");
  Serial.println(WiFi.localIP());

  pinMode(irSensorPin, INPUT); // Initialize the IR sensor pin as input
  garageDoorServo.attach(5, 500, 2400);
  garageDoorServo.write(0); // Set servo to default position
  delay(1000);

  server.on("/", handleRoot);
  server.on("/open", handleOpen);
  server.on("/close", handleClose);

  server.begin();
  Serial.println("HTTP server started");
}

void loop() {
  server.handleClient();
}

bool isObstacleDetected() {
  return digitalRead(irSensorPin) == LOW; // Assuming LOW means obstacle
detected
}
```

```
void handleRoot() {

  String html = "<!DOCTYPE html><html lang=\"en\"><head><meta
charset=\"UTF-8\">"

                "<meta name=\"viewport\" content=\"width=device-width,
initial-scale=1.0\">"

                "<title>Smart Garage Door System</title><style>body {
background-color: #282E4A; font-family: Bebas Neue; text-align: center; }"

                "#container { padding: 50px; }h1 { color: white; font-
size: 100px; }button { border: none; color: white; padding: 20px 50px;
font-family: Bahnschrift;"

                "text-align: center; text-decoration: none; display:
inline-block; font-size: 24px; margin: 4px 2px; cursor: pointer; width:
250px; }"

                ".open-button { background-color: #4CAF50; } .close-button
{ background-color: #F80000; }</style>"

                "<script>function openGarage() { var xhr = new
XMLHttpRequest(); xhr.open(\"GET\", \"/open\", true); xhr.send();"

                "alert(\"Garage Door Opened\"); } function closeGarage() {
var xhr = new XMLHttpRequest(); xhr.onreadystatechange = function() {"

                "if (this.readyState == 4 && this.status == 200) {
alert(xhr.responseText); }}; xhr.open(\"GET\", \"/close\", true);
xhr.send();"

                "}</script></head><body><div id=\"container\"><h1>SMART
GARAGE DOOR SYSTEM</h1>"

                "<p><button class=\"open-button\"
onclick=\"openGarage()\">OPEN</button></p>"

                "<p><button class=\"close-button\"
onclick=\"closeGarage()\">CLOSE</button></p></div></body></html>";



  server.send(200, "text/html", html);

}
```

```
void handleOpen() {

  garageDoorServo.write(180); // Move servo to open position

  Serial.println("Opening Garage Door"); // Print message to serial
monitor

  delay(50);

}




void handleClose() {

  if (!isObstacleDetected()) {

    garageDoorServo.write(0); // Move servo to closed position

    Serial.println("Closing Garage Door"); // Print message to serial
monitor

    delay(50);

  } else {

    Serial.println("Obstacle detected, cannot close the door."); // Print
obstacle detection message

    server.send(200, "text/plain", "Vehicle Detected! Door cannot be
closed.");

  }

}
```