

**Report on IoT-Based Smart House Door System**

DUE DATE: 31st May 2024

Prepared for: Proff Zac

PREPARED BY: **SELMA KAUNALENGA VELIKOSHI**

STUDENT NUMBER: **22011792**

Acknowledgements

Firstly, I would like to thank my Lord and my savior Jesus Christ for the strength and comfort He gave me throughout the procedure of this project. He will always be the first to thank in everything that I do.

It is also my great fortune that I got an opportunity to carry out this project work under the supervision of Prof. Zac Oyedokun in the Department of Mechanical, Industrial and Electrical Engineering of the Namibian University of Science and Technology (NUST). I express my sincere thanks and deepest sense of gratitude to our guide for his constant support, unparalleled guidance, and limitless encouragement.

Table of Contents

[INTRODUCTION 4](#_Toc168089440)

[OBJECTIVES OF THE PROJECT 4](#_Toc168089441)

[LITERATURE REVIEW 5](#_Toc168089442)

[Limits of the Study 6](#_Toc168089443)

[DESIGN METHODOLOGY 7](#_Toc168089444)

[HARDWARE DESIGN 7](#_Toc168089445)

[SOFTWARE COMPONENTS 8](#_Toc168089446)

[SYSTEM FLOWCHART 8](#_Toc168089447)

[TESTING AND RESULTS 9](#_Toc168089448)

[CONCLUSION AND FUTURE SCOPE 12](#_Toc168089449)

[REFERENCES 12](#_Toc168089450)

[APPENDIX 14](#_Toc168089451)

ABSTRACT

The evolution of smart locks represents a significant advancement in home security, aligning with the rise of smart home technologies such as Amazon’s Alexa and Google Home. This project aims to develop a smart door access control system using an RFID sensor, an ESP8266 microcontroller, and the Blynk IoT platform. The system eliminates the need for physical keys by granting access through RFID authentication, enhancing convenience and security. Upon successful authentication, the system first opens the gate and then uses an ultrasonic sensor to detect a person’s presence within a 10 cm range before opening the door. The ESP8266 controls the gate and door servos, with LEDs indicating access status. Additionally, a manual button inside the house allows for door operation. The Blynk app provides remote monitoring and control, including manual gate and door operation and real-time distance measurements from the ultrasonic sensor. This smart, cost-effective system offers a modern solution for secure and convenient home access control.

## INTRODUCTION

The advent of smart home technology has revolutionized the way we interact with and secure our living spaces. Traditional locks, while reliable, come with the inherent risk of lost or duplicated keys, prompting the need for more advanced and secure solutions. Smart locks, which can be controlled remotely and provide biometric access, offer a modern alternative. They not only enhance security but also integrate seamlessly with other smart home devices, contributing to a more connected and automated living environment.

This project focuses on developing a sophisticated door access control system using an RFID sensor, the ESP8266 microcontroller, and the Blynk IoT platform. By leveraging these technologies, the system provides a robust and user-friendly solution for managing home entry. Upon successful RFID authentication, the system initiates a sequence where the gate opens first, followed by the door, contingent on the presence of a person detected by an ultrasonic sensor. This ensures that the door only opens when it is necessary, enhancing both security and convenience. Additionally, the system features LEDs to indicate access status and a manual button for indoor door operation.

The integration of the Blynk application further elevates the system by enabling remote control and monitoring capabilities. Users can operate the gate and door from their smartphones, view real-time distance measurements, and receive status updates, making home access management more efficient and responsive. This project not only demonstrates the practical application of smart technology in home security but also highlights the potential for cost-effective, DIY solutions in the creation of intelligent living spaces.

## OBJECTIVES OF THE PROJECT

* To develop a smart door lock system using Arduino and ESP8266.
* To integrate an ultrasonic sensor for proximity-based access.
* To provide manual access control using a button.
* To control and monitor the system wirelessly through the Blynk application.

## LITERATURE REVIEW

Smart home automation systems leverage technology to address domestic challenges by offering control and administration of appliances and devices within a house through hardware and software integration [1]. This study builds upon this concept by proposing a low-cost, internet-operated automatic door lock system.

The proposed system adheres to the Internet of Things (IoT) paradigm, which emphasizes software and hardware integration for communication and task execution [1]. Our research involved developing a smart door lock with an accompanying IoT application. This involved utilizing a Wi-Fi enabled microcontroller and establishing communication between connected devices and user applications using the MQTT protocol. This enables remote locking and unlocking of the door, eliminating the need for users to be physically present.

Existing door lock systems encompass a range of technologies, including traditional mechanisms, RFID-based access control, touch-based interfaces, Bluetooth connectivity, and GSM cellular communication [2]. The ever-evolving technological landscape continuously introduces new possibilities for secure door lock systems. This study recognizes the paramount importance of security in today's world. By comprehensively exploring various relevant research topics, we aim to empower individuals to make informed decisions regarding the most suitable technology for their specific needs. This knowledge can be valuable for both new projects and for upgrading existing systems to incorporate the latest advancements in secure door lock technology.

According to [3], secure and convenient door lock systems are crucial for smart homes within the IoT landscape. Existing research explores image capture on unauthorized access attempts and remote access functionalities via mobile applications. This study builds upon these advancements by proposing an IoT-enabled door lock system that prioritizes both security and convenience. It incorporates features like image capture on unauthorized access and tamper detection alerts, alongside remote access, and access information monitoring through a mobile application.

## Limits of the Study

While this study lays a solid foundation for an RFID-based smart door lock with presence detection and remote access, it has limitations. Security assessments, user testing for diverse needs, and real-world implementation factors like cost and compatibility need further exploration. Additionally, the research could benefit from including the latest advancements in the field.

## DESIGN METHODOLOGY

The smart door access system design involves both hardware and software components. The hardware includes an Arduino, ESP8266 module, ultrasonic sensor, servos for door and gate control, and a manual button. The software is developed using the Arduino IDE and Blynk application for wireless control:

### HARDWARE DESIGN

Components: Arduino, ESP8266, ultrasonic sensor (HC-S R04), servos, LEDs, push button, power supply RFID sensor.

A close-up of a microchip

Description automatically generated A blue electronic device with a key chain

Description automatically generated

*Figure 1: NODE MCU Development board Figure 2 : RFID sensor*

A close-up of a device

Description automatically generated 

*Figure 3: Ultrasonic sensor Ffigure 4: servo motor*

*Figure 5: Arduino UNO board*

### SOFTWARE COMPONENTS

Programming: The Arduino IDE is used to program the control logic. The Blynk library integrates the project with the Blynk app for remote control.

Control Logic: The system reads the ultrasonic sensor data to check for proximity. Upon access approval (via the Blynk app or manual button), the servos are activated to unlock the door or gate.

### SYSTEM FLOWCHART

**START**

Is ESP8266 connected to WIFI

Access card scanned

Access approved

YES

NO

NO

YES

Open BLYNK app

ESP8266 receive command viaBLYNK

Arduino receives signal

Arduino sends signal to ESP8266

NO access

ESP8266 sends signal to the servo

ESP8266 activates servo motor to unlock

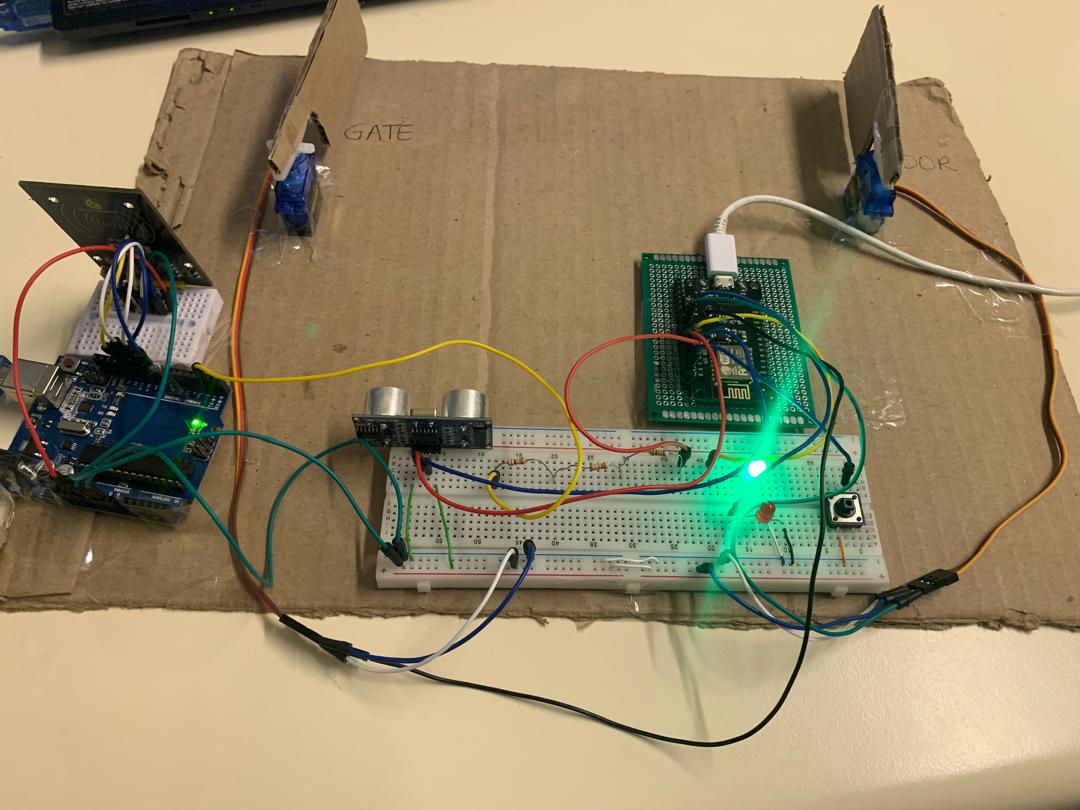
Servo door/gate unlocks

**END**

## TESTING AND RESULTS

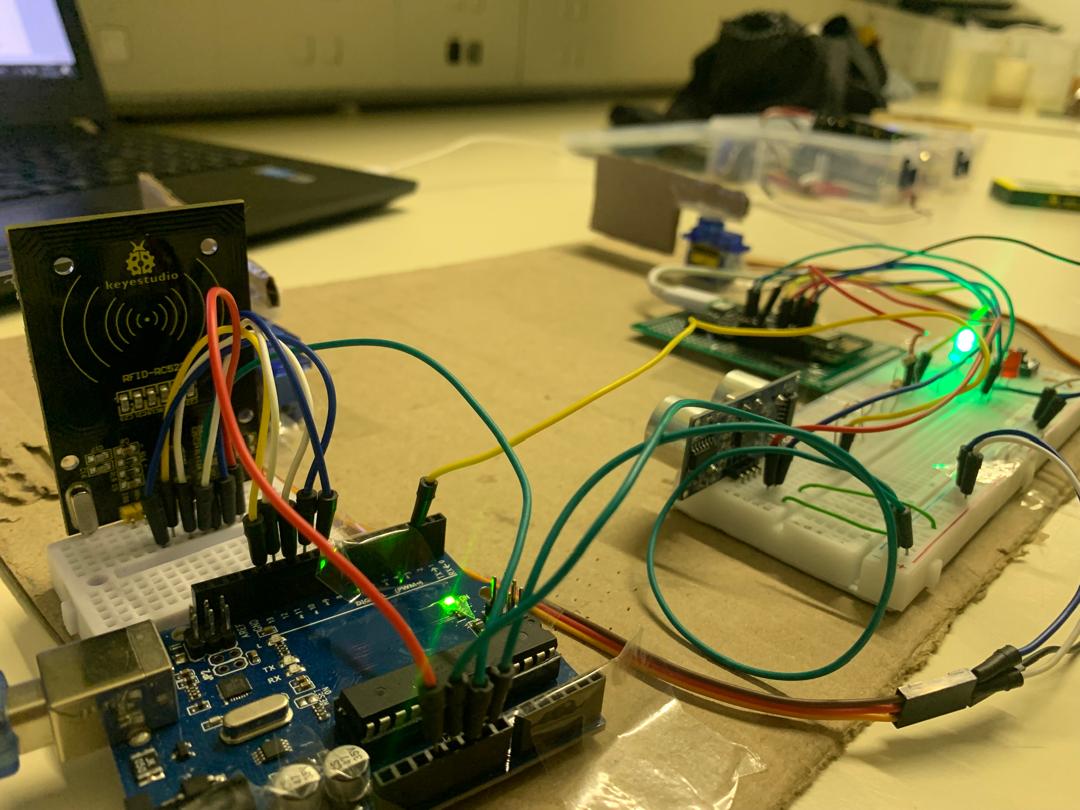
Individual components such as the ultrasonic sensor, servos, and ESP8266 were tested for proper functionality. The entire system was then assembled and tested to ensure the components work together as expected. The complete system was tested in various scenarios to validate its reliability and performance.

The smart door lock system successfully controlled access based on proximity detection and manual button input. The system operated smoothly with the Blynk app, providing real-time status updates and control.



*Figure 6: shows the full assembled project overview*

The system is designed to open the gate and door servos once access has been granted. The green LED goes on to show that access was granted otherwise, the red LED immediately goes on and shuts the whole system in response. This is assuming that an intruder is trying to walk in. the only way to open the door/gate is by either using the correct access user card or remotely opening it or by the push of a button.



*figure 7: closer angle of hardware setup*

A screenshot of a smart home system

Description automatically generated

*figure 8: shows the BLYNK console for the door access system.*

The door and gate can be accessed remotely by the press of a button. This is highly convenient in cases where visitors come over or when one has lost their Access card. However, its limitations are that it only operates when there is WIFI access.

## CONCLUSION AND FUTURE SCOPE

The development of the smart door lock system using Arduino and the ESP8266 module successfully integrates modern technology into home security. This project achieved its primary objective by creating a reliable and user-friendly smart lock system that enhances both convenience and security. The dual-mode operation proximity detection via an ultrasonic sensor and manual override with a button ensures versatile and secure access control. Additionally, wireless control through the Blynk application adds convenience by allowing remote monitoring and management.

Benefits:

* Enhanced security with remote monitoring
* Convenience of keyless entry
* Integration with smart home systems

Future Improvements:

* Integration with camera for visual verification
* Addition of fingerprint sensor for enhanced security

## 

## REFERENCES

[1] Taslim, Hazeem & Md Lazam, Nor Azlinah & Mohd Yahya, Nor. (2021). Development of Smart Home Door Lock System. 10.1007/978-3-030-70917-4\_13. <https://www.researchgate.net/publication/349959106_Development_of_Smart_Home_Door_Lock_System/citation/download>

[2] Yu, Yuan-Chih. (2018). A practical digital door lock for smart home. 1-2. 10.1109/ICCE.2018.8326305.

[3] Ha, Ilkyu. (2015). Security and Usability Improvement on a Digital Door Lock System based on Internet of Things. International Journal of Security and Its Applications. 9. 45-54. 10.14257/ijsia.2015.9.8.05. <https://www.researchgate.net/publication/283703807_Security_and_Usability_Improvement_on_a_Digital_Door_Lock_System_based_on_Internet_of_Things/citation/download>

## APPENDIX

Code snippet:

#define BLYNK\_TEMPLATE\_ID "TMPL26ONk-mm1"

#define BLYNK\_TEMPLATE\_NAME "Quickstart Template"

#include <Servo.h>

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

#define BLYNK\_PRINT Serial

// WiFi credentials

char auth[] = "Z3W1fwcUVUiNjdeEx6C0GrMynfm-WZb\_";  // Blynk Auth Token

char ssid[] = "selma";  // Your WiFi SSID

char pass[] = "123456789";  // Your WiFi Password

// Pin definitions for ESP8266

#define ACCESS\_PIN D1 // Digital input pin for access status from Arduino

#define TRIG\_PIN D2

#define ECHO\_PIN D3

#define GATE\_SERVO\_PIN D5

#define DOOR\_SERVO\_PIN D6

#define RED\_LED D7

#define GREEN\_LED D4

#define BUTTON\_PIN D0 // Digital input pin for the manual button

Servo gateServo;

Servo doorServo;

// Blynk virtual pin definitions

#define V0 0

#define V1 1

#define V\_DISTANCE V2 // Virtual pin for distance display

#define V\_ACCESS\_LED V3 // Virtual pin for access granted LED

bool accessGranted = false;

void setup() {

  Serial.begin(9600); // Initialize serial communications

  pinMode(ACCESS\_PIN, INPUT); // Set access status pin as input

  pinMode(RED\_LED, OUTPUT); // Set red LED pin as output

  pinMode(GREEN\_LED, OUTPUT); // Set green LED pin as output

  pinMode(BUTTON\_PIN, INPUT\_PULLUP); // Set button pin as input with internal pull-up

  gateServo.attach(GATE\_SERVO\_PIN);

  doorServo.attach(DOOR\_SERVO\_PIN);

  pinMode(TRIG\_PIN, OUTPUT);

  pinMode(ECHO\_PIN, INPUT);

  // Initialize servos to closed position

  gateServo.write(0);

  doorServo.write(0);

  // Connect to WiFi and Blynk

  Blynk.begin(auth, ssid, pass);

  Serial.println("Setup complete. Waiting for card scan...");

}

void loop() {

  Blynk.run(); // Run Blynk

  // Read access status from Arduino

  int accessStatus = digitalRead(ACCESS\_PIN);

  int buttonState = digitalRead(BUTTON\_PIN);

  if (accessStatus == HIGH && !accessGranted) {

    accessGranted = true;

    Serial.println("Access granted. Opening gate...");

    openGate();

    digitalWrite(GREEN\_LED, HIGH); // Turn on green LED

    digitalWrite(RED\_LED, LOW); // Turn off red LED

    Blynk.virtualWrite(V\_ACCESS\_LED, 255); // Turn on virtual LED in Blynk app

  } else if (accessStatus == LOW && accessGranted) {

    accessGranted = false;

    Serial.println("Access denied or reset. Closing gate and door.");

    closeGate();

    closeDoor();

    digitalWrite(GREEN\_LED, LOW); // Turn off green LED

    digitalWrite(RED\_LED, HIGH); // Turn on red LED

    Blynk.virtualWrite(V\_ACCESS\_LED, 0); // Turn off virtual LED in Blynk app

  }

  if (accessGranted) {

    // Check ultrasonic sensor and display distance

    int distance = measureDistance();

    Serial.print("Distance: ");

    Serial.print(distance);

    Serial.println(" cm");

    Blynk.virtualWrite(V\_DISTANCE, distance); // Send distance to Blynk app

    // Open door only if the button is pressed and distance is within 10cm

    if (buttonState == HIGH && distance <= 10) {

      Serial.println("Button pressed and person detected within 10cm. Opening door...");

      openDoor();

    }

  }

  delay(100); // Small delay to prevent excessive looping

}

void openGate() {

  gateServo.write(90); // Open gate

  delay(5000); // Wait for 10 seconds

  gateServo.write(0); // Stop gate

  delay(2000); // Wait for 3 seconds before opening the door

}

void closeGate() {

  gateServo.write(0); // Close gate

}

void openDoor() {

  doorServo.write(90); // Open door

  delay(7000); // Wait for 10 seconds

  doorServo.write(0); // Stop door

}

void closeDoor() {

  doorServo.write(0); // Close door

}

int measureDistance() {

  // Send a pulse to the trigger pin

  digitalWrite(TRIG\_PIN, LOW);

  delayMicroseconds(2);

  digitalWrite(TRIG\_PIN, HIGH);

  delayMicroseconds(10);

  digitalWrite(TRIG\_PIN, LOW);

  // Measure the length of time for the echo to return

  long duration = pulseIn(ECHO\_PIN, HIGH);

  // Calculate the distance (in cm)

  int distance = duration \* 0.034 / 2;

  return distance;

}

// Blynk function to manually open and close the gate

BLYNK\_WRITE(V0) {

  int pinValue = param.asInt();

  if (pinValue == 1) {

    openGate();

  } else {

    closeGate();

  }

}

// Blynk function to manually open and close the door

BLYNK\_WRITE(V1) {

  int pinValue = param.asInt();

  if (pinValue == 1) {

    openDoor();

  } else {

    closeDoor();

  }

}