IOT BASED RFID SMART DOOR LOCK SYSTEM USING NODE MCU AND BLYNK

A PROJECT REPORT

Submitted by

RAUNAK MALLICK

ROLL – 10071021038

REG. – 211000571010038

Supervised by

DR. JADAV CHANDRA DAS

in partial fulfillment for the award of the degree

of

MASTERS OF COMPUTER APPLICATION

IN

INFORMATION TECHNOLOGY

YEAR: 2023

MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY, WEST BENGAL

BONAFIDE CERTIFICATE

Certified that this project report **"IoT based RFID Smart Door Lock System using Node MCU and Blynk Application"** is the bonafede work of **Raunak Mallick** [Roll: 10071021038] who carried out the project work under my supervision.

SIGNATURE

DR. DEBASIS GIRI

HEAD OF THE DEPARTMENT

Department of Information Science & Technology

Maulana Abul Kalam Azad University of Technology, West Bengal

SIGNATURE

DR. JADAV CHANDRA DAS

SUPERVISOR

Assistant Professor
Department of Information Science & Technology

Maulana Abul Kalam Azad University of Technology, West Bengal

SIGNATURE

External Examiner:
Date:

// ACKNOWLEDGEMENT

I wish to express my whole heartedly and deep gratitude to **Dr. Jadav Chandra Das**, Assistant Professor of Maulana Abul Kalam Azad University of Technology, for his valuable guidance, constant support and encouragement throughout my project work.

This has been a precious opportunity for me not only to gain knowledge and skill but also to learn much more about approaches, attitudes towards work and interpersonal relationship.

I am sincerely thankful and indebted to **Dr. Debasis Giri**, Head of the Department in Department of Information Science & Technology, for their constant encouragement and continuous valuable suggestions throughout my project work.

With regards,

Raunak Mallick

Roll – 10071021038
Registration No. – 211000571010038
Masters of Computer Application

Date – 22/05/2023

PLAGIARISM REPORT

Final Sem Docx - Copy.docx

As of: May 19, 2023 10:59:23 AM
2,169 words - 34 matches - 23 sources

Similarity Index
19%

Mode: Similarity Report

paper text:

CHAPTER 1: INTRODUCTION It is necessary to adopt IoT-based smart security systems since they increase security by combining AI to improve the functionality of gadgets like CCTV cameras, smart lights, doorbells, and fire sensors. You can feel secure while you're abroad thanks to intelligent security solutions. For instance, intelligent doorbells may recognize visitors and initiate communication with them prior to opening your front gate. IoT devices control the monitoring in and around the house and, if the doors have smart locks, also keep track of who has access to them. IoT also contributes significantly to establishing security. IoT also contributes significantly to establishing security. The goal of an IoT-based smart security surveillance system is to increase home security. The goal of the research is to create the smartest doorbell possible with increased security, adaptability, and connectivity. 1.1. Objective and Motivation about this project: The project's goal is to build an intelligent, multi-alert smart IoT-based door lock security system with the least amount of user intervention feasible in order to increase a door's security.

The goal of the project **is to** create **an IoT-based smart** locking **system** 14

that can be remotely controlled via a smartphone app and that can send email notifications when

a door is opened or closed. The 10

technique is applicable to automated devices without human operators, such as KIOSKs, vending machines, and ATMs.

The physical key used in unmanned automation equipment like KIOSKs, vending machines, and ATMs 4

raises safety concerns, prompting

the development of the smart door locking **system based on** better security strategy. **The** 18

ABSTRACT

The Internet of Things (IoT) now has smart buildings as its foundation. Utilizing the internet is boosted by tying together the appliances in houses to make the spaces more cozy, safe, and enjoyable. Old-fashioned home locks are becoming obsolete. Most homes and businesses now utilize smart door locks. Smart door locks make homes more secure while seeming more upscale. You don't have to physically open the door when using a smart door lock.

In addition, smart door locks differ from conventional digital doors in that they can only be controlled wirelessly. Because the smart door lock can remotely lock the door, time is not wasted looking for the door lock. The suggested method handles the door lock mechanism, a security feature of smart home technology.

By enabling the owner to monitor the buildings using a Smart phone-controlled, Wi-Fi-connected system employing Node MCU ESP8266, the door lock system decides the security. For this project, a microcontroller is used to build experiments and develop security flaws. By installing the created Android application on devices like tablets, smartphones, computers, etc. and entering login credentials, users may open or close the door lock.

Keywords: Door lock, microcontroller, internet of things, security.

COMPARATIVE STUDIES

Paper Name	Key Features	Drawbacks	Reference
1. **Smart Door Locking System using IOT**	• Bluetooth HC05 • Broken Lock • Arduino Nano • Servo Motors • Wires 10 cm • Battery 5v	**Depends on the Bluetooth (HC05) module.** Can't access the door if you are out of the Bluetooth range (~30ft).	https://www.researchgate.net/publication/341508373_Smart_Door_Locking_System_using_IoT
2. **Smart Door Lock and Lightning system using IOT**	✓ PIC MC ✓ LCD ✓ Keypad ✓ Motor Driver ✓ Relay Driver ✓ Door Motor ✓ Relay ✓ GPRS Module	**The whole system based on GPRS module.** If there any cases of connection failure through GPRS and unable to read Server file then system can't be authenticated.	https://www.ijcsit.com/docs/Volume%209/vol9issue5/ijcsit2018090503.pdf
3. **IOT enabled Smart Door Lock**	▪ Raspberry Pi ▪ Relay Driver Circuit ▪ Camera Module ▪ PIR Motion Sensor ▪ Display ▪ Loud Speaker ▪ Electromagnetic Door Lock	**Fully depended on Cloud Server.** Any cases of server issue can affect the whole module functionality. Can't access the door at that period of time.	https://ijarsct.co.in/Paper2767.pdf

Comparison to my project named "*IoT based RFID Smart Door Lock System using Node MCU & Blynk Application*":

The proposed system of my project is not depending on a single module's failure. If internet is not available for some moment but the system will be fully active or online. Therefore, if Blynk not available due to internet errors, but the RFID mechanism will be online and guests can be access it using RFID cards.

Here you can see all of these project modules (1,2 & 3) are depended on a particular module or server function. But my proposed system is these kinds of function independent.

CONTENTS

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

It is necessary to adopt IoT-based smart security systems since they increase security by combining AI to improve the functionality of gadgets like CCTV cameras, smart lights, doorbells, and fire sensors. You can feel secure while you're abroad thanks to intelligent security solutions. For instance, intelligent doorbells may recognize visitors and initiate communication with them prior to opening your front gate. IoT devices control the monitoring in and around the house and, if the doors have smart locks, also keep track of who has access to them. IoT also contributes significantly to establishing security.

IoT also contributes significantly to establishing security. The goal of an IoT-based smart security surveillance system is to increase home security. The goal of the research is to create the smartest doorbell possible with increased security, adaptability, and connectivity.

1.1. Objective and Motivation about this project:

The project's goal is to build an intelligent, multi-alert smart IoT-based door lock security system with the least amount of user intervention feasible in order to increase a door's security. The goal of the project is to create an IoT-based smart locking system that can be remotely controlled via a smartphone app and that can send email notifications when a door is opened or closed.

The technique is applicable to automated devices without human operators, such as KIOSKs, vending machines, and ATMs. The physical key used in unmanned automation equipment like KIOSKs, vending machines, and ATMs raises safety concerns, prompting the development of the smart door locking system based on better security strategy. The smart lock makes use of Internet of Things (IoT)-enabled sensors to run keyless entry systems that let users open doors remotely using a smartphone or other internet-connected device.

1.2. Why IoT based Smart Door Security system is needed?

Because it offers better protection for the safety risk caused by the physical key used in unmanned automation equipment like KIOSKs, vending machines, and ATMs, IoT-based Smart Door protection System is required. The suggested method is intended to assist a person in protecting his or her home from theft; if a door is broken or the incorrect password is entered, it immediately notifies the concerned homeowner. Since it gives you complete control over everything inside and outside of your home, the smart home security system is an excellent tool for protecting your property. With the development of technology, smart door locking systems have gotten increasingly sophisticated. Even if an automatic password-based door lock system offers a more secure manner of locking and unlocking the system, it is exceedingly risky to touch the keypads, which are often used by various people, due to the current COVID problem.

Therefore, RFID tags and Node MCU are utilized in this project. The user's access is discovered and verified by the RFID card reader. Data from the card is read by the reader when the card is in close proximity, and it is sent to Node MCU.

1.3. Organization of the report:

Next chapter is all about the Literature Reviews and the previous works on Smart Door lock system and RFID cards authentication. Chapter 3 is all about System Overviews which dictates about the hardware and software tools which is needed for this project. Chapter 4 is all about the System Design which includes block diagrams, circuit diagram and the algorithm of work functionality. The Experimental Results are included in the chapter 5 whereas shown the snapshots of the project, google sheet data sharing and IDE serial monitor output. Chapter 6 deals with the conclusion and future work part. Chapter 7 is about the bibliography section and in the last section the Appendix are included.

CHAPTER 2: LITERATURE REVIEW

Making things simple and intelligent is the primary goal of the majority of sectors' efforts in the past several years in the areas of machine learning, artificial intelligence, big data analytics, and IoT-based projects. These become necessary for digitizing with numerous security technologies as a result of making locks in our daily lives smarter and introducing the locks that can be moved by a stepper motor and require a digital number pad for user input, in order to operate all of these gadgets, it must also include an infrared or Bluetooth module. Face recognition door locks differ significantly from conventional door locks in that there is no requirement for a stepper motor because the application recognizes faces using photographs that have been stored in the application programmed. Incompatible stepper motor and driver components from the previous models have been removed. We have included more advanced and unmatched facial detection features as a point of access to open or close the door. Relay module and solenoid lock are combined to unlock the door in this special and user-friendly way.

There are many fraud schemes and thefts going on right now, which have become major problems for everyone. Even if we have locks on the doors, anyone with a key can get inside and enter, but if we have a password lock, they can also get ahold of it using today's technology. In these fields, facial recognition doors are made safer since we may utilize biometric face recognition to prevent unauthorized entry. Face recognition is crucial for security and privacy in the modern era. These face recognition systems recognized individuals and, if the face matches a database entry, instantly unlocks the door; otherwise, an alert message is delivered to the owner.

2.1. Related work on Smart Door Lock system:

A lock and a real key are the most widely used locking and unlocking mechanism for doors. The whole procedure is mechanical. The complete locking mechanism needs to be replaced if the key is lost, forgotten, or stolen. When it comes to large corporations, where employees are required to carry many keys for various entrances, the issue with physical keys becomes more serious. In addition to adding weight, all the keys increase the risk of loss. RFID (Radio-Frequency Identification) is being employed as a replacement for physical keys. As pass keys, RFID cards are employed. The RFID card reader device has been placed close to the door. The reader detects the card's radio frequency when it is brought close, which allows it to confirm the key. The device can pair with many cards. However, they are once more susceptible to theft or loss. Additionally, it does not accomplish the goal of not carrying a key.

CHAPTER 3: SYSTEM OVERVIEW

3.1. Hardware Tools:

Node MCU ESP8266	Based on Espressif Systems' ESP8266 Wi-Fi SoC, NodeMCU is open-source firmware. It is a free and open-source IoT platform that at first comprised hardware based on the ESP-12 module and firmware that runs on Espressif Systems' ESP8266 Wi-Fi SoC. The ESP-12E module, which houses the ESP8266 chip with Tensilica Xtensa 32-bit LX106 RISC microprocessor, is included with the NodeMCU ESP8266 development board.	
RFID MFRC 522	Based on the MFRC522 controller from NXP semiconductor, the MFRC522 is a 13.56MHz RFID module. It is intended to communicate with RFID tags (ISO 14443A standard tags) and generate an electromagnetic field at 13.56 MHz. With a maximum data rate of 10 Mbps, the reader may interface with a microcontroller using a 4-pin SPI connector. It is frequently used in applications for person/object identification, such as attendance systems.	

Single Channel 5v Relay Module	A single channel 5V relay module often functions as a relay interface board that may be directly controlled by a variety of microcontrollers, including Arduino, AVR, PIC, ARM, 8051, and others. A coil and two contacts, such as ordinarily open (NO) and normally closed (NC), are typically included. DC energizes the relay coil so that contact switches can be opened or closed. Solenoids, motors, lights, fans, and other AC and DC appliances are among those it can control.	
Solenoid Lock	An electromagnetic lock called a 12V solenoid is used to lock things like vending machines, storage shelves, filing cabinets, and more. It boasts a long lifespan, consistent performance, energy efficiency, anti-theft features, and a shockproof design. The lock operates when the power is turned off and unlocks when it is turned back on. It has a long lifespan and is reliable, strong, and energy-efficient.	
LM2596 Buck Converter	A voltage regulator, the LM2596 Buck Converter is mostly used to step down the voltage or to drive loads that draw less than 3A. The buck converter, also known as a DC-to-DC power converter, steps down the voltage from the input source to the output load. During this voltage step-	

	down procedure, the current increases. It features outstanding load and line regulation and comes in set output voltages like 3.3V, 5V, and 12V. Additionally, it has a customized output version that allows you to adjust the output voltage to meet your needs.	
Buzzer Module	An electronic gadget used to produce sound is referred to as an IoT buzzer module. It is a particular kind of transducer that transforms electrical energy into sound. It is utilized in many different applications, including timers, alarms, and other devices that call for an audio signal. In order to provide sound alerts based on the input received from sensors or other devices, the buzzer module can be linked to an IoT device like an Arduino or Raspberry Pi.	

3.2. Software Tools:

- **Arduino IDE –**
  The open-source Arduino IDE software is used to programmed Arduino boards. It is a cross-platform application that runs on Windows, macOS, and Linux. Use the software to create and upload code to the Arduino board. It offers a user-friendly interface that makes learning to programmed simple for newcomers. The programmed has a bootloader that enables the uploading of the code to the board, a code editor, and a compiler. Programming languages C and C++ are supported by the Arduino IDE.
- **Blynk IoT Application –**
  An IoT device can be managed via the mobile app Blynk. It is a platform that enables users to create unique IoT applications without having to know how to code. The ESP8266, Raspberry Pi, Arduino, and other IoT devices may all be managed through the

Blynk app. With the app's user-friendly design, getting started with IoT is simple even for beginners. The software comes with a number of widgets that can be used to control IoT devices, including buttons, sliders, and graphs. Other features of the app include data recording, email alerts, and notification features.

The platform consists of three main parts:

- **Blynk App** - allows you to use the many widgets we offer to create stunning interfaces for your projects.
- **Blynk Server** - in charge of overseeing all hardware-to-smartphone communications. You can host your private Blynk server locally or utilize our Blynk Cloud. It can even be started on a Raspberry Pi, is open-source, and has no trouble supporting thousands of devices.
- **Blynk Libraries** - Enable communication with the server and handle all incoming and outgoing commands for all popular hardware platforms.

✓ ***Functionality of Blynk App in this project –***

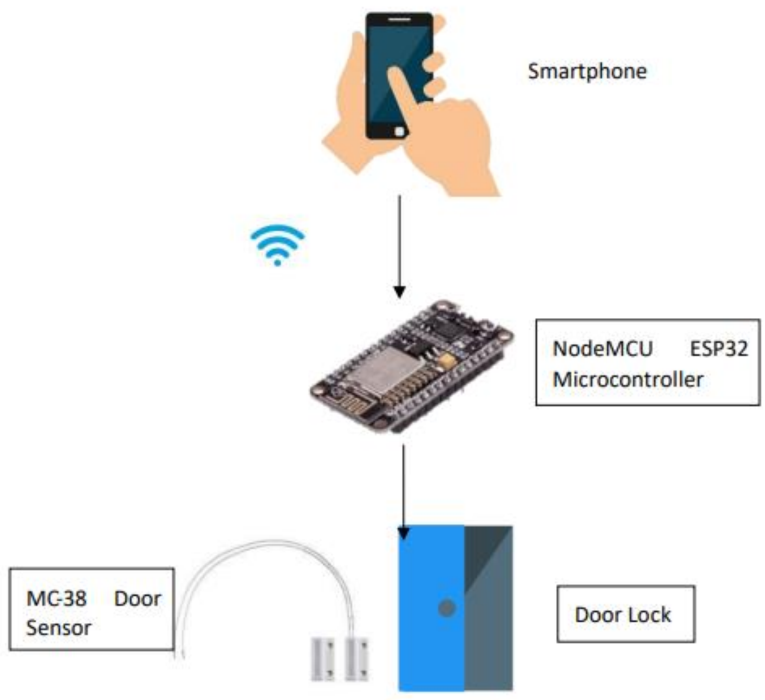


Fig 1: Blynk Application Functionality

CHAPTER 4: SYSTEM DESIGN

4.1. Block Diagram:

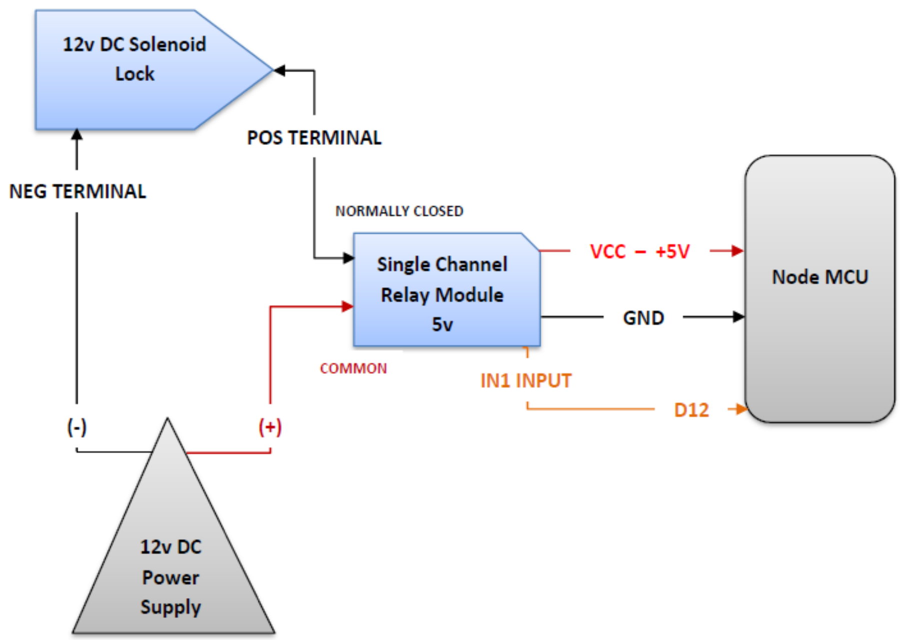


Fig 2: Solenoid Lock connections with Node MCU

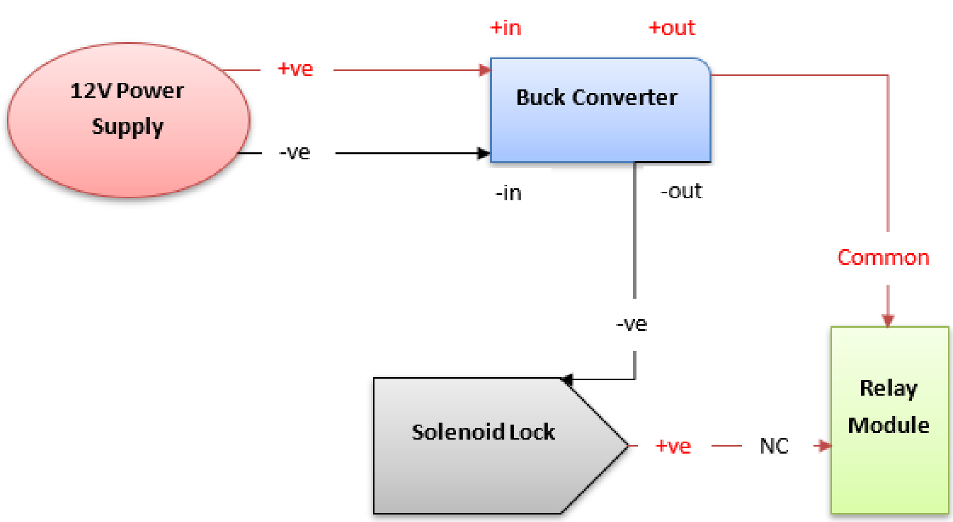


Fig 3: Voltage step down using Buck Converter

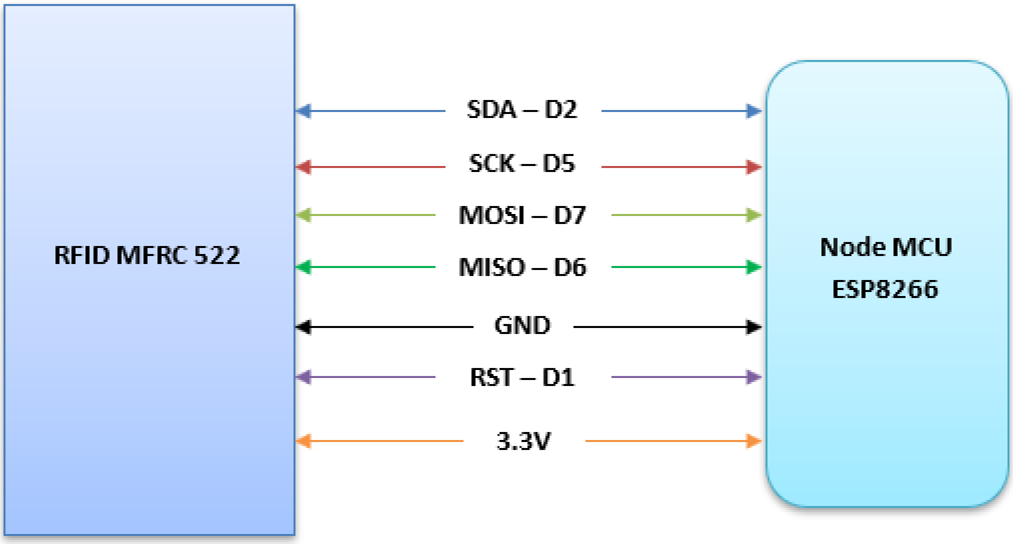


Fig 4: RFID MFRC 522 connection with Node MCU

4.2. Circuit Diagram:

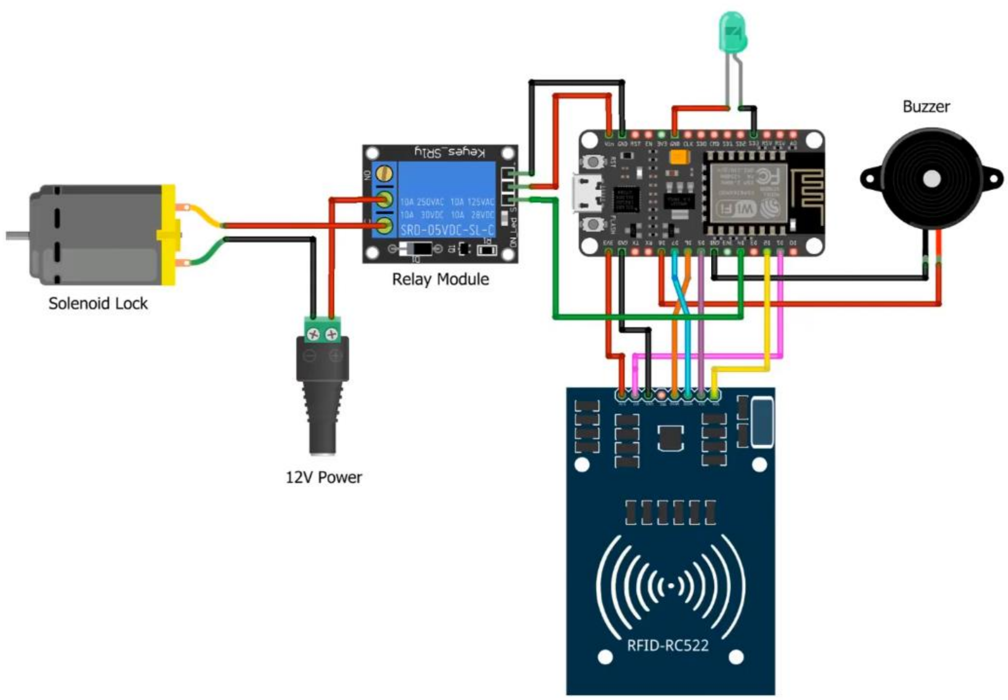
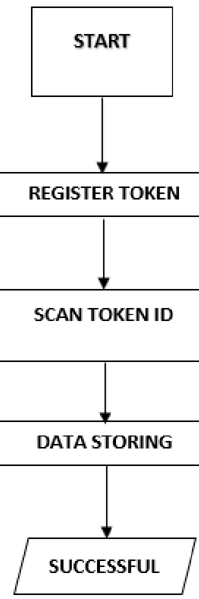
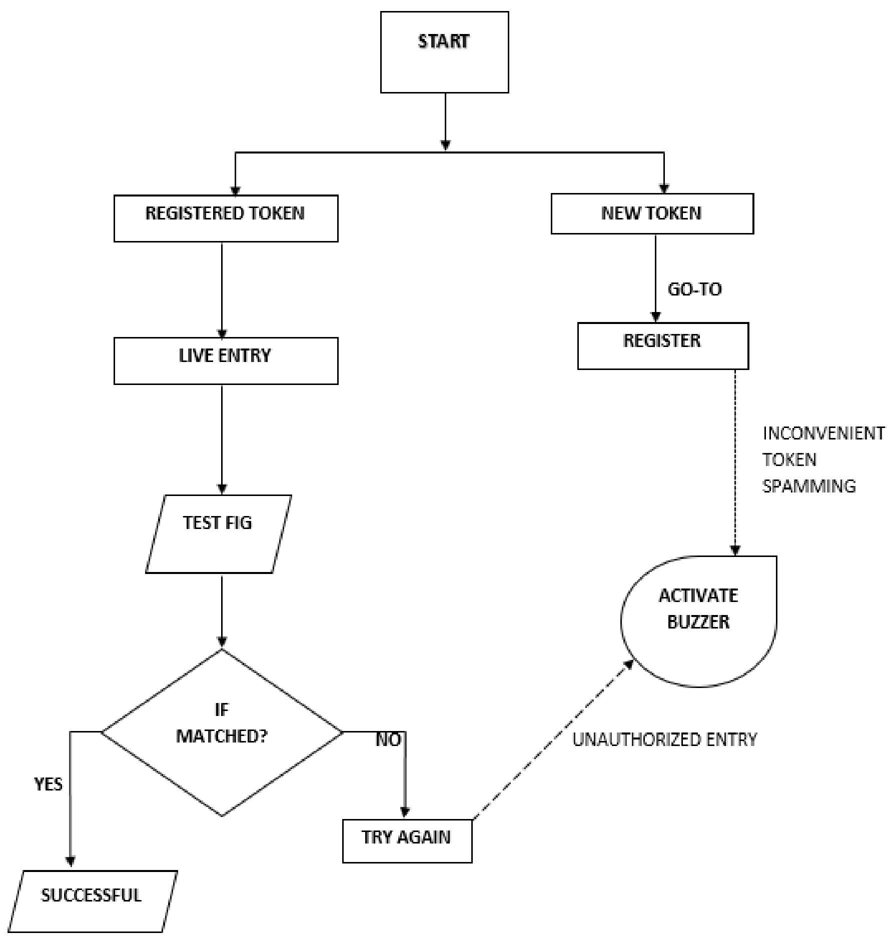


Fig 5: Circuit diagram of the project module

4.3. Flowchart:





CHAPTER 5: EXPERIMENTAL RESULTS

5.1. Snapshots:

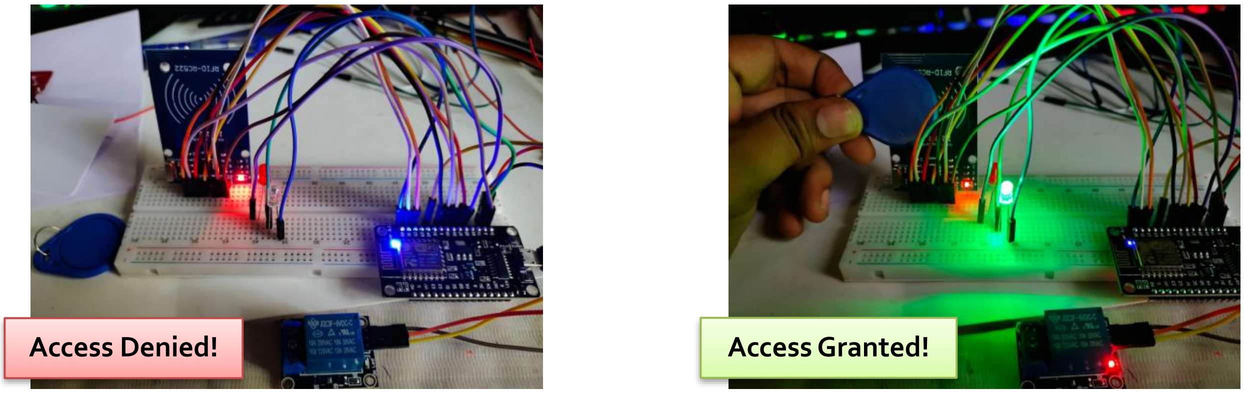


Fig 6: Card Authentication using RFID module

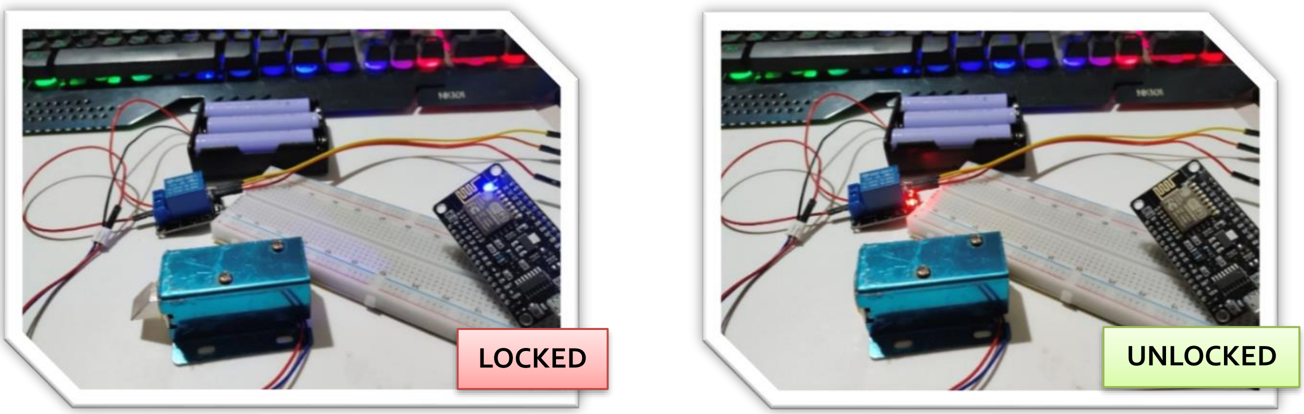


Fig 7: Solenoid lock connectivity with Relay

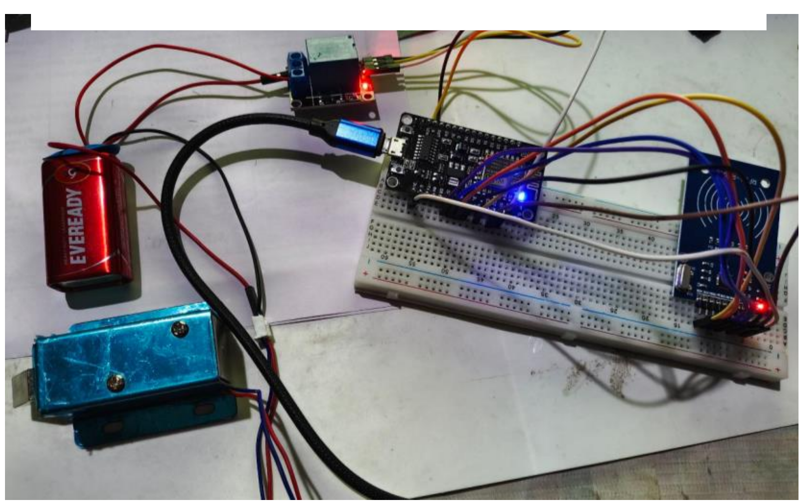


Fig 8: Full Connectivity of 12v Solenoid Lock

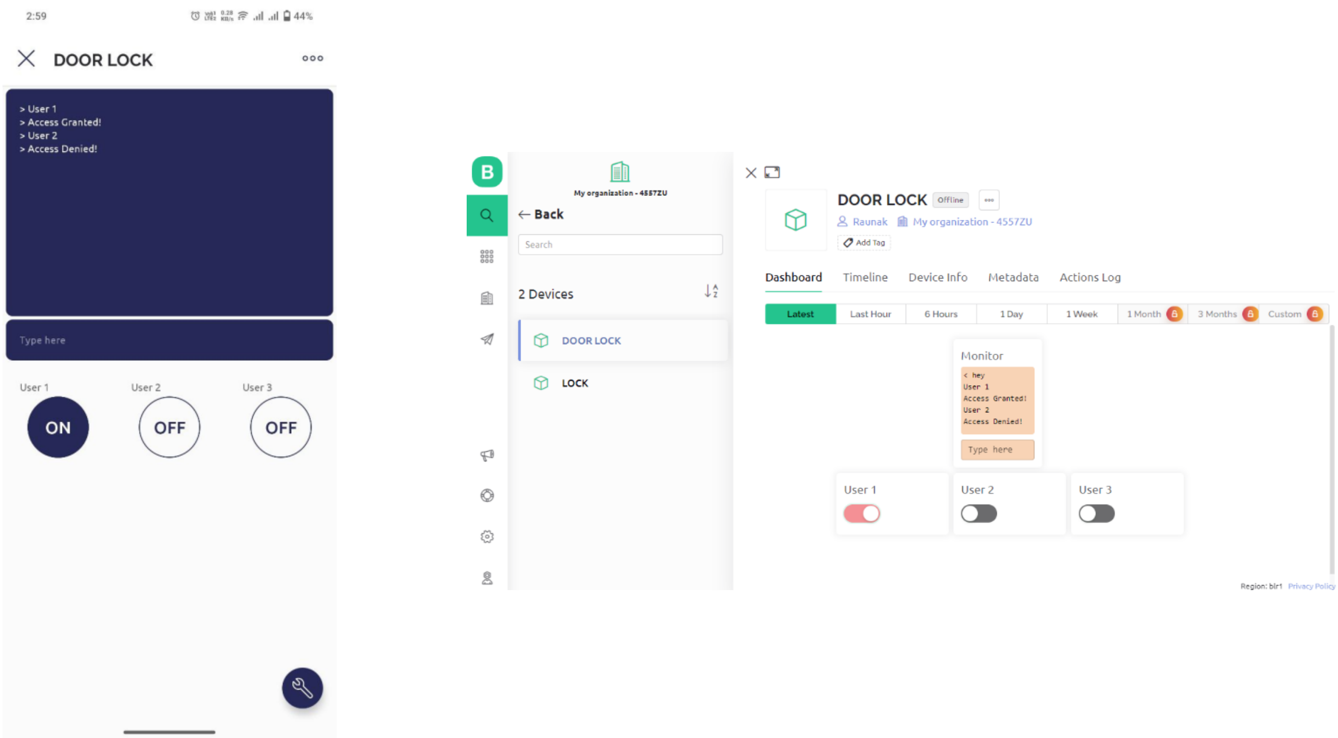


Fig 9: Blynk Application Mobile & Web Dashboard results

5.2. Arduino IDE Serial Monitor Output:

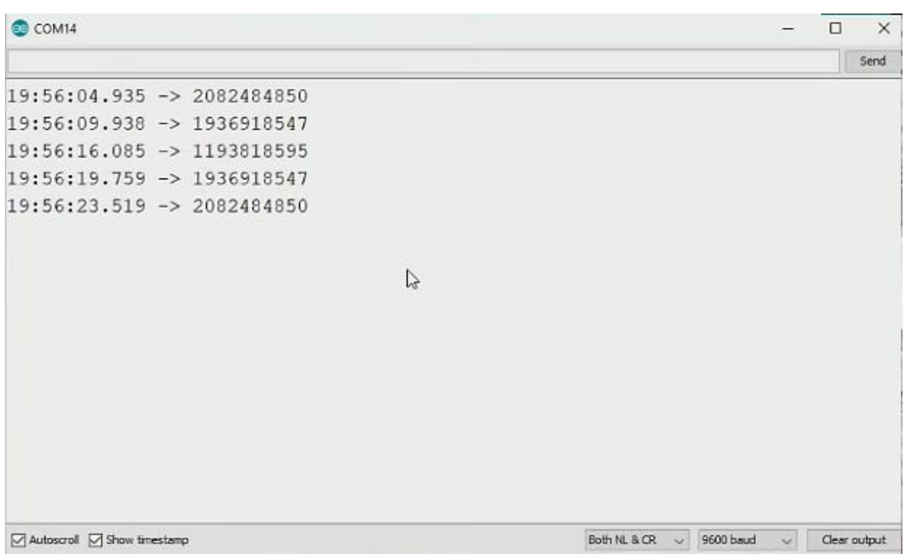


Fig 10: Card UID results in Serial Monitor

```
02:51:51.795 -> ���[75] Connecting to RAUNAK DIR-615-85CA
02:51:58.264 -> [6406] Connected to WiFi
02:51:58.264 -> [6406] IP: 192.168.0.136
02:51:58.297 -> [6406]
02:51:58.297 ->     ___  __          __
02:51:58.331 ->    / _ )/ /_ _____  / /__
02:51:58.364 ->   / _  / / // / _ \/  '_/
02:51:58.397 ->  /____/_/\_, /_//_/_/\_\
02:51:58.431 ->         /___/ v1.2.0 on ESP8266
02:51:58.464 ->
02:51:58.464 ->  #StandWithUkraine    https://bit.ly/swua
02:51:58.497 ->
02:51:58.497 ->
02:51:58.497 -> [6532] Connecting to blynk.cloud:80
02:51:58.656 -> [6793] Ready (ping: 51ms).
```

Fig 11: Blynk authentication with Node MCU results

CHAPTER 6: CONCLUSION

This paper discusses the design and execution of a smart home security system that includes email interaction alerts, a mobile web browser, and a door access system. Administrators can instantly communicate control information to field equipment through the Internet and acquire parameters for many remote units using a PC with a low-cost single-chip processor that can perform.

The developed system can also be used in business and industrial settings like offices, warehouses, and other places where certain areas are only accessible to authorized personnel, or other places where safety and security are top priorities, like the internet server room of a significant MNC where corporate data can be stolen. For added security, the system is also simply upgradeable to integrate other security features like cameras, motion detection sensors, etc. The Internet of Things (IoT), which has potential revolutionary implications, is growing in popularity in many major cities. Smart door locks are one of the uses for IoT. Smart door locks are made to give another level of security to your house or place of business. They are manageable remotely through a tablet or smartphone.

Future work on IoT smart door lock projects may involve enhancing the lock system's security features. Data transmission and reception for the IOT-based Smart Door Lock system are vulnerable to hacking and unauthorized access. Leaving a smart door lock exposed to the internet is the least secure way to create a network between a user and a smart lock.

CHAPTER 7: BIBLIOGRAPHY

1. **SMART DOOR LOCK**, ISOEVA-2022: 6th INTERNATIONAL SYMPOSIUM OF EDUCATION AND VALUES.

2. **IMPLEMENTASI ESP32-CAM DAN BLYNK PADA WIFI DOOR LOCK SYSTEM.** *Saiful Nurarif, Trinanda Syahputra, Muhammad Syahril, Dedi Setiawan, Hendra Jaya*

3. **International Journal for Scientific Research & Development (IJSRD):** Intrusion Detection System for Smart Home using Internet of Things – *Govinda K, Sai Krishna Prasad K and Sai Ram Susheel.*

4. **IIOP Conf. Series: Materials Science and Engineering 263:** Home Security System using Internet of Things – *Anitha A, VIT Vellore.*

5. **Smart Door Locking System Using IoT—A Security for Railway Engine Pilots:** *K. Sujatha, N. P. G. Bhavani, U. Jayalatsumi, T. Kavitha, B. Latha, A. Ganesan & A. Kalaivani*

APPENDIX

Code and Semantics:

```
#include <SPI.h>

#include <MFRC522.h>

#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>


#define SS_PIN 4

#define RST_PIN 2


int lock = D1;


MFRC522 mfrc522(RST_PIN, SS_PIN);        // Create MFRC522 instance.

char auth[] ="u69cegZi35QXQLDuyecg2pNFYJrrMRTj";      //Blynk Authentication Token – from Blynk web dashboard


char ssid[] = "RAUNAK DIR-615-85CA";   //Enter WiFi Name

char pass[] = "987654321";   //Enter Wifi Password


SimpleTimer timer;

int fflag = 0;

int eflag = 0;

int jflag = 0;

WidgetTerminal terminal(V2);


void setup() {
```

```
    Serial.begin(9600);      // Initialize serial communications with the PC
    Blynk.begin(auth, ssid, pass);
   pinMode(lock,OUTPUT);
   digitalWrite(lock, LOW);
    SPI.begin();             // Init SPI bus
    mfrc522.PCD_Init();      // Init MFRC522 card
    //Serial.println("Scan a MIFARE Classic PICC to demonstrate Value Blocks.");
     timer.setInterval(1000L, iot_rfid);
}

void loop() {
  timer.run(); // Initiates SimpleTimer
  Blynk.run();
}
void iot_rfid()
{
    // Prepare key - all keys are set to FFFFFFFFFFFFh at chip delivery from the factory.
    MFRC522::MIFARE_Key key;
    for (byte i = 0; i < 6; i++) {
        key.keyByte[i] = 0xFF;
    }
    // Look for new cards
    if ( ! mfrc522.PICC_IsNewCardPresent()) {
        return;
    }
    // Select one of the cards
    if ( ! mfrc522.PICC_ReadCardSerial()) {
        return;
```

```
}
// Now a card is selected. The UID and SAK is in mfrc522.uid.
// Dump UID
Serial.print("Card UID:");
for (byte i = 0; i < mfrc522.uid.size; i++) {
    Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
    Serial.print(mfrc522.uid.uidByte[i], DEC);
}
Serial.println();
// Dump PICC type
byte piccType = mfrc522.PICC_GetType(mfrc522.uid.sak);
//    Serial.print("PICC type: ");
//Serial.println(mfrc522.PICC_GetTypeName(piccType));
if (        piccType != MFRC522::PICC_TYPE_MIFARE_MINI
    &&        piccType != MFRC522::PICC_TYPE_MIFARE_1K
    &&        piccType != MFRC522::PICC_TYPE_MIFARE_4K) {
    //Serial.println("This sample only works with MIFARE Classic cards.");
    return;
}
// Enter RFID Tag ID here
if( ((mfrc522.uid.uidByte[0] == 11) && (mfrc522.uid.uidByte[1] == 22) && (mfrc522.uid.uidByte[2] ==
33) && (mfrc522.uid.uidByte[3] == 44)) && (fflag == 1) )
{
 Serial.println("User01");   //Enter User1 Name
 Blynk.virtualWrite(V2, "User01" );   //Enter User1 Name
 digitalWrite(lock, HIGH);
 delay(1000);
 digitalWrite(lock, LOW);
}
```

```
else
if(( (mfrc522.uid.uidByte[0] == 11) && (mfrc522.uid.uidByte[1] == 12) && (mfrc522.uid.uidByte[2] ==
13) && (mfrc522.uid.uidByte[3] == 14))&& (eflag == 1) )
{
 Serial.println("User02");   //Enter User2 Name
 Blynk.virtualWrite(V2, "User02" );   //Enter User2 Name
 digitalWrite(lock, HIGH);
 delay(1000);
 digitalWrite(lock, LOW);
}
else
if( ((mfrc522.uid.uidByte[0] == 21) && (mfrc522.uid.uidByte[1] == 22) && (mfrc522.uid.uidByte[2] ==
23) && (mfrc522.uid.uidByte[3] == 24))&& (jflag == 1) )
{
 Serial.println("User03");   //Enter User3 Name
 Blynk.virtualWrite(V2, "User03" );   //Enter User3 Name
 digitalWrite(lock, HIGH);
 delay(1000);
 digitalWrite(lock, LOW);
}
else
Serial.println("Unregistered User");
}
// in Blynk app writes values to the Virtual Pin 3
BLYNK_WRITE(V3)
{
   fflag = param.asInt(); // assigning incoming value from pin V3 to a variable
  // Blynk.virtualWrite(V2, fflag );
```

```
}
// in Blynk app writes values to the Virtual Pin 4
BLYNK_WRITE(V4)
{
  eflag = param.asInt(); // assigning incoming value from pin V4 to a variable
}
BLYNK_WRITE(V5)
{
  jflag = param.asInt(); // assigning incoming value from pin V5 to a variable
}
```