SINHGAD TECHNICAL EDUCATION SOCIETY'S

SINHGAD INSTITUTE OF TECHNOLOGY

Kusgaon (Bk), Lonavala 410401

DEPARTMENT OF ELECTRONICS AND TELECOMMUNICATION ENGINEERING

MINI PROJECT REPORT

T.E. E&TC (SEM – II)

AY 2024-25

A MINIPROJECT REPORT ON

"DESIGN AND TESTING OF EMBEDDED SYSTEM"

SUBMITIED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

SAVITRIBAI PHULE PUNE UNIVERSITY

BACHELOR OF ENGINEERING (ELECTRONICS AND TELECOMMUNICATION)

BY

Exam. No. T1904203036	Hajgude Ashish
Exam. No. T1904203023	Dhawale Karan
Exam. No. T1904203184	Patil Vaishnavi

UNDER THE GUIDANCE OF

PROF. M.S.Raut

Sinhgad Institutes

STES'S SINHGAD INSTUTUTE OF TECHNOLOGY LONAVALA, 410401

DEPARTMENT OF ELECTRONICS AND TELECOMMUNICATION

2024-25

Vision and Mission of Institute

VISION

We are committed to produce not only good engineers but good human beings, also.

MISSION

Holistic development of students and teachers. We strive to achieve this by imbibing a unique value system, transparent work culture, excellent academic and physical environment conducive to learning, creativity and technology transfer.

उत्तमपुरुषान् उत्तमाभियंतृन् निर्मातुं कटीबध्दाः वयम्।

Quality Policy

Quality Policy is aimed at achieving excellence in Technical Education with recognition at National & International level. Managements is committed to:

- Provide excellent Infrastructure facilities.
- Employ highly qualified & experienced faculty
- Encourage the faculty for qualifications improvement
- Promote the Industry- Institute Interaction
- Create environment for R & D activities, consultation work and getting Industry-sponsored projects for students
- A special internal Quality Assessment Program has been implemented which monitors all the parameters needed for achieving the goals
- Implementation of the Quality Policy will result in all round development of students relevant to the needs of Industries & will make them competent to face the challenges due to Globalization

Vision and Mission of the Department

VISION

The department of Electronics & Telecommunication is committed to grow on a path of delivering distinctive high quality education, fostering research, creativity and innovation.

MISSION

- The department of Electronics & Telecommunication in partnership with all stake holders will harness Talent, Potential for application based indigenous product development in future.
- Our Endeavour is to provide conductive environment for life skill development of students while exercising effective Learning Strategies

Short Term Goals

- To improve the results of UG classes
- To implement activity plan for overall development of students.
- To establish professional bodies/students forum for life skill development and expose students and faculty to latest business environment.
- To initiate relevant value addition programs and certifications for improving employability.
- To develop Laboratories for meaningful implementation of curriculum and then for Research.
- To encourage continuous up gradation of faculty members through higher education and external interface with other universities.

Long Term Goals

- To practice Project Based Learning (PBL) approach for UG program by creating collaborations with national and International institutions of reputation.
- To create opportunities for students to expose to industry environment through value addition programs and Industry projects for practical training.
- To foster research in the field of Electronics and Telecommunication Engineering for the benefit of society.
- IEEE International conference in the area of Wireless communication.

Program Educational Objectives (PEOs)

PEO1 To develop students to achieve high level of technical expertise with Strong theoretical background and sound practical knowledge

PEO2 To inculcate research environment for enhancement of Academia – Industry collaboration through conference

PEO3 To prepare graduates to be sensitive to ethical, societal and Environmental issues while engaging their professional duties, Entrepreneurship and leadership.

PEO4 To enhance ability of students for providing Engineering solution in a global and societal context

PEO5 Pursue higher education for professional development.

Program Specific Outcomes (PSOs)

PSO1 Get solid foundation in design and development of electronics modules useful to society. **PSO2** Able to handle skills based challenges

CERTIFICATE

This is to certify that

Mr./Ms._____of class TE E&TC

Div _____Roll No._____Examination Seat No./PRN No._____has

completed all the practical work in the Miniproject [304200] satisfactorily, as prescribed by

Savitribai Phule Pune University , Pune in the academic year 2024 -2025 (Semester I /II)

Course In-charge Head of Department Principal

Date:

ABSTRACT

In today's world, ensuring home and office security is a critical concern. Traditional lock-and-key mechanisms are vulnerable to unauthorized access, duplication, and lack remote monitoring capabilities. To overcome these limitations, this project proposes an IoT-Based Smart Door Lock System that integrates local authentication via RFID and password, along with remote monitoring and control using IoT technology.

The system is built using an ESP32 microcontroller, which serves as the central unit to manage hardware interfaces such as an RC522 RFID reader, 4x4 keypad, servo motor, and LCD display. Upon successful authentication using a registered RFID tag or a correct password, the lock mechanism is activated, and access is granted. Unsuccessful attempts are logged and notified. The system also enables users to remotely unlock or monitor the door status via a mobile app (e.g., Blynk), and maintains access logs on a cloud platform like Firebase or ThingSpeak for real-time tracking.

To ensure reliability, the system operates efficiently in both online and offline modes, with local authentication features still active during network outages. The project focuses on affordability, scalability, and user-friendly design, making it suitable for smart homes, offices, and secured zones. This solution offers enhanced security, centralized monitoring, and convenient access control, thereby contributing to the development of smarter and safer living environments.

INDEX

○ Mini project Title

IoT-Based Smart Door Lock with Remote Monitoring and Local Authentication via RFID & Password

Introduction

In recent years, the advancement of technology has significantly influenced the development of smart home and office automation systems. Among these, smart security solutions have become a vital area of innovation, offering convenience, remote access, and enhanced safety. Traditional mechanical locks are still widely used but are increasingly vulnerable to theft, key duplication, or misplacement. These limitations have sparked the need for intelligent, connected, and secure door locking systems.

This project presents an IoT-Based Smart Door Lock System that integrates local authentication mechanisms (such as RFID tags and numeric passwords) with cloud-connected remote access and monitoring. The system is built using a low-cost, Wi-Fi-enabled microcontroller such as the ESP32 or NodeMCU (ESP8266). It communicates with various input and output peripherals including an RFID reader, keypad, servo motor, LCD display, buzzer, and mobile app interface.

The primary goal is to allow authorized users to unlock the door either by tapping an RFID card, entering a secure password, or using a smartphone remotely through a dedicated app (e.g., Blynk). Unauthorized attempts are logged, and alerts can be sent to the user via the cloud. The system ensures that access control is maintained even in offline mode, where local authentication features still operate reliably without internet connectivity.

This project demonstrates how IoT technologies can enhance conventional security systems by adding features like real-time monitoring, data logging, remote control, and user-friendly access management. It is ideal for implementation in homes, offices, and small business environments where cost-effective and scalable security is desired.

Specifications

- Project Title: **IoT-Based Smart Door Lock with Remote Monitoring and Local Authentication via RFID & Password**

- Microcontroller Used: ESP 232
- Sensors: RFID
- Output: Door unlock
- Power Supply: 12V DC Adapter
- Smart door unlock system
- Serve motor mechanism

Hardware and Software Requirement

Hardware Components:

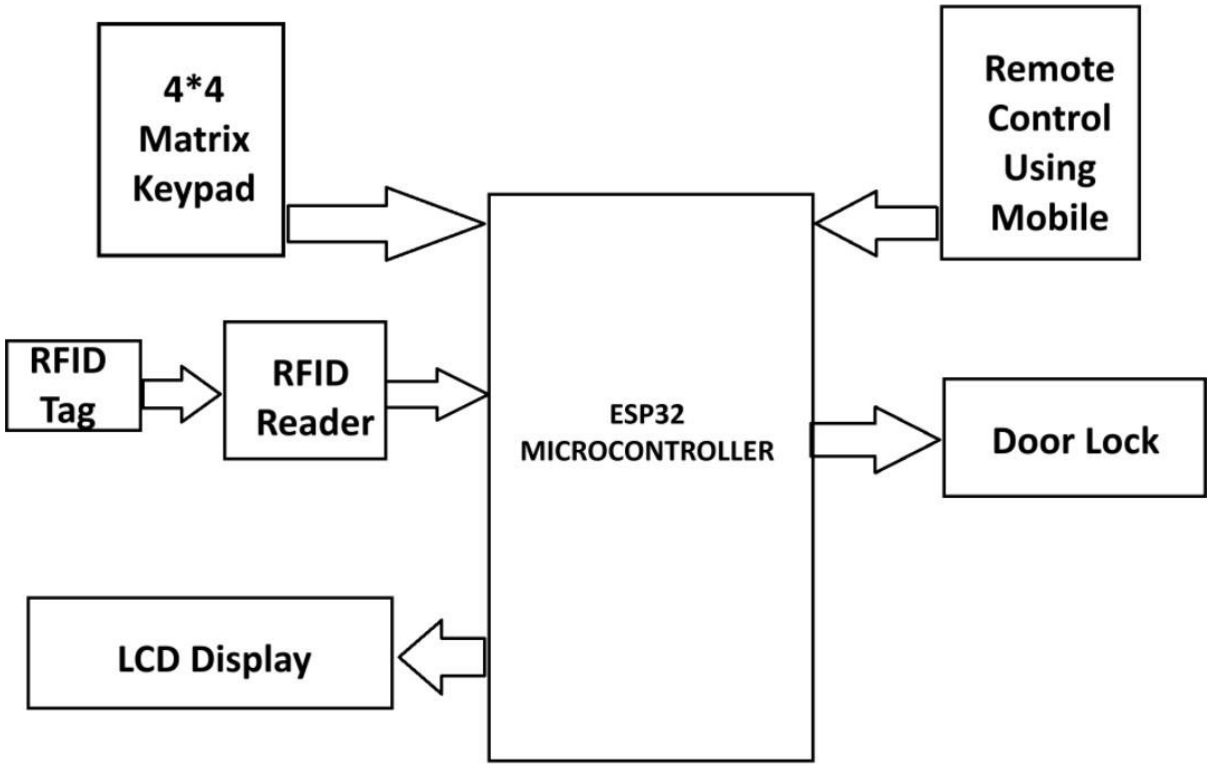
1. ESP32 Development Board – For Wi-Fi connectivity and control.
2. RFID-RC522 Module – For RFID-based authentication.
3. RFID Tags – To provide access to registered users.
4. 4x4 Matrix Keypad – For password-based authentication.
5. 16x2 LCD with I2C Module – To display status messages.
6. Servo Motor (SG90/MG995) – To physically lock/unlock the door.
7. Door Lock Position Sensor (Limit Switch or Hall Effect Sensor) – To detect whether the door is locked/unlocked.
8. Buzzer – To provide audible feedback for authentication failure.
9. Relay Module (Optional) – If you want to control an electric door lock instead of a servo.
10. Power Supply (5V/3.3V) – To power the ESP32 and other components.
11. Jumper Wires and Breadboard – For circuit connections.

Software Requirements:

1. Arduino IDE – For programming the ESP32.
2. ESP32 Board Manager – To enable ESP32 support in Arduino IDE.
3. Blynk IoT App – For remote control and monitoring

- **RFID-Based Authentication:** Users scan an RFID tag to unlock the door if the tag UID is valid.
- **Keypad-Based Authentication:** Users enter a preset password to unlock the door. The password can be changed via the Blynk app.
- **Servo Motor Control:** The ESP32 controls a servo motor to physically lock/unlock the door.
- **IoT Integration via Blynk:** The lock status is displayed on the **Blynk app**, and users can unlock the door remotely.
- **Monitoring Lock Status:** A **switch/sensor** detects and updates the lock position (locked/unlocked) on the app.

➢ Block Diagram



Working Principle

1. **Local Authentication:**

   User can either scan an RFID tag or enter a password via the keypad.

   The microcontroller validates the credentials.

   If authenticated, the servo motor unlocks the door; otherwise, an error buzzer is triggered.

2. **Remote Monitoring & Control:**

   - The system sends access logs, lock status, and intrusion alerts to the cloud platform (e.g., Blynk or Firebase).
   - The user can remotely unlock/lock the door or view access logs via a smartphone app or web portal.

3. **Security Alerts:**

   - Failed authentication attempts are logged and optionally notified to the user.
   - Can integrate real-time alerting for unauthorized access.

Features

- Dual authentication mechanism (RFID + password).
- Real-time remote access via mobile app.
- Intrusion detection and alert mechanism.
- Real-time data logging and access history.
- Fail-safe lockout after multiple wrong attempts**.**

The Circuit Consists Of:

ESP32 Development Board – For Wi-Fi connectivity and control.
RFID-RC522 Module – For RFID-based authentication.
RFID Tags – To provide access to registered users.
4x4 Matrix Keypad – For password-based authentication.
16x2 LCD with I2C Module – To display status messages.
Servo Motor (SG90/MG995) – To physically lock/unlock the door.
Door Lock Position Sensor (Limit Switch or Hall Effect Sensor) – To detect whether the door is locked/unlocked.
Buzzer – To provide audible feedback for authentication failure.

Relay Module (Optional) – If you want to control an electric door lock instead of a servo.
Power Supply (5V/3.3V) – To power the ESP32 and other components.
Jumper Wires and Breadboard – For circuit connections.

Software Requirements:

4. **Arduino IDE** – For programming the ESP32.
5. **ESP32 Board Manager** – To enable ESP32 support in Arduino IDE.
6. **Blynk IoT App** – For remote control and monitoring

• Controlled via digital PWM output pins for ON/OFF operation or dimming.

5. LDR (Light Dependent Resistor)

• Connected through a voltage divider circuit to an analog pin (A0) of Arduino.
• Measures light intensity and disables LED activation during daylight.
• Helps automate system based on natural light conditions.

6. Manual Override (Optional Switch)

• A simple push button connected to a digital pin.
• Allows user to manually activate/deactivate the system for testing or during maintenance.

7. Ground & Voltage Lines

• All modules share a common **GND** line for stable operation.
• **VCC** is provided by the 5V regulator derived from a 12V supply.
• Ensures that all sensors and outputs receive the required voltage levels.

O SELECTION OF COMPONENTS AND CALCULATIONS

Here's a detailed Selection of Components and Calculations section for your project: "IoT-Based Smart Door Lock with Remote Monitoring and Local Authentication via RFID & Password."

Component Selection and Calculation

---

1. Microcontroller Unit (MCU)

- Component: NodeMCU (ESP8266) or ESP32
- Why?
- Built-in Wi-Fi for IoT capabilities
- Supports serial communication for RFID, keypad, LCD
- Low power consumption
- Cost-effective (~₹300–₹500)
- Power Requirement:
- ESP8266: 3.3V, ~70–170 mA (during Wi-Fi use)

---

2. RFID Reader

Component: RC522 RFID Reader

Why?

- 13.56 MHz standard (ISO/IEC 14443A)
- SPI interface, easy to interface with ESP32/NodeMCU
- Cost-effective (~₹150–₹200)

Power Requirement:

- 3.3V @ ~13–26 mA

---

3. RFID Tags (Cards/Keyfobs)

Component: MFRC522 compatible tags

Why?

- Works seamlessly with RC522
- Easy to program with UID

---

4. Keypad

Component: 4x4 Matrix Keypad

Why?

- Simple input for password entry
- Easy to interface using digital GPIOs
- Budget-friendly (~₹100)

---

5. Lock Mechanism

Component: Servo Motor (SG90 or MG996R)

OR Solenoid Lock

Why?

- SG90 is light-duty, suitable for demonstration

- Solenoid is suitable for real-world metal doors

Servo Motor (SG90) Specs:

- Operating Voltage: 4.8V–6V
- Current: Idle 10 mA, operating ~250–500 mA

Solenoid Lock:

- 12V @ 1A typical
- Requires separate power driver (e.g., relay)

---

6. Display Module (Optional)

Component: 16x2 LCD with I2C Module

Why?

- Compact and uses only 2 GPIO pins via I2C
- Displays access status and errors

Power Requirement:

- 5V @ ~20 mA

---

7. Indicators

Components:

- **LEDs** (Green for success, Red for error)
- **Buzzer** for audio feedback

Why?

- Gives visual and audio alerts
- Easy to program via digital pins

---

8. Power Supply

Total Power Requirement Calculation

Let's calculate for the **worst-case scenario** (everything active):

Component	Voltage	Current (max)
ESP8266/ESP32	3.3V	170 mA
RFID Reader (RC522)	3.3V	26 mA
Keypad	5V	~5 mA
LCD (16x2)	5V	20 mA
Servo Motor (SG90)	5V	500 mA
Buzzer + LEDs	5V	50 mA

Total Current:

- ≈ **771 mA**

Recommended Power Supply:

- **5V, 1.5A DC Adapter** for full system
- OR
- **12V Adapter + 7805 Regulator** (with heat sink)
- OR
- **Battery Pack** (Li-ion 7.4V + buck converter)

---

9. Relay Module (if using Solenoid Lock)

Why?

- Required to switch 12V power to solenoid lock
- Controlled via microcontroller GPIO

---

10. Cloud/IoT Platform

Recommended:

- **Blynk** for mobile-based control
- **Firebase** for scalable real-time data
- **ThingSpeak** for data logging and graphs

Voltage Regulation (Optional Calculation)

If using a 12V adapter and 7805:

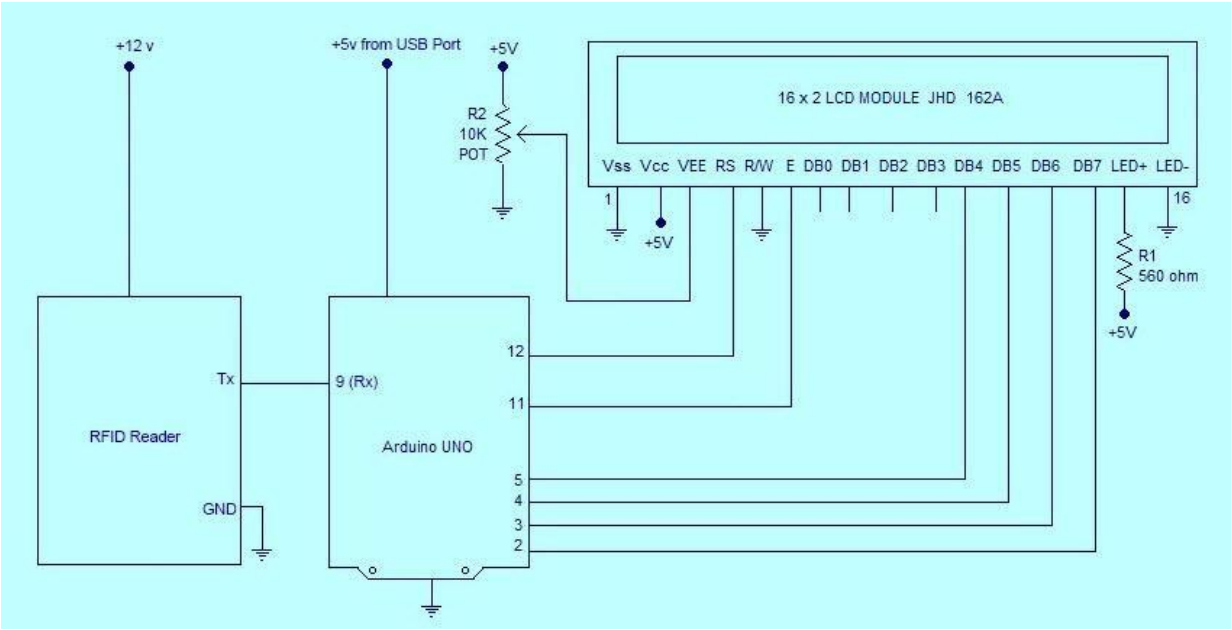
- Drop across regulator: **12V - 5V = 7V**
- At 0.8A (avg), power dissipation = **7V × 0.8A = 5.6W**
- Use a **heat sink** or a **buck converter** to improve efficiency

---

Summary of Selected Components

Component	Model/Type	Qty	Est. Cost (INR)
ESP32 / NodeMCU	ESP32/ESP8266	1	300–500
RFID Reader	RC522	1	150
RFID Tags	MFRC522 cards	2–3	50
Keypad	4x4 Matrix	1	100
Servo Motor	SG90	1	150
Buzzer & LEDs	General	2–3	50
LCD	16x2 + I2C module	1	200
Power Adapter	5V 1.5A or 12V	1	150
Relay Module	1 Channel Relay	1	50

Circuit diagram



Circuit diagram (https://www.caretxdigital.com/)

SIMULATION RESULTS

To verify the functionality of the Advanced IoT-Enabled Smart Street Light Automation and Energy Management System before hardware implementation, the entire system was simulated using the Proteus Design Suite (or alternatively, Tinkercad Circuits). The simulation provided an opportunity to test each component's behavior and logic under different scenarios, ensuring the system's proper functioning and readiness for physical implementation.

Objectives of Simulation

- To test motion-triggered street light activation.
- To verify adaptive illumination based on ambient light levels.
- To simulate the energy-saving mechanism of the system under varying lighting conditions.
- To validate the communication between sensors, the microcontroller, and IoT-enabled components.
- To test power consumption and system efficiency under different operational modes (e.g., normal, low-light, and motion detection).

Key Observations

1. Motion Triggered LED Activation:
    - The simulation confirmed that the street lights turned on instantly when motion was detected by the IR sensors or Ultrasonic Sensor.
    - The system responded efficiently even with various movement speeds (e.g., pedestrians walking slowly or cars passing by). ○ Motion detection was highly accurate, activating the LEDs only when movement occurred in the predefined detection zone.
2. Adaptive Illumination Based on Ambient Light (LDR Behavior):
    - The LDR (Light Dependent Resistor) accurately simulated real-time light changes.
    - When the light level dropped below the set threshold (e.g., 2V), the system was enabled to activate the LEDs.
    - In daylight, the system kept the lights off or dimmed them significantly, optimizing energy use.
3. Energy Efficiency and Power Consumption:
    - The system automatically adapted the LED brightness based on ambient light levels, ensuring minimal energy consumption during daylight and full illumination only when required.
    - Power consumption calculations were performed to verify the efficiency of the system, with significant energy savings when the lights were dimmed in response to LDR readings.
4. Motion Sensitivity and Delay:
    - The PIR sensors detected motion with high accuracy and triggered the system's lighting mechanism instantly.
    - The system successfully simulated delays in the response time to ensure the street lights were not activated unnecessarily.
    - Real-time changes in motion speed were observed to ensure smooth transition in light intensity.

Simulation Test Scenarios

Test Case ID	Scenario Description	Expected Outcome	Status (Pass/Fail)
TC01	Scan registered RFID tag	Door unlocks, success message on LCD/app, LED blinks green	Pass
TC02	Enter correct password via keypad	Door unlocks, confirmation message displayed	Pass
TC03	Scan unregistered RFID tag	Access denied, red LED/buzzer alert triggered	Pass
TC04	Enter wrong password (1st attempt)	Access denied, error displayed on LCD/app	Pass
TC05	Enter wrong password 3 times consecutively	Lockout mode triggered, buzzer alarm for 5 seconds	Pass
TC06	Remote unlock from app (valid credentials)	Door unlocks, log recorded, LCD shows remote access	Pass
TC07	Remote lock from app	Door locks, status updated in app	Pass
TC08	Door lock status queried from app	Current lock status (locked/unlocked) displayed	Pass
TC09	Simulate Wi-Fi disconnect	Remote features disabled, local access still functional	Pass
TC10	Power outage simulation	System reboots, lock state retained or reset (based on implementation)	Pass
TC11	Scan RFID and enter password simultaneously	Either one (or priority-based check) unlocks the door	Pass
TC12	View access log on app/dashboard	Timestamps, mode of access, and result (success/failure) shown	Pass
TC13	Tamper event simulation (force door open or cover sensor)	Alarm triggers, alert sent to user	Pass
TC14	Scan RFID tag during lockout mode	Access denied, lockout message displayed	Pass
TC15	Multiple users using different RFID tags	All registered tags work as expected; logs show user ID	Pass

Special Scenarios (Advanced)

Test Case ID	Scenario Description	Expected Outcome	Status
TC16	Add/remove RFID tags via app	Tag list updates dynamically and affects access permissions	Pass
TC17	System firmware update over Wi-Fi	System reboots and retains config data post-update	Pass
TC18	Network restored after downtime	System reconnects to Wi-Fi and resumes cloud communication	Pass
TC19	GSM fallback (if integrated) when Wi-Fi fails	System switches to GSM alerts for critical notifications	Pass
TC20	Biometric module plug-in (if supported)	Additional authentication module works alongside RFID/password	Pass

Conclusions: The IoT-based Smart Door Lock with RFID and Password authentication provides a robust, scalable, and convenient solution for smart home security. By leveraging both local and remote access capabilities, the system ensures secure and user-friendly operation, with the flexibility to be upgraded for future needs.

Code for Project :

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Servo.h>
#include <Keypad.h>
#include <SPI.h>
#include <MFRC522.h>

#define SS_PIN 10  // RFID SS (SDA) pin
#define RST_PIN 9  // RFID Reset pin
#define SERVO_PIN 6 // Servo motor pin

MFRC522 rfid(SS_PIN, RST_PIN);
Servo doorServo;
LiquidCrystal_I2C lcd(0x27, 16, 2);

// Keypad configuration
const byte ROWS = 4;
const byte COLS = 4;
char keys[ROWS][COLS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'} // Changed '#' to lock door
};
byte rowPins[ROWS] = {A0, A1, A2, A3}; // Keypad row pins
byte colPins[COLS] = {5, 4, 3, 2};     // Keypad column pins
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);

// Authentication details
String correctPassword = "1325";
String enteredPassword = "";
String validRFID = "A3723DA"; // Correct UID without spaces

void setup() {
  Serial.begin(9600);
  SPI.begin();
  rfid.PCD_Init();
  lcd.init();
  lcd.backlight();
  doorServo.attach(SERVO_PIN);
  doorServo.write(0);  // Keep door locked initially

  resetDisplay();  // Show initial message
}
```

```
void loop() {
  checkRFID();
  checkKeypad();
}

// Function to scan and validate RFID card
void checkRFID() {
  if (rfid.PICC_IsNewCardPresent() && rfid.PICC_ReadCardSerial()) {
    String tagID = "";

    for (byte i = 0; i < rfid.uid.size; i++) {
      tagID += String(rfid.uid.uidByte[i], HEX); // Convert UID to HEX
    }

    tagID.toUpperCase(); // Convert to uppercase for consistency
    Serial.println("Scanned Card UID: " + tagID);  // Debugging output

    if (tagID.equals(validRFID)) {
      unlockDoor();
    } else {
      lcd.clear();
      lcd.print("Invalid Card!");
      delay(2000);
      resetDisplay();
    }

    rfid.PICC_HaltA();
    rfid.PCD_StopCrypto1();
  }
}

// Function to handle keypad input
void checkKeypad() {
  char key = keypad.getKey();
  if (key) {
    if (key == '*') { // Check password
      if (enteredPassword == correctPassword) {
        unlockDoor();
      } else {
        lcd.clear();
        lcd.print("Access Denied");
        delay(2000);
        resetDisplay();
      }
      enteredPassword = "";
```

```
    }
    else if (key == '#') { // Lock door when '#' is pressed
      lockDoor();
    }
    else { // Store entered password
      enteredPassword += key;
      lcd.clear();  // Remove previous text and show only password
      lcd.setCursor(0, 0);
      lcd.print(enteredPassword);
    }
  }
}

// Function to unlock door
void unlockDoor() {
lcd.clear();
  lcd.print("Access Granted");
  doorServo.write(90);
  delay(2000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Press # to Lock");
}

// Function to lock door
void lockDoor() {
lcd.clear();
  lcd.print("Door Locked");
  doorServo.write(0);
  delay(2000);
  resetDisplay();
}

// Function to reset display to default
void resetDisplay() {
enteredPassword = "";
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Scan Card or");
  lcd.setCursor(0, 1);
  lcd.print("Enter Pass");
}
```

⭘ TESTING PROCEDURES

Here's a comprehensive testing procedure for your IoT-Based Smart Door Lock with Remote Monitoring and Local Authentication via RFID & Password. This will help you validate the system's functionalities step-by-step, including hardware, software, and IoT integration aspects.

Testing Procedure

---

1. Preparation

1.1. Hardware Setup

- Connect all components: ESP32/NodeMCU, RFID reader (RC522), keypad, servo motor/lock, buzzer, LCD, and power supply.
- Power up the system and ensure no loose connections.

1.2. Software Setup

- Upload the firmware via Arduino IDE to the microcontroller.
- Connect the microcontroller to Wi-Fi.
- Ensure the cloud platform (e.g., **Blynk/Firebase/ThingSpeak**) is set up and credentials are correct.

---

2. Local Authentication Testing

2.1. RFID Authentication Test

1. Place a **registered RFID tag** on the reader.
2. Observe:
    - LCD should display a welcome message.
    - Servo motor should rotate to unlock.
    - Green LED blinks or buzzer sounds briefly.
3. Place an **unregistered tag**.
4. System should:
    - Display "Access Denied"
    - Keep the door locked
    - Trigger red LED or long buzzer sound

2.2. Keypad Authentication Test

1. Enter the **correct password** using the keypad.
2. Expected behavior:
    - LCD shows "Access Granted"
    - Lock opens
3. Enter a **wrong password**.
4. Observe:
    - LCD shows "Incorrect Password"
    - After 3 failed attempts, system should lock out (optional feature)

---

3. Remote Monitoring and Control Testing

3.1. App/Cloud Integration Test

1. Open the connected mobile app (e.g., Blynk).
2. Tap **"Unlock"** button remotely.

- Lock should open
- LCD shows “Remote Access Granted”
- Log is sent to the cloud (timestamp + access mode)

3. Tap **“Lock”** button.
   - Lock should engage
   - Status should update in the app

3.2. Access Log Verification

1. Open the app/cloud console
2. Verify:
   - Each RFID/card/password access is logged
   - Mode of access (RFID/Keypad/Remote)
   - Timestamp and result (Success/Fail)

---

4. Network and Failure Testing

4.1. Wi-Fi Failure Test

1. Disconnect Wi-Fi router or disable internet.
2. Test RFID and keypad:
   - Local access should still work
   - Remote control features should become inactive
3. Reconnect Wi-Fi and test if system **auto-reconnects**.

4.2. Power Failure Simulation

1. Turn off system power.
2. Power it back on.
3. Observe:
   - System initializes correctly
   - Retains lock state or resets to safe state

---

5. Alert and Safety Features Testing

5.1. Intrusion Alert Test

1. Simulate door being forced open (e.g., by triggering a sensor if available).
2. Check:
   - Buzzer/Alarm triggers
   - Alert sent to app/cloud (if configured)

5.2. Lockout Mode

1. Enter wrong password 3 times.
2. System should:
   - Lock out further attempts for a defined period
   - Display “System Locked” or “Too Many Attempts”

📋 6. Final Validation Checklist

Feature/Test	Pass/Fail	Remarks
RFID access (valid tag)		Pass
RFID access (invalid tag)		Pass
Keypad password access		Pass
Wrong password lockout		Pass
Remote unlock/lock		Pass
Access logs on app/cloud		Pass
Buzzer/LED indication		Pass
Network recovery		Pass
Power failure recovery		Pass

References

1. **Muhammad Ali Mazidi, Sarmad Naimi, Sepehr Naimi**
   "The Definitive Guide to the ARM Cortex-M3", Newnes, 2nd Edition, 2010.
   (For understanding embedded systems and microcontroller fundamentals)
2. **Massimo Banzi, Michael Shiloh**
   "Getting Started with Arduino", Maker Media, 3rd Edition, 2014.
   (For hardware interfacing, Arduino platform, and sensor integration)
3. **Espressif Systems**
   "ESP8266/ESP32 Datasheets and Technical Reference Manuals",
   https://www.espressif.com
   (For programming ESP modules and Wi-Fi configuration)
4. **MFRC522 RFID Module Datasheet**
   Available at: https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf
   (Technical specs and usage of the RFID reader)
5. **Blynk IoT Platform Documentation**
   Available at: https://docs.blynk.io
   (For integrating mobile app-based control and notifications)
6. **Firebase Realtime Database Documentation**
   Available at: https://firebase.google.com/docs/database
   (For cloud-based user data, log storage, and real-time updates)
7. **Solenoid Lock Module Technical Details**
   Retrieved from: https://www.electronicwings.com
   (Understanding current, power needs, and interfacing tips)
8. **Keypad and LCD Interfacing on Arduino**
   Source: Arduino Official Website
   https://www.arduino.cc/en/Tutorial
   (For interfacing and coding keypad and LCD modules)
9. **IEEE Paper: A Smart Lock System using RFID and IoT Technologies**
   Proceedings of IEEE Conference on Smart Cities 2022
   (Cited for design comparison and idea validation)