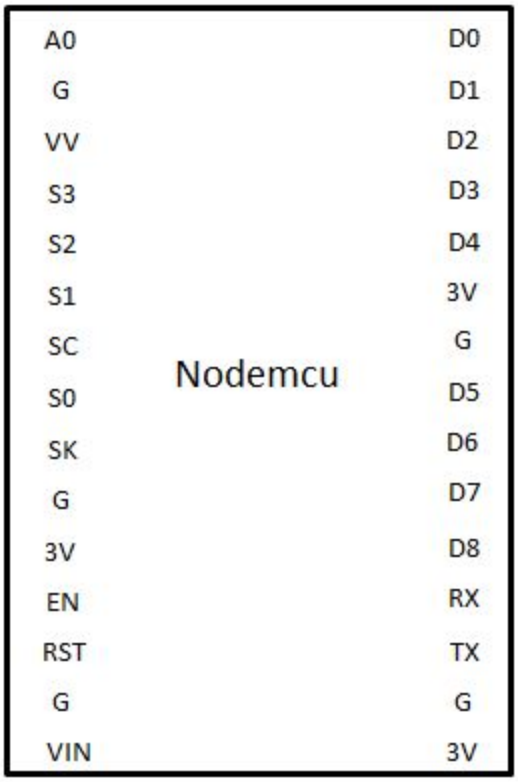
Automatic Door Lock System
Design Manual

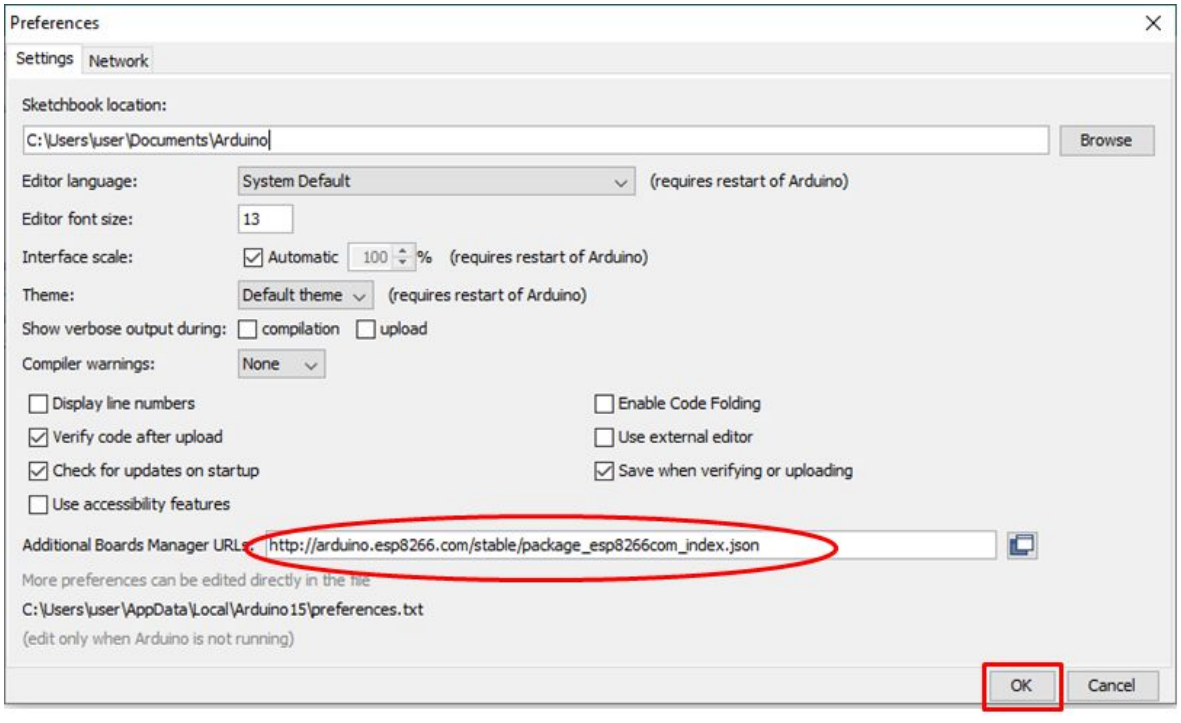
Table of Contents

1. Nodemcu (ESP8266)



Install CH340G driver (depends on the Nodemcu type you are using) to your computer and connect the nodemcu using the USB cable.

Go to File > Preferences. In the "Additional Boards Manager URLs" field, type (or copy-paste) *http://arduino.esp8266.com/stable/package_esp8266com_index.json*



Then go to Tools > Board > Board Manager. Type "esp8266" in the search field. The entry "esp8266 by ESP8266 Community" should appear. Click that entry and look for the install button on the lower right.

Once the download is complete, you can start coding!

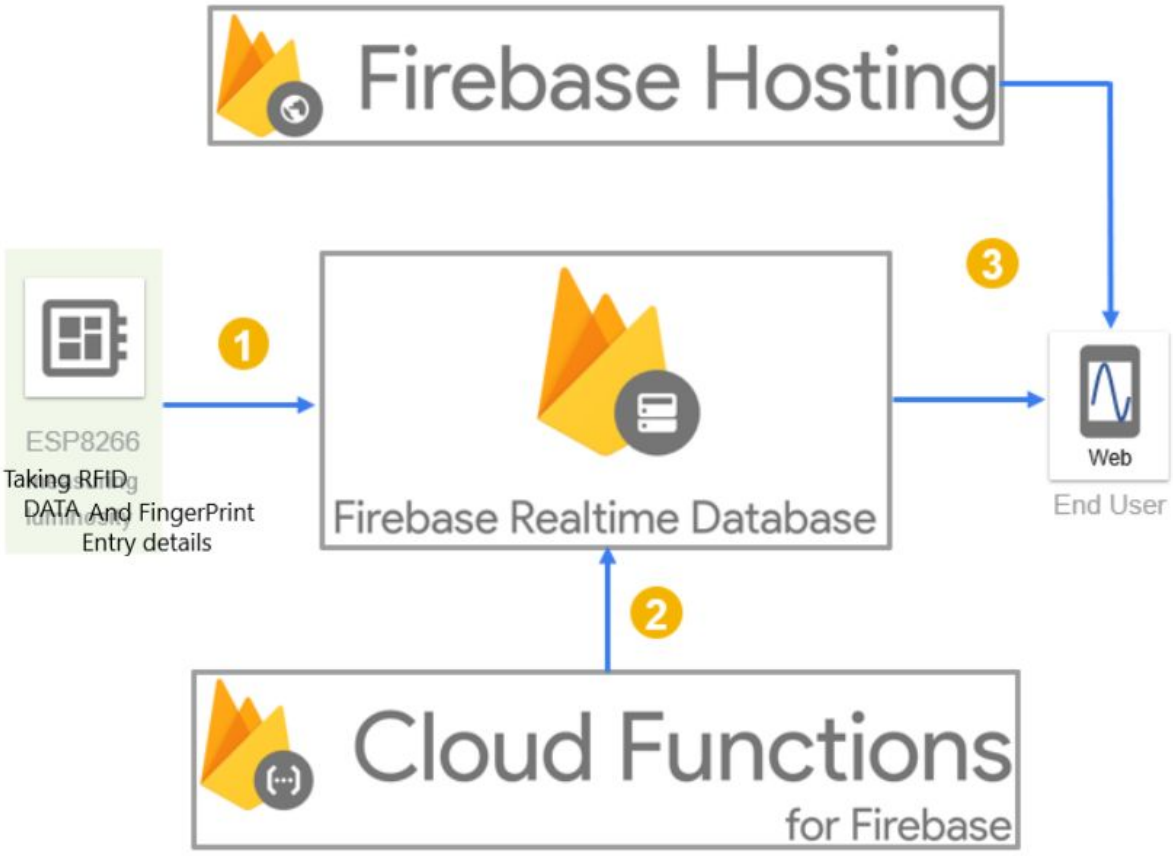
In arduino IDE select the board as NodeMCU 1.0 (ESP - 12E Module).

To check whether it is working, use the following code.

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```

2. Firebase as Database and Server



Step 1 :

First have to set up a Firebase project using https://console.firebase.google.com and set configurations of the project using https://firebase.google.com/docs this link includes all the documents about how to use firebase for store data and how to use those stored data in your website.

Step 2:

To get connected to our LAN, we just have to copy the **void setup()** function of one of the "Arduino core for ESP8266" examples, for instance: [link] (or you can reach it by: **Arduino_IDE>File>Examples>ESP8266Wifi>WifiClient**).

Below is the code for wifi set up and replace the values of WIFI_SSID and WIFI_PASSWORD according to your LAN

```
#include <ESP8266WiFi.h>

#define WIFI_SSID "YOUR_WIFI_SSID"
#define WIFI_PASSWORD "YOUR_WIFI_PASSWORD"

void setup() {
  Serial.begin(115200);
  delay(10);

  // We start by connecting to a WiFi network

  Serial.println();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(WIFI_SSID);

  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}

void loop() {
  // Network stuff will come here.
```

Step 3:

Connect Nodemcu to the firebase. There is a Firebase SDK to perform all these actions for several languages (JavaScript, C++ for instance) and operating systems (Android, iOS). This https://firebase.google.com/docs include all about them.

Have to include the header `#include <FirebaseArduino.h>` for Nodemcu to firebase connection.

On the FirebaseArduino API Reference [https://github.com/FirebaseExtended/firebase-arduino/issues/224] , the only function to connect to a Firebase project is `Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH)` and it requires two parameters like the ones we've seen before:FIREBASE_HOST (a URL) and FIREBASE_AUTH (a secret).

So first we have to define FIREBASE_HOST (a URL) and FIREBASE_AUTH (a secret) as below

```
#define FIREBASE_HOST "yourFirebaseProjectName" // your firebase project name
```

```
#define FIREBASE_AUTH "yourFirebaseDatabaseSecretKey" // your firebase database secret key
```

Step 4:

How we used firebase realtime database with RFID reader

1. To register new users (new tags)

   First get the registered uid values from the firebase to a json object

```
JsonObject& Registered_Tags =
Firebase.get("/Employers").getJsonVariant().asObject();
```

In above "Employers" is the firebase realtime database name.

Append new uidTag to the Registered_Tags and push that to the firebase realtime database using below command.

```
Firebase.pushString("Registered_Tags", uidTag);
```

2.To recognize weather entered tag is authorized or not

When RFID tag is recognized by the RFID reader. It compares that Uid tag with the database entries under above Registered_Tags and allow user (unlock the door) if it matches any entry in the database otherwise door remain closed and buzzer sounds.

Have to update firebase realtime database with the entry details like date and time including the header `#include <NTPClient.h>`

-When there is an authorized user :

update the employer entered date in the firebase database

```
Firebase.setString("Employers/" + slotkey + "/enter_date", dayStamp);
```

update the employer entered time in the firebase database

```
Firebase.setString("Employers/" + slotkey + "/enter_time", timeStamp);
```

update the history

```
    Firebase.pushString("History_Of_Entered/",uidTag + "u" + dayStamp + "u" +
timeStamp);
```

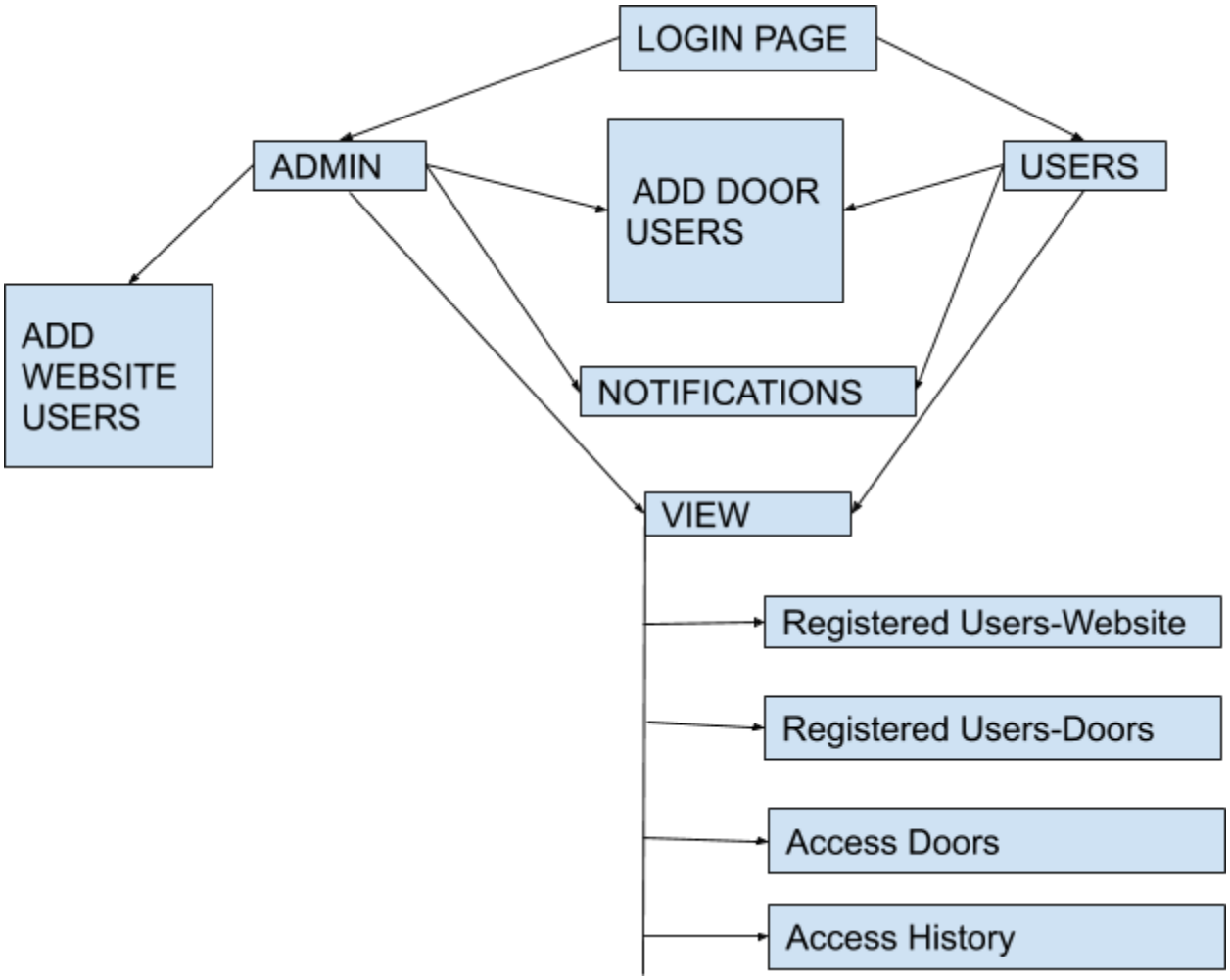
-When unauthorized access is recognized:

Have to update database with the time and date that the unauthorized access attempt is recognized.

`timeClient.update(); //` update the time calculation
`Firebase.pushString("Access_Denied", timeClient.getFormattedDate());//` update the database with the unauthorized access attempt details

3.Website

Sitemap of the website



LOGIN PAGE

- Need to be a Page with a form which has fields to fill credentials.
- Connect this page with firebase Authentication.
- Have to have a way to separate admin and other users

ADD WEBSITE USERS

- Only accessible for admin panel.
- Connect this page with firebase Authentication.
- Connect this page with firebase firestore to store user data other than the user credentials.
- There should be a way to add new admins and other new users separately.
- A page with a form to fill user details.

ADD DOOR USERS

- Accessible to both admins and other users.
- Connect this page with firebase realtime database.
- A page with a form to fill user details.

NOTIFICATIONS

- Should notify unauthorized access attempts in the doors with time and date it happened.
- Connect this page with firebase realtime database.
- Accessible to both admins and other users.

VIEW

- Key Page to four other sub pages
    - Registered users - website
    - Registered users - doors
    - Main entrance Access
    - Main entrance Access History
    - Server room Access
    - Server room Access History

Registered users - website

- Can view all registered users to use the website here
- Have to connect this page to firebase firestore database which stored website user details.

Registered users - doors

- Can view all registered users to enter through doors here
- Have to connect this page to firebase realtime database which stored website user details.

Access Doors

- Can view last door entry details of each user.
- Have to connect this page to firebase realtime database which stored website user details.

Access History

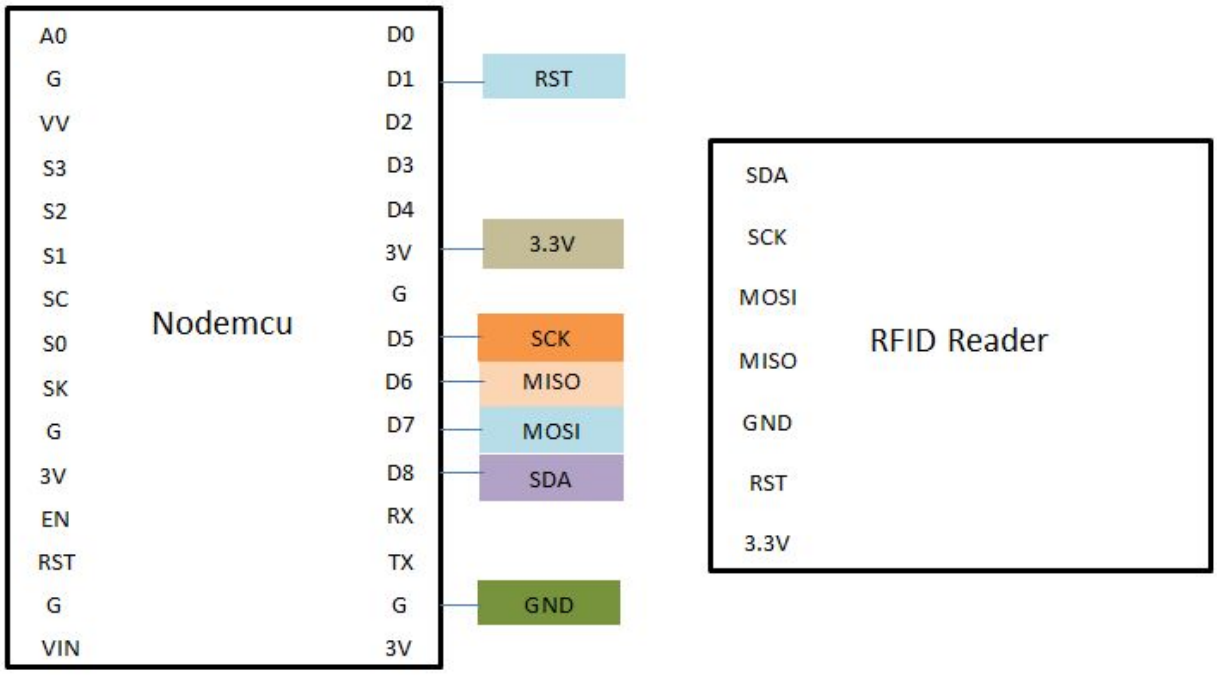
- Can view access details of each door but does not show any unauthorized access attempts
- Have to connect this page to firebase realtime database which stored website user details.

4. Node 1 : RFID Door Lock System

4.1. Connections with nodemcu

4.1.1. RFID reader

RFID reader uses serial communication to communicate with the nodemcu.

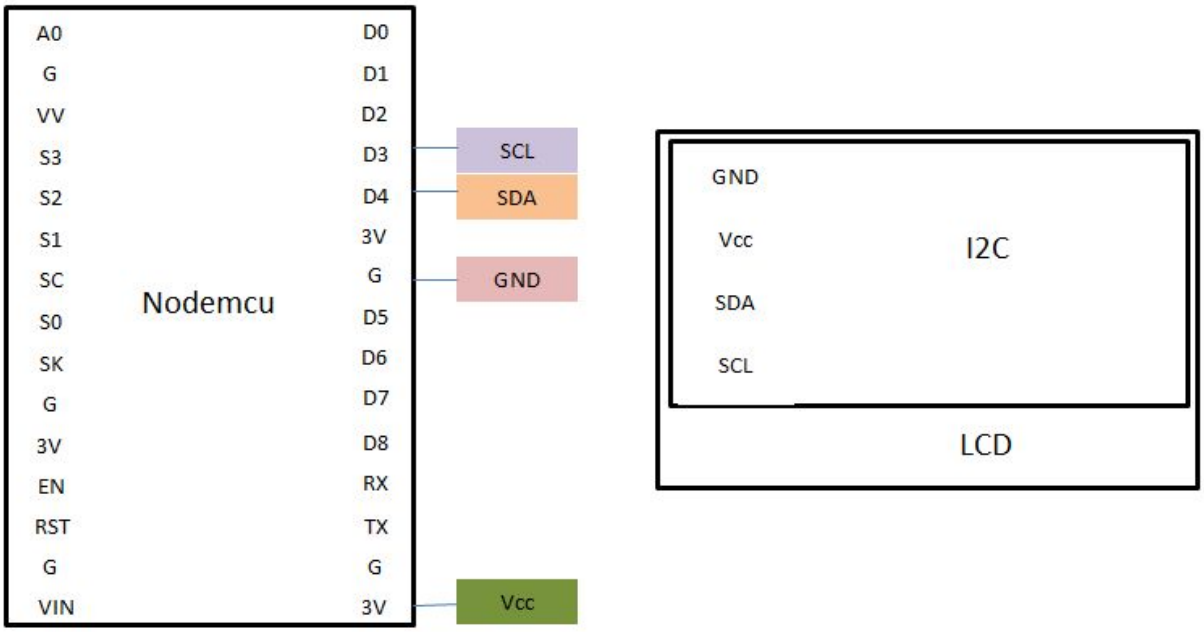


After welding the external pins to the RFID reader they should be connected to the nodemcu as shown in the above diagram.

Then add the library MFRC522 to arduino. (You can download it from here)

To check whether the RFID reader is working fine, upload the example code DumpInfo included in the MFRC522 library. This will help to read the RFID tags and it will be printed on the serial monitor. (Special note: you should change the RST_PIN as 5 and SS_PIN as 15)

4.1.2. LCD display



I2C is a SPI Serial Interface Module for Arduino. And it helps to reduce the 16 pins of LCD display to 4 pins. You should get an I2C which is compatible with your LCD display.

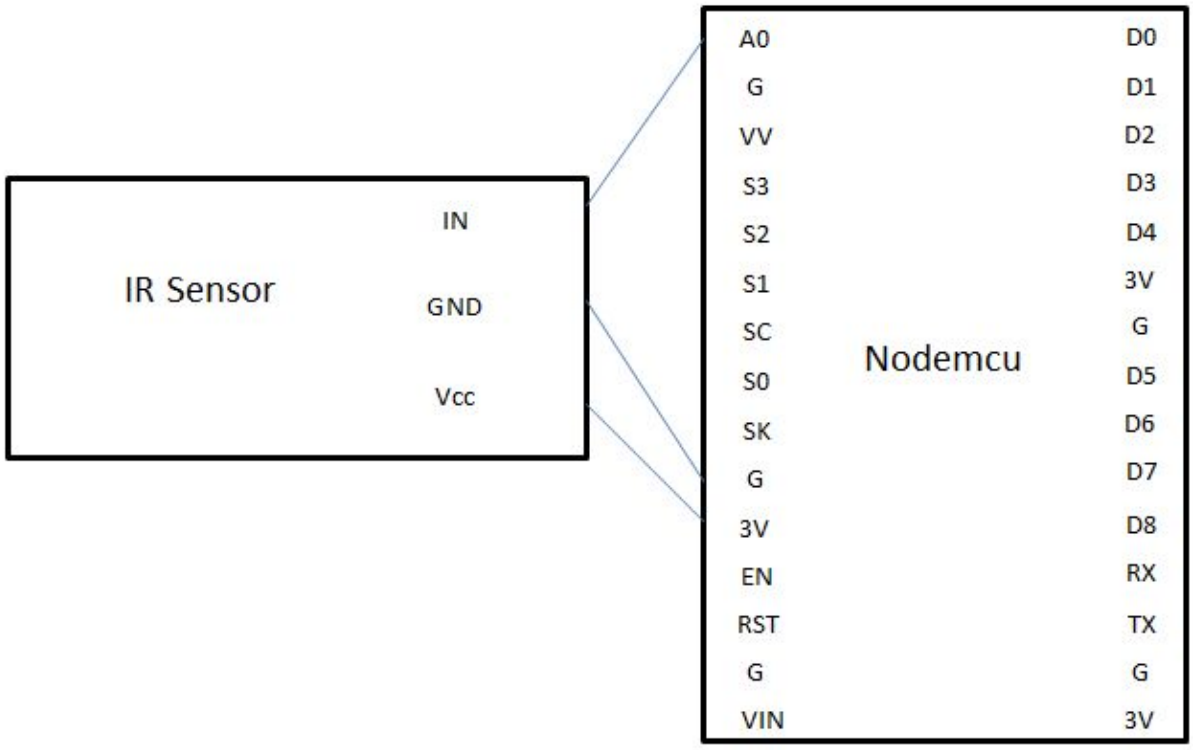
First you should weld the I2C pins to the LCD display.

Then connect it to the nodemcu as shown in the above diagram.

Then add the LiquidCrystal_I2C library to arduino. (you can download it from here)

To check whether it is working fine, upload the example code Helloworld which is included in the LiquidCrystal_I2C library. It should display some text in the LCD display. (Special Note :change this as LiquidCrystal_I2C lcd(0x27,16,2))

4.1.3. IR sensor



It is used to avoid any unauthorized access.

Connect the IR sensor as shown in the above diagram.

To check whether it is working fine, upload the simple code below.

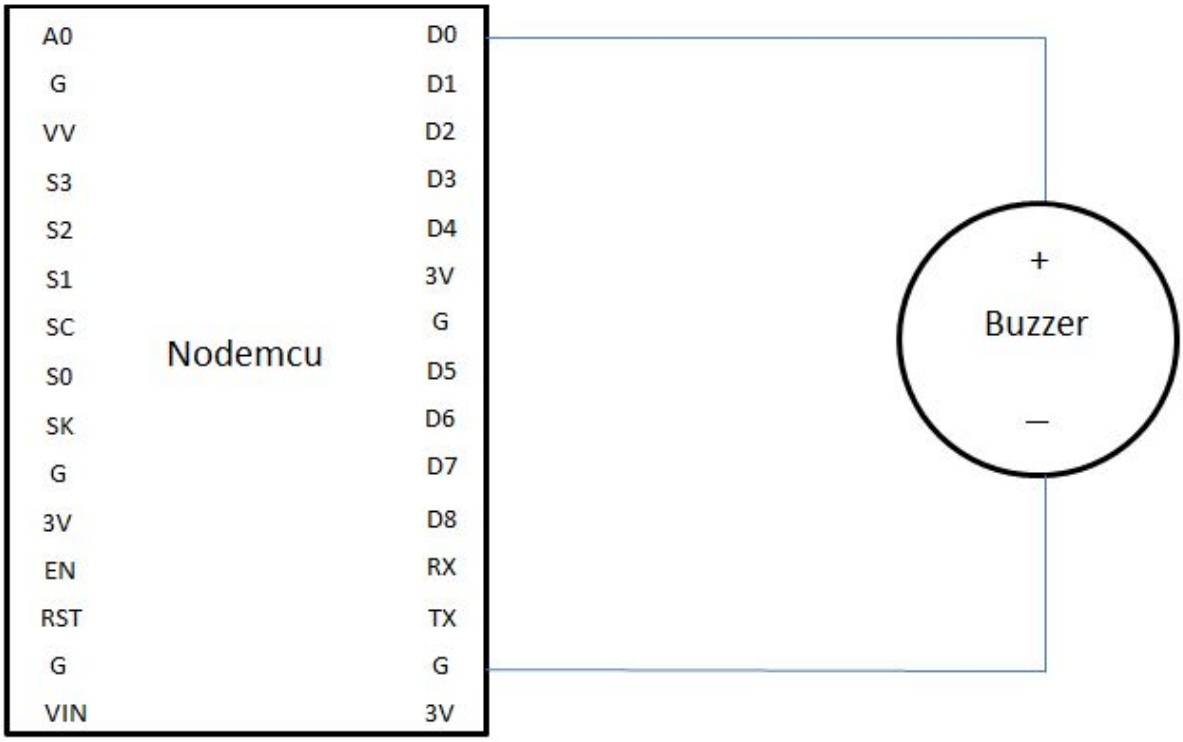

```
#define IRSensor  A0

void setup() {
  pinMode(IRSensor,INPUT);
}

void loop() {
 Serial.println(analogRead(IRSensor));
}
```

See how the values are changing when you move your hand in front of the IR sensor.

4.1.4. Buzzer



Connect the buzzer as shown in the above diagram.

To check whether the buzzer is working fine, upload the code below.

sketch_oct06a | Arduino 1.8.10

File Edit Sketch Tools Help

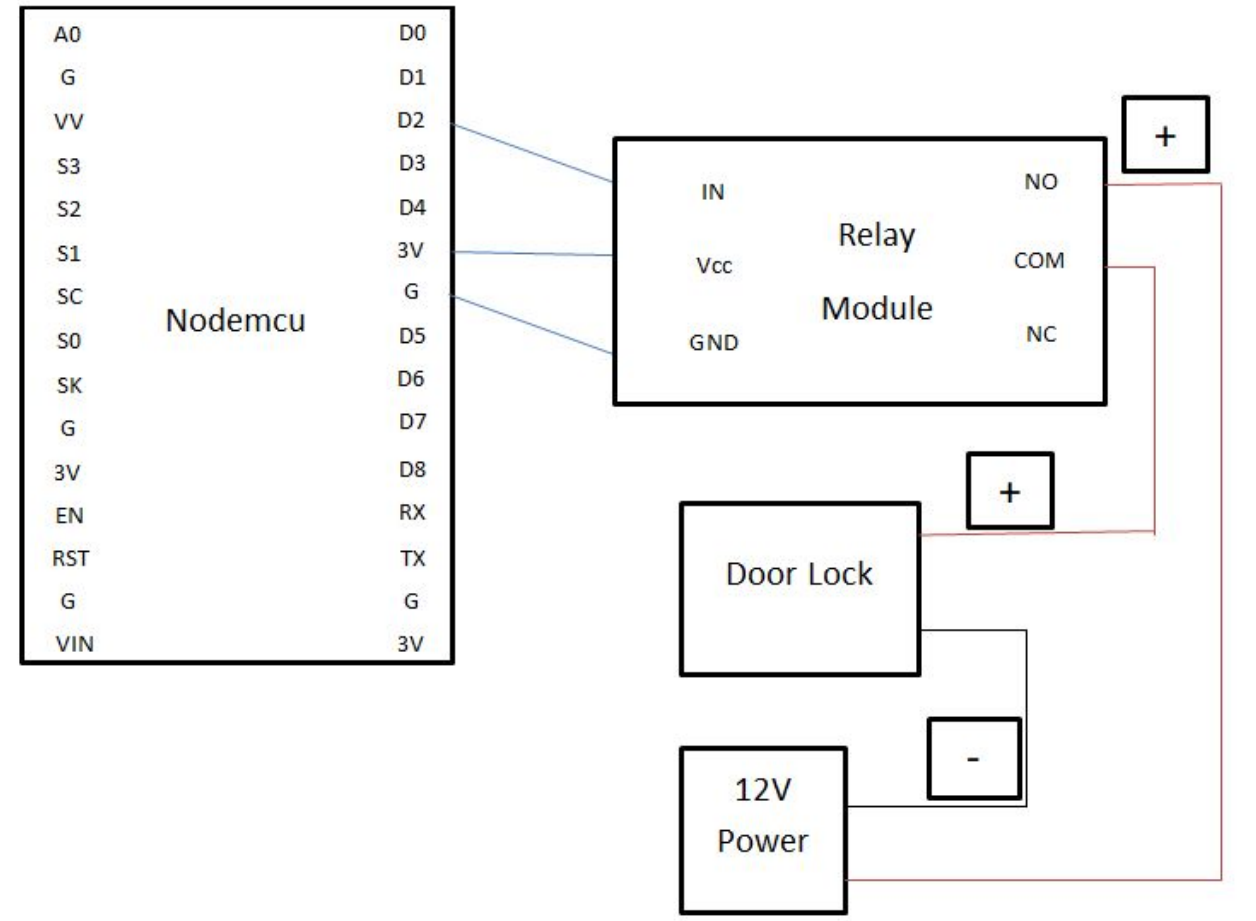
sketch_oct06a §

```
#define buzzer 16 //define buzzer - connected to D0 pin of nodemcu

void setup() {
  pinMode(buzzer, OUTPUT);
}

void loop() {
  digitalWrite(buzzer, HIGH);
  delay(1000);
  digitalWrite(buzzer, LOW);
  delay(1000);
}
```

4.1.5. Solenoid Door Lock



Connect all the components as shown in the above diagram.

To check whether the solenoid door lock works fine, upload the below code.

sketch_oct06a | Arduino 1.8.10

File Edit Sketch Tools Help

sketch_oct06a §

```
int relay_pin = 4; // define the relay input pin D4

void setup() {
  pinMode( relay_pin , OUTPUT);// set the relay pin to output
  digitalWrite( relay_pin , HIGH);//close the door
}

void loop() {
  digitalWrite( relay_pin , LOW);//door open
  digitalWrite( relay_pin , HIGH);//close the door
}
```

4.2. Completed Node 1

Now connect all the components (RFID,IR,Buzzer and Door Lock) with the nodemcu as stated above.

Then download our code, from here.

In the code change the WIFI_SSID <your wifi name> and WIFI_PASSWORD <your wifi password>

You have to download and install ArduinoJason library as it is used to retrieve the firebase data from the database as a jason object. (You can download it from here)

You have to download and install NTPClient library as it is used to calculate the access date and time. (You can download it from here)

Finally upload the code to the nodemcu and enjoy the product.

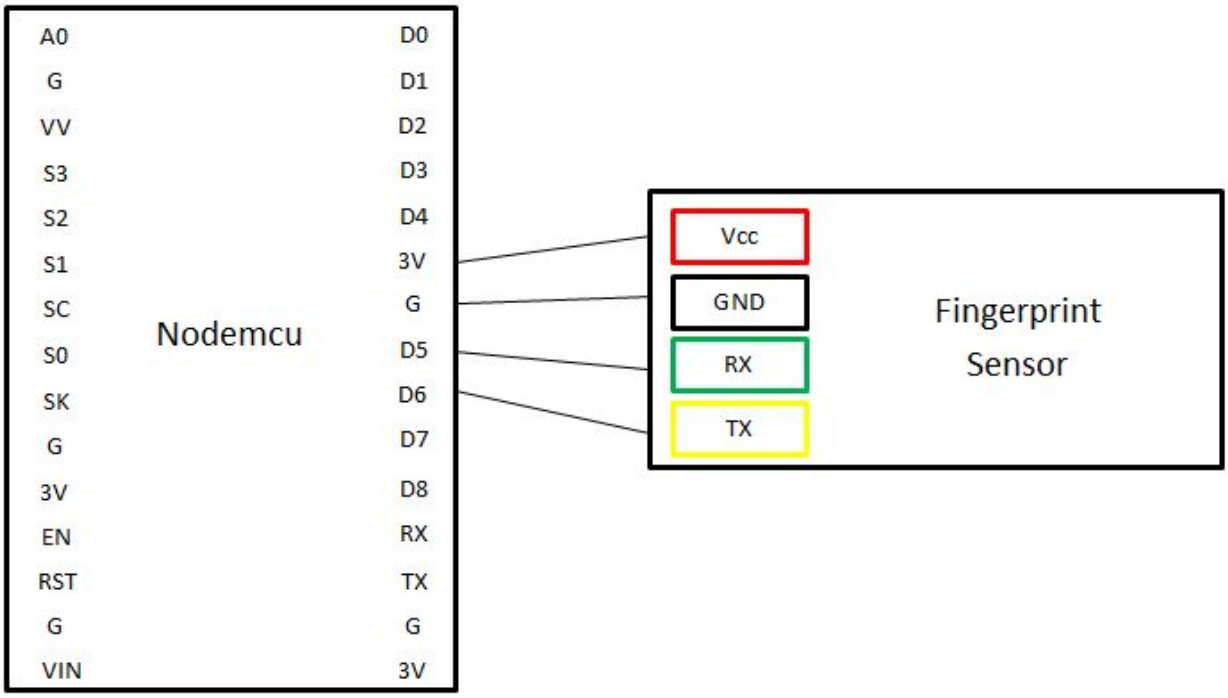
4.2.1. Data structures and algorithms

When retrieving the data from the database, they are retrieved as a json object.

5. Node 2 : Fingerprint Door Lock System

5.1. Connections with nodemcu

5.1.1. Fingerprint sensor



Connect the fingerprint sensor with the nodemcu as shown above.

To check whether the fingerprint sensor is working fine use the following codes.
For the enrollment of fingerprints
For the identification of stored fingerprints

You can watch a sample demo of how fingerprint sensor functions with the above codes from here.

In this node we are not going to use RFID reader, instead we are using the fingerprint sensor. Apart from RFID reader all the other components and the connections are the same as in node 1.

5.2. Completed Node 2

Now connect all the components (Fingerprint Sensor,IR,Buzzer and Door Lock) with the nodemcu as stated above.

Then download our code, from here.

In the code change the WIFI_SSID <your wifi name> and WIFI_PASSWORD <your wifi password>

You have to download and install ArduinoJason library as it is used to retrieve the firebase data from the database as a jason object. (You can download it from here)

You have to download and install NTPClient library as it is used to calculate the access date and time. (You can download it from here)

Finally upload the code to the nodemcu and enjoy the product.

5.2.1. Data structures and algorithms

When retrieving the data from the database, they are retrieved as a json object.

6.Testing

6.1. Performance Testing

Objective: Measure minimum response time between reading two authorized RFID tags

Inputs: RFID tags and our system
Measured Parameters: Response time
Expected Output: Minimum response time
Assumptions made:

- We check 10 times and take response time at each time. And assume that minimum of those 10 is the minimum response time between reading two authorized RFID tags.
- Assumed human errors at each time are the same.

Testing Process: In the code take the started time when the LCD displays WELCOME and display the time in LCD display. Then take the end time when it again displays WELCOME and display the time in the display.
The difference will be the response time. Repeat this 10 times and take minimum response time from those 10.

6.2. Load Testing for web application

Objective: Discover maximum number of concurrent users that system can handle
Inputs: web application of our system
Measured Parameters: Maximum number of concurrent users that system can handle
Assumptions made:

- The values depend on several factors like current server load, our internet speed, our CPU power etc. Hence, it's very unlikely that we will get the same results if these factors change. So, we assume that these factors do not get changed.

Testing Environment: An environment that Current server load, internet speed, CPU power etc. kept nearly constant without any change.

Method:
Using the tool “Apache Jmeter” analyze the performance using throughput because throughput represents the ability of the server to handle the heavy load.
We are doing a load analysis of our website for a specific number of users.

Before testing our web application, we should determine-

- Normal Load: Average number of users visit the website
- Heavy Load: The maximum number of users visit the website
- What is our target in this test?

Here is the roadmap of how we are going to do the test,

1. Add Thread Group
2. Adding JMeter elements

3. Adding Graph result
4. Run Test and get the test result

6.3. Unit Testing

Objective: Form validation – Validity of passwords
Measured Parameters: Length of a password

Input	Expected Output
Password length < 6	Weak Password
6 < Password length < 10	Average Password
Password length > 10	Strong Password

Testing Process:
We add javascript form validation to our web site so that it can handle the password length.