```
/*
   Created by TheCircuit
*/
#include<Servo.h>
Servo servo;

#define SS_PIN 15  //D8
#define RST_PIN 9

#include <SPI.h>
#include <MFRC522.h>
#include <LiquidCrystal.h>


MFRC522 mfrc522(SS_PIN, RST_PIN);   // Create MFRC522 instance.
int statuss = 0;
int out = 0;

LiquidCrystal lcd(9, 10, 2, 4, 5, 1); //These are the nos. of the gpios corresponding to the nodemcu

int len = 2;
int count[] = {0, 0};
int p = 0, turn = 0;
unsigned long int t = 0;

void setup()
{
  Serial.begin(9600);   // Initiate a serial communication
  servo.attach(3);
  servo.write(45);
  SPI.begin();      // Initiate  SPI bus
  mfrc522.PCD_Init();   // Initiate MFRC522
  lcd.begin(16, 2);
  lcd.clear();
  lcd.print("Welcome to");
  lcd.setCursor(0, 1);
  lcd.print("Robotics Room");
}

String card_id[2] = {"B9 02 56 73", "3D BE 40 52"};
String card_id_person[2] = {"Santanu Banerjee", "Gaurav Bansal"};

String close_id = "B5 1C 1A 77";

void loop()
{
  p = 0;
  for ( int j = 0; j < 2; j++)
```

```
{
  if (count[j] == 1)
    p++;
}
if (millis() - t > 1000 && p == 0)
{
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Welcome to");
  lcd.setCursor(0, 1);
  lcd.print("Robotics Room.");
}
else if (millis() - t > 2000 && turn == 0)
{
  turn = 1;
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Welcome to");
  lcd.setCursor(0, 1);
  lcd.print("Robotics Room.");
  t = millis();
}
else if (millis() - t > 1000 && turn == 1)
{
  turn = 0;
  lcd.clear();
  lcd.print("No. of persons=");
  lcd.setCursor(0, 1);
  lcd.print(" ");
  lcd.print(p);
  t = millis();
}
// Look for new cards
if ( ! mfrc522.PICC_IsNewCardPresent())
{
  return;
}
// Select one of the cards
if ( ! mfrc522.PICC_ReadCardSerial())
{
  return;
}

//Show UID on serial monitor
//Serial.println();
//Serial.print(" UID tag :");
String content = "";
byte letter;
```

```
for (byte i = 0; i < mfrc522.uid.size; i++)
{
  //Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
  //Serial.print(mfrc522.uid.uidByte[i], HEX);
  content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
  content.concat(String(mfrc522.uid.uidByte[i], HEX));
}
content.toUpperCase();
//Serial.println();
int i;
for (i = 0; i < len; i++)
{
  if (content.substring(1) == close_id)
  {
     lcd.clear();
     delay(100);
     lcd.print("Door Closed");
     t = millis();
     servo.write(45);
     p = 0;
     for(int j = 0; j < len; j++)
       count[j] = 0;
     break;
  }
  else if (content.substring(1) == card_id[i]) //change UID of the card that you want to give access
  {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Access Granted");
    delay(1500);
    lcd.clear();
    delay(100);
    lcd.setCursor(0, 0);
    if (count[i] == 0)
    {
      lcd.print("Welcome");
      lcd.setCursor(0, 1);
      lcd.print(card_id_person[i]);
      delay(1000);
      lcd.clear();
      delay(100);
      if ( p == 0 )
        lcd.print("Opening door..");
      else
        lcd.print("Door open.Go in.");
      t = millis();
      //lcd.clear();
      count[i] = 1;
```

```
      servo.write(125);
    }
    else if (count[i] == 1)
    {
      lcd.print("Thanks");
      lcd.setCursor(0, 1);
      lcd.print("For Visit");
      delay(1000);
      lcd.clear();
      delay(100);
      lcd.print("GoodBye");
      lcd.setCursor(0, 1);
      lcd.print(card_id_person[i]);
      delay(1000);
      lcd.clear();
      delay(100);
      if (p == 1)
        lcd.print("Close door");
      t = millis();
      count[i] = 0;
      if(p == 1)
        servo.write(45);
    }

    break;
   }
  }
  if (i == len)
  {
   lcd.clear();
   delay(100);
   lcd.print("Access Denied");
   delay(2000);
  }
}
```