

**گزارش پروژه**

**عنوان پروژه :IOT**

**دانشجو : مصطفی دست غیبی شیرازی**

**استاد راهنما : دکتر محمد خانی**

**مقطع : کارشناسی نرم افزار**



# فهرست

[فهرست 3](#_Toc155439092)

[چکیده 4](#_Toc155439093)

[مقدمه 5](#_Toc155439094)

[مفهوم اینترنت اشیاء 6](#_Toc155439095)

[قطعات مورد استفاده در پروژه 7](#_Toc155439096)

[Wemos esp8266 D1 Mini 7](#_Toc155439097)

[ماژول رله 5 ولت دو کاناله 11](#_Toc155439098)

[ماژول کارت خوان RC522 13](#_Toc155439099)

[شرح و اهدف پروژه 15](#_Toc155439100)

[زبان های مورد استفاده در این پروژه 16](#_Toc155439101)

[نحوه کار پروژه 17](#_Toc155439102)

[لینک ها 18](#_Toc155439103)

[تصاویر 19](#_Toc155439104)

[کد های esp8266 22](#_Toc155439105)

# چکیده

در این پروژه قصد داریم وسایل مختلفی از جمله درب های ورودی ، روشنایی ، درب پارکینگ و دستگاه های مختلف را کنترل و هوشمند سازی کنیم .

همچنین کاربر هنگامی که بخواهد از وب استفاده کند یک احراز هویتی صورت می گیرد.

**در این پروژه کاربر می تواند از دو طریق کنترل داشته باشد :**

1-بر بستر وب

2-RFID

**کلمات کلیدی :**

* Iot
* internet of things
* هوشمند سازی
* اینترنت اشیاء

# مقدمه

مصطفی دست غیبی شیرازی با شماره دانشجویی 00221033705005 هستم که در ترم 4 مقطع کارشناسی رشته نرم افزار در دانشگاه شهید شمسی پور مشغول به تحصیل هستم و واحد پروژه را با موضوع iot (اینترنت اشیا ) اخذ کردم .

با تشکر از استاد راهنما دکتر محمدخانی که مرا در این پروژه راهنمایی کردند .

در این پروژه من سعی کردم تمام علاقه مندی ها و دانش و مهارت های خودم که شامل (نرم فزار ، الکترونیک ، شبکه، سخت افزار ) بود را مورد استفاده و بهره برداری قرار دهم و این پروژه را به نحو احسنت و توانم به پایان رسانم .

# مفهوم اینترنت اشیاء

اینترنت اشیاء( Internet of Things) که گاهی به نام اینترنت اجسام(Internet of Objects) یا به اختصار I.O.T نیز نامیده می‌شود، به ارتباط اینترنتی بین اشیاء و تجهیزاتی می‌گویند که در محیط پیرامون ما قرار دارند.

این لوازم یا اشیاء متصل به شبکه اینترنت می‌توانند با استفاده از نرم‌افزار‌های موجود در تلفن‌های هوشمند، تبلت‌ها، رایانه‌ها، گجت یا ابزارک‌ها، ساعت‌های هوشمند، تلویزیون‌ها و هر شیء دیگری به صورت کنترل از راه دور مدیریت شوند.

اینترنت اشیاء به معنای ارتباط گیرنده‌ها و دستگاه‌ها با شبکه اینترنت است که از طریق این ارتباط و تعامل بین لوازم متصل به شبکه و کاربران دارای دسترسی مجاز به این شبکه، امکان دیدن و کنترل لوازم متصل به شبکه برای کاربران آن فراهم می‌شود. این مفهوم می‌تواند به سادگی ارتباط یک گوشی هوشمند با تلویزیون یا به پیچیدگی نظارت بر زیرساخت‌های شهری و ترافیکی باشد. این شبکه بسیاری از دستگاه‌های اطراف ما را از جمله ماشین لباس شویی و یخچال گرفته تا پوشاک را دربرمی‌گیرد.

به عبارتی دیگر اینترنت اشیاء یعنی مجموعه‌ای از اشیا و تجهیزات که به وسیله اتصال به یک شبکه با یکدیگر ارتباط برقرار می‌کنند. این فناوری می‌تواند به مصرف‌ کنندگان خدمات گوناگونی ارائه کند، برای مثال می‌توان به کنترل برخی از وسیله‌ها از راه دور، اطلاع از شرایط محیطی خانه و غیره اشاره کرد.

# قطعات مورد استفاده در پروژه

# **Wemos esp8266 D1 Mini**

برد Wemos D1 Mini یک برد توسعه یافته برای اجرای برنامه‌های کاربردی برای اینترنت اشیا IoT است. این برد بر اساس میکروکنترلر ESP8266 طراحی شده و دارای ابعاد کوچکی است که به راحتی در پروژه‌های کوچک و متوسط ​​قابل استفاده است. Wemos D1 Mini از مجموعه ای از پایه ها و ماژول ها تشکیل شده است که شامل ورودی/خروجی دیجیتال و آنالوگ، ارتباط WiFi، پورت USB، و رابط‌های شیلد قابل ارتقا و اتصال است. این برد با استفاده از زبان برنامه نویسی Arduino IDE و یا MicroPython قابل برنامه ریزی است. از ویژگی های وموس دی 1 مینی می توان به قیمت ارزان، کارایی بالا، طراحی کوچک، مصرف کم انرژی، و کتابخانه های آماده اشاره کرد که موجب شده است برای کاربرانی که می خواهند به سادگی پروژه های IoT را بر روی اینترنت بیاورند یکی از انتخاب های مناسب باشد. به عبارت دیگر اگر بردی با امکانات Nodemcu ولی کوچکتر نیاز دارید، این برد انتخاب مناسبی خواهد بود. این برد فقط دارای یک پایه آنالوگ با حداکثر ولتاژ ورودی 3.2 ولت است. از پورت micro usb بهره برده و با میکروپایتون و آردوینو قابل راه اندازی است. ابعاد این این برد در حدود 34.2 × 25.6 میلی‌متر است. با توجه به دیتاشیت این برد سرعت کلاک MHz 80/160 است.

**مشخصات فنی برد wemos D1 Mini**

ولتاژ عملیاتی: 3.3 ولت DC

ورودی DC: 5 ولت DC از طریق پورت USB

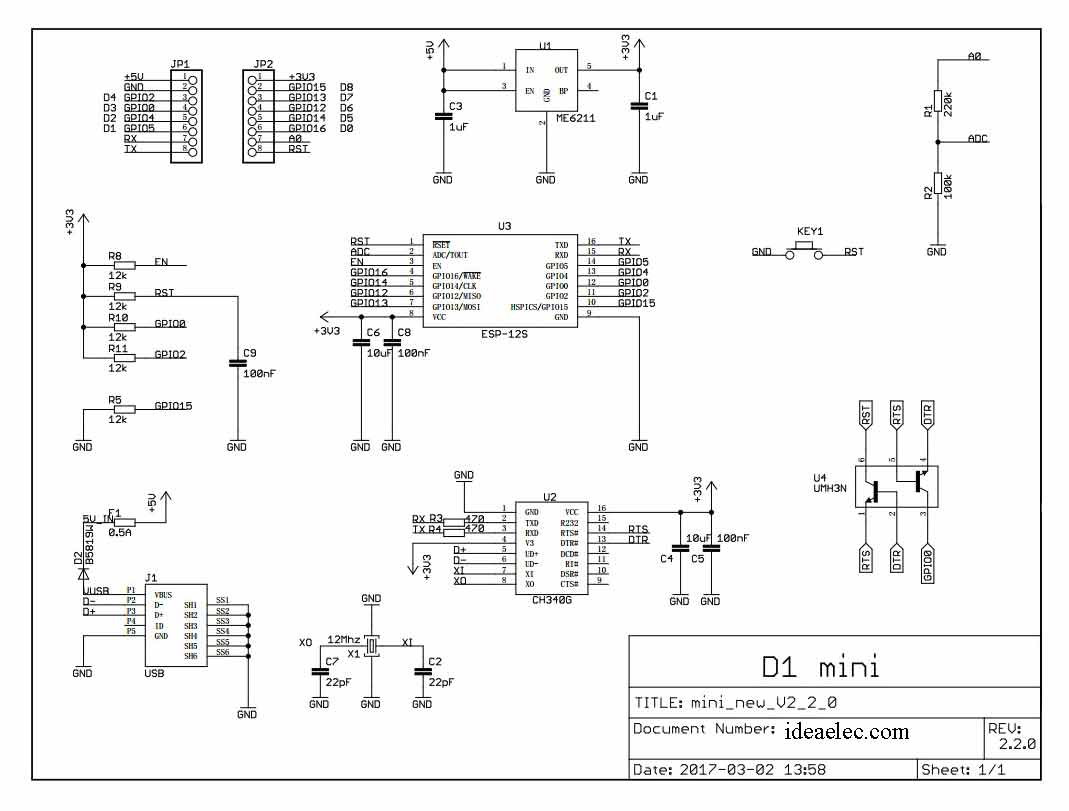
ورودی آنالوگ: 1 پورت ورودی آنالوگ با دقت 10 بیت (ADC) با ورودی حداکثر 3.2 ولت

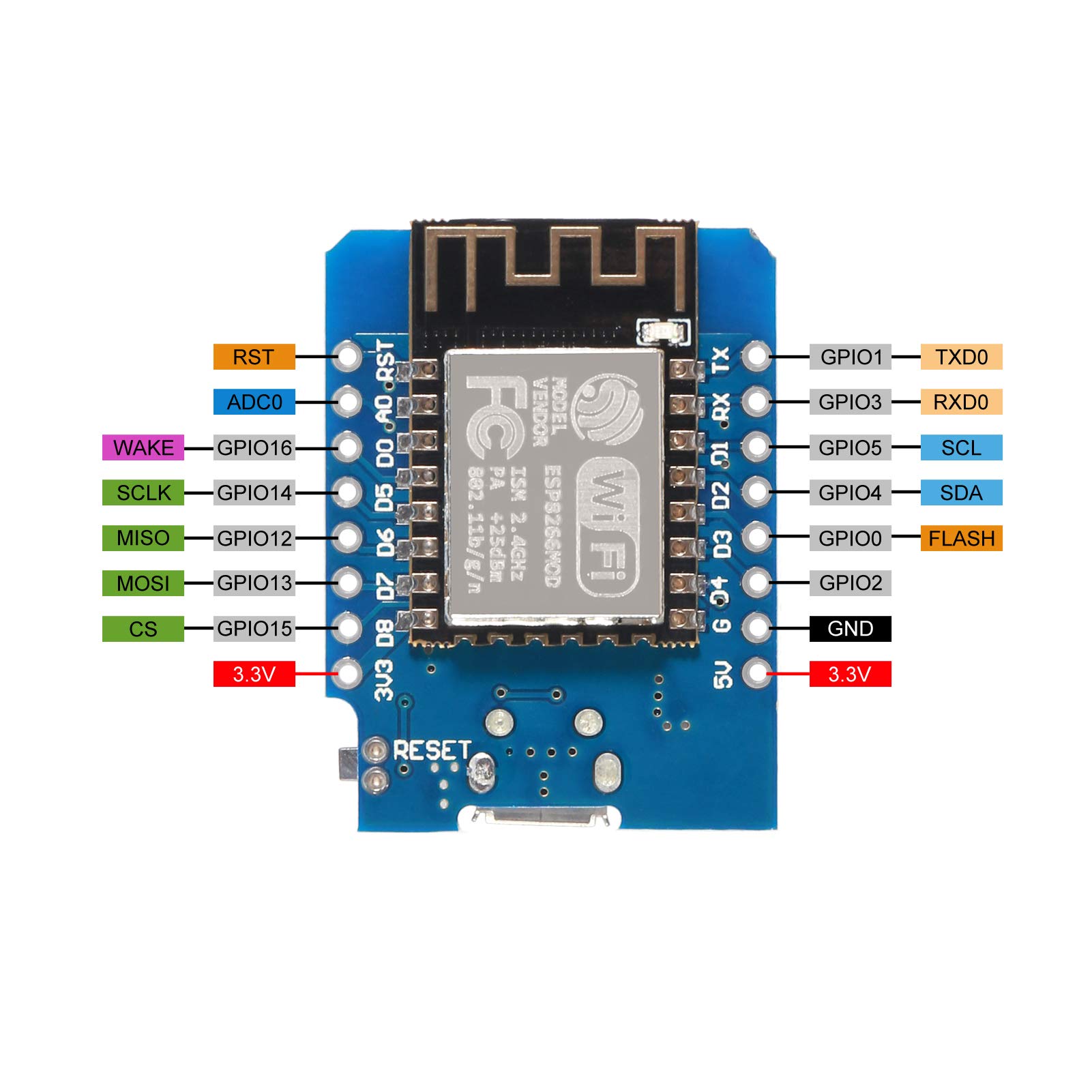
ورودی دیجیتال: 11 پورت ورودی/خروجی دیجیتال با پشتیبانی از سطوح ولتاژ 3.3 ولت

رابط شبکه: Wi-Fi 802.11 b/g/n

پورت USB: Micro USB برای اتصال به کامپیوتر و تغذیه از طریق آن

حافظه: 4MB فلش داخلی

ابعاد: 34.2mm x 25.6mm x 7.2mm

**بررسی پایه های ماژول Wemos D1 mini**

VCC: منبع تغذیه +3.3 ولتی

GND: زمین

D0: پایه تغذیه (خروجی با دقت 3.3 ولت)

D1: پایه GPIO5 و همچنین پایه SCL برای رابط I2C

D2: پایه GPIO4 و همچنین پایه SDA برای رابط I2C

D3: پایه GPIO0، همچنین برای راه اندازی با کد فلشر در حالت Boot استفاده می شود.

D4: پایه GPIO2 و همچنین خروجی 3.3 ولتی که برای کنترل LED به صورت پیشفرض در نظر گرفته شده است.

D5: پایه GPIO14 و همچنین پایه CLK برای رابط SPI

D6: پایه GPIO12 و همچنین پایه MISO برای رابط SPI

D7: پایه GPIO13 و همچنین پایه MOSI برای رابط SPI

D8: پایه GPIO15 و همچنین پایه SS برای رابط SPI، همچنین می‌توان آن را برای کنترل خروجی های دیجیتال ورودی/خروجی استفاده کرد.

RX: پایه ورودی سریال

TX: پایه خروجی سریال

A0: پایه ورودی آنالوگ

# ماژول رله 5 ولت دو کاناله

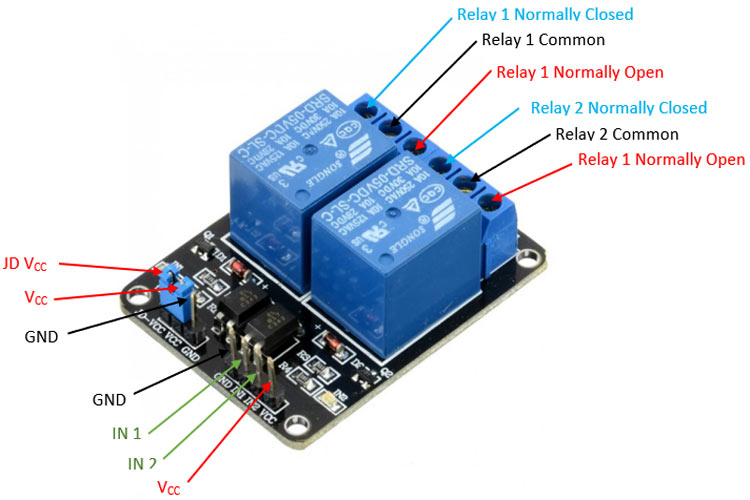
ماژول رله دو کانال، یک دستگاه الکترونیکی است که دارای دو کانال خروجی رله است. هر کانال از این ماژول می‌تواند برای کنترل یک مدار الکتریکی یا الکترونیکی استفاده شود.رله‌ها از نوع سوئیچ هستند که به وسیله‌ی جریان الکتریکی، می‌توانند یک مدار را روشن یا خاموش کنند. ماژول رله دو کانال می‌تواند به عنوان یک واسط بین مدارهای الکتریکی و کنترل کننده‌ی آنها مورد استفاده قرار گیرد.استفاده از ماژول رله دو کانال در سیستم‌های خانگی، صنعتی و تجاری بسیار رایج است و به کاربران امکان کنترل دستگاه‌های مختلف را با استفاده از میکروکنترلر، ماژول‌های سنسور و دیگر وسایل الکترونیکی می‌دهد.

**ویژگی های ماژول رله 5 ولت دو کاناله**

* دارای دو کانال جهت کنترل دو وسیله برقی
* قابلیت تحمل تا 250 ولت AC در خروجی رله
* مجهز به اپتوکوپلر جهت ایزولاسیون و حذف نویز
* دارای ولتاژ ورودی 5 ولت جهت فرمان به خروجی
* قابلیت تحمل تا 10 آمپر جریان خروجی به ازای هر کانال
* تعداد کانال: دو کانال
* نوع رله: رله‌های نیمه‌هادی (solid-state)   
  ولتاژ کاری: 5 ولت DC یا 12 ولت DC  
  جریان کاری: تا 10 آمپر برای هر کانال  
  ولتاژ خروجی: حداکثر 250 ولت AC و 30 ولت DC برای هر کانال  
  تحمل جریان بار: حداکثر 10 آمپر برای هر کانال  
  زمان واکنش: کمتر از 10 میلی‌ثانیه

**بررسی پایه های ماژول رله دو کانال**

پایه‌های ماژول رله دو کانال بسته به نوع ماژول و تولید کننده متفاوت باشد، اما برای بیان یک نمونه، پایه‌های ماژول رله دو کانال را به صورت زیر می‌توان بیان کرد:

* VCC:پایه‌ای که ولتاژ تغذیه به ماژول رله دو کانال متصل می‌شود. معمولاً این پایه به ولتاژ 5 ولت یا 12 ولت DC نیاز دارد.
* GND:پایه‌ای که به زمین متصل می‌شود.
* IN1:پایه ورودی برای کنترل رله اول. این پایه به یک پورت دیجیتال از
* : :پایه مشترک برای رله اول.
* NO1:پایه خروجی آزاد برای رله اول. اگر رله فعال باشد، این پایه به COM1 متصل می‌شود.
* NC1:پایه خروجی بسته برای رله اول. اگر رله غیرفعال باشد، این پایه به COM1 متصل می‌شود.
* IN2:پایه ورودی برای کنترل رله دوم. این پایه به یک پورت دیجیتال از میکروکنترلر، برد آردوینو یا سایر سیستم‌های کنترلی متصل می‌شود.
* COM2:پایه مشترک برای رله دوم.
* NO2:پایه خروجی آزاد برای رله دوم. اگر رله فعال باشد، این پایه به COM2 متصل می‌شود.
* NC2 :پایه خروجی بسته برای رله دوم. اگر رله غیرفعال باشد، این پایه به COM2 متصل می‌شود.

# ماژول کارت خوان **RC522**

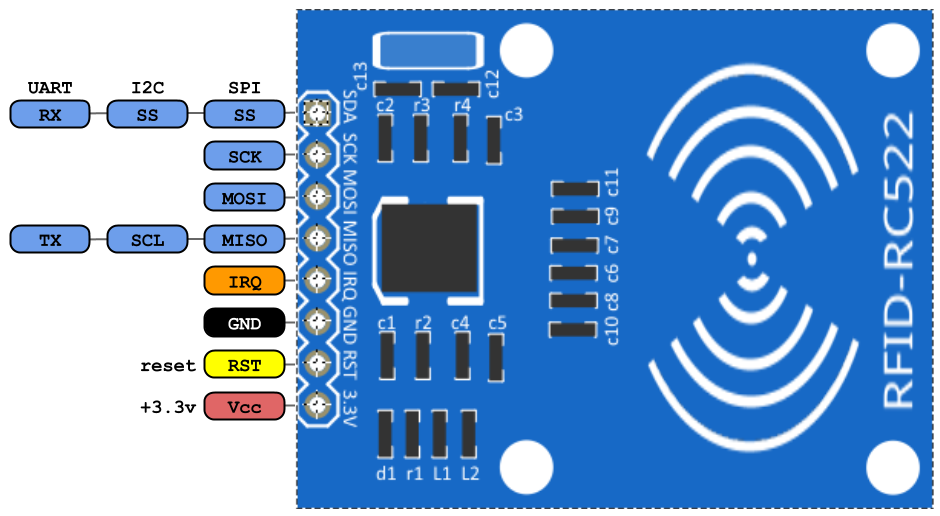
ماژول کارت خوان RFID RC522 یک ماژول کارت‌خوان NFC/RFID با فرکانس 13.56 مگاهرتز است که به راحتی می‌تواند انواع کارت‌های RFID و NFC را تشخیص دهد و اطلاعات موجود در آن‌ها را بخواند.این ماژول از یک الگوریتم تشخیص و ارتباط با کارت‌های RFID و NFC با نام MFRC522 پشتیبانی می‌کند و با اتصال به میکروکنترلرهایی مانند Arduino و ESP8266، به راحتی می‌تواند از آن برای پروژه‌های مختلفی استفاده کرد. از کاربردهای این ماژول می‌توان به پروژه‌های امنیتی و اعتبارسنجی مثل درب بازکن اتوماتیک، سیستم حضور و غیاب و سیستم‌های دسترسی فیزیکی اشاره کرد. همچنین این ماژول در صنایع مختلفی مانند حمل و نقل، پیشرفته‌سازی شهری و هوشمندسازی ساختمان‌ها نیز کاربرد دارد.

**مشخصات فنی RFID RC522**

* فرکانس عملیاتی: 13.56 مگاهرتز
* محدوده فاصله از کارت: تا 3 سانتی‌متر
* پشتیبانی از کارت‌های RFID: MIFARE 1 S50، MIFARE 1 S70، MIFARE Ultralight، MIFARE Pro و NTAG203
* پشتیبانی از کارت‌های NFC: NTAG213 و NTAG215
* پشتیبانی از رابط SPI با سرعت بالا
* دارای یک آنتن کوچک داخلی جهت ارتباط با کارت‌ها
* دارای 8 پین ورودی و خروجی برای ارتباط با میکروکنترلرها
* ولتاژ کاری: 3.3 ولت
* جریان مصرفی: کمتر از 13 میلی‌آمپر
* ابعاد فیزیکی: 60x40 میلی‌متر

**بررسی پایه های RFID RC522**

پایه‌های برد RFID RC522 به شرح زیر می‌باشد:

* :GNDاتصال به منبع منفی (گراند)
* RSTا:تصال به یک پایه دلخواه در آردوینو
* 3.3V: اتصال به منبع تغذیه با ولتاژ 3.3 ولت
* :SDAاتصال به پایه دیجیتال دلخواه در آردوینو
* :SCKاتصال به پایه دیجیتال دلخواه در آردوینو
* :MOSIاتصال به پایه دیجیتال دلخواه در آردوینو
* :MISOاتصال به پایه دیجیتال دلخواه در آردوینو
* :IRQ
* اتصال به یک پایه دلخواه در آردوینو

# شرح و اهدف پروژه

در این پروژه قصد داریم در حوزه اینترنت اشیاء(iot) کار کنیم هدف از این پروژه کنترل درب بر بستر وب و RFID می باشد .

البته که درب در اینجا یک مثال است پس از اتمام پروژه می توانیم هر وسیله ای که با برق 220V کار می کند را به رله وصل و آن را کنترل کنیم .

لازم به ذکر است که هنگامی که کاربر قرار است از وب استفاده کند باید مورد احراز هویت (Authentication) قرار گیرد.

با توجه به موضوع پروژه من بهترین و مناسب ترین برد از نظر (دقت، کیفیت ، هزینه ، کامل بودن امکانات و کتابخانه ) بردesp8266 بود که من هم این برد را انتخاب کردم .

هچنین من برای کنترل وسایل برقی که در ایران از ولتاژ 220v استفاده می کنند باید رله هم برای قطع و وصل جریان استفاده می کردم .

# زبان های مورد استفاده در این پروژه

این پروژه 0 تا 100 کد نویسی بوده و از هیچ CMS بهره نگرفته شده البته که ب راحتی میتوان از آن ها بهره برد تا هم سرعت بالا رفته و هم عملکرد بهینه تری داشته باشیم.

**برخی از زبان های مورد استفاده در پروژه:**

* HTML برای ساخت وب سایت
* CSS برای ساخت وب سایت
* PHP برای backend
* C++ برنامه نویسی esp 82266
* SQL برای مدیریت کاربران

# نحوه کار پروژه

ابتدا که برق 5 ولت را به برد می رسانیم دستگاه شروع به کار کرده و در عرض چند میلی ثانیه (با روشن شدن چراغ سبز ) آماده به کارمی باشد .

همچنین در حال جستجو در ssid های اطراف خود برای برای برقراری اتصال به شبکه (اینترنت) می باشد .

با نزدیک کردن nfc و یا کنترل از طریق وب می توان رله را کنترل کرد .

# لینک ها

**آدرس ایمیل :** [mostafa.ms607@gmail.com](mailto:mostafa.ms607@gmail.com)

**آدرس وب سایت برای کنترل پروژه :**

<https://payner.ir/Project/>

**User:**admin

**passwod:**admin

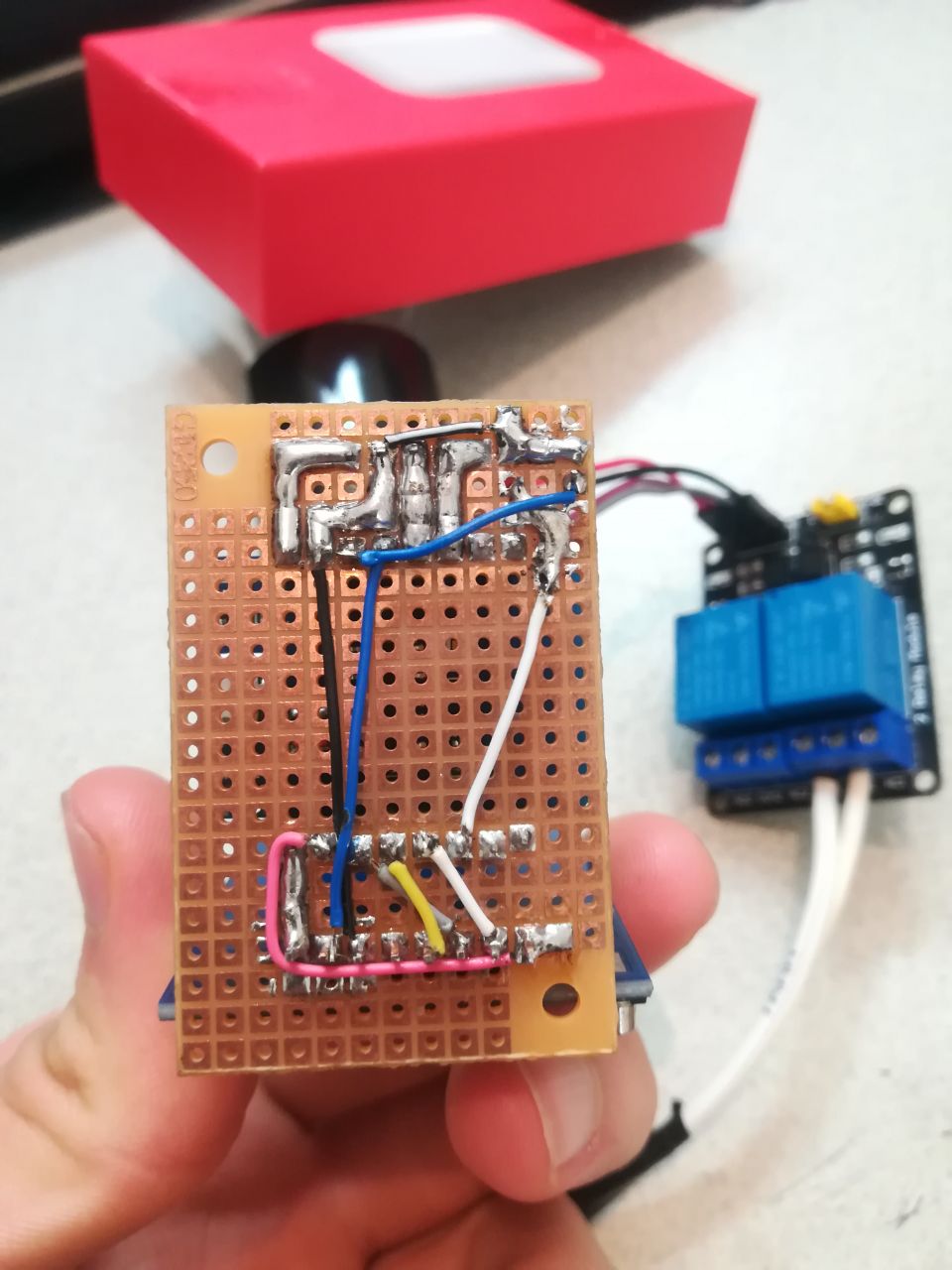
**آدرس گیت هاب برای دریافت کدها :** <https://github.com/mostafadastghibi/IoT>

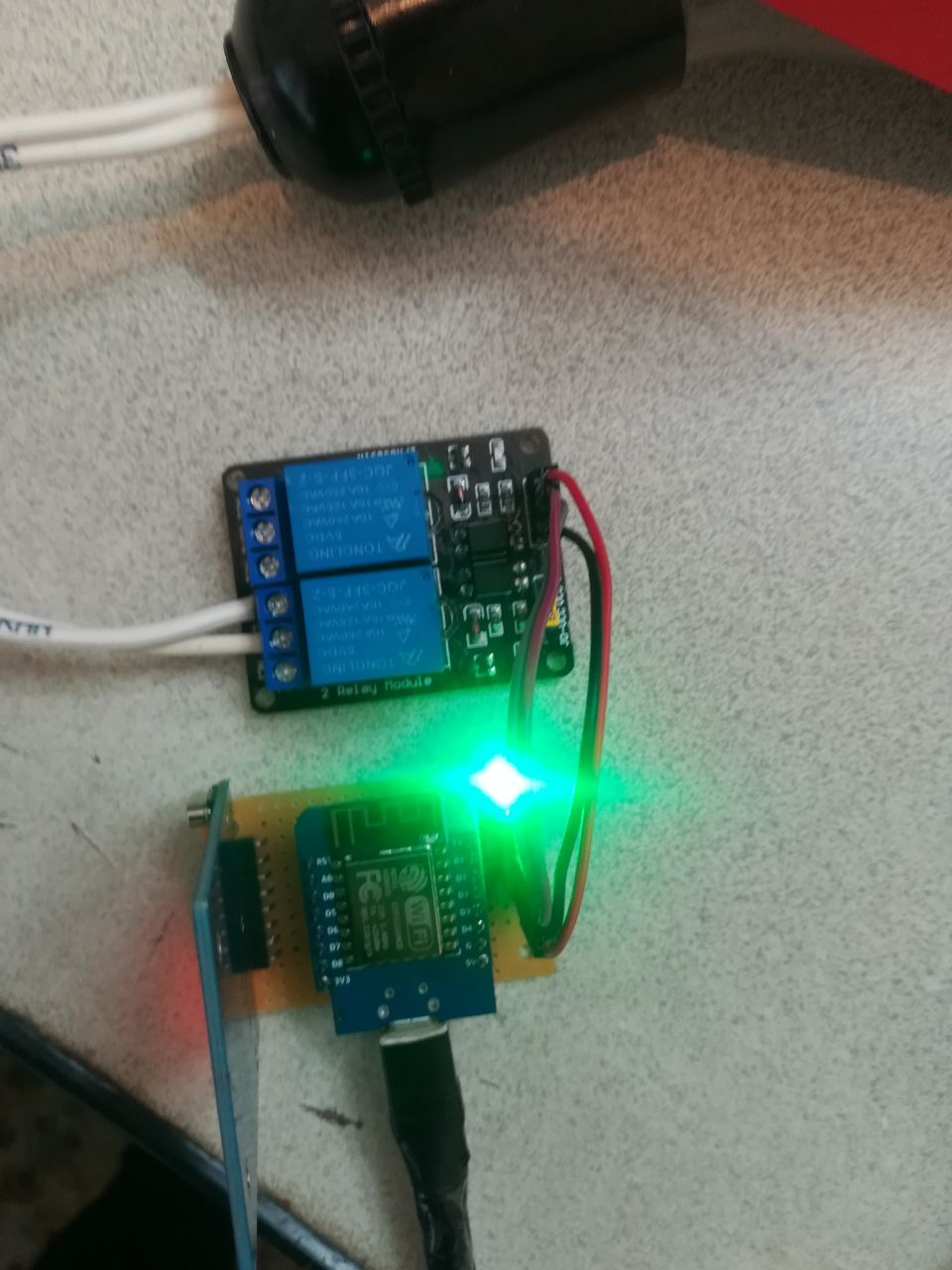
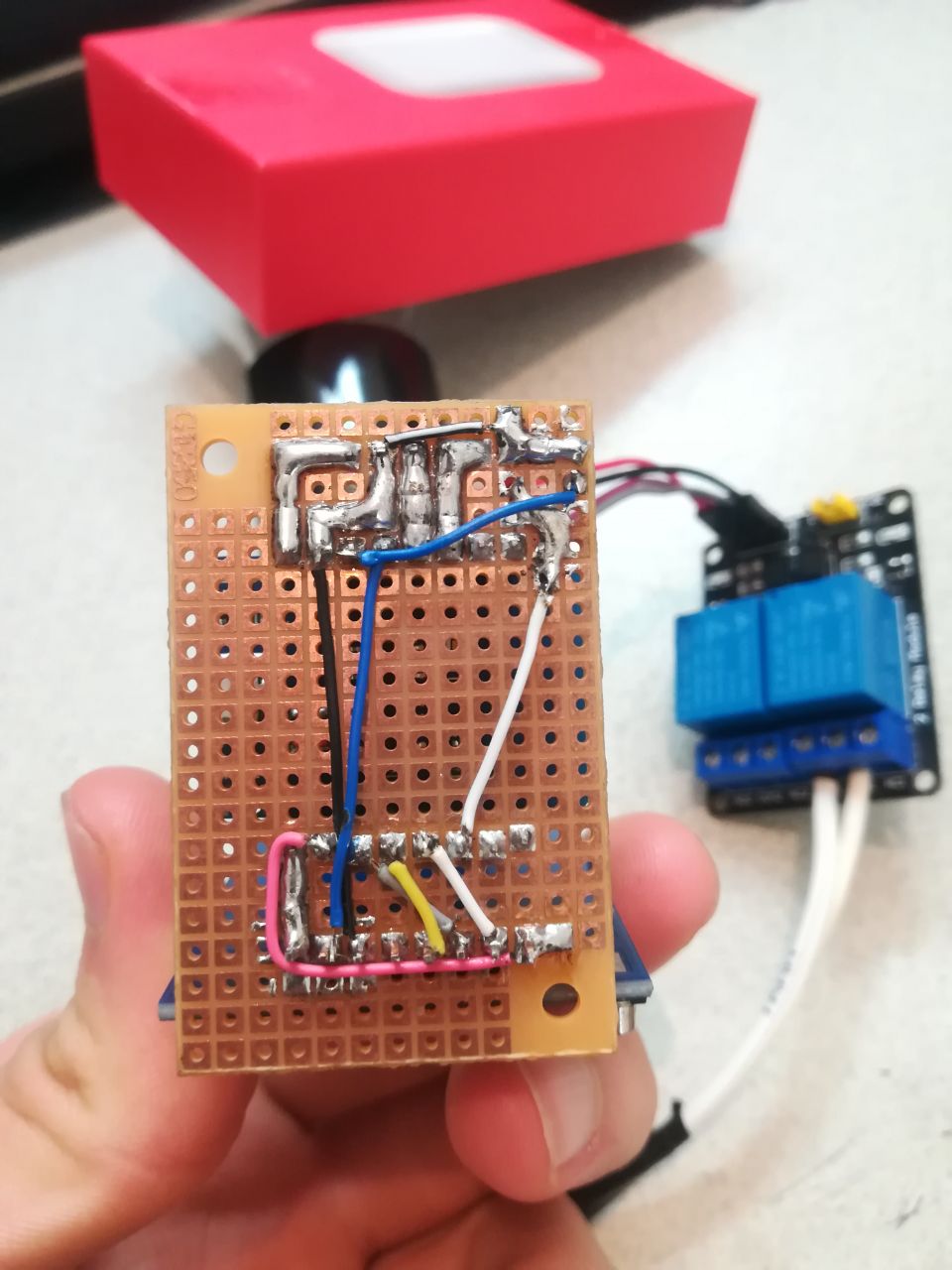
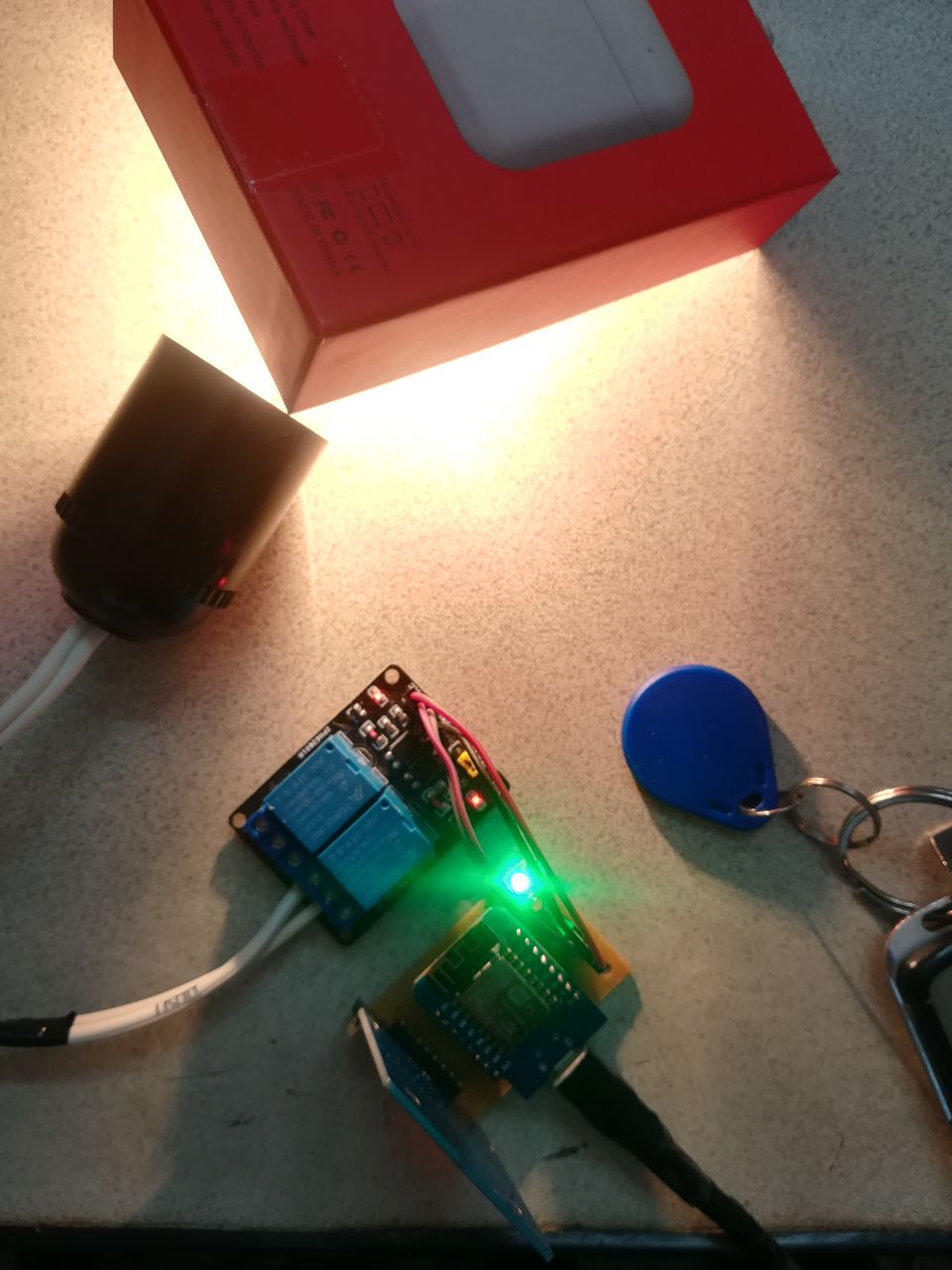
**مشخصات wi-fi**

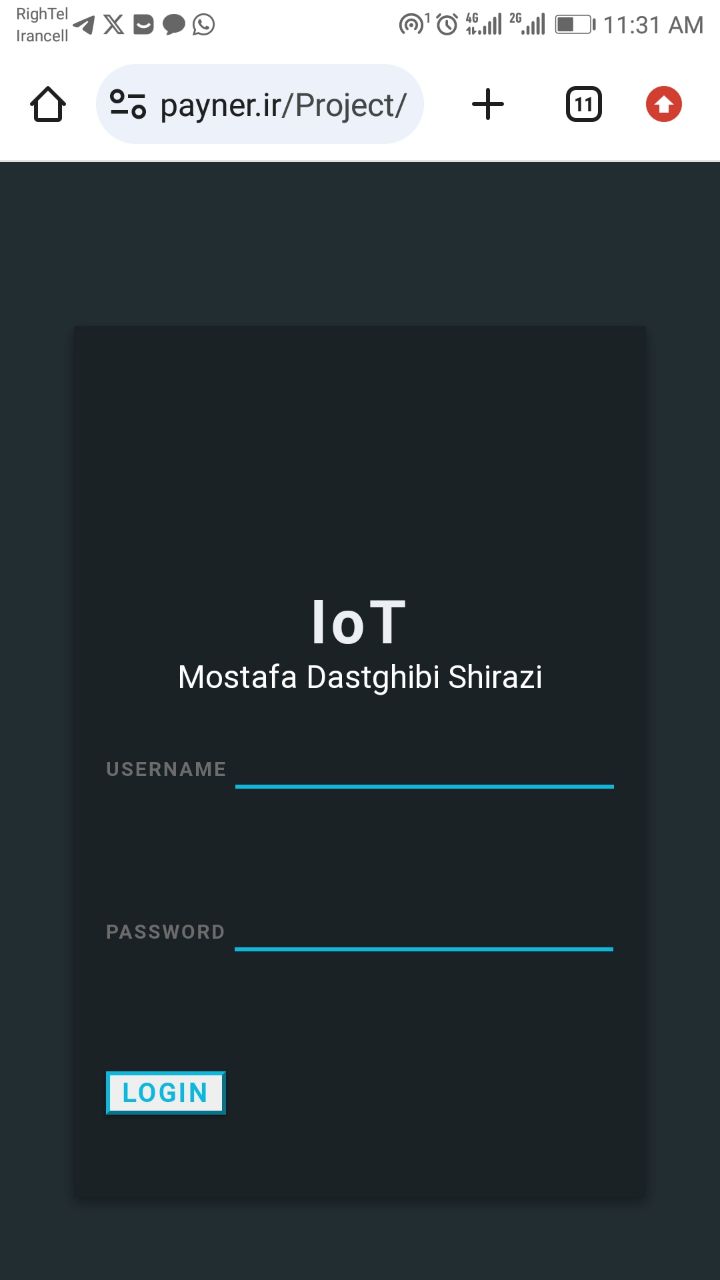
**Ssid:**” NTW”

**Password:**” 0303301306004”

# تصاویر







# کد های esp8266

//وارد کردن کتابخانه ها

#include <EEPROM.h> // کتابخانه حافظه eeprom

#include <ESP8266WiFi.h> // کتابخانه esp8266

#include <MFRC522.h> // کتابخانه ماژول ار اف ای دی

#include <SPI.h>

#include <ESP8266HTTPClient.h> // کتابخانه راه اندازی درخواست http

#include <WiFiClient.h>

// تعریف پایه های رله و ال ای دی و پایه های ماژول ار اف ای دی

#define RST\_PIN D3

#define SS\_PIN D8

#define RELAY\_1\_PIN D1

#define RELAY\_2\_PIN D2

#define LED\_GREEN\_PIN D4

#define LED\_RED\_PIN D0

// رمز و پسورد وای فای مورد نظر

const char\* ssid = "NTW"; // The SSID (name) of the Wi-Fi network you want to connect to

const char\* password = "0303301306004"; // The password of the Wi-Fi network

// کارت خوانده شده

byte presentedCard[4];

boolean state = false;

// آدرس سرور

const char\* serverAddress = "http://www.payner.ir:80/Project/process.php"; //Your Domain name with URL path or IP address with path

unsigned long lastTime = 0;

// Set timer to 5 seconds (5000)

unsigned long timerDelay = 8000;

// مقدار دهی اولیه ماژول ار اف ای دی

MFRC522 rf(SS\_PIN, RST\_PIN);

void setup() {

// راه اندازی سریال در بادریت مورد نظر

Serial.begin(115200);

delay(100);

EEPROM.begin(512);

delay(100);

// تعیین ورودی یا خروجی بودن پین ها

pinMode(RELAY\_1\_PIN, OUTPUT);

pinMode(RELAY\_2\_PIN, OUTPUT);

pinMode(LED\_GREEN\_PIN, OUTPUT);

pinMode(LED\_RED\_PIN, OUTPUT);

// وضعیت اولیه پین

digitalWrite(RELAY\_1\_PIN, HIGH);

digitalWrite(RELAY\_2\_PIN, HIGH);

digitalWrite(LED\_RED\_PIN, LOW);

// راه اندازی esp8266 در مود station

WiFi.mode(WIFI\_STA);

// اتصال به وای فای مورد نظر

WiFi.begin(ssid, password);

// 10 ثانیه چشمک زدن چراغ قرمز به منظور اتصال به وای فای

int count = 0;

while(count < 10){

digitalWrite(LED\_RED\_PIN, HIGH);

delay(500);

digitalWrite(LED\_RED\_PIN, LOW);

delay(500);

count++;

}

// شروع کار ماژول rfid

SPI.begin();

rf.PCD\_Init();

}

void loop() {

// شرط برقراره در صورتی که کارتی به ماژول نزدیک نشه - اگه کارتی تشخیص داده بشه وارد شرط نمیشه

if ( ! rf.PICC\_IsNewCardPresent() || ! rf.PICC\_ReadCardSerial() ) {

if ((millis() - lastTime) > timerDelay) { // اگه از تایمی کی بالا تعریف کردیم گذشته بود وارد شرط میشه

digitalWrite(LED\_GREEN\_PIN, LOW);

if(WiFi.status() == WL\_CONNECTED) { // برسی اتصال به وای فای

digitalWrite(LED\_RED\_PIN, HIGH);

String response = httpGETRequest(serverAddress); // ایحاد درخواست

Serial.print("Server -> ");

Serial.println(serverAddress);

Serial.print("Response -> ");

Serial.println(response);

if(response != "{}") {

if(response == "off"){

state = false;

} else if (response == "on"){

state = true;

}

control();

}

} else {

digitalWrite(LED\_RED\_PIN, LOW);

}

lastTime = millis();

}

digitalWrite(LED\_GREEN\_PIN, HIGH);

delay(50);

return;

}

for (byte i = 0; i < rf.uid.size; i++) { // خواندن کارت

presentedCard[i] = rf.uid.uidByte[i];

}

if (checkID(presentedCard)) { // بررسی وجود کارت در حافظه - اگه بود رله رو کنترل کن

control();

} else { // اگه نبود بیا برو تو حالت افزودن کارت

int count = 0;

while(! rf.PICC\_IsNewCardPresent() || ! rf.PICC\_ReadCardSerial()) { // تا 5 ثانیه اگه کارت قدیمی نزد کارت جدیدو اضافه نکن

digitalWrite(LED\_RED\_PIN, LOW);

digitalWrite(LED\_GREEN\_PIN, LOW);

delay(250);

digitalWrite(LED\_RED\_PIN, HIGH);

digitalWrite(LED\_GREEN\_PIN, HIGH);

delay(250);

if (count >= 10) {

return;

}

count++;

}

byte userCard[4];

for (byte i = 0; i < rf.uid.size; i++) {

userCard[i] = rf.uid.uidByte[i];

}

if(checkID(userCard)){

addNewUsers(presentedCard);

}

}

delay(400);

}

// تابع کنترل رله ها

void control() {

if(!state) {

digitalWrite(RELAY\_1\_PIN, HIGH);

digitalWrite(RELAY\_2\_PIN, HIGH);

state = true;

} else {

digitalWrite(RELAY\_1\_PIN, LOW);

digitalWrite(RELAY\_2\_PIN, LOW);

state = false;

}

}

// تابع برسی آی دی کارت خوانده شده در حافظه

boolean checkID( byte card[] ) {

byte users = EEPROM.read(0);

byte storedCard[4];

if(users <= 0){

addNewUsers(card);

return true;

}

for ( byte user = 0 ; user < users ; user++) {

for ( byte i = 0 ; i < 4 ; i++) {

storedCard[i] = EEPROM.read( user \* 4 + 1 + i );

}

boolean match = true;

for ( byte i = 0; i < 4; i++ ) {

if ( card[i] != storedCard[i]) {

match = false;

break;

}

}

if(match){

return true;

}

}

return false;

}

// تابع افزودن کاربر جدید در حافظه فلش

void addNewUsers(byte newUser[]) {

byte users = EEPROM.read(0);

users += 1;

EEPROM.write(0, users);

for ( byte i = 0 ; i < 4 ; i++ ) {

EEPROM.write( (users - 1) \* 4 + 1 + i , newUser[i] );

}

if (!EEPROM.commit()) {

Serial.println("ERROR! EEPROM commit failed");

}

}

// تابع درخواست http به صورت get

String httpGETRequest(const char\* serverName) {

WiFiClient client;

HTTPClient http;

// تایم اوت

http.setTimeout(3000);

http.begin(client, serverName);

// Send HTTP GET request

int httpResponseCode = http.GET();

String payload = "{}";

if (httpResponseCode == 200) { // اگر کد 200 بود اطلاعات صفحه رو بخون

payload = http.getString();

}

http.end();

return payload;

}