

# Desenvolvimento de Protótipos de Internet das Coisas com foco em Robótica Educacional

Iury Anderson Fernandes Coelho<sup>1</sup>

<sup>1</sup>Instituto Federal de Ciência e Tecnologia da Paraíba -(IFPB)  
Grupo de Pesquisa de Sistemas Embarcados e Inteligência Computacional  
Campina Grande –PB– Brasil.  
[iury.fernandes@academico.ifpb.edu.br](mailto:iury.fernandes@academico.ifpb.edu.br)

## TUTORIAL 01

### MANUAL DE INSTALAÇÃO DA PLACA NODEMCU ESP8266 NA IDE DO ARDUINO

Neste pequeno manual passo-a-passo pretende-se mostrar de forma clara como instalar e configurar os arquivos necessários na IDE do Arduino para que a placa NodeMcu funcione corretamente.

#### Atividade 1

O módulo NodeMcu utiliza linguagem de programação Lua. Entretanto existe várias maneiras de programa-lo. Uma das formas de se programar o NodeMcu na Linguagem C/C++ é utilizando a IDE do Arduino, que por sua vez oferece “caminhos” para que seja possível. A primeira atividade consiste em possuir os componentes de hardwares fundamentais. Você precisa ter os seguintes componentes de hardware:

- 1 x Computador com sistema operacional Windows;
- 1 x NodeMcu com o chip ESP8266;
- 1 x Cabo USB com suporte a transferência de dados.

#### Atividade 2

Depois de reunidos os componentes de hardwares necessários é hora de fazer o download dos arquivos:

1. Primeiro você deve entrar no site: [arduino.cc](http://arduino.cc);
2. Procure por software e downloads, tente isto: <https://www.arduino.cc/en/Main/Software>;
3. Faça o download do instalador do Arduino em formato .zip como mostra a figura 1.

Figura 1: Download da IDE do Arduino no formato .zip

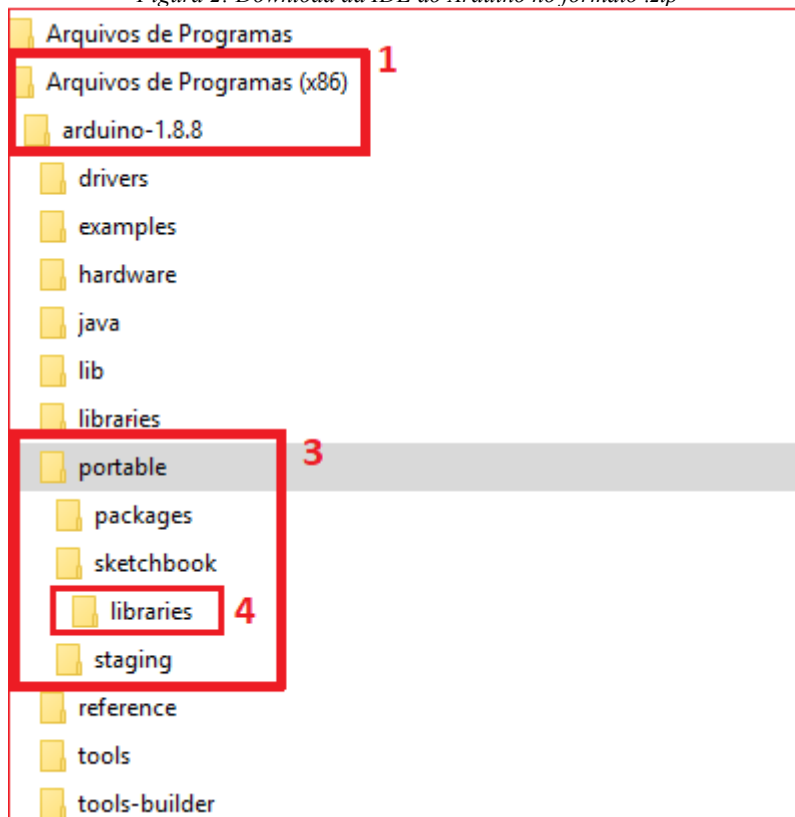


### Atividade 3

Agora é a hora de organizar os diretórios do Arduino e fazer as devidas configurações de arquivos.

1. Descompacte a pasta [arduino-1.8.8-windows.zip](#) no diretório C:\Program Files (x86) do seu computador;
2. Entre no diretório que você descompactou: C:\Program Files (x86)\arduino-1.8.8 e crie a pasta: portable;
3. Entre na pasta portable: C:\Program Files (x86)\arduino-1.8.8\portable e crie as pastas: package, staging e sketchbook.
4. Entre na pasta sketchbook crie a pasta: libraries;
5. A árvore de arquivos deve ficar da forma mostrada na figura2.

Figura 2: Download da IDE do Arduino no formato .zip



### Atividade 4

Depois de ter descompactado a IDE do Arduino e organizado a árvore de arquivos no diretório de Arduino você deve seguir os seguintes passos para a instalação

1. Abra a IDE do Arduino;
2. Clique no menu arquivos/preferências como mostra na figura3;
3. Note que a pasta portable já está como o caminho das preferências da IDE do Arduino C:\Program Files (x86)\arduino-1.8.8\portable\preferences.txt;

4. Na aba preferencias preencha o campo URLs Adicionais para Gerenciamento de Placas com o seguinte endereço: [http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json) de acordo com a figura 4;

Figura 3: menu de preferencias na IDE do Arduino

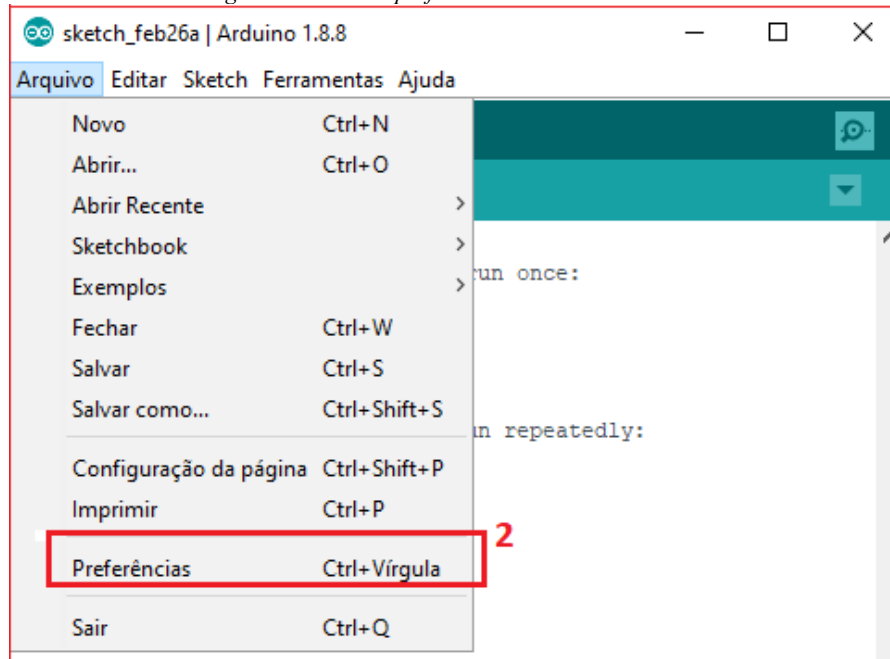
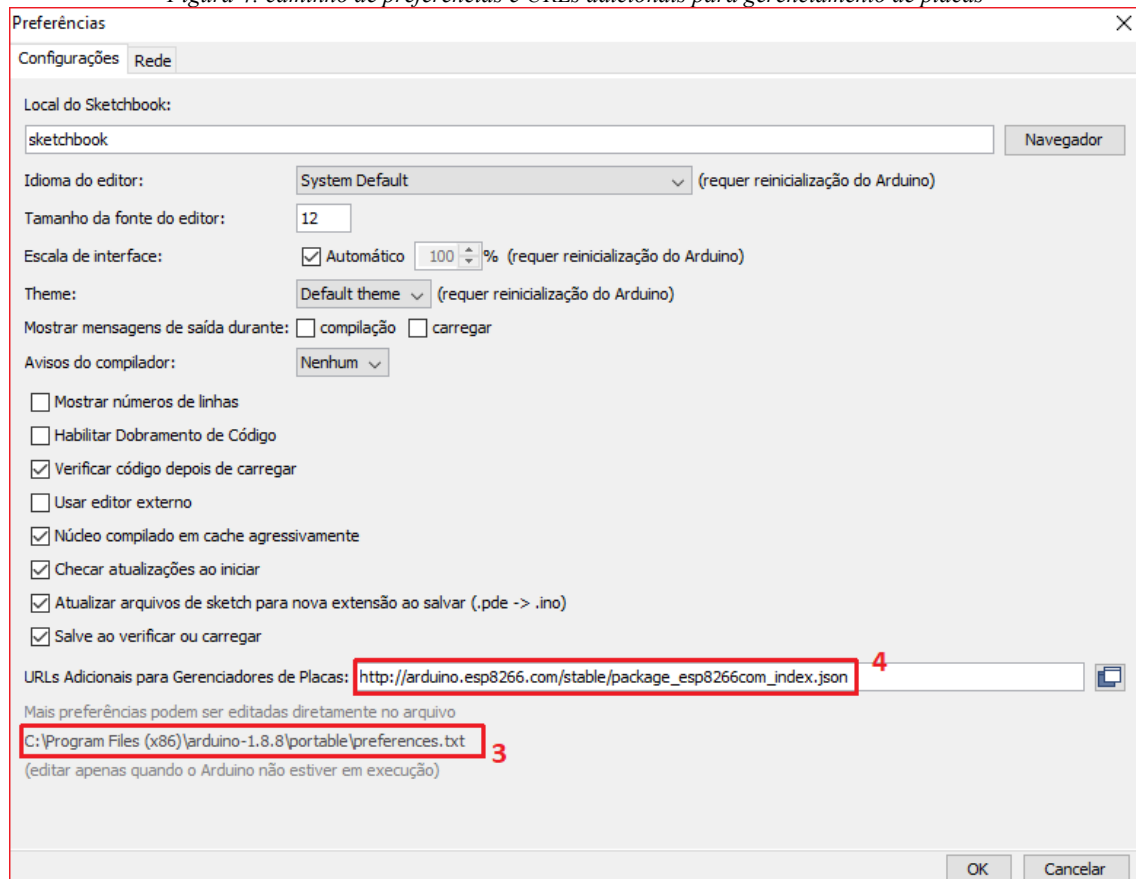


Figura 4: caminho de preferências e URLs adicionais para gerenciamento de placas



## Atividade 5

Tendo configurado a url de gerenciamento de placas corretamente você deve instalar as placas que contém o módulo ESP8266 na IDE do Arduino pela comunidade ESP8266.

1. Abra o menu ferramentas/ placas/ gerenciamento de placas como mostra a figura 5;
2. Na aba Gerenciador de Placas espere o carregamento dos índices e plataformas e em seguida faça uma busca por: esp8266;
3. Sua pesquisa deve achar o gerenciador de placas: esp8266 by ESP8266 Community de acordo com a figura 6;
4. Clique em instalar e aguarde a instalação ser concluída;

Figura 5: pesquisa e instalação do esp8266 by ESP8266 Community

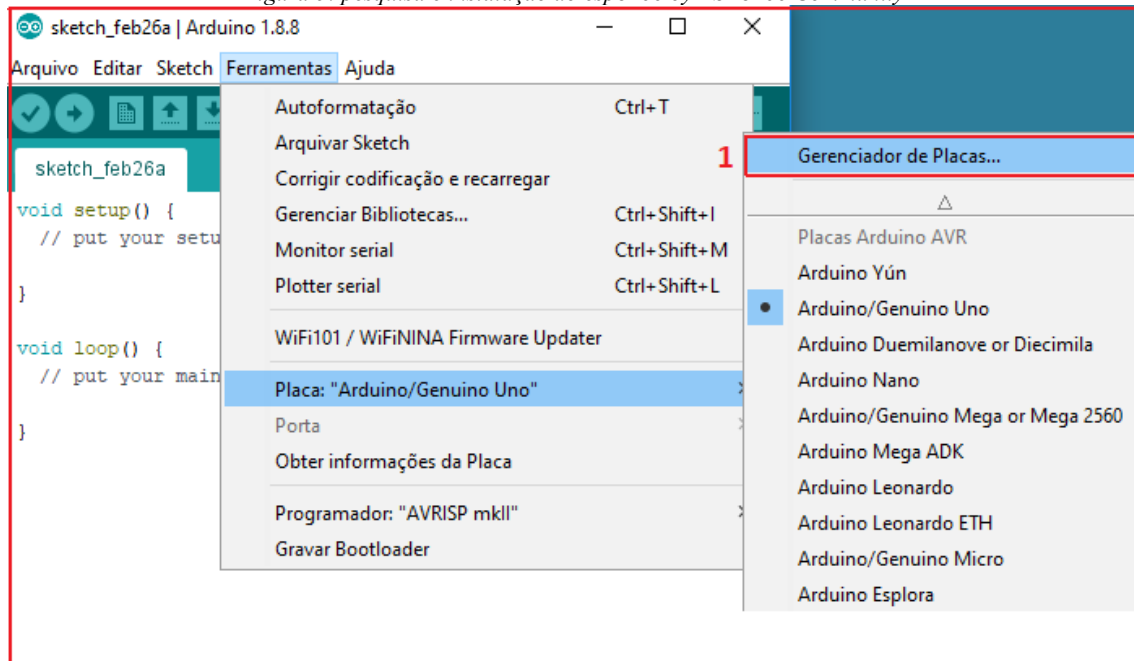
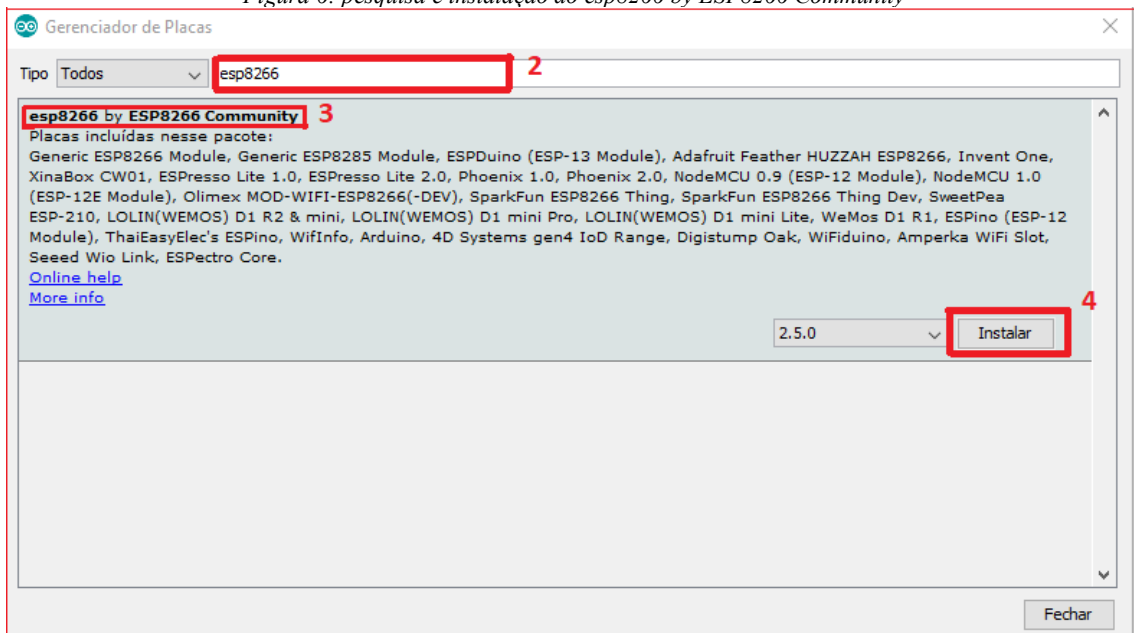


Figura 6: pesquisa e instalação do esp8266 by ESP8266 Community



## Atividade 6

Agora você precisa selecionar o módulo NodeMcu ESP8266 na IDE do Arduino e fazer as devidas configurações

1. Abra o menu ferramentas/placa, procure e selecione a placa: NodeMCU1.0(ESP-12E Module);
2. Após selecionar o modulo correto irá aparecer alguns ajustes no menu ferramentas
3. Espete o módulo NodeMcu na porta USB e identifique em qual porta serial ele está ligado.
4. Você pode verificar isto usando o gerenciador de dispositivo do Windows.
  - 4.1 Abra o menu pesquisar no Windows pesquise por gerenciador de dispositivos, abra o gerenciador clique em Portas (COM e LPT). Feito isso devemos visualizar a porta USB-SERIAL ao qual o node está ligado. A porta serial está escrita na forma COM (número da entrada USB). Exemplo COM3 como segue na figura 7
5. Na IDE no Arduino você deve configurar o módulo node de acordo com a figura8
  - 5.1 Selecione a porta serial ao qual o node está ligado. Neste caso COM3
  - 5.2 Em Flash Size marque o campo: 4M (3M SPIFFS)
  - 5.3 Em Upload Speed marque o campo: 115200

Figura 7: identificação da porta serial USB

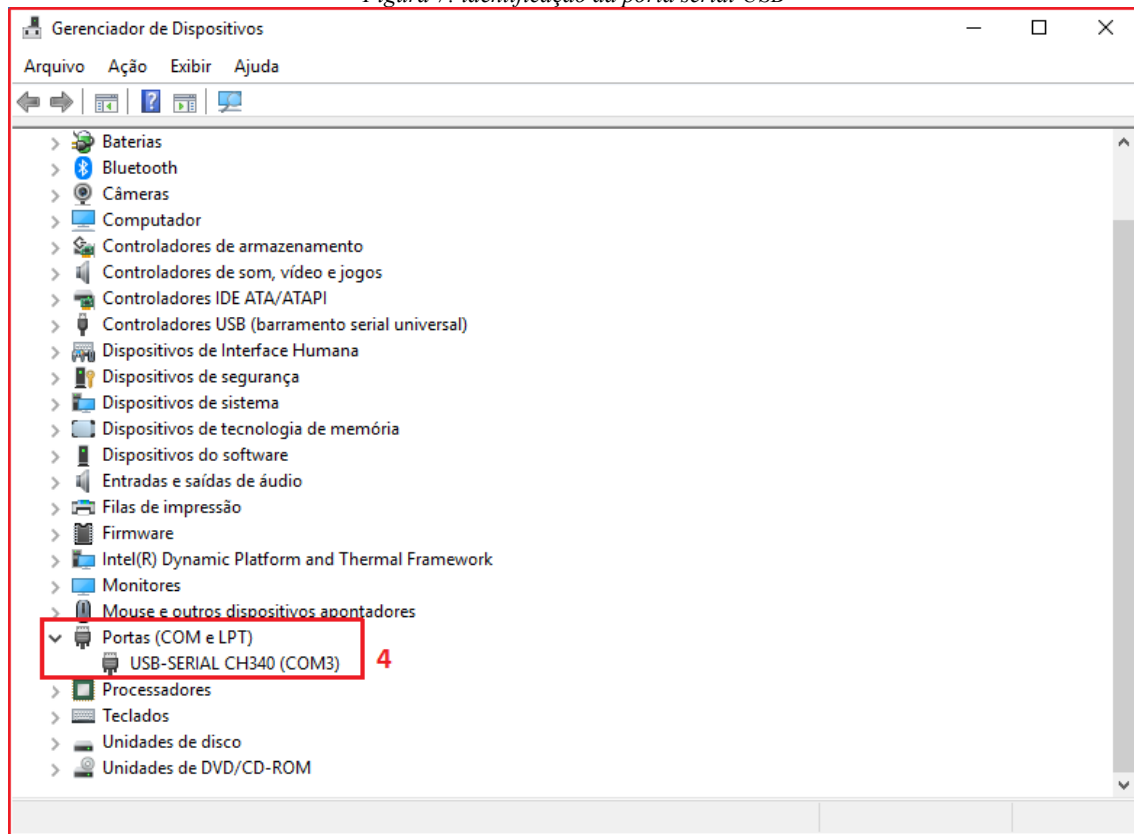
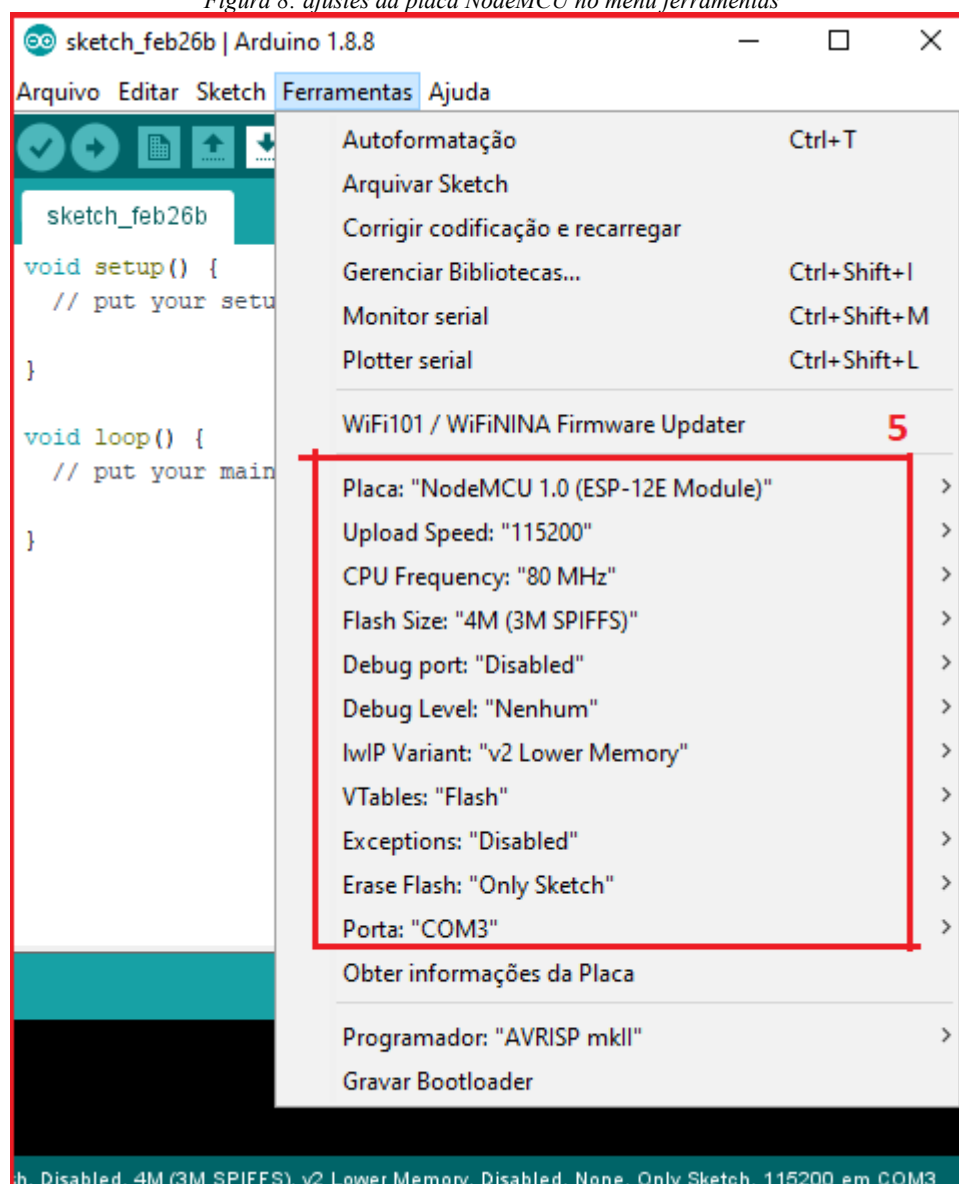


Figura 8: ajustes da placa NodeMCU no menu ferramentas



## Atividade 7

Depois de concluir todas as atividades anteriores com êxito, finalmente você pode carregar um algoritmo de teste para confirmar a correta instalação da placa NodeMcu na IDE do Arduino.

1. Resete a IDE do Arduino;
2. Copie e cole o na sua IDE o algoritmo da tabela 1 e em seguida carregue o código no seu módulo; NodeMcu. O passo 2 deve estar de acordo com a figura 9;
3. Depois de carregado o código você deve observar um led azul piscando no seu módulo NodeMcu;
4. Feito os passos anteriores com sucesso, a IDE do Arduino estará pronta e testada para programação do de protótipos de internet das coisas com o NodeMcu;

Tabela 1: Algoritmo Piscar Led do NodeMCU ESP8266 1.0

```

/*
  ESP8266 piscar led por Iury Fernandes
  O algoritmo a seguir é responsável por piscar o led do Modulo NodeMcu ESP8288 1.0
*/
const int led = 2; // declaração da constante contendo a numeração do pino ao qual estão nodeMCU

void setup() {
  pinMode(led, OUTPUT);    // Inicializa a saída no pino led = 2
}

void loop() {
  digitalWrite(led, LOW);  // Desliga a saída do led
  delay(500);              // Espera por meio segundo
  digitalWrite(led, HIGH); // Liga a saída do led
  delay(1000);             // Espera um segundo
}

```

Figura 9: carregamento do algoritmo piscarled.ino no módulo NodeMcu

