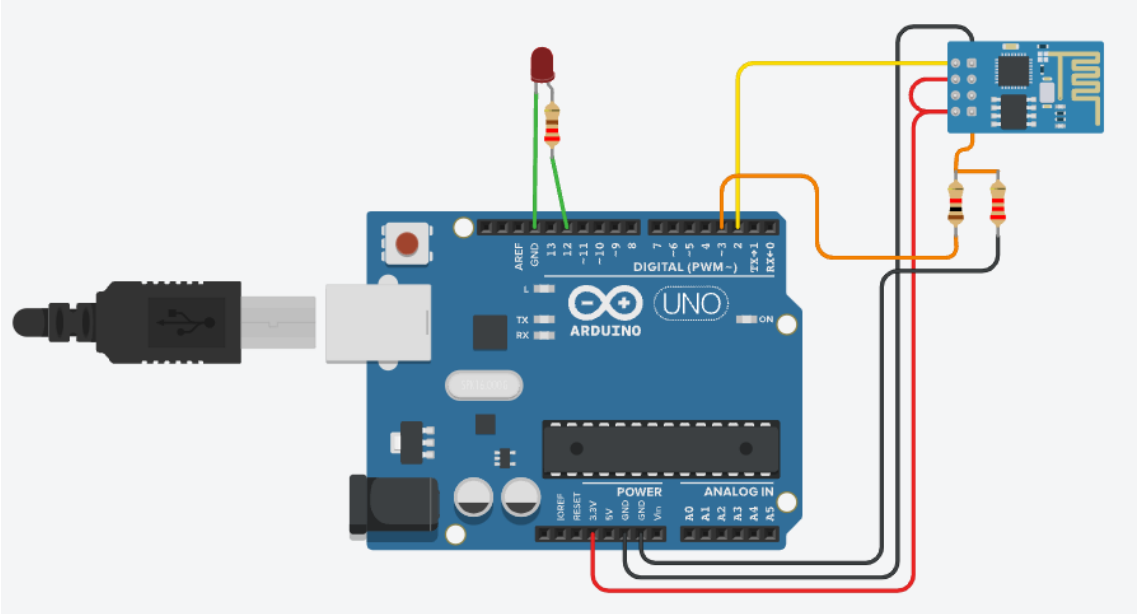
PROJETO EM ARDUINO PARA AV1 – UTILIZANDO ARDUINO UNO R3, ESP8266 e LED

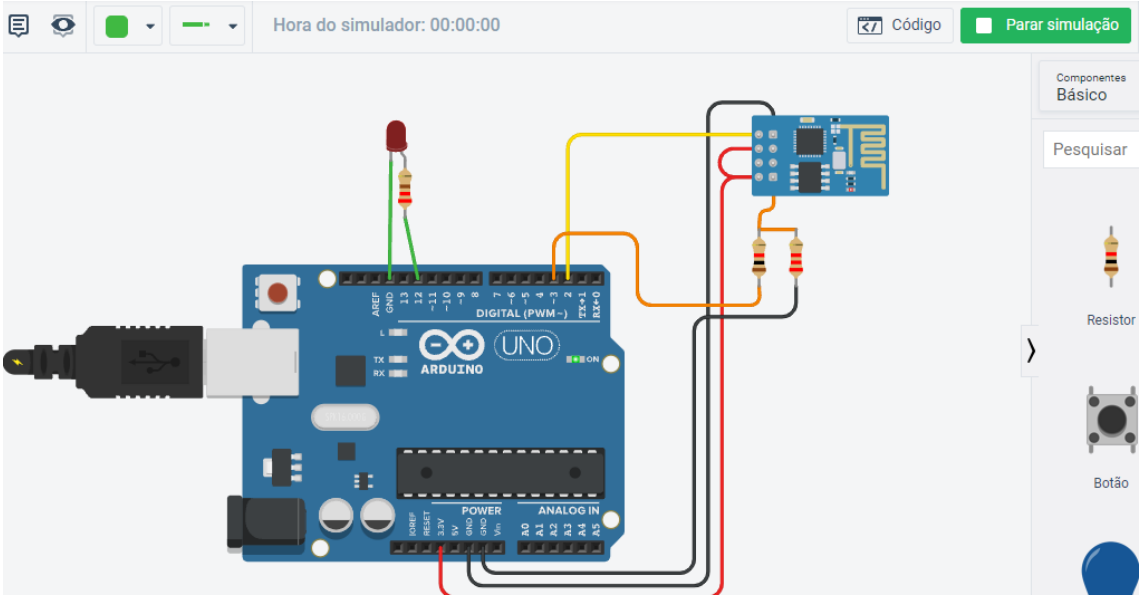
ALUNO: Vitor Sousa Mesquita

MATRÍCULA: 201802163204

Imagem do esquema:

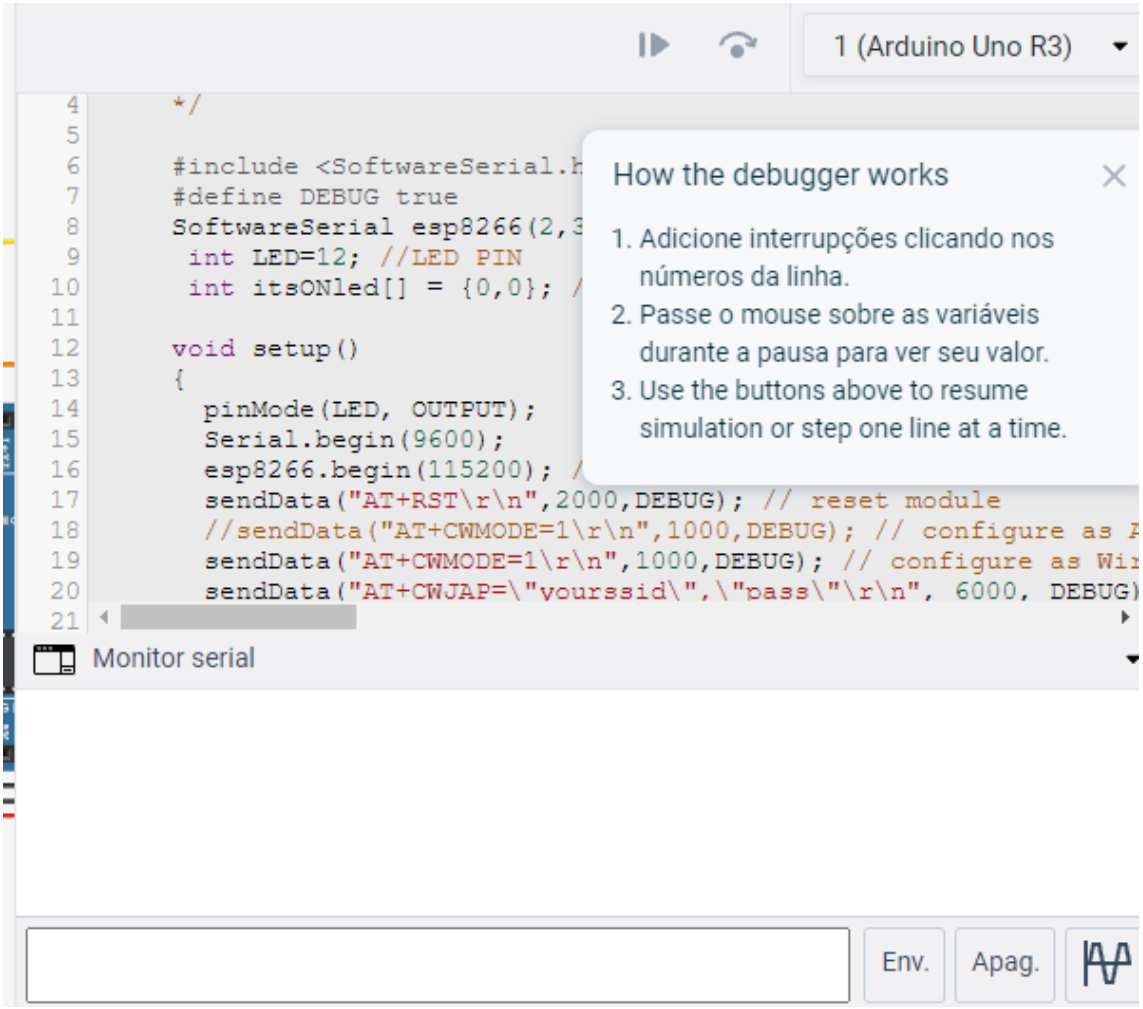


Funcionando:



PS: não funciona totalmente devido ao erro do Tinkercad que bloqueia comunicação com a internet, o que é vital para que um projeto baseado em WIFI funcione corretamente, o único

feedback que temos é a luz do Arduino indicando que está ligado e ausência de mensagem de erro



Código:

```
    #include <SoftwareSerial.h>

    #define DEBUG true

    SoftwareSerial esp8266(2,3); // make RX Arduino line is pin 5, make TX
Arduino line is pin 6.

     int LED=12; //LED PIN

     int itsONled[] = {0,0}; // LED STATUS ARRAY eg- ON or OFF at startup.


    void setup()

    {

      pinMode(LED, OUTPUT);

      Serial.begin(9600);
```

```
      esp8266.begin(115200); // your esp's baud rate might be different

      sendData("AT+RST\r\n",2000,DEBUG); // reset module

      //sendData("AT+CWMODE=1\r\n",1000,DEBUG); // configure as Access point
mode

      sendData("AT+CWMODE=1\r\n",1000,DEBUG); // configure as Wireless Station
mode

      sendData("AT+CWJAP=\"yourssid\",\"pass\"\r\n", 6000, DEBUG); //Put Your
SSID and password if activate as Station mode else comment down the line

      sendData("AT+CIFSR\r\n",2000,DEBUG); // get ip address

      sendData("AT+CIPMUX=1\r\n",1000,DEBUG); // configure for multiple
connections

      sendData("AT+CIPSERVER=1,80\r\n",1000,DEBUG); // turn on server on port 80

    }


void loop()

{


  if(esp8266.available()) // check if the esp is sending a message

  {

    if(esp8266.find("+IPD,"))

    {

     // subtract 48 because the read() function returns

     // the ASCII decimal value and 0 (the first decimal number) starts at 48

     int connectionId = esp8266.read()-48;

     //To read the url sent by the client

     String msg;

     esp8266.find("?");

     delay(100);

     msg = esp8266.readStringUntil(' ');

     String command1 = msg.substring(0);

     // HTML START

     String webpage = "<html><head><title>ESP8266 WEB SWITCH</title>";

     webpage += "<meta name=\"viewport\" content=\"width=device-width, initial-
scale=1.0\"><style>.button {background-color: orange;border: none;color:
white;padding: 15px 32px;text-align: center;display: inline-block;font-size:
16px;} .centre {text-align: center;}</style>";

     webpage += "</head><body class=\"centre\"><h1 class=\"centre\">ESP8266 WEB
SWITCH</h1>";

     //COMMANDS TO TURN ON or OFF LED RECEIVE BY WEB

                  if (command1 == "T"){
```

```
                    if (itsONled[1] == 1)
                    {
                      digitalWrite(LED, LOW);
                      itsONled[1] = 0;
                      webpage += "<p>LED STATUS OFF</p>";
                    }
                    else
                    {
                      digitalWrite(LED, HIGH);
                      itsONled[1] = 1;
                      webpage += "<p>LED STATUS ON</p>";
                    }
                }
     webpage += "<a class=\"button\" href=\"?T\">TAP</a></body></html>";
     String cipSend = "AT+CIPSEND=";
     cipSend += connectionId;
     cipSend += ",";
     cipSend +=webpage.length();
     cipSend +="\r\n";
     sendData(cipSend,500,DEBUG);
     sendData(webpage,500,DEBUG);
     //BELOW THIS LINE CLOSE THE CONNECTION
     String closeCommand = "AT+CIPCLOSE=";
     closeCommand+=connectionId; // append connection id
     closeCommand+="\r\n";
     sendData(closeCommand,500,DEBUG);
    }
  }

}

//PROGRAM TO SEND COMMAND TO ESP8266
void sendData(String command, const int timeout, boolean debug)
{
    esp8266.print(command); // send the read character to the esp8266
    long int time = millis();
```

```
    while( (time+timeout) > millis())
    {
      while(esp8266.available())
      {
        // The esp has data so display its output to the serial window
        Serial.write(esp8266.read());
      }
    }
}
```