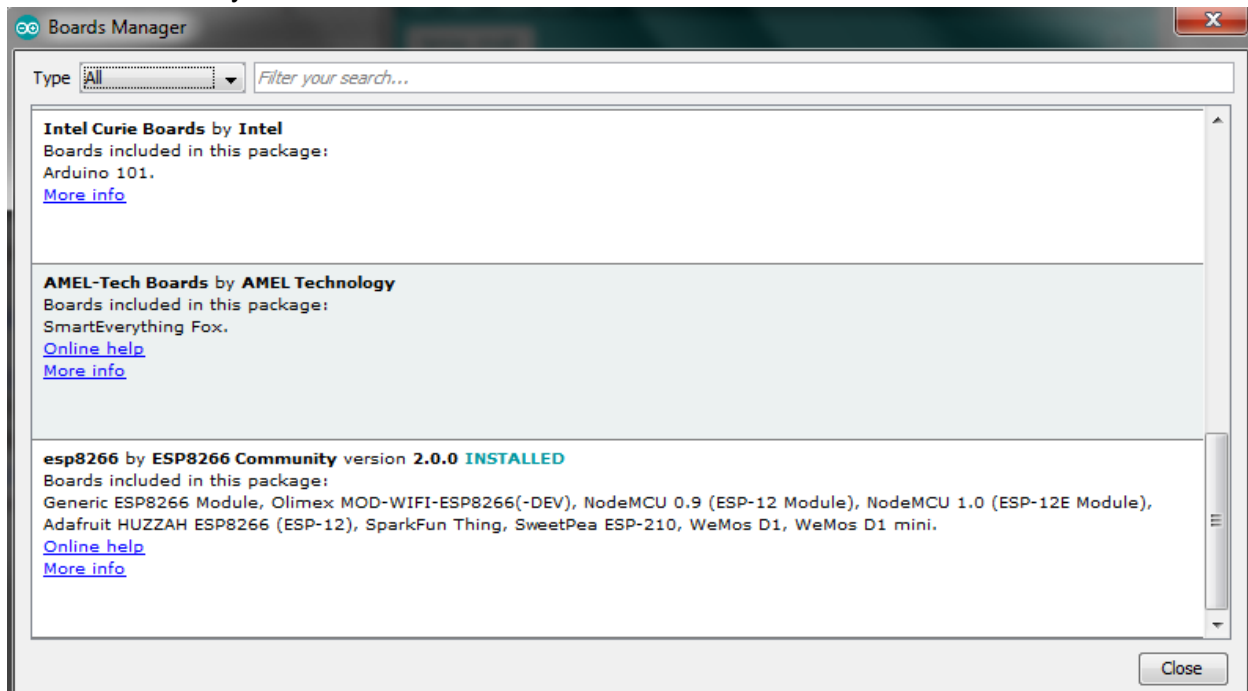


# Nodemcu Amica Wifi setup for Windows

If the steps get confusing, there are alternate resources that can teach you how to set it up such as <http://www.seeedstudio.com/recipe/1107-getting-started-with-nodemcu-devkit-in-arduino-ide.html>

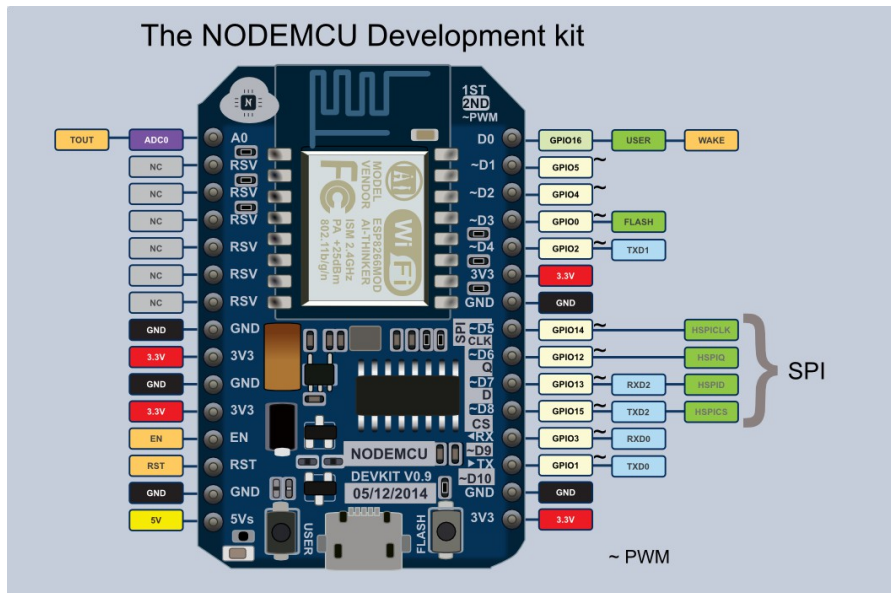
- 1) Install Arduino IDE from <https://www.arduino.cc/en/Main/Software>
- 2) Open up IDE and go to Files>Preferences
- 3) Go to Additional Board Manager URLs and paste in the UR:  
[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json) and click OK
- 4) Go to Tools>Board>Boards Manager and scroll down to esp8266 by esp8266 community and install it.



After you've completed all these steps, your IDE should be able to program the Nodemcu Amica board.

## Connecting Nodemcu to Wifi and turn on LED using

# Browser



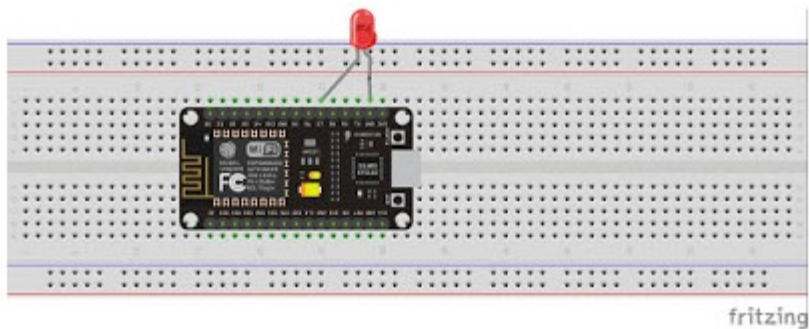
This is the original diagram, however, Zephyr found some errors and here is the correct layout.

J1			J2		
ADC EX	A0	1	PIN1	1	D0 GPIO16
ADC		2	PIN2	2	D1 GPIO5
RESV		3	PIN3	3	D2 GPIO4
SD D3	GPIO10	D12	PIN4	4	D3 GPIO0
SD D2	GPIO9	D11	PIN5	5	D4 GPIO2 TXD1
SD D1	<del>SPI INT</del> MOSI	6	PIN6	6	VDD3V3
SD CMD	<del>SPI MOSI</del> CS	7	PIN7	7	GND
SD D0	<del>SPI MISO</del> MISO	8	PIN8	8	D5 GPIO14 HSPICLK
SD CLK	<del>SPI CLK</del> CLK	9	PIN9	9	D6 GPIO12 HSPICQ
GND		10	PIN10	10	D7 GPIO13 RXD2 HSPID
VDD3V3		11	PIN11	11	D8 GPIO15 TXD2 HSPICS
EN		12	PIN12	12	D9 RXD0 GPIO3
nRST		13	PIN13	13	D10 TXD0 GPIO1
GND		14	PIN14	14	GND
VDD5V		15	PIN15	15	VDD3V3

THT\_Male\_P\_1x15 THT\_Male\_P\_1x15

## The set-up

- 1) Place your Nodemcu unit onto a breadboard
- 2) Connect a resistor to d7 (13 in IDE), a LED in series with that resistor, and connect the other end of the LED to ground as shown in the diagram below.



(the diagram doesn't show the resistor in series but you should use a resistor so the led doesn't burnout)

3) Plug the nodemcu into your computer and you're ready for the coding part.

### Coding

- 1) Open Arduino IDE and go to Tools>Boards and make sure that the board selected is "NodeMCU 1.0 (ESP-12E Module)"
- 2) Copy and paste the code below (taken from the Arduino Robotics division)

---

```
#include <ESP8266WiFi.h>

const char* ssid = "go";
const char* password = "goGogoG0";

int ledPin = 13; // GPIO13
WiFiServer server(80);

void setup() {
  Serial.begin(9600);
  delay(100);

  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, LOW);

  // Connect to WiFi network
  Serial.println();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  WiFi.begin(ssid, password);

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

  // Start the server
  server.begin();
  Serial.println("Server started");
```

```

// Print the IP address
Serial.print("Use this URL to connect: ");
Serial.print("http://");
Serial.print(WiFi.localIP());
Serial.println("");
}

void loop() {
  // Check if a client has connected
  WiFiClient client = server.available();
  if (!client) {
    return;
  }

  // Wait until the client sends some data
  Serial.println("new client");
  while(!client.available()){
    delay(1);
  }

  // Read the first line of the request
  String request = client.readStringUntil('\r');
  Serial.println(request);
  client.flush();

  // Match the request

  int value = LOW;
  if (request.indexOf("/LED=ON") != -1) {
    digitalWrite(ledPin, HIGH);
    value = HIGH;
  }
  if (request.indexOf("/LED=OFF") != -1) {
    digitalWrite(ledPin, LOW);
    value = LOW;
  }

  // Set ledPin according to the request
  //digitalWrite(ledPin, value);

  // Return the response
  client.println("HTTP/1.1 200 OK");
  client.println("Content-Type: text/html");
  client.println(""); // do not forget this one
  client.println("<!DOCTYPE HTML>");
  client.println("<html>");

  client.print("Led pin is now: ");

  if(value == HIGH) {
    client.print("On");
  } else {
    client.print("Off");
  }
  client.println("<br><br>");
  client.println("<a href='\"/LED=ON\"'><button>Turn On </button></a>");
  client.println("<a href='\"/LED=OFF\"'><button>Turn Off </button></a><br />");
  client.println("</html>");
}

```

```
delay(1);  
Serial.println("Client disconnected");  
Serial.println("");  
}
```

---

3) Replace ssid and password with your own ssid and password

Note: Since the UCI network makes it impossible for devices to communicate with each other, you must use hotspot from your phone in order to connect the Nodemcu to wifi.

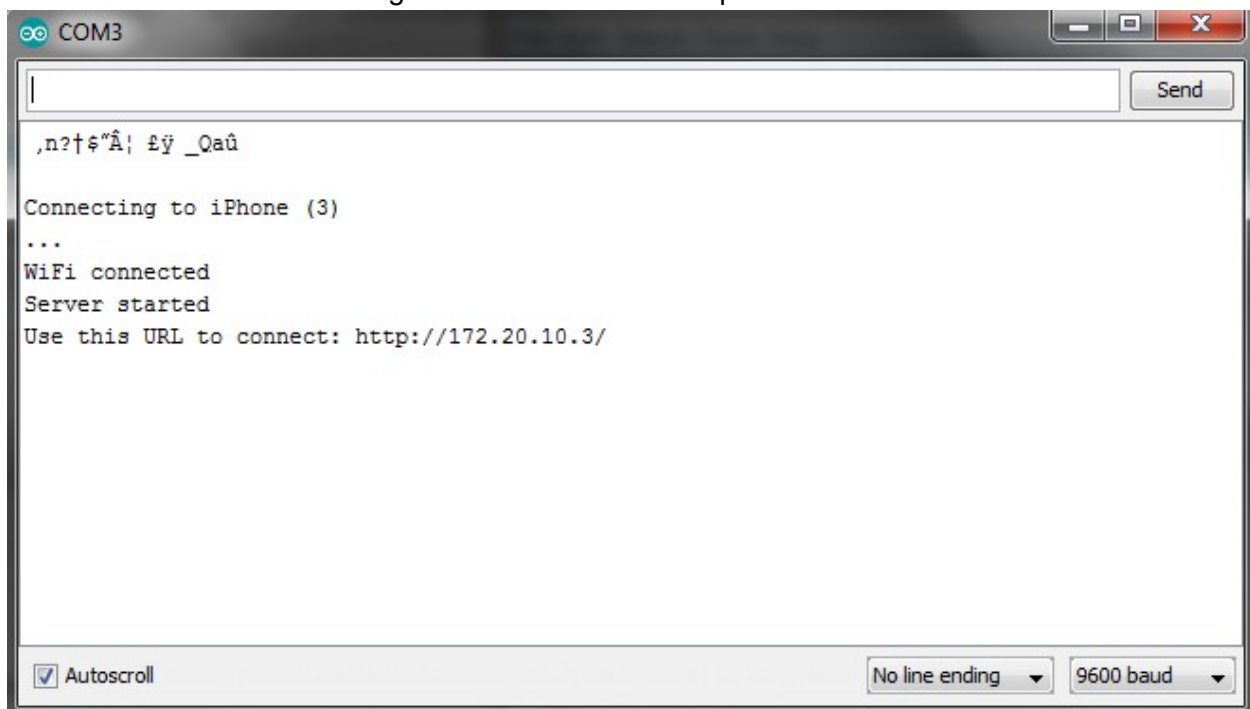
4) Upload your code using the arrow going to the right



5) Wait until your code has completely finished upload and go to Tools>Serial Monitor

(Make sure baud rate at the bottom right is set to 9600)

6) Wait until the Nodemcu has connected to your wifi and it will send you a URL. Your serial monitor should look something like this after it has completed



7) Connect your laptop to the Nodemcu (it is now acting as its own access point)

8) Paste the given url into your browser and you should be able to turn the LED on and off using the buttons on the browser.