

## **CopperCloud IoT Foundation Workshop (IFW)**

### **Software Set-up Instructions for Participant's PC**

This document describes the process to setup the software required on Participant's computer for the IoT Foundation Workshop.

These instructions will also be covered in the pre-workshop meeting with all Participants.

For help with any of these instructions, please send an email to [iot@coppercloud.in](mailto:iot@coppercloud.in).

The following software need to be installed on the Participant's computer:

1. Arduino IDE with support for NodeMCU board
2. MQTTBox (Messaging Client)
3. Node Red – Visual Programming tool for IoT

## **1. Installation of the Arduino Software (IDE)**

### **1.1 Procedure for Installation of Arduino IDE SETUP for ESP 8266:**

- a. Download Arduino IDE: <https://www.arduino.cc/en/Main/Software>
- b. Open you IDE and click on "File -> Preferences".
- c. In "Additional Boards Manager URLs" add this line:  
[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)  
and click on "OK":
- d. Go to "Tools -> Board -> Boards Manager", type "ESP8266" and install it.
- e. Go again to "Tools -> Board" and select "Generic ESP8266 Module".

### **1.2 Procedure for Installation of USB drivers (only if required. Please try uploading the blink program [step1.3 below] before installing new drivers. If it works, then additional drivers are not needed):**

*[This section of the document has been deprecated because newer versions of the Arduino IDE do not require additional drivers. However, in case attendees face problems uploading code to the NodeMCU, do contact us and CopperCloud team will help resolve the issues.]*

### 1.3 Procedure for NodeMCU programming:

- Connect Your NodeMCU to the Computer
- Go to “Tools -> Board” in a menu section entitled ESP8266 Boards. Select the appropriate board configuration from the list for your device (NodeMCU 1.0 (ESP-12E module)).
- You may need to manually select communication port in “Tools -> Port “if Arduino IDE can’t automatically detect it.
- Go to “File->Examples-> Basic-> Blink and click on it. Blink program is open.
- Click upload button on button bar and wait for downloading.

Ref: <https://mechatronicsblog.com/nodemcu-programming-first-steps/>

## 2. Installation of MQTT BOX:

**2.1** Installing MQTTBox on windows is straightforward. Download from here and double click on .exe file to install: <http://workswithweb.com/mqttbox.html>

**2.2** MQTT client setting: When creating new MQTT client from MQTTBox app, there are wide range of connection settings you can specify.

- MQTT Client Name:** Name to identify MQTT client and display on dashboard. It can be any string value. e.g: client\_test\_1
- Client ID:** The client identifier is an identifier of each MQTT client connecting to a MQTT broker.
- Protocol:** Network protocol used by MQTT client to connect with MQTT broker.e.g. **mqtt/tcp**.
- Host:** MQTT host to connect. For this workshop, the host:port will be: **3.214.158.175:1883**
- Will:** QoS- 0 –at most once

The screenshot shows the MQTTBox application window with the 'MQTT CLIENT SETTINGS' tab selected. The form contains the following fields and settings:

- MQTT Client Name:** AWSMQTT (highlighted with a red box)
- MQTT Client ID:** d5fd5bc8-e659-4aaa-ab2a-f991e6fb
- Protocol:** mqtt / tcp (highlighted with a red box)
- Host:** 3.214.158.175:1883 (highlighted with a red box)
- Username:** (empty)
- Password:** (empty)
- Reconnect Period (milliseconds):** 1000
- Connect Timeout (milliseconds):** 30000
- KeepAlive (seconds):** 10
- Will - Topic:** Will - Topic
- Will - QoS:** 0 - Almost Once
- Will - Retain:** No
- Will - Payload:** (empty)
- Append timestamp to MQTT client id?** Yes
- Clean Session?** Yes
- Reschedule Pings?** Yes
- Broker is MQTT v3.1.1 compliant?** Yes
- Auto connect on app launch?** Yes
- Queue outgoing QoS zero messages?** Yes

Buttons for 'Save' and 'Delete' are visible at the bottom of the form.

**\*Only fill data in the fields marked by red boxes above**

### 3. Installation of NODE Red:

To install Node-RED locally you will need a [supported version of Node.js](#).

- a. **Install Node.js** Download the latest 12.x LTS version of Node.js from the official [Node.js home page](#)
- b. Once installed, open a command prompt and run the following command to ensure Node.js and npm are installed correctly. Using cmd: `node --version && npm --version`. You should receive back output that looks similar to:

**V12.15.0**

**6.14.5**

- c. **Install Node Red:** Installing Node-RED as a global module adds the command `node-red` to your system path. Execute the following at the command prompt:

**Npm install -g --unsafe-perm node-red**

- d. **Run Node-RED:** Once installed, you are ready to [run Node-RED](#). the simple way to run Node-RED is to use the `node-red` command in a command prompt

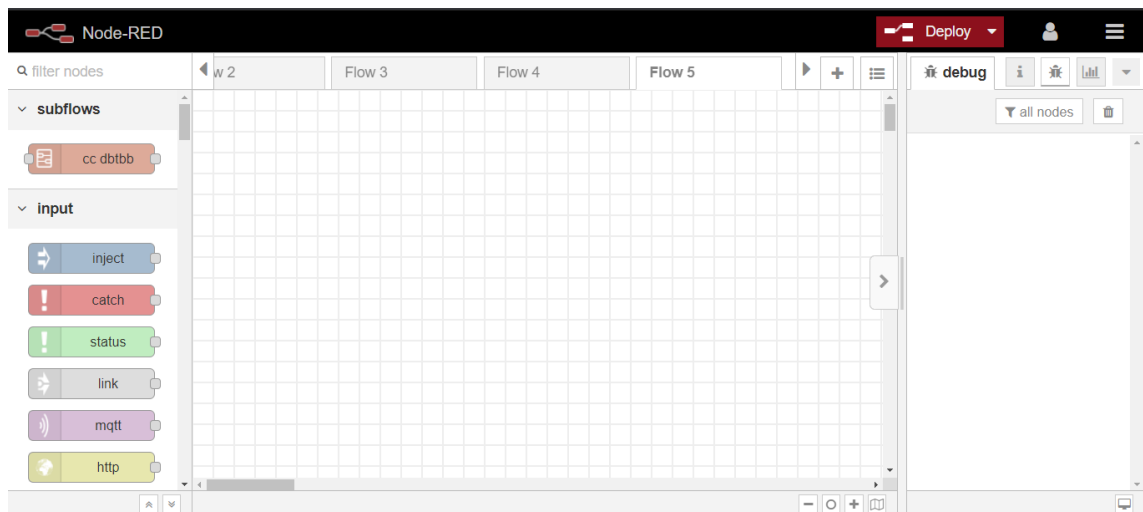
**C:>node-red**

- e. This will output the Node-RED log to the terminal. You must keep the terminal open in order to keep Node-RED running.

- f. **Open Node RED:**

➔ Go to the browser and visit this url: **http://localhost:1880** (or, **http://127.0.0.1:1880**)

➔ If successfully installed, you should see a page like this:



Further reading (these items will be covered in the workshop as well):

<https://nodered.org/docs/getting-started/windows>

<https://nodered.org/docs/getting-started/local>

---- End of installation of local tools ----