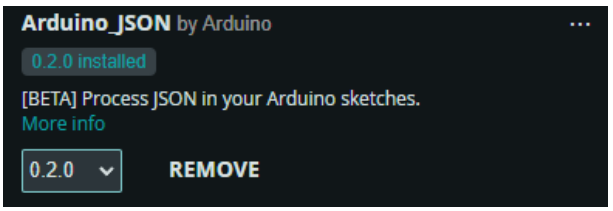
[1] Arduino Sketch Configuration

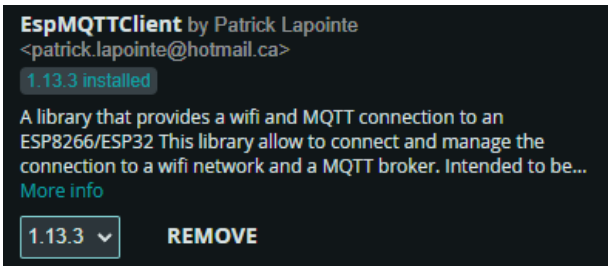
Disclaimer: This project was developed using Arduino IDE 2.3.3 version

Step 1: Install Required Libraries

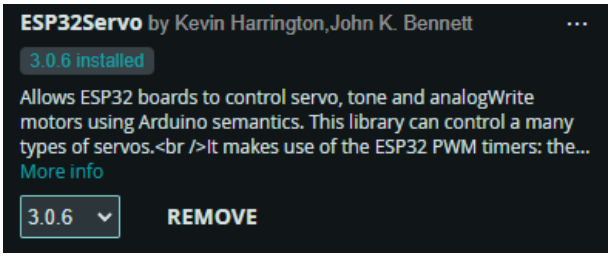
1. Open the Arduino IDE.
2. Navigate to **Sketch > Include Library > Manage Libraries**.
3. Use the library manager to search for and install the following libraries:
   - **Arduino_JSON** by Arduino



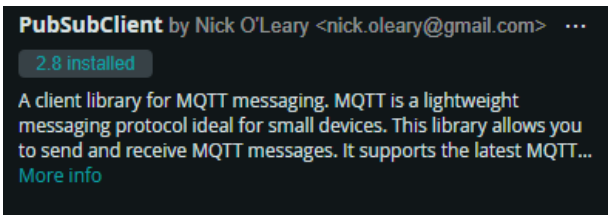
   - **EspMQTTClient** by Patrick Lapointe



   - **ESP32Servo** by Kevin Harrington



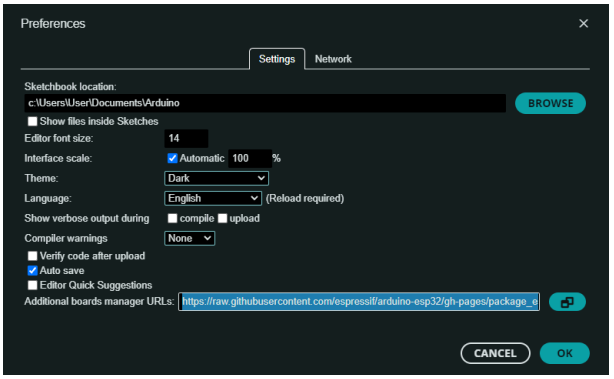
   - **PubSubClient** by Nick O'Leary



Step 2: NodeMCU-32S Board Configuration

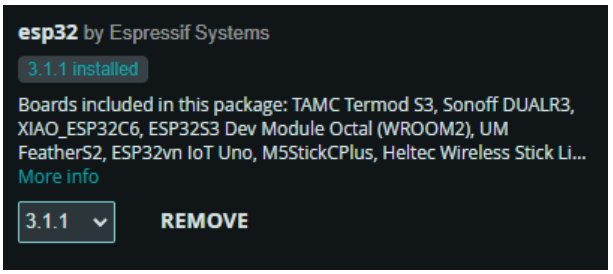
1. Navigate to **File > Preferences.**
2. In the Additional Boards Manager URLs textbox, paste the following URL:

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json



Step 3: Install ESP32 Board Manager

1. Navigate to **Tools > Board > Boards Manager**.
2. Use the search bar to search for esp32 and install:
   - **ESP32 boards manager** by Espressif Systems.



Step 4: Configure Board and Port

1. To choose the board, navigate to **Tools > Board > esp32 > NodeMCU-32S**
2. To choose the COM port, **Tools > Port > Your COM Port** (e.g., COM5)
3. Connect ESP32 board to the computer.

Step 5: Upload Code

1. Open the **WakeEase.ino** file in Arduino IDE.
2. Update the following values in the code:

Important Notes: The ESP32 only supports the 2.4GHz Wi-Fi band (not compatible with 5GHz). If using a dual-band hotspot (2.4GHz and 5GHz), configure it to either 2.4GHz only or Mixed mode (both 2.4GHz and 5GHz enabled).

- **WIFI_SSID**: Your Wi-Fi network name.
- **WIFI_PASSWORD**: Your Wi-Fi network password.
- **MQTT_SERVER**: Enter the external IP address of your VM instance, which can be found on the Google Cloud Platform. Ensure that the VM instance has been properly set up beforehand.

```
#include <PubSubClient.h>
#include <WiFi.h>
#include <time.h>
#include <ESP32Servo.h>

// Wi-Fi Credentials
const char* ssid = "yourwifi";  // Replace with your Wi-Fi SSID
const char* password = "yourwifipassword";    // Replace with your Wi-Fi password

// MQTT Credentials
const char* MQTT_SERVER = "yourExternalIP"; // Replace with your VM External IP
const int MQTT_PORT = 1883;               // Non-TLS communication port
WiFiClient espClient;
PubSubClient client(espClient);
```

3. Click the **Upload** button in Arduino IDE.
4. Once the upload is complete, the code will execute automatically.

You should see the output shown in the figure below if the setup and connection are configured correctly without any issues.