รายงาน

เรื่อง Web Embedded สำหรับ Setup/Configuration

จัดทำโดย

นาย ชัยภัทร เนื่องจำนงค์ รหัส 61033803

เสนอ

ดร. ผิน ฉัตรแก้วมณี

คณะ เทคโนโลยีสารสนเทศ

สาขา วิศวกรรมคอมพิวเตอร์

วิชา CPE 405

มหาวิทยาลัยศรีปทุม

คู่มือติดตั้ง Web Embedded

- มีการเสียบจ่ายไฟให้กับอุปกรณ์
- ได้มีการที่เราได้เชื่อมต่อกับอุปกรณ์ผ่าน Wi-Fi ใน Smartphone หรือ Tablet ที่มี Internet

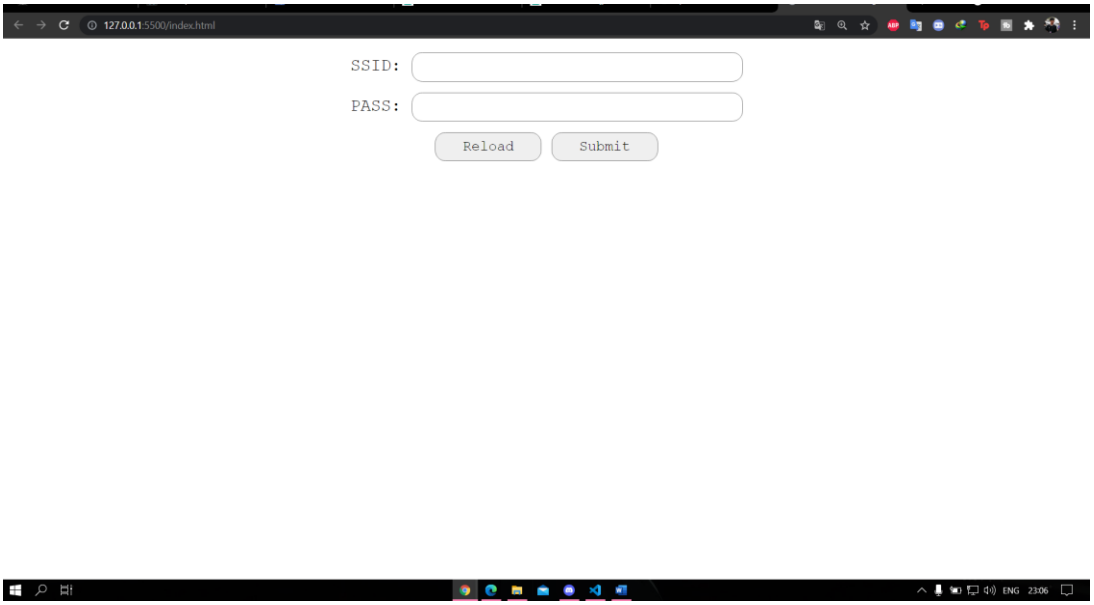
- ส่วนหน้าเข้าไปใน IP address /apSetup

ในส่วนหน้านี้ผมยังเข้าไปไม่ได้เพราะผมไม่มีบอร์ดในการเทสถ้าเข้าได้มันจะประมานนี้ครับ ตย.

- เข้า IP address 192.168.4.1/apsetup

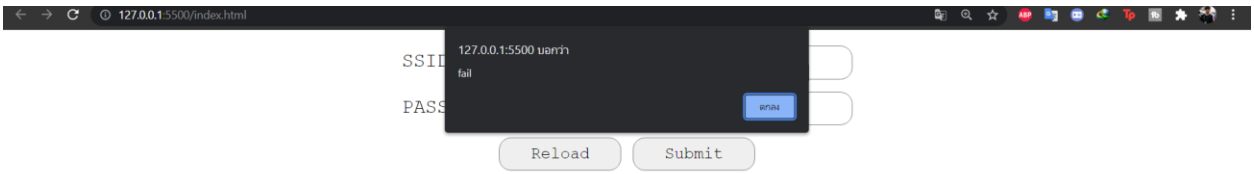
```
.........
Fail To Connect..
AccessPoint ssid: SPU_2G-[A4F544BF713C]
IP address(AccessPoint Mode): 192.168.4.1
HTTP server started
```

- อันนี้จะเป็นในส่วนหน้า SSID และ PASS ของ Wi-Fi ที่เราจะใช้

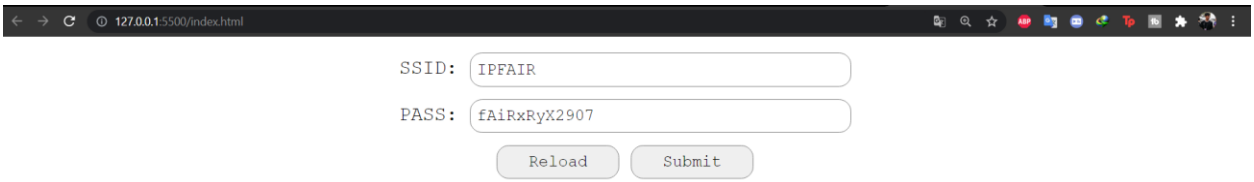


- ถ้าเวลาเราจะกด Summit มันจะแจ้งขึ้นมาว่า Sending Success

แต่ตอนนี้ของผมนั้นยังขึ้น Fail ยังไม่ได้ทำการเชื่อมต่อกันในทางบอร์ด ESP32



- ในการ Setup ตัว Wi-Fi

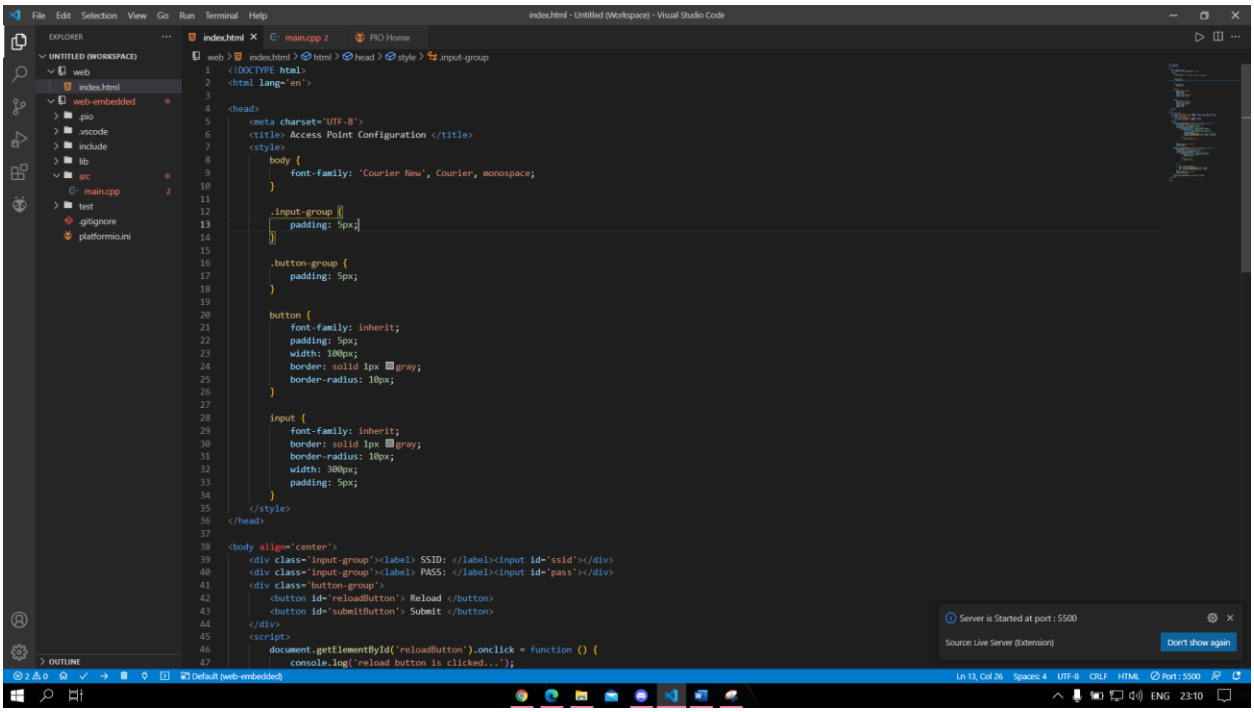


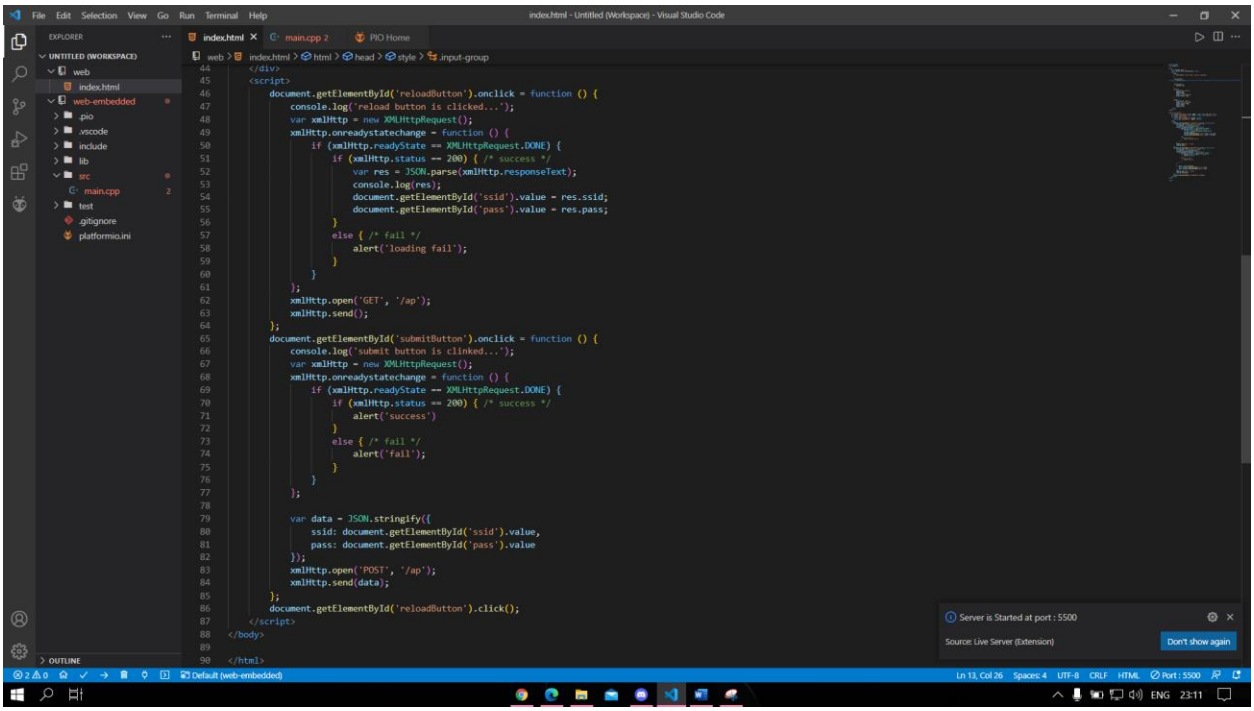
- จะมีการ reboots borad 1 ครั้ง

ตย.

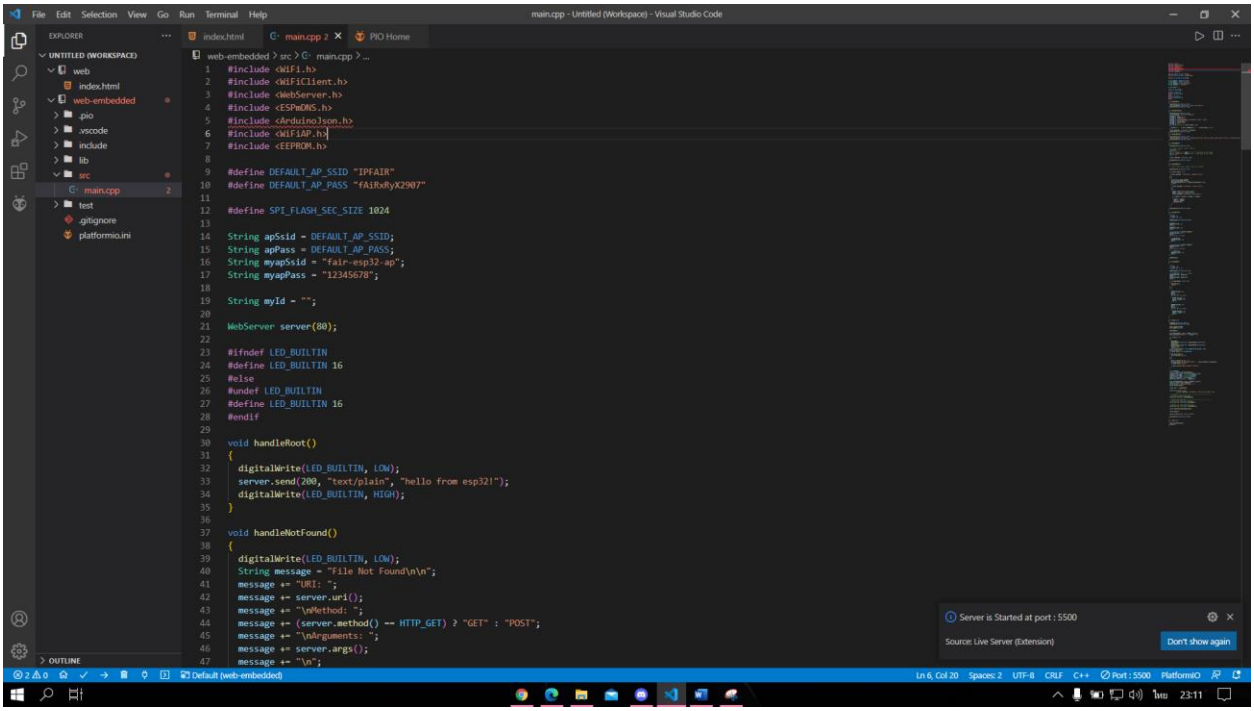
```
Connection Success..
IP address(STA mode): 192.168.1.51
MDNS responder started
AccessPoint ssid: SPU_2G-[A4F544BF713C]
IP address(AccessPoint Mode): 192.168.4.1
HTTP server started
```

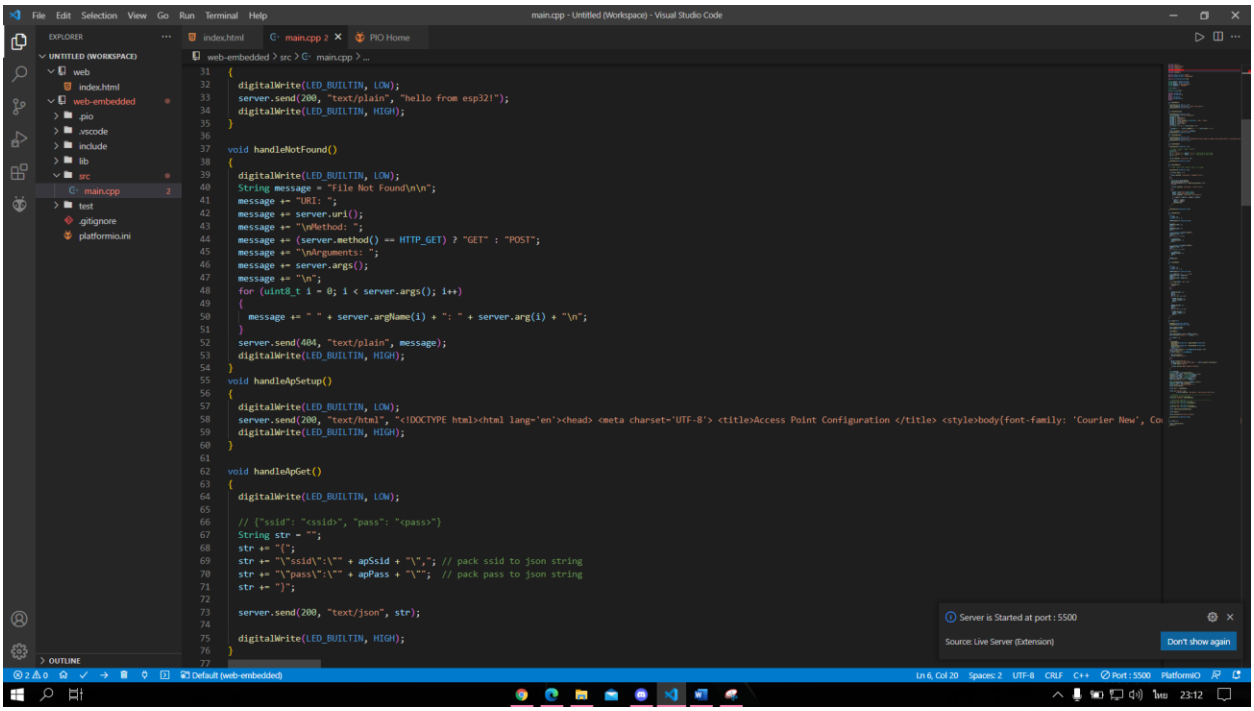
Code HTML

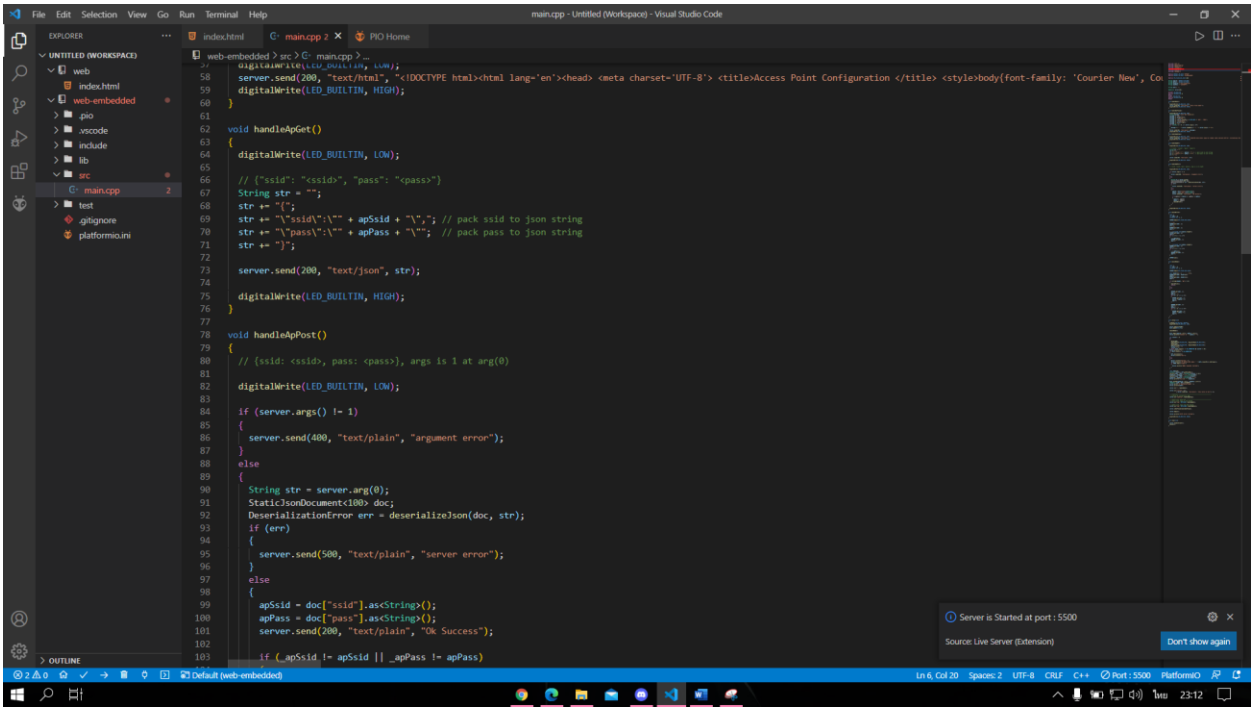


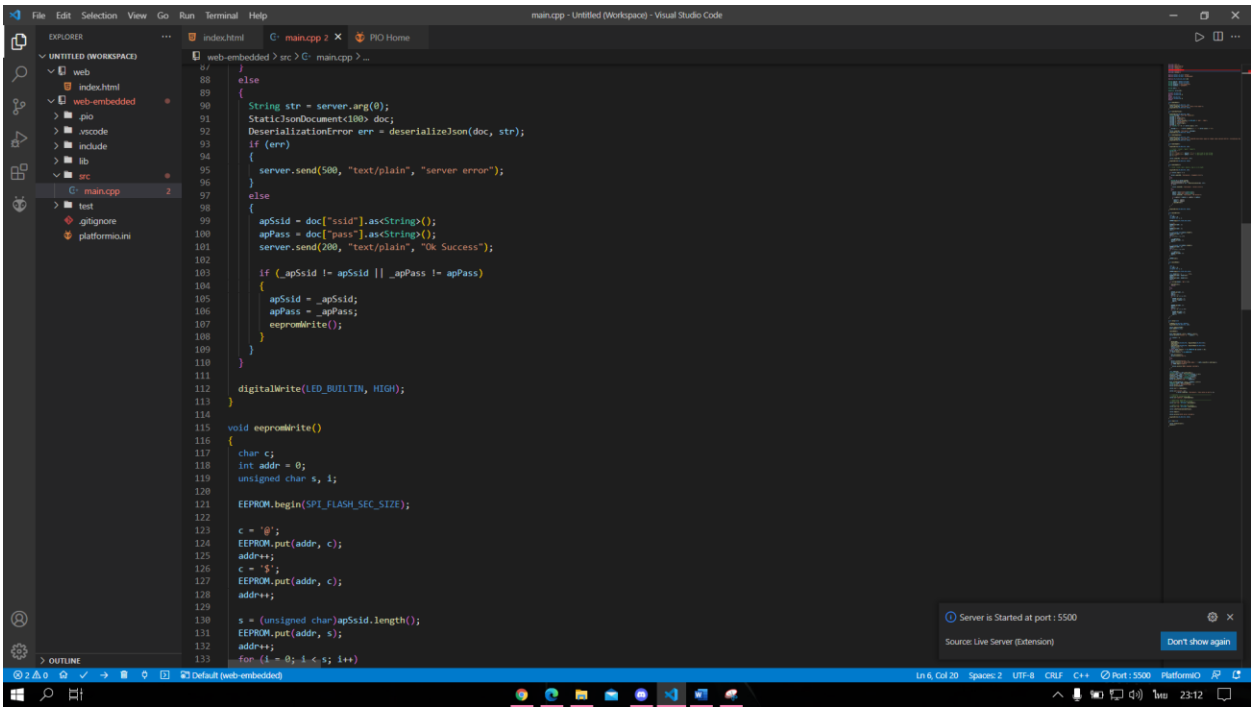


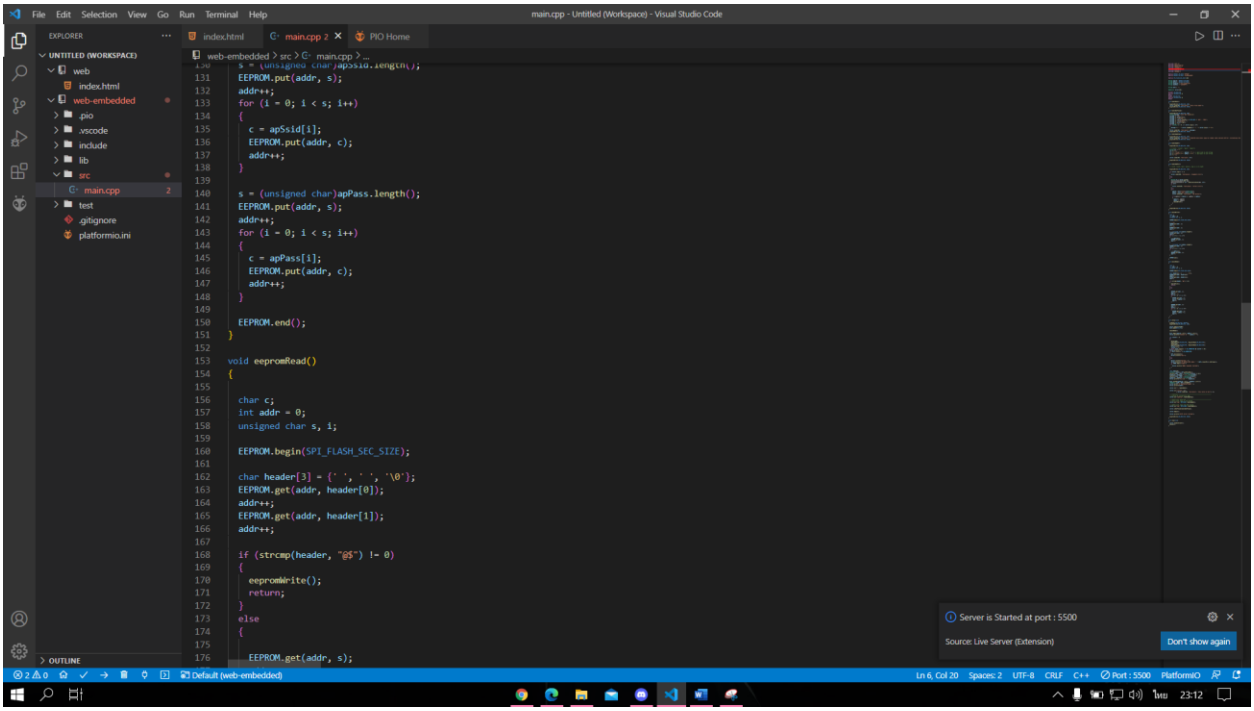
Code Main ในส่วน IO

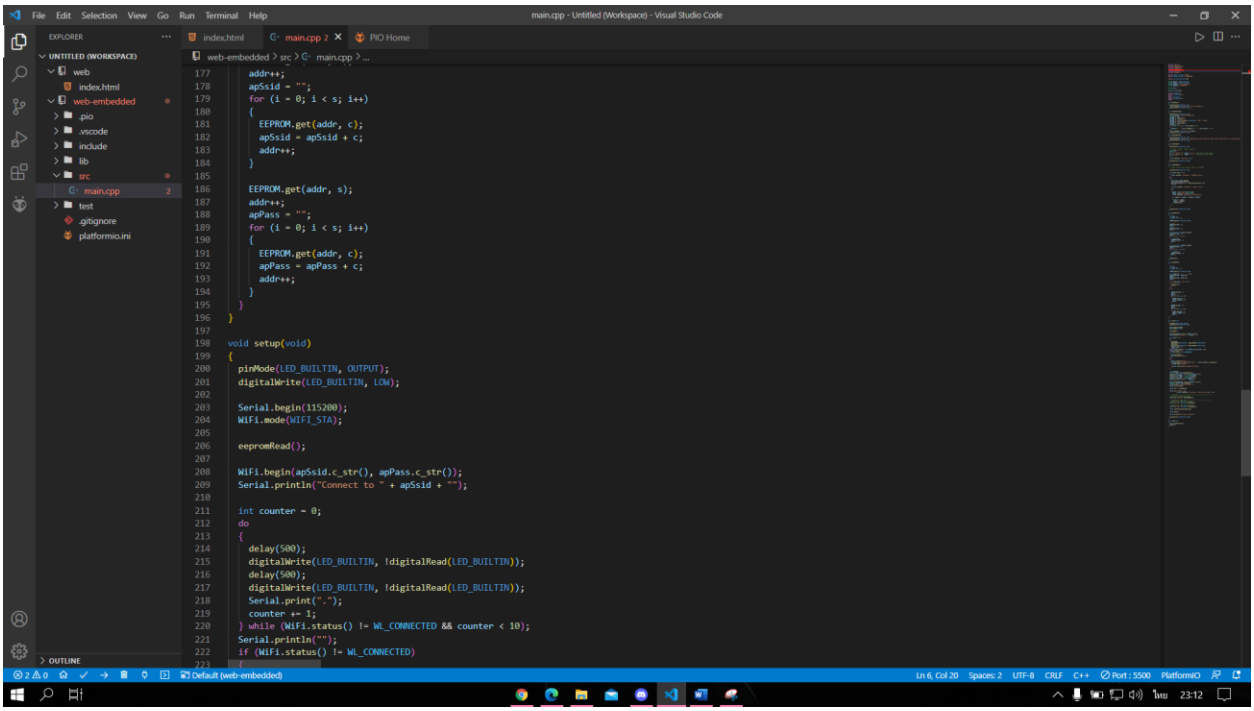


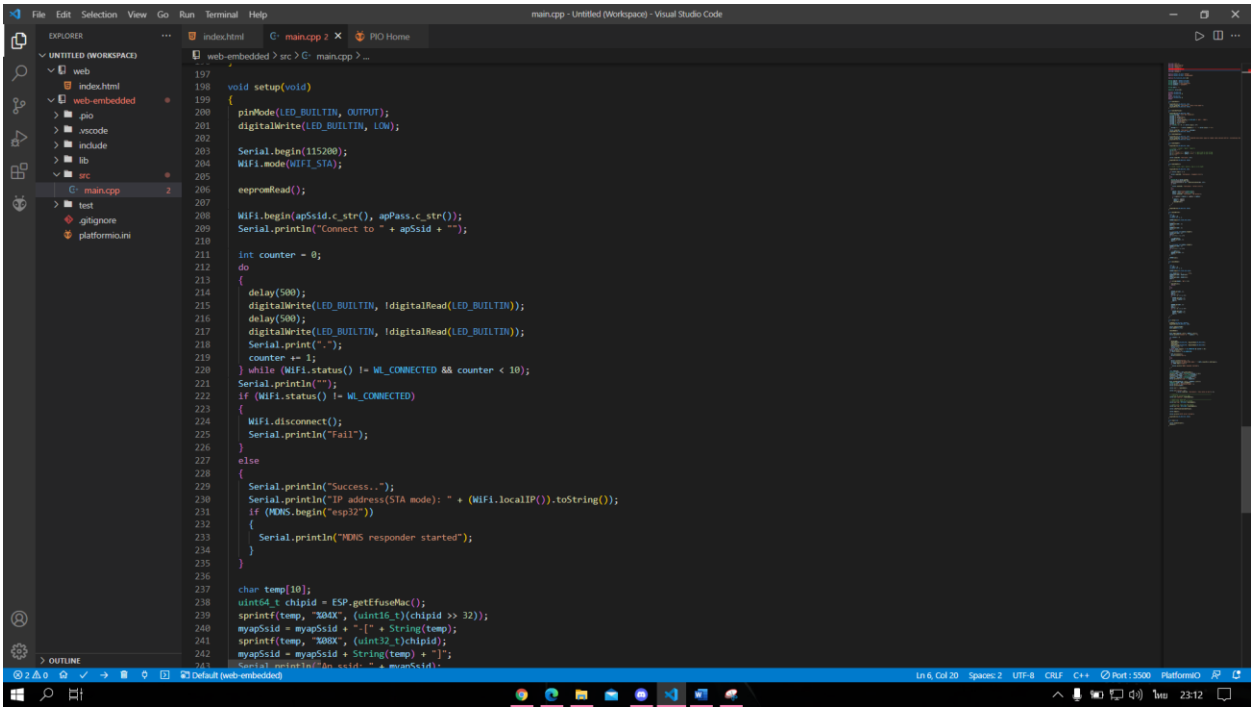

File Edit Selection View Go Run Terminal Help
main.cpp - Untitled (Workspace) - Visual Studio Code
EXPLORER
UNTITLED (WORKSPACE)
web
index.html
web-embedded
.pio
.vscode
include
lib
src
main.cpp
test
.gitignore
platformio.ini
index.html
main.cpp 2
PIO Home
web-embedded > src > main.cpp > ...
{
digitalWrite(LED_BUILTIN, LOW);
server.send(200, "text/plain", "hello from esp32!");
digitalWrite(LED_BUILTIN, HIGH);
}
void handleNotFound()
{
digitalWrite(LED_BUILTIN, LOW);
String message = "File Not Found\n\n";
message += "URI: ";
message += server.uri();
message += "\nMethod: ";
message += (server.method() == HTTP_GET) ? "GET" : "POST";
message += "\nArguments: ";
message += server.args();
message += "\n";
for (uint8_t i = 0; i < server.args(); i++)
{
message += " " + server.argName(i) + ": " + server.arg(i) + "\n";
}
server.send(404, "text/plain", message);
digitalWrite(LED_BUILTIN, HIGH);
}
void handleApSetup()
{
digitalWrite(LED_BUILTIN, LOW);
server.send(200, "text/html", "<!DOCTYPE html><html lang='en'><head> <meta charset='UTF-8'> <title>Access Point Configuration </title> <style>body{font-family: 'Courier New', Co
digitalWrite(LED_BUILTIN, HIGH);
}
void handleApGet()
{
digitalWrite(LED_BUILTIN, LOW);
// {"ssid": "<ssid>", "pass": "<pass>"}
String str = "";
str += "{";
str += "\"ssid\":\"" + apSsid + "\","; // pack ssid to json string
str += "\"pass\":\"" + apPass + "\""; // pack pass to json string
str += "}";
server.send(200, "text/json", str);
digitalWrite(LED_BUILTIN, HIGH);
}
Server is Started at port : 5500
Source: Live Server (Extension)
Don't show again
OUTLINE
Default (web-embedded)
Ln 6, Col 20 Spaces: 2 UTF-8 CRLF C++ Port : 5500 PlatformIO
23:12

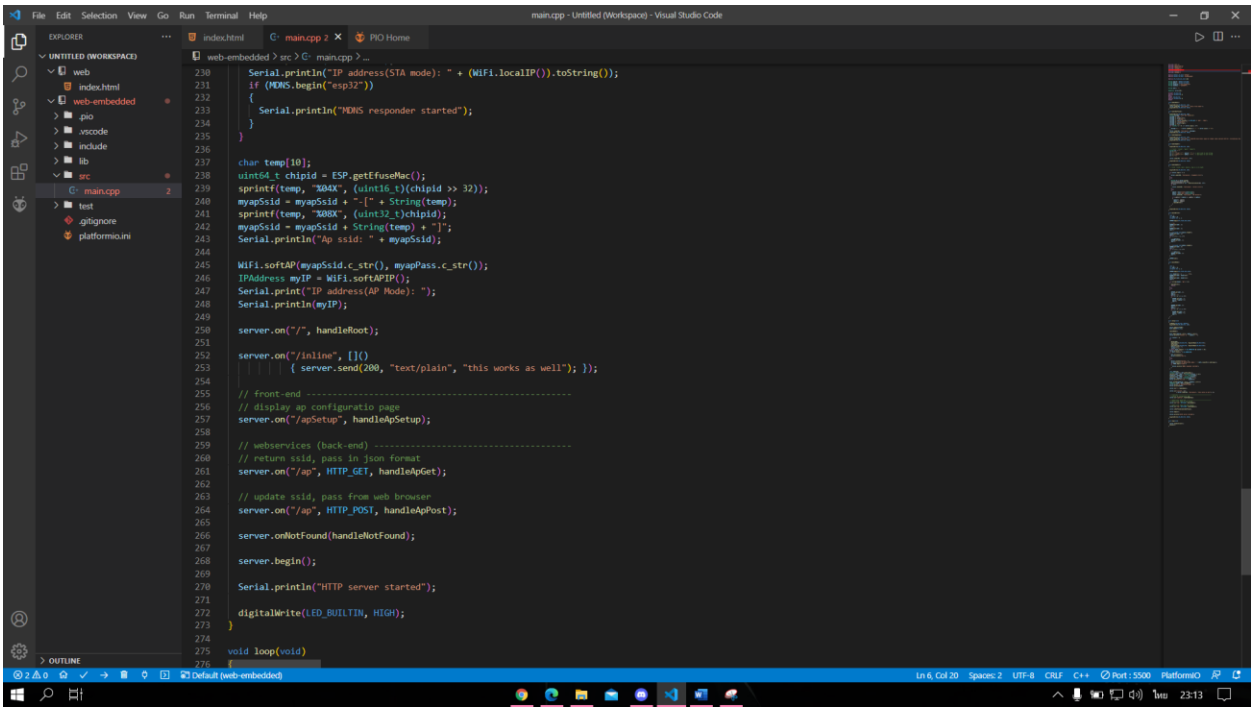

File Edit Selection View Go Run Terminal Help
main.cpp - Untitled (Workspace) - Visual Studio Code
EXPLORER
UNTITLED (WORKSPACE)
web
index.html
web-embedded
.pio
.vscode
include
lib
src
main.cpp
test
.gitignore
platformio.ini
index.html
main.cpp 2
PIO Home
web-embedded > src > main.cpp > ...
server.send(200, "text/html", "<!DOCTYPE html><html lang='en'><head> <meta charset='UTF-8'> <title>Access Point Configuration </title> <style>body{font-family: 'Courier New', Co
digitalWrite(LED_BUILTIN, HIGH);
}
void handleApGet()
{
digitalWrite(LED_BUILTIN, LOW);
// {"ssid": "<ssid>", "pass": "<pass>"}
String str = "";
str += "{";
str += "\"ssid\":\"" + apSsid + "\","; // pack ssid to json string
str += "\"pass\":\"" + apPass + "\""; // pack pass to json string
str += "}";
server.send(200, "text/json", str);
digitalWrite(LED_BUILTIN, HIGH);
}
void handleApPost()
{
// {ssid: <ssid>, pass: <pass>}, args is 1 at arg(0)
digitalWrite(LED_BUILTIN, LOW);
if (server.args() != 1)
{
server.send(400, "text/plain", "argument error");
}
else
{
String str = server.arg(0);
StaticJsonDocument<100> doc;
DeserializationError err = deserializeJson(doc, str);
if (err)
{
server.send(500, "text/plain", "server error");
}
else
{
apSsid = doc["ssid"].as<String>();
apPass = doc["pass"].as<String>();
server.send(200, "text/plain", "Ok Success");
if (_apSsid != apSsid || _apPass != apPass)
Server is Started at port : 5500
Source: Live Server (Extension)
Don't show again
OUTLINE
Default (web-embedded)
Ln 6, Col 20 Spaces: 2 UTF-8 CRLF C++ Port : 5500 PlatformIO
23:12


File Edit Selection View Go Run Terminal Help
main.cpp - Untitled (Workspace) - Visual Studio Code
EXPLORER
UNTITLED (WORKSPACE)
web
index.html
web-embedded
.pio
.vscode
include
lib
src
main.cpp
test
.gitignore
platformio.ini
OUTLINE
index.html
main.cpp 2
PIO Home
web-embedded > src > main.cpp > ...
}
else
{
String str = server.arg(0);
StaticJsonDocument<100> doc;
DeserializationError err = deserializeJson(doc, str);
if (err)
{
server.send(500, "text/plain", "server error");
}
else
{
apSsid = doc["ssid"].as<String>();
apPass = doc["pass"].as<String>();
server.send(200, "text/plain", "Ok Success");
if (_apSsid != apSsid || _apPass != apPass)
{
apSsid = _apSsid;
apPass = _apPass;
eepromWrite();
}
}
}
digitalWrite(LED_BUILTIN, HIGH);
}
void eepromWrite()
{
char c;
int addr = 0;
unsigned char s, i;
EEPROM.begin(SPI_FLASH_SEC_SIZE);
c = '@';
EEPROM.put(addr, c);
addr++;
c = '$';
EEPROM.put(addr, c);
addr++;
s = (unsigned char)apSsid.length();
EEPROM.put(addr, s);
addr++;
for (i = 0; i < s; i++)
Server is Started at port : 5500
Source: Live Server (Extension)
Don't show again
Default (web-embedded)
Ln 6, Col 20 Spaces: 2 UTF-8 CRLF C++ Port : 5500 PlatformIO
23:12

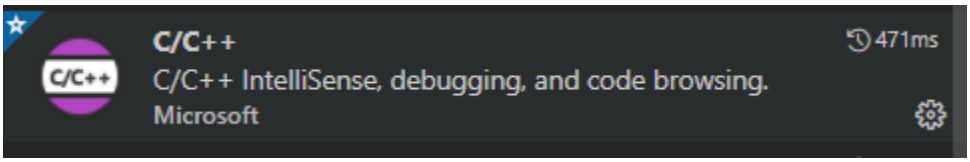

File Edit Selection View Go Run Terminal Help
main.cpp - Untitled (Workspace) - Visual Studio Code
EXPLORER
UNTITLED (WORKSPACE)
web
index.html
web-embedded
.pio
.vscode
include
lib
src
main.cpp
test
.gitignore
platformio.ini
OUTLINE
index.html
main.cpp 2
PIO Home
web-embedded > src > main.cpp > ...
s = (unsigned char)apSsid.length();
EEPROM.put(addr, s);
addr++;
for (i = 0; i < s; i++)
{
c = apSsid[i];
EEPROM.put(addr, c);
addr++;
}
s = (unsigned char)apPass.length();
EEPROM.put(addr, s);
addr++;
for (i = 0; i < s; i++)
{
c = apPass[i];
EEPROM.put(addr, c);
addr++;
}
EEPROM.end();
}
void eepromRead()
{
char c;
int addr = 0;
unsigned char s, i;
EEPROM.begin(SPI_FLASH_SEC_SIZE);
char header[3] = {' ', ' ', '\0'};
EEPROM.get(addr, header[0]);
addr++;
EEPROM.get(addr, header[1]);
addr++;
if (strcmp(header, "@$") != 0)
{
eepromWrite();
return;
}
else
{
EEPROM.get(addr, s);
Server is Started at port : 5500
Source: Live Server (Extension)
Don't show again
Default (web-embedded)
Ln 6, Col 20 Spaces: 2 UTF-8 CRLF C++ Port : 5500 PlatformIO
23:12

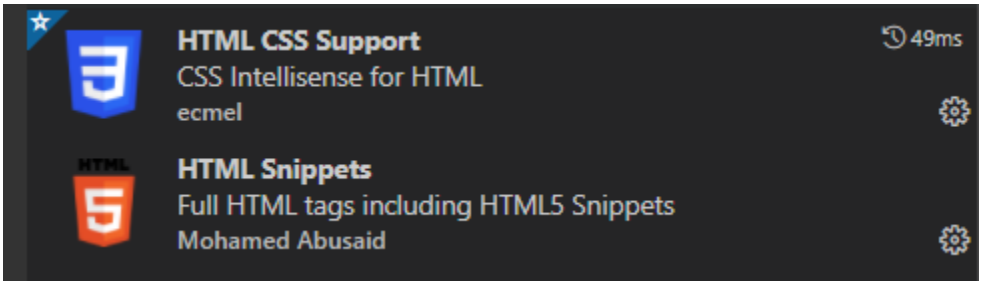

File Edit Selection View Go Run Terminal Help
main.cpp - Untitled (Workspace) - Visual Studio Code
EXPLORER
UNTITLED (WORKSPACE)
web
index.html
web-embedded
.pio
.vscode
include
lib
src
main.cpp
test
.gitignore
platformio.ini
OUTLINE
index.html
main.cpp 2
PIO Home
web-embedded > src > main.cpp > ...
177 addr++;
178 apSsid = "";
179 for (i = 0; i < s; i++)
180 {
181 EEPROM.get(addr, c);
182 apSsid = apSsid + c;
183 addr++;
184 }
185
186 EEPROM.get(addr, s);
187 addr++;
188 apPass = "";
189 for (i = 0; i < s; i++)
190 {
191 EEPROM.get(addr, c);
192 apPass = apPass + c;
193 addr++;
194 }
195 }
196 }
197
198 void setup(void)
199 {
200 pinMode(LED_BUILTIN, OUTPUT);
201 digitalWrite(LED_BUILTIN, LOW);
202
203 Serial.begin(115200);
204 WiFi.mode(WIFI_STA);
205
206 eepromRead();
207
208 WiFi.begin(apSsid.c_str(), apPass.c_str());
209 Serial.println("Connect to " + apSsid + "");
210
211 int counter = 0;
212 do
213 {
214 delay(500);
215 digitalWrite(LED_BUILTIN, !digitalRead(LED_BUILTIN));
216 delay(500);
217 digitalWrite(LED_BUILTIN, !digitalRead(LED_BUILTIN));
218 Serial.print(".");
219 counter += 1;
220 } while (WiFi.status() != WL_CONNECTED && counter < 10);
221 Serial.println("");
222 if (WiFi.status() != WL_CONNECTED)
223 {
Default (web-embedded)
Ln 6, Col 20 Spaces: 2 UTF-8 CRLF C++ Port : 5500 PlatformIO
23:12

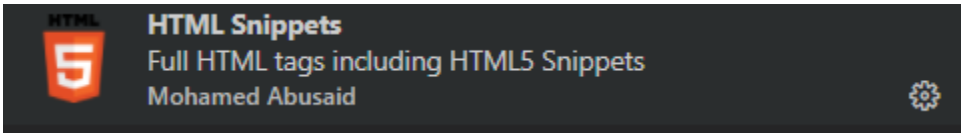

File Edit Selection View Go Run Terminal Help
main.cpp - Untitled (Workspace) - Visual Studio Code
EXPLORER
UNTITLED (WORKSPACE)
web
index.html
web-embedded
.pio
.vscode
include
lib
src
main.cpp
test
.gitignore
platformio.ini
OUTLINE
index.html
main.cpp 2
PIO Home
web-embedded > src > main.cpp > ...
197
198 void setup(void)
199 {
200 pinMode(LED_BUILTIN, OUTPUT);
201 digitalWrite(LED_BUILTIN, LOW);
202
203 Serial.begin(115200);
204 WiFi.mode(WIFI_STA);
205
206 eepromRead();
207
208 WiFi.begin(apSsid.c_str(), apPass.c_str());
209 Serial.println("Connect to " + apSsid + "");
210
211 int counter = 0;
212 do
213 {
214 delay(500);
215 digitalWrite(LED_BUILTIN, !digitalRead(LED_BUILTIN));
216 delay(500);
217 digitalWrite(LED_BUILTIN, !digitalRead(LED_BUILTIN));
218 Serial.print(".");
219 counter += 1;
220 } while (WiFi.status() != WL_CONNECTED && counter < 10);
221 Serial.println("");
222 if (WiFi.status() != WL_CONNECTED)
223 {
224 WiFi.disconnect();
225 Serial.println("Fail");
226 }
227 else
228 {
229 Serial.println("Success..");
230 Serial.println("IP address(STA mode): " + (WiFi.localIP()).toString());
231 if (MDNS.begin("esp32"))
232 {
233 Serial.println("MDNS responder started");
234 }
235 }
236
237 char temp[10];
238 uint64_t chipid = ESP.getEfuseMac();
239 sprintf(temp, "%04X", (uint16_t)(chipid >> 32));
240 myapSsid = myapSsid + "-[" + String(temp);
241 sprintf(temp, "%08X", (uint32_t)chipid);
242 myapSsid = myapSsid + String(temp) + "]";
Default (web-embedded)
Ln 6, Col 20 Spaces: 2 UTF-8 CRLF C++ Port : 5500 PlatformIO
23:12

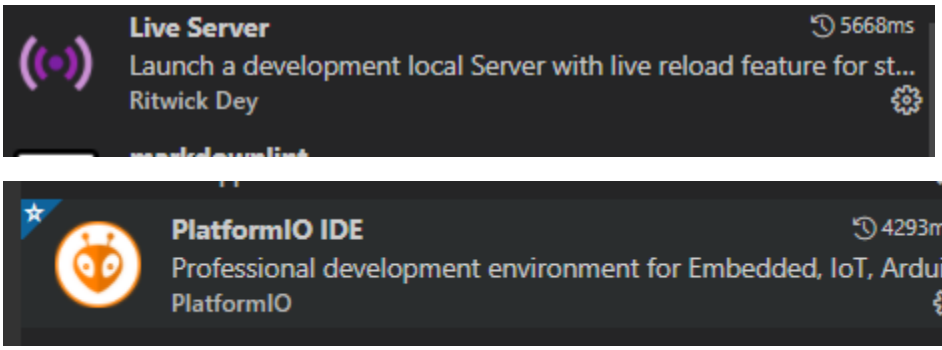


ส่วนเสริมที่ต้องใช้คร่าวๆที่ผมได้ลง








Live Server
5668ms
Launch a development local Server with live reload feature for st...
Ritwick Dey

PlatformIO IDE
4293ms
Professional development environment for Embedded, IoT, Ardui...
PlatformIO