ESP Smart Planter – Documentation

Below are pictures of the custom PCB

A purple circuit board with yellow dots

Description automatically generatedA purple circuit board with yellow dots

Description automatically generated

# Hardware:

* ESP32 Devkit v1 (30 pins)
* PN532 RFID reader
* LDR
* (Capacitive) soil moisture sensor
* I2C 20x4 LCD
* Passive buzzer (or speaker)
* Ds18b20 soil temperature sensor
* BME280
* WS2812B LED (COB) strip
* 4-channel (5v) relay module
* Water pump (12v)
* Fan (12v)
* Power supply (12v)
* Step-down buck converter (12 -> 5v or a variable variant if it can output 5v)
* Wires

# The software:

* Grafana
* InfluxDB
* Python (MQTT -> InfluxDB)
* C++ firmware (for ESP32)

# The Grafana setup

In Grafana we add the data source ‘InfluxDB’ using the created credentials and localhost because the database is running on the same device as the Grafana server.

We can then use graphs to display the temperatures and light values as well as use labels to display statuses of, for example, the fan or lights.

The ESP32 sends the following data:

* Air temperature
* Soil temperature
* Soil moisture
* Light level
* Water level
* Fan status
* Lights status
* Pump Status
* Current target (which plant)

All these datapoints are named appropriately in the influxdb.

# Installing the software and hardware

You can flash the ESP32 using PlatformIO or the Arduino IDE.  
Make sure to have all the required libraries installed!

All the required libraries are listed at the top of the C++ code.

If you are unsure of what pins to connect the sensors to, refer to the code or the provided schematic to see what sensors connect to which pins.

**Please make sure to change the WiFi SSID, Password & MQTT details BEFORE flashing the firmware to the board!**These can be found near the top of the code with the appropriately defined variables “WIFI\_SSID”, “WIFI\_PASSWORD”, “mqtt\_server”, “mqtt\_port”, “mqtt\_username” & “mqtt\_password”.

If you connect your sensors to other pins, you can change those too. All sensors are defined somewhere at the top portion of the code.

I recommend you either solder screw terminals to the PCB or solder wires that connect the sensors to the PCB, any other (semi-) permanent way is fine too except for breadboards as the tend to get some bad connections or disconnected wires sometimes.

# Important info:

Please refer to the sensor datasheets as some sensors should not be connected to the input of the ESP32 directly but instead need a pullup resistor or other special adjustments.  
These resistors are not shown in the schematic but are required to be present for the planter to work properly.

If your fan is **not** 12v but instead a 5v fan you can use the positive from the buck converter instead of the 12v wire, so you don’t blow up your fan motor.

Refer to the documentation of each service such as Grafana or MQTT on how to set it up.

If the ESP32 is no longer sending data, please check if your WiFi network is still available. Unplug the ESP32 immediately if it gets wet! If the ESP32 is unresponsive, unplug wait a few moments and plug it in again to force it to reboot and reconnect.

Schematics and other info can be found on the GitHub repository, please refer to the ReadMe file on there for the locations of these.