Sat composteur

Objectif

surveiller la température du compost.

Hardware

- micro-contrôleur : ESP32C3 de chez Seeed (seeed_xiao_esp32c3)
- 2 sondes de température DS18B20

Yaml ESPHome

composteur.yaml

```yaml
substitutions:
  devicename: composteur
  friendly_devicename: Composteur

esphome:
  name: ${devicename}
  friendly_name: ${friendly_devicename}

esp32:
  variant: ESP32C3
  board: seeed_xiao_esp32c3
  framework:
    type: arduino

# Enable logging
logger:
  # level: VERY_VERBOSE

# Enable Home Assistant API
api:
  encryption:
    key: "0987PÉUIÈ!VDLJ

ota:
  - platform: esphome
    password: "09rstceiupdl09876543ldvop"

wifi:
  ssid: !secret wifi_ssid
```

```
  password: !secret wifi_password

  # Enable fallback hotspot (captive portal) in case wifi connection
fails
  ap:
    ssid: "${devicename} Fallback Hotspot"
    password: "09PO09546XYP"

captive_portal:

one_wire:
  # D6
  - platform: gpio
    pin: GPIO21
    id: bus_ds18b20_1

  # D7
  - platform: gpio
    pin: GPIO20
    id: bus_ds18b20_2

sensor:
  - platform: dallas_temp
    one_wire_id: bus_ds18b20_1
    name: "Température 1 (${devicename})"
    resolution: 12
    update_interval: 60s

  - platform: dallas_temp
    one_wire_id: bus_ds18b20_2
    name: "Température 2 (${devicename})"
    resolution: 12
    update_interval: 60s

  - platform: wifi_signal
    name: "signal WiFi (${devicename})"
    update_interval: 60s

  - platform: uptime
    name: "${devicename} Uptime Sensor"
    id: ${devicename}_uptime_sensor
    update_interval: 60s
    on_raw_value:
      then:
        - text_sensor.template.publish:
            id: ${devicename}_uptime_human
            state: !lambda |-
              int seconds =
round(id(${devicename}_uptime_sensor).raw_state);
              int days = seconds / (24 * 3600);
```

```
              seconds = seconds % (24 * 3600);
              int hours = seconds / 3600;
              seconds = seconds % 3600;
              int minutes = seconds /  60;
              seconds = seconds % 60;
              return (
                (days ? String(days) + "j " : "") +
                (hours ? String(hours) + "h " : "") +
                (minutes ? String(minutes) + "m " : "") +
                (String(seconds) + "s")
              ).c_str();

text_sensor:
  - platform: wifi_info
    ip_address:
      name: "adresse IP (${devicename})"
      id: ${devicename}_ip_address

  - platform: template
    name: "Uptime (${devicename})"
    id: ${devicename}_uptime_human
    icon: mdi:clock-start

  - platform: version
    name: "Version d'ESPHome installée"
    id: ${devicename}_ESPHome_Version
```

Intégration Home Assistant