

HTW wintersemester 23/24

Internet of Things



Lead by
Prof. Alexander Huhn

Ting Ting Yang (s0592188)



Summary

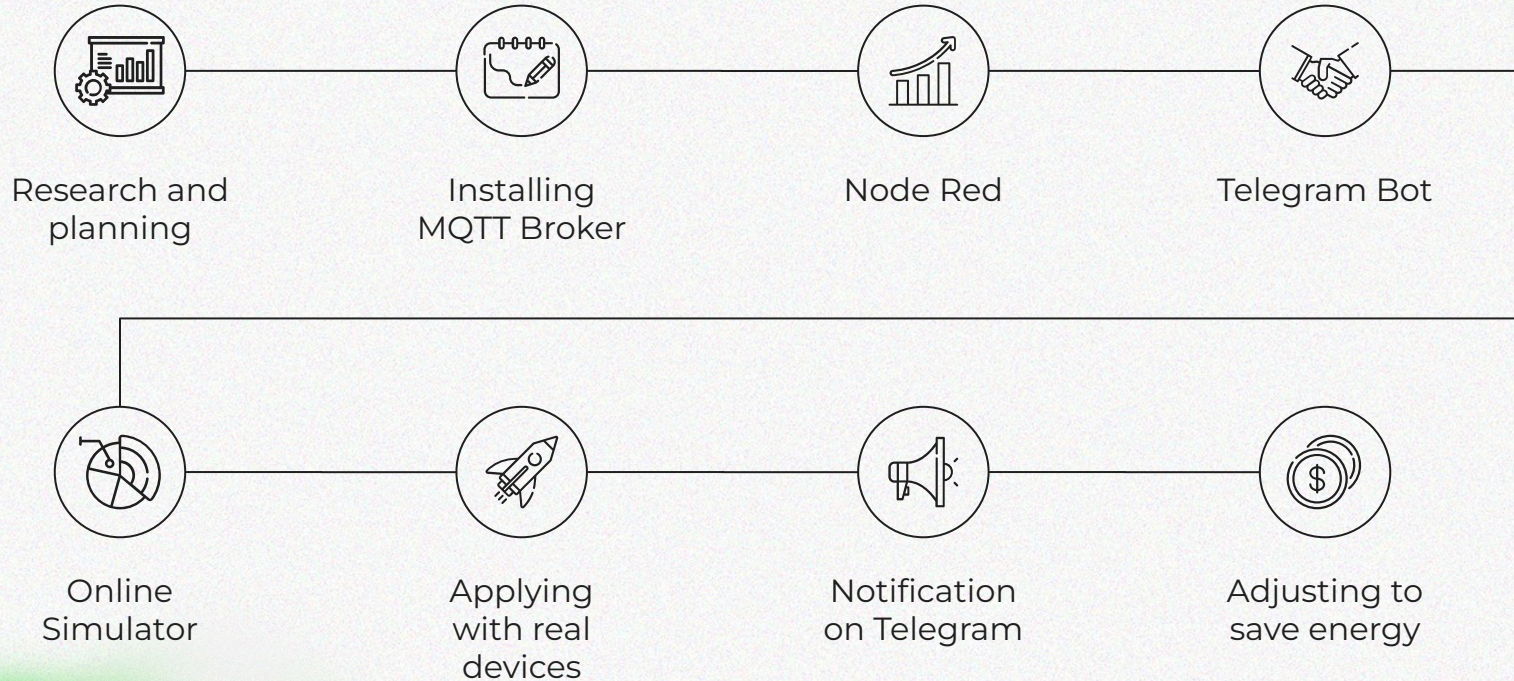


This project aims to **save energy bills** after the global natural gas crisis.

By using Micropython, Wokwi, MQTT, Telegram Bot API, and Node Red and equipped with devices such as microcontrollers, Raspberry Pis, and sensors. After a certain degree of humidity and temperature is detected, the user can decide whether the heating should be turned on.



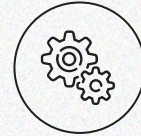
Project Development



Actual Implementation



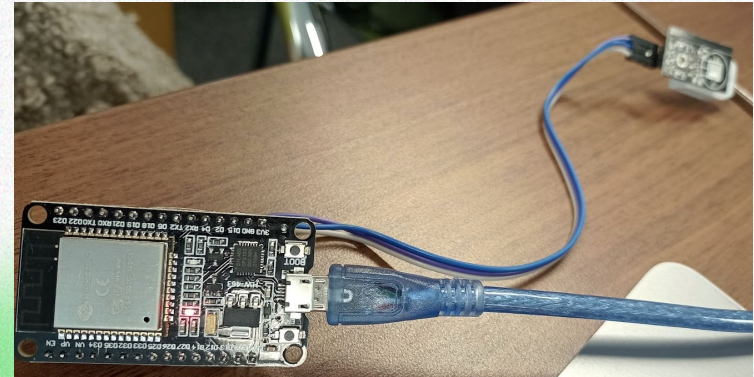
MQTT Weather Stimulator
(ESP32)



Microcontroller & Sensor
(In Real Life)

```
WOKWI SAVE SHARE MicroPython MQTT Weather Logger (ESP32) by urish
main.py diagram.json Library Manager Simulation
1 WOKWI_PROJECT = 'WOKWI-00031'
2
3 # WiFi
4
5 # MQTT
6
7 # DHT11
8
9 WiFi_PASSWORD = ''
10 MQTT_CLIENT_ID = 'Ting'
11 MQTT_BROKER = 'broker.hivemq.com'
12 MQTT_TOPIC_TEMP = 'M_IoT/ting/temp'
13 MQTT_TOPIC_HUMI = 'M_IoT/ting/humi'
14 DHT_PIN = 15
15
16 # Thresholds
17 TEMP_THRESHOLD = 12.0 # Celsius
18 HUMI_THRESHOLD = 80.0 # %
19
20 # Initialize WiFi
21 wlan = network.WLAN(network.STA_IF)
22 wlan.active(True)
23 wlan.connect(WIFI_SSID, WIFI_PASSWORD)
24 while not wlan.isconnected():
25     pass
26
27 # Read Temperature and Humidity
28 temp, humi = dht.read()
29
30 # Publish Temperature and Humidity
31 mqtt.publish(MQTT_TOPIC_TEMP, temp)
32 mqtt.publish(MQTT_TOPIC_HUMI, humi)
33
34 # Delay
35 time.sleep(10)
36
37 # Loop
38 while True:
39     # Read Temperature and Humidity
40     temp, humi = dht.read()
41
42     # Publish Temperature and Humidity
43     mqtt.publish(MQTT_TOPIC_TEMP, temp)
44     mqtt.publish(MQTT_TOPIC_HUMI, humi)
45
46     # Delay
47     time.sleep(10)
48
49 # End of Program
```

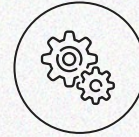
Temp is: 68.5
Humidity is: 26.5



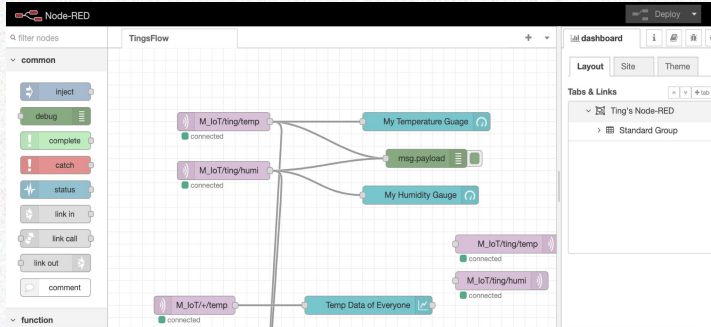
Actual Implementation



Node Red



Telegram Bot



HTW_IoT_Winter2324

bot

Temperature is below 12 degrees Celsius! Please turn on the heater!

17:58

Temperature is below 12 degrees Celsius! Please turn on the heater!

17:58

Temperature is below 12 degrees Celsius! Please turn on the heater!

18:04

Temperature is below 12 degrees Celsius! Please turn on the heater!

18:04



Question -
How IoT application help us to save bills?



Problem base :

The heater switch is mostly controlled manually. Without any force, what stays on will continue. It would easily produce additional costs.

Solution :

By applying IoT technology and having a sensor to detect temperature to instruct user when to turn on/off heater to make heater running more efficiently and reduce unnecessary cost

Step1

In your local terminal, start Node Red and connect your home device to the microcontroller to monitor changes in temperature.

Here we have ESP 32 run by Python, for example.



Step2

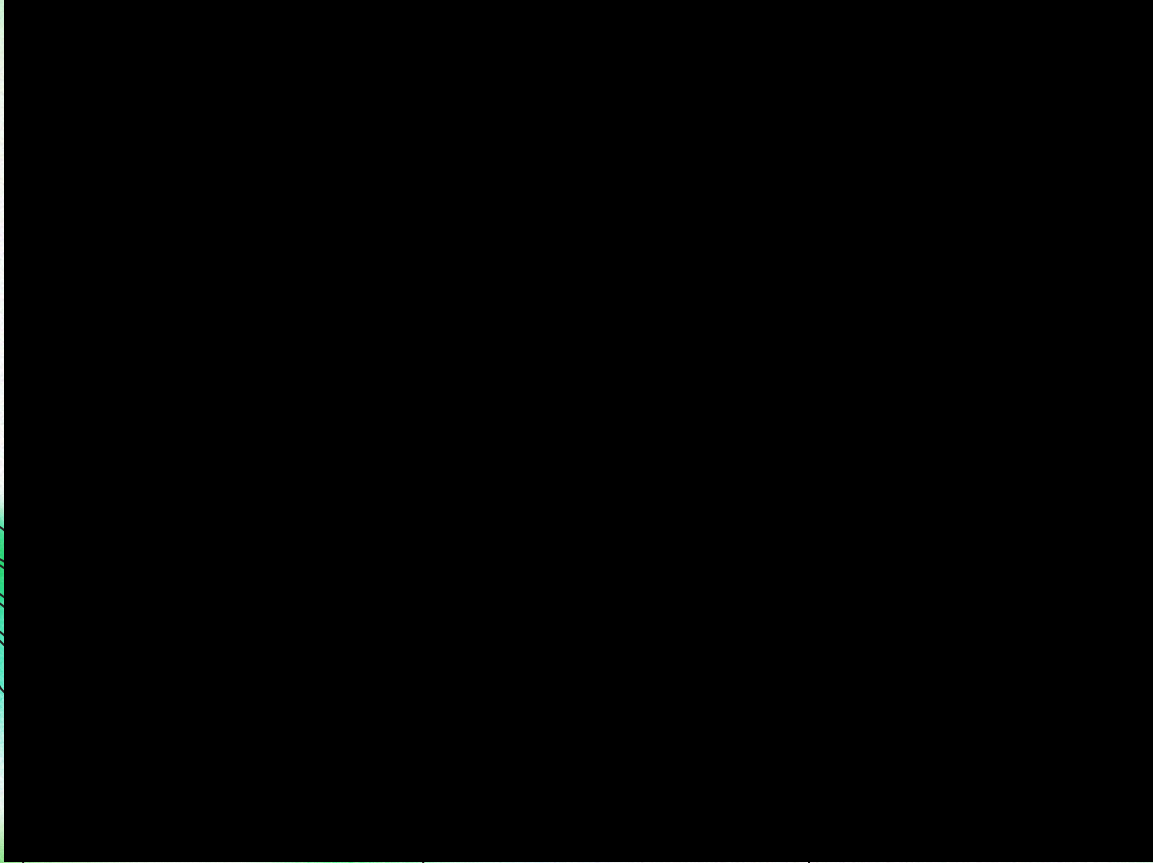
Sensor detects temperature and sends a message to Telegram in order to adjust the heater.





Showcase







Documentation

- https://gitlab.rz.htw-berlin.de/huhn/proit-d_ws2324/-/tree/s0592188/Documentation

Resources

- www.freepik.com
- https://de.123rf.com/photo_168964803_thermostat-on-a-wall-reading-71-degrees-fahrenheit.-selective-focus.-background-blur-and-foreground-blur.html
- <https://nodered.org/>

Thanks!

