



PUB TEMPERATURE MONITORING

Davide Rendina 830730
Andrei Gabriel Taraboi 829904

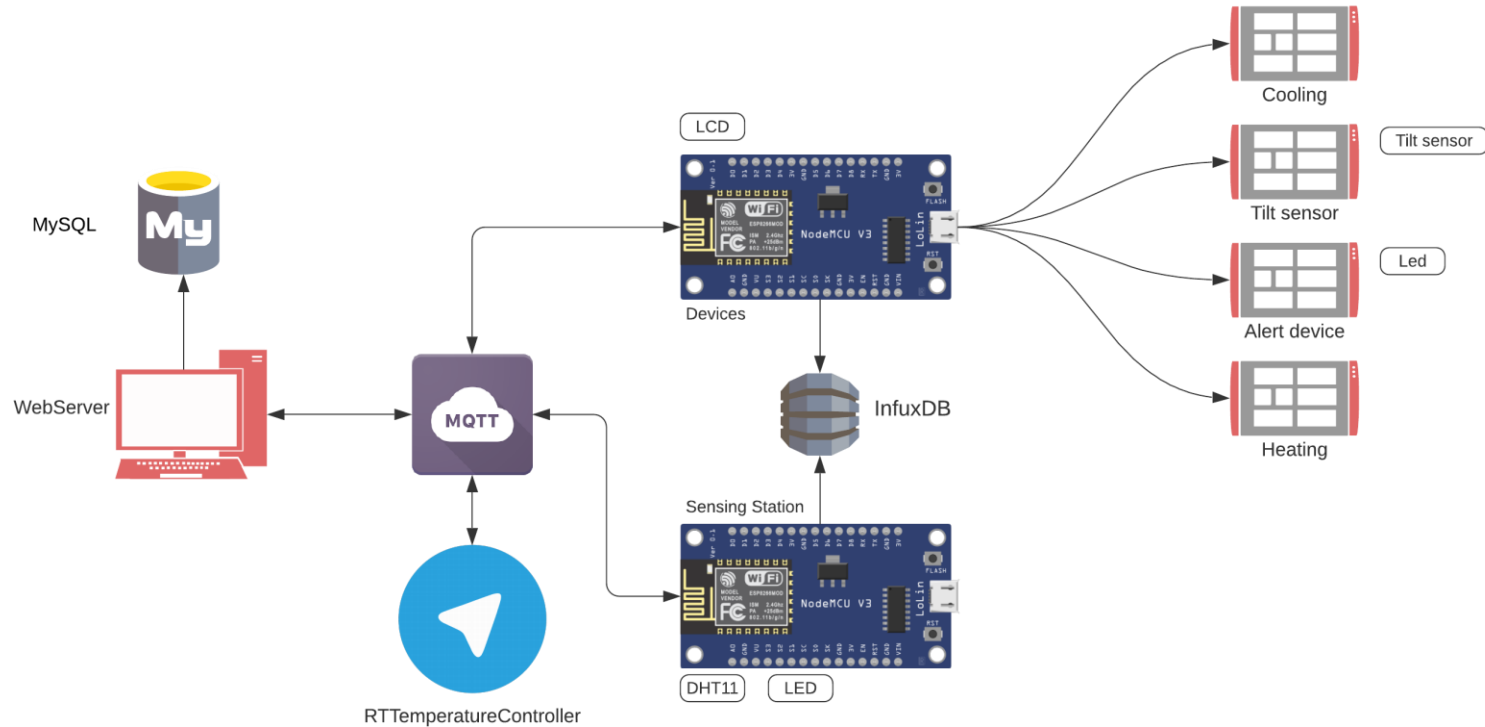


Obiettivi

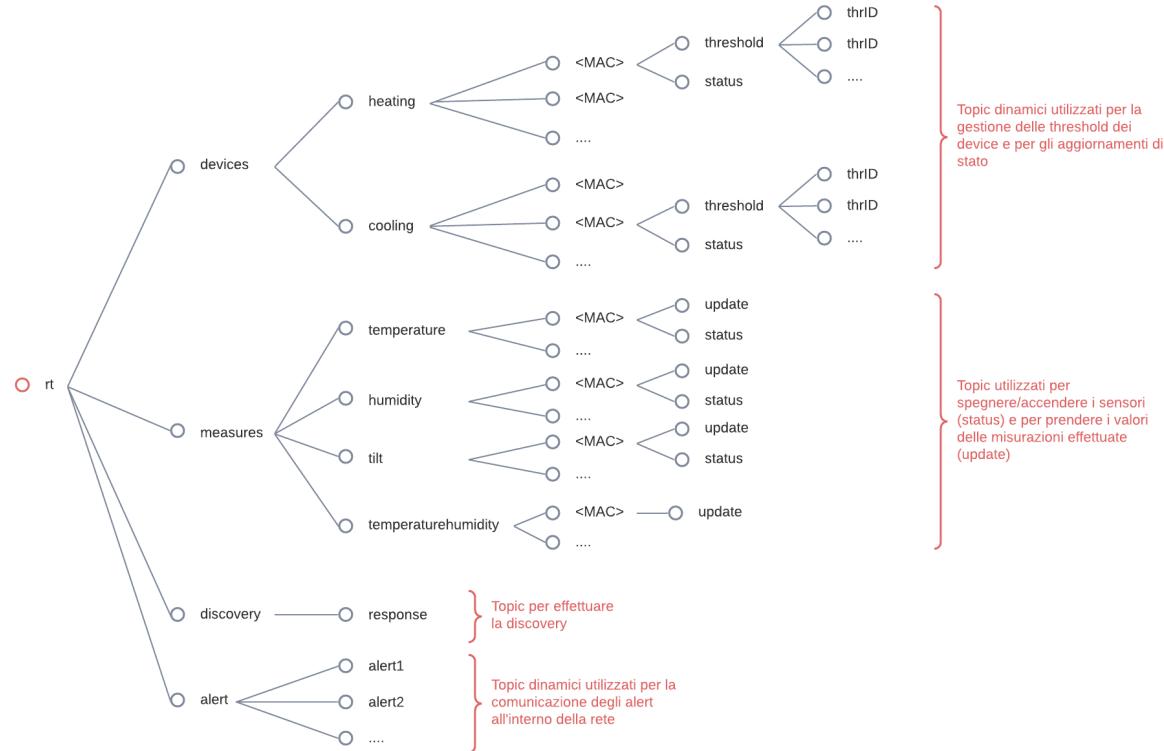


- Realizzare un sistema che permetta la gestione dei dispositivi di temperatura all'interno di un Bar.
- Eseguire un controllo sull'apertura e chiusura delle finestre dell'ambiente.
- Connettere i nodi della rete con MQTT.
- Realizzare una smart network che permetta l'autoconfigurazione dei dispositivi che si collegano.
- Utilizzare le Weather API per monitorare le previsioni meteo.
- Creare un'interfaccia web dinamica
- Costruire un sistema che offra grande dinamicità

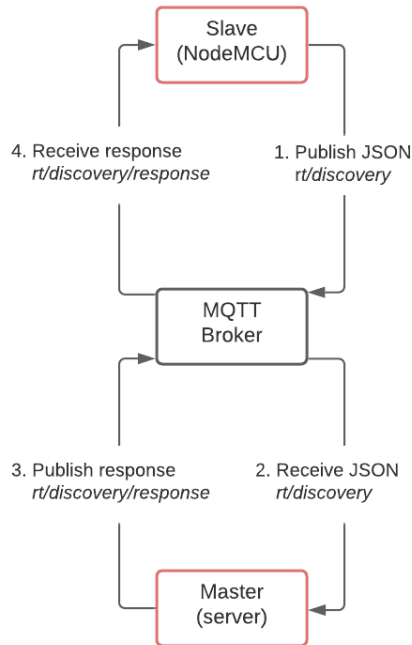
Struttura del sistema



MQTT



Fase di discovery



JSON discovery request structure

Device general info
(mac, name, description, type)

Threshold
(id, name, description, type, value)

Observes
(measureID)

Sensors
(id, name, status)

Alerts
(alertID)

JSON discovery response structure

General info
(mac, statusTopic)

Threshold topic
(lthresholdD : topic)

Observes topic
(measureID : topic)

Sensors topic
(sensorID : topic)

Alerts topic
(alertID : topic)

Database info
(connectInfo, pointDevice)

Controllo dispositivo acceso



1. Notifica

Ricezione di una notifica
se viene aperta/chiusa
una nuova finestra
oppure un device
cambia stato.



2. Controllo

Controlla che non ci siano
finestre aperte e device
accesi nello stesso
momento.



3. Comunicazione

Se il controllo ha esito
positivo, comunica l'alert
al sistema.

Controllo previsioni meteo



1. Chiamata API

Effettua una chiamata ogni 30 minuti alle API di *openweather.com* per chiedere le previsioni meteo.



2. Controllo

Controlla che non piovano nelle successive due ore e che non ci sia alcuna finestra aperta.



3. Comunicazione

Se il controllo ha esito positivo, comunica l'alert al sistema.

Risparmio energia



Sensing station

- Deep sleep ogni 10 minuti
- Log sulla rete ogni 30 minuti
- Utilizzo EEPROM



Tilt sensor

Deep sleep con risveglio tramite il sensore di tilt *



Heating/cooling

Nessuna strategia di risparmio necessaria

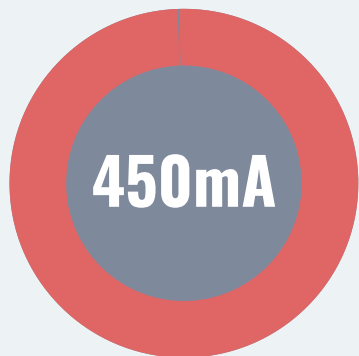


Alert Device

Deep sleep ad intervalli regolari di 5 minuti *

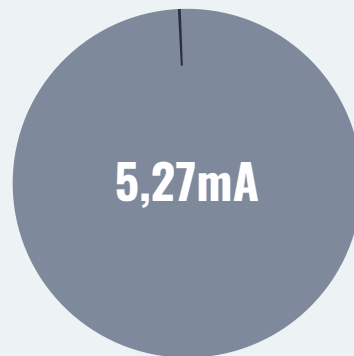


Consumo di corrente (teorico)



No sleep

Senza utilizzare strategie deep sleep, ogni dispositivo alimentato a batteria (8000mAh) rimarrebbe attivo per 47 ore continue.



Deep sleep

Utilizzando strategie deep sleep, *sensing station* e *alert device* rimarrebbero attivi per 99 giorni circa mentre il *tilt sensor* per 3 anni circa.

Interazione: Web APP



Home Temperature

Windows Count 0

Devices on 0

Alert

Alert window open: **off**

Heating

off

40:F5:20:04:3C:94-HT

Available sensors

Temperatura spegnimento

22

Cambia

Temperatura accensione

19.00

Cambia

Cooling

off

40:F5:20:04:3C:94-AC

Available sensors

Temperatura media

28.00

Cambia

Temperatura alta

32.00

Cambia

Umidità alta

80

Cambia

Alert device

40:F5:20:04:3C:94-AL

Available sensors

Tilt sensor

40:F5:20:04:3C:94-TL

Available sensors

tilt

Tilt sensor :
close

Acceso

Sensing station

40:F5:20:04:9E:DE

Available sensors

temperature

humidity

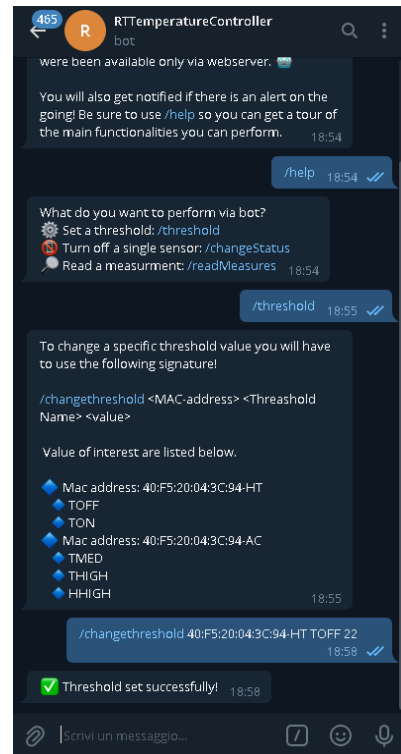
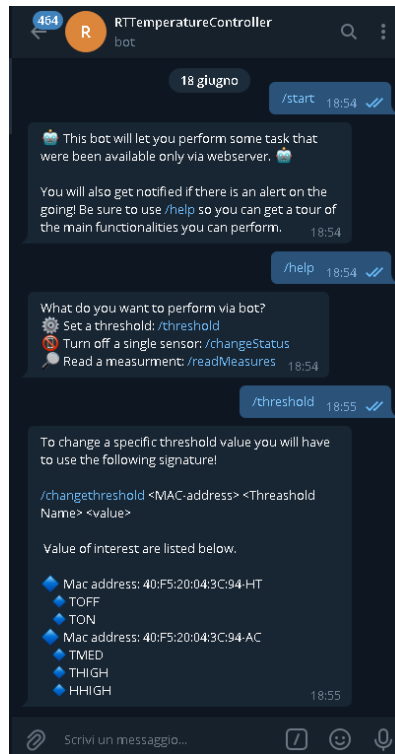
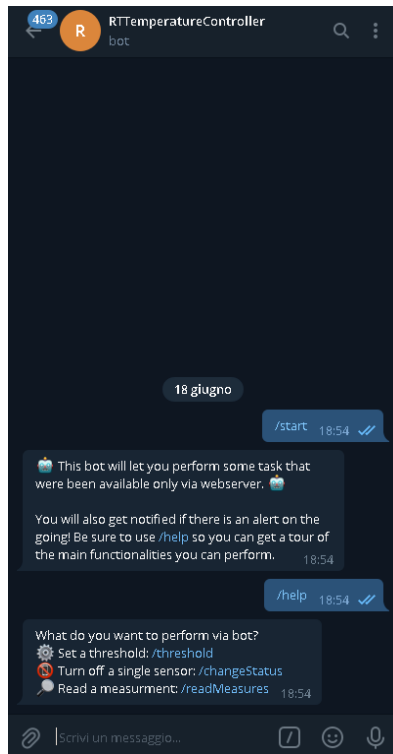
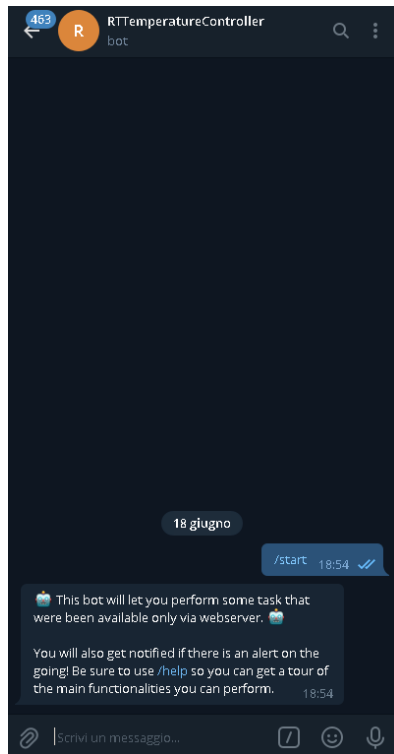
Temperature :
26

Acceso

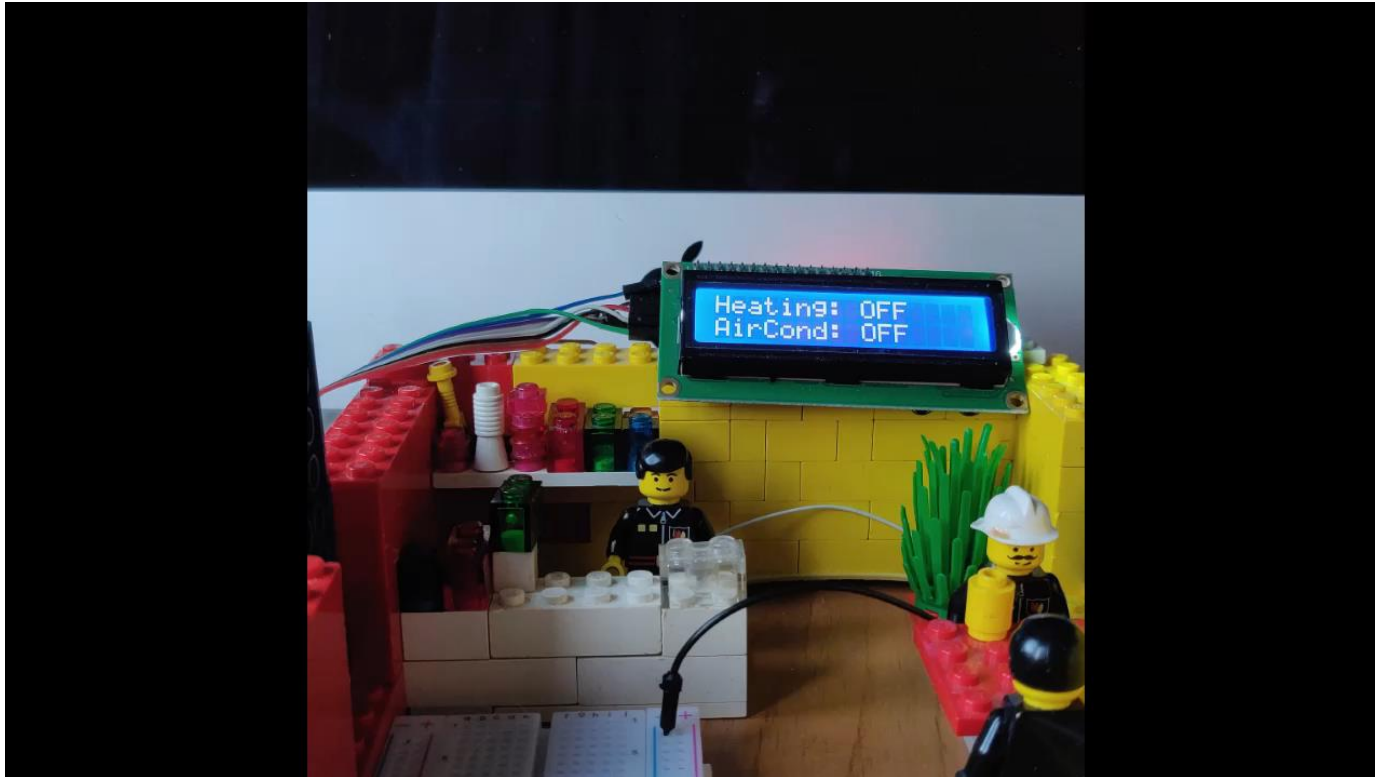
Humidity : 72

Acceso

Interazione: Telegram



Dimostrazione



Dimostrazione



The screenshot displays a web application for temperature control. On the left is a chat window titled 'RTTemperatureController bot'. The chat history shows commands like `/setThreshold`, `/changeStatus`, and `/readMeasures`. A recent message shows the bot's response to `/getMeasure`, listing available sensors: Tilt sensor, Temperature (26.00), and Humidity (72.00). The main panel, titled 'Home Temperature', features an 'Alert' section with a status 'off'. Below this are four control cards: 'Heating' (status 'off'), 'Cooling' (status 'off'), 'Alert device' (MAC address 40:F5:20:04:3C:94-AL), and 'Tilt sensor' (MAC address 40:F5:20:04:3C:94-TL). Each card lists available sensors and provides input fields for temperature and humidity, along with 'Cambia' buttons. At the bottom, a 'Sensing station' card shows the MAC address 40:F5:20:04:9E:DE and lists available sensors: temperature (26) and humidity (72), with 'Accesso' buttons.

Dimostrazione



The screenshot displays a web application interface for temperature control. On the left, a chat window titled "RTTemperatureController bot" shows a conversation. The chat history includes a command to change a threshold, a confirmation message, and a command to read measures. The current message in the chat shows the results of the "readMeasures" command: "Tilt sensor: open", "Temperature: 26.00", and "Humidity: 72.00". This message is highlighted with a red box. The main dashboard on the right, titled "Home Temperature", features a top navigation bar with "Windows Count 1" and "Devices on 0". Below this, there are several sections: "Alert" (Alert window open: off), "Heating" (off), "Cooling" (off), "Alert device" (40:F5:20:04:3C:94-AL), and "Tilt sensor" (40:F5:20:04:3C:94-TL). Each section has a list of available sensors and their current values. The "Sensing station" section shows the temperature and humidity values for the selected sensor (40:F5:20:04:9E:DE).

Chat Window (RTTemperatureController bot):

- Command: `/threshold`
- Message: To change a specific threshold value you will have to use the following signature!
`/changeThreshold <MAC address> <Threshold Name> <value>`
- Message: Value of interest are listed below.
 - Mac address: 40:F5:20:04:3C:94-HT
 - TOFF
 - TION
 - Mac address: 40:F5:20:04:3C:94-AC
 - TMED
 - THIGH
 - HHIGH
- Command: `/changeThreshold 40:F5:20:04:3C:94-HT TOFF 22`
- Message: Threshold set successfully!
- Command: `/readMeasures`
- Message: From here you can perform two different task.
Either you just ask all the measures now available via `/getMeasure`, or you could use the following signature!
`/getMeasure <MAC address> <Name>`
- Message: Value of interest are listed below.
 - Mac address: 40:F5:20:04:3C:94-TL
 - Tilt sensor
 - Mac address: 40:F5:20:04:9E:DE
 - Temperature
 - Humidity
- Command: `/getMeasure`
- Message: Tilt sensor: open
Temperature: 26.00
Humidity: 72.00

Main Dashboard (Home Temperature):

- Windows Count: 1
- Devices on: 0
- Alert: Alert window open: off
- Heating: off
- Cooling: off
- Alert device: 40:F5:20:04:3C:94-AL
- Tilt sensor: 40:F5:20:04:3C:94-TL
- Sensing station: 40:F5:20:04:9E:DE

Dimostrazione



The screenshot displays a web application interface for a temperature control system, divided into two main sections: a chat window on the left and a control panel on the right.

Chat Window (Left):

- Header: RTTemperatureController bot
- Content: A series of messages and commands. Key messages include:
 - Value of interest are listed below.
 - Mac address: 40:F5:20:04:3C:94-HT
 - Mac address: 40:F5:20:04:3C:94-AC
 - Mac address: 40:F5:20:04:3C:94-TL
 - Mac address: 40:F5:20:04:9E:DE
 - Temperature
 - Humidity
- Commands: /changeThreshold, /readMeasures, /getMeasure
- Status: Threshold set successfully!
- Alert: We caught an alert for you! Alert window open: is now on

Control Panel (Right):

- Header: Home Temperature
- Alert: Alert window open: on
- Heating: off
- Cooling: on
- Tilt sensor: 40:F5:20:04:3C:94-TL
- Alert device: 40:F5:20:04:3C:94-AL
- Sensing station: 40:F5:20:04:9E:DE

The interface also features a top navigation bar with 'Windows Count 1' and 'Devices on 1'.

Dimostrazione



RTTemperatureController
bot

- Mac address: 40:F5:20:04:3C:94-HT
- TOFF
- TON
- Mac address: 40:F5:20:04:3C:94-AC
- TIMED
- THIGH
- HHIGH

18:59

/changethreshold 40:F5:20:04:3C:94-HT TOFF 22

18:58

Threshold set successfully!

18:58

/readMeasures

19:26

From here you can perform two different task.

Either you just ask all the measures now available via
/getMeasure, or you could use the following signature:
/getMeasure <MAC address> <Name>

Value of interest are listed below.

- Mac address: 40:F5:20:04:3C:94-TL
- Tilt sensor
- Mac address: 40:F5:20:04:9E:DE
- Temperature
- Humidity

19:26

/getMeasure

19:26

Tilt sensor: close
Temperature: 26.00
Humidity: 72.00

19:26

/getMeasure

19:59

Tilt sensor: open
Temperature: 26.00
Humidity: 72.00

19:59

Will start raining at 00:00, be sure to close your windows.

20:03

⚠ We caught an alert for you! ⚠

Alert window open: is now on

20:12

⚠ We caught an alert for you! ⚠

Alert window open: is now off

20:13

Temperature control

127.0.0.1:8181

Home Temperature

Windows Count 0 Devices on 1

Alert
Alert window open: off

Heating
off

40:F5:20:04:3C:94-HT

Available sensors

Temperatura spegnimento
21.00 Cambia

Temperatura accensione
19.00 Cambia

Cooling
on

40:F5:20:04:3C:94-AC

Available sensors

Temperatura media
28.00 Cambia

Temperatura alta
32.00 Cambia

Umidità alta
80 Cambia

Tilt sensor

40:F5:20:04:3C:94-TL

Available sensors
tilt

Tilt sensor : close Acceso

Alert device

40:F5:20:04:3C:94-AL

Available sensors

Sensing station

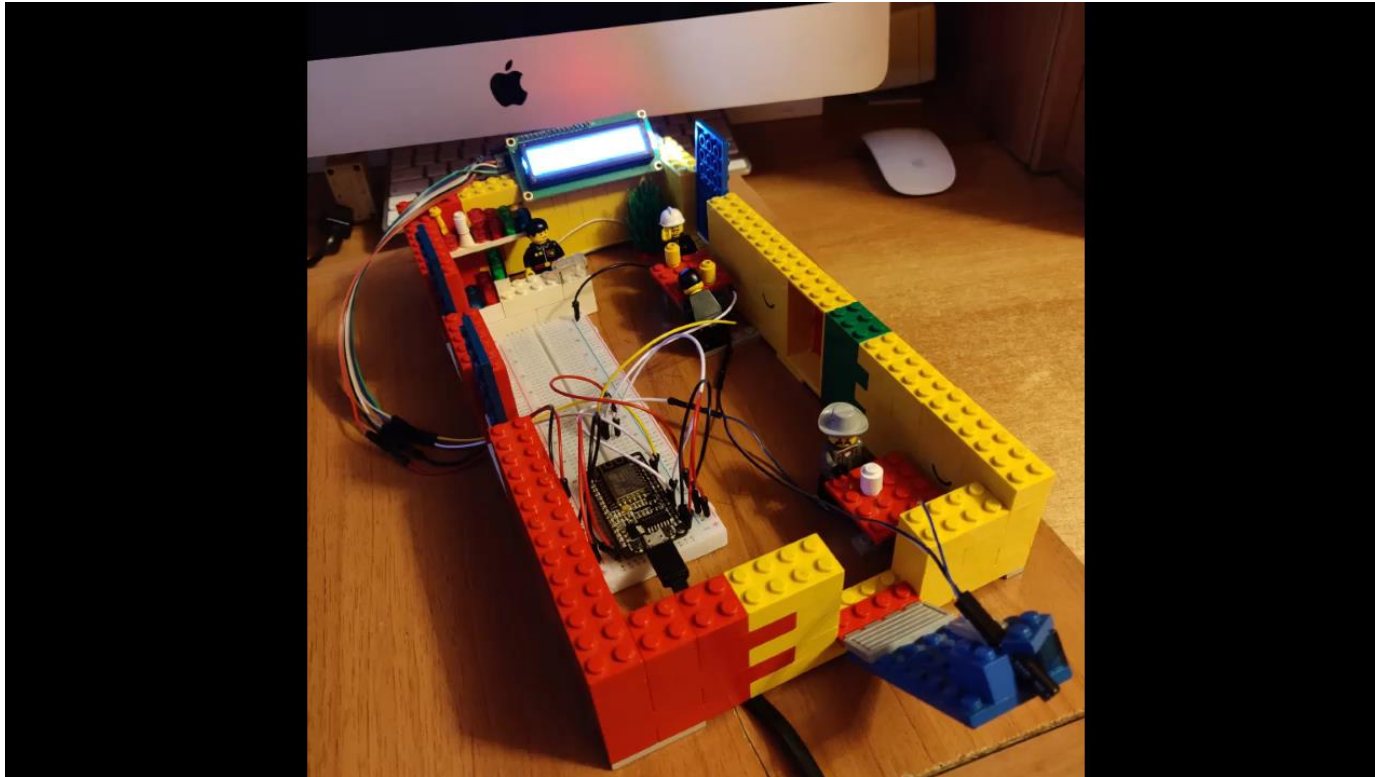
40:F5:20:04:9E:DE

Available sensors
temperature
humidity

Temperature : 26 Acceso

Humidity : 70 Acceso

Dimostrazione



Conclusioni

01

Devices

Utilizzo di un NodeMCU per ogni device, in modo da togliere la simulazione degli stessi.

02

Risparmio energetico

Adottare tecniche di deep sleep anche per i due dispositivi per cui non è stato possibile farlo (*alert device* e *tilt sensor*).





Grazie dell'attenzione

Davide Rendina 830730
Andrei Gabriel Taraboi 829904

