
This is the **published version** of the bachelor thesis:

Guzmán Ventura, Meritxell Anna; Aragonés Ortíz, Raul, dir. Mesurador de qualitat de l'aire per nodes de baix cost fent servir MQTT, Flutter, i Grafana. 2021. (958 Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/257834>

under the terms of the  license

Mesurador de qualitat de l'aire per nodes de baix cost fent servir MQTT, Flutter, i Grafana.

Meritxell Guzmán Ventura

Resum– Aquest projecte és l'ampliació d'una estació meteorològica creada amb un microcontrolador i diferents sensors de baix cost que monitorea aspectes mediambientals mitjançant el pas de missatges amb el protocol MQTT. Pretén arribar a l'autonomia energètica i sostenible garantint el mínim impacte ambiental. A més, és visible l'ús IoT per al desenvolupament del control de mètriques que permeten millorar la qualitat de vida i controlar la gestió de la nostra empremta ecològica. Utilitza un sistema de monitoratge de codi obert amb gràfiques que permeten visualitzar els històrics i tenir una visió generalitzada del funcionament del dispositiu. A més s'utilitzen serveis cloud per garantir que cada node es pugui utilitzar com un equip independent i permetre el seu desplegament sense limitacions geogràfiques.

Paraules clau– MQTT, monitoratge, IoT, autonomia energètica, microcontrolador, sistema distribuït

Abstract– This project is the extension of a weather station created with a microcontroller and different low cost sensors that track environmental aspects by passing messages with the MQTT protocol. It aims to achieve energy and sustainable autonomy by guaranteeing the minimum environmental impact. In addition, makes visible the use of IoT for the development of control metrics that improve the quality of life and control the management of our ecological footprint. It uses an open source monitoring system with graphics that allow you to view the data history and have an overview of the operation of the device. In addition, cloud services are used to ensure that each node can be used as a standalone computer and allow its deployment without geographical limitations.

Keywords– MQTT, monitoring system, IoT, autonomy, microcontroller, distributed system



1 INTRODUCCIÓ - CONTEXT DEL TREBALL

AQUEST projecte té l'objectiu de millorar una estació meteorològica creada amb un microcontrolador i diversos sensors per tal de millorar les seves presenciacions i crear una xarxa de nodes que permeti monitorar diferents espais connectats en una mateixa xarxa. Mitjançant l'anàlisi d'eines com node-red i HiveMQ em trobat una sol·lució per tenir tots els elements connectats sense importar on es trobin. En altres paraules, cada node no té per què estar a la mateixa xarxa WiFi per a formar part de tot el conjunt que es vol monitorar.

Avui en dia, amb la conscienciació sobre el canvi climàtic, es torna una necessitat el monitoratge de diversos factors que es tornen rellevants pel que fa a l'economia, energia i seguretat ciutadana i benestar. De fet, és un informe del Programa de les Nacions Unides, que diu que entre 1998 i 2017, van causar la mort de 1,3 milions de persones i uns 4,400 milions de ferits. [1] És per això que controlar els nivells de temperatura, humitat o qualitat de l'aire poden ser útils per preveure possibles incidents i establir estadístiques per obtenir una visió aproximada del futur i prendre consciència així com mesures que permetin canviar les mostres tal com siguin necessàries. Així doncs, l'ús de l'IoT és una gran eina a l'hora d'estudiar climes i permetre canviar els pronòstics que tenim actualment sobre el futur. Aquesta nova tendència de convertir en informatitzar les coses ve definida per la Indústria 4.0 en el seu intent de buscar una automatització de tots els processos mitjançant la intel·ligència artificial per tal que puguin analitzar i diagnos-

- E-mail de contacte: meritxellanna.guzman@autonoma.cat
- Menció realitzada: Enginyeria de Computadors
- Treball tutoritzat per Raül Aragonés
- Curs 2021/22

ticar problemes sense necessitat d'intervenció humana. Es per tal, que la posada en marxa d'una xarxa d'estacions meteorològiques SMART suposa el futur d'una tendència que podem veure en les smart cities d'avui en dia i que permet la millora en la distribució dels recursos a través de l'estalvi energètic o la distribució d'equips especialitzats en cas d'emergència de manera ràpida i eficient.

2 OBJECTIUS

- Crear un projecte end to end completament funcional mitjançant la utilització de la placa D1 R32.
- Realitzar la connectivitat de la placa amb un broker mitjançant un protocol MQTT i utilitzar un dispositiu mòbil per visualitzar les dades mesurades.
- Crear un sistema de missatges i notifikacions segons els nivells de contaminació de l'aire i altres sensors.
- Crear una aplicació mòbil utilitzant Flutter que permeti veure la informació captada pels sensors de manera intuïtiva i amigable.
- Crear un prototip funcional i modular, és a dir, que es puguin afegir altres tipus de sensors si s'escau.
- Gestionar una base de dades que ens permeti tractar les dades i crear gràfiques i anàlisis de manera visual.
- Ampliar els nodes de la xarxa MQTT per a monitorar més d'un espai diferent.
- Convertir el prototip en autònom i que consumeixi el mínim possible mitjançant el mode de suspensió de baix consum.
- Dissenyar un model en 3D per poder se exportat en una impressora 3D per a encapsular el dispositiu.

3 ESTAT DE L'ART

IoT permet a milions d'objectes percebre i comunicar informació. Aquests estímuls poden ser captats per altres nodes i així crear una xarxa interconnectada a través de diferents xarxes que és sensible a esdeveniments i actua amb relació a aquests. L'Internet de les Coses ha estat la revolució d'un món que sol estava dedicat als ordinadors i ara està present en gairebé tots els àmbits de la nostra vida en objectes etiquetats com a SMART.

Tot i això, queda molta feina per fer pel fet que aquestes noves tendències suposen un greu impacte en el consum energètic del nostre dia a dia. A més, una de les més grans preocupacions dels desenvolupadors i usuaris és la seguretat de les nostres dades que viatgen d'un punt a un altre a través de diferents servidors i que poden ser intervinguts per tercers. [9]

4 METODOLOGIA

S'ha seguit una metodologia basada en tasques amb Kanban i un diagrama de Gantt que mostra la planificació inicial que ha hagut de ser modificada, ja que no s'ha pogut treballar

amb la placa en certs moments i que han esdevingut un impediment per al desenvolupament del projecte. Tot i això, el període de finalització no s'ha vist afectat per aquests canvis.

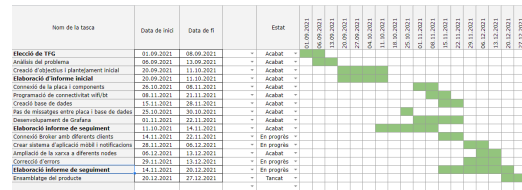


Fig. 1: Planificació Gantt

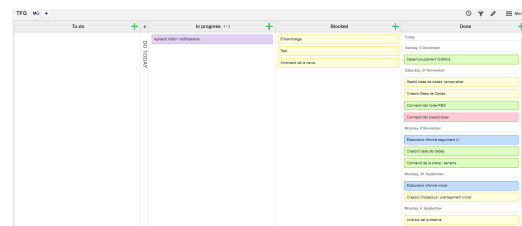


Fig. 2: Kanban

5 ARQUITECTURA UTILITZADA

S'ha utilitzat la placa D1 R32 [2] per a la realització d'aquest projecte. És una placa compatible amb l'entorn Arduino i de baix cost. Aquesta placa disposa de Wi-Fi a 2.4 i 2.5 Ghz (Això implica que les xarxes de 5 Ghz no seran detectades per aquesta placa) i Bluetooth integrats per al desenvolupament de projectes IoT. Conté un ESP32 WROOM-32 que integra un xip de 32 bits Tenselica. Disposa de 512kB de RAM. L'esquema pinout és el següent:

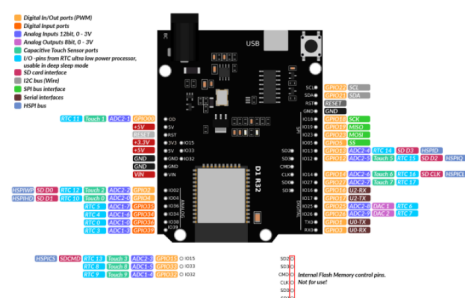


Fig. 3: Esquema Pinout

5.1 Sensors

5.1.1 DHT11

El DHT11 és un sensor digital que mesura la temperatura i la humitat relativa. Es connecta a 3,3V i necessita 1 segon per calcular la temperatura. El seu rang de temperatura varia de 0 a 50°C i té una precisió de ± 2.0 °C. Pel que fa a la humitat té un rang d'un percentatge de 20 a 90 RH i necessita la llibreria DHT.h per al seu funcionament.

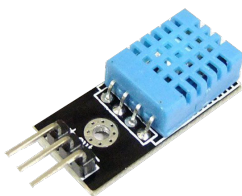


Fig. 4: DHT11

5.1.2 MQ-2



Fig. 5: MQ2

MQ-2 Pertany a una sèrie de sensors de gas que permeten monitorar la presència de gasos a l'aire. Cada sensor té una sensibilitat diferent i aquest en concret a gas metà, butà, LPG i fum. Tot i que aquest sensor s'ha utilitzat en les fases inicials del projecte, no està present en el prototip físic final com que no s'ha considerat per fer experimentació utilitzar gasos durant períodes llargs de temps i, per tant, no mostren mesures rellevants en els gràfics de Grafana. Aquest sensor es pot considerar útil com a detector de fum per a evitar concentracions d'aquest en un espai on es trobi el node.[8]

Per a la correcta utilització del sensor és necessari realitzar una calibració prèvia de 50 mesures per aconseguir el valor R0 que es farà servir posteriorment per al càlcul del ppm.

5.1.3 MQ-135



Fig. 6: MQ135

Aquest és el sensor mesurador de la qualitat de l'aire en ppm i també pertany a la sèrie de sensors de la família de MQ. Proporciona una sortida analògica i digital i pot detectar els gasos segons els diferents percentatges de gas a l'aire com pot ser, a part de CO₂, alcohol o amoníac. Abans de començar a mesurar, s'hi ha de preescalfar durant vint segons per a obtenir mesures estables. Per al calibratge s'ha utilitzat la llibreria MQ135.h

5.2 Mode baix consum

Necessitem crear un projecte energèticament sostenible. Aquesta placa ha de poder ser autònoma i no dependre d'u-

na font d'energia constant per al seu funcionament. Si volem optar a bateries, hem d'aconseguir que no estigui funcionant a ple rendiment tota l'estona. Si volem una mostra cada 5 minuts, la placa únicament necessita 10 segons per a realitzar la seva mesura. El temps restant ha d'estar en mode suspensió per evitar consumir energia de manera innecessària.

Després d'analitzar diferents modes de baix consum i de mirar diferents possibilitats de fer-ho ens decantem per utilitzar instruccions específiques per a la placa esp32. Primerament, es volia fer ús la llibreria LowPower.h, però no era compatible amb la placa. Els diferents modes de baix consum per la placa són els següents:

- Actiu: Aquest es el mode per defecte de la placa i està en ple funcionament tota l'estona.
- Modem-sleep: En aquest mode la CPU està activa i la placa es capaç de canviar de mode depenent si s'està utilitzant la comunicació WiFi. També desactiva els perifèrics, el Bluetooth i la ràdio.
- Light-sleep: L'especialitat d'aquest mode respecte a l'anterior és que el codi de la placa entra en un estat de pausa i per tant la CPU entra en el mode sleep.
- Deep-sleep: Aquest es el mode que menys consumeix i únicament està activat la ULP que es un coprocessador que s'activa únicament quan el sistema està en modes d'estalvi i el RTC.

Per activar el mode de deep sleep utilitzem la funció `esp_sleep_enable_timer_wakeup(TIME_TO_SLEEP * uS_TO_S_FACTOR)` posant els segons que volem que estigui en baix consum.

La funció `esp_sleep_pd_config(ESP_PD_DOMAIN_RTC_PERIPH, ESP_PD_OPTION_ON)`; serveix per definir la configuració dels RTC.

I finalment la funció `esp_deep_sleep_start` que activa el mode deep sleep.

D'aquesta manera, l'impacte energètic de la placa a causa de estar cinc minuts actiu i la seva connexió Wi-Fi es veu reduït per a poder ser autònom.[5]

5.3 Connexió WiFi

Aquesta placa disposa de connexió WiFi i Bluetooth integrada i per realitzar la connexió Wi-Fi de la placa s'ha d'utilitzar la llibreria `Wifi.h` disponible a l'entorn Arduino. Com a característica important d'aquesta connexió, és que la placa és capaç de detectar una xarxa de 2.4Ghz i que, per tant, si la xarxa és de 5Ghz no serà detectada i no s'hi podrà connectar.

Aquesta manera de connectar-se és la més senzilla i la que s'ha usat per dur a terme les proves amb la placa. Tot i això, hem contemplat diferents alternatives que s'expliquen a continuació.

5.3.1 Eduroam

La manera més senzilla per connectar-se a una xarxa que requereix usuari i contrasenya és mitjançant un programa nomenat Connectify Hotspot a l'ordinador i l'utilitzem per

compartir l'Internet posant un nom i una contrasenya especials i permetent a la placa connectar-s'hi.

Pel que fa a codi, hi ha una altra opció que consisteix a fer servir PEAP per la connexió utilitzant la variable EAP_IDENTITY com a nom d'usuari i EAP_PASSWORD com a contrasenya. Tot i això, aquesta manera conté encara alguns errors [6]

5.3.2 Portal captiu

Un portal captiu és una pàgina que es mostra als usuaris que es connecten a una xarxa abans de proveir-los d'accés a Internet. Un dispositiu com un microcontrolador pot simular que és un usuari que vol connectar-se a la xarxa de manera automàtica.

Primerament, cada portal captiu funciona de manera diferent i hi hauria d'acceptar-se la petició en l'àmbit de codi. Per aquesta connectivitat s'ha d'utilitzar un ordinador que es connecti al portal captiu per mirar totes les peticions i respostes que s'envien i reben durant el procés.

Finalment un cop s'han observat totes les accions entre la xarxa i l'ordinador es pot emular el mateix comportament utilitzant peticions http. D'aquesta manera es pot simular que s'ha fet clic sobre el botó que permet l'entrada com a convidat.[10]

5.4 Protocol MQTT

MQTT és un protocol de comunicació que requereix recursos mínims i dissenyat per a connectivitat M2M en entorns IoT. Les seves característiques principals[7] són les següents:

- Més lleuger que el protocol HTTP per transmetre grans quantitats de dades sense sobre costos i distribuïts en paquets petits.
- Permet un enviament asíncron de les dades
- Està orientada a esdeveniments i pot enviar dades d'un a molts clients
- És un protocol de baix consum respecte a altres alternatives
- És segur i capaç de transmetre dades de manera privada
- Permet una escalabilitat ràpida i poc costosa.

El format del missatge consisteix en el Byte1, que conté el tipus de missatge i el seu respectiu tag, que pot ser DUP, QoS o TETAIN, i el Byte2 que conté les dades que volen transferir.

El tag QoS, permet gestionar la robustesa de l'enviament de missatges al client davant una fallida en l'operació.

- QoS 0 unacknowledged (at most one): El missatge s'entrega una única vegada. Si hi ha un error el missatge podria no ser entregat.
- QoS 1 acknowledged (at least one): El missatge s'envia fins que és garanteix que s'ha arribat. Si hi ha un error, el subscriptor pot rebre missatges duplicats.

- QoS 2 assured (exactly one): És el més robust. garanteix que cada missatge s'entrega al subscriptor únicament una vegada.

6 HIVEMQ

HiveMQ és la solució que busquem a l'hora de trobar un broker de manera senzilla, flexible i segura. Hem triat aquesta plataforma, ja que té la possibilitat d'utilitzar-se de manera distribuïda com a clúster de manera gratuïta. Això permet que es pugui connectar qualsevol node que tingui la configuració del nostre broker des de qualsevol posició o xarxa. Per a aquest apartat, s'ha usat una connexió al servidor públic d'exemple que recapta els tòpics que publiquem, en aquest cas publicariem les dades que la nostra placa recapta. Per observar el tràfic de les nostres connexions podem emprar una aplicació alternativa anomenada MQTT Explorer[3], que ha estat molt útil a l'hora de veure les dades enviades i rebudes.

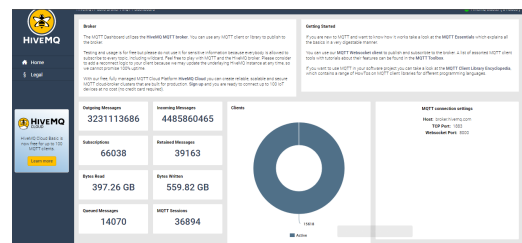


Fig. 7: Mostra del servidor públic de HiveMQ

Cal recalcar que en aquest Broker tothom hi té accés i que la publicació d'informació sensible podria ser recaptada per tercers. Per aquest motiu, s'ha creat un servei Cloud gratuït i privat per a publicar els nostres tòpics.

En el mode Cloud podem crear utilitzant servidors com Azure o Amazon AWS un broker privat on gestionar les connexions. A més disposem d'un WebSocket que ens serveix per comprovar la connexió i crear una subscripció o una publicació amb qualsevol tòpic.

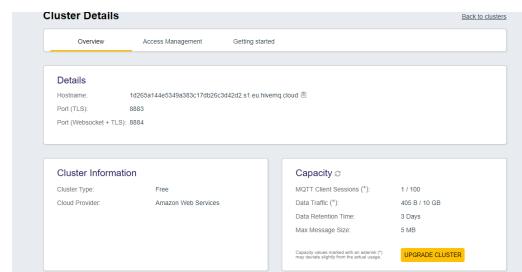


Fig. 8: Mostra del servidor privat de HiveMQ

Com mostrem a continuació ens connectem al servidor i accedim a totes les possibilitats d'aquesta eina. Es considera que ha estat un sistema molt potent a l'hora de desenvolupar una estructura de broker i que té moltes eines i documents per a diferents tipus de projecte, ja que HiveMQ també disposa d'exemples en altres llenguatges i un suport a la comunitat a l'hora de trobar solucions als

problemes dels seus usuaris

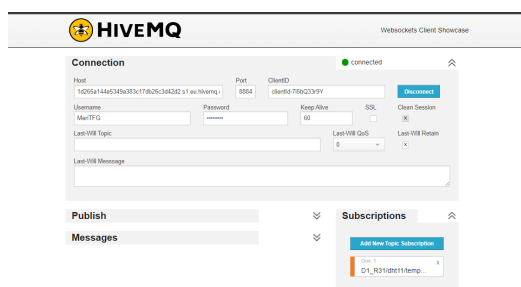


Fig. 9: Mostra del Websocket de HiveMQ

7 NODE-RED

Node-RED és l'eina que s'utilitza per connectar el hardware a les diferents API's que volem utilitzar. Se subscriurà als tòpics publicats al broker HiveMQ que envia la placa i s'encarregarà d'enviar-los a la base de dades temporal en forma numèrica, ja que rep les dades del broker en format text.

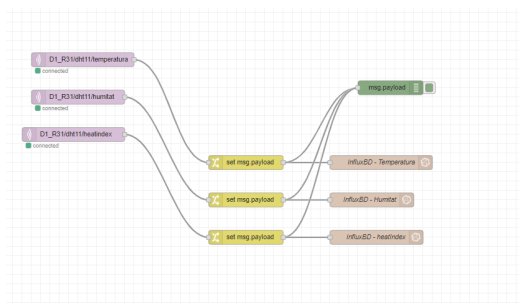


Fig. 10: Esquema Node-RED de la placa amb el sensor dht11

Aquesta eina està escrita en JavaScript en un entorn Node.JS i ens permet unir de manera visual mitjançant el seu panell completament personalitzable amb diferents mòduls que es poden afegir mitjançant paquets de tercers.

8 INFLUXDB CLOUD

InfluxDB és la base de dades temporal que ens permet guardar les mesures que generem. La versió utilitzada és la 2.0, que permet connectar-se des de Node-RED mitjançant un token. A més, aquesta versió destaca per les seves utilitats com el seu Dashboard on podem mostrar en diferents formats els resultats. També compta amb la possibilitat de crear tasques en un api unificada. [4]

Quan es crea una base de dades mitjançant InfluxDB, es genera un token, que és un codi alfanumèric per a connectar-ho a una API de manera segura, en aquest cas node-red. Un cop creat el token no pots canviar l'accés que ell proporciona i únicament és borra quan esborrem la base de dades que està lligat a ell.

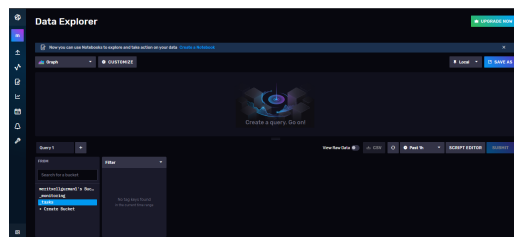


Fig. 11: Interfície inicial de InfluxDB Cloud

9 GRAFANA

Grafana és una eina de software lliure creada per visualitzar i monitorar dades temporals. Et permet dividir en parts el que vols monitorar i s'encarregarà automàticament d'anar-se mostrant periòdicament en pantalla.

Des d'InfluxDB podem exportar les dades a Grafana mitjançant Flux. A la figura següent observem un exemple per mostrar el monitora de la temperatura amb aquest llenguatge.

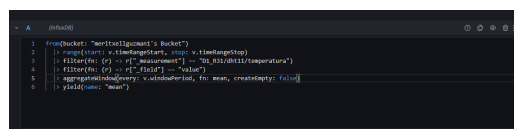


Fig. 12: Mostra de la temperatura amb Flux



Fig. 13: Mostra del Dashboard amb Grafana

Grafana permet crear diferents panells que continguin informació diferent i, per tant, poder diferenciar entre panells dades rellevants. També disposa d'un mode que refresca automàticament l'escena actualitzant al moment les mètriques i, per tant, és útil per als panells que continguin informació diària.

9.1 Alertes

Grafana permet definir alertes i notificacions gràcies al fet que en la seva nova versió 8.0 ha integrat el seu propi sistema amb l'ajuda de Prometheus. Per connectar un canal de notificacions podem posar un correu o Telegram, però s'ha acabat decantant per Telegram perquè una mala gestió de les notificacions enviades al correu podria catalogar-se com a correu brossa i passar desapercebut a l'usuari. A més es pot crear un panell on es pot veure l'estat de les alertes i fins i tot un històric d'aquestes. Hi ha dues maneres de generar una alerta; la primera, serveix per a fer una alerta instantània, cada cert temps s'encarregarà de mirar el valor de la mètrica que li hem demanat vigilar i alertarà en conseqüència. La segona manera és similar, la diferència consisteix en el fet que cada cert temps vigilarà la mètrica durant una estona que nosaltres triem i determinarà l'estat de l'alerta passat

aquest temps. D'aquesta manera, s'eviten alertes falses o irrelevantes.

Les alertes disposen de quatre estats diferenciats.

- **OK:** Aquest estat es manté quan una condició d'alerta no es compleix, garantint que el sistema funciona de manera correcta.
- **NO DATA:** Aquesta alerta apareix quan Grafana no detecta dades i avisa que possiblement algun component ha deixat de funcionar.
- **PENDING:** Aquesta alerta està analitzant el comportament de les mètriques per determinar si alertar o si simplement s'ha complert la condició d'alerta de manera puntual.
- **WARNING:** Aquesta alerta indica que s'ha complert la condició que s'ha establert i per tant envia un avís mitjançant el canal que li hem especificat. Es mantindrà en aquest estat fins que es deixi de complir la condició.

A continuació es mostra una secció on podem trobar alguns dels diferents estats quant a l'alerta. Es pot observar que en verd significa el punt en el fet que la temperatura ha deixat de complir la condició d'alerta i en vermell quan la temperatura comença a baixar. Finalment, quan el dispositiu està apagat, mostra en gris, que ja no hi ha dades.

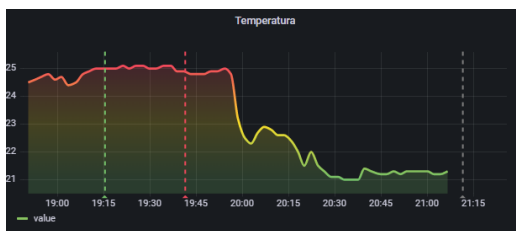


Fig. 14: Alerta en el gràfic

L'alerta es pot crear basant-se en una mitjana de valors, màxims, mínims, comptadors i sumes, entre d'altres i és una eina molt potent per a controlar de manera automàtica valors com la temperatura d'una sala. No serà necessari que s'estigui pendent de les gràfiques, ja que s'ha desenvolupat un sistema de notifikacions mitjançant l'aplicació Telegram que permetrà tenir les alertes a qualsevol dispositiu mòbil que disposi d'aquesta aplicació.

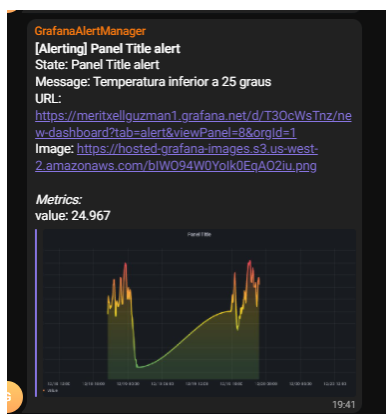


Fig. 15: Mostra d'una alerta amb Telegram

10 FLUTTER

Flutter és un framework multiplataforma que utilitza el llenguatge Dart per al disseny d'aplicacions per a dispositius electrònics. I per a la realització d'aquest s'ha utilitzat Android Studio que ens ha permès escriure el codi i comprovar mitjançant un emulador el funcionament.[11]

Primerament, les diferents parts d'una aplicació necessiten informació que és descàrrega de l'arxiu de dependències pubspec.yaml on afegim el paquet de webview.

La idea darrere de l'aplicació conté dues pantalles diferenciades. La principal, que conté un botó amb la connexió a Grafana.

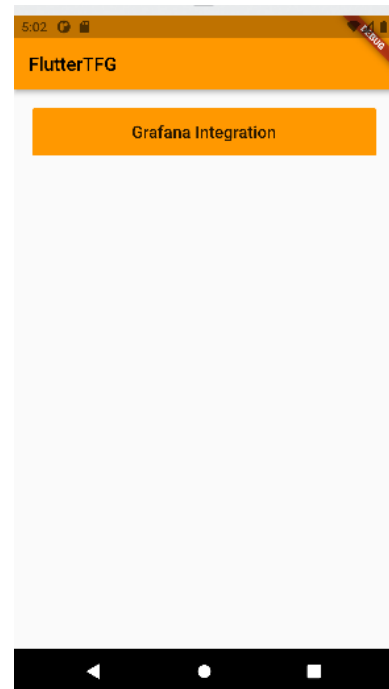


Fig. 16: Pantalla 1

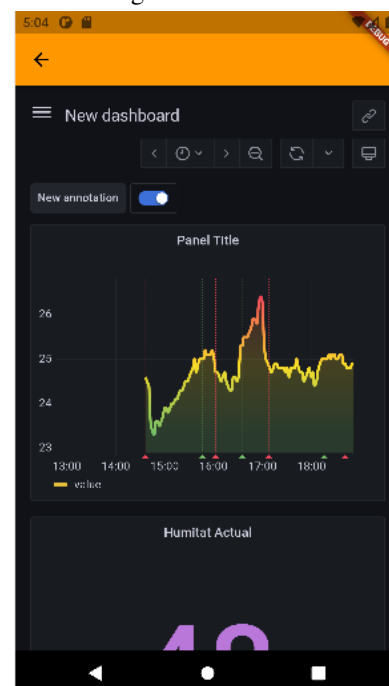


Fig. 17: Pantalla 2

Una vegada s'accedeix a l'apartat observem la visualització del corresponent panell de Grafana. Això s'aconsegueix creant una snapshot interactiva i creant un contenidor per a visualitzar l'enllaç que li enviem.

11 ENCAPSULAMENT DEL DISPOSITIU

Durant la realització de l'experimentació dels diferents sensors s'ha trobat que els sensors MQ generen una quantitat de calor residual mentre són actius.

Aquest inconvenient afecta a la mesura de la temperatura del sensor DHT11 i, per tant, s'ha buscat una alternativa per a mantenir-los de manera separada.

Per tal de fer el prototip més segur i resistent s'ha creat un encapsulament resistent llest per imprimir en una impressora 3D gràcies al programa Tinkercad, una eina de modelatge 3D creada per Autodesk que disposa de formes i eines per a dissenyar diferents models 3D de manera gratuïta. Aquest model conté un Arduino Uno al seu interior com a substitut de la placa disposada i espai suficient al seu interior per a posar el regulador de càrrega i les bateries, per mostrar això s'ha disposat d'un contenidor de piles amb botó on/off. El disseny es pot importar en formats .obj, .stl o .glb o fins i tot enviar directament a una impressora per a la seva impressió 3D o en format .svg si es vol crear el model utilitzant tall làser.

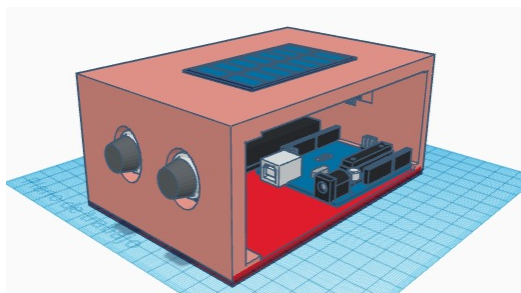


Fig. 18: Disseny 3D del prototip

12 LÍNIES FUTURES

Es vol optimitzar l'aplicació flutter mitjançant un sistema de control que permeti enviar opcions al node dispositiu mitjançant un missatge amb MQTT, aquest factor no s'ha contemplat gràcies al fet que el sistema de deep-sleep del dispositiu no rebria l'ordre fins que es despertés cada cinc minuts i s'ha volgut prioritzar el rendiment energètic a la comunicació entre usuari i node.

El node amb el qual s'ha treballat es pot connectar a una bateria i un panell solar mitjançant un regulador de càrrega per a panells solars. Es vol ampliar la xarxa de nodes per a monitorar més d'un espai, ja que disposem d'un broker que permet fins a 100 dispositius que es connectin a la vegada per transmetre dades.

També es vol millorar el sistema de deep-sleep guardant al RTC els valors de la lectura anterior i connectant-se a internet en el cas que hi hagi un canvi rellevant en les diferents lectures i, per tant, reduint el consum energètic d'això.

13 CONCLUSIONS

Aquest projecte ha demostrat les capacitats que pot tenir una placa de baix cost i la rellevància que pot tenir en un entorn SMART. S'ha treballat utilitzant eines gratuïtes i de codi obert que permeten que el cost real del producte consisteixi únicament en el mateix hardware que consisteix en la placa i els seus sensors.

Elaborar un projecte end-to-end m'ha donat la capacitat d'analitzar i veure el procés del projecte d'inici a final, podent fer canvis segons el desenvolupament i ha permès al projecte adaptar-se a les dificultats produïdes durant les diferents fases.

Des de l'inici s'ha tingut molt clar cap a on es volia arribar i s'han complert tots els objectius que s'han proposat. Pel que fa a la metodologia ha estat molt útil per tal de planificar i completar totes les tasques sense endarrerir l'entrega final.

14 AGRAÏMENTS

Aquest projecte no hagués estat possible sense l'ajuda del meu tutor Raúl Aragonés que m'ha ajudat durant totes les fases d'aquest treball i a la meua família pels ànims i l'interès que han mostrat durant el desenvolupament d'aquest.

REFERÈNCIES

- [1] Saeed, N., Al Naffouri, T. Y., and Alouini, M. S. (2020). Around the world of IoT/Climate monitoring using internet of X things . IEEE Internet of Things ne, 3(2), 82-83.
- [2] AZ-Delivery. (s. f.). Placa de desarrollo ESP32 D1 R32 con CH340G y WiFi + Bluetooth (placa de desarrollo de Internet). Recuperado 7 de octubre de 2021, de <https://www.az-delivery.de/es/products/esp32-d1-r32-board>
- [3] L. (2021, 8 junio). MQTT Explorer, un cliente genérico para MQTT. Luis Llamas. <https://www.luisllamas.es/mqtt-explorer-un-cliente-generico-para-mqtt/>
- [4] Savage, R. (2019, 24 enero). Introducing the Next-Generation InfluxDB 2.0 Platform. InfluxData. <https://www.influxdata.com/blog/introducing-the-next-generation-influxdb-2-0-platform/>
- [5] C. (2021a, mayo 7). Guide to Reduce the ESP32 Power Consumption by 95. DIYIoT. <https://diyi0t.com/reduce-the-esp32-power-consumption/>
- [6] 2018 sketch for connecting esp32 to Eduroam network. Worked under PEAP + MsCHAPv2 or EAP-TTLS under WPA/WPA2 Enterprise networks. (s. f.). Gist. Recuperado 11 de enero

de 2022, de <https://gist.github.com/martinius96/8579d9a5e7f9ab367a6f6b88359520bd>

- [7] Hillar, G. C. (2017). MQTT Essentials-A lightweight IoT protocol. Packt Publishing Ltd.
- [8] designthemes. (s. f.). Sensores de gas serie MQ. Prometec. <https://www.prometec.net/sensores-de-gas-serie-mq/>
- [9] Andalucía Conectada. (2021, 17 junio). Estado de arte y tendencias internet de las cosas e internet de todo. Observatorio - Andalucía Conectada. <https://observatorio.andaluciaconectada.es/publicaciones/estado-de-arte-y-tendencias-internet-de-las-cosas-e-internet-de-todo/>
- [10] Exemple de connexió a un portal captiu mitjançant una placa ESP8266 <https://yoursunny.com/t/2017/ESP8266-captive-login/>
- [11] Google. (s. f.). Flutter - Build apps for any screen. Flutter. <https://flutter.dev/>