

Pol Culleré Farré

**PLATAFORMA MÒBIL CONNECTADA MULTISENSOR
PERSONALITZABLE**

TREBALL DE FI DE GRAU

dirigit pel Sr.Iker Bilbao Aresté

Grau en Enginyeria de Sistemes i Serveis de Telecomunicacions



UNIVERSITAT ROVIRA I VIRGILI

**Tarragona
2024**

AGRAÏMENTS

Vull dedicar aquest espai del treball per tal donar les gràcies al tutor d'aquest Treball de Fi de Grau, l'Iker Bilbao Aresté, que tot i ser el seu primer Treball de Fi de Grau ha mostrat un altíssim grau d'implicació i d'assessorament durant tot el desenvolupament del projecte.

RESUM DEL PROJECTE

(Català) Aquest Treball de Fi de Grau es centra en el desenvolupament d'una plataforma mòbil connectada i multisensor per tal de dur a terme una monitorització ambiental utilitzant la tecnologia LoRaWAN. En un inici, es presenta la contextualització de la IoT i la tecnologia LoRaWAN, explicant la seva importància i justificant la seva selecció per aquest projecte. L'objectiu principal és desenvolupar una plataforma capaç de recollir dades ambientals en temps real, processar i visualitzar aquestes dades per millorar la gestió ambiental. Posteriorment, es detalla un marc teòric on s'inclou l'estudi detallat de la tecnologia LoRaWAN. També es parlarà dels diferents tipus de sensors que s'utilitzaran, mostrant les seves aplicacions i els paràmetres ambientals que poden mesurar. Tot seguit, en la fase de disseny de la plataforma, es defineix l'arquitectura general del sistema i es seleccionen els sensors més adients per al projecte. La plataforma està integrada amb *The Things Network* i es configura l'entorn de programació necessari per al seu funcionament. Durant la fase d'integració i configuració del flux de treball, els nodes MQTT i InfluxDB es configuren a Node-RED per permetre la recollida i l'emmagatzematge de dades. La visualització de dades es fa mitjançant Grafana, una eina de creació de taulers personalitzats que permet l'anàlisi de dades en temps real. El projecte conclou analitzant els resultats obtinguts, identificant les limitacions del sistema i recomanant possibles millores així com treballs futurs per optimitzar i ampliar el sistema de seguiment ambiental.

(English) This Final Degree Project focuses on the development of a connected and multi-sensor mobile platform to carry out environmental monitoring using LoRaWAN technology. Initially, the contextualization of IoT and LoRaWAN technology is presented, explaining its importance and justifying its selection for this project. The main objective is to develop a platform capable of collecting environmental data in real-time, processing, and visualizing this data to improve environmental management. Subsequently, a theoretical framework is detailed, which includes a detailed study of LoRaWAN technology. Different types of sensors that will be used are also discussed, showing their applications and the environmental parameters they can measure. Next, in the platform design phase, the overall system architecture is defined and the most suitable sensors for the project are selected. The platform is integrated with The Things Network and the necessary programming environment for its operation is configured. During the integration and workflow configuration phase, MQTT and InfluxDB nodes are configured in Node-RED to allow data collection and storage. Data visualization is done using Grafana, a custom dashboard creation tool that allows real-time data analysis. The project concludes by analysing the obtained results, identifying the system's limitations, and recommending possible improvements as well as future work to optimize and expand the environmental monitoring system.

ÍNDEX

1. INTRODUCCIÓ	1
1.1 Contextualització de la IoT i LoRaWAN.	1
1.2 Justificació de la selecció de la tecnologia LoRaWAN.	2
1.3 Objectius del projecte.	4
2. MARC TEÒRIC	5
2.1 Estudi de la tecnologia LoRaWAN.	5
2.1.1 És possible usar LoRa sense LoRaWAN?	5
2.1.2 Tipus d'antenes usades en LoRaWAN.	6
2.1.3 Funcionament de la tecnologia LoRaWAN.	7
2.1.4 Arquitectura d'una xarxa LoRaWAN.	9
2.1.5 Característiques principals.	10
2.1.6 Limitacions de la tecnologia LoRaWAN.	11
2.1.7 Estructura d'un paquet LoRaWAN.	12
2.2 Gateway.	14
2.2.1 Característiques principals.	14
2.2.2 Gateways comercials.	15
2.3 Network Server.	18
2.3.1 Metadata disponible en el Network Server.	19
2.4 Join Server	21
2.5 Regulacions regionals.	22
2.5.1 Principals Regions i les seves especificacions.	22
2.5.2 Subbandes i canals.	23
2.6 Data Rate.	24
2.7 Cicle de treball.	25

2.8 Mecanismes d'activació.	26
<i>2.8.1 Procés d'activació (OTAA).</i>	<i>27</i>
2.9 Característiques avançades.	28
<i>2.9.1 FUOTA.</i>	<i>28</i>
<i>2.9.2 Roaming.</i>	<i>30</i>
<i>2.9.3 Geolocalització.</i>	<i>31</i>
<i>2.9.4 LoRaWAN Relay.</i>	<i>32</i>
2.10 Revisió de sensors (GPS, CO2, soroll, gasos, etc.).	33
<i>2.10.1 Tipus de sensors i les seves aplicacions.</i>	<i>33</i>
<i>2.10.2 Gasos i paràmetres ambientals mesurables.</i>	<i>35</i>
2.11 Monitorització de la qualitat ambiental.	35
<i>2.11.1 Avenços en sistemes intel·ligents de monitorització del entorn mitjançant IoT i sensors.</i>	<i>36</i>
<i>2.11.2 Anàlisi sobre els sistemes de Monitorització Intel·ligent del Medi Ambient (SEM).</i>	<i>38</i>
2.12 Importància de la recollida de dades en temps real i aplicacions.	40
<i>2.12.1 Per què és important recolzar-se en el Machine Learning?</i>	<i>40</i>
<i>2.12.2 Beneficis de les dades en temps real.</i>	<i>40</i>
<i>2.12.3 Per què les empreses implementen aquesta tècnica ?</i>	<i>41</i>
3. DISSENY DE LA PLATAFORMA	43
3.1 Arquitectura general del sistema.	43
3.2 Selecció i justificació dels sensors: Temperatura, Pressió, CO2, TVOC.	43
3.3 Integració dels dispositius amb la xarxa TTN.	45
<i>3.3.1 Donar d'alta un sensor a la xarxa TTN.</i>	<i>45</i>
<i>3.3.2 Donar d'alta un gateway a la xarxa TTN.</i>	<i>49</i>
3.4 Configuració de l'IDE d'Arduino.	52
4. CONFIGURACIÓ DEL SERVIDOR AWS	55

4.1 Creació d'una instància EC2 a AWS.....	55
<i>4.1.1 Selecció del tipus de màquina.</i>	<i>55</i>
<i>4.1.2 Configuració de les claus d'accés.</i>	<i>56</i>
<i>4.1.3 Configuració de seguretat (ports oberts).</i>	<i>57</i>
4.2 Connexió a la màquina virtual amb PuTTY.....	57
<i>4.2.1 Conversió de la clau PEM a PPK.....</i>	<i>57</i>
<i>4.2.2 Connexió via SSH.</i>	<i>58</i>
5. INSTAL·LACIÓ DE NODE-RED, INFLUXDB I GRAFANA A LA MÀQUINA VIRTUAL.....	62
5.1 Instal·lació de Node-RED.	62
<i>5.1.1 Configuració inicial i posada en marxa.</i>	<i>63</i>
5.2 Instal·lació d'InfluxDB.	64
<i>5.2.1 Configuració de la base de dades.</i>	<i>65</i>
<i>5.2.2 Creació d'usuaris.....</i>	<i>66</i>
5.3 Instal·lació de Grafana.	67
<i>5.3.1 Configuració inicial i posada en marxa.</i>	<i>68</i>
6. INTEGRACIÓ I CONFIGURACIÓ DEL FLUX DE TREBALL A NODE-RED.....	70
6.1 Configuració dels nodes MQTT a Node-RED.....	70
<i>6.1.1 Subscripció als temes MQTT.</i>	<i>71</i>
<i>6.1.2 Comprovació del missatge rebut.....</i>	<i>73</i>
6.2 Configuració dels nodes InfluxDB a Node-RED.....	74
<i>6.2.1 Preparació de dades per a la inserció a la base de dades.....</i>	<i>74</i>
<i>6.2.2 Inserció de dades a InfluxDB.....</i>	<i>75</i>
6.3 Configuració dels nodes per enviar dades a Telegram.	77
<i>6.3.1 Preparació dels nodes per connectar-se a Telegram.</i>	<i>77</i>

6.3.2 Creació del Bot, Configuració del node Telegram Sender i Resultat Final.....	79
7. VISUALITZACIÓ I ANÀLISI DE DADES AMB GRAFANA.....	81
7.1 Creació de dahsboards a Grafana.	81
7.2 Alertes i notificacions a Grafana.	83
8. PROVES I RESULTATS.....	88
8.1 Preparació i muntatge.....	88
8.2 Resultats obtinguts.....	89
8.3 Autonomia de la plataforma mòbil multisensor.	96
9. CONCLUSIONS I LÍNIES FUTURES DEL PROJECTE	97
9.1 Conclusions.....	97
9.2 Línies futures.....	97
10. REFERÈNCIES BIBLIOGRÀFIQUES	99
11. ANNEXOS.....	101
11.1 Estructura WBS i diagrama de Gantt.	101
11.2 Codis Arduino.....	102
11.2.1 Programa sensor HW-611.....	102
11.2.2 Programa sensor CCS811.....	105
11.2.3 Programa principal amb els sensors unificats.....	109

1. INTRODUCCIÓ

L'Internet de les Coses (IoT¹) ha emergit com una força transformadora en el panorama tecnològic actual. Aquesta tecnologia està reconfigurant la manera en què interactuem amb el món, des de la manufactura fins a la salut personal. Aquest treball de final de grau té com a objectiu contribuir a aquest camp en ràpida evolució mitjançant el desenvolupament d'una Plataforma Mòbil Connectada Multisensor Personalitzable. Aquesta solució permet integrar diversos sensors i recollir dades en temps real, oferint una solució adaptable i de gran abast.

En les properes pàgines, presentaré els objectius específics del projecte, incloent el disseny, la implementació i la validació de la plataforma. Així mateix, detallaré la justificació del perquè l'elecció de LoRaWAN és no només estratègica sinó també necessària en el context actual, posant èmfasi en les seves capacitats de xarxa de baixa potència i ampli abast (LPWAN) que permeten una nova onada de connectivitat IoT.

1.1 Contextualització de la IoT i LoRaWAN.

La Internet de les Coses (IoT) representa una evolució significativa en la manera com interactuem amb el món digital. Com a concepte, l'IoT es refereix a la interconnexió de dispositius físics, vehicles, electrodomèstics i altres elements incorporats amb sensors, software, actuadors i connectivitat que els permet recopilar i intercanviar dades. La IoT té l'objectiu d'integrar el món físic amb sistemes d'informació per millorar l'eficiència, la precisió i el benefici econòmic, a més de reduir la intervenció humana.

La utilització de l'IoT es fa present en múltiples sectors, incloent la manufactura intel·ligent (Indústria 4.0), la domòtica, la salut (e-salut), l'agricultura intel·ligent (agritech), ciutats intel·ligents (smart cities) i molt més. Aquests dispositius "intel·ligents" recullen dades del seu entorn, que poden ser processades localment o enviades a una plataforma en el núvol per a ser analitzades i utilitzades per a una gran varietat d'aplicacions.

Tanmateix, un dels reptes més importants en la implementació d'IoT és la connectivitat, especialment quan els dispositius han de funcionar durant llargs períodes de temps, de vegades en llocs remots o de difícil accés on les connexions Wi-Fi o mòbils tradicionals com 3G o LTE poden ser poc pràctiques. o massa car.

Segons les previsions actuals, el volum de negoci de l'IoT mundial arribarà a 1.1 trilions de dòlars el 2025. Actualment, el nombre de dispositius IoT connectats a nivell mundial supera els 14.000 milions i s'espera que aquesta xifra augmenti fins a més de 30.000 milions el 2030. Aquest creixement és impulsat per la demanda creixent de solucions intel·ligents en diversos sectors, així com per la millora de les infraestructures de connectivitat com LoRaWAN.

¹ Internet of Things.

LoRaWAN (Long Range Wide Area Network) és un protocol de comunicacions i una arquitectura de xarxa per a xarxes cel·lulars de baix consum i llarg abast (LPWAN) dissenyades per a comunicacions IoT. Utilitza una tecnologia de modulació anomenada LoRa (Long Range) per a la transmissió de dades, la qual és propietat de SemTech², fet que implica que els detalls específics d'aquesta modulació no són accessibles públicament. No obstant això, la resta de l'ecosistema LoRaWAN és obert, permetent que desenvolupadors i enginyers puguin crear solucions i aplicacions sense restriccions significatives. Aquesta modulació permet comunicar-se a distàncies molt més llargues que les xarxes Wi-Fi, amb un abast de més de 10 quilòmetres en zones rurals de baixa densitat.

Un dels principals avantatges de LoRaWAN és el seu consum d'energia extremadament baix, que permet que els dispositius funcionin amb bateries durant molts anys. Això el fa ideal per a aplicacions on els dispositius necessiten enviar petites quantitats de dades de manera regular, com ara sensors de monitorització ambiental o sistemes de seguiment d'actius.

LoRaWAN també ofereix funcions clau com ara el xifratge d'extrem a extrem, la capacitat d'operar en diverses bandes de freqüència i l'adaptabilitat de la velocitat de dades per aconseguir un equilibri entre el consum d'energia, l'abast i el rendiment de càrrega de dades.

A causa d'aquestes característiques, LoRaWAN és una solució òptima per a molts casos d'ús d'IoT que requereixen connectivitat a gran escala i de baix cost. A més, l'existència de xarxes LoRaWAN públiques com The Things Network redueix encara més les barreres per a l'adopció d'IoT permetent als usuaris connectar els seus dispositius sense desplegar la seva pròpia infraestructura de xarxa. Aquesta xarxa està basada en una infraestructura comunitària on qualsevol persona pot contribuir afegint les seves pròpies passarel·les. Això significa que qualsevol usuari pot ampliar la cobertura de la xarxa instal·lant i configurant una passarel·la TTN, permetent així una major accessibilitat i cobertura a nivell global.

En resum, IoT i LoRaWAN es complementen, IoT aporta un gran nombre de dispositius al món digital i LoRaWAN proporciona la columna vertebral de connectivitat que permet que aquests dispositius funcionin de manera eficient i còmoda. A mesura que avancem cap a un futur més connectat, IoT i LoRaWAN juguen un paper clau en la transformació digital de la societat, alliberant el potencial sense aprofitar mitjançant la innovació i la connectivitat.

1.2 Justificació de la selecció de la tecnologia LoRaWAN.

En aquest treball de fi de grau, l'elecció de LoRaWAN com a tecnologia principal per al desenvolupament de plataformes IoT mòbils i multisensor es basa en una sèrie de consideracions tècniques i estratègiques que la fan únicament atractiva en comparació amb altres tecnologies sense fil del mercat, com Wi-Fi, Bluetooth, Sigfox, Wize i NB-IoT. Analitzarem els motius d'aquesta elecció des de diferents perspectives tècniques i empresarials.

² És una empresa líder en el disseny i subministrament de semiconductors analògics i de senyal mixta.

Des d'una perspectiva d'eficiència energètica i autonomia, la tecnologia LoRaWAN és coneguda per la seva eficiència energètica superior. El protocol està optimitzat per a comunicacions de baix consum, la qual cosa permet que els dispositius durin mesos o fins i tot anys amb una sola bateria. Això supera amb escreix les capacitats del Wi-Fi o Bluetooth, que, tot i que són ideals per a transferències de dades d'alta velocitat i baixa latència en entorns locals, es troben limitats en termes d'eficiència energètica, requerint una recàrrega freqüent que no seria viable en aplicacions IoT disperses o de difícil accés. No obstant això, com que la majoria dels dispositius IoT funcionen amb piles, això genera un volum significatiu de deixalles electròniques futures. A banda de mirar que les piles durin el màxim possible, es pot considerar l'ús de tècniques *d'Energy Scavenging*, que consisteixen en recuperar part de l'energia que generem i es dispersa. Això pot incloure energia del propi cos, del sistema que monitoritza, o fonts renovables.

Per altra banda, pel que fa al rang de cobertura si ho comparem amb tecnologies LAN com el Wi-Fi i Bluetooth, LoRaWAN ofereix un rang molt més ampli, que pot arribar a diversos quilòmetres en zones obertes, i és capaç de proporcionar connectivitat en entorns urbans complexos amb obstacles significatius. Això obre la porta a una gran varietat d'aplicacions on els dispositius necessiten ser monitoritzats a llarga distància, des de la gestió de l'agricultura intel·ligent fins a la infraestructura urbana connectada.

A més a més, la tecnologia LoRaWAN, amb la seva capacitat per penetrar en edificis i altres estructures urbanes, supera a alternatives com el Wi-Fi i el Bluetooth, que sovint es veuen limitades per la seva curta distància de transmissió i la pèrdua de senyal a través d'obstacles. Això és especialment crític en entorns urbans densos on els senyals poden ser interromputs per múltiples fonts d'interferències. La robustesa de la senyal LoRaWAN assegura una connectivitat fiable que és fonamental per a aplicacions crítiques de IoT que requereixen una comunicació consistent i fiable. [\[1\]](#)

Mentre que tecnologies com Sigfox són també de baix consum i ofereixen una cobertura àmplia, LoRaWAN aporta l'avantatge addicional de permetre una xarxa mesurable i controlada per l'usuari final. Les xarxes LoRaWAN poden ser desplegades de forma privada, donant a les empreses i a les institucions la capacitat de construir i gestionar les seves pròpies xarxes de sensors sense dependre de tercers. Això és significativament avantatjós respecte a les xarxes cel·lulars, com LTE o NB-IoT, que depenen d'operadors de xarxa i poden implicar costos recurrents més alts.

Finalment, un dels factors més crítics en la selecció de LoRaWAN és el seu model de costos favorable. El desplegament de la tecnologia LoRaWAN pot ser extremadament cost-efectiu en comparació amb les alternatives cel·lulars com LTE o NB-IoT, ja que no requereix l'ús de freqüències llicenciades ni el pagament de tarifes de subscripció mensuals als operadors de telecomunicacions. Aquest baix cost d'entrada i la falta de costos de funcionament recurrents fan de LoRaWAN una opció atractiva per a *startups* i investigadors que busquen implementar solucions IoT amb pressupostos ajustats.

Considerant tots aquests punts, LoRaWAN emergeix com la tecnologia més alineada amb els objectius del projecte, que requereix una plataforma IoT escalable, de baix consum, amb una àmplia cobertura i amb l'autonomia suficient per suportar un desplegament efectiu de sensors mòbils en un entorn urbà complex. Així doncs, l'elecció de LoRaWAN representa una decisió estratègica, orientada a maximitzar la funcionalitat, la viabilitat i la sostenibilitat del projecte de recerca en qüestió.

	WiFi	LoRa	Sigfox	Wize	NB-IoT
Eficiència energètica	Baixa	Alta	Alta	Alta	Mitja/Alta
Rang	Curt	Llarg	Llarg	Llarg	Mitjà
Encriptació	AES256	AES128 (NW/APP)	AES128	AES128/Custom	Custom
Ample de banda	50-1000Mbps	250-11000bps	10-1000bps	2400-9600kbps	20-250kbps
Desplegament	Ubicu	Dispers	Ciutats i vies	Algunes ciutats	Pilots
Modulació	CCK / PSK / QAM	Wide band CSS (chip lock-in)	Narrow band BPSK	Narrow band FSK	DMA narrow band
Banda	2.4 – 5 GHz	868 MHz	868 MHz	169 MHz	700 MHz – 2.2 GHz
Model de negoci	Xarxa privada	Xarxa privada	Xarxa operada	Xarxa privada	Car
Cost de la infraestructura	< 100€	200-1000€	?	~ 2000€	Car

Taula 1.1. Comparativa entre diferents tecnologies de comunicació per a la IoT.

1.3 Objectius del projecte.

Els principals objectius que es pretenen assolir amb la realització d'aquest treball són els següents:

- Desenvolupament i implementació d'una solució de monitorització ambiental de baix cost, replicable i sense cost de llicència, utilitzant recursos de programari lliure.
- Realització d'una investigació exhaustiva sobre la tecnologia LoRaWAN, incloent els seus principis de funcionament, característiques, avantatges...
- Realització d'una sèrie de proves pilot per tal de demostrar l'escalabilitat i la fiabilitat de la plataforma mòbil en diferents escenaris d'ús real, per validar la seva aplicabilitat en el camp de l'IoT.

2. MARC TEÒRIC

En aquest apartat s'exposaran els diversos aspectes clau de la tecnologia LoRaWAN i la seva aplicació en el món de l'IoT. Cada un dels diferents punts, serà detallat amb exemples pràctics i estudis de cas per tal de demostrar la viabilitat i els beneficis d'ús de LoRaWAN en aplicacions de monitorització ambiental i gestió de dades en temps real.

2.1 Estudi de la tecnologia LoRaWAN.

LoRaWAN (Long Range Wide Area Network) és una tecnologia de xarxa de àrea àmplia i de baix consum (LPWAN) dissenyada per a la comunicació de dispositius IoT (Internet of Things) a grans distàncies. Utilitza la modulació LoRa (Long Range) per a la transmissió de dades, que permet un abast molt més ampli que les xarxes Wi-Fi o Bluetooth, amb un abast que pot superar els 10 km en zones rurals o de baixa densitat.

2.1.1 *És possible usar LoRa sense LoRaWAN?*

Aquesta és una pregunta molt comú en el món de les tecnologies sense fil. La resposta és sí, ja que els desenvolupadors poden implementar els seus propis protocols de comunicació personalitzats sobre la capa física LoRa, adequats per les seves necessitats específiques d'aplicació. Alguns escenaris on això podria arribar a ser útil podrien ser:

- **Comunicació punt a punt:** En aplicacions simples on solament es necessita comunicació directa entre dos dispositius, com un sensor remot i una estació base, es pot utilitzar LoRa per transmetre dades directament sense la infraestructura de xarxa LoRaWAN.
- **Xarxes tancades:** En situacions on una xarxa tancada amb un número limitat de dispositius necessita comunicar-se dins d'una àrea definida, es pot establir un protocol personalitzat sobre LoRa per manejar la comunicació i el control d'accés.
- **Aplicacions de baixa complexitat:** Per a projectes que no requereixen l'escalabilitat, la gestió de xarxa o les característiques de seguretat LoRaWAN, com certs projectes de hobby o educatius, LoRa pot ser una solució més senzilla i directa.

Malgrat que en certs escenaris existeixen avantatges en el fet d'usar LoRa sense l'estàndard LoRaWAN damunt, això ens limita en la majoria dels casos, ja que al prescindir de LoRaWAN es perden els beneficis propis d'aquest, com per exemple la interoperabilitat entre dispositius i xarxes, les funcions avançades de seguretat i la gestió de la xarxa. A més a més, un desenvolupament d'un protocol personalitzat podria requerir més temps i recursos.

2.1.2 Tipus d'antenes usades en LoRaWAN.

En el context de LoRaWAN, les antenes desenvolupen un paper fonamental en l'habilitació de comunicacions sense fils de llarga distància i baix consum energètic. És un component clau i extremadament important, que tendeix a estar en segon pla. Els tipus d'antenes més importants usades en LoRaWAN son:

- **Antenes Dipol:** Àmpliament usades en dispositius finals degut a la seva simplicitat, eficiència i mesura compacta. Consisteixen en dos elements conductors d'igual longitud, orientats en la línia amb el transmissor. Son un tipus d'antena omnidireccional i ofereixen un compromís raonable entre rendiment i mida. A part, son fàcils de fabricar i el seu funcionament es pot entendre de manera relativament fàcil, per tant, s'utilitzen també amb fins didàctics. Per una banda de freqüència de 868 MHz, una antena dipol tindria una longitud d'uns 17,25 cm.



Figura 2.1. Antena Dipol.

- **Antenes Monopol:** Les antenes dipol consisteixen en un únic element conductor vertical muntat sobre algun tipus de pla de terra, que podria ser una superfície metàl·lica o inclús el propi xassís del dispositiu. Son essencialment una versió de mitja longitud d'un dipol, amb el pla de terra actuant com el mirall de l'element faltant. Son comuns en aplicacions on l'antena necessita ser compacta i menys propensa a danys o interferències, com en dispositius IoT muntats en exteriors. Per una banda de freqüència de 868 MHz, una antena monopol tindria una longitud d'uns 8.625cm.



Figura 2.2. Antena Monopol.

- **Antena Helicoidal:** Son les anomenades antenes hèlix, les quals son especialment populars en aplicacions on es requereix polarització circular, com per exemple en comunicacions per satèl·lit, radioastronomia o bé LoRaWAN actualment. Degut al seu disseny relativament simple i la seva còmoda mida, actualment les podem trobar en diverses plaques amb suport per comunicacions LoRa. Per una banda de freqüència de 868 MHz, una antena helicoidal pot tenir una longitud de 8 a 10 cm, depenent de la configuració específica de la bobina.



Figura 2.5. Petita antena Helix en una placa Heltec Lora 32.

2.1.3 Funcionament de la tecnologia LoRaWAN.

Una xarxa desenvolupada amb aquesta tecnologia té com a objectiu principal connectar dispositius entre sí. En LoRaWAN, els nodes (dispositius finals) poden operar en una de les 3 classes diferents: Classe A, Classe B i Classe C. Cada classe defineix el comportament de comunicació del dispositiu, especialment en termes de com i quan poden rebre missatges del servidor (*downlink*³). Aquestes classes permeten diferents compromisos entre la complexitat del dispositiu, el consum d'energia i les necessitats de l'aplicació.

Classe A (Dispositius Bidireccionals Bàsics).

Tots els dispositius LoRaWAN han de suportar la Classe A. En aquesta classe, la comunicació s'inicia sempre pel dispositiu final amb un missatge de *uplink*⁴. Després de cada missatge de *uplink*, hi ha dos breus finestres de temps on el dispositiu escolta els missatges de *downlink*. Immediatament després de transmetre, el dispositiu obre dues finestres de recepció (Rx1 i Rx2). Si no rep cap missatge durant aquestes finestres entra en mode de baix consum fins la pròxima transmissió. La Classe A és la més eficient en termes d'energia, ja que el dispositiu està majoritàriament en mode de baix consum i solament es desperta per transmetre o durant les finestres de recepció.

³ Comunicació que es produeix des d'una estació base o servidor central cap a un dispositiu final a la xarxa.

⁴ Comunicació que es produeix des d'un dispositiu final a la xarxa cap a una estació base o servidor central.

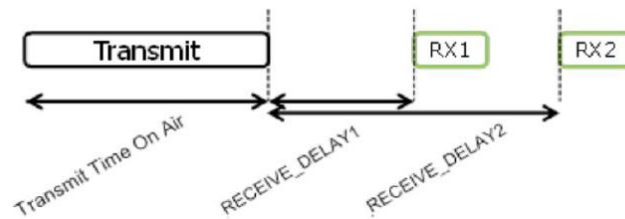


Figura 2.6. Finestra de RX a la Classe A.

Classe B (Dispositius amb Finestres de Recepció Programada).

Els dispositius Classe B inclouen tot el comportament de la Classe A però afegixen finestres de recepció programades a més de les finestres post-transmissió. Això permet al servidor de xarxa saber quan estarà el dispositiu escoltant activament, el que facilita la planificació de les comunicacions *downlink*. Per sincronitzar aquestes finestres de recepció programades, les portes d'enllaç emeten *beacons*⁵ que els dispositius Classe B utilitzen per sincronitzar els seus rellotges interns i saber quan obrir les seves finestres de recepció addicionals. La Classe B proporciona una major oportunitat per a la comunicació de *downlink* a costa d'un consum energètic lleugerament major, degut a que està periòdicament escoltant els *beacons* i les finestres de recepció programades.

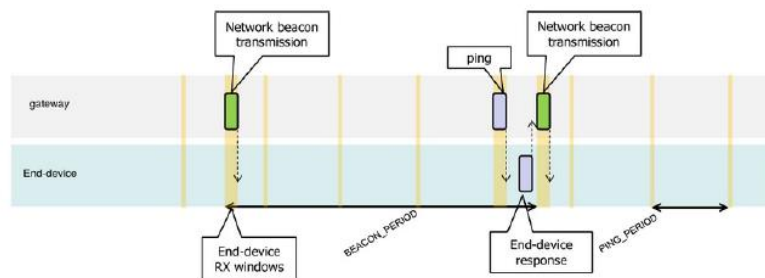


Figura 2.7. Finestra de RX a la Classe B.

Classe C (Dispositius de Recepció Contínua).

Els dispositius de Classe C tenen finestres de recepció obertes de manera quasi contínua, tancant-se solament quan el dispositiu està transmetent. Això significa que poden rebre missatges de *downlink* en qualsevol moment, sense necessitat d'esperar una transmissió d'*uplink*. Aquesta classe és la menys eficient en termes d'energia, ja que el dispositiu necessita mantindre el seu receptor actiu quasi tot el temps. Per tant, és més adequat per dispositius que no depenen de bateries o per aplicacions que requereixen una alta disponibilitat de comunicació *downlink*.

⁵ Actuen com senyals de referència que ajuden a mantenir la sincronització temporal necessària per a la comunicació eficient entre els dispositius i les portes d'enllaç.

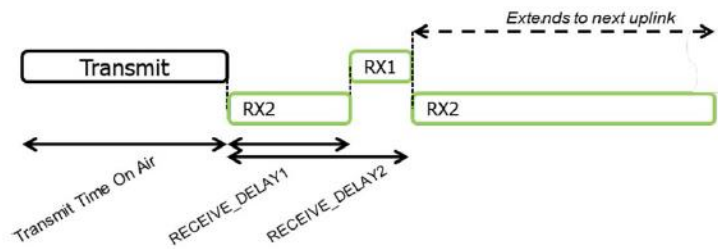


Figura 2.8. Finestra de RX a la Classe C.

La tria entre Classe A, B o C depèn dels requisits específics de l'aplicació i del balanç desitjat entre la eficiència energètica i la necessitat de comunicacions de *downlink*.

2.1.4 Arquitectura d'una xarxa LoRaWAN.

L'arquitectura de LoRaWAN es basa en una topologia en estrella, on diversos dispositius finals de baix consum es comuniquen amb passarel·les LoRaWAN. Aquestes passarel·les actuen com a ponts entre els dispositius finals i un servidor central anomenat Network Server (NS). El NS s'encarrega de gestionar les comunicacions, emmagatzemar les dades i proporcionar una interfície per a l'aplicació final.

Els components principals d'una xarxa LoRaWAN són:

- **Nodes:** Son els sensors, actuadors o altres dispositius IoT que necessiten comunicar-se dins la xarxa. Aquests dispositius són normalment alimentats per bateries i dissenyats per ser de baix cost i consum energètic.
- **Portes d'Enllaç (Gateway):** Funcionen com ponts entre els dispositius finals i el servidor de xarxa. Reben les transmissions dels dispositius finals i les reenvien al servidor de xarxa a través de connexions de banda ampla com Ethernet, Wi-Fi o cel·lular. Una sola porta d'enllaç pot manejar milers de dispositius finals simultàniament.
- **Servidor de Xarxa (Network Server):** És el component central de la xarxa LoRaWAN, responsable de la gestió de la xarxa, incloent el control d'accés, l'enrutament de missatges, la programació de la comunicació descendent i la gestió de la taxa de dades adaptativa (ADR⁶).
- **Servidor d'Aplicació:** Processa totes les dades d'aplicació rebuts del servidor de xarxa. Aquí és on s'implementa la lògica específica de l'aplicació, com per exemple l'anàlisi de dades, la visualització i la presa de decisions basada en les dades rebudes dels dispositius finals.

⁶ Permet que el servidor TTN controli la velocitat de les dades i la potència de transmissió del dispositiu o node.

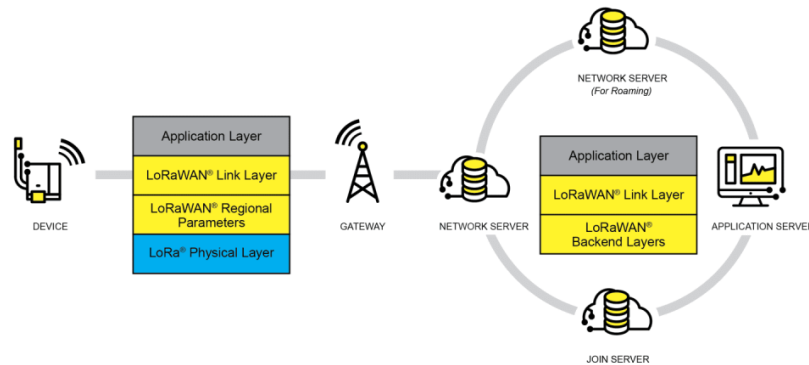


Figura 2.9. Arquitectura general d'una xarxa LoRaWAN.

2.1.5 Característiques principals.

Hi ha una sèrie de característiques que fan de LoRaWAN una tecnologia especial i fonamental. Actualment, la necessitat de crear xarxes IoT segures, fiables i de llarg abast és cada vegada més gran. Davant d'aquesta realitat, LoRaWAN es converteix en una solució molt adequada, tot gràcies a aquestes característiques:

- **Connexió de dispositius:** Un sol *gateway* permet connectar milers de dispositius, una xifra que va en augment cada cop que s'afegeix una nova porta. D'aquesta forma, es facilita la creació d'una xarxa barata, amplia i d'un consum energètic baix. Els aparells lligats enviaran les seves dades en temps real i durant llargs períodes. Per això, existeixen diferents formes de cobertura LoRaWAN: xarxes privades, xarxes de tercers i operadores de dispositius.
- **Baix consum d'energia:** LoRaWAN està optimitzat per a la comunicació de baix consum, permetent que els dispositius funcionin amb bateries durant anys o fins i tot dècades. Això el fa ideal per a aplicacions on els dispositius necessiten enviar petites quantitats de dades de manera periòdica, com els sensors de monitorització ambiental o els sistemes de seguiment d'actius.
- **Ampli abast:** Aquesta és una de les característiques més destacades per les implicacions que comporta. La tecnologia està dissenyada per funcionar en ciutats i a travessar tot tipus d'obstacles. LoRaWAN pot proporcionar connectivitat a distàncies superiors a 10 km en zones rurals o de baixa densitat, i fins a 3 km en entorns urbans complexos. Això el fa ideal per a aplicacions que necessiten monitoritzar dispositius dispersos en una àrea àmplia, com ara l'agricultura intel·ligent o la gestió de la ciutat intel·ligent.
- **Transmissió de dades baixa:** La velocitat de transmissió de dades a LoRaWAN és baixa, amb valors que oscil·len entre 0,3 i 50 Kbps. No obstant això, aquesta característica no suposa un inconvenient per a la majoria de les aplicacions previstes per a aquesta tecnologia.
- **Escalabilitat:** Els dispositius que es poden utilitzar són molt variats i depenen de l'objectiu de la xarxa. Ajuden a avançar-se a problemes incipients i prevenir-los de manera adequada. Alguns exemples són els sensors d'humitat, temperatura o de conductivitat elèctrica. Aquestes són les seves principals aplicacions:

- Smart cities.
- Smart industry.
- Agricultura.
- Logística.
- Utilities.
- Cadena de subministrament.

- **Cost baix:** El desplegament i la operació de xarxes LoRaWAN són relativament econòmics.
- **Seguretat:** La seguretat és primordial en el desplegament IoT i LoRaWAN defineix dues capes de criptografia. Les dades del sensor (*payload*) estan encriptades amb l'AppSkey (AES128). El missatge està signat amb el MIC (**codi d'integritat del missatge**), que es calcula amb el *payload*, el *devaddr*, el *fcnt* i fent servir la NwkSkey.

La xarxa no pot de cap manera saber què s'està enviant, només l'aplicació. TTN permet descodificar el missatge en el *backend*⁷, per tant es recomana fer servir un *handler*⁸ segur per connectar-se (MQTT sobre SSL). [2]

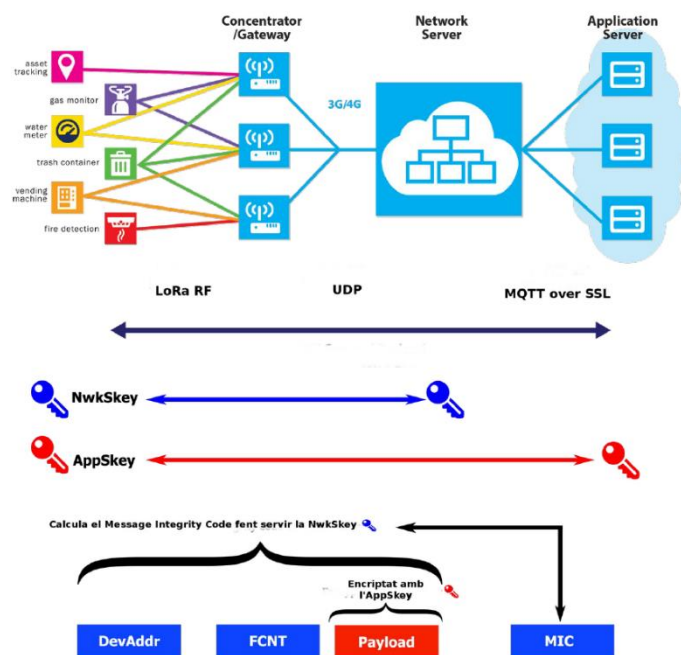


Figura 2.10. Seguretat en xarxes LoRaWAN.

2.1.6 Limitacions de la tecnologia LoRaWAN.

Si bé es cert que la tecnologia LoRaWAN presenta grans avantatges significatius, també cal tenir en compte algunes de les limitacions abans d'implementar-la en algun projecte o aplicació específica. Les principals limitacions d'aquesta tecnologia es basen en la velocitat

⁷ El backend de la aplicació administra la funcionalitat general de la aplicació web.

⁸ Component o una part del software que gestiona o processa determinades accions o dades.

de dades baixa, ja que té una velocitat de dades relativament baixa en comparació amb les altres tecnologies com el Wi-Fi o bé el Bluetooth. Per aquest motiu, no seria una tecnologia ideal per aplicacions que requereixen dades d'alta velocitat, com per exemple, la transmissió de vídeo o de música.

Per altra banda, també té una latència alta en comparació amb les altres tecnologies, el que no la fa ideal per aplicacions que requereixen resposta en temps real.

Finalment, la principal limitació seria la dependència de les passarel·les. La tecnologia LoRaWAN depèn en gran part de les passarel·les, que actuen com a ponts entre els dispositius finals i el servidor central. La ubicació estratègica i l'adequat manteniment de les diferents passarel·les són crucials per garantir un funcionament en condicions de la xarxa.

A més, els servidors de xarxa poden representar un altre punt de limitació important. La capacitat de processament i la disponibilitat del servidor central són crucials per a la gestió i el processament de grans volums de dades que arriben dels dispositius finals. Si el servidor de xarxa no és capaç de gestionar aquesta càrrega eficientment, pot provocar colls d'ampolla que afecten el rendiment general del sistema. Aquesta limitació pot ser especialment problemàtica en aplicacions que requereixen una alta disponibilitat i fiabilitat del servei.

2.1.7 Estructura d'un paquet LoRaWAN.

L'estructura d'un paquet LoRaWAN està dissenyada per optimitzar l'eficiència en entorns de baix ample de banda i potència limitada. Un paquet LoRaWAN consta de diverses parts, cada una amb un propòsit específic per garantir la comunicació segura, eficient i fiable entre els dispositius finals i la xarxa. A continuació, veurem les diferents parts d'un paquet LoRaWAN:

MAC PAYLOAD

El MAC Payload és on s'emmagatzema la informació útil del missatge. Està compost pels següents elements:

- **FHDR (Frame Header):** Aquest encapçalament inclou la informació important pel processament del missatge.
- **FPort:** Aquest camp indica el port d'aplicació al que es destina el missatge. Un valor de FPort de 0 està reservat pels comandaments MAC, mentre que els valors de 1 a 223 estan disponibles per l'ús d'aplicacions específiques. Els valors de 224 a 225 estan designats per usos futurs i per a extensions específics de la xarxa.
- **FRMPayloads:** Aquí es troba la càrrega útil del missatge, és a dir, les dades reals que el dispositiu està enviant o rebent. La naturalesa d'aquestes dades pot variar àmpliament segons l'aplicació, des de lectures de sensors fins a comandaments per actuadors.

FHDR	FPORT	FRMPayload
7-23 bytes	0-1 bytes	0-N bytes

Taula 2.1. ESsquema MAC Payload.

MAC HEADER (MHDR)

El MHDR és un camp d'1 byte que especifica el tipus de missatge (M-Type). Aquest camp és crucial per determinar com s'ha de processar la resta del paquet. El tipus de paquet especificat pot ser de diversos tipus:

Camp M-Type	Tipo de missatge
000	Join Request
001	Join Accept
010	Unconfirmed Data Up
011	Unconfirmed Data Down
100	Confirmed Data Up
101	Confirmed Data Down
110	RFU
111	Proprietary

Taula 2.2. Esquema MAC Header

PHY Payload

El *Payload* de capa física compta amb una capçalera MAC, el MAC Payload i un Message Integrity Code (MIC⁹), un codi de 4 bytes que es calcula a partir de la NwkSkey. Conté la informació que s'ha d'enviar dins d'una trama de la capa d'enllaç. La mida màxima d'aquesta ha de ser de 255 bytes, ja que la mida del *payload* queda definida en un únic byte a la capçalera de la trama.

MHDR	MACPayload	MIC
1 byte	1-M bytes	4 bytes

Taula 2.3. Esquema PHY Payload.

Frame Header (FHDR)

El FHDR conté informació essencial per a l'encaminament i la gestió dels missatges, els quals poden variar en mesura. Està compost pels següents elements:

-DevAddr: Un identificador de 4 bytes que denota el dispositiu final dins de la xarxa LoRaWAN.

-FCtrl: Un camp d'1 bytes que inclou diversos *flags*¹⁰ de control, com per exemple els indicadors de ACK i ADR.

-FCnt: Un comptador de 2 bytes que s'utilitza per portar el compte dels missatges enviats i rebuts per evitar la repetició i garantir la seqüència correcta dels missatges.

-FOpts: Un camp opcional que pot contenir comandaments MAC específics. [\[3\]](#)

⁹ El Message Integrity Code (MIC) és una clau per autenticar el missatge generat a partir dels camps MHDR y MACPayload.

¹⁰ Bits o petits conjunts de bits dins d'una unitat de dades que s'utilitzen per transmetre informació o indicar diferents estats i condicions.

DevAddr	FCtrl	FCnt	FOpts
4 bytes	1 byte	2 bytes	0-15 bytes

Taula 2.4. Esquema FHDR.

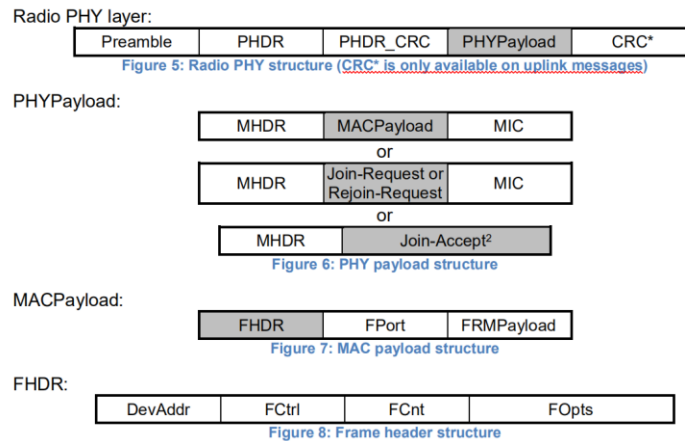


Figura 2.11. Dissecció d'un paquet LoRaWAN.

2.2 Gateway.

Anteriorment, ja hem observat que la funció principal d'un *gateway* és rebre senyals de radiofreqüència (RF) transmeses pels dispositius finals utilitzant la tecnologia LoRa i reenviar aquesta informació al *Network Server*, generalment a través d'una connexió a Internet.

2.2.1 Característiques principals.

Les principals característiques son:

- **Conversió de senyals:** Els *gateways* LoRaWAN converteixen les senyals RF de LoRa en paquets de dades que poden ser transmesos a través de xarxes IP al servidor de xarxa LoRaWAN, i viceversa pels missatges de *downlink* des del servidor de xarxa cap als dispositius finals.
- **Cobertura de llarg rang i baix consum:** Aprofitant la tecnologia LoRa, els *gateways* poden cobrir distàncies significatives, de fins a diversos kilòmetres en àrees urbanes i molt més en àrees rurals. És per aquest motiu, que els *gateways* implementen electrònica d'alta sensibilitat, podent arribar a tenir una sensibilitat major a senyals molt dèbils, de menys de -140 dB.
- **Capacitat de maneig de dispositius múltiples:** Un sol *gateway* LoRaWAN pot manejar comunicacions de mils de dispositius finals simultàniament, gràcies a la capacitat de LoRa d'utilitzar diferents factors d'eixamplament (*Spreading Factors*¹¹) i canals.
- **Funcions de reenviament de dades:** Els *gateways* no interpreten ni analitzen les

¹¹ La seva funció principal és ajustar la relació entre el rang i la taxa de transferència de dades.

dades transmises; simplement els reenvien al servidor de xarxa i entreguen els missatges de *downlink* als dispositius finals segons les instruccions del servidor.

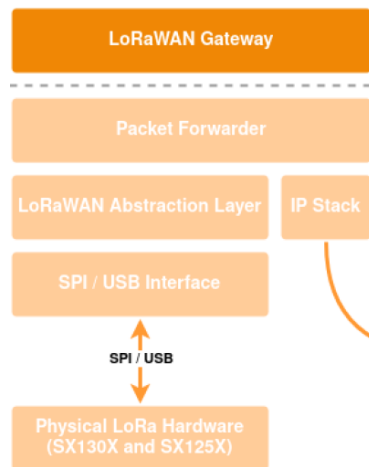


Figura 2.12. Components interns d'un gateway LoRaWAN.

Per complir amb l'objectiu de reenviar paquets des de la xarxa LoRa fins al *Network Server* el *gateway* necessita electrònica interna que implementi en *hardware* la demodulació i modulació LoRa, a més a més de comptar amb un *software* que s'encarregui de la lògica de reenviar aquests paquets a la troncal, que poden ser Ethernet, Wifi, Celular... i que al final del dia conduirà els paquets al *Network Server* a través d'internet. Aquest component es denomina "*Packet Forwarder*".

2.2.2 Gateways comercials.

Algunes de les principals passarel·les del mercat actualment son:

- Kerlink Wirnet

La Wirnet Station Kerlink va ser una de les primeres passarel·les LoRaWAN del mercat (disponible a l'any 2014):

Els paràmetres tècnics més rellevants son:

- ARM processor, 128MB RAM, 8GB eMMC
- Linux OS
- Semtech packet forwarder
- GPS
- PoE
- 9 channels (49 demodulators)
- Backhauls 2G/3G and ethernet.
- Filters for near-by channels
- From 900€



Figura 2.13. Gateway LoRaWAN Kerlink Wirnet.

- MultiTech Conduit

Solució modular amb diferents configuracions.

Els paràmetres tècnics més rellevants son:

- Interior/Exterior
- ARM9 processor at 400MHz and 256MB Flash
- Backhauls 2G/3G/4G, WiFi and ethernet
- GNSS
- 8/16 channels
- Max 27dBm RX

La solució estàndard d'exteriors té un preu superior als 1000€.



Figura 2.14. Gateway LoRaWAN MultiTech Conduit.

- WisGate Edge Pro

La RAK7289 és una passarel·la de RAK *wireless*, d'exteriors, amb una molt bona relació qualitat/preu, i múltiples opcions de connectivitat.

Els paràmetres tècnics més rellevants son:

- Basada en el RAK634 (MT7628N) 32 bits MIPS CPI with 1 core at 580MHz, 16Mb flash and 128Mb DDR2
- WisGateOS basat en OpenWRT
- Fins a 2x RAK2287/RAK5146 SX1302/3 LoRa concentrators (16 channels)
- Quectel EG95 for LTE connectivity
- GPS
- Active PoE
- A partir dels 370€



Figura 2.15. Gateway LoRaWAN WisGate Edge Pro.

- Mikrotik WPA8-LoRa

Mikrotik és una marca molt coneguda al món de radio enllaços.

Els paràmetres tècnics més rellevants son:

- MIPSBE arcuitecture, QCA9531 core at 650MHz, 64MB RAM, 16MB Flash
- Semtech packet forwarder
- Passive PoE
- -137dB @ SF12
- Class A and C devices supported (no GPS)
- WiFi and Ethernet backhaul
- 150€!



Figura 2.16. Gateway Mikrotik WPA8 LoRa.

- TTN GW / TTI Indoor GW

Solucions d'interior de baix cost.

Els paràmetres tècnics més rellevants son:

TTN Gateway:

Indoor:

- PIC32MZ2048EFM144 processor @ 200MHz
- Microchip concentrator based on the SX1301
- 8 channels
- Socket XBee



Figura 2.17. Gateway LoRaWAN TTN GW.

TTI Indoor Gateway:

- ESP8266 microcontroller
- 8 channels
- BasicStation compatible, but not configurable [\[4\]](#)



Figura 2.18. Gateway LoRaWAN TTI Indoor

2.3 Network Server.

Anteriorment, hem vist que en l'estàndard LoRaWAN, el *Network Server* desenvolupa un paper central en la gestió i operació de la xarxa. Funciona com el cervell de la xarxa LoRaWAN, gestionant la comunicació entre els dispositius finals i les aplicacions, i assegurant que les dades es transmeten de manera eficient i segura. Les principals funcions i responsabilitats del *Network Server* en una xarxa LoRaWAN son:

- **Gestió de Dispositius:** El *Network Server* gestiona l'activació dels dispositius finals, ja sigui mitjançant Activació per Personalització (ABP) o Activació per Aire

(OTAA), assignant identificadors únics i assegurant una unió segura de la xarxa. A més a més, manté l'estat de les sessions dels dispositius, incloent identificadors, comptadors de trames i claus de seguretat, per assegurar una comunicació continua i segura.

- **Encaminament de missatges:** L'encaminament de Uplink rep missatges d'*uplink* dels dispositius finals a través de les portes d'enllaç i els encamina a les aplicacions corresponents. Per altra part, programa i envia missatges de *downlink* des de les aplicacions fins als dispositius finals, gestionant les finestres de recepció i la prioritització de missatges.
- **Control de Taxa de Dades i Potència de Transmissió:** L'encarregat d'aquest procés és la *Adaptive Data Rate (ADR)* que optimitza la taxa de dades i la potència de transmissió dels dispositius finals per tal de millorar l'eficiència de la xarxa i la duració de la bateria dels dispositius, adaptant-se a les condicions canviants de l'enllaç de ràdio.
- **Gestió de canals:** Administra l'assignació de canals RF als dispositius finals per complir amb les regulacions i optimitzar l'ús de l'espectre.
- **Seguretat:** Gestiona les claus de seguretat per al xifrat i l'autenticació de missatges, assegurant la integritat i confidencialitat de les dades en la xarxa. A més a més, assegura que solament els dispositius autoritzats poden unir-se i operar a la xarxa, i que les dades es dirigeixin a les aplicacions adequades.
- **Deduplicació de missatges:** Elimina el duplicats dels missatges *uplink* rebuts per múltiples portes d'enllaç, assegurant que les aplicacions reben una única instància de cada missatge.
- **Gestió de la QoS^{12} :** Supervisa el rendiment de la xarxa i els dispositius, proporcionant diagnòstics i alertes per al manteniment i la resolució de problemes.
- **Interoperabilitat:** Assegura la interoperabilitat entre diferents fabricants de dispositius i aplicacions, complint amb l'estàndard LoRaWAN per garantir una àmplia compatibilitat.
-

2.3.1 *Metadata disponible en el Network Server.*

Segons l'estàndard LoRaWAN , des del *Network Server* es poden accedir a diversos paràmetres i metadades que proporcionen informació valuosa sobre la qualitat de l'enllaç, el rendiment de la xarxa i el comportament dels dispositius finals.

Alguns d'aquests paràmetres inclouen:

- **SNR (Signal-to-Noise Ratio):** El SNR mesura la relació entre el nivell de la senyal útil i el nivell de soroll de fons. És un indicador crucial de la qualitat del enllaç de comunicació, ja que un SNR alt suggereix una senyal clara i distintiva en comparació amb el soroll.
- **Data Rate:** La taxa de dades utilitzada per la transmissió d'un missatge específic.

¹² Quality of Service.

Això pot proporcionar informació sobre el *Spreading Factor* i l'ample de banda utilitzats, a més a més, pot indicar la distància aproximada al *gateway* i les condicions de l'enllaç de ràdio.

- **Freqüència de transmissió:** Es tracta de la freqüència específica en la que es transmet un missatge. Aquest és important per al seguiment de l'ús de l'espectre.
- **Gateway o gateways que van rebre el missatge:** Informació sobre quines portes d'enllaç han rebut el missatge d'un dispositiu final, el que pot incloure la quantitat de portes d'enllaç i les seves identitats. Això és útil per entendre la cobertura de la xarxa i la redundància del senyal.
- **Ubicació de les portes d'enllaç:** Si les portes d'enllaç estan equipades amb capacitats de geolocalització, les seves ubicacions poden ser reportades junt amb els missatges. Això, pot ajudar a determinar al ubicació aproximada dels dispositius finals mitjançant tècniques de triangulació.
- **Timestamps:** Les marques de temps de quan es van rebre els missatges i les portes d'enllaç i quan van ser processades pel *Network Server*. Això és essencial per la sincronització i l'anàlisi temporal de les dades.
- **ADR Status:** Informació sobre si l'ADR està activat per un dispositiu i com s'estan ajustant dinàmicament els paràmetres de transmissió, com la taxa de dades i la potència de transmissió, per optimitzar l'eficiència i el rendiment de la xarxa.
- **Marge de l'Enllaç (Link Margin):** El marge de l'enllaç indica quin marge té la senyal per damunt del mínim necessari per una comunicació fiable. És un bon indicador de la robustesa del enllaç de comunicació.
- **Potència de transmissió:** La potència amb la que es transmet un missatge des d'un dispositiu final, la qual pot influir en la cobertura i el consum d'energia del dispositiu.

A continuació, podem observar un exemple d'aquesta informació, obtinguda pel meu dispositiu Heltec, amb un sensor de temperatura i pressió HW-611. Aquestes dades son obtingudes a través de MQTT i es troba en format JSON:

```

topic: "v3/v01-test-app@ttn/devices/eui-70b3d57ed0066305/up"
payload: object
  end_device_ids: object
    device_id: "eui-70b3d57ed0066305"
  application_ids: object
    application_id: "v01-test-app"
  dev_eui: "70B3D57ED0066305"
  join_eui: "0000000000000000"
  dev_addr: "260BC83E"
  correlation_ids: array[1]
    0: "gs:uplink:01HXBJDP8ZJP993BZE3PP8FZX"
    received_at: "2024-05-08T11:31:41.589636023Z"
  uplink_message: object
    session_key_id: "AY9X7afuAFI+1LBqA6ZMug=="
    f_port: 2
    f_cnt: 37
    frm_payload: "C18GD114"
  decoded_payload: object
    pressio: 1016690.48
    temperatura: 26.07
  rx_metadata: array[1]
    0: object
      gateway_ids: object
        gateway_id: "ttn-cat-gw51"
        eui: "5031395325434750"
        time: "2024-04-28T16:44:54.838567Z"
        timestamp: 1512569044
        rssi: -92
        channel_rssi: -92
        snr: 11.5
        uplink_token: "ChkKFwoLdHRuY2F0LWd3NTE5CFAXOVMIQ0dQENTxn9EFGgwIncXtsQYQ08artwEgoPiP4YIs"
        received_at: "2024-05-08T11:31:41.356395922Z"
      settings: object
        data_rate: object
          frequency: "868100000"
          timestamp: 1512569044
          time: "2024-04-28T16:44:54.838567Z"
          received_at: "2024-05-08T11:31:41.385636045Z"
          consumed_airtime: "0.102912s"
        version_ids: object
        network_ids: object
      qos: 0

```

Figura 2.19. Dades obtingudes a través de MQTT en format JSON.

2.4 Join Server.

El *Join Server* és el responsable de la autenticació de dispositius que intenten unir-se a la xarxa. Aquest procés es fa mitjançant el maneig del procés de “*join-request*” i “*join-accept*” entre el dispositiu i la xarxa.

A continuació alguna de les característiques comunes d’un *Join Server*:

- **Autenticació de dispositius:** El *Join Server* autentica dispositius que intenten unir-se a la xarxa LoRaWAN. Utilitza la informació única d’identificació del dispositiu, com el *DevEUI* (Identificador Únic de Dispositius) i una clau precompartida (*AppKey*), per verificar que el dispositiu sol·licita unir-se a la xarxa és legítim i està autoritzat a fer-lo.
- **Gestió de claus:** Un cop que un dispositiu és autenticat, el *Join Server* genera les claus de sessió necessàries per la comunicació segura dins de la xarxa. Això inclou la *AppSKey* (Application Session Key), que s’utilitza per xifrar i desxifrar les dades de l’aplicació (*payload*), i la *NwkSKey* (Network Session Key), que s’utilitza per la

seguretat a nivell de xarxa, incloent la validació de missatges i la identificació de dispositius.

- **Maneig del procés de Join:** El *Join Server* gestiona el procés de *join-request* i *join-accept*. Quan un dispositiu desitja unir-se a la xarxa, envia un *join-request* al *Network Server*, que després el reenvia al *Join Server*. Si el dispositiu és autènticat de manera exitosa, el *Join Server* respon amb un *join-accept*, que inclou les claus de sessió xifrades i altres paràmetres necessaris per la comunicació.
- **Seguretat millorada:** Al centralitzar la autènticació i la gestió de claus, el *Join-Server* millora la seguretat de la xarxa. Això permet implementar pràctiques de seguretat robustes, com ara la rotació periòdica de claus i la gestió segura de credencials de dispositius, minimitzant el risc d'atacs i compromís de seguretat.
- **Desacoblament de funcions:** Al separar les responsabilitats de seguretat de la resta de la gestió de la xarxa, el *Join Server* permet una major flexibilitat i escalabilitat. Els operadors de xarxa poden actualitzar o modificar polítiques de seguretat, gestionar les claus i autènticar dispositius sense interferir amb les operacions del *Network Server*, millorant la mantenibilitat i la gestió de xarxa.
- **Interoperabilitat i flexibilitat:** En alguns casos, el *Join Server* pot operar independentment del proveïdor de la xarxa LoRaWAN, el que permet als dispositius unir-se a diferents xarxes sense necessitat de tornar a configurar les claus de seguretat o les credencials, facilitant la interoperabilitat entre diferents operadors i simplificant la gestió de dispositius a gran escala.

2.5 Regulacions regionals.

Les regulacions regionals de LoRaWAN son un conjunt de especificacions adaptades a les lleis i normatives locals sobre l'ús del espectre de ràdio en diferents parts del món. Aquestes regulacions afecten a diversos aspectes de la comunicació LoRaWAN, incloses les bandes de freqüència permeses, la potència de transmissió màxima, els canals disponibles i les polítiques d'ús de l'espectre... L'adaptació a aquestes regulacions és crucial per assegurar que les xarxes i dispositius LoRaWAN operen legalment i optimitzen el seu rendiment i rang dins de les restriccions locals.

2.5.1 Principals Regions i les seves especificacions.

-**Europa (EU868):** A Europa, LoRaWAN opera principalment en la banda de 868 MHz. Les regulacions inclouen un *duty cycle* limitat (generalment 1% en la majoria de canals), el que significa que un dispositiu solament pot transmetre durant un 1% del seu temps per tal de minimitzar la interferència amb altres dispositius.

-**Nord Amèrica (US915):** Als EE.UU, LoRaWAN utilitza la banda de 915 MHz. Aquesta regió permet un enfocament de *frequency hopping*¹³ amb 72 canals de 125 kHz i 8 canals de 500 kHz per transmissions de velocitat o de control. No hi ha restriccions de *duty cycle*, però

¹³ És una tècnica de comunicació que canvia la freqüència portadora d'una transmissió de ràdio de forma periòdica, per evitar interferències i augmentar la seguretat.

s'apliquen altres normatives, com la potència de transmissió màxima.

-Austràlia (AU915): Similar a US915, però amb algunes diferències en els canals de potència. Austràlia també utilitza la banda de 915 MHz i adopta un enfocament similar referent al salt de freqüència.

-Àsia (AS923): La especificació AS923 és utilitzada per diversos països asiàtics i es basa en la banda de 923 MHz. Té regulacions específiques pel *duty cycle* i la potència de transmissió, a més a més, es caracteritza per oferir una banda plana, sense salt de freqüència obligatori. Alguns països dins d'aquesta regió tenen requisits addicionals, com Japó amb la seva banda AS923-1.

-Equador i Brasil: S'utilitza la regió AU915.

-Altres: Les altres regions, com Amèrica Llatina i Àfrica, poden seguir una de les especificacions mencionades o tindre les seves pròpies regulacions basades en les condicions locals de l'espectre de ràdio.

2.5.2 Subbandes i canals.

Les subbandes i canals son conceptes utilitzats per organitzar l'espectre de radiofreqüència disponible per la transmissió de dades. L'estructura de subbandes i canals permet optimitzar l'ús de l'espectre, millorar la capacitat de la xarxa i complir amb les regulacions regionals sobre l'ús de l'espectre.

Una subbanda és un segment específic de l'espectre de ràdio assignat per LoRaWAN dins d'una banda de freqüència més àmplia. Les regulacions regionals de LoRaWAN divideixen la banda de freqüència disponible en diverses subbandes, cada una amb les seves pròpies normes. Es pot entendre com un agregador de canals.

Dins de cada subbanda, l'espectre es divideix en canals més estrets que son utilitzats per les transmissions de LoRa. Cada canal té una freqüència central específica i un ample de banda definit.

En la següent taula podem observar els canals *uplink* i *downlink* en la regió AU915:

AU915 Uplink Channels – BW: 125KHz, Unit: MHz, Spacing: 200KHz								
Channel 0-7	915.2	915.4	915.6	915.8	916.0	916.2	916.4	916.6
Channel 8-15	926.4	926.6	926.8	927.0	927.2	927.4	927.6	927.8
Channel 16-23	918.4	918.6	918.8	919.0	919.2	919.4	919.6	919.8
Channel 24-31	920.0	920.2	920.4	920.6	920.8	921.0	921.2	921.4
Channel 32-39	921.6	921.8	922.0	922.2	922.4	922.6	922.8	923.0
Channel 40-47	923.2	923.4	923.6	923.8	924.0	924.2	924.4	924.6
Channel 48-55	924.8	925.0	925.2	925.4	925.6	925.8	926.0	926.2
Channel 56-63	926.4	926.6	926.8	927.0	927.2	927.4	927.6	927.8
AU915 Downlink Channels – BW: 500KHz, Unit: MHz, Spacing: 1.6MHz								
Channel 64-71	915.9	917.5	919.1	920.7	922.3	923.9	925.5	927.1

Taula 2.5. Canals d'uplink i downlink en la regió AU915

2.6 Data Rate.

Quan parlem del concepte *Data Rate* es refereix a la velocitat a la que les dades son transmeses des dels dispositius finals a les portes d'enllaç i viceversa. Aquesta taxa és un factor crític en el rendiment de la xarxa ja que influeix en aspectes com el rang de la comunicació, la capacitat de la xarxa i la duració de la bateria dels dispositius finals. En conclusió, depèn de dos factors: l'ample de banda (BW) i el factor de dispersió (SF).

Teòricament és possible calcular-lo de manera aproximada seguint la següent fórmula:

$$\text{Bit rate} = \text{SF} \cdot \frac{\text{BW}}{2^{\text{SF}}} \cdot \text{CR} \quad (2.1)$$

On,

CR: Code Rate¹⁴, normalment 4/5.

BW: Ample de banda.

SF: Spreading factor.

LoRaWAN codifica les diverses combinacions d'ample de banda i de factor de dispersió i a cada combinació li assigna un identificador, que és bàsicament un número enter del 0 al 15.

Una modulació més lenta, caracteritzada per un *Data Rate* més baix, té l'avantatge d'augmentar la cobertura de transmissió. Això es deu al fet que els senyals més lents tenen una millor capacitat per traspasar obstacles i mantenir-se al rang de recepció a distàncies majors. No obstant, tot i que una modulació més lenta pot augmentar la cobertura, això també implica estar ocupant el canal durant més temps, per tant, això es tradueix en que es poden enviar menys dades al dia.

Un altre aspecte crític de la modulació lenta és la seva relació amb el consum d'energia. Degut a que la seva transmissió de dades es més duradora en el temps, el dispositiu transmissor necessita estar actiu durant períodes més alts, i per tant, comporta un augment considerable en el consum de bateria, el que pot desencadenar en una disminució considerable de la vida útil del dispositiu.

A continuació, podem observar els paràmetres regionals de *Data Rate* per la banda EU868:

¹⁴ Fa referència a la proporció entre la quantitat de dades útils que s'envien i el total de dades transmeses.

Date Rate	Configuració	Bit Rate
0	LoRa: SF12 / 125 kHz	250 bps
1	LoRa: SF11 / 125 kHz	440 bps
2	LoRa: SF10 / 125 kHz	980 bps
3	LoRa: SF9 / 125 kHz	1760 bps
4	LoRa: SF8 / 125 kHz	3125 bps
5	LoRa: SF7 / 125 kHz	5470 bps
6	LoRa: SF7 / 250 kHz	11000 bps
7	FSK: 50 kbps	50000 bps

Taula 2.6. Paràmetres regionals de Data Rate.

2.7 Cicle de treball.

El cicle de treball és conegut com *duty cycle* i es tracta d'un concepte regulatori que limita el percentatge de temps que un dispositiu pot transmetre per tal de minimitzar la interferència en les bandes de freqüència compartides, com ara les bandes ISM¹⁵, les quals son comunament utilitzades per LoRaWAN i altres tecnologies sense fils.

El cicle de treball màxim està regulat per l'ETSI¹⁶. Aquest defineix per la banda 868 MHz un ocupació de l'1% del temps, és a dir, un dispositiu pot transmetre durant 36 segons per hora. Considerant una transmissió típica de dades en LoRaWAN, que pot trigar entre 0,1 i 1 segon depenent de la càrrega de dades i la velocitat de transmissió, es pot calcular el nombre de missatges que es poden enviar. Si suposem una durada mitjana de 0,5 segons per transmissió amb 36 segons disponibles i una durada de 0,5 segons per missatge, es permeten fins a 72 missatges *uplink* per hora. Per altra banda, la restricció és similar per als missatges *downlink*, amb la limitació addicional de la política específica de la xarxa, com la TTN que limita a 10 missatges *downlink* per dia.

Aquestes restriccions varien segons la regió i estan definides per les autoritats reguladores locals. Per exemple, als EE.UU (banda US915), no s'aplica un límit de cicle de treball però s'utilitzen altres regulacions com el "Listen Before Talk".

Per altra banda, diferents xarxes poden imposar cicles més restrictius. Per exemple, TTN defineix una política de *Fair Access Policy* la qual imposa un temps màxim de 30 segons cada 24 hores per enviar (*uplink*) i 10 missatges cada 24 hores per rebre (*downlink*). Aquesta política està dissenyada per assegurar que tots els dispositius tinguin una oportunitat justa d'utilitzar la xarxa, evitant que alguns dispositius acaparin l'ample de banda disponible.

¹⁵ Industrial, Scientific and Medical.

¹⁶ European Telecommunications Standards Institute.

2.8 Mecanismes d'activació.

En la tecnologia LoRaWAN, els nodes han de ser activats i associats a una xarxa abans de que puguin comunicar-se de forma segura. Hi ha dos mecanismes principals d'activació: Activació per Personalització (ABP¹⁷) i Activació per Aire (OTAA¹⁸). Cada un té les seves pròpies característiques, avantatges i inconvenients. No obstant, tractarem més profundament la OTAA ja que serà la que farem servir a la part pràctica.

Activació per Personalització (ABP)

En l'Activació per Personalització, els identificadors i claus de seguretat es programen directament en el dispositiu abans del seu desplegament. Això inclou la direcció del dispositiu (DevAddr), la clau de xarxa (NwkSKey) i la clau d'aplicació (AppSKey). Les seves principals avantatges son:

- **Simplicitat:** No requereix un procediment d'unió, el que simplifica el procés de desplegament.
- **Connectivitat immediata:** Els dispositius poden comunicar-se amb la xarxa immediatament després de ser engegats, ja que totes les seves claus necessàries ja estan configurades.

No obstant, les seves principals desavantatges son:

- **Seguretat:** Si un dispositiu és compromès, les claus han de ser canviades manualment, el que podria ser impracticable per dispositius ja desplegats.
- **Escalabilitat:** No és ideal per xarxes grans, degut a que gestionar les claus de seguretat de molts dispositius pot esdevenir complicat. [\[5\]](#)

Activació per Aire (OTAA)

En l'Activació per Aire, el dispositiu realitza un procediment d'unió amb la xarxa cada cop que s'encén o necessita re associar-se, durant el qual es negocien dinàmicament un nou DevAddr i claus de sessió (NwkSKey i AppSKey). Les seves principals avantatges son:

- **Seguretat millorada:** Les claus de sessió es generen dinàmicament per cada sessió, millorant la seguretat.
- **Flexibilitat:** Permet la reassignació de direccions de dispositius i la actualització de claus, el que facilita la gestió de dispositius en xarxes de creixement.
- **Adaptabilitat:** Els dispositius poden adaptar-se a canvis en la configuració de la xarxa, com la reassignació a una porta d'enllaç.

No obstant, les seves principals desavantatges son:

- **Complexitat addicional:** El procés d'unió OTAA és més complex que la configuració directa de ABP.
- **Retard en la connectivitat:** Els dispositius han de completar el procediment d'unió abans de poder enviar o rebre dades útils. [\[6\]](#)

¹⁷ Activation By Personalization.

¹⁸ Over-The-Air Activation.

2.8.1 Procés d'activació (OTAA).

Anteriorment, hem vist en què consisteix l'Activació per Aire. A continuació, descriurem el procés d'OTAA en detall, inclosos els paràmetres necessaris.

Paràmetres requerits en el node

Per iniciar el procés OTAA, el dispositiu final necessita els següents identificadors i claus úniques, les quals son preconfigurades en el dispositiu:

- **DevEUI (Identificador Únic del Dispositiu):** És un identificador globalment únic (EUI-64) pel dispositiu final, similar a una direcció MAC en xarxes Ethernet, la qual assegura una identificació única a nivell mundial.
- **AppEUI (Identificador Únic de l'Aplicació):** És també conegut com JoinEUI en versions més recents de LoRaWAN. És un identificador globalment únic (EUI-64) per l'aplicació a la que el dispositiu està intentant unir-se. Serveix per assegurar que el missatge d'unió arribi a l'aplicació correcta en el servidor de xarxa.
- **AppKey (Clau de l'Aplicació):** És una clau AES¹⁹ de 128 bits precompartida entre el dispositiu i la xarxa, utilitzada per xifrar els missatges d'unió i derivar les claus de sessió (NwkSKey i AppSKey) després d'un procés d'unió amb èxit.

Procés d'activació OTAA

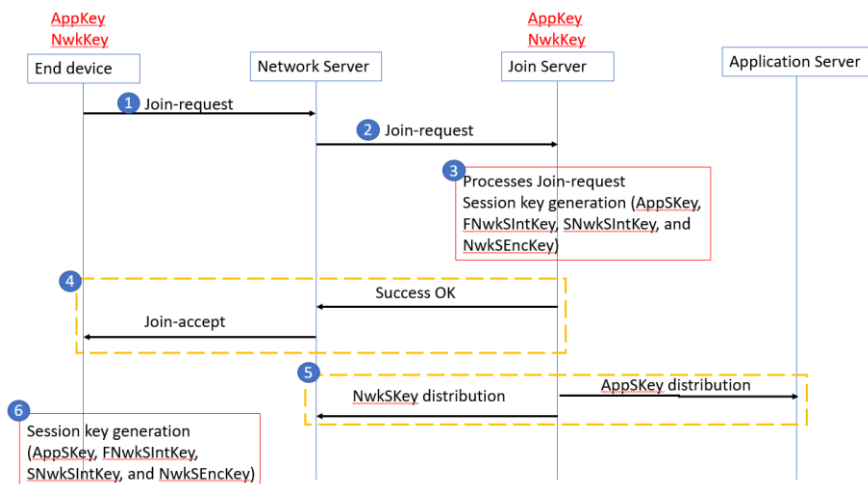


Figura 2.20. Intercanvi de missatges en una activació OTAA

1. Missatge de sol·licitud d'unió (Join Request): El dispositiu envia un missatge de sol·licitud d'unió que inclou el seu DevEUI, AppEUI i un Nonce, el qual és un número utilitzat un sol cop per assegurar la "frescura" del missatge. Aquest missatge està xifrat amb la AppKey.

2. Recepció i verificació: La xarxa, ja sigui el servidor de xarxa o el *Join Server*, depenent de l'arquitectura, rep el missatge de sol·licitud d'unió, verifica la validesa del DevEUI i

¹⁹ Advanced Encryption Standard.

AppEUI, i utilitza la AppKey per desxifrar i autenticar el missatge.

3. Missatge d'acceptació d'unió (*Join Accept*): Si la verificació és exitosa, la xarxa genera una resposta d'acceptació d'unió, la qual inclou els paràmetres necessaris per la sessió de comunicació, com la direcció del dispositiu (DevAddr) i les claus de sessió (NwkSKey per la capa de xarxa i AppSKey per la capa d'aplicació). Aquest missatge també inclou un comptador (JoinNonce) i es xifra utilitzant la AppKey.

4. Establiment de la sessió: El dispositiu rep el missatge d'acceptació d'unió, desxifra els paràmetres amb la AppKey i estableix els paràmetres de sessió. A partir d'aquest moment, totes les comunicacions anteriors es xifren utilitzant les claus de sessió derivades (NwkSKey i AppSKey).

5. Comunicació segura: Amb les claus de sessió establertes, el dispositiu pot ara comunicar-se de forma segura amb la xarxa, enviant i rebent missatges xifrats.

Característiques de seguretat

Al generar dinàmicament les claus de sessió per cada cicle d'unió, OTAA ofereix una seguretat robusta, ja que les claus de sessió no son preconfigurades i canvien amb cada reinici o reassociació.

Els dispositius poden realitzar el procés OTAA cada cop que es reinicien o quan és necessari re associar-se amb la xarxa. Això, permet actualitzar les claus de sessió i mantenir la seguretat de la comunicació. [\[7\]](#)

2.9 Característiques avançades.

A continuació es descriuran algunes de les característiques avançades que fan de LoRaWAN una opció potent per a diverses aplicacions IoT.

2.9.1 FUOTA.

FUOTA significa *Firmware Update Over The Air*, el qual permet actualitzar el firmware dels dispositius sense fils de manera automàtica. És una funcionalitat que ha estat definida pel Lora Alliance²⁰ i es troba per damunt de l'estàndard LoRaWAN, ja que en l'estàndard mateix no es contemplen certes de les característiques requerides per a que FUOTA funcioni, com per exemple els missatges *multicast*.

FUOTA és especialment important per als dispositius IoT desplegats en el camps, on les actualitzacions manuals serien poc pràctiques o impossibles degut al seu nombre, ubicació o accessibilitat.

L'estàndard LoRaWAN abasta FUOTA amb un conjunt de mecanismes dissenyats per

²⁰ És una organització sense ànim de lucre que agrupa empreses, institucions i altres organitzacions interessades en el desenvolupament i la promoció de l'estàndard LoRaWAN.

facilitar l'entrega segura i eficient d'actualitzacions de firmware als dispositius finals, considerant les limitacions inherents de LoRaWAN, com per exemple, el baix ample de banda i l'alt cost energètic associat a la transmissió de grans quantitats de dades.

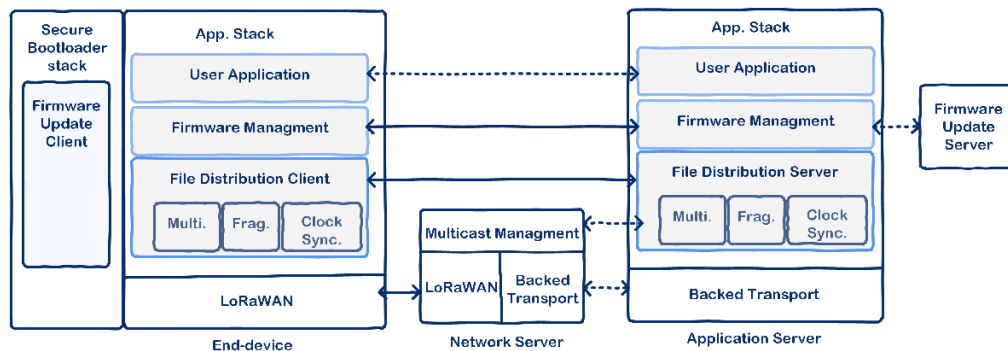


Figura 2.21. Esquema general per a FUOTA proposat per Lora Alliance.

Aspectes claus de FUOTA en LoRaWAN

- **Fragmentació:** Degut a l'ample de banda i a la dimensió limitada del *payload* en LoRaWAN, els arxius d'actualització de firmware que poden ser bastant grans, han de fragmentar-se en paquets més petits per a la seva transmissió. El dispositiu final després torna a muntar aquests fragments per reconstruir el firmware complet.
- **Multicast:** Per millorar l'eficiència i reduir el consum d'energia i l'ús del espectre, FUOTA sovint utilitza transmissions *multicast*, permetent que múltiples dispositius finals rebin la mateixa actualització de firmware simultàniament.
- **Seguretat:** La integritat i autenticitat de les actualitzacions de firmware son crítiques per evitar la instal·lació de firmware maliciós. FUOTA inclou mecanismes de seguretat, com ara el xifrat i la verificació de firmes, per assegurar que solament les actualitzacions autoritzades i autèntiques son aplicades.
- **Gestió d'energia:** Degut a que molts dispositius LoRaWAN son alimentats per bateries, FUOTA està dissenyada per minimitzar el consum d'energia durant el procés d'actualització. Per exemple, permet que els dispositius romanguin en mode de baix consum fins que sigui necessari rebre o processar les dades de l'actualització.
- **Recuperació d'errors:** L'estàndard inclou mecanismes per detectar i recuperar paquets perduts o corruptes durant la transmissió, assegurant que el firmware complet i sense errors pugui ser reconstruït pel dispositiu final.
- **Comptabilitat amb versions:** FUOTA ha de ser capaç de manejar dispositius amb diferents versions de hardware i firmware, permetent actualitzacions selectives i compatibilitat amb versions anteriors. [8]

Cal mencionar que FUOTA és implementada a nivell d'aplicació i per tant, no modifica cap aspecte de l'estàndard LoRaWAN, el que el fa perfectament compatible amb qualsevol implementació operativa existent.

2.9.2 Roaming.

El *Roaming* a LoRaWAN es refereix a la capacitat dels dispositius finals per comunicar-se amb una xarxa diferent a la seva xarxa d'origen, permetent així la mobilitat dels dispositius entre diferents cobertures de xarxa i la interoperabilitat entre operadores de xarxa. El concepte de *roaming* és crucial per aplicacions on els dispositius necessiten operar en múltiples regions geogràfiques o dins d'àrees cobertes per diferents xarxes LoRaWAN.

La LoRa Alliance, que és l'organisme que supervisa el desenvolupament de l'estàndard LoRaWAN, ha definit especificacions per al *roaming* en LoRaWAN, incloent els mecanismes per al *roaming* passiu i *roaming* actiu.

Roaming passiu

El *roaming* passiu és l'enfocament més simple per al *roaming* a LoRaWAN i es basa en l'idea de que les portes d'enllaç poden reenviar els missatges dels dispositius finals a múltiples servidors de la xarxa. En el *roaming* passiu:

- Les portes d'enllaç reenvien els missatges *uplink* dels dispositius finals al seu propi servidor de xarxa i també als servidors de xarxa d'altres xarxes amb les quals tenen acords *roaming*.
- El servidor de xarxa que reconeix el dispositiu, ja sigui perquè el dispositiu està registrat en aquella xarxa o té un acord de *roaming*, processarà el missatge, inclús si és necessari enviarà un missatge de *downlink* de tornada al dispositiu a través de la porta d'enllaç.
- El *roaming* passiu no requereix que els dispositius finals o portes d'enllaç realitzin cap mena d'operació especial de *roaming*.

Roaming actiu

El *roaming* actiu introdueix una comunicació més directa i controlada entre els servidors de xarxa involucrats en el *roaming*, permetent una coordinació més complexa però també oferint més control i funcionalitats. En el *roaming* actiu:

- S'estableix una comunicació directa entre el servidor de xarxa local (la xarxa en la qual es troba actualment el dispositiu) i el servidor de xarxa d'origen (la xarxa a la qual originalment pertany el dispositiu).
- Els missatges d'*uplink* i *downlink* s'intercanvien entre els servidor de xarxa a través d'interfícies de *roaming* definides en les especificacions LoRaWAN.
- El *roaming* actiu permet una millor gestió dels missatges de *downlink*, la sincronització de comptadors de trames i altres funcionalitats avançades.

Consideracions per al roaming

Per a que el *roaming* funcioni hi ha d'haver acords entre els operadors de les xarxes involucrades, incloent aspectes com l'intercanvi de dades, la seguretat i la facturació. A més a més, els dispositius han de ser capaços d'operar en les bandes de freqüència utilitzades per la xarxa en la qual es troben durant el *roaming*. Finalment, la seguretat és una consideració

crucial en el *roaming*, especialment per garantir que els missatges entre servidors de xarxa i dispositius finals estan degudament autenticats i xifrats. [9]

Packet Broker

El *Packet Broker* és una plataforma desenvolupada per TTN que facilita l'intercanvi de paquets de dades entre diverses xarxes LoRaWAN de forma segura. A través del *Packet Broker*, les xarxes poden intercanviar dades de forma segura, permetent així que els dispositius finals es connectin a qualsevol *gateway* disponible, independentment de la seva xarxa principal.

En el nostre cas pràctic, un dispositiu LoRaWAN d'un servei de monitorització ambiental pot estar subscrit a una xarxa privada específica. Quan el dispositiu es mou fora de l'àrea de cobertura d'aquesta xarxa, pot utilitzar *gateways* d'altres xarxes connectades al *Packet Broker* per continuar transmetent dades.

En resum, el Packet Broker és una eina essencial per habilitar i gestionar el *roaming* en xarxes LoRaWAN, permetent una interconnexió segura entre diverses xarxes i millorant la cobertura dels serveis IoT basats en LoRaWAN.

2.9.3 Geolocalització.

La geolocalització LoRaWAN es basa en l'ús de la tecnologia LoRa per determinar la ubicació de dispositius sense necessitat de GPS, el que ofereix una solució eficient en termes d'energia i costos per xarxes d'àrea ampla de baixa potència (LPWAN). Aquesta funcionalitat es suporta mitjançant dispositius finals LoRaWAN existents, eliminant costos addicionals i sense requerir més potència de processament.

En la pràctica, la geolocalització LoRaWAN s'implementa mitjançant multilateració, utilitzant senyals amb marques de temps provinents de tres o més portes d'enllaç LoRaWAN. Aquestes senyals ajuden a convertir la Diferència de Temps d'Arribada (TDOA²¹), en la distància entre les portes d'enllaç i el dispositiu IoT, obtenint així la seva ubicació. Aquest enfocament no sol és fàcil d'implementar i executar, sinó que també permet que cada sensor i dispositiu final IoT de LoRaWAN es localitzi potencialment sense la necessitat d'agregar un mòdul GPS costós al dispositiu, optimitzant el consum d'energia i reduint el cost del hardware del dispositiu. [10]

²¹ Time Difference of Arrival.

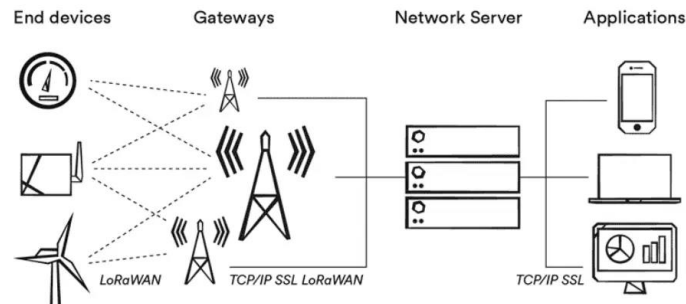


Figura 2.22. Geolocalització en LoRaWAN.

2.9.4 LoRaWAN Relay.

Relay és un dispositiu que actua com un intermediari de missatges entre els nodes finals i la infraestructura de la xarxa LoRaWAN. S'utilitza per estendre la cobertura en ambients on no és rentable instal·lar més *gateways*. Gràcies a aquesta característica, alguns nodes que es troben més allà de la cobertura *gateway* encara poden comunicar-se, utilitzant altres dispositius (anomenats *Nodes Relay*) per enviar els seus missatges.

Els nodes *Relay* no compleixen la funció de nodes regulars, tan sols s'encarreguen de passar els missatges dels nodes propers al gateway. És per això, que no s'ha de confondre pensant que aquesta és una infraestructura de red mesh²².

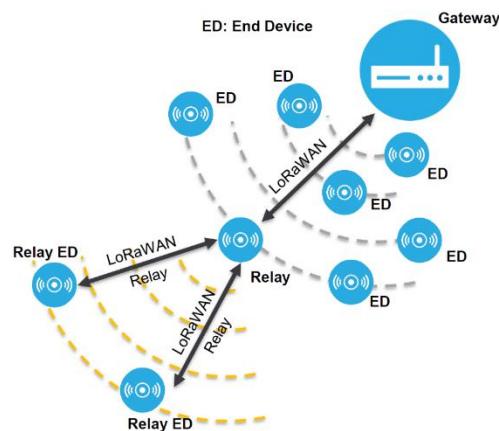


Figura 2.23. Relay en LoRaWAN.

Per implementar *Relay*, és necessari despertar el node al qual s'ha de reenviar els paquets (Node Relay). Això és necessari ja que en la majoria de casos, els nodes passen la majoria del temps en mode *sleep*. És a dir, els nodes per estalviar bateria normalment es troben en un estat de baix consum de bateria fins que els toca transmetre. És per això, que per implementar el *Relay* es necessari despertar el Node *Relay*.

²² És un tipus de xarxa en el qual els nodes estan interconnectats de manera que cada node pot connectar-se directament amb diversos altres nodes.

Per tal de poder dur-ho a terme, s'utilitza una tècnica denominada WOR (*Wake On Radio*), que s'utilitza en dispositius de comunicació sense fils per tal d'optimitzar el consum d'energia. Al combinar WOR amb LoRaWAN els Nodes *Relay* poden configurar-se per despertar-se en intervals programats per verificar si hi ha senyals de ràdio que indiquen que necessiten realitzar alguna acció, com per exemple transmetre dades recopilades o rebre instruccions. Fora d'aquests intervals, el dispositiu pot tornar al mode *sleep* per conservar energia.

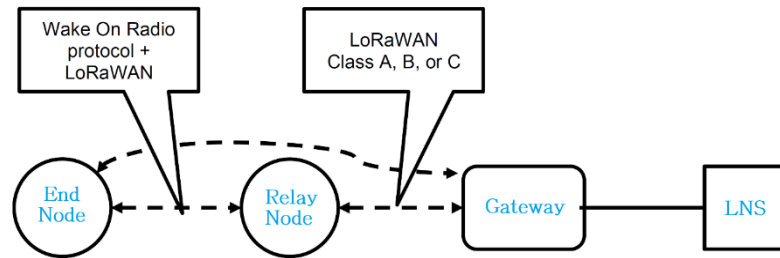


Figura 2.24. Missatge WOR per despertar el node Relay.

El WOR s'implementa simplement incrementant la mesura del preàmbul²³ de la modulació LoRa, arribant a ser tan llarg com un segon. D'aquesta manera, el dispositiu *Relay*, té una alta probabilitat que al despertar-se detecti la senyal de WOR si està sent transmesa. [11]

2.10 Revisió de sensors (GPS, CO2, soroll, gasos, etc.).

Un cop feta la recerca sobre la tecnologia LoRaWAN, hem observat que aquesta ens ofereix la capacitat de connectar una àmplia varietat de sensors, com ara els de qualitat de l'aire, temperatura, humitat i detecció de gasos, i transmetre les seves dades de manera fiable a llarga distància. Aquesta característica resulta ser un element crucial en el desenvolupament d'un sistema exhaustiu de seguiment, el qual recopila dades de múltiples indicadors ambientals. A més a més, la compatibilitat de LoRaWAN amb eines com Node-RED, MQTT, InfluxDB i Grafana, possibilita un maneig avançat de les dades, incloent el seu processament en temps real, l'emmagatzematge i la seva representació gràfica, constituint una eina integral per al seguiment ambiental en el camp de l'IoT.

2.10.1 Tipus de sensors i les seves aplicacions.

A continuació us mostraré una llista de sensors i les seves respectives aplicacions:

-GPS: Sensor utilitzat per a la geolocalització de dispositius en temps real. Essencial per a aplicacions de seguiment de la qualitat de l'aire.

²³ Seqüència de símbols enviats al començament d'un paquet de dades que serveixen diverses funcions clau en la comunicació.



Figura 2.25. GPS Milesight.

-Sensors de CO₂: Aquest tipus de sensors mesuren les concentracions de diòxid de carboni, els quals son indicadors clau per determinar la qualitat de l'aire.



Figura 2.26. Sensor de qualitat de l'aire MQ-135.

-Sensors de soroll: Essencials per a la monitorització de la contaminació acústica, especialment en zones urbanes densament poblades.

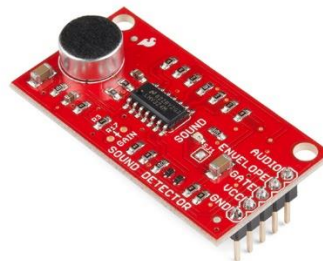


Figura 2.27. Sensor de so Sparkfun.

-Sensors de gasos: Aquests sensors permeten detectar una gran quantitat de compostos orgànics volàtils (alcohols, aldehids, cetones, àcids orgànics, amines, hidrocarburs alifàtics i aromàtics) a més de proporcionar una lectura de càlcul equivalent de diòxid de carboni (eCO₂). Disposa d'interfície i2c per a una comunicació més senzilla amb els nostres microcontroladors.

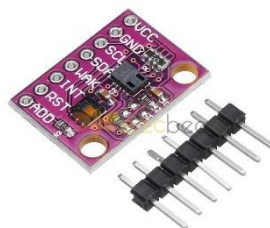


Figura 2.28. Sensor de gasos CCS811.

-Sensors de temperatura i pressió / Sensors de temperatura i humitat: Sensors essencials per analitzar les condicions meteorològiques que poden arribar a afectar a la dispersió i la concentració de contaminants.

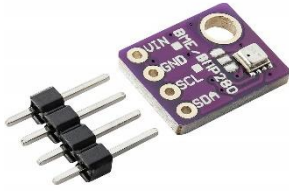


Figura 2.29. Sensor de temperatura i pressió BMP280.

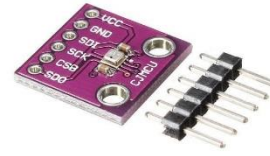


Figura 2.30. Sensor de temperatura i humitat BME280

2.10.2 Gasos i paràmetres ambientals mesurables.

A continuació podem observar els gasos i paràmetres crítics que podem mesurar amb sensor IoT amb la finalitat de poder monitoritzar la qualitat ambiental:

- Monòxid de Carboni (CO)
- Diòxid de Nitrogen (NO2)
- Sulfur Diòxid (SO2)
- Ozó (O3)
- Partícules en suspensió (PM10 i PM2.5)
- Compostos orgànics volàtils (VOCs)
- Amoníac (NH3)
- Temperatura ambiental
- Humitat relativa
- Pressió atmosfèrica

2.11 Monitorització de la qualitat ambiental.

La monitorització de la qualitat ambiental utilitzant sensors IoT implica la instal·lació de xarxes de sensors que recullen en temps real. El desenvolupament i la implementació de tecnologies de monitorització ambiental han guanyat importància a mesura que la preocupació per la salut ambiental continua creixent., a més a més, aquestes permeten una gestió més eficient dels recursos ambientals i ajuden a respondre ràpidament a les emergències de contaminació.

Actualment, hi ha 4 objectius clars i concisos respecte la monitorització ambiental:

- **Prevenió de la contaminació:** Detectar les principals fonts de contaminació per tal de controlar i reduir així l'impacte mediambiental.
- **Conservació de recursos:** Monitoritzar la qualitat i l'ús dels recursos naturals per tal de promoure la seva conservació.
- **Salut pública:** Avaluar els riscos ambientals a que estan exposats els humans i per tant, que poden atemptar contra la seva salut. Per exemple, la qualitat de l'aire, l'aigua...
- **Compliment legal:** Assegurar-se que les activitats industrials i altres pràctiques compleixen la legislació ambiental actual.

No obstant, malgrat tots els avenços tecnològics fets fins avui en dia, encara hi han diversos reptes. El primer de tots, la cobertura i l'escalabilitat, ja que estendre la xarxa de monitorització a tots els àmbits és molt costós. En segon lloc, és difícil mantenir la integritat de les dades degut a la fiabilitat de les dades recollides. Finalment, el repte més gran a que s'enfronta la monitorització és la de transformar grans volums de dades en informació que pugui ser realment útil. [12]

Finalment, cal mencionar que el futur de la monitorització ambiental està completament vinculat a la integració i la completa automatització de les tecnologies IoT i IA. És per això, que l'adopció d'algoritmes d'aprenentatge automàtic per processar i analitzar les dades ambientals milloren considerablement la detecció precoç de riscos mediambientals i en faciliten una resposta més ràpida. [13]

2.11.1 Avenços en sistemes intel·ligents de monitorització del entorn mitjançant IoT i sensors.

En l'era de la informació i la tecnologia, la monitorització intel·ligent del medi ambient (SEM)²⁴ ha emergit com una necessitat ineludible per afrontar els creixents reptes mediambientals globals. La integració de la Internet de les Coses (IoT) i els sensors avançats ha possibilitat el desenvolupament de sistemes de monitorització que no només detecten, sinó que també reaccionen proactivament davant les fluctuacions ambientals. Aquesta revisió s'endinsa en els avenços recents i els desafiaments persistents en l'aplicació del SEM, destacant les seves aplicacions en l'agricultura, la qualitat de l'aigua i de l'aire, i més enllà.

[14]

L'objectiu principal de la monitorització intel·ligent del medi ambient és abordar els reptes mediambientals com la contaminació de l'aire, aigua i radiació, així com optimitzar l'agricultura mitjançant tecnologies avançades com la Internet de les Coses (IoT) i els sensors intel·ligents. Aquesta tecnologia permet una vigilància més precisa i efectiva, essencial per a la sostenibilitat global i la salut pública.

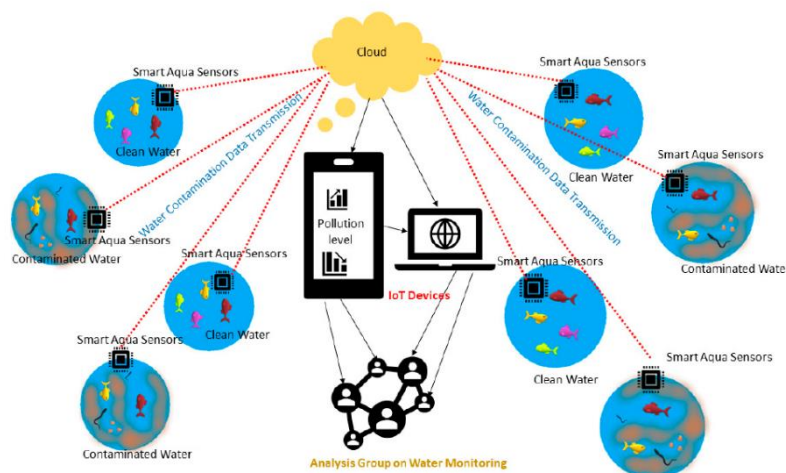


Figura 2.26. Sistema SEM que destaca la contaminació de l'aigua i la seva monitorització mitjançant la connexió al núvol de IoT i sensors.

²⁴ Smart Environment Monitoring.

L'evolució del SEM està intrínsecament lligada als progressos en IoT i la tecnologia de sensors. Aquests dispositius, cada vegada més petits, eficients i connectats, recullen dades de múltiples fonts ambientals en temps real. La capacitat de processar aquestes dades mitjançant algoritmes d'Intel·ligència Artificial (IA) i Machine Learning permet una resposta ràpida i precisa, transformant així la gestió mediambiental en una tasca més dinàmica i predictiva. El SEM es manifesta en diverses formes, cadascuna adaptada a necessitats específiques:

- **Sistemes de Monitorització de l'Agricultura Intel·ligent (SAM)²⁵:** Els SAM utilitzen sensors per mesurar factors crítics com la humitat del sòl, pH, i nivells de nutrients, facilitant així pràctiques agrícoles més informades i sostenibles.
- **Sistemes de Monitorització de la Contaminació de l'Aigua Intel·ligents (SWPM)²⁶:** Aquests sistemes s'apliquen per detectar contaminants perillosos en fonts d'aigua, ajudant en la preservació de la salut pública i dels ecosistemes aquàtics.
- **Sistemes de Monitorització de la Qualitat de l'Aire Intel·ligents (SAQM)²⁷:** El SAQM és vital per supervisar i gestionar la presència de contaminants aeris, com els òxids de nitrogen i les partícules fines, que afecten tant la salut humana com la climatologia local.

Tot i que el SEM presenta avantatges significatius, també enfronta desafiaments notables. La interoperabilitat entre diferents plataformes tecnològiques continua sent un obstacle major, limitant la capacitat dels sistemes per interactuar eficaçment. A més, la dependència de la infraestructura de xarxa pot restringir l'aplicació del SEM en regions remotes o subdesenvolupades.

Mirant cap al futur, és essencial que els investigadors i desenvolupadors se centrin en superar les barreres existents en la implementació del SEM. L'exploració de noves tecnologies de sensors, juntament amb l'adopció de normes globals per a la recollida i l'intercanvi de dades, potencialitzarà la capacitat del SEM per gestionar els desafiaments mediambientals de manera més eficient.

A continuació es presenten diversos estudis on s'ha implementat la caracterització de la qualitat de l'aire usant sensors heterogenis i mètodes d'aprenentatge automàtic.

En aquest article [\[15\]](#) s'ha estudiat l'ús de diferents mètodes i sistemes de monitorització de la qualitat de l'aire (SAQM), presentant un resum en la *Taula 2.1* sobre les diverses aproximacions utilitzades. S'ha implementat la caracterització de la qualitat de l'aire usant sensors heterogenis i mètodes d'aprenentatge automàtic. També s'ha realitzat una avaluació de la qualitat de l'aire usant nodes de sensors fixos i mòbils, que permeten comprovar la qualitat de l'aire tant en estat estacionari com en moviment.

Per a l'avaluació en ambients mòbils, s'han desplegat sensors compatibles que funcionen de

²⁵ Smart Agriculture.

²⁶ Smart Water Pollution Monitoring

²⁷ Smart Air Quality Monitoring

manera satisfactòria en entorns dinàmics. Les dades capturades pels nodes sensors intel·ligents han estat processades i analitzades mitjançant tècniques d'aprenentatge automàtic. En un altre estudi sobre el control de la qualitat de l'aire, s'han utilitzat tècniques IoT i d'aprenentatge automàtic centrades en la valoració de la contaminació de l'aire, desplegant sensors de gas que ajuden a capturar partícules i analitzar els contaminants barrejats a l'aire.

Per ser una mica més concrets, en alguns dels estudis s'han usat mecanismes de control de la qualitat de l'aire basats en RFID i la implementació de sensors de gas per determinar el nivell de contaminació de l'aire, usant IoT per analitzar les dades sensorials capturades. La tecnologia RFID ha estat utilitzada principalment per a la detecció de contaminants i la comunicació amb xarxes de sensors sense fils (WSN) amb l'ajuda de dispositius IoT connectats a través d'una arquitectura WSN. Per altra banda, un sistema SAQM es va estudiar usant LoRaWAN, essent aquest treball molt útil per detectar components com temperatura, pols, humitat i diòxid de carboni a l'aire. A més, es van detectar components com PM10²⁸, PM2.5, SO2, òxids de nitrogen (NOx), O3, plom, CO i benzè.

Research	Purpose	Data and Technique
Air quality characterization [58]	Air quality monitoring	Heterogeneous sensors; machine learning based predictive model
Air quality modeling [75]	Air quality monitoring	Mobile nodes
Air pollution [76]	Air quality monitoring	Gas sensors from mobile vehicle data, IoT and machine learning
Air quality in vehicular sensor network [77]	Air quality monitoring	Sensors in mobile nodes
Detection of VOC in air [25]	Organic compound detection	Infrared sensors, spectroscopy and machine learning
PM2.5 estimation [78]	Air quality in terms of PM2.5 concentration levels	Spatio-temporal geographic data, Extreme machine learning technique
Urban air [44]	Urban air pollution in terms of O3, NO2 and SO2 concentrations	Forecasting models
Air pollution prediction [79]	Air pollution control	RFID, Gas sensors and IoT
Smart air quality [80]	Air quality	Temperature, humidity, dust and carbon dioxide sensor; LoRaWAN
Intelligent air quality system [83]	Air quality for detection of CO2, NOx, temperature and humidity	UV light, AI and sensors
Ozone, PM10 and PM2.5 [84]	PM10, PM2.5, SO2, Oxides of nitrogen (NOx), O3, lead, CO and benzene	Machine learning and spatio-temporal data
Air quality [85]	Air quality	Heterogeneous sensors and SVM
Abnormal O3 [84]	Ozone (O3)	Ozone data and deep learning
Wearable sensors [86]	Temperature and humidity monitoring	Wireless and wearable sensor technology
CO2 monitoring [87]	Monitoring of carbon dioxide	IoT and cloud technologies
Indoor air quality [88]	Air quality monitoring in indoor environment	IoT, VOC: volatile organic compound; LoRaWAN

Taula 2.7. Investigació sobre sistemes SAQM que utilitzen Machine Learning i IoT.

2.11.2 Anàlisi sobre els sistemes de Monitorització Intel·ligent del Medi Ambient (SEM).

Es van analitzar les tendències de recerca per avaluar el volum de recerca realitzat en l'àrea de SEM. L'estudi de tendències es va fer mitjançant una cerca de publicacions en les bases de dades de Science Direct [16] de manera anual des de l'any 1995 fins al 2020. Es pot veure clarament en la Taula 2.2 que el volum de recerca sobre SEM ha anat augmentant amb el temps en ambdós casos, tant en la recerca que emprava IoT i xarxes de sensors sense fils

²⁸ Partícules en suspensió amb un diàmetre aerodinàmic de fins a 10 µm.

(WSN), com en la recerca que utilitza IoT i aprenentatge automàtic, indicant un avanç significatiu en l'aplicació d'aquestes tecnologies per millorar la monitorització de l'aire, l'aigua i els sistemes agrícoles. Un fet interessant que es desprèn de la taula és que la recerca que utilitza mètodes moderns d'aprenentatge automàtic encara està una mica per darrere de la que no utilitza cap aprenentatge automàtic.

Year	Research Using IoT and WSN	Research Using IoT and Machine Learning
1995–2000	21	2
2001–2005	7	7
2006–2010	22	2
2010–2015	541	175
2015–2020	6181	3004

Taula 2.8. Recull de contribucions a la investigació utilitzant IoT i WSN, e IoT i Machine Learning²⁹

A més a més, s'ha detectat que, tot i l'augment en la recerca, hi ha desafiaments importants com la interoperabilitat³⁰ entre dispositius heterogenis que encara no s'han superat completament. Aquesta falta d'estandardització pot complicar la integració de dades i la precisió dels sistemes SEM. [17]

Recomanacions Clau

Desenvolupament d'un marc de mètodes d'aprenentatge automàtic: És crucial establir un conjunt robust de models de classificació, predicció i previsió que funcionin eficaçment amb qualsevol tipus de dades i per a qualsevol finalitat dins del SEM.

Estandardització de la recollida i anàlisi de dades: Implementar normes que facilitin la compatibilitat entre diferents tipus de sensors i sistemes per millorar la qualitat i interoperabilitat de les dades recollides.

Focus en zones rurals i infraestructures menys desenvolupades: Encara que el SEM té un gran potencial en zones urbanes, és fonamental expandir aquesta tecnologia a zones on la infraestructura pot ser limitada, especialment en països en desenvolupament.

Integració de tecnologia avançada en SEM: Incorporar noves tecnologies com la intel·ligència artificial (IA) i l'aprenentatge profund per desenvolupar sistemes experts que puguin preveure i gestionar els problemes ambientals de manera més efectiva.

En conclusió, la investigació en SEM ha avançat significativament, mostrant la capacitat de la tecnologia per abordar problemes mediambientals complexos. No obstant això, per aprofitar plenament aquests avanços, es necessiten solucions innovadores que abordin els reptes de interoperabilitat i precisió en la recollida de dades. La implementació de les

²⁹ Disciplina de la IA que a través d'algoritmes dota als ordenadors de la capacitat d'identificar patrons en dades massives.

³⁰ Capacitat de les aplicacions i els sistemes per intercanviar dades de forma segura i automàtica.

recomanacions esmentades pot ajudar a dissenyar sistemes SEM més robustos i eficients, amb el potencial de superar els desafiaments ambientals globals amb una major eficàcia. [\[18\]](#)

2.12 Importància de la recollida de dades en temps real i aplicacions.

Avui en dia, el nostre món canvia tan ràpidament que de vegades pot ser difícil de comprendre. Estem constantment bombardejats amb milions de punts de dades que informen decisions, ja sigui de manera conscient o no. Els científics estimen que l'humà mitjà pren més de 35.000 decisions al dia.

Els sistemes moderns d'intel·ligència empresarial no són diferents dels nostres cervells en la manera en què recullen, processen i analitzen dades entrants contínuament. Això és un gran canvi respecte a una època en què les dades es processaven per lots. El procés continu és la base del processament i anàlisi de dades en temps real.

L'anàlisi en temps real significa que es poden processar i consultar noves dades immediatament a mesura que es creen per informar decisions en el moment i guiar la presa de decisions empresarials. Per alguns científics, es veu com una disciplina que aplica lògica i matemàtiques a les dades amb l'objectiu de proporcionar visions per prendre millors decisions ràpidament. [\[19\]](#)

2.12.1 Per què és important recolzar-se en el Machine Learning?

Un dels reptes més grans en l'anàlisi de dades en temps real és la capacitat de capturar i processar grans volums de dades de manera contínua. Aquestes dades es generen a una velocitat impressionant, exigint l'ús de tècniques i algoritmes especialitzats que facilitin el seu processament i anàlisi automàtic.

En aquest context, es fan servir algoritmes d'aprenentatge automàtic i mètodes de processament de dades en paral·lel. És crucial disposar d'una infraestructura robusta, com ara serveis en el núvol, que suporti l'emmagatzematge i processament d'aquestes informacions. També és essencial tenir la capacitat d'identificar patrons i tendències dins les dades analitzades, utilitzant algoritmes d'aprenentatge automàtic juntament amb tècniques estadístiques.

Finalment, comptar amb un equip de científics de dades és vital per interpretar correctament i actuar segons els resultats obtinguts de les anàlisis realitzades. [\[20\]](#)

2.12.2 Beneficis de les dades en temps real.

Un cop coneguda la importància de les dades en temps real, a continuació podem veure tots els seus beneficis:

-Agilitat en la presa de decisions: Les dades actualitzades instantàniament permeten una presa de decisions ràpida i ben fonamentada. Això facilita l'avaluació de situacions i la identificació d'oportunitats, permetent respondre de manera més efectiva als canvis del mercat.

-Detecció immediata de problemes: L'accés continu a la informació permet monitoritzar les operacions en tot moment i detectar qualsevol incidència de manera instantània, evitant així repercussions més greus. Un exemple clar és el maneig de crisi en xarxes socials, on una detecció i resposta ràpides són clau per mantenir la reputació.

-Millora de la qualitat del servei: Tenir la informació correcta en el moment adequat és crucial per evitar danys en l'experiència del client i la possible pèrdua d'aquests. La capacitat per corregir errors de servei de forma immediata distingeix les empreses en l'era digital.

-Personalització i millora de l'experiència del client: Utilitzar les dades en temps real per entendre millor les preferències i necessitats dels clients permet oferir una experiència molt més personalitzada i ajustada, incrementant la seva satisfacció.

-Optimització de processos: Les dades instantànies sobre rendiment operatiu permeten identificar àrees de millora, optimitzant processos i realitzant ajustos per augmentar l'eficiència i reduir costos.

-Avantatges competitiu: Estar al corrent de tot el que passa en cada instant ofereix un avantatge competitiu, ja que permet prendre decisions més ràpides i precisament ajustades a les demandes del mercat.

-Més seguretat: La capacitat de detectar i reaccionar a amenaces de seguretat de manera ràpida i eficient és un altre avantatge de les dades en temps real, permetent una millor protecció dels actius digitals. [\[21\]](#)

2.12.3 Per què les empreses implementen aquesta tècnica ?

Dins del món empresarial, l'anàlisi de dades en temps real es desplega per tal de millorar l'eficàcia dels processos, elevar-ne la rendibilitat de l'empresa i enriquir l'experiència del client.

Imagineu una empresa de comerç electrònic que pot ajustar els preus dels seus productes en funció de la demanda en temps real, maximitzant les vendes i els beneficis. O una empresa de transport que optimitza les rutes dels seus camions en funció del trànsit actual, reduint costos i millorant l'eficiència. Aquestes són només algunes de les possibilitats que ofereix l'anàlisi de dades en temps real.

És per això, que el potencial de l'anàlisi de dades en temps real s'estén a una àmplia gamma d'indústries:

- **Finances:** Detecció de transaccions fraudulentas, anàlisi de mercats financers i presa de decisions d'inversió basades en informació actualitzada.
- **Salut:** Monitorització de l'estat dels pacients, detecció de malalties en temps real i optimització dels recursos sanitaris.

- **Seguretat i intel·ligència:** Detecció de patrons d'activitat sospitosa, prevenció d'atacs cibernètics i terroristes, i monitoratge del trànsit de xarxes per a identificar activitat maliciosa.

Independentment de la seva grandària, les empreses que adopten l'anàlisi de dades en temps real poden obtenir beneficis tangibles:

- **Decisions immediates i estratègiques:** Actuar ràpidament davant situacions inesperades, aprofitar oportunitats i minimitzar riscos.
- **Plans de negoci efectius:** Augmentar la demanda, millorar el posicionament en el mercat i desenvolupar estratègies competitives.
- **Resolució de problemes oportuna:** Identificar i solucionar problemes de manera immediata, minimitzant el seu impacte negatiu.
- **Reducció de costos:** Optimitzar processos, eliminar ineficiències i estalviar recursos.
- **Màxima rendibilitat:** Prendre decisions basades en dades que maximitzen el retorn de les inversions.
- **Precisió i prevenció d'errors:** Evitar errors operatius, administratius o de comercialització basant-se en informació en temps real. [\[22\]](#)

L'anàlisi de dades en temps real no és només una tecnologia, sinó un canvi de paradigma en la manera com les empreses operen i prenen decisions. En un món en constant evolució, la capacitat d'adaptar-se i actuar ràpidament s'ha convertit en una clau per a l'èxit. L'anàlisi de dades en temps real proporciona les eines necessàries per navegar en aquest entorn dinàmic i convertir els reptes en oportunitats. Aquelles que adoptin aquesta tecnologia de manera efectiva estaran en una posició privilegiada per a navegar en un entorn dinàmic i competitiu, aconseguir un avantatge competitiu sostenible i impulsar el seu creixement a llarg termini.

En resum, l'anàlisi de dades en temps real es converteix en un motor transformador que redefineix la manera com les empreses operen, competeixen i prenen decisions. No es tracta d'una simple eina tecnològica, sinó d'un canvi de paradigma que impulsa l'agilitat, l'eficiència, la innovació i el creixement en un entorn dinàmic i en constant evolució. [\[23\]](#)

3. DISSENY DE LA PLATAFORMA

El disseny de la plataforma implica tenir una estructuració molt ben organitzada per tal de garantir una implementació eficaç i escalable.

3.1 Arquitectura general del sistema.

A continuació, podem veure d'una forma gràfica l'esquema de la solució proposada per al desplegament de la meua plataforma de sensors amb tecnologia LoRaWAN.

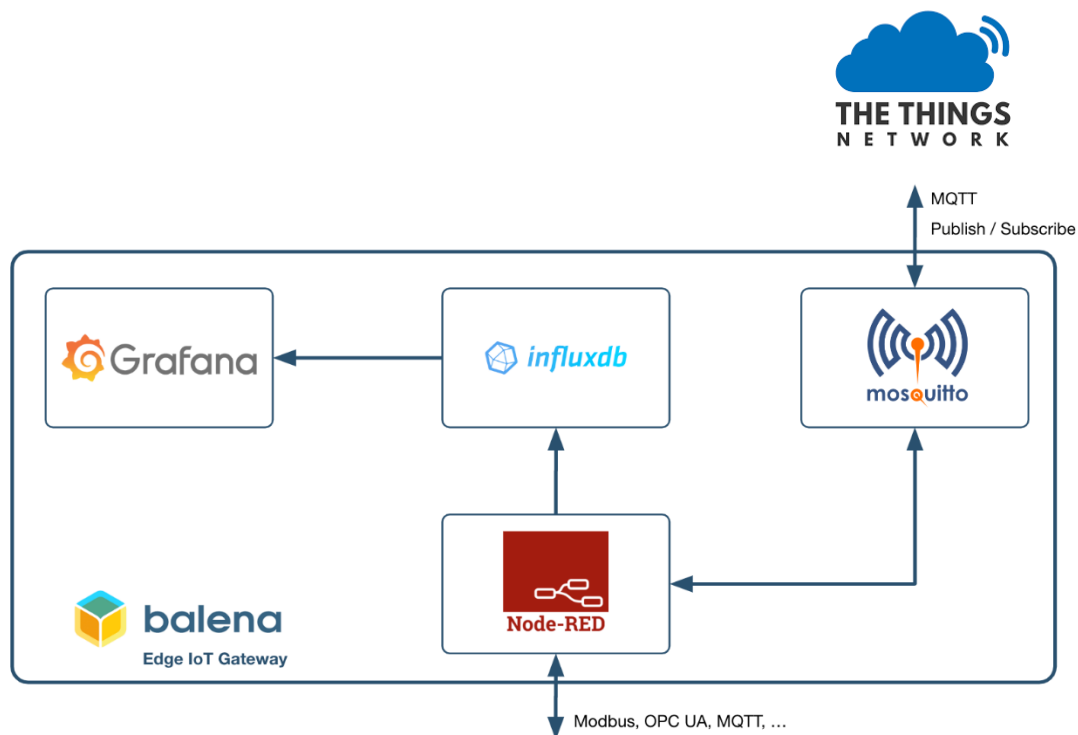


Figura 3.1. Arquitectura de la plataforma.

3.2 Selecció i justificació dels sensors: Temperatura, Pressió, CO2, TVOC.

Finalment, de tots els sensors exposats a la part inicial del projecte, m'he decantat pels següents sensors:

En primer lloc, el sensor de temperatura i pressió BMP280. Té una gran capacitat per proporcionar lectures precises de la pressió i la temperatura, a més a més, és ideal per aplicacions on es requereix monitoratge ambiental.

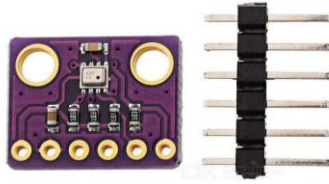


Figura 3.2. Sensor BMP280.

En segon lloc, tenim el sensor CCS-811, el qual està dissenyat per a la detecció de compostos orgànics volàtils, el que el fa realment bo en la monitorització de la qualitat de l'aire. A més a més, a part de poder detectar una àmplia gamma de gasos, està realment optimitzat per tal de consumir molt poca energia. A part, la tria d'aquest sensor ve basada per les referències dels diferents estudis que informen que el diòxid de carboni és un gas que, tot i no ser tòxic en concentracions habituals a l'aire lliure, és un indicador important de la qualitat de l'aire interior i de la ventilació en espais tancats. L'acumulació de CO₂ en espais tancats pot indicar una ventilació inadequada, la qual cosa pot portar a una acumulació d'altres contaminants més perillosos i afectacions en el benestar i la salut de les persones. Segons [Sunny et al. \(2020\) \[12\]](#), el monitoratge de CO₂ és crucial per identificar la qualitat de l'aire i implementar mesures correctives en entorns interiors.

Per altra banda, trobem els compostos orgànics volàtils (VOCs) els quals són una categoria de contaminants que inclouen una gran varietat de productes químics, molts dels quals poden tenir efectes adversos sobre la salut a llarg termini, com ara càncer o problemes respiratoris. Els TVOCs són emesos per una àmplia gamma de productes, incloent pintures, solvents, i alguns materials de construcció, així com processos industrials. L'estudi de [Ranjan Laha et al. \(2022\) \[13\]](#) ressaltava la importància de detectar aquests compostos per protegir la salut humana i el medi ambient.

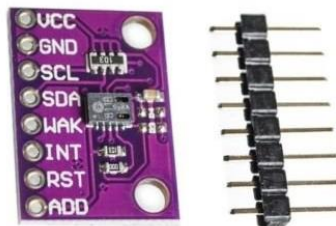


Figura 3.3. Sensor CCS-811.

L'ús d'aquests dos sensors en una plataforma de sensors connectada mitjançant LoRaWAN és justificat bàsicament per la seva aplicabilitat en els escenaris reals, on la combinació de les mesures de pressió, temperatura i la composició de l'aire pot ser de caire essencial per a l'anàlisi ambiental. A part, la seva integració en una xarxa com pot ser LoRaWAN permet un desplegament eficient pel que fa a energia i consum.

3.3 Integració dels dispositius amb la xarxa TTN.

TTN³¹ és una xarxa d'IoT global gratuïta, descentralitzada i oberta que permet a qualsevol persona aportar el seu coneixement d'IoT per millorar contínuament la plataforma. Va ser fundada a Amsterdam l'estiu de 2015 per Johan Stokking i Wienke Giezeman, entre d'altres, per promoure el desenvolupament i disseny de la web. La plataforma admet GW (*gateways*) distribuïts arreu del món i té una infraestructura de fons que proporciona funcionalitat de servidor web i d'aplicacions.

TTN està desenvolupat com a codi obert i el codi de fons està disponible públicament a GitHub, cosa que facilita l'accés i la contribució de qualsevol persona. Actualment, TTN té aproximadament 9.000 GW registrats utilitzats per aproximadament 80.000 usuaris a tot el món, cosa que demostra una comunitat en creixement. Això proporciona a moltes persones un accés fàcil i econòmic a un entorn IoT que es pot desplegar projectes de diversa complexitat.

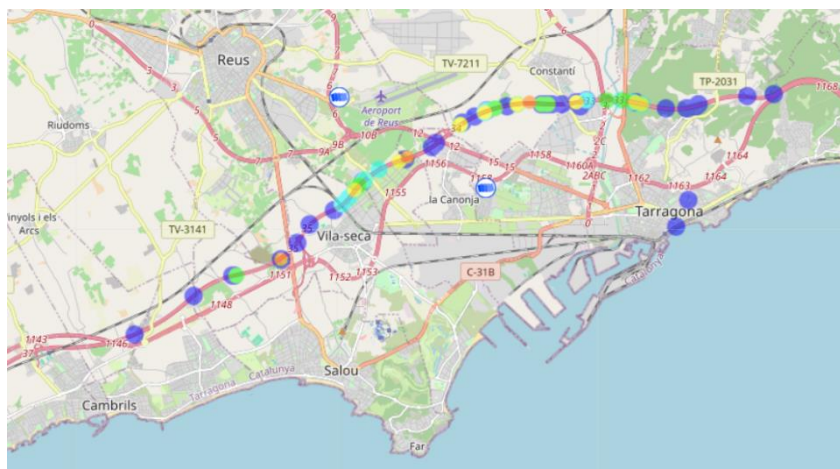


Figura 3.4. Cobertura de GW a nivell de Tarragona i voltants.

3.3.1 Donar d'alta un sensor a la xarxa TTN.

Un cop tenim el codi Arduino funcionant correctament [**Annexos**], amb els sensors pertinents, és hora de donar d'alta el dispositiu corresponent a la xarxa The Things Network. En el nostre treball, hem optat pel mecanisme d'activació OTAA.

En primer lloc, crearem la nostra aplicació, on podrem afegir els nostres dispositius:

³¹ The Things Network.

Create application

Within applications, you can register and manage end devices and their network data. After setting up your device fleet, use one of our many integration options to pass relevant data to your external services.
Learn more in our guide on [Adding Applications](#).

Application ID *

v01-test-app

Application name

Pol Culleré

Description

Description for my new application

Optional application description; can also be used to save notes about the application

Create application

Figura 3.5. Creació de l'aplicació al TTN.

Un cop creada l'aplicació, és hora de registrar el nostre dispositiu Heltec Cubecell AB01. En el nostre cas, Heltec Automation és una marca registrada al TTN, per tant el procediment és més senzill.

Register end device

Does your end device have a LoRaWAN® Device Identification QR Code? Scan it to speed up onboarding.

Scan end device QR code | [Device registration help](#)

End device type

Input method

Select the end device in the LoRaWAN Device Repository

Enter end device specifics manually

End device brand * | Model * | Hardware Ver. * | Firmware Ver. * | Profile (Region) *

HelTec AutoMation | HTCC-AB01(Class... | Unkno... | 1.0 | EU_863_870

HTCC-AB01(Class A OTAA)
LoRaWAN Specification 1.0.2, RP001 Regional Parameters 1.0.2 revision B, Over the air activation (OTAA), Class A

The Heltec HTCC-AB01 CubeCell - Dev-Board is a LoRaWAN® development board based on ASR605x (ASR6501, ASR6502). The ASR605x chip is integrated with the PSoc® 4000 series MCU (ARM® Cortex® M0+ Core) and SX1262. HTCC-AB01 allows connecting various sensors and supports the Arduino® development environment.

[Product website](#) | [Data sheet](#)

Frequency plan *

Europe 863-870 MHz (SF9 for RX2 - recommended)

Figura 3.6. Registre del nostre dispositiu al TTN.

Posteriorment, és hora d'afegir la descripció, la informació del clúster i les metadades, la configuració de la capa de xarxa LoRaWAN, el comportament i la sessió de la capa d'aplicació, les claus arrel i les configuracions de xarxa per l'activació del dispositiu final.

Basic
Description, cluster information and metadata

End device ID *
eui-70b3d57ed0066305

AppEUI *
00 00 00 00 00 00 00 00

DevEUI *
78 B3 D5 7E D8 06 63 05

End device name
My new end device

End device description
Optional end device description; can also be used to save notes about the

Network Server address
eu1.cloud.thethings.network

Application Server address
eu1.cloud.thethings.network

External Join Server
 Enabled

Join Server address
eu1.cloud.thethings.network

Attributes
+ Add attributes
Attributes can be used to set arbitrary information about the entity, to be used by scripts, or simply for your own organization

Network layer
LoRaWAN network-layer settings, behavior and session

Frequency plan *
Europe 863-870 MHz (SF9 for RX2 - recommended)

LoRaWAN version *
LoRaWAN Specification 1.0.2

Regional Parameters version *
RP001 Regional Parameters 1.0.2 revision B

LoRaWAN class capabilities
 Supports class B
 Supports class C

Activation mode *
 Over the air activation (OTAA)
 Activation by personalization (ABP)
 Define multicast group (ABP & Multicast)

Device address *
26 88 D1 E1

NwkSKey *
* * * * *

Application layer
Application-layer behavior and session

Application-layer behavior and session

Payload crypto override
 Use application default
 Enforce skipping payload crypto
 Enforce payload crypto

AppSKey *
* * * * *

Join settings

Root keys and network settings for end device activation

Home NEID
* * * * *
ID to identify the LoRaWAN network

Application Server ID
The AS-ID of the Application Server to use

Application Server KEK label
The KEK label of the Application Server to use for wrapping the application session key

Network Server KEK label
The KEK label of the Network Server to use for wrapping the network session key

Resets Join requests
 Enabled

Reset used DevEUIs

AppKey *
* * * * *

Figura 3.7. Configuració general del dispositiu Heltec al TTN.

Finalment, així es com es veuria el menú principal del TTN un cop donat d'alta el nostre dispositiu:

eui-70b3d57ed0066305
ID: eui-70b3d57ed0066305

↑ 148 ↓ n/a • Last activity 17 seconds ago

[Overview](#) [Live data](#) [Messaging](#) [Location](#) [Payload formatters](#) [General settings](#)

General information

End device ID: eui-70b3d57ed0066305

Frequency plan: Europe 863-870 MHz (SF9 for RX2 - recommended)

LoRaWAN version: LoRaWAN Specification 1.0.2

Regional Parameters version: RP001 Regional Parameters 1.0.2 revision B

Created at: Mar 25, 2024 13:17:34

Hardware

Brand: heltec

Model: cubecell-dev-board-class-a-otaa

Hardware version: _unknown_hw_version_

Firmware version: 1.0

Activation information

AppEUI: 00 00 00 00 00 00 00 00

DevEUI: 78 B3 D5 7E D8 06 63 05

AppKey: * * * * *

Session information

Session start: May 14, 2024 23:12:07

Device address: 26 88 D1 E1

NwkSKey: * * * * *

AppSKey: * * * * *

MAC data

Live data

00:03:48 Schedule data downLink for transmission on Gateway Server DevAddr: 26
00:03:48 Forward upLink data message DevAddr: 26 88 D1 E1 Payload: [0
00:03:48 Successfully processed data message DevAddr: 26 88 D1 E1
00:03:19 Schedule data downLink for transmission on Gateway Server DevAddr: 26
00:03:19 Forward upLink data message DevAddr: 26 88 D1 E1 Payload: [0
00:03:19 Successfully processed data message DevAddr: 26 88 D1 E1

Location

No location information available

Figura 3.8. Menú principal del dispositiu al TTN

Quan el dispositiu està donat d'alta, ja podem connectar-hi els sensors pertanyents. En aquest cas en concret, hem connectat el sensor BMP280 per fer la prova de funcionament. Per tant, podem observar el tràfic rebut pels sensors a TTN. En aquest cas, podem veure la temperatura i la pressió:

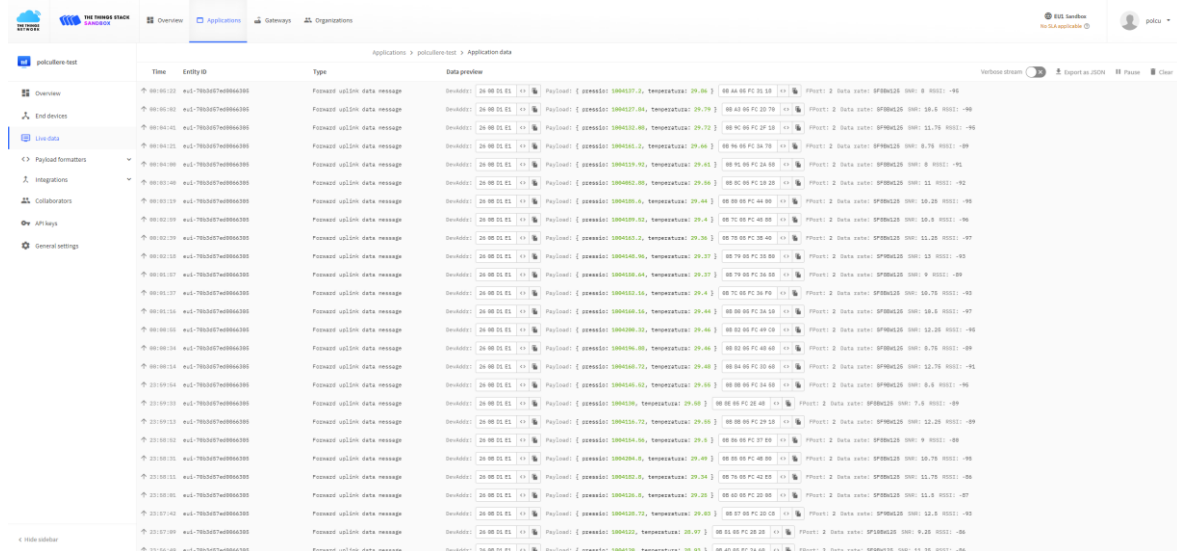


Figura 3.9. Live Data del tràfic rebuts dels sensors a TTN.

No obstant, per tal de poder observar la temperatura i pressió en format decimal, s'ha de modificar el *payload formatter*³², per tal de transformar les dades hexadecimals que rep del sensor BMP280, en dades amb valors numèrics. Independentment del sensor, sempre s'haurà de modificar el *payload formatter*, per tal de convertir les dades hexadecimals que rep del sensor en dades numèriques referents al sensor que s'estigui usant.

Uplink Downlink

Setup

Formatter type*

Custom Javascript formatter

Formatter code*

```

1 function decodeUplink(input) {
2   var temp = (input.bytes[0] << 8) | input.bytes[1];
3   var pressRaw = (input.bytes[2] << 24) | (input.bytes[3] << 16) | (input.bytes[4] << 8)
4   | (input.bytes[5]);
5   |
6   if (temp > 32767) { // Convertir de complement a dos per valors negatius
7     temp = temp - 65536;
8   }
9
10  var temperature = temp / 100; // Convertir de nou a valor amb decimals
11
12  var pressure = pressRaw / 100.0; // Toxna a convertir a Pascales amb dos decimals
13
14  return {
15    data: {
16      temperatura: temperature,
17      pressio: pressure
18    }
19  };
20 }
21
22

```

Figura 3.10. Payload Formatter en format Javascript.

³² Una funció o un conjunt d'eines que s'utilitza per formatar i processar les dades (payload) que es transmeten entre dispositius en una xarxa de comunicació.

3.3.2 Donar d'alta un gateway a la xarxa TTN.

En el nostre cas, vam decidir instal·lar un *gateway Mikrotik wAP LR8 kit*, per tal de tenir sempre cobertura i així poder enviar dades constantment dels dispositius al TTN. No obstant, vam haver de seguir un parell de passos per tal de dur a terme la instal·lació del *gateway*.

En primer lloc, vam endollar el *gateway* i vam connectar un cable Ethernet des d'un dels ports LAN del nostre router cap a l'entrada Ethernet del nostre *gateway*.



Figura 3.11. Connexió del gateway al router via cable Ethernet.

Un cop connectat al router i a la font d'energia, ens connectem a la xarxa Wifi del equip MikroTik-6D9687 i des d'un navegador ingressem al lloc web <http://192.168.88.1/> per començar amb la configuració de l'equip.

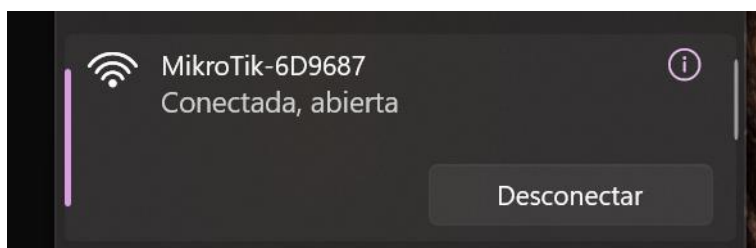
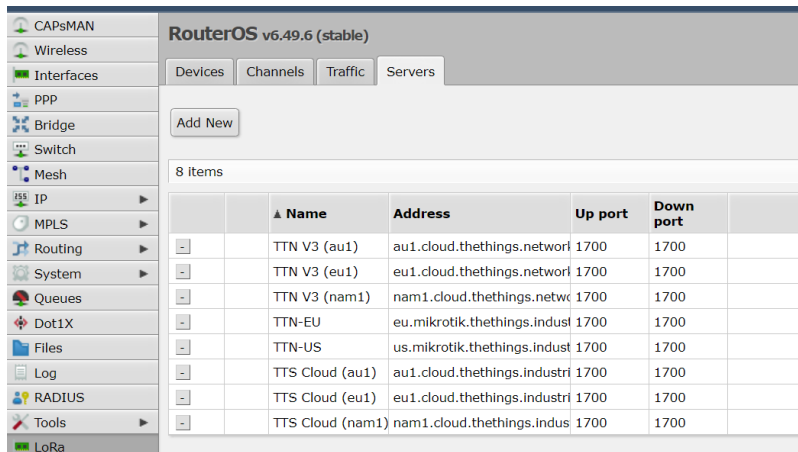


Figura 3.12. Connexió a la xarxa Wifi del gateway.

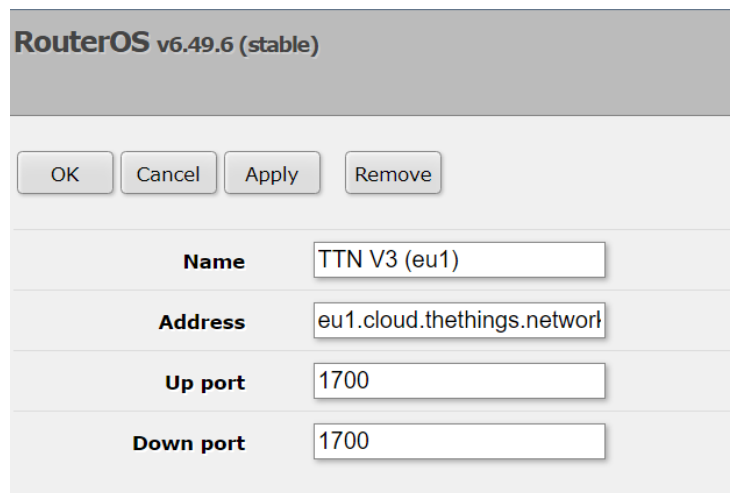
Un cop hem obert el lloc web, al menú de l'esquerra, seleccionem **Lora** i seleccionem **Servers** a la part superior dreta.



	Name	Address	Up port	Down port
-	TTN V3 (au1)	au1.cloud.thethings.network	1700	1700
-	TTN V3 (eu1)	eu1.cloud.thethings.network	1700	1700
-	TTN V3 (nam1)	nam1.cloud.thethings.network	1700	1700
-	TTN-EU	eu.mikrotik.thethings.indust	1700	1700
-	TTN-US	us.mikrotik.thethings.indust	1700	1700
-	TTS Cloud (au1)	au1.cloud.thethings.industri	1700	1700
-	TTS Cloud (eu1)	eu1.cloud.thethings.industri	1700	1700
-	TTS Cloud (nam1)	nam1.cloud.thethings.indus	1700	1700

Figura 3.13. Llistat de servidors del gateway.

En el nostre cas, com que el servidor ens el va deixar el professor, ja està el servidor creat, i seria el **TTN v3 (eu1)**. Els seus paràmetres serien els següents per si haguéssim de crear-lo de nou:



RouterOS v6.49.6 (stable)

OK Cancel Apply Remove

Name TTN V3 (eu1)

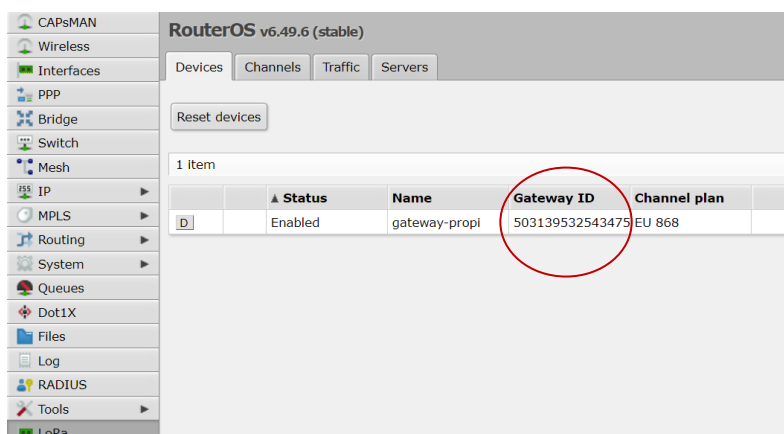
Address eu1.cloud.thethings.network

Up port 1700

Down port 1700

Figura 3.14. Paràmetres del servidor TTN V3.

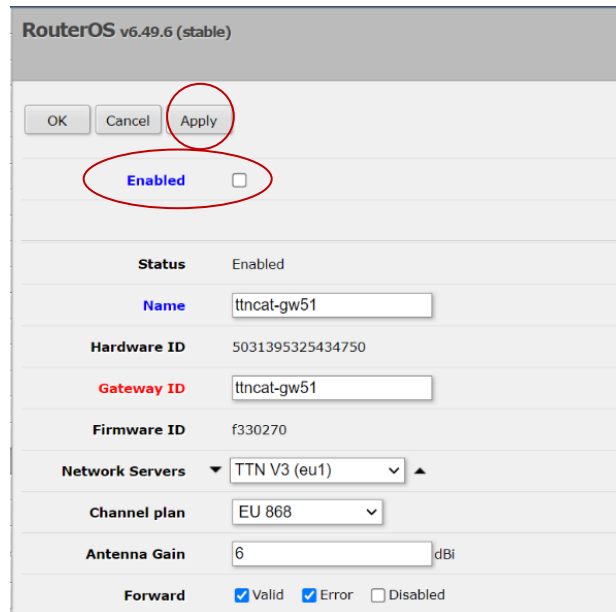
Posteriorment, al menú superior, seleccionarem la pestanya **Devices** i farem clic a l'**ID de la passarel·la** per configurar-lo:



	Status	Name	Gateway ID	Channel plan
D	Enabled	gateway-propi	503139532543475	EU 868

Figura 3.15. Pestanya per veure els dispositius connectats.

Finalment, un cop hem entrat dins de la configuració dels dispositius, ajustem els paràmetres adients al nostre *gateway* i fem clic a **apply** i **enabled** un cop hem acabat la configuració:



RouterOS v6.49.6 (stable)

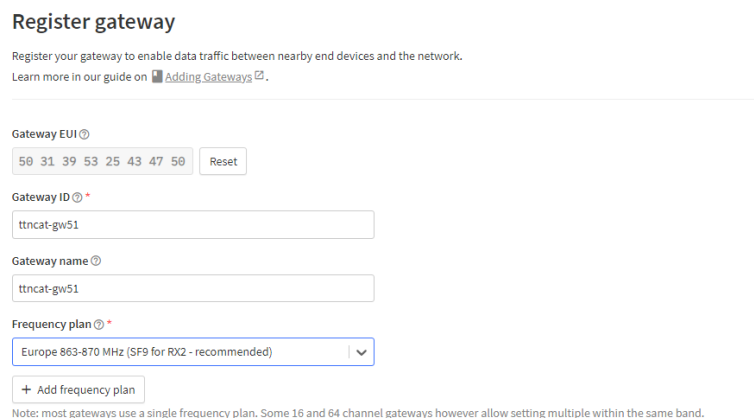
OK Cancel **Apply**

Enabled

Status	Enabled
Name	ttncat-gw51
Hardware ID	5031395325434750
Gateway ID	ttncat-gw51
Firmware ID	f330270
Network Servers	TTN V3 (eu1)
Channel plan	EU 868
Antenna Gain	6 dBi
Forward	<input checked="" type="checkbox"/> Valid <input checked="" type="checkbox"/> Error <input type="checkbox"/> Disabled

Figura 3.16. Configuració del nostre gateway.

Finalment, un cop hem configurat tots els paràmetres adients al nostre *gateway*, és el moment de registrar-lo a la TTN. El primer que farem és posar el *Gateway EUI* que marca el fabricant. Seguidament, afegirem el ID que vulguem, en el nostre cas, per tal de seguir un ordre amb els diferents *gateways* registrats per la XOIC³³, vam optar per posar **ttncat-gw51**. Per acabar, afegirem la freqüència de 863-870 MHz, que és la que es correspon a Europa.



Register gateway

Register your gateway to enable data traffic between nearby end devices and the network.
Learn more in our guide on [Adding Gateways](#).

Gateway EUI

50 31 39 53 25 43 47 50

Gateway ID *

ttncat-gw51

Gateway name

ttncat-gw51

Frequency plan *

Europe 863-870 MHz (SF9 for RX2 - recommended)

Note: most gateways use a single frequency plan. Some 16 and 64 channel gateways however allow setting multiple within the same band.

Figura 3.17. Registre del gateway a la TTN.

Un cop dona't d'alta el *gateway* a la TTN, cada cop que s'envia una dada des del nostre dispositiu a la TTN, el gateway mostra diferents paràmetres, com per exemple la DevAddr, l'estat del *gateway*, el missatge de càrrega i el descàrrega... Bàsicament, seria el live data del MikroTik.

³³ Xarxa Oberta per la Internet de les Coses.

Time	Type	Data preview
11:56:57	Send downlink message	Tx Power: 16.15 Data rate: SF9BW125
11:56:56	Receive uplink message	DevAddr: 26 00 64 84 FCnt: 4 FPort: 2 Data rate: SF9BW125 SNR: 12 RSSI: -85
11:56:52	Receive gateway status	Metrics: { rxin: 1, rxok: 1, rxfw: 1, ackr: 100, txin: 1, txok: 1 } Versions: { ttn-lw-gateway-server: "3.30.0-rc0-SNAPSHOT-2f391bec0a" }

Figura 3.18. Live data del nostre gateway.

A continuació, ja podem veure quin *gateway* ha capturat el nostre paquet. Això és útil ja que de vegades hi ha més d'un *gateway* que pot arribar a capturar el teu paquet. Com podem veure a la següent imatge, el nostre paquet ha estat capturat pel **ttncat-gw51**, ni més ni menys que el nostre *gateway*. Aquestes dades proporcionen una visió completa de com es transmet i processa un missatge de pujada en una xarxa LoRaWAN gestionada per TTN, permetent fer un seguiment de la qualitat de la comunicació i dels dispositius involucrats.

```

"received_at": "2024-05-14T23:35:53.645143894Z",
"uplink_message": {
  "session_key_id": "AY948ptZdPNZxPhai5+AQQ==",
  "f_port": 2,
  "f_cnt": 416,
  "frm_payload": "CkIF/BVA",
  "decoded_payload": {
    "pressio": 1004065.92,
    "temperatura": 26.26
  },
},
"rx_metadata": [
  {
    "gateway_ids": {
      "gateway_id": "ttncat-gw51",
      "eui": "5031395325434750"
    },
    "time": "2024-05-14T23:35:53.122676Z",
    "timestamp": 3617536324,
    "rssi": -83,
    "channel_rssi": -83,
    "snr": 11.75,
    "uplink_token": "ChkKFwoLdHRuY2F0LWd3NTE5CFAX0VMlQ0dQEMTY",
    "channel_index": 1,
    "received_at": "2024-05-14T23:35:53.410625271Z"
  }
]

```

Figura 3.19. Paquet capturat pel nostre gateway.

3.4 Configuració de l'IDE d'Arduino.

En aquesta secció, s'indicaran els passos específics per la correcta configuració del nostre entorn de desenvolupament pel dispositiu Heltec Cubecell AB01. A continuació, detallaré pas a pas la configuració de l'IDE d'Arduino:

1. Obrirem l'IDE d'Arduino, i farem clic a la pestanya **Arxiu-->** i després clicarem a **Preferències**.

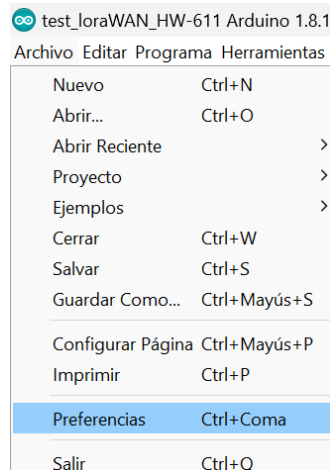


Figura 3.20. Inici de la configuració del IDE.

2. A la finestra de **Preferències** haurem de pegar la URL de la biblioteca de gestor de targetes per tal de poder treballar amb dispositius Heltec Cubecell d'Arduino, la següent direcció és:

http://resource.heltec.cn/download/package_CubeCell_index.json

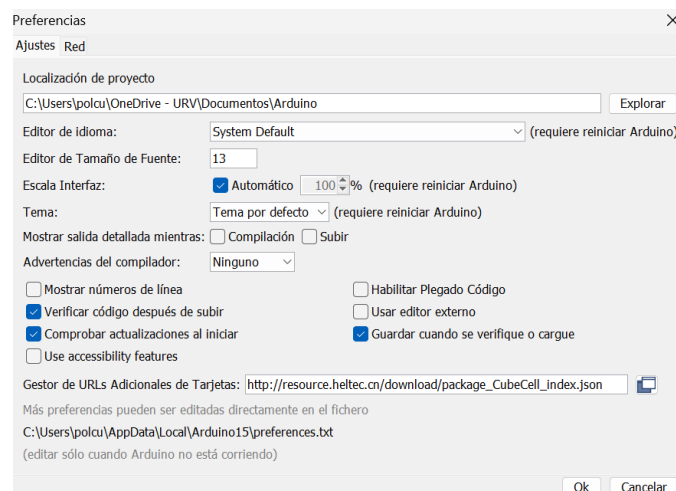


Figura 3.21. Configuració de la biblioteca del gestor de targeta Cubecell.

En la finestra **Eines-->**, buscarem el menú d'**Administrador de Biblioteques** i procedirem a instal·lar la llibreria **LualtekCubecell**:

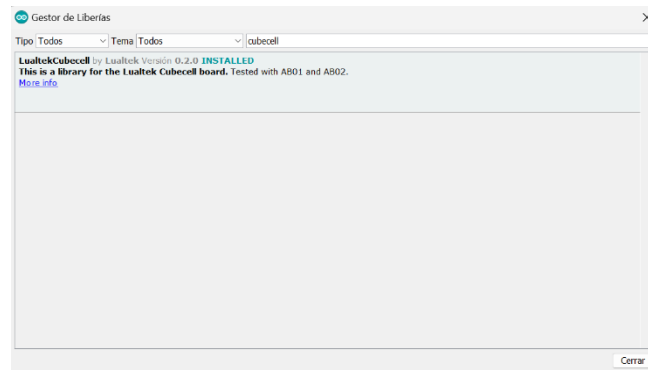


Figura 3.22. Instal·lació correcta de la llibreria.

Per últim, ens dirigim al menú **Eines-->** i podem observar que la placa Heltec Cubecell ja està instal·lada i configurada correctament.

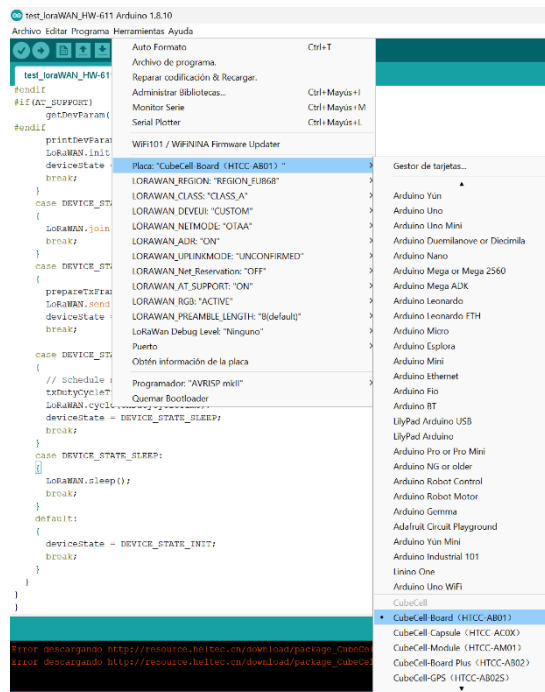


Figura 3.23. Configuració correcta del IDE Arduino.

4. CONFIGURACIÓ DEL SERVIDOR AWS

En aquest apartat, detallaré el procés de configuració d'un servidor en el núvol utilitzant Amazon Web Services. AWS es tracta d'una plataforma de serveis al núvol proporcionada per la companyia Amazon, la qual ofereix una àmplia gamma de serveis de computació, emmagatzematge, base de dades i altres funcionalitats les quals permeten desplegar aplicacions de forma escalable i segura.

Per al meu projecte en concret, configurarem una instància EC2 (*Elastic Compute Cloud*), per tal de funcionar com a servidor centralitzat el qual gestionarà les dades dels sensors de temperatura, pressió, CO2, TVOC i GPS. Aquesta instància estarà configurada per tal d'executar una màquina virtual amb Ubuntu, on instal·laré i configuraré les eines que facin falta per tal de rebre, emmagatzemar i visualitzar les dades enviades pels sensors.

4.1 Creació d'una instància EC2 a AWS.

Aquest apartat detalla els passos necessaris per tal de crear i configurar una instància EC2 a Amazon Web Services. Una instància EC2 és bàsicament una màquina virtual que proporciona la capacitat de computació necessària per tal d'executar aplicacions al núvol.

4.1.1 Selecció del tipus de màquina.

En primer lloc, caldrà accedir al portal d'AWS i iniciar sessió amb les credencials corresponents. Posteriorment, haurem d'anar al panell de EC2 i seleccionar **Llançar una instància**.

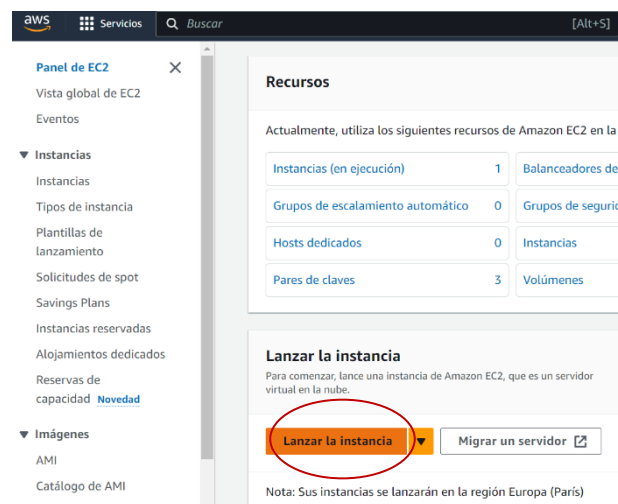


Figura 4.1. Llançament de la instància EC2.

Posteriorment, haurem de triar una AMI que contingui el sistema operatiu desitjat. En el nostre cas, seleccionarem “Ubuntu Server 20.04 LTS (HVM), SSD Volume Type”.

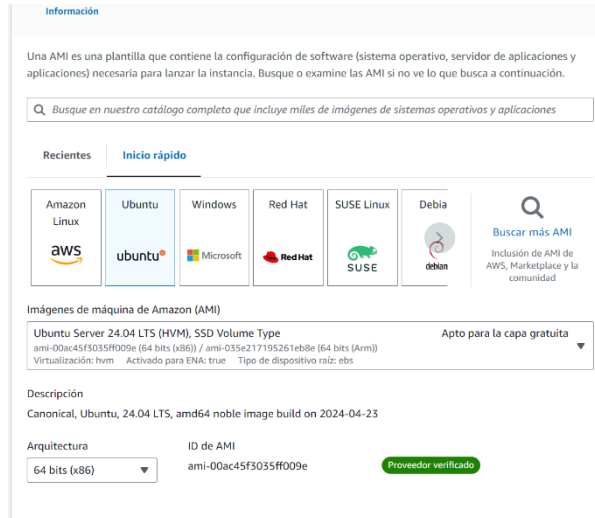


Figura 4.2. Selecció de la màquina virtual.

Finalment, escollirem el tipus d'instància en funció de les necessitats de recursos. Al tractar-se d'un projecte que no requereix d'un ús avançat de recursos, el tipus "t2.micro" és suficient.



Figura 4.3. Selecció del tipus d'instància.

4.1.2 Configuració de les claus d'accés.

És important la creació de les claus d'accés, per a posteriori poder accedir a la teva instància via SSH. En el nostre cas, n'hi ha 3 de creades ja que les primeres em van servir per anar fent proves, però amb una clau és suficient.

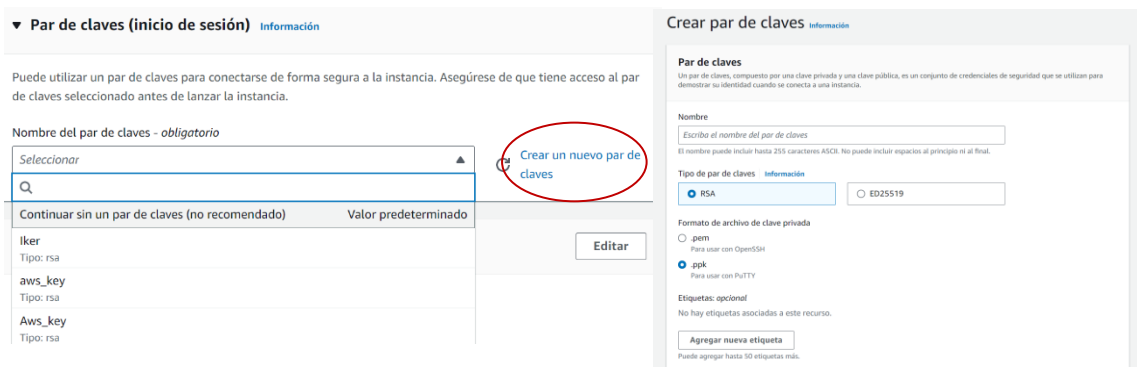


Figura 4.4. Creació de les claus d'accés.

4.1.3 Configuració de seguretat (ports oberts).

En aquest pas, és important crear diferents grups de seguretat per tal de permetre l'accés a la instància des del Node-Red, MQTT, Grafana i SSH. Per exemple, el SSH està associat al port 22, el HTTP està associat al port 80 i serveix per fer servir aplicacions web, el Grafana està associat al port 3000 i el Node-red està associat al port 1880, entre d'altres.

Nombre del grupo de seguridad	ID de grupo de seguridad	Tipo	Protocolo	Intervalo de puertos	Origen	Descripción
default	sg-03160cea7d363bcb6	Todo el tráfico	Todo	Todo	sg-03160cea7d363bcb6	-
grafana	sg-01eb468936721dfc7	TCP personalizado	tcp	1806	0.0.0.0/0	-
grafana	sg-01eb468936721dfc7	HTTP	tcp	80	0.0.0.0/0	-
grafana	sg-01eb468936721dfc7	ssh	tcp	22	0.0.0.0/0	-
grafana	sg-01eb468936721dfc7	TCP personalizado	tcp	8086	0.0.0.0/0	-
grafana	sg-01eb468936721dfc7	TCP personalizado	tcp	3000	0.0.0.0/0	-
launch-wizard-1	sg-03c96721d4ff08951	ssh	tcp	22	0.0.0.0/0	-
node-red	sg-0c42ac97869565ff7	TCP personalizado	tcp	1880	0.0.0.0/0	-

Figura 4.5. Llista dels diferents grup de seguretat.

4.2 Connexió a la màquina virtual amb PuTTY.

Un cop creada la instància EC2 a AWS, el pròxim pas és connectar-se a ella per tal de poder configurar el SO³⁴ i les aplicacions que facin falta. És per aquest motiu, que generalment s'usa el protocol SSH. PuTTY és un client popular per a sistemes Windows que permet aquest procés.

4.2.1 Conversió de la clau PEM a PPK.

AWS proporciona les claus d'accés en format PEM. PuTTY et demana que aquestes claus estigui en format PPK. És per això, que es pot fer ús de l'eina PuTTYGen, la qual ve instal·lada amb PuTTY. Per tant, abans de tot hem de seguir els següents passos:

1. Descarreguem la versió 64-bit x86.

Package files

You probably want one of these. They include versions of all the PuTTY utilities (except the new and slightly experimental Windows pterm).
(Not sure whether you want the 32-bit or the 64-bit version? Read the [FAQ entry](#).)

We also publish the latest PuTTY installers for all Windows architectures as a free-of-charge download at the [Microsoft Store](#); they usually take a few days to appear there after we release them.

MSI ("Windows Installer")

64-bit x86: [putty-64bit-0.80-installer.msi](#) ([signature](#))
 64-bit Arm: [putty-arm64-0.80-installer.msi](#) ([signature](#))
 32-bit x86: [putty-0.80-installer.msi](#) ([signature](#))

Unix source archive

.tar.gz: [putty-0.80.tar.gz](#) ([signature](#))

Figura 4.6. Fitxers d'instal·lació PuTTY.

³⁴ Sistema operatiu.

- Obrim PuTTY Key Generator i fem clic al botó **“Load”**. Posteriorment, seleccionem el fitxer PEM que vam descarregar quan vam crear la clau d'accés. Un cop carregada la clau, fem clic al botó **“Save private key”** i dessem el fitxer amb extensió .ppk. Aquest serà el fitxer que PuTTY usarà per autenticar la connexió SSH.

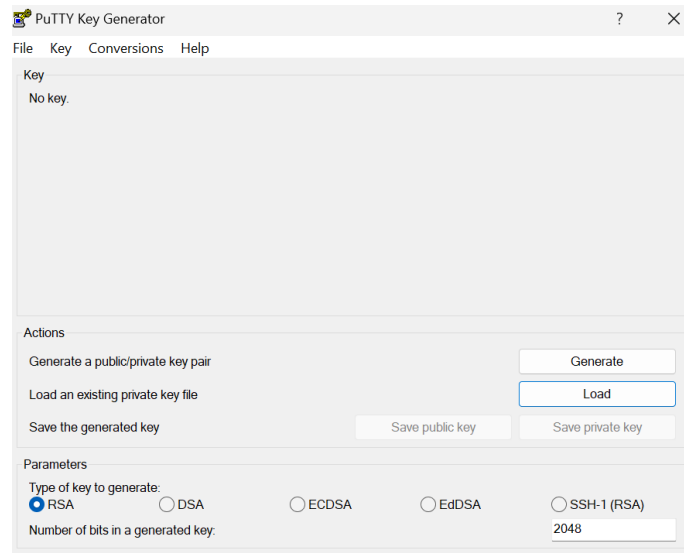


Figura 4.7. PuTTY Key Generator.

4.2.2 Connexió via SSH.

Un cop ja tenim creada la clau en format .ppk, ja podem seguir els següents passos per connectar-nos a la instància EC2. En primer lloc, observem els resum de la nostra instància EC2, on es pot observar la direcció IPv4 pública i privada del nostre servidor, el DNS³⁵ de la IPv4 pública....

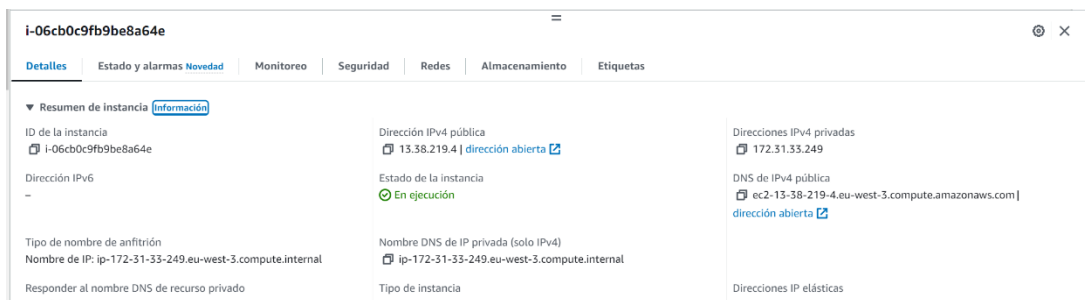


Figura 4.8. Resum de la instància EC2.

³⁵ Domain Name System.

Ara comença la configuració del nostre PuTTY per tal de poder accedir a la nostra instància a través d'aquest:

1. Afegim el DNS de la IPv4 pública i els detalls de connexió. Confirmem que el Port està configurat a 22 i que el tipus de connexió SSH està seleccionat.

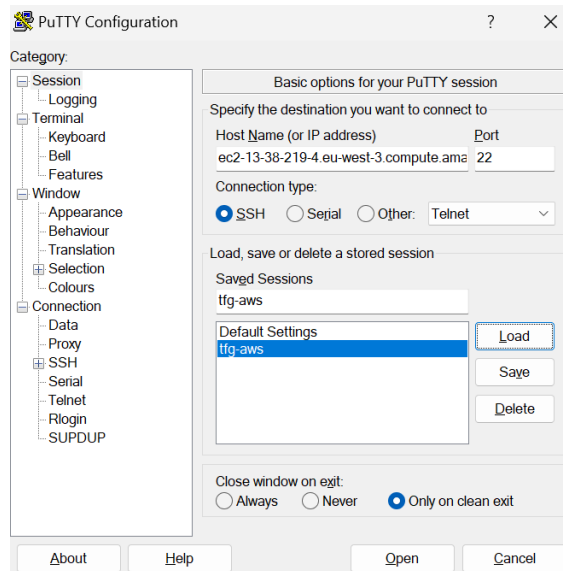


Figura 4.9. Configuració del PuTTY.

2. Verifiquem el protocol SSH. Fem clic a **Connection-->SSH** i ens assegurem que s'hagi seleccionat el número 2 per a la versió del protocol SSH.

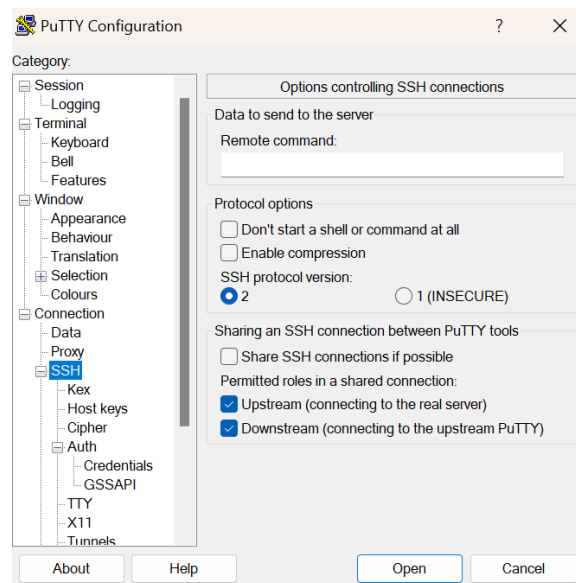


Figura 4.10. Configuració de la connexió SSH.

- Amb la clau SSH creada anteriorment i guardada al nostre PC local, busquem la secció de **Auth-->Credentials** i allà on posa **Browse**, busquem el nostre fitxer de clau privada acabat amb l'extensió **.ppk**.

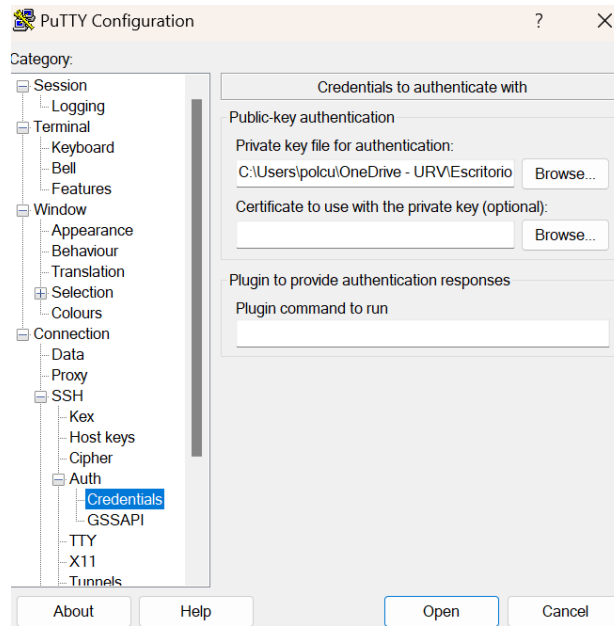


Figura 4.11. Credencials per autenticar l'accés.

- A l'apartat de **Connection-->Data**, introduïm el nom d'usuari del nostre servidor. Per a la configuració inicial, aquest hauria de ser **root**, que és l'usuari administratiu del nostre servidor. Posteriorment, podrem afegir el nostre usuari personalitzat, que en el nostre cas és **Ubuntu**.

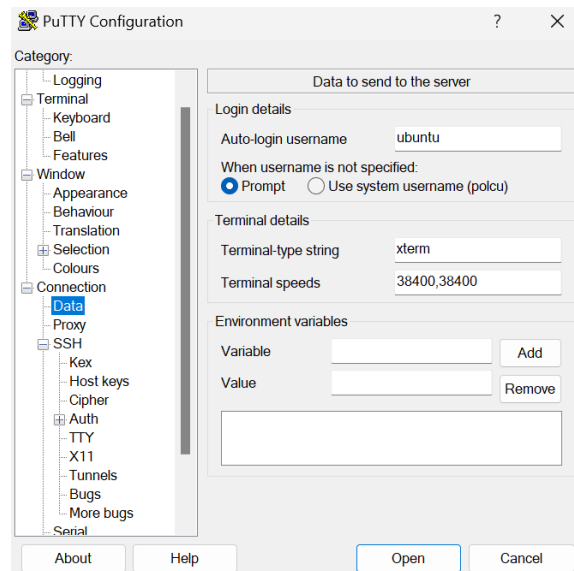


Figura 4.12. Configuració del nom d'usuari.

5. Finalment, podem desar aquestes preferències per evitar escriure-les manualment en el futur. Fem clic a **Session** i a continuació, afegim un nom al quadre de text sota **Saved Sessions** i fem clic a **Save**.

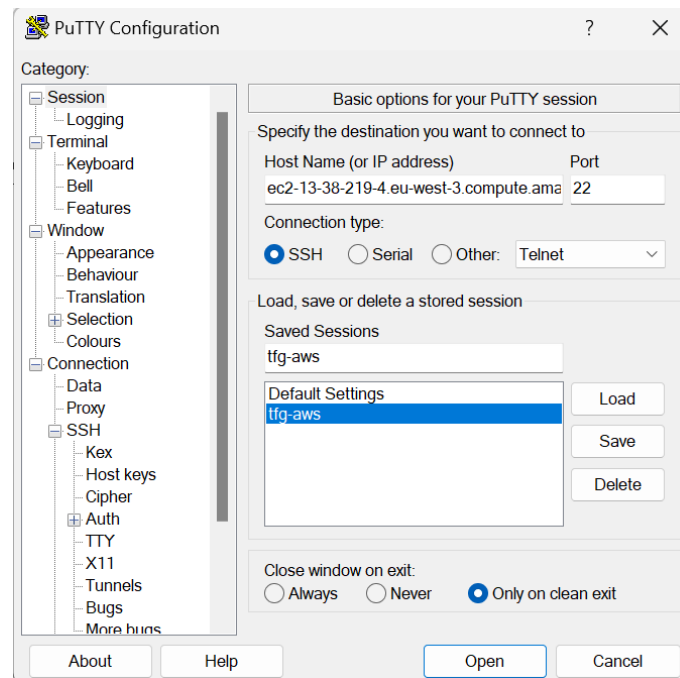


Figura 4.13. Guardem les preferències de la sessió PuTTY.

6. Un cop carregades les preferències, fem clic a **Open** per tal de connectar-nos a la nostra instància EC2. Ja hem accedit a la nostra màquina virtual de AWS amb l'usuari Ubuntu i ja la podem usar sense problema.

```

Using username "ubuntu".
Authenticating with public key "imported-openssh-key" from agent
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1016-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Mon Apr 15 13:49:50 UTC 2024

System load:  0.0          Processes:    104
Usage of /:   50.3% of 7.57GB   Users logged in:  1
Memory usage: 40%          IPv4 address for eth0: 172.31.33.249
Swap usage:  0%

=> There is 1 zombie process.

 * Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

  https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

11 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

1 additional security update can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Thu May 16 15:06:14 2024 from 84.88.201.56
ubuntu@ip-172-31-33-249:~$

```

Figura 4.14. Màquina virtual Ubuntu corrent la instància EC2.

5. INSTAL·LACIÓ DE NODE-RED, INFLUXDB I GRAFANA A LA MÀQUINA VIRTUAL

Node-Red és una plataforma de desenvolupament basada en el navegador que permet connectar diferents dispositius de hardware, APIs i serveis online d'una manera visual i intuïtiva. Mitjançant un editor de fluxos, es possible crear aplicacions complexes mitjançant blocs senzills que s'interconnecten entre ells.

InfluxDB és una base de dades especialitzada en guardar i processar moltes dades produïdes per sensors, sistemes de monitoratge, aplicacions IoT, etc. Té una gran capacitat d'escalabilitat i és molt eficient en l'accés a les dades.

Grafana és una eina de visualització de dades de codi obert que permet crear panells interactius per monitorar i analitzar dades en temps real. Grafana s'adapta sense dificultat a InfluxDB i altres fonts de dades per oferir visualitzacions personalitzables i detallades.

5.1 Instal·lació de Node-RED.

Abans d'instal·lar Node-RED, el més recomanable és actualitzar la llista de paquets del sistema i actualitzar els paquets instal·lats.

```
sudo apt upgrade -y
sudo apt update
```

Figura 5.1. Comandes per actualitzar els paquets.

Node-RED es basa en Node.js, és per això que cal instal·lar Node.js abans de continuar amb la instal·lació de Node-RED. Per tant, la següent comanda és la que instal·la el Node.js:

```
curl -sL https://deb.nodesource.com/setup_14.x | sudo -E bash -
sudo apt install -y nodejs
```

Figura 5.2. Comandes per instal·lar el Node.js

Un cop ja tenim instal·lat el Node.js, ja podem instal·lar el Node-RED de forma global utilitzant npm³⁶. La comanda és:

³⁶ Gestor de paquets de Node.js.

```
sudo npm install -g --unsafe-perm node-red
```

Figura 5.3. Comanda per instal·lar el npm.

Finalment, per assegurar-nos que Node-RED s’ha instal·lat de forma correcta, haurem d’executar la següent comanda:

```
node-red
```

Figura 5.4. Comanda per instal·lar de forma correcta Node-RED.

Un cop realitzats tots aquests passos, s’haurà iniciat Node-RED i es mostrarà la URL on podrem accedir a la interfície web:

```
13.38.219.4:1880/#flow/574f54071ef66436
```

Figura 5.5. URL per accedir al Node-RED.

5.1.1 Configuració inicial i posada en marxa.

És molt important configurar correctament el Node-RED per tal de que s’iniciï de forma automàtica quan obrim la nostra màquina virtual. Això es pot aconseguir creant un servei de systemd. Per tant, cal crear un fitxer de servei per a Node-RED:

```
sudo nano /lib/systemd/system/nodered.service
```

Figura 5.6. Comanda per crear un fitxer de servei.

A continuació hem d’editar el fitxer de servei i afegir el següent contingut al fitxer:

```
[Unit]
Description=Node-RED
After=syslog.target network.target

[Service] ExecStart=/usr/bin/env node-red-pi
ExecStop=/bin/kill -SIGINT $MAINPID
Restart=on-failure
KillMode=process

[Install]
WantedBy=multi-user.target
```

Finalment, després de crear el fitxer de servei, cal activar i iniciar el servei Node-RED.

```
sudo systemctl enable nodered.service
sudo systemctl start nodered.service
```

Figura 5.7. Comandes per activar i iniciar el Node-RED.

En termes generals, configurar Node-RED és fàcil, només cal instal·lar Node.js, descarregar Node-RED amb npm i ajustar el sistema per tal de que Node-RED s'iniciï automàticament. Un cop complerts tots aquests passos, ja podem accedir a la interfície web de Node-RED per tal de començar a crear els diferents fluxos i connectar dispositius i serveis de forma visual i intuïtiva.

5.2 Instal·lació d'InfluxDB.

A continuació es mostren els diferents passos per tal de d'instal·lar i configurar InfluxDB, una part essencial del nostre projecte, si volem representar les dades de forma correcta.

El primer pas és importar la clau pública utilitzada pel sistema de paquets:

```
wget -qO- https://repos.influxdata.com/influxdb.key | sudo apt-key add -
```

Figura 5.8. Clau pública del sistema de paquets.

Posteriorment, hem d'afegir el repositori APT d'InfluxDB:

```
source /etc/lsb-release
echo "deb https://repos.influxdata.com/${DISTRIB_ID,,} ${DISTRIB_CODENAME} stable"
```

Figura 5.9. Comandes per afegir el repositori APT.

A continuació, hem d'actualitzar el sistema de paquets i instal·lar InfluxDB: 1

```
sudo apt update
sudo apt install influxdb -y
```

Figura 5.10. Comandes per actualitzar el sistema i instal·lar InfluxDB.

Un cop hem actualitzat el sistema de paquets, iniciarem el servei d'InfluxDB:

```
sudo systemctl start influxdb
sudo systemctl enable influxdb
```

Figura 5.11. Comandes per iniciar el servei d'InfluxDB.

Finalment, verificarem que InfluxDB estigui funcionant correctament:

```
ubuntu@ip-172-31-33-249:~$ sudo systemctl status influxdb
● influxdb.service - InfluxDB is an open-source, distributed, time series database
   Loaded: loaded (/lib/systemd/system/influxdb.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2024-05-08 11:24:49 UTC; 1 week 4 days ago
     Docs: https://docs.influxdata.com/influxdb/
   Main PID: 368 (influxd)
    Tasks: 10 (limit: 1121)
   Memory: 91.3M
      CPU: 2h 37min 43.445s
   CGroup: /system.slice/influxdb.service
           └─368 /usr/bin/influxd -config /etc/influxdb/influxdb.conf
```

Figura 5.12. Estatus del servei d'InfluxDB

Com podem observar, ens surt que la base de dades està *active*, i per tant està corrent correctament.

5.2.1 Configuració de la base de dades.

A continuació configurarem la base de dades per tal de poder recollir les dades que anem rebent a través del sensor.

En primer lloc accedirem a la consola d'InfluxDB:

```
influx
```

Figura 5.13. Comanda per accedir a la consola d'InfluxDB.

A continuació, crearem una base de dades nova per tal de poder accedir-hi posteriorment:

```
CREATE DATABASE TFG_base_dades
```

Figura 5.14. Comanda per crear la base de dades.

La següent comanda ens mostra les diferents bases disponibles:

```
> SHOW DATABASES
name: TFG_base_dades
>
```

Figura 5.15. Llista de bases de dades creades.

Finalment, ja podem usar la base de dades creada anteriorment:

```
> USE TFG_base_dades
Using database TFG_base_dades
```

Figura 5.16. Comanda per usar la base de dades creada.

Un cop hem entrat a la nostra base de dades, ja podem accedir a les dades que anem emmagatzemant contínuament de forma regular:

```
> SELECT * FROM TFG_base_dades
```

Figura 5.17. Comanda per accedir a les dades de la nostra DB.

5.2.2 Creació d'usuaris.

És important crear un usuari amb permisos d'administrador. En el meu cas, poso aquest com a exemple per privacitat:

```
CREATE USER admin WITH PASSWORD 'password' WITH ALL PRIVILEGES
```

Figura 5.18. Comanda per crear un usuari amb permisos d'administrador.

Aquesta seria la llista d'usuaris associats a aquest projecte:

```
> SHOW users
pol true
```

Figura 5.19. Llistat d'usuaris associades a la base de dades.

5.3 Instal·lació de Grafana.

A continuació es mostren els diferents passos per tal de d'instal·lar i configurar Grafana, la part del treball més visual i on es veu reflectit tot el treball que hi ha darrere.

El primer pas és afegir el repositori de Grafana:

```
sudo apt-get install -y software-properties-common
sudo add-apt-repository "deb https://packages.grafana.com/oss/deb stable main"
```

Figura 5.20. Comandes per afegir el repositori APT.

Posteriorment, hem d'afegir la clau GPG³⁷ de Grafana:

```
wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -
```

Figura 5.21. Comanda per afegir la clau GPG.

A continuació, hem d'actualitzar el sistema de paquets i instal·lar Grafana:

```
sudo apt-get update
sudo apt-get install grafana -y
```

Figura 5.22. Comandes per actualitzar el sistema i instal·lar Grafana.

Un cop hem actualitzat el sistema de paquets, iniciarem el servei de Grafana:

```
sudo systemctl start grafana-server
sudo systemctl enable grafana-server
```

Figura 5.23. Comandes per iniciar el servei de Grafana.

³⁷ GNU Privacy Guard.

Finalment, verificarem que Grafana estigui funcionant correctament:

```
buntu@ip-172-31-33-249:~$ sudo systemctl status grafana-server
● grafana-server.service - Grafana instance
   Loaded: loaded (/lib/systemd/system/grafana-server.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2024-05-08 11:24:49 UTC; 1 week 4 days ago
     Docs: http://docs.grafana.org
    Main PID: 370 (grafana)
      Tasks: 15 (limit: 1121)
     Memory: 91.6M
        CPU: 1h 36min 29.686s
    CGroup: /system.slice/grafana-server.service
            └─370 /usr/share/grafana/bin/grafana server --config=/etc/grafana/grafana.ini --pidfile=
```

Figura 5.24. Estatus del servei de Grafana.

Com podem observar, ens surt que la instància de Grafana està *active*, i per tant està corrent correctament.

5.3.1 Configuració inicial i posada en marxa.

Un cop hem realitzat la instal·lació de Grafana, és moment de d'accedir a la interfície web de Grafana. Obrirem el navegador i ens dirigirem a **http://13.38.219.4:3000/**.

Posteriorment, ens demanarà iniciar sessió amb les credencials predeterminades. Com que es tracta del primer cop que accedim, l'usuari i la contrasenya seran **admin**.

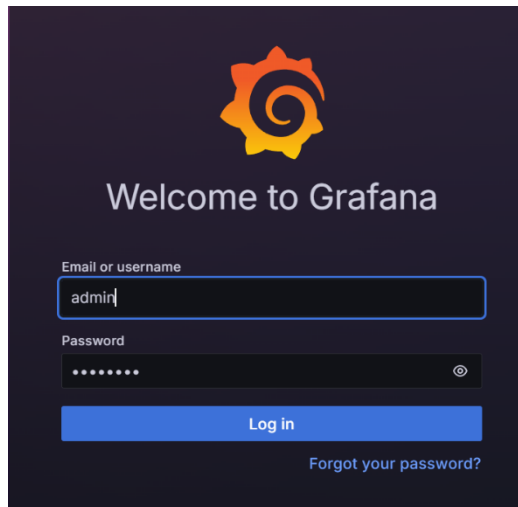


Figura 5.25. Inici de sessió a Grafana.

Finalment, posarem en marxa la connexió amb InfluxDB. No obstant, en aquest apartat tan sols veurem la configuració del *Data Source*, ja que posteriorment veurem la configuració dels panells i els diferents gràfics.

Per tant, obrim el desplegable que ens ofereix Grafana i busquem a **Connections-->Data Sources**.

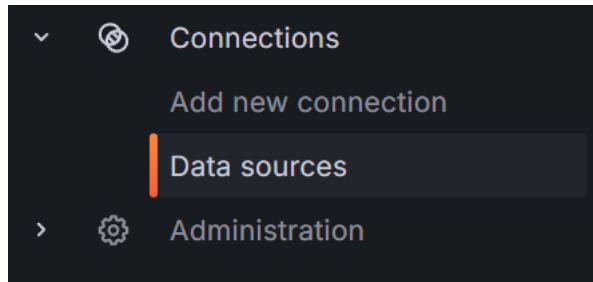


Figura 5.26. Desplegable de Connections dins de Grafana.

Un cop estem dins de *Data Sources*, de totes les bases de dades disponibles, seleccionem InfluxDB:

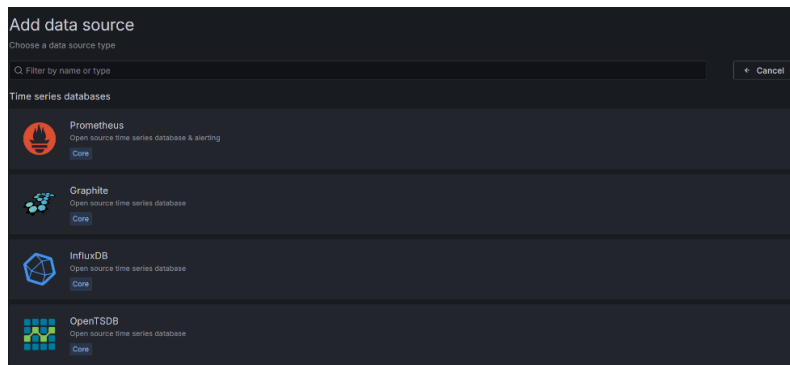


Figura 5.27. Llista de bases de dades disponibles per vincular.

Finalment, configurem els detalls de la connexió, com ara el nom de la nostra base dades que hem creat anteriorment, la URL amb la qual accedim a la interfície web, però en comptes del port 3000 s'usa el port 8086, el qual és el port per defecte utilitzat per InfluxDB per acceptar connexions HTTP.

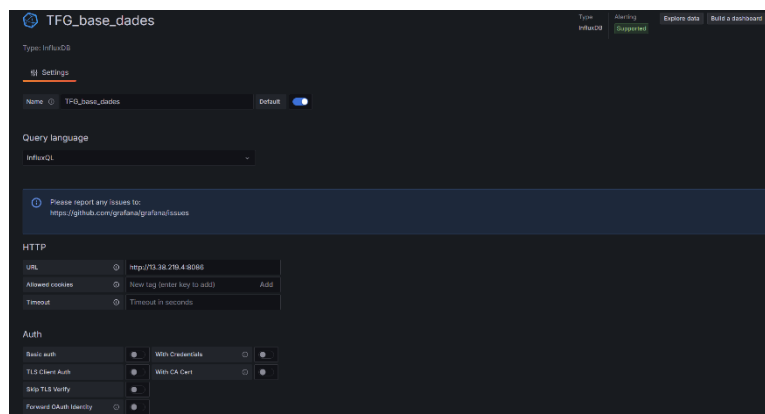


Figura 5.28. Configuració dels detalls de la connexió a InfluxDB

6. INTEGRACIÓ I CONFIGURACIÓ DEL FLUX DE TREBALL A NODE-RED

En el context d'un projecte relacionat amb IoT, com per exemple el monitoratge de dades de sensors, Node-Red és una eina perfecta per tal de gestionar el flux de de dades des dels sensors fins la visualització i l'anàlisi. Aquest procés d'integració i configuració del flux de treball consta de diverses etapes importants, des de rebre les dades a través del protocol MQTT, processar i emmagatzemar-les de en una base de dades com InfluxDB, fins a la visualització d'aquestes en temps real amb Grafana.

És per aquest motiu, que per dur a terme el projecte ens hem decantat per treballar amb aquesta eina. A continuació tenim el diagrama de flux que hem utilitzat per tal de fer funcionar el projecte:

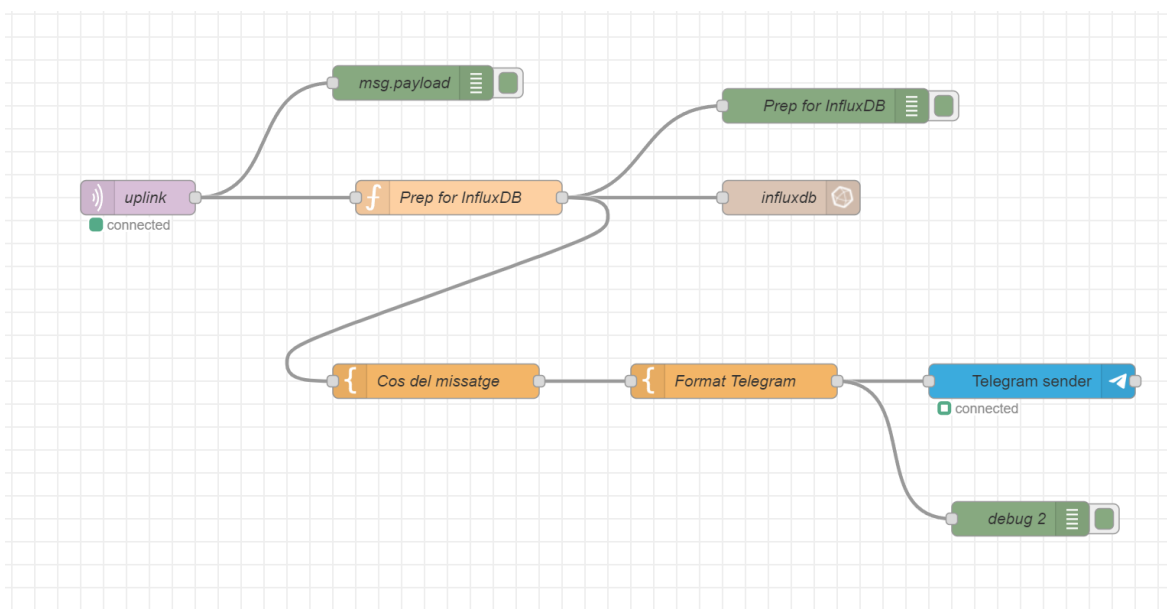


Figura 6.1. Diagrama de flux del projecte.

6.1 Configuració dels nodes MQTT a Node-RED.

MQTT³⁸ és un protocol de missatgeria basat en estàndards, o un conjunt de normes, que s'utilitza per a la comunicació d'un equip a un altre. Els sensors intel·ligents, els dispositius portàtils i altres dispositius d'IoT generalment tenen que transmetre i rebre dades a través d'una xarxa amb recursos restringits i un ample de banda limitat. Aquests dispositius IoT utilitzen MQTT per a la transmissió de dades, ja que resulta fàcil d'implementar i poder comunicar dades IoT de manera eficient. MQTT admet la missatgeria entre dispositius al núvol i el núvol al dispositiu. [24]

³⁸ Message Queing Telemetry Transport.

6.1.1 Subscripció als temes MQTT.

En primer lloc, accedirem a la TTN i de les integracions que ofereix aquesta, farem ús de MQTT. El més important és guardar l'usuari i la contrasenya ja que necessitarem aquestes dades per després al Node-Red.

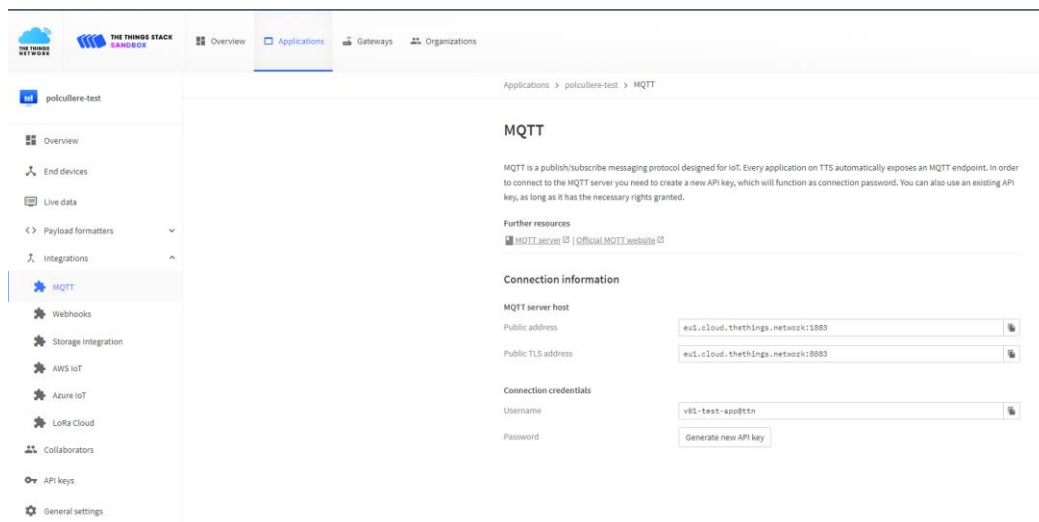


Figura 6.2. Integració MQTT al TTN.

Pel que fa al bloc violeta del diagrama de flux de la *Figura 6.1*, és el que correspon al protocol MQTT. Aquest bloc s'encarrega de connectar-se a un intermediari (*broker*) i en subscriure's als missatges del tema especificat (*topic*), és a dir, el que fa és rebre els missatges publicats en un topic.

Per configurar aquest node, cal especificar les característiques de connexió del servidor, com l'adreça i el port, així com les credencials d'aplicació, que inclouen el nom de l'aplicació de TTN d'on es volen extreure les dades i la contrasenya corresponent a la clau API. Aquests paràmetres els hem vist a la *Figura 6.2*.

A més a més, cal omplir el camp *Topic*, ja que és essencial per tal d'establir la connexió amb el servidor i accedir a l'aplicació on es troben els sensors. Segons la documentació de The Things Network, per subscriure's al tràfic de pujada, el format que s'ha de seguir per omplir el camp del *Topic* és el següent:

```
v3/{application id}@{tenant id}/devices/{device id}/up
```

En la part de *{application id} @ {tenant id}* posarem el nom de l'aplicació juntament amb la identificació del servidor TTN, en aquest cas seria *v01-test-app@ttn*.

Al camp de *{device id}* posarem l'identificador de cadascun dels sensors.

Finalment, el camp del topic quedaria de la següent forma:

```
v3/v01-test-app@ttn/devices/eui-70b3d57ed0066305/up
```

Figura 6.3. Configuració del node MQTT.

Figura 6.4. Configuració del servidor del node MQTT.

Figura 6.5. Configuració de la seguretat del servidor.

6.1.2 Comprovació del missatge rebut.

En aquest apartat s'explica el bloc verd del diagrama de flux de la *Figura 6.1*, que és el que correspon al *msg.payload*. Aquest bloc s'encarrega de mostrar el missatge que surt del node MQTT en format JSON. Aquest bloc és molt útil, ja que et permet observar que és el que realment està rebent del node MQTT, i per tant, jutjar si les dades són correctes o cal modificar alguna cosa.

Bàsicament, en el que ens centrarem serà en el camp *decoded_payload*, el qual és una part crucial d'aquest missatge, ja que conté les dades decodificades que el dispositiu ha enviat. En aquest cas, hem fet ús del sensor de pressió i temperatura i del sensor de co2 i tvoc. A més a més, també podem veure el voltatge de la bateria que s'està fent servir. Més endavant, afegirem el sensor GPS.

A més a més, a la següent figura, podrem observar a la part superior, el *topic*, el qual especifica el tema MQTT al qual s'ha subscrit Node-RED per rebre aquest missatge:

```

22/5/2024, 19:59:30 node: msg.payload
v3/v01-test-app@ttn/devices/eui-70b3d57ed0066305/up : msg : Object
▼ object
  topic: "v3/v01-test-app@ttn/devices/eui-70b3d57ed0066305/up"
▼ payload: object
  ▶ end_device_ids: object
  ▶ correlation_ids: array[1]
  received_at: "2024-05-22T17:59:30.291955396Z"
  ▼ uplink_message: object
    session_key_id: "AY+hW+U1DrnbIMEqPPqvSA=="
    f_port: 2
    f_cnt: 60
    frm_payload: "AwzPBgr9aAGWAAA0eg=="
  ▼ decoded_payload: object
    batteryVoltage: 3.706
    co2: 406
    pressure: 101383.528
    temperature: 32.79
    tvoc: 0
  ▶ rx_metadata: array[2]
  ▶ settings: object
    received_at: "2024-05-22T17:59:30.088042536Z"
    consumed_airtime: "0.123392s"
  ▶ version_ids: object
  ▶ network_ids: object
qos: 0
retain: false
_msgid: "d817130aedbef044"

```

Figura 6.6. Node msg.payload.

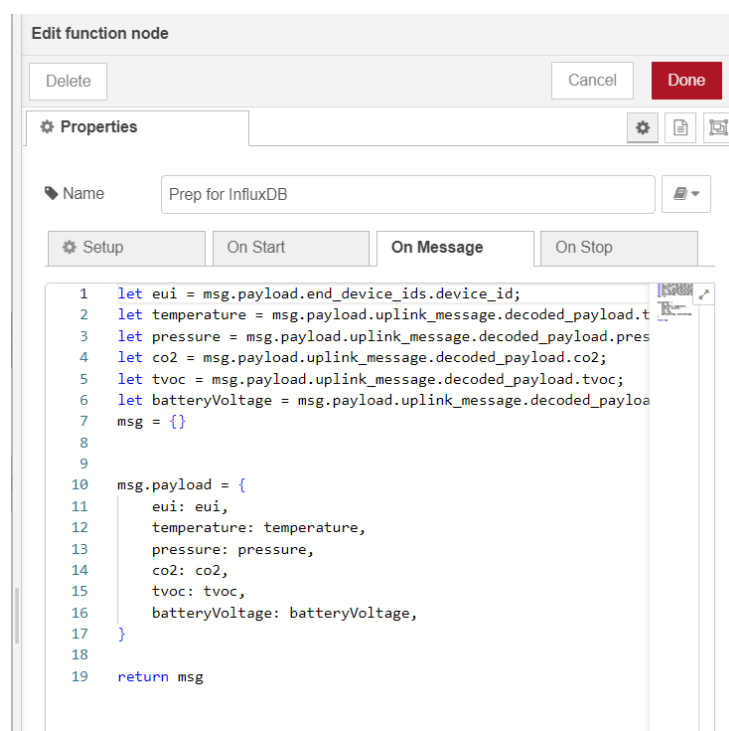
6.2 Configuració dels nodes InfluxDB a Node-RED.

Abans d’inserir les dades a InfluxDB cal prepara-les de forma adequada. Al Node-RED això significa utilitzar nodes que prenguin les dades que reben via MQTT i les transformin en un format compatible amb InfluxDB. Per altra banda, cal mencionar que en aquesta part del projecte encara no s’ha inclòs el sensor GPS i per tant la configuració dels nodes sol tracten els sensors inclosos fins ara.

6.2.1 Preparació de dades per a la inserció a la base de dades.

En aquest apartat s’exposen aquells blocs que fan possible la preparació de les dades per tal de poder-les inserir a la base de dades. El responsable d’això és el bloc anomenat *Prep for InfluxDB* de la *Figura 6.1*. Bàsicament, el que fa és “parsejar”, és a dir, identificar i separar els elements de l’objecte *uplink message* que es rep per part de l’intermediari MQTT, amb l’objectiu de tenir els valors de les mesures obtinguts per després poder-los inserir a la base de dades.

A continuació podem veure la configuració d’aquest node per tal de dur a terme tot allò mencionat anteriorment:



```

1 let eui = msg.payload.end_device_ids.device_id;
2 let temperature = msg.payload.uplink_message.decoded_payload.t
3 let pressure = msg.payload.uplink_message.decoded_payload.pres
4 let co2 = msg.payload.uplink_message.decoded_payload.co2;
5 let tvoc = msg.payload.uplink_message.decoded_payload.tvoc;
6 let batteryVoltage = msg.payload.uplink_message.decoded_payloa
7 msg = {}
8
9
10 msg.payload = {
11   eui: eui,
12   temperature: temperature,
13   pressure: pressure,
14   co2: co2,
15   tvoc: tvoc,
16   batteryVoltage: batteryVoltage,
17 }
18
19 return msg

```

Figura 6.7. Configuració del node Prep for InfluxDB.

Posteriorment, el que hem fet ha estat posar un node *debug* com ja hem fet anteriorment, per tal de poder observar les dades que surten pel node *Prep for InfluxDB*. Com podem veure a la següent figura, veiem els valors de les mesures obtinguts, ja amb el format correcte per inserir a la base de dades:

```

22/5/2024, 19:46:38  node: Prep for InfluxDB
msg : Object
  ▼ object
    ▼ payload: object
      eui: "eui-70b3d57ed0066305"
      temperature: 32.03
      pressure: 101375.408
      co2: 789
      tvoc: 58
      batteryVoltage: 3.698
      _msgid: "d07270f3b37b1f30"

```

Figura 6.8. Node.Prep for InfluxDB.

6.2.2 Inserció de dades a InfluxDB.

Finalment, queda inserir les dades ja preparades, a la base de dades d'InfluxDB. Per dur a terme aquest procés, primer s'ha d'instal·lar la llibreria *node-red-contrib.influxdb*. Aquesta llibreria és molt important ja que proporciona nodes per interactuar amb la base de dades InfluxDB dins de Node-RED. Els principals 4 nodes son InfluxDB In, InfluxDB Out, InfluxDB Batch i InfluxDB Query. Per dur a terme el nostre projecte, usarem el InfluxDB Out, el qual s'utilitza per inserir dades a la base de dades InfluxDB.

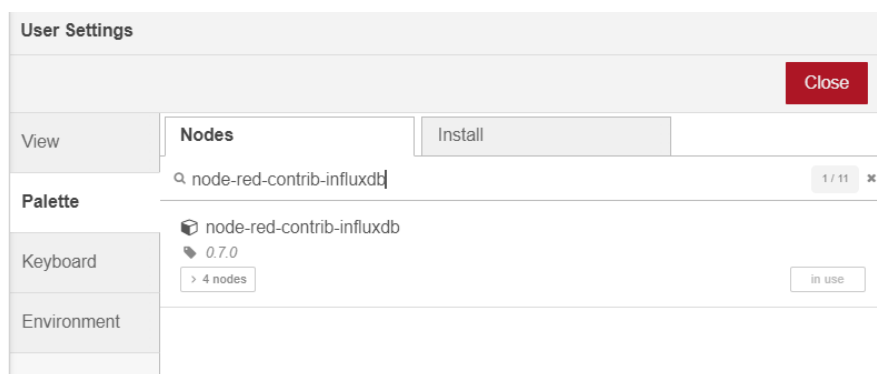


Figura 6.9. Instal·lació de la llibreria *node-red-contrib-influxdb*.

Un cop tenim les llibreries instal·lades, és moment de configurar el node *influxdb*, vist a la *Figura 6.1*. En primer lloc, posarem nom a la nostra base de dades, en aquest cas TFG_base_dades. Posteriorment, seleccionarem la versió 1.x ja que és la versió que tenim instal·lada a la màquina virtual. A continuació, posarem l'adreça IP que tenim assignada a la nostra instància i el port 8086, que és el port per defecte al qual InfluxDB està escoltant. Finalment, afegim a la base de dades el nom que hem creat anteriorment i posarem l'usuari i la contrasenya creats anteriorment a la *Figura 5.19*.

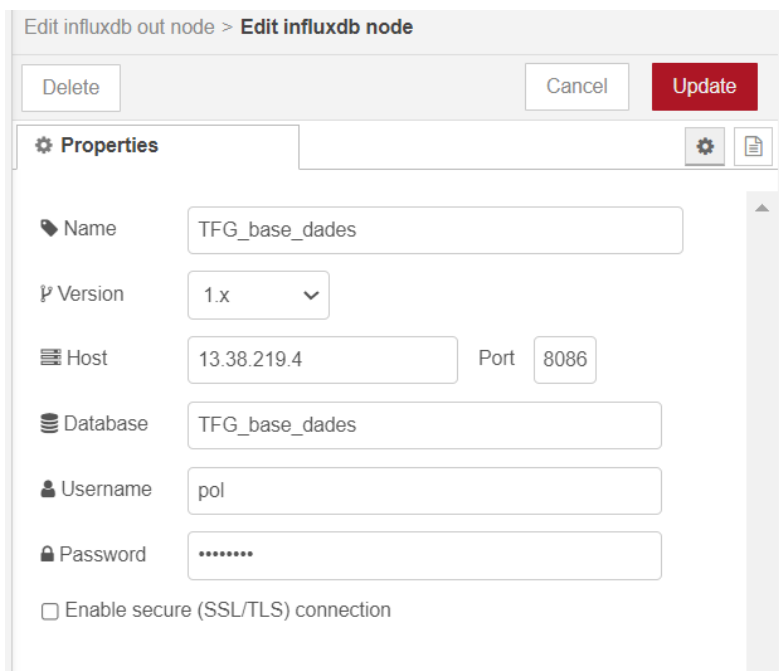


Figura 6.10. Configuració del node influxdb.

Finalment, afegirem a l'apartat *Server* la configuració creada anteriorment a la *Figura 6.10* i a l'apartat de *Measurement* posarem el nom de la nostra base de dades.

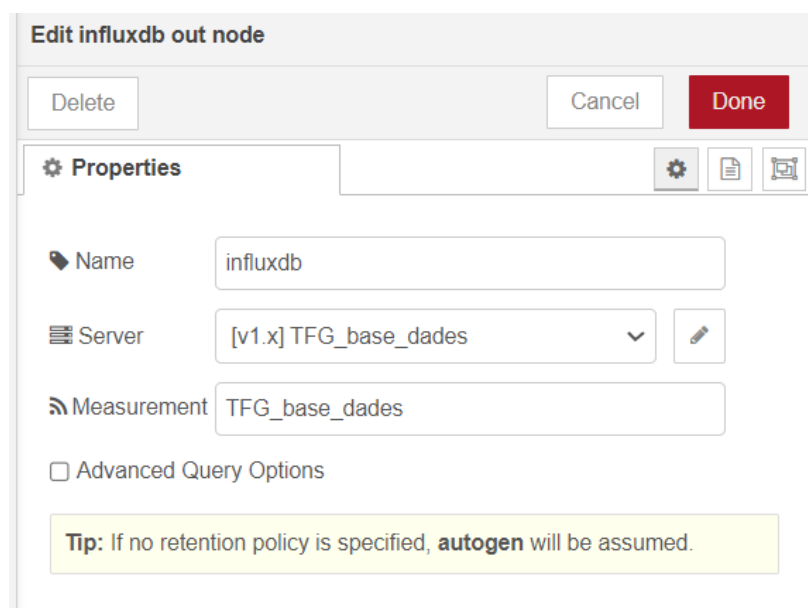


Figura 6.11. Configuració del node influxdb.

Un cop hem finalitzat tot el procediment per a la inserció de dades a InfluxDB, ja podem accedir a la nostra base de dades d'InfluxDB a través de la nostra màquina virtual. Aquí és on emmagatzemarem totes les dades de temperatura, pressió, voltatge de la bateria, co2 i tvoc, i posteriorment, les dades referents al sensor GPS.

time	batteryVoltage	co2	eui	pressure	temperature	tvoc
1716419112172202458	3.798	415	eui-70b3d57ed0066305	101466.624	30.43	1
1716419138534140645	3.822	406	eui-70b3d57ed0066305	101473.424	30.56	0
1716419165099183866	3.738	424	eui-70b3d57ed0066305	101470.216	30.44	3
1716419192115739422	3.722	426	eui-70b3d57ed0066305	101467.16	30.33	3
1716419218991552517	3.812	420	eui-70b3d57ed0066305	101468.184	30.18	3
1716419245421227882	3.812	406	eui-70b3d57ed0066305	101465.872	30.24	0
1716419272701161586	3.812	416	eui-70b3d57ed0066305	101463.28	29.91	1
1716419299396360126	3.698	421	eui-70b3d57ed0066305	101466.792	29.88	3
1716419325644430159	3.814	608	eui-70b3d57ed0066305	101468.176	30.04	31

Figura 6.12. Base de dades InfluxDB.

6.3 Configuració dels nodes per enviar dades a Telegram.

Telegram és una aplicació de missatgeria instantània que ha guanyat molta popularitat degut a la seva seguretat, flexibilitat i facilitat per fer servir. Enviar dades a Telegram des de sistemes automatitzats com Node-RED, té diversos avantatges com la comunicació immediata, l'accessibilitat ràpida i la automatització eficient de les dades. L'enviament de dades a Telegram combina la potència de la comunicació en temps real amb la flexibilitat i accessibilitat de la plataforma de missatgeria. Això esdevé una eina molt potent per a la monitorització, notificació i interacció, tot millorant l'eficiència operativa i la capacitat de resposta davant de qualsevol situació. [25]

6.3.1 Preparació dels nodes per connectar-se a Telegram.

En primer lloc, és important instal·lar les llibreries pertanyents, per tal de poder fer ús dels nodes per connectar-se a Telegram.

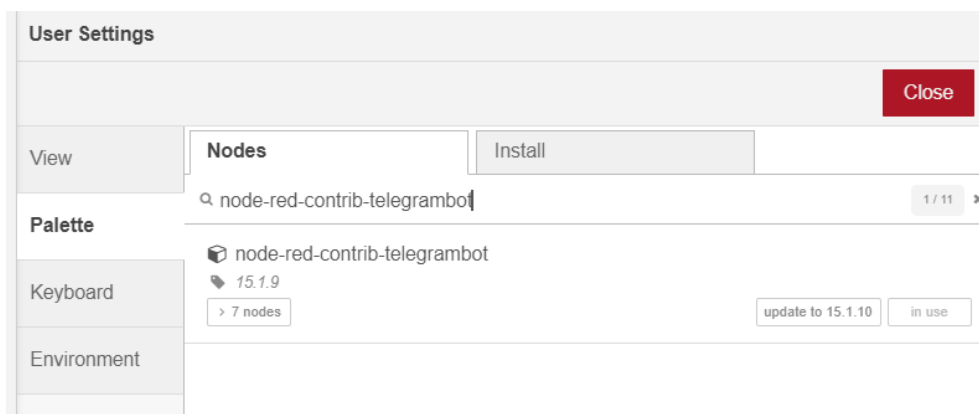


Figura 6.13. Instal·lació de la llibreria node-red-contrib-telegrambot.

A continuació, el que s'ha fet és agafar les dades que enviem a la base de dades, tot adaptant-les al format que accepta Telegram. És per això, que com podem veure a la Figura 6.1 hem creat dos nodes per tal d'adequar el format del missatge per a l'arquitectura que pot suportar Telegram.

En el primer node, el que s'observa és el cos del missatge que posteriorment podrem observar a través de la plataforma de Telegram. Com podem observar, es basa en el format *payload*, ja vist anteriorment.

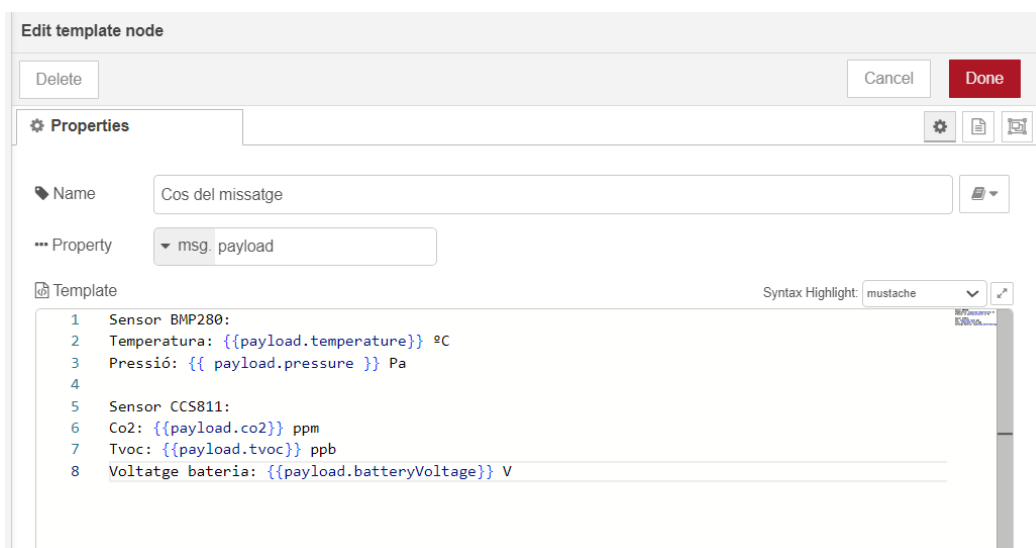


Figura 6.14. Cos del missatge de Telegram.

A continuació, el que hem de fer és adaptar el cos del missatge que volem enviar, al format que accepta la plataforma de Telegram. No obstant, primer hem de conèixer el *chatId*, que no és més que l'ID del xat de Telegram al qual es vol enviar aquest missatge. Per tal de conèixer-lo, primer cal crear aquest diagrama de flux de treball:

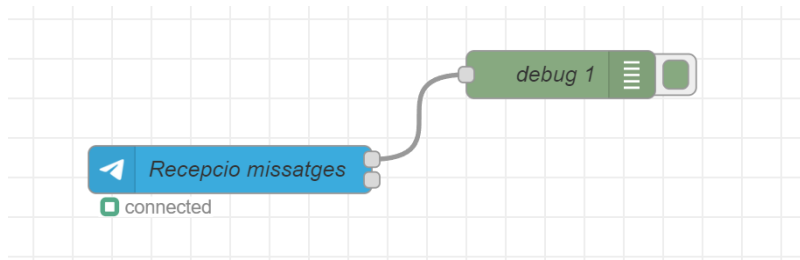


Figura 6.15. Diagrama de flux de treball per conèixer el chatId.

El format de la plataforma Telegram, consta de 3 parts; *chatID*, *type*, *content*.

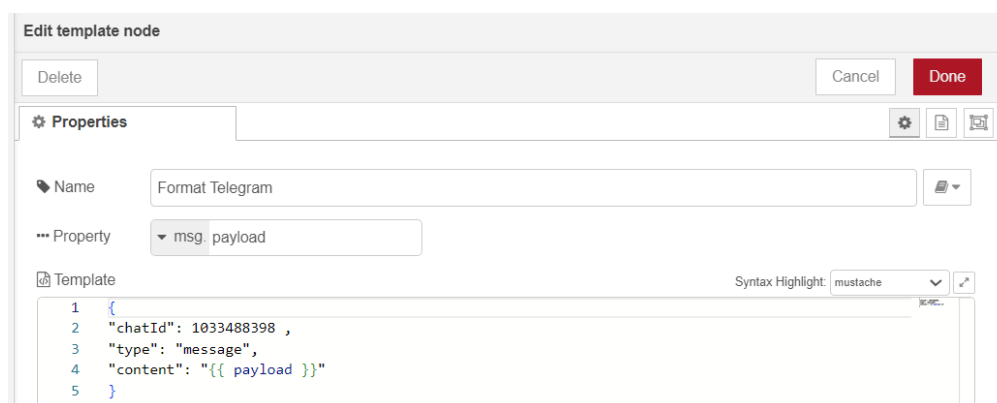


Figura 6.16. Missatge en format Telegram.

6.3.2 Creació del Bot, Configuració del node Telegram Sender i Resultat Final.

Posteriorment, cal crear el canal de Telegram per on s'enviaran les dades dels sensors. Primer de tot cal buscar un canal anomena *BotFather*. Aquí, li indicarem el nom del canal que volem crear i el nom de l'usuari associat a aquest canal. Seguidament, ell et crea el canal i et dona un *token*, el qual és una cadena única de caràcters que actuen com una clau d'autenticació per permetre que aplicacions externes, en aquest cas Node-RED, es puguin comunicar amb el nostre bot de Telegram.

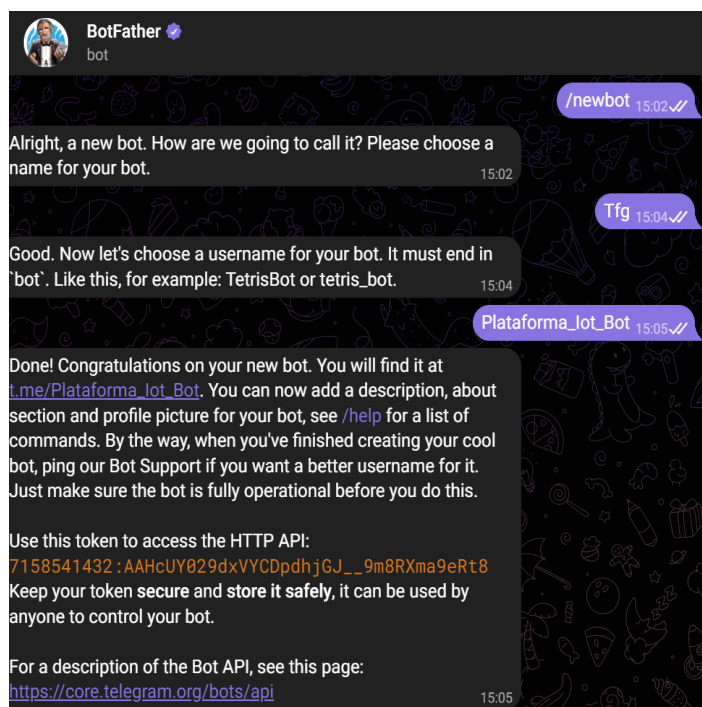


Figura 6.17. Creació del canal de Telegram per mitjà del BotFather.

Un cop creat el canal, cal configurar el node *Telegram Sender* que hem vist a la *Figura 6.1*, per tal de que la connexió entre Node-RED i Telegram sigui profitosa. En aquest cas, caldrà posar el nom d'usuari i el *token* creats anteriorment.

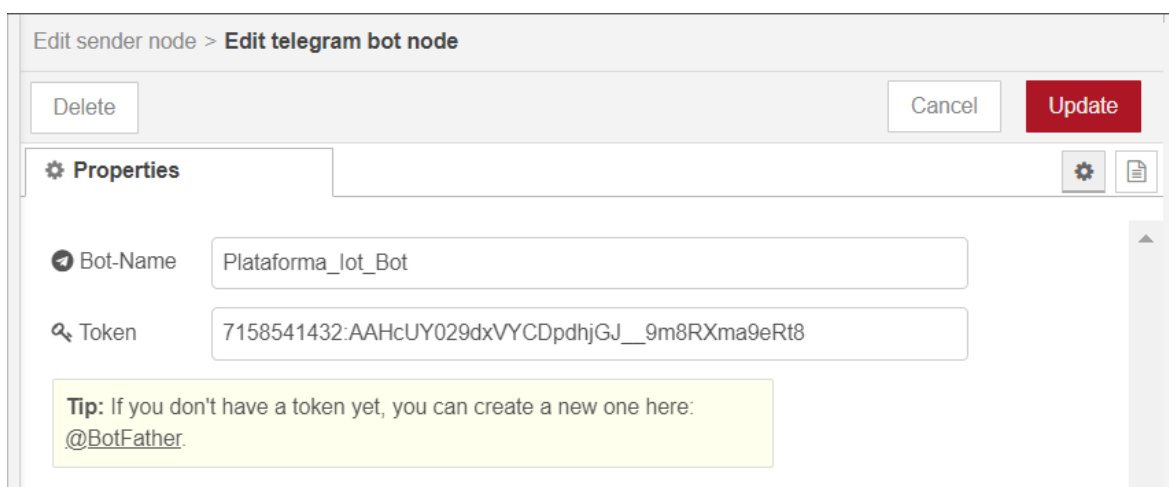


Figura 6.18. Configuració del node Telegram sender.

Finalment, un cop realitzada tota la tasca de creació del bot i configuració del node *Telegram sender*, ja podem observar el resultat final, on cada missatge que envien els diferents sensors, es veu de la següent forma:

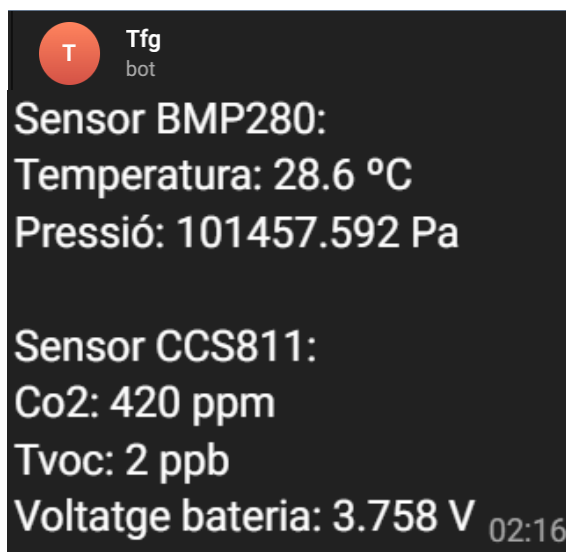


Figura 6.19. Missatge rebut per la plataforma Telegram.

7. VISUALITZACIÓ I ANÀLISI DE DADES AMB GRAFANA

Un cop arribats a aquest punt, solament ens queda fer ús de l'eina Grafana amb la finalitat de visualitzar i analitzar les dades rebudes en temps real. És per això, que s'han de crear uns *dashboards* personalitzats, amb els quals podem visualitzar les dades i tenir una visió completa de tot allò que estem rebent. A més a més, aquesta plataforma s'integra perfectament amb una àmplia varietat de fonts de dades.

7.1 Creació de dashboards a Grafana.

Un cop realitzada la configuració inicial de Grafana i la posada en marxa, podem observar que Grafana ofereix una àmplia gamma d'opcions per a la configuració de gràfics i *widgets*. En aquest apartat, es mostra la configuració del *dashboard* per tal de captar les dades de la base de dades i assegurar-se que es presenten d'una manera clara i eficaç.

Per tant, aquesta és la configuració de la *Query*³⁹ en Grafana, una eina molt popular en la qual es tria la font d'on es trauran les dades, es seleccionen els camps específics i les funcions que s'aplicaran a aquests camps.

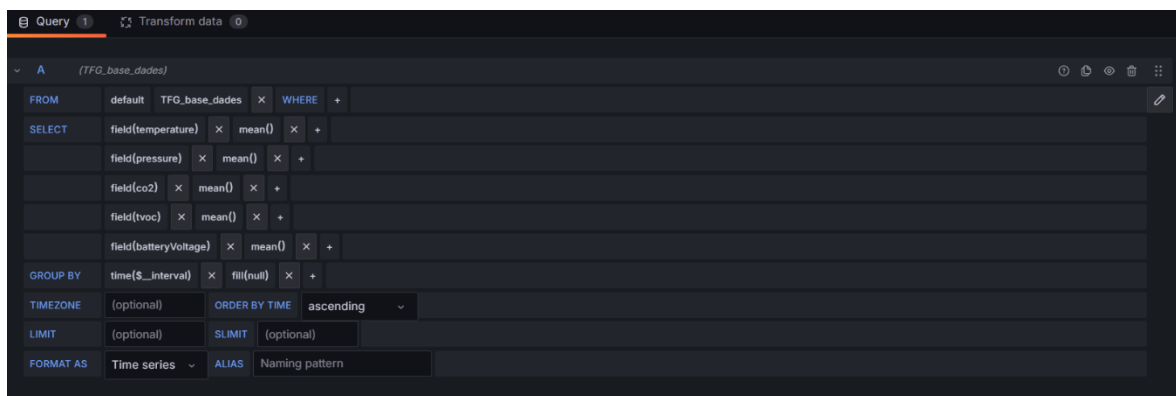


Figura 7.1. Configuració de la Query.

Posteriorment, és moment de crear els *dashboards* on podrem monitoritzar les dades enviades pel sensor a temps real, i posteriorment analitzar-les i treure'n conclusions. No obstant, primer cal configurar els *override*⁴⁰ referents a cada *dashboard*, per tal de mostrar les dades que toquen per cada un d'ells:

³⁹ És una consulta que s'utilitza per extreure informació específica d'una base de dades. En el context de bases de dades, un query permet als usuaris cercar i recuperar dades mitjançant un llenguatge de consulta.

⁴⁰ Són ajustaments que es poden aplicar a sèries de dades individuals en un panell per canviar la seva aparença o comportament.

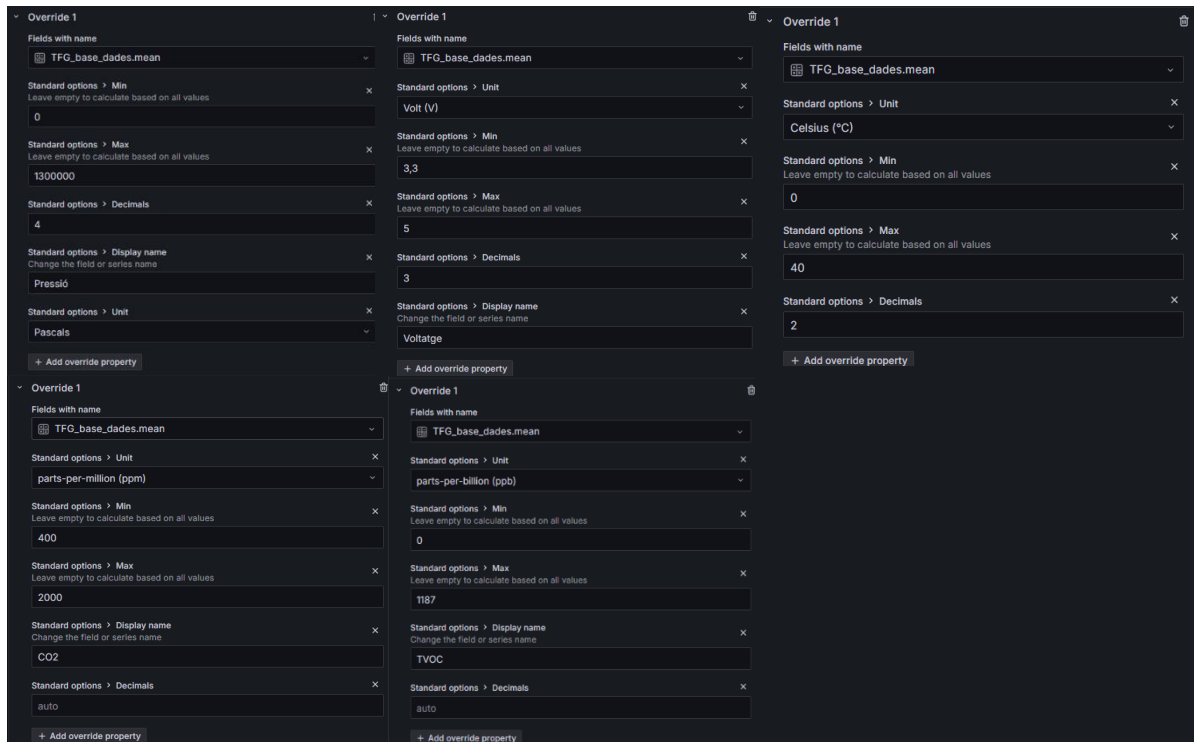


Figura 7.2. Configuració dels override.

Finalment, aquesta és la pantalla principal que observem quan accedim a Grafana. Cal destacar que està en ús la implementació del sensor de temperatura i pressió, i el sensor de co2 i tvoc. A més a més, també es pot observar el voltatge de la bateria que alimenta la placa Heltec Cubecell i que per tant, alimenta als sensors.



Figura 7.3. Dashboard del projecte.

7.2 Alertes i notificacions a Grafana.

Finalment, Grafana també ofereix la capacitat d'establir alertes i notificacions basades en les dades monitoritzades. Aquesta funcionalitat és de caire essencial per a la detecció de problemes i per a la resposta ràpida de possibles situacions crítiques. És per això, que la configuració adequada d'aquestes alertes garanteix el bon funcionament dels seus sistemes i aplicacions.

El primer pas és crear una alerta. En el nostre cas, hem creat un total de 4 alertes, una per cada camp que poden mesurar els sensors:

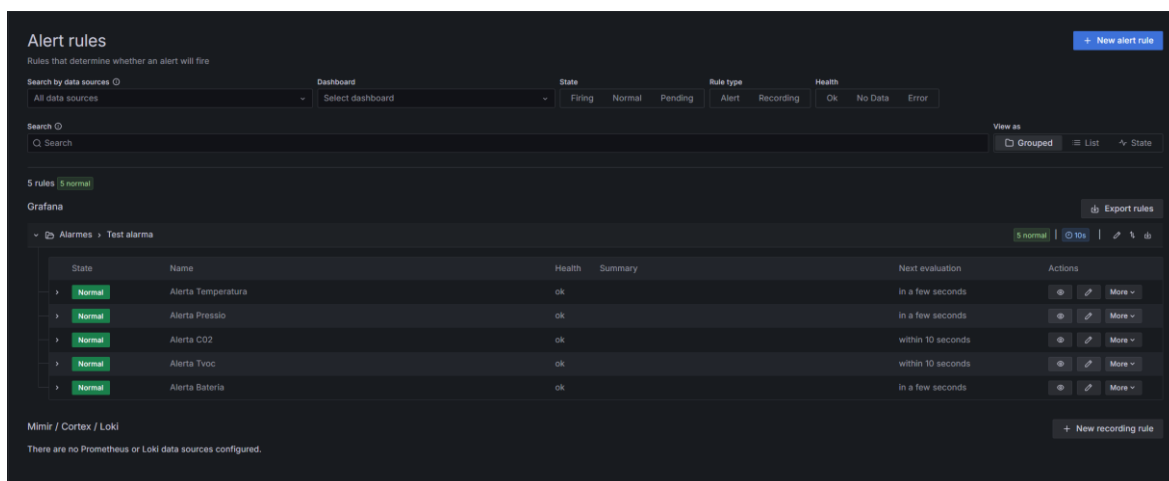


Figura 7.4 Creació de les alertes.

Cada una d'aquestes alertes consta de la següent configuració, depenent del camp mesurat. Per exemple, a continuació es mostra la configuració de l'Alerta de Temperatura, no obstant, totes segueixen el mateix patró de configuració:

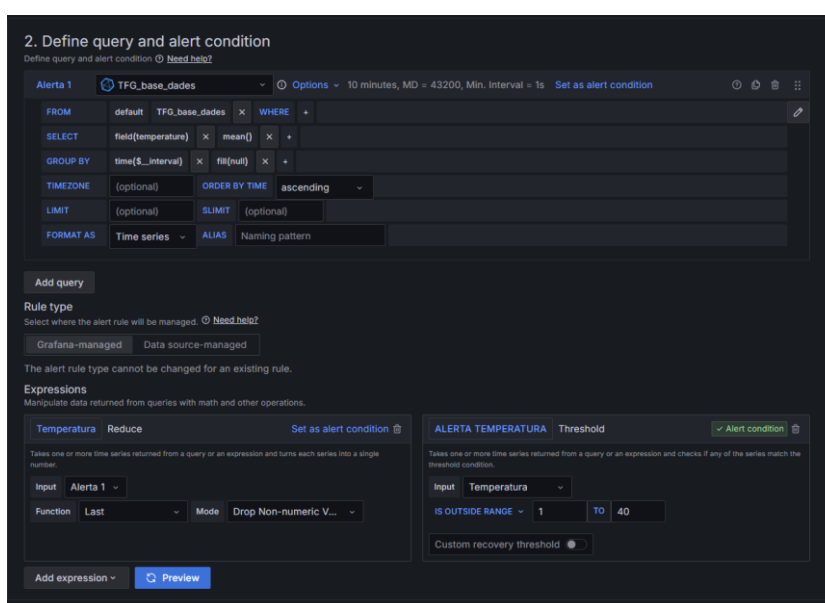


Figura 7.5. Configuració de l'Alerta de Temperatura.

En resum, el funcionament de la configuració d'aquesta alerta consta de dues parts. La primera part és on es realitza la consulta de dades, on es recuperen els valors de la temperatura, en aquest cas en concret, de la base de dades *TFG_base_dades*. La segona part és on s'utilitzen les expressions de reducció, la qual aplica la funció *last* per utilitzar l'últim valor de la sèrie temporal de temperatures. A posterior, s'usa una condició d'alerta, la qual es dispara si l'últim valor de la temperatura està fora del rang de 1 a 40 graus centígrads.

Aquestes configuracions permeten monitoritzar contínuament els valors rebuts i generar alertes quan es detecten valors fora del rang especificat. A més a més, en aquesta part de configuració de les alertes també hem afegit el període en el qual una regla d'alerta pot estar infringint la condició fins que la regla d'alerta s'activi, en el nostre cas 10 segons. Això es fa principalment per si hi ha algun valor que supera el nostre rang, però es troba en un període transitori, que no es detecti com a alerta. Finalment, afegirem una etiqueta *priority = test* per tal de poder filtrar les alertes posteriorment a les *notification policies*.

3. Set evaluation behavior
Define how the alert rule is evaluated. [Need help?](#)

Folder
Select a folder to store your rule.
Alarmes or + New folder

Evaluation group
Rules within the same group are evaluated concurrently over the same time interval.
Test alarma or + New evaluation group

All rules in the selected group are evaluated every 10s [🔗](#)

Pending period
Period in which an alert rule can be in breach of the condition until the alert rule fires.
10s

Pause evaluation [🔗](#)

> Configure no data and error handling

4. Configure labels and notifications

Labels
Add labels to your rule to annotate your rules, ease searching, or route to a notification policy. [Need help?](#)

priority = test [🗑️](#)

[🔗 Add label](#)

Alert instance routing preview
When you have your folder selected and your query and labels are configured, click "Preview routing" to see the results here. [Preview routing](#)

Figura 7.6. Configuració de l'Alerta de Temperatura.

Seguidament, configurarem els *contact points*. En el nostre cas, volem que quan salti una alerta, ens envii un missatge a la plataforma Telegram, és per això que hem d'indicar-li el Chat ID del nostre canal creat anteriorment. A més a més, li hem d'indicar el missatge que volem que ens envii cada cop que s'activi l'alarma.

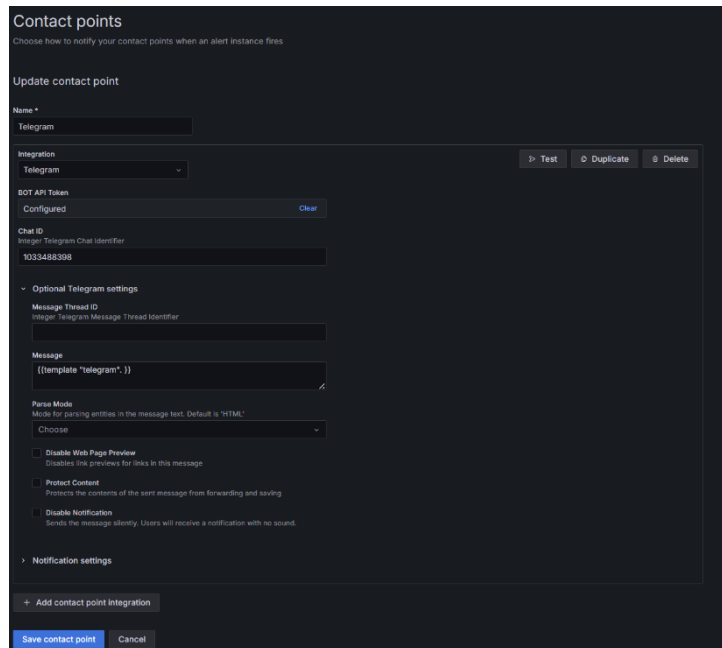


Figura 7.7. Configuració dels Contact points.

A més a més, en aquest mateix apartat de *contact points* crearem el que s'anomena la plantilla de notificació. Aquesta plantilla defineix el contingut del missatge d'alerta que s'enviarà a Telegram (*telegram.message_alert*). Utilitza etiquetes (*.Labels.alertname*) per mostrar el nom de l'alerta, l'estat (*.Status*) i el moment d'inici (*.StartsAt*). També recorre els valors (*.Values*) per mostrar les claus i els valors associats. Aquestes configuracions permeten que quan una alerta s'activi o es resolgui, es notifiquin els punts de contacte amb un missatge personalitzat, en aquest cas, a través de Telegram. Les plantilles de notificació garanteixen que la informació rellevant sigui clara i consistent en totes les notificacions enviades. A continuació, podem observar el codi usat per aquesta plantilla de notificació:

```

{{ define "telegram.message_alert" }}
<strong>{{ .Labels.alertname }}</strong> {{ .Status }} {{ .StartsAt }}
{{ range $k, $v := .Values }} {{ $k }}={{ $v }} {{ end }}
{{ end }}

{{ define "telegram" }}

{{ if .Alerts.Firing }}
  {{ range .Alerts.Firing }}
    🚨 {{ template "telegram.message_alert" . }}
  {{ end }}
{{ end }}

{{ if .Alerts.Resolved }}
  {{ range .Alerts.Resolved }}
    ✅ {{ template "telegram.message_alert" . }}
  {{ end }}
{{ end }}

{{ end }}

```

Finalment, hem creat una *notification policy* que utilitzi el nostre *contact point* per entregar les alertes que nosaltres vulguem. El que hem fet és usar el label *priority = test*, el qual hem definit anteriorment a la nostra alerta, el qual serà el responsable d'enviar els missatges al nostre *contact point*. Posteriorment, hem definit amb quina freqüència i en quin moment s'envien aquests missatges al nostre *contact point*.

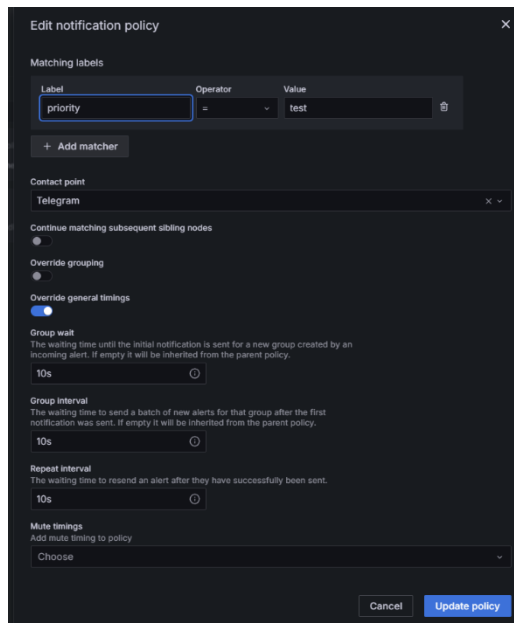


Figura 7.8. Configuració de la notification policy.

Un cop realitzades totes les configuracions pertanyents, és hora de veure reflectit el resultat i observar l'alerta un cop ha saltat, i un cop ha estat resolta. En aquest cas, el que he fet és forçar l'alerta pel sensor de temperatura, tot desendollat un dels cables de la connexió, i aquest ha estat el resultat:

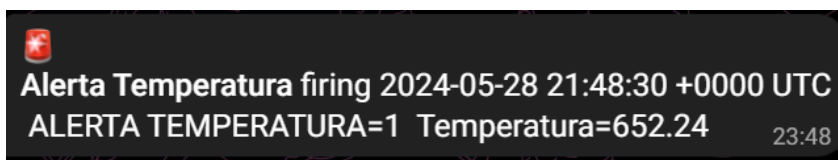


Figura 7.9. Notificació d'alerta de temperatura a Telegram.

A més a més, si observem les alertes al Grafana, podem observar com l'alerta de la temperatura està en l'estat *firing*, el que significa que l'alerta ha saltat i que està sobrepassant el rang establert.

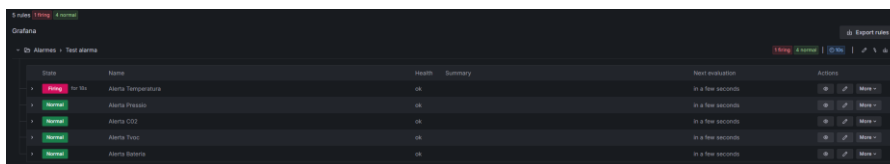


Figura 7.10. Estat de les alertes al Grafana.

Finalment, un cop solucionat el problema de connexió, el qual significa que l'alerta ha estat resolta, aquest és el missatge que rebem per la plataforma de Telegram:

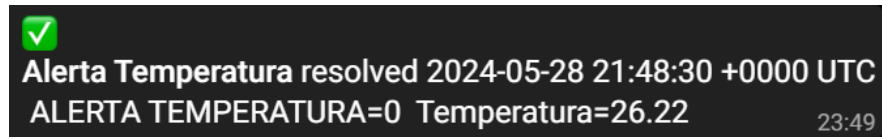


Figura 7.11. Notificació d'alerta de temperatura resolta a Telegram.

8. PROVES I RESULTATS

Un cop hem arribat fins aquest punt, és moment d'avaluar el conjunt de tota l'arquitectura vista fins ara en un entorn real. No obstant, cal remarcar que es tracta d'una prova de conceptes on l'important és validar la plataforma de captura de dades i d'enviament no pas la validesa de la mesura, que en tot cas es qualitativa i depèn molt del pressupost disponible per als sensors. En el següents apartats, es detalla el muntatge complet de la plataforma mòbil i s'avaluen els resultats de la prova en un entorn real realitzada el dia 30 de maig del 2023.

8.1 Preparació i muntatge.

En primera instància, es mostrarà la preparació del material necessari per tal de dur a terme la prova.

En primer lloc, s'ha usat la placa Heltec Cubecell HTCC-AB01 connectada al sensor HW-611 i al sensor CCS811. Aquesta placa, ha estat alimentada per una bateria de polímer de liti que proporciona 3,7 V i 1200 mAh. A continuació, podem observar la taula de connexions que s'ha dut a terme seguint les instruccions del fabricant:

Heltec Cubecell HTCC-AB01	HW-611	CCS811
3.3V	VCC	VCC
GND	GND	GND
SCL	SCL	SCL
SDA	SDA	SDA
GND	CSB	WAK
GND	SDD	INT

Taula 8.1. Taula de connexions per connectar els dos sensors.

Un cop realitzades les connexions pertanyents entre la placa i els sensors, i la placa i la bateria, aquest és el resultat del muntatge de la plataforma mòbil:

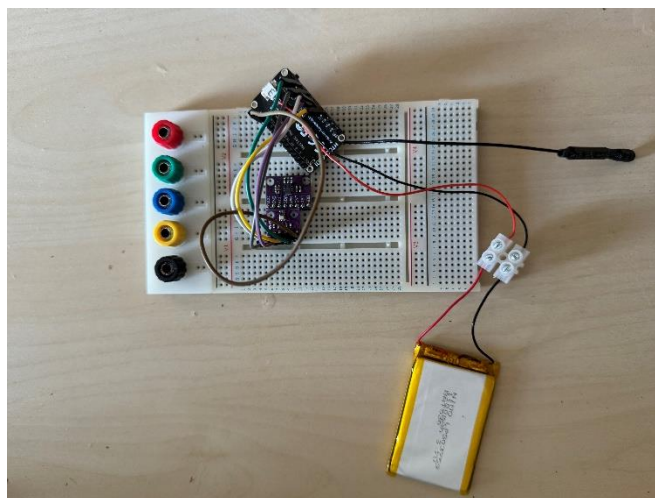


Figura 8.1. Muntatge de la plataforma mòbil.

A més a més, al ser un prototip el qual pot ésser exposat a diverses condicions climatològiques adverses, o inclús pot estar estàtic en un lloc durant cert temps, s'ha optat per la caixa estanca IP55⁴¹, la qual proporciona protecció contra la pols i protecció contra l'aigua, dos factors clau per a la preservació de tota la plataforma.

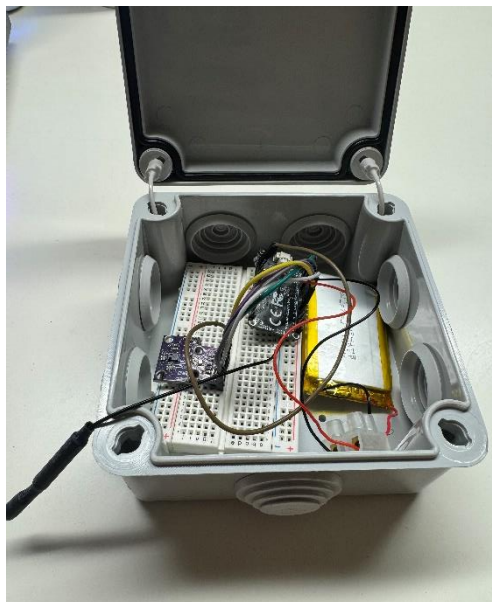


Figura 8.2. Plataforma integrada a la caixa IP55.

8.2 Resultats obtinguts.

Un cop hem muntat la plataforma mòbil, és hora de realitzar la prova de funcionament. Per tal de dur a terme aquesta prova, ens hem desplaçat fins al barri de Bonavista, on hi ha un *gateway* instal·lat el qual proporciona cobertura.

El cas pràctic consta de dues rutes d'aproximadament 1,3 km al voltant del *gateway*, amb la finalitat de demostrar que la plataforma mòbil funciona i envia dades amb sentit mentrestant té cobertura i dura la bateria. A més a més, també s'han anotat els punts on el *gateway* ha deixat de tenir cobertura, i així poder valorar també la distància la qual pot abastir aquest.

Aquesta ha estat la primera ruta realitzada d'aproximadament 20 minuts, on hem començat des de l'Escola Joan XXIII, que és el punt màxim de cobertura del *gateway*, i hem donat una volta pels voltants d'aquest. No obstant, hem marcat 9 punts considerables on hem anotat específicament els valors rebuts pels sensors.

⁴¹ Regida per la norma internacional IEC 60529 la qual defineix les classificacions IP (Ingress Protection) per establir el grau de protecció que ofereixen les caixes i altres carcasses contra la penetració de sòlids i líquids.

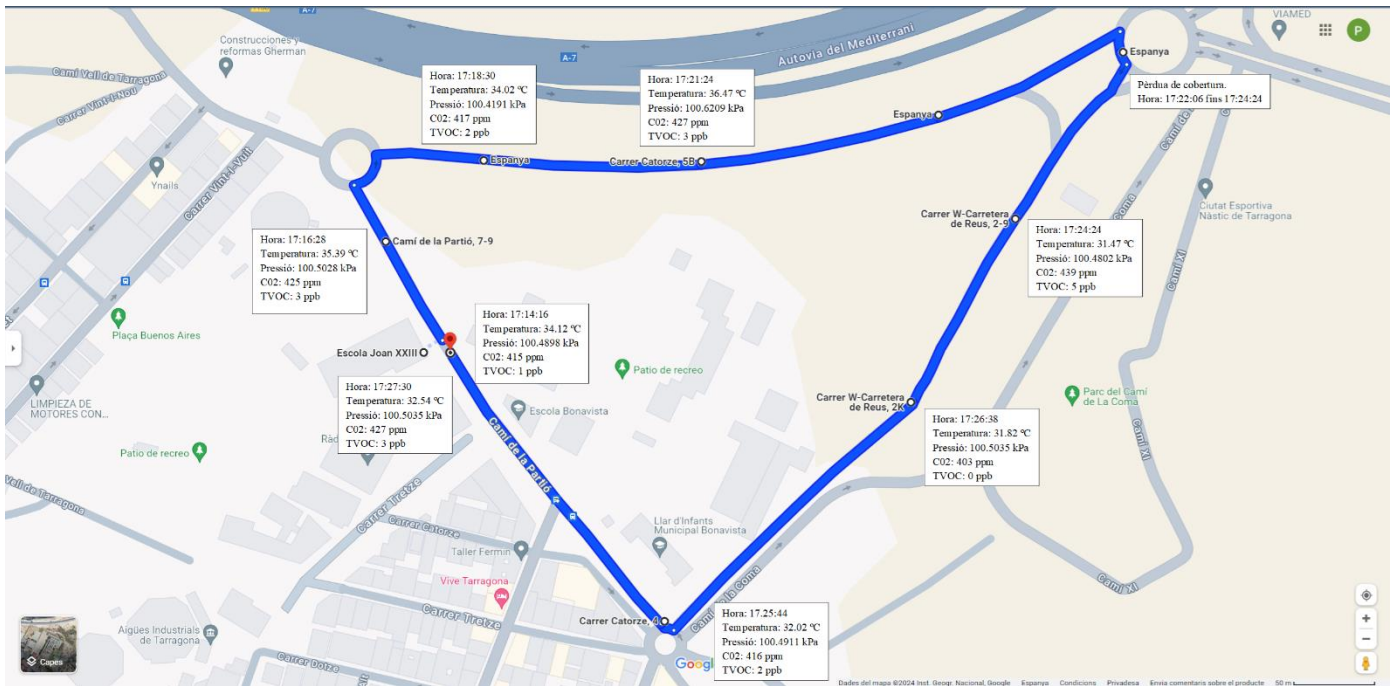


Figura 8.3. Ruta 1 realitzada.

Aquestes han estat les dades dels diferents paràmetres enviades cada 25 segons aproximadament, captats per la plataforma Grafana durant el primer trajecte:



Figura 8.4. Gràfica de la temperatura captada.

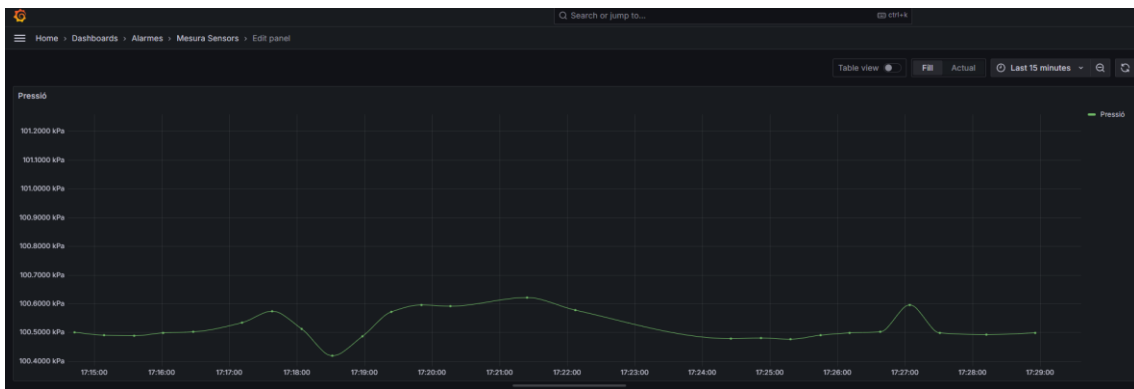


Figura 8.5. Gràfica de la pressió captada.

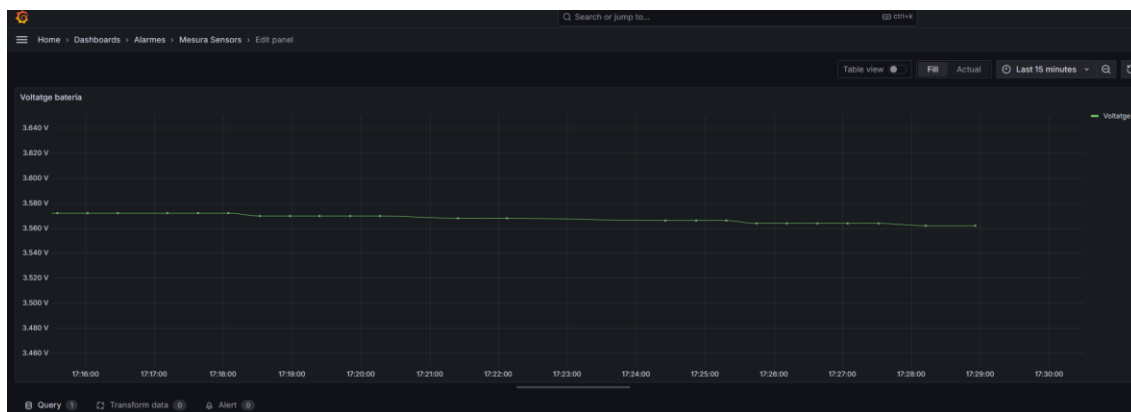


Figura 8.6. Gràfica del voltatge de la bateria captat.



Figura 8.7. Gràfica del CO2 captat.



Figura 8.8. Gràfica del TVOC captat.

Com podem observar, els valors són bastant coherents, tot i que hi ha petits pics de temperatura per exemple, degut a l'exposició dels diferents sensors al sol durant un període llarg de temps. Pel que fa a la contaminació de l'aire, el nivell de CO₂, tret d'un moment concret, es troba dins del rang de 0 a 500 ppm. Per altra banda, el nivell de TVOC es troba tota l'estona dins del rang de 0 a 200 ppb.

CO2 (ppm)	CO2 (mg/m ³)	Ambiente	Mensaje
de 0 a 500	de 0 a 900	Muy bueno	Aconsejado
de 501 a 1000	de 901 a 1800	Bueno	Bueno
de 1001 a 1500	de 1801 a 2700	Moderadamente malo	Consejo:ventilar
de 1501 a 2000	de 2701 a 3600	Muy malo	VENTILAR
mas de 2000	mas de 3600	Inaceptable	Desalojar

Taula 8.2. Valors de referència pel CO2.

VOC (ppb)	Ambiente	Mensaje
de 0 a 200	Muy bueno	Aconsejado
de 201 a 600	Bueno	Bueno
de 601 a 1000	Moderadamente malo	Consejo:ventilar
de 1001 a 2000	Muy malo	VENTILAR
mas de 2000	Inaceptable	Desalojar

Taula 8.3. Valors de referència pel TVOC.

Finalment, cal remarcar que hem tingut cobertura del *gateway* durant tota la prova, menys en un punt situat a 650 m d'aquest, on hem perdut la cobertura amb el *gateway* i ens ha aparegut el següent missatge per la consola del The Things Network:

```

↑ 17:44:00 eui-78b3d57ed0866305 Forward join-accept message DevAddr: 26 08 60 15 <> JoinEUI: 00 00 00 00 00 00 00 00 <> DevEUI: 78 B3 D5 7E 08 06 63 05 <>
↑ 17:43:59 eui-78b3d57ed0866305 Successfully processed join-request DevAddr: 26 08 6A E6 <> JoinEUI: 00 00 00 00 00 00 00 00 <> DevEUI: 78 B3 D5 7E 08 06 63 05 <>
@ 17:43:58 eui-78b3d57ed0866305 Accept join-request DevAddr: 26 08 60 15 <> JoinEUI: 00 00 00 00 00 00 00 00 <> DevEUI: 78 B3 D5 7E 08 06 63 05 <>
↓ 17:43:00 eui-78b3d57ed0866305 Transmit downlink Downlink transmission failed with result TOO_EARLY
    
```

Figura 8.9. Missatge rebut a través del TTN.

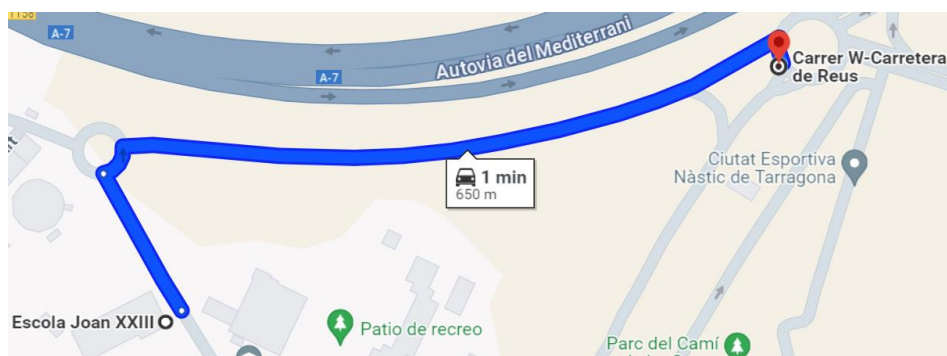


Figura 8.10. Punt de la ruta on hem perdut la cobertura amb el gateway.

Seguidament, hem realitzat la segona ruta d'aproximadament 15 minuts, on hem començat des del mateix punt que anteriorment, i hem donat una volta pels voltants d'aquest. No obstant, hem marcat 9 punts considerables on hem anotat específicament els valors rebuts pels sensors.

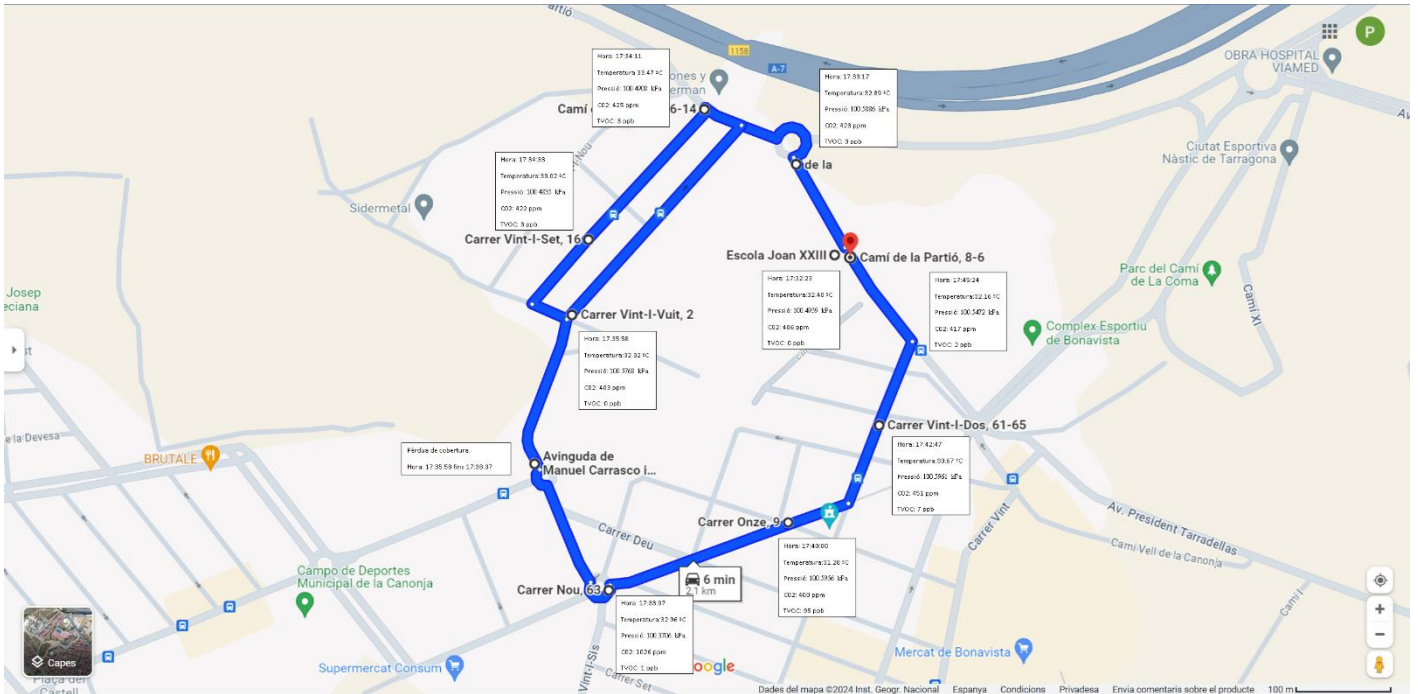


Figura 8.11. Ruta 2 realitzada.

Aquestes han estat les dades dels diferents paràmetres enviades cada 25 segons aproximadament, captats per la plataforma Grafana durant el segon trajecte:



Figura 8.12. Gràfica de la temperatura captada.



Figura 8.13. Gràfica de la pressió captada.



Figura 8.14. Gràfica del voltatge de la bateria captat.



Figura 8.15. Gràfica del CO2 captat.

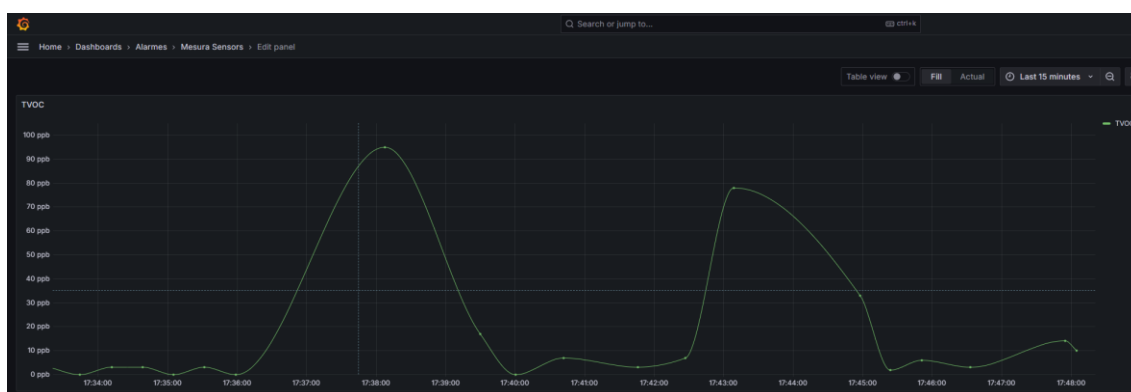


Figura 8.16. Gràfica del TVOC captat.

Com podem observar, i seguint el patró vist anteriorment, els valors són bastant coherents, tot i en aquest cas hi ha hagut algun pic respecte el paràmetre de CO₂. Això es degut a algun desajust en el sensor CCS811, el qual ha causat un augment considerable d'aquest paràmetre. Pel que fa a la contaminació de l'aire, el nivell de CO₂, tret d'un moment concret, es troba dins del rang de 0 a 500 ppm. Per altra banda, el nivell de TVOC es troba tota l'estona dins del rang de 0 a 200 ppb.

Finalment, en aquesta segona ruta també hem pogut tenir cobertura del *gateway* durant tota la prova, menys quan ens hem allunyat 750 m del punt inicial i l'hem perduda. El missatge per la consola del The Things Network ha tornat a aparèixer en aquest instant.

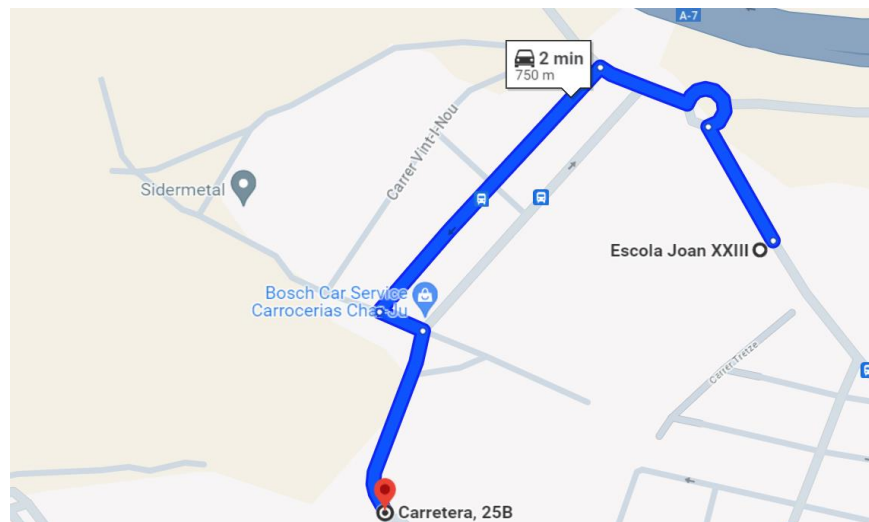


Figura 8.17. Punt de la ruta on hem perdut la cobertura amb el *gateway*.

Per concloure els resultats obtinguts, mencionar que durant les dues rutes hem pogut enviar dades gràcies a la cobertura que ens ha proporcionat el *gateway_id*: "ej23-01". A continuació, podem observar un dels paquets captats per aquest *gateway*:

```

31   "received_at": "2024-05-30T15:48:04.877407149Z",
32   "uplink_message": {
33     "session_key_id": "AY/KLM81qnHUoVaNIYVGFQ==",
34     "f_port": 2,
35     "f_cnt": 5,
36     "fzm_payload": "AwzZBF5CkAHXAAoN2A==",
37     "decoded_payload": {
38       "batteryVoltage": 3.544,
39       "co2": 471,
40       "pressure": 100549.264,
41       "temperature": 32.89,
42       "tvoc": 10
43     },
44     "rx_metadata": [
45       {
46         "gateway_ids": {
47           "gateway_id": "ej23-01",
48           "eui": "3133303719004800"
49         },
50         "time": "2024-05-30T15:48:02.892361Z",
51         "timestamp": 200090820,
52         "rssi": -110,
53         "channel_rssi": -110,
54         "snr": -2,
55         "location": {
56           "latitude": 41.12318357229092,
57           "longitude": 1.1927601102615883,
58           "source": "SOURCE_REGISTRY"

```

Figura 8.18. Paquet captura pel nostre *gateway*.

8.3 Autonomia de la plataforma mòbil multisensor.

L'autonomia és un aspecte crucial en les plataformes mòbils multisensor, especialment en aplicacions de monitorització ambiental, seguretat, agricultura de precisió i exploració. Una plataforma amb una autonomia adequada pot operar durant períodes prolongats sense necessitat de recàrrega o manteniment freqüent, augmentant l'eficiència i reduint els costos operatius. A més a més, una bona autonomia permet que la plataforma es desplegui en llocs remots o de difícil accés, assegurant una recollida contínua de dades.

En aquest apartat, es valora l'autonomia de la nostra plataforma mòbil multisensor connectada personalitzable. Per tant, en la següent imatge s'observa l'evolució d'aquesta autonomia durant el transcurs del temps.

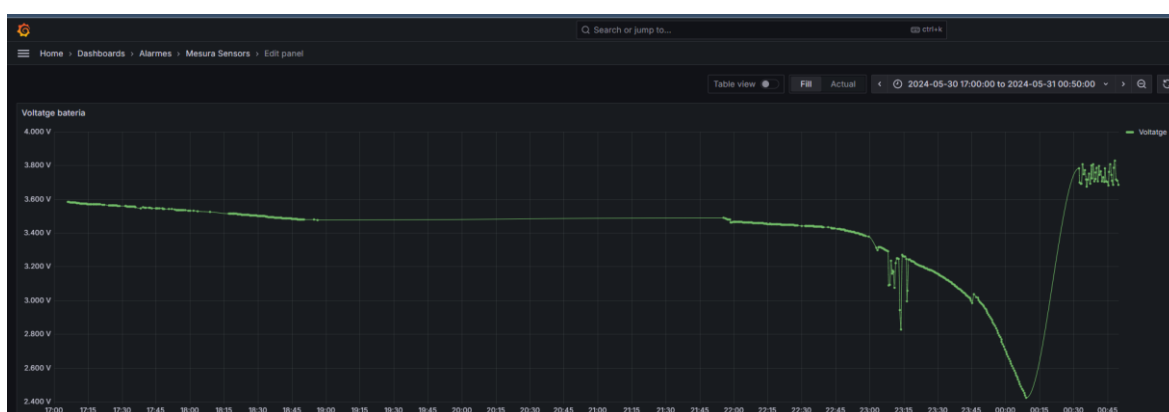


Figura 8.19. Evolució de l'autonomia de la plataforma mòbil.

Com podem observar, la ruta s'inicia a les 17:15h aproximadament amb un pic de 3.586 V fins les 18.30h. Posteriorment, fins les 22:00h la bateria ha esdevingut desconnectada per motius de falta de cobertura amb el *gateway*. Durant aquest període de temps, el voltatge de la bateria ha disminuït uns 0.1 V aproximadament. Finalment, la bateria ha estat en ús des de les 22:00h fins les 00:10h, on ha arribat fins als 2.426 V i per tant, el regulador de tensió no ha pogut mantenir els 3.3 V necessaris per al funcionament del circuit, fent que aquest deixi d'alimentar-se correctament. Un cop la bateria ha deixat de proporcionar voltatge, s'ha tornat a connectar la plataforma via USB i aquesta s'ha alimentat un altre cop fins als 3.785 V.

En termes generals, l'autonomia de la bateria ha estat de tres hores i mitja aproximadament, enviant dades quasi cada 25 segons de manera contínua. No obstant, la duració d'aquesta podria augmentar si es possessin els sensors en mode *sleep*.

9. CONCLUSIONS I LÍNIES FUTURES DEL PROJECTE

Per concloure, en aquesta última part del treball, es revisaran els objectius principals marcats a l'inici del projecte i s'extrauran conclusions respecte la feina feta durant tot aquest procés. Així mateix, es formalitzaran les millores i línies d'investigació futures a partir de les activitats plasmades en aquest Projecte Final de Carrera.

9.1 Conclusions.

En aquest treball de fi de grau, s'han assolit amb èxit els tres principals objectius establerts al començament del projecte. A continuació es detallen les conclusions extretes respecte a cada un dels punts.

Tal i com hem pogut observar, s'ha aconseguit desenvolupar i implementar una solució de monitorització ambiental que és no només de baix cost sinó també replicable i sense cost de llicència, utilitzant recursos de programari lliure. La solució inclou una plataforma mòbil equipada amb sensors capaços de mesurar diferents paràmetres ambientals com CO₂ i TVOCs. Aquesta plataforma ha demostrat ser efectiva en la recollida i transmissió de dades, oferint una eina potent i accessible per a la monitorització ambiental en temps real.

Per altra banda, s'ha realitzat una investigació detallada sobre la tecnologia LoRaWAN, que ha inclòs els seus principis de funcionament, característiques i avantatges. La investigació ha revelat que LoRaWAN és una tecnologia idònia per a aplicacions IoT gràcies a la seva capacitat de transmetre dades a llargues distàncies amb un consum d'energia mínim.

Pel que fa a la realització de les proves pilot, s'ha demostrat l'escalabilitat i la fiabilitat de la plataforma mòbil en diversos escenaris d'ús real. Aquestes proves han validat l'aplicabilitat de la plataforma en el camp de l'IoT, demostrant que és capaç de proporcionar dades precises i fiables en diferents condicions ambientals. Les proves han inclòs la mesura de paràmetres com la temperatura, humitat, i concentracions de CO₂ i TVOC, confirmant que la plataforma pot operar de manera contínua i eficient.

En conclusió, el projecte ha demostrat que la tecnologia LoRaWAN ofereix una plataforma viable per a la monitorització ambiental efectiva i econòmica. Les conclusions obtingudes representen un punt de partida sòlid per a futures investigacions i millores, amb l'objectiu d'ampliar les capacitats de la plataforma, augmentar la seva autonomia i escalabilitat, i aprofitar al màxim el potencial de les dades recollides per a la gestió ambiental sostenible.

9.2 Línies futures.

Respecte les diverses línies futures del projecte, hi ha diverses línies de recerca que es poden explorar per tal de millorar i ampliar les capacitats de la plataforma mòbil multisensor.

En primer lloc, treballar en ampliar la gamma de sensors incloent nous tipus de mesuradors que puguin captar altres contaminants o paràmetres ambientals com el NO₂, SO₂, o fins i tot sensors de soroll. A més a més, treballar en l'anàlisi de la precisió dels sensors actuals i explorar la possibilitat d'implementar sensors de major precisió o amb capacitat de calibratge automàtic.

En segon lloc, planificar i executar estudis de camp més extensos en diversos escenaris reals, com per exemple zones urbanes, rurals i industrials, amb la finalitat de validar l'eficiència de la plataforma en diversos contextos geogràfics. Paral·lelament, es podria col·laborar amb autoritats locals i institucions acadèmiques amb la finalitat d'obtenir suport i accés a diferents opcions que puguin enriquir el desenvolupament del projecte.

Finalment, es mostra un prototip d'un esquema de connexió d'un mòdul Heltec CubeCell amb una bateria i un panell solar. Aquesta configuració no es va arribar a implantar en aquest projecte, però es considera una solució ideal per proporcionar autonomia energètica a llarg termini. La integració d'aquest sistema permetria que la plataforma operés sense necessitat de connexió a una font d'alimentació externa, facilitant la seva utilització en ubicacions remotes

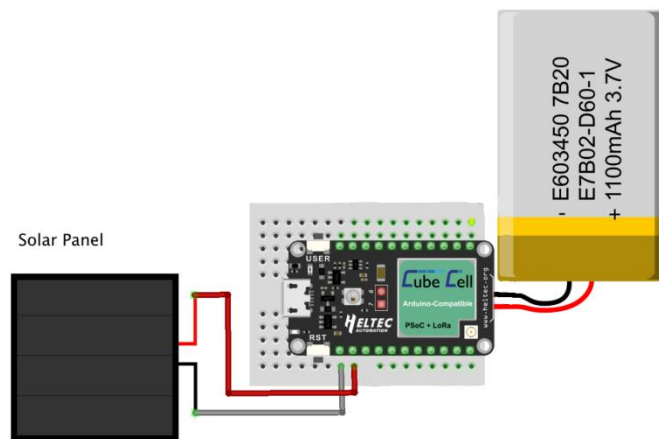


Figura 9.1. Esquema de connexió amb una bateria i un panell solar.

10. REFERÈNCIES BIBLIOGRÀFIQUES

- [1] Mesura de soroll ambiental i Smart Cities: un repte actual. (2016, junio 1). Keacoustics | Enginyeria experta en acústica, soroll i vibracions; Keacoustics - Consultoria Acústica Barcelona.
- [2] Qué es LoRa, cómo funciona y características principales. (2022, agosto 25). enco Electrónica. <https://www.vencoel.com/que-es-lora-como-funciona-y-caracteristicas-principales/>
- [3] Hector. (2020, June 1). *LoRaWAN – Parte 3 – Arquitectura y tramas*. El Bosque de Silicio. <https://elbosquedesilicio.es/lorawan3/>
- [4] *LoRa Gateways*. (n.d.). Maker Store by ALLNET. Retrieved May 6, 2024, from <https://maker-store.es/iot-lora-nb-iot-rfid-m2m-co/lora/lora-gateways/>
- [5] *What is the difference between OTAA and ABP Devices*. (2016, July 7). The Things Network. <https://www.thethingsnetwork.org/forum/t/what-is-the-difference-between-otaa-and-abp-devices/2723>
- [6] Reimondo, G. (2019, January 15). *Seguridad en Redes LoRaWAN™*. Tecnología Humanizada. <https://humanizationoftechnology.com/seguridad-en-redes-lorawan/revista/iot/01/2019/>
- [7] Rocha, A. M. D., Oliveira, M. A. D., José F. M., P., & Cavalheiro, G. G. H. (2023). ABP vs. OTAA activation of LoRa devices: an Experimental Study in a Rural Context. *2023 International Conference on Computing, Networking and Communications (ICNC)*, 630–634.
- [8] Schäfer, B. (n.d.). *LoRaWAN Firmware Update Over the Air - FUOTA*. Stackforce.com. Retrieved May 11, 2024, from <https://stackforce.com/en/about-us/news/lorawan-firmware-update-over-the-air-fuota>
- [9] Lavrut, F. (2020, June 22). *LoRaWAN roaming goes global thanks to actility's ThingPark Exchange* –. Actility. <https://www.actility.com/lorawan-roaming-goes-global-thanks-to-actilitys-thingpark-exchange/>
- [10] Reimondo, G. (2019, March 22). *Geolocalización en LoRaWAN*. Tecnología Humanizada. <https://humanizationoftechnology.com/geolocalizacion-en-lorawan/revista/2019/volumen-2-2019/03/2019/>
- [11] Semtech Corporation. (n.d.). *LoRaWAN® relay: A battery-powered network extender*. Semtech.com. Retrieved May 12, 2024, from <https://info.semtech.com/lorawan-relay-white-paper>
- [12] Sunny, A. I., Zhao, A., Li, L., & Kanteh Sakiliba, S. K. (2020). Low-cost IoT-based sensor system: A case study on harsh environmental monitoring. *Sensors (Basel, Switzerland)*, 21(1), 214. <https://doi.org/10.3390/s21010214>
- [13] Ranjan Laha, S., Pattanayak, B. K., & Pattnaik, S. (2022). Advancement of environmental monitoring system using IoT and sensor: A comprehensive analysis. *AIMS environmental science*, 9(6), 771–800. <https://doi.org/10.3934/environsci.2022044>
- [14] Sayed, E.; Ahmed, A.; Yousef, M.E. Internet of things in Smart Environment: Concept, Applications, Challenges, and Future Directions. (2019) www.mdpi.com/journal/sensors
- [15] Ullo, S.L.; Sinha, G.R. "Advances in Smart Environment Monitoring Systems Using IoT and Sensors." (2020)
- [16] Ayele, T.W.; Mehta, R. Air pollution monitoring and prediction using IoT. In Proceedings of the 2nd International Conference on Inventive Communication and Computational Technologies (ICICCT 2018), Coimbatore, India, 20–21 April 2018; pp. 1741–1745.
- [17] Primo-Peña, E., & Estrada-Lorenzo, J.-M. (2009). Las bases de datos bibliográficas españolas, un instrumento para el conocimiento y la difusión de la producción científica. *Seminarios de la Fundacion Espanola de Reumatologia*, 10(4), 132–141. <https://doi.org/10.1016/j.semreu.2009.09.004>

- [18] Kamal, R. Lesson 11 Internet Connected Environment (Weather, Air Pollution and Forest Fire) Monitoring. 2017, 141. Available online: <https://www.dauniv.ac.in/public/frontassets/coursematerial/InternetofThings/IoTCh12L11EnvironmentMonitoring.pdf> (accessed on 6 April 2020).
- [19] Air Quality Monitoring Using IoT and Big Data. Available online: https://www.gsma.com/iot/wp-content/uploads/2018/02/iot_clean_air_02_18.pdf (accessed on 31 May 2020).
- [20] Caurell, D. L. (2023, noviembre 17). Importancia de una base de datos en tiempo real. *ExpandLatam*. <https://www.expandlatam.com/blog/base-de-datos-en-tiempo-real/>
- [21] Keyrus. (2023, julio 10). La importancia de los datos en tiempo real: cómo ser relevante para tus clientes. LinkedIn.com. <https://www.linkedin.com/pulse/la-importancia-de-los-datos-en-tiempo-real-c%C3%B3mo-ser-relevante-para/>
- [22] Ekon, E. (2021, junio 22). Datos en tiempo real: cómo benefician a la gestión de empresas. Ekon; Cegid Ekon. <https://www.ekon.es/blog/como-procesar-y-analizar-datos-en-tiempo-real-en-la-nube/>
- [23] *La Importancia de la Monitorización en tiempo real para empresas*. (2024, febrero 28). ToBeIT. <https://tobeit.es/la-importancia-de-la-monitorizacion-en-tiempo-real-para-empresas/>
- [24] Follow, D. (2020, July 23). *Introduction of message queue telemetry transport protocol (MQTT)*. GeeksforGeeks. <https://www.geeksforgeeks.org/introduction-of-message-queue-telemetry-transport-protocol-mqtt/>
- [25] Fernández, Y. (2022, September 23). *Telegram: 46 trucos y consejos para exprimir al máximo una app que no sólo sirve para mensajería*. Xataka.com; Xataka Basics. <https://www.xataka.com/basics/telegram-45-trucos-consejos-para-exprimir-al-maximo-app-que-no-solo-sirve-para-mensajeria>

11. ANNEXOS

L'apartat d'annexes consta, en primer lloc, del diagrama de *Gantt* i de l'estructura WBS⁴² referent a tota la planificació del projecte. Tot seguit, s'adjunten els diferents codis Arduino usats per tal de dur a terme tot el desenvolupament del projecte, els quals no han estat presentats anteriorment en els apartats destinats a la seva aparició.

11.1 Estructura WBS i diagrama de Gantt.

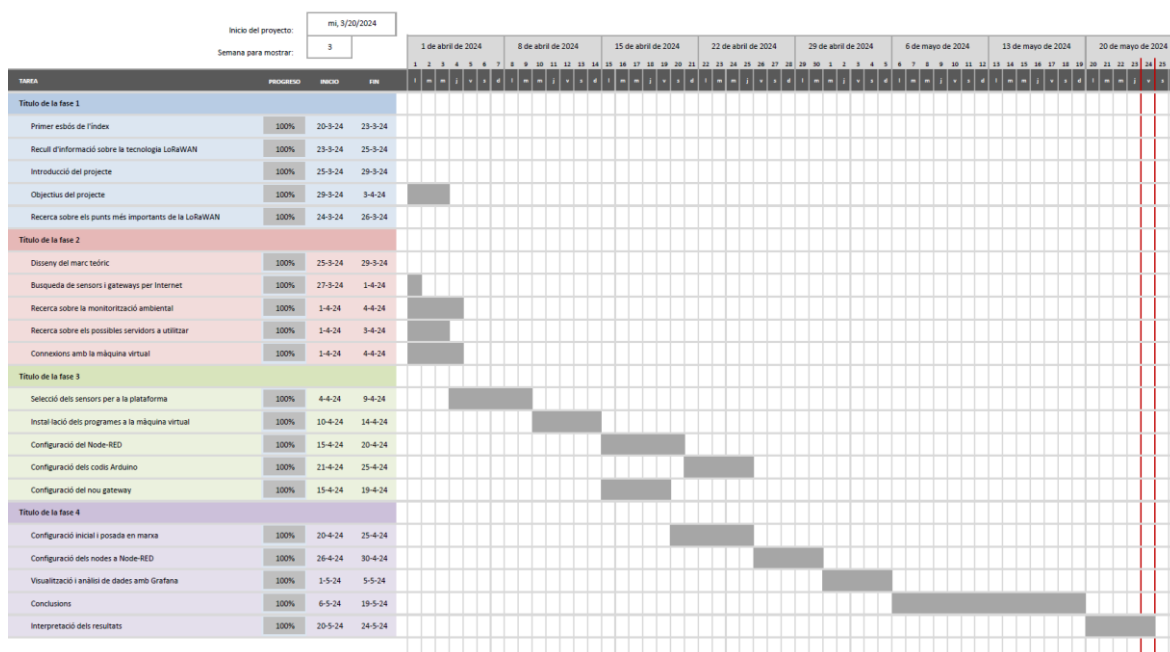


Figura 11.1. Diagrama de Gantt de la planificació del projecte.

La fase 1, en la qual es van realitzar les tasques inicials, incloent la recerca i preparació de la base teòrica sobre la tecnologia LoRaWAN va del 20 al 26 de març.

La fase 2, en la qual es va fer recerca sobre el mercat dels sensors ambientals, investigació sobre el maquinari i el programari necessari i disseny del marc teòric va del 25 de març al 4 d'abril.

La fase 3, en la qual es va fer la selecció i la configuració dels sensors, la plataforma de gestió i el sistema Node-RED va del 5 al 19 d'abril.

La fase 4, la qual es va dur a terme la integració final del sistema i la interpretació dels resultats obtinguts va del 20 d'abril al 24 de maig.

⁴² Work Breakdown Structure.

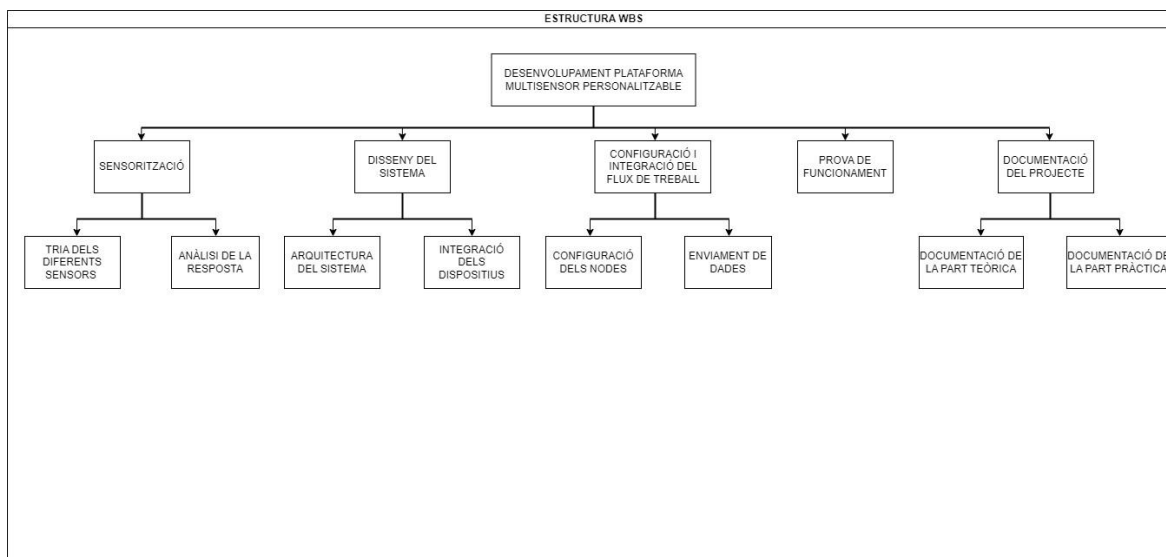


Figura 11.2. Estructura WBS del projecte.

11.2 Codis Arduino.

11.2.1 Programa sensor HW-611.

```

#include <Adafruit_BMP280.h>
#include <Adafruit_Sensor.h>
#include "LoRaWan_APP.h"
#include "Arduino.h"
#include <Wire.h>

/*
 * set LoraWan_RGB to Active, the RGB active in loraWan
 * RGB red means sending;
 * RGB purple means joined done;
 * RGB blue means RxWindow1;
 * RGB yellow means RxWindow2;
 * RGB green means received done;
 */
Adafruit_BMP280 bmp; // objecte sensor BMP280
/* OTAA para*/
uint8_t devEui[] = { 0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x06, 0x63, 0x05 };
uint8_t appEui[] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };
uint8_t appKey[] = { 0x57, 0x23, 0xAF, 0x79, 0xB5, 0x23, 0x5D, 0x56, 0xC9,
0x81, 0xF9, 0xE9, 0x47, 0x59, 0x65, 0x27 };

/* ABP para*/
uint8_t nwSKey[] = { 0x68, 0xD8, 0x27, 0x15, 0x3E, 0x3D, 0x81, 0x38, 0x8A,
0xC5, 0xC8, 0x3D, 0x83, 0xFB, 0x90, 0xCD };
uint8_t appSKey[] = { 0xA1, 0xB6, 0x4B, 0xEC, 0xC6, 0xF5, 0xBD, 0xA0, 0xA0,
0x12, 0xE9, 0x4C, 0xA6, 0x87, 0xD7, 0x26 };
uint32_t devAddr = ( uint32_t )0x260b2ea8;

/*LoraWan channelmask, default channels 0-7*/
uint16_t userChannelsMask[6]={ 0xFF00,0x0000,0x0000,0x0000,0x0000,0x0000
};

/*LoraWan region, select in arduino IDE tools*/
LoRaMacRegion_t loraWanRegion = ACTIVE_REGION;

```

```

/*LoraWan Class, Class A and Class C are supported*/
DeviceClass_t loraWanClass = LORAWAN_CLASS;

/*the application data transmission duty cycle. value in [ms].*/
uint32_t appTxDutyCycle = 20000;

/*OTAA or ABP*/
bool overTheAirActivation = LORAWAN_NETMODE;

/*ADR enable*/
bool loraWanAdr = LORAWAN_ADR;

/* set LORAWAN_Net_Reserve ON, the node could save the network info to
flash, when node reset not need to join again */
bool keepNet = LORAWAN_NET_RESERVE;

/* Indicates if the node is sending confirmed or unconfirmed messages */
bool isTxConfirmed = LORAWAN_UPLINKMODE;

/* Application port */
uint8_t appPort = 2;
/*!
* Number of trials to transmit the frame, if the LoRaMAC layer did not
* receive an acknowledgment. The MAC performs a datarate adaptation,
* according to the LoRaWAN Specification V1.0.2, chapter 18.4, according
* to the following table:
*
* Transmission nb | Data Rate
* -----|-----
* 1 (first)      | DR
* 2              | DR
* 3              | max(DR-1,0)
* 4              | max(DR-1,0)
* 5              | max(DR-2,0)
* 6              | max(DR-2,0)
* 7              | max(DR-3,0)
* 8              | max(DR-3,0)
*
* Note, that if NbTrials is set to 1 or 2, the MAC will not decrease
* the datarate, in case the LoRaMAC layer did not receive an acknowledgment
*/
uint8_t confirmedNbTrials = 4;

/* Prepares the payload of the frame */
static void prepareTxFrame( uint8_t port )
{
    float temperature = bmp.readTemperature(); // Llegeix la temperatura
    float pressure = bmp.readPressure(); // Llegeix la pressió

    // Imprimeix les lectures al monitor serial
    Serial.print("Temperatura: "); Serial.println(temperature);
    Serial.print("Pressió: "); Serial.println(pressure);

    int16_t temp_int = (int16_t)(temperature * 100); // Per exemple, 25.12
    C esdevindria 2512
    uint32_t pres_int = (uint32_t)(pressure * 1000); // Converteix Pa a
    hPa
    /*appData size is LORAWAN_APP_DATA_MAX_SIZE which is defined in
"commissioning.h".
    *appDataSize max value is LORAWAN_APP_DATA_MAX_SIZE.
    *if enabled AT, don't modify LORAWAN_APP_DATA_MAX_SIZE, it may cause

```

```

system hanging or failure.
    *if disabled AT, LORAWAN_APP_DATA_MAX_SIZE can be modified, the max
value is reference to lorawan region and SF.
    *for example, if use REGION_CN470,
    *the max value for different DR can be found in
MaxPayloadOfDatarateCN470 refer to DataratesCN470 and BandwidthsCN470 in
"RegionCN470.h".
    */
    appDataSize = 6;
appData[0] = (temp_int >> 8) & 0xFF; // Agafa els bits més significatius
de la temperatura.
appData[1] = temp_int & 0xFF; // Agafa els bits menys significatius de la
temperatura.
appData[2] = (pres_int >> 24) & 0xFF; // Agafa el byte més significatiu de
la pressió.
appData[3] = (pres_int >> 16) & 0xFF; // Agafa el segon byte més
significatiu de la pressió.
appData[4] = (pres_int >> 8) & 0xFF; // Agafa el tercer byte més
significatiu de la pressió.
appData[5] = pres_int & 0xFF; // Agafa els bits menys significatius de la
pressió.

    // Imprimeix les dades convertides al monitor seri
    Serial.print("Dades enviades: ");
    for (int i = 0; i < appDataSize; i++) {
        Serial.print(appData[i], HEX);
        Serial.print(" ");
    }
}

void setup() {
    Serial.begin(115200);
    // Inicia el bus I2C
    Wire.begin();
    if (!bmp.begin(0x76)) { // Inicialitza el BMP280
        Serial.println(F("Could not find a valid BMP280 sensor, check
wiring!"));
        while (1);
    }
#ifdef AT_SUPPORT
    enableAt();
#endif
    deviceState = DEVICE_STATE_INIT;
    LoRaWAN.ifskipjoin();
}

void loop()
{
    switch( deviceState )
    {
        case DEVICE_STATE_INIT:
        {
#ifdef LORAWAN_DEVEUI_AUTO
            LoRaWAN.generateDeveuiByChipID();
#endif
#ifdef AT_SUPPORT
            getDevParam();
#endif
            printDevParam();
            LoRaWAN.init(loraWanClass, loraWanRegion);
            deviceState = DEVICE_STATE_JOIN;
        }
    }
}

```

```

        break;
    }
    case DEVICE_STATE_JOIN:
    {
        LoRaWAN.join();
        break;
    }
    case DEVICE_STATE_SEND:
    {
        prepareTxFrame( appPort );
        LoRaWAN.send();
        deviceState = DEVICE_STATE_CYCLE;
        break;

    case DEVICE_STATE_CYCLE:
    {
        // Schedule next packet transmission
        txDutyCycleTime = appTxDutyCycle + randr( 0,
APP_TX_DUTYCYCLE_RND );
        LoRaWAN.cycle(txDutyCycleTime);
        deviceState = DEVICE_STATE_SLEEP;
        break;
    }
    case DEVICE_STATE_SLEEP:
    {
        LoRaWAN.sleep();
        break;
    }
    default:
    {
        deviceState = DEVICE_STATE_INIT;
        break;
    }
}
}
}

```

Codi 1. Codi programa sensor HW-611.

11.2.2 Programa sensor CCS811.

```

#include "LoRaWan_APP.h"
#include "Arduino.h"
#include <Wire.h>
#include <CCS811.h>

/*
 * set LoraWan_RGB to Active,the RGB active in loraWan
 * RGB red means sending;
 * RGB purple means joined done;
 * RGB blue means RxWindow1;
 * RGB yellow means RxWindow2;
 * RGB green means received done;
 */

/* OTAA para*/
uint8_t devEui[] = { 0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x06, 0x63, 0x05 };
uint8_t appEui[] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };
uint8_t appKey[] = { 0x57, 0x23, 0xAF, 0x79, 0xB5, 0x23, 0x5D, 0x56, 0xC9,
0x81, 0xF9, 0xE9, 0x47, 0x59, 0x65, 0x27 };

/* ABP para*/
uint8_t nwksKey[] = { 0x68, 0xD8, 0x27, 0x15, 0x3E, 0x3D, 0x81, 0x38, 0x8A,

```

```

0xC5, 0xC8, 0x3D, 0x83, 0xFB, 0x90, 0xCD };
uint8_t appSKey[] = { 0xA1, 0xB6, 0x4B, 0xEC, 0xC6, 0xF5, 0xBD, 0xA0, 0xA0,
0x12, 0xE9, 0x4C, 0xA6, 0x87, 0xD7, 0x26 };
uint32_t devAddr = ( uint32_t )0x260b2ea8;

/*LoraWan channelsmask, default channels 0-7*/
uint16_t userChannelsMask[6]={ 0x00FF,0x0000,0x0000,0x0000,0x0000,0x0000
};

/*LoraWan region, select in arduino IDE tools*/
LoRaMacRegion_t loraWanRegion = ACTIVE_REGION;

/*LoraWan Class, Class A and Class C are supported*/
DeviceClass_t loraWanClass = LORAWAN_CLASS;

/*the application data transmission duty cycle. value in [ms].*/
uint32_t appTxDutyCycle = 15000;

/*OTAA or ABP*/
bool overTheAirActivation = LORAWAN_NETMODE;

/*ADR enable*/
bool loraWanAdr = LORAWAN_ADR;

/* set LORAWAN_Net_Reserve ON, the node could save the network info to
flash, when node reset not need to join again */
bool keepNet = LORAWAN_NET_RESERVE;

/* Indicates if the node is sending confirmed or unconfirmed messages */
bool isTxConfirmed = LORAWAN_UPLINKMODE;

/* Application port */
uint8_t appPort = 2;
/*!
* Number of trials to transmit the frame, if the LoRaMAC layer did not
* receive an acknowledgment. The MAC performs a datarate adaptation,
* according to the LoRaWAN Specification V1.0.2, chapter 18.4, according
* to the following table:
*
* Transmission nb | Data Rate
* -----|-----
* 1 (first)      | DR
* 2              | DR
* 3              | max(DR-1,0)
* 4              | max(DR-1,0)
* 5              | max(DR-2,0)
* 6              | max(DR-2,0)
* 7              | max(DR-3,0)
* 8              | max(DR-3,0)
*
* Note, that if NbTrials is set to 1 or 2, the MAC will not decrease
* the datarate, in case the LoRaMAC layer did not receive an acknowledgment
*/
uint8_t confirmedNbTrials = 4;

/* Prepares the payload of the frame */
CCS811 ccs;

static void prepareTxFrame( uint8_t port )
{
    /*appData size is LORAWAN_APP_DATA_MAX_SIZE which is defined in

```

```

"commissioning.h".
 *appDataSize max value is LORAWAN_APP_DATA_MAX_SIZE.
 *if enabled AT, don't modify LORAWAN_APP_DATA_MAX_SIZE, it may cause
system hanging or failure.
 *if disabled AT, LORAWAN_APP_DATA_MAX_SIZE can be modified, the max value
is reference to lorawan region and SF.
 *for example, if use REGION_CN470,
 *the max value for different DR can be found in MaxPayloadOfDatarateCN470
refer to DataratesCN470 and BandwidthsCN470 in "RegionCN470.h".
 */
pinMode(Vext, OUTPUT);
digitalWrite(Vext, LOW);
pinMode(GPIO0, OUTPUT);
digitalWrite(GPIO0, LOW);
pinMode(GPIO4, OUTPUT);
digitalWrite(GPIO4, HIGH);
uint16_t co2=0;
uint16_t tvoc=0;
uint8_t i=0;
ccs.begin();
while(!ccs.available());

delay(5000); //wait for sensor

//Read 5 times to get the average value
while(i<5)
{
  ccs.available();
  {
    if(!ccs.readData())
    {
      co2 += ccs.getCO2();
      tvoc += ccs.getTVOC();
      i++;
    }
  }
}
co2/=i;
tvoc/=i;
Wire.end();
digitalWrite(Vext, HIGH);
digitalWrite(GPIO0, HIGH);
digitalWrite(GPIO4, LOW);
uint16_t batteryVoltage = getBatteryVoltage();

appDataSize = 6;
appData[0] = (uint8_t)(co2>>8);
appData[1] = (uint8_t)co2;
appData[2] = (uint8_t)(tvoc>>8);
appData[3] = (uint8_t)tvoc;
appData[4] = (uint8_t)(batteryVoltage>>8);
appData[5] = (uint8_t)batteryVoltage;

Serial.print("CO2: ");
Serial.print(co2);
Serial.print(" ppm,");
Serial.print(" TVOC: ");
Serial.print(tvoc);
Serial.print(" ppb,");
Serial.print("BatteryVoltage:");
Serial.println(batteryVoltage);

```

```
// Print data in hexadecimal format
Serial.print("Dades enviades: ");
for (int i = 0; i < appDataSize; i++) {
    Serial.print(appData[i], HEX);
    Serial.print(" ");
}
Serial.println();
}

void setup() {
    Serial.begin(115200);
#ifdef AT_SUPPORT
    enableAt();
#endif
    deviceState = DEVICE_STATE_INIT;
    LoRaWAN.ifskipjoin();
}

void loop()
{
    switch( deviceState )
    {
        case DEVICE_STATE_INIT:
        {
#ifdef LORAWAN_DEVEUI_AUTO
            LoRaWAN.generateDeveuiByChipID();
#endif
#ifdef AT_SUPPORT
            getDevParam();
#endif
            printDevParam();
            LoRaWAN.init(loraWanClass, loraWanRegion);
            deviceState = DEVICE_STATE_JOIN;
            break;
        }
        case DEVICE_STATE_JOIN:
        {
            LoRaWAN.join();
            break;
        }
        case DEVICE_STATE_SEND:
        {
            prepareTxFrame( appPort );
            LoRaWAN.send();
            deviceState = DEVICE_STATE_CYCLE;
            break;
        }
        case DEVICE_STATE_CYCLE:
        {
            // Schedule next packet transmission
            txDutyCycleTime = appTxDutyCycle + randr( 0, APP_TX_DUTYCYCLE_RND );
            LoRaWAN.cycle(txDutyCycleTime);
            deviceState = DEVICE_STATE_SLEEP;
            break;
        }
        case DEVICE_STATE_SLEEP:
        {
            LoRaWAN.sleep();
            break;
        }
    }
}
```

```

    default:
    {
        deviceState = DEVICE_STATE_INIT;
        break;
    }
}
}

```

Codi 2. Codi programa sensor CCS811.

11.2.3 Programa principal amb els sensors unificats.

```

#include <Adafruit_BMP280.h>
#include <Adafruit_Sensor.h>
#include "LoRaWAN_APP.h"
#include "Arduino.h"
#include <Wire.h>
#include <CCS811.h>

/* OTAA para */
uint8_t devEui[] = { 0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x06, 0x63, 0x05 };
uint8_t appEui[] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };
uint8_t appKey[] = { 0x57, 0x23, 0xAF, 0x79, 0xB5, 0x23, 0x5D, 0x56, 0xC9,
0x81, 0xF9, 0xE9, 0x47, 0x59, 0x65, 0x27 };

/* ABP para */
uint8_t nwksKey[] = { 0x68, 0xD8, 0x27, 0x15, 0x3E, 0x3D, 0x81, 0x38, 0x8A,
0xC5, 0xC8, 0x3D, 0x83, 0xFB, 0x90, 0xCD };
uint8_t appSKey[] = { 0xA1, 0xB6, 0x4B, 0xEC, 0xC6, 0xF5, 0xBD, 0xA0, 0xA0,
0x12, 0xE9, 0x4C, 0xA6, 0x87, 0xD7, 0x26 };
uint32_t devAddr = ( uint32_t )0x260b2ea8;

/* LoRaWAN channelmask, default channels 0-7 */
uint16_t userChannelsMask[6]={ 0xFF00,0x0000,0x0000,0x0000,0x0000,0x0000
};

/* LoRaWAN region, select in arduino IDE tools */
LoRaMacRegion_t loraWanRegion = ACTIVE_REGION;

/* LoRaWAN Class, Class A and Class C are supported */
DeviceClass_t loraWanClass = LORAWAN_CLASS;

/* the application data transmission duty cycle. value in [ms]. */
uint32_t appTxDutyCycle = 20000;

/* OTAA or ABP */
bool overTheAirActivation = LORAWAN_NETMODE;

/* ADR enable */
bool loraWanAdr = LORAWAN_ADR;

/* set LORAWAN_Net_Reserve ON, the node could save the network info to
flash, when node reset not need to join again */
bool keepNet = LORAWAN_NET_RESERVE;

/* Indicates if the node is sending confirmed or unconfirmed messages */
bool isTxConfirmed = LORAWAN_UPLINKMODE;

/* Application port */
uint8_t appPort = 2;

/* Number of trials to transmit the frame, if the LoRaMAC layer did not

```

```

receive an acknowledgment. */
uint8_t confirmedNbTrials = 4;

Adafruit_BMP280 bmp; // objecte sensor BMP280
CCS811 ccs;

/* Prepares the payload of the frame */
static void prepareTxFrame( uint8_t port )
{
    float temperature = 0;
    float pressure = 0;
    uint16_t co2 = 0;
    uint16_t tvoc = 0;
    uint8_t i = 0;
    uint16_t batteryVoltage = 0;
    uint8_t sensorType = 0; // 0 = no sensor, 1 = CCS811, 2 = BMP280, 3 =
    both

    // Comprovem si el sensor BMP280 està disponible
    if (bmp.begin(0x76) || bmp.begin(0x77)) {
        sensorType |= 0x02;
        temperature = bmp.readTemperature(); // Llegeix la temperatura
        pressure = bmp.readPressure(); // Llegeix la pressió
    }

    // Comprovem si el sensor CCS811 està disponible
    pinMode(Vext, OUTPUT);
    digitalWrite(Vext, LOW);
    pinMode(GPIO0, OUTPUT);
    digitalWrite(GPIO0, LOW);
    pinMode(GPIO4, OUTPUT);
    digitalWrite(GPIO4, HIGH);
    if (ccs.begin()) {
        sensorType |= 0x01;
        delay(5000); // wait for sensor

        // Read 5 times to get the average value
        while (i < 5) {
            if (ccs.available() && !ccs.readData()) {
                co2 += ccs.geteCO2();
                tvoc += ccs.getTVOC();
                i++;
            }
        }
        co2 /= i;
        tvoc /= i;
    }
    Wire.end();
    digitalWrite(Vext, HIGH);
    digitalWrite(GPIO0, HIGH);
    digitalWrite(GPIO4, LOW);
    batteryVoltage = getBatteryVoltage();

    // Imprimeix les lectures al monitor serial
    Serial.print("Temperatura: "); Serial.println(temperature);
    Serial.print("Pressió: "); Serial.println(pressure);
    Serial.print("CO2: "); Serial.print(co2); Serial.print(" ppm,");
    Serial.print(" TVOC: "); Serial.print(tvoc); Serial.print(" ppb,");
    Serial.print("BatteryVoltage:"); Serial.println(batteryVoltage);

    int16_t temp_int = (int16_t)(temperature * 100); // Per exemple, 25.12

```

```

C esdevindria 2512
uint32_t pres_int = (uint32_t)(pressure * 1000); // Converteix Pa a hPa

appDataSize = 13; // Un byte més per al sensorType
appData[0] = sensorType; // Primer byte indica el tipus de sensor
appData[1] = (temp_int >> 8) & 0xFF; // Agafa els bits més significatius
de la temperatura.
appData[2] = temp_int & 0xFF; // Agafa els bits menys significatius de
la temperatura.
appData[3] = (pres_int >> 24) & 0xFF; // Agafa el byte més significatiu
de la pressió.
appData[4] = (pres_int >> 16) & 0xFF; // Agafa el segon byte més
significatiu de la pressió.
appData[5] = (pres_int >> 8) & 0xFF; // Agafa el tercer byte més
significatiu de la pressió.
appData[6] = pres_int & 0xFF; // Agafa els bits menys significatius de
la pressió.
appData[7] = (uint8_t)(co2 >> 8);
appData[8] = (uint8_t)co2;
appData[9] = (uint8_t)(tvoc >> 8);
appData[10] = (uint8_t)tvoc;
appData[11] = (uint8_t)(batteryVoltage >> 8);
appData[12] = (uint8_t)batteryVoltage;

// Imprimeix les dades convertides al monitor serial
Serial.print("Dades enviades: ");
for (int i = 0; i < appDataSize; i++) {
  Serial.print(appData[i], HEX);
  Serial.print(" ");
}
Serial.println();
}

void setup() {
  Serial.begin(115200);
  // Inicia el bus I2C
  Wire.begin();
  if (!bmp.begin(0x76)) { // Inicialitza el BMP280 amb adreça 0x76
    Serial.println(F("Could not find a valid BMP280 sensor at 0x76, trying
0x77..."));
    if (!bmp.begin(0x77)) { // Inicialitza el BMP280 amb adreça 0x77
      Serial.println(F("Could not find a valid BMP280 sensor at 0x77, check
wiring!"));
      while (1);
    }
  }
}
#ifdef AT_SUPPORT
  enableAt();
#endif
deviceState = DEVICE_STATE_INIT;
LoRaWAN.ifskipjoin();
}

void loop()
{
  switch (deviceState)
  {
    case DEVICE_STATE_INIT:
    {
      #if(LORAWAN_DEVEUI_AUTO)
        LoRaWAN.generateDeveuiByChipID();
      #endif
    }
  }
}

```

```
#endif
#if (AT_SUPPORT)
    getDevParam();
#endif
printDevParam();
LoRaWAN.init(loraWanClass, loraWanRegion);
deviceState = DEVICE_STATE_JOIN;
break;
}
case DEVICE_STATE_JOIN:
{
    LoRaWAN.join();
    break;
}
case DEVICE_STATE_SEND:
{
    prepareTxFrame(appPort);
    LoRaWAN.send();
    deviceState = DEVICE_STATE_CYCLE;
    break;
}
case DEVICE_STATE_CYCLE:
{
    // Schedule next packet transmission
    txDutyCycleTime = appTxDutyCycle + randr(0, APP_TX_DUTYCYCLE_RND);
    LoRaWAN.cycle(txDutyCycleTime);
    deviceState = DEVICE_STATE_SLEEP;
    break;
}
case DEVICE_STATE_SLEEP:
{
    LoRaWAN.sleep();
    break;
}
default:
{
    deviceState = DEVICE_STATE_INIT;
    break;
}
}
}
```

Codi 3. Codi programa sensors unificats.