



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Escola Superior d'Enginyeries Industrial,
Aeroespacial i Audiovisual de Terrassa

Disseny i implementació d'una plataforma IoT industrial per a la gestió de dades industrials

Document:

Memòria

Autor/Autora:

Pau Viñas Francisco

Director - Codirector:

Miguel Delgado Prieto /
Ángel Fernández Sobrino

Titulació:

Grau en Enginyeria Electrònica
Industrial i Automàtica

Convocatòria:

Tardor 2021 Pròrroga 2022.

TREBALL DE FI D'ESTUDIS



Resum

El present treball de Fi d'Estudis té com a motivació l'estudi, el disseny, la implementació de l'arquitectura de comunicacions i dispositius necessaris pel desplegament d'una plataforma IIoT, a partir la qual connecti dispositius propis d'una planta de producció, amb aplicacions al núvol, per habilitar serveis remots de supervisió i control de producció. L'aplicació experimental s'ha realitzat als laboratoris de investigació MCIA de l'edifici GAIA i al laboratori Schneider de l'ESEIAAT, instal·lacions de la Universitat Politècnica de Catalunya ubicades a Terrassa. Concretament, el treball s'ha centrat en l'adquisició i gestió de dades d'un sistema automatitzat industrial i el posterior desplegament d'una aplicació de supervisió i control de producció, juntament amb el tractament de les dades adquirides per l'extracció d'informació rellevant.

L'estudi es focalitza en el marc de l'automatització basada en la indústria 4.0, donant importància en digitalitzar el procés productiu, tractar-lo remotament i fer un ús intel·ligent d'aquestes dades, amb l'objectiu de crear entorns productius automatitzats més flexibles, més econòmics, amb una comunicació més vertical i autosuficients, aportant recerca i estudi d'acord al concepte de Smart Factory, tant per extreure-hi dades útils i rellevants pel procés el qual s'apliqui, com per la col·laboració i transformació d'una societat cada cop més sostenible.

Abstract

This final degree project is motivated by the study, design and implementation of the communications architecture and devices necessary for the deployment of an IIoT platform, from which devices of a production plant are connected with cloud applications, to enable remote monitoring and production control services. The experimental application has been carried out in the MCIA research laboratory of the GAIA building, and in the Schneider laboratory of the ESEIAAT, facilities of the Polytechnic University of Catalonia located in Terrassa. Specifically, the work has focused on the acquisition and management of data from an industrial automated system and the subsequent deployment of a production monitoring and control application, together with the processing of the acquired data for the extraction of relevant information.

The study focuses on the framework of automation based on Industry 4.0, placing importance on digitising the production process, treating it remotely and making intelligent use of this data, with the aim of creating more flexible, more economical automated production environments, with more vertical and self-sufficient communication, providing research and study in accordance with the Smart Factory concept, both to extract useful and relevant data for the process being applied, and for the collaboration and transformation of an increasingly sustainable society.



Índex

RESUM.....	1
ABSTRACT.....	1
ÍNDEX	3
ÍNDEX DE TAULES	5
ÍNDEX DE FIGURES	6
LLISTA D'ABREVIATURES/GLOSSARI	8
1 INTRODUCCIÓ.....	9
1.1 OBJECTE	9
1.2 ABAST	9
1.3 REQUERIMENTS	10
1.4 JUSTIFICACIÓ.....	11
2 ANTECEDENTS I REVISIÓ DE L'ESTAT DE LA QÜESTIÓ	13
2.1 CONTEXT	13
2.1.1 INDÚSTRIA 4.0.....	13
2.1.2 SENSORITZACIÓ I ANALITZADOR DE XARXES.....	15
2.2 CAS D'ESTUDI	16
3 DISSENY DE LA SOLUCIÓ.....	18
3.1 INTRODUCCIÓ DE LA SOLUCIÓ PROPOSADA	18
3.2 DIAGRAMA CONCEPTUAL	19
4 IMPLEMENTACIÓ	21
4.1 INTRODUCCIÓ DE LA IMPLEMENTACIÓ	21
4.2 CONFIGURACIÓ ENTORN VIRTUAL	22
4.3 FABRICACIÓ ARQUITECTURA PLATAFORMA IOT.....	25
4.3.1 INSTAL·LACIÓ DE SOFTWARE EN CONTENIDORS	25
4.3.2 POSADA EN MARXA CONTENIDORS	27
4.3.3 NODE-RED.....	28
4.3.4 INFLUXDB.....	30
4.3.5 GRAFANA.....	32
4.4 CREACIÓ Y ABOCAMENT DE DADES A LA PLATAFORMA	34
4.5 TRACTAMENT DE DADES.....	36
4.6 VISUALITZACIÓ INFORMACIÓ ÚTIL I RESULTATS.....	40
5 ASSAJOS I VALIDACIONS	42
5.1 INTRODUCCIÓ DE LES PROVES REALITZADES	42
5.2 PROVES OPERACIONALS	42
5.2.1 VERIFICACIÓ CORRECTA INSTAL·LACIÓ DOCKER	42
5.2.2 CONNEXIÓ AL PLC I A L'ANALITZADOR DE XARXA	43
5.2.3 GENERACIÓ DE SENYALS ARTIFICIALS.....	46
5.2.4 ACCÉS A LA INFORMACIÓ DE L'API PÚBLICA	47
5.3 PROVES FUNCIONALS	48
5.3.1 CAPTACIÓ I ESCRIPTURA DE DADES DE L'ANALITZADOR	48
5.3.2 IMPLEMENTACIÓ SISTEMA FINAL DE TEMPS REAL.....	49

6	RESUM DEL PRESSUPOST I/O ESTUDI DE VIABILITAT ECONÒMICA	50
7	ANÀLISI I VALORACIÓ DE LES IMPLICACIONS AMBIENTALS I SOCIALS	50
8	CONCLUSIONS	51
9	REFERÈNCIES	53



Índex de taules

Títol i número de totes les taules per ordre d'aparició en el text.

<i>Taula 1. Llegenda figura 7. Elements de l'arquitectura de la plataforma IIoT.</i>	<i>19</i>
<i>Taula 2. Característiques primera màquina virtual configurada.</i>	<i>24</i>
<i>Taula 3. Característiques segona màquina virtual configurada.</i>	<i>28</i>
<i>Taula 4. Paràmetres de lectura de l'analitzador.</i>	<i>38</i>

Índex de figures

Títol i número de tots els gràfics per ordre d'aparició en el text.

<i>Figura 1. Eix temporal revolucions industrials.</i>	14
<i>Figura 2. Creixement dels IPF en tecnologies 4IR per centres globals d'innovació</i>	15
<i>Figura 3. Conjunt analitzador de xarxa i passarel·la del laboratori UPC.</i>	16
<i>Figura 4. Imatge de cel·la industrial del laboratori d'automatització.</i>	17
<i>Figura 5. Placa de característiques motor WEG instal·lat.</i>	17
<i>Figura 6. Imatge motor WEG instal·lat a la cel·la industrial.</i>	17
<i>Figura 7. Estructura general sistema CPS i plataforma IIoT.</i>	19
<i>Figura 8. Imatge captura pantalla Azure.</i>	23
<i>Figura 9. Connexió SSH amb la VM</i>	25
<i>Figura 10. Captura de pantalla edició fitxer .yml del docker-compose.</i>	26
<i>Figura 11. Aixecament dels tres contenidors.</i>	27
<i>Figura 12. Interfície de Node-Red.</i>	28
<i>Figura 13. Interfície de línies d'ordre d'InfluxDB.</i>	31
<i>Figura 14. Ordres de filtració i visualització de InfluxDB.</i>	32
<i>Figura 15. Interfície de Grafana.</i>	33
<i>Figura 16. Esquema proves de creació de missatges.</i>	34
<i>Figura 17. Codi JavaScript node funció.</i>	35
<i>Figura 18. Configuració node Influx batch.</i>	36
<i>Figura 19. Tractament dades 1.</i>	37
<i>Figura 20. Codi de requeriment node Modbus-Flex-Getter.</i>	37
<i>Figura 21. Tractament dades 2.</i>	39
<i>Figura 22. Interfície per programar alertes.</i>	40
<i>Figura 23. Panell de visualització de dashboards.</i>	41
<i>Figura 24. Verificació correcta instal·lació Docker.</i>	43
<i>Figura 25. Verificació connexió IP PLC.</i>	44
<i>Figura 25. Informació connexió UPCLink.</i>	45

<i>Figura 26. Verificació connexió IP Analitzador de Xarxa.</i>	<i>45</i>
<i>Figura 27. Codi JavaScript creació senyal sinus</i>	<i>46</i>
<i>Figura 28. Visualització per Grafana del sinus</i>	<i>47</i>
<i>Figura 29. Visualització depuració petició API.</i>	<i>48</i>
<i>Figura 30. Visualització dades enregistrades a InfluxDB</i>	<i>49</i>
<i>Figura 31. Visualització depuració dades captades.</i>	<i>49</i>
<i>Figura 31. Visualització dades enregistrades a InfluxDB</i>	<i>49</i>

Llista d'abreviatures/Glossari

IloT, Industrial internet of things, conjunt de sensors, instruments i dispositius autònoms connectats a través de internet a aplicacions industrials.

Smart Factory, fàbrica intel·ligent, aquella a la que s'aplica tecnologies emergents com la intel·ligència artificial, robòtica, Big Data i IloT, amb l'objectiu d'optimitzar els recursos i els processos pel fi de tenir més control i eficiència.

Indústria 4.0, quarta revolució industrial, nou model d'organització i de control de les cadenes de valor a partir de sistemes de fabricació recolzat per las tecnologies de la informació digitalitzades i la tecnologia IoT.

CPS, Cyberphysical System, sistema informàtic en el qual un mecanisme està controlat o monitoritzat per un ordinador mitjançant l'ús d'algoritmes i en els quals els components físics i de programari estan profundament entrelaçats.

Cloud computing, serveis de gestió de dades en llocs web.

TCP, Transmission Control Protocol, protocol orientat a la connexió que garanteix la transmissió correcta i ordenada de dades.

IP, Internet Protocol, protocol no orientat a la connexió, responsable de l'adreçament i fragmentació de paquets de dades en les comunicacions d'internet.

M2M, Machine-to-machine, comunicacions directes entre dues màquines i en qualsevol tipus de canal.

WAN, Wide Area Network o xarxa de llarg abast, conjunt de xarxes informàtiques d'àrea local que destaca per la seva extensió, normalment a nivell regional o estatal, fins i tot a nivell mundial com per exemple Internet.

1 Introducció

1.1 Objecte

El present Treball Final d'Estudis (TFE) té com a objecte principal l'estudi, el disseny, la implementació i la posterior validació de l'arquitectura de comunicacions i dispositius necessaris pel desplegament d'un sistema ciberfísic que connecti dispositius propis d'una planta de producció, amb aplicacions al núvol, per habilitar serveis remots de supervisió i control de producció. L'aplicació experimental es realitza als laboratoris de investigació MCIA de l'edifici GAIA i al laboratori Schneider de l'ESEIAAT, instal·lacions de la Universitat Politècnica de Catalunya ubicades a Terrassa.

Concretament, el treball es centra en l'adquisició i gestió de dades d'un sistema automatitzat industrial i el posterior desplegament d'una aplicació de supervisió i control de producció. Els desenvolupaments inclouen les prestacions d'una plataforma IIoT, amb capacitat de gestionar fluxos de dades i enregistrar-los en una base de dades al núvol segura, així com la monitorització remota de la informació en panells de visualització.

Aquest projecte es focalitza en el marc de l'automatització basada en la indústria 4.0, donant importància en digitalitzar el procés productiu, tractar-lo remotament i fer un ús intel·ligent d'aquestes dades, amb l'objectiu de crear entorns productius automatitzats més flexibles, més econòmics, amb una comunicació més vertical i autosuficients, aportant recerca i estudi d'acord al concepte de *Smart Factory*.

1.2 Abast

L'abast del treball considera el desenvolupament de la plataforma IoT, el disseny i la implementació dels flux de gestió de dades, i el desenvolupament de l'aplicació al núvol per la gestió de la seva visualització. Juntament amb corresponent protocol de comunicació industrial i la programació pertinent. D'altra banda, l'estudi no abasta la selecció de dispositius i components per la posada en marxa de la plataforma ja que integra els existents a les instal·lacions de la universitat, tant com el procés de programació del laboratori d'automatització a partir del PLC. Per tant es delimita a poder generar l'abocament de dades a la plataforma, tractar-les i gestionar-les per posteriorment visualitzar-les, extraient informació rellevant del procés industrial. La simulació del procés industrial que no entra a l'abast del projecte, sinó que s'agafaran dades d'execució de la planta del laboratori, posant-la en marxa, sense cap gestió programada dels PLCs que faci de simulació.

Es consideren principalment tres aspectes a definir en detall, dels quals es divideixen diferents paquets de treball.

1. Desenvolupament de la plataforma IIoT.
 - a. Configuració de l'entorn virtual (*cloud computing*) utilitzant una màquina virtual i un sistema de encapsulat de software. Posteriorment creació de l'arquitectura IIoT on es podrà desenvolupar els aspectes relacionats amb les comunicacions industrials.
 - b. Posada en marxa de les aplicacions que realitzaran funcions específiques. Es desitja dissenyar i implementar els següents aspectes:

En primer lloc, els fluxos de dades del sistema, a través del software inicialment definit Node-RED.

En segon lloc, la creació d'un registre a través de la connexió inicialment definida amb InfluxDB, una base de dades de codi obert desenvolupada per InfluxData, amb possibilitat d'altres opcions similars.

Finalment, la monitorització i pantalla de visualització de les dades registrades a través del software definit inicialment Grafana.

2. Implementació de l'arquitectura del sistema d'aplicació al núvol, les comunicacions i el processament de dades: Es desitja introduir els fluxos de dades provinents de la planta a la plataforma IIoT, mitjançant un protocol de comunicació, en aquest cas TCP/IP.
3. Integració d'anàlisis de dades al núvol: Estudi del procés industrial i disseny de l'extracció de dades significatives a partir d'algoritmes, processats de dades, intel·ligència artificial, etc. Implementació sobre MATLAB Online d'un únic script amb tots els desenvolupaments aconseguits. Posteriorment integració a l'arquitectura IIoT generada, l'aplicació al núvol de l'anàlisi de dades, de forma que es pugui visualitzar per Grafana.

1.3 Requeriments

Amb la finalitat que aquest projecte sigui concloent, i serveixi d'utilitat dins del marc de l'automatització a l'indústria 4.0, així com a l'estudi de noves formes descentralitzades, remotes i globals de poder analitzar la informació en un entorn productiu, es defineixen els següents requeriments:

- L'automatització de la gestió de fluxos de dades, en aquest cas duta a terme inicialment pel software Node-RED.
- L'enregistrament de les dades adquirides en una *database*, en aquest cas duta a terme inicialment pel software InfluxDB.
- La visualització de les dades i monitorització d'aquestes remotament, en aquest cas duta a terme inicialment pel software Grafana.
- Disseny i implementació d'una estructura de contenidors de software (containers) mitjançant Docker.
- Execució d'operacions de lectura i escriptura a bases de dades des d'aplicacions capaces de processar les dades i obtenir informació de valor afegit.
- Possibilitat d'emmagatzemar l'arquitectura desenvolupada en un servidor remot, inclús de poder-la clonar.
- Anàlisis d'informació per veure el consum del procés industrial i desenvolupament amb la finalitat de executar un procés més sostenible.

- Capacitat de transmetre les dades del sistema automatitzat en temps real a la plataforma IIoT.

1.4 Justificació

Actualment els processos industrials o entorns productius estant entrant a una progressió a nivell d'automatització més avançada, on la personalització extrema del procés juntament amb una alta flexibilitat, la reducció de costos, la traçabilitat dels productes, la gestió remota, la velocitat i sobretot eficiència són factors molt importants per l'òptim funcionament i per destacar al sector industrial. El control, la supervisió i la millora d'aquests processos automatitzats, ja sigui per adquisició de dades, organització dels fluxos d'informació, la gestió de les comunicacions, entre d'altres, fa que es converteixin en una eina fonamental dins d'aquests processos, on cada cop és més comú i es busquen noves tecnologies per administrar les dades de manera descentralitzada, des de diferents dispositius, amb el registre i control dels mateixos de forma remota.

És per això, que aquest projecte pretén abordar el disseny i la implementació d'una arquitectura CPS, juntament amb la rellevància del IIoT, ja que permet a tots els dispositius està connectats a la mateixa xarxa, en aquest cas internet (WAN), donant-li un alt potencial d'aplicacions i solucions. Aquest treball tracte d'ampliar el contingut que hi ha sobre aquesta temàtica, polir i millorar els diferents sistemes que hi ha de gestionar els fluxos de dades i l'adquisició d'informació, ajudant així a perfeccionar un servei el qual es pugui utilitzar a la indústria real, a una empresa o fàbrica en concret.

Primer de tot ens hem de centrar en la primera capa del procés, els equips electrònics que avui en dia controlen en temps real, processos seqüencials a base del seu programa lògic intern i executen els diferents actuadors del sistema, rebent diferent informació, ja sigui per diferents sensors o captadors, en els sistemes automatitzats actuals, són els Controladors Lògics Programables (PLC). La importància que aquest estigui programat correctament per actuar a l'entorn productiu sense cap problema i rebí informació veraç amb els diferents protocols el qual es comunica amb els diferents sensors i actuadors, és la mateixa amb la que aquest ha de comunicar-se amb el núvol, ja que serà des d'allà, on es controlarà i es gestionaran les dades captades d'aquest projecte. Els diferents dispositius de planta normalment son connectats via industrial ethernet (LAN), l'entrada al *cloud* es fa amb un dispositiu central o ordinador de planta connectat tant a la LAN com a la WAN, on a partir del protocols TCP/IP i a través d'un software de gestió de fluxos d'informació com node-RED, pots derivar i enregistrar la informació cap a una base de dades al núvol, emmagatzemada en algun servidor remot. La tendència en l'àmbit de la indústria 4.0 serà que cada vegada hi hagi més descentralització fins i tot dins la mateixa fàbrica on cada màquina es podrà connectar a internet, així establir una comunicació màquina a màquina (M2M), amb un software que permeti autogestionar-se i que estarà ubicat al núvol.

Per altra banda, la interacció amb els humans segueix sent igual d'important, i a nivell pràctic dins les empreses es necessita per el control i supervisió dels sistemes, per corroborar que no hi hagi cap error i per veure el correcte funcionament.

L'actual projecte tracta de donar una solució vàlida tant a la gestió de l'entrada i sortida de dades a la plataforma IIoT com a la monitorització i supervisió per pantalla dels processos que estant passant i informació que s'està rebent tant de les màquines a través de l'analitzador de xarxes com del mateix procés automatitza't, a partir de sensors, donant una capacitat de resposta del sistema a temps real, o detecció d'anomalies.

Finalment, el processat de dades a partir de eines online dins de l'entorn CPS i el qual podem emmagatzemar en un servidor remot, permet un control de les dades que es va generant a la línia industrial, extreure dades significatives i poder prendre diferents decisions, fins i tot predir el comportament de la màquina o del sistema automatitza't. Així doncs dona una personalització, adaptació a l'entorn i a noves tasques de les màquines

que fa canviar tot el paradigma de les gestions dins d'una fàbrica. Els avantatges que genera aquesta tecnologia són entre d'altres generar una resposta intel·ligent ja programada en base al temps de processament dels productes, en base les dates d'entrega o en base a objectius de benefici o sostenibilitat.

2 Antecedents i revisió de l'estat de la qüestió

2.1 Context

Un dels motius pels quals s'efectua la revolució cognitiva en l'ésser humà és pel llenguatge que els sàpiens vam saber desenvolupar.

“¿Què té de tant especial el nostre llenguatge, doncs?”

La resposta més habitual és que el nostre llenguatge és increïblement flexible. Podem enllaçar un nombre limitat de sons i símbols per produir un nombre infinit de frases, totes amb un significat diferent. Per tant, podem rebre, emmagatzemar i comunicar una quantitat prodigiosa d'informació sobre el món que ens envolta.” (Yuval Noah Harari 2011)

L'actual estudi es contextualitza com bé s'ha justificat a l'apartat anterior en la implementació i desenvolupament actual de l'anomenada quarta revolució industrial on la distribució de la informació i la connexió a internet juguen un paper fonamental per tal d'aconseguir un flux de dades major, la possibilitat de compartir i comunicar molta més informació del món que ens envolta, més concretament en el món industrial, dades que indirectament compartim a nivell humà, però que directament es traspassen a nivell de màquina, donant-li així un potencial exageradament gran a desenvolupar intel·ligències en aquest cas artificials.

En les dos últimes dècades s'han generat varis estudis i avenços, s'han implementat diverses solucions d'aquest camp de la ciència tant ampli, actualment gran part de les indústries d'arreu del món comencen a utilitzar aquest tipus de tecnologies per tal de globalitzar i poder tractar amb sistemes CPS. Avenços que venen molt lligats de la 3a revolució industrial on la electrònica i la informàtica van permetre ja l'automatització de tasques, càlculs i tractament complex de la informació.

2.1.1 Indústria 4.0

La indústria 4.0 és la conseqüència de la quarta revolució industrial. Segons el diccionari de l'institut d'estudis catalans la revolució industrial ve definida com a *“moviment social i econòmic caracteritzat per l'aparició d'una gran quantitat de màquines que substitueixen l'esforç humà en les tasques productives.”* És sabut que a la segona meitat del segle XVIII a partir de la mecanització d'enginyers els quals funcionaven a partir d'aigua i vapor, s'inicià la primera revolució industrial la qual transformà els éssers humans que fins aquell moment es centraven en una societat agrícolta i ramadera, en una societat manipuladora de màquines, i en una societat més industrial on les ciutats van agafar gran importància. Aquests canvis alteraren posteriorment les bases econòmiques globals.

Seguidament va venir la segona revolució industrial a partir de l'aprofitament de noves energies, d'entre elles una de les més importants, l'electricitat, que va permetre fer funcionar i permet actualment encara, tot tipus de màquines les quals faciliten i realitzen tasques que fins el moment feien els humans, alhora que va permetre el començament de la producció en massa, i en conseqüència nous sistemes socioeconòmics, com per exemple el capitalisme.

La tercera revolució industrial ve definida per l'aparició de l'electrònica i els sistemes en tecnologia de la informació, també coneguda com a revolució digital, va transformar totalment la gestió de les dades a partir de la digitalització de senyals elèctrics on es podia tractar, emmagatzemar i automatitzar càlculs i tasques de gran quantitat d'informació. Els dispositius analògics i mecànics queden en un segon pla, i sorgeixen tota mena de dispositius digitals com els primers robots industrials o màquines de control numèric, entre

d'altres. Per tant des del punt de vista industrial a part de producció en massa obtenim l'automatització de la producció a gran escala.

Finalment molt lligada a l'anterior i augmentant exponencialment tant les possibilitats, solucions a aplicar, com la nostra manera de viure, treballar i relacionant-se, sorgeix la quarta revolució industrial, la qual no es podria entendre sense les anteriors, i on els sistemes ciberfísic, que combinen infraestructura física amb software, sensors, nano tecnologia i electrònica digital, juntament amb la connexió de totes aquestes dades a una gran xarxa de compartició com podria ser la internet a partir de tecnologia IoT, ja han començat a transformar la indústria actual, anomenant així l'origen de la indústria 4.0. Els avenços tecnològics emergents com la intel·ligència artificial, el big data, la robòtica, l'internet de les coses, els vehicles autònoms, la impressió 3D, la biotecnologia, la ciència dels materials, l'emmagatzematge d'energia, la computació quàntica, entre d'altres, desenvoluparan les indústries, sorgint així les anomenades fàbriques intel·ligents.

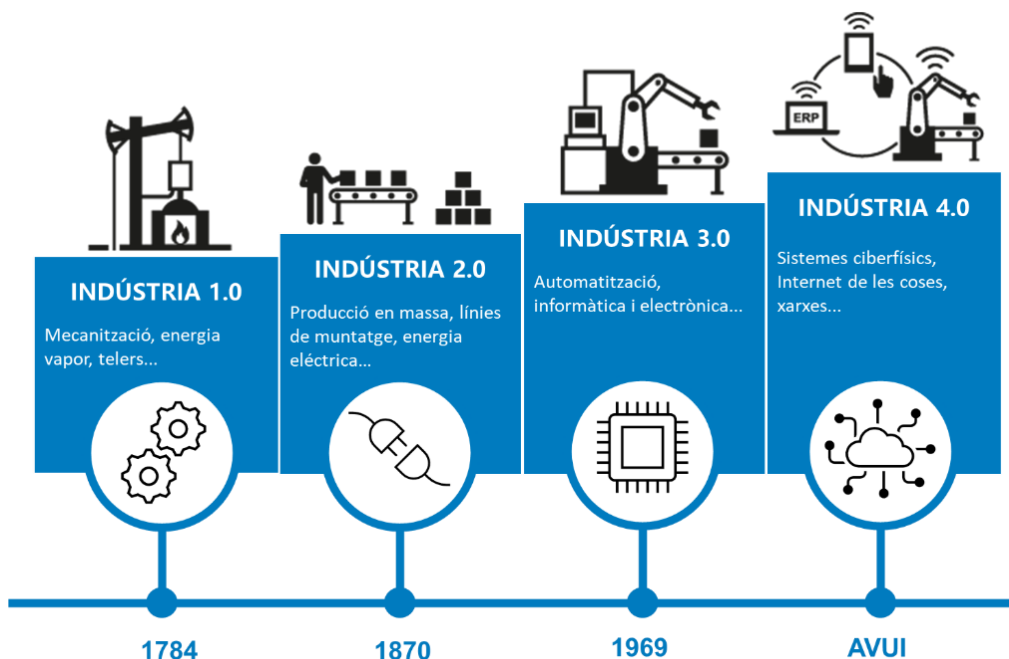


Figura 1. Eix temporal revolucions industrials.

Font: <https://kemptechnologies.com/blog/the-4th-industrial-revolution/>

Elaboració pròpia

Per altre banda tots aquests avenços donaran lloc a la possibilitat de milers de milions de persones connectades per dispositius mòbils, amb una potència de processament, capacitat d'emmagatzematge i accés al coneixement sense precedents, les quals multiplica encara més la velocitat de la transformació, deixant-la així sense límits. Les constel·lacions de dispositius intel·ligents, connectats, que es comuniquen entre ells, en gran majoria per internet, i que comparteixen dades de manera autònoma en temps real, representaran una quarta part de tot el trànsit d'Internet el 2025.

El desembre de 2020, l'EPO¹ va presentar un estudi basat en la darrera informació disponible sobre patents per analitzar les tendències d'innovació en la indústria 4.0. Es va

¹ EPO (European Patent Office): Organització encarregada de l'aplicació administrativa del Conveni sobre la Patent Europea.

analitzar totes les famílies de patents internacionals, es van identificar 264.565 IPF² amb data de presentació entre l'any 2000 i 2018. Aquesta amplia cohort³ d'invents proporciona una visió extraordinària de l'estat de la transformació digital que afecta l'economia global.

Les principals troballes de l'estudi són que el ritme de la innovació global en tecnologies 4IR⁴ ha accelerat vigorosament durant l'última dècada, amb una taxa de creixement anual mitjana de patents propera al 20% entre 2010 i 2018, en comparació amb el 12,8% entre 2000 i 2009 (figura 2). L'augment anual de famílies de patents internacionals (IPF) per a tecnologies 4IR ha estat gairebé cinc vegades més gran que el creixement de les IPF en tots els camps des del 2010 (4,2%). Com a resultat, els objectes connectats intel·ligents van representar més de l'11% de tota l'activitat de patents a tot el món el 2018, amb prop de 40.000 IPF nous només el 2018.

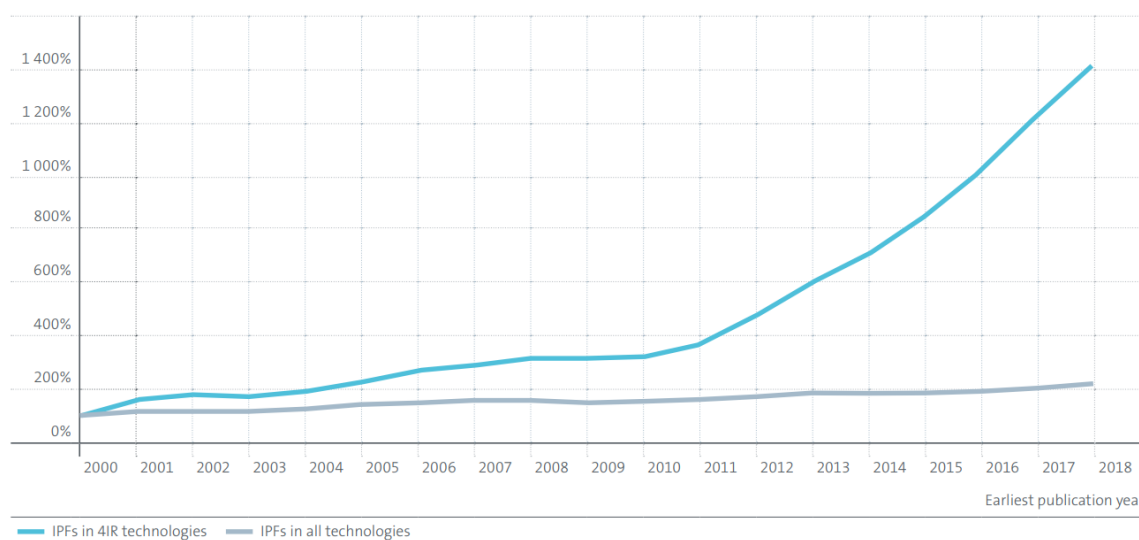


Figura 2. Creixement dels IPF en tecnologies 4IR per centres globals d'innovació, 2000-2018.

Font: European Patent Office, 2020, *Patents and the Fourth Industrial Revolution*.

Fa milers d'anys el llenguatge humà ens va donar la possibilitat de captar, emmagatzemar i comunicar informació del món que ens envolta, i en conseqüència l'esser humà va evolucionar cognitivament, avui en dia aquesta possibilitat de compartir i intercanviar informació entre intel·ligències ja siguin humanes o artificials és molt més extensa i global, fet que dona pas a aquesta transformació o revolució, en el món manufacturer i de la fabricació, anomenada indústria 4.0.

2.1.2 Sensorització i Analitzador de xarxes

A l'hora de captar dades i informació del que està passant als processos industrials i ha una gran gama d'eines de mesura que passen des de sensors, sortides digitals o analògiques dels PLC, analitzadors de xarxa, etc.

² **IPF (International Patent Family)**: Invencions per a les quals s'han presentat sol·licituds de patent en dos o més territoris.

³ **Cohort**: en estadística, conjunt de persones o casos que se sotmeten a observació durant un període concret de temps.

⁴ **4IR (4th Industrial Revolution)**: Quarta revolució industrial

En el nostre cas ens centrarem en l'analitzador de xarxa ja que serà l'eina utilitzada en el treball per extreure informació.

Un analitzador de xarxes es un equip dissenyat per la mesura i el càlcul dels paràmetres elèctrics de la línia elèctrica situada aigües avall del punt de mesura o connexió. En l'actual mercat existeixen diferents tipus d'analitzador de xarxa, ofereixen diverses prestacions com per exemple diferent tipus de comunicacions, algunes més ràpides que d'altres, amb major o menor precisió de mesura, amb més o menys paràmetres de lectura, etc.

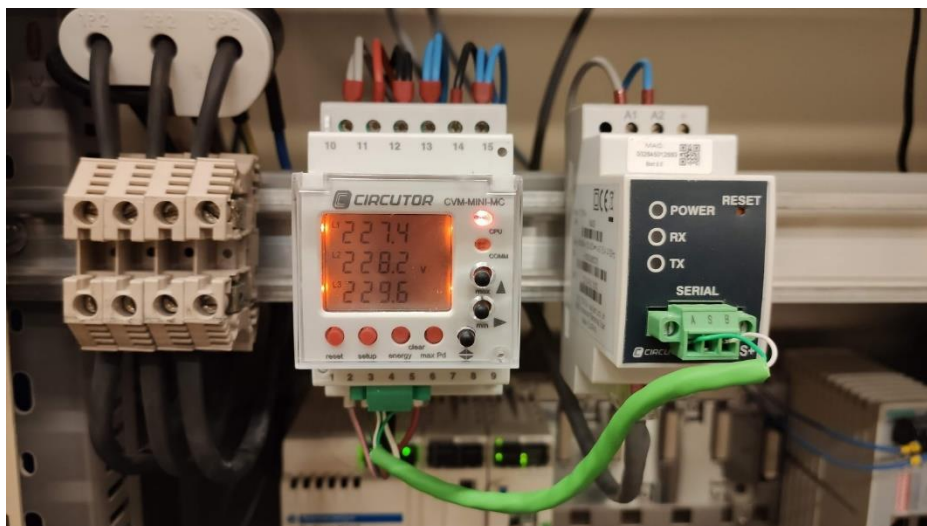


Figura 3. Conjunt analitzador de xarxa i passarel·la del laboratori UPC.

Font: Elaboració pròpia.

2.2 Cas d'estudi

Els antecedents dels quals es parteix en el present estudi i que es continuen desenvolupant són d'anteriors treballs finals d'estudis i de projectes del màster universitari en Enginyeria de Sistemes Automàtics i Electrònica Industrial. En els quals es fa un estudi i automatització de les diferents etapes d'una aplicació industrial basada en el transport i la gestió de peces.

És per això important exposar el cas d'estudi en el qual es treballa en aquest projecte i on es dur a terme, en aquest cas l'aplicació experimental es realitza a la cel·la industrial de producció (*figura 4*) basada en transport i gestió de peces ubicada a l'edifici TR2 de la Universitat Politècnica de Catalunya a la localitat de Terrassa.

La funció de la cel·la és bàsicament simular un procés automatitzat de transport de safates o palets mitjançant corretges accionades per motors, juntament amb subjectadors per parar el flux i amb plataformes per canviar la seva direcció. A partir de PLCs es gestiona els diferents elements connectats, programant adequadament depenent del flux que es vulgui a la cel·la. En total hi ha quatre tipus de comunicació de bus de camp a la instal·lació els quals són Profibus, CAN, Ethernet i ASI-INTERFACE.



Figura 4. Imatge de cel·la industrial del laboratori d'automatització.

Font: Elaboració pròpia

Així doncs els diferents elements de la cel·la industrial la qual s'ha basat aquest estudi són:

WEG Württ. Elektromotoren G.m.b.H. Balingen					
Typ	OD 732 T			3 842 503	
220-240/380-415		◀ Volt ▶	Y 440-480		
50	2,2/1,25	◀ Amp. ▶	1,1		
Hz	2800	◀ min ⁻¹ ▶	3360		
0,550 kW		cos φ	0,85	0,85	0,550 kW
Mot.-Nr.	2 330 076	FD	565	IP	54 IsKI B
DIN VDE 0530			Made in Germany		

Figura 5. Placa de característiques motor WEG instal·lat.

Font: Elaboració pròpia

Motors trifàsics: Generen el moviment mecànic de les cintes a partir d'electricitat.

Els motors instal·lats a la cel·la són de la firma WEG, de baixa potència ja que no superen el kW de potència nominal, com es pot observar a la figura cinc, al ser motors trifàsics poden funcionar tant a 440V com a 230V, depèn de si estan connectats en estrella o en triangle, a una freqüència de 50Hz, l'estàndard Europa.

Cintes transportadores: permeten transportar les safates.

Sensors inductius: Permeten detectar elements metàl·lics.

Electrovàlvules: Vàlvules que s'actuen mitjançant electricitat, segons el principi del camp magnètic creat a l'eix d'un bobinat quant a través d'aquest i circula una certa intensitat. Permeten accionar els subjectadors o plataformes a través de la neumàtica.



Figura 6. Imatge motor WEG instal·lat a la cel·la industrial.

Font: Elaboració pròpia

Plataformes: Elements situats a les interseccions els quals permeten canviar o realitzar correctament el flux de safates.

Subjectadors: Elements que permeten subjectar i retenir les safates a través de les electrovàlvules.

Polsadors: *Switch* que permet interrompre el pas del corrent elèctric mentre es manté polsat, al deixar-lo anar torna a la seva posició inicial. En el cas del **polsador d'emergència** permet realitzar una parada en una situació de perill. El mecanisme incorpora un enclavament que manté l'estat una vegada accionat.

Quadre elèctric: Altrament anomenat quadre de distribució, és l'encarregat de gestionar la potència elèctrica adequada per cada equip o aparell de la instal·lació i que per tant anirà connectat a la xarxa elèctrica, per altre banda també conté fusibles, proteccions magneto tèrmiques i diferencials que protegeixen els circuits que componen la instal·lació.

3 Disseny de la solució

3.1 Introducció de la solució proposada

La possibilitat de poder simular el flux de safates o palets a la cel·la industrial de la universitat, activant els diferents línies amb els seus motors pertinents, amb diferents parades i arrencades, amb més temps de funcionament o menys, imitant així el que podria ser el funcionament en una indústria real, permet poder adquirir dades de la cel·la mitjançant PLCs, l'analitzador de xara, diferents sensors, etc.

La solució proposada en l'actual treball consta d'implementar la plataforma IIoT amb els protocols necessaris per poder captar les dades de la cel·la, a partir de l'analitzador de xarxa i la passarel·la TCP/IP, consultar-les remotament i processar-les a la plataforma per extreure informació útil, seguidament fer un monitoratge remot via web, qual cosa permetrà actuar en conseqüència les dades captades o analitzar i estudiar els diferents comportaments, amb un entorn de connexió, de gestió del flux de dades i el seu tractament molt més mal·leable, comparat amb una gestió només a planta, obrint així una infinitat de possibilitats a l'hora de gestionar la informació, per obtenir-ne benefici.

Com a solució també es presenta poder captar i fer una auditoria energètica del consum simulat de la cel·la, extraient-ne així el consum tant sostenible com econòmic i poder veure l'estalvi que es pot generar, juntament amb l'estudi dels harmònics totals per tal de poder millorar eficiència o alertar si hi ha pics de soroll. La monitorització en temps real i el tractament de les dades permet tenir control del que està passant físicament de manera virtual o informàticament des de qualsevol punt remot en connexió al núvol.

El disseny d'aquest estudi s'ha realitzant pensant amb les diferents opcions de software i les seves estructures, per tal de poder complir els objectius proposats, la solució passa per implementar aquesta plataforma en un servidor remot, amb els softwares necessaris per tal de captar les dades de l'analitzador de xarxa a partir de protocol TCP/IP, poder gestionar i tractar les dades, tenir capacitat d'emmagatzemar-les i poder-les consultar, i finalment, capacitat de mostrar visualment aquestes dades i poder fer-ne la monitorització, disseny que en el següent apartat es pot visualitzar a l'esquema conceptual.

3.2 Diagrama conceptual

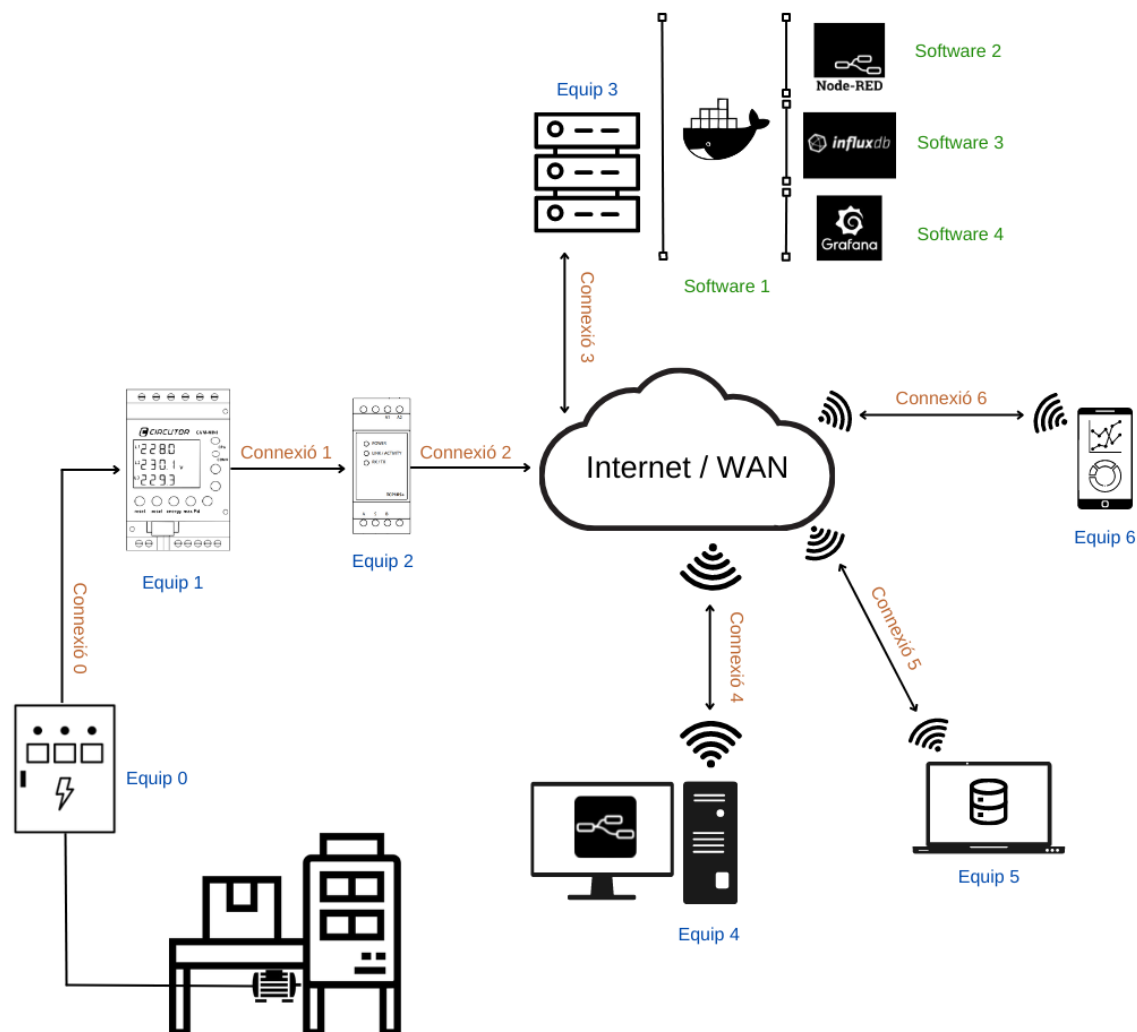


Figura 7. Estructura general sistema CPS i plataforma IIoT.

Font: Elaboració pròpia

Referència	Descripció	Referència	Descripció
Equip 0	Quadre elèctric	Connexió 0	Línia elèctrica trifàsica
Equip 1	Analitzador de xarxa CVM-MINI-MC	Connexió 1	RS485 (Modbus RTU®)
Equip 2	Passarel·la TCP1RS+	Connexió 2	Ethernet (TCP/IP)
Equip 3	Servidor remot (VM)	Connexió 3	Ethernet (TCP/IP)
Equip 4	Qualsevol PC que accedeix plataforma	Connexió 4	Ethernet (TCP/IP)
Equip 5	Qualsevol portàtil que accedeix plataforma	Connexió 5	Ethernet (TCP/IP)
Equip 6	Qualsevol mòbil que accedeix plataforma	Connexió 6	Ethernet (TCP/IP)
Software 1	Software Docker	Software 3	Software InfluxDB
Software 2	Software Node-RED	Software 4	Software Grafana

Taula 1. Llegenda figura 7. Elements de l'arquitectura de la plataforma IIoT.

En el diagrama de la figura set es pot observar l'esquema de la solució dissenyada, amb la seva arquitectura, les diferents connexions, equips i softwares. Cal destacar que es un diagrama de caire conceptual on l'objectiu es poder-se fer una idea de l'arquitectura utilitzada com a solució d'aquest projecte, de manera visual, i no pretén ser un esquema d'alta precisió i detall. A la *Taula 1* s'observa la llegenda de la figura set amb els diferents elements de l'arquitectura de la plataforma IIoT, on hi ha la descripció de les referències de l'esquema.

A baix a l'esquerra s'hi pot observar una representació del laboratori d'automatització on hi ha les cintes transportadores de safates mecanitzades a través dels motors trifàsics, els quals estan connectats a la xarxa elèctrica trifàsica a través del quadre elèctric, referenciat com a equip zero a la figura. De la línia trifàsica connectada al quadre elèctric, referenciada com a connexió zero és on és llegeixen les diferents dades a partir de l'analitzador de xarxa. En aquest punt, és on comença el flux de dades dirigit a l'equip 2, a través de la connexió u que funciona amb el protocol RS485 (Modbus RTU®). En aquest equip, es converteix la informació de protocol perquè pugui ser consultada a través d'ethernet TCP / IP, des de la màquina virtual ubicada al servidor remot. Fins aquest punt cal destacar que a través de la connexió dos i tres que utilitzen el mateix protocol la informació només és transfereix cap a un sentit, ja que de la passarel·la només és consulten dades no se n'hi envien. Totes les altres connexions restants de l'esquema que també son a través d'ethernet són en doble sentit, aquestes estan connectades a la WAN anomenada Internet a través de protocol ethernet TCP / IP.

A l'esquema es pot observar també l'estructura de software de l'equip tres, és a dir de la màquina virtual on s'allotjarà la plataforma IIoT creada en aquest projecte, on a partir de Docker es generen tres contenidors diferents on s'ubicaran els softwares dos, tres i quatre, Node-Red, InfluxDB i Grafana, respectivament, els quals pretenen solucionar l'objectiu de poder gestionar les diferents dades que arribin, tractar-les, poder-les emmagatzemar en una base de dades i alhora poder-les llegir, i finalment poder visualitzar aquesta informació tal com s'ha esmentat anteriorment a partir d'un panell de *dashboards*, gràfics i alertes, en una interfície fàcilment modificable.

Els equips quatre, cinc i sis s'han posat com a exemple per donar a entendre que al estar la plataforma IIoT connectada al núvol, des de un equip remot, sempre i quant la màquina virtual sigui encesa i funcionant, s'hi pot accedir des de qualsevol dispositiu amb accés a internet, simplement amb la direcció IP i el port TCP, en aquest cas tots tindran la mateixa IP, però cada software estarà ubicat en un port diferent, amb la possibilitat d'editar i consultar les dades tractades en gràfics via el port de Grafana, poder gestionar el flux de dades via Node-Red o simplement gestionar la base de dades a InfluxDB via comandes HTTP, totes aquestes gestions ja siguin des d'un PC, un portàtil, un dispositiu mòbil o qualsevol altre dispositiu amb connexió a internet.

Cal especificar que aquest esquema no contempla la connexió directa a la màquina virtual ja que el més adient es des d'un dispositiu amb protocol SSH instal·lat el qual s'explica més endavant, i des d'un dispositiu on sigui fàcil executar ordres al terminal, per tant des d'un dispositiu mòbil es podria connectar però seria molt més complex.

La figura del núvol a l'esquema representa una certa importància ja que la solució d'aquest projecte es focalitza en generar la plataforma adient per poder gestionar i traspasar les dades recollides físicament a planta a través d'internet. I tal i com s'observa a l'esquema tota la solució passar per Internet, ja que permet que qualsevol dispositiu amb accés en aquest núvol, pugui participar del desenvolupament de la plataforma o dels resultats d'aquesta.

4 Implementació

4.1 Introducció de la implementació

Aquest capítol tracta sobre com s'ha dut a terme la implementació de la solució dissenyada de l'estudi, dividit en diferents apartats, ordenats segons l'ordre d'implementació i seguint el transcurs de l'estudi, s'explica des de la creació de l'entorn virtual escollit, fins a la visualització de la informació útil i les dades tractades, explicant en detall tots els procediments intermedis seguits.

Els procediments que es plantegen per implementar la solució són el següents:

1. Configuració de l'entorn virtual (*cloud computing*) mitjançant una màquina virtual Linux de la plataforma Microsoft Azure.
2. Aprenentatge i recerca de la utilització d'entorns de softwares de gestió de fluxos d'informació i les seves bases de dades, juntament amb la transmissió de dades i com visualitzar-les al núvol.
3. Generació de l'arquitectura IIoT base sobre la que es desenvoluparan els aspectes relacionats amb les comunicacions industrials. Disseny i implementació de l'estructura de contenidors de software mitjançant Docker.
4. Posada en marxa de les aplicacions que realitzen les funcions específiques, com operacions de lectura i escriptura a bases de dades, o visualització de les dades a través del següent software: Node-Red, InfluxDB i Grafana.
5. Execució de proves i validacions a Node-Red, d'abocament de dades i emmagatzematge a la base de dades, juntament amb la creació de dashboards que permetin visualitzar les dades adquirides i les calculades, a través de gràfics a diferents panells editables.
6. Realització dels diferents càlculs i tractaments de dades a partir de blocs de programació, diferents algorismes i connexions en l'entorn de Node-Red, aplicats a l'estudi. Extreure les diferents magnituds per la consultoria energètica, alhora que el consum, cost i emissions del procés. Captació de pics de soroll.
7. Integració de l'anàlisi de dades processades i configuració per tal de poder veure per pantalla monitoritzada, el panell amb els diferents dashboards o valors indicatius.
8. Implementació de la plataforma IoT al laboratori d'automatització, amb l'aplicació industrial de gestió de safates, la qual farà el tractament i càlcul de dades dissenyat en aquest estudi, juntament amb un panell de visualització final on poder consultar les dades rellevants de l'estudi, aplicades al laboratori.

4.2 Configuració entorn virtual

Per a la posterior fabricació de l'arquitectura d'aplicacions per a gestionar la plataforma, es necessita un entorn virtual on poder executar-les. La tendència d'aquest entorn virtual on poder processar les dades en les indústries està derivant cap al anomenat *edge computing*⁵ ja que com analitzen grans companyies com per exemple IBM, la proximitat a les dades de la seva font pot oferir grans avantatges empresarials, com ara coneixements més ràpids, temps de resposta millorats o una millor disponibilitat d'ample de banda. Per tant a l'hora d'implementar un projecte com el tractat en aquest estudi, a la indústria actual, es decanta cap a configurar per exemple un entorn virtual en un PC industrial o en un dispositiu especialitzat (edge computer) a la mateixa planta.

Tanmateix en l'actual estudi, com bé s'ha especificat al disseny de la solució, s'inclina per configurar aquest entorn en una VM⁶, allotjada en un servidor de Microsoft Azure, on no hi ha necessitat d'obtenir l'equip físicament a la planta i té d'avantatge de poder-hi treballar remotament sempre que s'hi vulgui configurar o modificar alguna part del software. Hi ha empreses de la indústria que també es decanten per aquesta solució ja que té certs avantatges, a parts dels esmentats, com per exemple no tenir manteniment dels equips físics o no haver d'implementar una capa de ciberseguretat al teu propi *edge* ja que normalment les grans companyies on s'allotgen les VM ja en tenen.

Azure és una plataforma flexible que aporta diferents serveis integrats al núvol per a la gestió, control i emmagatzematge de les dades i preservació de continguts informàtics digitals, la qual ha estat desenvolupada i pertany a la corporació de Microsoft Corporation. Hi ha diferents opcions i competències a l'hora d'escollir on allotjar la teva plataforma, les avantatges més importants que conté l'escollida són:

- **Estalvi de costos:** l'execució de diversos entorns virtuals en una única infraestructura significa que es pot reduir dràsticament la superfície física de la infraestructura. Això augmenta els beneficis, ja que redueix la necessitat de mantenir tants servidors i els costos de manteniment i electricitat.
- **Agilitat i velocitat:** la posada en marxa d'una màquina virtual és relativament fàcil i ràpida, i és molt més senzilla per als desenvolupadors que l'aprovisionament d'un entorn nou complet. La virtualització fa que el procés d'execució d'escenaris de desenvolupament i proves sigui molt més ràpid.
- **Temps d'inactivitat reduït:** les màquines virtuals són molt portables i fàcils de migrar o clonar en un equip diferent, per la qual cosa són una solució excel·lent per a còpies de seguretat, en el cas que el host deixi de funcionar de forma inesperada.
- **Escalabilitat:** les màquines virtuals permeten escalar més fàcilment les aplicacions afegint més servidors virtuals o físics per distribuir la càrrega de treball entre diverses màquines virtuals. Com a resultat, podeu augmentar la disponibilitat i el rendiment de les aplicacions.
- **Avantatges de seguretat:** atès que les màquines virtuals s'executen en diversos sistemes operatius, l'ús d'un sistema operatiu convidat a una màquina virtual

⁵ **Edge Computing:** Marc informàtic distribuït que acosta les aplicacions empresarials a fonts de dades com ara dispositius IoT o servidors perifèrics locals.

⁶ **VM (Virtual machine):** Màquina virtual. Programari que crea una capa independent on s'emula el funcionament d'un ordinador real amb tots els components de maquinari que es necessita per funcionar (disc dur, memòria RAM, targetes de xarxa, targeta gràfica, etc.)

permet executar aplicacions d'una seguretat dubtosa i protegir el sistema operatiu host. Les màquines virtuals també permeten una millor anàlisi forense de la seguretat i solen utilitzar-se per estudiar virus informàtics de forma segura, aïllant-los per evitar riscos a l'equip host.

Així doncs, com a solució a implementar, l'entorn es configura en una màquina virtual amb sistema operatiu Linux, amb un Servidor Ubuntu i una direcció IP pública, el qual s'hi accedirà mitjançant connexió SSH⁷, un protocol de línia d'ordres que ens permet accedir-hi, molt semblant a programes de connexió remota d'escriptoris com podria ser el TeamViewer. La utilització d'aquests servidors lògicament té un cost per l'usuari, en funció de les característiques de la màquina virtual que vulguis utilitzar, es podria comparar a paraules generals com al lloguer d'un equip remotament. Depenent de la ubicació que estiguin aquests servidors i de les característiques que t'ofereixin tindrà un cost o un altre, i normalment va per tarifa d'hores utilitzades. A l'apartat de pressupostos s'aprofundeix amb xifres quin cost té la VM utilitzada en aquest projecte, no és un cost molt elevat, en comparació amb la compra d'un equip *d'edge* o PC industrial, lògicament en un projecte a la indústria s'ha de fer un balanç segons els avantatges i inconvenients i les hores que s'utilitzin els equips per si val la pena invertir amb un o l'altre.

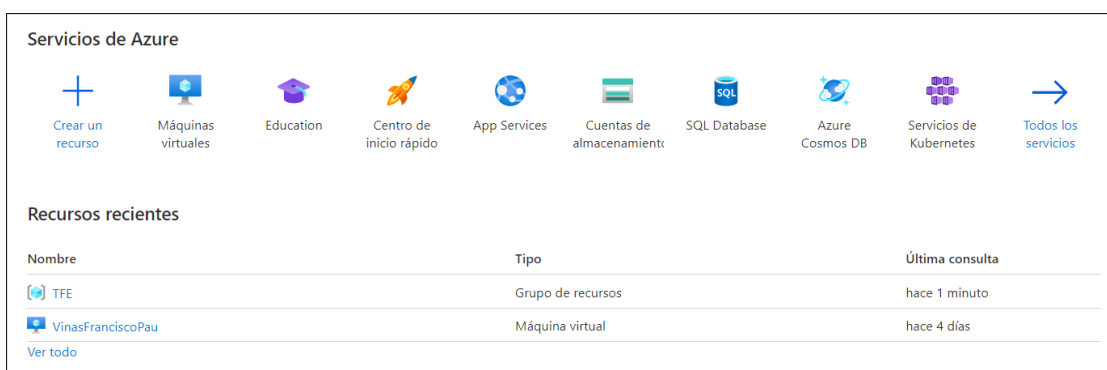


Figura 8. Imatge captura pantalla Azure.

Font: <https://portal.azure.com/#home>

Cal indicar que en el cas d'aquest projecte s'ha pogut adquirir una subscripció d'Azure per estudiants, amb la que es té accés a alguns Serveis de la plataforma, amb 100 dòlars de saldo que la plataforma assigna gratuïtament, per tant no ha tingut cap cost real per l'autor. Per a la creació de la màquina i l'obtenció de la subscripció s'ha creat un compte personal d'Azure assignat al email de la universitat. Posteriorment s'activa la subscripció per estudiants i un cop es rep el saldo ja es pot accedir al portal Azure de la web de Microsoft Azure, on tenim les diferents opcions de treball, com es pot observar a la figura set.

Un cop es tenen els serveis activats pots consultar el saldo que tens a la plataforma, i es pot procedir a crear la màquina virtual. En primer lloc, una màquina amb els requeriments necessaris per poder instal·lar amb facilitat els softwares necessaris per el projecte i els diferents paquets, és a dir una màquina més potent que quan només es posin en marxa els contenidors amb els diferents softwares instal·lats ja.

Les característiques de la primera màquina virtual configurada al projecte són les següents:

⁷ **SSH (Secure Shell)**: Protocol de xarxa que proporciona xifratge per operar els serveis de xarxa de manera segura en una xarxa no segura, proporciona també un inici de sessió de la línia d'ordres, executa ordres remotes i transfereix arxius de forma segura.

Nom	Característica
Mida VM	B2s (estàndard)
Família	ús general
vCPU	2
RAM	4 GiB
Discos de dades	4
E/S màxim per segon ⁸	1280
Emmagatzematge temporal	8 GiB
Cost / mes	31,99 €

Taula 2. Característiques primera màquina virtual configurada.

La direcció IP pública assignada a la VM és 40.89.152.162, la ubicació del servidor és a França central (segons la ubicació també varia el preu).

Des del portal d'Azure podem iniciar, parar o reiniciar la VM entre d'altres opcions com actualitzar-la o directament eliminar-la. Ens permet també poder connecta'ns-hi, amb diferents protocols, tal i com s'ha esmentat anteriorment, en el nostre cas, s'ha fet a partir de SSH, el qual s'havien creat dues claus prèviament, una pública i una privada la qual s'emmagatzema personalment. Definint la ruta de la clau privada a Azure, ens proporciona directament una ordre, amb la direcció IP pública, per executar al nostre terminal, la qual connecta directament amb la màquina virtual.

En aquest cas l'ordre al terminal per tal d'executar la connexió amb la VM és el següent:

```
PS C:\Users\pauvi> ssh-i C:\Users\pauvi\.ssh\TestClave_key.pem VinasPau@40.89.152.162
```

Tal i com s'observa a la part inferior de la figura vuit, s'indica l'usuari i el nom de la màquina virtual creada. Per tant en aquest punt ja es poden executar ordres sobre la VM a partir d'aquella línia. També s'indica alguna informació com l'última connexió o el tant per cent de memòria usada.

⁸ **E/S màxim per segon:** Les màquines virtuals d'Azure tenen límits de rendiment i d'operacions d'entrada/sortida (E/S) per segon en funció del tipus i mida de la màquina virtual.

```
PS C:\Users\pauvi> ssh -i C:\Users\pauvi\.ssh\TestClave_key.pem VinasPau@40.89.152.162
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-1062-azure x86_64)

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

System information as of Mon Nov 22 19:16:16 UTC 2021

System load:  1.12                Processes:           129
Usage of /:   11.2% of 28.90GB    Users logged in:    0
Memory usage: 6%                 IP address for eth0: 10.0.0.4
Swap usage:   0%

 * Super-optimized for small spaces - read how we shrank the memory
   footprint of MicroK8s to make it the smallest full K8s around.

https://ubuntu.com/blog/microk8s-memory-optimisation

17 updates can be applied immediately.
14 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Last login: Wed Nov 17 08:34:07 2021 from 139.47.125.73
VinasPau@VinasFranciscoPau:~$
```

Figura 9. Connexió SSH amb la VM

Font: Elaboració pròpia

4.3 Fabricació arquitectura plataforma IoT

4.3.1 Instal·lació de software en contenidors

Una vegada és té la màquina virtual operativa, s'instal·len els diferents recursos que es faran servir sobre ella:

- **Node-Red**, un gestor de comunicacions amb la planta i altres aplicacions al núvol, que permet a més a més executar programes.
- **Grafana**, una interfície gràfica de baixa complexitat que permet presentar dades en una web en forma de gràfics, mapes etc. És fàcilment editable per treballar amb influxDB.
- **InfluxDB**, un servidor de base de dades de codi obert desenvolupada per l'empresa InfluxData. Està escrit en el llenguatge de programació Go per emmagatzemar i recuperar dades de sèries temporals en camps com ara la supervisió d'operacions, mètriques d'aplicacions, dades de sensors d'IoT i anàlisi en temps real.

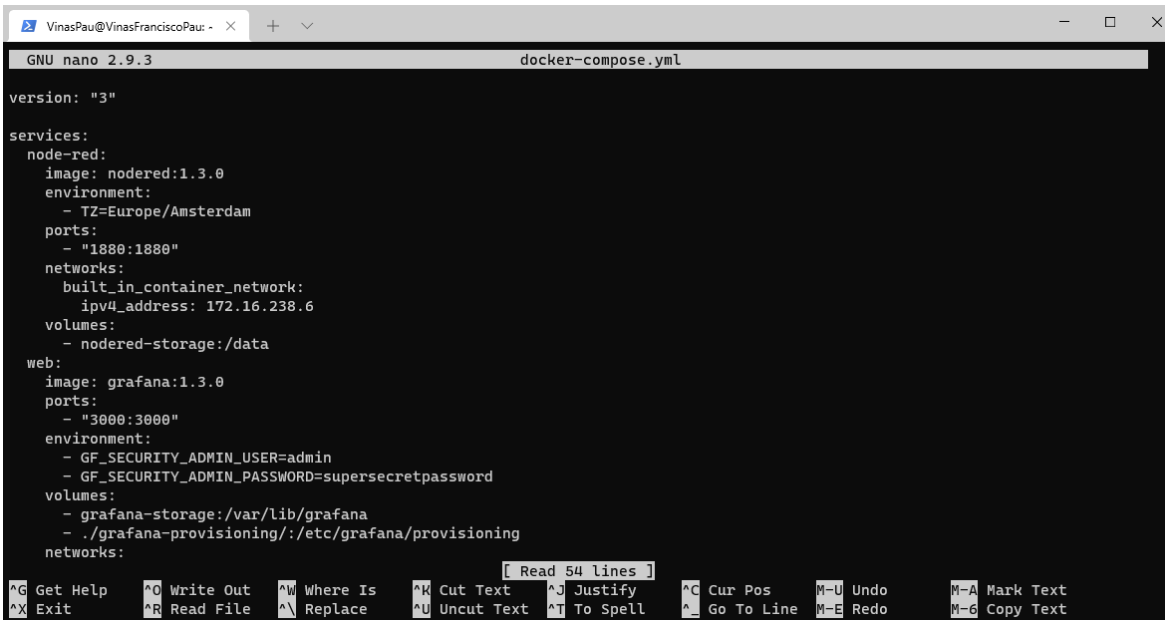
Per poder instal·lar aquets recursos correctament i funcionin adequadament, la solució tecnològica passa per implementar cada un d'aquests softwares en un contenidor aïllat, el qual ens permet executar cada aplicació independentment, alhora que ens permet generar entorns els quals es poden clonar fàcilment. Les característiques que fan que sigui interessant aquest tipus d'implementació són les següents:

- Els contenidors són més lleugers que les màquines virtuals ja que en comptes d'executar un sistema operatiu complet, el que fan és compartir els recursos del sistema operatiu propi, és a dir del *host* o amfitrió sobre el qual s'estan executant.
- No és necessita la instal·lació d'un sistema operatiu per contenidor, per tant augmenta l'eficiència del servidor i redueix els costos del servidor i de la llicència.
- L'ús dels recursos de la màquina són menors que una VM.

- Possibilitat de gran quantitat de contenidors per equip físic o VM.
- Tenen una òptima portabilitat⁹.

Tot i que existeixen una gran varietat de solucions per implementar contenidors, com per exemple: *Virtuozzo*, *DC/OS*, *OpenVZ*, *LXC/LXD*, *RedHat*, etc.

En aquest estudi s'ha optat per la solució oferta per **Docker**, un projecte de codi obert que permet la implementació dels contenidors, i que és dels més populars i utilitzats. A partir de les instruccions de la seva pàgina web on s'indica les diferents ordres a executar (per més detall consultar als annexes) per un sistema Ubuntu, es descarrega i es realitza la instal·lació del software. Seguidament s'implementa el *Docker Compose*, una eina del programari que permet definir i executar diferents contenidors al mateix temps i que és essencial per un projecte com l'actual, ja que per exemple es pot estar monitoritzant dades alhora que s'està captant i gravant una dada a la seva base corresponent. Permet també a partir d'una sola ordre, crear i posar en marxa els diferents contenidors. Per definir els serveis s'utilitza un fitxer YAML¹⁰.



```
GNU nano 2.9.3 docker-compose.yml
version: "3"

services:
  node-red:
    image: nodered:1.3.0
    environment:
      - TZ=Europe/Amsterdam
    ports:
      - "1880:1880"
    networks:
      built_in_container_network:
        ipv4_address: 172.16.238.6
    volumes:
      - nodered-storage:/data
  web:
    image: grafana:1.3.0
    ports:
      - "3000:3000"
    environment:
      - GF_SECURITY_ADMIN_USER=admin
      - GF_SECURITY_ADMIN_PASSWORD=supersecretpassword
    volumes:
      - grafana-storage:/var/lib/grafana
      - ./grafana-provisioning:/etc/grafana/provisioning
    networks:
```

Figura 10. Captura de pantalla edició fitxer .yml del docker-compose.

Font: Elaboració pròpia

Un cop s'ha instal·lat i s'han fet les validacions necessàries (es recullen totes a l'apartat de validacions d'aquest projecte, on es poden consultar), es pot aixecar les diferents aplicacions que requereix el projecte, en diferents contenidors, amb els seu corresponent programari. A partir del protocol SCP¹¹, que està basat en SSH, per tant l'ordre també anirà amb la clau privada, es transfereixen els fitxers necessaris per la instal·lació de cada contenidor, ja que es requereixen certs paquets per cada un d'ells, s'aconsegueix així crear

⁹ **Portabilitat:** En informàtica, és la capacitat que té un software (programa o aplicació) d'executar-se a diferents plataformes, entorns o sistemes, de forma efectiva i eficient.

¹⁰ **YAML:** És un llenguatge de programació de serialització de dades, se sol utilitzar en el disseny d'arxius de configuració o en aplicacions on s'emmagatzemen o es transmeten dades.

¹¹ **SCP (Security Copy Protocol):** És un mètode basat en SSH amb la finalitat de transferir de manera segura fitxers informàtics entre un host local i un host remot, o entre dos host remots. El protocol intenta ser una versió més segura del RCP (Remote Copy Protocol), el qual serveix per transferir fitxers a través de la xarxa.

una imatge *custom* de cada contenidor, adequada a les seves necessitats. Indicant la direcció d'origen i la de destí a l'ordre, es transfereixen fàcilment els documents, els quals es verifiquen s'hi han arribat correctament verificant el directori on s'han enviat.

Un cop realitzats aquests passos, es procedeix a crear les imatges corresponents a partir dels arxius transferits, els quals es poden veure en detall en els annexes d'aquest projecte, a la figura nou es pot observar com seria mes o menys l'esquema del fitxer YAML, el qual s'edita des de la terminal de la VM mateix, canviant-li el paràmetre anomenat tag de la imatge creada corresponent abans d'executar-lo. Així doncs es realitza aquest procés tant per la imatge de Node-Red com la de Grafana.

4.3.2 Posada en marxa contenidors

Després de la preparació de l'arquitectura de la solució, finalment ve la posada en marxa dels contenidors, l'ordre d'aixecament agafa la informació recollida per generar els contenidors adients, en aquest cas, segons es pot observar a l'estructura del docker-compose.yml aixeca tres contenidors en total, un per cada software.

Cal mencionar que aquesta acció, es dur a terme cada cop que es vol utilitzar la plataforma d'aquest projecte, és a dir cada cop que es vol connectar amb la VM, ja que encara que es deixin els contenidors muntats, s'eliminen quan s'apaga la màquina virtual, la qual des de les opcions d'Azure està programada per apagar-se automàticament cada dia a les 00:00h per tal de que no quedi encesa i generi un cost elevat o no esperat pel projecte.

```

VinasPau@VinasFranciscoPau:~/TFE/Base folder$ sudo docker ps
CONTAINER ID   IMAGE                                COMMAND                                CREATED
727c0a812080   grafana:1.3.0                       "/run.sh"                              5 months ago
6e427d290b23   influxdb:1.8.4-alpine               "/entrypoint.sh infl..."             5 months ago
6401a3626a4f   nodered:1.3.0                       "npm --no-update-not..."            5 months ago

STATUS        PORTS                                NAMES
Up 3 minutes  0.0.0.0:3000->3000/tcp, :::3000->3000/tcp  basefolder_web_1
Up 3 minutes  0.0.0.0:8086->8086/tcp, :::8086->8086/tcp  basefolder_database_1
Up 3 minutes  0.0.0.0:1880->1880/tcp, :::1880->1880/tcp  basefolder_node-red_1
  
```

Figura 11. Aixecament dels tres contenidors.

Font: Elaboració pròpia

Una vegada ja està tota l'arquitectura de la plataforma IoT creada, juntament amb els contenidors aixecats i llestos per poder funcionar correctament, hi ha la possibilitat de modificar les prestacions de la VM degut a que ja s'ha realitzat la instal·lació del software necessari, tasca que requeria d'una potència i característiques de la màquina més elevades. S'opta per disminuir les característiques, realitzant un canvi de mida de la màquina virtual, conseqüentment disminuint així els costos del projecte, i optimitzant també les necessitats actuals amb el potencial de la màquina, fent-la més eficient.

Les característiques de la segona màquina virtual configurada al projecte són les següents:

Nom	Característica
Mida VM	B1s
Família	ús general
vCPU	1
RAM	1 GiB

Discos de dades	2
E/S màxim per segon	320
Emmagatzematge temporal	4 GiB
Cost / mes	7,99 €

Taula 3. Característiques segona màquina virtual configurada.

4.3.3 Node-Red

“Node-Red és una eina de programació basada en fluxos de dades, desenvolupada originalment per l'equip de serveis de tecnologia emergent d'IBM i que ara forma part de la Fundació OpenJS.” (OpenJS Foundation, 2019)

És l'aplicació de codi obert que s'utilitza per aquesta part de desenvolupament on és configuren i programen els flux de dades. Node-Red és una eina de programació visual, que s'implementa en dispositius controladors de hardware, per tal d'organitzar fluxos d'informació. Connecta varis dispositius al mateix temps, tant de hardware, normalment enfocats a internet de les coses, tant com serveis d'internet. És adequat per treballar en entorns dedicats al processos industrials o com a solució per equips de planta que necessitin una comunicació adient. Facilita el nivell de programació requerit, ja que es molt més visual, i té certes opcions, procediments o blocs de programació predeterminades o configurades.

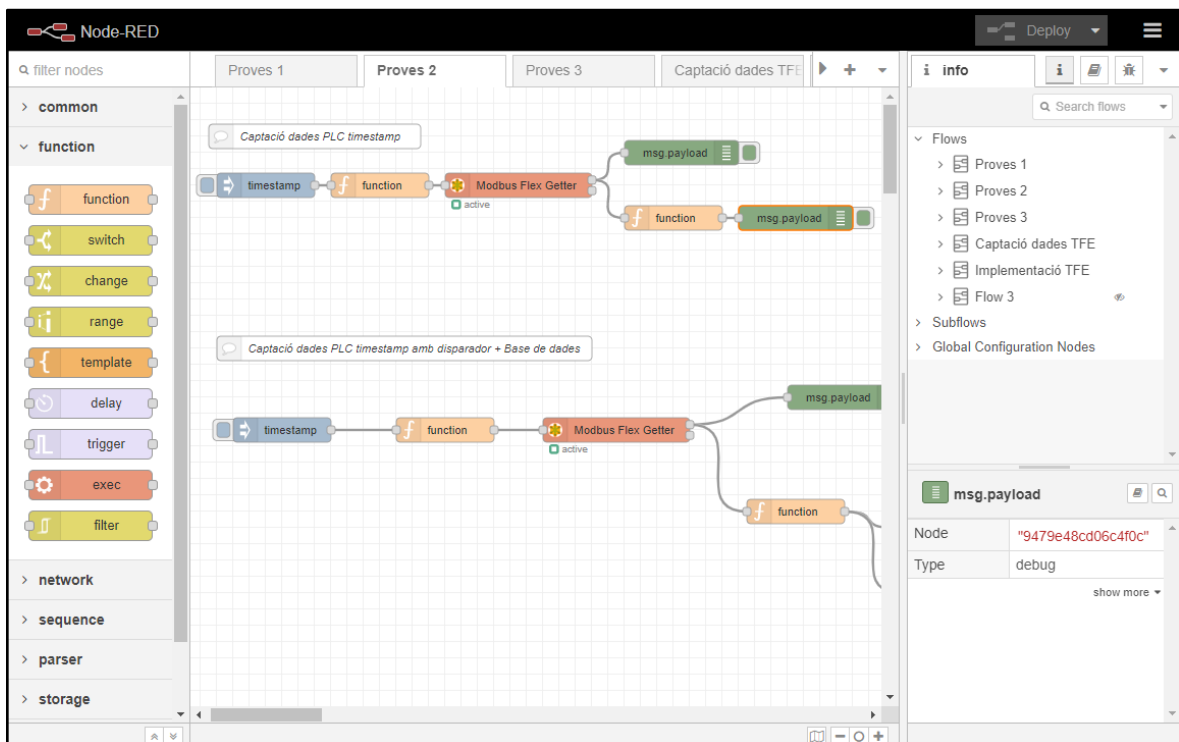


Figura 12. Interfície de Node-Red.

Font: Elaboració pròpia

La programació d'aquest tipus, basada en fluxos és una forma de descriure el comportament de l'aplicació com una xarxa de caixes negres, o nodes, tal i com els hi denominen des de Node-Red. Tots els nodes tenen un objectiu ben definit, ja que han de

rebre o capturar certes dades, realitzar-ne algun tractament, per després enviar a un o certs nodes els quals està enllaçat.

Node-Red s'executa a un dels tres contenidors aixecats a la plataforma, per tant a la màquina virtual ubicada físicament al servidor remot. La utilització d'aquest software per tal d'automatitzar la gestió dels fluxos de dades és un dels requeriments del projecte.

Mitjançant la direcció IP pública de la VM (40.89.152.162) i el port assignat al contenidor de Node-Red (1880) s'accedeix al software. D'aquesta manera es pot accedir al programa que gestiona els fluxos de dades i permet l'edició i la programació d'aquests, de forma remota. Es possible canviar el port del contenidor des de la configuració de la VM.

Així doncs, per accedir mitjançant el navegador a Node-Red s'utilitza la següent adreça:

<http://40.89.152.162:1880/>

Com s'observa a la figura dotze, la interfície consta principalment de tres àrees diferents, la primera, un menú pels diferents tipus de nodes els quals es vol treballar, la segona, la zona de treball on hi ha les diferents pestanyes per configurar les diferents connexions entre nodes i esquemes de flux d'informació, i finalment la tercera, on hi ha l'àrea d'informació, d'ajuda, de depuració dels missatges, de configuració dels nodes, de dashboards i context de nodes. Per altra banda a dalt a la dreta hi ha el botó per desplegar els canvis realitzats i un menú general de l'aplicació.

Arrossegant els diferents blocs disponibles del lateral esquerra, cap a l'àrea de treball, es generen els nous fluxos, amb la possibilitat de configurar els paràmetres i els atributs de cada node. Els nodes es connecten entre si mitjançant cables de connexió els quals uneixen les diferents entrades i sortides dels nodes. Cada bloc es pot editar i especificar una etiqueta amb l'objectiu de fer més entenedor l'esquema i poder entendre ràpidament la funció o rol que té dins del procés de gestió o tractament d'informació.

Els diferents blocs de nodes es divideixen amb els següents apartats:

- **Comú:** S'utilitzen més freqüentment i serveixen per fer les accions principals, on hi podem trobar per exemple el node d'injectar missatges en el flux, el de depurar missatges, el de completar, el de captació d'errors, o el de simplement comentar, entre d'altres.
- **Funció:** Relacionats en tractar i manipular la informació provinent de l'anterior node, ja sigui amb el bloc funció, el qual et permet executar codi JavaScript per programar el missatge, o les variants ja siguin per canviar, filtrar, executar, enrederir un senyal o missatge.
- **Xarxa:** Hi trobem els nodes per poder enviar outputs o rebre inputs de dades dels diferents protocols de xarxa, com per exemple MQTT, HTTP, WebSockets, TCP, UDP.
- **Seqüència:** Permeten ajuntar, dividir, barrejar o seqüenciar els diferents missatges rebuts.
- **Analitzador sintàctic:** Faciliten analitzar i convertir missatges en diferent formats com per exemple, csv, html, json, xml, yaml.
- **Emmagatzematge:** Permeten escriure, llegir i visualitzar els missatges per tal de emmagatzemar-los. També en bases de dades com InfluxDB en aquest cas havent descarregat l'extensió i paquet de contribució pertinent.

- **Modbus:** Possibiliten intercanviar missatges a partir de sèrie Modbus ja sigui amb TCP, C701, buffer RTU, etc.
- **Dashboard:** Permeten generar interfície visual per variar o generar missatges, o visualitzar informació. Ja sigui amb botons, sliders, inputs de texts, gràfics, notificacions, etc.

Els nodes utilitzats en aquest projecte s'expliquen detalladament amb els paràmetres configurats en cada cas, en l'apartat de tractament de dades. Així com l'esquema de gestió de la informació i el flux de dades, amb les connexions pertinents.

4.3.4 InfluxDB

El software que s'utilitza per la gestió de les bases de dades en aquest projecte és InfluxDB, un programari de codi obert desenvolupat per InfluxData.

Al registrar dades d'un procés industrial, es generen en molt poc temps, quantitats d'informació molt extenses, les quals han de processar-se juntament amb alguna referència tal com la marca temporal del moment del mesurament, o com l'orde de petició, etc. Aquestes dades, necessiten ordenar-se de forma coherent i classificada. InfluxDB ens permet realitzar aquesta tasca, ja que permet generar diferents bases de dades. Possibilita crear també varies series de dades o diferents mesures dins de la mateixa base de dades, sent independents cadascuna, series anomenades mesuraments.

InfluxDB és una plataforma de sèries temporals, és a dir que cada dada que s'emmagatzema se li assigna una timestamp¹² o marca temporal. L'hora que se li assigna a la dada és normalment la que entra a la base de dades tot i que es pot configurar per definir la hora que es vulgui emmagatzemar la dada.

Així doncs la funcionalitat d'aquest software es limita a rebre, emmagatzemar, mostrar y enviar dades si es necessari.

InfluxDB s'executa a un dels tres contenidors aixecats a la plataforma, per tant a la màquina virtual ubicada físicament al servidor remot. La utilització d'aquest software per l'enregistrament de dades és un dels requeriments d'aquest projecte.

Mitjançant la direcció IP pública de la VM (40.89.152.162) i el port assignat al contenidor de InfluxDB (8086) s'accedeix al software. D'aquesta manera es pot accedir al programa que emmagatzema les diferents dades, de forma remota. Es possible canviar el port del contenidor des de la configuració de la VM.

Per accedir mitjançant el navegador a InfluxDB però, és necessitar realitzar a través del protocol HTTP algun tipus de petició com per exemple un GET al "endpoint" query per poder visualitzar en format json la base de dades. Per poder visualitzar les diferents bases de dades creades per exemple, s'utilitza la següent adreça:

<http://40.89.152.162:8086/query?q=show%20databases&pretty=true>

En aquest estudi però s'ha accedit a InfluxDB via la màquina virtual directament és a dir pel terminal dedicat del contenidor que executa el software InfluxDB, tant per la seva comoditat, com perquè s'ha realitzat l'aprenentatge de les diferents ordres des d'allà, i no per ordres amb protocol HTTP.

¹² **Timestamp:** Marca temporal o segellat de temps, seqüència de caràcters que denoten la hora i la data en el que ha ocorregut un determinat esdeveniment, com per exemple la captació de certa dada.

Així doncs s'ha entrat a la interfície de línies d'ordres de InfluxDB, és a dir a la terminal del contenidor, ha partir d'una ordre executada en la VM.

```
$ sudo docker exec -it basefolder_database_1 bash
```

On, "basefolder_database_1" és el nom del contenidor el qual es vol executar la seva terminal. Un cop hi ets, introduint la paraula "influx" accedeixes a la interfície de línies d'ordres de InfluxDB, tal i com es pot veure a la figura tretze.

```
VinasPau@VinasFranciscoPau:~/TFE/Base folder$ sudo docker exec -it basefolder_database_1 bash
sudo: unable to resolve host VinasFranciscoPau
bash-5.0# influx
Connected to http://localhost:8086 version 1.8.4
InfluxDB shell version: 1.8.4
> use TFE_database
Using database TFE_database
> show measurements
name: measurements
name
----
Consum_x_hora
Dades_AdX
Prova_dades_PLC
Serie_1
Serie_2
Sinus
```

Figura 13. Interfície de línies d'ordre d'InfluxDB.

Font: Elaboració pròpia

Per realitzar les diferents accions possibles de la interfície s'han de executar les ordres específiques per cada tasca. A la figura tretze es visualitzen dos ordres, la primera per usar la base de dades denominada "TFE_database" i la segona per visualitzar els mesuraments que hi ha dins d'aquesta base de dades, els quals a l'apartat de tractament de dades i validacions s'expliquen per que han servit cadascun d'ells. Cal destacar que la base de dades "TFE_database" s'ha creat també partir de l'ordre específica "Create Database", la qual és d'interès, com també el "Drop Database" el qual permet eliminar la base de dades. La visualització de la informació també es realitza des d'aquesta interfície a través d'ordres de selecció on podem seleccionar les dades a visualitzar segons la mesura, la columna a la qual pertanyin i filtrar-les segons timestamp o directament el tipus de dada. A la figura catorze es mostra un exemple, que permet entendre la lògica del llenguatge emprat i alguns casos pràctics executats.

En l'exemple mostrat, es pot observar com a partir d'ordres amb un llenguatge anomenat InfluxQL, molt semblant als que s'utilitza en bases de dades com per exemple l'SQLite, s'ordena quina columna es vol visualitzar d'un mesurament concret. Posant l'asterisc es seleccionen totes les columnes per visualitzar. Així doncs s'observa en la primera línia d'ordre que es seleccionen totes les columnes gravades de la mesura "Consum_x_hora", mentre que després que la interfície ensenyi les dades requerides, a la segona ordre es fa una petició del mateix mesurament però seleccionant només la columna de "consum".

Les dades es poden filtrar també a partir de condicions imposades a les ordres de selecció, tal i com es veu a la tercera ordre de la figura catorze, on s'imposa la condició de seleccionar només les dades les quals tinguin un cost superior a "0.15" o a la cinquena ordre on s'imposa seleccionar les dades de la columna "randomVal" que siguin superior a quatre. Aquest llenguatge tant comú i estàndard en bases de dades permet moltes possibilitats de filtratge i ordenació de dades per tal de poder classificar les que es volen consultar o llegir de forma òptima.

```

> select * from Consum_x_hora
name: Consum_x_hora
time          consum          cost          emissions
-----
1650541852529256007 0.8967448979591827 0.23161127224489766 0.22418622448979567
1650542789197956427 0.8967448979591827 0.1497653654081631 0.22418622448979567
> select consum from Consum_x_hora
name: Consum_x_hora
time          consum
-----
1650541852529256007 0.8967448979591827
1650542789197956427 0.8967448979591827
> select * from Consum_x_hora where cost>0.15
name: Consum_x_hora
time          consum          cost          emissions
-----
1650541852529256007 0.8967448979591827 0.23161127224489766 0.22418622448979567
> select * from Serie_1
name: Serie_1
time          numVal  randomVal          strVal
-----
1636650786104689745 10      4.310478935890341  word
1637762926105890041 10      3.54245579109304  word
1637762936524686939 10      8.27788588375231  word
1640271812406934080 10      0.8256326072816011 word
> select randomVal from Serie_1 where randomVal>4
name: Serie_1
time          randomVal
-----
1636650786104689745 4.310478935890341
1637762936524686939 8.27788588375231

```

Figura 14. Ordres de filtració i visualització de InfluxDB.

Font: Elaboració pròpia

4.3.5 Grafana

Per la visualització gràfica de les dades emmagatzemades, s'utilitza el software Grafana, un software dissenyat per la configuració de panells de gràfics o dashboards, per tal de controlar i visualitzar d'un cop d'ull o fàcilment els indicadors clau de rendiment rellevants d'un procés o consultar l'estat en que es troben. Està orientat a dades amb marca temporal.

Grafana és un programari de codi obert, creat al gener de 2014, dissenyat prèviament per Torkel Ödegaard, un desenvolupador suec el qual continua desenvolupant programari lliure. Està escrit en llenguatge de programació Go i Node.js juntament amb un conjunt ampli d'interfície de programació d'aplicacions (API).

Es caracteritza per la rapidesa, flexibilitat i facilitat per crear, guardar i compartir els diferents panells configurats per diferents usuaris. On els noms de grup, els hosts o elements poden canviar ràpidament canviant una variable dins de la plantilla. Aquesta plataforma també facilita la definició de condicions per alertes de forma visual per mesures que sigui necessàries. Contínuament el software supervisa les condicions i envia notifikacions de diferents formes en cas que s'activi una d'aquestes. Proporciona la connexió fàcilment de diferents bases de dades, com per exemple InfluxDB.

Grafana s'executa a un dels tres contenidors aixecats a la plataforma, per tant a la màquina virtual ubicada físicament al servidor remot. La utilització d'aquest software per la visualització de les dades i monitorització d'aquestes remotament, és un dels requeriments d'aquest projecte.

Mitjançant la direcció IP pública de la VM (40.89.152.162) i el port assignat al contenidor de Grafana(3000) s'accedeix al software. D'aquesta manera es pot accedir al programa que permet visualitzar els diferents dashboards, de forma remota. És possible canviar el port del contenidor des de la configuració de la VM.

Així doncs, per accedir mitjançant el navegador a Grafana s'utilitza la següent adreça:

<http://40.89.152.162:3000/>

Al entrar a la interfície, demana unes credencials les quals estan indicades al arxiu .yml del docker compose.



Figura 15. Interfície de Grafana.

Font: Elaboració pròpia

Una vegada s'accedeix a la interfície, el primer que s'ha de configurar és la base de dades de la qual es vol que Grafana faci la lectura de dades. Com es visualitza a l'esquerra de la figura quinze, des de configuració, fonts de dades, es selecciona una base de tipus InfluxDB, posteriorment se li defineix un nom a la font i s'indica la direcció URL del contenidor. En el cas d'aquest estudi serà una base de dades associada a InfluxDB, on s'indicarà a Grafana que ha de consultar les dades a dibuixar de la base de dades que s'ha generat. Com que la base de dades que s'ha de consultar es troba al mateix ordinador és a dir a la VM, s'utilitza el localhost per indicar l'adreça on consultar juntament amb el port associat a InfluxDB. És important, definir la base de dades que es vol consultar en aquest cas la creada anomenada "TFE_database".

A la figura quinze s'observa l'estructura de la resta d'interfície on al centre hi ha un gràfic d'exemple que indica la potència activa amb les diferents referències en kilowatts i al llarg de les diferents marques temporals, a dalt es pot escollir en quins moments es vol observar les dades. Gràfics com aquests se'n poden configurar varis fins i tot contrarestar diferents dades en el mateix gràfic. En l'apartat de visualització útil i resultats s'explica un dels panells definitius realitzats en aquest estudi.

Grafana permet configurar varis panells i guardar-los per poder consultar de diferents disposicions o preferències segons pertorqui de la mateixa base de dades.

4.4 Creació y abocament de dades a la plataforma

Un cop es té la plataforma operativa, s'ha de configurar l'abocament de dades adequat, és per això que és fan un seguit de validacions a partir de la creació de dades artificials abans de captar-ne de reals. A l'apartat de validacions s'expliquen en detall les més rellevants, dutes a terme en aquest projecte.

Consisteixen bàsicament en crear senyals a partir de l'entorn Node-Red i poder-los depurar per veure si arriben correctament al destí, juntament en la creació de la base de dades a InfluxDB i corroborar que s'escriuen i s'emmagatzemen les dades correctament.

En aquesta etapa es realitza un aprenentatge de la plataforma i el seu funcionament.

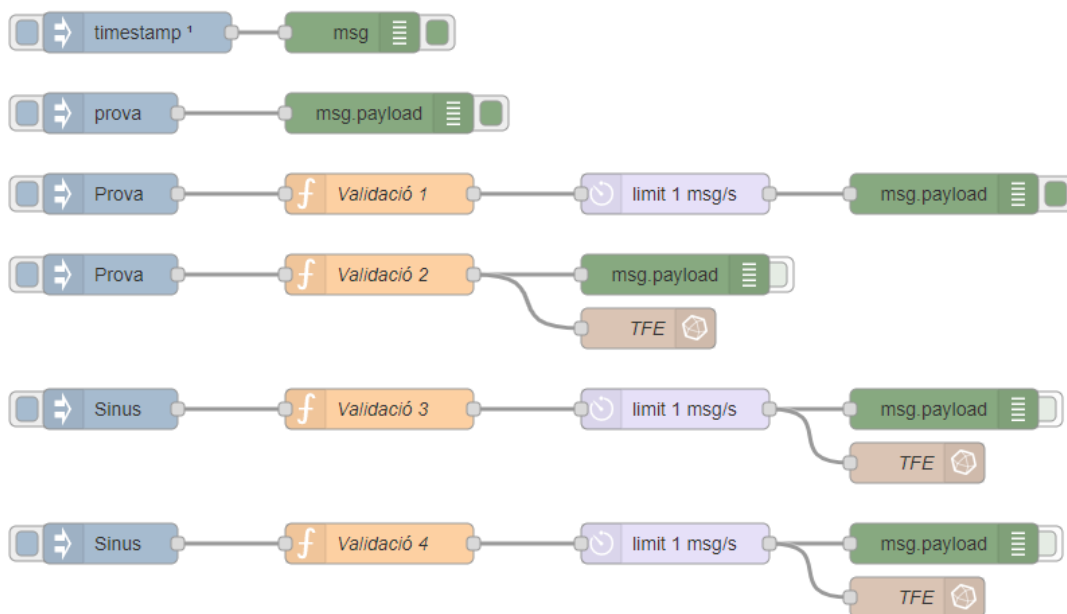


Figura 16. Esquema proves de creació de missatges.

Font: Elaboració pròpia

A la figura setze s'observa un dels primers esquemes de proves realitzats en aquest apartat. A partir dels blocs blaus s'envien senyals d'inicialització on es pot configurar un primer missatge, com a un string, un número, un booleà, un timestamp, entre d'altres. Aquesta informació es guarda en la carrega o payload del missatge, i per altre banda se'l pot etiquetar amb un tòpic. Aquest node d'injecció de missatges es pot configurar que injecti un sol cop o ho faci després d'un cert temps o que repeteixi la injecció periòdicament cada cert temps. Els blocs verds, anomenats de depuració, permeten depurar els fluxos programats, ja que reben el missatge que se'ls hi envia i l'emeten per la pantalla de depuració o *debugging*. A la pantalla de depuració s'indica la data i la hora del missatge rebut, l'identificador del node el qual li arriba el missatge, quin tipus de missatge és ja sigui un objecte, un vector, o un string, i finalment el missatge rebut, on normalment és un objecte amb diferents característiques com per exemple un identificador de missatge, un payload o carrega on hi ha el missatge que es vol traslladar, i un tòpic per tenir una referència del que s'està enviant. Els dos primers exemples de la figura s'envia simplement un timestamp per rebre'l i verificar-lo en el *debugging* en forma d'objecte el primer i en els segon s'envia una paraula de prova en format string per rebre solament la carrega és a dir la paraula en el *debug*.

Per la creació de dades és essencial introduir a part del node d'injecció, el node de funcions, els qual permet executar un tros de codi en JavaScript a la que rep el missatge. Alhora permet treballar sobre el missatge rebut. Al cinquè esquema de l'exemple de la figura setze, en la validació 3 s'implementa un tros de codi que genera una senyal sinus a partir d'un bucle el qual va enviant missatges cap a un node temporitzador, que fa que arribi al destí, un missatge cada un segon.

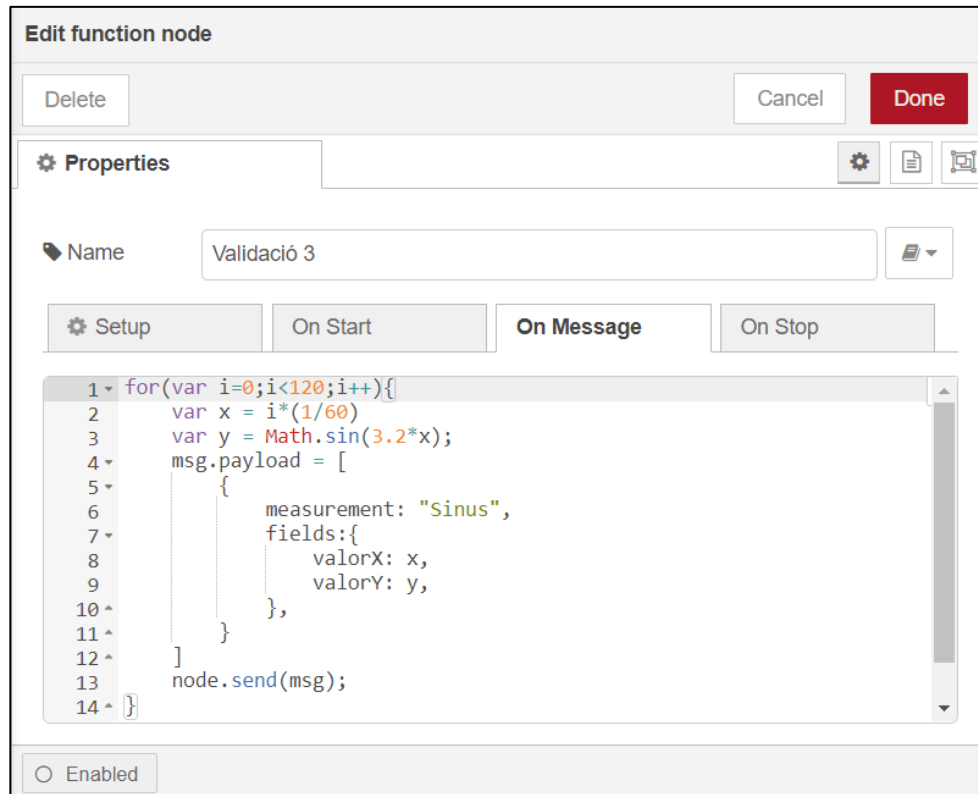


Figura 17. Codi JavaScript node funció.

Font: Elaboració pròpia

A la figura 17 es veu el codi implementat en la validació 3, i permet visualitzar la interfície d'edició d'un bloc o node funció. El codi simplement genera un sinus de longitud d'ona igual a 2 i amplitud 1. Per fer-ho recórrer 120 cops un bucle *for*, calculant dos variables *x* i *y*, i que envia en format d'objecte cap a la base de dades del TFE. Per tal de poder visualitzar després la senyal per Grafana. Aquesta validació s'explica en detall a l'apartat de validacions.

Finalment i no menys important en aquest apartat, és el node Influx batch el qual ens permet enviar els senyals creats a la base de dades corresponent. És a dir que funciona com a input de la base de dades, pels missatges que rep. En la figura divuit es pot veure la configuració del server on hi ha la base de dades on bàsicament s'hi posa el host amb la IP pública de la VM i el port del contenidor on està situat InfluxDB, per acabar cal indicar quina base de dades es vol utilitzar.

Es pot observar tant en la figura disset com en la divuit a baix a l'esquerra que els nodes tenen l'opció d'activar-se o desactivar-se en funció del que es vulgui realitzar en el flux de dades.

Edit influx batch node > Edit influxdb node

Delete Cancel Update

⚙️ Properties ⚙️ 📄

📌 Name Name

📌 Version 1.x

📌 Host 40.89.152.162 Port 8086

📌 Database TFE_database

📌 Username

📌 Password

Enable secure (SSL/TLS) connection

Enabled ⓘ 9 nodes use this config On all flows

Figura 18. Configuració node Influx batch.

Font: Elaboració pròpia

4.5 Tractament de dades

Una vegada s'han creat dades artificials i abocat a la plataforma, entra l'etapa de l'estudi per tractar les dades que es captaran a la plataforma i poder-ne observar la informació necessària i requerida correctament, així com la informació útil i rellevant que se'n pugui extreure.

A la plataforma IIoT dissenyada, el tractament de dades realitzat en aquest estudi es divideix en tres parts:

- Tractament de les dades captades per l'analitzador per extreure els valors adients.
- Càlcul del consum, cost i emissions per hora en temps real.
- Programació d'alertes per pics de distorsió harmònica.

Els dos primers tractaments de dades estan realitzats directament des de l'entorn de Node-Red a partir de blocs de programació en JavaScript, mentre l'últim el qual és més senzill es realitza directament a l'entorn de Grafana.

El tractament de les dades captades per l'analitzador és basa en un esquema de nodes el qual gestiona la informació rebuda per l'analitzador combina els dos vectors rebuts per a partir d'aquí calcular i reescriure les dades amb les unitats i els valors correctament.

Les dades es consulten de l'analitzador via protocol ModbusTCP / IP, a partir dels nodes vermells Modbus-Flex-Getter que permeten la lectura connectant-se a un servidor Modbus TCP, a la que els hi entra un cert missatge a partir d'un node de funció, amb les dades de configuració de la consulta que es vol fer.

A partir de dos vectors de 40 registres cadascun, es sol·liciten les dades de l'analitzador de xarxa, les quals venen en un format de 16 bytes de grandària. Aquests vectors tal i com s'observa a la figura dinou es combinen amb el node d'unió groc, el qual uneix seqüències de missatges i esta configurat de manera manual per generar un vector resultant dels dos inputs. Està programat de tal manera que envia un missatge després de rebre dos.

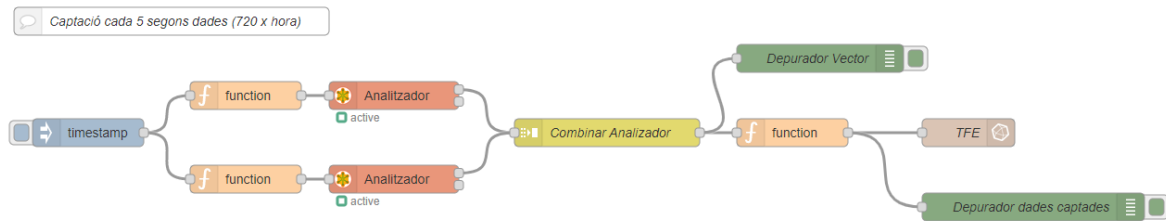


Figura 19. Tractament dades 1.

Font: Elaboració pròpia

Els missatges que reben els nodes Modbus són els que es mostren al codi de programació de la figura vint. Serveixen per configurar el requeriment del GET on el paràmetre FC indica quin tipus de lectura es vol demanar, en aquest cas la tipo tres, que són registres existents, en aquest cas les diferents mesures de l'analitzador. Per altre banda també és defineixen l'adreça la qual es vol començar a llegir, en el primer cas la zero, en el segon a partir de la quaranta, ja que es llegeix la segona part de les dades, i també la quantitat desitjada, en aquest cas es fa una petició de quaranta registres a llegir a partir de l'adreça indicada.

```

1 msg.payload = {
2   value: msg.payload,
3   'fc': 3,
4   'unitid': 1,
5   'address': 0,
6   'quantity': 40
7 }
8 return msg
9

```

Figura 20. Codi de requeriment node Modbus-Flex-Getter.

Font: Elaboració pròpia

La lectura del diferents paràmetres ocupa dos registres a causa de la mida d'alguns i la seva resolució. Seguidament es mostra una taula de gran utilitat pel disseny de la funció que tracta i reescriu les dades per guardar-les a la base de dades correctament. La taula consta dels diferents paràmetres que registra l'analitzador de xarxa, relacionats amb cinc dades per variable, el símbol de cada paràmetre, la seva posició inicial de memòria on es consulten traduïda a sistema decimal (Dades extretes del manual estès de l'analitzador de xarxa, on estaven amb sistema hexadecimal), la posició que agafaran dins del vector de lectura de la plataforma, i les unitats les quals estan les dades consultades i a les que es converteix després de la funció de tractament de dades de la plataforma, i per tant es guarden al registre de la base de dades.

Mesures analitzador de xarxa					
Símbol	Magnitud	Posició memòria analitzador	Posició vector lectura	Unitats	
				Analitzador	Plataforma IloT
V1	Tensió fase 1	0	0	[dV]	[V]
A1	Corrent línia 1	2	2	[mA]	[A]
W1	Potència activa 1	4	4	[W]	[kW]
Q1	Potència reactiva 1	6	6	[VAr]	[kVAr]
FdP1	Factor de potència 1	8	8	-	-
V2	Tensió fase 2	10	10	[dV]	[V]
A2	Corrent línia 2	12	12	[mA]	[A]
W2	Potència activa 2	14	14	[W]	[kW]
Q2	Potència reactiva 2	16	16	[VAr]	[kVAr]
FdP2	Factor de potència 2	18	18	-	-
V3	Tensió fase 3	20	20	[dV]	[V]
A3	Corrent línia 3	22	22	[mA]	[A]
W3	Potència activa 3	24	24	[W]	[kW]
Q3	Potència reactiva 3	26	26	[VAr]	[kVAr]
FdP3	Factor de potència 3	28	28	-	-
WIII	Potència activa III	30	30	[W]	[kW]
VArLIII	Potència inductiva III	32	32	[VAr]	[kVAr]
VArCIII	Potència Capacitativa III	34	34	[VAr]	[kVAr]
cosPhiIII	Cos Phi III	36	36	-	-
FdPIII	Factor de Potència III	38	38	-	-
THDv1	% THD V L1	48	40	[%]	[%]
THDv2	% THD V L2	50	42	[%]	[%]
THDv3	% THD V L3	52	44	[%]	[%]
THDi1	% THD I L1	54	46	[%]	[%]
THDi2	% THD I L2	56	48	[%]	[%]
THDi3	% THD I L3	58	50	[%]	[%]
kWhIII	Energia activa III	60	52	[Wh]	[kWh]
kVArLIII	Energia reactiva inductiva	62	54	[VAr]	[kVAr]
kVArCIII	Energia reactiva capacitativa	64	56	[VAr]	[kVAr]
VAIII	Potència aparent III	66	58	[VA]	[kVA]
In	Corrent del neutre	72	64	[mA]	[A]
VA1	Potència aparent 1	74	66	[VA]	[kVA]
VA2	Potència aparent 2	76	68	[VA]	[kVA]
VA3	Potència aparent 3	78	70	[VA]	[kVA]
°C	Temperatura	80	72	[d°C]	[°C]
kVAhIII	Energia Aparent III	86	78	[VA]	[kVA]

Taula 4. Paràmetres de lectura de l'analitzador.

Un cop es tenen les dades en un únic vector es tracta a partir del tercer node de funció on es modifiquen i passen totes les unitats que s'han de canviar i s'ajusten els valors corresponents, juntament amb la correcció de les falses interpretacions que pot fer de sol·licitud a l'analitzador, com per exemple en els valors negatius sol·licitats, aquestes dades com que son enviades en format complement a dos podrien ser mal interpretades com a decimals, per tant s'ha fet un petit tractament a partir de condicionants i bucles per corroborar que estan bé i si no ho estan modificar i rectificar la dada. Als annexes es troba tot el codi utilitzat en aquest tractament de dades.

Finalment es registren els paràmetres tractats a la base de dades per poder consultar al moment o posteriorment, i es visualitzen els diferents paràmetres amb un node de depuració per visualitzar que surten correctament. Recordar que aquest tasca es realitza cada cinc segons.

El segon tractament de dades es realitza cada hora un cop s'activa. En aquesta gestió de flux de dades es pretén calcular els diferents consums per hora tant d'energia, quilowatts per hora, com econòmicament, euro per cada quilowatts per hora i consums sostenibles en aquest cas s'ha decantat per la opció del consum de CO₂, en quilograms de CO₂ per cada quilowatts per hora. Així com la realització del càlcul de previsió de consum el qual no s'emmagatzema però es pot consultar si fos adient.

L'esquema del segon tractament es pot visualitzar a la figura vint-i-u.

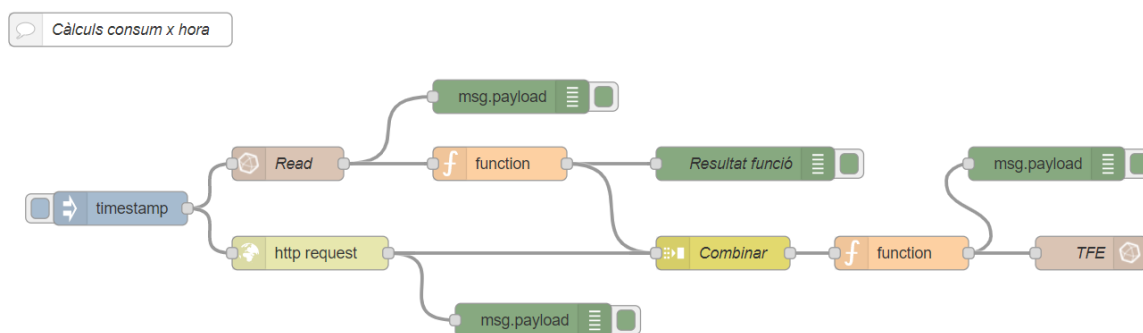


Figura 21. Tractament dades 2.

Font: Elaboració pròpia

El flux comença amb un node d'injecció el qual periòdicament (cada hora), injecta un timestamp per activar un node de lectura Influx el qual configurat amb el mateix servidor on hi ha la base de dades executa un codi de consulta:

```
SELECT WIII FROM Dades_AdX WHERE time > now()- 1h
```

Selecciona tots els registres de la potencia activa de la mesura de les dades de l'analitzador de la última hora, i els envia cap al següent node de funció on es calcula la mitjana per posteriorment poder calcular en el següent node de funció, els càlculs de consum.

Paral·lelament es consulta en temps real, és a dir cada hora el preu de la llum en una API pública que obté els valors de la REE (Red Eléctrica de España). Aquesta consulta es genera amb un node de petició HTTP el qual indicant que es vol un mètode GET, és a dir de consulta, i una URL de la API pública, et retorna en format json els valors requerits. A partir del mateix node d'injecció, s'activa aquest node. Un cop consultat, es depura el resultat de node de consulta per veure que els resultats són correctes.

Cal destacar que els càlculs de consum d'emissions es calculen a partir del mix elèctric extret de la pàgina web oficial de la generalitat i que coincideixen amb els consultats als informes de garanties i etiquetatge de la electricitat, de la CNMC (Comisión Nacional de los

Mercados y la Competencia). Als annexes d'aquest estudi es pot consultar el document sencer de l'informe.

En acabar els càlculs de consum es registren a la base de dades i també es depuren per si hi hagués quelcom error.

Finalment el tractament de dades que es fa des de Grafana el qual és senzillament programant un seguit de condicions per alertar en notificacions que certa mesura ha passat un llindar. En aquest estudi s'ha realitzat per la distorsió harmònica total del voltatge i la intensitat de les tres línies el qual en línia amb el IEEE519 es recomana que els sistemes generals, no tinguin més del 8% de distorsió. Per tant sempre i quant es passi aquest llindar per causa indiferent, la plataforma IloT avisarà via notificació a Grafana que s'esta sobrepasant al llindar.

Aquestes alertes es creen des de la interfície de Grafana, en el menú d'alertes dins d'un panell, on es poden gestionar les diferents regles, condicions, i paràmetres com destinatari de la notificació ja sigui via email, MicrosoftTeams, Discord, Hangouts, entre molts d'altres. En aquest estudi es genera una notificació que s'envia via email. Es pot observar ala interfície de creació d'alarmes a la figura vint-i-dos.

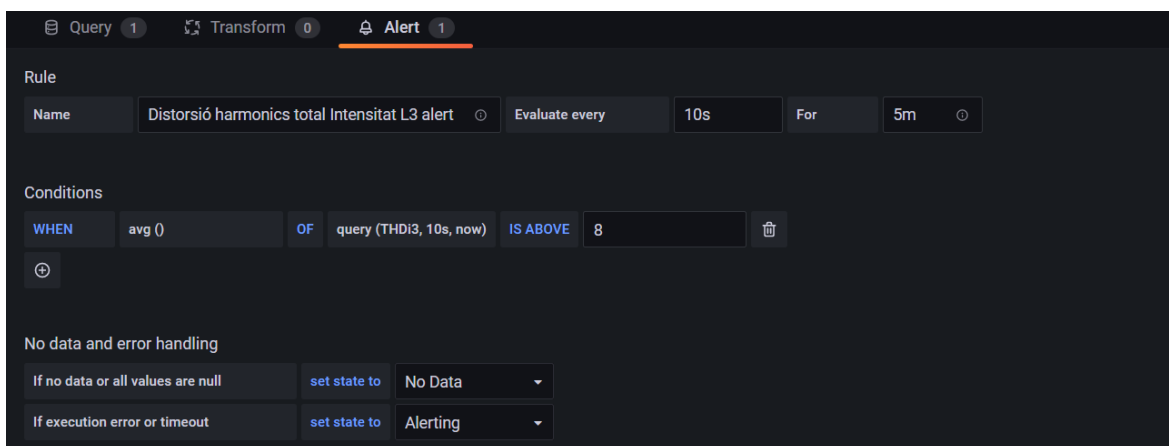


Figura 22. Interfície per programar alertes.

Font: Elaboració pròpia

4.6 Visualització informació útil i resultats

Amb el tractament de dades realitzat, l'últim pas de la solució proposada és la correcta visualització de la informació útil en *dashboards* o certs indicadors, per poder observar els resultats.

La proposta de l'actual projecte és la realització d'un sol panell on poder monitoritzar en temps real per Grafana, estant la plataforma IoT dissenyada en funcionament, de les dades de l'auditoria energètica de la cel·la industrial, la solució en un cas real a la indústria passaria per l'usuari en qüestió de la plataforma, el qual decidiria quines dades de les captades i tractades són de més utilitat pel seu cas. La interfície de Grafana fa que sigui fàcilment personalitzable cada panell en cas de necessitat. En aquest projecte però s'han posat algunes de les dades captades d'exemple, creient també la seva utilitat dins del laboratori on s'han fet les proves. Per altre banda en aquest únic panell també es pot observar les dades de consum calculades juntament amb el cost i emissions de l'hora actual. Finalment es poden observar els harmònics del sistema, juntament amb les alertes

preestablertes per avisar en cas de pic no desitjat o que pugui causar grans interferències o danys al sistema.

A la figura vint-i-tres es pots observar un dels resultats creat a partir de Grafana:



Figura 23. Panell de visualització de dashboards.

Font: Elaboració pròpia

Aquest tipus de panell podria ajustar-se a qualsevol procés industrial on es vulgui fer una auditoria energètica de consum a nivell econòmic i sostenible. On per exemple es consulta la informació tractada per la plataforma IIoT i s'observen els diferents paràmetres registrats.

En el cas concret de la figura es visualitza la gràfica d'un cert interval de temps escollit de la potència activa comparada amb l'aparent en kW. Per altre banda també mostra les gràfiques del mateix interval de temps de la Distorsió harmònica total de la línia dos i tres en tant per cent i on hi ha un llindar de vuit per cent el qual si s'apropés el valor es mostraria a la gràfica també, que si es passa enviar una alerta. I finalment es visualitzen els últims valors obtinguts de consum és a dir de la mitjana de consum de la última hora.

5 Assajos i validacions

5.1 Introducció de les proves realitzades

Els assajos i validacions d'aquest estudi es divideixen en dos tipus de proves, que s'agrupen en el següents grups:

- Proves Operacionals
- Proves Funcionals

Les proves operacionals tenen com a objectiu comprovar que el sistema, sub-sistema o el component estan en condició operativa, sense cap tipus d'error poden aplicar la acció que tenen predeterminada. Aquests tipus de proves es duen a terme com a raó de valorar i depurar els sub-mòduls o sub-blocs del sistema dissenyat o estudiat. Com podria ser per exemple la correcta configuració d'un node que llegeix dades.

Per altre banda les proves funcionals, tenen com a objectiu comprovar que els sistemes desenvolupats funcionin correctament, segons la funció o missió que tenen i es comportin amb els requisits estimats. En aquest estudi es centren en detectar els defectes de disseny o programació del sistema establert. Aquest tipus de proves es duen a terme com a raó de corroborar que el sistema, procediment, etc., es comporten adequadament. Com podria ser per exemple la correcte programació d'un flux de dades.

Al llarg d'aquest estudi s'han realitzat varies i diverses proves continuades per corroborar el correcte desenvolupament del treball realitzat i verificar els diferents procediments realitzats, tant per veure funcionalitats i operativitat dels diferents sistemes. S'ha realitzat proves d'emmagatzematge de dades, de visualització de panells, de codis en varis llenguatges de programació degut al sistema complet el qual s'implementa, de lectura i escriptura de dades, de gestió d'aquestes, d'ordres de línia, com podrien ser per exemple proves de correcta instal·lació o implementació d'un determinat software, entre moltes d'altres. No s'ha fet recull de cadascuna d'elles, degut a que a la gran quantitat, però si que es pretén detallar certes validacions, tant per veure el procediment realitzat en cada una d'elles com per afegir valor documental en aquest estudi.

5.2 Proves Operacionals

5.2.1 Verificació correcta instal·lació Docker

Descripció de la prova:

Per tal de implementar una correcte instal·lació del Docker s'executen una sèrie d'ordres a la terminal de la maquina virtual. Una vegada connectats mitjançant SSH a la VM, i després d'executar totes les ordres que indica la documentació oficial de Docker, s'ha de verificar primer de tot que s'hagi instal·lat correctament i s'hagin executat sense cap error cada una de les ordres, i segon verificar que la versió de Docker Compose hagi realitzat correctament la instal·lació ja que és un dels eix vertebrals del sistema de software implementat. Per tal de verificar el procediment s'executen dos ordres de verificació, respectivament:

```
$ sudo docker run hello-world
```

```
$ sudo docker-compose --version
```

Resultats esperats:

Com a resultats esperats es pretén visualitzar la correcte execució de les ordres sense cap tipus d'anomalia o error.

Modificació:

En aquest cas no hi ha modificació ja que els resultats obtinguts són els esperats.

Resultats obtinguts:

Com es visualitza a la figura vint-i-quatre les ordres s'executen correctament i ens indiquen que tant el Docker ha estat instal·lat correctament com la versió del Docker Compose també.

```
VinasPau@VinasFranciscoPau:~/TFE/Base folder$ sudo docker run hello-world
sudo: unable to resolve host VinasFranciscoPau

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

VinasPau@VinasFranciscoPau:~/TFE/Base folder$ sudo docker-compose --version
sudo: unable to resolve host VinasFranciscoPau
docker-compose version 1.28.5, build c4eb3a1f
```

Figura 24. Verificació correcta instal·lació Docker.

Font: Elaboració pròpia

5.2.2 Connexió al PLC i a l'Analitzador de xarxa

Descripció de la prova:

Per tal de comprovar el correcte funcionament de connexió amb el PLC i l'analitzador de xarxes amb la seva adreça IP, després que programant correctament els nodes Modbus a l'esquema de Node-Red i sense resposta de lectura, es procedeix a executar directament a la terminal de la VM una ordre per tal de corroborar que te connexió primer amb l'adreça d'uns dels PLCs de la universitat i després concretament amb l'adreça IP del analitzador de xarxa el qual es consulten les dades d'aquest estudi.

L'ordre que s'executa és una eina de diagnòstic que permet fer la verificació del estat d'una determinada connexió de un host local almenys a un equip remot contemplat en una xara de tipo TCP/IP. Permet veure si la direcció IP especifica és accessible o no.

Les ordres a executar són les següents:

\$ ping-c 4 147.83.83.29 (Adreça IP: PLC)

\$ ping-c 4 10.7.0.39 (Adreça IP: Analitzador de Xarxa)

```
VinasPau@VinasFranciscoPau:~/TFE/Base folder$ ping -c 4 147.83.83.29
PING 147.83.83.29 (147.83.83.29) 56(84) bytes of data.
64 bytes from 147.83.83.29: icmp_seq=1 ttl=50 time=19.2 ms
64 bytes from 147.83.83.29: icmp_seq=2 ttl=50 time=18.5 ms
64 bytes from 147.83.83.29: icmp_seq=3 ttl=50 time=18.4 ms
64 bytes from 147.83.83.29: icmp_seq=4 ttl=50 time=18.3 ms

--- 147.83.83.29 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 18.360/18.641/19.266/0.388 ms
```

Figura 25. Verificació connexió IP PLC.

Font: Elaboració pròpia

Resultats esperats:

Com a resultats esperats és pretén obtenir de les dos IP una correcta transmissió dels bytes de prova, ja que en l'ordre d'execució s'indica que es vol enviar quatre paquets de prova, s'espera rebre el cent per cent dels paquets enviats i en un temps raonable.

Modificació:

Després d'executar varis cops l'ordre ping per l'IP de l'analitzador de xarxa, no es rep cap tipus de resposta, és a dir que existeix un error de connexió.

Es configura l'accés UPCLink a la màquina virtual, ja que per temes de seguretat s'impedeix la connexió d'un dispositiu no identificat en una IP pròpia del centre en aquest cas la universitat UPC.

Es segueix el protocol ofert pel centre per instal·lar el programari F5FPC de F5 el qual permetrà establir la VPN en una distribució Linux per comandes. Així doncs després de la instal·lació del programari, simplement connectant amb UPCLink amb el nom d'usuari UPC i la contrasenya corresponent, s'entra a l'entorn segur de la Xarxa UPC. Per tal de veure si estem connectats ens fixarem en l'estat de la connexió executant la comanda de informació F5FPC. Si ens surt una connexió establerta, voldrà dir que tenim accés sense cap problema. Tal i com es pot observar a la figura vint-i-sis.


```
VinasPau@VinasFranciscoPau:~/TFE/Base folder$ f5fpc --info
Connection Status: session established
Favorites Information:
-----
fav-Id   fav-Type   fav-Status   fav-Name
672     vpn        established   /Common/VPN_upclink.upc.edu_Estudiants

Favorites Extended Info:
-----
Fav-Id: 672 Fav-Name: /Common/VPN_upclink.upc.edu_Estudiants
Tunnel Port: 443
Tunnel Protocol: TCP
Tunnel Security Protocol: TLSv1.2
Tunnel Cipher Strength: 128
Tunnel Hash Algorithm: AEAD
Tunnel Cipher Algorithm: AESGCM(128)
Tunnel PKI Algorithm: ECDH
Tunnel Client IPv4 Address: 10.6.10.189
Tunnel Client IPv6 Address:
Tunnel Server IPv4 Address: 1.1.1.1
Tunnel Server IPv6 Address:
Tunnel GZip Compression: Disabled
Tunnel Bytes In: 6149
Tunnel Bytes In (Low): 196
Tunnel Bytes Out: 1926
Tunnel Bytes Out (Low): 557
```

Figura 25. Informació connexió UPCLink.

Font: Elaboració pròpia

Resultats obtinguts:

Finalment i després de la modificació, com a resultat obtingut s'obté la correcta connexió amb la IP de l'analitzador de Xarxa i l'entorn UPC. Podent així consultar dades d'aquesta adreça. En la figura vint-i-sis podem visualitzar com efectivament s'envien els paquets de dades amb un temps raonable.

```
VinasPau@VinasFranciscoPau:~/TFE/Base folder$ ping -c 4 10.7.0.39
PING 10.7.0.39 (10.7.0.39) 56(84) bytes of data:
64 bytes from 10.7.0.39: icmp_seq=1 ttl=61 time=23.6 ms
64 bytes from 10.7.0.39: icmp_seq=2 ttl=61 time=18.6 ms
64 bytes from 10.7.0.39: icmp_seq=3 ttl=61 time=18.7 ms
64 bytes from 10.7.0.39: icmp_seq=4 ttl=61 time=19.1 ms

--- 10.7.0.39 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 18.646/20.031/23.607/2.075 ms
```

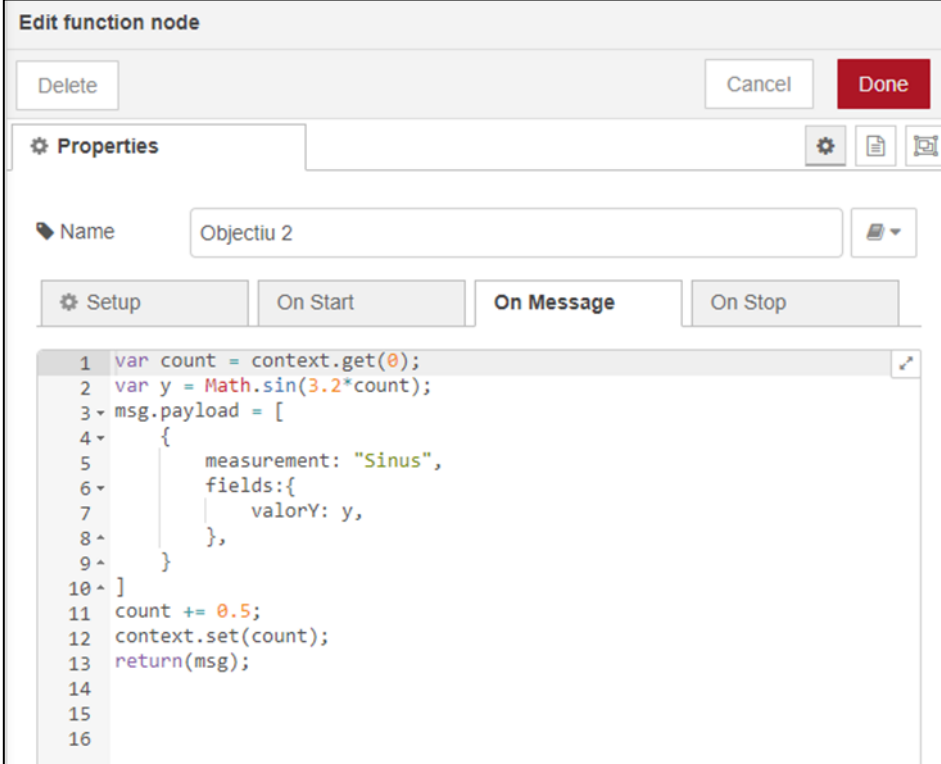
Figura 26. Verificació connexió IP Analitzador de Xarxa.

Font: Elaboració pròpia

5.2.3 Generació de senyals artificials

Descripció de la prova:

Per tal d'assajar amb dades artificials abans de captar-ne de reals, tal i com s'explica a l'apartat de creació o abocament de dades a la plataforma. Una la qual es detalla en aquest apartat és la de generació de senyals artificials en aquest cas concret una senyal sinus.



```
1 var count = context.get(0);
2 var y = Math.sin(3.2*count);
3 msg.payload = [
4   {
5     measurement: "Sinus",
6     fields:{
7       valorY: y,
8     },
9   }
10 ]
11 count += 0.5;
12 context.set(count);
13 return(msg);
14
15
16
```

Figura 27. Codi JavaScript creació senyal sinus .

Font: Elaboració pròpia

En el codi es pot observar una variable comptador la qual se li emmagatzema l'últim valor calculat amb la funció de context. Seguidament es calcula la imatge que s'emmagatzemarà a la variable generada y. Es genera el missatge indicant la mesura a la qual es vol registrar, i els diferents camps, en aquest cas només el valor y. Finalment se suma el comptador mig segon, es carrega el context i s'envia el missatge.

Per corroborar que aquest assaig és correcte i el node on hi ha el tros de codi és operatiu es genera un panell per observar si en el cert temps que s'ha generat el sinus es dibuixa correctament en un gràfic.

Resultats esperats:

Com a resultat esperat es pretén poder visualitzar per Grafana la dada artificial generada correctament i de manera visual.

Modificació:

En aquest cas no hi ha modificació ja que els resultats obtinguts són els esperats.

Resultats obtinguts:

Com a resultats obtinguts d'aquest assaig realitzat en el procés de creació i abocaments de dades s'obté un dibuix visual correcte del sinus, observat en cert temps i emmagatzemat correctament a la base de dades. Tal i com es pot veure a la figura vint-i-vuit.

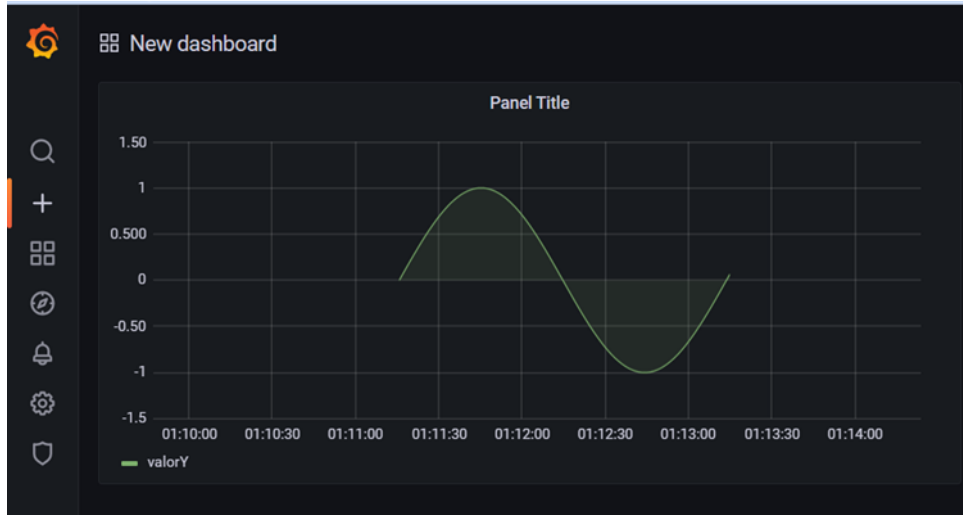


Figura 28. Visualització per Grafana del sinus .

Font: Elaboració pròpia

5.2.4 Accés a la informació de l'API pública

Descripció de la prova:

Per tal de comprovar que es consulta correctament les dades de la API pública que s'utilitza en aquest estudi, és a dir que el node de petició HTTP és operatiu. Es mirat amb un debug si la informació en format json coincideix amb la proposada a la pagina web de l'organització.

<https://api.preciodelaluz.org/>

Resultats esperats:

La URL que s'utilitza es per agafar el preu de la llum en el moment de la consulta per tant hauria de coincidir amb el següent json.

```
{  "date": "26-04-2022",  
  "hour": "16-17",  
  "is-cheap": true,  
  "is-under-avg": true,  
  "market": "PVPC",  
  "price": 283.17,  
  "units": "€/Mwh"  }
```

Modificació:

En aquest cas no hi ha modificació ja que els resultats obtinguts són els esperats.

Resultats obtinguts:

S'obté el json esperat a partir de la finestra de visualització de depuracions.

```
msg.payload : Object
  ▼ object
    date: "26-04-2022"
    hour: "16-17"
    is-cheap: true
    is-under-avg: true
    market: "PVPC"
    price: 283.17
    units: "€/Mwh"
```

Figura 29. Visualització depuració petició API.

Font: Elaboració pròpia

5.3 Proves Funcionals

5.3.1 Captació i escriptura de dades de l'analitzador

Descripció de la prova:

Assaig per captar dades i escriure-les en la base de dades correctament. A través de nodes de depuració es corrobora que la forma dels objectes dels missatges rebuts i l'esquema de gestió del flux d'informació sigui correcte.

Resultats esperats:

Rebre correctament en un objecte els diferents valors dels paràmetres amb el seu camp especificat. Tal i com s'ha programat en el codi. Posteriorment visualitzar per interfície de InfluxDB si s'han enregistrat correctament les dades. Podent dir així que aquest primer sistema funciona correctament.

Modificació:

En aquest cas no hi ha modificació ja que els resultats obtinguts són els esperats.

Resultats obtinguts:

Com a resultat obtingut s'obtenen correctament els valors esperats en la forma adient. Tal i com està programat en un format objecte. Es recorda que es pot veure en detall la programació d'aquest bloc al document d'annexes d'aquest estudi.

time	A1	V1	W1	Q1	FP1
1648143083210519955	0.441	227.8	0.063	-0.025	0
1648143102629231532	4.372	227.4	0.743	0.655	0
1648143132588057164	3.881	226.9	0.592	0.642	0
1648143234748369939	0.441	228.2	0.063	-0.025	0
1648143265505127818	0.428	228.2	0.05	-0.025	0
1648143268400078946	0.441	227.8	0.05	-0.037	0
1648143273391621645	0.441	228.3	0.063	-0.025	0
1648143278394695373	0.428	228	0.05	-0.025	0
1648143283395364081	0.441	228.4	0.05	-0.037	0
1648143288402524101	0.441	228.8	0.063	-0.025	0
1648143293409758610	0.441	228.6	0.05	-0.037	0
1648143298404055661	0.441	226.5	0.05	-0.025	0
1648143303408807839	0.454	228.1	0.063	-0.025	0
1648143308412112101	0.441	227.9	0.063	-0.025	0
1648143313414269847	0.454	228	0.05	-0.037	0

```

fd6ef1cc5e08b2bb : msg.payload : array[1]
  ▼ array[1]
    ▼ 0: object
      measurement: "Dades_AdX"
      ▼ fields: object
        A1: 0.466
        V1: 228.9
        W1: 0.063
        Q1: -0.025
        FP1: 0
  
```

Figura 30. Visualització dades enregistrades a InfluxDB Figura 31. Visualització depuració dades captades.

Font: Elaboració pròpia

Font: Elaboració pròpia

5.3.2 Implementació sistema final de temps real

Descripció de la prova:

Per tal de corroborar la funcionalitat de l'esquema programat a Node-Red que permet la gestió pel tractament de dades pel càlcul de consums. Es vol observar la seva correcta visualització per pantalla en Grafana.

Resultats esperats:

Els resultats esperats en aquest apartat són la correcta visualització dels valors de consum, cost, emissions pels panells configurats a Grafana. Així com un completa funcionalitat del esquema dissenyat juntament amb la correcta configuració de Grafana.

Modificació:

S'indica que es fa una modificació en l'estat del panell de visualització perquè consulti l'última dada consultada dels paràmetres d'emissió. En comptes de consultar cada cinc segons com la resta de dashboards en temps real.

Resultats obtinguts:

Com a resultats obtinguts i tal i com es veu a la figura trenta-u es veuen els consums després de configurar correctament la dada consultada com a la última registrada, ja que si consultava cada cinc segons el sistema no trobava dada emmagatzemada.



Figura 31. Visualització dades enregistrades a InfluxDB

Font: Elaboració pròpia

6 Resum del pressupost

El fet que aquest treball s'hagi implementat com a forma d'estudi, genera una sèrie de tasques les quals s'indiquen com a pressupost establert per les hores aplicades, juntament amb els honoraris professionals pel cost tècnic de redacció del projecte.

El pressupost total d'aquest projecte ve definit pels costos directes per les tasques realitzades i les hores de personal invertides en cada fase, i els costos indirectes de l'estudi com per exemple el cost de l'energia elèctrica emprada, el cost de combustible per desplaçament, el cost de la màquina virtual en marxa, el cost de les llicències de software utilitzat, entre d'altres. Els costos indirectes representen un 15% del total de costos directes.

La despesa total per costos directes es comptabilitzen en les diferents fases del projecte : l'estudi previ, el procés de desenvolupament aplicació, la implementació i posada en marxa i la redacció de la memòria

El pressupost per costos directes totals de 7.584€, mentre que el pressupost per costos indirectes total és de 1.137,6€.

S'ha de tenir present que en aquest càlcul no es té en compte i per tant no s'aplica el corresponent d'IVA, ni tampoc és en compte el cost d'ús de la plataforma i el servei que pot oferir, sinó simplement el pressupost d'estudi i implementació de la plataforma IoT. Es pot consultar tot el detall en el document de pressupost.

El pressupost total del projecte és de 8.721,6€.

7 Anàlisi i valoració de les implicacions ambientals

L'actual estudi no presenta cap gran impacte pel medi ambient tant positivament com negativament en implicacions ambientals.

Seguidament es fa una valoració dels avantatges i inconvenients de la implementació de la plataforma IoT en implicacions ambientals.

Els principals inconvenients bàsicament són que la plataforma necessita energia elèctrica per estar en funcionament ja que a part dels dispositius electrònics que s'utilitzin per connectar-s'hi a través del núvol que ja tenen un cert consum, la plataforma està ubicada en un servidor remot d'una gran empresa de serveis al núvol, per tant es necessita alimentar un gran nombre de dispositius electrònics. En el cas concret el servidor està ubicat a França, un país on més de la meitat d'electricitat produïda, prové de centrals nuclears.

Per contra aquest estudi busca implementar procediments per mirar de conscienciar aquest consum elèctric i desenvolupar la societat cap a una forma de fer més sostenible. És per això que un dels tractaments de dades rellevants són les emissions de CO₂. Actualment el món industrial es mesura per quilograms d'emissions de CO₂ el qual serveix per mesurar l'impacte generat ambiental. Per tant una eina com la que ofereix la plataforma permet poder fer un càlcul de la petjada ecològica del procés industrial si es fa algun tipus d'estalvi, etc. Factors que ajuden de manera important a generar aquest canvi de mindset a la societat, i el més important ajuden al seu manteniment del nostre planeta.

8 Conclusions

Un cop finalitzat aquest estudi i les seves validacions, com síntesi final es pretén obtenir unes conclusions i fer una valoració dels objectius complerts a partir de la interpretació dels resultats assolits.

L'objectiu principal d'aquest treball ha sigut l'estudi, el disseny i la implementació d'una arquitectura de comunicacions i dispositius necessaris pel desplegament d'una plataforma IIoT de consulta de dades en remot, el disseny i la implementació dels flux de gestió de dades i la implementació de l'aplicació al núvol.

S'ha aconseguit implementar tots els objectius i realitzar l'estudi satisfactòriament. Després de varis i diversos assajos i validacions es pot corroborar que s'han complert els requeriments de l'estudi i s'ha assolit la implementació de la plataforma IIoT.

Així doncs es pot afirmar que s'ha aconseguit el disseny i implementació d'una estructura de contenidors de software mitjançant Docker on ubicar la plataforma. L'automatització de la gestió de fluxos de dades, finalment duta a terme pel software Node-RED. L'enregistrament de les dades adquirides en una data base, finalment duta a terme pel software InfluxDB. La visualització de les dades i monitorització d'aquestes remotament, finalment duta a terme pel software Grafana. L'execució d'operacions de lectura i escriptura a bases de dades. I la possibilitat d'emmagatzemar l'arquitectura desenvolupada en un servidor remot, facilitant així la clonació de la màquina virtual i per tant així de la plataforma.

Per tant es pot afirmar que la plataforma generada és funcional i s'adequa a les especificacions requerides.

Per altra banda també s'ha complert la anàlisi d'informació per veure el consum del procés industrial i el seu desenvolupament, amb la finalitat de executar un procés més sostenible juntament amb la capacitat de transmetre les dades del sistema automatitzat en temps real a la plataforma IIoT.

Aspectes que enriqueixen la plataforma i la converteixen en una aplicació més avantatjosa, beneficiosa i eficaç, tant per extreure-hi dades útils i rellevants pel procés el qual s'apliqui, com per la col·laboració i transformació d'una societat cada cop més sostenible.

Es conclou també, que s'ha realitzat la part experimental correctament a les instal·lacions del laboratori d'automatització de la universitat, captant així, les dades correctament i sent la plataforma totalment funcional, podent calcular i realitzar el tractament de dades amb dades reals del laboratori, amb el matís però, que eren dades que no simulaven un procés real.

Un cop s'han complert els objectius, encara que no estava previst s'ha pogut ampliar les funcionalitats de la plataforma llegint en temps real d'una base de dades no prevista com en el cas de la consulta a l'API pública el preu de l'electricitat, donant així més precisió als càlculs i un valor afegit. Juntament en fer una senzilla previsió dels consums respecte el consumit fins al moment, per tal d'enriquir el potencial de la plataforma.

En conclusió, el treball ha complert l'objecte considerat inicialment, i addicionalment, ha ofert major capacitat de seguiment del procés industrial i la possibilitat de fer un seguiment cibernètic d'un sistema que està actuant físicament, des de qualsevol punt remot amb connectivitat a internet i en temps real.

Des del punt de vista acadèmic i tecnològic l'assoliment de tots els punts esmentats m'han format en el món de la indústria 4.0 i en l'entorn dels serveis al núvol com la monitorització per el diagnòstic d'un procés, o la preparació de dades per la posterior analítica.

Millorant les competències en diversos camps com els protocols de comunicacions industrials, els diferents llenguatges de programació emprats a l'estudi o l'aplicació de la

tecnologia de la quarta revolució industrial per tal de globalitzar i augmentar la utilitat dels sistemes tant de hardware com de software.

Finalment cal indicar també que actualment l'ús d'aquest tipus de plataforma es gairebé necessari per continuar l'evolució que s'està fent, sobretot en un món on la captació de les dades, el seu anàlisi i posterior diagnòstic a través de intel·ligències artificials cada cop és més important i està desenvolupant a les indústries a nivells exponencials. On cada cop hi ha més fluxos de dades i a més gran escala, i on la seguretat serà un factor essencial pel correcte desenvolupament i gestió de les dades.

Per tot això aquest treball dona peu a futurs estudis i extensions, o futurs TFE els quals el poden completar i enriquir, com podria ser la integració d'un sistema d'intel·ligència artificial per poder prendre decisions respecte els dades captades o la creació d'un sistema ciberfísic total on hi hagi un bessó digital del procés físic estudiat, on a partir de les dades captades a la plataforma i tractades es pugui generar.

Per altra banda també dona peu a la integració d'un sistema de registre de la petjada de carboni en la societat en que vivim, la qual indústries d'arreu del món poguessin mitjançant una plataforma com la del estudi enregistrar les dades de consum i d'emissions de CO₂, amb un hardware capaç de poder ser consultat a través del núvol. Amb possibilitat d'emmagatzemar en una base de dades encriptada o que en quedes constància en un sistema de manteniment de registre com podria ser en una base de dades amb tecnologia de cadena de blocs.

9 Referències

- [1] E. Y. Nakagawa, P. O. Antonino, F. Schnicke, R. Capilla, T. Kuhn, i P. Liggesmeyer, «Industry 4.0 reference architectures: State of the art and future trends», *Comput. Ind. Eng.*, vol. 156, núm. March, p. 107241, 2021, doi: 10.1016/j.cie.2021.107241.
- [2] Ö. Uygun i M. E. Aydin, «Digital transformation: Industry 4.0 for future minds and future society», *Comput. Ind. Eng.*, vol. 157, núm. April, p. 107362, 2021, doi: 10.1016/j.cie.2021.107362.
- [3] Schneider Electric, Las redes Industriales amplían las funciones de los PLC. [En línia]. Disponible a: <https://www.se.com/es/es/about-us/press/press-release/2019/industrial-networking-expands-plc-functionality.jsp> (Consulta: 20 setembre 2021).
- [4] Influxdata, Get started with InfluxDB OSS| InfluxData Documentation. [En línia]. Disponible a: <https://docs.influxdata.com/influxdb/v1.8/introduction/> (Consulta: 25 setembre 2021).
- [5] Ecu Red, Ethernet - EcuRed. [En línia]. Disponible a: <https://www.ecured.cu/Ethernet> (Consulta: 1 octubre 2021).
- [6] del Val Román, J. Industria 4.0: la transformación digital de la industria. Conferencia de directores i decans d'enginyeria informàtica, facultat d'enginyeria de la universitat de Deusto, Bilbao, 2016.
- [7] Pumares Cerezo, J. Estudio y automatización de una aplicación industrial basada en el transporte y la gestión de piezas. treball final d'estudis, Universitat Politècnica de Catalunya, Terrassa, 2020.
- [8] Torecilla Matencio, M. Estudio de las etapas de automatización de un proceso industrial. treball final d'estudis, Universitat Politècnica de Catalunya, Terrassa, 2020.
- [9] A. R. Rao i D. Clarke, «Perspectives on emerging directions in using IoT devices in blockchain applications», *Internet of Things (Netherlands)*, vol. 10, 2020, doi: 10.1016/j.iot.2019.100079.
- [10] Y. N. Harari, «Sapiens: A Brief History of Humankind by Yuval Noah Harari», *Guard.*, 2014.
- [11] We forum. [En línia]. Disponible a: <https://www.weforum.org/agenda/2016/01/the-fourth-industrial-revolution-what-it-means-and-how-to-respond/> (Consulta: 3 desembre 2021).
- [12] Y. Ménière, J. Philpott, J. P. Rodríguez, I. Rudyk, S. Wewege, i N. Wienold, «Patents and the Fourth Industrial Revolution (EPO)», 2020.

Apart dels estudis i projectes acadèmics consultats en deteniment, hi ha certes consultes de procediments i dades.

Seguidament és citen pàgines web consultades segons el subjecte treballat al llarg de l'estudi:

Quadre elèctric

<https://electricidadbevia.es/cuadros-electricos-industriales-caracteristicas-y-ventajas-2021/>

Edge

<https://www.ibm.com/cloud/what-is-edge-computing>

<https://www.redhat.com/en/topics/edge-computing/what-is-edge-computing>

Virtual Machine

<https://www.redhat.com/es/topics/virtualization/what-is-a-virtual-machine>

<https://azure.microsoft.com/es-es/overview/what-is-a-virtual-machine/#what-used-for>

SSH

<https://docs.microsoft.com/es-es/azure/active-directory/fundamentals/auth-ssh>

E / s màxim per segon

<https://docs.microsoft.com/es-es/azure/virtual-machines/disks-performance>

Docker

<https://www.oracle.com/es/cloud-native/container-registry/what-is-docker/>

<https://aws.amazon.com/es/docker/>

<https://www.redhat.com/es/topics/containers/what-is-docker>

Instal·lació Docker

<https://docs.docker.com/engine/install/ubuntu/>

SCP

<https://www.venafi.com/blog/what-secure-copy-protocol-and-how-use-it>

Docker compose

https://docs.docker.com/get-started/08_using_compose/

<https://docs.microsoft.com/es-es/azure/cognitive-services/containers/docker-compose-recipe>



YAML

<https://www.redhat.com/es/topics/automation/what-is-yaml>

KgCo2

https://canviclimatic.gencat.cat/es/actua/factors_demissio_associats_a_lenergia/

Node-RED

<https://nodered.org/about/>

<https://flows.nodered.org/node/node-red-contrib-modbus>

Consum CO₂

https://canviclimatic.gencat.cat/es/actua/factors_demissio_associats_a_lenergia/

https://gdo.cnmec.es/CNE/resumenGdo.do?informe=garantias_etiquetado_electricidad

Standard IEEE

<https://standards.ieee.org/ieee/519/3710/>