

Servei de monitoratge i reporting per a instal·lacions fotovoltaïques dins de la plataforma de Wattwin

Roger Física Regatero

Resum– En aquest projecte s'ha dissenyat un servei de monitoratge de plaques fotovoltaïques amb la intenció d'optimitzar els recursos dels quals disposa la plataforma de SIIE. El seu objectiu principal és el d'automatitzar uns processos que actualment s'estan fent de forma manual per l'equip d'operacions de l'empresa.

Per tal de fer això possible, s'aplicarà un procés d'extracció, transformació i càrrega (ETL) de les dades que recullen els dispositius que tenen instal·lats les plaques fotovoltaïques. Un cop passat aquest procés, totes aquestes dades ens permetran generar gràfiques que ens donaran de manera ràpida i detallada de com estan funcionant les plantes. A més de poder complementar aquesta informació amb un sistema d'alarmes personalitzable per l'usuari.

Paraules clau– Monitoratge, Fotovoltaïques, ETL, Dades de Sèries de Temps, InfluxDB, Angular, NodeJS, Alarmes, Anàlisis, Automatització.

Abstract– In this project it has been designed a service of monitoring of photovoltaic panels with the intention to improve the sources which the platform SIIE disposes. The main goal is to automate processes which now a days we are making manually through the company's operations team.

To make this possible, it will be applied a process of extraction, transformation and load (ETL) of the data collected by the devices that are installed on the photovoltaic panels. After this process is done, all the data will allow us to generate graphics that will give us the quickly and detailed method of how are working the panels. Furthermore, to complete this information with a personalized alarm system for the user.

Keywords– Monitoring, Photovoltaic, ETL, Time Series Data, InfluxDB, Angular, NodeJS, Alarms, Analysis, Automation.

1 INTRODUCCIÓ

SERVICIOS Integrales de Instalaciones Eléctricas (SIIE) és una petita empresa que ofereix un software com a servei, anomenat Wattwin, a empreses que es dediquen a vendre d'instal·lacions de producció d'energia renovable mitjançant una externalització dels serveis d'enginyeria associats.

Aquest treball consistirà a implementar un nou servei dins de la plataforma Wattwin per monitorar les instal·lacions realitzades de plaques fotovoltaïques. Aquest servei per-

metrà al propietari de les plaques, a l'equip de SIIE i als instal·ladors veure el consum i l'estat de la instal·lació i la possibilitat de rebre informes mensuals.

Per poder imaginar una mica millor la infraestructura que es pretén monitorar, a continuació es comenta com està format el sistema capaç de recollir les dades de les instal·lacions o plantes.

Cada planta, compta amb uns dispositius anomenats *loggers* que seran els que es comuniquin amb les bases de dades, en aquest cas ABB, i enviïn la informació recollida per uns altres dispositius anomenats *devices*. És per això, que les dades que es guarden seran a nivell de *device*, per així després poder filtrar i tenir una informació més detallada, tot i que després s'haurà de mostrar la informació de tots els *devices* d'una mateixa planta de manera unificada en cas de ser necessari.

• E-mail de contacte: roger.fisica@e-campus.uab.cat

• Menció realitzada: Tecnologies de la Informació

• Treball tutoritzat per: Josep Maria Basart Muñoz (Departament d'Enginyeria de la Informació i de les Comunicacions)

• Curs 2019/20

1.1 Motivació

SIIE és una empresa que encara es troba en un procés de creixement i afiançament dins del sector comercial de les energies renovables, per això és important seguir implementant noves tecnologies i serveis, tant als seus usuaris actuals com als de futur.

Abans de la realització d'aquest projecte, per conèixer l'estat de les instal·lacions era necessari consultar manualment les dades a terceres plataformes, recopilar informació i generar informes, una tasca que era simple però força costosa en temps.

És per això, que del correcte desenvolupament d'aquest servei de monitoratge s'espera que pugui captar nous clients i ser aprofitat tant pel nostre equip d'operacions com pels usuaris que actualment fan servir la plataforma.

1.2 Objectius

Quan es vol oferir un nou producte a un client, és important pensar que és el que pot necessitar i com es poden cobrir aquestes necessitats. Per tal de poder oferir un bon producte, juntament amb l'empresa es van definir un conjunt d'objectius a complir.

Aquests objectius que permetran obtenir un bon resultat final són els següents:

- Dissenyar un servei per establir connexió amb la API de ABB i fer les peticions desitjades.
- Servei capaç de comparar les dades d'ABB amb les dades que ja estan guardades a Wattwin i afegir a la nostra base de dades la nova informació dels nous dispositius de ABB.
- Dissenyar nous models de dades que ens permeti guardar la informació necessària de les plantes monitorades.
- Indexar a Elasticsearch [6] el model de les plantes monitorades per poder realitzar cerques ràpides d'aquestes dins la plataforma web.
- Dissenyar un servei per establir connexió amb la API de InfluxDB [3] i fer les peticions desitjades.
- Guardar en InfluxDB [3] les noves dades captades de manera periòdica a les instal·lacions.
- Preparar els serveis creats perquè puguin ser cridats per missatges de batch de forma periòdica.
- Crear un Endpoint per demanar i filtrar les dades guardades en InfluxDB [3].
- Preparar un mòdul a la plataforma on mostrar un llistat de totes les plantes que estan registrades.
- Mostrar dins la descripció de les plantes tota la informació de forma clara
- Implementar tots els elements desenvolupats a la vista com a components per si algun cop poden ser reaprofitats en altres llocs de la plataforma.
- Tractar les dades de InfluxDB [3] per crear gràfiques on l'usuari pugui filtrar pels camps que l'interessin.
- Adaptar la introducció periòdica de dades perquè funcioni igualment en cas d'extreure les dades d'altres bases de dades diferents de ABB.
- Un llistat de configuracions on hi haurà una llista amb les diferents fonts d'on extreure la informació de les plantes, amb l'objectiu de poder afegir noves fonts des de la plataforma. Aquestes fonts podrien ser de les bases de dades de ABB o d'altres empreses de subministraments que guardin dades de les plantes dels nostres clients.
- Dissenyar un sistema d'alarmes, amb el que l'usuari pugui personalitzar diferents tipus d'alarmes associades a les plantes, per conèixer de manera més detallada l'estat en el qual es troben aquestes.

1.3 Metodologia

Pel tipus de treball que es durà a terme es va considerar que la barreja de les metodologies Kanban [1] i Scrum [2] seria la forma més idònia de desenvolupar un bon projecte.

Kanban [1] és un tipus de metodologia àgil idònia per gestionar de manera simple com es completen les tasques. Aquesta metodologia permetrà dividir el projecte en tasques i subtasques, dur un control més detallat de l'estat en què es troben aquestes i afegir de manera fàcil noves tasques en cas de ser necessari. Per dur-la a terme s'utilitzarà el Trello, un software per administrar projectes des d'una interfície web, que permet crear targetes i classificar-les en columnes segons el seu estat.

A més, per part de la metodologia Scrum [2], setmanalment s'ha programat una reunió de seguiment amb el tutor de l'empresa per fer un seguiment del projecte i solucionar possibles dubtes que hagin sorgit durant la setmana. La idea de fer aquestes reunions és la d'evitar l'estancament en alguna de les parts del projecte, o simplement, tenir un altre punt de vista sobre com desenvolupar el projecte. A causa de l'estat en què ens trobem actualment, provocat pel coronavirus, aquestes reunions s'han realitzat virtualment.

Al final, es pot observar com cada tasca a dur a terme depèn en gran part a alguna tasca feta anteriorment, per aquest motiu, és necessari comprovar que funcionin de manera correcta cadascuna de les parts realitzades.

1.4 Planificació

En aquest apartat es mostra la planificació que s'ha dut a terme pel correcte desenvolupament del projecte.

En la taula 1 es pot veure una divisió de tasques en etapes i les hores que ha ocupat cadascuna d'aquestes. Com era d'esperar, les tasques més costoses i que més temps han ocupat han estat els primers dissenys a Frontend, ja que es van dissenyar amb el màxim nombre de components possibles. També han estat costosos els serveis de comunicació amb ABB i InfluxDB [3], a causa de les seves peculiaritats a l'hora d'accedir a les seves dades.

A la figura A1 del apèndix es poden veure com han estat repartides aquestes tasques durant el temps de desenvolupament del projecte gràcies al Diagrama de Gantt.

TAULA 1: DISTRIBUCIÓ DE LES TASQUES EN HORES

Etapes	Tasques	Hores
Planificació	Definició projecte	4h
	Disseny Mockups	6h
Backend	Servei ABB	5h
	ETL	20h
	Servei InfluxDB	20h
	Escollar la Batch	5h
	Lectura InfluxDB	25h
	Gestió alarmes	25h
	Dissenyar models	25h
Frontend	Llistat plantes	50h
	Detall planta	35h
	Disseny gràfiques	15h
	Configuració planta	10h
	Llistat configuració	2h
	Disseny alarmes	25h
	Disseny estat	2h
Test	Test implementacions	15h

2 ESTAT DE L'ART

Wattwin com moltes altres plataformes web, es divideix en Frontend i Backend. Quan parlem de Frontend, entenem que és la part del software que interactua directament amb l'usuari. Backend en canvi, és el software que processa per darrere les necessitats de Frontend, fent peticions a la base de dades o executant operacions costoses.

Actualment a Frontend s'utilitza com a *framework* una barreja entre Angular [9] i AngularJS. Això és perquè no fa gaire temps es va començar una migració de AngularJS a Angular [9], però aquest és un procés força costós per ser realitzat de cop i es va decidir migrar l'esquelet i anar migrant la resta a poc a poc. Provocant així que aquest projecte sigui dels primers a la plataforma a desenvolupar-se amb Angular [9].

Angular [9] és un *framework* per desenvolupar aplicacions web de tot tipus d'una manera més potent que AngularJS. A més compta amb molts components que ens faciliten la feina a l'hora de crear elements molt comuns a pàgines web. A l'hora de crear les gràfiques s'utilitzarà Ngx-charts [7], un *framework* molt potent enfocat al disseny de gràfiques de tota mena, que satisfà les necessitats del projecte pel que fa a la visualització de les dades monitorades.

Pel que fa al Backend, funciona amb NodeJS [8], un llenguatge de programació asíncron que permet crear un desenvolupament homogeni amb Frontend. A més, NodeJS [8] es caracteritza per ser compatible amb moltes llibreries.

Respecte a la base de dades, actualment Wattwin compta amb dues: MongoDB [5] i Elasticsearch [6]. MongoDB [5] és una base de dades NoSQL que ens permet fer peticions molt potents, amb les quals es poden fer agregacions per operar amb múltiples col·leccions. Les dades es guarden en estructures molt semblants al que seria un JSON, fet que ens permet que l'estructura de dades no hagi de canviar gaire al fer les peticions des de NodeJS [8]. En canvi, Elasticsearch [6] és una base de dades pensada per poder buscar text i filtrar dades de forma ràpida. Per això Wattwin l'utilitza per a guardar dades que després hauran de ser afegides

a llistats.

A més, per tal de guardar i fer peticions per crear les gràfiques de monitoratge, s'utilitza la base de dades de InfluxDB [3], una base de dades que no s'havia utilitzat encara a Wattwin. InfluxDB [3] és un servei de base de dades de sèries de temps, o *time series*, ideal per guardar dades destinades a generar gràfiques. Tot i això no es descarta migrar en un futur a la base de dades *time series* de AWS quan l'eina estigui disponible, ja que tota la nostra infraestructura es troba en AWS.

3 REQUISITS DEL PROJECTE

En aquest apartat s'indiquen quins són els requisits funcionals i no funcionals amb el que s'ha plantejat el projecte.

3.1 Requisits funcionals

Els requisits funcionals són tots aquells que defineixen una funcionalitat del software, i en el cas d'aquest projecte han estat els següents:

- Donar la possibilitat de poder filtrar i buscar les dades recollides de plataformes externes des de la propia plataforma de Wattwin.
- Poder crear noves font de dades des del llistat de configuració.
- Carregar les dades de les plantes que estan sent monitorades i registrar noves plantes monitorades a Wattwin quan la configuració ha afegit algun nou dispositiu de recollida de dades.
- Poder afegir manualment nous dispositius a les plantes ja existents.
- Poder editar els valors de les plantes per tal de poder-les vincular fàcilment amb dades que ens són més familiars.
- Poder vincular les plantes monitorades amb instal·lacions que han gestionat els enginyers de SIIE per tal de tenir una informació molt més completa de la planta.
- Poder vincular els *loggers* i *devices* amb productes del nostre catàleg per tal de tenir més informació dels dispositius.
- Filtrar les dades de l'apartat d'anàlisis per poder llegir les dades dels dispositius i el període de temps que es desitja.

3.2 Requisits no funcionals

Els requisits funcionals, són aquelles característiques del software que ens demostren que el projecte ha estat desenvolupat seguint els criteris pactats i per tant compleixen amb certs criteris de qualitat.

- Preparar els serveis de cerca de noves plantes i el de càrrega de dades, perquè siguin capaços de funcionar amb diferents fonts de dades, ja que en un futur s'espera no només agafar aquestes dades de ABB.

- Disseny de components a Frontend que siguin molt personalitzables i útils per altres programadors. Això permetria a la llarga estalviar feina i recursos, a més de garantir una estètica igual a tota la plataforma i la no duplictat del codi.
- Automatitzar el procés de recollida de les dades que abans havien de ser afegides manualment dins d'un Excel.
- Mostrar de manera clara i comprensible les dades que són recollides a la plataforma de Wattwin.
- Donar l'opció de carregar les dades d'un cert període de temps en cas de saber que durant aquest hi ha hagut algun tipus d'error i les dades que hi ha actualment s'hi troben buides.
- Nous dissenys a Frontend que mantinguin l'estètica de la resta de la plataforma.
- Actualitzar i monitorar únicament les dades de les configuracions que es troben en estat actiu.

4 IMPLEMENTACIÓ DEL PROJECTE

En aquest apartat, s'expliquen els passos que s'han seguit per tal d'assolir els objectius del projecte i un bon producte final. Per fer-ho dividirem els diferents passos a seguir en blocs de tasques.

4.1 Captació de dades

El primer d'aquests blocs comença un cop es va definir el projecte i es van dissenyar els Mockups del projecte. Aquest bloc consisteix en el procés ETL amb el que captarem les dades de les plantes. La primera tasca a realitzar, serà el desenvolupament del servei encarregat de consultar de manera periòdica la base de dades de ABB per tal de conèixer si s'han incorporat nous dispositius per a monitorar les plantes, o si directament hi ha noves plantes de les quals s'estan recollint dades. Per això ha estat necessari crear un servei capaç d'establir connexió amb ABB i consultar sobre els dispositius dels quals conté informació. Un cop estan tots els dispositius dels quals ABB registra informació, es comparen amb la base de dades de Wattwin, i s'afegeixen tots aquests dispositius a la nostra base de dades per començar a monitorar-los.

Per tal de poder seguir avançant i poder guardar tota la informació que es recull de les plantes, primer de tot es necessitava crear un servei que es connectés amb la API de InfluxDB [3], la base de dades *time series*. Aquest servei haurà de ser capaç no només d'establir connexió i escriure les noves dades, sinó que també haurà de ser capaç d'adaptar les peticions que es realitzen des de Frontend, al format que demana InfluxDB [3], per tal de poder llegir les dades que es sol·liciten. A més, aquest servei funcionarà com un *mixin*, per tal de poder-lo utilitzar fàcilment amb altres models de dades a part del de monitoratge d'instal·lacions. Un *mixin* és una classe que està adaptada per ser heretada de manera còmoda per subclasses, però que alhora no està pensada per ser autònoma.

Un cop la base de dades té tota la informació necessària de les plantes i els dispositius, utilitzant el servei creat

per comunicar-nos amb ABB, es consulten les noves dades que han estat captades pels dispositius de les plantes. Tota aquesta informació s'adaptarà i serà guardada a la base de dades *time series* amb el servei creat per comunicar-nos amb InfluxDB [3].

A la figura (1) es pot veure en la interfície que ens ofereix InfluxDB [3], part de les dades que hi ha guardades de les diferents plantes que ja s'estan monitorant. També es pot observar la columna *field* amb els camps dels quals estem guardant informació per a cada dispositiu. Aquests camps seran ampliat un cop el projecte estigui en una fase final, per tal de tenir més i millor informació de les plantes.

Totes aquestes funcionalitats mencionades anteriorment, han d'estar executant-se de forma periòdica, i és per això que s'ha hagut d'adaptar aquests serveis perquè siguin capaços d'escollir la cua de la Batch i executar-se en aquesta. Per entendre a què ens referim quan parlem de la Batch, podríem dir a resumits comptes, que és una dúplica de la API de Wattwin, que té com a objectiu processar els còmputos que puguin resultar més pesats. A més, s'ha creat una nova cua, per tal de no saturar la cua que ja hi havia. Aquesta execució periòdica serà la que ens permetrà conèixer en tot moment si hi ha noves plantes o dispositius a monitorar i a poder tenir una bona base de dades actualitzada en temps real.

El següent pas que es va dur a terme, va ser el de preparar el model de la nostra base de dades, per a indexar la informació bàsica de les plantes a Elasticsearch [6] per a després, poder realitzar cerques ràpides des de la plataforma web de Wattwin. Les dades a indexar, són els valors pels quals podrem filtrar, o es consideren important per ser mostrats en un petit resum de la planta.

4.2 Representació de les dades captades

En aquest punt del projecte, comença la feina a Frontend, on es mostren les dades recollides de les plantes i on s'ha de plasmar el significat d'aquestes. Per tal de mantenir la mateixa estètica a tota la plataforma i facilitar la migració a Angular [9], s'ha hagut de dissenyar tota la vista pensant en fer components que després ens permetin migrar a Angular [9] i fer les vistes de la plataforma de manera més fàcil i amb una estètica comuna.

A la figura (2), es pot veure com apareix a la barra de navegació lateral la nova pestanya de monitoratge, una pestanya que només podran veure els usuaris que tinguin els permisos necessaris. Aquesta pestanya condueix a un llistat de plantes monitorades, on cadascuna d'aquestes és representada amb una carta, que ens mostra més o menys informació depenent de si han estat vinculades amb una instal·lació gestionada des de Wattwin o no. Tota la informació que mostrem en aquest llistat prové de totes aquelles dades que anteriorment havíem indexat a Elasticsearch [6].

Aprofundint més en el llistat de les plantes, a la següent figura (3) es mostra un component dissenyat capaç de filtrar pels paràmetres que es desitgi, i es pot aplicar en tots els llistats que hi ha a la plataforma.

En aquest cas, es podran buscar les plantes monitorades per l'estat en el qual es troben, les etiquetes que l'usuari podrà posar per facilitar la seva cerca, per si tenen una instal·lació associada, per la potència de la planta i per la data en què es

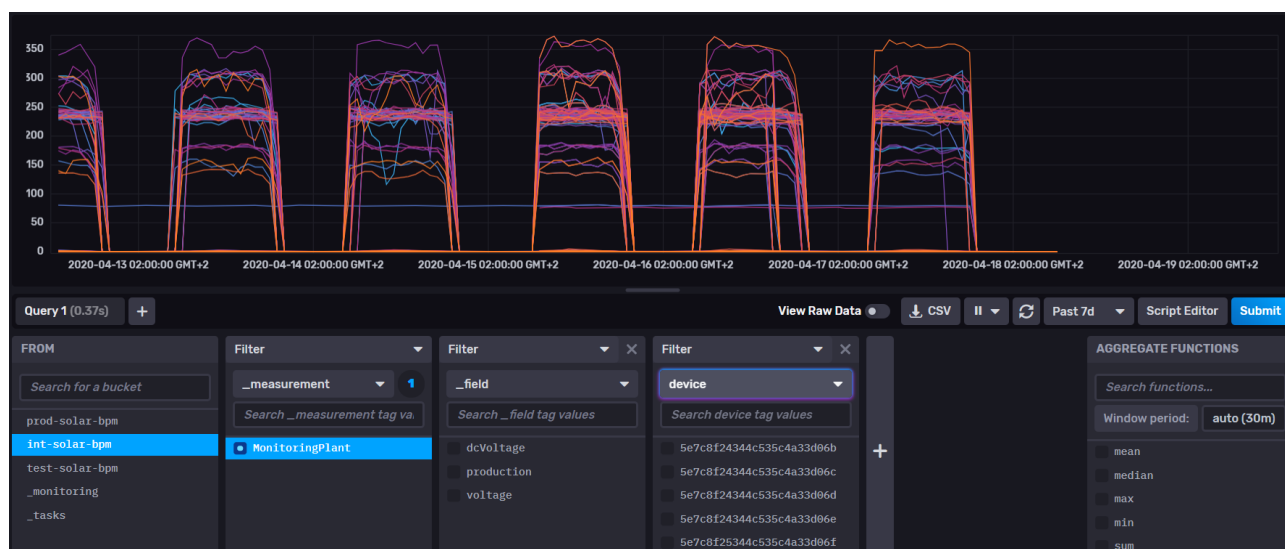


Fig. 1: Interfície de InfluxDB on es poden veure algunes de les dades guardades.

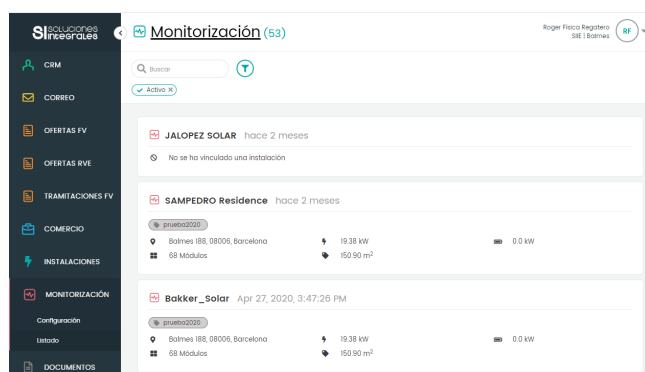


Fig. 2: Llistat amb totes les plantes monitorades.

van crear a la base de dades.

Per poder imaginar la mida dels components que s'estan realitzant, està bé saber que aquest component de cerca avançada està utilitzant un total de 9 components més en el seu interior. Comentar també, que gràcies als components dissenyats, la migració d'altres llistats de la plataforma està sent molt fàcil, i molts d'aquests llistats ja han estat migrats.

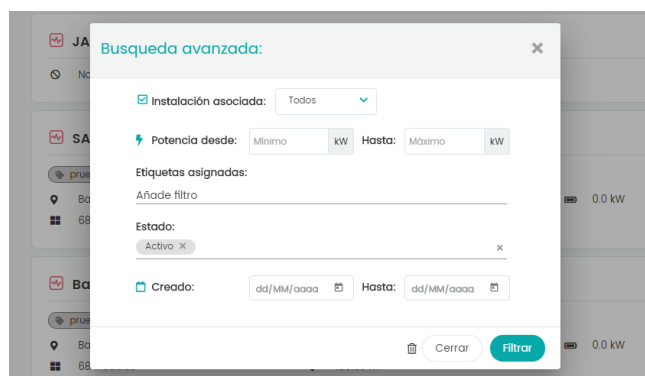


Fig. 3: Component amb diferents filtres de cerca.

Un cop el llistat està finalitzat, es defineix el disseny del detall de la planta, on mostrarem de forma organitzada les dades que es llegeixen de ABB.

Tal com es pot veure a la figura (6), a la part superior hi ha

dues targetes amb diferents pàgines cadascuna. A la targeta de l'esquerra es mostra la informació de la planta que ha pogut estar modificada i a la segona pàgina les dades originals. A la targeta de la dreta es pot veure en primera cara les dades de l'última vegada que han estat monitorades i a la segona més informació extra de la planta. A més, a la part inferior de la imatge, es veuen quatre pestanyes, una d'anàlisis, una amb l'estat de la planta, un altre per definir les alarmes de la planta i una de la configuració de la planta.

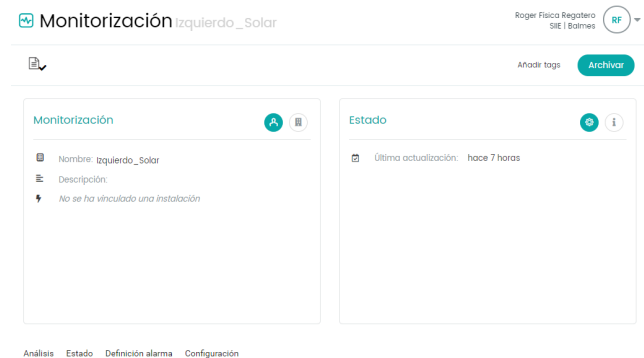


Fig. 4: Detall de la planta, on es mostra tota la informació d'aquesta.

A l'apartat d'anàlisis, a la figura (5), es poden veure les dades que han recollit les plantes en temps real de forma clara gràcies a la implementació d'una gràfica en la qual podem filtrar per períodes de temps, per *loggers*, *devices* o per grups de dades. Aquesta gràfica pertany a la llibreria de Ngx-charts [7] i permetrà fer un component d'aquesta per tal de reaprofitar-la a altres llocs de la web.

Per tal d'agafar les dades de InfluxDB [3], s'ha utilitzat el servei dissenyat per poder fer lectures a la base de dades i preparar la resposta des de Backend, per tal de fer quadrar les dades amb el format correcte per plasmar la informació a la gràfica estalviant el còmput a Frontend. A la figura (6) es pot veure quina va estar l'última implementació d'aquest bloc de tasques, com és la implementació de l'apartat de configuració logística dels dispositius de la planta.

En aquest apartat, el que hi ha és un llistat amb els diferents

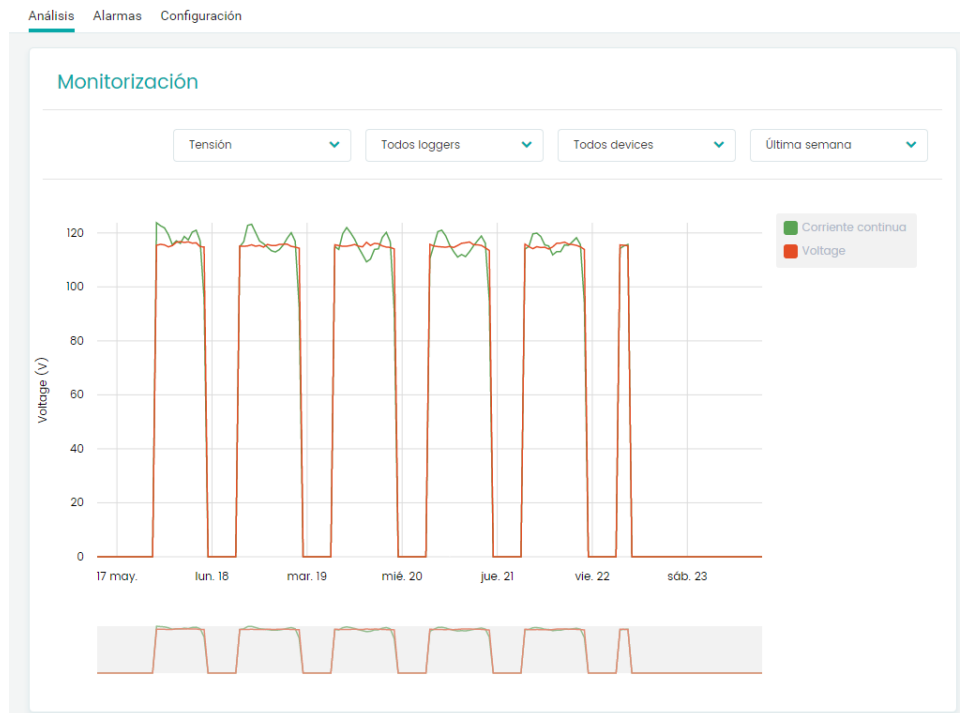


Fig. 5: Gràfica amb les dades guardades en ABB.

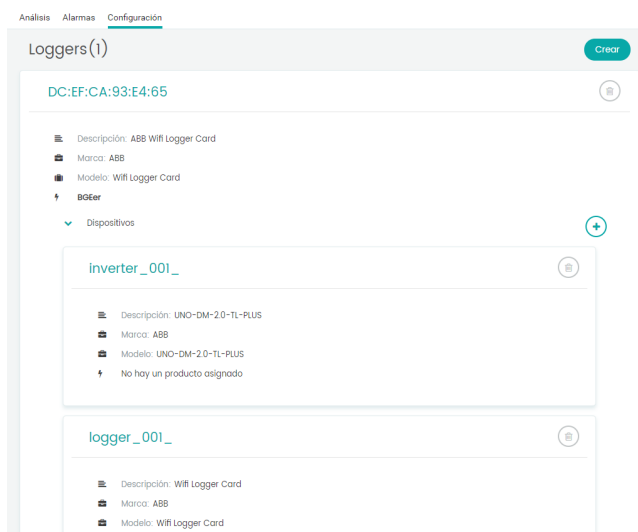


Fig. 6: Llistat amb els *loggers* i *devices* de la planta, i les seves característiques.

loggers i *devices* amb els quals s'està fent el monitoratge de la planta. Dins d'aquest llistat, es poden veure algunes de les dades originals recollides dels dispositius, però tots aquests camps podran ser modificats per tal de poder treballar de la manera més còmoda.

Dins d'aquesta pestanya, també es podran crear nous *loggers* i *devices* associats a la planta en què ens trobem, amb el pop-up de la figura (7), on podrem donar un nom, una descripció i nomenar el model i la marca del dispositiu. A més, s'ha dissenyat un pop up perquè cada logger i *devices* pugui ser vinculat a un producte que es troba al catàleg dels productes que ofereix la plataforma, tal com es mostra a la figura (8).

Una altra funcionalitat que resideix dins d'aquest llistat és



Fig. 7: Pop-up per crear nous *loggers* i *devices* a la planta.

el de poder afegir nous *loggers* i *devices* associats a la planta. Una funcionalitat que serà molt útil en un futur si es vol prescindir d'agafar les dades de terceres empreses com pot ser ABB.

4.3 Configuració de les dades

En aquest apartat, s'ha implementat un sistema que ens permeti de manera fàcil crear noves fonts de dades per tal de monitorar més plantes, o bé, carregar noves dades en qual-sevol moment.

El primer pas, ha estat crear un el llistat de la figura (9) aprofitant els components creats, però per això, abans és important preparar el model de configuració per a poder ser indexat a Elasticsearch [6].

Aquest llistat funciona exactament igual que el llistat de plantes i en seleccionar una configuració ens dirigeix al seu detall, tal com mostra la figura (10).

Dins d'aquest detall, es pot veure i editar tota la informació

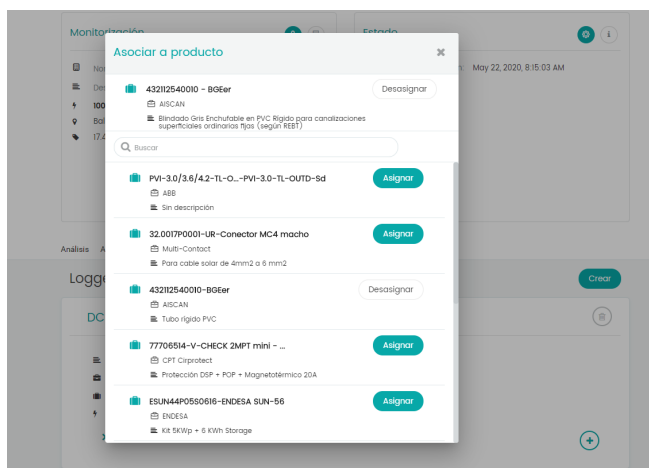


Fig. 8: Pop-up per associar els dispositius amb els productes del catàleg.

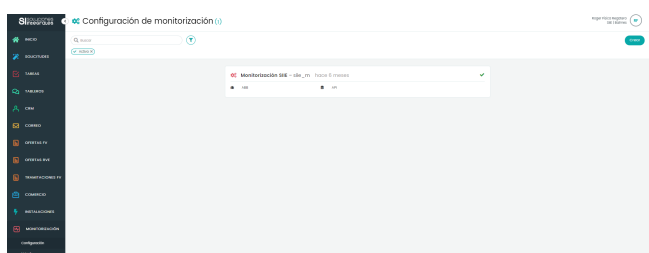


Fig. 9: Llistat amb les diferents configuracions.



Fig. 10: Detall de la configuració de ABB.

de la configuració que després permet establir connexió a ABB amb un usuari i agafar les dades de les plantes associades a la configuració.

També veiem com dins la targeta de la dreta hi ha tres botons, un per tenir la possibilitat de provar si la connexió funciona correctament, i els altres dos per dur a terme les accions de buscar noves plantes i carregar les dades que han registrat les plantes un cert període de temps.

A més, a la part inferior trobem un llistat de les plantes que pertanyen a aquesta configuració i permet veure les dades que estem monitorant de la configuració concreta de manera fàcil.

4.4 Gestió d'alarmes

En aquesta última activitat l'objectiu ha estat crear un sistema d'alarmes que fos personalitzable des de la plataforma de Wattwin per qualsevol usuari.

Aquestes alarmes controlaran l'estat de les plantes, i permetran veure si han deixat de notificar noves dades o si les dades que estan captant les plantes compten amb alguna anomalia.

Perquè això sigui possible, s'utilitzarà el sistema d'alarmes amb el que conta InfluxDB [3], que ens permetrà executar aquestes alarmes de manera periòdica. Com ja s'esmenta anteriorment, hi haurà dos tipus d'alarmes que són les següents:

- **Threshold:** Comprova si les dades són dins dels valors desitjats.
- **Deadman o llindar:** Comprova que entrin dades i que les que entren no siguin únicament zeros.

Aquest servei per a la gestió d'alarma el trobarem dins del detall de la planta juntament amb les pestanyes d'anàlisi i configuració que s'han vist anteriorment.

El servei que ens ajudarà a crear la definició de les alarmes, es veu a la figura (12). Aquí l'usuari veurà un llistat amb totes les alarmes que té definides la planta i podrà crear noves definicions d'aquestes per tal de veure amb més detall la informació registrada per cadascun dels dispositius de la planta, amb el pop-up de la figura (11).

Per fer tot això possible, prèviament s'hi hauran de

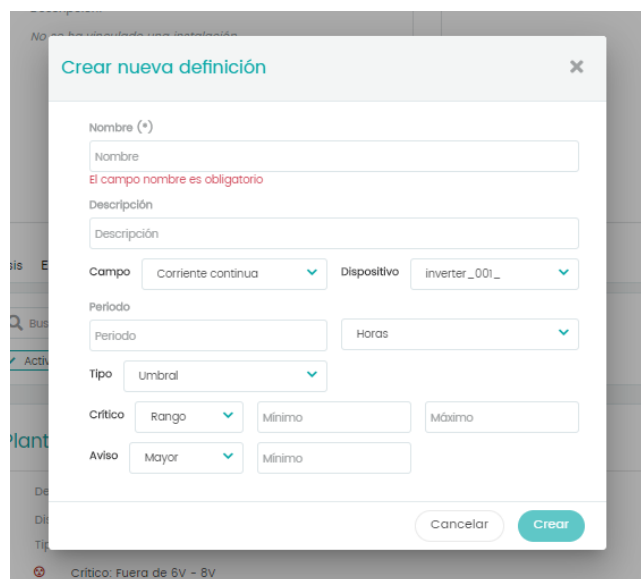


Fig. 11: Pop-up per crear noves alarmes associades a la planta actual.

realitzar un parell de tasques a Backend. Primer de tot preparar un model de dades per guardar a les bases de dades de MongoDB [5] i Elasticsearch [6] les alarmes definides, i a més, s'haurà d'estendre el servei de comunicació amb InfluxDB [4], per tal que sigui capaç de gestionar via API, la gestió de les alarmes que creem des de la nostra plataforma. Perquè aquest servei compleixi amb les necessitats del projecte, ha de ser capaç de crear, actualitzar, eliminar i rebre els resultats de les alarmes.

Tal com podem veure, a l'hora de crear l'alarma, es podrà especificar el tipus d'alarma que volem i en cas de ser de tipus threshold els rangs de valors pels quals es vol que notifiquin l'estat del dispositiu.

Aquestes alarmes definides seran guardades a la nostra base de dades de MongoDB [5] per tal de poder-les indexar a

ElasticSearch [6] i definir el llistat que hi ha a la figura (12). Un cop aquestes definicions d'alarma siguin guardades a Wattwin, s'agafarà la ID d'aquesta i es crearan a InfluxDB [3] amb el nom d'aquesta ID, de manera que després es puguin demanar els resultats d'aquestes.

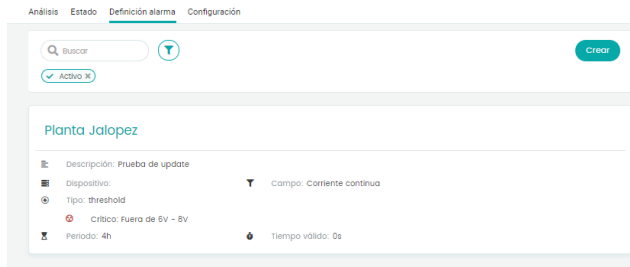


Fig. 12: Llistat amb les alarmes associades a dispositius de la planta.

Si salten les alarmes quedaran guardades a una nova taula a InfluxDB [3] i després podrem consultar aquesta amb el mateix servei de lectura que es va dissenyar per la lectura de les dades de l'anàlisi.

A la figura (13) es veu la pestanya de l'estat de la planta on imprimirem els resultats generats per les alarmes de la planta en qüestió. Aquí es mostra una taula amb diferents columnes que mostren l'alarma que ha saltat, el dispositiu a la que està vinculada, el tipus d'alarma que era, quan s'ha produït i l'estat en què es troba el dispositiu respecte a l'alarma.

5 RESULTATS

En aquest apartat, es mostra el resultat final obtingut en finalitzar el projecte i els problemes que s'han trobat durant el desenvolupament del projecte.

5.1 Resultat del projecte

Donat que l'objectiu principal del projecte era el de dissenyar un servei capaç de monitorar les plantes de les plantes que la seva instal·lació ha estat gestionada pels enginyers de l'empresa de SIIE, es pot dir que el projecte ha estat un èxit.

El servei dissenyat és capaç de detectar si han entrat noves plantes a monitorar de manera automàtica gràcies al fet que disposem d'una cua FIFO a la que entren missatges periòdicament recordant a la Batch que ha de fer una comparació de les plantes que ja estan sent monitorades a Wattwin amb les plantes que conta la configuració d'origen de les dades.

Paral·lelament a tot això, la Batch també està rebent, amb molta més freqüència, un missatge per fer l'extracció, transformació i càrrega de les dades que han recollit els *loggers* i *devices* dins de la base de dades de sèries de temps InfluxDB [3].

Totes aquestes plantes, es veuen de manera clara dins de la nostra plataforma web dins d'un llistat que segueix l'estètica de la pàgina. Dins d'aquest llistat podrem veure un breu resum de la informació de cadascuna de les plantes

i podrem fer cerques avançades gràcies al component que es va dissenyar per això.

En seleccionar una d'aquestes plantes, això conduirà a una nova vista amb el detall de la planta, que ens mostrarà d'una manera més ordenada i clara la resta de la informació que tenim de la planta. En aquest detall, tindrem un històric amb les dades agafades de ABB, però les que es mostren en primera plana podran ser editades per tal que el client pugui treballar de manera més clara des de la nostra plataforma.

Encara dins del detall de la planta, trobarem quatre pestanyes que tractaran de donar-nos la resta d'informació necessària de la planta. Aquestes pestanyes són les següents:

Anàlisis: Mostra la informació recollida de la planta fent ús d'una gràfica. La informació de la gràfica podrà ser filtrada pel device o logger de la planta que es volen veure, per període de temps o per grup de valors.

Estat: Mostra les últimes alarmes que ha fet saltar la planta, i per tant, ens dona informació de l'estat actual de la planta i els seus dispositius.

Definició d'alarmes: Permet generar noves definicions d'alarmes per reportar l'estat de la planta i els seus dispositius. A més, permet veure un llistat amb les definicions d'alarmes associades a aquesta planta i editar aquestes en cas de ser necessari.

Configuració: Mostra un llistat de la configuració de la planta amb els diferents *loggers* i *devices* que la conforma. Aquests dispositius podran ser vinculats amb productes del nostre catàleg i es podrà editar la informació d'aquests dispositius.

Hi ha també un llistat amb les diferents configuracions, on cada configuració dona la informació necessària per poder accedir a les bases de dades que tenen la informació de les plantes monitorades. En aquest llistat, es pot crear noves configuracions que estaran en estat de pausa mentre no hagi estat validat l'accés a la nova font de dades.

Igual que amb el llistat de les plantes, podrem accedir al seu detall i editar les seves característiques, a més de poder arxivar la configuració per tal de deixar de llegir dades d'aquest.

Cal destacar que totes les implementacions que s'han realitzat a Frontend, s'ha fet en components tal com es demanava per tal de facilitar la migració Angular [9].

Per fer tot això possible, s'han implementat també els serveis corresponents a Backend, per tal d'establir una comunicació completa tant amb la API de ABB, com amb la de InfluxDB [3], de tal manera que qualsevol altre servei encara que no fos de monitoratge de plantes pugui aprofitar aquestes eines.

5.2 Problemes trobats

És cert que durant el desenvolupament del projecte no ha sorgit cap problema greu que hagi pogut comprometre la finalització exitosa del treball, però com és normal en un projecte d'aquestes magnituds, i sobretot, en un projecte

Análisis

Estado

Definición alarma

Configuración

Estado de la planta:

Fecha	Definición	Dispositivo	Tipo	Nivel	Campo
13 de jun. de 2020 2:00	int - dcVoltage	inverter_001_	deadman	Crítico	Corriente continua
13 de jun. de 2020 2:00	int - dcVoltage	Device_002_	deadman	Crítico	Corriente continua
13 de jun. de 2020 2:00	int - dcVoltage	logger_001_	deadman	Crítico	Corriente continua
13 de jun. de 2020 2:00	int - dcVoltage	meter_001_	deadman	Crítico	Corriente continua
13 de jun. de 2020 2:00	int - dcVoltage	meter_002_	deadman	Crítico	Corriente continua
13 de jun. de 2020 2:00	int - dcVoltage	inverter_001_	deadman	Crítico	Corriente continua

Fig. 13: Taula amb els resultats de les alarmes de la planta.

on gairebé tot el que fas és nou per a tu, han sorgit alguns problemes o canvis respecte a la planificació que es va fer inicialment.

5.2.1 Canvis en la planificació

El projecte realitzat s'ha mantingut molt estàtic respecte els temps de desenvolupament des de l'inici del mateix i tampoc ha patit cap canvi en què respecta la seva funcionalitat. Tot i això, si es va produir un canvi en la seva planificació, i és que en un inici, la idea era la de crear o implementar un connector amb InfluxDB [3], que ens permetis crear una connexió directa del nostre Backend amb una base de dades NoSQL.

Tot i que la idea de crear aquest connector fos segurament la millor idea per tal d'integrar molt millor InfluxDB [3] en la nostra infraestructura, i ens donés un mecanisme de comunicació molt més adient, juntament amb el tutor de l'empresa es va decidir que crear un connector de zero era massa costos per un projecte d'aquestes característiques.

Això va fer que la idea del disseny d'un connector quedés aparcada com una futura possible millora, però que actualment, tota la comunicació amb InfluxDB [3] la gestionaríem nosaltres des de la nostra API fent crides directament contra la seva API.

5.2.2 Problemes trobats

El problema més gran que s'ha trobat durant el desenvolupament del projecte va ser que sense previ avís ABB va tancar l'URL a la que estàvem realitzant les peticions de la API. Això va provocar que el ETL deixés de funcionar als tres entorns que té l'empresa (int, test i producció).

La causa del problema va trigar a ser detectada, ja que en cap moment es va sospitar de què l'URL a la que sempre havíem fet les peticions hagués deixat de funcionar. Per tal de trobar l'error, ens vam posar en contacte amb ABB per conèixer si estàvem realitzant malament algun pas o si algun dels camps de què hi havia a la configuració de ABB era errònia.

Finalment, es va acabar detectant l'error gràcies a anar pas

a pas revisant tot el procés de connexió amb ABB, i va permetre trobar algun error que no estàvem capturant correctament, fet que va provocar, que a la cua de la Batch que escoltava els missatges de monitoratge arribés a acumular més de 300 missatges per llegir.

Aquest problema, també va provocar l'aparició d'un nou botó al detall de la configuració que permetés actualitzar les dades d'un període de temps per tal de no tenir buits de dades en la nostra base de dades de temps de sèries.

Altres problemes que s'han trobat tampoc han tingut gran complexitat a l'hora de buscar i solucionar l'error.

6 CONCLUSIÓ

En aquest apartat es pot veure un breu resum de tot el que s'ha mencionat anteriorment al document i al final d'aquest, s'explica una mica per sobre les futures millores que es faran al projecte.

El resultat del desenvolupament del projecte ha estat força satisfactori, ja que tots els objectius que es van marcar juntament amb l'equip de desenvolupadors sèniors de SIIE durant la definició del projecte han estat complerts dins dels terminis del Treball de Fi de Grau. Es pot dir que s'hi ha aconseguit dissenyar un servei de monitoratge per a les plaques fotovoltaïques dins de la plataforma de Wattwin amb èxit.

És cert que s'han trobat problemes com va estar la impossibilitat de lectura de dades de ABB durant un cert període de temps o la no implementació d'un connector a InfluxDB [3], però tots aquests problemes han estat solucionats sense gaires dificultats.

Encara que s'hagin complert tots els objectius que hi havia, el projecte es troba en un estat de proves, en el que es tractarà de veure quins resultats donen les alarmes a InfluxDB [3] per tal de treure el màxim de profit possible.

A més, com a tasques futures ja fora del Treball de Fi de Grau, es treballarà per seguir millorant el servei de Backend que es comunica amb InfluxDB [3], per tal d'explotar al màxim el seu potencial, ja que s'està veient que és una eina molt interessant. Algun exemple seria el d'habilitar el sistema d'alarmes perquè pugui crear alarmes per a grups de

plantes des de la plataforma o rebre les notificacions d'aquestes directament a la plataforma, sense haver de preguntar per elles. També seria interessant el fet de programar tasques perquè les dades guardades un temps enrere quedin agrupades en comptes de per hores per mesos, mostrant-nos així un altre tipus d'informació amb la qual no s'haurà d'operar si no interessa.

AGRAÏMENTS

Primer de tot, agrair a l'empresa SIIE per deixar-me fer el Treball de Fi de Grau amb un projecte com aquest i sobretot, al Jordi Alborch, qui ha estat el meu tutor a l'empresa i ha mostrat molt interès per veure en quin estat es trobava aquest.

Per altra banda, també vull agrair al meu tutor acadèmic Josep Maria Basart, ja que tot i que no he requerit la seva ajuda, des del principi del projecte a la reunió inicial em va deixar clar que si tenia cap mena de problema tractaria de fer el possible per ajudar-me.

També m'agradaria agrair als companys més sèniors que m'han donat un cop de mà sempre que ho he necessitat i han estat presents a totes les reunions de seguiment per tal d'aportar noves idees.

REFERÈNCIES

- [1] Kanban [online]. Disponible a internet: <https://es.wikipedia.org/wiki/Kanban>
- [2] Scrum [online]. Disponible a internet: <https://proyectosagiles.org/que-es-Scrum/>
- [3] InfluxDB [online]. Disponible a internet: <https://www.influxdata.com/>
- [4] Influx API Service [online]. Disponible a internet: <https://v2.docs.influxdata.com/v2.0/api/>
- [5] MongoDB [online]. Disponible a internet: <https://docs.MongoDB.com/manual/>
- [6] Elasticsearch [online]. Disponible a internet: <https://www.elastic.co/es/elastic-stack/features>
- [7] Ngx-charts [online]. Disponible a internet: <https://swimlane.gitbook.io/ngx-charts/>
- [8] NodeJS [online]. Disponible a internet: <https://nodejs.org/es/docs/>
- [9] Angular [online]. Disponible a internet: <https://angular.io/>

APÈNDIX

A.1 PLANIFICACIÓ DEL PROJECTE

