



**UNIVERSIDAD DE CASTILLA-LA MANCHA**

ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA

**GRADO EN INGENIERÍA INFORMÁTICA**

TECNOLOGÍA ESPECÍFICA DE

INGENIERÍA DEL SOFTWARE

TRABAJO FIN DE GRADO

**“**[**Uso de tecnología CEP para la detección de desgaste en aerogeneradores**](https://www.esiiab.uclm.es/asig.php?curso=2017-18&codasig=42358&codtfg=1397)**”**

Enrique Brazález Segovia



Junio de 2018



**UNIVERSIDAD DE CASTILLA-LA MANCHA**

ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA

**GRADO EN INGENIERÍA INFORMÁTICA**TECNOLOGÍA ESPECÍFICA DE

INGENIERÍA DEL SOFTWARE

TRABAJO FIN DE GRADO

[**Uso de tecnología CEP para la detección de desgaste en aerogeneradores**](https://www.esiiab.uclm.es/asig.php?curso=2017-18&codasig=42358&codtfg=1397)

|  |  |
| --- | --- |
| Autor: | Enrique Brazález Segovia. |
| Directores: | Gregorio Diáz Descalzo. |

Junio 2018

A todos aquellos que me han ayudado y formado parte de estos 4 años de carrera. Gracias.

Declaración de Autoría

 Yo, Enrique Brazález Segovia, con DNI 74519608E, declaro que soy el único autor del Trabajo Fin de Grado titulado “Uso de tecnología CEP para la detección de desgaste en aerogeneradores” y que el citado trabajo no infringe las leyes en vigor sobre propiedad intelectual y que todo el material no original contenido en dicho trabajo está propiadamente atribuido a sus legítimos autores.

Albacete, a jueves, 15 de marzo de 2018.



 Fdo.: Enrique Brazález Segovia.

Resumen

Internet of Things es un concepto tecnológico que está revolucionando la forma de concibir nuestro día a día estos últimos años. La integración de sistemas informáticos en nuestros hogares, en nuestros puestos de trabajo, en la industria es muy notable, y en algunos casos, vital. Son muchas las tecnologías las que toman participación en este concepto, y una de ellas es “Complex Event Processing” o “Procesamiento de Eventos Complejos” (CEP).

CEP, es una nueva forma de analizar datos. Una aplicación de Fast Data, que dota a los sistemas que lo soportan de una capacidad de sintésis, y análisis de datos abrumadora. Como toda tecnología tiene sus ventajas y desventajas, pero sin duda alguna, hemos de destacar que el procesamiento de evento complejos dota de una de las capacidades de análisis más rápidas del mercado, permitiendo la toma de decisiones a tiempo real.

El escenario en el que nos encontramos en este trabajo es el de las energías renovables, en concreto, la energía eólica. A rasgos generales, el funcionamiento de un molino de viento es muy sencillo, a través de la fuerza que el viento ejerce sobre sus palas se generá energía mecánica sobre el rotor, y este a su vez electricidad a través de un generador. Lo que intentaremos en este trabajo de fin de grado es aplicar CEP, para la optimización de obtención de energía eléctrica a través aerogeneradores.

En este trabajo de fin de grado se desarrolla un sistema de predicción del desgaste en aerogeneradores en base a los datos obtenidos por su sistema de monitorización. El objetivo del sistema a desarrollar es, utilizando el procesamiento de eventos complejos y tecnologías de código abierto, implementar un sistema que nos permita predecir cuando el comportamiento los aerogeneradores dentro de un parque eólico es anómalo. Por tanto, aquel aerogenerador que tenga un comportamiento poco común tendrá mas probabilidades de tener alguna avería, actualmente, o a corto-medio plazo.

Abstract

Internet of Things is a technological concept that is revolutionizing the way we conceive our day to day in recent years. The integration of computer systems in our homes, in our jobs, in the industry is very remarkable, and in some cases, vital. There are many technologies that participate in this concept, and one of them is the "Processing of Complex Events" or "Procesamiento de Eventos Complejos” (CEP).

CEP, is a new way to analyze data. An application of Fast Data, which endows the systems that support it with syntax capability, and overwhelming data analysis. As each technology has its advantages and disadvantages, but without any doubt, we must emphasize that complex event processing provides one of the fastest analysis capabilities of the market, allowing real-time decision making.

The scenario in which we find ourselves in this work is renewable energy, specifically, wind energy. In general terms, the operation of a windmill is very simple, through the force applied by the wind in its blades, mechanical energy is generated in the rotor, generating electricity through a generator. What we will try in this end-of-degree project is to apply CEP, for the optimization of obtaining electricity through wind turbines.

In this end-of-course project, a wind weathering prediction system is developed based on the data obtained by its monitoring system. The objective of the system is to develop, using the processing of complex events and open source technologies, implement a system that allows us to predict when the behavior of wind turbines within a wind farm is anomalous. Therefore, the wind turbine that has an unusual behavior is more likely to be damaged, currently, or a short-medium term.

Agradecimientos

Me gustaría dar las gracias en primer lugar a mis padres, por brindarme la oportunidad de poder estudiar la carrera que quería desde un primer momento. Por apoyarme día sí, día también desde que empecé mi camino como ingeniero informático y hacerme ver que no todo en la vida es ser bueno en tu profesión, sino también hay que ser buena persona, buen hijo y buen amigo.

A mi director de trabajo de fin de grado, Gregorio Diáz Descalzo y a Juan Boubeta Puig, por ayudarme en el desarrollo de este proyecto e introducirme en el mundo del Procesamiento de Eventos Complejos.

A mis hermanos y a mi novia, que han sabido reconducirme y tranquilizarme en todo momento, aguantando mis nervios.

Y por último a mis amigos por su incondicional y gran apoyo.

Índice de Contenido

Declaración de Autoría iii

Resumen v

Abstract vi

Agradecimientos ix

CAPÍTULO 1. Introducción 1

1.1. Motivación 1

1.2. Objetivos 2

1.3. Estructura de la memoria 3

CAPÍTULO 2. Antecedentes y estado de la cuestión 5

2.1. Fundamentos teóricos 6

2.1.1. Procesamiento de eventos complejos (CEP) 6

2.1.2. Arquitectura orientada a Servicios (SOA) 8

2.1.3. Arquitectura dirigida por eventos (EDA) 9

2.1.4. Arquitectura orientada a servicios dirigido por eventos (SOA 2.0) 9

2.1.5. ESPER EPL. Lenguaje de programación para CEP. 9

2.1.6. Energía Eólica 10

2.2. Estado del Arte 10

2.3. Discusión de metodologías 10

CAPÍTULO 3. ENTORNO DE TRABAJO 11

3.1. Medios Hardware. 11

3.2. Medios Software. 11

CAPÍTULO 4. Metodología y desarrollo 15

4.1. SPRINT 1. Introducción a cep. 15

4.2. SPRINT 2. Inicio del proyecto 15

4.2.1. Reunión 1. “Conocemos al cliente, primera toma con experto” 15

4.2.1.1 Información 15

4.2.1.2 Asuntos que tratar 15

4.2.1.3 Información recogida 16

4.2.1.4 Próximos pasos 19

4.2.2. Reunión 2. “Inicio del proyecto de trabajo de fin de grado” 20

4.2.2.1 Información. “Conocemos al cliente, primera toma con experto” 20

4.2.2.2 Asuntos que tratar 20

4.2.2.3 Información recogida 21

4.2.2.4 Próximos pasos 22

4.2.3. Reunión 3. “Dudas y problemas” 22

4.2.3.1 Información 22

4.2.3.2 Asuntos que tratar 23

4.2.3.3 Información recogida 23

4.2.3.4 Próximos pasos 24

4.2.4. Reunión 4. “Coordinación y análisis de la situación” 24

4.2.4.1 Información 24

4.2.4.2 Asuntos que tratar 25

4.2.4.3 Información recogida 26

4.2.4.4 Próximos pasos 28

4.2.5. Reunión 5. Clase tutorizada de energía eólica de Francisco José Polo 29

4.2.5.1 Información 29

4.3. Asuntos que tratar 29

4.3.1.1 Información recogida 29

4.3.1.2 Próximos pasos 33

4.3.2. Reunión 6. Reunión de revisión del sprint y retrospectiva del Sprint Meter aquí al Product Owner 33

4.3.2.1 Información 33

4.3.2.2 Asuntos que tratar 34

4.3.2.3 Información recogida 34

4.3.2.4 Próximos pasos 36

4.3.3. Retrospectiva del Sprint 37

4.4. SPRINT 2. Adaptación al caso real 38

4.4.1. Reunión 7. Reunión del comienzo del segundo 38

4.4.1.1 Información 38

4.4.1.2 Asuntos que tratar 39

4.4.1.3 Información recogida 39

4.4.1.4 Próximos pasos 41

4.4.2. Reunión 8. Reunión para explicación de 41

4.4.2.1 Información 41

4.4.2.2 Asuntos que tratar 41

4.4.2.3 Información recogida 42

4.4.2.4 Próximos pasos 46

4.4.3. Reunión 9. Reunión para concretar y validar los patrones con el cliente 47

4.4.3.1 Información 47

4.4.3.2 Asuntos que tratar 47

4.4.3.3 INformación recogida 47

4.4.3.4 Próximos pasos 49

4.4.4. Reunión 10. Reunión para revisión de código y comunicación de patrones definitivos 49

4.4.4.1 Información 49

4.4.4.2 Asuntos que tratar 50

4.4.4.3 Información recogida 50

4.4.4.4 Próximos pasoS 51

CAPÍTULO 5. Experimentos y resultados 53

CAPÍTULO 6. Conclusiones y propuestas 55

6.1. Conclusiones 55

6.2. Trabajo futuro 55

Bibliografía 57

Anexos 59

A.1. Ejemplo de uso de la herramienta 59

A.2. Manual de usuario 59

Índice de Ilustraciones

[Ilustración 1. Infraestructura del entorno de Eolic Event Consumer. 18](#_Toc508713945)

[Ilustración 2. Incorporación de CEP al sistema actual. 19](#_Toc508713946)

[Ilustración 3. Formaciones comunes y no comunes de un parque eólico. 21](#_Toc508713947)

[Ilustración 4. Jerarquía de factores. 22](file:///C:\Users\Kike%20Brazález\git\EolicSimulatorCEP\docs\TFG_CEP_Enrique%20Brazález.docx#_Toc508713948)

[Ilustración 5. Ejemplo de curva de potencia. 23](#_Toc508713949)

[Ilustración 6. Plantilla de enunciado de patrones. 27](#_Toc508713950)

[Ilustración 7. Grafíco de potencia producida el día 1 de enero. 35](file:///C:\Users\Kike%20Brazález\git\EolicSimulatorCEP\docs\TFG_CEP_Enrique%20Brazález.docx#_Toc508713951)

Índice de Tablas

[Tabla 1.Conclusiones de curso introductorio de energía eólica. 12](#_Toc508797126)

[Tabla 2. Factores enfrentados. 36](#_Toc508797127)

[Tabla 3. Mediciones rendimiento el día 1 de enero. 38](#_Toc508797128)

[Tabla 4. Medidas de productividad el día 1 de enero. 40](#_Toc508797129)

2. Introducción

“Internet of Things” (IoT), o “Internet de las cosas” en español, es el tema que ha inspirado este proyecto. Cómo poder comunicar todos los elementos que influyen en nuestro día a día puede parecer poco viable, debido a la gran inversión y trabajo que se necesita. No obstante, cada día que pasa dicho esfuerzo es menor [1]. Así, la evolución del IoT cambiará nuestro futuro. Nuestra forma de percibir la realidad cambiará totalmente y no sólo en nuestra vida cotidiana. si no que irá mucho más allá afectando a ámbitos como la educación, la comunicación, las empresas, la ciencia, y el gobierno. En este trabajo de fin de grado nos centraremos en el ámbito científico e industrial, en concreto el que afecta a las energías renovables.

* 1. Motivación

Este proyecto nace de la necesidad de optimizar la obtención de energía a un bajo coste y teniendo en cuenta la preservación del medio ambiente. Las fuentes de energía no renovables son aquellas que se encuentran de forma limitada en nuestro planeta dado que su consumo es mayor que su “regeneración” [2].

La integración de los sistemas de información en el sector industrial está al orden del día, es decir, la industria está vitalmente unida a los sistemas software en su enorme mayoría. Si la tecnología decae, cualquier actividad profesional relacionada está destinada al fracaso como es el caso de la generación de energía [3]. Existen multitud de aplicaciones software para controlar las herramientas de extracción de energía por medios renovables. Por ejemplo, en placas solares móviles podemos observar un sistema para ajustar la orientación con respecto al sol y optimizar su producción y, por otro lado, en el sistema de control de capacidad en un embalse podemos decidir cuándo nos conviene aprovechar la energía producida por su caudal.

A lo que energía renovable se refiere, hemos de destacar la energía eólica y la implicación que tienen los aerogeneradores en la misma. La integración del software y sistemas electrónicos en todos sus componentes tanto a nivel individual, así como, a nivel de parque eólico es abrumador. Un ejemplo destacable sería el flujo de información que se produce entre los aerogeneradores y las estaciones centrales. Sin embargo, todo ese volumen de información no se aprovecha nada más que para labores muy sencillas de mantenimiento y monitorización.

El objetivo de este proyecto es ir un paso más allá y en base a este gran volumen de información predecir el desgaste de un aerogenerador. Como dijo Bill Gates: “La información es poder”. Llevando esta frase a la práctica. Si tenemos toda la infraestructura de sensorización, ¿por qué no aprovechar esos grandes volúmenes de datos no solamente para labores de control y monitorización sino también para realizar predicciones?

* 1. Objetivos

Fruto de la motivación para la elaboración de este proyecto surgen los siguientes objetivos, aunque hemos de destacar en concreto el principal: “Implementar un sistema experto que nos permita predecir los eventos relacionados con el desgaste de un aerogenerador en base a los datos obtenidos por sus sistemas de monitorización”.

Para considerar como satisfecho este objetivo hemos de alcanzar las siguientes metas a, que serían:

1. Estudio de energía eólica y familiarizarnos con el entorno de los aerogeneradores, tanto desde un punto de vista técnico, como desde uno más general.
2. Controlar el “Enterprise Service Bus” de Mule, para poder analizar el flujo de datos.
3. Familiarizarnos con el lenguaje para implementar los patrones en Esper EPL.
4. Detectar qué eventos son de interés en las medidas dadas en un aerogenerador para implementar los patrones adecuados.
5. Desarrollo de una aplicación gráfica que nos permita visualizar todos los eventos simples, y complejos fruto de los patrones implementados.
6. Generar alertas en base a los patrones implementados.
   1. Estructura de la memoria

El esqueleto interno de esta memoria por capítulos es el siguiente:

**CAPÍTULO 1. Introducción**

Se trata del capítulo donde nos encontramos, y en el que se han desarrollado temas que sirven como introducción para el correcto entendimiento de los conceptos de este proyecto.

**CAPÍTULO 2. Antecedentes y estado de la cuestión**

En esta sección veremos reflejados todos y cada uno de los fundamentos teóricos en los que se basa “Eolic Event Consumer”. Se definirá el funcionamiento del Procesamiento de Eventos Complejos y la energía eólica. Terminando con el estado del arte, donde se analizan las tecnologías existentes, revelando sus ventajas y desventajas, en busca de la mejor o la más viable para la solución del problema.

**CAPÍTULO 3. Metodología y desarrollo**

En este capítulo se detalla la metodología utilizada, SCRUM mezclado con prototipado, haciendo una pequeña comparación y discusión de porqué hemos decidido utilizar esta metodología ágil. En la segunda parte del capítulo, veremos reflejados el trabajo realizado sprint a sprint, con toda la información detallada, desde varios diagramas UML hasta lo recogido en las reuniones con el cliente y el equipo.

**CAPÍTULO 4. Experimentos y resultados**

En este capítulo quedan reflejadas las pruebas llevadas a cabo en el sistema, siendo contrastadas con el modelo de calidad ISO-IEC 9126. Describiendo algunos de los aspectos visuales y técnicos de la aplicación gráfica, y de la aplicación CEP que trabaja por debajo.

**CAPÍTULO 5. Conclusiones y propuestas**

Por último, se expondrán una serie de conclusiones, culminando con el planteamiento de futuros trabajos que pueden realizarse utilizando como punto de partida “Eolic Event Consumer”.

2. Antecedentes y estado de la cuestión

Hoy en día la incorporación del análisis de datos en los sistemas informáticos es un elemento indispensable para su éxito. Con el transcurso del tiempo, el anhelo de conocimiento de ingenieros y tecnólogos produce que la evolución en la industria crezca cada día mas. Ultimamente, estos avances están relacionados con el desarrollo de sistemas que pretenden optimizar la productividad y eficiencia de cualquier actividad, ya bien a sea en nuestros hogares, a través de domótica, o a nivel comercial, a través del análisis de la tendencia en el mercado utilizando técnicas de Machine Learning.

Ese incremento en la productividad y eficiencia, con respecto la competencia, puede significar que nos posicionemos como pioneros en nuestra actividad profesional. Y esto también tiene su lugar en el contexto que nos encontramos, la energía eólica. Optimizar la producción energía frente al resto de multinacionales puede significar que nos coloquemos como cabeza en la lucha por ser líderes de producción energética renovable.

La incorporación de Internet of Things en los parques eólicos puede significar un cambio en el modo de producir energía eléctrica. Los sistemas de monitorización de los aerogeneradores producen una ingesta cantidad de información, de ahí que actualmente se intente utilizar toda esa cantidad de datos, para hacer análisis predictivo y optimizar la producción de energía.

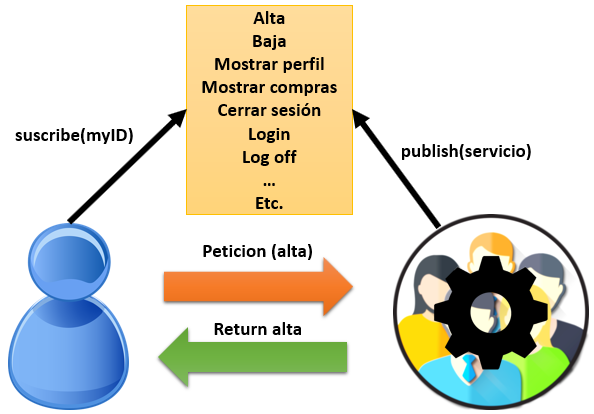
Cada vez son más las productoras de energía renovables como Endesa y Acciona, que integran en sus aerogeneradores estos sistemas predictivos, para poder optimizar su actividad. Cabe destacar en este campo a uno de los fabricantes más famosos de aerogeneradores, Gamesa, que el 29 de octubre de 2008[4], que consolidó una tecnología propia para el mantenimiento predictivo mediante el análisis de vibraciones a través del uso de acelerómetros para detectar el desgaste en los elementos mecánicos del aerogenerador. De ahí que más adelante estudiemos la implantación del procesamiento de los eventos complejos, en una arquitectura orientada a servicios, pero dirigida por eventos.

Volviendo al hilo principal, dividimos este capítulo en dos. Por un lado, tenemos la definición y desarrollo de los fundamentos teóricos de los que nace “Eolic Event Consumer”, como herramienta para el mantenimiento predictivo, y, por otro lado, un análisis comparativo de las posibles soluciones al problema planteado.

* 1. Fundamentos teóricos.

En este punto veremos todos y cada de los puntos teóricos que rodean este trabajo de fin de grado, tanto a nivel de arquitectura, como de software, como a nivel de contexto, puesto que nos encontramos dentro del escenario de la energía eólica.

* + 1. Arquitectura orientada a Servicios (SOA)

de La arquitectura orientada a servicios es un estándar basado en componentes que relaciona entre si diferentes funcionalidades de un nodo, estas funcionalidades se denominan “servicios”[5], por medio de tecnologías e interfaces que facilitan la interacción con el usuario.

Un servicio es un elemento reutilizable que puede ser parte o todo de un proceso de negocio, a mayor o menor escala. Un servicio puede ser una simple petición GET a un servidor para obtener el saldo que nos queda en nuestro banco. Además, puede servir como parte en un sistema completo de logeo en una página web. Hemos de añadir que a lo mujer esa componente del sistema de logeo por sí sola no aporta ninguna funcionalidad, pero en conjunto con más servicios sí que lo hace. Tal y como vemos en la Ilustración 1, normalmente las arquitecturas orientadas a servicios se implantan a través de protocolos publish/suscribe, donde el usuario se suscribe al proveedor de servicios para solicitar esas funcionalidades.

Ilustración 1. Funcionamiento de SOA a través de publish/suscribe.

Una parte imprescindible para el buen funcionamiento de una arquitectura orientada a servicios es la interfaz, ha de definirse de forma neutral, independiente a lo que trabaja por debajo y proporciona el servicio. Esto desencadena que estas funcionalidades puedan interactuar correctamente con el usuario de forma fácil, sencilla, uniforme y universal sea cual sea su perfil.

La arquitectura orientada a servicios [6] permite que los componentes estén débilmente acoplados, reutilizando los servicios entre sí optimizando su funcionamiento y compartiendo los componentes internas de la arquitectura con todos los usuarios. Las características facilitan este débil acoplamiento son:

* Sigue estándares abiertos que permiten la fácil compatibilidad.
* Aseguran la compatibilidad con multitud de tecnologías.
* Separa el código de la interacción del usuario a través de interfaces.
* Modular. Puede ejecutarse como parte o todo de un proceso de negocio, y pueden ser remplazables.
* Sin estado. El servicio no mantiene el estado entre invocaciones, bajo la petición de un servicio se toma el parámetro introducido y se invoca aisladamente la función correspondiente.
* Comunicación. Permite la comunicación entre nodos distintos que proveen de servicios externos.
* Ubicación. El usuario descnoce la ubicación real de la máquina que ofrece el servicio.
  + 1. Arquitectura dirigida por eventos (EDA).

Según dos de los pioneros en lo que al procesamiento de eventos complejos se refiere, David Luckham y W. Roy Schulte, un evento es un suceso que está ha llamado a suceder, [9] y que tiene una relevancia en un contexto concreto. Por ejemplo, la notificación cada diez minutos de un sensor de temperatura, o la llegada de un correo electrónico a nuestra bandeja de entrada. Por tanto, si aplicamos su definición de evento para describir EDA, podemos decir se trata de un estilo de arquitectura cuyos componentes son los eventos, dirigidos y comunicados por ellos mismos.

Este patrón arquitectural permite la producción de eventos, monitorización, así como su visualización y análisis [8]. Tenemos cuatro elementos que son claves para el funcionamiento normal de este tipo de arquitectura:

* **Productor de eventos.** Es aquel componente que tiene como misión generar los eventos a raíz de lo que ocurre en su entorno. Si lo aplicamos a “Eolic Event Consumer”, diremos que los productores de eventos son los aerogeneradores, mandando cada diez minutos un registro con todo lo que ha captado su sistema de monitorización.
* **Consumidores de eventos.** Un componente que consume la información generada por el productor para efectuar la toma de decisiones. En el caso de este trabajo de fin de grado, podríamos decir que es el motor inteligente que procesará toda esa información utilizando los patrones que nosotros como desarrolladores elaboraremos.
* **Canal.** Es el medio por el que se transmiten los eventos.
* **Acciones.** Se trata de las acciones reales que se llevan a cabo, fruto del análisis ofrecido por el consumidor de eventos.
  + 1. Procesamiento de eventos complejos (CEP)

Dando respuesta a la pregunta lanzada en la motivación: “¿por qué no aprovechar esos grandes volúmenes de datos no solamente para labores de control y monitorización sino también para realizar predicciones?” Surge como posible solución el Procesamiento de Eventos Complejos (tecnología CEP) como aplicación de “Fast Data”. La mejor forma de tomar decisiones en estos ámbitos es hacerlo cuanto antes, de ahí que se examine toda la información a tiempo real para de esta forma dirigir satisfactoriamente la toma de decisiones o acciones correctivas en el ámbito industrial. Hay que saber distinguir que se habla de “Fast Data” y no de “Big Data” [10]. ¿Por qué? Fast Data tiene una ventaja, y es que no almacena ningún tipo de información. Conforme los datos son generados, una vez procesados por primera vez, se desechan, es decir, se analizan conforme van siendo generados para después desecharlos, lo que lo hace muy rentable. Si se almacenara hablaríamos de “Big Data”. Por lo tanto, como se genera, se procesa, y se desecha sin almacenarla hablamos de “Fast Data”, evitándonos el gran coste que llevaría salvaguardar tanta información.

La ventaja del procesamiento de eventos complejos tal y como Fast Data determina, es que puede dominar cualquier volumen de datos, a una velocidad en tiempo real y de orígenes muy diversos. CEP, es una tecnología emergente capaz de monitorizar múltiples flujos de eventos, procesarlos mediante la determinación de patrones implementados en Esper Event Processing Language (EPL) [7], y así tomar decisiones tanto predictivas como correctivas en base a la información que el contexto nos aporta.

Como conclusión a la problemática generada por la necesidad de optimizar la producción en un parque eólico, este trabajo de fin de grado tiene como meta la predicción del desgaste de aerogeneradores a través de un sistema SOA 2.0. SOA 2.0, es una arquitectura orientada a servicios, aunque dirigida por eventos [11], en la sección XXX se explicarán como es su funcionamiento.

Cabe destacar que CEP [12] tiene varios aspectos comúnes con una arquitectura basada en servicios, donde el usuario sabe cual es el servicio para solicitar, sin embargo, en el contexto de CEP, son los productores de eventos los que manda las peticiones a los consumidores (usuarios) para que procesen toda esa información. Podemos decir que la arquitectura donde se implantan se implanta la tecnología basada en el procesamiento de eventos complejos es reactiva (bajo la ocurrencia de situaciones de interés) y desacoplada (al poder haber más de un único productor de eventos).

Debido a las pequeñas diferencias y similitudes con SOA, podemos concluir que SOA 2.0 nace como fruto de la combinación con una arquitectura dirigida por eventos, EDA, ajustándonos a ese funcionamiento reactivo y desacoplado.

Si analizamos el funcionamiento de CEP [13], podemos enumerar diferentes características como son:

* **Comunicación múltiple**. Por lo general los productores de eventos envían la información que es de interés a cada uno de los consumidores, que se han suscrito a ese productor.
* **Inmediatez.** La comunicación de los eventos es a tiempo real en el momento en el que se detecta alguna situación de interés.
* **Asincronismo.** El sistema que publica un evento no espera a que los sistemas que lo reciben lo procesen. Publica y continúa.
* **Eventos de grano fino.** Todo el análisis surge a raíz de eventos simples con datos de grano fino, pudiendo así, simplificar el flujo de información y facilitar en análisis y la toma de decisiones en base a la información obtenida. Cuanto más se simplifique el flujo de eventos, mejor
* **Nomenclatura y categorización.** Se puede presentar la jerarquización de los eventos permitiendo la discretización de la información, o la clasificación de eventos en base a su relevancia o lecturas.
* **Procesamiento de Eventos Complejos.** El sistema interpretará y monitorizará las relaciones entre los eventos. Por ejemplo, la agregación de eventos (un patrón de eventos implica un evento de mayor nivel) o relación de causalidad (un evento es originado por un evento previo).
  + 1. Arquitectura orientada a servicios dirigido por eventos (SOA 2.0)
    2. ESPER EPL. Lenguaje de programación para CEP.

CEP and event streams almost inevitably go together; CEP can in this case be viewed as a transforming agent, taking as input a stream of lower-level events and emitting as output either a filtered set of these events or aggregated higher-level events. Paper [18] deals with high-performance CEP over streams as related to RFID readings coded as events. It introduces the SASE (apparently not an acronym) event matching language, which structures queries in the form: EVENT [WHERE ] [WITHIN ] The is a major event type (e.g., a book being scanned), the can filter on event object properties, and the is a temporal constraint. SASE is interesting as an earlier (2006) pioneering language that improved on existing languages due to its query format which was highly optimizable due to similarity to relational database query formats, compared to existing products such as TelegraphCQ. However, it has no aggregate operators and is improved upon by SASE+ or NFAb -based systems.

* + 1. Energía Eólica
  1. Estado del Arte

Discusión de porque utilizamos CEP contra otro tipo de tecnologías.

Porque elegimos el ESB de MuleSoft

Porque elegimos Medit4CEP y Eclipse.

* 1. Discusión de metodologías

1. ENTORNO DE TRABAJO

En este apartado hablaremos de todos los medios tanto software como hardware tenemos pensado utilizar para llevar a cabo el proyecto en primera instancia.

* 1. Medios Hardware.

Los medios hardware que utilizaremos para llevar a cabo el proyecto serán los siguientes:

* **Ordenador personal.** Se trata de un ordenador ASUS bastante común, utilizado durante el transcurso de la carrera. Procesador i7-5610QM CPU de 2,30 GHz y memoria RAM de 6 GB, con un tipo de sistema de 64 bits y con un sistema operativo Windows 10.
* **PLC de aerogenerador.** Se trata del PLC que controla todas las mediciones de las centralitas PT100 que nutren el sistema de monitorización del aerogenerador. Esta será la fuente de datos del sistema, aunque en primer lugar trataremos un conjunto de datos sintético en formato Excell o .db con el que simularemos el flujo de información, debido a que no tenemos acceso directo a ese PLC.
  1. Medios Software.

En esta sección veremos cuáles son los elementos software que tomarán participación en nuestro entorno de trabajo. Desde los programas, IDEs y demás, hasta las librerías particulares que hacen singular este proyecto.

* **Eclipse.** Eclipse es un entorno de desarrollo integrado de Java (IDE), con gran cantidad de plugins de Open Source con lenguajes integrados como C / C++ y PHP. Puede combinar fácilmente el soporte de idiomas y otras funciones en cualquiera de los paquetes que la nube de Eclipse facilita en los paquetes predeterminados, y Eclipse Marketplace permite una personalización y extensión prácticamente ilimitadas. Hemos elegido Eclipse y no otro entorno porque el ESB está implementado sobre Eclipse, de esta forma podemos unificar paralelamente todo el código, tanto el del simulador como el del Enterprise Service Bus [14].
* **EGit**. EGit es un proveedor de Eclipse Team para el sistema de control de versiones de Git. Git es un SCM distribuido, lo que significa que cada desarrollador tiene una copia completa de todo el historial de cada revisión del código, lo que hace que las consultas contra el historial sean muy rápidas y versátiles. El proyecto EGit está implementando herramientas Eclipse además de la implementación JGit Java de Git [15].
* **Team Foundation Service.** Se trata de un producto Software que nos permitirá llevar a cabo la gestión del trabajo. En concreto la enumeración de historias de usuario, de tareas, y generación de artefactos como sería el Sprint Burndown.
* **Anypoint Studio, ESB de Mulesoft**. Es el entorno de desarrollo de integración basado en Eclipse de MuleSoft para diseñar y probar aplicaciones Mule. Puede implementar la aplicación y ejecutarla en su servidor Mule. El mismo editor también le permite editar archivos de definición de API (en RAML y WSDL) y crear dominios que definan recursos compartidos. Anypoint Studio ofrece dos pestañas paralelas que puede utilizar para diseñar y crear sus aplicaciones: Editor visual; y un editor XML [16]**.**
* **Librería poi-3.11 para el tratamiento de documentos Excel.** Se trata de una librería de Apache para la manipulación de documentos Excel que en un principio se generarían por el PLC del aerogenerador y nutrirían al motor CEP.
* **Librería de Esper EPL.** Esper ofrece un lenguaje específico de dominio (DSL) para el procesamiento de eventos. El lenguaje de procesamiento de eventos (EPL) es un lenguaje declarativo para tratar datos de eventos basados ​​en el tiempo de alta frecuencia. EPL cumple con el estándar SQL-92 y se amplía para analizar series de eventos y con respecto al tiempo [17].
* **Medit4CEP.**  Es una herramienta para la generación de código Esper EPL, a través de un editor gráfico. Esta aplicación sirve es una solución para acercar la tecnología de procesamiento de eventos complejos a cualquier usuario independientemente del nivel de programación que posea, para que, de esta forma, se facilite la toma de decisiones [18].

1. Metodología y desarrollo
   1. SPRINT 1. Introducción a cep.
   2. SPRINT 2. Inicio del proyecto

En esta iteración se empezó

* + 1. Reunión 1. “Conocemos al cliente, primera toma con experto”
       1. Información

En esta reunión tomaremos un primer contacto oficial con el cliente, “Ingeteam”.

* Dirección: Parque Científico y Tecnológico, Paseo de la Innovación,3, 02006 Albacete. Oficinas de Ingeteam S.A.
* Hora y fecha de la reunión. A las 8:15 el día 8 de enero de 2018.

Los asistentes de la reunión son:

* Vicente Requena Montejano. Ingeniero Industrial. Especializado en aerogeneradores.
* Antonio Fernández Diez. Ingeniero Industrial. Especializado en placas fotovoltaicas.
* Enrique Brazález Segovia. Desarrollador y encargado de defender este trabajo de fin de grado.
  + - 1. Asuntos que tratar

Debido a que ninguno de los asistentes es Pedro Salazar, enlace directo con la empresa, tendremos que explicar cuáles son los objetivos de nuestro proyecto, cuál es la tecnología que utilizaremos y la capacidad de análisis que tiene el procesamiento de eventos complejos. Los temas que se tomaron en la reunión fueron:

* Presentación y recepción de contactos.
* Situación actual del contexto.
* Clase introductoria de energía eólica.
* ¿Qué tipo de aerogenerador vamos a analizar?
  + - 1. Información recogida

La situación con el cliente en un primer momento es buena. Los asistentes conversan y tratan los problemas sin ningún problema. A priori no parece que vayan a haber conflictos a lo largo del ciclo de vida del proyecto. En primer lugar, el contexto del proyecto se ha concretado en el estudio de un aerogenerador estándar de 1500 W. El acceso a los datos obtenidos por los PLCs de control del sistema de monitorización es algo complicado debido a que los clientes no quieren en un primer momento proporcionar sus datos. Sin embargo, en Burgos hay un parque cuyo servicio es dado por Ingeteam, y sus datos sí que son accesibles. A continuación, se da una clase teórica para adentrarnos en el mundo de la energía eólica, los datos más representativos son los recogidos en Tabla 1.

Tabla 1.Conclusiones de curso introductorio de energía eólica.

|  |  |  |
| --- | --- | --- |
| ***Nombre*** | ***Descripción*** | ***Trababjamos con*** |
| Potencia nominal | Se trata de la potencia sacada por el aerogenerador en su rendimiento más alto | 1500 W |
| Rango de viento |  | 4 -25 m/s |
| Tensión | La tensión que sale del generador para la estación distribuidora | 690 V – 12000 V |
| Giro del rotor | El giro del rotor | 10 – 30 rpm |
| Palas | Son 3 palas que van acopladas al eje central, dos posiciones extremas. Atención los grados son respecto el eje central de la junta de las palas “CONO”. A más palas más coste, 3 es lo óptimo en cuanto a relación dinero/optimización | 2 posiciones  0º Barlovento (máximo rendimiento)  90 º Bandera (mínimo rendimiento) |
| Nacelle | Góndola | - |
| Rotor | Cono y palas | - |
| Pitch | Giro de las aspas sobre el cono para cambiar su grado (de 0 a 90) | Es un pitch eléctrico |
| Multiplicadora | Al ser un aerogenerador con dipolo la velocidad del rotor es muchísimo menor a la que le hace falta al generador para producir energía eléctrica de forma que necesitamos una multiplicadora para adecuar ambas velocidades a la de sincronismo. 1 etapa planetaria y 2 paralelas | Sí, trabajamos con un aerogenerador con multiplicadora eléctrica. |
| Etapa planetaria | Comunica directamente la parte acoplada al rotor y transmite la velocidad de este a la fase paralela | ATENCIÓN LA LUBRICACIÓN ES IMPORTANTE, debido a la generación de partículas que producen desgaste. |
| Etapa paralela | Conjunto de engranajes para derivar una velocidad baja transmitida por la fase planetaria al generador con la velocidad de sincronismo adecuada. | ATENCIÓN LA LUBRICACIÓN ES IMPORTANTE, debido a la generación de partículas que producen desgaste. |
| Lubricación y refrigeración en multiplicadora | Es importante debido a que SIEMPRE ha de estar a una temperatura y nivel de aceite adecuado. La temperatura es controlable, pero ES INEVITABLE el cambio de aceite debido a su gran uso | NO SABEMOS RANGOS  ¿Filtros se desgastan?  ¿Resistencia calefactora peta?  Prestar atención en:   * Rodamientos. * Engranajes. * Resistencia calefactora. * PT100 – Sensor térmico del sistema de lubricación. |
| Junta rotativa | Se trata de una junta rotativa que comunica la multiplicadora con el generador ya que uno al estar estático y otro que es móvil entonces debemos de incorporar para impedir que se líen cables | Puede haber más de una junta rotativa |
| Generador doble alimentado | Se trata del que hace la conversión y genera la energía en watios que se mandará a la estación eléctrica. Incorpora otro PT100. Circuito de producción para la red (el normal), otro circuito de alimentación que alimenta al generador para producir energía en caso de haber poco viento de esta forma siempre se podrá generar energía puesto que se ayuda al rotor a girar. | Con una tensión que se va a la estación de 690 V de forma que hay que subirla para que no haya pérdidas  1500 rpm  10 % deslizamiento  2 – 4 pares de polos.  Tener en cuenta:   * PT100. * Cuerpo de anillos. * ¿Refrigeración aire o agua? |
| Polipasto | Grúa de ayuda para llevar herramienta pesada en pequeñas reparaciones. |  |
| Sistema del YAW | El YAW es el giro de la góndola en 360º con respecto el eje vertical del aerogenerador. | Es un sistema de YAW eléctrico. |
| Freno de fricción | Frena el movimiento de YAW en algún momento. | ¿Calor? ¿Número de veces utilizado? |
| Dispositivo cuentavueltas | Cuentavueltas de góndola para evitar enrollamiento de los cables de potencia. | Máximo dos vueltas, se tiende a desenrollar tras las dos o para optimizar a desenrollar cuando no tiene un buen rendimiento el aerogenerador. |
| Grupo hidráulico | Controla toda la hidráulica del rotor y las palas | Control de aceite.  Acumulador de nitrógeno.  Preostatos.  PT100. |
| Armario TOP | Donde se ubica toda la lógica software del sistema, desde el movimiento del YAW, los controles de las temperaturas y sistemas hidráulicos hasta las rotaciones del PITCH. |  |
| Armario GROUND | Controla toda la potencia y las funcionalidades de mantenimiento y demás de los aerogeneradores. |  |
| Transformador | Elevación de la tensión de 690 V obtenidos por el generador hasta elevarlo al voltaje requerido para que no haya perdida de energía. |  |
| Celda de maniobra | Para actividades de maniobra o corte de tensión del aerogenerador. |  |

* + - 1. Próximos pasos

En esta reunión se han acordado las siguientes tareas:

* Petición de datos sintéticos al parque eólico de Arroyal, Burgos. El encargado de recoger esta información es Vicente, para después proporcionar los datos al desarrollador, Enrique.
* Elaboración de modelo de datos a partir de los datos sintéticos obtenidos por Vicente, para comenzar con el primer prototipo de la iteración primera del proyecto y estudio de los conocimientos aportados por la clase introductoria de energía eólica.
* Antonio deberá de hablar con Pedro para coordinar al desarrollador, y fijar el encargado de controlar el progreso de las reuniones.
  + 1. Reunión 2. “Inicio del proyecto de trabajo de fin de grado”
       1. Información. “Conocemos al cliente, primera toma con experto”

En esta reunión comenzaremos la iteración 1 “Inicio del proyecto”.

* Dirección. Edificio Infante Don Juan Manuel. Avda. de España s/n. Albacete. Despacho 0. B.8.
* Hora y fecha de la reunión. A las 18:00 el día 11 de enero de 2018.

Los asistentes de la reunión son:

* Gregorio Diaz Descalzo. Jefe de proyecto de este trabajo de fin de grado. Titular de la Universidad de Castilla-La Mancha.
* Enrique Brazález Segovia. Desarrollador y encargado de defender este trabajo de fin de grado.
* Fernando Luján Martínez. Desarrollador de trabajo de fin de grado paralelo, su tutor también es Gregorio.
  + - 1. Asuntos que tratar

En resumen, esta reunión es para atender aspectos relacionados con la elaboración del trabajo de fin de grado de cara a la universidad. Es decir, de cara a la presentación del anteproyecto y los criterios de la convocatoria. Los temas que se trataron fueron:

* Criterios de evaluación de la convocatoria.
* Aclaración de las fechas de las que disponemos.
* Explicación de cómo hacer el anteproyecto.
* Definir planificación del proyecto.
* ¿Qué metodología vamos a utilizar? ¿Qué iteraciones nos vamos a marcar?
* Explicación de sistema de citas.
  + - 1. Información recogida

Se repasaron con atención todos y cada uno de los criterios para la evaluación del trabajo de fin de grado, desde su evaluación como a los requisitos mínimos que debería de tener. En concreto se repasaron todos y cada uno de los capítulos de la memoria del trabajo de fin de grado para que no haya dudas en su redacción.

Hablando de las fechas nos imponemos como fecha límite para tenerlo todo terminado, el día 17 de junio de 2018, a efectos de la presentación para la defensa que va desde el día 25 al 29 de junio de 2018. Hay que destacar que el día 18 de junio se ha de hacer el depósito, de ahí que tengamos que tener todo preparado para el día 17. Utilizaremos SCRUM combinándolo con prototipado, de forma que tendremos 4 iteraciones y al final de cada iteración tendremos un prototipo listo y funcionando.

En cuanto al anteproyecto, hemos discutido sobre todo los aspectos referentes a la justificación de las competencias. En concreto hemos tomado las siguientes notas, tanto Fernando como yo:

* **Primera competencia.** Aportar que vamos a usar una metodología ágil. El uso de la tecnología CEP de forma eficiente y con pruebas de la evaluación (pruebas de carga y estrés). Aportar que es barato puesto que no necesitamos ningún tipo de base de datos para guardar la información.
* **Segunda competencia.** Se van a recoger todos y cada uno de los requisitos necesarios para conseguir que el proyecto sea aceptable y cumpla con todo lo impuesto por el cliente.
* **Tercera competencia.** Descripción de las metodologías y herramientas que vamos a utilizar.
* **Cuarta competencia.** Discusión y debate acerca qué tecnologías utilizar y porqué CEP es la más viable.
* **Quinta competencia.** Realizar un plan de gestión de riesgos tal y como hemos visto en Gestión de Proyectos Software.
* **Sexta competencia.** Control de recursos humanos, fallos en las herramientas, contacto continuo con el cliente.

Por último, en la última parte de la reunión se hace muchísimo hincapié en la importancia de la inclusión de bibliografía debido a que es una de las partes clave del trabajo de fin de grado. De esta forma, no tendremos problemas en aspectos referentes a la propiedad intelectual.

* + - 1. Próximos pasos

En esta reunión se han acordado las siguientes tareas:

* El anteproyecto ha de estar hecho a primeros de febrero.
* Establecer un primer flujo de datos en esta primera iteración que coja los datos a través de “Thingspeak” con un pequeño muestreo de datos. No hace falta con el modelo de datos real, aunque sí se puede mejor.
* Utilizar algún control de versiones como podría ser GitHub. Como utilizaremos Eclipse, se toma la decisión de acoplarlo todo utilizando EGit, que es la distribución de Git para el IDE de Eclipse.
* Utilizar algún programa tipo “Kunagi” para llevar la gestión de las tareas e historias de usuario de los sprints.
* Hacer una planificación inicial con todos los hitos que tendría el proyecto, a nivel de reuniones, fechas de fin de iteración, y demás.
  + 1. Reunión 3. “Dudas y problemas”
       1. Información

En esta reunión se resolvieron dudas del anteproyecto y ciertos problemas de colaboración por parte del cliente.

* Dirección. Edificio Infante Don Juan Manuel. Avda. de España s/n. Albacete. Despacho 0. B.8.
* Hora y fecha de la reunión. A las 16:00 el día 23 de enero de 2018.

Los asistentes de la reunión son:

* Gregorio Diaz Descalzo. Jefe de proyecto de este trabajo de fin de grado. Titular de la Universidad de Castilla-La Mancha.
* Enrique Brazález Segovia. Desarrollador y encargado de defender este trabajo de fin de grado.
* Fernando Luján Martínez. Desarrollador de trabajo de fin de grado paralelo, su tutor también es Gregorio.
  + - 1. Asuntos que tratar

En resumen, esta reunión sirve como recursos para atender ciertos problemas a la hora de hacer el anteproyecto, y aspectos referentes a la poca colaboración del cliente en el proyecto. Los temas que se trataron fueron:

* Dudas a la hora de realizar el anteproyecto, posibles restricciones, aspectos positivos, aspectos negativos.
* ¿Dónde está Pedro?
* Poco compromiso por parte del cliente.
* No sabemos de qué información bebe el motor de datos, hay que decidir cómo se nutre al motor que procesa los eventos.
* Recogida de requisitos.
  + - 1. Información recogida

En esta reunión se vieron las dudas que hubo a la hora de llevar a cabo el anteproyecto, sirvió para que Gregorio diera el visto bueno ya que el enunciado de las competencias es muy confuso. Tanto Fernando como yo, necesitábamos realizar esta prueba de aceptación para cerciorarnos de que íbamos por el buen camino.

Pedro fue nuestro primer enlace con Ingeteam para llevar a cabo este proyecto, sin embargo, por circunstancias que desconocemos se fue de la empresa sin comentarnos la situación. Un error por su parte al irse habiéndose comprometido con el proyecto, y una falta de profesionalidad bastante grande puesto que no se ha pronunciado al respecto. Sin embargo, se acordó establecer una reunión con Antonio y Vicente la semana próxima para solucionar todos los problemas de coordinación y enmarcar los límites de nuestro proyecto. De esta forma, fijaríamos qué debe de incluir nuestro proyecto y no desarrollar ni de menos ni de más.

La problemática de la entrada de datos era muy grande, debido a que no teníamos acceso directo al PLC. Por lo tanto, tomamos la decisión de que se simularía a través de un servicio FTP, donde se colocarían archivos de un formato X (.db, csv, o Excel) en un directorio y el Enterprise Service Bus se encargaría de procesar para detectar los eventos. Sin embargo, debido a una falta de tener un coordinador dentro de la empresa, no sabíamos en qué formato tomar los datos. En base a la facilidad de procesamiento, utilizaríamos archivos en formato csv, obviando las necesidades del cliente debido a su desinterés.

Finalmente, decidimos tomar los requisitos en formato de storyboards, y en lenguaje natural aquellos aspectos que no se pudieran llevar a cabo con esta técnica. Deberemos de enunciar todos los patrones implementados en EPL, a través de la herramienta Medit4CEP, ya que es mucho más visual que el propio código. Es importante que sepamos si el cliente tiene alguna preferencia en cuanto al lenguaje utilizado para desarrollar la aplicación gráfica.

* + - 1. Próximos pasos

En esta reunión se han acordado las siguientes tareas:

* Enrique debe de contactar con la empresa y hacer que alguien, por ejemplo, Vicente o Antonio se haga cargo de la coordinación del proyecto dentro de la empresa. Para ello a la semana siguiente a esta reunión se ha de concertar una reunión para hablar estos aspectos.
* Terminar el anteproyecto para repasarlo con Gregorio, y dar últimas guías para la redacción tanto del mismo como el de la memoria final del trabajo de fin de grado.
* Preparar storyboards.
* Preparar el plan de gestión de riesgos.
* Documentar todos los fallos que vayamos teniendo a lo largo de todo el proyecto.
  + 1. Reunión 4. “Coordinación y análisis de la situación”
       1. Información

En esta reunión sirvió en su mayor parte para dar solución al problema de una coordinación inexistente por parte del cliente, y volver explicar la mecánica de la tecnología CEP al cliente.

* Dirección: Parque Científico y Tecnológico, Paseo de la Innovación, 3, 02006 Albacete. Oficinas de Ingeteam S.A.
* Hora y fecha de la reunión. A las 16:30, el día 1 de febrero de 2018.

Los asistentes de la reunión son:

* Vicente Requena Montejano. Ingeniero Industrial. Especializado en aerogeneradores.
* Francisco José Polo Sánchez. Jefe de departamento de I+D+i de Ingeteam Service.
* Antonio Fernández Diez. Ingeniero Industrial. Especializado en placas fotovoltaicas.
* Gregorio Diaz Descalzo. Jefe de proyecto de este trabajo de fin de grado. Titular de la Universidad de Castilla-La Mancha.
* Enrique Brazález Segovia. Desarrollador y encargado de defender este trabajo de fin de grado.
* Fernando Luján Martínez. Desarrollador de trabajo de fin de grado paralelo, su tutor también es Gregorio.
  + - 1. Asuntos que tratar

En resumen, esta reunión sirve como recursos para volver a explicar la mecánica de la tecnología CEP debido a que Pedro se fue, y hacer hincapié donde tendría cabida la aplicación, bien dentro del sistema de Ingeteam o a parte. Los temas que se trataron fueron:

* ¿Quién es el encargado de coordinar el proyecto como Pedro se fue?
* Explicar otra vez cómo es la mecánica de la tecnología CEP.
* ¿Dónde vamos a integrar la aplicación?
* ¿Cómo tenemos que mostrar la ocurrencia de los eventos complejos?
* ¿Cuál es la nueva forma de trabajar después de irse Pedro?
* Ingeboards y SCADA.
  + - 1. Información recogida

El encargado debido a la ausencia de Pedro en la empresa será Francisco José Polo. Se explicó cómo funcionaba la tecnología CEP haciendo mucho hincapié en que los desarrolladores no iban a hacer labores de estadísticos para descubrir por ellos mismos cuales son las causas por las que se desgastan los aerogeneradores. Los patrones han de ser implementados en base el enunciado de estos por el cliente en el formato: “Quiero detectar X, afectan Y, W, …, t variables y cada Z de tiempo.”

Se introdujeron dos conceptos nuevos que por parte de los desarrolladores y el jefe de proyecto eran desconocidos. Hablamos de Ingeboards y SCADA. Tal y como vemos en la Ilustración 1, observamos dos sistemas o estructuras, que colaboran entre sí. Por un lado, tenemos SCADA, que es el sistema que tiene almacenados todos los datos recogidos por los sistemas de monitorización de los aerogeneradores, además, de los logs de averías, y, por otro lado, tenemos Ingeboards, plataforma donde se visualiza la información dada por SCADA de una forma más legible. A través de Ingeboards, los técnicos encargados de las reparaciones de los aerogeneradores mandarán al sistema los informes técnicos que han llevado a cabo una vez terminadas sus reparaciones. Gracias a Ingeboards se puede determinar el rendimiento de los técnicos.

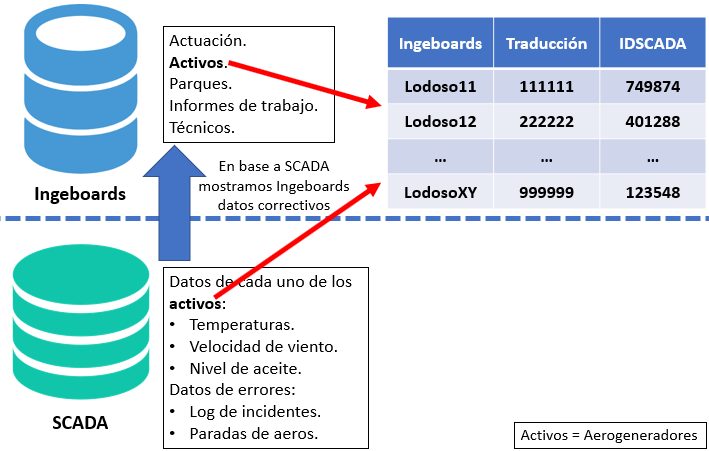


Ilustración 2. Infraestructura del entorno de Eolic Event Consumer.

Nuestra aplicación tiene que interactuar con ambos sistemas tal y como vemos en la Ilustración 2. Como “Eolic Event Consumer” predice en base a los datos obtenidos de los activos, nuestra aplicación trabajará debajo de “Ingeboards”, y paralelamente con “SCADA”. De esta forma, tras procesar los datos que son enviados desde “SCADA” y detectar un evento de interés, se insertaría una tupla a una tabla de la base de datos de “Ingeboards”, desplegando la información en forma de alarma.

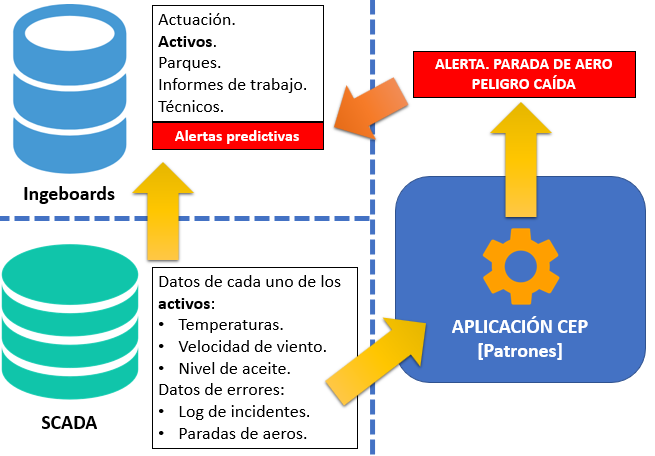


Ilustración 3. Incorporación de CEP al sistema actual.

Habiendo visto esto, como punto de partida, diremos que “NO TODOS” los campos captados por el sistema de monitorización son representativos para analizar. Solo tendremos en cuenta los factores más significativos, estos son la temperatura, la presión, la velocidad del viento y la potencia producida. Todos y cada uno de estos factores son los más importantes si examinamos el rendimiento de un aerogenerador. En concreto, haremos mucho hincapié en examinar la curva de potencia, que representa la potencia producida frente a la velocidad del viento.

No obstante, esto será nuestro punto de partida, en base a las conclusiones sacadas tras analizar estos cuatro factores iremos teniendo en cuenta el resto.

A la hora de analizar los datos deberemos de agruparlos por aerogenerador ya que las mediciones que se nos proporcionan son de un parque eólico. Por lo tanto, también sería interesante ver el rendimiento frente a otros parques para ver cuál es óptimo. Cada parque eólico y sus medidas se han de contextualizar debido a que la situación geográfica es distinta. Tendremos que filtrar los datos puesto que hay mediciones que no son significativas y son despreciables de forma que estas pueden afectar negativamente a nuestras predicciones. Ambos sistemas tanto SCADA como Ingeboards trabajan sobre bases de datos MySQL, de forma que a la hora de detectar un evento de interés bebiendo de los datos de SCADA podemos insertar una tupla en la tabla “Alarmas” de Ingeboards.

Todo el sistema de Ingeboards está implementado con PHP 5 o posterior (no se sabía exactamente), de aquí podemos concluir en qué lenguaje deberemos de implementar la aplicación gráfica.

* + - 1. Próximos pasos

En esta reunión se han acordado las siguientes tareas:

* Integrar la lectura de datos en formato .db.
* Filtrar los datos leídos debido a la lectura de valores nulos y registros diez minútales duplicados, porque afecta muchísimo al análisis de los datos.
* Establecer un flujo de datos donde se suelten .db en un directorio y se procese toda la información recogida.
* Mandar anteproyecto e información acerca el procesamiento de eventos complejos a Francisco José Polo.
* Dar información en diez minútales del último año.
* Antonio explicará a Fernando, todo lo referente acerca placas fotovoltaicas.
  + 1. Reunión 5. Clase tutorizada de energía eólica de Francisco José Polo
       1. Información

Se trata de una pequeña reunión que sirvió para conocer qué factores son claves para analizar las mediciones de un aerogenerador.

* Dirección: Parque Científico y Tecnológico, Paseo de la Innovación, 3, 02006 Albacete. Oficinas de Ingeteam S.A.
* Hora y fecha de la reunión. A las 8:15, el día 2 de febrero de 2018.

Los asistentes de la reunión son:

* Francisco José Polo Sánchez. Jefe de departamento de I+D+i de Ingeteam Service.
* Enrique Brazález Segovia. Desarrollador y encargado de defender este trabajo de fin de grado.
  1. Asuntos que tratar

Esta reunión es una clase para recoger toda la información que puede ser clave a la hora de tener que hacer el análisis. Los puntos de la reunión fueron:

* Explicación del modelo de datos utilizado.
* ¿Qué es el “SystemNumber”?
* Nociones de energía eólica para tener en cuenta.
  + - 1. Información recogida

Se dieron diferentes nociones, que son las siguientes:

* **Formaciones en parques eólicos.** Tal y como vemos en la Ilustración 3 podemos observar que no es trivial la formación que hacen los aerogeneradores en un parque eólico. Es un balance de costes y rendimiento. En un primer momento podríamos pensar que cuantos mas aerogeneradores en un mismo espacio pues más viento aprovecharíamos. Sin embargo, esto no es así, ya que una vez que pasa el viento a través de los aerogeneradores el viento genera turbulencias de forma que el viento pierde fuerza, y no es igual de aprovechable que al principio. Es un balance costes y rendimiento, ya que a mayor número de aerogeneradores más costes, a más distancia entre activos más costes debido al uso de cableado, de forma que en cada situación geográfica la formación óptima es distinta.

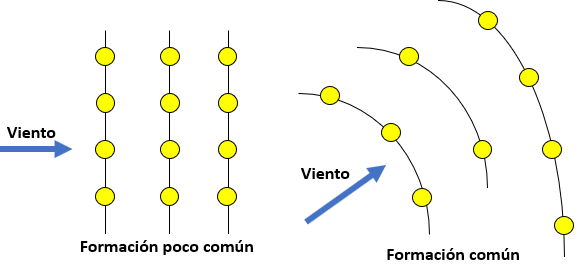


Ilustración 4. Formaciones comunes y no comunes de un parque eólico.

No hay una formación ideal, debido a que la dirección del viento predominante no es siempre igual y en cada lugar es distinto. Por esta razón los aerogeneradores se ubican en lo alto de los cerros. El aire que fluye a través de la montaña es impulsado hacia la parte de arriba del terreno por lo que aprovechamos el doble de su fuerza reduciendo el espacio por donde el viento fluye.

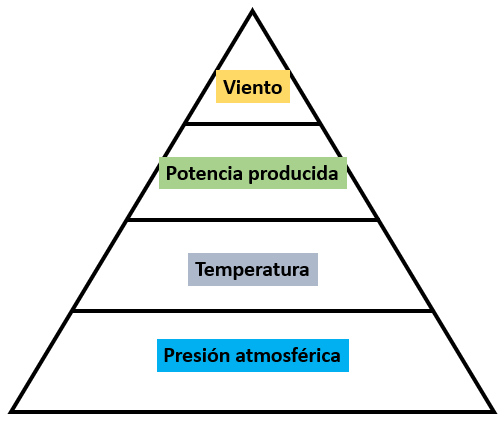
* **Traducción del nombre de los activos.** Cada uno de los aerogeneradores están identificados por un número, este número es el “SystemNumber”. Este identificador es el que tiene dentro del sistema de SCADA. Sin embargo, en “Ingeboards” tiene otro identificador. Por ejemplo, en SCADA podríamos denominar a un aerogenerador con el id “92874621”, y en Ingeboards podríamos nombrar a ese mismo con el alias “LodosoA1”. Esto sucede porque el uso de la información en un sistema y otro es distinto, Ingeboards es utilizado por técnicos y SCADA es más para un uso más estadístico.
* **Factores involucrados.** A la hora de analizar los datos proporcionados por SCADA hemos de tener en cuenta una pequeña jerarquía de factores. Esta jerarquía es la vista en la Ilustración 4. A más viento mayor es la potencia producida. A más temperatura del aire, menor densidad tiene el viento, de forma que, al tener menos fuerza, la potencia producida es menor. Es decir, cuanto más frío sea el viento más fuerza produce, y mayor es la potencia producida. Por otro lado, la presión también es importante, aunque en menor medida. Cuanta más presión hay, mayor es la fuerza que produce el viento sobre las palas del aerogenerador, por lo que la potencia producida es mayor.

Ilustración 5. Jerarquía de factores.

* **Curva de potencia.** Todo el análisis de datos girará en torno a la curva de potencia. Esta curva esquematiza el rendimiento del aerogenerador, enfrentando el viento en ese momento contra la potencia producida tal y como vemos en la Ilustración 5. Podemos ver que hay tres fases dentro de la curva de potencia, el arranque, la situación normal y la etapa en la que está a pleno rendimiento.

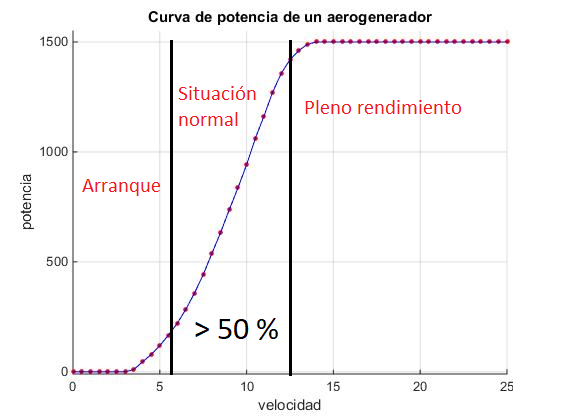


Ilustración 6. Ejemplo de curva de potencia.

Cabe destacar dos aspectos muy relevantes en cuanto a las etapas de la curva de potencia. Por un lado, diremos que más de la mitad de las mediciones de un aerogenerador se encuentran en la fase de “situación normal”, por lo que los activos están diseñados para trabajar en esas condiciones al encontrarse la mayor parte del tiempo ahí. Por otro lado, tenemos la etapa de pleno rendimiento que es cuando la velocidad de viento excede de los 15 m/s, ya que la potencia es constante. Esto se debe a que los aerogeneradores sufren un mayor desgaste a velocidades muy altas, por ello se limitan para producir lo mismo en estas situaciones.

* **Defectos en la curva de potencia.** La curva de potencia se dibuja a través de una nube puntos dependiendo de las mediciones reales de un aerogenerador, sin embargo, estas mediciones tienen ruido. Este ruido podría ser por ejemplo 7 puntos con un viento de 20 m/s, pero con potencia reducida igual a 0. Esto podría ser porque el aerogenerador en cuestión está siendo reparado y se haya parado para ejercer labores de mantenimiento. Por lo que deberemos de quitar ese ruido a la hora de analizar los datos. Los defectos que podemos predecir examinando la curva de potencia son:
  + Error referenciado de pitch 0º en PLC > 0º en físico.
  + Suciedad en palas.
  + Errores en tren mecánico.
  + Defectos en las palas.
  + Defectos eléctricos generados.
  + Quitan diez minútales con error asociado.
* **Mediciones de los fabricantes para vender aeros.** No podemos comparar nuestras conclusiones con los datos que los fabricantes otorgan, ya que se obvian muchos detalles que pueden ser muy relevantes.
  + - 1. Próximos pasos

En esta reunión se han acordado las siguientes tareas:

* Estudio de la información aportada.
* Sacar conclusiones para analizar los datos.
* Limpiar de ruido la información proporcionada.
* Tener en cuenta que se ha de agrupar las mediciones por aero, y hacer comparaciones a nivel individual (por activo) o bien a nivel de grupo (por parque).
  + 1. Reunión 6. Reunión de revisión del sprint y retrospectiva del Sprint Meter aquí al Product Owner
       1. Información

Es la reunión para revisar que es lo que se ha realizado a lo largo de esta primera iteración, ver qué hemos hecho correctamente y qué problemas hemos visto, para de esta forma hallar cómo resolverlos para que no vuelvan a afectar negativamente en el proyecto.

* Dirección. Edificio Infante Don Juan Manuel. Avda. de España s/n. Albacete. Despacho 0. B.8.
* Hora y fecha de la reunión. A las 16:30 el día 14 de febrero de 2018.

Los asistentes de la reunión son:

* Gregorio Diaz Descalzo. Jefe de proyecto de este trabajo de fin de grado. Titular de la Universidad de Castilla-La Mancha.
* Enrique Brazález Segovia. Desarrollador y encargado de defender este trabajo de fin de grado.
* Fernando Luján Martínez. Desarrollador de trabajo de fin de grado paralelo, su tutor también es Gregorio.
* Francisco José Polo Sánchez. Jefe de departamento de I+D+i de Ingeteam Service.
  + - 1. Asuntos que tratar

Esta reunión sirve como revisión de lo generado a lo largo de lo que se lleva de proyecto, ver qué aspectos positivos y negativos hemos visto, y en qué podemos mejorar. Los puntos de la reunión son:

* Revisión del prototipo.
* Revisión de la documentación.
* Revisión de como generar los requisitos.
* Como genero los storyboards.
* ¿Cómo insertar la documentación generada en las reuniones?
  + - 1. Información recogida

En cuanto a la revisión del prototipo. Hemos acordado que hemos cumplido con lo propuesto e incluido algún avance más como el ajuste de los datos, ya que el flujo de datos contiene ruido que influye negativamente en las predicciones. Queda validado, tanto el formato de datos en formato “csv” como la detección de eventos complejos y alarmas. En teoría el formato de entrada iba a ser en formato “.db”, sin embargo, debido a las dificultades que conlleva su transformación, la empresa colaboradora se compromete a establecer un “job” o batch, desarrollado con “Pentaho”, para pasar los datos en formato “.db” a “.csv” cada cierto tiempo, para alimentar el motor CEP.

De momento el primer prototipo, recoge todos los datos obtenidos en “.csv”, y registro a registro alimenta al motor CEP. Los registros son diez minútales, cosa que deberemos de tener en cuenta, puesto que no son datos a tiempo real exactos, si no que están ponderados por la media de lo medido en esos diez minutos. En resumen, el prototipo generado en este primer sprint cumple las expectativas impuestas con creces, ya que no solo tenemos un incremento en el que manda información a ThingSpeak y lo retorna al ESB, sino que, además, nos saltamos la dependencia de ThingSpeak y lo alimentamos directamente a través de ficheros “.csv”. De esta forma, minimizamos dependencias y, por si fuera poco, incrementamos la lectura de los datos notablemente. En la revisión del prototipo se detectaron ciertas deficiencias en cuanto al modelo establecido, no obstante, eran simples erratas en el código que daban conflictos sintácticos muy sencillos de solucionar.

Si nos centramos en la documentación generada, se revisaron los siguientes artefactos:

* **Anteproyecto**. En cuanto a lo que el anteproyecto se refiere, he de destacar que había multitud de erratas a lo largo de todo el texto, fruto de mi inexperiencia a la hora de redactar textos formales. Se revisó cuidadosamente la justificación de las competencias, y se pidió expresamente que se incorporara alguna referencia al modelo de calidad actual. El jefe de proyecto hizo hincapié en justificar correctamente, sin aportar más información de la necesaria. Un anteproyecto corto y conciso, es mejor que uno largo y redundante. Motivación, descripción del proyecto y método, fases de desarrollo y objetivos correctos.

En cuanto a los medios a utilizar, no sabemos con exactitud todo lo que nos hará falta, ya que hay decisiones que no hemos tomado todavía. Por ejemplo, no hemos decidido aún la tecnología que utilizaremos para hacer la aplicación gráfica, si usar node.js, o usar algo más tradicional a través de “Apache” y “Php”. Personalmente, mi postura es utilizar “Php” puesto que puede servirme como reto personal ya que nunca había utilizado “Php”, y “node.js” ya lo había visto en asignaturas como “Procesos de Ingeniería del Software” y “Web Engineering and Services”.

* **Storyboards para captura de requisitos de patrones.** Los storyboards preparados para capturar los requisitos de los patrones eran correctos, y útiles a la hora de implementar todos y cada uno de los patrones. Echando un vistazo a las viñetas, teniendo nociones muy básicas de energía eólica, cualquier persona puede entender lo que pretenden los patrones implementar. Un ejemplo de storyboard puede ser el de la Ilustración 6.

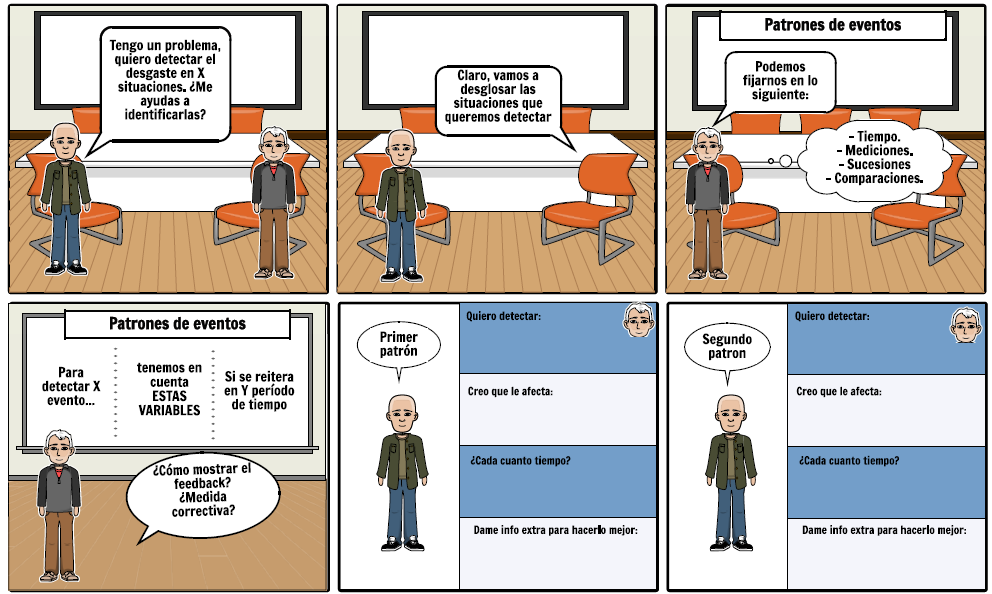
****

Ilustración 7. Plantilla de enunciado de patrones.

* **Actas de reuniones.** Una de las dudas del Team, es donde iban a tener cabida toda la documentación generada en cada una de las reuniones. Debido a que esta documentación es la correspondiente a la generada en una metodología SCRUM, la plasmaremos en el capítulo 3 de esta memoria, donde se verán todas las actividades desarrolladas en cada uno de los Sprints.
* **Arquetipos.** Este artefacto se ha añadido como extra por parte propia. Debido a la poca profesionalidad de nuestro primer enlace con la empresa, Pedro, nos hemos visto obligados a tomar precauciones a nivel organizativo, esquematizando y elaborando unos arquetipos de las personas con las que tenemos contacto. De esta forma, ante una necesidad o consulta, nos dirigiremos a la persona indicada instantáneamente.
* **Información técnica clave.** Había dudas de cómo plasmar en la memoria del trabajo de fin de grado toda la información técnica acerca de la energía eólica que se recogerían en el segundo sprint. Se decidió que tendría cabida en el capítulo del estado del arte.
  + - 1. Próximos pasos

Se han acordado la incorporación de las siguientes historias de usuario para el siguiente sprint:

* **Estudio de factores externos distintos de la potencia producida y el viento.** En concreto, el estudio de lo que influye la temperatura y la presión, en la productividad y rendimiento.
* **Estudio de la curva de potencia.** Deberemos de hacer un análisis individual a nivel de aerogenerador y otra a nivel de parque eólico. Buscar fuentes de información de otras universidades, o artículos referentes a la energía eólica en busca de posibles comportamientos anómalos en los aerogeneradores. Sacar conclusiones y enunciar con ayuda del Product Owner, el enunciado de los patrones.
* **Implementar patrones.** Implementar una primera versión de los patrones a raíz de las conclusiones sacadas con el análisis.
* **Documentación**. Para adelantar y no dejar para el último momento la redacción de la memoria, se pone como tarea escribir el primer y segundo capítulo. Cuidar el estilo de la memoria, así como incorporar en el capítulo pertinente toda la información correspondiente al primer Sprint.
  + 1. Retrospectiva del Sprint

Una vez hecha la reunión de revisión del Sprint, el Product Owner abandonó la sala, para que el Team, junto con el Scrum Master, tuvieran la retrospectiva. El objetivo de esta reunión fue responder a las siguientes preguntas:

**¿Qué salió bien en la iteración?**

* Se hizo un buen prototipo, que tenía más funcionalidades de las esperadas.
* Dio tiempo a hacer parte de la documentación referente a todas y cada una de las reuniones.
* Se solucionó la coordinación con el cliente, pese al incidente de la ida de Pedro.
* La elección del ESB de MuleSoft, puesto que tiene multitud de funcionalidades.
* Hay muy buena comunicación entre el Team, y el Scrum Master. No hay ningún tipo de problema a la hora de reunirnos.
* La documentación está muy bien detallada.
* Los Storyboards plasman de forma correcta todo.

**¿Qué no salió bien en la iteración?**

* Debido a la mala configuración de TFS, el Sprint Burndown no refleja correctamente lo introducido en el servicio de Microsoft. Suponemos que es debido a que las tareas se redactaron conforme avanzó el Sprint.
* La conversión de los ficheros de “.db” a “.csv”. No se pudo llevar a cabo de forma automática debido a que el único lector de .db que permitía procesos “batch” que los transformaba era “paradox-reader”, sin embargo, no funciona correctamente puesto que la tarea se queda colgada y permite la conversión de dos ficheros seguidos.

**¿Qué mejoras vamos a implementar en la próxima iteración?**

* La implementación de los patrones.
* Amoldamiento de los patrones al ESB.
* Instauración de la base de datos MySQL donde introduciremos las alarmas de interés.
* Si sobra tiempo, integrar de forma automática la conversión de ficheros “.db”.
* Empezar la memoria del trabajo de Fin de Grado.
  1. SPRINT 2. Adaptación al caso real

A continuación, tendremos la explicación de lo que se desarrolló en el Sprint 2.

* + 1. Reunión 7. Reunión del comienzo del segundo
       1. Información

En resumen, esta reunión sirvió como punto de inicio de este sprint. En ella se establecieron los objetivos a conseguir y se estipuló la funcionalidad que debería de cumplir este prototipo en la segunda iteración.

* Dirección. Edificio Infante Don Juan Manuel. Avda. de España s/n. Albacete. Despacho 0. B.8.
* Hora y fecha de la reunión. A las 17:00 el día 24 de febrero de 2018.

Los asistentes de la reunión son:

* Gregorio Diaz Descalzo. Jefe de proyecto de este trabajo de fin de grado. Titular de la Universidad de Castilla-La Mancha.
* Enrique Brazález Segovia. Desarrollador y encargado de defender este trabajo de fin de grado.
* Fernando Luján Martínez. Desarrollador de trabajo de fin de grado paralelo, su tutor también es Gregorio.
  + - 1. Asuntos que tratar

Esta reunión sirve como revisión de lo generado a lo largo de lo que se lleva de proyecto, ver qué aspectos positivos y negativos hemos visto, y en qué podemos mejorar. Los puntos de la reunión son:

* ¿Qué va a tener nuestro prototipo?
* ¿Qué patrones he detectar? ¿Qué va a ser objeto de estudio?
* ¿En qué plataforma hacemos la aplicación gráfica? ¿Node.js (sin complicación, ya sabía en su día utilizarlo) o usar PHP (un reto, no sabía utilizarlo)?
* ¿Algún problema con el cliente?
  + - 1. Información recogida

Lo más representativo de esta reunión fue el planteamiento de objetivos de este sprint. Los objetivos que se deberían de cumplir serían los siguientes:

* **El prototipo deberá de desplegar los patrones que se implementen dentro del ESB.** Lo más relevante de este sprint es el desarrollo de estos patrones, deberemos de centrarnos en el estudio de la curva de potencia tanto a nivel de parque eólico como a nivel de aerogenerador individual.
* **Los eventos complejos de interés deberán de insertarse en una base de datos MySQL.** El Scrum Master sugirió utilizar un todo en uno para incorporar tanto PHP como algún sistema gestor de base de datos como es MySQL, en concreto se tomó la decisión de utilizar XAMPP. Se eligió XAMPP porque es gratuito, y era el que más documentación tenía para poder consultar dudas.
* Deberemos de traducir los patrones una vez implementados e integrados en el ESB, a través de Medit4CEP, ya que es mucho más visual su representación que el propio código. Sirve como apoyo adicional al código para su comprensión.

Otro punto de la reunión fue el estipular que patrones deberíamos de implementar. En su día en una de las reuniones con el cliente, nos requirió examinar con escrupulosidad la curva de potencia. Según las reuniones con el cliente se hizo mucho hincapié en estudiar el comportamiento de los aerogeneradores en los distintos bines de viento, es decir, el rendimiento que tienen en intervalos de viento concretos. También es interesante estudiar productividad, factor que enfrenta potencia producida y tiempo. Tendremos que normalizar el patrón de alguna forma puesto que el factor geográfico es determinante a la hora de que un aerogenerador es más productivo que otro. También se sugiere contrastar las conclusiones que saquemos con el log de errores para así comprobar que la alerta que el motor CEP ha sacado es correcta, esto último se llevaría a cabo solo en caso de terminar el proyecto antes de tiempo. Por el momento, solo abarcaremos todo lo que tenga que ver con rendimiento en los diferentes bines de viento, y productividad.

A continuación, se trató el tema de la aplicación gráfica que se iba a implementar. Se discutieron varios tipos de tecnologías, todas y cada una de ellas, servicios web. Sobre todo, hemos de destacar la discusión de usar Node.js y un sistema tradicional a través del uso de PHP. Por un lado Node.js, nos daba la ventaja de que ya lo habíamos utilizado, cosa que no nos suponía ningún problema, sin embargo, el cliente utilizaba PHP en su sistema, y su equipo técnico no tenía ninguna noción acerca de Node.js, cosa que dificultaba mucho su integración en el sistema. Por esta razón, y por mis ganas de aprender con el desarrollo de este trabajo de fin de grado, se decidió utilizar PHP, tal y como el servicio técnico de informáticos del cliente utilizaba. Esta decisión facilitó la integración de nuestro trabajo, con el sistema de “Ingeboards” que el cliente estaba desarrollando y me ayudó a adquirir más conocimientos acerca de una tecnología que hasta el momento me era desconocida.

Debido a los problemas que habíamos tenido anteriormente de compromiso por parte del cliente, dedicamos un extracto de la reunión para hablar de los posibles problemas que había de comunicación. Exceptuando el problema de coordinación que se solucionó gracias a la intervención de Francisco José Polo, no hubo ningún otro problema que destacar.

* + - 1. Próximos pasos

Como consecuencia de esta primera reunión queda fijado llevar a cabo los objetivos nombrados en el punto 3 de esta reunión.

* + 1. Reunión 8. Reunión para explicación de
       1. Información

En resumen, esta reunión sirvió como punto de inicio de este sprint. En ella se establecieron los objetivos a conseguir y se estipuló la funcionalidad que debería de cumplir este prototipo en la segunda iteración.

* Dirección. Edificio Infante Don Juan Manuel. Avda. de España s/n. Albacete. Despacho 0. B.8.
* Hora y fecha de la reunión. A las 17:00 el día 22 de febrero de 2018.

Los asistentes de la reunión son:

* Gregorio Diaz Descalzo. Jefe de proyecto de este trabajo de fin de grado. Titular de la Universidad de Castilla-La Mancha.
* Enrique Brazález Segovia. Desarrollador y encargado de defender este trabajo de fin de grado.
* Fernando Luján Martínez. Desarrollador de trabajo de fin de grado paralelo, su tutor también es Gregorio.
  + - 1. Asuntos que tratar

En esta reunión se propusieron las primeras versiones de los patrones, para que el Scrum Master

* Fundamentos teóricos y explicación de patrones.
* Aceptación o denegación de los patrones presentados.
  + - 1. Información recogida

Tras unos días estudiando las nociones teóricas de la energía eólica junto con algunos expertos procedentes de la empresa cliente, hemos sacado varias conclusiones y un primer acercamiento a los patrones a implementar. Los patrones que he pensado por el momento miden productividad y rendimiento. La gran diferencia entre ellos la podemos ver en la Tabla 2

Tabla 2. Factores enfrentados.

|  |  |
| --- | --- |
| Rendimiento | Productividad |
| Viento VS Potencia producida | **Tiempo VS Potencia producida** |

Como podemos ver, por un lado, uno enfrenta viento y potencia producida al intervalo de tiempo que deseemos. Por otro lado, tenemos otro que enfrenta la potencia producida diariamente. A continuación, veremos la explicación de los mismos.

1. **Patrón de Rendimiento**

Para medir el rendimiento nos centraremos en el estudio de la curva de potencia de los aerogeneradores. Sin embargo, hay ciertos problemas a la hora de comparar “el buen rendimiento”, ya que no todos los molinos, reciben en un instante X, una velocidad de viento Y. El viento en una región tan amplia como un parque eólico es variable, ya bien sea por la disposición de los aerogeneradores (en matriz, en filas irregulares, etc.) o bien por la disposición geógrafica en la que se encuentran (uno encima de un cerro, otro en mitad de una llanura, etc.). Por tanto, este patrón lo que intenta es ver qué aerogeneradores son los mejores en distintos intervalos de viento. Los intervalos de análisis son unitarios, es decir, de 0 a 1, de 1 a 2, y así hasta 18, puesto que a partir de 18 se da la potencia nominal del aerogenerador que estamos estudiando. Por ejemplo, en la Tabla 3, vemos el estudio del rendimiento en el intervalo que va de 6 a 7 y de 7 a 8, en el día 1 de enero.

Estas mediciones son comparables entre sí, y sí que se podrían analizar por algún tipo de criterio sin tener en cuenta la disposición geográfica. Por ejemplo, se podrían ver las desviaciones en tanto por ciento con respecto la media, o entre ellos. A raíz de ahí, veremos que aerogeneradores son los “peores”. En teoría, todos los aeros son iguales, sin embargo, si hay una diferencia significativa puede que haya un fallo que el sistema de monitorización no haya detectado. Además, gracias a este control podremos prevenir este tipo de deficiencias. Para discretar este análisis y verlo más visual e intuitivo podemos hacer un análisis por ranking de rendimiento en cada intervalo de tiempo.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID de aero | Intervalo m/s | Media del intervalo | Sigma | Media del aero | Predicción | Posición ranking |
| 1 | [6,7) | 427.80 W | 75.61 W | 423 W | A determinar | 3º |
| 2 | [6,7) | 427.80 W | 75.61 W | 425 W | A determinar | 2º |
| 3 | [6,7) | 427.80 W | 75.61 W | 450 W | A determinar | 1º |
| 1 | [7,8) | 658.27W | 90.12 W | 650 W | A determinar | 1º |
| 2 | [7,8) | 658.27W | 90.12 W | 610 W | A determinar | 3º |
| 3 | [7,8) | 658.27W | 90.12 W | 636 W | A determinar | 2º |

Tabla 3. Mediciones rendimiento el día 1 de enero.

Teniendo en cuenta la tabla anterior hemos visto que en el intervalo de tiempo [6,7), que las posiciones en función de la media obtenida por aerogenerador es en primera posición el aero 3, en segunda el aero 2, y en tercero el aero 1.

Si lo representamos en un evento,

tendría la siguiente forma:

Veremos pues que los elementos del evento quedan ordenados, según los datos obtenidos. Bien, supongamos que el día 2 de enero se obtiene, el siguiente evento:

Como podemos observar dentro de este ranking, de un día para otro, el aero 3 ha bajado una posición. Esa posición a nivel mecánico puede significar algún tipo de desperfecto en la multiplicadora, una subida de temperatura que no debiera de darse, etc. Esto puede ser un evento de interés a ser analizado. La idea es que el algoritmo revele estos desplazamientos dentro del ranking para hacer predicciones a corto-medio plazo.

1. **Patron de Productividad**

En cuanto a lo que productividad se refiere, enfrentamos dos factores muy importantes que son la potencia producida y el tiempo. Esta vez, como hemos comentado anteriormente, el factor geográfico y la posición de los aerogeneradores es muy relevante para encontrar un criterio que nos permita decir cuál es más productivo que otro. En la Ilustración 7, vemos lo que producen 6 aerogeneradores a lo largo de un día. Como podemos observar sus líneas de tendencia lineales indican cual es más productivo el uno con respecto el otro. Sin embargo, estas mediciones son dependientes del viento. Por tanto, el que reciba más viento, es el que más produce.

Ilustración 8. Grafíco de potencia producida el día 1 de enero.

Dicho lo anterior, hemos de incorporar el estudio del contexto de cada generador, tanto el factor geográfico, como su lugar dentro de la formación del parque. Debido a estos dos criterios, en condiciones normales, todos y cada uno de los aeros están destinados a ser uno más productivos que otros. Por lo tanto, no podemos hacer un análisis cuantitativo únicamente fijándonos en los valores numéricos de su producción, obviando la situación dentro del parque del aerogenerador.

Por tanto, lo que haremos es analizar los desplazamientos dentro de ese orden al que están destinados a producir cada uno de los aerogeneradores. En la Tabla 4, vemos un ejemplo de las mediciones de los días 1 y 2 de enero.

Tabla 4. Medidas de productividad el día 1 de enero.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID de aero | Día | Potencia producida (en día) | Nº Eventos representativos | Media del aero | Posición ranking |
| 1 | 1 de enero | 4200 W | 3 | 1400 W | 2º |
| 2 | 1 de enero | 3000 W | 2 | 1500 W | 1º |
| 3 | 1 de enero | 3000 W | 3 | 1000 W | 3º |
| 1 | 2 de enero | 4100 W | 3 | 1366 W | 1º |
| 2 | 2 de enero | 3200 W | 3 | 1066 W | 3º |
| 3 | 2 de enero | 4000 W | 3 | 1333 W | 2º |

Supongamos que el orden de producción es el conjunto ordenado descendientemente por productividad. Veremos que el día uno de enero no cumple con el orden ideal, podremos decir que el parque ha producido lo que se esperaba el día 1 de enero, puesto que . Sin embargo, si analizamos el día dos, vemos que . Si analizamos R2 y R3, vemos que la productividad del aero 2 ha bajado muchísimo, esto no debería de pasar, puesto que el aero 2 en condiciones normales, debería de estar de los primeros. Esta bajada, puede ser señal de que hay algún fallo mecánico, alguna reparación, un evento inesperado etc. En base a las traslaciones de esta especie de ranking podremos implantar medidas preventivas cuando más nos convenga.

ATENCIÓN. “Que el aerogenerador dibuje una curva de potencia más elevada con respecto el resto, no implica que produzca más (sea el más productivo)”

Una vez expuestos los patrones a implementar, el Scrum Master, dio el visto bueno a los patrones a implementar, era una buena solución y con una dificultad adecuada. Sin embargo, quedó pendiente que el cliente lo verificará y viera el visto bueno o las apreciaciones necesarias para ajustarse a sus necesidades.

* + - 1. Próximos pasos

Como consecuencia de esta reunión y antes de empezar a implementar los patrones, deberemos de comunicarnos con el cliente y cerciorarnos de que las ideas planteadas son correctas y solventa sus necesidades. Por tanto, Enrique se deberá de reunir a solas con Francisco José Polo para confirmar el enunciado de los patrones, así como estructurarlos y definir de una vez por todas qué patron se va a implementar, y cuál es el límite de nuestro proyecto.

* + 1. Reunión 9. Reunión para concretar y validar los patrones con el cliente
       1. Información

En esta reunión se procedió a la validación de los fundamentos funcionales que tienen los patrones ideados, así como posibles retoques que mejorarán la exactitud de los eventos de interés a detectar.

* Dirección: Parque Científico y Tecnológico, Paseo de la Innovación, 3, 02006 Albacete. Oficinas de Ingeteam S.A.
* Hora y fecha de la reunión. A las 10:00 el día 23 de febrero de 2018.

Los asistentes de la reunión son:

* Francisco José Polo Sánchez. Jefe de departamento de I+D+i de Ingeteam Service.
* Enrique Brazález Segovia. Desarrollador y encargado de defender este trabajo de fin de grado.
  + - 1. Asuntos que tratar

Los puntos que se desarrollarán en esta reunión son los siguientes:

* Fundamentos teóricos y funcionamiento de los patrones.
* Aceptación o denegación de los patrones ideados.
* Posibles mejoras en los patrones.
* Apreciaciones que se podrían hacer extras utilizando el log de errores.
  + - 1. INformación recogida

En primer lugar, Enrique expusó todos los puntos que definen los patrones ideados, que intentan medir productividad y rendimiento. Una vez hecho esto Francisco José dio el visto bueno a los patrones. Sin embargo, corregió algunos aspectos que eran incorrectos, mejorando así el funcionamiento del algoritmo CEP.

Dio la aprobación completa al patrón para esquematizar la productividad, le pareció interesante. Destacó que analizar cualititativamente los aerogeneradores como si fueran corredores en una carrera, normalizando las condiciones geográficas, mediante los desplazamientos en un ranking, era una buena idea.

Una de las apreciaciones más importantes a la hora de llevar a cabo el análisis, es que a la hora de estudiar el rendimiento que los intervalos de viento no fueran unitarios, si no que fueran por bines de viento. Es decir, que se analicen intervalos de un medio de dominio, tal que si un evento era:

,

Después de la intervención del cliente estipulamos que un evento queda representado como:

Las variaciones de potencia producida en los intervalos centrales de la curva son demasiado grandes, de tal forma que al minimizar el intervalo obtendremos más precisión en el análisis, ya que reducimos las desviaciones en los diferentes dominios de viento. Además, sugirió separar el análisis a través de tres patrones distintos pero complementarios entre sí, son los siguientes:

* **Análisis de desviaciones.** Cada X tiempo se recogerían los datos de los distintos aerogeneradores, de tal forma que haríamos un análisis de sus desviaciones típicas y consideraremos que aquel que tenga una desviación mayor, es más irregular, y, por tanto, peor con respecto el resto.
* **Análisis de puntos fuera de los intervalos de confianza en los bines de viento**. Este patrón nace a raíz del anterior, ya que, a través de la desviación típica obtenida de cada aerogenerador en un bin de viento, podremos obtener el intervalo de confianza de sus mediciones. Contaremos cuantos puntos se quedan fueran de ese intervalo, de tal forma que aislaremos qué puntos no son normales dentro de los datos obtenidos en ese tiempo. Estos puntos pueden ser el indicio de que pueda haber alguna avería mecánica a corto-medio plazo. Por lo general, podremos tener la hipótesis de que el aerogenerador que más puntos tenga fuera de su intervalo de confianza será el peor. También se sugirió hacer un ranking con este criterio.
* **Análisis de media y bajadas en tanto por ciento del unos con respecto otros**. A través de las medias obtenidas por cada aerogenerador en los diferentes bines de viento, analizaremos las bajadas de rendimiento, con respecto sus propias medias o de las del resto de aerogeneradores. Es interesante analizar para un bin de viento, porqué un aerogenerador ha bajado su rendimiento en un tanto por ciento con respecto lo obtenido en el día anterior.
  + - 1. Próximos pasos

En esta reunión se confirmaron todos y cada uno de los patrones a implementar, hasta este momento no se acotó la amplitud del proyecto, ni las funcionalidades que iba a tener el prototipo de esta iteración. Hecho esto, aquí empezó la implementación de los patrones. Las tareas fruto de esta reunión son:

* Implementar el patrón para medir productividad.
* Implementar el patrón para medir rendimiento.
* Cambiar el filtrado de datos para que los intervalos sean los correspondientes a los bines de viento, y no en intervalos unitarios.
* Cambiar el flujo de envío de alertas del ESB, para que se añada una tupla a la base de datos MySQL correspondiente al evento complejo detectado.
  + 1. Reunión 10. Reunión para revisión de código y comunicación de patrones definitivos
       1. Información

En resumen, esta reunión sirvió como punto de inicio de este sprint. En ella se establecieron los objetivos a conseguir y se estipuló la funcionalidad que debería de cumplir este prototipo en la segunda iteración.

* Dirección. Edificio Infante Don Juan Manuel. Avda. de España s/n. Albacete. Despacho 0. B.8.
* Hora y fecha de la reunión. A las 17:00 el día 1 de marzo de 2018.

Los asistentes de la reunión son:

* Gregorio Diaz Descalzo. Jefe de proyecto de este trabajo de fin de grado. Titular de la Universidad de Castilla-La Mancha.
* Enrique Brazález Segovia. Desarrollador y encargado de defender este trabajo de fin de grado.
* Fernando Luján Martínez. Desarrollador de trabajo de fin de grado paralelo, su tutor también es Gregorio.
  + - 1. Asuntos que tratar

Esta reunión sirvió como recurso para comunicar los patrones definitivos y para labores de coordinación dentro del trabajo de fin de grado. Los puntos de la reunión fueron:

* Comunicación de los patrones definitivos.
* Exposición de dudas en un patrón, puesto que no se sabía en su día como plantearlo.
* Coordinación del trabajo de fin de grado.
  + - 1. Información recogida

Se puso en conocimiento al Scrum Master de cuáles iban a ser los patrones definitivos a implementar, ya dado el visto bueno por parte del cliente. El Scrum Master vio correcto el enunciado de los patrones, y que eran posibles de implementar.

A continuación, ya se había empezado a implementar el patrón que tenía que ver con la productividad. Sin embargo, la solución propuesta no estaba completa ya que a la hora de hacer un SELECT con una ventana temporal de tipo BATCH, no se conseguía añadir el ROW NUMBER para poder añadir una columna más que indique la posición de los aeros dentro del ranking. Sin embargo, se pensó a hacer de otra forma, capturando tantos eventos seguidos como el número de aerogeneradores multiplicado por dos. De esta forma, capturaremos el ranking obtenido en dos días consecutivos y compararemos los unos con respecto los otros para ver si han cambiado las posiciones de un día para otro.

Solucionado este problema a nivel de implementación, pasamos a la coordinación del trabajo de fin de grado. Debido a que solo quedarían tres meses para la entrega del mismo, el Scrum Master mandó a ambos alumnos empezar a elaborar la memoria en la medida de lo posible, en concreto, con los capítulos del estado del arte y a documentar el primer sprint. Además, se estipuló que iba a ver una reunión semanal a partir de la semana del 19 de marzo debido a que había tener un mayor control de lo que se iba haciendo, y así evitar posibles atascos en las tareas del proyecto.

* + - 1. Próximos pasoS

Las tareas que deberemos desarrollar después de esta reunión son principalmente terminar de elaborar los patrones, empezar la memoria de trabajo de fin de grado, y continuar las tareas impuestas en el backlog.

1. Experimentos y resultados

En este apartado metemos todas las pruebas funcionales unitarios y de integración llevadas a cabo en este proyecto. Así como pruebas de calidad.

1. Conclusiones y propuestas
   1. Conclusiones
   2. Trabajo futuro

Bibliografía

En esta sección veremos todas las referencias bibliográficas de las que ha nacido este trabajo de fin de grado a través del estándar IEE.

[1] X. Zhao, H. Fan, H. Zhu, Z. Fu, and H. Fu, “The Design of the Internet of Things Solution for Food Supply Chain,” *Proc. 2015 Int. Conf. Educ. Manag. Inf. Med.*, vol. 49, no. Emim, pp. 314–318, 2015.

[2] M. Antonio and R. Borja Díaz, “Energía eólica,” vol. 1, pp. 54–57, 2013.

[3] IBM, “Business Rules Management System | IBM Digital Business Automation.” [Online]. Available: https://www.ibm.com/cloud/automation-software/business-decision. [Accessed: 29-Jan-2018].

[4] R. E. SIEMENS Gamesa, “Gamesa consolida una tecnología propia para el mantenimiento predictivo de sus aerogeneradores,” *29-10-2008*, 2018. [Online]. Available: http://www.siemensgamesa.com/es/comunicacion/noticias/gamesa-consolida-una-tecnologia-propia-para-el-mantenimiento-predictivo-de-sus-aerogeneradores.html?idCategoria=0&fechaDesde=&especifica=0&texto=&fechaHasta=. [Accessed: 14-Mar-2018].

[5] Microsoft Corporation, “La Arquitectura Orientada a Servicios (SOA) aplicada al mundo real,” 2006.

[6] S. O. Tortosa, A. O. Baíllo, and R. B. Plata, “Arquitectura Orientada a Servicios Para La Distribuidos,” pp. 3–10, 2006.

[7] M. Döhring *et al.*, “The convergence of workflows, business rules and complex events: Defining a reference architecture and approaching realization challenges,” *ICEIS 2010 - Proc. 12th Int. Conf. Enterp. Inf. Syst.*, vol. 3 ISAS, no. January 2010, pp. 338–343, 2010.

[8] S. Moreno Saiz, “Estudio de Arquitecturas Software para Servicios de Internet de las Cosas,” 2015.

[9] D. Luckham and R. Schulte, “Event Processing Glossary - Version 2.0,” *Processing*, no. July, pp. 1–19, 2011.

[10] P. Russom, “Big data analytics,” *TDWI Best Pract. Rep.*, pp. 1–35, 2011.

[11] J. Boubeta, “Model-Driven Development of Domain-Specific Interfaces for Complex Event Processing in Service-Oriented Architectures.,” p. 223, 2014.

[12] J. B. Puig, G. O. Bellot, and I. M. Bulo, “Procesamiento de Eventos Complejos en Entornos SOA: Caso de Estudio para la Detecci{ó}n Temprana de Epidemias,” *VII Jornadas Cienc. En Ing. Serv.*, pp. 63–76, 2011.

[13] Alberto del Barrio de Pablos, “Introducción al procesamiento de Eventos Complejos I - Decide Soluciones,” *2013-09-15*, 2018. [Online]. Available: http://decidesoluciones.es/introduccion-al-procesamiento-de-eventos-complejos-i/. [Accessed: 15-Mar-2018].

[14] Eclipse, “Eclipse - The Eclipse Foundation open source community website.” [Online]. Available: https://www.eclipse.org/. [Accessed: 29-Jan-2018].

[15] Eclipse Egit, “EGit.” [Online]. Available: http://www.eclipse.org/egit/. [Accessed: 29-Jan-2018].

[16] MuleSoft, “Anypoint Studio // Documentación de MuleSoft.” [Online]. Available: https://docs.mulesoft.com/anypoint-studio/v/6/. [Accessed: 29-Jan-2018].

[17] EsperTech, “Esper - EsperTech.” [Online]. Available: http://www.espertech.com/esper/. [Accessed: 29-Jan-2018].

[18] J. Boubeta-Puig, G. Ortiz, and I. Medina-Bulo, “MEdit4CEP: A model-driven solution for real-time decision making in SOA 2.0,” *Knowledge-Based Syst.*, vol. 89, pp. 97–112, Nov. 2015.

Anexos

* 1. Ejemplo de uso de la herramienta

adflñkajf qelrkj qer lqewrj hqlkrj qlhr lqjr lkqrk ckzfjasdlfh qenrl jqelrkj qleh hnlqwerj qw

lajfa lnfladsjf asdfn ladfj aldfladjf ladjflñaeyrqehrn lqwerj oqewrh nqer

* 1. Manual de usuario

lkadfjla jdflqjer qertkj qer'ijqtej qoier hnqert

kllñkalñkdfg laesrfj lqñwer hnladf fjowqehtn lkwrjeoyhk lñkjje wlkjwenr ñljwer k