

---

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

Piferrer Iglesias, Didac; Ribas i Xirgo, Lluís, dir. Monitoreo automático enfocado en la industria 4.0. 2022. (958 Enginyeria Informàtica)

---

This version is available at <https://ddd.uab.cat/record/264207>

under the terms of the  license

# Monitoreo automático enfocado en la industria 4.0

Didac Piferrer Iglesias

**Resumen**– Este documento recoge el procedimiento de automatización del proceso de monitorización de una parte del sistema de producción de Boehringer Ingelheim el cual inicialmente se realizaba de forma manual. En los procesos recurrentes y de carácter tecnológico, normalmente, las intervenciones humanas no solo ralentizan estos procesos, sino que también pueden llegar a ejecutarlos con algunas imprecisiones. En un entorno industrial, como es el farmacéutico, es muy importante conseguir el mayor valor del proceso en el mínimo tiempo posible y de la manera más precisa posible. Es aquí donde entra la digitalización y automatización de procesos que proporcionan una mejora considerable tanto en velocidad como en calidad de ejecución respecto la realización manual del proceso que lleva a cabo el ser humano.

**Palabras clave**– Industria 4.0, ISA 95, monitorización, automatización, Boehringer Ingelheim, farmacéutica, entorno productivo.

**Abstract**– This document includes the automation procedure for the monitoring process of a part of the Boehringer Ingelheim production system, which was initially carried out manually. In recurrent and technological processes, normally, human interventions not only slow down these processes, but can also execute them with some inaccuracies. In an industrial environment, such as the pharmaceutical one, it is very important to obtain the greatest value from the process in the shortest possible time and in the most precise way possible. This is where the digitization and automation of processes comes in, which provide a considerable improvement in both speed and quality of execution compared to the manual execution of the process carried out by the human being.

**Keywords**– Industry 4.0, ISA 95, monitoring, automation, Boehringer Ingelheim, pharmaceutical, production environment.



## 1 INTRODUCCIÓN

DENTRO de Boehringer Ingelheim [<https://www.boehringer-ingelheim.es/>], se encuentra un equipo que ejerce de puente entre el departamento de IT y el de ingeniería. Juntos, apoyan al negocio en la implementación de la estrategia digital. Este equipo más general está dividido en dos grandes equipos:

**El primer equipo** es el primer contacto con el departamento de Ingeniería. Proporciona orientación en el uso de los servicios de IT existentes, lo que permite que los equipos de ingeniería se concentren en su trabajo principal.

**El segundo equipo** es donde se implementan los casos de uso de negocio que apoyan la transformación digital de las fábricas. Actualmente, la tecnología principal usada es Inmation [1].

Estos dos equipos se coordinan para llevar a cabo un programa que se enfoca en brindar casos de uso digital al negocio de Boehringer Ingelheim. Dentro de los casos de uso se destaca el “Continuos process monitoring” que se desarrolla en este proyecto con el equipo donde se han estado realizando las prácticas.

Principalmente, el caso de uso es definido como el completo y continuo proceso de monitoreo del entorno de producción para un óptimo proceso de control, detección de errores y la prevención de estos.

- E-mail de contacto: didacpiferrer@hotmail.com
- Mención realizada: Ingeniería de computadores
- Trabajo tutorizado por: Lluís Ribas Xirgo (Departamento de microelectrónica y sistemas electrónicos)
- Curso 2021/22

## 1.1. Objetivos

La visión empresarial del equipo es no intervenir en el proceso de monitoreo y que se realice automáticamente para que los responsables puedan manejar ellos mismos sus máquinas. De tal manera que la ayuda del equipo solo sea requerida en casos puntuales.

Los objetivos planteados del proyecto en el intervalo de tiempo definido han sido los siguientes:

- Disponer de un sistema unificado de registros en libros de cálculo de Excel [<https://www.microsoft.com/es-es/microsoft-365/excel>] obtenidos con la *WebApp* de Inmation.
- Tener un sistema de correo para la comunicación automática de la información relevante del monitoreo diario.

Estos objetivos se desglosan en las siguientes tareas:

1. Mejorar los análisis de los registros mediante el complemento de Excel previamente desarrollado.
  - a) Corregir errores en las fechas de los mensajes.
  - b) Añadir una nueva funcionalidad que permita, en otra hoja, organizar los mensajes por objeto, consiguiendo así una mejor relación de los registros.
2. Suprimir Confluence  
[<https://www.atlassian.com/es/software/confluence>] como herramienta de reporte y fusionarlo con Excel, añadiendo otra nueva funcionalidad al complemento que genere una hoja con las tablas correspondientes para hacer el reporte de los objetos del sistema.
3. Mejorar la obtención de las métricas para eliminar falsos errores.
4. Crear una nueva función que obtenga de forma organizada el estado de todos los objetos e indique por cada fábrica cuantos elementos se encuentran en un estado no deseado.
5. Diseñar un nuevo panel para visualizar los nuevos datos obtenidos.
6. Documentar el proceso completo para su externalización (sin incluir la implementación del sistema automático de correo electrónico). Orientado sobre todo a la obtención de los logs y a como se realiza el proceso de monitoreo. La externalización será temporal hasta que el proceso esté completamente automatizado.
7. Implementar un aplicativo de gestión automático vía correo electrónico para suprimir la comunicación manual entre la persona que realiza el monitoreo y los distintos responsables de las máquinas.
  - a) Diseño del sistema.
    - 1) Tablas que se van a necesitar en la base de datos.
    - 2) Diagramas UML.(A.2 y A.3)
  - b) Implementación del sistema.
  - c) Realización de tests.

## 1.2. Planificación y metodología

Para conseguir los objetivos, la metodología que se ha usado es Scrum [<https://www.atlassian.com/es/agile/scrum>] con *sprints* de 2 semanas, realizando los *sprint review* con el equipo cada martes.

Los puntos 1, 2 y 7a se han realizado durante el inicio del semestre. El siguiente punto completado fue el 3, y paralelamente a este, se realizó el objetivo 6 que se había convertido en objetivo prioritario ya que se quería externalizar, junto con otros procesos, el proceso de monitoreo. Seguidamente, se ejecutó el punto 4 y 5 respectivamente, obteniendo así un nuevo panel.

Por último, se planteó finalizar las tareas 7 (7b y 7c), todo y que gracias a los diagramas UML (A.2 y A.3) se ha conseguido avanzar en la programación y desarrollar, esta última tarea no se ha dado por concluida en el intervalo definido de este trabajo. Tal y como se irá explicando durante todo el informe, este ha sido un proyecto de bajo riesgo en un entorno de trabajo real, lo que supone muchas veces repetir, aplazar o modificar tareas para tratar de contribuir el máximo valor al equipo en ese momento.

## 2 ARQUITECTURA

### 2.1. Niveles ISA 95

ISA 95 [4] es un estándar de comunicación industrial internacional para permitir la comunicación entre diferentes niveles de una empresa de fabricación. Ayuda a que los datos del campo o de la planta se conecten con los sistemas y aplicaciones empresariales. Con los estándares ISA 95, se hace posible la comunicación entre diferentes actores en un proceso de fabricación, como proveedores, línea de fabricación y cadena de suministro. El modelo de datos ISA 95 define 5 niveles diferentes de automatización donde los datos se crean, consumen o tramitan en un entorno de fabricación. Este modelo de datos se basa en UML (lenguaje de modelado unificado) que ya está en uso para el intercambio de datos entre los sistemas MOM (gestión de operaciones de fabricación) y ERP (planificación de recursos empresariales).



Fig. 1: Arquitectura de niveles típica siguiendo los estándares de la ISA 95

**Nivel 0** : Aquí es donde comienza la creación de datos y representa el proceso de producción real.

**Nivel 1** : En este nivel se pueden encontrar sensores y equipos inteligentes que crean o recopilan datos en un proceso de fabricación. Estos datos e inteligencia se utilizan además para manipular los equipos y dispositivos de la línea.

**Nivel 2** : En el nivel dos, la información se emplea para impulsar acciones que impactan el rendimiento del equipo, la línea y la planta. El hardware y el software de control y automatización se unen para monitorear y controlar los procesos físicos.

**Nivel 3** : Esta capa implica flujos de trabajo que cumplen los objetivos de la línea de producción. Implica acciones como la programación, el equilibrio de la carga de trabajo, la optimización de la producción y otros procesos que ayudan a lograr los objetivos de fabricación. Involucra sistemas MOM como MES (sistemas de ejecución de fabricación), sistemas de programación de producción e historiadores.

**Nivel 4** : Esta capa consta de sistemas empresariales que administran las operaciones de fabricación de acuerdo con los objetivos comerciales. ERP, CRM (gestión de relaciones con el cliente), SCM (gestión de la cadena de suministro) son algunos de los sistemas empresariales que utilizan datos del nivel 3 para impulsar las decisiones comerciales.

Las diferentes capas se definen de tal forma que es independiente de la industria o producto fabricado. Puede aplicarse a cualquier tipo de fabricación como discreta, de proceso o híbrida.

## 2.2. Enfoque de Boehringer Ingelheim

En su momento el equipo contempló diferentes soluciones para poder alcanzar los objetivos de digitalización ya que buscaban un software que permitiera una comunicación fluida entre los diferentes niveles del ISA 95 (Figura 1), entre las cuales están:

- OSI Soft PI (Ahora es parte de Aveva) [5]
- SAP MII [6]
- Inmation [1]

Antes de la unión a Aveva, la arquitectura de OSI Soft PI contemplaba dos tipos de servicios, la PI interfaces, que son aquellos servidores para capturar datos, y los servidores que gestionan el almacenamiento de los datos. PI ofrece redundancia a nivel de PI interfaces y servidores para garantizar la continuidad del negocio. La limitación en cuanto a la flexibilidad y la programación la solucionaba ofreciendo un módulo AF (*asset framework*). Sin embargo, este módulo trabajaba bajo una base de datos relacional, lo cual se vió como una limitación. Así pues, para una colección de determinadas funciones es posible que se necesiten varias aplicaciones, con las respectivas licencias. Por todas estas razones PI fue descartado.

SAP MII e Inmation son aplicaciones parecidas, un solo sistema para poder interconectar múltiples sistemas. A la vez, proveen de conexión directa entre la planta y las

aplicaciones corporativas. Inmation disponía de un catálogo mucho mayor de protocolos de conexión, un modelo orientado a servicios versátiles y con la posibilidad de optimizar recursos en servidores. Un punto muy importante fue las diferentes opciones que ofrecía para garantizar la continuidad del negocio. Además, otra razón que hizo descartar SAP MII fue que el sector farmacéutico no era conocido para ellos, mientras que Inmation ya venía tiempo trabajando con empresas farmacéuticas como químicas. Por esta razón finalmente, el equipo se decantó por el software Inmation.

Actualmente, según Owlter [https://www.owler.com/company/inmation], Aveva sigue estando entre los 10 principales competidores de Inmation.

## 2.3. Estructura en Boehringer Ingelheim

Actualmente, hay dos sistemas (basados en el software de Inmation) que están configurados como sistemas globales para ser utilizados en toda la organización. Lo que Boehringer Ingelheim ha conseguido es que los datos relevantes de la planta sean accesibles en tiempo real en una plataforma, integrar todos los datos de producción en la plataforma, conectar a las personas con las fábricas y guardar los datos valiosos de producción para el futuro. Propiciando y generando información de datos, transparencia para la toma rápida de decisiones e incremento en la productividad, la calidad y el cumplimiento del proceso.

## 3 INMATION

System:inmation [1] es un poderoso *middleware* de datos: una infraestructura de datos escalable, corporativa y en tiempo real, que consolida una variedad de diferentes fuentes de datos en un espacio de nombres estandarizado, brindando a los usuarios autorizados acceso a datos operativos granulares.

Las características básicas de System:inmation se enumeran a continuación:

- Adquisición de datos de diferentes tipos de fuentes de datos (por ejemplo, *Classic* OPC, OPC UA, RDBMS, *Files*).
- Consolidación y almacenamiento de datos estandarizados en un repositorio central sin SQL.
- Historización y agregación de datos, gestión de alarmas y eventos, tendencias en tiempo real y más.
- Acceso fácil y seguro a todos los datos utilizando formatos abiertos (por ejemplo, OPC y JSON).

### 3.1. Resumen de componentes

La infraestructura de datos en tiempo real proporcionada por inmation consolida cualquier cantidad de datos operativos en un espacio de nombres estandarizado. La información utiliza una arquitectura orientada a servicios (SOA) basada en clases. El uso de componentes

independientes acoplados libremente permite ajustar la infraestructura de datos en tiempo real para adaptarse con precisión a cualquier necesidad arquitectónica.

A continuación se listan los principales bloques de construcción de la arquitectura y los que van a formar parte de una forma u otra en el proyecto.

- **DataSource:** una *data sources* expone información, que es recuperada por System:inmation. Los “data sources” también se denominan puntos finales.
- **Connector:** un servicio de Windows que recupera datos de una o varias fuentes de datos. El conector también se conoce como el servicio del conector.
- **Core:** un servicio de Windows que recupera datos de uno o varios conectores. El Core procesa y almacena todos los datos en el repositorio. El Core también se conoce como el servicio Core.
- **Data Store Groups:** agrupación de *Data Stores*, estos proporcionan diferentes tipos de almacenes dependiendo de los tipos de datos que se archivan. La cantidad de tipos de datos diferentes producidos por diversas fuentes de datos significa que un diseño de almacenamiento de datos no es suficiente para garantizar un almacenamiento y una recuperación efectivos y eficientes para cada tipo. Por lo tanto, el historiad de inmation MongoDB tiene almacenes de datos separados para diferentes tipos de datos.
- **Repository:** Todos los datos se almacenan en el repositorio central utilizando formatos estándar. System:inmation emplea MongoDB, una base de datos NoSQL gratuita y de código abierto. MongoDB es infinitamente escalable para manejar grandes cantidades de datos en tiempo real.
- **Server:** un servicio de Windows que expone datos de acuerdo con las últimas especificaciones de OPC. Se puede usar cualquier cliente de terceros para conectarse al servidor para leer o escribir datos. El servidor también se conoce como el servicio del servidor.
- **Client Application:** el cliente accede a los datos. DataStudio es la aplicación de cliente que se incluye con el sistema. Las aplicaciones cliente personalizadas pueden acceder a todos los datos desde el repositorio central (JSON) o desde el servidor de información (OPC).
- **I/O Items:** Un elemento con nombre adquirido de un sistema orientado a elementos, como un servidor OPC UA, o creado mediante importaciones de texto de dropzone o secuencias de comandos SQL.
- **Holder Items:** El elemento de titular de datos permite el almacenamiento de datos en tiempo real. Puede ser configurado por Clientes OPC externos.

### 3.2. Arquitectura y Escalabilidad

Se mostrará un escenario de ejemplo a escala empresarial para explicar resumidamente la arquitectura que se usa en el entorno de producción.

En un caso con alta complejidad de los requisitos de infraestructura y la cantidad de datos que se almacenan, como es el sistema de producción de Boehringer Ingelheim, System:inmation se ajusta para adaptarse a cualquier escala de entorno de datos en tiempo real. La escalabilidad completa se logra agregando instancias de componentes System:inmation adicionales a una de las diferentes capas de arquitectura. La cantidad de instancias de servicio de *Connector* e instancias de servicio de *Core*, así como la escala de MongoDB, se ajustan al tamaño y al requisito de rendimiento.

A medida que crecen los servidores OPC y elementos de datos son agregados al sistema, la capacidad de flujo de datos se garantiza a través de una creciente infraestructura de capas de componentes piramidales. Al mismo tiempo, los componentes individuales, en el caso de Boehringer Ingelheim, están repartidos por todo el mundo a través de la conexión a internet. Por lo tanto, cada tipo de entorno de datos en tiempo real puede administrarse y configurarse de manera centralizada.

Cada conexión de red individual entre componentes de información solo requiere un único puerto TCP. En consecuencia, las reglas del cortafuegos se configuran y administran fácilmente para permitir que los componentes internos envíen y reciban datos.

El siguiente esquema muestra una posible disposición de los componentes del sistema en un sistema a gran escala. El entorno System:inmation a gran escala puede usar muchas ramas donde cada componente administra un subconjunto de componentes de la siguiente capa, lo que finalmente da como resultado que System:inmation consolide una gran cantidad de *Endpoints* (p.e, servidores OPC) en un único sistema grande.

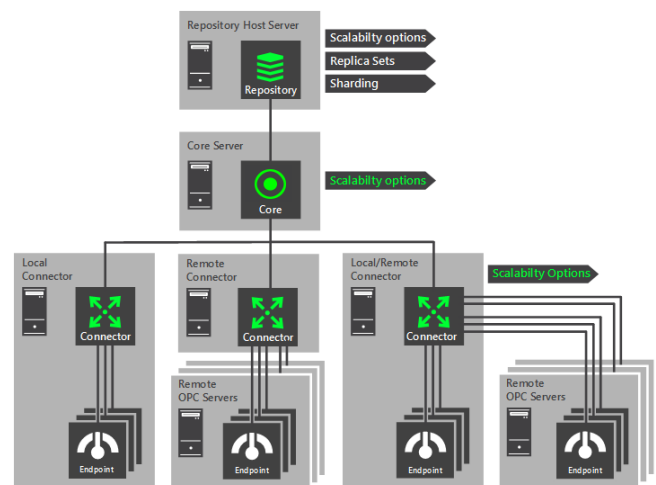


Fig. 2: Escenario escalable de empresa

Al requerir un sistema de escala empresarial, todos los sistemas están escalables, y estos funcionan con un *Master Core* que puede englobar tanto otro *Core* llamado *Local Core* como *Connectors*, *Data Store Groups*, *Data Sources* y dentro de estos diferentes *I/O Items* y *Data Holder Items*. Y los *Local Cores* pueden englobar el mismo tipo de objetos que el *Master Core* a excepción de otro *core*.

Esta distribución es la típica arquitectura de un sistema donde se busca descentralizar los datos y el volumen de cómputo así como obtener un balanceo de carga óptimo para el correcto funcionamiento del sistema de producción.

### 3.3. Herramientas

Para realizar el desarrollo del proyecto se ha utilizado la aplicación de cliente de DataStudio y la *WebApp* (WebStudio) proporcionadas ambas por la plataforma de Inmation. Usando Lua [2] como lenguaje de programación y formatos de texto tipo JSON para la visualización de los paneles.

#### 3.3.1. DataStudio

- Es la principal aplicación de cliente para System:inmation, y está diseñada para ser una interfaz segura y única para acceder a toda su red de fuentes de datos.
- Proporciona un acceso completo y rápido a los datos históricos y en tiempo real con un conjunto de opciones de visualización dinámicas y efectivas.
- Permite configurar y controlar el espacio de trabajo, como también una gestión de los perfiles de usuario y la seguridad.

#### 3.3.2. WebStudio

WebStudio es una aplicación web que permite compilar un tablero de widgets (basados en estructura JSON) de respuesta dinámica para visualizar los datos del sistema de inmation en un navegador web. WebStudio utiliza la API web para recuperar datos del sistema y permite el acceso a sus datos desde cualquier lugar con conexión a Internet.

### 3.4. Entorno de ejemplo

Se supondrá un entorno de ejemplo extraído del entorno de *Training*. Durante todo el informe, las fotos estarán basadas en el entorno de trabajo de la figura 3.

## 4 MONITORIZACIÓN: PUNTO DE PARTIDA

### 4.1. Contexto

La tarea/proceso <sup>1</sup> de monitoreo se realiza una vez por semana, y hasta agosto de 2021 tenía una duración de un día laboral por semana (teniendo en cuenta que la realiza una persona, y que se requiere a otra para su verificación). Esta revisión se hacía a través de DataStudio la cual permite tener una visión global de todo el entorno de producción y acceder a las diferentes métricas de los objetos como el acceso a los diferentes registros producidos por el sistema. Las métricas garantizarán que los datos se están transmitiendo de forma correcta y de que no hay ningún tipo de sobrecarga o problema que pueda dañar el sistema en un futuro próximo. Inicialmente, se monitorizaban las métricas

<sup>1</sup>Es llamada también tarea porque dentro del equipo se trata como una tarea. No confundir con las tareas que sirven para mejorar la tarea/proceso de monitoreo.

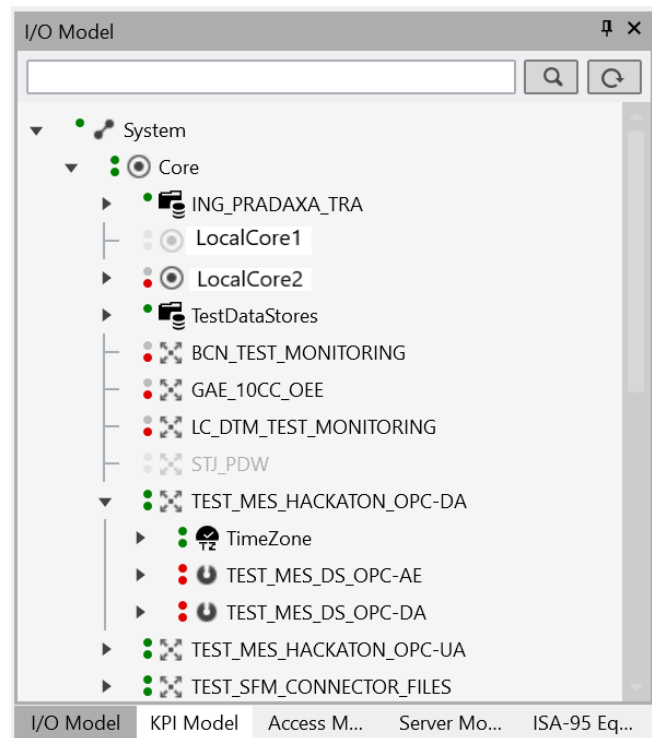


Fig. 3: Árbol ejemplo del entorno de *Training* visto desde el aplicativo de DataStudio

de los siguientes objetos: *System*, *Master and Local Cores*, *Connectors*, *DataSources*, *Data Store Groups*.

### 4.2. Proceso manual

Inicialmente, se tenían que comprobar las métricas de rendimiento y de transmisión de datos de los objetos de forma manual, accediendo al aplicativo DataStudio y navegando por él. Una tarea muy tediosa y que daba lugar a la posibilidad de errores humanos. Paralelamente, se iba rellenando un documento en la plataforma de Confluence donde quedaba reflejado que todos los objetos estaban en buen estado, y los que no, necesitaban ser reportados específicamente en otra sección.

De igual manera, los registros también se buscaban manualmente en el aplicativo DataStudio, y teniendo en cuenta que había semanas que se tenían 99.000 mensajes en los cuales participaban 100 objetos diferentes, esto se traducía en otra faena demasiado tediosa en la que se tenía que filtrar de forma manual y apuntar los mensajes únicos que iban apareciendo. Agrupando por cada mensaje todos los objetos que en algún momento de la semana habían emitido ese error.

Una vez se tenían los reportes de todo lo que estaba en un estado no deseado, se procedía a avisar a los responsables de cada fábrica mediante correo electrónico. Haciendo aún más tediosa la tarea, ya que se tenía que tener un control del flujo de la comunicación entre el equipo que realizaba el monitoreo y cada responsable de departamento de ingeniería de la fábrica en cuestión.

### 4.3. Proceso semiautomático

Gracias a WebStudio y a la programación realizada con Lua en el back-end, se ha conseguido realizar un programa que obtenga todas las métricas necesarias de los diferentes objetos. Entonces, los compañeros del front-end con WebStudio diseñaron un panel donde todos los miembros de los equipos pueden ver de forma resumida todos los objetos del sistema de producción (Figura 4).

Name	Type	Status	State	Comm.
Core	Core	BAD	●	●
TEST_MES_HACKATON_OPC-DA	Connector	PERFECT	●	●
TEST_MES_DS_OPC-AE	Data Source	●	●	●
TEST_MES_DS_OPC-DA	Data Source	●	●	●
TEST_MES_HACKATON_OPC-UA	Connector	MEDIUM	●	●
TEST_SFM_CONNECTOR_FILES	Connector	PERFECT	●	●

Name	Type
Core	Core
ING_PRADAXA_TRA	Data Store Group
ING_PRADAXA_TRA_AE_01	Custom Event Data Store
ING_PRADAXA_TRA_PT_01	Custom Prod. Tracking Data Store
ING_PRADAXA_TRA_TS_03	Custom Time Series Data Store

Fig. 4: Panel de visualización de la estructura de árbol de la figura 3 a través de WebStudio. Ir al apéndice A.1 para ver el panel completo

Además, por la parte que corresponde a los registros del sistema, se desarrolló un complemento de Excel programado en VBA ( *Visual Basic for Application* ) [3] que evita el filtrado de los registros manualmente. Con la exportación de los registros de la semana deseada a Excel y la ejecución del complemento, estos registros se ordenan automáticamente, haciendo mucho más fácil su gestión.

Estas dos mejoras significativas redujeron la tarea del monitoreo a la mitad de tiempo. Ya que aún había que mirar el panel y apuntar manualmente en la hoja de Confluence. Además, el análisis de los registros y la comunicación por correo electrónico con los responsables de cada *site* también han seguido ocupando parte del tiempo del proceso de monitoreo.

## 5 MONITORIZACIÓN: CONTINUACIÓN DEL PROYECTO

A continuación, se resumirá cada tarea realizada para la continuación del proyecto general de empresa con una breve explicación de lo realizado en la misma y el valor aportado de esta en el proyecto.

### 5.1. Macro de Excel

La macro de Excel previamente implementada tenía errores en la captación de las fechas. La solución a esto

fue optar por otra manera de obtener las fechas y en vez de utilizar funciones de filtrado, que Excel proporcionaba y que no acababan de adaptarse a lo que se deseaba, (había problemas con la primera y última fecha del set de datos bruto) se definieron las funciones de búsqueda y se hizo el código más robusto.

Cada registro extraído de DataStudio tiene los siguientes campos: Hora y fecha, Gravedad, Mensaje, Detalle y Ruta del objeto en el árbol del DataStudio (p.e, registro generado por Core1/Connector2/DataSource6 ).

Así pues, en la primera hoja se organizan los registros por mensaje, sacando así los mensajes únicos de la franja de tiempo seleccionada. Y por cada mensaje se listan los objetos que han estado involucrados. Por otro lado, se ha añadido una segunda hoja que muestra una forma diferente de organizar los datos. Se organizan por objeto, es decir, ahora el número de filas de esta tabla será el número de objetos por el número de veces que ese objeto tenga un mensaje diferente <sup>2</sup>.

#### 5.1.1. Dificultades

La principal dificultad fue desarrollar la segunda hoja. Aunque desde un inicio parecían dos algoritmos muy parecidos, el hecho de registrar cada objeto y después por cada objeto contemplar cuantos mensajes diferentes tenía, aumentó la complejidad del algoritmo a  $O(N^2)$ .

Otra dificultad surgió cuando el equipo quiso externalizar este proceso. El hecho de que la externalización se realizaría a gente fuera de España, significaba que tenían la herramienta de Excel 365 en inglés, y se tuvo que adaptar y generalizar partes del código.

#### 5.1.2. Valor

Tener una ordenación de los registros permite un ahorro de tiempo a la hora de buscar incidencias ocurridas. Cuando un *core* se ha desconectado, en vez de dirigirse a los registros del DataStudio, se extraían los que interesaban y se analizaban en Excel. Además, esta herramienta permitía una mayor comunicación entre los miembros del equipo para ayudar a encontrar los problemas de las incidencias.

## 5.2. Reportes en OneDrive

Confluence es una herramienta muy útil para los entornos de trabajo ágiles, como puede ser orientado a la documentación de procesos, páginas de contacto, documentos informativos, técnicos, etc. Pero, no era el adecuado para un proceso orientado a realizarse semanalmente y de tamaño considerablemente grande y en continuo crecimiento (cada vez se iban integrando más fábricas). El resultado de esto era un proceso de edición del documento muy lento y la desesperación por parte del miembro del equipo que realizase el monitoreo esa semana.

<sup>2</sup>Tanto en la primera hoja como en la segunda, el campo de detalle no es tenido en cuenta, ya que se extrajo para tenerlo en cuenta a la hora de analizarlos e ir a los datos en bruto, para así tener una idea más clara de que lo estaba sucediendo

Como solución, se trasladó todo el sistema de reportes al sistema de carpetas compartido de OneDrive. Para ello fue necesario añadir una hoja más en Excel que constase del reporte de las métricas de los objetos del sistema, substituyendo lo realizado en Confluence, y organizar el sistema de carpetas para que el entorno de trabajo sea cómodo para los miembros del equipo.

### 5.2.1. Dificultades

Cuando el proceso se externalizó, también se tuvieron que cambiar ciertas partes del código para generalizarlo. Además, al ser un proceso totalmente nuevo, tuvo que ser explicado a los miembros del equipo y esto llevó un pequeño tiempo de adaptación.

### 5.2.2. Valor

Con esta nueva metodología, se conseguía tener un sistema de archivos más ordenado con todos los monitoreos ejecutados, y se podían llegar a realizar comparaciones mucho más rápido.

## 5.3. Supresión de falsos errores

En la primera versión, los valores estipulados para las métricas de *Backlog* y de *Buffer Disk*, estaban limitados a 50 *counts* (unidad de medida en Inmation). Ya que en las pautas a seguir cuando el monitoreo se realizaba de manualmente se indicaba que estos dos parámetros tenían que ser 0 o próximos a 0.

En la nueva versión, se decide implementar unos rangos mínimos de seguridad y después controlar este crecimiento obteniendo los parámetros medios del día y del mes. Haciendo la comparativa, si la media del día superaba con creces la del mes, quería decir que esa máquina no estaba trabajando de manera natural y correcta.

### 5.3.1. Dificultades

El principal problema era que se desconocía la naturaleza de los parámetros (*Backlog* y *Buffer Disk* y que esta dependía de la máquina (Pe. No es lo mismo una máquina que procesa 10 lotes/h que una que procesa 1000 lotes/h). Así pues, a medida que fueron apareciendo diferentes incidencias, se vió que este valor podía ser relativamente alto (1000), pero lo más importante era que este no aumentara exponencialmente.

### 5.3.2. Valor

Esta solución proporciona una mayor calidad de los datos expuestos en el panel, y una mayor claridad a la hora de realizar la tarea del monitoreo. Además, para conseguir un proceso homogéneo y evitar fallas en el sistema, en el documento oficial de Boehringer Ingelheim se recalca que cada vez que haya un indicador de error, este tiene que ser comprobado en DataStudio. Por lo que si se evitan falsos errores que tienen que ser revisados, se está mejorando el rendimiento del proceso.

## 5.4. Extracción de los *I/O Items* y los *Data Holders*

Un punto importante en el proceso de monitoreo es conocer que ciertos objetos que van creando e incorporando *tags* al sistema estén funcionando correctamente, concretamente los *DataSource*, los *I/O Items* y los *Data Holder Items*. Por esta razón, se ha implementado una función nueva que recolecte el estado de estos dos últimos objetos, y que en el panel principal aparezcan los que están en estado correcto y erróneo por cada *Connector* y el total. Además, si se quiere mirar con detalle el *Connector*, el usuario podrá ver todos los ítems y los distintos estados en los que se encuentran (error, correcto, advertencia, neutral, sin confirmar, vacío).

### 5.4.1. Dificultades

Definir la metodología de extracción fue la decisión clave a realizar en esta tarea. Se podía ejecutar por consulta a la base de datos (NoSQL), proporcionando un código mucho más limpio a cambio de una ejecución relativamente lenta (segundos). O bien, hacer una función recursiva que explorase todo el árbol, siendo mucho más rápida pero a la vez más compleja.

Finalmente, se decidió por la consulta a la base de datos, ya que el tiempo no era un problema, puesto que era una función que no iba a ser llamada en intervalos cortos de tiempo. De esta manera, se ahorró tiempo de desarrollo de código.

### 5.4.2. Valor

Realizó ser una tarea crucial, puesto que el requerimiento de obtener el estado de estos objetos se tenía desde el inicio en el proceso manual, pero por diferentes circunstancias la tarea se decidió posponer hasta que la nueva funcionalidad ha sido añadida. Por otro lado, cuando se generan los registros de error, es un valor añadido poder ver en directo el estado del objeto que está generando el error.

## 5.5. Nuevo panel en la Web API

Debido a que los miembros del equipo del front-end tenían muchas tareas, se decidió desarrollar el diseño del panel en la plataforma de WebStudio. Una nueva vista fue proporcionada junto con las vistas de las nuevas funcionalidades de la tarea 5.4.

### 5.5.1. Dificultades

Debido a que el entorno del front-end era totalmente desconocido, fueron necesarias algunas reuniones de soporte con los compañeros. Además, hay que añadir los problemas con las versiones del propio Inmation. Habían *widgets* que funcionaban en la versión antigua, pero no en la nueva. Lo que compartaba que se realizasen reuniones con los responsables de Inmation para solucionar estos problemas.

### 5.5.2. Valor

La nueva vista era más clara, y más fácil de manejar para el usuario.



## 5.6. Documentación del proceso

El requerimiento más importante para externalizar el proceso era la realización de un documento que siguiese todos los procesos legales requeridos por la empresa de Boehringer Ingelheim.

Al tratarse de un documento oficial, la estructura y el contenido se han diseñados con idea propia, pero ha sido necesaria la supervisión y aprobación del System Lead del equipo.

### 5.6.1. Dificultades

Al tratarse de un proceso en el cual el becario ha estado trabajando la mayor parte de su estancia en prácticas, la primera versión del documento era escasa en algunas explicaciones. Gracias a los compañeros que usaron este documento, se pudo realizar una segunda versión mucho más clara.

### 5.6.2. Valor

Con el proceso documentado se ha conseguido externalizar el proceso y que la tarea de monitoreo que realizaba el equipo cambiase a una sola evaluación global de estos reportes obtenidos por el agente externo. Además, al tratarse de un entorno controlado y tener el documento calificado, permite seguir en línea con la normalización de todos los procesos dentro del equipo.

## 5.7. Automatización de los reportes

La automatización de los reportes gracias a la herramienta de Inmation es parte de la solución para la desaparición por completo de la interacción interna del equipo en la tarea del monitoreo. Consiste en que se ejecute el monitoreo de forma periódica y que de forma automática el reporte obtenido sea enviado al responsable correspondiente.

### 5.7.1. Dificultades

Las principales dificultades han estado siendo la necesidad de usar de forma más completa el software de Inmation. Ya que para formar la estructura de árbol se han usado diferentes tipos de objetos los cuales se ha tenido que consultar la documentación.

Además, debido al rediseño necesario de la tarea y la interrupción de otras más prioritarias que daban valor inmediato al equipo, esta ha ido siendo aplazada. Impidiendo la finalización completa y alargando la estimación para que esté funcional en el entorno de producción.

### 5.7.2. Valor

El valor añadido es máximo, ya que no solo se elimina el tiempo dedicado de la tarea y se hace completamente automática, sino que además se aumenta la precisión de forma notable puesto que no es un ser humano el que la realiza.

TABLA 1: COSTES Y VALOR DE LAS DIFERENTES FASES DE LA TAREA DE MONITOREO

	F1	F2	F3	F4	F5
Reportes/semana	1	1	1	14	14
Horas/semana	8	4	1	1 + 7(ext)	1(ext)

## 6 IMPACTO DE LOS RESULTADOS

El proyecto realizado se encuentra dentro de los proyectos de bajo riesgo en una empresa, ya que, en este caso, es un proyecto que no afecta directamente al negocio ni a la producción. Por lo que desde un inicio no se han hecho cálculos exhaustivos previos al desarrollo del proyecto, sin embargo, sí que se ha ido discutiendo con los superiores las diferentes tareas y su viabilidad.

Puesto que este proyecto ha ido evolucionando durante todo su transcurso, se puede decir que la tarea del monitoreo se ha ido encontrando en diferentes fases. Estas fases se van a diferenciar principalmente por el coste de la tarea y por el valor aportado de la tarea (Costes y valor medido en reportes/semana en la tabla 1):

- Fase 1 (F1): Realización de la tarea del monitoreo totalmente manual. (abril/2021 - octubre/2021)
- Fase 2 (F2): Realización de la tarea del monitoreo de forma semiautomática al iniciar el proyecto. Se dispone del primer panel de visualización de los objetos. (noviembre/2021 - febrero/2022)
- Fase 3 (F3): Realización de la tarea del monitoreo de forma semiautomática antes de finalizar el proyecto. Se realiza el proceso con la nueva metodología a través de la plantilla de Excel. (marzo/2022 - mayo/2022).
- Fase 4 (F4): La realización de la tarea del monitoreo se externaliza. Se multiplica por 14 el número de reportes semanales realizados, puesto que la empresa externa realiza 2 diarios (30min/reporte). Esta externalización en la tabla 1 se indica con las letras "ext". Para el equipo interno la tarea en este momento solo consiste en realizar un análisis general de los 14 reportes obtenidos y reportar mediante correo electrónico a los responsables lo más destacable. (junio/2022 - septiembre/2022)
- Fase 5 (F5): Se hace la suposición de que la realización de la tarea del monitoreo junto con el acto de reportar a los responsables lo más destacable se automatiza por completo a través de la plataforma de Inmation. Y además, se mantiene externalizada la tarea de obtención de los registros. Esta última, sería realizada por la empresa externa. Cabe destacar que esta suposición se realiza hasta abril de 2023 para concluir el análisis de la tarea a los 2 años exactos. Sin embargo, esta se va a seguir desarrollando y mejorando. El equipo ya está trabajando para trasladar la obtención de registros a una plataforma llamada Splunk [7] que está enfocada principalmente en el monitoreo de sistemas. (octubre/2022 - abril/2023).

Se ha realizado a posteriori del proyecto un análisis aproximado de los costes del proyecto y del tiempo necesario para que éste se amortice. Para realizar este estudio, se hará la comparativa de la tarea evolutiva pasando por las diferentes fases, con la tarea inicial, ya que se quiere saber si todo el esfuerzo y tiempo dedicado ha aportado valor respecto a la tarea inicial.

## 6.1. Suposición de costes de los recursos humanos

En este estudio se tendrá en cuenta el salario de un ingeniero medio en España, 38.000€ [<https://es.calcuworld.com/cuantos/cuanto-cobra-un-ingeniero/>] y el salario del becario medio en el sector de ingeniería en España, 9.600€ [<https://es.gowork.com/blog/12-cuanto-cobra-un-becario-en-espana-para-cada-sector/>].

Sin olvidar que se tiene que calcular cual es el coste real para la empresa el contratar un ingeniero. Según la página web [<https://es.talent.com/tax-calculator>], con la cantidad definida por el ingeniero medio en España, el coste total es de 49.362€. Por otro lado, no ocurre lo mismo para el becario, ya que este es un convenio de colaboración entre la empresa, la universidad y el estudiante. Por lo tanto, suponiendo 22 días laborales al mes, el ingeniero cuesta a la empresa 23,38€/h y el becario 4,55€/h.

Además, se hará la suposición de forma mucho más general y sin fuentes que lo puedan corroborar, ya que se trata de información confidencial, de que los servicios que se han externalizado tienen un coste de 25€/h (principalmente son técnicos/ingenieros).

## 6.2. Análisis aproximado de los costes

Una vez hechos los cálculos correspondientes, sabiendo la fase en la que se encuentra la tarea a lo largo de estos dos años, las horas de trabajo de la tarea al mes y el coste de los recursos humanos, se obtienen dos gráficas representativas.

### 6.2.1. Evolución de la tarea de monitoreo

La primera gráfica se encuentra en el apéndice A.4 y muestra como han ido evolucionando los diferentes costes (tarea y becario).

El coste del becario empieza en agosto de 2021 con un valor de 800€ puesto que ese mes se estaba trabajando a jornada completa y fue a principios de este cuando se empezó el proyecto de mejora del monitoreo. Los demás meses se ha estado trabajando en el proyecto a media jornada.

La línea base (línea gris) horizontal marca el coste inicial mensual de la tarea sin aplicarle ningún tipo de mejora y va a ser la base para las diferentes mejoras que se apliquen. En el momento en el que se empieza a desarrollar el proyecto, los costes sobrepasan considerablemente la línea base; se está realizando una inversión para la futura

mejora de la tarea. En la F2, donde la tarea se reduce a la mitad, se observa que el coste del becario sumado al de la tarea sobrepasa ligeramente el coste inicial de la tarea en la F1 y una vez llegada a la F3 ya se empieza a tener amortización en el proyecto puesto que la tarea reduce considerablemente su coste.

Al haber ejecutado todas las tareas necesarias para externalizar el proyecto, en la F4 se ve como la tarea aumenta considerablemente. Esto es debido a la externalización de la tarea y a que ahora se realizan 14 reportes/semana, más una tarea de análisis de los 14 reportes obtenidos, proporcionando un valor añadido a la tarea muy importante para el equipo. Esto, sumado al coste de desarrollo del becario pone al proyecto de nuevo por encima de la línea base, ya que se continua el desarrollo del proyecto.

A partir de la F5 se realizan suposiciones. Se prevee que en octubre de 2022 el proyecto de automatización finalice, por lo cual los costes del becario desaparecen y parte de la automatización queda completa. Es en esta F5 donde se volverá a obtener amortización del proyecto llegando a amortizarlo por completo en enero de 2023.

### 6.2.2. Cálculo de la amortización

La amortización del proyecto se podría obtener observando la figura del apéndice A.4 y calculando el sumatorio de las áreas por encima y por debajo de la línea base. Sin embargo, para obtener el punto exacto en el cual el proyecto se encuentra amortizado se ha realizado la gráfica que se encuentra en el apéndice A.5.

Gracias a los costes acumulativos se puede observar como la tarea de monitoreo comparada con la tarea de monitoreo en la F1 empieza a distarse al iniciarse el proyecto (primera fase de inversión al involucrar al becario en el proyecto) y esta se mantiene por encima hasta enero de 2023.

Se observa en la gráfica del apéndice A.5 tal y como se ha comentado al principio de este apartado, que se trata de un proyecto de bajo riesgo, puesto que la diferencia monetaria de la tarea de monitoreo aplicando las mejoras comparadas con la tarea de monitoreo sin aplicar las mejoras (F1) no es muy considerable.

En la F4 si que se puede observar como la distancia entre las dos líneas dista un poco más debido al aumento en el número de reportes tal y como se ha comentado anteriormente. En octubre, sin embargo, se observa como la línea se allana de forma considerable puesto que solo se tiene el coste de la obtención de los registros de forma externa. Y es a partir de ahí, cuando se empieza a amortizar por completo el proyecto.

Así pues tal y como se ha mencionado antes, el proyecto se amortiza completamente en enero de 2023 y en abril de 2023 se obtendría un beneficio, no solo económico puesto que hay que recordar que el valor del proceso ha sido multiplicado por 14, de 1.815,92€.

## 7 CONCLUSIONES

Trabajar en un entorno controlado de producción ayuda a ver situaciones que en otros proyectos no se dan. El hecho añadido de formar parte de un proyecto real, muestra lo importante que es que las tareas que se estén realizando tengan constantemente un valor, ya que muchas veces se han tenido que atrasar tareas para dedicarle la máxima atención a las más prioritarias y a las que iban a aportar valor más rápido al equipo.

Gracias al trabajo realizado se ha conseguido optimizar un proceso de monitoreo inicialmente muy costoso. Mejorar el panel de visualización y su funcionalidad, añadir funcionalidades extra en los reportes para relacionar mejor los datos, la mejora del entorno para trabajar con los reportes obtenidos y externalizar el proceso, son las acciones que han permitido que los miembros del equipo enfoquen más parte de su tiempo en desarrollar y que el proceso sea mucho más valioso, óptimo y preciso.

Sin embargo, debido a las dificultades surgidas durante el desarrollo del proyecto, no se ha llegado a finalizar la automatización completa del proceso de monitoreo, dejando esta última tarea a mitad de ejecución con la expectativa de terminarse en septiembre de 2022.

Además, tal y como se ha comentado anteriormente, el equipo ya está pensando en la plataforma Splunk para que sumado a la automatización ya realizada se consiga un sistema de control total e independiente a la interacción de los miembros del equipo. Permitiendo al equipo de desarrollo centrarse en las principales tareas y ayudando al equipo a tener una visión constante del estado y salud del entorno productivo, un punto muy importante en una industria farmacéutica como es Boehringer Ingelheim.

## AGRADECIMIENTOS

Primero de todo, quiero agradecer el apoyo otorgado por mi familia y amigos que en todo momento y a su manera, me han brindado y facilitado la ayuda que he necesitado.

Por otro lado, agradecer a mi tutor del TFG, Lluís Ribas Xirgo por guiarme de forma tan clara y precisa la tutoría de este trabajo ofreciendo en todo momento su disponibilidad y apoyo.

Por último, este trabajo no hubiera sido posible sin la ayuda de todos mis compañeros/as de trabajo los/las cuales han estado constantemente a mi lado proporcionándome todo tipo de ayuda.

En especial, quiero agradecer a mi tutora de prácticas de empresa en Boehringer Ingelheim y responsable del equipo, Montse Portella, al *Product Owner* del back-end, Sergi Alà, al *Product Owner* del front-end, Marco Gallo, y al *System lead*, Josep Reverter, que me han ayudado en todo momento y han confiado en mi y en mi criterio para realizar con gran libertad la gestión y realización del proyecto de monitoreo. Solo añadir que en el equipo era conocido por ser el "Monitoring Man".

## REFERENCIAS

- [1] "Industrial Information Management System & Software — inmation". Industrial Information Management System & Software — inmation. <https://www.inmation.com/> (accedido el 6 de marzo de 2022).
- [2] "Manual de Referencia de Lua 5.1 - contenido". The Programming Language Lua. <https://www.lua.org/manual/5.1/es/> (accedido el 6 de marzo de 2022).
- [3] "Referencia de Visual Basic para Aplicaciones (VBA) para Office". Developer tools, technical documentation and coding examples — Microsoft Docs. <https://docs.microsoft.com/es-es/office/vba/api/overview/> (accedido el 6 de marzo de 2022).
- [4] "What is ISA 95 — Oden Technologies". Oden Technologies. <https://oden.io/glossary/isa-95/> (accedido el 1 de junio de 2022).
- [5] "PI System: conexión de datos, operaciones y personas — OSISoft". OSISoft — Inteligencia operativa — PI System. <https://www.osisoft.es/pi-system/> (accedido el 3 de junio de 2022).
- [6] "SAP MII". SAP Support Portal Home. <https://support.sap.com/en/alm/sap-focused-run/expert-portal/integration-cloud-monitoring/sap-mii.html> (accedido el 3 de junio de 2022).
- [7] "Splunk — The Data Platform for the Hybrid World". Splunk. <https://www.splunk.com/> (accedido el 8 de junio de 2022).

## APÉNDICE

### A.1. Visualización completa de la primera versión del panel en WebStudio

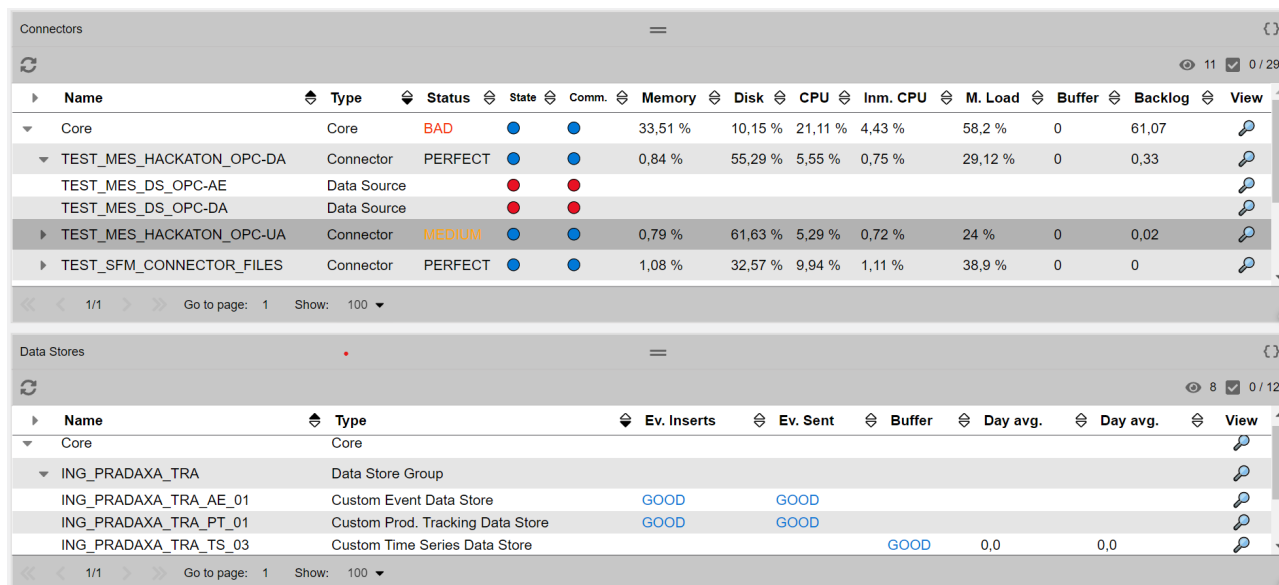


Fig. 5: Panel de visualización de los objetos a monitorear con sus respectivas métricas en el entorno de *Training*

### A.2. Diagrama UML de casos de uso de la tarea “5.7. Automatización de los reportes”

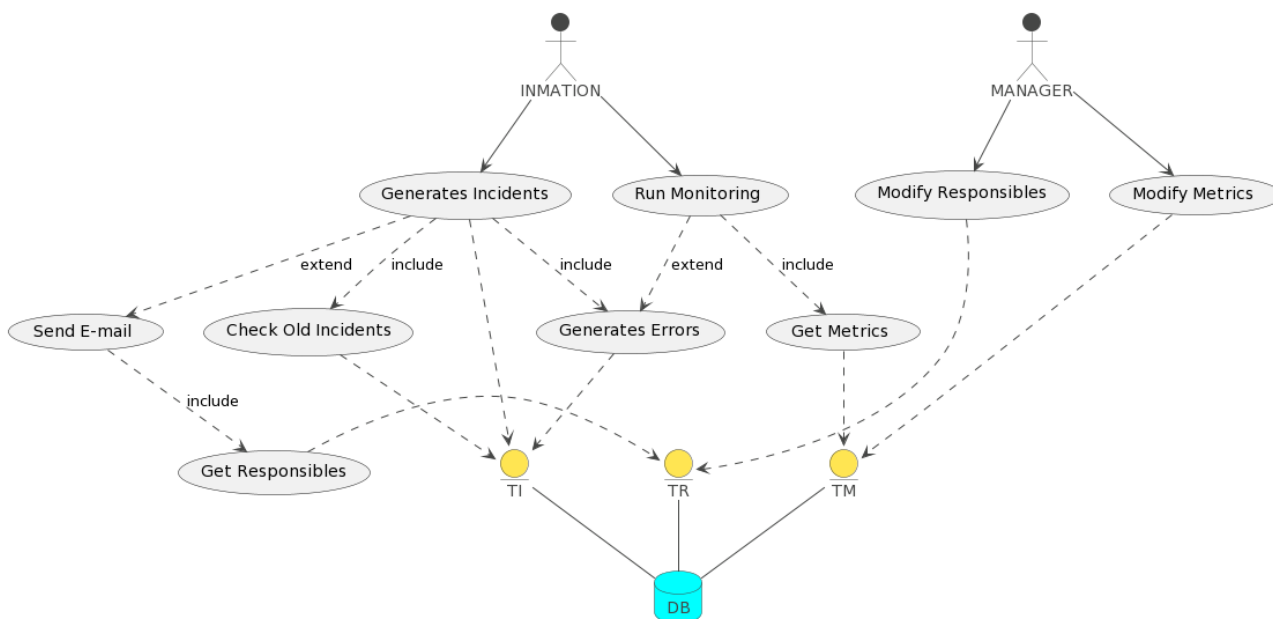


Fig. 6: Diagrama UML de casos de uso para el diseño de las funcionalidades del sistema de reportes automáticos usado para transmitir una idea clara de lo que se quiere realizar a los miembros del equipo

### A.3. Diagrama UML de secuencia de la tarea “ 5.7. Automatización de los reportes”

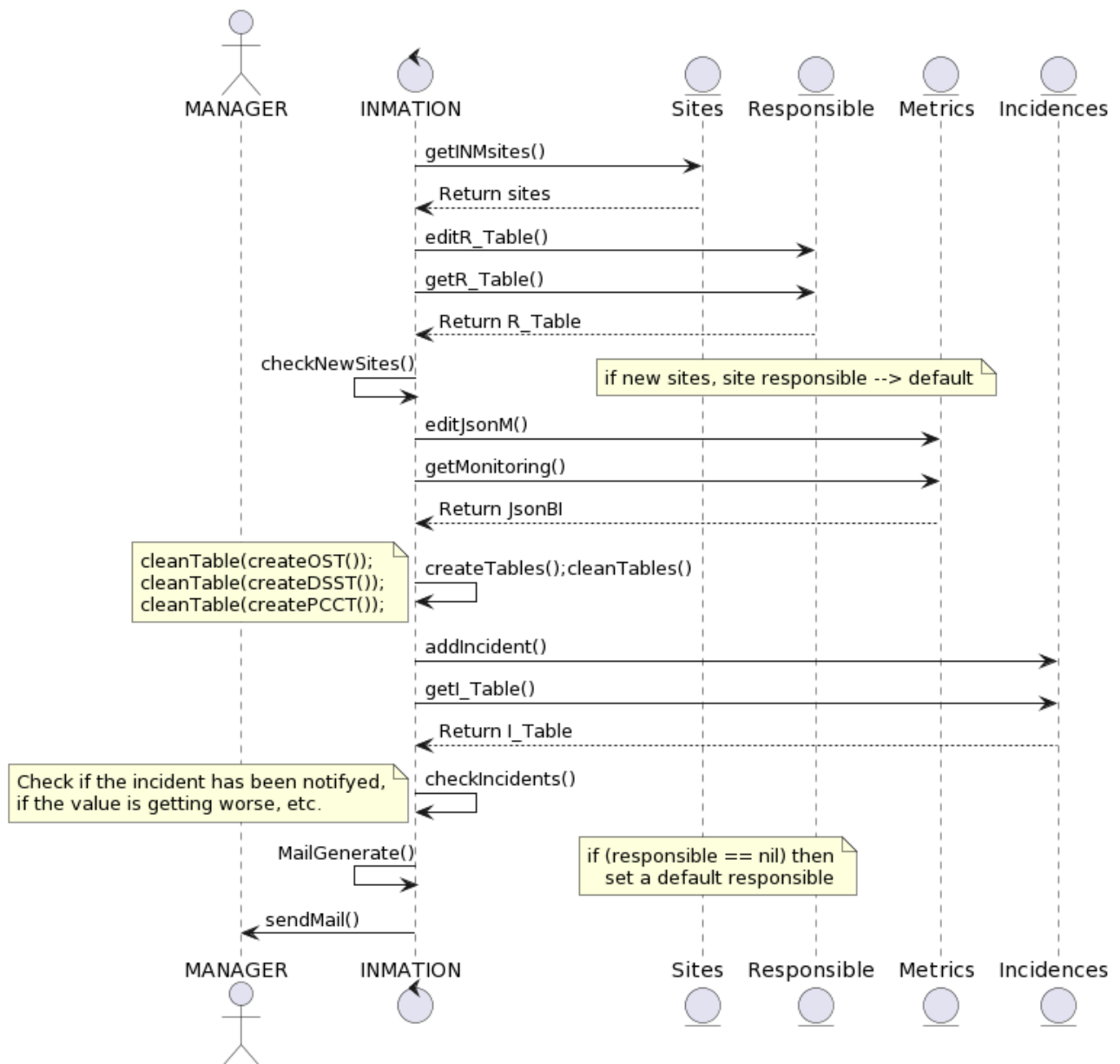


Fig. 7: Diagrama UML de secuencia para el diseño base del protocolo de comunicación usado como pauta y base para la programación y el montaje de la estructura de objetos de Inmation

### A.4. Evolución de los costes en la tarea de monitoreo

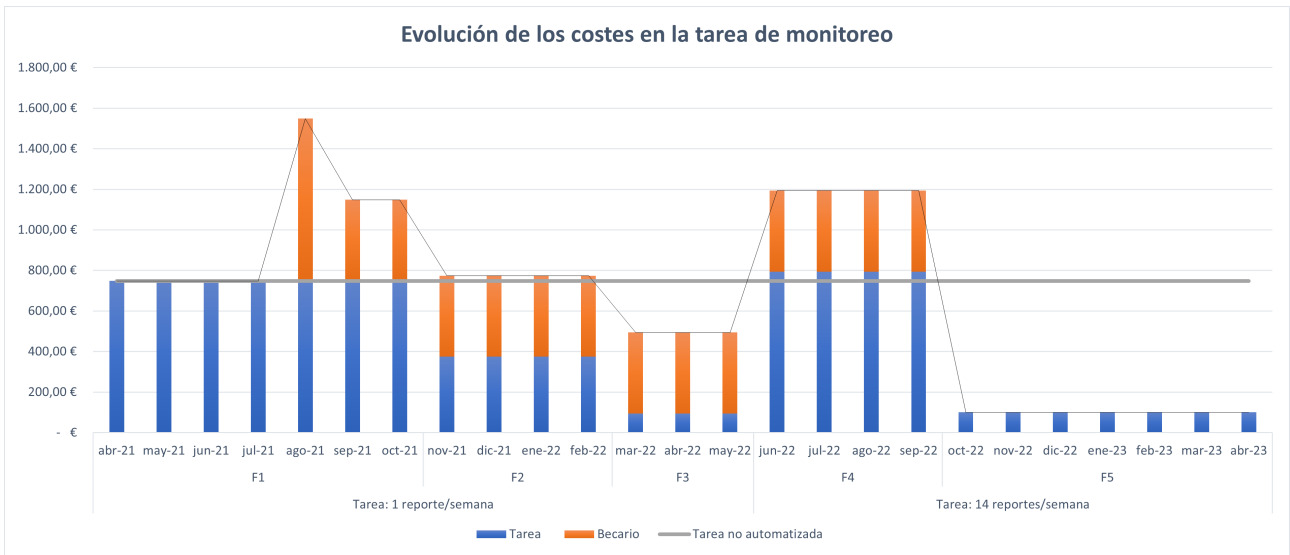


Fig. 8: Transcurso de los costes de la tarea de monitoreo y del becario desde abril de 2021 hasta abril de 2023

### A.5. Previsión de amortización a través de los costes acumulativos



Fig. 9: Amortización de la tarea de monitoreo desde abril de 2021 hasta abril de 2023