



# **ESCUELA POLITÉCNICA DE INGENIERÍA DE GIJÓN**

## **GRADO EN INGENIERÍA INFORMÁTICA EN TECNOLOGÍAS DE LA INFORMACIÓN**

### **Lenguajes y Sistemas Informáticos**

### **TRABAJO FIN DE GRADO/MÁSTER Nº ???**

### **Explotación, integración y visualización de múltiples fuentes de datos mediante un Data Lake**

**Mier Montoto, Juan Francisco**

#### **TUTORES:**

**D. Augusto Alonso, Cristian**

**D. Morán Barbón, Jesús**

**D. Vázquez Faes, Eduardo**

**FECHA: junio 2024**

# Índice de contenido

<b>Índice de contenido</b>	<b>1</b>
<b>Índice de figuras</b>	<b>2</b>
<b>1. Introducción</b>	<b>3</b>
1.1. Antecedentes . . . . .	3
1.1.1. Análisis y visualización de datos . . . . .	3
1.2. Motivación . . . . .	5
1.3. Finalidad del proyecto . . . . .	5
1.4. La empresa . . . . .	6
<b>2. Fundamento teórico</b>	<b>7</b>
2.1. Data lake . . . . .	7
2.2. Procesos ETL . . . . .	7
2.2.1. Definición . . . . .	7
2.2.2. Características . . . . .	7
2.2.3. Funcionamiento . . . . .	8
2.2.4. Alternativas . . . . .	9
2.3. Dashboards . . . . .	10
<b>3. Planificación</b>	<b>12</b>
3.1. Metodología . . . . .	12
3.1.1. Scrum . . . . .	12
3.1.2. Visualización de la planificación . . . . .	12
3.1.3. Comunicación . . . . .	13
3.2. Presupuesto . . . . .	14
<b>4. Análisis</b>	<b>15</b>
4.1. Partes interesadas (stakeholders) . . . . .	15
4.2. Definición del sistema . . . . .	15
4.3. Análisis de alternativas . . . . .	15
<b>5. Diseño del sistema</b>	<b>16</b>
5.1. Arquitectura del sistema . . . . .	16
5.2. Modelo de datos . . . . .	16
<b>6. Implementación</b>	<b>17</b>
<b>7. Resultados</b>	<b>18</b>
<b>8. Conclusiones y trabajo futuro</b>	<b>19</b>
<b>Bibliografía</b>	<b>20</b>

# Índice de figuras

2.1. Ejemplo de flujo con virtualización . . . . .	9
3.1. Roadmap de tareas . . . . .	13

# 1. Introducción

—TODO— introducción a la introducción

## 1.1. Antecedentes

En la actualidad, la cantidad de datos que se generan y almacenan es cada vez mayor <sup>1</sup>, una tendencia que por supuesto se traduce a las empresas. Estos datos provienen de múltiples fuentes y en múltiples formatos, lo que dificulta su análisis y explotación: a esta característica de la información se le conoce como *heterogeneidad*<sup>2</sup>.

A su vez, el progreso tecnológico ha permitido la creación de nuevas herramientas y técnicas que facilitan la recogida, almacenamiento y análisis de estos datos. Una de estas técnicas son los *data lakes* (ver 2.1 *Data lake*), que permiten almacenar grandes cantidades de datos de diferentes tipos y formatos, para poder analizarlos y explotarlos de forma más eficiente.

### 1.1.1. Análisis y visualización de datos

La visualización de datos es una técnica que permite representar la información de manera visual, para facilitar su análisis y comprensión. La visualización de datos es una parte importante del proceso de análisis de datos, ya que permite identificar patrones, tendencias y anomalías en los datos de forma más rápida y sencilla.

La evolución de la visualización de datos ha ido de la mano de la evolución de la tecnología, y actualmente existen múltiples herramientas y técnicas que permiten visualizar datos de forma más eficiente y efectiva. Una de estas técnicas es el modelo DIKW.

La pirámide DIKW<sup>3</sup> es un modelo que describe la relación entre los datos, la información, el conocimiento y la sabiduría. Según este modelo, los datos son la materia prima de la información, que a su vez es la materia prima del conocimiento, que a su vez es la materia prima de la sabiduría.

La visualización de datos es una técnica que permite representar los datos de forma visual, para facilitar su análisis y explotación. Los dashboards (ver 2.3 *Dashboards*) son una

<sup>1</sup><https://www.statista.com/statistics/871513/worldwide-data-created/>

<sup>2</sup><https://www.sciencedirect.com/topics/computer-science/data-heterogeneity>

<sup>3</sup>[https://en.wikipedia.org/wiki/DIKW\\_pyramid](https://en.wikipedia.org/wiki/DIKW_pyramid)

herramienta que permite visualizar los datos de forma sencilla y eficiente, para poder tomar decisiones informadas sobre los mismos.

## 1.2. Motivación

El proyecto surge de la necesidad de la empresa (ver *1.4 La empresa*) de extraer información y conocimiento de las múltiples y heterogéneas fuentes de datos de las que se disponen, tanto internas (e.g. bases de datos, archivos de registros, APIs, entre otros), como externas (e.g. APIs o datos de webs de terceros, datos de fuentes públicas. . . ).

En la actualidad, la empresa dispone de una gran cantidad de datos que se encuentran en diferentes formatos y en diferentes ubicaciones, lo que dificulta su análisis y explotación. Por otra parte, se depende de la consulta manual o de servicios de terceros (como dashboards en NewRelic o AWS CloudWatch) para poder analizar estos datos, lo que supone un coste adicional.

Además del uso interno, la empresa también quiere ofrecer a sus clientes la posibilidad de consultar estos datos de forma visual y sencilla, para que puedan analizarlos y explotarlos de forma autónoma, lo que supondría un valor añadido para los mismos. Este tipo de dashboards son diferentes a los dashboards de monitorización antes mencionados, ya que permiten al usuario final la consulta de datos de negocio, y no de infraestructura.

## 1.3. Finalidad del proyecto

El objetivo de este sistema es centralizar y unificar las fuentes de datos heterogéneas cuya consulta se realiza de manera manual, con la finalidad de analizar los datos de forma más eficiente.

El cumplimiento de este objetivo permitirá a la empresa obtener una serie de beneficios:

- Eliminar el tiempo invertido en la consulta manual de los datos.
- Reducir los costes asociados a servicios de terceros.
- Mejorar la toma de decisiones, al poder analizar los datos de forma más eficiente.
- Incrementar la calidad de los servicios ofrecidos a los clientes, al poder ofrecerles la posibilidad de consultar los datos de forma visual y sencilla.
- Explotar económicamente este servicio, ofreciéndolo a terceros.

Además de la mejora de los procesos ya existentes, la explotación mediante esta herramienta abrirá la puerta a nuevas posibilidades de análisis y explotación de los datos, como la

detección de anomalías en la infraestructura o la predicción de patrones y eventos futuros.

## 1.4. La empresa

Okticket es una startup nacida en Gijón en 2017 cuyo producto principal es un servicio software que reduce los costes y el tiempo que invierten las empresas en contabilizar y manejar los gastos de viaje de los profesionales mediante el escaneo automático de tickets y notas de gastos.

La empresa tienen su sede principal en el Parque Tecnológico de Gijón, aunque cuenta con un número de sedes creciente en varios países, como Francia, Portugal o, más recientemente, México. En esta oficina principal se encuentran los departamentos de ventas y marketing, así como el equipo de desarrollo y soporte.

Okticket es una de las empresas que más crecen tanto del sector como del propio Parque Tecnológico. Debido a este rápido crecimiento, el equipo está en constante desarrollo y cambio, tanto aquí en España como en el resto de sedes. Este crecimiento se refleja en la recepción de un gran número de galardones y reconocimientos <sup>4 5 6 7</sup>

La parte principal del negocio es el núcleo del software como servicio (Software as a Service en inglés, en adelante *SaaS*), es decir, la aplicación completa tanto para administradores como para empleados. Este SaaS se oferta a empresas de cualquier tamaño, cuyo precio final varía en función del número de usuarios, las características e integraciones que requiera la empresa cliente y el soporte que se ofrezca.

Recientemente se han añadido nuevas propuestas a la cartera de servicios ofertada por Okticket, como la OKTCard - una tarjeta inteligente que gestiona automáticamente los gastos, así como la inclusión de nuevos “módulos” de gestión de gastos y viajes.

---

<sup>4</sup>Okticket en el especial startups 2023 de Forbes (LinkedIn)

<sup>5</sup>Arcelor y Okticket, premios nacional de Ingeniería Informática (EL COMERCIO)

<sup>6</sup>Okticket recibe el sello Pyme Innovadora (okticket.es)

<sup>7</sup>Okticket, empresa emergente certificada (okticket.es)

## 2. Fundamento teórico

—TODO— introducir fundamento teórico

### 2.1. Data lake

Los *data lakes*<sup>1</sup> son almacenes de datos que almacenan grandes cantidades de datos de manera no estructurada [1].

A diferencia de los *data warehouses*, los *data lakes* no tienen un esquema definido, lo que permite almacenar datos *heterogéneos*. Esto permite almacenar grandes cantidades de información sin tener que definir un esquema de antemano, lo que puede ser útil en aquellos casos en los que no se conoce la estructura de los datos que se van a almacenar.

### 2.2. Procesos ETL

#### 2.2.1. Definición

Los procesos ETL [1] son procesos que combinan datos de múltiples fuentes en un único destino, transformando los datos en un formato común. Estos procesos se utilizan para extraer datos de diferentes fuentes, transformarlos en un formato común y cargarlos en un destino común.

#### 2.2.2. Características

Los procesos ETL cuentan con una serie de características que los hacen útiles para la integración de datos:

- **Adaptabilidad:** los procesos ETL deben de adaptarse a la estructura de los datos de la fuente de origen, ya que dichas fuentes pueden tener diferentes estructuras y tener tipos de datos diferentes (la característica de *heterogeneidad* de los datos que ya se ha mencionado).
- **Escalabilidad:** otra de las características clave de los procesos ETL es que sean escalables, ya que los datos que se muestran en los dashboards suelen ser datos que se

---

<sup>1</sup><https://aws.amazon.com/es/what-is/data-lake/>



generan de manera continua, y por lo tanto los procesos ETL deben ser capaces de procesar grandes cantidades de datos de manera eficiente. En ocasiones, los procesos ETL se pueden realizar en *streaming*, lo que significa que los datos se procesan en tiempo real a medida que se generan.

### **2.2.3. Funcionamiento**

#### **2.2.3.1. Extracción**

En este proceso se extraen los datos de las fuentes de datos, que pueden ser bases de datos, logs, APIs, etc. En este paso, se pueden aplicar filtros para extraer solo los datos que se necesiten, y se pueden extraer datos de múltiples fuentes.

Frecuentemente, los datos brutos se almacenan temporalmente en una zona de almacenamiento intermedia llamada staging area (que es estrictamente transitoria).

En algunos casos, los datos se pueden extraer de manera incremental, es decir, solo se extraen los datos que han cambiado desde la última extracción. Esto puede ser útil para reducir el tiempo de procesamiento y el volumen de datos que se almacenan.

En otros casos, los datos se pueden extraer de manera continua, es decir, se extraen los datos en tiempo real según se van generando. Esto puede ser útil para procesar datos que se generan en tiempo real, como logs o datos de sensores.

#### **2.2.3.2. Transformación**

En este proceso se transforman los datos extraídos en un formato común, normalmente tablas relacionales.

Una transformación básica de datos es la limpieza, revisión y corrección de los datos extraídos, para asegurar que los datos que se almacenan son correctos y consistentes. Otras operaciones más complejas pueden ser la agregación de datos, la conversión de formatos, la normalización de datos, el cifrado, etc.

#### **2.2.3.3. Carga**

En este proceso se cargan los datos transformados en el destino final. Frecuentemente, los datos se almacenan en una data warehouse o data lake para su posterior análisis.

En algunos casos, los datos se pueden cargar de manera incremental, es decir, solo se

cargan los datos que han cambiado desde la última carga. Esto puede ser útil para reducir el tiempo de procesamiento y el volumen de datos que se almacenan. En otros casos, los datos se pueden cargar de manera continua, es decir, se cargan los datos en tiempo real según se van generando. Esto puede ser útil para procesar datos que se generan en tiempo real, como logs o datos de sensores.

#### 2.2.4. Alternativas

Aunque lo más común es el flujo anteriormente explicado de *extracción*, *transformación* y *carga*, existen algunos flujos y procesos alternativos que evitan algunos de estos pasos, normalmente en casos específicos que se beneficien del cambio:

- **Virtualización:** en lugar de extraer los datos de las fuentes, se crea una capa virtual que permite acceder a los datos de las fuentes sin necesidad de extraerlos. Esto permite ahorrar espacio de almacenamiento y tiempo de procesamiento, pero puede ser menos eficiente en algunos casos.
- **Proceso *ELT*:** en lugar de transformar los datos antes de cargarlos en el destino, se cargan los datos en bruto y se transforman en el destino. Funciona bien para grandes conjuntos de datos sin estructura que requieran una carga (o recarga) continua, aunque, al igual que la virtualización, puede ser menos eficiente en algunos casos.

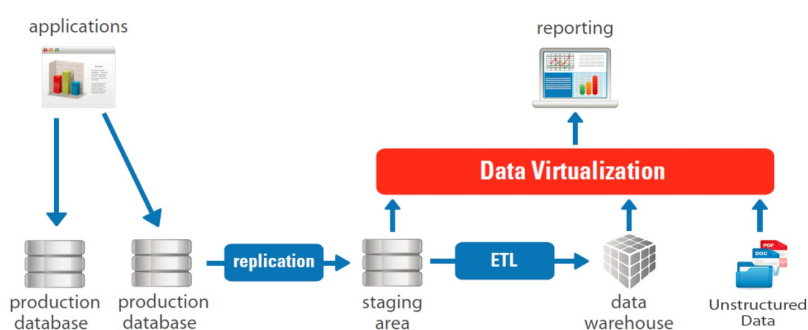


Figura 2.1: Ejemplo de flujo con virtualización

## 2.3. Dashboards

**Definición** [1] La palabra *dashboard*, que traducido de manera literal significa *cuadro de mandos*, es un término que se utiliza para referirse a cualquier interfaz gráfica que muestre información relevante de manera visual sobre un proceso o negocio. Aunque el término se utiliza en muchos ámbitos: indicadores comerciales, de producción, de marketing, de calidad, de recursos humanos...

En el ámbito de este proyecto, los dashboards reflejan en tiempo real el rendimiento de actividades o procesos de negocio, y se utilizan para tomar decisiones informadas sobre los mismos. En el caso de una empresa, un dashboard puede mostrar desde el rendimiento de la plataforma en tiempo real hasta un reflejo del de las ventas, y permitir a los directivos tomar decisiones informadas sobre el futuro de la empresa.

**Características** Los dashboards cuentan con una serie de características que los hacen útiles para la toma de decisiones:

- **Visualización de datos:** es la característica fundamental de cualquier cuadro de mandos, y aquella que determina su utilidad. La visualización de datos es la ciencia de presentar los datos de manera que se pueda extraer información útil y realizar decisiones informadas sobre ellos. Un buen dashboard cuenta con gráficas, tablas, indicadores, etc. que permiten al usuario entender la información que se está presentando con un conocimiento técnico mínimo.
- **Interactividad y personalización:** un dashboard debe permitir al usuario interactuar con los datos (filtrarlos, ordenarlos, profundizar en ellos...) y ajustar la información que se muestra a cada proceso o negocio que se esté evaluando. Esta capacidad asegura que el dashboard se adapte tanto a las necesidades actuales como a las evoluciones futuras de lo que se esté analizando.
- **Accesibilidad y portabilidad:** un dashboard debe ser accesible desde una variedad de situaciones y dispositivos, manteniendo su funcionalidad y forma. Aunque normalmente los dashboards se analizan en pantallas grandes, es importante que también se puedan consultar en otras circunstancias, como dispositivos móviles.

**Dashboards planteados** Para el sistema que se describe, se plantean dos tipos de dashboards diferentes:

- **Dashboards internos:** que reflejan el rendimiento de la plataforma en tiempo real.
- **Dashboards externos:** que reflejan el rendimiento de las ventas y permiten a los clientes tomar decisiones informadas sobre su negocio.

## 3. Planificación

La planificación de un proyecto es fundamental para su correcto funcionamiento y desarrollo, dentro de los plazos y costes establecidos. En este apartado se detallan las tareas que se llevarán a cabo en el proyecto, así como los recursos necesarios y los plazos de ejecución.

### 3.1. Metodología

#### 3.1.1. Scrum

Para la planificación del proyecto se ha escogido *Scrum*, una metodología “ágil” que se basa en la realización de iteraciones cortas y en la adaptación a los cambios. La metodología *Scrum* se estructura en *sprints* (iteraciones cortas de una duración fija), en las que se llevan a cabo una serie de tareas que se han planificado previamente.

El primer paso de la metodología *Scrum* es la creación de un *product backlog*, una lista ordenada de las tareas a realizar durante el desarrollo del producto, a partir de los requisitos del sistema, que a su vez son una versión refinada de los requisitos iniciales del proyecto. A partir de este *product backlog* se planifican las tareas que se llevarán a cabo en cada *sprint*, de manera que sea posible cumplir con los objetivos del proyecto en el tiempo establecido.

#### 3.1.2. Visualización de la planificación

Para la visualización de la planificación se ha utilizado la herramienta de gestión de proyectos de *GitHub*, que permite múltiples visualizaciones de tareas e *issues* en tableros separados.

- Se utiliza un tablero de *requisitos* al estilo *Kanban* para visualizar los requisitos del proyecto y su estado, siguiendo con la metodología *Scrum*.
- Se utiliza un *roadmap* de tareas, donde se visualizan las tareas y los hitos del proyecto, así como su estado y sus fechas límite.



Figura 3.1: Roadmap de tareas

### 3.1.3. Comunicación

La comunicación con los tutores y con el equipo de desarrollo se considera fundamental para el correcto desarrollo del proyecto. Puesto que el trabajo se desarrolla de manera presencial en la oficina de la empresa, la comunicación con el equipo de desarrollo se realiza de manera frecuente y directa, mientras que la comunicación con los tutores se realiza de manera remota pero igual de frecuente, manteniendo el contacto mediante correo electrónico y Teams para pedir revisiones e informar sobre el estado del trabajo en todo momento.

## **3.2. Presupuesto**

## 4. Análisis

—TODO— introducir análisis

### 4.1. Partes interesadas (stakeholders)

Las partes interesadas en el proyecto son aquellas personas o entidades que tienen un interés en el mismo, ya sea porque se ven afectadas por el resultado del proyecto, o porque tienen algún tipo de interés en el mismo. Las partes interesadas en este proyecto son las siguientes:

1. **Okticket:** la empresa es la principal parte interesada en el proyecto, ya que es la que se beneficiará directamente de los resultados del mismo, así como de las oportunidades de negocio que se abren con la explotación de los datos.
  - **Equipo de desarrollo:** el equipo de desarrollo es otra parte interesada en el proyecto, ya que son los encargados de llevar a cabo la implementación del sistema y de garantizar su correcto funcionamiento.
  - **Equipo de soporte:** el sistema planteado ahorraría tiempo al equipo de soporte, ya que les permitiría analizar los datos de forma más eficiente e identificar problemas antes de que tener que resolver las peticiones de los clientes afectados.
2. **Clientes:** los clientes de la empresa también son partes interesadas, puesto que se beneficiarán de los nuevos servicios que se ofrecen, como los dashboards de negocio que se han descrito anteriormente.
3. **Investigador y desarrollador:** el desarrollador del proyecto tiene la oportunidad de aplicar los conocimientos adquiridos en el desarrollo de un proyecto real, y de adquirir nuevos conocimientos en el proceso.

### 4.2. Definición del sistema

### 4.3. Análisis de alternativas



## **5. Diseño del sistema**

### **5.1. Arquitectura del sistema**

### **5.2. Modelo de datos**

## 6. Implementación

## **7. Resultados**

## 8. Conclusiones y trabajo futuro

# Bibliografía

- [1] J. Mier, “Presentación de datos: dashboards y procesos ETL.” Primera entrega de teoría de la asignatura Inteligencia de Negocio, EPI Gijón, curso 23-24, 2023.
- [2] J. Mier, “latexTemplate.” <https://github.com/miermontoto/latexTemplate>, 2024. Plantilla de L<sup>A</sup>T<sub>E</sub>X personal para trabajos académicos.
- [3] J. Mier, “Minería de anomalías.” Segunda entrega de teoría de la asignatura Inteligencia de Negocio, EPI Gijón, curso 23-24, 2024.