

2023

# DEVELOPING A DATALAKE

(Alternative edition)

2º Curso de Ciencia e Ingeniería de Datos  
Escuela de Ingeniería  
Informática Universidad de Las Palmas de  
Gran Canaria

## **Esta es la memoria sobre el proyecto de Developing a Datalake (alternative edition)**

Este programa va a permitir que un usuario a través de un web service, realice consultas SQLITE de los sitios con mayor temperatura y menor temperatura de la isla en un cierto rango de días, de los datos meteorológicos extraídos y almacenados de la AEMET, a través de una API REST.

El programa accederá a la API de AEMET para así crear un datalake conteniendo en forma de eventos diarios, el registro de temperatura de los sensores. De este mismo datalake, se extraerá un registro de las temperaturas máximas y mínimas de cada día y se almacenarán en una base de datos SQLITE de la que se podrán realizar peticiones a través de una API REST.

**Índice: Recursos utilizados..... ¡Error! Marcador no definido.**

**Diseño ..... 2**

**Conclusiones ..... 3**

**Líneas futuras..... 4**

**Bibliografía ..... 5**

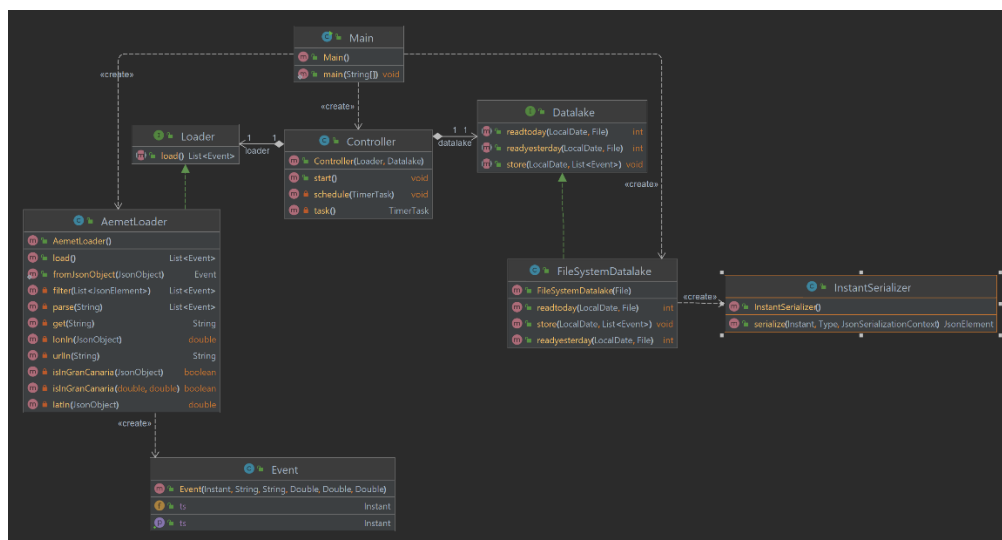
## Recursos Utilizados:

Entre los recursos utilizados encontramos como entorno de desarrollo la aplicación IntelliJ IDEA Community Edition en su versión 2022.2.2. No se utilizó ninguna herramienta de control de versiones y la herramienta de documentación utilizada es Word.

## Diseño

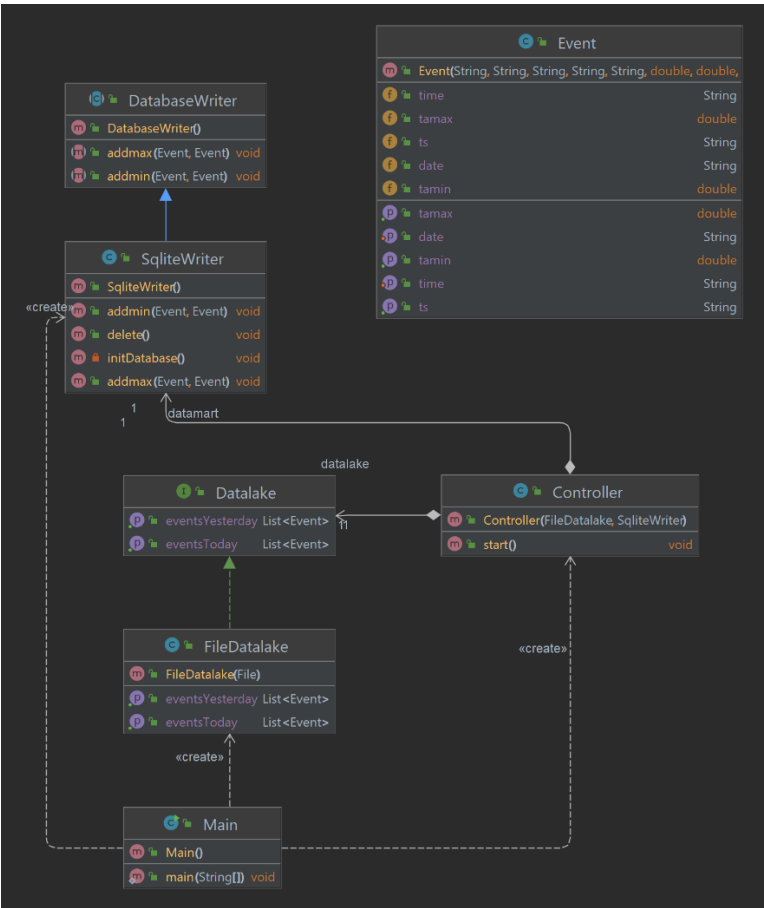
La estructura del código consiste en un archivo llamado “aemetproyect”, que nada mas abrirlo vemos que se segmenta en tres distintos directorios que trabajaran de manera asíncrona. Cada directorio tendrá una responsabilidad distinta, los tres son necesarios para el correcto funcionamiento del programa.

El primer directorio “Feeder”, tendrá la función de conectarse a la API de la AEMET y cada hora pedirá los datos de las estaciones. La clase “AemetLoader” se encargará de conectarse a la API y filtrar el stream de los datos, a través de la función *filter*, de manera que se coleccionen solo aquellos datos que cumplen la latitud correspondiente a la isla de Gran Canaria. Todos los datos serán almacenados como eventos de clase “Event”, para luego procesarlos el “FileSystemDatalake” para almacenarlos en la carpeta *datalake* que se creara tomándola de los argumentos de lanzamiento del programa (especificar C:\ruta...\datalake) e ira creando cada día un fichero .events con sus respectivos eventos.

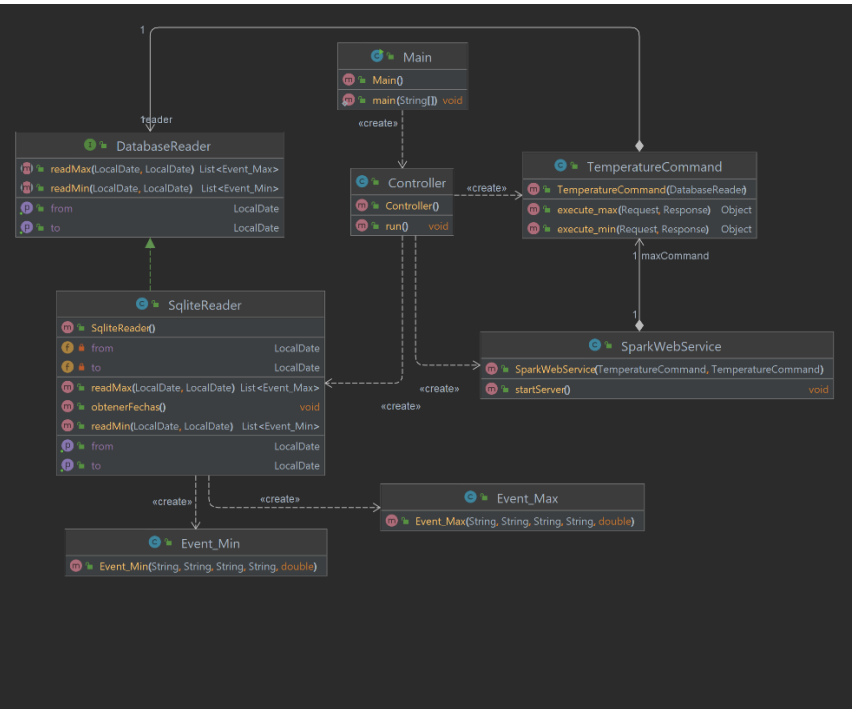


El segundo directorio “Datamart-builder” se encargará de acceder a la ruta donde se encuentre la carpeta *datalake* para así leer cada fichero de eventos guardados y almacenarlos en listas. Estas mismas serán operadas por el “Controller”, que se encargara de extraer los eventos con la máxima temperatura de cada día y los de temperatura más baja de cada día. Luego se introducirán en la base de datos SQLITE creada por la función “SqliteWriter”, quien se encargará de poner según se trate el

evento en su tabla correspondiente (tabla de registro temperaturas máximas o tabla de registro temperaturas mínimas).



El tercer directorio “Temperatura-API” será quien inicialice la API REST, accediendo a través de la clase “SqliteReader” a los registros de la base de datos y según que petición realice el usuario en el web service, la clase “SparkWebService” se encargara de devolver los datos correspondientes.



## Conclusiones

Este proyecto me ha resultado bastante atrevido, con una serie de desafíos a los que cualquier alumno inicializado en programación en java le costara superar. Sin embargo, considero que el aprendizaje obtenido durante todo el desarrollo del proyecto sin duda es el mas elevado, puesto que conforme a más desafíos se enfrente el alumno, mas posibilidades de aprendizaje podrá obtener de ellos.

Un aspecto muy importante y que valoro positivamente de el planteamiento de este proyecto, es que sigue muy fielmente los conocimientos obtenidos desde el principio de la asignatura. Y el motivo del que decido destacar previo aspecto mencionado es que este, a mi parecer, fue la mayor ayuda durante el desarrollo del proyecto. Digamos que el Spotify API fue la pieza necesaria para aprender como funciona un sistema de claves en una API externa, como funciona una API con sus respectivas request, y como crear bases de datos en SQLITE. El trabajo de Scraper me aporoto el conocimiento necesario de como aprender a manejar volúmenes de datos y proporcionarlos a través de un web service. Y es precisamente de estos trabajos, de donde obtuve el conocimiento suficiente para poder enfocar y desarrollar este proyecto final.

Otra conclusión, a mi considerar, relevante para el profesorado es que el planteamiento de todo el proyecto sin lugar a duda fue lo más complicado. He aprendido que en un programa se te pueden presentar varias dificultades con las que tener que lidiar y muchas veces no saber cómo solucionarlas, es por eso por lo que considero indispensable tres ayudas en este aprendizaje, las del profesorado José Juan y Octavio (por la parte del planteamiento y el comienzo del programa), la ayuda de mi compañera Alejandra Ruiz de Adana, con quien mano a mano he logrado no solo trabajar, sino que enfrentarme a decenas (si no son centenas) de problemas y errores y finalmente el CHAT GPT, quien me ha ayudado a aprender infinidad de elementos gramáticos, funcionales y arquitectónicos del lenguaje *java*.

## Líneas futuras

Se podría utilizar la base de datos para analizar las tendencias climáticas a lo largo del tiempo, y proporcionar dicha información a investigadores y científicos con el fin de estudiar cómo el clima puede estar cambiando en la isla, y como esto podría afectar a las comunidades y a los ecosistemas de la isla.

Además, esta información también podría ser utilizada para desarrollar estrategias para adaptarse y mitigar los efectos del cambio climático en la isla, como la planificación de infraestructura y la gestión de los recursos naturales.

## Bibliografía

- <https://openai.com/blog/chatgpt/> (el uso de el chat gpt, herramienta que me recomendó Jose Juan, de donde aprendí a hacer relaciones entre clases y la utilización de varios métodos en general)
- [https://www.youtube.com/watch?v=Bq7WBXd6gEg&t=17s&ab\\_channel=JavaHangout](https://www.youtube.com/watch?v=Bq7WBXd6gEg&t=17s&ab_channel=JavaHangout)
- <https://www.javadoc.io/doc/com.google.code.gson/gson/2.8.0/com/google/gson/TypeAdapter.html>