Proyecto datalake to datamart

Desarrollo de Aplicaciones para Ciencia de Datos

2º año de Grado de Ingeniería en Ciencia de Datos.

Escuela Universitaria de Informática. Universidad de Las Palmas de Gran Canaria.

Irene Guerra Déniz

Versión: v1.0

Fechas de revisión: 30/01/2022, 03/01/2023, 04/01/2023, 05/01/2023, 10/01/2023, 11/01/2023, 13/01/2023

Resumen:

El proyecto se divide en tres módulos, uno que corresponde a la obtención de los datos de AEMet y el guardado de la información en un datalake, otro que se encarga de leer la información del datalake, procesarla y guardarla en un datamart, y por último uno que proporciona una API REST para realizar consultas sobre las localizaciones con temperaturas máximas y mínimas.

En primer lugar, para conectarse a la API de AEMet y obtener los datos en formato JSON de las estaciones meteorológicas se debe obtener una API key, que será insertada en el programa a través de los argumentos de la clase Main (del módulo del datalake), siendo este su único argumento, y que deberá escribir la persona que quiera ejecutar el código.

Para el desarrollo del módulo del datalake primero se ha accedido a los datos a través de la API key ya mencionada, se han extraído los datos en formato JSON y se ha accedido evento por evento escogiendo únicamente los que tenían los valores de latitud y longitud dentro de un intervalo definido para recoger solo los datos de Gran Canaria. De estos eventos se han cogido los atributos de la clase Weather: fecha, estación, ubicación, temperatura máxima y mínima, creando objetos de este tipo. Todos los objetos de esta clase se han introducido en una lista. Seguidamente se crea un directorio llamado “datalake” comprobando si existe o no y se accede a él. Se recorren los objetos de tipo Weather de la lista mencionada con anterioridad, obteniendo la fecha para que cada vez que se examine un objeto, se cree (si no existe) un fichero que posea como nombre esa fecha, y que de esta manera haya un fichero por día. Además, se comprueba que el objeto (pasado a formato JSON) que se vaya a escribir en el fichero no haya sido escrito previamente, para no tener información repetida. Si efectivamente no ha sido introducido con anterioridad se añade desde donde último se ha escrito, para no sobrescribir información y así perderla. Los eventos se escribirán línea por línea.

En cuanto el módulo del datamart, se accede a la ruta donde se ha creado el datalake y se obtienen los nombres de los ficheros que contiene el directorio. Se leen línea por línea los eventos de cada uno de los ficheros y por cada día se guardan en una lista de JSONObjects. Esta lista se filtra con la función stream(), obteniendo, por una parte, el evento con la temperatura mínima del día, comparando todas las mínimas de ese día entre ellas y cogiendo la menor de todas. El JSONObject que la posee se guarda en una lista general para los eventos con la temperatura mínima de cada día, por otra parte, se hace exactamente lo mismo con las temperaturas máximas guardándose en otra lista. Estas dos listas se introducen en dos tablas de una base de datos SQlite, siendo una para la lista de máximas y otra para la de mínimas.

Por último, el módulo de la API REST en la que se pueden realizar consultas de las localizaciones de las temperaturas máximas o mínimas entre dos fechas (ambas inclusive). Las fechas se deberán escribir en este formato ‘yyyy-MM-dd’. De este módulo se recogen los registros de la base de datos de la tabla correspondiente filtrándolos por el intervalo de fechas elegido por el usuario y se le pasa en una lista las localizaciones de los registros al WebService para que las muestre en formato JSON en la API.

Para finalizar, se ha añadido un TimerTask a los módulos datalake y datamart para que se ejecuten cada hora y así se actualice la información. Y se ha creado además una cola de ejecución para que ejecute las tres clases Main.

Índice

Recursos utilizados

Entorno de desarrollo

Se ha utilizado IntelliJ para el desarrollo del proyecto y el plugin de SimpleSqliteBrowser para visualizar la base de datos SQlite. Además, para utilizar y probar la API se ha empleado la extensión de Google Chrome Talend API Tester.

Herramienta de control de versiones

Para el control de versiones se ha empleado Git, y se ha subido el proyecto a GitHub.

Herramienta de documentación

Para la documentación del proyecto se ha utilizado Microsoft Word y luego se ha pasado a PDF.

Diseño

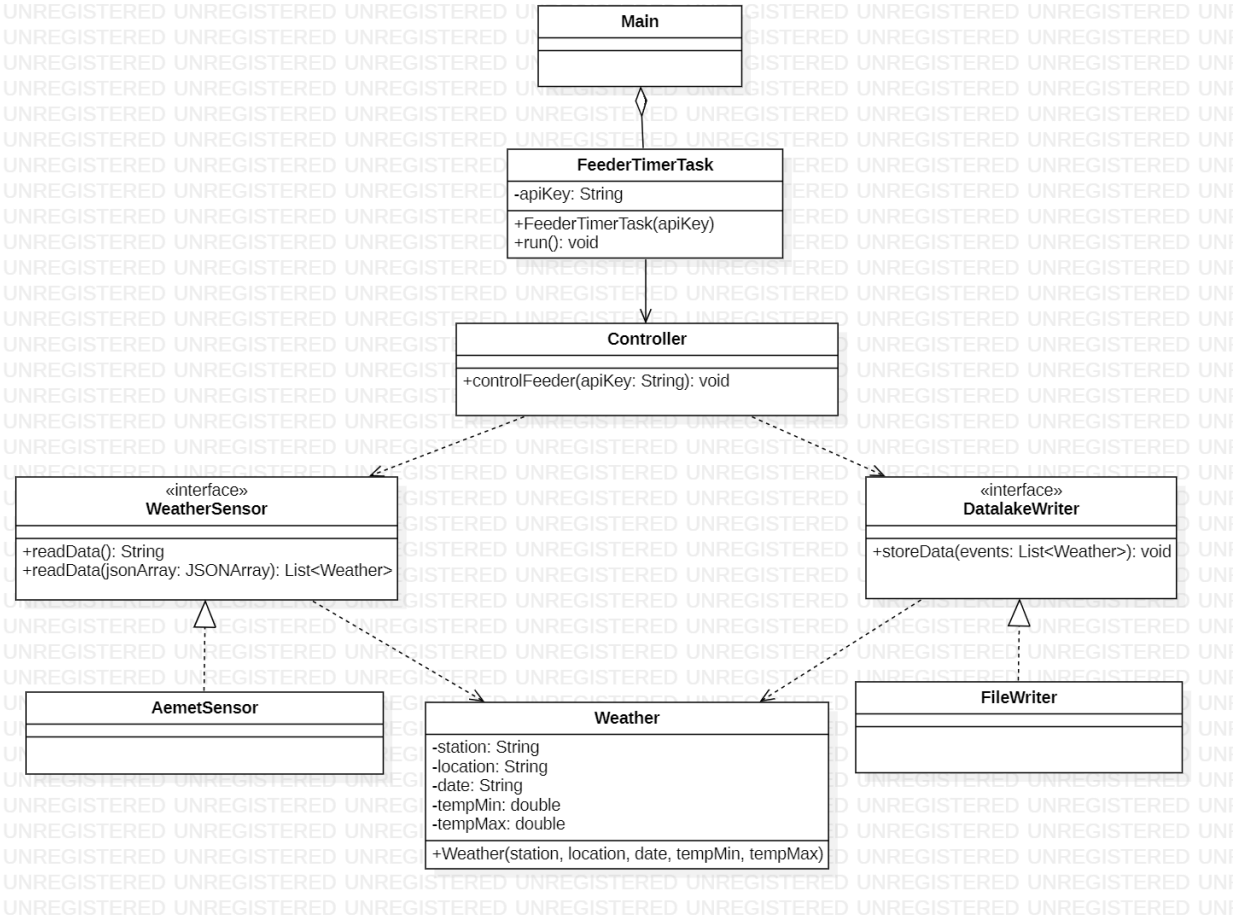
Patrones y principios de diseño

Se ha aplicado el estilo de arquitectura software model-view-controller (modelo, vista, controlador) sin tener definida la vista ya que no se posee una interfaz de usuario.

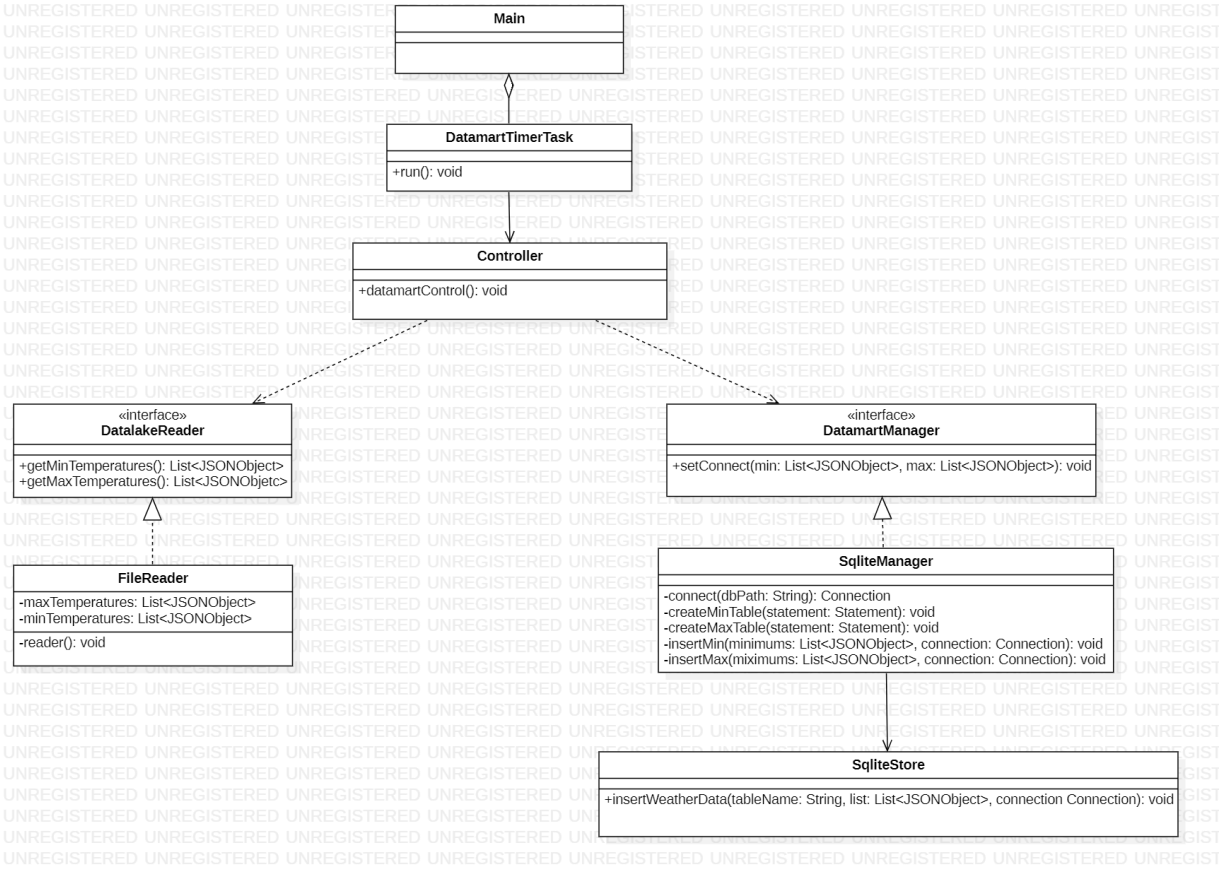
Además, se ha aplicado una arquitectura lambda, y los principios de diseño SOLID.

Diagramas de clase

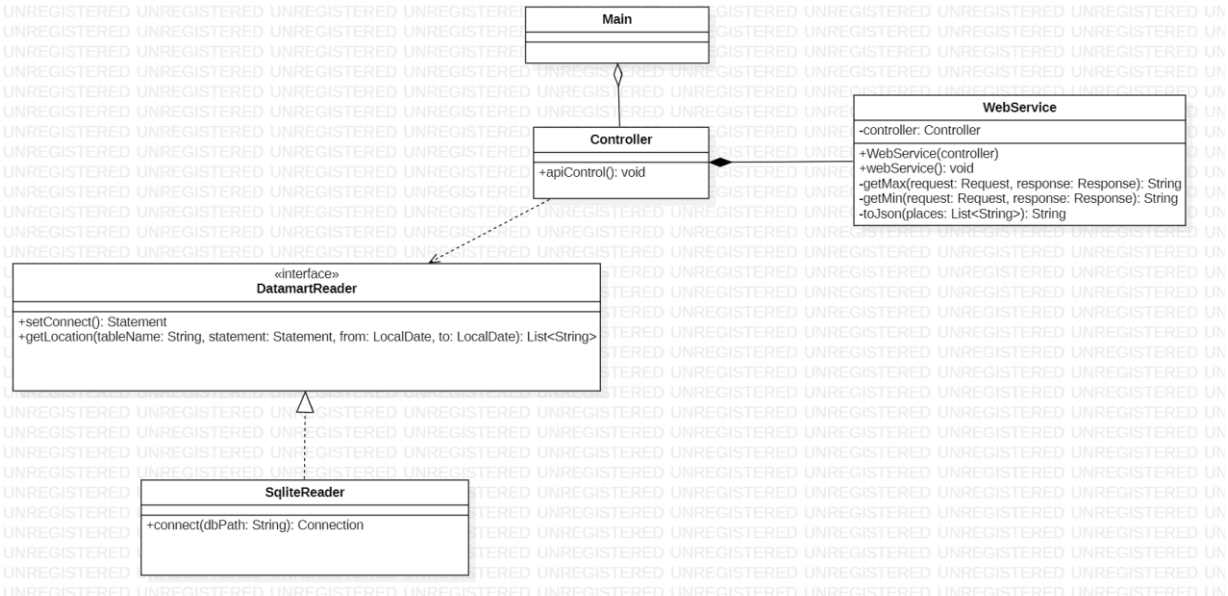
Módulo datalake:



Módulo datamart:



Módulo API REST:



Conclusiones

Con este proyecto he repasado como manejar ficheros y modificarlos, he aprendido a usar e implementar interfaces, he practicado el manejo del formato JSON, JSONArray, JSONObjects, la gestión de bases de datos SQlite, el filtrado a través de stream(), manejar los parámetros de la API REST, y he aprendido a usar un TimerTask que aunque sea sencillo es bastante útil.

Este proyecto asimismo me ha ayudado a interiorizar realmente los conceptos de datalake y datamart y a saber utilizarlos en la práctica.

Si empezase de nuevo el trabajo me haría un diagrama de clases claro antes de empezar a programar, ya que al tener este proyecto tantas clases me hubiese ayudado a organizarme mejor desde un principio.

Líneas futuras

Con el proyecto actual se pueden saber los lugares más cálidos y fríos de la isla. Pero si se hiciesen otros datamarts en el proyecto como uno para las velocidades del viento y otro para las precipitaciones, y además añadir información sobre la altura y otras características del lugar, se podrían realizar estudios más completos en cuanto a la relación entre estos atributos y las propiedades del lugar y podría ayudar con las predicciones y a realizar comparaciones con fechas pasadas.

Bibliografía

https://www.sqlitetutorial.net/

https://stackoverflow.com/

https://opendata.aemet.es/