* Justificación de patrones utilizados.

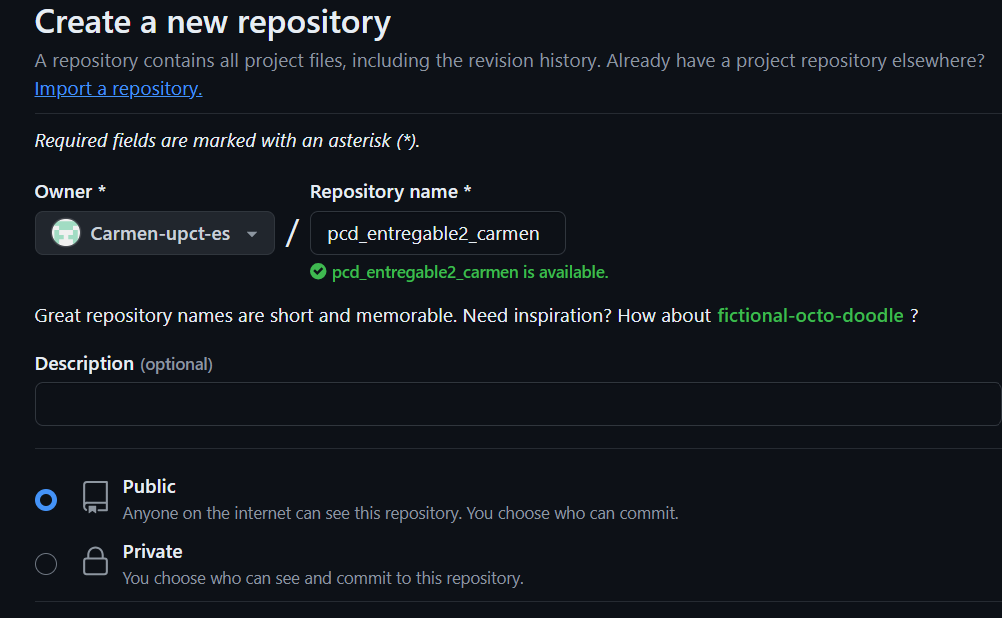
Para cumplir con **R1**, utilizamos el patrón **Singleton**. Este garantiza que una clase sólo tenga una instancia la cuál proporcione un punto de acceso único a ella. IoTSystem gestiona solo todo el entorno, por lo que sólo debe haber una única instancia en esta clase. De esta manera evitamos la duplicación de recursos y todas las operaciones se realizarán sobre él.

Para cumplir con **R2**, utilizamos el patrón **Observer**, el cuál te garantiza una dependencia de uno a muchos de manera que si una cambia de estado todos son actualizados y notificados. TemperatureSensor es el notificador del nuevo valor de temperatura y de esta manera los observadores podrán conocer y reaccionar a los cambios de temperatura de la manera más rápida posible.

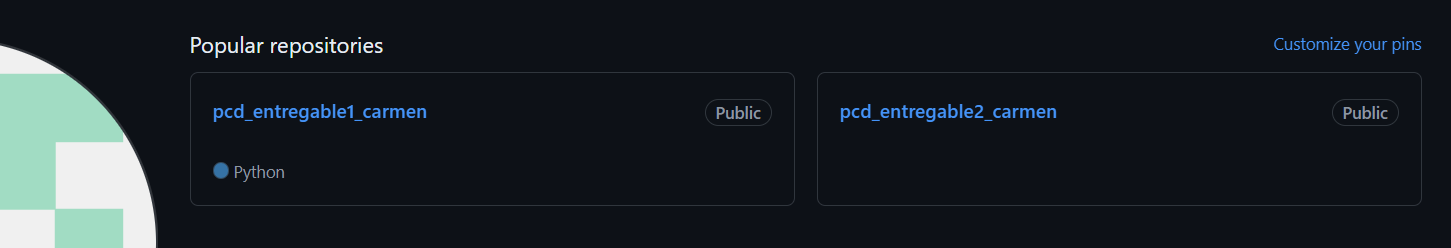
Para cumplir con **R3**, utilizamos **Chain of Responsability**. Esta permite que los manejadores puedan procesar solicitudes sin tener que remitir al receptor. De esta manera podremos procesar la temperatura en cadena con cada manejador realizando diferentes acciones como el cálculo de estadísticos, por ejemplo.

Para cumplir con **R4**, utilizamos el patrón **Strategy**, el cual nos permite encapsular un algoritmo en la familia definida. Así, podremos tener los diferentes métodos para el cálculo de los estadísticos sobre la temperatura y de esta manera no tendremos que modificar el código de las clases.

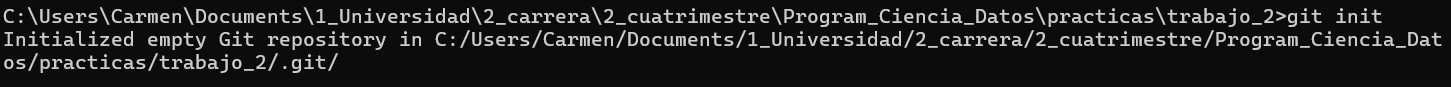
* Comencemos creando el repositorio en GitHub:



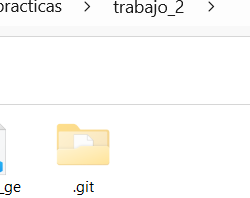
Vemos que el repositorio se ha creado correctamente:



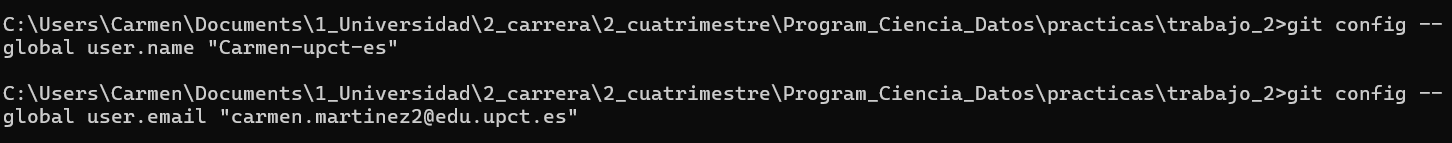
* url del repositorio: <https://github.com/Carmen-upct-es/pcd_entregable2_carmen>
* Creamos el .git en nuestra carpeta de trabajo con *git init*:



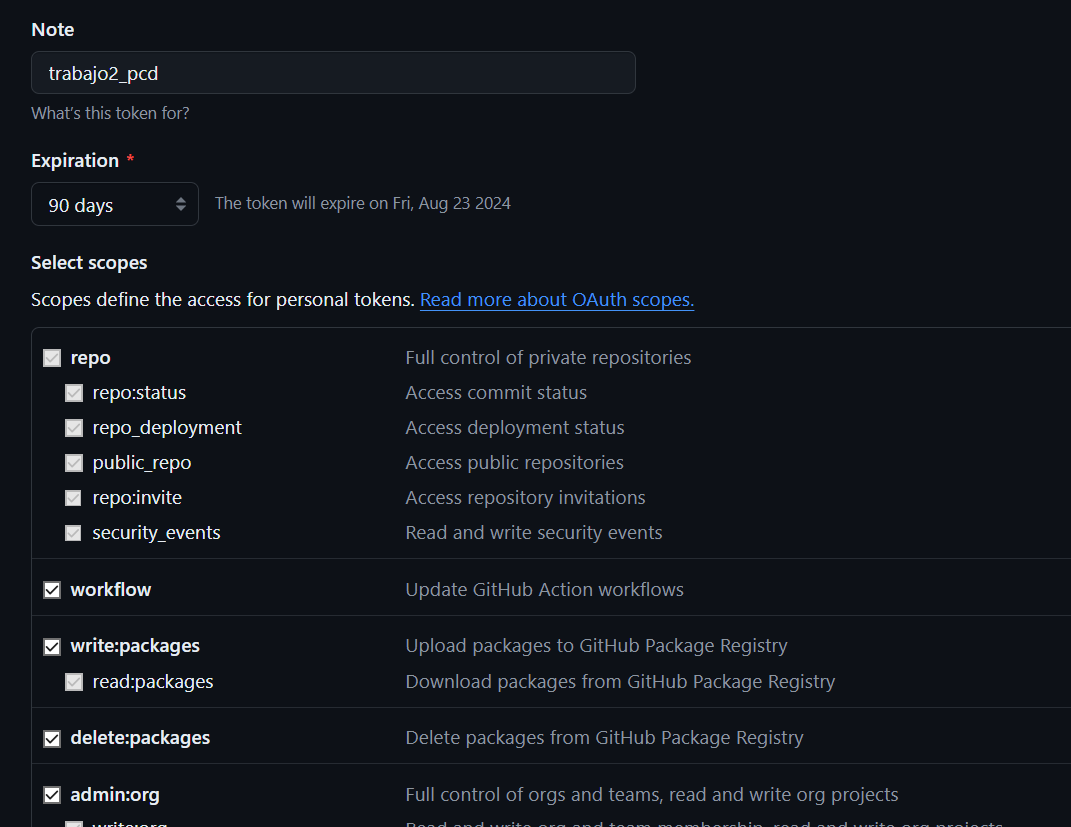
Comprobamos que se ha creado correctamente:

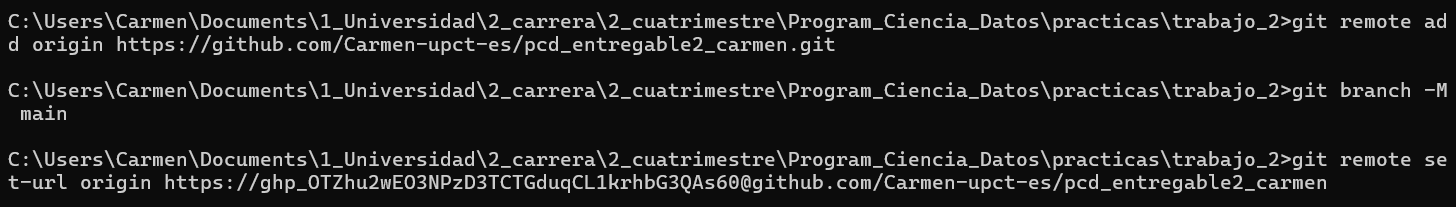


* Añadimos los datos para conectar con nuestro repositorio git con *git config:*

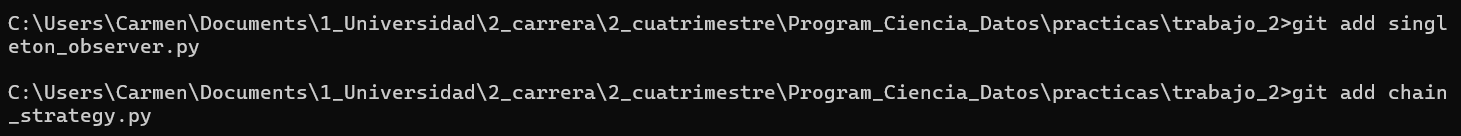


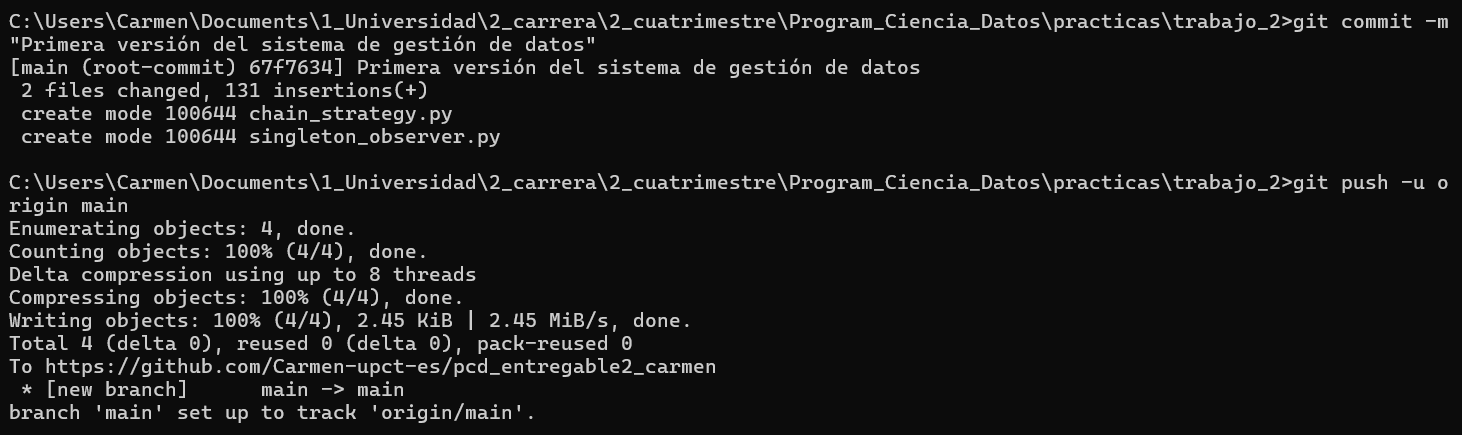
* Generamos un token y conectamos el repositorio externo a local. Creamos la rama:



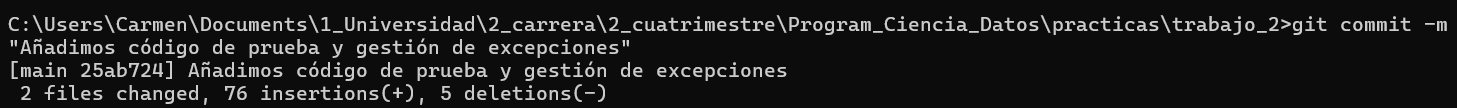


* Ahora podemos añadir los bocetos del sistema de gestión en main:

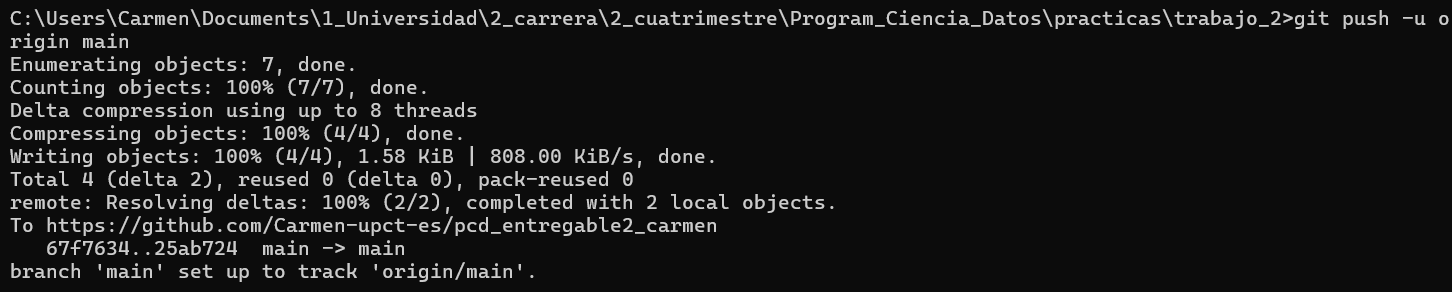
(el add se podría haber hecho de una vez también)



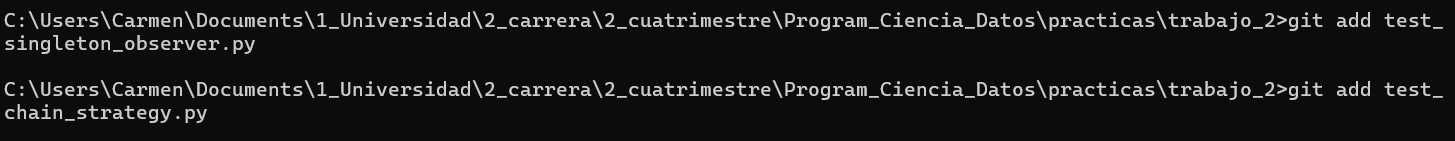
* Hacemos el siguiente commit una vez añadidas gestión de excepciones y código de prueba y añadimos un tag a esos dos ficheros:

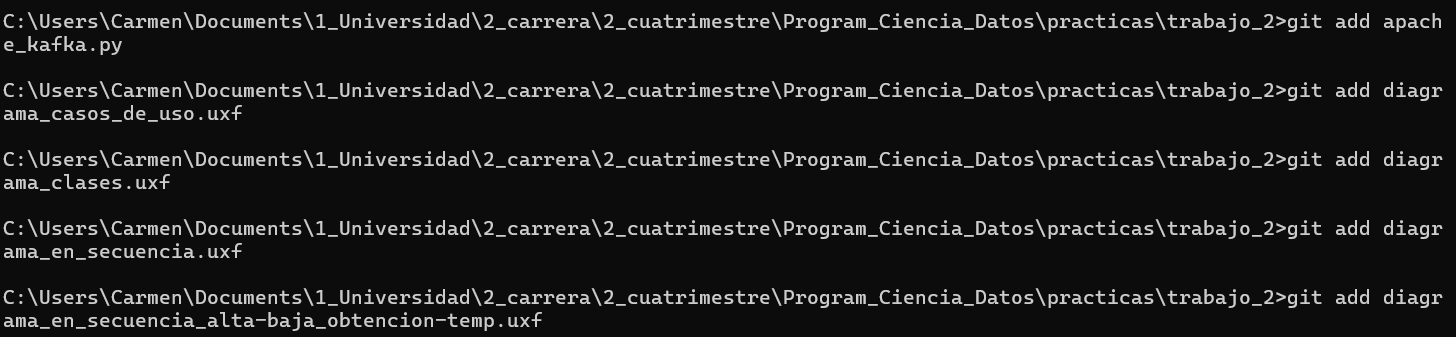


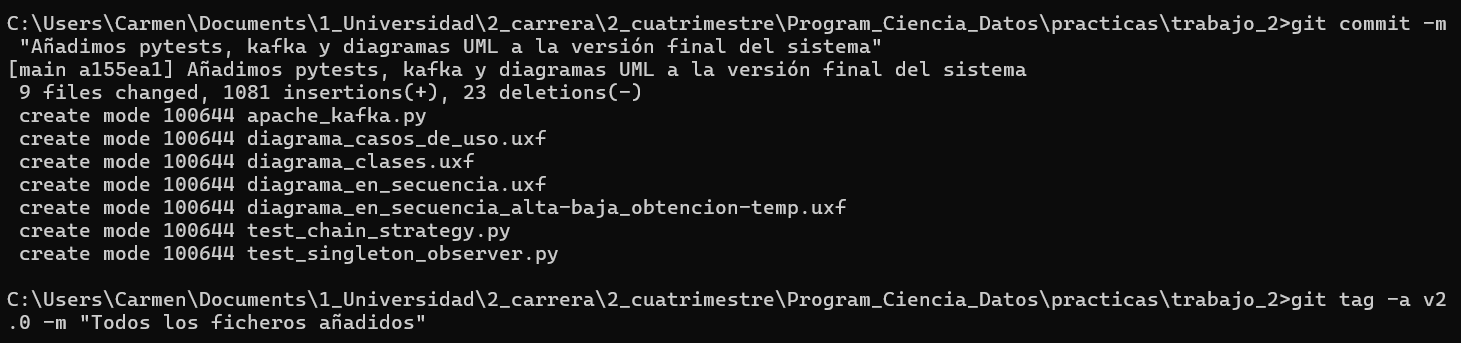




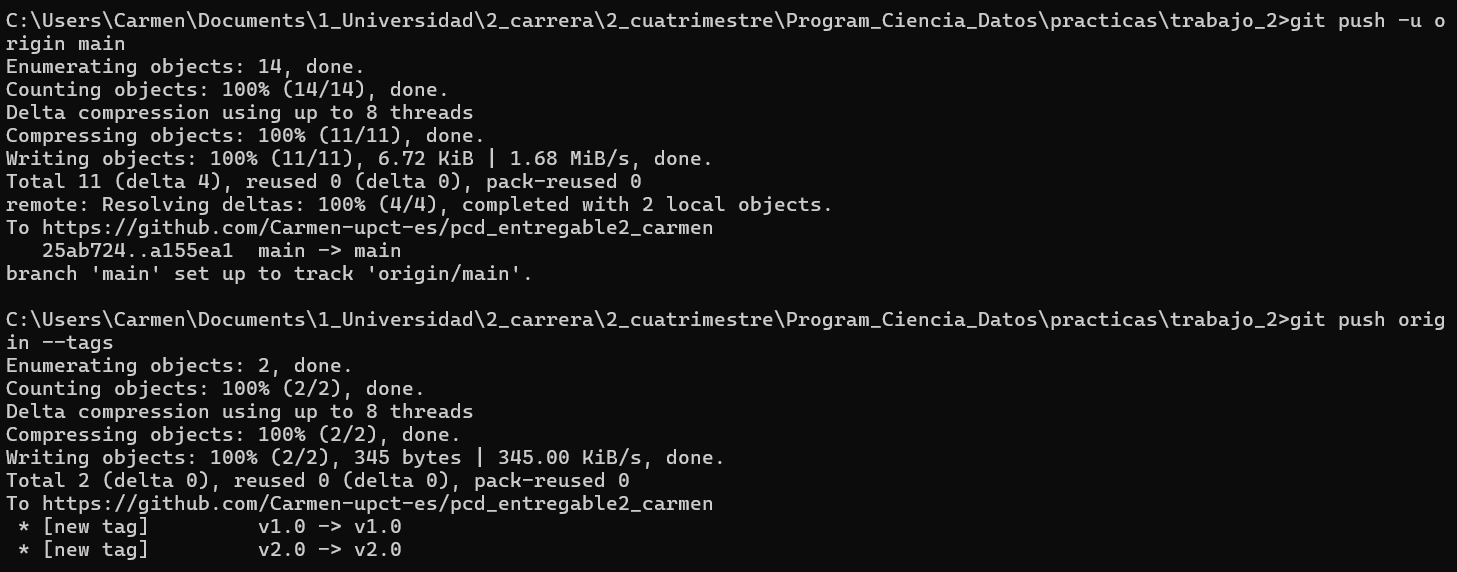
* Añadimos los tests unitarios, kafka y los diagramas UML y hacemos commit poniendo otro tag v2.0:







* Subimos los ficheros a GitHub junto con las etiquetas:



* Por último, subimos este fichero a GitHub.