• Información general del proyecto: para qué sirve, cuál es la estructura general del diseño, qué grandes retos de diseño enfrenta (i.e. ¿qué es lo difícil?). Deben incluir la URL para consultar el proyecto.

El proyecto es para guardar la información de las acciones hechas por el usuario, como el historial de un navegador, donde registra acciones y direcciones que el usuario realiza en orden cronológico.Diagrama

Descripción generada automáticamente

Los retos que presenta este patrón son:

1.Gestion eficiente del historial

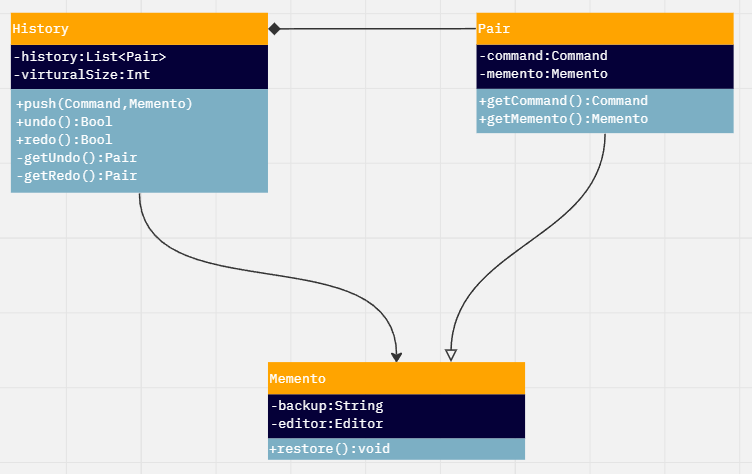
2.SIncronizacion entre componentes

3.Implementacion lógica de deshacer/rehacer

4.Integracion con la interfaz de usuario

<https://refactoring.guru/es/design-patterns/memento/java/example#example-0--history-History-java>

• Información y estructura del fragmento del proyecto donde aparece el patrón. No se limite únicamente a los elementos que hacen parte del patrón: para que tenga sentido su uso, probablemente va a tener que incluir elementos cercanos que sirvan para contextualizarlo.



En este caso el proyecto se basa alrededor del patrón de diseño memento, usa sus 3 clases principales junto a sus funciones presentes en diferentes clases y relaciones, los métodos undo y redo de History son los encargados de la funcionalidad de deshacer o rehacer las acciones del usuario dentro de la aplicación. Los get sustraen la información del comando, push añade información acerca de las acciones del usuario a history y backup almacena el estado de respaldo.

• Información general sobre el patrón: qué patrón es y para qué se usa usualmente.

El patrón Memento es un patrón de diseño de comportamiento que se utiliza para capturar y externalizar un estado interno de un objeto de manera que el objeto pueda ser restaurado a este estado después. Este patrón permite implementar funcionalidades de deshacer y rehacer, así como la capacidad de almacenamiento y recuperación el estado de un objeto en un momento dado.

• Información del patrón aplicado al proyecto: explicar cómo se está utilizando el patrón dentro del proyecto.

El proyecto como tal se construye alrededor del patrón, no como tal en la estructura sino en la funcionalidad, el proyecto sigue tal cual el funcionamiento del patrón de almacenar y retornar información sin embargo lo reorganizan para que las acciones de undo y redo se hagan en una misma clase envés de separadas al igual que la obtención de estados.

• ¿Por qué tiene sentido haber utilizado el patrón en ese punto del proyecto? ¿Qué ventajas tiene?

El proyecto al ser un historial, y el patrón memento una forma de usar las mismas funcionalidades que puede tener un historial es justo lo que se buscara que tenga la aplicación, en este caso la ventaja es que este patrón cobija todas las funciones de un historial en él.

• ¿Qué desventajas tiene haber utilizado el patrón en ese punto del proyecto?

El sistema al basarse en este patrón, consume en gran medida memoria RAM, por lo que si se registran muchas acciones del usuario el programa se demorara en mostrar sus resultados además de consumir más recursos del sistema.

• ¿De qué otras formas se le ocurre que se podrían haber solucionado, en este caso particular, los problemas que resuelve el patrón?

Una manera potenciar este patrón memento, que debido a las funcionalidades que le provee a la aplicación no seria conveniente quitarlo, seria mezclarlo con otros patrones de diseño. Para reducir su impacto en memoria RAM podría implementarse junto al patrón Flyweight, evitando la duplicación de datos y junto a la reducción de uso de memoria se lograría un mejor rendimiento.