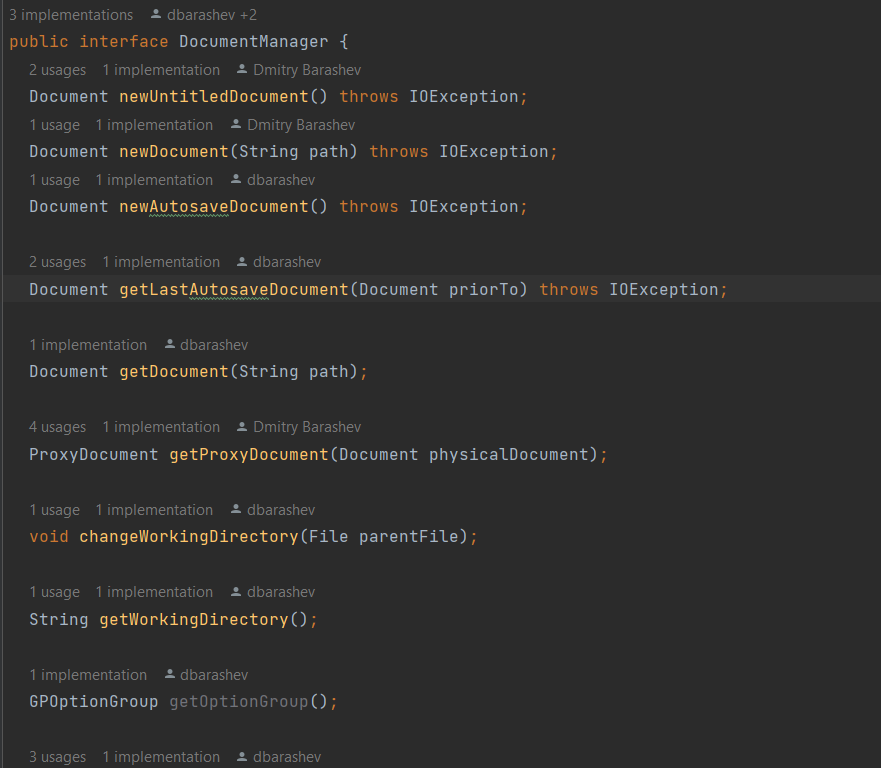
# *Design Patterns*

## *Eis 3 Design Patterns encontrados no código:*

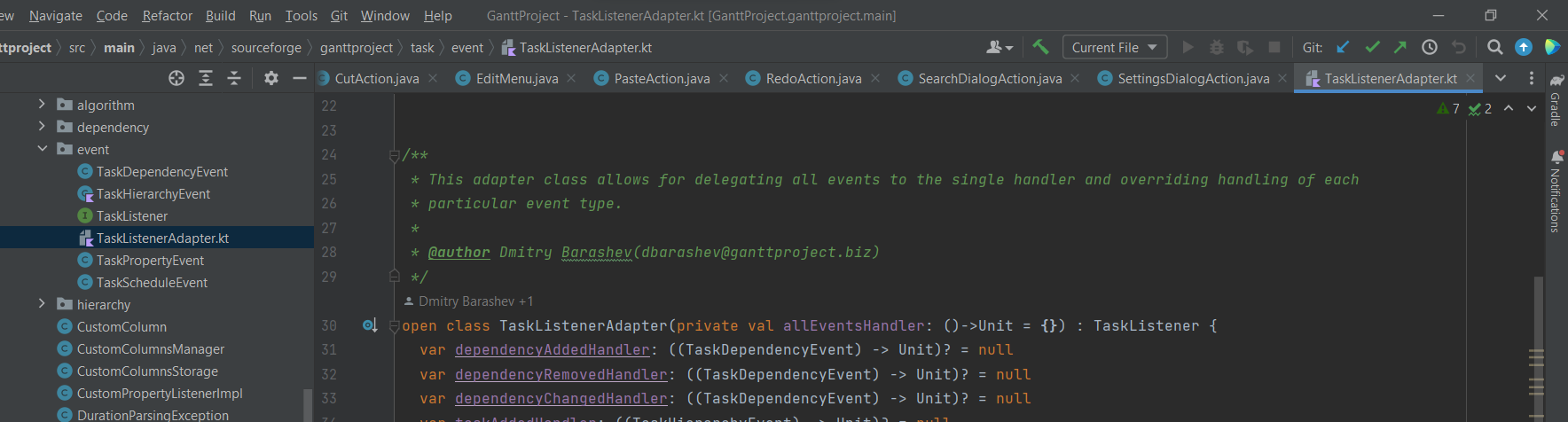
### Memento Pattern

Esta interface e as respetivas implementações preservam o estado antigo, corrente e futuro do documento a editar. Isto é o foco do Memento Pattern, uma vez que ele permite que em qualquer momento seja possível ao utilizador (des)faça quaisquer operações realizadas no texto/documento a tratar.

 Pelas linhas 33-49 de ganttproject/src/main/java/net/sourceforge/ganttproject/document/DocumentManager.java temos, por exemplo:

### The Adapter Pattern

Este *Design Pattern* foca-se em facilitar a comunicação entre dois sistemas/objetos, através de uma interface compatível para ambos. E é isto que observamos neste exemplo. A TaskListenerAdapter é uma interface de apoio ao manipulador de eventos consoante cada tipo de evento.

 Pelas linhas 24 a 33 de ganttproject/src/main/java/net/sourceforge/ganttproject/task/event/TaskListenerAdapter.kt, temos, por exemplo:

### Factory Pattern

Como se pode ver neste exemplo, é definida a interface ActionStateChangedListener para criar todos os produtos distintos deste tipo, mas deixa a criação real do produto para a classe concreta do Produto.

Apresento as linhas 80-97 de ganttproject/src/main/java/net/sourceforge/ganttproject/action/ArtefactAction.java (do produto) sendo a localização da interface: ganttproject/src/main/java/net/sourceforge/ganttproject/action/ActionStateChangedListener.javaText

Description automatically generated