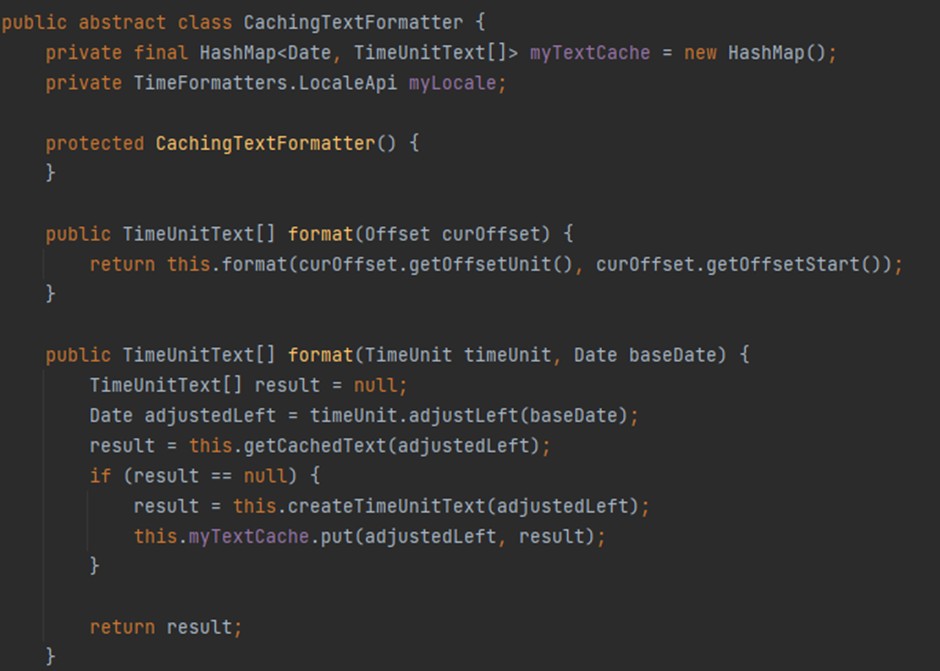
# Mafalda Batalha

* **Template Method**

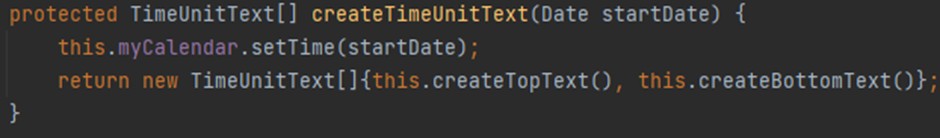
Package: biz.ganttproject.core.chart.text (biz.ganttproject.core.src.main.java)

A classe abstrata CatchingTextFormatter contém um método inicializado CreateTimeUnixText(), utilizado no método implementado format(), que será implementado de forma diferente por cada classe que a estende.

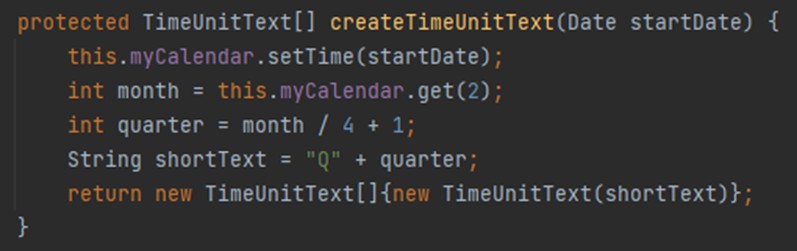


*Classe CatchingTextFormatter*

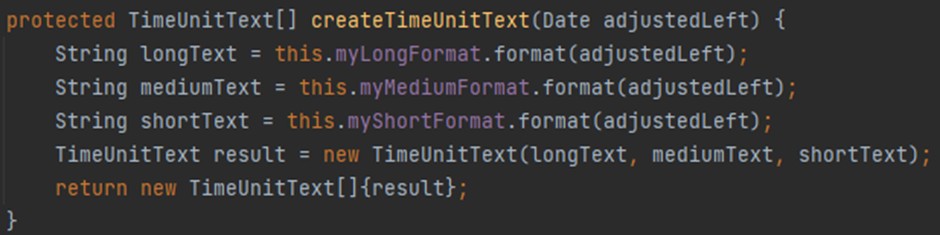
As classes WeekTextFormatter, QuarterTextFormatter, MonthTextFormatter e YearTextFormatter são as classes mencionadas anteriormente que estendem a classe abstrata.



*Implementação de WeekTextFormatter*



*Implementação de QuarterTextFormatter*



*Implementação de MonthTextFormatter*



*Implementação de YearTextFormatter*

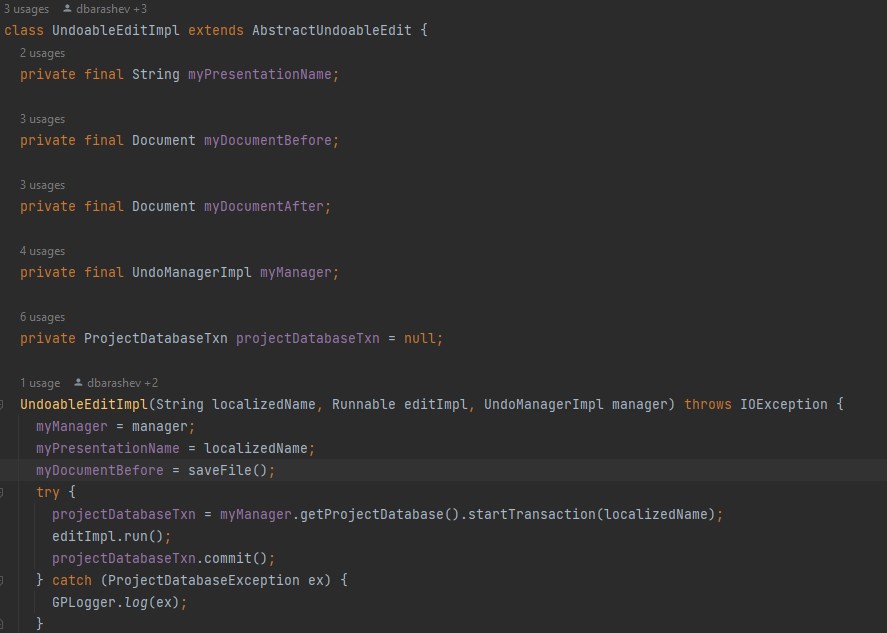
## Revisto por:

Bruno David Vasco Malta

# Memento

Package: net.sourceforge.ganttproject.undo (ganttproject.src.main.java)

A classe UndoableEditImpl implementa o Memento Pattern uma vez que guarda duas versões de um objeto Document (myDocumentBefore e myDocumentAfter), permitindo, assim, executar operação de “redo” e “undo”.



*Implementação de UndoableEditImpl onde é possível observar que são guardadas diferentes versões de um objeto*



*Método de redo*



*Método de undo*

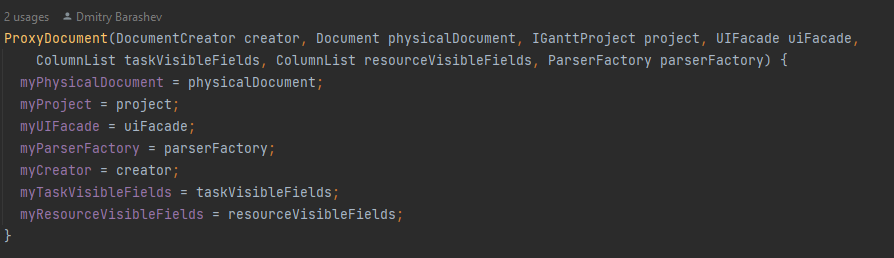
## Revisto por:

Rafael Mira

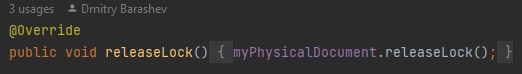
# Proxy

Package: net.sourceforge.ganttproject.document (ganttproject.src.main.java)

Tal como o próprio nome indica, a classe ProxyDocument demonstra o design pattern Proxy. Esta implementa a mesma interface (Document) que a classe do objeto real (myPhysicalDocument) que contém e sobre a qual delega algum dos pedidos.



*Construtor da classe ProxyDocument*



*Exemplo de um método que demonstra a delegação de pedidos para o objeto original*

## Revisto por:

Gonçalo Cerveira