

TransactionMS

# Pruebas unitarias y checkStyle



YULY SANDRA CHOQUE RAMOS

Contenido

[Pruebas unitarias y checkStyle 0](#_Toc184145523)

[Tecnologías Utilizadas 2](#_Toc184145524)

[Pruebas unitarias con Junit y Mockito: 2](#_Toc184145525)

[TransactionControllerTest 3](#_Toc184145526)

[TransactionCreatorTest 3](#_Toc184145527)

[TransactionCreatorTest 4](#_Toc184145528)

[TransactionServiceImplTest 5](#_Toc184145529)

[DepositOperationImpl 5](#_Toc184145530)

[TransferOperationImplTest 6](#_Toc184145531)

[WithdrawalOperationImplTest 6](#_Toc184145532)

[TransactionServiceTTest 6](#_Toc184145533)

[AccountServiceTTest 7](#_Toc184145534)

[Cálculo de cobertura de código (jacoco) 7](#_Toc184145535)

[Cobertura del proyecto TransactionMS 7](#_Toc184145536)

[Revisión del COVERAGE en Intellij IDEA 9](#_Toc184145537)

[Checkstyle 9](#_Toc184145538)

[Implementacion del archivo CheckStyle.xml 9](#_Toc184145539)

# Tecnologías Utilizadas

JUnit:

Se utiliza **JUnit** para las pruebas unitarias. Permite la creación de casos de prueba para verificar que los microservicios funcionen como se espera y garantizar que los cambios futuros no rompan funcionalidades existentes.

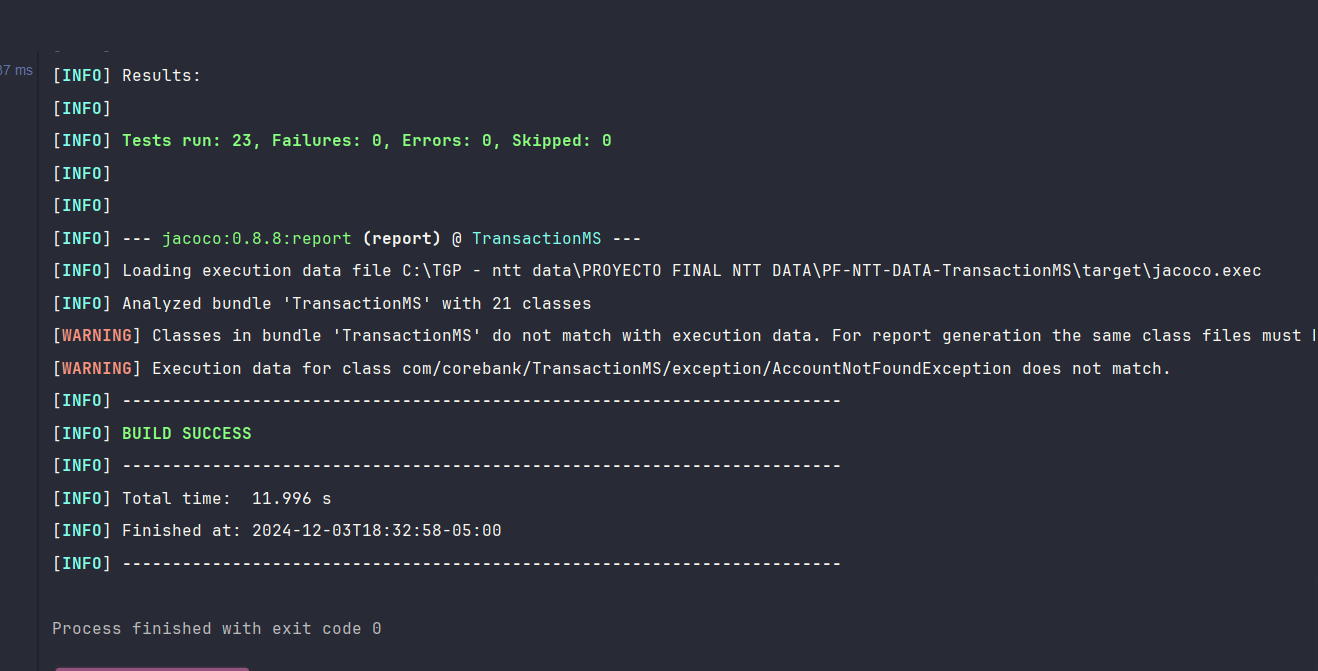
Mockito:

**Mockito** se usa para crear mocks y stubs que simulan el comportamiento de dependencias externas. Esto asegura que las pruebas unitarias estén aisladas de servicios externos como bases de datos y APIs de terceros, resultando en pruebas más confiables y mantenibles.

JaCoCo:

**JaCoCo** está integrado para medir la cobertura de código. Proporciona información detallada sobre la efectividad de las pruebas unitarias, asegurando que las áreas críticas del código estén bien probadas.

# Pruebas unitarias con Junit y Mockito:



Se realizo la implementación de pruebas unitarias de los métodos y diferentes validaciones según la lógica de los métodos.

## TransactionControllerTest

* **testPostDeposit**



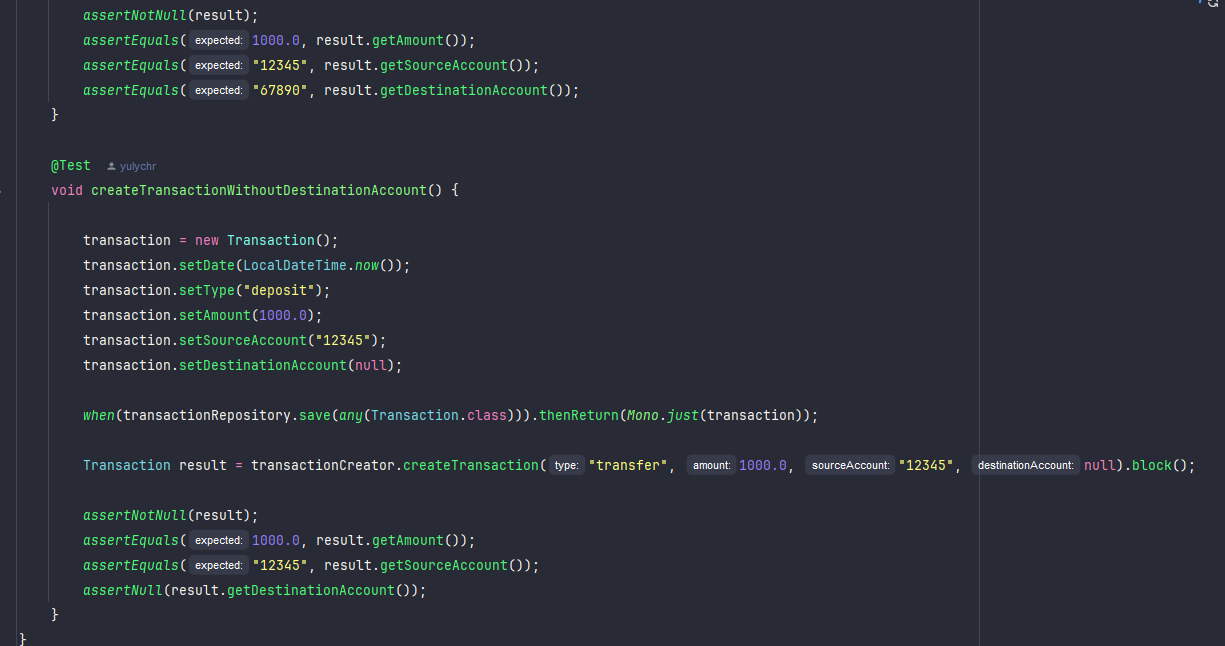
* **testPostWithdrawal**



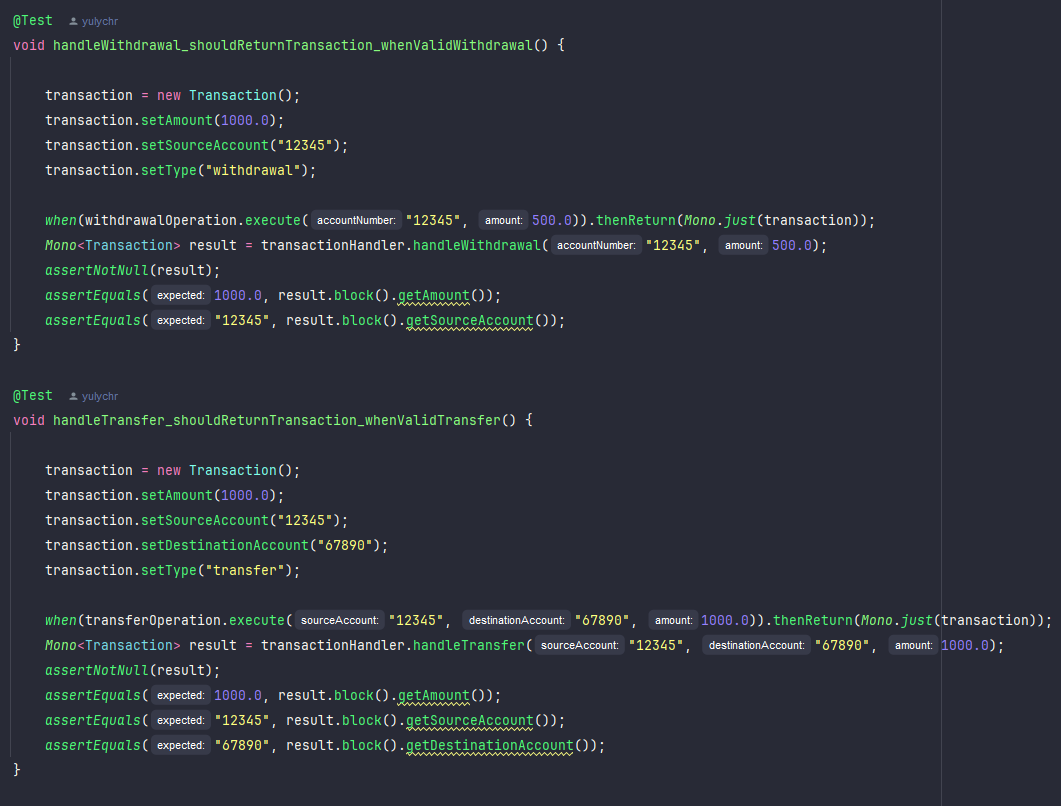
* **testPostTransfer**



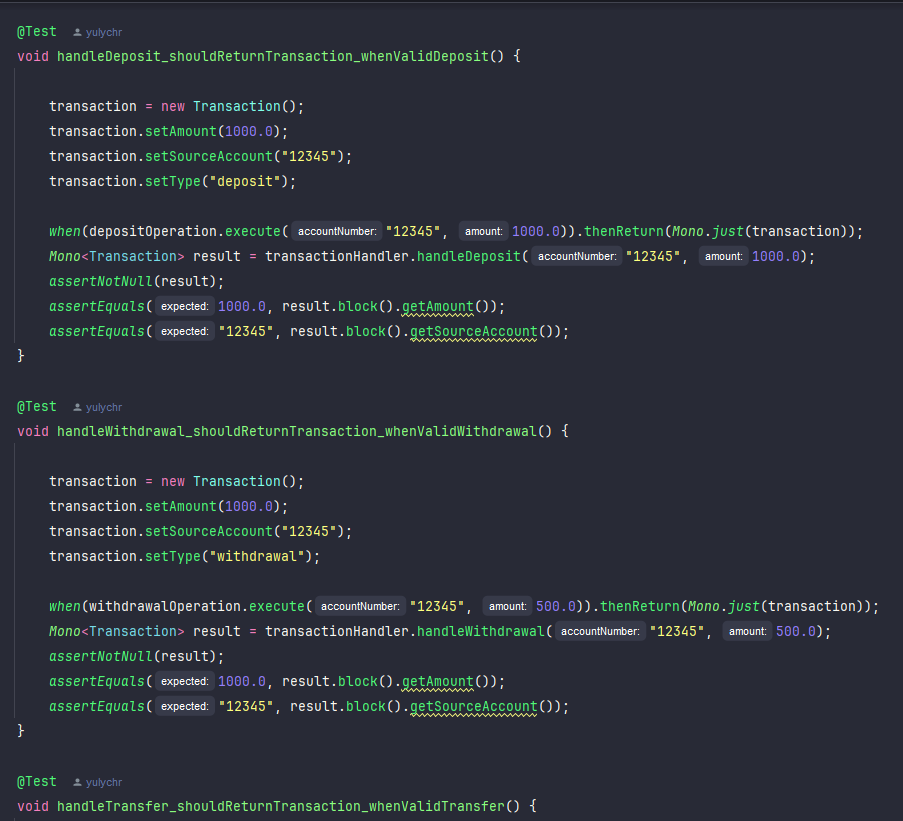
## TransactionCreatorTest



## TransactionCreatorTest



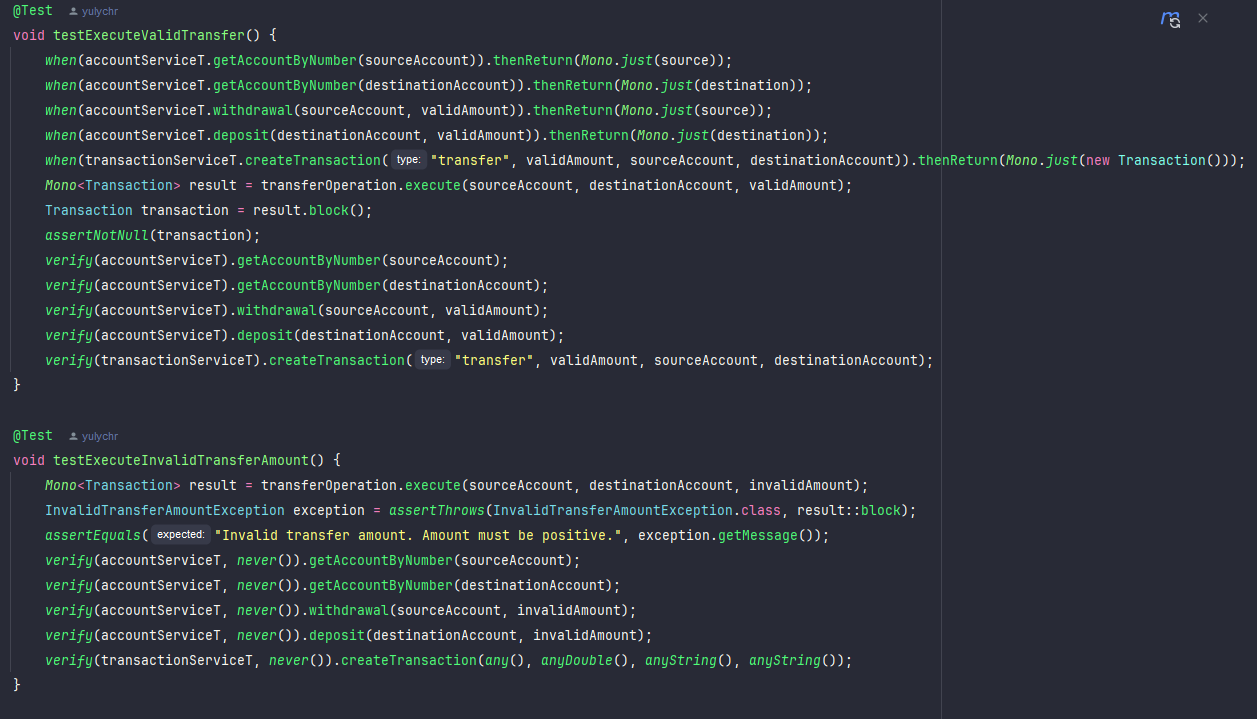
## TransactionServiceImplTest



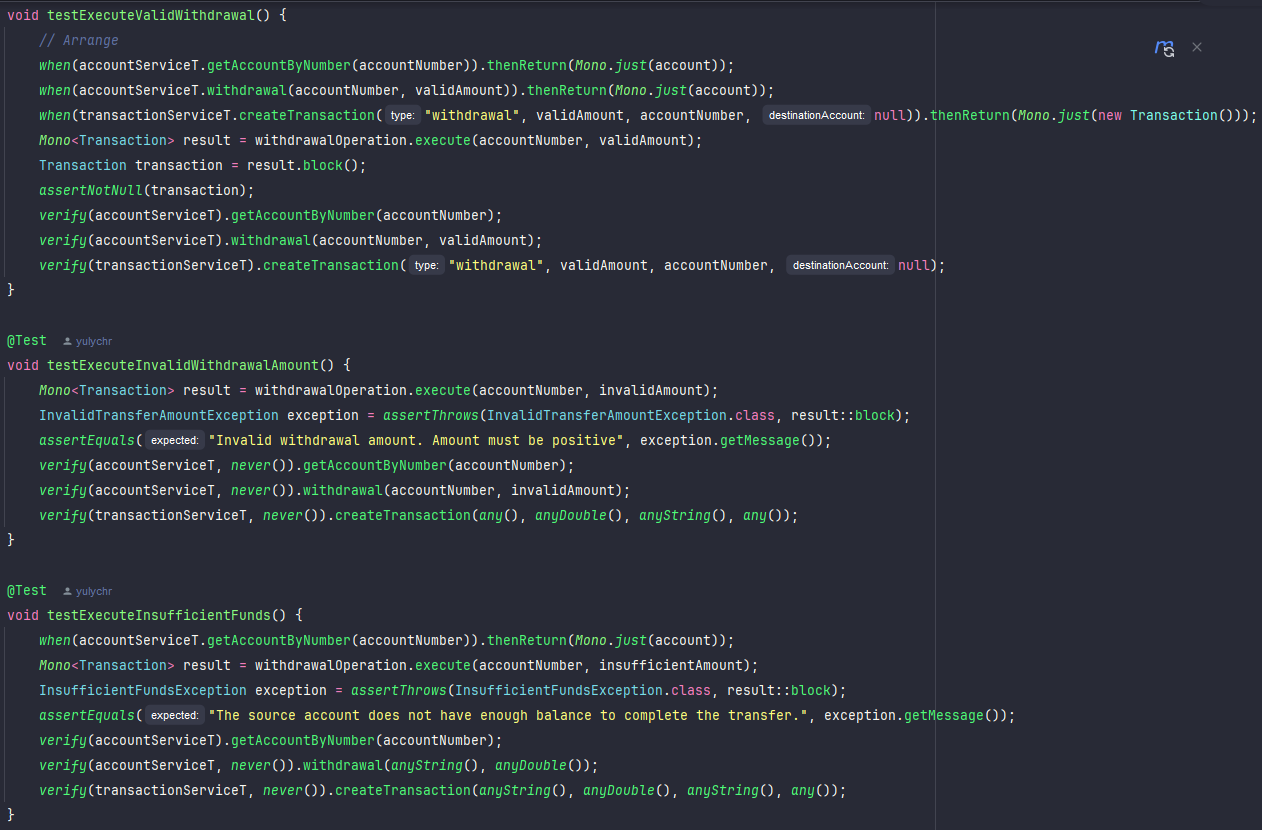
## DepositOperationImpl



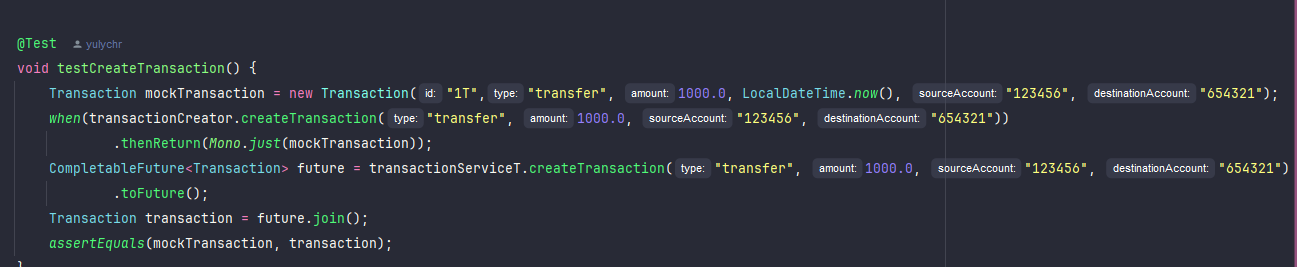
## TransferOperationImplTest



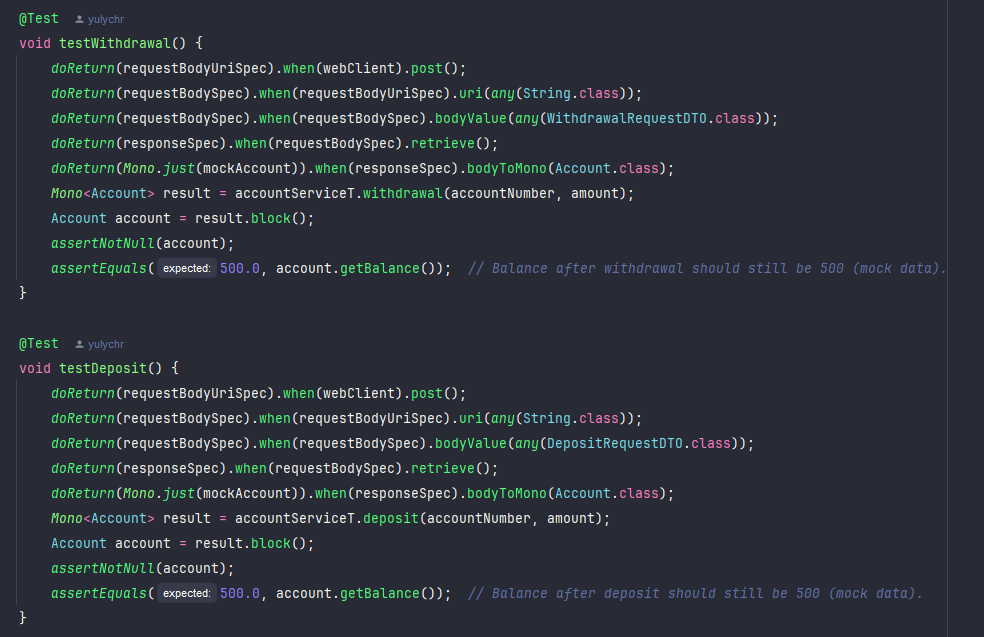
## WithdrawalOperationImplTest



## TransactionServiceTTest

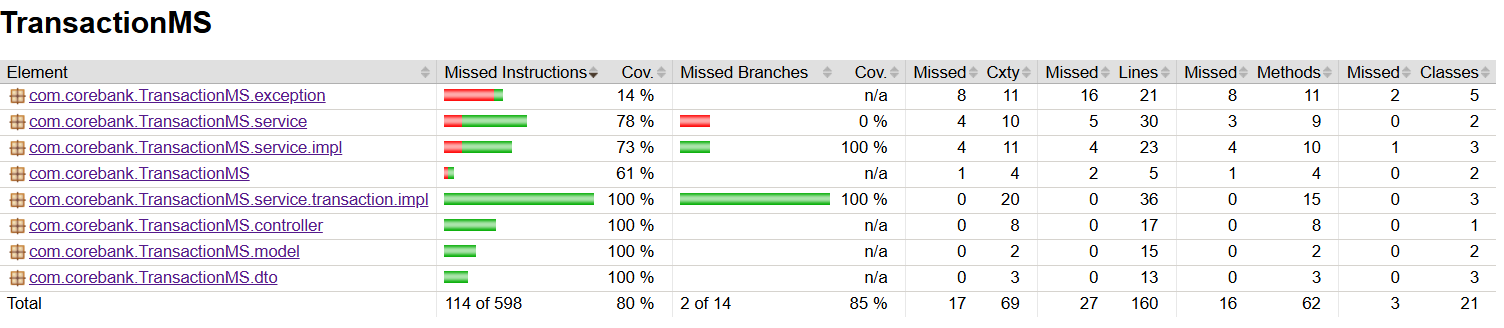


## AccountServiceTTest

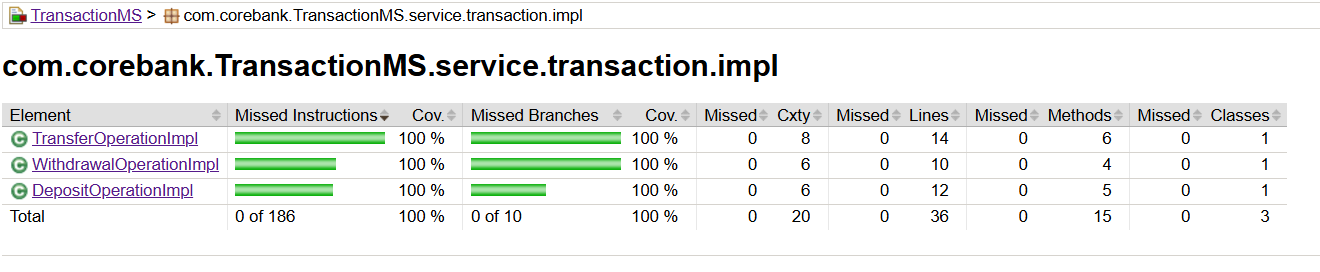


# Cálculo de cobertura de código (jacoco)

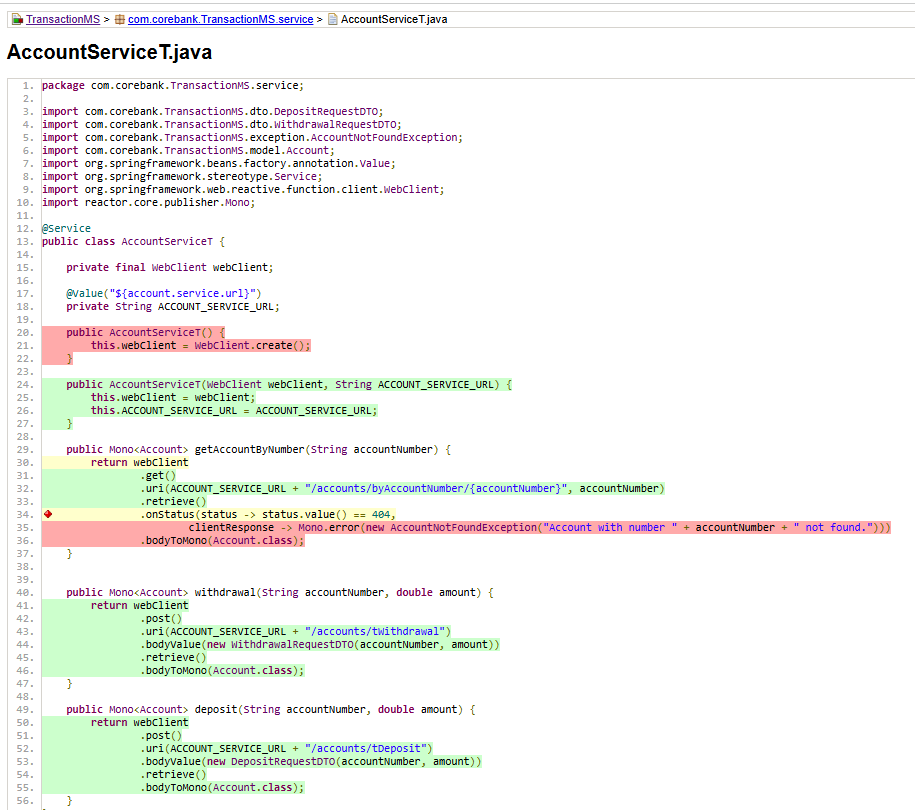
## Cobertura del proyecto TransactionMS



Cobertura de transaction.impl

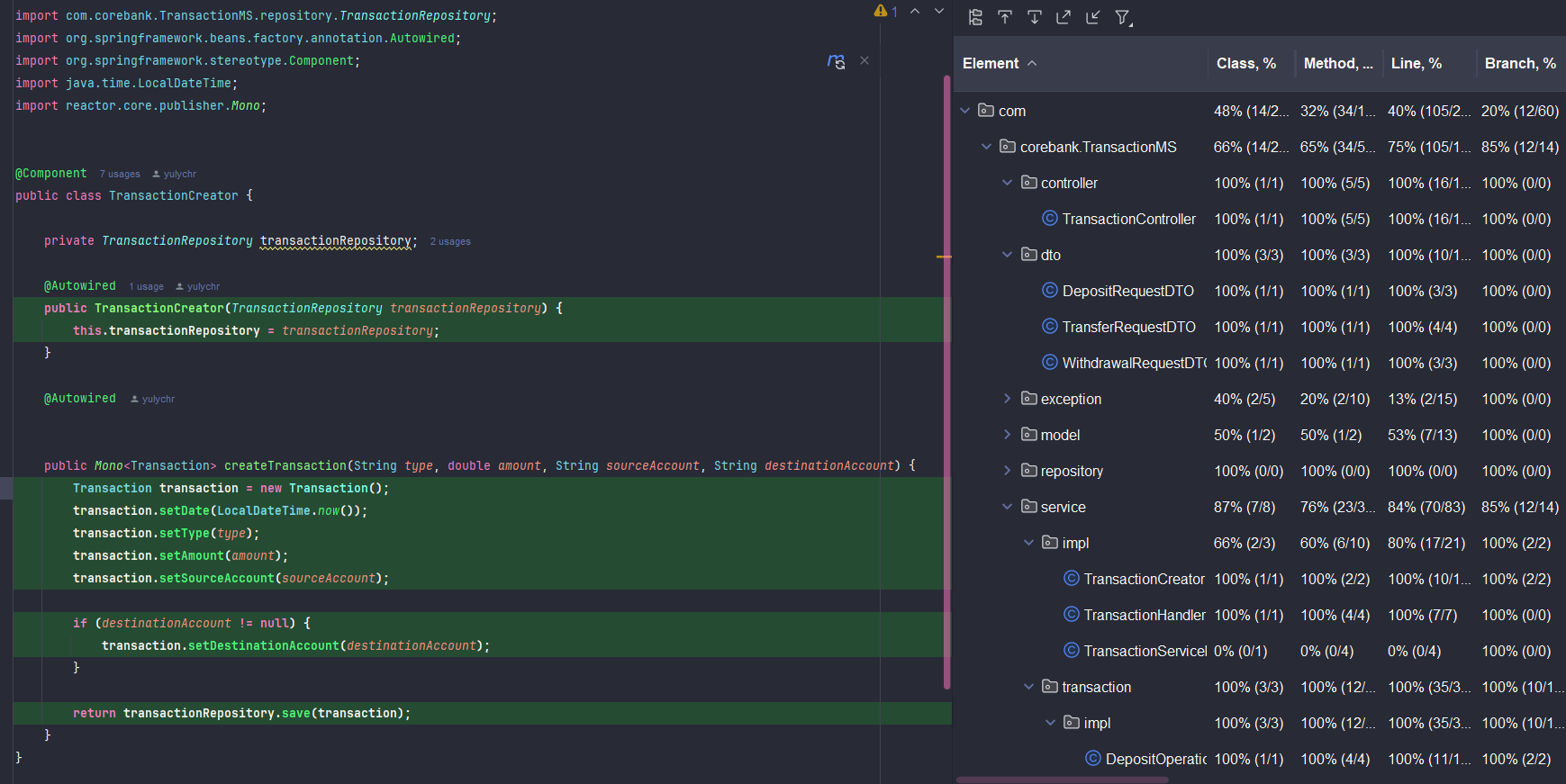


Revisión detallada de la cobertura



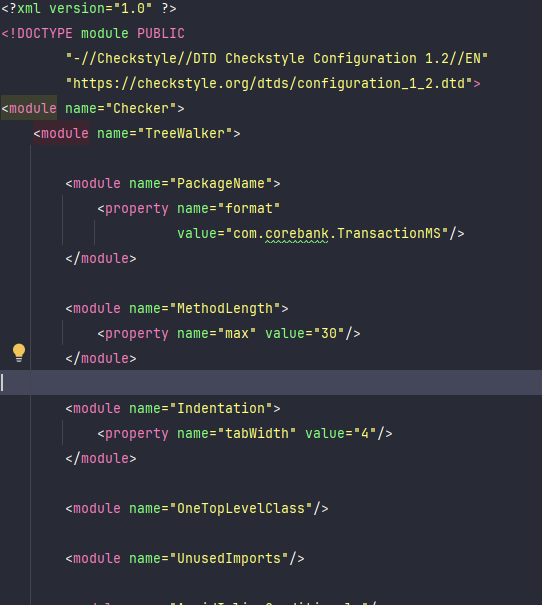


## Revisión del COVERAGE en Intellij IDEA



# Checkstyle

## Implementacion del archivo CheckStyle.xml



Revisión del del proyecto con CheckStyle

