



PBTest



Matheus Ale



“

O sistema tem como objetivo ser uma ferramenta que auxilie na documentação e visualização e execução de testes de software utilizando o framework de teste de unidade JUnit.

1.

Mas o que é
teste de
software?



Vamos falar um
pouco sobre
testes!



Por que testar?



Para que testar?

Teste para aumentar a confiança de sistemas de software.



Como?

Executar o sistema com um conjunto de entradas e avaliar se o conjunto de saídas é válido.

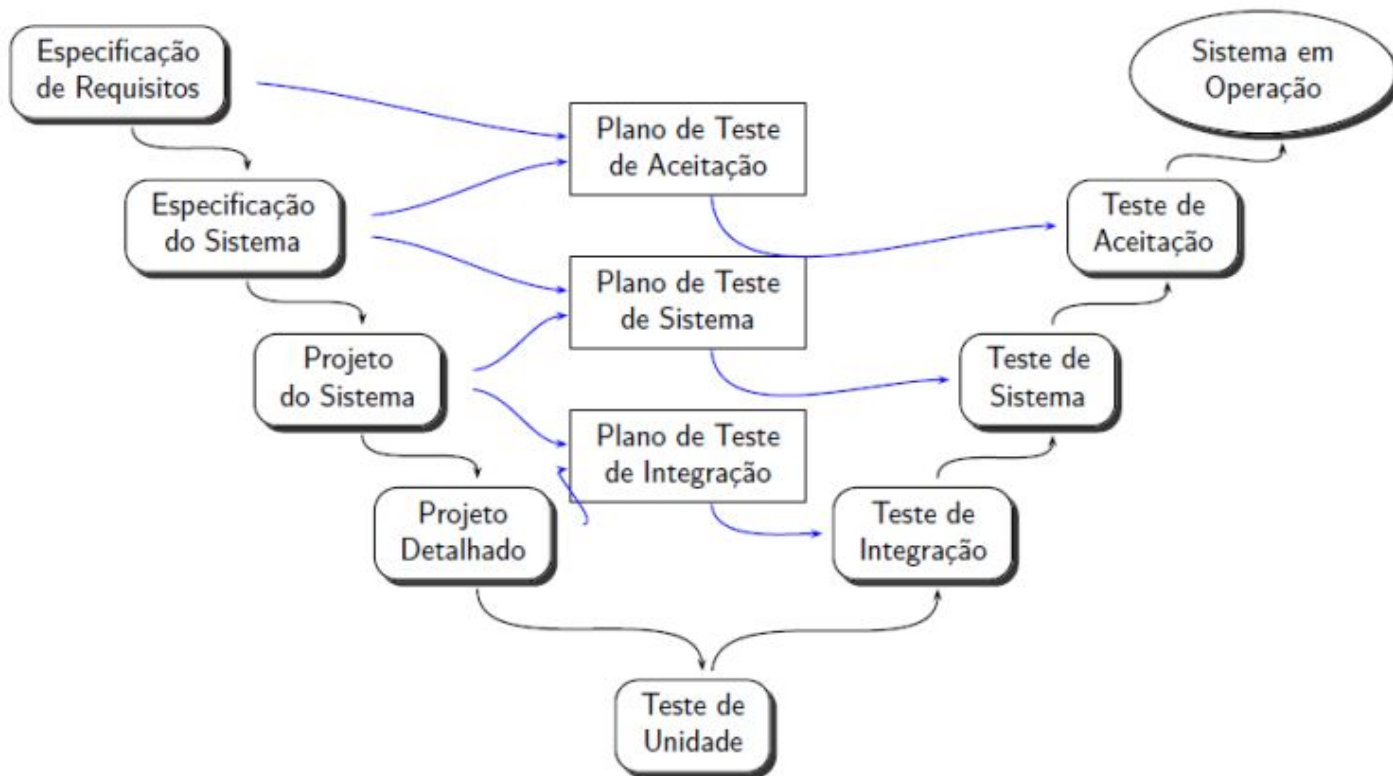


Testar para encontrar o que?

Teste com objetivo de revelar a presença de defeitos.

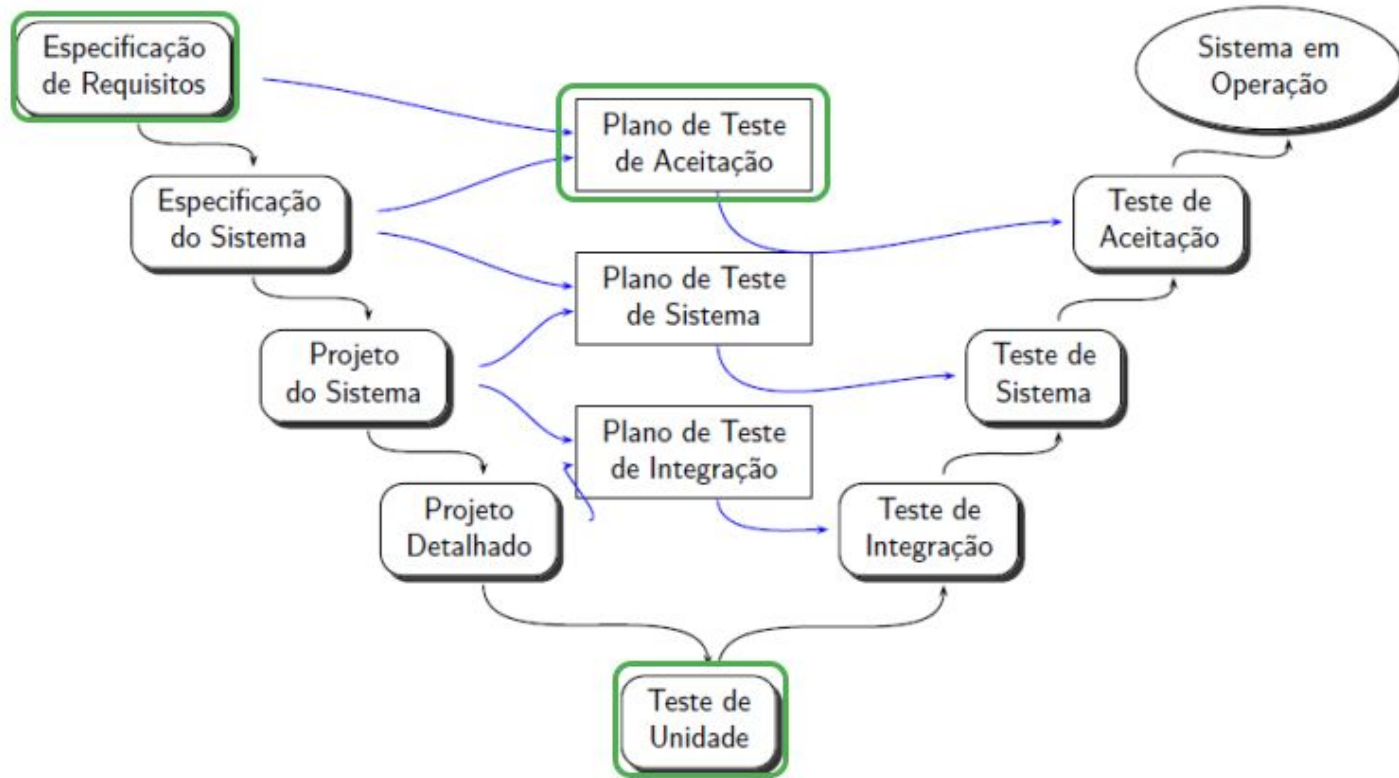


Fases da atividade de teste





O software cobre estas fases





Aprofundado... mas não muito

Especificação de requisitos

Fase da coleta dos requisitos e descoberta de funcionalidades do sistema.

Plano de teste de aceitação

Planejamento de uma sequência de teste de unidade para avaliar uma funcionalidade.

Teste de unidade

O teste de unidade concentra esforços na menor unidade do projeto de software.

2.

Requisitos do sistema



Vamos falar um
sobre os
requisitos



Requisitos do sistema

Com base nestas fases foram extraídos os requisitos do sistema.



Principais requisitos de sistema

O usuário pode criar um projeto

Para um projeto ser criado é necessário que o usuário informe nome do projeto, descrição, prefixos e os diretórios de sistema para os artefatos do projeto e para os casos de teste.

O usuário pode especificar casos de uso

Um caso de uso é composto de uma descrição, título, objetivo e atores. Todos esses atributos são obrigatórios.

O usuário pode vincular casos de teste a artefatos

É composto por: nome descrição textual que explique em linguagem natural o que está sendo testado. Além de ser obrigatoriamente **vinculado a um caso de uso**.



Principais requisitos de sistema

O usuário pode criar sequências de teste

Uma sequência de testes deve conter nome, descrição e **pelo menos um caso de teste**.

O usuário pode executar sequências de teste

Uma sequência de teste deve poder ser **executada pelo sistema**.

O usuário pode ver o resultado dos teste

Após a execução de uma sequência, o usuário pode **ver o resultado** dos teste de unidade.

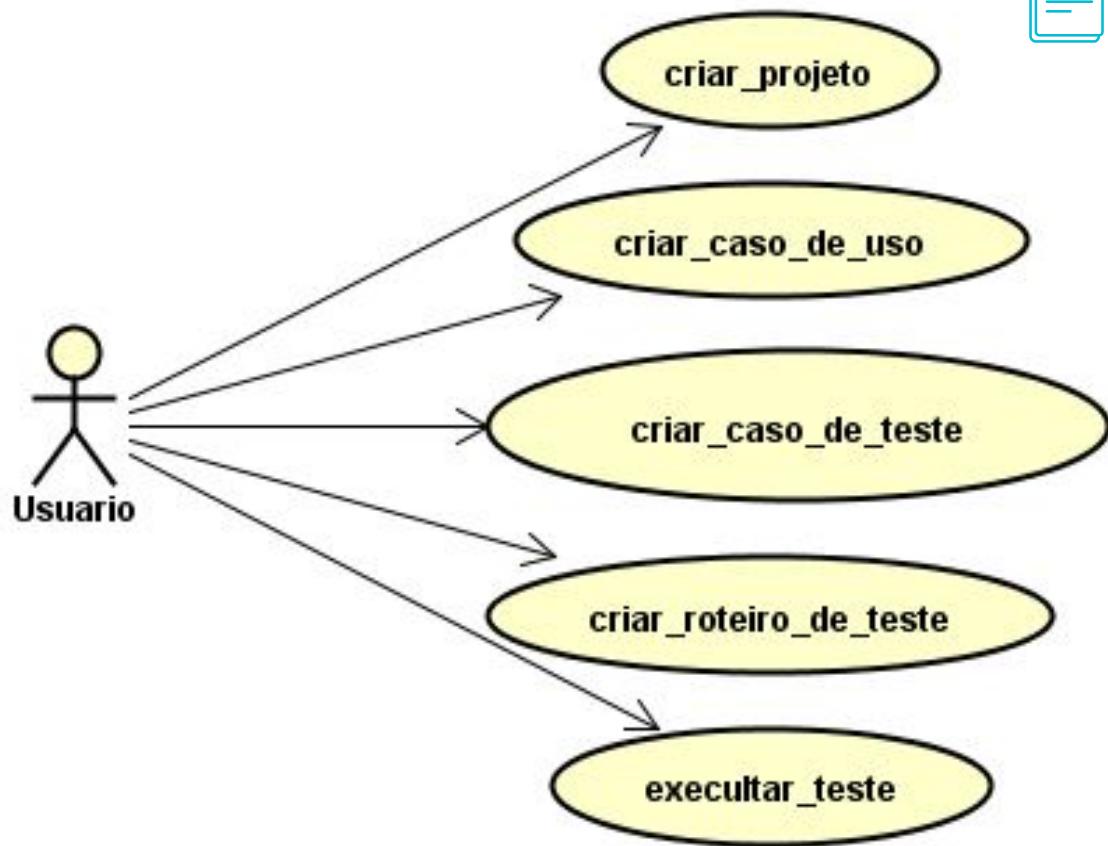
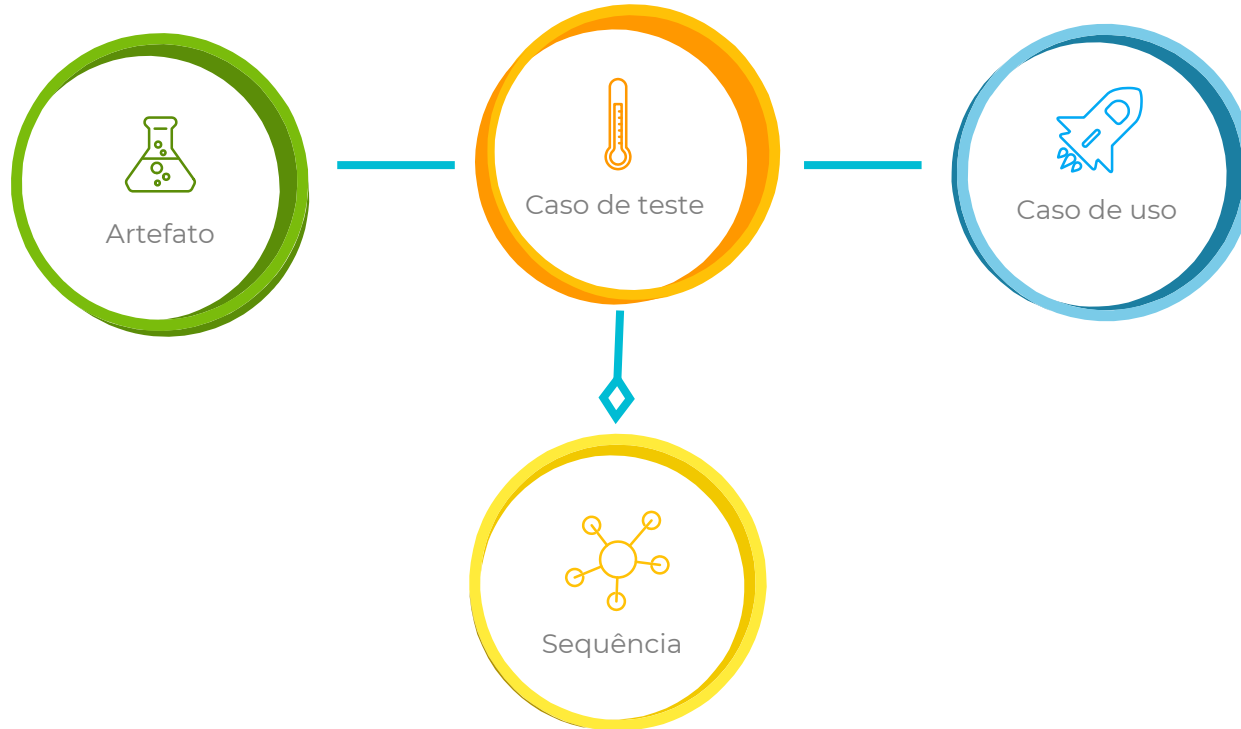


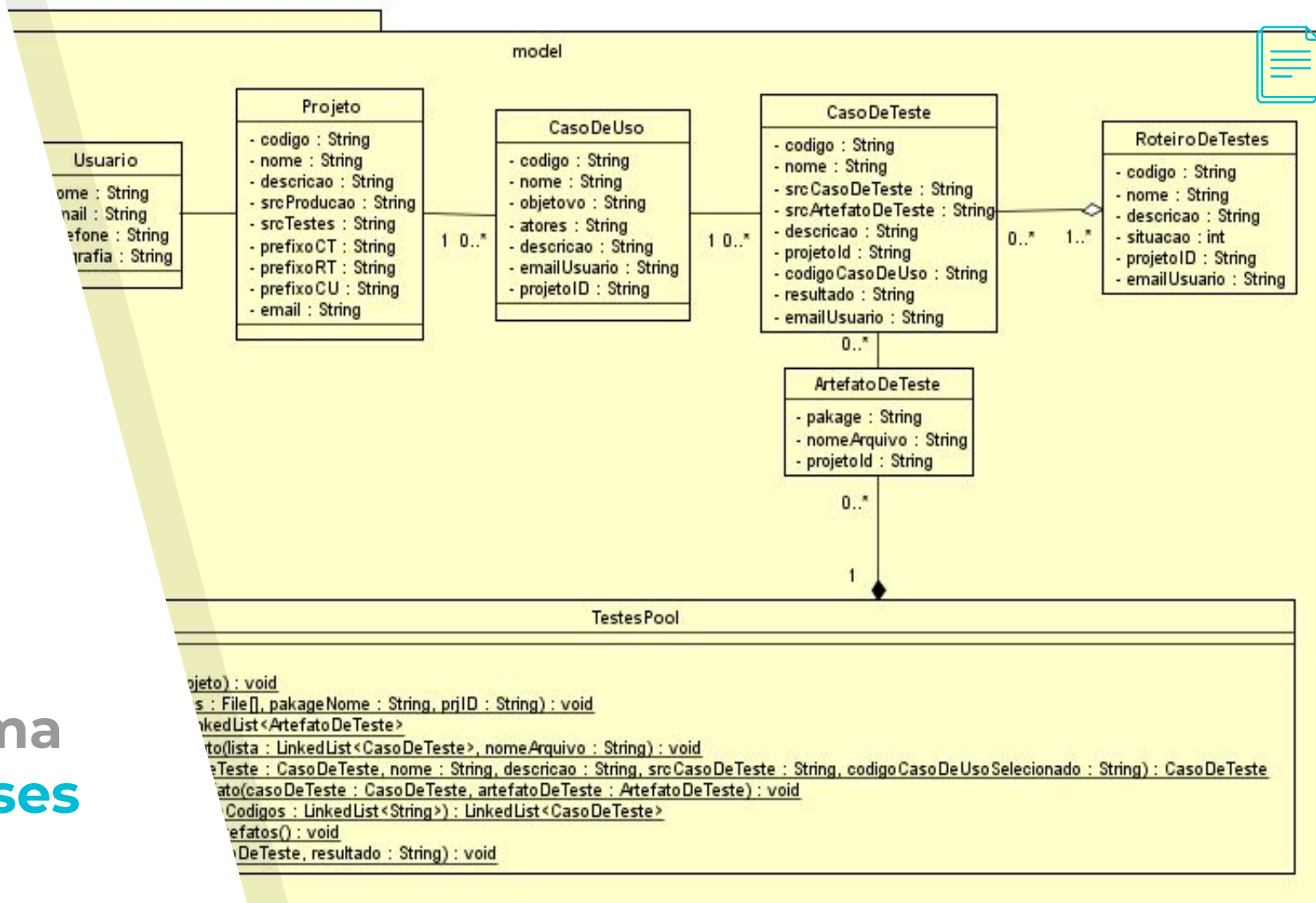
Diagrama
de **casos**
de **uso**



Relação entre as informações dentro do sistema



No diagrama de classes





Criar um **Projeto**

Como criar um projeto dentro do sistema.



Criar um **Caso** **de Uso**

Como criar um caso de uso dentro do sistema.



Criar um **Caso** **de teste**

Como criar um caso de teste dentro do sistema.



Caso de uso do sistema

Como o usuário do sistema cria um caso de teste?



Caso de uso do sistema

Pré-condições:

Ter um projeto ativo; Estar na tela de casos de teste;



Cenários Principais:

1 - Selecionar um artefato de teste.



2 - Clicar no botão “novo caso de teste”



3 - Informar o nome e descrição do caso de teste.



4 - Vincula o caso de teste a um caso de uso existente.

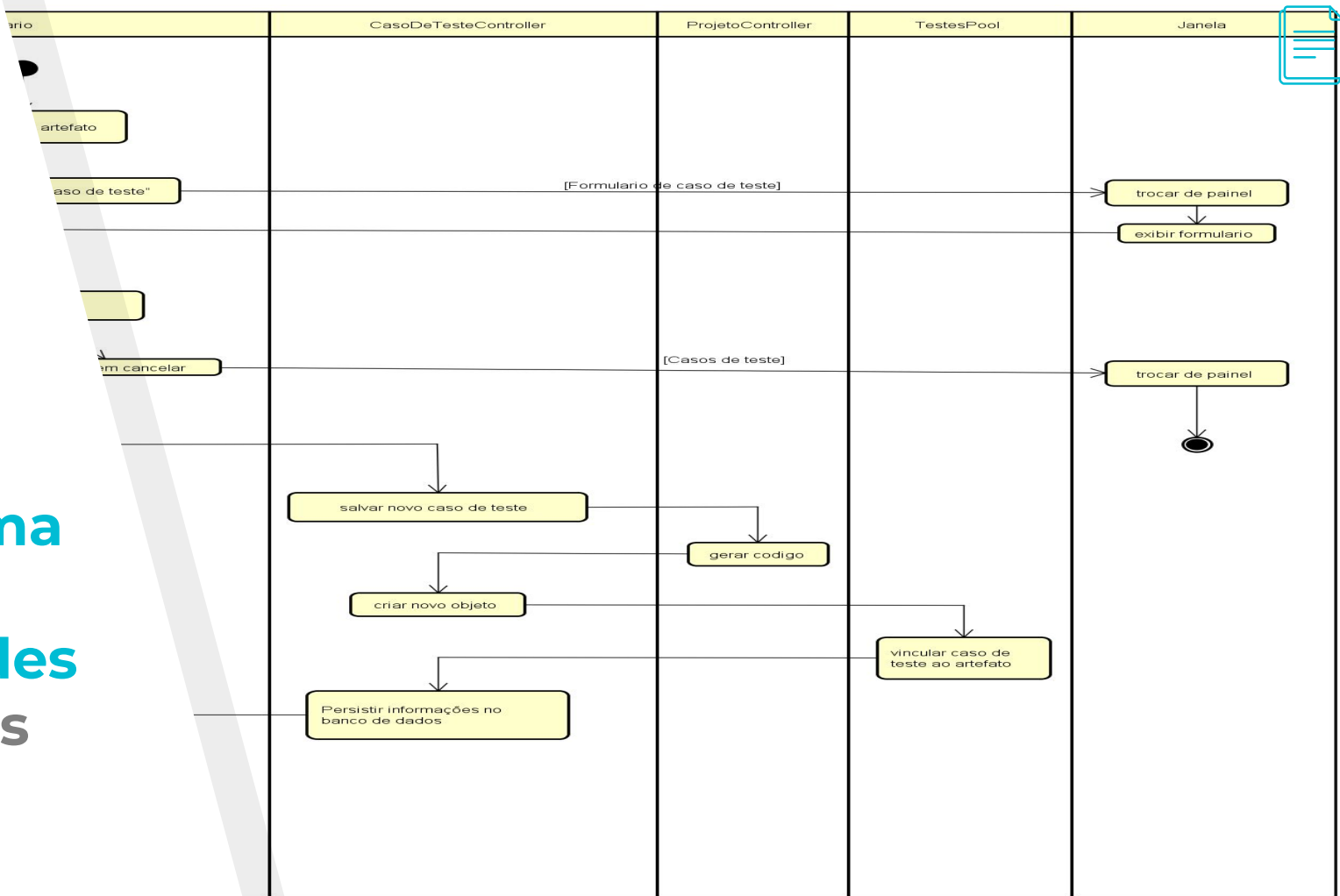


5 - Informar a classe de teste.



6 - Clicar no botão “salvar”.

Diagrama de atividades do casos de uso





Criar um **Roteiro** **de teste**

Como criar um sequência de casos de teste dentro do sistema.



Executar um **Roteiro de teste**

Como executar uma sequência de casos de teste dentro do sistema.

3.

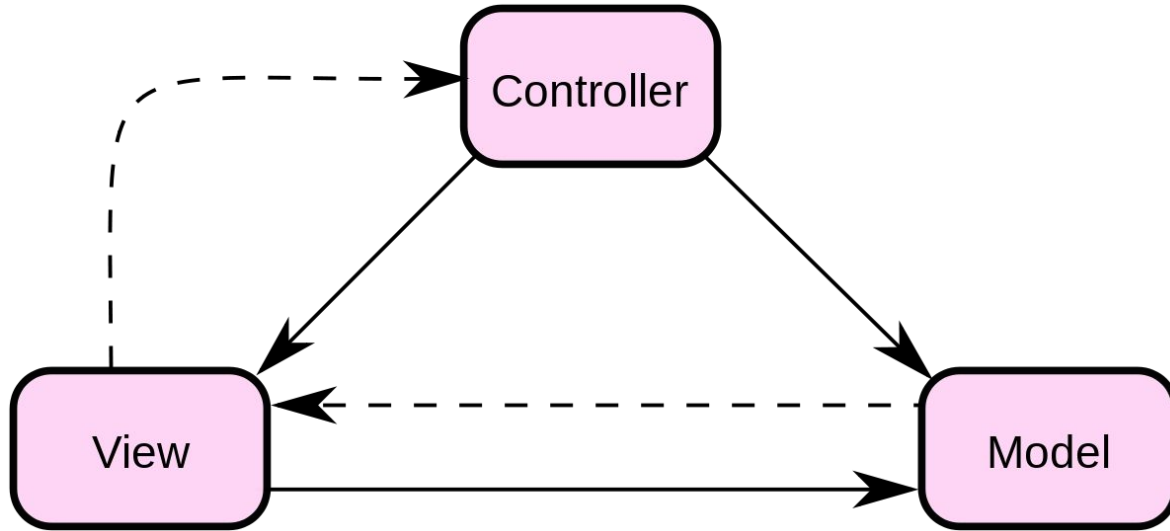
Modelagem do sistema



Vamos falar um
sobre a
modelagem

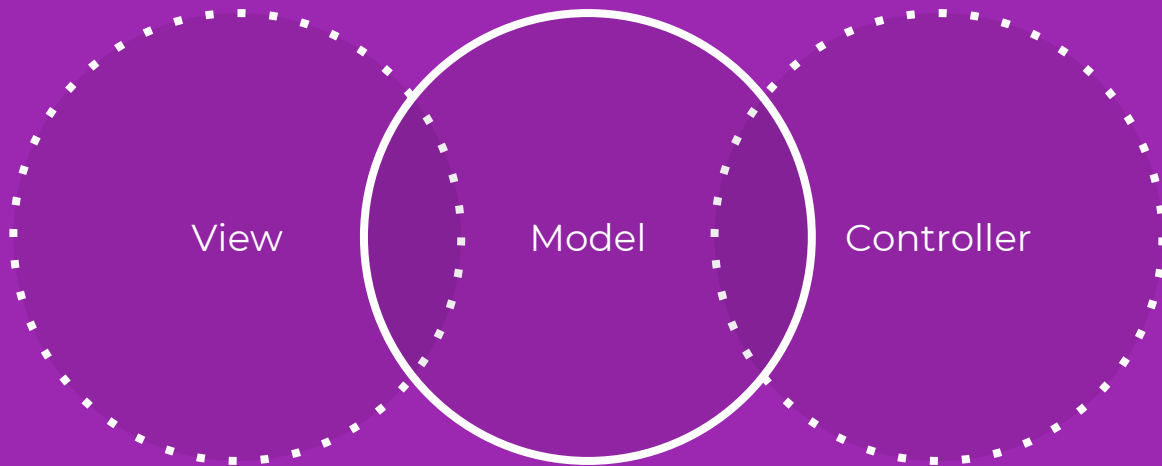


Padrão arquitetural do sistema



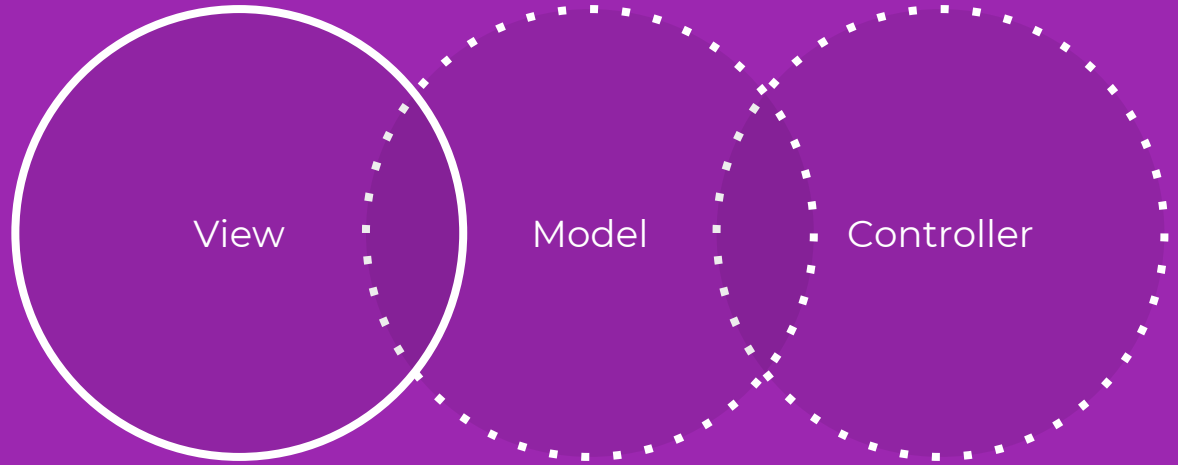


No
diagrama
de **classes**



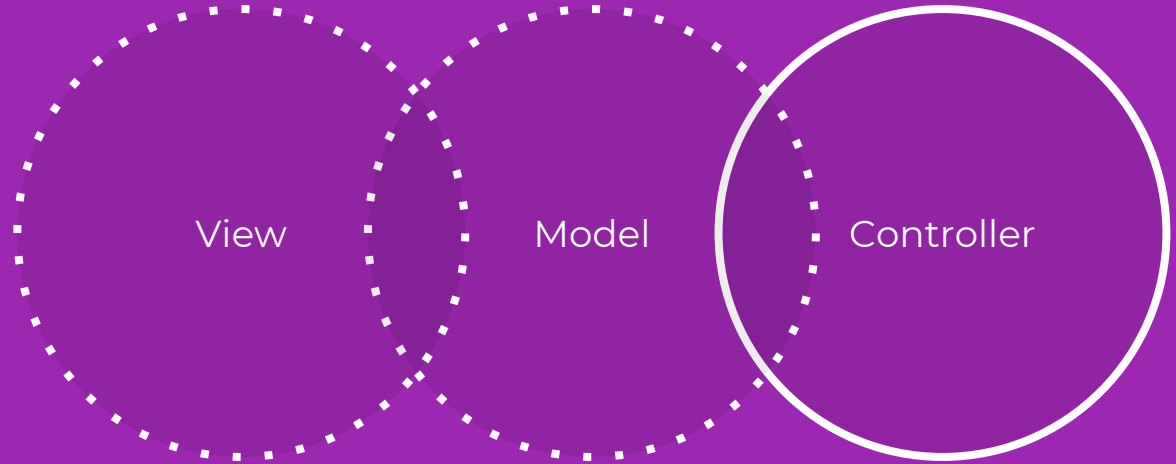


No
diagrama
de **classes**





No
diagrama
de **classes**



4.

Trabalhos futuros



Melhorias que
futuras para o
sistema.



Novas funcionalidades para o sistema

Relatorios

Implementar a extração da documentação de um projeto para o formato pdf ou odg.

Compilar as classes

Fazer com que o sistema compile as classes. Sem bagunçar os arquivos do projeto.

Estatísticas

Painel que permita ao usuário visualizar estatísticas sobre um projeto. Porcentagem de falhas e acertos de e casos de teste, erros em um caso de uso, etc.

Relacionamento entre artefato e documentação

Mostrar em um grafo ou diagrama a relação entre os casos de uso, artefatos casos de teste e sequências.

Leitura de arquivos de propriedades

Ler o documento de propriedades de um projeto gerado pela IDE.

Execução modular

Criar Módulos de execução de teste. Para que a execução de teste seja independente do sistema. Para permitir a execução de testes em projetos implementados em outras linhagens.



Referencias

CARLOS MALDONADO, J. et al. Introdução ao teste de software. Instituto de Ciências Matemáticas e de Computação, v. 65, n. 01, 2004.

FUNABASHI JORGE, P.; HENRIQUE KUROISHI, P. Introdução ao Teste de Software: técnicas, ferramentas e pesquisas atuais, 2016.

ANICHE, M. Testes automatizados de software Um guia prático. 1. ed. Vila Mariana – São Paulo – SP – Brasil: Casa do Código, 2017.



THANKS!

Perguntas?

Todos os arquivos do projeto estão disponíveis em:

<https://github.com/matheusAle/PBTest>