

1. Explique o conceito de teste unitário?

O conceito do teste unitário é fazer testes isolados, visando que aquela parte em específico resulte de acordo com o que estava previsto, seja para sucesso do método ou alguma exceção já esperada.

2. Descreva como fazer um código de teste.

Utilizando a IDE IntelliJ é possível fazer uma classe teste com os comandos Alt + insert na classe que deseja testar, vai abrir uma opção de ações e selecione teste, feito isso, será gerado uma classe de teste derivada da classe que executou o comando.

Já o código deve ser pensado de forma isolado, deve entender a entrada do método e a saída, injetar a dependência necessária, e mockar as dependências da dependência, exemplo.: no contexto do spring boot o teste de um serviço, deve injetar com @Autowired o objeto service, e utilizar a anotação @MockBean para as dependências como o objeto repository.

3. Qual o intuito do teste unitário?

Validar se o método de forma isolada e sem depender de outras funcionalidades, funciona de acordo com o planejado

4. Quais são as ferramentas que utilizamos para realizar testes unitários(2 pelo menos).

Junit e MockMvc

5.

Busca os lançamentos com sucesso	
Text Stack Trace	Explicação
No tests found matching Method buscarTodosLancamentosPorDependen tes	O método não havia sido anotado com a anotação @Test
Não há lançamentos cadastrados com os valores fornecidos ValidateException(status=404 NOT_FOUND, msg=null, code=null, detail=null) at com.trilha.back.finacys.service.impl.Lan camentoServiceImpl.buscarLancamento PorDependentes(LancamentoServicelm pl.java:129)	O erro foi causado por que não mockei um valor na classe entidade, então ele chamou o método e seguiu o fluxo natural e não encontrou na entidade a filtragem correta. Identifica a linha e a classe em que o erro ocorreu
Não há lançamentos cadastrados com os valores fornecidos ValidateException(status=404 NOT_FOUND, msg=null, code=null, detail=null)	O erro havia se repetido pq era necessário fazer o Mock da classe de entidade de lançamento.

at com.trilha.back.finacys.service.impl.Lan camentoServiceImpl.buscarLancamento PorDependentes(LancamentoServicem pl.java:129)	
--	--

Retorno exception	
Text Stack Trace	Explicação
preencha o campo data ValidateException(status=400 BAD_REQUEST, msg=null, code=null, detail=null) at com.trilha.back.finacys.service.impl.Lan camentoServiceImpl.buscarLancamento PorDependentes(LancamentoServicem pl.java:111)	O erro ocorreu pois ainda não estava definido que o que se esperava era justamente o lançamento da exceção sendo assim, o método do serviço foi chamado e esbarrou em uma parte que não era o retorno, a exception
Expected: an instance of java.lang.NullPointerException but: <ValidateException(status=400 BAD_REQUEST, msg=null, code=null, detail=null)> is a com.trilha.back.finacys.exception.Valida teException. Stacktrace was: ValidateException(status=400 BAD_REQUEST, msg=null, code=null, detail=null) at com.trilha.back.finacys.service.impl.Lan camentoServiceImpl.buscarLancamento PorDependentes(LancamentoServicem pl.java:111)	O erro ocorreu pois o exception que foi inserido como esperado não foi o que saiu de fato foi configurado como teste o NULLPOINTEREXCEPTION retorno do método: ValidateException
Expected: (an instance of com.trilha.back.finacys.exception.Valida teException and exception with message a string containing "test") but: exception with message a string containing "test" message was "preencha o campo data" Stacktrace was: ValidateException(status=400 BAD_REQUEST, msg=null, code=null, detail=null) at com.trilha.back.finacys.service.impl.Lan	O erro ocorreu pois a mensagem de erro que me retornou foi a diferente do que configurei como esperado configurado: "test" retorno do método: "preencha o campo data"

camentoServiceImpl.buscarLancamento PorDependentes(LancamentoServiceIm pl.java:111)	
---	--