

FAST

FORMAÇÃO ACELERADA EM
SOLUÇÕES DE TECHDESIGN



C . E . S . A . R .

Metodologias Ágeis e BDD

Todos os direitos reservados.

Este material, ou qualquer parte dele, não pode ser reproduzido,
divulgado ou usado de forma alguma sem autorização escrita.



Cenários Eficientes com BDD e Cucumber

Metodologias Ágeis e BDD

Aula 3



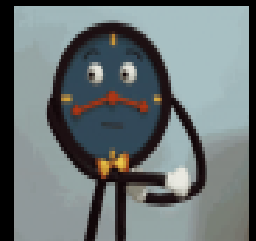
O que veremos hoje?

- Conteúdo 01 - Escrita e Análise de Cenários com Gherkin e Cucumber
- Conteúdo 02 - Ferramentas e boas práticas na construção de cenários
- Conteúdo 03 - Prática guiada

Bora revisar?

1. O que é entrega contínua e como ela afeta o papel do QA?
2. Diferença entre critérios de aceite e definição de pronto?
3. Como o BDD melhora a colaboração entre Dev, QA e PO?
4. Estrutura do Gherkin: Given – When – Then.
5. Diferença entre BDD, TDD e ATDD.

Kahoot time







Escrita e Análise de Cenários com Gherkin e Cucumber

Metodologias Ágeis e BDD

Aula 3



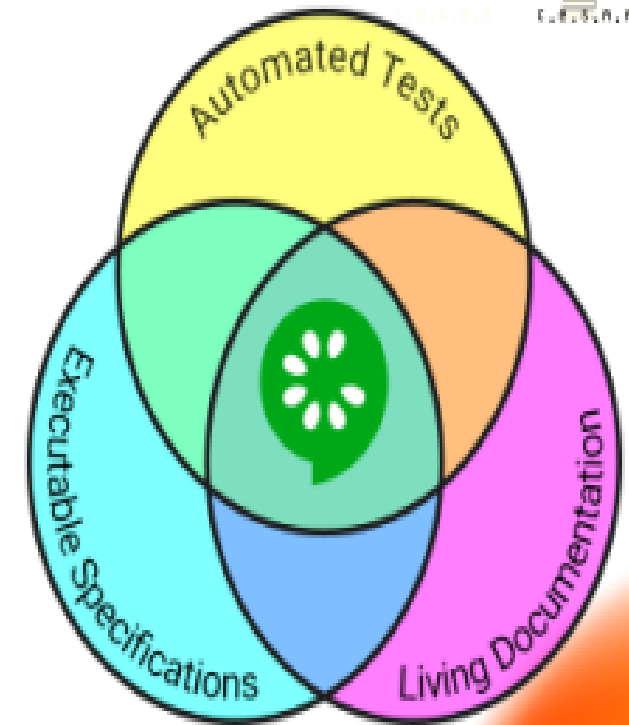
Gherkin vs Cucumber

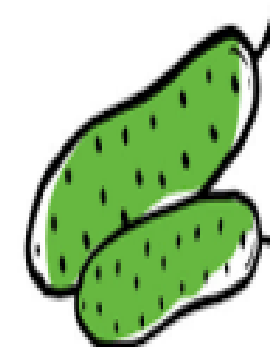
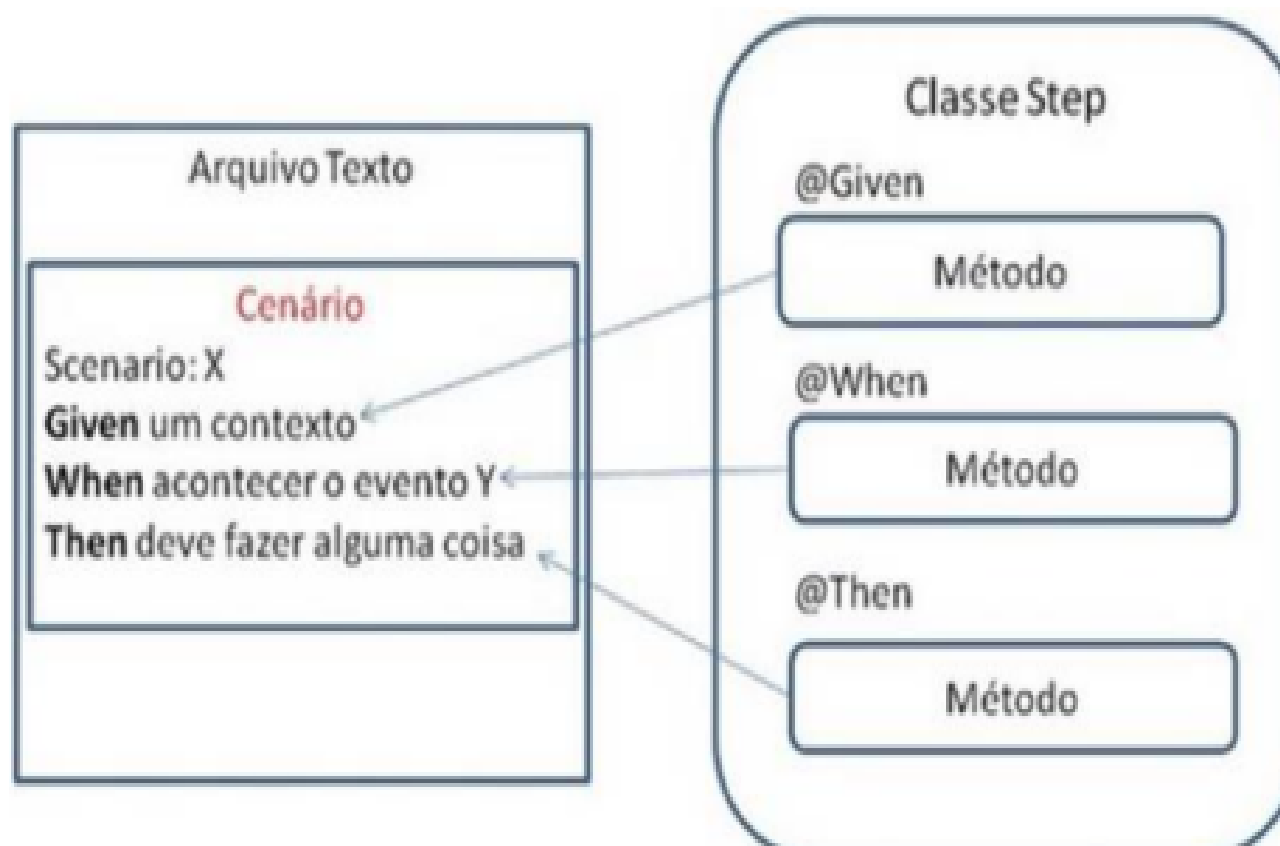
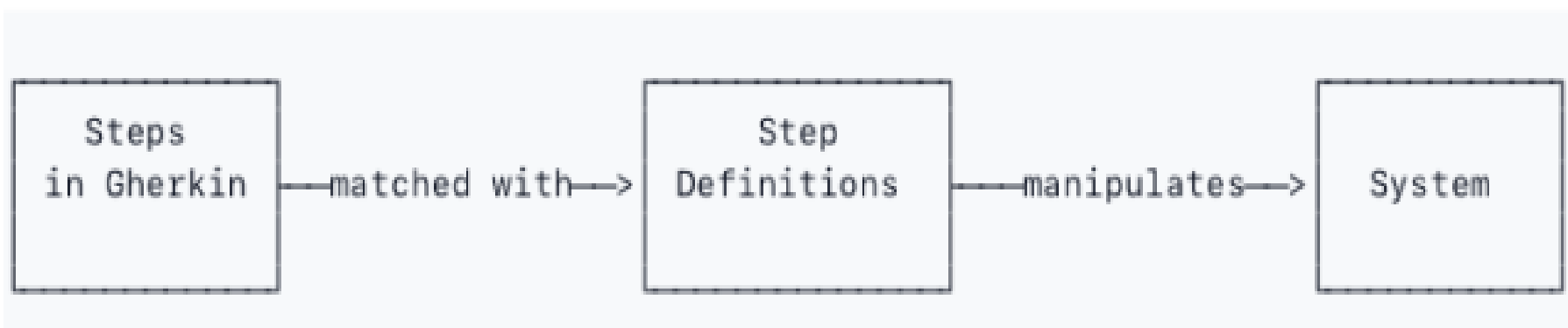
- Gherkin é uma linguagem simples, mas estruturada, utilizada para descrever o comportamento de um software, enquanto o **Cucumber** é uma ferramenta que utiliza o **Gherkin** para automatizar testes baseados nesse comportamento.
- O Cucumber lê especificações executáveis escritas em texto simples e valida se o software cumpre o que essas especificações exigem. As especificações consistem em múltiplos *exemplos* ou *cenários*
- Cada cenário é uma lista de *etapas* a serem executadas pelo Cucumber. O Cucumber verifica se o software está em conformidade com as especificações e gera um relatório indicando  sucesso ou  falha para cada cenário.
- Para que o Cucumber entenda os cenários, ele deve seguir algumas regras básicas de sintaxe, chamadas **Gherkin**.

"Resumidamente, a prática do BDD gera o Gherkin que por sua vez é utilizado por algumas ferramentas, a exemplo do Cucumber."

Para que serve o cucumber?

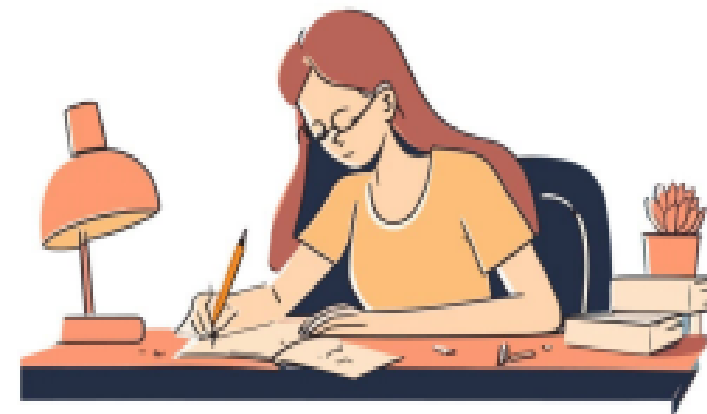
- **Leitura e execução de especificações em linguagem natural**
 - Isso permite que os membros da equipe que não são desenvolvedores compreendam o que está sendo testado.
- **Suporte para múltiplas linguagens**
 - Cucumber pode ser usado com várias linguagens de programação, incluindo Ruby, Java, e [JavaScript](#).
- **Integração com ferramentas de teste**
 - Pode ser integrado com outras ferramentas de teste como Selenium para testes de interface de usuário, ou com frameworks de mock e stub para testes de serviço.





cucumber

Bons cenários cucumber



1. Use a linguagem Gherkin
2. **Seja conciso e focado** -> Evite cenários longos e complexos.
3. **Evite detalhes irrelevantes** -> Evitando detalhes de implementação que não contribuam para o entendimento do cenário.
4. **Escreva cenários declarativos, não imperativos** -> Descreva o que o sistema deve fazer, em vez de como deve ser feito.
5. **Utilize exemplos** -> Use a estrutura "Exemplo" (Scenario Outline) para testar diferentes entradas e saídas com os mesmos passos, otimizando a cobertura de testes.
6. **Mantenha a consistência**
7. **Separe os cenários** -> Utilize linhas em branco para separar os cenários e facilitar a leitura do arquivo.
8. **Use ganchos (hooks)** -> Utilize ganchos como @Before e @After para configurar e limpar o ambiente de teste, evitando repetição de código.
9. **Colabore com a equipe**

Vamos praticar

Scenario: Teste de login

Given o sistema está rodando

When o usuário tenta entrar

Then ele entra no sistema

1. Termos vagos ("o sistema está rodando")
2. Ação imprecisa ("tenta entrar")
3. Sem clareza no resultado esperado



Onde está o erro?

Vamos tentar de novo?



Scenario: Login com sucesso

Given o usuário está na página de login

When ele informa um e-mail e senha válidos

Then ele deve ver a página inicial do sistema

Passo a passo: do requisito ao cucumber

Vamos usar como exemplo o seguinte requisito funcional:

Como cliente, quero recuperar minha senha, para poder acessar minha conta caso eu a esqueça.

- ☐ Etapa 1: Entenda o comportamento esperado
 - ☐ Quando o usuário pode usar essa funcionalidade?
 - ☐ O que deve acontecer se o e-mail for válido?
 - ☐ O que deve acontecer se o e-mail não estiver cadastrado?
- ☐ Etapa 2: Escreva os critérios de aceite
 - ☐ Ex1: O sistema deve enviar um e-mail com link de recuperação válido por 24h
 - ☐ Ex2: Se o e-mail informado não estiver cadastrado, uma mensagem deve ser exibida

Passo a passo: do requisito ao cucumber

Como cliente, quero recuperar minha senha, para poder acessar minha conta caso eu a esqueça.

❑ Etapa 3: Escreva os cenários em Gherkin

Feature: Recuperação de senha

Scenario: Recuperação com e-mail válido

Given o usuário está na página "Esqueci minha senha"

When ele informa um e-mail válido

Then um e-mail de recuperação deve ser enviado

Scenario: Recuperação com e-mail não cadastrado

Given o usuário está na página "Esqueci minha senha"

When ele informa um e-mail inexistente

Then uma mensagem de erro deve ser exibida

Passo a passo: do requisito ao cucumber

Como cliente, quero recuperar minha senha, para poder acessar minha conta caso eu a esqueça.

- ❑ **Etapa 4: Automatize os cenários com Cucumber** (Exemplo em JavaScript com Cypress + Cucumber)

```
Given('o usuário está na página "Esqueci minha senha"', () => {  
  cy.visit('/esqueci-senha')  
})  
  
When('ele informa um e-mail válido', () => {  
  cy.get('#email').type('teste@exemplo.com')  
  cy.get('#enviar').click()  
})  
  
Then('um e-mail de recuperação deve ser enviado', () => {  
  cy.contains('Um e-mail foi enviado').should('be.visible')  
})
```

INTERVALO!



20 min.



Ferramentas e boas práticas na construção de cenários

Metodologias Ágeis e BDD

Aula 3



Ferramentas BDD em destaque 2025

© CESAR 2025 | Todos os Direitos Reservados



beanSpec

cucumber



SMARTBEAR

specflow

ACCELO

jbehave

FitNesse

Tricentis

DAVE

TestLeft

behat

concordion

Ferramentas BDD em Destaque



Codeless com modelo visual, integração com Jira/Jenkins



Código-fonte aberto, suporte a várias linguagens



.NET, editor de Gherkin online



BDD para Java



Baseado em wiki, testes como especificações



Relatórios em HTML e execução como JUnit



Framework BDD para PHP



Java, integração com JMock2 e Hamcrest

Qual a melhor ferramenta???



Critérios de escolha de ferramentas

1. Compatibilidade
2. Integração com ferramentas de automação
3. Suporte à sintaxe Gherkin
4. Geração de documentação viva
5. Colaboração
6. Custo
7. Curva de aprendizado
8. Flexibilidade
9. Comunidade e suporte
10. Opinião do time



Vamos juntos!

1. Remover item do carrinho

Como cliente durante uma compra,
quero remover um item do meu carrinho,
para corrigir ou ajustar minha lista de compras antes do
pagamento.

1. Receber notificações de envio

Como cliente que fez uma compra,
quero receber notificações sobre o status de envio do
pedido,
para me manter informado até a entrega.



Vamos praticar!

Escrita colaborativa de cenários

- Transformar o **user stories** em **cenários claros e validados**.
 - a. Cada aluno (ou dupla) deve escolher **2 user stories** e **escrever cenários Gherkin baseados nelas**.
 - b. Um cenário de **sucesso** e um cenário alternativo/**erro**.
 - c. Verificar se os critérios da história estão representados.
 - d. Alunos (duplas) serão sorteados para lerem seus cenários.
 - e. Todos deverão enviar suas respostas via [Google forms](#) usando o formulário fornecido.



Vamos praticar!

1 Atualizar dados cadastrais

Como usuário logado,
quero atualizar meu endereço de entrega,
para garantir que minhas compras cheguem
corretamente.

2 Adicionar produto ao carrinho

Como cliente navegando na loja,
quero adicionar um produto ao carrinho,
para finalizar a compra depois.

3 Avaliar um produto comprado

Como cliente,
quero deixar uma avaliação em um produto que comprei,
para ajudar outros usuários na decisão de compra.

4 Filtrar pedidos por status

Como administrador da loja,
quero filtrar os pedidos por status (pago, pendente,
enviado),
para acompanhar melhor o andamento das vendas.

DÚVIDAS?!

:)

The background of the slide features several large, overlapping circles in shades of orange and yellow, creating a vibrant, abstract pattern on the right side.



C . E . S . A . R

MUITO OBRIGADO!

© CESAR | Todos os direitos reservados