**TTI 202 - Desenvolvimento Ágil - AULA 10**

**TESTES DE SOFTWARE ÁGIL**

**TDD (*Test-Driven Development*)**

**BDD (*Behavior-Driven Development*)**

**Profa. Ana Cláudia Tiessi**

**Profa. Ana Paula Serra**

**1. Recomendação**

Utilizar os artigos indicados para realizar os demais itens dessa tarefa.

**1.1 Introdução ao TDD (Conceitos, Ciclo, Vantagens):**

* **Artigos:**
  + **5 passos do TDD**
    - [**https://developer.ibm.com/articles/5-steps-of-test-driven-development/**](https://developer.ibm.com/articles/5-steps-of-test-driven-development/)

**1.2 Introdução ao BDD (Conceitos, Gherkin, Vantagens):**

* **Artigos:**
  + ***What is BDD?***
    - <https://cucumber.io/docs/bdd/>
  + ***Gherkin Reference***
    - <https://cucumber.io/docs/gherkin/reference/>

**2. Estudos de Caso Simplificados**

Leia os estudos de caso e respondam a questões.

**2.1 Estudo de caso : Login**

"Um sistema de login deve validar se o email e a senha estão corretos. Se estiverem, o acesso é permitido; caso contrário, uma mensagem de erro é exibida."

Agora vocês devem responder essas questões:

1. Como eles testariam essa funcionalidade manualmente?

a. Abrir a interface de login do sistema

b. Digitar diferentes combinações de e-mails e senhas para verificar se o sistema responde corretamente em cada caso:

i. E-mail e senha corretos.

ii. E-mail correto e senha incorreta.

iii. E-mail incorreto e senha correta.

iv. Ambos incorretos.

v. Campos em branco.

1. Quais seriam os possíveis casos de teste?

a. Acesso permitido: i. Login com e-mail e senha corretos

b. Erro: “Senha inválida”: i. Login com e-mail correto e senha incorreta

c. Erro: “Usuário não encontrado”: i. Login com e-mail incorreto e senha correta

d. Erro genérico: “Credenciais inválidas”: i. Login com ambos incorretos

e. Erro: “Preencha todos os campos”: i. Login com campos vazios

f. Erro: “E-mail inválido”: i. Login com e-mail em formato inválido

1. Como eles poderiam descrever o comportamento esperado dessa funcionalidade para alguém que não é técnico?

Ao tentar entrar no sistema, a pessoa deve digitar seu email e senha. Se os dois estiverem certos, ela entra normalmente. Se um deles estiver errado, o sistema avisa com uma mensagem de erro. Se o email estiver mal escrito (por exemplo, sem o '@'), o sistema também avisa. E se não preencher algum campo, o sistema lembra que é obrigatório.

1. Quais seriam os critérios de aceitação e como testar?

Cenário 1: Login bem-sucedido

Dado que o usuário está na tela de login

Quando ele insere um email e senha válidos

Então ele deve ser redirecionado para a página principal

Cenário 2: Login com senha incorreta

Dado que o usuário está na tela de login

Quando ele insere um email válido e senha incorreta

Então ele deve ver a mensagem "Email ou senha incorretos"

Cenário 3: Login com email em formato inválido

Dado que o usuário está na tela de login

Quando ele insere um email sem "@"

Então ele deve ver a mensagem "Formato de email inválido"

Cenário 4: Campos obrigatórios não preenchidos

Dado que o usuário está na tela de login

Quando ele tenta enviar o formulário sem preencher email e senha

Então ele deve ver mensagens dizendo que os campos são obrigatórios

**2.2 Estudo de Caso: Agendamento**

"Vocês fazem parte da equipe de desenvolvimento de um sistema de agendamento de consultas. O PO criou a seguinte história de usuário:

*Como paciente, quero agendar uma consulta com meu médico pelo aplicativo, para evitar filas e ligações.*

Com base nessa história, façam:

1. A definição de critérios de aceitação (como o PO).

O paciente deve poder escolher uma data e um horário disponíveis na agenda do médico.

O sistema deve impedir o agendamento de horários já ocupados.

O sistema deve exibir uma mensagem clara ao confirmar o agendamento.

O sistema deve registrar a consulta no histórico do paciente.

1. A escrita de testes no formato Gherkin (BDD), de acordo, com os critérios de aceitação do item 1.

Cenário 1: Agendamento de consulta com sucesso

Dado que o paciente está logado no aplicativo

E acessa a lista de médicos disponíveis

Quando ele seleciona um médico e um horário livre

E confirma o agendamento

Então o sistema deve registrar o agendamento

E exibir uma mensagem de confirmação

Cenário 2: Tentativa de agendar em horário já ocupado

Dado que o paciente está na tela de agendamento

E o horário selecionado já está ocupado

Quando ele tenta confirmar o agendamento

Então o sistema deve exibir a mensagem "Horário indisponível"

Cenário 3: Tentativa de agendamento com dados incompletos

Dado que o paciente não preencheu todos os campos obrigatórios

Quando ele tenta confirmar o agendamento

Então o sistema deve exibir a mensagem "Preencha todos os campos obrigatórios"

Cenário 4: Visualização de horários disponíveis

Dado que o paciente seleciona um médico

Quando ele acessa os horários

Então o sistema deve exibir apenas os horários livres daquele médico

1. O esboço de testes unitários no estilo TDD (pseudocódigo ou código na linguagem escolhida). Lembrem de fazer o programa de teste e programa do agendamento.

TESTE "Agendamento com sucesso"

criar sistema de agendamento

criar médico com horário livre às 10h

criar paciente

resultado = sistema.agendar(medico, paciente, "2025-05-16 10:00")

ASSERT resultado == "Agendamento confirmado"

TESTE "Agendamento em horário ocupado"

agendar consulta com médico às 10h

tentar novo agendamento mesmo médico e horário

ASSERT resultado == "Horário indisponível"

TESTE "Agendamento com dados incompletos"

tentar agendar consulta sem paciente ou horário

ASSERT resultado == "Preencha todos os campos obrigatórios"

**3 Questionário**

Respondam as seguintes questões

1. O que significa a fase 'Red' no ciclo TDD?
2. Qual o principal objetivo do BDD?
3. Gherkin é uma linguagem utilizada em qual abordagem?
4. "DD foca mais nos aspectos técnicos da unidade, enquanto BDD foca no comportamento do sistema sob o ponto de vista do usuário. (Verdadeiro/Falso)
5. Os critérios de aceitação influenciaram o design dos testes?
6. Quais práticas você consideraria adotar em projetos reais?

RESPOSTAS:  
Questão 3:

1 -) A fase 'Red', no ciclo TDD, significa que o código ainda não funciona

2 -) O principal objetivo do BDD é trazer uma solução para os times de desenvolvimento trabalharem próximos aos membros do business e outros membros técnicos, encorajando a comunicação & colaboração, implermentando iterações pequenas e rápidas, que fazem o uso de 'plugins' que aprimoram as entregas, e documentação dinâmica.

3 -) Gherkin é uma linguagem baseada em 'palavras-chave' cujo foco é definir o comportamento esperado de um software de uma perspectiva de negócios. Cada palavra-chave descreve uma etapa ou processo dentro do programa, servindo como um tipo de 'guia'

4 -) Verdadeiro. O BDD foca no comportamento do sistema sob o ponto de vista do usuário e das partes interessadas

5 -) Sim, os critérios de aceitação determinam quais resultados devemos atingir nos testes, portanto influenciam em seu design

6 -) O uso do framework SCRUM em conjunto com o quadro Kanban e processos do XP, como programação em pares, são práticas muito úteis que consideraria adotar em projetos reais.