**Prova QA**

1. Quais cenários de teste você especificaria para validar a funcionalidade do aplicativo bancário desenhado abaixo? Liste todas as premissas adotadas.

Texto

Descrição gerada automaticamente

Baseado na pirâmide de testes para a validação da tela do aplicativo apresentada eu especificaria os seguintes testes.

**Testes Unitários e Testes de Componentes**

Os testes unitários e de componentes comumente são de responsabilidade do desenvolvedor. Certificaria-me que o desenvolvedor criou os testes unitários e de componentes para a cobertura de todo o código relacionado a página.

**Testes de Integração**

Os testes de Integração são de responsabilidade do desenvolvedor e do Especialista de qualidade. Para os testes de integração especificaria os seguintes cenários:

- Validar a API que retorna o valor de Saldo apresentado na tela.

-Validar a API que envia os valores do campo “Cliente destino”, “valor” e “Data de efetivação”.

-Validar os valores limites e dados aceitos dos campos que são enviados pela API.

- Validar o acionamento da API no botão “Transferir”

- Validar a integração entre a API que envia os dados da transferência com a API que busca o saldo em conta para que não ultrapasse o valor disponível.

**Testes End to End**

Os testes End to End ou testes funcionais são de responsabilidade principalmente do Especialista de qualidade. Para os testes funcionais especificaria os seguintes cenários:

- Validar se os dados do campo Saldo são apresentados corretamente

-Validar o envio correto dos valores dos campos “Cliente destino”, “valor” e “data de efetivação” ao clicar no botão “Transferir”.

- Validar o bloqueio do envio ou mensagem de erro ao tentar realizar a transferência com algum campo em branco ou com valores fora dos valores permitidos para preenchimento de cada campo.

- Validar o cancelamento da transferência ao clicar no botão “Cancelar”

- Validar que os campos destinados para preenchimento de “Cliente destino”, “valor” e “data de efetivação” aceitam somente dados que estejam de acordo o formato especificado para cada campo.

- Validar o layout da Tela, títulos, textos, campos de input de dados e botões se estão de acordo com o especificado pela equipe de Experiência do Cliente.

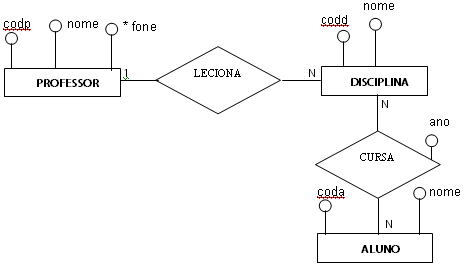
**Testes de acessibilidade**

Para os testes de acessibilidade especificaria os seguintes testes

- Validar a navegação da página em todos os campos por comandos de teclado

-Validar a navegação e usabilidade da página com a alteração do contraste da tela para usuários de baixa visão.

- Validaria a navegação da página com o uso do talkback

1. Considere o esquema Entidade / Relacionamento abaixo:

* Escreva a consulta SQL para listar o nome de todos os alunos matriculados na disciplina de Cálculo do professor João.

SELECT Aluno.nome

FROM Aluno

JOIN Professor ON Aluno.Disciplina\_que\_cursa = Professor.Disciplina\_que\_leciona

WHERE Professor.nome = 'João' AND Disciplina.nome = 'Cálculo';

* Escreva a consulta SQL para exibir a quantidade de alunos por disciplinas.

SELECT Disciplina.nome, COUNT(Aluno.Coda) AS quantidade\_alunos

FROM Disciplina

LEFT JOIN Aluno ON Disciplina.Codd = Aluno.Disciplina\_que\_cursa

GROUP BY Disciplina.nome;

* Escreva a consulta SQL para listar as disciplinas que todos os professores lecionam.

SELECT Disciplina.nome

FROM Disciplina

JOIN Professor ON Disciplina.Codd = Professor.Disciplina\_que\_leciona

GROUP BY Disciplina.nome

HAVING COUNT(DISTINCT Professor.Codp) = (SELECT COUNT(\*) FROM Professor);

* Escreva a consulta SQL que exibe o total de professores.

SELECT COUNT(\*) AS total\_professores

FROM Professor;

* Escreva a consulta SQL para listar todos os alunos que cursaram alguma disciplina do ano 2000 até 2020.

SELECT Aluno.nome

FROM Aluno

WHERE Aluno.Ano BETWEEN 2000 AND 2020;

1. **Para validarmos a sua técnica de automação de testes, vamos utilizar uma api de desenvolvimento referente a plataforma Trello.**

Caso não conheça o Trello, acessar: www.trello.com

Link da api para desenvolvimento: <https://developers.trello.com/docs/api-introduction>

Com o end-point acima, crie testes automatizados para as seguintes ações:

* **Cadastrar um board**
* **Cadastrar um card**
* **Excluir um card**
* **Excluir um board**

**Obs: Utilizar o Cypress como ferramenta de automação.**

**Para a automação de testes realizada, ilustre através de um mapa mental, a sua estratégia de testes para construção do seu projeto.**

**Após a conclusão do teste, publique no github, todo material produzido, e não esqueça de nos disponibilizar o link da sua avaliação.**

Em todas as interações com a API do Trello verificar que a Resposta seja com Status 200 e que não tenha um Body vazio

Envio um Delete para excluir o board criado. Utilizo o id do Board que recebi no response da criação do Board

Envio um Delete para excluir o card criado. Utilizo o id do Card que recebi no response da criação do card

API Trello

Envio um POST para a criação de uma nova Lista utilizando o id do Board criado. Com o response registro o id da Lista.

Envio um POST para a criação de um novo Card utilizando o id da Lista criada. Ao receber o response registro o id do Card.

Envio um POST para a criação de um novo Board. Com o response registro o id do Board.