

**ENGENHEIRO DE QUALIDADE DE SOFTWARE**

Giovanna Paolinelli Shinhe

Análise de Qualidade

Florianópolis

2023

# RESUMO

Aqui você deve fazer um breve resumo do seu projeto. Aborde um pouco de tudo, mas não entre profundamente em nada. O “resumo” em um trabalho acadêmico “serve” para mostrar ao leitor se o conteúdo é de seu interesse ou não. Mas é um resumo, um breve relato de, no máximo 200 palavras.

# SUMÁRIO

[1. RESUMO 2](#_Toc99483086)

[2. SUMÁRIO 3](#_Toc99483087)

[3. INTRODUÇÃO 4](#_Toc99483088)

[4. O PROJETO 5](#_Toc99483089)

[4.1 Estratégia de teste 6](#_Toc99483090)

[4.2 Critérios de aceitação 6](#_Toc99483091)

[4.3 Casos de testes 6](#_Toc99483092)

[4.4 Repositório no Github 7](#_Toc99483093)

[4.5 Testes automatizados 7](#_Toc99483094)

[4.6 Integração contínua 8](#_Toc99483095)

[4.7 Testes de performance 8](#_Toc99483096)

[5. CONCLUSÃO 9](#_Toc99483097)

[6. REFERÊNCIAS BIBLIOGRÁFICAS 9](#_Toc99483098)

# INTRODUÇÃO

Na introdução você deve fazer um apanhado geral do seu cenário para o leitor. Escreva a prévia do que teríamos no trabalho, o que irá tratar, o que espera do projeto, etc.

# O PROJETO

Para este trabalho de conclusão de curso **Profissão: Engenheiro de Qualidade de software**, você deve utilizar o conhecimento adquirido ao longo do curso para elaborar uma estratégia de testes adequada para validar o e-commerce EBAC Shop (<http://lojaebac.ebaconline.art.br/>). Você deve considerar as histórias de usuário já refinadas como se você estivesse participando de um time ágil. As funcionalidades devem seguir todo o fluxo de trabalho de um *Quality Engineer* (QE), desde o planejamento até a entrega. Siga as etapas dos sub-tópicos para se orientar no trabalho.

**ATENÇÃO**:

* Conforme a sua estratégia, você pode executar os testes no endereço disponibilizado ou utilizando as imagens disponíveis no Docker Hub:
  + Banco de Dados: [ernestosbarbosa/lojaebacdb](https://hub.docker.com/repository/docker/ernestosbarbosa/lojaebacdb)
  + Loja EBAC: [ernestosbarbosa/lojaebac](https://hub.docker.com/repository/docker/ernestosbarbosa/lojaebac)
    - Comandos para subir os containers:

*docker network create --attachable ebac-network*

*docker run -d --name wp\_db -p 3306:3306 --network ebac-network ernestosbarbosa/lojaebacdb:latest*

*docker run -d --name wp -p 80:80 --network ebac-network ernestosbarbosa/lojaebac:latest*

*Após subir os containers a loja estará em* [*http://localhost:80*](http://localhost:80)

## Estratégia de teste



## Critérios de aceitação

|  |  |  |  |
| --- | --- | --- | --- |
| **[US-0001] – Adicionar item ao carrinho** | | | |
| **Situação** | Em Andamento | | |
| **Projeto** | EBAC-SHOP | | |
| **Versão Afetada** | 1.0 | | |
| **Tipo:** | História | **Prioridade:** | Média |
| **Solicitante:** | Fábio Araújo | **Responsável:** |  |
| **Resolução:** |  | **Pontuação:** | 13 |
| **Tempo Estimado:** |  | | |
| **Anexos** |  | | |
| **Descrição** | | | |
| **Como** cliente da EBAC-SHOP  **Quero** adicionar produtos no carrinho  **Para** realizar a compra dos itens  **Regras de negócio:**   * Não é permitido inserir mais de 10 itens de um mesmo produto ao carrinho; * Os valores não podem ultrapassar a R$ 990,00; * Valores entre R$ 200 e R$ 600 , ganham cupom de 10% * Valores acima de R$ 600 ganham cupom de 15% | | | |
| **Critérios de Aceitação** | | | |
| **Contexto:**  **Dado** que eu acesse a página do <produto>  **Esquema de Cenário:** Adicionar item ao carrinho com sucesso  **Quando** eu adicionar <quantidade> <produto> ao carrinho  **Então** deve aparecer mensagem de sucesso  **E** o <produto> deve ser adicionado ao carrinho  **Exemplos**   |  |  | | --- | --- | | **quantidade** | **produto** | | **1** | [Moleton Rosa Feminino](http://lojaebac.ebaconline.art.br/product/moleton-rosa-feminino/) | | **10** | [Moleton Rosa Feminino](http://lojaebac.ebaconline.art.br/product/moleton-rosa-feminino/) |   **Esquema de Cenário:** Adicionar mais de 10 itens do mesmo produto ao carrinho  **Quando** eu adicionar <quantidade> <produto> ao carrinho  **Então** deve aparecer mensagem de erro  **E** o <produto> não deve ser adicionado ao carrinho  **Exemplos**   |  |  | | --- | --- | | **quantidade** | **produto** | | **11** | [Moleton Rosa Feminino](http://lojaebac.ebaconline.art.br/product/moleton-rosa-feminino/) |   **Esquema de Cenário:** Adicionar diferentes itens ao carrinho  **Dado** que eu tenha um <produto1> no carrinho  **Quando** eu adicionar <quantidade> <produto2> ao carrinho  **Então** deve aparecer mensagem de sucesso  **E** o <produto1> e <produto2> devem estar no carrinho  **Exemplos**   |  |  |  | | --- | --- | --- | | **quantidade** | **produto1** | **produto2** | | **1** | Josie Yoga Jacket | [Moleton Rosa Feminino](http://lojaebac.ebaconline.art.br/product/moleton-rosa-feminino/) |   **Esquema de Cenário:** Adicionar mais de 10 itens do mesmo produto ao carrinho  **Dado** que eu tenha 10 itens do <produto> no carrinho  **Quando** eu adicionar <quantidade> <produto> ao carrinho  **Então** deve aparecer mensagem de erro  **E** o <produto> não deve ser adicionado ao carrinho  **Exemplos**   |  |  | | --- | --- | | **quantidade** | **produto** | | **1** | [Moleton Rosa Feminino](http://lojaebac.ebaconline.art.br/product/moleton-rosa-feminino/) |   **Esquema de Cenário:** Teste de valores do carrinho  **Dado** que eu  **Quando** eu adicionar <valor> ao carrinho  **Então** o valor deve ser <limite>  **E** o <cupom> deve ser aplicado ao carrinho   |  |  |  | | --- | --- | --- | | **valor** | **cupom** | **limite** | | **199,99** | **0** | **Válido** | | **200** | **10** | **Válido** | | **600** | **10** | **Válido** | | **600,01** | **15** | **Válido** | | **990** | **15** | **Válido** | | **990,01** | **0** | **Inválido** | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **[US-0002] – Login na Plataforma** | | | |
| **Situação** | Em Andamento | | |
| **Projeto** | EBAC-SHOP | | |
| **Versão Afetada** | 1.0 | | |
| **Tipo:** | História | **Prioridade:** | Média |
| **Solicitante:** | Fábio Araújo | **Responsável:** |  |
| **Resolução:** |  | **Pontuação:** | 8 |
| **Tempo Estimado:** |  | | |
| **Anexos** |  | | |
| **Descrição** | | | |
| **Como** cliente da EBAC-SHOP  **Quero** fazer o login (autenticação) na plataforma  **Para** visualizar meus pedidos  **Regras de negócio:**   * Somente usuários ativos podem fazer login; * Deve exibir uma mensagem de erro caso o usuário erre o login e senha; * Se o usuário errar por 3 vezes a senha, deve travar por 15 minutos o login | | | |
| **Critérios de Aceitação** | | | |
| **Contexto:**  **Dado** que eu acesse a página login  **Cenário:** Login com sucesso  **Quando** eu inserir <email> e <senha>  **Então** deve aparecer mensagem de sucesso  **E** o usuário deve ser logado na plataforma  **Exemplos**   |  |  | | --- | --- | | **email** | **senha** | | [**usuarioativo@teste.com**](mailto:usuarioativo@teste.com) | teste@123 |   **Esquema de Cenário:** Login com senha errada  **Quando** eu inserir <email> e <senha>  **Então** deve aparecer mensagem de erro  **E** o usuário deve não ser logado na plataforma  **Exemplos**   |  |  | | --- | --- | | **email** | **senha** | | [**usuarioativo@teste.com**](mailto:usuarioativo@teste.com) | senhaerrada |   **Esquema de Cenário:** Login com usuário inativo  **Quando** eu inserir <email> e <senha>  **Então** o usuário deve não ser logado na plataforma  **Exemplos**   |  |  | | --- | --- | | **email** | **senha** | | [**usuarioinativo@teste.com**](mailto:usuarioinativo@teste.com) | teste@123 |   **Esquema de Cenário:** Travar o login após erro  **Quando** eu inserir <email> e <senha> 3 vezes  **Então** deve aparecer mensagem de erro  **E** o usuário deve não ser logado na plataforma  **E** o login deve ficar bloqueado por 15 min  **Exemplos**   |  |  | | --- | --- | | **email** | **senha** | | [**usuarioativo@teste.com**](mailto:usuarioativo@teste.com) | senhaerrada | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **[US-0003] – API de Cupons** | | | |
| **Situação** | Em Andamento | | |
| **Projeto** | EBAC-SHOP | | |
| **Versão Afetada** | 1.0 | | |
| **Tipo:** | História | **Prioridade:** | Média |
| **Solicitante:** | Fábio Araújo | **Responsável:** |  |
| **Resolução:** |  | **Pontuação:** | 13 |
| **Tempo Estimado:** |  | | |
| **Anexos** |  | | |
| **Descrição** | | | |
| **Como** admin da EBAC-SHOP  **Quero** criar um serviço de cupom  **Para** poder listar e cadastrar os cupons  Autenticação da API:   * Usuário: admin\_ebac * Senha: @admin!&b@c!2022 * Authorization: Basic YWRtaW5fZWJhYzpAYWRtaW4hJmJAYyEyMDIy   **Regras de negócio:**  GET:   * Deve listar todos os cupons cadastrado ou listar buscando por ID do cupom * Documentação do serviço:   http://lojaebac.ebaconline.art.br/rest-api/docs/#/coupons/get\_wc\_v3\_coupons  http://lojaebac.ebaconline.art.br/restapi/docs/#/coupons/get\_wc\_v3\_coupons\_\_id\_  POST:   * Deve cadastrar os cupons com os campos obrigatórios abaixo:   + Código do cupom: Exemplo: “Ganhe10”   + Valor: “10.00”   + Tipo do desconto: “fixed\_product”   + Descrição: “Cupom de teste” * Nome do cupom não pode ser repetido; * Os outros campos são opcionais. * Documentação do serviço:   http://lojaebac.ebaconline.art.br/rest-api/docs/#/coupons/post\_wc\_v3\_coupons   * Exemplo do body:   body: {  "code": "nomeCupom",  "amount": "10",  "discount\_type": "fixed\_product",  "description": "Cupom de desconto de teste"  } | | | |
| **Critérios de Aceitação** | | | |
| **Contexto:**  **Dado** que o usuário não tenha sido autenticado  **Cenário:** Get de cupons sem autenticação  **Quando** eu pedir para listar todos os cupons  **Então** a API deve retornar 401 Unauthorized  **Cenário:** Get de cupons sem autenticação  **Quando** eu pedir para listar cupom por ID  **Então** a API deve retornar 401 Unauthorized  **Cenário:** Post de cupons sem autenticação  **Quando** eu cadastrar cupom com body  body: {  "code": "ganhe10",  "amount": "10",  "discount\_type": "fixed\_product",  "description": "Cupom de desconto de teste"  }  **Então** a API deve retornar 400 Unauthorized  **Contexto:**  **Dado** que eu tenha autenticado a sessão como administrador  **Cenário:** Get de cupons  **Quando** eu pedir para listar todos os cupons  **Então** todos os cupons devem ser listados  **Esquema de Cenário:** Get de cupons por ID  **Quando** eu pedir para listar o cupom pela ID <ID>  **Então** o cupom de ID <ID> deve ser listado   |  | | --- | | **ID** | |  |   **Esquema de Cenário:** Post de cupons  **Quando** eu cadastrar um cupom com <body>  **Então** o cadastro deve ser <permissão>   |  |  | | --- | --- | | **codigo\_cupom** | **permissão** | | body: {  "code": "ganhe10",  "amount": "10",  "discount\_type": "fixed\_product",  "description": "Cupom de desconto de teste"  } | Permitido | | body: {  "code": "ganhe20",  "amount": "20",  "discount\_type": " fixed\_cart",  "description": "Cupom de desconto de teste"  } | Permitido | | body: {  "code": "ganhe30",  "amount": "30",  "discount\_type": " percent",  "description": "Cupom de desconto de teste"  } | Permitido | | body: {  "code": "ganhe10",  "amount": "10",  "discount\_type": "fixed\_product",  "description": "Cupom de desconto de teste"  } | Não Permitido | | body: {  "amount": "10",  "discount\_type": "fixed\_product",  "description": "Cupom de desconto de teste"  } | Não Permitido | | body: {  "code": "ganhe1",  "discount\_type": "fixed\_product",  "description": "Cupom de desconto de teste"  } | Não Permitido | | body: {  "code": "ganhe3",  "amount": "10",  "description": "Cupom de desconto de teste"  } | Não Permitido | | body: {  "code": "ganhe10",  "amount": "10",  "discount\_type": "fixed\_product"  } | Não Permitido | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **[US-0004] – Catálogo de Produtos** | | | |
| **Situação** | Em Andamento | | |
| **Projeto** | EBAC-SHOP | | |
| **Versão Afetada** | 1.0 | | |
| **Tipo:** | História | **Prioridade:** | Média |
| **Solicitante:** | Fábio Araújo | **Responsável:** |  |
| **Resolução:** |  | **Pontuação:** | 8 |
| **Tempo Estimado:** |  | | |
| **Anexos** |  | | |
| **Descrição** | | | |
| **Como** cliente da EBAC-SHOP  **Quero** uma acessar uma página de catálogo de produtos  **Para** escolher o produtos para comprar  **Regras de negócio:**   * Os produtos devem aparecer com o nome, imagem, nota e valor de cada produto. * Deve ordenar por padrão, popularidade, média de classificação, mais recente, preço: menor para maior, preço: maior para menor. * Deve trocar a visualização entre grid e lista * Ao passar o mouse em cima do produto, devem aparecer opções: ver opções, visualizar, favoritar, comparar produtos. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **[US-0005] – Painel Minha Conta** | | | |
| **Situação** | Em Andamento | | |
| **Projeto** | EBAC-SHOP | | |
| **Versão Afetada** | 1.0 | | |
| **Tipo:** | História | **Prioridade:** | Média |
| **Solicitante:** | Fábio Araújo | **Responsável:** |  |
| **Resolução:** |  | **Pontuação:** | 8 |
| **Tempo Estimado:** |  | | |
| **Anexos** |  | | |
| **Descrição** | | | |
| **Como** cliente da EBAC-SHOP  **Quero** acessar uma página individual  **Para** visualizar minhas informações pessoais e pedidos  **Regras de negócio:**   * A página deve ser acessada somente após autenticação com login e senha; * A tela deve apresentar os pedidos, endereços, downloads e detalhes da conta; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **[US-0006] – Meus Pedidos** | | | |
| **Situação** | Em Andamento | | |
| **Projeto** | EBAC-SHOP | | |
| **Versão Afetada** | 1.0 | | |
| **Tipo:** | História | **Prioridade:** | Média |
| **Solicitante:** | Fábio Araújo | **Responsável:** |  |
| **Resolução:** |  | **Pontuação:** | 8 |
| **Tempo Estimado:** |  | | |
| **Anexos** |  | | |
| **Descrição** | | | |
| **Como** cliente da EBAC-SHOP  **Quero** visualizar meus pedidos  **Para** conferir minhas compras  **Regras de negócio:**   * Todos os pedidos devem aparecer na página * Os pedidos devem apresentar status: pending, processing, on-hold, completed, cancelled, refunded, failed. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **[US-0007] – Endereços** | | | |
| **Situação** | Em Andamento | | |
| **Projeto** | EBAC-SHOP | | |
| **Versão Afetada** | 1.0 | | |
| **Tipo:** | História | **Prioridade:** | Média |
| **Solicitante:** | Fábio Araújo | **Responsável:** |  |
| **Resolução:** |  | **Pontuação:** | 8 |
| **Tempo Estimado:** |  | | |
| **Anexos** |  | | |
| **Descrição** | | | |
| **Como** cliente da EBAC-SHOP  **Quero** editar meus endereços cadastrados  **Para** atualizar meus dados  **Regras de negócio:**   * Endereço de entrega não é obrigatório * Campo Nome, Sobrenome, País, Endereço, Cidade, Estado, CEP, Telefone, e-mail são obrigatórios. * Uma vez inserido endereço de entrega, não é possível excluir, somente alterar. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **[US-0008] – Detalhes da Conta** | | | |
| **Situação** | Em Andamento | | |
| **Projeto** | EBAC-SHOP | | |
| **Versão Afetada** | 1.0 | | |
| **Tipo:** | História | **Prioridade:** | Média |
| **Solicitante:** | Fábio Araújo | **Responsável:** |  |
| **Resolução:** |  | **Pontuação:** | 8 |
| **Tempo Estimado:** |  | | |
| **Anexos** |  | | |
| **Descrição** | | | |
| **Como** cliente da EBAC-SHOP  **Quero** visualizar minhas informações cadastradas  **Para** poder editá-las  **Regras de negócio:**   * Atualização de senha só deve ser permitida após inserir a senha atual * A senha deve ter pelo menos doze caracteres, incluindo letras, números e símbolos * A senha e confirmação de senha devem ser iguais * Campos Nome, Sobrenome, Display name e email são obrigatórios * Campo de nova senha deve indicar senha “muito fraca”, “fraca”, “média” e “forte” | | | |

## Casos de testes

|  |
| --- |
| **[US-0001] – Adicionar item ao carrinho** |
| |  |  |  |  | | --- | --- | --- | --- | | **#** | **Teste** | **Resultado esperado** | **Tipo de teste** | | 1 | Adicionar 1 produto ao carrinho | Deve adicionar o produto ao carrinho | Automatizado | | 2 | Adicionar 10 produtos ao carrinho | Deve adicionar 10 produtos ao carrinho | Automatizado | | 3 | Adicionar 11 produtos ao carrinho | Não deve permitir adicionar produto | Automatizado | | 4 | Adicionar 10 produtos iguais e em seguida mais 1 produto igual ao carrinho | Não deve permitir adicionar produto | Automatizado | | 5 | Adicionar itens diferentes ao carrinho | Deve adicionar ambos produtos ao carrinho | Automatizado | | 6 | Adicionar 199,99 reais ao carrinho | Não deve ter cupom de desconto | Automatizado | | 7 | Adicionar 200 reais ao carrinho | Deve inserir cupom de 10% de desconto | Automatizado | | 8 | Adicionar 600 reais ao carrinho | Deve inserir cupom de 10% de desconto | Automatizado | | 9 | Adicionar 600,01 reais ao carrinho | Deve inserir cupom de 15% de desconto | Automatizado | | 10 | Adicionar 990 reais ao carrinho | Deve inserir cupom de 15% de desconto | Automatizado | | 11 | Adicionar 990,01 reais ao carrinho | Não deve permitir finalizar compra | Automatizado | |
| **[US-0002] – Login na Plataforma** |
| |  |  |  |  | | --- | --- | --- | --- | | **#** | **Teste** | **Resultado esperado** | **Tipo de teste** | | 1 | Fazer login com sucesso | Deve logar usuário | Automatizado | | 2 | Fazer login com senha errada | Usuário não deve ser logado  Mensagem de senha errada deve aparecer | Automatizado | | 3 | Fazer login com email não cadastrado | Usuário não deve ser logado | Automatizado | | 4 | Fazer login com email inativo | Usuário não deve ser logado | Automatizado | | 5 | Fazer login com senha errada 3 vezes | Usuário não deve ser logado  Usuário deve ficar bloqueado por 15 min | Automatizado | |
| **[US-0003] – API de Cupons** |
| |  |  |  |  | | --- | --- | --- | --- | | **#** | **Teste** | **Resultado esperado** | **Tipo de teste** | | 1 | Listar cupons sem autenticação | A API deve retornar 401 Unauthorized | Automatizado | | 2 | Listar cupons por ID sem autenticação | A API deve retornar 401 Unauthorized | Automatizado | | 3 | Cadastrar cupons sem autenticação | A API deve retornar 401 Unauthorized | Automatizado | | 4 | Listar cupons com autenticação | API deve retornar 200 e listar cupons | Automatizado | | 5 | Listar cupons por ID com autenticação | API deve retornar 200 e listar cupom por ID | Automatizado | | 6 | Cadastrar cupons com autenticação tipo “fixed\_product” com  body:  {  "code": "ganhe10",  "amount": "10",  "discount\_type": "fixed\_product",  "description": "Cupom de desconto de teste"  } | API deve retornar 201 e cadastrar cupom | Automatizado | | 7 | Cadastrar cupons com autenticação tipo “fixed\_cart” com  body:  {  "code": "ganhe1230947",  "amount": "20",  "discount\_type": " fixed\_cart",  "description": "Cupom de desconto de teste"  } | API deve retornar 201 e cadastrar cupom | Automatizado | | 8 | Cadastrar cupons tipo “percent” com autenticação com  body:  {  "code": "ganhe401294309",  "amount": "30",  "discount\_type": " percent",  "description": "Cupom de desconto de teste"  } | API deve retornar 201 e cadastrar cupom | Automatizado | | 9 | Cadastrar cupons com código já existente com autenticação com  body:  {  "code": "ganhe10",  "amount": "10",  "discount\_type": "fixed\_product",  "description": "Cupom de desconto de teste"  } | API deve retornar 400  Error: Bad Request com a mensagem:  {  "code": "woocommerce\_rest\_coupon\_code\_ already\_exists",  "message": "O código de cupom já existe",  "data": {  "status": 400  }  } | Automatizado | | 10 | Cadastrar cupons com autenticação sem o código do cupom com  body:  {  "amount": "10",  "discount\_type": "fixed\_product",  "description": "Cupom de desconto de teste"  } | API deve retornar 400  Error: Bad Request  {  "code": "rest\_missing\_callback\_param",  "message": "Parâmetro(s) ausente(s): code",  "data": {  "status": 400,  "params": [  "code"  ]  }  } | Automatizado | | 11 | Cadastrar cupons com autenticação sem o valor com  body:  {  "code": "ganhe1",  "discount\_type": "fixed\_product",  "description": "Cupom de desconto de teste"  } | API deve retornar 400  Error: Bad Request  {  "code": "rest\_missing\_callback\_param",  "message": "Parâmetro(s) ausente(s): amount",  "data": {  "status": 400,  "params": [  "amount"  ]  }  } | Automatizado | | 12 | Cadastrar cupons com autenticação sem o tipo de desconto com  body:  {  "code": "ganhe23124549",  "amount": "10",  "description": "Cupom de desconto de teste"  } | API deve retornar 400  Error: Bad Request  {      "code": "rest\_missing\_callback\_param",      "message": "Parâmetro(s) ausente(s): discount\_type ",      "data": {          "status": 400,          "params": [              " discount\_type "          ]      }  } | Automatizado | | 13 | Cadastrar cupons com autenticação sem campo de descrição  body:  {  "code": "ganhe13943810",  "amount": "10",  "discount\_type": "fixed\_product"  } | API deve retornar 400  Error: Bad Request  {      "code": "rest\_missing\_callback\_param",      "message": "Parâmetro(s) ausente(s): description",      "data": {          "status": 400,          "params": [              "code"          ]      }  } | Automatizado | |
| **[US-0004] – Catálogo de Produtos** |
| |  |  |  |  | | --- | --- | --- | --- | | **#** | **Teste** | **Resultado esperado** | **Tipo de teste** | | 1 | Alterar visualização para lista | Os produtos devem aparecer em lista ao invés de grid | Manual | | 2 | Alterar visualização para grid | Os produtos devem aparecer em grid ao invés de lista | Automatizado | | 3 | Ordenar por padrão | Deve ordenar por padrão | Manual | | 4 | Ordenar por popularidade | Deve ordenar por popularidade | Manual | | 5 | Ordenar por média de classificação | Deve ordenar por média de classificação | Manual | | 6 | Ordenar por mais recente | Deve ordenar por mais recente | Manual | | 7 | Ordenar por preço: menor para maior | Deve ordenar por preço: menor para maior | Manual | | 8 | Ordenar por preço: maior para menor | Deve ordenar por preço: maior para menor | Manual | | 9 | Passar o mouse em cima do produto | Deve aparecer ver opções, visualizar, favoritar, comparar produtos. | Manual | | 10 | Clicar no produto | Deve ir para tela de produto | Automatizado | | 11 | Favoritar produto | Deve adicionar o produto à lista de desejos.  Deve mostrar mensagem na tela de que o produto foi adicionado à lista de desejos | Automatizado | | 12 | Clicar em “ver opções” | Deve ir para tela do produto | Automatizado | | 13 | Clicar em comparar produto | Deve mostrar tela de comparação de produto | Automatizado | | 14 | Clicar em visualizar | Deve mostrar na mesma página, o produto e informações do produto | Automatizado | |
| **[US-0005] – Painel Minha Conta** |
| |  |  |  |  | | --- | --- | --- | --- | | **#** | **Teste** | **Resultado esperado** | **Tipo de teste** | | 1 | Abrir Painel Minha Conta | Deve aparecer a mensagem “Olá, <usuário> (não é <usuário>? Sair)  A partir do painel de controle de sua conta, você pode ver suas compras recentes, gerenciar seus endereços de entrega e faturamento, e editar sua senha e detalhes da conta.” | Automatizado | | 2 | Verificar menu de opções | Deve aparecer opção: Pedidos, Endereços, Downloads, Detalhes de Conta, Logout | Automatizado | | 3 | Fazer logout | Deve deslogar da conta | Automatizado | | 4 | Acessar pedidos | Deve direcionar para tela de pedidos | Automatizado | | 5 | Acessar endereços | Deve direcionar para tela de endereços | Automatizado | | 6 | Acessar downloads | Deve direcionar para tela de download | Automatizado | | 7 | Acessar detalhes de conta | Deve direcionar para tela de detalhes de conta | Automatizado | | 8 | Sair pela mensagem do painel | Deve realizar logout | Automatizado | |
| **[US-0006] – Meus Pedidos** |
| |  |  |  |  | | --- | --- | --- | --- | | **#** | **Teste** | **Resultado esperado** | **Tipo de teste** | | 1 | Visualizar pedido na tela pedidos | Deve abrir uma tela com os detalhes do pedido corretamente | Automatizado | | 2 | Cliente sem pedido | Deve aparecer mensagem “Nenhum pedido feito ainda” e um link para ver produtos | Automatizado | | 3 | Cliente sem pedido, clicar em ver produtos | Deve ser redirecionado para página de produtos | Automatizado | | 4 | Finalizar uma compra e verificar o pedido | O pedido deve aparecer na tela de pedidos corretamente | Automatizado | | 5 | Alterar o status de um pedido para aguardando | Deve alterar o status do pedido na tela | Automatizado | | 6 | Alterar o status de um pedido para concluído | Deve alterar o status do pedido na tela | Automatizado | | 7 | Alterar o status de um pedido para reembolsado | Deve alterar o status do pedido na tela | Automatizado | | 8 | Alterar o status de um pedido para mal sucedido | Deve alterar o status do pedido na tela.  Deve mostrar opções Pagar, Visualizar e Cancelar pedido | Automatizado | | 9 | Pagar pedido mal sucedido | Deve ser redirecionado para tela de pedido recebido  Pedido deve estar com status aguardando |  | | 10 | Cancelar pedido mal sucedido | Deve ser redirecionado para painel inicial com mensagem “seu pedido foi cancelado”  Pedido deve continuar na página de pedidos, com status cancelado |  | | 11 | Alterar o status de um pedido para pagamento pendente | Deve alterar o status do pedido na tela.  Deve mostrar opções Pagar, Visualizar e Cancelar pedido | Automatizado | | 12 | Pagar pedido pendente | Deve ser redirecionado para tela de pedido recebido  Pedido deve estar com status aguardando | Automatizado | | 13 | Cancelar pedido pendente | Deve ser redirecionado para painel inicial com mensagem “seu pedido foi cancelado”  Pedido deve continuar na página de pedidos, com status cancelado | Automatizado | |
| **[US-0007] – Endereços** |
| |  |  |  |  | | --- | --- | --- | --- | | **#** | **Teste** | **Resultado esperado** | **Tipo de teste** | | 1 | Editar endereço de faturamento | Deve abrir nova tela com o endereço. | Automatizado | | 2 | Editar endereço de entrega | Deve abrir nova tela com o endereço. | Automatizado | | 3 | Editar todos os campos do endereço de faturamento e salvar | Deve alterar endereço de faturamento | Automatizado | | 4 | Deixar algum dos campos obrigatórios vazios e salvar | Não deve alterar endereço  Deve mostrar mensagem <campo> é um campo obrigatório. | Automatizado | | 5 | Editar todos os campos do endereço de entrega e salvar | Deve alterar endereço de entrega | Automatizado | | 6 | Apagar todos os campos do endereço de entrega | Não deve alterar endereço  Deve mostrar mensagem <campo> é um campo obrigatório para todos os campos | Automatizado | |
| **[US-0008] – Detalhes da Conta** |
| |  |  |  |  | | --- | --- | --- | --- | | **#** | **Teste** | **Resultado esperado** | **Tipo de teste** | | 1 | Editar nome, sobrenome, display name e email e salvar | Deve salvar dados | Automatizado | | 2 | Salvar informações com algum dos campos obrigatórios nome, sobrenome, display name e email e salvar vazio | Não deve salvar  Deve mostrar mensagem <campo> é obrigatório | Automatizado | | 3 | Inserir senha atual  Inserir senha nova com doze caracteres, letras, números e símbolos e confirmação correta | Deve alterar senha | Automatizado | | 4 | Não inserir senha atual  Inserir senha nova com doze caracteres, letras, números e símbolos e confirmação correta | Não deve alterar senha  Deve mostrar mensagem “Digite sua senha atual.” | Automatizado | | 5 | Inserir senha atual  Inserir senha nova com doze caracteres, letras, números e símbolos e confirmação incorreta | Não deve alterar senha  Deve mostrar mensagem  “As novas senhas não são iguais.” | Automatizado | | 6 | Inserir uma senha “muito fraca” | Botão de salvar deve ficar desabilitado  “Muito fraca - Digite uma senha segura.” | Automatizado | | 7 | Inserir uma senha “fraca” | Botão de salvar deve ficar desabilitado  Deve aparecer a mensagem: “Fraca - Digite uma senha segura.” | Automatizado | | 8 | Inserir uma senha “média” | Botão de salvar deve ficar habilitado  Deve aparecer a mensagem: “Médio” | Automatizado | | 9 | Inserir uma senha “forte” | Botão de salvar deve ficar habilitado  Deve aparecer a mensagem: “Forte” | Automatizado | |

## Repositório no Github

Link do repositório: https://github.com/giovannapaolinelli/Projeto\_EBAC

## Testes automatizados

* + 1. Automação de UI
* Crie um projeto de automação WEB com o framework e a linguagem que preferir
* Justifique a sua escolha através de um comparativo entre ao menos 3 opções de ferramentas e linguagem.
* Crie uma pasta chamada UI para os testes WEB dos casos de teste que forem automatizados
* Utilize ao menos um *Testing Pattern* (à sua escolha) na implementação dos testes.
  + 1. Automação de API
* Crie uma pasta chamada API para os testes de API dos casos de teste que forem automatizados
* Você deve utilizar a ferramenta Supertest para criar seus testes de API
* Não esqueça de validar os contratos! ☺
  + 1. Automação Mobile
* Considere para os APPs apenas a funcionalidade de Catálogo de Produtos
* Você pode encontrar os APPs em:
  + *Android*: <https://github.com/EBAC-QE/testes-mobile-ebac-shop/tree/main/app/android>
  + *iOS*: <https://github.com/EBAC-QE/testes-mobile-ebac-shop/tree/ios-tests/app/ios>
* Crie uma pasta chamada Mobile para os testes em aplicativos dos casos de teste que forem automatizados
* Utilize ao menos um *Testing Pattern* (à sua escolha) na implementação dos testes.
* Você deve implementar testes para ao menos uma das plataformas Mobile (*Android* ou *iOS*)
* Observações:
  + Considere todas as boas práticas aprendidas até aqui
  + Não esqueça de implementar a geração de relatórios
* Referência: Módulos 11, 12, 14, 16, 17, 22, 23, 24, 29 e 30

## Integração contínua

* Execute os testes automatizados em integração contínua utilizando o Github Actions
* Referência: Módulo 26

## Testes de performance

* Usando o K6, implemente um teste de performance em ao menos 2 casos de testes
* Referência: Módulo 28
* Configurações do teste de performance:

-Usuários virtuais: 20  
-Tempo de execução: 2 minutos  
-RampUp: 20 segundos  
-Massa de dados: Usuário / senha:

user1\_ebac / psw!ebac@test  
user2\_ebac / psw!ebac@test  
user3\_ebac / psw!ebac@test  
user4\_ebac / psw!ebac@test  
user5\_ebac / psw!ebac@test

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

# CONCLUSÃO

Coloque sua experiência na realização do trabalho, o que aprendeu, quais lições pode aplicar em sua vida profissional etc.

# REFERÊNCIAS BIBLIOGRÁFICAS

Seguir regras ABNT