

Respostas do processo da vaga de Analista de QA na Blu.

Candidato: Rodrigo Santos de Moraes

Email: rsm.santos@live.com

Índice:

1 – Como Rodar os Testes Automatizados no Eclipse	Pag. 2
2- Escopo de Testes	Pag. 2
3– Suíte de Teste: Documentação	Pag. 3
4 – Suíte de Teste: API	Pag. 3
5- Relatório de Bugs encontrados	Pag. 4

1 – Como Rodar os Testes Automatizados no Eclipse:

1.1 - Definição do Pacote de execução do teste:

Main: contém os 3 arquivos JUnit que tem por finalidade executar as rotinas de testes.

- **MainTest_Compra:**

Este é o arquivo de rotinas principal do projeto, nele esta todas as rotinas solicitadas na avaliação, contendo o “Cadastro de um Novo Cliente”, “Login na Plataforma”, “Pesquisa de Produtos”, “Seleção de Produtos”, “Envio de Produtos para o Carrinho”, “Seleção de forma de Pagamento” e “verificação de Status da Compra”.

Tudo que foi pedido na avaliação esta automatizado nestas rotinas.

- **MainTest_Cadastro e MainTest_Login:**

Conforme sugerido no enunciado da avaliação, seria bem-vindo se novos casos de testes fossem inseridos além dos que já estavam sendo solicitados. Estes dois arquivos foram inseridos como sugestão de testes específicos para a tela de Cadastro e Login, fazendo que estes dois serviços pudessem ter uma “bateria” de testes automatizados somente para testar as suas funções, dando uma maior cobertura e segurança a plataforma. Foram implementados dois casos para cada arquivo; um testando uma rotina com sucesso e outro simulando que o teste automatizado encontrou um possível erro.

OBS: os outros pacotes do projeto, possuem nas suas classes, todas as orientações e definições sobre a sua implementação e função no projeto.

1.2 - Passos para “rodar” o Testes Automatizado.

- A. Ao abrir o projeto no eclipse, acesse o arquivo MainTest_Compra: **SRC-> Main-> MainTest_Compra.**
- B. Clique com o botão direito sobre o arquivo e selecione a opção **RUN AS -> JUnit Test.**

OBS: O teste já está configurado para rodar todas as etapas e também já está programado para gerar um email aleatório para um novo cadastro. Ao final, será possível ver o retorno do teste nos logs do eclipse. Os testes bônus do MainTest_Cadastro e MainTest_Login seguem o mesmo procedimento de execução informados acima.

O Diretório padrão do projeto é: C:\ProcessoPagNet_BluPagamentos\ProjetoAvaliacaoBluPagamentos, assim, se for descompactado direto no C:/ verifique se o diretório acima está correspondendo.

2- Escopo de Testes:

→ Fazer login na plataforma, selecionar um produto, realizar uma compra e verificar o status.

2.1 Suíte de Testes definidos (foram Automatizados no Eclipse):

- 1 – Cadastrar Cliente.
- 2 – Fazer Login na Plataforma.
- 3 – Pesquisar Produtos.
- 4 – Enviar Produtos para o Carrinho.
- 5 – Selecionar Forma de Pagamento.
- 6 - Verificar status da compra.

2.2 Suíte de Testes Adicionais (foram Automatizados no Eclipse):

Como sugerido, foram adicionados JUnit com casos de testes para crescimento e trabalhos futuros.

- **A– Cadastrar Cliente** (JUnit específica para realizar diversos testes com foco no cadastro).
 - A.1- Tentar Cadastrar Cliente sem informar campos obrigatórios.
 - A.2- Tentar Cadastrar Cliente já existente na base.
- **B– Login** (JUnit específica para realizar diversos testes com foco no Login na Plataforma).
 - B.1 - Tentar fazer Login com email inexistente.
 - B.2 - Tentar fazer Login com email errado (sem o @xx.com)

3– Suíte de Teste: Documentação

Aqui vou descrever um caso de teste que faz parte da Suíte de Teste “Realizar Compra na Plataforma”.

3.1 - Forma Tradicional de Documentação de Testes.

Suíte de Teste: Realizar Compra na Plataforma.

Caso de Teste: Pesquisar Produtos

Etapas e Condições: Cliente “Logado” e visualizando a tela principal da loja na plataforma.

Passo 1:

Ação: Cliente clica no campo “pesquisar”.

Sistema: seleciona o campo “pesquisar” e permite a digitação.

Passo2:

Ação: Cliente digita o nome do produto e pressiona “Enter”.

Sistema: verifica o nome digitado, realiza a busca e retorna na tela o produto ou a lista de produtos relacionados.

3.2 - No formato BDD e Cucumber (Pronto para Automação).

Conforme sugerido, vou acrescentar a forma que eu gosto de fazer hoje, pois, além de deixar os casos prontos para Automação dos Testes, também facilita muito o trabalho dos Desenvolvedores BackEnd na hora de criar seus testes unitários.

Suíte de Teste: Realizar Compra na Plataforma.

Caso de Teste: Pesquisar Produtos

Etapas e Condições: Cliente “Logado” e visualizando a tela principal da loja na plataforma.

Roteiro de Teste(Cucumber):

Dado que o cliente está realizando uma busca de um produto

E o mesmo digita o nome do produto no campo “pesquisar”.

Quando solicitar para efetua a pesquisa

Então o sistema deverá buscar as informações no BD

E deverá exibir na tela o produto(s) relacionado(s) com a pesquisa feita pelo cliente.

4 – Suíte de Teste: API (exemplo de documentação)

Abaixo segue o roteiro para configurar a ferramenta e executar o teste.

4.1 - Definições:

- **Ferramenta:** POSTMAN
- **Suíte de Teste:** Realizar Compra na Plataforma.
- **Caso de Teste:** Cadastrar Clientes

4.2 - Configuração para o teste no Postman (dados fictícios para ilustração)

Nome Request: Cadastro de Cliente

Tipo Request: POST

URL Resquest: http://xxx_diretorio/xxx_servico

Na Aba **Headers:**

- Key: Content-Type / Value: application/json

Na Aba **Body** (segue o exemplo de alguns campos que contem na plataforma)

```
{
  "email": "xxxx@com.com",
  "frist_name": "nome do cliente",
  "last_name": "sobrenome cliente",
  "password": "xxxxxxx";
  "addresses": [
    {
      "street": "xxxxxxxx",
      "number": "xx",
      "city": "xxxxxxxx"
    }
  ]
}
```

4.3 - Resultado da Request:

- **Status:** 201 (pois foi realizado um cadastro)
- **Body:** Mensagem do cadastro realizado com sucesso

5- Relatório de Bugs encontrados:

5.1 - **Erro:** Email secundário para recuperação de senha aceitando o mesmo email que está sendo cadastrado:

Onde Ocorre: campo "Assign an address alias for future reference." da tela "cadastro de clientes"

Como simular: Na tela "Create an Account", informe o mesmo email que está sendo cadastrado e finalize o cadastro.

5.2 - **Erro:** Quantidade de produtos para enviar para o carrinho aceitando números muito grandes (99999999)

Onde Ocorre: campo "Quantity" da tela de "add to cart".

Como simular: Ao selecionar um produto, informe no campo "Quantity" um alto para o item e adicione no carrinho.

5.3 - **Erro:** Mesmo fechando o browser o cliente continua logado no site.

Onde Ocorre: Site da plataforma.

Como simular: Ao fazer o Login, fechar o browser e abrir novamente, quando acessar o site o cliente continuará "logado".

5.4 - **Erro:** Ao tentar confirmar uma compra com itens maior que do estoque, a plataforma retorna 500 na tela.

Onde Ocorre: Na tela de confirmação de pagamento etapa "Payment".

Como simular: Ao enviar para o carrinho um produto com o número de itens maior que o estoque e tentar confirmar a opção de pagamento para finalizar a compra. Então é exibido o erro de retorno 500 na tela.