



## **PROGRAMA DE BOLSAS - TESTE COM RUBY**

### **PLANO TESTE SITE BUGBANK**

**Nome: Iura Kurtz Filho**

## **1. Introdução**

O site da BugBank é uma aplicação que tem como finalidade, a mais próxima assemelhação com um projeto real, para que os usuários possam realizar praticar fielmente a automação de testes em sites diversos. Para garantir que o site continue a oferecer uma experiência de usuário satisfatória, é importante realizar testes regulares em suas funções e recursos.

## **2. Objetivos**

Este projeto de plano de testes tem como objetivo estabelecer um conjunto de testes para os fluxos principais do site. Os testes em questão ajudarão a garantir que os usuários possam utilizar o site com facilidade e praticidade.

## **3. Escopo**

O plano de testes para o site da Bug Bank incluirá apenas testes prioritários para os fluxos principais das operações, que envolvem as seguintes etapas:

- I. Acessar o site da BugBank e realizar cadastro(ou login caso já possua conta).
- II. Selecionar a operação que deseja realizar.
- III.Caso a operação "Extrato" seja selecionada.
- IV. Visualizar o saldo disponível, as movimentações e o saldo atual no dia das movimentações.
- V. Caso a operação "Transferência" seja selecionada.
- VI. Preencher campos necessários para realizar transferência.
- VII. Finalizar a compra e verificar a confirmação do pedido.

Os testes serão executados em diferentes navegadores e dispositivos, para garantir a compatibilidade e a usabilidade do site. Os critérios de aceitação, os casos de teste, os dados de teste e os resultados esperados serão documentados em uma planilha. Os defeitos encontrados serão reportados em uma ferramenta de gerenciamento de defeitos, com as evidências necessárias para a reprodução e a correção dos mesmos.

## **4. Suíte de Casos**

### **Caso de Teste CT1: Login**

- CT 101: Login de forma convencional;
- CT 102: Validar a autenticação dos dados;
- CT 103: Validar preenchimento de campos;
- CT 104: Validar login sem preenchimento de todos os campos
- CT 105: Validar login com conta inválida

### **Caso de Teste CT2: Cadastro**

- CT 201: Validar preenchimento de campos obrigatórios;
- CT 202: Verificar igualdade de campo de confirmação de senha e campo de senha;
- CT 203: Validar duplicidade do cadastro.
- CT 204: Validar comportamento sem preenchimento de todos os campos

### **Caso de Teste CT03: Home**

- CT 301: Validar redirecionamento de links para página de transferência;
- CT 302: Validar redirecionamento de links para página de extrato;
- CT 303: Validar exibição do saldo correspondente;
- CT 304: Validar exibição do Nome correspondente;
- CT 305: Validar botão de Logout

### **Caso de Teste CT 04: Transferências**

- CT 401: Validar inserção de uma conta existente;

- CT 402: Validar exibição de pop up de sucesso ou de erro.
- CT 403: Validar comportamento com valores negativos
- CT 404: Validar transferência para conta inválida
- CT 405: Validar transferência sem preenchimento de campos

#### **Caso de Teste CT05: Extrato**

- CT 501: Validar se o saldo atual é exibido corretamente;
- CT 502: Validar se movimentações da conta são exibidas nos dias correspondentes;
- CT 503: Validar se saldo atual no dia das movimentações é exibido corretamente;

## **5. Estratégia de teste**

Dado que o site da BugBank é um site que visa se assimilar com um Banco, e tem como principal objetivo a prática de planejamento de testes de automação, então priorizamos todas funcionalidades disponíveis no site.

Assim, para cada funcionalidade, será testado as diversas formas que o cliente pode percorrer usando o site.

## **6. Prioridades**

Como já mencionado anteriormente, a prioridade deste plano de teste, é validar os fluxos de Transferência e visualização de Extrato do site BugBank, pois, são essas as principais funcionalidades do mesmo. Através deste plano, será testado as diversas variantes do fluxo, até que o usuário esteja satisfeito

## **7. Testes Candidatos à automação**

Devido ao plano de teste ter como finalidade a validação dos principais fluxos do

site, que são os fluxos de Transferências e visualização de Extrato, todas as variantes dos casos de testes são candidatas a automação.

Para isso, deverão ser organizadas as variantes de cada funcionalidade de forma que as mesmas sigam um fluxo, desde quando o usuário chegue ao site até a realização do que é desejado.

Segue um exemplo:

Automação 01:

- CT101: Login de forma convencional;
- CT106 : Validar login com conta inválida;
- CT 201: Validar preenchimento de campos obrigatórios;
- CT 302: Selecionamento da operação de transferência;
- CT402: Validar preenchimento de campos necessários;
- CT 501: Confirmar transferência;
- CT 504: Validar exibição de mensagem de Transferência realizada;
- CT 504: Validar mudança no saldo da conta.

## **8. Ferramentas**

Para realização do plano de teste em questão as ferramentas utilizadas foram:

- Testes exploratórios, para tomada de decisões de prioridades de testes;
- Cucumber e BDD, para facilitação de feedback dos stakeholders;
- Linguagem de programação Ruby, com gems de cucumber, capybara, rspec, siteprism e selenium-webdriver, para a automação dos testes.

## **9. Como os resultados do teste serão divulgados**

a. Um relatório com os resultados do teste, no formato PDF, deverá ser entregue na plataforma Moodle. Esse relatório deverá ter a seguinte estrutura:

1. Informações sobre o site
2. Dados obtidos

3. Análise estatística dos resultados. Para cada pergunta deverá ser apresentada uma tabela com todos os tempos obtidos em segundos. Além disso, a média e o desvio padrão devem ser calculados. O grupo deverá dissertar sobre os resultados obtidos.
- b. Os testadores do período matutino e noturno terão datas diferentes para enviar seus relatórios e apresentar os resultados
  - Matutino: 22/09 e 23/09
  - Noturno: 22/09 e 23/09

## 10. Cronograma

ATIVIDADES	05/03 09/09	10/09 12/09	13/09 14/09	15/09 18/09	19/09 19/09	20/09 21/09	22/09 22/09
Tirar dúvidas							
Testes exploratórios							
Revisão dos requisitos e preparação do ambiente de teste							
Execução dos testes							
Análise dos resultados dos testes e relatório de bugs							
Revisão final e preparação do relatório de teste							

ID	Funcionalidade/componente	Título	Passo-a-passo para execução	Resultado Esperado	Status	Observação
CT101	Login	Validar login de forma convencional	1: Acessar página inicial do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: preencher os campos de email e senha; 3:Clicar em Acessar	Deverá ser redirecionado para a página de Home	OK	
CT102	Login	Validar autenticação de dados	1: Acessar página inicial do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: preencher os campos de email sem o @ e o campo de senha apenas com um dígito; 3:Clicar em Acessar	Deverá não ser possível logar-se e mensagens aparecerão nos campos	OK	
CT103	Login	Validar Preenchimento de campos	1: Acessar página inicial do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: preencher os campo de email; 3: preencher campo de senha; 4: clicar em botão “Acessar”	Campos devem exibir corretamente as informações inseridas	OK	
CT104	Login	Validar Login sem preenchimento dos campos	1: Acessar página inicial do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: não preencher os campo de email; 3:não preencher campo de senha; 4: clicar em botão “Acessar”	Botão de “Acessar” deve ficar desabilitado	OK	
CT105	Login	Validar Login com conta Inválida	1: Acessar página inicial do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: preencher os campos de email não cadastrado; 3: preencher campo de senha; 4: clicar em botão “Acessar”	Uma mensagem de conta inválida deverá ser apresentada	OK	
CT201	Register	Validar preenchimento de campos obrigatórios	1: Acessar página de registro do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: preencher os campos de email cadastrado; 3: preencher campo de Nome. 3: preencher campo de senha; 5:preencher campo de confirmação de senha; 4: clicar em botão “Acessar”	Campos deverão apresentar corretamente as informações inseridas e usuário poderá se cadastrar	OK	
CT202	Register	Verificar igualdade de campo de confirmação de senha e confirmação de senha	1: Acessar página de registro do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: preencher os campos de email cadastrado; 3: preencher campo de Nome. 4: preencher campo de senha; 6:preencher campo de confirmação de senha; 7: Verificar se as senhas são iguais	Sistema deverá apresentar senhas iguais que o usuário informou corretamente	OK	

CT203	Register	Validar duplicidade de cadastro	1: Acessar página de registro do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: preencher os campos de email cadastrado; 3: preencher campo de Nome. 4: preencher campo de senha; 6:preencher campo de confirmação de 7:Clicar em “Acessar” 8: Cadastrar novamente uma conta com mesmo email	Deverá ser impossibilitada a possibilidade e mostrar uma mensagem de conta já existente	OK	
CT204	Register	Validar comportamento sem preenchimento de todos os campos	1: Acessar página de registro do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: Não preencher nenhum dos campos de cadastro 3:Clicar em “Acessar”	Deverá ser apresentado uma mensagem de campos obrigatórios para preenchimento	OK	
CT301	Home	Validar redirecionamento de links para página de transferência	1: Acessar página de Home do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: Clicar no botão “Transferência”	Deve ser redirecionado para página de transferências	OK	
CT302	Home	Validar redirecionamento de links para página de Extrato	1: Acessar página de Home do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: Clicar no botão “Extrato”	Deve ser redirecionado para página de Extrato	OK	
CT303	Home	Validar exibição do saldo correspondente	1: Acessar página de Home do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: Encontrar texto que mostra saldo atual na Home 3:Validar se saldo é correspondente com o atual	Deve ser apresentado corretamente o saldo atual	OK	
CT304	Home	Validar exibição do Nome do usuário	1: Acessar página de Home do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: Encontrar texto que mostra o nome do usuário 3:Validar se o Nome é informado corretamente	Deve ser apresentado corretamente o nome do usuário	OK	
CT305	Home	Validar botões de Logout	1: Acessar página de Home do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2:Clicar no botão “Sair”	Deverá ser redirecionado para página de login	OK	
CT401	Transferência	Validar inserção de um conta existente	1: Acessar página de Transferência do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: preencher campo de número da conta com uma conta existente 3: preencher campo de valor de transferência	Transferência deve ser realizada com sucesso e exibir uma mensagem de	OK	



			4: preencher descrição 5: Clicar em “Efetuar transferência”	sucesso;		
CT402	Transferência	Validar exibição de pop up de sucesso ou erro	1: Acessar página de Transferência do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: preencher campo de número da conta; 3: preencher campo de valor de transferência; 4: preencher descrição; 5: Clicar em “Efetuar transferência”	Se a conta realmente existir deverá ser apresentado um popup de sucesso, caso contrário deverá ser apresentado um popup de conta inválida	OK	
CT403	Transferência	Validar comportamento com valores negativos	1: Acessar página de Transferência do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: preencher campo de número da conta; 3: preencher campo de valor de transferência com número negativo; 4: preencher descrição; 5: Clicar em “Efetuar transferência”	Deverá ser apresentado um popup de transferência inválida	OK	
CT404	Transferência	Validar transferência para conta inválida	1: Acessar página de Transferência do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: preencher campo de número de uma conta inexistente; 3: preencher campo de valor de transferência; 4: preencher descrição; 5: Clicar em “Efetuar transferência”	Deverá ser apresentado um popup de conta inválida	OK	
CT405	Transferência	validar transferência sem preenchimento de campos	1: Acessar página de Transferência do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: Deixar os campos em branco; 5: Clicar em “Efetuar transferência”	Deverá ser apresentado um popup de transferência inválida	OK	
CT501	Extrato	Validar se o saldo atual é exibido corretamente	1: Acessar página de Extrato do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: Encontrar texto que exibe o saldo atual 5: Verificar se é correspondente ao saldo atual real	O saldo atual exibido deve ser correspondente ao saldo real	OK	
CT502	Extrato	Validar movimentações são exibidas corretamente	1: Acessar página de Extrato do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: Encontrar texto que exibe as movimentações da conta 5: Verificar se as movimentações são correspondente ao aos dias efetuadas	As informações das movimentações devem ser exibidas corretamente	OK	
CT503	Extrato	Validar se saldo atual é exibido corretamente no dia	1: Acessar página de Extrato do site: <a href="https://bugbank.netlify.app/">https://bugbank.netlify.app/</a> 2: Encontrar texto que exibe o saldo	O saldo deve ser alterado exibido	OK	

		das movimentações	atual no dia das movimentações da conta 5: Verificar se o saldo corresponde corretamente	corretamente conforme as movimentações		
--	--	-------------------	---	--	--	--