

**PLANO DE TESTES PARA O SITE PARABANK**

**07/09/2023**

**Sumário**

**Introdução3**

**Objetivos3**

**Escopo3**

**Mapa Mental4**

**Suíte de Caso de Testes4**

**Estratégia de Testes6**

**Prioridades6**

**Testes Candidatos a Automação6**

**Ferramentas7**

**Cronograma7**

1. **Introdução**

ParaBank é um site de demonstração, da atividade do setor bancário, que serve para a utilização de softwares de testes da empresa ParaSoft. Assim, através dos conhecimentos adquiridos na trilha, sobre testes e automação de testes, serão executados os testes para verificar as funcionalidades deste site.

# Objetivos

Este projeto de plano de testes tem como objetivo estabelecer um conjunto de testes para validar as principais funcionalidades do site. Os testes em questão ajudarão a garantir que funcionalidades como criar uma conta, fazer uma transferência ou um depósito estejam funcionando conforme o objetivo do site.

# Escopo

O plano de testes para o site da ParaBank incluirá além dos testes prioritários, que são as princiais funcionalidades de um banco, como criar uma conta, fazer uma transferência, fazer um depósito, gerar um extrato bancário, também irá verificar funcionalidades secundárias do site, como validação de links de redirecionamento e páginas de informações sobre a empresa.

Os testes serão executados em diferentes navegadores, 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.

# Mapa Mental

# 

# Suíte de Casos

Suíte de Teste CT01: Login

* CT101: Validar preenchimento de campos;
* CT102: Validar link “Forgot Login Info?”;
* CT103: Validar link “Register”;
* CT104: Validar login com uma conta válida;
* CT105: Validar login com uma conta inválida;

Suíte de Teste CT02: Register

* CT201: Validar preenchimento de campos;
* CT202: Validar cadastro com dados válidos;
* CT203: Validar cadastro com dados inválidos;
* CT204: Validar duplicidade de cadastro.

Suíte de Teste CT03: Forgot Login Info?

* CT301: Validar preenchimento de campos;
* CT302: Validar recuperação dos dados de login de conta existente;
* CT303: Validar recuperação dos dados de login de conta inexistente.

Suíte de Teste CT04: Account Services

* CT401: Validar abertura de conta (“Open new account”);
* CT402: Validar visão geral das contas (“Accounts Overview”);
* CT403: Acessar detalhes de uma conta;
* CT404: Validar transferencia de fundos (“Transfer Funds”);
* CT405: Validar pagamento de contas (“Bill Pay”);
* CT406: Validar extratos de conta (“Find Transations”);
* CT407: Validar atualizar informações de contato (“Update Contact Info”);
* CT408: Validar solicitar empréstimo (“Request Loan”);
* CT409: Validar Log Out;

Suíte de Teste CT05: Header Links e Footer Links

* CT501: Validar redirecionamento de links (“Menu superior a esquerda”);
* CT502: Validar redirecionamento de links (“Botões a direita”);
* CT503: Validar redirecionamento de links (“Footer”);
* CT504: Validar formulário de contato;

1. **Estratégia de teste**

Neste plano de teste para o site ParaBank, dividimos os testes pelas funcionalidades e criamos suítes de caso de testes, priorizando as funcionalidades necessárias para que de fato a aplicação cumpra com eu papel como banco.

Como pode ser visto, primeiramente estão as funcionalidades de login, seguido das funcionalidades do setor bancário e por fim as funcionalidades secundárias.

1. **Prioridades**

Como itens prioritários de testes foram adotados os casos:

* Suíte de Teste CT01: Login;
* Suíte de Teste CT02: Register;
* Suíte de Teste CT04: Account Services;

Foram definidos como prioritários as suítes de testes acima, visto que em caso de mal funcionamento ou a não execução, impedem o principal objetivo do site, que é criar uma conta e fazer movimentações bancárias.

1. **Testes Candidatos à automação**

Todos os casos de testes elencados neste plano são candidatos à automação e deverão após serem testados manualmente, serem automatizados.

1. **Ferramentas**

Para a realização do plano de testes em questão as ferramentas utilizadas serão:

* Testes exploratórios e manuais;
* Cucumber e BDD, para facilitação de feedback dos stakeholders;
* Linguagem de programação Ruby para a automação dos testes juntamente com a ferramente Selenium Web Driver;

1. **Cronograma**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ATIVIDADES** |  | **05/set** | **06/set** | **08/set** | **11/set** | **12/set** | **13/set** | **14/set** | **15/set** |
| **Planejamento** |  |  |  |  |  |  |  |  |  |
| **Execução do Plano de Testes** |  |  |  |  |  |  |  |  |  |
| **Testes exploratórios** |  |  |  |  |  |  |  |  |  |
| **Relatório de Bugs encontrados** |  |  |  |  |  |  |  |  |  |
| **Automação de testes** |  |  |  |  |  |  |  |  |  |
| **Apresentação relatório de testes** |  |  |  |  |  |  |  |  |  |