**PLANO DE TESTE: Soturno**

Rodrigo Lopes

Leonardo Ravaleli

Cento universitário Senac Santo Amaro

**Histórico de revisões**

|  |  |  |  |
| --- | --- | --- | --- |
| Versão | Data | Autor | Descrição |
| 1.0 | **10/10/2023** | Rodrigo |  |

PLANO DE TESTE

O objetivo deste documento é descrever o Plano de Teste do Projeto de Software XXXXXX. Ele contém a definição do escopo do Projeto de Software, a especificação dos requisitos do sistema, a estratégia de teste e o esforço necessário para sua realização. Na estratégia de teste estão definidos as técnicas, níveis e tipos de teste que serão executados na iteração e os objetivos que devem ser atingidos. O documento apresenta ainda a definição do ambiente de testes, a matriz de responsabilidade e a relação dos casos de testes. Em seguida são listados os riscos de teste, os critérios para conclusão dos testes e o cronograma do projeto de teste.

<Para descrever cada capítulo/item deste plano, procurem relacioná-los aos conceitos discutidos na disciplina de Teste de *Software*. Para isso, utilizem referências de autores da área de engenharia e teste de *software*. Todas as obras pesquisadas e citadas no texto devem estar relacionadas no final deste documento.>

Sumário

[1 ESCOPO DO PROJETO DE sOFTWARE 4](#_Toc67924660)

[*1.1* *Objetivo do Projeto* 4](#_Toc67924661)

[*1.2* *Metodologia de desenvolvimento* 4](#_Toc67924662)

[2 Matriz de rastreabilidade de requisitos de sistema 5](#_Toc67924663)

[*2.1* *Requisitos Funcionais* 5](#_Toc67924664)

[*2.2. Requisitos Não Funcionais* 6](#_Toc67924665)

[3 ESTRATÉGIA DE TESTE 7](#_Toc67924666)

[4 AMBIENTE de TESTE 8](#_Toc67924667)

[5 MATRIZ DE RESPONSABILIDADE 9](#_Toc67924668)

[6 RELAÇÃO DOS CASOS DE TESTES 10](#_Toc67924669)

[7 RISCOS DO PROJETO DE TESTE DE SOFTWARE 11](#_Toc67924670)

[8 CRITÉRIOS DE CONCLUSÃO 12](#_Toc67924671)

[9 CRONOGRAMA DO PROJETO DE TESTE DE SOFTWARE 13](#_Toc67924672)

[REFERÊNCIAS 14](#_Toc67924673)

# ESCOPO DO PRODUTO DE sOFTWARE

A declaração do escopo do produto de software descreve, em detalhes, as entregas do produto e o trabalho necessário para criar essas entregas. Fornece um entendimento comum do escopo do produto a todas as partes interessadas no projeto e descreve os principais objetivos do projeto. Além disso, permite que a equipe do projeto realize um planejamento mais detalhado, orientando a execução do projeto, fornecendo linha de base para avaliar solicitações de mudanças ou trabalho adicional.

## *Objetivo do Projeto*

*[*Trata-se do norte, onde este projeto deve nos levar, o que deseja-se alcançar*.]*

## *Metodologia de desenvolvimento*

< Nesta parte descreva o processo utilizado para desenvolvimento do software. Pode ser acrescentado artefatos que trazem maior clareza da solução proposta, tais como desenho do processo de desenvolvimento, metodologia de desenvolvimento, protótipo, arquitetura do sistema.>

# Matriz de rastreabilidade de requisitos de sistema

<Descreva aqui a importância da rastreabilidade dos requisitos do sistema para o planejamento de teste de software.>

## *Requisitos Funcionais*

Descreve as características do produto, serviço ou resultado para cuja criação o projeto será realizado. Essas características terão normalmente menos detalhes nas fases iniciais e mais detalhes nas fases posteriores, conforme as características do produto forem progressivamente elaboradas. Embora a forma e o conteúdo das características variem, as descrições dos requisitos funcionais devem sempre fornecer detalhes suficientes para dar suporte ao planejamento do projeto de software e teste.

|  |  |  |
| --- | --- | --- |
| **REQUISITOS FUNCIONAIS (RF)** | | |
| **ID** | **Nome do RF** | **Descrição** |
| RF01 | logar | O usúario pode logar, existindo 2 tipos de contas, a do cliente e a do a adiministrador |
| RF02 | comprarPoção | O cliente pode efetuar a compra de uma poção |
| RF03 | comprarFeitiço | O cliente pode efetuar a compra de um feitiço |
| RF04 | cadastrarPoção | O administrador pode cadastrar uma poção |
| RF05 | cadastrarFeitiço | O administrador pode cadastrar um feitiço |
| RF06 | consultarPoção | O administrador pode consultar uma poção |
| RF07 | consultarFeitiço | O administrador pode consultar um feitiço |
| RF08 | cadastrarCliente | O cliente pode criar uma conta, caso não tenha |
|  |  |  |
|  |  |  |
|  |  |  |

## *2.2. Requisitos Não Funcionais*

Segundo Pressman (2016), requisitos não-funcionais são os requisitos relacionados ao uso da aplicação em termos de desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenção e tecnologias envolvidas. Estes requisitos dizem respeito a como as funcionalidades serão entregues ao usuário do software.

|  |  |  |  |
| --- | --- | --- | --- |
| **REQUISITOS NÃO-FUNCIONAIS** | | | |
| **RNF** | **Categoria** | **Nome** | **Descrição** |
| 01 | Desempenho |  |  |
|  |  |
|  |  |
| 02 | Usabilidade |  |  |
|  |  |
|  |  |
| 03 | Segurança |  |  |
|  |  |
|  |  |
| 04 | Portabilidade |  |  |
|  |  |
|  |  |
| 05 | ...... |  |  |

|  |  |
| --- | --- |
|  |  |

# ESTRATÉGIA DE TESTE

A estratégia de teste define quais técnicas, níveis e tipos de testes serão utilizados no projeto.

<Descreva aqui de forma detalhada quais técnicas, níveis e tipos de testes serão utilizados no projeto, relacionando com os requisitos de sistema.>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ESTRATÉGIA DE TESTE** | | | | |
| **NÍVEIS DE TESTE** | | | | |
| **ID** | **Nome do Teste** | **Técnicas** | **Descrição** | **Requisitos de Sistemas** |
| 01 | Teste Unitário |  |  | RF01, RF02... |
| 02 | Teste de Integração |  |  |  |
| 03 | Teste de Sistema |  |  |  |
| 04 | Teste de Aceitação |  |  |  |
| **TIPOS DE TESTES** | | | | |
| **ID** | **Tipo do Teste** | **Técnicas**  **Descrição** | | **Requisitos de Sistemas** |
| 01 | Desempenho |  | |  |
| 02 | Usabilidade |  | |  |
| 03 | Portabilidade |  | |  |
| ... | ..... |  | |  |

# AMBIENTE de TESTE

O ambiente de testes mostra as condições em que serão executados os testes.

<Descreva aqui a importância da definição do ambiente de teste no planejamento de teste com base nos autores da área de engenharia e teste de software.>

| **AMBIENTE DE TESTE** | |
| --- | --- |
| **Recurso** | **Descrição** |
| **Computador cliente** |  |
| **Mobile** |  |
| **Servidor de aplicação** |  |
| **Servidor de banco de dados** |  |
| **Criação de relatórios** |  |
| **Browser cliente** |  |
| **SO cliente** |  |
| **<< Outro recurso>>** | <Informe a descrição completa. Apague esta linha da tabela se não existir outro recurso.> |

# MATRIZ DE RESPONSABILIDADE

<Descreva aqui a importância da definição da matriz de responsabilidade no planejamento de teste com base nos autores da área de engenharia e teste de software.>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MATRIZ DE RESPONSABILIDADE** | | | | |
| **Fase do Projeto** | **Atribuição Técnica** | **Responsabilidade** | **Nome** | **Descrição da Atividade** |
| Planejamento | Líder de Projeto de Teste | Técnico responsável pela liderança de um projeto de teste específico, geralmente relacionado a um sistema de desenvolvimento, seja um projeto novo ou uma manutenção. |  |  |
| Arquiteto de teste |  |  |  |
| Especificação | Analista de Teste |  |  |  |
| Execução | Testador |  |  |  |
| Resultados |  |  |  |  |

# RELAÇÃO e descrição DOS CASOS DE TESTES

Em [engenharia de software](https://pt.wikipedia.org/wiki/Engenharia_de_software), o caso de teste é um conjunto de condições usadas para [teste de software](https://pt.wikipedia.org/wiki/Teste_de_software). Neste plano de teste consta a relação de todos os cenários e seus respectivos casos de testes a serem especificados pelos analistas de teste.

|  |  |  |  |
| --- | --- | --- | --- |
| **CASOS DE TESTES** | | | |
| **ID** | **Cenário de Teste** | **Caso de Teste** | **Descrição** |
| 01 | CT\_LOGIN | CT01\_Novo Usuário | Caso de teste para validar o cadastramento de um novo usuário. |
| CT02\_Usuário Cadastrado | Caso de teste para validar o acesso de um novo usuário já cadastrado. |
| CT03\_Esqueci a senha | Caso de teste para validar a reinicialização de senha para o usuário. |
| 02 |  |  |  |
|  |  |
|  |  |
| 03 |  |  |  |
|  |  |
|  |  |
| 04 |  |  |  |
|  |  |
|  |  |
| 05 |  |  |  |
|  |  |
|  |  |

## *Descrição dos principais cenários de testes*

<Utilizando as técnicas de caixa preta descreva os principais cenários de testes do projeto (mínimo 3).>

# RISCOS DO PROJETO DE TESTE DE SOFTWARE

Este capítulo contém um sumário dos riscos chaves do projeto de teste, que se acontecerem, poderão afetar os resultados esperados e o projeto de software.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RISCOS DO PROJETO DE TESTE** | | | | |
| **ID** | **Risco** | **Descrição** | **Grau do Risco** | **Resposta** |
| 01 | Hardware | Risco do equipamento (hardware) utilizado para execução da homologação do teste falhar. | Baixo | Mitigar – equipamento de reserva |
| 03 |  |  |  |  |
| 04 |  |  |  |  |
| 05 |  |  |  |  |

# CRITÉRIOS DE CONCLUSÃO

Os critérios de conclusão/continuidade dos testes determinam qual a tolerância a falhas, ou seja, partir deste momento se os testes serão suspensos. Detalhando quais critérios serão utilizados para suspensão de um teste e retomada do desenvolvimento, ou mesmo continuidade apesar dos cenários de falha, ou seja, se após executado um determinado número de cenários de testes obtiver-se um número significativo de falhas que os testes são suspensos e a entrega deverá passar por uma revisão completa, ou mesmo se determinada falha ocorrer os testes não prosseguirão até que a mesma esteja corrigida. Aqui também se define as condições as quais o teste prosseguirá ou não se os dados de teste não estiverem em conformidade com a massa especificada anteriormente.

<Defina quais os critérios para aceitação, parada e retomada dos testes.>

| **Critério** | **Descrição** |
| --- | --- |
| **Cobertura** | * Todos os requisitos definidos no escopo deste projeto deverão ser testados. * Testar fluxos básico, alternativo e de exceção de cada requisito. * Todas as regras de negócio serão testadas. |
| **Quando parar de testar** | * Quando nenhuma falha for encontrada após executar todos os casos de teste pelo menos uma vez. * Quando for encontrado um número excessivo de falhas logo no início dos testes. * Quando for encontrada alguma falha que inviabilize a continuidade dos testes. * Quando atingir no mínimo os seguintes critérios:   + Casos de Teste     - 100% dos casos de teste executados   + Defeitos     - 100% dos defeitos de severidade 1 corrigidos e verificados     - 90% dos defeitos de severidade 2 corrigidos e verificados     - 75% dos defeitos de severidade 3 corrigidos e verificados |
| **Em que momento reportar as falhas** | * Reportar todas as falhas imediatamente após encontrá-las. |
| **Teste de regressão** | * O teste de regressão será feito sobre a falha encontrada, sendo um fluxo básico e um fluxo alternativo do caso de uso. |

# CRONOGRAMA DO PROJETO DE TESTE DE SOFTWARE

<Descreva aqui a importância do cronograma do projeto no planejamento de teste com base nos autores da área de engenharia e teste de software, e insira o cronograma das atividades a serem executadas no projeto de teste.>

# REFERÊNCIAS

PRESSMAN, Roger S; MAXIM, Bruce R. **Engenharia de Software:** Uma Abordagem Profissional. 8ª ed. Editora McGraw-Hill,2016.

<Todos os autores citados no texto devem ser incluídos na lista de referências, que seguirão a formatação da NBR 6023:2002. A referência é composta pelos seguintes elementos: autor, título, edição, local, editora e data de publicação. Devem ser alinhadas à margem esquerda do texto com espaço simples e separadas umas das outras por um espaço simples. Usar o mesmo tipo e tamanho de fonte do texto do artigo, *Arial*, tamanho 12. Não deve ser numerada.>

EXEMPLOS E SUGESTÕES:

DELAMARO, Márcio Eduardo; MALDONADO, José Carlos; JINO, Mario. **Introdução ao Teste de Software.** São Paulo: Elsevier, 2007.

MOLINARI, Leonardo. **Inovação e Automação de Testes de Software**. Rio de Janeiro: Érica, 2010.

RIOS, Emerson. **Análise de riscos**: em projetos de teste de software. Rio de Janeiro: Alta Books, 2005.

RIOS, Emerson; Moreira Filho, Trayahú. **Teste de Software.** Rio de Janeiro:Alta Books, 2006.

SOMMERVILLE, Ian. **Engenharia de software**. 9. ed. São Paulo: Pearson, 2011.