**Sistema de Teste Estrutural**

***Plano de Testes***

**Resumo**

Este documento tem o objetivo de apresentar o planejamento dos testes do Projeto Teste Estrutural para programas escritos em PHP.

***Índice***

[1. Expectativas de alto nível 2](#_Toc498686802)

[2. Quais atributos de qualidade e confiabilidade 3](#_Toc498686803)

[3. Pessoas, coisas e lugares 3](#_Toc498686804)

[4. Definições 3](#_Toc498686805)

[5. Responsabilidades Inter grupos 3](#_Toc498686806)

[6. Fases de teste 3](#_Toc498686807)

[7. Estratégias de teste 4](#_Toc498686808)

[7.1. Aplicação dos Testes 4](#_Toc498686809)

[7.2. Tipos de Evidência Utilizados 4](#_Toc498686810)

[7.3. Equipe de Testes 4](#_Toc498686811)

[8. Recursos necessários 4](#_Toc498686812)

[9. Atribuições dos testadores 5](#_Toc498686813)

[10. Planejamento do teste 5](#_Toc498686814)

[11. Casos de Teste 5](#_Toc498686815)

[11.1. Release 1 6](#_Toc498686816)

[11.1.1. Testes Unitários do Release 1 6](#_Toc498686817)

[11.1.2. Testes de Integração 6](#_Toc498686818)

[11.1.3. Testes de Aceitação 6](#_Toc498686819)

[12. Relatórios de Erros 7](#_Toc498686820)

[13. Métricas e Estatísticas 7](#_Toc498686821)

[14. Riscos e problemas 7](#_Toc498686822)

[15. Planejamento dos Testes 7](#_Toc498686823)

[16. Critérios para Aprovação de Versão 7](#_Toc498686824)

# Expectativas de alto nível

Este documento apresenta o plano de teste para o desenvolvimento de uma ferramenta de teste estrutural para programas escritos em PHP. Nela estão implementados os critérios de teste baseados em fluxo de controle.

# Quais atributos de qualidade e confiabilidade

Os atributos de qualidade e confiabilidade indispensáveis para o projeto são:

* Tolerância a falhas;
* Gravidade das falhas;
* Possibilidade de recuperação;
* Possibilidade de previsão;
* Exatidão;
* Tempo médio entre falhas;
* Testabilidade;
* Acurácia.

# Pessoas, coisas e lugares

Todos os membros do grupo desempenham todas as funções; implementador, gerenciador de testes e testador do sistema. O local de trabalho de cada um é individual, feito em sua própria máquina localmente e depois integrado através de um sistema de controle de versões, o GITHUB. Haverá reuniões presenciais para acertos e ajustes.

# Definições

# Responsabilidades Inter grupos

O implementador terá a responsabilidade de desenvolver a ferramenta proposta, além de manter-se em contato com o testador do sistema para validar o que foi feito. O testador do sistema deve executar a rotina de testes e verificar se os mesmos passam na porcentagem de aprovação definida. O gerenciador de testes coordena os testadores e define a estratégia de teste a ser executada.

# Fases de teste

|  |  |
| --- | --- |
| **Fase** | **Descrição** |
| Planejar Teste | * Identificar Requisitos para o Teste * Avaliar Risco * Desenvolver Estratégia de Teste * Identificar Recursos de Teste * Criar Planejamento * Gerar Plano de Teste |
| Projetar Teste | * Análise de Carga de Trabalho * Desenvolver Conjunto de Teste * Identificar e Descrever Casos de Teste * Identificar e Estruturar Scripts de Teste * Revisar e Acessar Cobertura de Teste |
| Implementar Teste | * Configurar Ambiente de Teste * Registrar ou Programar Scripts de Teste * Desenvolver Drivers e Stubs de Teste * Identificar funcionalidade específica do Teste no design e modelo de implementação * Estabelecer Conjuntos de Dados Externos |
| Executar Teste | * Executar Script de Teste * Avaliar Execução do Teste * Recuperar-se de Teste Interrompido * Verificar os Resultados * Investigar Resultados Inesperados * Registrar Defeitos |
| Avaliar Teste | * Avaliar a Cobertura dos Casos de Teste * Avaliar Cobertura do Código * Analisar Defeitos * Determinar se os Critérios de Conclusão do Teste e os Critérios de Êxito foram alcançados * Criar Relatório de Avaliação do Teste |

# Estratégias de teste

## Aplicação dos Testes

A finalidade deste documento é apresentar as informações necessárias para o planejamento e o controle dos testes de sistema a serem efetuados no projeto. Ele descreve a abordagem dada ao teste do software e é o plano para auxiliar na coordenação do esforço de teste.

Os estágios de teste a serem considerados são:

* Teste de Unidade: onde são executados testes para validar uma única unidade do sistema. Essas unidades são testadas a testa de maneira isolada, geralmente simulando as prováveis dependências que aquela unidade tem
* Teste de Integração: onde são executados testes para validar a integração entre duas partes do seu sistema.
* Teste de Aceitação: onde a versão entregue ao cliente será testada com a finalidade de determinar ou não a sua aprovação.

## Tipos de Evidência Utilizados

Como evidências da realização dos testes serão utilizados:

* Testes de unidade
* Testes de Integração
* Teste de Sistemas

## Equipe de Testes

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome** | **Função** | **Telefone/Ramal** | **E-mail** |
| Katyelen Costa | Tester | 98829-3197 | katyelenjf@gmail.com |
| André Archanjo | Tester | 99935-1904 | archanjoandre@yahoo.com.br |
| Matheus Marques | Tester | 98848-9048 | matheusocmarques@gmail.com |

# Recursos necessários

|  |  |  |
| --- | --- | --- |
| **Recursos Humano** | | |
| **Trabalhador** | **Recursos Mínimos** | **Responsabilidades Específicas/Comentários** |
| Gerente de Testes | 1. André Archanjo 2. Lucas Silveira 3. João Paulo 4. Katyelen Costa 5. Laercio Pioli 6. Matheus Marques | * Fornece supervisão de gerenciamento * Responsabilidades:   1. Fornecer direção técnica   2. Adquirir recursos apropriados   3. Relatório de gerenciamento |
| Testador do Sistema |  | * Executa os testes * Responsabilidades:   1. Executar testes   2. Registrar resultados   3. Recuperar-se de erros   4. Documentar defeitos |
| Implementador |  | * Implementa e faz o teste de unidade das classes de teste e pacotes de teste * Responsabilidades:   Cria os pacotes e classes de teste implementados no Conjunto de Teste. |

|  |  |
| --- | --- |
| **Recursos do Sistema** | |
| **Recurso** | **Nome / Tipo / Número de Série** |
| Pacote para representação e manipulação de grafo. | vis.js |
| Servidor Apache... | PHP |
| Xamp | Xamp |

# Atribuições dos testadores

# Planejamento do teste

# Casos de Teste

O plano de testes abrange os Testes de Sistema e Aceitação referentes aos casos de uso definidos abaixo:

|  |  |
| --- | --- |
| **Release** | **Caso de Uso** |
| *Release 1* | *UC –* |
| *UC-* |
| *UC-* |
| *UC-* |
| *UC-* |
| *UC-* |

## Release 1

### Testes Unitários do Release 1

|  |  |
| --- | --- |
| **Estratégia de Teste** | Tipo de Teste: Caixa Branca  Técnicas: Teste automatizado para validação de Caminho Básico e Condição e Teste de Regressão |
| **Critério de Sucesso** | Todas as classes devem ter sido testadas e estarem funcionando corretamente. |
| **Critérios de Interrupção** | Caso exista algum defeito que impeça a continuação dos testes o mesmo teve ser interrompido até que o problema seja resolvido. |
| **Ambiente de Teste** |  |

### Testes de Integração

|  |  |
| --- | --- |
| **Estratégia de Teste** | A estratégia usada será a mesma definida no Plano de Integração:  Testes Automatizados |
| **Critério de Sucesso** | Após integração, todos os componentes devem estar funcionando corretamente. |
| **Critérios de Interrupção** | Caso exista algum defeito que impeça a continuação dos testes o mesmo teve ser interrompido até que o problema seja resolvido. |
| **Ambiente de Teste** |  |

### Testes de Aceitação

|  |  |
| --- | --- |
| **Estratégia de Teste** | Será utilizada a mesma estratégia definida para os testes de Sistema. Os DETs serão enviados ao cliente para que os testes de aceitação sejam realizados. |
| **DETs a serem testados** | DETUC\_. |
| **Critério de Sucesso** | Todas as funcionalidades devem ter sido testadas e devem estar funcionando corretamente. |
| **Critérios de Interrupção** | Caso exista algum defeito que impeça a continuação dos testes o mesmo teve ser interrompido até que o problema seja resolvido. |
| **Ambiente de Teste** |  |
| **Observação** |  |

# Relatórios de Erros

# Métricas e Estatísticas

# Riscos e problemas

A tabela abaixo contém a análise inicial dos riscos envolvidos no projeto.

**Probabilidade:** percentual de probabilidade que o risco ocorra (Alta, média e baixa).

**Impacto:** percentual que deve indicar a grandeza do impacto do evento de risco no projeto.

**Responsável:** qual o membro da equipe do projeto que poderá responder pela situação de risco.

**Contingência:** O que deve ser feito em caso de o risco ocorrer.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Descrição** | **Probabilidade** | **Impacto** | **Responsável** | **Contingência** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Planejamento dos Testes

# Critérios para Aprovação de Versão

* Todas os DETs devem ter sido executados pelo menos uma vez;
* Não existir defeitos críticos;
* Os testes devem ter cobertura mínima de 80%;
* 90% dos bugs tenham sido corrigidos.