

Sati

Plano de Teste

Versão 1.0

Histórico da Revisão

Data	Versão	Descrição	Autor
05/12/2015	1.0	O foco desta versão é inspecionar os módulos de Fluxo de Caixa e Gerenciamento de Minicursos.	Giovane

Introdução	6
Finalidade	6
Escopo	6
Missão de Avaliação e Motivação dos Testes	6
Missão de Avaliação	6
Itens-alvo dos Testes	6
Abordagem dos Testes	6
Técnicas e Tipos de Teste	6
Teste de Unidade	6
Teste de Integração	7
Teste da Aceitação	7
Critérios de Entrada e de Saída	8
Plano de Teste	8
Critérios de Entrada de Plano de Teste	8
Critérios de Saída de Plano de Teste	8
Critérios de Suspensão e de Reinício	8
Ciclos de Teste	8
Critérios de Entrada de Ciclo de Teste	8
Produtos Liberados	8
Sumários de Avaliação de Testes	8

<u>Registros de Incidentes e Solicitações de Mudança</u>	8
<u>Produtos de Trabalho Adicionais</u>	9
<u>Resultados Detalhados dos Testes</u>	9
<u>Scripts de Teste Funcionais Automatizados Adicionais</u>	9
<u>Guia de Teste</u>	9
<u>Necessidades Ambientais</u>	9
<u>Elementos de Software Básicos do Ambiente de Teste</u>	9
<u>Ferramentas de Produtividade e de Suporte</u>	9
<u>Configurações do Ambiente de Teste</u>	10
<u>Aprovação e Encerramento</u>	10

Índice Analítico

<u>Introdução</u>	6
<u>Finalidade</u>	6
<u>Escopo</u>	6
<u>Missão de Avaliação e Motivação dos Testes</u>	6
<u>Missão de Avaliação</u>	6
<u>Itens-alvo dos Testes</u>	6
<u>Abordagem dos Testes</u>	6
<u>Técnicas e Tipos de Teste</u>	6
<u>Teste de Unidade</u>	6
<u>Teste de Integração</u>	7
<u>Teste da Aceitação</u>	7
<u>Critérios de Entrada e de Saída</u>	8
<u>Plano de Teste</u>	8
<u>Critérios de Entrada de Plano de Teste</u>	8
<u>Critérios de Saída de Plano de Teste</u>	8
<u>Critérios de Suspensão e de Reinício</u>	8
<u>Ciclos de Teste</u>	8

<u>Critérios de Entrada de Ciclo de Teste</u>	8
<u>Produtos Liberados</u>	8
<u>Sumários de Avaliação de Testes</u>	8
<u>Registros de Incidentes e Solicitações de Mudança</u>	8
<u>Produtos de Trabalho Adicionais</u>	9
<u>Resultados Detalhados dos Testes</u>	9
<u>Scripts de Teste Funcionais Automatizados Adicionais</u>	9
<u>Guia de Teste</u>	9
<u>Necessidades Ambientais</u>	9
<u>Elementos de Software Básicos do Ambiente de Teste</u>	9
<u>Ferramentas de Produtividade e de Suporte</u>	9
<u>Configurações do Ambiente de Teste</u>	10
<u>Aprovação e Encerramento</u>	10

Plano de Teste

1. Introdução

1.1 Finalidade

A finalidade do Plano de Teste de Iteração é reunir todas as informações necessárias ao planejamento e ao controle do esforço de teste referente a uma iteração específica. Ele descreve a abordagem dada ao teste do software e é o plano de nível superior gerado e usado pelos gerentes para coordenar o esforço de teste.

Este Plano de Teste referente ao Gerenciador de SATIs suporta os seguintes objetivos:

- Descrever as metodologias utilizadas para os testes.
- Identificar os elementos a serem testados.
- Define os recursos necessários para realização dos testes.
- Identifica os elementos que alcaçaram o critério de aceitação.

1.2 Escopo

Serão realizados teste de Unidade e Integração procurando testar Funcionalidade, Usabilidade, Confiabilidade, Desempenho e Suportabilidade que serão abordados por este **Plano de Teste**.

2. Missão de Avaliação e Motivação dos Testes

É o foco desta versão inspecionar os módulos de Fluxo de Caixa e Gerenciamento de Minicursos.

2.1 Missão de Avaliação

- Localizar erros de input de dados
- Localizar erros de persistência
- Verificar se os módulos estão em um padrão
- Verificar qualquer elemento faltante nos módulos.
- Certificar a qualidade de interface.

3. Itens-alvo dos Testes

São alvos dos testes as classes do controle e persistência dos módulos.

4. Abordagem dos Testes

4.1 Técnicas e Tipos de Teste

4.1.1 Teste de Unidade

Serão realizados testes em cada classe do sistema, para verificar o funcionamento da unidade individualmente e simular suas dependências.

Objetivo da Técnica:	Verificar se a unidade está funcionando e quais são suas dependências.
----------------------	--

Técnica:	Será utilizada a técnica do teste de caixa branca, onde será trabalhado diretamente no código da aplicação, para verificar se a classe traz os resultados esperados.
Estratégias:	Gerar scripts para automatizar os testes, utilizando o framework phpunit.
Ferramentas Necessárias:	Framework PHP Unit
Crítérios de Êxito:	O script deverá ser executado sem interrupções ocasionadas por falhas.
Considerações Especiais:	Os processos deverão ser disparados manualmente. E deverá ser feito o acompanhamento da execução dos scripts para análise dos erros.

4.1.2 Teste de Integração

Será realizado o teste de integração entre as unidades que tem dependências entre elas, .

Objetivo da Técnica:	Verificar se as unidades do sistema estão funcionando corretamente em conjunto.
Técnica:	Será utilizada a técnica do teste de caixa branca, onde será trabalhado diretamente no código da aplicação, para verificar se as classes estão funcionando corretamente em conjunto e trazendo os resultados esperados.
Estratégias:	Gerar scripts para automatizar os testes, utilizando o framework phpunit.
Ferramentas Necessárias:	Framework PHP Unit
Crítérios de Êxito:	O script deverá ser executado sem interrupções ocasionadas por falhas.
Considerações Especiais:	Os processos deverão ser disparados manualmente. E deverá ser feito o acompanhamento da execução dos scripts para análise dos erros.

4.1.3 Teste da Aceitação

Será realizado o teste de aceitação para verificar se o software está de acordo com as especificações do cliente, se o sistema funciona como um todo e também a qualidade de uso do sistema. Por isso, será realizado quando o software estiver concluído e pronto para ser implantado.

Objetivo da Técnica:	Verificar se está de acordo com as especificações do cliente, se o sistema funciona como um todo e a usabilidade do sistema.
Técnica:	Será utilizado a técnica de teste de caixa preta para verificar os requisitos da aplicação.
Estratégias:	Serão criados casos de testes para cada funcionalidade e para a integração das funcionalidades.
Ferramentas Necessárias:	
Crítérios de Êxito:	O sistema deverá funcionar corretamente como um todo, de acordo com os casos de testes e com os requisistos estabelecidos pelo cliente.

Considerações Especiais:	O analista de teste deverá reportar todos os erros encontrados na aplicação.
--------------------------	--

5. Critérios de Entrada e de Saída

5.1 Plano de Teste

5.1.1 Critérios de Entrada de Plano de Teste

Os testes de unidade poderão ser iniciados a partir do momento que for finalizado o desenvolvimento da classe, pelo próprio desenvolvedor. Os testes de integração poderão ser realizados a partir do momento que tiver classes que dependem uma da outra. E os testes de aceitação serão realizados com o sistema estiver concluído e pronto para implantação.

5.1.2 Critérios de Saída de Plano de Teste

O teste será considerado concluído quando não for encontrado mais erros na funcionalidade. Quando for encontrado algum erro, deverá ser corrigido e testado novamente a funcionalidade inteira, para verificar se a alteração no impactou em outra parte do funcionalidade e no caso dos testes de integração, em outra funcionalidade do sistema.

5.1.3 Critérios de Suspensão e de Reinício

Os testes serão suspensos caso seja encontrado algum erro que impeça o testador de continuar a verificação da funcionalidade. E será reiniciado após a correção do erro.

5.2 Ciclos de Teste

5.2.1 Critérios de Entrada de Ciclo de Teste

Um novo ciclo de teste poderá ser iniciado quando for corrigido um grupo de funcionalidades que se integram ou quando for realizada alguma alteração no sistema

6. Produtos Liberados

6.1 Sumários de Avaliação de Testes

Será criado um sumário com os resultados do teste e as principais medidas do teste para revisão e avaliação. Serão gerados ao fim de cada ciclo de teste.

6.2 Registros de Incidentes e Solicitações de Mudança

Será utilizada a ferramenta Test Case Web para gerenciamento dos testes, dos incidentes dos testes e das solicitações de mudança.

6.3 Produtos de Trabalho Adicionais

6.3.1 Resultados Detalhados dos Testes

Será gerado uma planilha do excel com as informações dos testes, do desenvolvedor e do testador. Também será informado a quantidade de erros, os tipos de erros encontrados.

6.3.2 Scripts de Teste Funcionais Automatizados Adicionais

Para cada teste realizado, será criado um script de automização com a ferramenta selenium IDE, que será disponibilizado para o restante da equipe através de um repositório.

6.3.3 Guia de Teste

Após realizar o teste, o responsável deverá gerar um guia de teste, explicando a funcionalidade e informando o que deve ser testado.

7. Necessidades Ambientais

7.1 Elementos de Software Básicos do Ambiente de Teste

São necessários os seguintes elementos de software básicos no ambiente de teste deste *Plano de Teste*.

Nome do Elemento de Software	Versão	Tipo e Outras Observações
Google Chrome	45 ou superior	Navegador da Internet
Mozilla Firefox	40 ou superior	Navegador da Internet
Network Associates McAfee Virus Checker		Software de Detecção e Recuperação de Vírus

7.2 Ferramentas de Produtividade e de Suporte

Serão utilizadas as seguintes ferramentas para suportar o processo de teste deste *Plano de Teste*.

Categoria ou Tipo de Ferramenta	Nome da Marca da Ferramenta	Fornecedor ou Desenvolvida Internamente	Versão
Gerenciamento de Teste	Teste Case Web	Keith Tyler	
Controle de Defeitos	Bugzilla	bugzilla.org	5.0.1
Automatização de testes	PHP Unit	Sebastian Bergmann	5.1
Ferramenta ASQ para teste funcional	Selenium	Selenium HQ	1.0.4
Gerenciamento de Projeto	PHP collab	Opensource CMS	2.5.1

7.3 Configurações do Ambiente de Teste

Devem ser fornecidas e suportadas as seguintes Configurações de Ambiente de Teste para este projeto.

Nome da Configuração	Descrição	Implementada na Configuração Física
Mínima configuração suportada	A mínima configuração necessária é ter instalado o navegador Google Chrome na versão 45 ou o navegador Mozilla Firefox na versão 40.	
Instalação de Rede (não cliente)		

8. Aprovação e Encerramento

Será realizado uma avaliação dos testes que serão realizados e das estratégias adotadas, para verificar se as verificações serão suficientes para garantir a qualidade do sistema. Essa avaliação será realizado pelo gerente do projeto junto com o analista de teste.