**Portfólio para Artesãos**

Documento de Teste de Software

Revisões

|  |  |  |
| --- | --- | --- |
| Data | Descrição | Autor / Responsabilidade |
| 25 de Fevereiro de 2011 | Inclusão da descrição (análise) dos testes | Thiago e Gabriel (analista de teste) |
| 25 de fevereiro de 2011 | Inclusão de testes no desenvolvimento do software | Miller Alves (Testador) |

Conteúdo

1 - Introdução 2

1.1 - Objetivos 2

1.2 - Público Alvo 2

1.3 - Organização do documento 3

2. - Recursos necessários para os testes 3

3. - Estratégias e técnicas de teste 3

4. - Casos de teste 3

5. - Aprovação Formal 4

Bibliografia 5

Anexos 5

# Introdução

## Objetivos

O objetivo deste documento é descrever o planejamento geral das atividades de teste do projeto Portfólio para artesãos, bem como critérios de aceitação dos artefatos a serem testados. A seguir, serão identificados os casos de teste a serem utilizados e descritas as estratégias a serem utilizadas.

## Público Alvo

Esse documento se destina principalmente aqueles relacionados às fases de requisitos, desenvolvimento, gerenciamento da qualidade e testes.

Os testes estão presentes dentro do desenvolvimento da *Sprint* dentro do processo, portanto, o presente documento é indicado a todos os envolvidos no ciclo de vida do software.

## Organização do documento

Este documento foi organizado em três sessões. A primeira se trata da introdução, contendo uma breve descrição do documento. A segunda sessão descreve recursos necessários para os testes. A terceira sessão descreve as estratégias e técnicas de teste.

# Recursos necessários para os testes

Os recursos necessários para a execução dos testes, incluem:

* Hardware – Um computador com configuração igual ou superior a: memória RAM de 2GB, HD com 160 GB livres, Core 2 Duo 2.0, Placa de Rede Ethernet 10/100.
* Software de apoio para a execução dos testes (ex.: sistema operacional, SGBD, ferramentas para geração ou carga de dados, ferramentas de teste);
* Software adicional a ser desenvolvido para os testes (ex.: *drivers*, *stubs*, geradores de arquivos de dados, programas de comparação de resultados esperados e obtidos);
* Recursos humanos - habilidades específicas do pessoal envolvido nos testes.

# Estratégias e estágios de teste

**Testes Unitários –** Ostestes de unidade de um código tem como objetivo detectar erros ou defeitos em certo nível de classe do sistema. Se trata de um teste Funcional, de Caixa Branca.

**Testes de Integração** – Depois de implementados, os módulos do sistema devem ser integrados. Os testes de integração consistem basicamente num conjunto dos testes unitários cumulativos construídos para testar especificamente o funcionamento de um componente individualmente, após a integração de componentes e/ou subsistemas. Nesta integração, podem ser empregadas ferramentas de automação. Assim como o Teste Unitário, este também é um teste de Caixa Branca.

**Testes de Sistema** - Os testes de sistema devem focar na iteração entre os componentes do sistema e, para isso, podem ser utilizadas ferramentas de automação de testes de interface e ferramentas de testes de unidade, que permitam a exploração e a iteração dos componentes nos trechos de códigos dependentes do funcionamento de vários componentes e subsistemas integrados. Os testes de sistema são tipicamente os da camada de fachada do sistema. Para os testes, cuja automação seja complexa, testes manuais podem ser bastante úteis para testar o software como um todo, inclusive com a vantagem de anteciparem possíveis problemas encontrados pelo cliente nos testes de aceitação. Este já se trata de um teste de Caixa Preta.

**Testes de Aceitação do Sistema** - Os testes de aceitação devem ser realizados pelos clientes na liberação de cada release, onde o cliente/usuário será convidado a interagir livremente pelo sistema na busca de erros. Esta iteração também favorecerá uma nova verificação junto aos requisitos do sistema. Apenas após certo grau de maturidade e dependendo das medidas da qualidade do sistema deve ser colocado testes de aceitação.

**Tipos de Teste**

## Teste de Carga

* Verificar se as informações do ator podem ser fornecidas e recuperadas.
* Verificar se o conteúdo e as categorias podem ser inseridos e exibidos.
* Verificar se os perfis do usuário e as informações de conta podem ser fornecidos e exibidos.
* Verificar se as informações de uso específicas do usuário são rastreadas.

## ****Teste Funcional - TF****

* Verificar se os usuários conseguem ver as informações sobre as quais solicitaram mensagens.
* Verificar se as mensagens são endereçadas aos usuários quando o conteúdo chega.
* Verificar se a inserção de conteúdo automático está funcionando.
* Verificar se a aprovação do editor provoca a inserção de conteúdo não-automático.
* Verificar se o conteúdo obsoleto foi excluído.
* Verificar se os relatórios são exportados devidamente em formato desejado do projeto.

## Teste de Configuração - TC

* Verificar a operação usando o Mozilla Firefox 39.0.3 ou superior, conforme Plano de Implantação.