

DoppelSoft

Oniria – Plano de Testes [v 1.0]

Grupo: Matheus “Feral” Luck Lucas

Histórico de Revisões

Data	Versão	Descrição	Autor
18/10/2010	v1.0	Versão inicial do Plano de testes	Matheus “Feral” Luck Lucas

Conteúdo

Histórico de Revisões	2
1. Introdução	4
1.1 Objetivos desse documento.....	4
1.2 O Oniria	4
1.3 Escopo.....	5
2. Requisitos a testar.....	5
2.1 Teste do Banco de Dados.....	5
2.2 Teste Funcional.....	6
2.3 Teste da Interface com o Usuário.....	6
2.4 Teste de Performance.....	6
2.5 Teste de Estresse	6
2.6 Teste de Volume.....	7
2.7 Teste de Falha/Recuperação.....	7
2.8 Teste de Instalação	7
2.9 Teste de Portabilidade.....	7
3. Estratégia de Teste	8
3.1 Tipos de Teste	8
3.1.1 Teste de Banco de Dados	8
3.1.2 Teste Funcional.....	8
3.1.3 Teste da Interface com o Usuário.....	8
3.1.4 Teste de Performance.....	9
3.1.5 Teste de Estresse	9
3.1.6 Teste de Volume	10
3.1.7 Teste de Falha/Recuperação	10
3.1.8 Teste de Instalação	11
3.1.9 Teste de Portabilidade	11
3.2 Ferramentas	12
4. Recursos Necessários	12
4.1 Recursos Humanos	12
4.2 Recursos de Sistema	12
5. Cronograma	13

1. Introdução

1.1 Objetivos desse documento

Os objetivos do Plano de Testes consistem em identificar e descrever os componentes do software a serem testados, bem como listar os Requisitos a Testar, recomendar e descrever as estratégias a serem empregadas em cada teste, identificar e estimar os recursos e o esforço necessários para a fase de testes e por fim listar os elementos resultantes do projeto de testes. Todas essas atividades visam prover maiores auxílio e organização para todo o processo de testes empregado no desenvolvimento do produto de software proposto.

1.2 O Oniria

O software Oniria consiste em uma ferramenta desenvolvida para servir ao mesmo tempo como sistema de catalogação e auxílio à criação de personagens e elencos para obras de ficção. O que antes seria um conjunto de anotações dispersas ou informações semi-conexas na mente do autor, pode através dessa ferramenta ser devidamente catalogado e descrito, ficando registrado para futura referência. O programa oferece a catalogação e descrição dos 4 tipos de entidades necessários: Autores, Obras, Elencos e Personagens, oferecendo a cada uma um conjunto de ferramentas de catalogação adequado. Para personagens, o sistema oferece a catalogação de informações específicas que vão desde descrição física a até perfil psicológico, bem como relacionamento/parentesco com outros personagens cadastrados. Há ainda a opção de anexar um repertório de imagens e textos referentes ao personagem. Para Elencos, é possível fazer uma descrição geral sobre o propósito do elenco, informações gerais, e tanto associar personagens a ele como associar o próprio elenco a uma ou mais obras. Estas, por sua vez, são catalogadas com uma descrição geral, gênero, recomendação de público, elencos participantes, autores e co-autores, bem como obras relacionadas. Por fim, os Autores são catalogados com uma lista de informações básicas, além de eventuais perfil e biografia. Tudo isso é estocado em um banco de dados interno.

A ferramenta Oniria provê ainda a geração de listas e relatórios de interações e relacionamentos, seja entre personagens, elencos, obras, autores ou outras combinações entre estes.

O software é disponível offline, mediante instalação no computador do usuário. Inicialmente, está sendo desenvolvido para a plataforma PC, sistema Microsoft Windows. Devido, porém, à escolha de linguagem e framework, está prevista a possibilidade da criação de versões portadas para outras plataformas.

1.3 Escopo

A fase de testes cobrirá testes de Unidade, Integração, Sistema e Aceitação. Testes de unidade lidarão com a corretude e performance de cada funcionalidade, enquanto que os testes de integração garantirão essas mesmas características ao conjunto. Testes de sistema garantirão estabilidade e desempenho à plataforma de software já montada, como um todo, enquanto testes de aceitação determinação se as características do software (entre usabilidade, funcionalidades, desempenho, confiabilidade entre outras) estão satisfatórias do ponto de vista do usuário e/ou cliente.

1.4 Identificação de Projeto

Documento	Criado ou Disponível	Recebido ou Revisado
Documento de Requisitos	Sim	Sim
Plano de Projeto	Sim	Sim
Modelo de Análise	Não	Não
Modelo de Projeto	Não	Não
Documento de Arquitetura	Não	Não
Protótipo	Não	Não
Manual do Usuário	Não	Não
Lista de Riscos	Sim (integrada ao Plano de Projeto)	Sim

2. Requisitos a testar

2.1 Teste do Banco de Dados

- Verificar se as informações dos autores podem ser inseridas, consultadas, alteradas e removidas;
- Verificar se as informações das obras podem ser inseridas, consultadas, alteradas e removidas;
- Verificar se as informações dos elencos podem ser inseridas, consultadas, alteradas e removidas;
- Verificar se as informações dos personagens podem ser inseridas, consultadas, alteradas e removidas;
- Verificar se as ligações entre as entidades podem ser criadas, alteradas e removidas;
- Verificar se o sistema de buscas ao banco de dados pode ser executado e retornar informações;

2.2 Teste Funcional

- Verificar se a alternância entre autores, elencos, e obras pode ser executada e revertida;
- Verificar se os relatórios de relacionamentos entre entidades possuem corretude;
- Verificar se todas as operações sobre as entidades podem ser realizadas;
- Verificar se a importação e exportação de dados é executada normalmente;

2.3 Teste da Interface com o Usuário

- Verificar se a interface gráfica é consistente e íntegra;
- Verificar se todos os elementos da interface podem ser acessados e correspondem a suas funções;
- Verificar se a disposição dos elementos permite conforto visual e fácil manuseio;
- Verificar os elementos visuais são identificáveis (i.e se os ícones estão identificáveis e as fontes e rótulos estão legíveis);
- Verificar se não há elementos visuais conflitantes durante a operação de menus e janelas;
- Verificar a ortografia e gramática de todos os textos da interface;

2.4 Teste de Performance

- Verificar o tempo de resposta na gravação de dados no banco de dados, para cada operação com cada entidade;
- Verificar o tempo de resposta nas buscas de relacionamentos para a geração de relatórios;
- Verificar a expansão de espaço em disco de acordo com a ocupação do banco de dados;
- Verificar o peso total do sistema, carregado com diferentes volumes de dados;

2.5 Teste de Estresse

- Verificar estabilidade, confiabilidade e resposta do sistema, durante e após operação intensa e prolongada, em uma máquina de desempenho mediano;
- Verificar a operação simultânea da ferramenta, utilizada de forma intensa, em conjunto com outros softwares paralelamente em um mesmo ambiente de trabalho (i.e editores de texto, editores de imagem etc);
- Executar novamente os testes em um netbook, de desempenho limitado;

2.6 Teste de Volume

- Verificar confiabilidade do sistema ao trabalhar com banco de dados povoado por um grande volume de registros e informações;
- Verificar o comportamento do sistema de associação de arquivos externos ao manipular um grande volume de arquivos simultaneamente;

2.7 Teste de Falha/Recuperação

- Forçar falhas através de operações não permitidas, e verificar o comportamento do sistema;
- Forçar o fechamento irregular do sistema e verificar a integridade de suas informações após a falha;
- Verificar como o sistema reage à corrupção parcial de alguma entrada em seu banco de dados;

2.8 Teste de Instalação

- Verificar se a instalação e desinstalação do software ocorre normalmente;
- Verificar se há variações no comportamento da instalação/desinstalação para um banco de dados muito povoado;

2.9 Teste de Portabilidade

- Verificar se o sistema possui funcionamento e comportamento uniformes nas suas versões portadas para outras plataformas;

3. Estratégia de Teste

3.1 Tipos de Teste

3.1.1 Teste de Banco de Dados

Objetivo do Teste	Verificar o comportamento correto e estável do banco de dados ao se manipularem informações.
Técnica	<ul style="list-style-type: none"> • Chamar e executar todos os métodos e processos de acesso ao banco de dados, em inserções, modificações e exclusões, de dados válidos e inválidos; • Verificar se a inserção e modificação de dados válidos ocorrem normalmente; • Verificar se a inserção de dados inválidos gera as mensagens de erro adequadas para o usuário, e o dado não é efetivamente inserido; • Verificar se todas as operações e eventos no banco de dados ocorrem sem gerar falhas ou conflitos;
Critério de Finalização	Todas as operações com o banco de dados ocorrem da forma esperada, de maneira consistente e sem falhas.
Considerações Especiais	<ul style="list-style-type: none"> • O teste utiliza uma ferramenta SGBD para manipulação dos dados, buscas e operações de forma direta; • As invocações dos testes são feitas de forma manual;

3.1.2 Teste Funcional

Objetivo do Teste	Garantir que cada Caso de Uso pré-determinado pode ser executado normalmente, com comportamento dentro do esperado.
Técnica	<ul style="list-style-type: none"> • Executar cada Caso de Uso, percorrendo seus fluxos de forma detalhada e ao mesmo tempo forçando bom e mau-uso da ferramenta, para cobrir diferentes operações de cada funcionalidade;
Critério de Finalização	Todos os Casos de Uso foram tratados, e o sistema funciona de forma correta e dentro do esperado em todos eles.
Considerações Especiais	Nenhuma.

3.1.3 Teste da Interface com o Usuário

Objetivo do Teste	Verificar se a apresentação visual e aparente ao usuário pode ser visualizada e utilizada da forma correta, gerando bom manuseio e garantindo que as funcionalidades possam ser aproveitadas pelo usuário.
Técnica	<ul style="list-style-type: none"> • Testar a interface do software com vários usuários diferentes, com

	<p>diferentes níveis de familiaridade com ferramentas do gênero, monitorando a navegabilidade, usabilidade das funções e visibilidade geral;</p> <ul style="list-style-type: none"> • Testar cada componente, menu e janela para verificar consistência visual e corretude de exibição;
Critério de Finalização	Todos os elementos da interface possuem usabilidade consistente e com nível de aceitação satisfatório para uma maioria aceitável de usuários.
Considerações Especiais	Nenhuma.

3.1.4 Teste de Performance

Objetivo do Teste	Verificar o desempenho do sistema, tanto do ponto de vista do tempo de resposta com as operações quando no consumo geral de recursos da plataforma onde está instalado o software.
Técnica	<ul style="list-style-type: none"> • Rodar funcionalidades específicas do sistema, analisando cada tempo de resposta de acordo com cada tipo de solicitação; • Executar as mesmas funcionalidades sob circunstâncias diferentes de hardware e configurações distintas de software no momento da execução; • Verificar o comportamento simultâneo do sistema em conjunto com outros softwares;
Critério de Finalização	O desempenho de cada uma das funcionalidades do software é considerado satisfatório.
Considerações Especiais	<ul style="list-style-type: none"> • Deverá haver espaço suficiente em disco para todas as operações e registros (i.e não deverá haver gargalos de armazenamento); • A execução individual de funcionalidades deve ser primeiramente executada sem outros softwares em paralelo; • Testes de execução simultânea devem ser feitos em paralelo com softwares de comportamento uniformemente controlável (i.e ferramentas de <i>benchmarking</i> e <i>load testing</i> programáveis, testes de aplicações usuais em configurações padronizadas);

3.1.5 Teste de Estresse

Objetivo do Teste	Verificar a estabilidade e o bom-comportamento do sistema durante utilizações prolongadas e em ambientes anti-propícios à sua execução normal, mantendo usabilidade e integridade aceitáveis.
--------------------------	---

Técnica	<ul style="list-style-type: none"> • Executar grandes números de funções de forma contínua e gradualmente elevada, monitorando eventuais mudanças de comportamento do sistema durante os procedimentos; • Executar testes de funcionalidades em sistemas sobrecarregados em diferentes níveis, verificando o comportamento do sistema durante os procedimentos; • Executar os mesmos testes em uma plataforma de hardware mais limitada (i.e um netbook com baixo poder de processamento);
Critério de Finalização	O comportamento do software ao longo de todos os testes se mostra satisfatório; estável, sem ocorrência de falhas, com garantia de integridade dos dados e sem prejuízos perceptíveis à usabilidade.
Considerações Especiais	Nenhuma.

3.1.6 Teste de Volume

Objetivo do Teste	Verificar o funcionamento estável e correto do sistema enquanto opera com grandes volumes de dados.
Técnica	<ul style="list-style-type: none"> • Executar as funcionalidades do sistema que utilizem acesso a dados, forçando um grande volume de informação em iterações curtas; • Executar as funcionalidades do sistema enquanto o banco de dados comporta um grande volume de informações;
Critério de Finalização	O sistema se mantém estável, correto e com desempenho satisfatório em um número aceitável de situações tratadas.
Considerações Especiais	<ul style="list-style-type: none"> • O espaço livre em disco da máquina usada nos testes deve ser suficiente para comportar os dados estocados ao final da operação;

3.1.7 Teste de Falha/Recuperação

Objetivo do Teste	Verificar se o sistema é capaz de continuar operante após a ocorrência de uma falha durante sua execução.
Técnica	<ul style="list-style-type: none"> • Forçar falhas do sistema em diferentes funcionalidades, observando como o sistema se recupera e se há perda de informações ou prejuízo na operação do sistema;
Critério de Finalização	O sistema sobrevive a falhas de forma aceitável, evitando e/ou limitando perda de informações em níveis aceitáveis para cada tipo de operação.
Considerações Especiais	Nenhuma.

3.1.8 Teste de Instalação

Objetivo do Teste	Verificar se os processos de instalação, atualização e desinstalação do software ocorrem de forma adequada.
Técnica	<ul style="list-style-type: none"> • Instalar, executar e desinstalar o software, e monitorar o comportamento do processo; • Repetir o processo, dessa vez após atualização do sistema; • Reinstalar e desinstalar o software, verificando uniformidade do comportamento;
Critério de Finalização	Os processos avaliados ocorrem de forma esperada, sem falhas e sem incorretudes (i.e vestígios inesperados na desinstalação, corrupção parcial na instalação etc)
Considerações Especiais	<ul style="list-style-type: none"> • Os testes devem ser feitos em diferentes circunstâncias de configuração de hardware e/ou software.

3.1.9 Teste de Portabilidade

Objetivo do Teste	Verificar o pleno funcionamento da ferramenta em outras plataformas operacionais, através dos recursos de portabilidade do Framework utilizado (C++ e Qt).
Técnica	<ul style="list-style-type: none"> • Após executada cada versão portada, rodar as mesmas baterias de testes anteriores na plataforma nova, analisando se há um mesmo comportamento.
Critério de Finalização	O software funciona da mesma forma, estável e funcional, com nível de desempenho razoavelmente uniforme, em todas as plataformas portadas.
Considerações Especiais	<ul style="list-style-type: none"> • As primeiras plataformas listadas para testes de portabilidade são Linux, Web e Mobile. • O teste para Mac depende da aquisição de plataforma de testes adequada.

3.2 Ferramentas

A lista de ferramentas de utilizadas durante o teste será:

- Rational Unified Process (para gerenciamento de teste)
- JUnit e Rational Rose (Projeto de Teste)
- Microsoft Project (Gerenciamento de Projeto)
- MySQL Database Server (Ferramenta de SGBD)
- SuperPi (Benchmark sintético para simulação de carga mono-tarefa)
- Lavalysis Everest (software para benchmarking e monitoramento em stress-tests)
- PCMark2010 (Benchmark sintético para simulação de carga multi-propósito)
- Microsoft Office suite (suite de programa de escritório para simular ambiente de operação usual)
- Adobe Master Suite (suite de programas de design, para simular ambiente de operação profissional usual)

4. Recursos Necessários

4.1 Recursos Humanos

Como a equipe é composta por apenas um membro, a ele serão alocadas todas as funções agregadas aos testes (Testador, Analista, Analista de Automação, Arquiteto, Líder e Gerente).

4.2 Recursos de Sistema

Serão utilizadas duas estações de trabalho para testes iniciais, com as seguintes configurações de hardware e software:

- Estação 1:
 - Computador Desktop, padrão PC
 - Phenom II X4 955 @3.2GHz
 - 2GB DDR2-800
 - HDDs 1x320GB 7200RPM SATA-300, 1x 80GB 720RPM PATA-133, 1x200GB 7200RPM PATA-133-to-USB2.0
 - Geforce 8600GT
 - Microsoft Windows 7 Professional (32bit)
 - Microsoft Windows 7 Professional (64bit)
 - Microsoft Windows XP Professional (32bit)

- Estação 2:
 - Netbook, padrão PC
 - Atom N270 @ 1.6GHz
 - 1GB DDR2-667
 - HDD 160GB 5400RPM SATA
 - Intel GMA500
 - Microsoft Windows 7 Professional (32bit)
 - Microsoft Windows XP Home (32bit)
 - Mandriva Linux 2010 (32bit)

5. Cronograma

Fase	Iteração	Data	Descrição da atividade
<i>Concepção</i>	Preliminar	03/09/2010	Levantamento inicial da idéia
		05/09/2010	Ponderação de alternativas
		13/09/2010	Entrega do Plano de Projeto
<i>Elaboração</i>	#1	15/09/2010	Ponderação sobre os requisitos elicitados e o escopo do projeto
		17/09/2010	Entrevistas para levantamento de requisitos
		21/09/2010	Análise final sobre os requisitos
		04/10/2010	Entrega Documento de Requisitos
<i>Análise</i>	#2	06/10/2010	Levantamento inicial sobre Arquitetura do Sistema
		08/10/2010	Pesquisa e análise de alternativas
		13/10/2010	Início da segunda fase de análise de alternativas
	#3	15/10/2010	Revisão e verificação da documentação de análise
		16/10/2010	Fechamento da primeira etapa do Documento de A&P
<i>Construção</i>	#4	17/10/2010	Revisão da estrutura e análise de testes, Início da codificação
		18/10/2010	Entrega do Plano e Projeto de Testes de Sistema
		19/10/2010	Início da segunda etapa do Documento de A&P
		20/10/2010	Análise sobre os componentes preliminares
		21/10/2010	Avaliação do desenvolvimento atual
	#5	22/10/2010	Implementação de componentes intermediários
		23/10/2010	Revisão dos <i>assets</i> atuais
		24/10/2010	Aplicação dos <i>assets</i> de conteúdo
	#6	25/10/2010	Acompanhamento dos componentes atuais
		26/10/2010	Análise final sobre o documento de projeto
27/10/2010		Entrega do Documento de Análise e Projeto	
<i>Transição (Testes e Implantação)</i>	#7	28/10/2010	Análise sobre a versão atual do sistema
		29/10/2010	Análise de falhas e erros
		01/10/2010	Aplicação das atividades de testes
		03/10/2010	Aplicação de correções necessárias
	Final	05/10/2010	Testes de estabilidade e usabilidade finais
		07/10/2010	Análise final do aplicativo
		08/11/2010	Entrega da Versão final do Sistema