

Sistema Datachk



Plano de Testes

Versão <1.0>

Zusammenarbeit
Ideias
Colaborativas

2010

Histórico de Revisões

Data	Versão	Descrição	Autores
13/05/2010	1.0	Criação da primeira versão do Plano de Testes	Angelo Brito Rodolfo Ferraz Sofia Lima Tiago Nascimento



Conteúdo

1. Introdução	4
1.1. Objetivo	4
1.2. O Sistema Datachk	4
1.3. Escopo	4
1.4. Identificação de Projeto	4
2. Requisitos a Testar	5
2.1. Teste do Banco de Dados	5
2.2. Teste Funcional	5
2.3. Teste da Interface do Usuário	5
2.4. Perfil do Desempenho	5
2.5. Teste de Carga	5
2.6. Teste de Segurança e de Controle de Acesso	5
2.7. Teste de Falha e Recuperação	6
2.8. Teste de Instalação	6
3. Estratégia de Teste	7
3.1. Tipos de Teste	7
3.1.1. Teste de Integridade de Dados e do Banco de Dados	7
3.1.2. Teste de Função	7
3.1.3. Teste de Interface do Usuário	7
3.1.4. Teste de Desempenho e de Carga	8
3.1.5. Teste de Segurança e de Controle de Acesso	8
3.1.6. Teste de Instalação	9
3.2. Ferramentas	9
4. Recursos	10
4.1. Trabalhadores	10
4.2. Sistema	11
5. Cronograma	12



1. Introdução

1.1. Objetivo

Este documento tem como objetivo identificar e definir os requisitos a serem testados, bem como descrever estratégias de teste a serem executadas. Além disso, este documento traz um levantamento dos recursos necessários, provendo uma estimativa dos esforços a serem empregados durante a etapa de testes.

1.2. O Sistema Datachk

O objetivo do **Sistema Datachk** é automatizar os processos que hoje são feitos manualmente no CRC – Recife, organizando todo o fluxo de trabalho e definindo protocolos nas fases de transição entre processos. Além disso, o **Sistema Datachk** se propõe a emitir os relatórios necessários para a prestação de contas do CRC – Recife para o Governo Federal.

1.3. Escopo

Atualmente, na unidade CRC – Recife, não há qualquer controle automático sobre o recebimento e o estoque de equipamentos e dispositivos doados. Visando garantir uma melhor eficiência neste setor, o sistema permitirá o registro de todas as peças em estoque, identificando cada uma com um número serial único e uma etiqueta com este número em código de barras.

A saída dos computadores recondicionados também não é automatizada e, portanto, a logística tem pouco controle sobre quantos equipamentos entraram, quantos saíram, quantos foram descartados ou quantos computadores podem ser montados a partir das peças estocadas. Além do mais, toda a contabilidade para a emissão de relatórios de prestação de contas é manual. Assim, quando as peças possuírem um código de barras e um número de referência e estiverem cadastradas no sistema, será possível emitir os relatórios automaticamente, bem como ter um maior controle sobre a entrada de peças e a saída de computadores recondicionados.

Esse sistema deve funcionar em uma rede interna, de modo que as seguintes etapas, representadas por grupos de trabalho: pré-teste, estoque e montagem/manutenção; estarão interligadas e aptas a consultar e/ou modificar o sistema, de acordo com a necessidade.

1.4. Identificação de Projeto

Abaixo, estão listados os recursos e a documentação disponíveis para a confecção deste documento.

Documento	Criado ou Disponível	Recebido ou Revisado
Plano de Projeto	<input checked="" type="checkbox"/> Sim <input type="checkbox"/> Não	<input checked="" type="checkbox"/> Sim <input type="checkbox"/> Não
Documento de Requisitos	<input checked="" type="checkbox"/> Sim <input type="checkbox"/> Não	<input checked="" type="checkbox"/> Sim <input type="checkbox"/> Não
Plano de Testes	<input type="checkbox"/> Sim <input checked="" type="checkbox"/> Não	<input type="checkbox"/> Sim <input checked="" type="checkbox"/> Não
Projeto de Testes	<input type="checkbox"/> Sim <input checked="" type="checkbox"/> Não	<input type="checkbox"/> Sim <input checked="" type="checkbox"/> Não
Documento de Análise e Projeto	<input type="checkbox"/> Sim <input checked="" type="checkbox"/> Não	<input type="checkbox"/> Sim <input checked="" type="checkbox"/> Não
Protótipo	<input type="checkbox"/> Sim <input checked="" type="checkbox"/> Não	<input type="checkbox"/> Sim <input checked="" type="checkbox"/> Não



2. Requisitos a Testar

Abaixo, estão listados os componentes do sistema considerados críticos e que, portanto, devem ser testados.

2.1. Teste do Banco de Dados

- Verificar se o sistema está apto a cadastrar ou remover equipamentos, além de alterar informações dos mesmos;
- Verificar se o sistema está apto a cadastrar ou remover computadores, além de trocar equipamentos dos mesmos;
- Verificar se o sistema está apto a cadastrar ou remover usuários, além de alterar informações dos mesmos;
- Verificar se o sistema faz buscas por códigos de barras de equipamentos ou de computadores;
- Verificar se o sistema busca pelo CPF dos usuários;
- Verificar se as informações mostradas pelo sistema correspondem à busca efetuada;
- Verificar se o sistema distingue equipamentos que foram para o descarte, dos que ficaram no estoque ou na sucata;
- Verificar se o sistema guarda a hora do sistema e o nome do usuário para todas as operações realizadas;
- Verificar se os dados do sistema estão sempre atualizados (se há consistência).

2.2. Teste Funcional

- Verificar o acesso dos usuários através de login e senha;
- Verificar se cada usuário pode efetuar as operações, através do sistema, a ele atribuídas;
- Verificar se os relatórios estão sendo emitidos no formato correto.

2.3. Teste da Interface do Usuário

- Verificar se não há redundância de acesso a um mesmo formulário (se não há duas formas de chegar ao mesmo formulário);
- Verificar se os menus são o mais simplificado possível;
- Verificar se a disposição dos dados na tela tem boa visibilidade;
- Verificar a gramática e a ortografia do português apresentado na interface gráfica.

2.4. Perfil do Desempenho

- Verificar o tempo de resposta do Servidor Web para operações de cadastro, remoção atualização e busca;
- Verificar o tempo de processamento da leitura do código de barras dos equipamentos e computadores.

2.5. Teste de Carga

- Verificar a resposta do Servidor Web para o número máximo de requisições suportado.

2.6. Teste de Segurança e de Controle de Acesso

- Verificar se cada usuário apenas tem acesso às informações a ele necessárias;



- Verificar se apenas o administrador do sistema tem acesso às operações de cadastro, de busca, de remoção e de atualização dos usuários;
- Verificar se apenas o gerente de estoque tem acesso a buscas mais elaboradas, bem como à emissão de relatórios;
- Verificar se apenas o gerente e o operador de estoque têm acesso ao cadastro, à busca por código de barras, à remoção e à atualização de equipamentos e computadores;
- Verificar se o usuário da montagem/manutenção apenas tem acesso à busca de equipamento pelas características;
- Verificar que todas as operações apenas podem ser efetuadas a partir da rede interna do CRC.

2.7. Teste de Falha e Recuperação

- Verificar se o sistema consegue manter a consistência dos dados, caso o servidor fique *off-line* de maneira inesperada;
- Verificar se o sistema consegue manter a consistência dos dados, caso o terminal de onde está partindo a requisição fique *off-line* de maneira inesperada.

2.8. Teste de Instalação

- Verificar se as configurações do servidor são suficientes para a instalação do sistema;
- Verificar se o sistema foi devidamente instalado no servidor principal;
- Verificar se os terminais conseguem acessar o sistema através do *browser* e se este funciona da forma esperada.



3. Estratégia de Teste

3.1. Tipos de Teste

3.1.1. Teste de Integridade de Dados e do Banco de Dados

Objetivo	Verificar se as rotinas de acesso ao Banco de Dados funcionam da maneira esperada e se a consistência dos dados é mantida.
Técnica	<ul style="list-style-type: none"> • Chamar cada uma das rotinas de acesso ao Banco de Dados com entradas válidas e inválidas; • Verificar se cada chamada de rotina, em conjunto com os dados de entrada, gerou uma resposta esperada no Banco de Dados.
Critério de Finalização	Todas as rotinas de acesso ao Banco de Dados funcionam da maneira esperada e os dados não são corrompidos.
Considerações Especiais	<ul style="list-style-type: none"> • São necessários <i>drivers</i> para simular o envio de dados da interface gráfica para o Banco de Dados; • Pequenas amostras de dados devem ser utilizadas, para facilitar a identificação de resultados não-esperados.

3.1.2. Teste de Função

Objetivo	Verificar as funcionalidades do sistema, desde a autenticação de usuário, até a emissão de relatórios.
Técnica	<p>Executar os fluxos de dados descritos nos Casos de Uso com dados válidos e inválidos para observar o seguinte:</p> <ul style="list-style-type: none"> • Quando dados válidos são inseridos, resultados esperados são observados; • Quando dados inválidos são inseridos, mensagens de erro são observadas.
Critério de Finalização	Todos os testes foram executados com sucesso e os eventuais erros foram corrigidos.
Considerações Especiais	Nenhuma.

3.1.3. Teste de Interface do Usuário

Objetivo	Verificar se a navegação é fácil e intuitiva, incluindo a verificação do uso de métodos de acesso e aparência da página (posicionamento, tamanho da fonte, organização dos menus, etc).
Técnica	<ul style="list-style-type: none"> • Com base nos Casos de Uso, passar por todos os fluxos de dados possíveis, alcançando todos os caminhos possíveis para verificar a facilidade de navegação e a organização dos menus; • Testar os métodos de acesso mais usados (<i>mouse</i>, tecla <i>tab</i>, tecla <i>enter</i>, etc); • Verificar se o tamanho da fonte e o posicionamento dos elementos da página são agradáveis aos olhos e, ao mesmo tempo, permitem que a maior parte da



	informação seja mostrada em tela com um uso mínimo da barra de <i>scroll</i> .
Critério de Finalização	Todo sistema foi varrido com base nos Casos de Uso e passou nos testes acima descritos.
Considerações Especiais	O teste de interface com o usuário é bastante subjetivo e, portanto, mais passível a falhas no que diz respeito ao que o cliente espera e o que será oferecido a ele.

3.1.4. Teste de Desempenho e de Carga

Objetivo	Verificar o desempenho do sistema nas seguintes situações:
Técnica	<ul style="list-style-type: none"> • Carga normal de trabalho; • Carga de trabalho no pior caso. • Gerar um número de requisições ao servidor considerado normal; • Gerar um número de requisições ao servidor considerado crítico.
Critério de Finalização	Finalização das requisições com sucesso, ou seja, sem nenhuma falha do servidor e com um tempo de resposta aceitável.
Considerações Especiais	<ul style="list-style-type: none"> • Para simular um número grande de requisições, são usadas ferramentas de Emulação de Terminal Remoto (RTE), que também podem ser usadas para carregar uma rede com “tráfego”; • Uma segunda etapa de testes de desempenho consiste em uma simulação de um número grande de requisições num ambiente mais real, com múltiplos terminais físicos tentando acessar o servidor ao mesmo tempo; • Para uma melhor mensuração, os testes de desempenho devem ser rodados em máquinas dedicadas; • O tamanho das bases de dados utilizadas nesses testes deve estar muito próximo do tamanho real.

3.1.5. Teste de Segurança e de Controle de Acesso

Objetivo	Verificar de apenas usuários autenticados podem fazer modificações no sistema e se a permissão de acesso de acordo com o tipo de usuário é respeitada.
Técnica	Criar casos de teste (incluindo todas as operações permitidas) para cada tipo de usuário e aplicá-los tanto ao usuário para os quais foram criados, quanto aos demais tipos de usuários.
Critério de Finalização	O teste acaba quando todos os casos de teste criados para um tipo específico de usuário forem bem-sucedidos, mas falharem ao serem aplicados aos demais tipos de usuários, desde que não haja uma intersecção entre as operações permitidas por um tipo de usuário e o outro.
Considerações Especiais	O teste de acesso ao sistema por usuários não



	autenticados deve ser anteriormente discutido com o administrador de redes, visto que esse teste pode não ser necessário, se já houver uma função para tal requisito na própria rede.
--	---

3.1.6. Teste de Instalação

Objetivo	<p>Verificar se o sistema é instalado com sucesso nos seguintes casos:</p> <ul style="list-style-type: none"> • O servidor não possui o Sistema Datachk instalado; • O servidor já possui a mesma versão do Sistema Datachk que se está tentando instalar; • O servidor possui uma versão mais antiga do Sistema Datachk.
Técnica	<p>Criar um <i>script</i> de instalação do sistema e rodá-lo para os casos acima descritos. Verificar, posteriormente, o funcionamento do sistema através da execução de operações específicas.</p>
Critério de Finalização	<p>O sistema é instalado sem falhas nos três casos.</p>
Considerações Especiais	<p>Deve-se tomar cuidado ao escolher as operações que devem ser rodadas para que o sistema seja testado de forma confiável.</p>

3.2. Ferramentas

As ferramentas listadas abaixo serão utilizadas para facilitar a execução dos testes e o *report* de erros:

Etapa	Ferramenta	Vendedor
Gerenciamento de Teste	Rational RequisitePro	IBM
Projeto de Teste	Rational Rose	IBM
Gerenciamento de Projeto	Microsoft Project	Microsoft
Ferramenta de Administração do Sistema de Gerenciamento de Banco de Dados	phpMyAdmin	Free



4. Recursos

Esta seção apresenta os recursos necessários para a etapa de testes do **Sistema Datachk**.

4.1. Trabalhadores

Os recursos humanos necessários estão listados abaixo:

Recursos Humanos		
Categoria de Trabalhador	Recursos Mínimos Recomendados	Responsabilidades Específicas ou Comentários
Gerente de Testes	Rodolfo Ferraz	Gerencia as atividades, prevenindo riscos e tomando decisões estratégicas. Responsabilidades: <ul style="list-style-type: none"> • Guiar a equipe; • Adquirir recursos apropriados; • Fornecer relatórios de gerenciamento.
<i>Designer</i> de Testes	Sofia Lima	Identifica e implementa os casos de teste. Responsabilidades: <ul style="list-style-type: none"> • Analisar o contexto; • Gerar plano de testes; • Avaliar o resultado do esforço aplicado na etapa de testes.
Testador	Tiago Nascimento	Executa os testes. Responsabilidades: <ul style="list-style-type: none"> • Registrar os resultados; • Reportar os eventuais erros; • Documentar as solicitações de mudança.
Administrador do Sistema de Testes	Angelo Brito	Gerencia e mantém as ferramentas do ambiente de testes. Responsabilidades: <ul style="list-style-type: none"> • Instalar as ferramentas necessárias para a etapa de testes; • Administrar e manter o ambiente de testes; • Controlar o acesso dos trabalhadores ao ambiente de testes.
Administrador do Banco de	Tiago Nascimento	Garante a gerência e manutenção dos



Dados		bens de teste. Responsabilidades: <ul style="list-style-type: none"> • Criar as bases de dados (bens de teste) necessárias para a execução dos testes; • Administrar os dados dos bens de teste.
<i>Designer</i>	Sofia Lima	Identifica e define as classes de teste. Responsabilidades: <ul style="list-style-type: none"> • Definir os pacotes de teste; • Definir as classes de teste; • Definir as operações, atributos e associações das classes de teste.
Desenvolvedores	Angelo Brito, Rodolfo Ferraz	Implementa módulos do sistema e executa os testes unitários correspondentes. Responsabilidades: <ul style="list-style-type: none"> • Cria as classes e pacotes de teste anteriormente definidos; • Implementa módulos do sistema; • Roda testes unitários sobre os módulos do sistema implementados.

4.2. Sistema

A tabela a seguir apresenta os recursos de *hardware* e *software* disponíveis para o Projeto de Testes:

Recursos do Sistema
Servidor de Banco de Dados MySQL DataBase Server
Terminais Clientes 3 Notebooks conectados via WLAN (ou LAN) e à Internet
Repositório de Testes Repositório SVN hospedado no Google Code 3 Notebooks para o Desenvolvimento de Testes



5. Cronograma

Milestone	Data de início	Data de término
Planejar Teste	13/05/2010	16/05/2010
Projetar Teste	17/05/2010	19/05/2010
Implementar Teste	26/05/2010	03/06/2010
Executar Teste	14/06/2010	18/06/2010
Avaliar Teste	21/06/2010	22/06/2010

