

Processo Seletivo

Plano de Teste

Versão 1.1

18/07/2016

Equipe:

Bruno Messias

João Gabriel

Jean Marcos

Michel Berigo

Tiago Damascena

Histórico de Revisões

Data	Versã o	Descrição	Autor
07/07/2016	1.0	Versão inicial	Jean Marcos, Bruno dos Santos, João Gabriel
18/07/2016	1.1	Estratégia de testes	Jean Marcos, Bruno dos Santos, João Gabriel

1. INTRODUÇÃO

1.1 Objetivo do Documento

Este documento tem por objetivo definir o planejamento e padrões necessários para a realização de atividades relacionadas ao teste do sistema ProcessoSeletivo.

Portanto, o restante do documento tem os objetivos de compreender o projeto, identificar os componentes necessários para a realização dos testes, listar os requisitos e casos de testes e definir as estratégias de teste.

1.2 Compreensão do Sistema

O Sistema ProcessoSeletivo é um sistema que gerencia a seleção, recrutamento e efetivação de recursos humanos para uma organização

Os interessados nas vagas disponíveis deverão acessar o sistema através de algum navegador de internet e se candidatar às vagas disponíveis, posteriormente alguns candidatos serão convocados para entrevistas e fases adicionais de seleção, à critério da organização.

O sistema validará a maioria das informações fornecidas pelos candidatos no exato momento do preenchimento (por exemplo, formato do nome, CPF, data de nascimento), informando-os de possíveis erros e solicitando prontamente a correção.

Os responsáveis pela seleção usarão o sistema para gerar uma lista com os candidatos que preenchem os requisitos e suas informações e meios de contato.

É importante salientar que o software ainda está em desenvolvimento

2. REQUISITOS DO TESTE

A lista a seguir identifica os itens (casos de uso, requisitos funcionais, requisitos não funcionais) que foram identificados como alvos do teste. Essa lista representa o que será testado. Detalhes sobre cada teste serão determinados posteriormente à medida que os Casos de Teste forem identificados e os Scripts de Teste forem desenvolvidos.

2.1 Teste Funcional

Os objetivos do teste funcional são:

- ❖ Verificar se o sistema permite a inclusão bem sucedida de cadastros com informações corretas preenchidas.
- ❖ Verificar se o sistema valida os campos preenchidos, informa os usuários de informações incorretas e impede a inclusão do cadastro antes que a correção ocorra.
- ❖ Verificar se o sistema é capaz de gerar corretamente uma lista com todos os candidatos que preencham os requisitos necessários à convocação para entrevista. Teste da Interface do Usuário
- ❖ Verificar se o sistema exige autenticação do usuário antes de permitir acesso às informações de cadastro armazenadas e, depois de autenticado, se o sistema só exibe as informações adequadas para aquele usuário.

Esses objetivos deverão ser atingidos utilizando as seguintes técnicas de teste funcionais:

2.1.1 Teste de Integração

Verificar possíveis falhas associadas às interfaces entre os módulos quando esses são integrados para construir a estrutura do software que foi estabelecida.

2.1.2 Teste de Regressão

Verificar se a cada nova versão do software ou a cada ciclo, todos os testes que já foram aplicados nas versões ou ciclos de teste anteriores do sistema resultam em sucesso.

2.1.3 Teste de Aceitação

Verificar se o produto atende as expectativas do usuário e foi implementado de acordo com suas prévias especificações

2.2 Teste Estrutural

O objetivo do teste estrutural é:

- ❖ Verificar o comportamento interno do componente de software diretamente sobre o código fonte para garantir que os software seja estruturalmente sólido se que funcionem no contexto técnico onde serão instalados

Esse objetivo deverá ser atingido utilizando as seguintes técnicas de teste funcionais:

2.2.1 Teste Unitário

Verificar se cada unidade do código funciona da forma desejada de acordo com as entradas inseridas.

Examinar o comportamento do código se determinado parâmetro for passado (ou não), o que ele retorna se determinada condição for verdadeira, os efeitos colaterais que ele causa durante a sua execução e se determinada exceção é lançada

2.2.2 Teste de Performance

Verificar se o sistema informa o usuário sobre o preenchimento incorreto de um campo e até 1 segundo após o usuário ter tirado o foco daquele campo.

Verificar se, sob condições ideais de rede, se o tempo de resposta do servidor para qualquer uma das páginas do processo de cadastro não ultrapassa 5 segundos.

2.2.3 Teste de Carga

Verificar a resposta do sistema com 10 usuários simultâneos.
--

Verificar a resposta do sistema com 50 usuários simultâneos.
--

Verificar a resposta do sistema com 100 usuários simultâneos.

Verificar a resposta do sistema com 200 usuários simultâneos.

3. Estratégia de Teste

Os tipos de teste são descritos abaixo. Foram adicionados testes para características ainda não presentes no sistema, como banco de dados e interface de usuário, pois espera-se que elas sejam incluídas posteriormente na implementação do sistema.

3.1 Testes Funcionais

O teste funcional tem o objetivo de garantir que os requisitos e as especificações do sistema tenham sido atendidos.

3.1.1 Teste de Integração

Objetivo do Teste:	Os módulos do sistema, depois de implementados, devem ser integrados. E para verificar sua correção após a integração é necessário realizar os testes de integração. Nesse tipo de teste é observada a integração de cada componente do sistema de acordo com o que foi especificado em sua arquitetura.
Técnica:	Caixa preta
Critério de Finalização:	Após a execução de todos os testes de integração, os componentes recém integrados apresentaram o comportamento esperado sem falhas
Considerações Especiais:	Será verificada a forma como os componentes do sistema estão integrados, de acordo com o que foi proposto na arquitetura do sistema.

3.1.2 Teste de Regressão

Objetivo do Teste:	O teste de regressão tem como objetivo verificar se uma nova build ou release ou versão, realiza com sucesso os testes de versões anteriores prevenindo efeitos colaterais de alterações.
Técnica:	Caixa preta
Critério de Finalização:	Todo o plano de teste da versão imediatamente anterior é realizado com sucesso.
Considerações Especiais:	Os testes a serem realizados devem ser os da versão imediatamente anterior, se houver. Isso garante uma cobertura maior pois na versão anterior o critério também foi esse, assim, até a primeira versão.

3.1.3 Teste de Aceitação

Objetivo do Teste:	O teste de aceitação tem como objetivo verificar se uma nova release ou versão, representa corretamente as expectativas do usuário e suas prévias especificações
Técnica:	Caixa preta
Critério de Finalização:	O cliente valida todas as funcionalidades da versão
Considerações Especiais:	Os testes devem ser realizados constantemente, garantindo a diminuição da refatoração do software.

3.2 Testes Estruturais

3.2.1 Teste Unitário

Objetivo do Teste:	O teste de unidade de código tem como objetivo detectar erros ou defeitos num nível de classe do sistema.
Técnica:	Caixa branca
Critério de Finalização:	Resultados dos testes unitários finalizados sem erros ou alertas.
Considerações Especiais:	Para cada classe, será implementada uma classe de teste contendo os métodos de teste equivalentes.

3.2.2 Teste de Performance

Objetivo do Teste:	<p>Verificar que os comportamentos de performance para as transações designadas ou funções de negócio sob as seguintes condições:</p> <ul style="list-style-type: none">▪ Carga de trabalho normal▪ Carga de trabalho no pior caso
Técnica:	<ul style="list-style-type: none">▪ Usar Procedimentos de Teste desenvolvidos para Teste da Função ou Ciclo de Negócio▪ Modificar os arquivos de dados para aumentar o número de transações ou os scripts para aumentar o número de iterações que ocorre a cada transação.▪ Scripts devem ser rodados em uma máquina (melhor caso para comparar um único usuário, uma única transação) e ser repetidas com múltiplos clientes (virtual ou real, ver Considerações Especiais abaixo).

Critério de Finalização:	<p>Único usuário ou transação: finalização com sucesso dos scripts de testes sem nenhuma falha e dentro da alocação de tempo por transação esperada ou requisitada.</p> <p>Múltiplas transações ou usuários: finalização bem sucedida dos scripts de teste sem qualquer falha e dentro da alocação de tempo aceitável.</p>
Considerações Especiais:	<p>Um teste abrangente de performance inclui ter uma carga de trabalho no servidor.</p> <p>O teste de performance deve ser executado em uma máquina dedicada ou em um tempo dedicado. Isso permite controle total e mensuração precisa.</p>

3.2.3 Teste de Carga

Objetivo do Teste:	Verifique o tempo de resposta para as transações designadas ou casos de negócio sob condições variantes de carga de trabalho.
Técnica:	<ul style="list-style-type: none"> ▪ Use testes desenvolvidos para o Teste do Ciclo de Negócio ou Função. ▪ Modifique os arquivos de dados para aumentar o número de transações ou os testes para aumentar o número de vezes que cada transação ocorre.
Critério de Finalização:	Múltiplas transações ou usuários: finalização bem sucedida dos testes sem qualquer falha e dentro da alocação de tempo aceitável.
Considerações Especiais:	<ul style="list-style-type: none"> ▪ O teste de carga deve ser executado em uma máquina dedicada ou em um tempo dedicado. Isso permite o controle total a e mensuração precisa. ▪ As bases de dados usadas para os testes de carga devem ou ser do tamanho real ou igualmente dimensionadas.

4. Casos de Teste

Esta seção apresenta os casos de teste que foram gerados para os casos de uso atuais existentes para o sistema. Conforme os sistema evolui e novas necessidade surgem alguns casos de teste podem

precisar de alterações, novos casos de teste podem ser necessários ou alguns casos de teste podem tornar-se desnecessários.

A tabela abaixo descreve os casos de uso implementados a partir das técnicas de particionamento de equivalência e análise de valor limite para a execução funcional do teste de cada caso de uso.

4.1 CSU01 - Publicar Vagas

Casos de testes relativos a execução do caso de uso Publicar Vagas.

[CSU01 - Publicar Vagas](#)

4.2 CSU02 - Realizar Inscrição

Casos de testes relativos a execução do caso de uso Realizar Inscrição.

[CSU02 - Realizar Inscrição](#)

4.3 CSU03 - Agendar Entrevista

Casos de testes relativos a execução do caso de uso Agendar Entrevista.

[CSU03 - Agendar Entrevista](#)

4.4 CSU04 - Entrevistar Candidato

Casos de testes relativos a execução do caso de uso Entrevistar Candidato.

[CSU04 - Entrevistar Candidato](#)

4.5 CSU05 - Avaliar Candidatos

Casos de testes relativos a execução do caso de uso Avaliar Candidatos.

[CSU05 - Avaliar Candidatos](#)

4.6 CSU06 - Reportar Resultado

Casos de testes relativos a execução do caso de uso Reportar Resultado.

4.7 CSU07 - Logar no Sistema

Casos de testes relativos a execução do caso de uso Logar no Sistema.

[CSU07 - Logar no Sistema](#)

4.8 CSU08 - Manter Usuários

Casos de testes relativos a execução do caso de uso Manter Usuários.

[CSU08 - Manter Usuários](#)

4.9 CSU09 - Enviar E-mail

Casos de testes relativos a execução do caso de uso Enviar E-mail.

[CSU09 - Enviar E-mail](#)

5. Recursos

Essa seção apresenta os recursos recomendados para o projeto do sistema Processo Seletivo, suas principais responsabilidades, e seus conhecimentos ou conjunto de habilidades.

5.1 Papéis

A tabela abaixo descreve os papéis necessários para a execução e condução dos testes para o sistema. Um integrante do projeto pode desempenhar mais um papel, além disso, determinados papéis podem ser compartilhados, variando de acordo com a necessidade do projeto e a disponibilidade da equipe.

Recursos Humanos	
Papel	Responsabilidades Específicas ou Comentários
Gerente de Teste,	Supervisiona os testes, delega tarefas e aprova prazo. Responsabilidades: <ul style="list-style-type: none">• Prover direcionamento técnico• Adquirir recursos apropriados• Fornecer relatórios de gerenciamento

	<ul style="list-style-type: none"> • Manter plano de testes atualizado • Assegurar a execução das tarefas conforme o plano
Testador	<p>Executa os testes.</p> <p>Responsabilidades:</p> <ul style="list-style-type: none"> • Executar os testes • Registrar os resultados • Documentar solicitações de mudança
Administrador do Sistema de Teste	<p>Garante que o ambiente e os bens de teste sejam gerenciados e mantidos.</p> <p>Responsabilidades:</p> <ul style="list-style-type: none"> • Administrar o sistema de gerenciamento teste • Instalar e gerenciar o acesso do trabalhador ao sistema de testes
Gerente de infraestrutura	<p>Garante que o ambiente e bens de teste de dados (banco de dados) sejam gerenciados e mantidos.</p> <p>Responsabilidades:</p> <ul style="list-style-type: none"> • Manter a base de dados • Efetuar testes na camada de persistência • Assegurar que as inconsistências são tratadas e possuem seus testes implementados
Desenvolvedor	<p>Implementa e faz os testes unitários das classes e pacotes de teste.</p> <p>Responsabilidades:</p> <ul style="list-style-type: none"> • Cria as classes e pacotes de teste implementados nos casos de testes

5.2 Sistema

A tabela seguinte expõe os recursos do sistema para o projeto de teste.

Recursos do Sistema
<p>Servidor de Banco de Dados</p> <ul style="list-style-type: none"> • MySQL DataBase Server
<p>Terminais Clientes</p> <ul style="list-style-type: none"> • Qualquer computador atenda às configurações da seção 2.2
<p>Repositório de Testes</p> <ul style="list-style-type: none"> • Repositório GitHub, no endereço: https://github.com/TiagoDamascena/VVS-Grupo09 • 3 PCs de Desenvolvimento de Teste

5.3 Ferramentas

As seguintes ferramentas serão empregadas para esse projeto:

	Ferramenta	Fornecedor
Gerenciamento de Teste	Travis CI	Travis
Ferramentas do SGBD	MySQL Control Center	MySQL

6. Cronograma

A tabela do cronograma de execução dos testes é apresentada abaixo, as datas utilizadas são fictícias e devem ser alteradas quando houver execução real dos testes.

Milestone	Data de Início	Data de Término
Planejar Teste	23/06/16	20/07/16
Projetar Teste	21/07/16	15/08/16
Implementar Teste	20/08/16	23/08/16
Executar Teste	24/08/16	27/08/16
Avaliar Teste	28/08/16	29/08/16