**Projeto Brazuka**

Plano de Testes

Versão 0.1

**Histórico de Revisão**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versão | Descrição | Autor |
| 02/04/2012 | 0.1 | Elaboração Inicial do Documento | Cyntia Dias |

**Sumário**

[1. Introdução 3](#_Toc277945009)

[1.1. Escopo 3](#_Toc277945010)

[1.2. Definições, Acrônimos, e Abreviações 3](#_Toc277945011)

[1.3. Visão Geral 3](#_Toc277945012)

[2. Estágios de Teste 3](#_Toc277945013)

[3. Tipos de Testes 3](#_Toc277945014)

[4. Detalhamento dos tipos de testes 3](#_Toc277945015)

[5. Alvos de Teste 3](#_Toc277945016)

[5.1. Requisitos Funcionais 3](#_Toc277945017)

[5.2. Requisitos Não Funcionais 3](#_Toc277945018)

[5.3. Requisitos Não Contemplados na Execução dos Testes 3](#_Toc277945019)

[6. Estratégia de Testes 3](#_Toc277945020)

[7. Recursos necessários 3](#_Toc277945021)

[7.1. Recursos Humanos 3](#_Toc277945022)

[7.2. Recursos Computacionais 3](#_Toc277945023)

[8. Riscos 3](#_Toc277945024)

[9. Produtos Gerados 3](#_Toc277945025)

[10. Referência 3](#_Toc277945026)

[11. Anexos 3](#_Toc277945027)

**Plano de Testes**

1. Introdução

Este documento relaciona os casos de uso a serem testados, os estágios de testes, método de qualificação, detalhamento dos tipos de testes, alvos de testes, a estratégia adotada para a execução dos testes, os recursos humanos necessários, bem como os produtos que serão gerados.

* 1. Escopo

[Uma breve descrição do escopo deste Plano de Teste; a que Projeto(s) ele está associado e tudo o mais que seja afetado ou influenciado por este documento.]

* 1. Definições, Acrônimos, e Abreviações

[Essa subseção fornece as definições de todos termos, acrônimos, e abreviações requeridas para a apropriada interpretação do Plano de Teste. Estas informações podem ser fornecidas pela referência ao Glossário do projeto.]

* 1. Visão Geral

[Esta subseção deve descrever o que o restante do Plano de Teste contém e explicar como o documento está organizado.]

1. Estágios de Teste

Definem o momento do ciclo de vida do software em que são realizados testes por pessoas diferentes daquelas que o programaram. Entretanto, considerando a divisão das tarefas de teste em quatro níveis relacionados ao escopo do software, estão previstos para o ***Projeto Brazuka*** os seguintes estágios de teste:

* **Teste Unitário:** Estágio mais baixo da escala de testes e são aplicados nos menores componentes de código criados, visando garantir que testes atendam as especificações, em termos de características e funcionalidade.
* **Teste de Integração:** São realizados para verificar basicamente se as unidades testadas de forma individual executam corretamente quando colocadas juntas, isto é, quando integradas.
* **Teste de Sistema:** Será revisando a execução do sistema, dentro de um ambiente operacional controlado, para validar a exatidão e perfeição na execução de suas funções.
* **Teste de Aceitação ou Homologação:** São os testes finais de execução do sistema, realizados pelos usuários, visando verificar se a solução atende aos objetivos do negócio e a seus requisitos, no que diz respeito à funcionalidade e usabilidade, antes da utilização no ambiente de produção.

1. Tipos de Testes

Seguem abaixo os tipos de testes a serem aplicados ao **Projeto Brazuka:**

* **Acessibilidade:** Verifica se a interface do usuário fornece o acesso apropriado às funções do sistema e a navegação adequada. Além disso, estes testes garantem que os objetos dentro da interface do usuário funcionem de acordo com os padrões definidos pelo cliente.
* **Carga:** Verifica o comportamento do sistema frente à variação de carga de trabalho diferente e grande quantidade de dados no sistema para determinar se os limites que podem causar a falha do software são alcançados.
* **Ciclo de negócio:** Verifica se o ciclo do caso de uso está de acordo, percorrendo todas as funcionalidades desde a entrada do dado até sua saída.
* **Configuração:** Verifica se o software está apto a rodar em diferentes versões ou configurações de ambientes (hardware e software), como, por exemplo, em diferentes browsers.
* **Controle de Segurança e Acesso:** Avalia o aplicativo quanto às restrições de acesso em duas principais áreas de segurança: segurança em nível de aplicação e em nível de sistema.
* **Disponibilidade:** Avaliam a capacidade do software em continuar operando mesmo quando algum elemento (software ou hardware) fica inoperante ou para de funcionar.
* **Estresse:** Verifica o comportamento do sistema durante condições limite ou fora da tolerância esperada.
* **Falha e Recuperação:** Assegura que o sistema pode, com sucesso, recuperar os dados após uma falha no funcionamento do hardware, do software ou de rede, quando existir perda dos dados ou da integridade dos mesmos.
* **Funcional:** Grupos de testes que avaliam se o que foi especificado foi implementado.
* **Instalação:** verifica que o sistema instalado em outra máquina, que não tenha sido utilizada anteriormente, funcione corretamente,
* **Integridade de dados:** Devem ser executados os métodos de acesso à base de dados da interface do usuário, de forma que seja possível observar e registrar o comportamento funcional incorreto ou a corrupção de dados.
* **Performance:** Mede e avalia o tempo de resposta de cada transação dos requisitos sensíveis ao tempo.
* **Usabilidade:** Verificam o nível de facilidade de uso do software pelos usuários.
* **Fumaça:** Verifica a capacidade do sistema de resistir a situações não previstas nos requisitos e especificações.
* **Regressão:** Verifica a ocorrência de novos defeitos após a resolução de defeitos.

1. Detalhamento dos tipos de testes

[Devem ser detalhados somente os tipos de testes a serem aplicados no projeto no nível do teste em questão. Deve ser incluído neste detalhamento informações específicas relacionadas ao projeto e a sua execução, podendo ser impacto real ou não. Para cada tipo de teste a ser realizado devem ser especificadas no mínimo: o objetivo do teste, a técnica/método a ser utilizada para sua execução, observações. Para maiores informações sobre os tipos de testes consulte o Guia de Tipos de Testes. ]

1. Alvos de Teste
   1. Requisitos Funcionais

Para o ***Projeto Brazuka*** serão utilizados os casos de uso que foram derivados a partir dos requisitos funcionais. Dessa forma, este tópico apresenta os casos de uso de maior prioridade que serão analisados durante a execução da primeira iteração da fase de elaboração.

| ID | Casos de Uso | Prioridade | Tipos de Teste a serem aplicados |
| --- | --- | --- | --- |
|  | Cadastrar Aluno | A | <especificar quais os testes que poderão ser aplicados para o caso de uso em questão> |
|  | Recuperar Senha | M |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* 1. Requisitos Não Funcionais

Este item apresenta os requisitos não-funcionais que serão analisados durante a execução dos testes.

| ID | Requisitos | Prioridade | Tipos de Teste a serem aplicados |
| --- | --- | --- | --- |
|  | <nome do requisito não funcional> | <A, M ou B> | <especificar quais os testes que poderão ser aplicados para o caso de uso em questão> |
|  |  |  |  |

* 1. Requisitos Não Contemplados na Execução dos Testes

Este item apresenta os requisitos funcionai ou não-funcionais que não serão analisados durante a execução dos testes.

| ID | Requisitos | Justificativa |
| --- | --- | --- |
|  | <nome do requisito não funcional ou Caso de Uso> | <especificar quais as justificativas para que o requisito ou caso de uso não seja contemplado na execução dos testes> |
|  |  |  |

1. Estratégia de Testes

[A Estratégia de Teste apresenta a estratégia recomendada para o teste da aplicação. Esta seção descreve como a aplicação será testada.

Em seguida podem ser descritas as principais considerações sobre as técnicas de teste a serem adotadas.

Em adição às considerações de cada um dos testes do item 3, os testes devem ser executados somente utilizando bases de dados conhecidas em ambientes estáveis e seguros.

Nessa seção também deverá está descrita a estratégia para automatização dos testes

As seguintes abordagens podem ser observadas:

Nível 1:

* Aplicar Teste Exploratório (teste fumaça) para avaliação preliminar da integração do build. Este procedimento pode ser realizado pelo integrador ou testador.
* Aplicar Teste Unitário, com apoio de checklist, para avaliar se os requisitos mínimos de qualidade foram atendidos para os componentes a serem integrados. Este procedimento pode ser realizado pelo desenvolvedor sobre o seu próprio código gerado.

Nível 2:

* Aplicar Teste Funcional para cada caso de uso disponibilizado, com o objetivo de prevenir defeitos antes do teste integrado.

Nível 3:

* Aplicar Teste Integrado para avaliar a efetividade da integração e a qualidade do produto a ser entregue ao cliente.
* Aplicar Teste de Regressão para cada caso de uso do build integrado que já havia sido submetido do teste de Nível 2. Para o teste de regressão pode ser utilizado o teste de tipo fumaça com o objetivo de minimizar o esforço tendo em vista que os defeitos mais graves já foram diagnosticados no teste de nível 2 já aplicado.

Nível 4:

* Aplicação de validação pelo usuário durante a homologação do produto entregue]

1. Recursos necessários
   1. Recursos Humanos

[Especificar os recursos humanos necessários para a execução dos testes. Caso essa informação esteja descrita em outro artefato do projeto, como por exemplo o Plano do Projeto, deve-se referenciar a informação em questão para o documento Plano do Projeto.]

* 1. Recursos Computacionais

[Especificar os recursos computacionais necessários para a execução dos testes, sejam eles de hardware ou software. Caso essa informação esteja descrita em outro artefato do projeto, como por exemplo, o Plano de GCS, deve-se referenciar a informação em questão para o documento Plano de GCS. No caso de software, deve ser especificada a quantidade de licenças necessárias para dimensionamento e atendimento ao projeto. Mencionar também as ferramentas para automação caso seja necessário]

[Caso exista detalhamento do ambiente de homologação ou teste no documento de arquitetura de software o mesmo pode ser referenciado evitando redundância na informação]

1. Riscos

[Listar quaisquer riscos, restrições ou contingências que possam afetar o projeto, desenvolvimento ou implementação do teste. Caso esteja em outro documento fazer a referência ao outro documento.]

1. Produtos Gerados

[Especificar os produtos que serão gerados durante a realização dos testes.

Para os artefatos de Planilha de Resultado de Teste e Sumário de Avaliação de Teste é necessário estabelecer o evento para geração dos artefatos conforme exemplo abaixo.

* Planilha de Resultado de Teste: Artefato gerado durante a execução dos ciclos de teste. Este artefato é produzido um para cada roteiro de teste.
* Sumário de Avaliação de Teste: Artefato gerado para cada build gerado com o objetivo a avaliar a qualidade do produto entregue. Este artefato é produzido um para cada build.]

1. Referência

[Listar as referências utilizadas neste documento. Para apresentação das referências, pode-se utilizar a norma aprovada pela ABNT (Associação Brasileira de Normas Técnicas) relativa à apresentação de referências bibliográficas, identificada como NBR 6023:2000 – Referências Bibliográficas]

| # | Documento |
| --- | --- |
|  | [nome do documento referenciado a este documento] |
|  |  |

1. Anexos

[Listar os anexos deste documento, com suas respectivas versões.]

| Anexo | Documento | Versão |
| --- | --- | --- |
|  | [nome do documento anexo a este documento] | [versão do documento] |
|  |  |  |