**BRUNO ALÍPIO BARBOSA**

**BRENO**

**DOUGLAS KARPINSKI**

**JORGE LUÍS BARBOSA DA SILVA**

**LUCAS DE ALMEIDA TARAIA**

**SISTEMA DE RELACIONAMENTO UNIVERSITÁRIO**

**TESTES DO SISTEMA**

Trabalho apresentado à matéria Engenharia de Software II, para o curso Ciência da Computação, da Universidade de Franca.

Professor: Heitor Cunha

**FRANCA**

**2016**

**Objetivos do Teste**

Teste de software é um elemento crítico da garantia de qualidade de software e representa a revisão final da especificação, projeto e geração de código.

1. Teste é um processo de execução de um programa com a finalidade de encontrar um erro.

2. Um bom caso de teste é aquele que tem alta probabilidade de encontrar um erro ainda não descoberto.

3. Um teste bem-sucedido é aquele que descobre um erro ainda não descoberto.

Esses objetivos implicam uma dramática mudança de ponto de vista. Eles vão contra a visão comum de que um teste bem-sucedido é aquele no qual não são encontrados erros. Nosso objetivo é projetar testes que descobrem sistematicamente diferentes classes de erros e fazê-lo com uma quantidade mínima de tempo ou esforço.

Se o teste for conduzido de maneira bem-sucedida (de acordo com os objetivos declarados anteriormente), ele descobrirá erros no software. Como benefício secundário, o teste demonstra que as funções do software parecem estar funcionando do acordo com a especificação de que os requisitos de comportamento e desempenho parecem estar sendo satisfeitos. Além disso, os dados coletados à medida que o teste é conduzido fornecem uma boa indicação da confiabilidade do software e alguma indicação da qualidade de todo o software.

**TESTES DOS REQUISITOS FUNCIONAIS**

Visam verificar se as funcionalidades especificadas foram devidamente implementadas.

Uso de métodos de testes de caixa-preta e caixa-branca:

* Teste de Usabilidade

Teste focado na experiência do usuário, consistência da interface, layout, acesso às funcionalidades etc.

* Teste Funcional

Testa os requisitos funcionais, as funções e os casos de uso. “A aplicação faz o que deveria fazer? ”.

* Teste de Unidade

Testa um componente isolado ou classe do sistema.

**TESTES DOS REQUISITOS NÃO FUNCIONAIS**

Visam determinar se a implementação do sistema satisfaz aos requisitos não funcionais.

Tipos de testes que serão realizados:

* Teste de Performance

O teste de performance se divide em 3 tipos:

- Teste de carga

Testa o software sob as condições normais de uso. Ex.: tempo de resposta, número de transações por minuto, usuários simultâneos etc.

- Teste de stress

Testa o software sob condições extremas de uso. Grande volume de transações e usuários simultâneos. Picos excessivos de carga em curtos períodos de tempo.

- Teste de estabilidade

Testa se o sistema se mantém funcionando de maneira satisfatória após um período de uso.

* Teste de Segurança

Testa se o sistema e os dados são acessados de maneira segura, apenas pelo autor das ações. (*Logon*/*Logoff*, Permissões, Ataques de Dicionário, requisitos mínimos de senha).

* Teste de Configuração

Testa se o software funciona no hardware e sistema a ser instalado.

* Teste de Compatibilidade

Testa se o sistema é capaz de rodar em diversas plataformas (Sistemas Operacionais e Navegadores).

**Ferramentas que serão utilizadas nos testes**

*Acunetix* – Scanner de sites em busca de vulnerabilidades. Procura falhas críticas e leves, entre as falhas estão: Injeção de SQL, XSS, Fopen entre outros.

*SQLmap* – Ferramenta open source para teste de penetração que automatiza o processo de detecção e exploração de vulnerabilidades a injeção de SQL.

*Selenium* – Ferramenta que testa diretamente pelo browser, exatamente como o usuário faria. Apresenta como uma extensão do Firefox.

*JMeter* – Ferramenta utilizada para testes de carga (Stress).

*Nessus* – ferramenta de verificação de falhas/vulnerabilidades de segurança

**Bibliografia**

Pressman, Roger S. Engenharia de Software MAKRON BOOKS 5º Edição – 2002.

Sommerville, Ian. Engenharia de Software. Addison Wesley, 6ª Edição – 2003.