

IFPB - Campina Grande

Disciplina: Testes de Software Professor: Emanuel Dantas

Avidade 01: Pesquisa sobre segurança de software

\* atividade a ser realizada de forma individual

01) Escolher um dos sistemas a seguir:

1. Sistema para controle acadêmico do IFPB (SUAP)
2. Sistema para compra e revenda de livros usados
3. Sistema para controle do estoque de um supermercado 4.
- Sistema para agendamentos de consultas em um Hospital

Para cada fator de qualidade (Segurança, Usabilidade, Portabilidade, Desempenho):

- o Mostrar porque o fator é importante no sistema escolhido. o Citar exemplos de situações onde o fator em questão está sendo violado no sistema escolhido
  - o Ranquear os fatores do mais significativo para menos significativo no sistema escolhido.

*A **Usabilidade** do software:*

- 1. É fácil de ser utilizado?*
- 2. Usuários que não entendem tanto do assunto podem aprender como usá-lo com facilidade?*

*São questões a serem feitas ao projetar o software! Se ambas as perguntas atenderem aos usuários, esse é um dos principais critérios para mensurar a qualidade de um software;*

*A **Segurança** é importante no controle de entrada, renovação, reserva, saída, cancelamento e serviços afins que não seja vulnerável a modificação não autorizada, tampouco deve possibilitar a um usuário, mal intencionado ou não, ter acesso a dados, podendo difundir, alterar ou mesmo destruí-los;*

A **Portabilidade** de um software precisa ser de fácil adaptação nas diferentes plataformas utilizadas pelo usuário para que os mesmos tenham uma melhor experiência. Para isso, é necessário que haja uma solução para diferentes plataformas e objetivos em que o software pode ser operado, sem a necessidade de demais ações ou meios para que ele funcione adequadamente ou tenha suas funcionalidades minimizadas;

O **Desempenho** que tem relação com a eficiência do software que se refere ao tempo de resposta para executar uma função desejada pelo usuário. Realizar a avaliação de qualidade de software é essencial para alcançar da melhoria à excelência na funcionalidade com o seu produto e aumentar a competitividade.

02) Realizar uma pesquisa sobre segurança de software. Mostrar como os testes podem ajudar a identificar falhas de segurança e porque essas falhas são prejudiciais ao software. Em sua pesquisa, você precisa responder pelo menos as seguintes questões:

- O que é segurança?
- Como um software pode ter sua segurança violada?
- Quais testes podem ser feitos para garantir a segurança do software?
- Pesquise um exemplo onde a segurança de um sistema “famoso” foi atingido. Pode ser de algum banco, empresa de telecomunicações, governo, etc.

### **Segurança**

O objetivo da segurança de software é produzir software que não seja vulnerável a modificação não autorizada, tampouco deve possibilitar a um usuário, mal intencionado ou não, ter acesso a dados, podendo difundir, alterar ou mesmo destruí-los.

Os testes de segurança de software são responsáveis por verificar se o programa está funcionando como esperado, verificando se há vulnerabilidades e como ele responde diante de diferentes situações que envolvam ameaças à segurança.

## **Violação**

*Uma violação de segurança é qualquer incidente que resulte em acesso não autorizado a dados, aplicativos, redes ou dispositivos de computador. O resultado é que as informações são acessadas sem autorização. Normalmente, isto ocorre quando um intruso é capaz de burlar os mecanismos de segurança.*

*O teste faz parte de um processo no desenvolvimento do programa, podendo ser feito pelos próprios desenvolvedores ou, em alguns casos, feito por profissionais especializados na área. O procedimento tem como objetivo antecipar e corrigir falhas e bugs que apareceriam para o usuário final.*

## **Testes**

*Para identificar e organizar os erros, os testes são separados em diferentes tipos:*

### **Testes de Caixa Branca**

*Por ter acesso ao código fonte, o profissional pode observar com mais atenção as determinadas etapas do código. Nesse processo, o usuário analisa por qual caminho ocorre o fluxo de dados e é possível verificar se há a passagem correta em todas as condições esperadas;*

### **Teste da Caixa Preta**

*Nesse procedimento o profissional não tem acesso ao código fonte e nem a sua estrutura. Como é baseado nos requisitos funcionais, ele também é chamado de teste funcional. Para que o teste seja feito com sucesso, o resultado deve sair como esperado pelo usuário.*

### **Teste de Regressão**

*Durante o desenvolvimento do programa, é comum termos as seguintes situações: ou a inclusão de uma nova funcionalidade pelo cliente ou encontrar um erro na lógica do código fonte.*

*Independentemente do cenário, o desenvolvedor terá de fazer a alteração na programação. O problema é que, em alguns casos, uma simples mudança pode comprometer toda a lógica já escrita, invalidando quaisquer testes básicos feitos no processo de produção;*

### **Teste de Usabilidade**

*Utilizando o ponto de vista do consumidor, o teste tem como objetivo verificar a experiência do usuário. Dessa forma, o responsável deve checar a organização dos itens disponíveis na tela, observar se o layout está correto e se os botões se comunicam corretamente entre as diferentes páginas do sistema;*

## **Segurança**

*Os testes de software evoluíram muito com o tempo e um know-how maior foi adquirido pelos profissionais da área. Dentre os novos recursos utilizados para garantir o pleno funcionamento de um programa de computação estão os testes de segurança;*

### *Integração*

*Nesse teste, em vez de se atestar funcionalidades do software, se analisa a integração entre as diferentes unidades que formam o sistema. São averiguados aspectos como a interface e a dependência entre os componentes;*

### *Performance*

*Atesta o desempenho do software. Se os comandos dados respondem rapidamente, se os componentes não demoram muito a carregar e se a experiência do usuário é satisfatória no produto testado. É de suma importância, pois por mais bem programado que um software possa ser, sua performance é que determinará o uso satisfatório das funções planejadas;*

### *Instalação*

*O teste de instalação verifica se sob diferentes condições como pouco espaço de memória, interrupções no sistema e demais entraves que podem comprometer esse processo, o programa consegue ser instalado ou se cede facilmente a essas limitações;*

### *Manutenção*

*Um software não é feito para durar pouco tempo, tampouco uma única versão. Atualizações constantes são necessárias a fim de aprimorar os recursos do programa e a própria experiência do usuário com aquele produto. Os testes de manutenção averiguam se esses aprimoramentos acontecem com sucesso e se são aceitos pelo sistema. Sem isso, corre-se o risco do software tornar-se defasado e até mesmo inoperante pela falta de atualização;*

### *Funcional*

*O teste funcional abrange tanto o teste de caixa branca quanto o de caixa-preta abordado anteriormente nesse post. Sua importância consiste justamente na capacidade de determinar se o que o software foi programado para fazer está de fato fazendo.*

### *Exemplo*

#### ***Netflix, LinkedIn, Last.FM e outros – 1,4 bilhão de senhas vazadas***

*No fim de 2017, foi encontrado um arquivo que reúne mais de 1,4 bilhão de nomes de usuários e senhas de diversos sites, como Netflix, LinkedIn, MySpace, Las.FM, Minecraft e YouPorn.*

*O arquivo era organizado por ordem alfabética e frequentemente atualizado pelos hackers. Ele poderia ser facilmente encontrado por meio de plataformas para download de torrents e na dark web (uma parte da deep web onde são compartilhadas informações ilegais).*

*Apesar de a Netflix ter negado o vazamento, especialistas que tiveram acesso ao arquivo que compilava as senhas afirmaram que muitas delas estavam corretas.*

