



www.devmedia.com.br

[versão para impressão]

Link original: <http://www.devmedia.com.br/articles/viewcomp.asp?comp=29799>

Introdução aos diferentes tipos de teste

Este artigo apresenta alguns dos tipos de teste menos comuns dentro de um Processo de Teste de Software, os Testes de Usabilidade, Confiabilidade, Portabilidade e Acessibilidade.

Fique por dentro

Este artigo apresenta alguns dos tipos de teste menos comuns dentro de um Processo de Teste de Software, os Testes de Usabilidade, Confiabilidade, Portabilidade e Acessibilidade. O objetivo é analisá-los demonstrando os ganhos que podem ser obtidos quando é destinada uma pequena fatia de tempo e investimento para a realização dos mesmos.

Esse tema é útil quando o objetivo dos envolvidos no projeto é agregar qualidade ao software, fugindo do “vício” habitual das empresas de executar apenas os testes funcionais mais comuns, aumentando assim a fidelização e a satisfação dos clientes.

Com o aumento da utilização de sistemas web, associado a uma busca contínua por mais qualidade, menos riscos e melhores resultados, adicionar o Teste de Software ao ciclo de vida de desenvolvimento do software torna-se cada vez mais importante.

O problema é que o Teste, uma área relativamente nova, ainda encara muitas barreiras. Isto se dá, às vezes, em função dos recursos humanos e financeiros reduzidos, outras, por causa do tempo e dos prazos limitados dos projetos, sem contar que existem situações em que o Teste é simplesmente deixado de fora do escopo do projeto por puro desconhecimento de causa ou até mesmo despreparo dos gestores envolvidos.

O que ocorre é que nos dias de hoje não dá mais para deixar o Teste como uma atividade irrelevante. É sabido que não é viável realizar uma centena de tipos de testes para garantir a qualidade de um software, mas que tal começar devagar, adicionando novos testes gradativamente, para que aos poucos essa atividade possa conquistar o seu devido espaço na empresa e a confiança do cliente?



É válido lembrar que cada projeto apresenta características distintas, que dependem do tamanho do software, da tecnologia utilizada para o seu desenvolvimento e de muitos outros fatores. Assim, a escolha adequada dos tipos de testes que serão adotados torna-se primordial.

Nesse artigo, o foco será dado aos testes de Usabilidade, Aceitação, Portabilidade e Confiabilidade. No entanto, antes de prosseguir, para auxiliar na compreensão do que representa cada um desses testes, vamos apresentá-los sucintamente, através dos seguintes questionamentos:

- **Teste de Usabilidade:** O software é fácil de usar?
- **Teste de Confiabilidade:** O quanto podemos contar com o correto funcionamento do software? Ele é tolerante a falhas?
- **Teste de Portabilidade:** É possível utilizar o software em diversas plataformas com pequeno esforço de adaptação?

- **Teste de Acessibilidade:** Qualquer usuário, deficiente ou não, consegue utilizar a aplicação?

Ao longo desse artigo, além dos testes já citados, serão apresentadas as vantagens e as ferramentas que viabilizam as suas respectivas execuções. Como referência principal, será utilizada a ISO-9126, uma norma que define características e subcaracterísticas voltadas para a qualidade de software.

Teste de Usabilidade

O Teste de Usabilidade tem como objetivo avaliar a usabilidade da aplicação, determinando até que ponto a interface do software é fácil e intuitiva de utilizar.

Esse tipo de teste possibilita detectar todas as ações dos usuários, analisar suas preferências, ajudando a determinar o que pode ser melhorado na aplicação.

Os problemas mais comuns que podem ser detectados com a execução de testes de usabilidade em aplicações web são:

- **Irrelevância:** informações exibidas na interface que não possuem nenhuma utilidade, sem pertinência às necessidades do usuário;
- **Inadequação:** abreviaturas usadas sem prévia explicação do termo completo;
- **Resposta inesperada:** ao acionar um link, a aplicação apresentar outra página totalmente diferente da solicitada;
- **Cliques:** efetuar muitos cliques para conseguir chegar à funcionalidade desejada.

Com a realização desse teste é possível analisar o tempo de resposta do servidor para cada requisição, e até mesmo a satisfação, mesmo que subjetiva, do usuário, em relação ao que está sendo apresentado.

Jakob Nielsen, um dos maiores especialistas na área de usabilidade, listou um conjunto de critérios básicos para analisar a usabilidade de um software. São eles:

- **Intuitividade:** o software deve ser fácil de ser utilizado, permitindo que mesmo um usuário sem experiência seja capaz de executar algum trabalho satisfatoriamente;
- **Eficiência:** o sistema deve ser eficiente em seu desempenho, viabilizando um alto nível de produtividade;
- **Memorização:** as interfaces da aplicação devem de fácil memorização, permitindo que usuários ocasionais consigam utilizá-lo mesmo depois de um longo intervalo de tempo;
- **Erro:** a quantidade de erros apresentados pelo software deve ser a menor possível. Além disso, a solução para os problemas deve ser simples e rápida, mesmo para usuários iniciantes. Erros graves ou sem solução não podem ocorrer;
- **Eficácia:** o software deve, sobretudo, cativar o usuário, sejam eles iniciantes ou avançados, possibilitando uma interação no mínimo agradável.

Analisar essas características de usabilidade em uma aplicação em desenvolvimento, ou até mesmo pronta, possibilita que muitos problemas sejam diagnosticados antes do software ser disponibilizado para o cliente.

Isso pode, no mínimo, evitar a insatisfação do cliente numa fase avançada do ciclo de vida do desenvolvimento do software, onde as correções se tornam mais caras e impactantes.

Como executar o teste de usabilidade

Muitas são as maneiras de medir a usabilidade de um software. A escolha depende de cada projeto e está diretamente ligada ao resultado que se deseja obter.

Executados em laboratório, em um ambiente controlado, ou em ambiente real, muitas técnicas de teste de usabilidade podem ser utilizadas. De uma maneira geral, o teste de usabilidade pode ser realizado seguindo quatro etapas principais:

- **Planejamento:** envolve determinar o objetivo do teste, bem como a metodologia que será utilizada. Define também o perfil dos participantes, como recrutá-los, o cenário e as tarefas que serão realizadas. Nessa primeira etapa, um ponto fundamental é saber exatamente o que será avaliado. Criar um pequeno checklist, como o exemplificado a seguir, pode facilitar essa etapa:

- o Qual o objetivo do teste?

- o Quando e onde o teste será executado?

- o Qual o tempo de duração para cada sessão de teste?

- o Qual ambiente e ferramentas necessárias para executá-lo?

- o Em qual estado o software deve estar no início do teste?

- o Quem serão os usuários envolvidos no teste?

- o Quais tarefas serão definidas para que os usuários executem?

- o Quais são as funcionalidades mais críticas?

- o Quais serão os dados coletados durante os testes?

- o Qual critério será utilizado para definir se um teste foi bem sucedido?

· **Preparação:** na segunda etapa, tudo deve estar pronto para por o teste em prática. É nesse momento que o usuário receberá uma explicação sobre como será executado o teste, e caso alguma ferramenta seja utilizada, um breve treinamento deve ser realizado;

· **Execução:** na terceira etapa é onde o teste é efetivamente executado. É aconselhável que antes de realizar o primeiro teste, um projeto experimental seja colocado em prática.

Nesse momento serão detectadas as informações primordiais para a avaliação posterior do resultado. É recomendável ainda utilizar um cronômetro, e é importante informar ao usuário que executará o teste o tempo limite para conclusão de cada tarefa. Outros pontos também devem ser seguidos, a saber:

o Não interfira no teste, no momento da execução;

o Documente os pontos de dificuldade citados pelos usuários;

o Anote as sugestões do(s) usuário(s) participante(s).

· **Análise:** esta é a etapa final dos testes de usabilidade e tem como objetivo analisar os dados obtidos. É o momento de revisar os problemas encontrados, definindo a frequência com que ocorrem, a gravidade e a prioridade para resolução dos mesmos.

Nessa etapa, é fundamental que todos os detalhes sejam avaliados, como por exemplo, a quantidade de cliques antes da conclusão de uma tarefa, o tempo de duração de cada atividade, quantas vezes o help foi acionado e, se for o caso, quantos erros foram apresentados.

Ferramentas

Nos últimos anos, houve um considerável aumento no número de ferramentas voltadas para o teste de usabilidade. Para exemplificar, apresentaremos três delas:

- **Loop11:** é uma ferramenta online voltada para a execução de testes de usabilidade não moderados (sem o acompanhamento presencial de um orientador) com usuários reais. Para utilizá-la, basta cadastrar algumas tarefas, solicitar que sejam realizadas e o Loop11 irá armazenar as ações.

Como resultado, através de relatórios você poderá rastrear a interação do usuário com a aplicação, o tempo de conclusão de cada tarefa, os caminhos percorridos e as falhas apresentadas;

- **Ethnio:** permite recrutar os visitantes da sua aplicação web para ajudá-lo com a pesquisa sobre a usabilidade do software. Ou seja, a ferramenta resolve o problema de tentar encontrar usuários reais para participar dos testes e ainda envia relatórios periódicos sobre os resultados obtidos;

- **Morae:** permite captar toda a interação do usuário, incluindo voz, vídeo, teclas, movimentos do mouse e ações na tela. Ou seja, essa ferramenta é um laboratório de testes de usabilidade completo, pois é capaz de obter uma sofisticada gama de informações, possibilitando uma rica análise da aplicação.

Vantagens do Teste de Usabilidade

A execução desse tipo de teste pode fornecer inúmeros benefícios ao software, a saber:

- Avaliação da qualidade das telas da aplicação;
- Identificação de problemas visando melhorias durante o desenvolvimento do software;
- Pode ser associada à prototipação, tornando a avaliação do protótipo mais efetiva.

Teste de Confiabilidade

A confiabilidade de um software é medida de acordo com a estabilidade e o desempenho da aplicação durante um determinado período de tempo, sob diferentes condições de teste.

O objetivo desse teste é garantir a integridade completa dos dados trafegados pelo software, monitorando e avaliando a capacidade que a aplicação tem de concluir as suas operações com sucesso, conforme especificado.

As informações obtidas ao longo dos testes de confiabilidade devem ser coletadas em todas as etapas do ciclo de vida do desenvolvimento de software, identificando sempre quando uma interrupção produzir uma falha.

Em suma, o teste de confiabilidade visa analisar a integridade, conformidade e interoperabilidade do software, incorporando os resultados de testes não funcionais, como os testes de segurança e estresse, aos testes funcionais. Esse teste pode ser descrito através de algumas características, a saber:

- **Recuperabilidade:** o software é capaz de voltar a funcionar após uma falha?
- **Maturidade:** com que frequência a aplicação apresenta falhas inesperadas?
- **Tolerância a falhas:** como o software se comporta quando uma falha ocorre?
- **Conformidade:** a aplicação está de acordo com normas e convenções previstas em leis?

Como executar o teste de confiabilidade

O teste de confiabilidade é considerado caro, em comparação com os demais citados ao longo desse artigo. Isto porque pode envolver outras técnicas de teste. Sendo assim, torna-se primordial executá-lo de forma planejada e estruturada. Para isso, alguns passos podem ser seguidos, conforme apresentado a seguir:

- **Estabeleça metas de confiabilidade:** o objetivo dessa primeira etapa é determinar qual será o desempenho satisfatório do software em termos que sejam significativos para o cliente. O ideal é que todos os envolvidos no ciclo de vida do desenvolvimento estejam cientes da priorização dessas metas;

- **Desenvolva perfis operacionais:** definir quem irá realizar esses testes também é muito importante. Feito isso, é preciso descrever exatamente o que cada usuário irá realizar, juntamente com o resultado que se deseja alcançar. Os recursos de hardware necessários para execução dos testes também devem ser especificados;

- **Planeje e execute os testes:** esse terceiro passo visa levantar os dados necessários para execução dos testes, bem como verificar as exigências dos requisitos, atribuir as responsabilidades de acordo com os perfis operacionais e, por fim, registrar todas as informações obtidas ao longo da execução do teste de confiabilidade;

- **Use os resultados do teste para tomar decisões:** após a execução do teste, tem-se o momento de avaliar os resultados. Este passo é fundamental para definir qual será a priorização da solução para os problemas apresentados.

Vantagens do Teste de Confiabilidade

A execução desse tipo de teste pode fornecer inúmeros benefícios ao software, a saber:

- Determina se o software atende aos requisitos de confiabilidade definidos pelos usuários, proporcionando a confiança de cumprir com os objetivos pretendidos;
- Verifica a existência de pontos fracos na aplicação;
- Analisa até que ponto o software está apto a suportar as falhas apresentadas.

Teste de Portabilidade

O Teste de Portabilidade tem como objetivo verificar o grau de portabilidade da aplicação em diferentes ambientes e situações, envolvendo desde o hardware até o software. Por exemplo, um grande desafio para quem desenvolve aplicações web é garantir que ela tenha o mesmo comportamento independente do navegador que o usuário esteja utilizando.

A execução desse tipo de teste consiste em avaliar algumas características fundamentais de uma aplicação. Dentre elas, destacamos:

- **Adaptabilidade:** É fácil adaptar o software a outras plataformas?
- **Coexistência:** A aplicação conseguirá compartilhar os recursos comuns com outros softwares independentes?
- **Capacidade de substituição:** É fácil substituir o software por outro especificado para o mesmo fim?

Esse tipo de teste pode ter o seu planejamento voltado para avaliar questões de hardware, browsers, de diferentes tipos, e sistemas operacionais, com suas várias versões e *service packs*.

Uma dica para testar a aplicação em diferentes plataformas é a utilização de máquinas virtuais (VMs). Através de emulação as VMs permitem que diferentes sistemas operacionais sejam executados em uma mesma máquina, sem a necessidade de *dual boot*. Além disso, as VMs permitem salvar o estado da máquina ou restaurá-lo facilmente.

Como executar o teste de portabilidade

Executar um teste de portabilidade não é complicado e nem requer um grande planejamento, basta utilizar o software na arquitetura desejada. Para que seja viável a execução desse tipo teste, algumas pré-condições devem ser seguidas:

- Verificar se o software foi ou está sendo desenvolvido seguindo os requisitos de portabilidade;
- Confirmar se a integração entre todos os componentes foi realizada;
- Analisar se o ambiente está preparado, apto para a execução do Teste de Portabilidade.

Ferramentas

Para execução do teste de Portabilidade, não é necessário que haja um computador para cada teste. Existem ferramentas que simulam diferentes situações numa mesma máquina, por exemplo:

- **IETester:** disponível gratuitamente em sua versão *alpha*, é uma ferramenta simples e intuitiva que possibilita emular o navegador Internet Explorer e de uma só vez utilizar desde a versão 5.5 até a versão 10;
- **Windows XP Mode:** é uma ferramenta para o Windows 7 que, em conjunto com o Windows Virtual PC, propicia emular uma versão do Windows XP;

- **VirtualBox:** é uma ferramenta da Oracle que permite instalar e executar diferentes sistemas operacionais em um único computador. Pode ser instalado no Windows, Linux, Solaris e Mac.
- **VMware Player:** é um software gratuito que funciona como uma máquina virtual, assim como o VirtualBox. Permite a instalação e utilização de um ou vários sistemas operacionais dentro de um mesmo computador. É possível até mesmo utilizar os dispositivos USB conectados à máquina física e a internet.

Vantagens do Teste de Portabilidade

A execução desse tipo de teste determina se o software pode ser portado para cada um dos ambientes especificados pelo usuário, além de avaliar características fundamentais para a sua correta utilização, como:

- Espaço necessário em disco;
- Velocidade do processador;
- Resolução do monitor;
- Navegadores e Sistemas Operacionais suportados.

Teste de Acessibilidade

O Teste de Acessibilidade tem como objetivo garantir que o software poderá ser utilizado por qualquer usuário, inclusive aqueles que possuam algum tipo de deficiência física. Esse teste verifica se as interfaces do software permitem uma navegação adequada para todos.

Visando as aplicações web, existem padrões, citados a seguir, que determinam se existe ou não acessibilidade no software. Essa questão é tão relevante que, para garantir que os serviços ou informações governamentais disponíveis na web sejam acessíveis a todos os usuários, em qualquer horário, local, ambiente ou dispositivo, o governo brasileiro definiu, em 2005, o “Modelo Brasileiro de Acessibilidade em Governo Eletrônico”, conhecido como e-Mag.

No mundo, com o objetivo de tornar a web acessível a um número cada vez maior de cidadãos, o World Wide Web Consortium (W3C), principal organização de padronização da Web, um consórcio internacional com quase 400 membros, criou o WAI (*Web Accessibility Initiative*), com a atribuição de manter grupos de trabalho elaborando conjuntos de diretrizes para garantir a acessibilidade ao conteúdo da Internet a pessoas com deficiências, ou para os que acessam a rede em condições especiais de ambiente, equipamento, navegador e outras ferramentas Web.

Em suma, diante de tantas padronizações, algumas características são fundamentais quando se trata de testar a acessibilidade de uma aplicação web. Dentre elas, destacamos:

- **Perceptível:** a informação e os componentes da interface da aplicação devem ser apresentados de modo que os usuários possam perceber o que é exibido na tela;
- **Operável:** a interface do software não pode exigir uma interação que o usuário não consiga executar. Além disso, os componentes da tela precisam ser operáveis;
- **Compreensível:** a informação e a operação da interface da aplicação devem ser de fácil compreensão;

- **Robusto:** o conteúdo precisa ser sólido o suficiente para que ele possa ser interpretado por diferentes usuários, ferramentas e tecnologias assistivas (recursos e serviços que contribuem para proporcionar ou ampliar habilidades funcionais de pessoas com deficiência).

Lembre-se: Para o teste de acessibilidade, quanto mais autoexplicativa é a aplicação, mais facilmente ela será utilizada. Isto permitirá também que os sites de busca encontrem mais rapidamente o conteúdo da sua aplicação web.

Como executar o teste de acessibilidade

Para medir a acessibilidade de um software algumas condições precisam ser seguidas. Segundo o *checklist* do WCAG1.0 da W3C um documento que fornece uma listagem com todos os pontos de verificação previstos nas “Diretrizes para Acessibilidade ao Conteúdo da Web”, o primeiro passo é definir a prioridade da funcionalidade baseada no seu impacto. Essas prioridades podem ser divididas em três categorias:

- **Prioridade1:** pontos que têm que ser totalmente desenvolvidos na aplicação. Caso contrário, um ou mais grupos de usuários ficarão impossibilitados de acessar as informações contidas no software;
- **Prioridade2:** pontos que o software deve satisfazer. Caso contrário, um ou mais grupos de usuários terão dificuldades em acessar as informações contidas na aplicação;
- **Prioridade3:** pontos facultados, ou seja, que a aplicação pode ou não contemplar. Caso contrário, um ou mais grupos poderão se deparar com algumas dificuldades em acessar a aplicação.

Diante de tantas padronizações vindas dos mais distintos setores, desde os públicos até os privados, o teste de acessibilidade possibilita que sejam analisadas inúmeras características na aplicação, como por exemplo:

- Verificar se os campos da interface permitem uma tabulação na ordem da disposição dos campos na tela, visando à impossibilidade de utilizar o mouse;
- Analisar se é viável a navegação por toda a aplicação apenas com o teclado;
- Verificar a existência de texto informativo nas imagens (*hint*), possibilitando que outros softwares possam informar ao usuário o seu respectivo conteúdo;
- Analisar se é possível alterar o tamanho do conteúdo da interface, como textos, tabelas e imagens;
- Verificar se em tabelas de dados são identificados os cabeçalhos de linha e de coluna;
- Verificar se é fornecida uma descrição sonora das informações importantes veiculadas em trechos visuais de apresentações multimídia;
- Analisar se as informações estão divididas em pequenos blocos, facilitando o gerenciamento;
- Verificar se os mecanismos de navegação estão sendo utilizados de maneira coerente e sistemática;
- Verificar se está especificado por extenso cada abreviatura ou sigla ao longo da aplicação;
- Verificar se é identificado o principal idioma utilizado pela aplicação.

Ferramentas

Nos últimos anos, houve um considerável aumento no número de ferramentas voltadas para o teste de usabilidade. Para exemplificar, apresentaremos três delas:

- **AChecker:** é uma ferramenta que permite que você informe uma URL e teste uma aplicação web, verificando a conformidade de acordo com diversas orientações de acessibilidade;
- **Wave Toolbar:** é uma barra de ferramentas do Firefox que fornece um mecanismo para a execução de testes diretamente do navegador. Um fator interessante é que nenhuma informação é enviada para o servidor Wave, garantindo assim uma acessibilidade 100% privada e segura. A ferramenta consegue avaliar a versão renderizada da aplicação e todo o conteúdo gerado dinamicamente;
- **NVDA:** NonVisual Desktop Access é uma ferramenta open source de leitura de tela. Ela permite que pessoas com algum nível de deficiência visual ouçam o conteúdo por meio de uma voz sintetizada. Suporta mais de 20 idiomas e não precisa ser instalada no computador, bastando utilizá-la diretamente a partir de um *pendrive*.

Vantagens do Teste de Acessibilidade

A execução desse tipo de teste pode fornecer inúmeros benefícios ao software, a saber:

- Disponibilizar uma aplicação para um público mais amplo permite melhoria na fidelização dos clientes;
- Todo usuário, independentemente de ser portador ou não de deficiência, terá acesso à aplicação;

- Reforça e amplia a acessibilidade como uma característica relevante do software.

Conclusão

Executar os testes de Usabilidade, Confiabilidade, Portabilidade e Acessibilidade não é uma atividade comum dentro de um Processo de Teste de Software. No entanto, à medida que são inseridos, de forma combinada ou não, podem contribuir significativamente para a ampliação do público alvo da aplicação.

Em termos de Portabilidade, hoje em dia estão acessíveis a praticamente todos os públicos os mais distintos sistemas operacionais e navegadores. Pensando na acessibilidade, estima-se que somente no Brasil, segundo o último censo do IBGE, 23,95% da população sofra de algum tipo de deficiência. Já a Usabilidade e Confiabilidade podem garantir, no mínimo, a fidelização do cliente, tornando o software mais eficaz, eficiente e compreensível.

Executados sozinhos, esses testes podem até não garantir um software (quase) perfeito, mas agregados a outros tipos de teste e diante dos diferentes aspectos expostos ao longo desse artigo, bons resultados podem ser obtidos.

Para acrescentá-los ao ciclo de vida de desenvolvimento do software, não será necessário nem tanto investimento e, talvez, nem tanto tempo como se imagina, considerando a melhoria da qualidade que pode ser obtida na aplicação.

Links

10 Usability Heuristics for User Interface Design.

<http://www.nngroup.com/articles/ten-usability-heuristics/>

Checklist para os pontos de verificação para a acessibilidade e ao conteúdo da Web – Diretrizes 1.0.

<http://www.maujor.com/w3c/clistcpointac.html>

Como testar a acessibilidade em websites? (Parte 1).

http://internativa.com.br/artigo_acessibilidade_03_06.html

Engenharia de Usabilidade.

<http://www.labiutil.inf.ufsc.br/hiperdocumento/>

[Engenharia de Usabilidade Nielsen.doc](#)

Reliability Guidelines.

http://www.testingstandards.co.uk/reliability_guidelines.htm

Um processo para o desenvolvimento de aplicações web acessíveis.

<http://www.dct.ufms.br/gestor/titan.php?target=openFile&fileId=479>



por Renata Eliza

Expert em Java e programação Web
