

UNIVERSIDADE FEDERAL DO MARANHÃO  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA  
CURSO DE CIÊNCIA DA COMPUTAÇÃO

**ANDRÉ DIAS BARROS**

**INCREMENTANDO A QUALIDADE DE APLICAÇÕES WEB  
(WEBAPPS) ATRAVÉS DE ESTUDOS DE USABILIDADE**

São Luís

2015

**ANDRÉ DIAS BARROS**

**INCREMENTANDO A QUALIDADE DE APLICAÇÕES WEB  
(WEBAPPS) ATRAVÉS DE ESTUDOS DE USABILIDADE**

Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Prof<sup>a</sup>. Ma. Maria Auxiliadora Freire.

São Luís

2015

Barros, André Dias.

Incrementando a Qualidade de Aplicações Web (WebApps) Através de Estudos de Usabilidade/ André Dias Barros. – São Luís, 2015.

78 f.

Orientadora: Profa. Msc. Maria Auxiliadora Freire.

Monografia (Graduação) – Universidade Federal do Maranhão, Curso de Ciência da Computação, 2015.

1. Aplicações web 2. Globalização 3. Qualidade 4. Usabilidade I. Título.

CDU 004.738.5


ANDRÉ DIAS BARROS


INCREMENTANDO A QUALIDADE DE APLICAÇÕES WEB  
(WEBAPPS) ATRAVÉS DE ESTUDOS DE USABILIDADE


Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

Aprovada em: 16 / 06 / 2015

BANCA EXAMINADORA

  
\_\_\_\_\_  
Prof.<sup>a</sup> Ma. Maria Auxiliadora Freire (Orientadora)  
Mestra em Ciência da Engenharia  
Universidade Federal do Maranhão

  
\_\_\_\_\_  
Prof. Me. Antônio de Abreu Batista Júnior  
Mestre em Ciência da Computação  
Universidade Federal do Maranhão

  
\_\_\_\_\_  
Prof. Dr. Samyr Beliche Vale  
Doutor em Ciência da Computação  
Universidade Federal do Maranhão

## AGRADECIMENTOS

Agradeço primeiramente a Deus, por me abençoar nessa caminhada e permitir que conseguisse chegar até o final.

A minha mãe e aos meus familiares, que acreditaram em mim, e sempre estiveram dispostos a me ajudar em tudo que fosse necessário.

Aos meus colegas de curso por esse período de convivência, troca de conhecimento e experiências tanto boas quanto más, que contribuíram para o meu desenvolvimento.

Aos meus professores, pelo conhecimento que me proporcionaram, e principalmente, à minha orientadora Maria Auxiliadora, pela valiosa ajuda nessa fase final do curso.

E a todos que de alguma forma contribuíram para que a realização desse trabalho fosse possível.

*“A vida me ensinou a nunca desistir.  
Nem ganhar, nem perder, mas procurar evoluir”.*

*(Charlie Brown Jr.)*

## RESUMO

Uma das áreas da tecnologia mais dinâmica é a área de desenvolvimento de aplicações web, que exige de seus profissionais contínuo processo de aprendizagem. E para que as aplicações web sejam produzidas com qualidade é necessário o uso de metodologias que conduzam o processo de maneira ágil, simples, porém eficazes e produtivas. Com a globalização, o crescimento da internet e a necessidade de serviços rápidos têm ocasionado o desenvolvimento de aplicações web. Diante deste cenário, o presente trabalho tem como objetivo promover o desenvolvimento da qualidade nas aplicações web. No desenvolvimento do projeto fez-se necessário a análise de diversos pontos de usabilidade nas aplicações web, com intuito de melhorar a experiência de uso do usuário nas aplicações web.

Palavras-chave: Aplicações web. Globalização. Qualidade. Usabilidade.

## ABSTRACT

One of the areas of the most dynamic technology is the web application development area that requires its professional ongoing learning process. And for that web applications are produced with quality using methodologies that lead the process of agile, simple but effective and productive is required. With globalization, the growth of the internet and the need for fast service have led to the development of web applications. Against this background, this paper aims to promote the development of quality in web applications. In project development it was necessary to analyze different points of usability in web applications, aiming to improve the user experience in the use of web applications.

Keywords: Web Applications. Globalization. Quality. Usability.



## LISTA DE FIGURAS

Figura 2.1 - Evolução da web. Fonte: Aplicação de Software Social 2011 .....	18
Figura 2.2 - Web 1.0. Fonte: Agregando Conhecimento e Informação .....	18
Figura 2.3 - Web 2.0. Fonte: Agregando Conhecimento e Informação .....	19
Figura 2.4 - Representação do modelo de janelas em cascata. Fonte: MATOS, A.; BRITTO, T, 2007	21
Figura 2.5 - Fluxo de navegação em rede (teia). Fonte: MATOS, A.; BRITTO, T, 2007 .....	22
Figura 2.6 - Arcabouço da WebE. Fonte: PRESSMAN, R.S. 2009 .....	24
Figura 2.7 - A pirâmide de projeto da WebApp. Fonte: PRESSMAN, R.S. 2009 .....	34
Figura 3.1 - Árvore de requisitos de qualidade. Fonte: Engenharia de Software .....	40
Figura 4.1 - Websites de notícias .....	55
Figura 4.2 - Websites institucionais .....	58
Figura 4.3 - Gráfico que representa o percentual de uso da ferramenta flash nos websites analisados .....	59
Figura 4.4 - Websites diversos .....	61
Figura 4.5 - Gráfico que representa o percentual de uso da área de contato nos websites analisados .....	64
Figura 4.6 - Websites diversos .....	63
Figura 4.7 - Gráfico que representa o percentual de uso de menus de navegação nos websites analisados .....	66
Figura 4.8 - Websites diversos .....	68
Figura 4.9 - Websites diversos .....	71

## LISTA DE TABELAS

Tabela 2.1 - Comparativo entre aplicação convencional e aplicação web. Fonte: MATOS, A.; BRITTO, T, 2007 .....	22
Tabela 4.1 - Síntese de análise dos principais pontos que afetam o carregamento de um website ....	56
Tabela 4.2 - Síntese dos principais pontos de melhorias para o carregamento do website R7 .....	56
Tabela 4.3 - Síntese de análise dos principais pontos que afetam a legibilidade de um website .....	57
Tabela 4.4 - Síntese dos principais pontos de melhorias para a legibilidade do website da Veja .....	58
Tabela 4.5 - Síntese de análise dos principais pontos de uso de flashes em um website .....	60
Tabela 4.6 - Síntese dos principais pontos de melhorias do uso da ferramenta flash no website Martini .....	60
Tabela 4.7 - Síntese de análise dos principais pontos de uso do logotipo em um website .....	62
Tabela 4.8 - Síntese dos principais pontos de melhorias para o uso do logotipo no website G1 .....	62
Tabela 4.9 - Síntese de análise dos principais pontos de uma homepage apresentável em um website .....	63
Tabela 4.10 - Síntese dos principais pontos de melhorias da homepage do website IBGE .....	64
Tabela 4.11 - Síntese de análise dos principais pontos da estrutura de navegação em um website ..	67
Tabela 4.12 - Síntese dos principais pontos de melhorias na estrutura de navegação do website R7 .....	67
Tabela 4.13 - Síntese de análise dos principais pontos de uso de um logotipo em um website .....	69
Tabela 4.14 - Síntese dos principais pontos de melhorias no uso do logotipo no website Gazeta do Litoral .....	69
Tabela 4.15 - Síntese de análise dos principais pontos de uso do campo de pesquisa em um website .....	70
Tabela 4.16 - Síntese dos principais pontos de melhorias no uso da barra de pesquisa nos websites do Guaraná Dolly, Stock Car e Coca-Cola .....	70
Tabela 4.17 - Síntese de análise dos principais pontos de uso de anúncios e pop-ups em um website .....	72
Tabela 4.18 - Síntese dos principais pontos de melhorias para o uso de anúncios e pop-ups no website do RD1 .....	72
Tabela 4.19 - Síntese de análise dos principais pontos de uso de títulos nas páginas em um website .....	73
Tabela 4.20 - Síntese dos principais pontos de melhorias para os títulos das páginas nos websites do IBGE e da Vivo .....	74

## LISTA DE SIGLAS

ARPA - Advanced Research and Project Agency

HTML - Hypertext Markup Language

URL - Uniform Resource Locator

XML - Extensible Markup Language

NBR - Norma Brasileira

ISO - International Organization for Standardization

IHC - Interação Homem-Computador

EBC - Empresa Brasil de Comunicação

IBGE - Instituto Brasileiro de Geografia e Estatística

iG - Internet Group

MTV - Music Television

UFPA - Universidade Federal do Pará

TCE/RJ - Tribunal de Contas do Estado do Rio de Janeiro

UFMG - Universidade Federal de Minas Gérias

USP - Universidade de São Paulo

## SUMÁRIO

<b>1 INTRODUÇÃO</b>	<b>13</b>
1.1 Motivação	15
1.2 Objetivos	15
1.3 Organização do Trabalho	15
<b>2 PROCESSO DE ENGENHARIA WEB</b>	<b>17</b>
2.1 Web	17
2.2 Aplicação Web	20
2.3 Engenharia Web	23
2.3.1 Comunicação	24
2.3.1.1 Formulação	24
2.3.1.2 Elicitação	25
2.3.1.3 Negociação	26
2.3.2 Planejamento	26
2.3.2.1 Estimação	27
2.3.2.2 Análise de risco	28
2.3.2.3 Escalonamento	29
2.3.2.4 Monitoração	30
2.3.3 Modelagem	30
2.3.3.1 Análise	31
2.3.3.2 Projeto	33
2.3.4 Construção	35
2.3.4.1 Teste	36
2.3.5 Implantação	39
<b>3 QUALIDADE NA WEBAPP</b>	<b>39</b>
3.1 Características de qualidade segundo a norma NBR ISO/IEC 9126	40
3.1.1 Funcionalidade	41
3.1.2 Confiabilidade	42

3.1.3 Usabilidade .....	42
3.1.4 Eficiência .....	43
3.1.5 Manutenibilidade .....	44
3.1.6 Portabilidade .....	45
<b>3.2 Avaliação de Usabilidade .....</b>	<b>46</b>
3.2.1 Avaliação Heurística .....	46
<b>3.3 Pontos da Usabilidade .....</b>	<b>48</b>
3.3.1 Acessibilidade .....	48
3.3.2 Identidade .....	49
3.3.3 Navegabilidade .....	50
3.3.4 Conteúdo .....	52
<b>4 ESTUDO DE CASO – USABILIDADE DE WEBAPPS .....</b>	<b>54</b>
4.1.1 Acessibilidade .....	54
4.1.2 Identidade .....	60
4.1.3 Navegabilidade .....	65
4.1.4 Conteúdo .....	70
<b>5 CONCLUSÃO .....</b>	<b>75</b>
REFERÊNCIAS .....	77

## 1 INTRODUÇÃO

A internet tem um papel fundamental no mundo globalizado, visto que, é possível trocar informações com pessoas no mundo inteiro e de inúmeras maneiras. Cabe salientar que, surgiu a partir de um projeto da agência norte-americana *Advanced Research and Project Agency* (ARPA), cujo seu objetivo era apenas conectar os computadores dos seus departamentos de pesquisa. Ou seja, tinha apenas como campo o ambiente acadêmico, tal que, a Internet nasceu a partir da ARPANET que interligava quatro instituições de ensino: Universidade da Califórnia; LA e Santa Bárbara; Instituto de Pesquisa de Stanford e Universidade de Utah, tendo início em 1969.

No entanto, só a partir de 1993, a Internet deixou de ser uma instituição de natureza apenas acadêmica e passou a ser explorada comercialmente, tanto para construção de novos *backbones* por empresas privadas como para fornecimento de serviços diversos.

Dessa forma, a internet passou a ser vista como uma rede que conecta milhares de computadores e permite que os mesmos consigam se comunicar e trocar informações através de protocolos que usam em comum. Segundo Gallo (2003), “a internet é como uma coleção de redes de computadores baseadas em um conjunto padrão de rede”.

Sabe-se, que a internet possibilita muitas coisas, como por exemplo, obter informações em tempo real de lados opostos do planeta que há alguns anos parecia impossível de ser realizado, mas hoje é algo normal. Um bom exemplo foi a Guerra do Iraque que começou em 20 de março de 2003, ou seja, foi um conflito que ocorreu em outro continente mais foi acompanhado pelo mundo todo através da internet, que permitiu essa troca de informação de maneira rápida.

Para que essa troca de informações ocorra, os usuários interagem com as aplicações Web por meio dos navegadores web. Fornecendo informações aos servidores, os quais processam e geram dinamicamente as repostas aos usuários. Assim, a troca de informações entre usuários e Web é bidirecional. Só que essas informações visualizadas pelos usuários podem ser informações enganosas e duvidosas não transmitindo assim, a credibilidade para quem visita a *WebApp*.

Portanto, utilizando esta perspectiva, o foco deste trabalho está associado ao desenvolvimento da qualidade nas aplicações Web – aperfeiçoar um conjunto de

atributos visando manter o seu nível de compreensão, entendimento dentro de condições estabelecidas do sistema melhorando a troca de informações entre usuários e Web – os tornando capazes de serem compreendidos, aprendidos, operados e atraentes.

## 1.1 Motivação

Apresentar recomendações relacionadas à qualidade nas aplicações web, visto que, há uma grande quantidade de sites ruins existentes hoje do ponto de vista dos usuários. Com isso, através da análise de aspectos de usabilidade, se visa aperfeiçoar ainda mais, a troca de informações entre usuários e aplicações web melhorando sua usabilidade.

## 1.2 Objetivos

Este trabalho tem como objetivo principal propor um conjunto de recomendações para melhorar a qualidade das aplicações web.

Os objetivos específicos são listados:

a) Evidenciar a estrutura para o desenvolvimento de aplicações web diferenciando-a da utilizada nas aplicações convencionais.

b) Identificar os problemas, tais como: tempo de carregamento do site, uso da ferramenta *flash*, títulos descritivos nas páginas e etc., com intuito de verificar quais problemas podem comprometer a usabilidade numa aplicação web.

c) Apresentar soluções para minimizar os problemas relacionados à qualidade nas aplicações web melhorando sua usabilidade.

## 1.3 Organização do Trabalho

A apresentação detalhada dos aspectos deste trabalho está distribuída da seguinte forma:

Neste primeiro capítulo foi realizada a introdução deste trabalho.

O segundo capítulo apresentará as principais diferenças entre desenvolver uma aplicação convencional e desenvolver uma aplicação web. E, além disto, será apresentado o arcabouço de desenvolvimento das aplicações web, ou seja, um conjunto de atividades que são necessárias para o seu desenvolvimento.

O terceiro capítulo apresenta como as aplicações web podem alcançar a qualidade de software. E também, como identificar os principais problemas que comprometem a usabilidade.

O quarto capítulo apresenta um estudo de caso onde serão analisados



diversos pontos de usabilidade e, dentre estes pontos, os principais problemas que podem ser observados nas diversas aplicações web. Com intuito de mostrar as possíveis soluções que podem ser feitas nas aplicações web, melhorando sua usabilidade.

O quinto capítulo apresenta a conclusão do trabalho, ressaltando as dificuldades e vantagens durante a elaboração do estudo de caso.

## 2 PROCESSO DE ENGENHARIA WEB

Algumas pessoas acreditam que uma organização pode definir totalmente um processo para desenvolvimento de um software. No entanto, as coisas não funcionam dessa maneira. É preciso que desenvolvedores e clientes tenham constante contato cujo objetivo é definir uma sequência de atividades interligadas tais como: comunicação, planejamento, modelagem e etc. que serão necessárias para o desenvolvimento do software.

### 2.1 Web

A Web também conhecida como *www* ou *world wide web*, significa um sistema de informações ligadas através de hipermídia que permite o usuário acessar uma infinidade de conteúdos através da internet por meio de um navegador (*browser*). Tornando-se assim, uma tecnologia indispensável para negócios, comunicação, comércio e etc.

Dessa forma, pode-se verificar que, a web está cada vez mais entrando no cotidiano das pessoas. Visto que, ela mudou as formas como se compra produtos (comércio eletrônico), entende-se o mundo (portais), se vê as notícias (mídia on-line) e ir à escola (aprendizado on-line). Isso se deve a facilidade de acesso à internet e sua velocidade que está cada vez mais rápida, pois, mesmo pessoas que não tenham internet em suas casas podem acessar a web através de *lan-houses*. Ou até então, acessar a web através de celulares, que estão cada vez mais contando com esse recurso.

Com o advento da *www*, houve inicialmente um grande crescimento de empresas denominadas *pontocom*, no qual foram criadas principalmente para explorar a comercialização de serviços ou produtos na Internet. Ou seja, baseavam seu modelo de negócios em número de acessos, visibilidade ou criação de comunidades virtuais. A princípio, essas empresas não tinham a necessidade de grandes instalações e o investimento necessário recaía apenas na tecnologia. Porém, foi observada como tendência a incorporação dos negócios eletrônicos, isto é, englobar o relacionamento e atividades de colaboração com clientes, com fornecedores e transações através da web pelas empresas.

Quem utiliza a internet com certa frequência há alguns anos pode perceber

certas significativas mudanças na maioria das páginas que visita. Até pouco tempo atrás, os sites estavam disponíveis como livros: as pessoas acessavam liam o que era de interesse e simplesmente fechavam a página. No máximo, a comunicação com os desenvolvedores pelo conteúdo era através de um e-mail. Assim como a humanidade, a web também evoluiu.

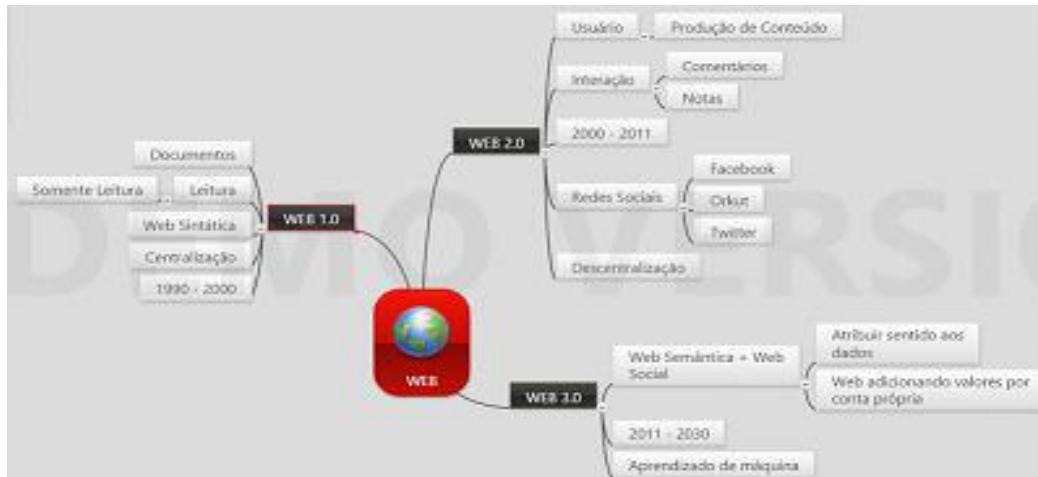


Figura 2.1: Evolução da web. Fonte: Aplicação de Software Social 2011<sup>1</sup>.

Como se pode verificar na Figura 2.1, a web possui três fases evolutivas. A primeira fase, conhecida como Web 1.0, é a internet como ela surgiu, isto é, consistia num veículo de informação onde os consumidores (usuários) tinham o poder de apenas consumir o conteúdo colocado pelas empresas. Havia uma grande quantidade de informação disponível com pouco grau de interação, o usuário não podia alterar o conteúdo. Portanto, a principal característica desta fase é a estaticidade. Mesmo assim, não impediu o crescimento e a popularização da rede, que depois se adaptou às diferentes demandas de usuários e exigências corporativas sendo disseminada por todo o planeta.

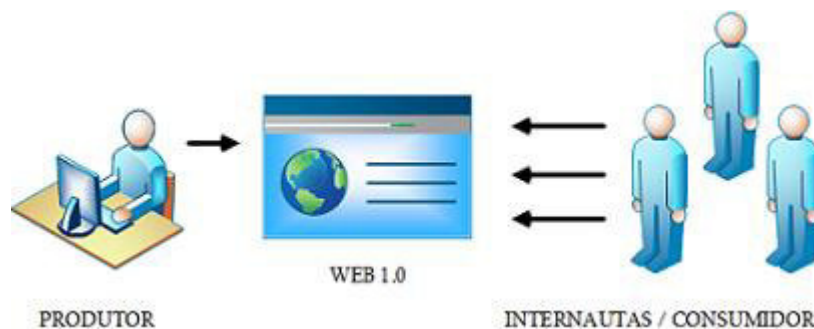


Figura 2.2: Web 1.0. Fonte: Agregando Conhecimento e Informação<sup>2</sup>.

<sup>1</sup> Disponível em: <<https://apssocial.wordpress.com/category/atividades-academicas/projeto-ii-principios-basicos-da-web-2-0-e-tendencias-de-evolucao/>>. Acesso em 03 jan. 2015.

<sup>2</sup> Disponível em: <<https://agregcom.wordpress.com/2014/09/09/a-evolucao-da-web-1-0-x-2-0-x-3-0/>>. Acesso

Com o passar do tempo, a *Hypertext Markup Language* (HTML) foi aumentada com ferramentas e tecnologia. Dessa forma, chega-se a segunda fase, conhecida como Web 2.0 ou “Web Social” conforme a Figura 2.3. Nesse momento, a internet se popularizou em todo mundo. Com isso, passaram a referenciar as tendências e os novos rumos que a internet tomou ao adotar características que fornecem aos usuários mais criatividade, informação e colaboração entre os usuários. Portanto, a principal característica desta fase é o conteúdo dinâmico gerado por usuários. Visto que, ele deixou de ser apenas um espectador e passou a fazer parte do movimento, podendo assim, alterar ou acrescentar conteúdo aos sites. Isso tudo, se deve, ao surgimento de serviços de *wikis*, vídeos e blogs, e ainda as redes sociais como o *Orkut*, *Twitter* e *Facebook*.



Figura 2.3: Web 2.0. Fonte: Agregando Conhecimento e Informação<sup>3</sup>.

Entretanto, como tudo tem seu lado bom e ruim, houve uma explosão de informação e o excesso de informação inútil.

Além da falta de estrutura, com o crescimento desordenado da rede, não existe ainda [...] padrões universais para a organização dos documentos, como os que existem para bibliotecas e arquivos. A indexação de páginas Web é feita de maneira precária (através de informações chamadas *metatags* que são adicionadas ao documento pelo criador das páginas) ou, na maioria das vezes, elas simplesmente não são indexadas. São poucos os recursos organizados por profissionais com o conhecimento de técnicas de indexação e catalogação (CENDÓN, 2007, p. 296).

Com intuito de controlar o excesso de informação, pode-se dizer que a web

em 03 jan. 2015.

<sup>3</sup> Disponível em: <<https://agrecom.wordpress.com/2014/09/09/a-evolucao-da-web-1-0-x-2-0-x-3-0/>>. Acesso em 03 jan. 2015.

chegou à sua terceira fase, conhecida como Web 3.0 ou “Web Inteligente”. Nesta fase, a web visa viabilizar a criação de categorias para a representação dos recursos informacionais existentes e disponíveis no ambiente web. Dessa forma, os conteúdos online estarão organizados de forma semântica, muito mais personalizada para cada internauta. Convém ressaltar que toda essa informação será obtida de forma organizada, não somente para que os humanos compreendam, mas também para as máquinas, assim elas podem nos ajudar respondendo pesquisas e perguntas com respostas concretas. Essa fase também é vista como, uma internet mais próxima da inteligência artificial. Portanto, a principal característica desta fase é a metainformação.

Sabendo disso, o veículo que adquire a informação, a estrutura, monta uma apresentação empacotada e a entrega é chamada de Aplicação Web<sup>4</sup> (WebApp).

## 2.2 Aplicação Web

Nos primeiros dias do *World Wide Web* (1990 – 1995), as aplicações convencionais consistiam em um conjunto de arquivos de hipertexto ligados, que apresentavam informações usando texto e gráficos simples. No entanto, o crescimento da internet e a necessidade de serviços rápidos e instantâneos têm ocasionado uma demanda significativa no desenvolvimento de aplicações que utilizam a web como ambiente operacional, caracterizadas principalmente pelos conceitos de interação, compartilhamento e comunicação. Daí surgiram as aplicações web ou *WebApps*.

Segundo Nielsen (2007), a experiência de um usuário de internet não é como na televisão, pois a televisão passa seu conteúdo passivamente, onde seu usuário não interage com o conteúdo, já em uma aplicação web o usuário participa ativamente, ele está sempre no controle, pois a interação é necessária.

Além disto, as aplicações desenvolvidas para web estão se tornando cada vez mais complexas devido às características de negócio da aplicação a ser desenvolvida. E também, não apenas oferecem funcionalidades isoladas ao usuário final, mas também foram integradas a banco de dados e aplicações corporativas e governamentais.

---

<sup>4</sup> Outras denominações são dadas como: Websites, Web-Based Information Systems (WBIS), Web Information Systems (WIS), Sistemas Web e Sistemas de Informação Web.

Segundo Pressman (2009), as seguintes características podem ser encontradas na grande maioria das *WebApps*:

- **Rede Intensiva:** as aplicações Web utilizam recursos de rede por natureza. Visto que, está em uma rede e deve atender as necessidades de diversos usuários. Dessa forma, podem estar disponíveis na Internet (permitindo a comunicação com o mundo todo), Intranet (permitindo a comunicação em uma organização) ou Extranet (permitindo a comunicação inter-redes).

- **Dirigido a conteúdo:** em muitos casos, a função primária das aplicações Web é usar hipermídia para apresentar textos, gráficos e vídeo para os usuários.

- **Evolução contínua:** ao contrário das aplicações convencionais que evoluem através de uma série de versões planejadas e lançadas em determinados intervalos de tempo, as aplicações Web evoluem continuamente. Visto que, é usual ser atualizado a cada hora.

Baseado nos modelos de navegação, Figura 2.4, as aplicações convencionais consolidam o modelo de janelas em cascatas, onde uma tela da aplicação leva a outra em uma direção linear. Ou seja, possuindo uma janela pai e as janelas filhas, e onde as ações realizadas se restringem apenas ao escopo da aplicação.

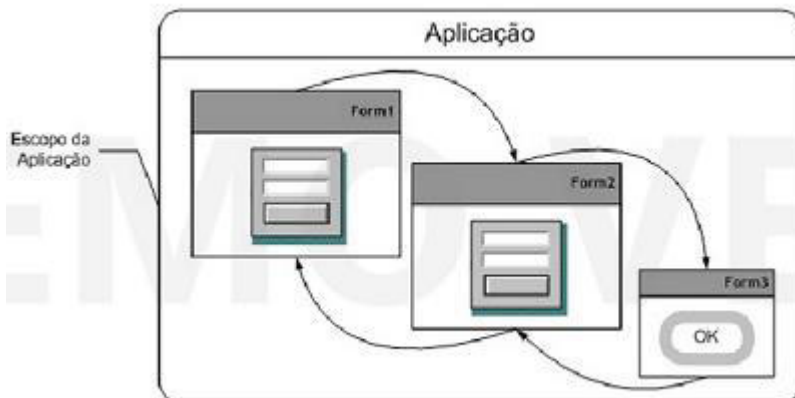


Figura 2.4: Representação do modelo de janelas em cascata. Fonte: MATOS, A.; BRITTO, T, 2007.

Já as *WebApps*, possuem um modelo de navegação em rede, ou teia, como também é chamado, Figura 2.5. Mesmo existindo a hierarquia entre determinadas páginas, diversos elementos contribuem para a formação de um fluxo de navegação não linear, como os botões de voltar e avançar no navegador, o acesso direto através da URL ou links contidos em diversas páginas que apontam para o mesmo caminho. Nas *WebApps* é possível, inclusive, concluir um processo referenciando

uma aplicação externa, um outro *website*, saindo do escopo daquela aplicação.

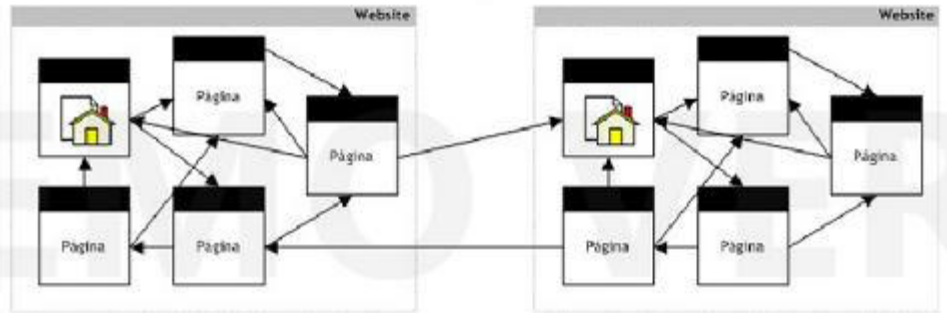


Figura 2.5: Fluxo de navegação em rede (teia). Fonte: MATOS, A.; BRITTO, T, 2007.

Tendo em vista estas observações e com base em algumas considerações de Murugesan [MATOS, A.; BRITTO, T. apud Murugesan and Ginige, 2004], é possível identificar as principais diferenças entre desenvolver uma aplicação convencional e desenvolver uma aplicação Web conforme a Tabela 2.1, são elas:

Tabela 2.1 - Comparativo entre aplicação convencional e aplicação web. Fonte: MATOS, A.; BRITTO, T, 2007.

<b>Aplicação Convencional</b>	<b>Aplicação Web</b>
Consolidou o modelo de janela em cascata	Modelo de navegação e interação por hipertexto.
Padrão de interfaces (cores, iconografia, posicionamento dos elementos) semelhante.	Interface possui características distintas dependendo do foco do projeto.
Geralmente é desenvolvido com base em apenas uma linguagem de programação.	Envolve técnicas e procedimentos com uso de várias linguagens.
Aplicações homogêneas, desenvolvidas para rodar em ambiente específico.	Aplicações heterogêneas: é mais difícil definir perfil de hardware, plataforma e largura de banda.
Linearidade e consistência de navegação: o usuário não consegue acessar uma tela de aplicação diretamente sem interagir com um componente, por exemplo, um botão.	Usuários podem acessar as páginas através da URL.
Precisa estar disponível apenas um tipo de dispositivo.	Precisa estar disponíveis em diferentes formatos e dispositivos e suportar hardware, software e redes com grande variedade de velocidade de acesso.

Sabendo disso, ao tentar desenvolver uma *WebApp* deve-se adotar umas das técnicas de desenvolvimento. A princípio se tem a do tipo *ad hoc* (conhecida como acertar-ou-errar), que é baseada na escola antiga. Isto é, informalidade, urgência, intuição e arte são as forças motrizes por trás das atividades da maioria dos desenvolvedores de sistemas que utilizam esta técnica. Afinal, *informalidade* leva a um ambiente de trabalho fácil – aquele em que você pode fazer o que quiser. *Urgência* leva a ação e tomada de decisão rápida. *Intuição* é uma qualidade que

permite sentir à sua maneira em situações complexas. E *arte* leva a forma estética e função – a algo que agrada àqueles que a encontraram. Se todas as coisas correrem bem, nascerá uma *WebApp* significativa.

Mas as coisas nem sempre funcionam bem. Com isso, a *WebApp* pode não fazer o que deveria; ela pode ser entregue atrasada ou nem ser entregue; ou ela pode ser difícil ou impossível de se corrigir.

Dessa forma, existe uma alternativa à técnica da escola antiga – que reduz (mas não elimina) o risco e tem uma probabilidade mais alta de sucesso quando *WebApps* de qualidade industrial precisam ser montadas. A alternativa é *engenharia Web* (WebE).

### 2.3 Engenharia Web

Ao mesmo tempo em que adota muitos princípios da Engenharia de Software, a Engenharia Web incorpora novas abordagens, metodologias, ferramentas e técnicas para atender os requisitos exclusivos dos sistemas para a Web.

Segundo Pressman (2009, p.11), a engenharia Web propõe um arcabouço ágil, porém disciplinado, para a montagem de *WebApps* de qualidade industrial.

A partir desta citação, deve-se saber que um *arcabouço* estabelece o alicerce para um processo de engenharia Web, identificando um pequeno número de atividades de arcabouço que se aplicam a todos os projetos de *WebApp*, independentemente de seu tamanho e complexidade. Sendo que, o arcabouço precisa ser adaptado às características específicas do projeto e às pessoas que especificarão os requisitos e realizarão o trabalho.

Por essa razão, as atividades de arcabouço ocorrerão repetidamente à medida que cada incremento de *WebApp* for desenvolvido e entregue.

Dessa forma, as atividades de WebE referente à Figura 2.6, se aplicam à grande maioria dos projetos de *WebApp*, tais como: comunicação, planejamento, modelagem, construção e implantação.





Figura 2.6: Arcabouço da WebE. Fonte: PRESSMAN, R.S. 2009.

Convém ressaltar que, cada atividade do arcabouço é preenchida por um conjunto de *ações de engenharia Web* – uma coleção de tarefas relacionadas, que produz um produto de trabalho.

### 2.3.1 Comunicação

O processo de WebE começa com a atividade de comunicação. Com referência à Figura 2.6, a comunicação estabelece o “destino” para um projeto de *WebApp*, ou seja, serve como ponto de entrada. É o local onde se engajam uma série de ações de WebE que: (1) perguntam e respondem a um conjunto de questões sobre a *WebApp* e seu contexto de negócio; (2) elicitam requisitos que servirão de base para as atividades seguintes; e (3) negociam as necessidades contra as realidades do tempo. Essas ações são chamadas de *formulação*, *elicitação* e *negociação*, respectivamente.

A atividade de comunicação durante a primeira interação começará pela ação denominada de *formulação*.

#### 2.3.1.1 Formulação

A formulação é uma ação da WebE que além de identificar as necessidades de negócio e estabelecer os objetivos da *WebApp*, é também responsável por definir as principais características e estabelecer uma base para a ação de elicitação seguinte. A formulação permite que os interessados e a equipe de WebE estabeleçam um conjunto comum de metas e objetivos para criação de cada

incremento da *WebApp*.

Dessa forma para que se tenham os objetivos estabelecidos. É preciso primeiramente identificar quem será alvo das perguntas, visto que, se interagirá com diversos interessados diferentes, cada um com um ponto de vista diferente.

Logo, a comunicação pode ser realizada junto à formulação usando-se uma ou mais das seguintes técnicas:

- **Grupo de foco tradicional:** caracteriza-se por reunir um pequeno grupo (normalmente, menos de 10 pessoas) de usuários finais. Cujo objetivo é discutir a *WebApp* a ser desenvolvida e, entender melhor os requisitos do sistema.

- **Grupo de foco eletrônico:** caracteriza-se por reunir um grupo maior de usuários finais. Logo, todos os usuários podem participar ao mesmo tempo e mais informações podem ser coletadas em um período de tempo mais curto.

- **Pesquisas iterativas:** consiste em uma série de pesquisas breves, focadas em usuários finais e solicitando respostas para questões específicas sobre a *WebApp*. As respostas são analisadas e usadas para ajustar a próxima pesquisa.

- **Pesquisa exploratória:** consiste em uma pesquisa baseada na Web vinculada a uma ou mais *WebApps* que possuem usuários semelhantes. No qual, os usuários seguem um link até a pesquisa e respondem a uma série de perguntas.

- **Criação do cenário:** seleciona usuários finais que são solicitados a criar cenários de uso<sup>5</sup> que descrevem interações específicas com a *WebApp*.

À medida que informações de vários pontos de vista são coletados, os requisitos podem ser inconsistentes ou podem entrar em conflito uns com os outros. Logo, é preciso categorizar todas as informações dos interessados. Portanto, a transição da formulação para elicitación ocorre quando o nível de abstração se move das questões gerais de negócios para requisitos mais específicos.

### 2.3.1.2 Elicitación

Passada a etapa de formulação, chega-se a elicitación dos requisitos. Cujo objetivo é descobrir ou levantar com os interessados, quais serão as funcionalidades da *WebApp* e como deverá oferecê-las. Parece uma tarefa fácil, mas na realidade, não é. Pois os clientes, por vezes, não sabem o que realmente querem, tornando

---

<sup>5</sup> Consiste em um ou dois parágrafos narrativos que descrevem como um usuário final interagiria com uma função da *WebApp*.

assim difícil a interpretação das reais necessidades da *WebApp*.

Cabe ressaltar que, esta identificação dos requisitos pode ocorrer por meio de reuniões. Cujo objetivo é realizar o levantamento de requisitos das informações obtidas da ação anterior. Tendo por sua vez, a realização de quatro técnicas para detecção:

1. Definir categorias de usuário e desenvolver descrições para cada categoria.
2. Refinar o conteúdo e a funcionalidade usando as listas que cada usuário preparou.
3. Considerar restrições e questões de desempenho específica.
4. Escrever cenários de uso para cada classe de usuário.

Portanto, a transição da elicitação para negociação ocorre quando os requisitos funcionais estão definidos e é preciso resolver detalhes com os interessados.

#### 2.3.1.3 Negociação

A negociação corresponde a última ação da atividade de comunicação, ou seja, é necessário que haja uma negociação dos requisitos coletados durante a fase de elicitação porque não é difícil ocorrer de clientes e usuários finais pedirem mais do que pode ser conseguido, considerando o fato de que os recursos são limitados em qualquer projeto (PRESSMAN, 2006). Dessa forma, pode-se afirmar que a elicitação é uma atividade que caminha de mãos dadas com a atividade de negociação de requisitos. Além disto, acontece muitas vezes dos requisitos conflitarem ou não coincidirem nas visões dos clientes do projeto. Por fora, também é preciso que os engenheiros do sistema acordem com estes clientes qual será o grau de prioridade das funções do software.

Portanto, a transição da negociação para o planejamento ocorre quando todos os detalhes dos requisitos estão resolvidos e, é preciso que ocorra o planejamento de seu desenvolvimento.

#### 2.3.2 Planejamento

Segundo a visão de Dilbert do planejamento de projetos [Pressman apud Dilbert, 2007] essa continua sendo uma atividade muito necessária quando as

*WebApps* são construídas. Visto que, não deve ser uma atividade única realizada no início do projeto. Ao contrário, planejamento é uma atividade recorrente na condução de projetos. A pergunta mais crítica que todo time de desenvolvimento de sistemas tem que responder é se está, de fato, construindo, o que deve ser feito, algo que tenha valor.

Portanto, o planejamento é uma atividade iterativa onde o time busca aperfeiçoar uma solução inicialmente concebida. Mas é importante reconhecer: (1) as limitações e incertezas inerentes às estimativas de custo e prazo para se alcançar um dado objetivo; (2) verificar os riscos durante o desenvolvimento; (3) selecionar e escalonar as tarefas a serem realizadas; e (4) acompanhar o projeto. Essas ações, com referência à Figura 2.6, são chamadas de *estimação, análise de risco, escalonamento e monitoração*, respectivamente.

A atividade de planejamento inicia-se coletando as informações sobre o escopo do projeto procurando entendê-las. Essa ação por sua vez é denominada de *estimação*.

### 2.3.2.1 Estimação

Quando se inicia um novo projeto, não existe uma forma exata de determinar quando será concluído. Pois, existem muitas incertezas, o escopo não está definido, os requisitos estão pouco entendidos e elaborados e a arquitetura ainda não foi concebida.

Neste caso entra em prática a estimativa, que é uma ação da WebE que consiste em coletar as informações sobre o escopo e os recursos do projeto e então é desenvolvido a duração que será considerada no cronograma<sup>6</sup>. Segundo Pressman (2009, p. 67), o escopo é definido respondendo-se às seguintes questões:

**Contexto:** Como a *WebApp* se encaixa em um contexto de negócios e que restrições são impostas por esse contexto?

**Objetivos da informação:** Que objetos de conteúdo visíveis ao cliente são usados e produzidos pelo incremento da *WebApp*?

**Funcionalidade:** Que funções são iniciadas pelo usuário final ou invocadas internamente pela *WebApp* para atender aos requisitos definidos nos cenários de

---

<sup>6</sup> É uma atividade que aloca o esforço estimado para tarefas de WebE específicas durante a linha de tempo planejada (duração) para construção de um incremento.

uso?

**Restrições e desempenho:** Que restrições técnicas e de ambiente causarão impacto nas atividades de arcabouço que se seguem? Que questões de desempenho especiais (incluindo questões de segurança e privacidade) exigirão esforço de projeto e construção?

É importante ressaltar que os seguintes fatores, podem fazer com que as estimativas deem errado, são eles:

- excesso de alterações de requisitos;
- otimismo;
- requisitos imprecisos e incompletos.

Portanto, a transição da estimação para análise de risco ocorre quando o escopo do projeto e os requisitos estão compreendidos e, é preciso considerar os riscos durante o desenvolvimento.

### 2.3.2.2 Análise de risco

A análise de risco se encontra no planejamento do projeto, porém é considerada uma atividade guarda-chuva<sup>7</sup>. Visto que, compreende uma série de tarefas que ajudam uma equipe de WebE a entender e gerenciar os problemas que podem assolar um projeto de *WebApp*. Convém ressaltar que, um risco é um problema em potencial – pode ocorrer no futuro e ser concretizado, ou não. Mas, independentemente do resultado, é fundamental avaliar sua probabilidade de ocorrência e estimar seu impacto.

Dessa forma, com o objetivo de melhorar a análise de risco, pode ser útil fazer o agrupamento por categorias. Através dessas informações pode-se formar uma base mais consolidada de informações, sendo possível ter uma visão melhor sobre as semelhanças, grau de completude de seus esforços e outras informações que facilitarão a identificação de riscos. Tendo as seguintes categorias:

- **Riscos de pessoas:** são problemas que podem ser rastreados diretamente em alguma ação ou falha humana. Por exemplo, é provável que o incremento da *WebApp* exija componentes desenvolvidos usando XML, mas a equipe atualmente não tem ninguém com experiência com XML.

---

<sup>7</sup> Aplicáveis durante todo o processo de software.

- **Risco de produto:** pode ser rastreados a problemas associados ao conteúdo, funções, restrições ou desempenho da *WebApp*. Por exemplo, um risco que poderia ter impacto sobre o prazo apertado para o incremento da *WebApp* é descoberto apenas depois de sua construção tiver sido iniciada.

- **Riscos de processo:** são problemas que estão ligados às ações e tarefas de arcabouço que foram escolhidas pela equipe. Por exemplo, a equipe decidiu realizar um modelo de análise completo e especificou ações e tarefas para realizar isso, com isso, houve muito tempo gasto nesta atividade.

Feita a identificação, o próximo passo é analisar os riscos listados, um por um detalhadamente, a fim de obter o grau de periculosidade de cada risco.

Portanto, a transição da análise de risco para escalonamento ocorre quando todos os riscos estão avaliados para o desenvolvimento do projeto. E então, é preciso selecionar as tarefas a serem realizadas.

### 2.3.2.3 Escalonamento

Sabe-se que o desenvolvimento de um software está propício a diversos problemas. Sendo que, a realidade de qualquer projeto é que centenas de tarefas pequenas precisam ocorrer para realizar uma meta maior. Enquanto outras tarefas se encontram no “caminho crítico”. Se essas tarefas críticas atrasarem, a data de término do incremento da *WebApp* será colocada em risco.

Neste caso, entra em prática o escalonamento, que é uma ação da WebE que lista todas as ações e tarefas de WebE para um incremento; identifica as tarefas que são críticas e depois rastreia seu progresso para garantir que qualquer atraso será reconhecido. Para conseguir isso, é preciso determinar o tempo de desenvolvimento necessário das tarefas estabelecidas através de um cronograma. Porém, é importante observar que o cronograma geral da *WebApp* evolui com o tempo.

Dessa forma, a equipe de WebE consulta e negocia com os interessados e desenvolve um cronograma macroscópico que identifica todos os incrementos da *WebApp* e projeta as datas em que cada um será implantando

Portanto, a transição do escalonamento para monitoração ocorre quando todos os incrementos estão identificados e estabelecidos seus prazos de desenvolvimento. E então, é preciso acompanhar o seu desenvolvimento.

#### 2.3.2.4 Monitoração

Antes que o incremento de um projeto Web esteja disponível, um engenheiro Web deve saber a situação deste incremento no cronograma de desenvolvimento. Ou seja, é nesta fase que entra a monitoração, é onde o engenheiro Web precisará acompanhar o progresso enquanto um incremento da *WebApp* está sendo desenvolvido.

Uma maneira de acompanhar o progresso enquanto um incremento da *WebApp* está sendo desenvolvido é realizar uma enquete com a equipe de WebE para determinar quais atividades de arcabouço foram completadas. Porém, essa técnica pode não ser confiável porque o término de uma atividade de arcabouço é uma indicação do progresso somente se produtos de trabalho apropriados tiverem sido desenvolvidos.

Outra técnica é determinar quantos cenários do usuário foram implementados e quantos cenários do usuário restam a serem implementados. Isso oferece uma indicação do grau relativo de completude do incremento do projeto.

Portanto, a transição da monitoração para modelagem ocorre quando se tem total controle do incremento a ser desenvolvido. E então, é preciso criar um modelo explicando as características e os comportamentos do produto.

#### 2.3.3 Modelagem

Sabe-se que a palavra escrita ou falada não é um modo preciso de representar o conteúdo ou as funções a serem entregues por uma *WebApp*. Pois, a linguagem natural pode ser ambígua, contraditória ou confusa.

Neste caso, entra em prática a modelagem, que é uma atividade de construir modelos que expliquem as características ou o comportamento de um software através da utilização de notações gráficas e textuais para representar as partes essenciais de um sistema.

Um modelo construído adequadamente tem o potencial de permitir que os engenheiros Web entendam os requisitos de negócios e do usuário, e o projeto que tem como finalidade representá-los. Dessa forma, a modelagem visa: (1) entender a natureza do problema a ser resolvido e a "forma" da *WebApp* que lhe permitirá resolver esse problema; e (2) trata de compreender a estrutura interna da *WebApp*

sendo desenvolvida. Essas ações, com referência à Figura 2.6, são chamadas de *análise e projeto*, respectivamente.

É importante ressaltar que durante a modelagem pode-se encontrar os seguintes desafios, são eles:

- deficiências em relação a suporte ferramental, isso se deve a flexibilidade para adaptação ao modo de trabalho das organizações;
- dificuldade de modelar, isso se deve ao fato dos requisitos estarem especificados de forma incorreta;
- aplicação cada vez mais complexa.

Portanto, a atividade de modelagem inicia-se ajudando a compreender com clareza a natureza do problema a ser desenvolvido e resolvido. Essa ação é denominada de análise.

#### 2.3.3.1 Análise

Sabe-se que o processo de WebE precisa ser ágil, por isso, os desenvolvedores Web alegam a necessidade da análise de requisitos para *WebApp*. Visto que, a análise do problema exige tempo, mas resolver o problema errado leva ainda mais tempo. Logo, a finalidade da modelagem de análise é refinar e modelar os requisitos da *WebApp* e, como consequência, estabelecer uma compreensão muito mais profunda da natureza da *WebApp* a ser construída. Segundo Steve Franklin [Pressman apud Steve Franklin, 2003]:

Frequentemente, os desenvolvedores vão diretamente à fase de codificação sem realmente entender o que estão tentando construir ou como eles desejam construí-lo. A codificação no lado do servidor normalmente é feita *ad hoc*, as tabelas de banco de dados são acrescentadas conforme a necessidade e a arquitetura evoluem de uma maneira às vezes não intencional. Mas alguma modelagem e engenharia de software disciplinada podem tornar o processo de desenvolvimento de software muito mais estável e garantir que o sistema Web possa ter uma manutenção mais fácil no futuro.

Dessa forma, os engenheiros Web devem entender o papel da análise bem o suficiente para saber exatamente o quanto dela precisa ser realizada. O grau ao qual a modelagem de análise deve ser aplicada depende dos seguintes fatores (Pressman, 2009, p. 124):

- Tamanho e complexidade do incremento da *WebApp* é grande e/ou



complexa;

- Número de interessados é grande (a análise pode ajudar a identificar requisitos em conflitos originários de fontes diferentes);
- Tamanho da equipe WebE é grande;
- Grau aos quais os membros da equipe de WebE trabalharam junto antes (a análise pode ajudar a desenvolver um entendimento comum do projeto).

Cabe ressaltar que, levantar requisitos e analisá-los é uma parte importante do desenvolvimento de uma *WebApp*. Mas, é preciso ter em mente as seguintes questões (Pressman, 2009, p. 127):

1. Quais são as entradas para a modelagem de análise?
2. Quais são as saídas da análise?
3. Que tarefas de análise podem-se e devem-se executar?
4. Que ferramentas pode-se usar para nos ajudar a modelar?

O processo de WebE incorpora uma atividade de comunicação que identifica a informação que se torna entrada para a modelagem de análise. Essa entrada inclui interessados e categorias de usuários, o contexto de negócios, requisitos gerais da *WebApp* e cenários de uso. Sendo representadas na forma de descrições em linguagem natural, esboços e desenhos. A análise apanha essa informação, produzindo um modelo mais rigoroso como saída que oferece uma indicação detalhada da estrutura verdadeira do problema.

Cabe ressaltar que, podem-se identificar quatro classes principais de modelos como saídas da análise, que inclui o (1) modelo de conteúdo – caracteriza-se por oferece uma visão importante dos requisitos de conteúdo para *WebApp*. Sendo composto dos seguintes elementos: *objetos de conteúdo* (por exemplo, texto, gráficos, áudio) que representam uma informação bruta usada dentro de uma *WebApp* e *classes de análise* – entidades visíveis ao usuário que são criadas e manipuladas à medida que um usuário interage com a *WebApp*; (2) modelo de interação – caracteriza-se por descrever a maneira com o usuário interage com a *WebApp*, sendo composto por: *casos de uso* – descrevem as formas como os usuários interagirão com a *WebApp*; *diagramas de sequência* - descrevem uma maneira como as ações do usuário colaboram com as classes de análise; *diagramas de estado* - descrevem o comportamento dinâmico da *WebApp* à medida que ocorre uma interação; e/ou *protótipos da interface do usuário* – consiste em implementar os principais links de navegação, visando ter uma representação física de uma

interação com o usuário na *WebApp*; (3) modelo funcional - consiste em definir as operações que serão aplicadas ao conteúdo da *WebApp* e descreve outras funções de processamento que são independentes do conteúdo, mas necessárias ao usuário final; e (4) modelo de configuração - descreve o ambiente e a infraestrutura em que a *WebApp* reside.

Portanto, a análise é onde a qualidade da *WebApp* é estabelecida. E a transição para o projeto ocorre quando é preciso compreender a estrutura interna da *WebApp* a ser desenvolvida.

### 2.3.3.2 Projeto

A criação de um projeto eficaz normalmente exige um conjunto diversificado de habilidades. Ou seja, é nesta fase que entra a modelagem de projeto, que é responsável por criar a forma da *WebApp* que foi identificada pelo modelo de análise.

Cabe ressaltar, que existem essencialmente duas abordagens básicas de projeto: o ideal artístico de expressar-se e o ideal de engenharia de solucionar um problema para um cliente. Durante a primeira década do desenvolvimento Web, muitos desenvolvedores escolheram uma abordagem baseada no ideal artístico. O projeto ocorria de maneira *ad hoc* e normalmente era realizado enquanto as páginas *HTML* eram geradas. O projeto evoluía de uma visão artística que, por sua vez, evoluía enquanto ocorria a construção da *WebApp*. No entanto, esta abordagem é aplicável somente para *WebApps* simples quando um único desenvolvedor pode projetar a arquitetura, enquanto mantém uma imagem mental detalhada do projeto.

Essa realidade nos leva à segunda abordagem - “o ideal de engenharia de solucionar um problema para um cliente”. Esta abordagem é aplicável quando o conteúdo e a função são complexos, quando o tamanho da *WebApp* abrange centenas de objetos de conteúdo, funções e classes de análise. Logo, a engenharia Web adota essa filosofia, visto que, um modelo de projeto deverá ser criado antes que a construção se inicie, reconhecendo assim que o projeto evoluirá à medida que a *WebApp* é construída.

O projeto de *WebApps* inclui o (1) projeto lógico – caracteriza-se por identificar aquelas coisas abstratas (por exemplo, comunicação de diferentes partes de uma *WebApp*) que precisam ocorrer dentro da *WebApp* a fim de atender os

requisitos dos interessados e estar conforme ao modelo de análise; e (2) projeto físico – mapeia nosso conhecimento do projeto lógico nos elementos físicos reais a serem implementados como parte da *WebApp* (por exemplo, servidores Web, servidores de banco de dados, páginas Web). A saída principal é um modelo de projeto que oferece informações suficientes para a equipe de WebE construir a *WebApp* final.

Dessa forma, o projeto Web focaliza muitos aspectos diferentes do sistema baseado na Web. Ou seja, é preciso definir um conjunto de ações de projeto que ocorrem quando um modelo de projeto completo é criado. Com referência à Figura 2.7, o projeto Web é composto por: projeto de interface, projeto estético, projeto de conteúdo, projeto de navegação, projeto arquitetural e projeto de componentes.



Figura 2.7: A pirâmide de projeto da WebApp. Fonte: PRESSMAN, R.S. 2009.

A Figura 2.7 começa com o *projeto da interface* que descreve a estrutura e a organização da interface do usuário e abrange uma representação do leiaute da tela, modos de interação e uma descrição de mecanismos de navegação. Sendo que, as melhores interfaces de *WebApp* são eficientes em sua apresentação de conteúdo e função, mas também devem ser esteticamente agradáveis. Com isso, entra em prática o *projeto estético*, que descreve o “aspecto” da *WebApp* e inclui combinação de cores, leiaute geométrico, tamanho, fonte e posicionamento de textos.

Logo, interfaces de usuário eficazes, navegação clara e funcionalidade rica são todos importantes, mas muitas *WebApps* que exibiram essas características falharam porque não tinham conteúdo significativo. Neste caso, entra em prática o *projeto de conteúdo*, que define o leiaute, a estrutura e o esboço para todo o conteúdo apresentado como parte da *WebApp*, permitindo o usuário alcançar os objetivos identificados durante a atividade de comunicação.

Para que os usuários alcancem seus objetivos, entra em prática o *projeto*

*navegacional*, que representa o fluxo de navegação entre objetos de conteúdo e para todas as funções da *WebApp*. O objetivo é ajudar os usuários experientes a alcançarem seus objetivos navegacionais mais rapidamente e fornecer a usuários inexperientes suporte de navegação adicional.

Sabe-se que o conteúdo pode ser o núcleo de uma *WebApp*, mas a funcionalidade a faz cantar e dançar. Neste caso, entra em prática o *projeto arquitetural*, que identifica a estrutura geral da *WebApp* estando ligado aos objetivos estabelecidos, ao conteúdo a ser apresentado, aos usuários que visitarão a página e à filosofia de navegação que foi estabelecida.

Convém ressaltar que, as *WebApps* modernas fornecem funções de processamento cada vez mais sofisticadas que: (1) executam processamento localizado para gerar recursos de navegação e conteúdo de forma dinâmica; (2) fornecem recursos de cálculo ou processamento de dados apropriados para o campo de aplicação da *WebApp*; (3) fornecem sofisticadas consultas e acesso a bancos de dados; e (4) estabelecem interfaces de dados com sistemas externos. Neste caso, entra em prática o *projeto de componentes*, que desenvolve a lógica de processamento detalhada para implementar os componentes funcionais que implementam uma função de *WebApp* completa.

Portanto, a modelagem de projeto é onde a qualidade da *WebApp* é incorporada. E a transição para construção ocorre quando é preciso implementar cada componente da *WebApp*.

#### 2.3.4 Construção

Independentemente do tipo de *WebApp* que precisa ser construída, uma equipe de WebE por fim precisa construir uma aplicação usando o projeto como guia e as tecnologias disponíveis.

Dessa forma entra em prática a construção, que consiste na seleção de ferramentas e tecnologia de WebE usadas para construir a *WebApp* que foi modelada. Sendo composta das seguintes ações: (1) *seleção* – envolve a identificação de componentes relevantes preexistentes, que podem ser reutilizados dentro do projeto proposto; (2) *codificação* – cobre a adaptação de componentes existentes ou criação de novos componentes; (3) *gerenciamento de conteúdo* – envolve a criação, migração e estruturação do conteúdo; (4) *autoria de páginas* –

envolve a integração do conteúdo bruto com o projeto gráfico e o mapeamento do conteúdo nas telas e páginas; (5) *integração* – compreende o vínculo do código, conteúdo e apresentação nos componentes finais a serem liberados; (6) *refatoração* – consiste em “polir” os componentes implantados a fim de melhorar sua estrutura, esclarecer e remover código redundante; e por fim, (7) envolve a verificação de que os diversos componentes e objetos estão corretos, denominada de *teste*.

#### 2.3.4.1 Teste

Uma *WebApp* é desenvolvida como uma série de incrementos. Quando o incremento da *WebApp* tiver sido construído, uma série de testes rápidos é realizada para garantir que os erros no projeto sejam descobertos. Dessa forma, entra em prática o teste, que consiste em exercitar cada uma das várias dimensões de qualidade da *WebApp* com intenção de encontrar erros ou descobrir tópicos que podem levar a falhas de qualidade antes que a *WebApp* se torne disponível aos seus usuários finais.

Dessa forma, o processo de teste para engenharia Web começa com testes que experimentam a funcionalidade do conteúdo e da interface que é imediatamente visível aos usuários finais. Sendo composto pelos seguintes tipos de teste:

**Teste de conteúdo:** tenta descobrir erros no conteúdo que estão presentes em uma parte da *WebApp*. E também, combina revisões e a criação de testes executáveis visando descobrir erros semânticos no conteúdo.

**Teste da interface:** consiste em experimentar os mecanismos de interação e validar os aspectos estéticos da interface com o usuário.

**Teste de navegação:** consiste em testar contra os casos de uso e caminhos de navegação, para garantir que quaisquer erros que impedem o término de um caso de uso sejam identificados e corrigidos.

**Teste de componentes:** experimenta o conteúdo e unidades funcionais dentro da *WebApp*. Onde uma *unidade* dentro da arquitetura funcional da *WebApp* é visto por oferecer serviço diretamente para um usuário final.

**Teste de configuração:** tenta testar um conjunto de configurações prováveis no cliente e no servidor para garantir que a experiência do usuário será a mesma em todas elas e isolar erros que podem ser específicos a determinada configuração.

**Teste de segurança:** incorpora uma série de testes elaborados para explorar

a vulnerabilidade na *WebApp* e seu ambiente

**Teste de desempenho:** compreende uma série de testes que são elaborados para avaliar: (1) como o tempo de resposta e a confiabilidade da *WebApp* são afetados por um maior tráfego de usuários e/ou complexidade funcional; (2) quais componentes da *WebApp* são responsáveis pela degradação do desempenho e quais características de uso causam a ocorrência de degradação; e (3) como a degradação do desempenho afeta os objetivos e requisitos gerais da *WebApp*.

Portanto, a transição da construção para implantação ocorre quando todos os testes foram aplicados, com intuito de detectar e corrigir os erros de um incremento. E então, é preciso implantá-lo imediatamente.

### 2.3.5 Implantação

Ao se terminar de construir um incremento da *WebApp* e este por sua vez, testado e corrigido os possíveis erros. A equipe de WebE precisará então implantar este incremento da *WebApp* em tantos ambientes quantos sejam necessários. Neste caso, entra em prática a implantação, que consiste em configurar a *WebApp* para o seu ambiente operacional, e depois entregá-lo aos usuários finais.

A atividade de implantação ocorre depois que um incremento de *WebApp* tiver sido verificado pela equipe de WebE. Como a engenharia Web é incremental por natureza, a implantação ocorre não uma vez, mas diversas vezes enquanto a *WebApp* se move em direção ao término de seu desenvolvimento. Sendo composto pelas seguintes ações:

**Empacotamento:** consiste em identificar o conjunto de componentes aptos a implantação.

**Liberação:** consiste em liberar os componentes do servidor de homologação para o servidor de produção, após os componentes individuais terem sido testados. Oferecendo assim aos usuários finais, um incremento operacional da *WebApp* que oferece funções e recursos úteis.

**Avaliação:** consiste em oferecer à equipe de *WebApp* por meio de feedback dos usuários, orientações importantes que resultam em modificações no conteúdo, funções, características e enfoque para o próximo incremento.

Portanto, a implantação de um incremento da *WebApp* representa um marco importante para qualquer projeto de engenharia Web, sendo que esta precisa

apresentar qualidade para que consiga alcançar seu propósito.

### 3 QUALIDADE NA WEBAPP

As *WebApps* estão cada vez mais complexas e têm se expandido rapidamente entre as organizações. No entanto, modelos de processo de produto bem definidos que promovam um ciclo de desenvolvimento repetível e eficiente, e modelos de processo e metodologias de avaliação que contribuem com a compreensão e melhora da qualidade das *WebApps*, não estão acompanhando este rápido crescimento observado. Torna-se necessário, ou até mesmo obrigatório, uma utilização sistemática e disciplinada de métodos, modelos e técnicas de Engenharia de Software para o desenvolvimento, manutenção e avaliação da qualidade das *WebApps*, principalmente em projetos médio ou grande porte.

A qualidade de software, segundo a norma NBR ISO 8402, é vista como a totalidade das características de uma entidade que lhe confere a capacidade de satisfazer às necessidades explícitas e implícitas. Entidade é o produto propriamente dito, as necessidades explícitas são as próprias condições e objetivos propostos pelo produtor e as necessidades implícitas são condições, mais subjetivas, como as diferenças entre as necessidades dos usuários, a evolução no tempo, as questões de segurança e outras.

Para que uma *WebApp* tenha qualidade, deve-se levar inúmeros requisitos em consideração. No entanto, para que esses requisitos sejam atendidos, há a necessidade de processos bem definidos para atender as expectativas do cliente, incluindo prazo e custo, sempre atentando tanto para a qualidade do processo, quanto para a qualidade do produto. Logo, a qualidade de uma *WebApp* é um fator muito importante que deve ser pensado durante seu processo de desenvolvimento.

Você tem menos de dois minutos para se comunicar na primeira vez em que um potencial cliente visita seu Website. Este é um fato básico sobre a experiência Web: no que diz respeito aos usuários, cada página deve justificar sua importância quando chamada. Se uma página não fizer isso imediatamente e de maneira clara, eles vão para outros websites. A maioria deles nem mesmo se dá ao trabalho de usar o *scroll* para ver o que ela contém. (NIELSEN, 2007, p.21)

Dessa forma, existem inúmeros fatores a se pensar quando se desenvolve uma *WebApp*, o leiaute, o conteúdo a ser exibido, o tamanho dos arquivos que serão exibidos nas páginas e a maneira como estes itens serão exibidos aos visitantes, que devem ser desenvolvidos visando sempre à facilidade de navegação para seus usuários, fazendo com que consigam entender a organização e localizar o conteúdo



que desejam na *WebApp*.

Na realidade, à medida que cada incremento da *WebApp* é introduzido, uma série de qualidade de produtos de software se aplicam às *WebApps* conforme a norma NBR ISO/IEC 9126.

### 3.1 Características de qualidade segundo a norma NBR ISO/IEC 9126

De acordo com a ISO 9126, a qualidade de produtos de software é “Um conjunto de atributos que têm impacto na capacidade do software de manter o seu nível de desempenho dentro de condições estabelecidas por um dado período de tempo (...)”. Ou seja, a especificação e avaliação da qualidade do produto de software são fatores-chave para garantir qualidade adequada. Isto pode ser alcançado pela definição apropriada das características de qualidade, levando em consideração o uso pretendido do produto de software.

Dessa forma, as mais relevantes dessas características – funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade – oferecem uma base útil para se avaliar a qualidade das *WebApps* as quais são, por sua vez, subdivididas em subcaracterísticas, conforme a Figura 3.1.

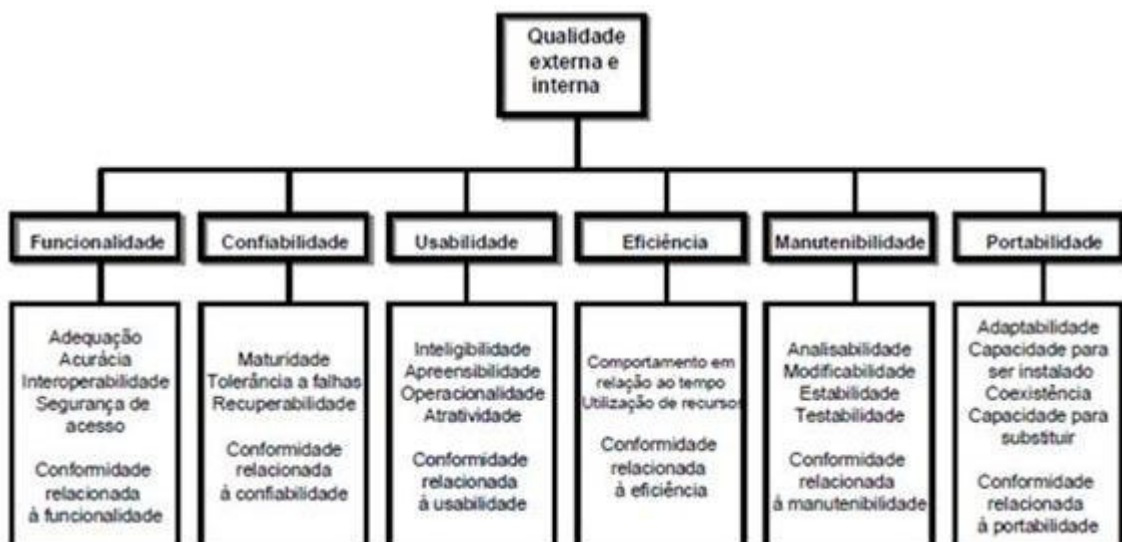


Figura 3.1: Árvore de requisitos de qualidade. Fonte: Engenharia de Software<sup>8</sup>.

Portanto, pode-se dizer que a norma ISO 9126 é útil para a definição,

<sup>8</sup>Disponível em: <<http://es-it.blogspot.com.br/p/engenharia-de-software.html>>. Acesso em 19 mar. 2015

avaliação e descrição dos requisitos, especificações, características, atributos e a avaliação antes e depois da entrega da *WebApp*.

### 3.1.1 Funcionalidade

Sabe-se que a demanda por qualidade de software tem motivado o desenvolvimento de software mais fácil de usar. Logo os desenvolvedores não podem menosprezar o papel do usuário/cliente, não podendo se esquecer das necessidades implícitas de uso. Uma vez que, cada cliente pode ter desejos e necessidades diferentes em relação a *WebApp* a ser desenvolvida e, por sua vez, são os mais interessados no uso da *WebApp*, na sua funcionalidade.

Dessa forma, segundo a norma NBR ISO/IEC 9126, a funcionalidade é vista como a capacidade do produto de software de prover funções que atendam as necessidades explícitas e implícitas, quando o software estiver sendo utilizado sob condições específicas.

Cabe ressaltar, que este atributo de qualidade apresenta as seguintes subcaracterísticas:

- **Adequação:** corresponde na capacidade do software de provar um conjunto apropriado de funções para tarefas e objetivos do usuário especificados.

- **Acurácia:** capacidade do produto de software de prover, com o grau de precisão necessário, resultados ou efeitos corretos ou conforme acordados. Por exemplo, sistemas que precisam de uma precisão de 3 casas decimais.

- **Interoperabilidade:** capacidade do produto de software de interagir com um ou mais sistemas especificados.

- **Segurança de acesso:** capacidade do produto de software de proteger informações e dados, de forma que, pessoas ou sistemas não autorizados não possam lê-los nem modificá-los e que não seja negado o acesso às pessoas ou sistemas autorizados.

- **Conformidade relacionada à funcionalidade:** capacidade do produto de software de estar de acordo com normas, convenções ou regulamentações previstas em leis relacionadas à funcionalidade.

Portanto, a funcionalidade está relacionada com o que software faz para atender às necessidades, enquanto que outras características estão principalmente relacionadas à quando e como ele atende às necessidades.

### 3.1.2 Confiabilidade

Está claro que produzir software de qualidade é uma meta essencial e básica, que oferece métodos, técnicas e ferramentas. O necessário mesmo é que o software seja confiável, eficaz e siga os padrões exigidos pelo contexto. Logo, é de suma importância considerar a confiabilidade de software no desenvolvimento de sistemas.

Dessa forma, segundo a norma NBR ISO/IEC 9126, a confiabilidade é vista como capacidade do produto de software de manter um nível de desempenho especificado, quando usado em condições especificadas.

Cabe ressaltar, que este atributo de qualidade apresenta as seguintes subcaracterísticas:

- **Maturidade:** capacidade do produto de software de evitar falhas decorrentes de defeitos no software.

- **Tolerância a falhas:** capacidade do produto de software de manter um nível de desempenho especificado em casos de defeitos no software ou de violação de sua interface especificada.

- **Recuperabilidade:** capacidade do produto de software de restabelecer seu nível de desempenho especificado e recuperar os dados diretamente afetados no caso de uma falha.

- **Conformidade relacionada à confiabilidade:** capacidade do produto de software de estar de acordo com normas, convenções ou regulamentações relacionadas à confiabilidade.

Portanto, a confiabilidade é vista como a probabilidade do sistema funcionar sem ocorrência de falhas num período e ambiente especificados.

### 3.1.3 Usabilidade

Desde a invenção da Web, a tecnologia para construção de *WebApps* vem sendo progressivamente incrementada permitindo o desenvolvimento de aplicações cada vez mais complexas que utilizam a Web não apenas para troca de informações, mas como plataforma para aplicações distribuídas tais como comércio eletrônico e Intranets. Durante este processo evolutivo, o número de usuários e de *WebApps* cresceram exponencialmente. E a Web tornou-se acessível a todas as

peças, passando a contar com uma grande variedade de aplicações. Contudo, observa-se que tal popularidade não implica necessariamente em satisfação dos usuários, que se confrontam muito frequentemente com problemas de usabilidade.

Sabe-se que a usabilidade não é derivada da estética, dos mecanismos de interação de última geração ou da inteligência artificial. Em vez disso, ela ocorre quando a arquitetura da interface se encaixa às necessidades das pessoas que vão utilizá-la.

Dessa forma, segundo a norma NBR ISO/IEC 9126, a usabilidade é vista como a capacidade do produto de software de ser compreendido, aprendido, operado e atraente ao usuário, quando usado sob condições especificadas.

Cabe ressaltar, que este atributo de qualidade apresenta as seguintes subcaracterísticas:

- **Inteligibilidade:** capacidade do produto de software de possibilitar ao usuário compreender se o software é apropriado e como ele pode ser usado para tarefas e condições de uso específicas. Por exemplo, facilidade de compreensão geral do site.

- **Apreensibilidade:** capacidade do produto de software de possibilitar ao usuário aprender sua aplicação. Por exemplo, uso de um modelo único dentro do sistema quanto às telas, relatórios e procedimento.

- **Operacionalidade:** capacidade do produto de software de possibilitar ao usuário operá-lo e controlá-lo. Por exemplo, facilidade de caminhar entre funções e/ou sistemas, respeitando a segurança de acesso.

- **Atratividade:** capacidade do produto de software de ser atraente ao usuário. Por exemplo, uso de cores e da natureza do projeto gráfico.

- **Conformidade relacionada à usabilidade:** capacidade do produto de software de estarem de acordo com normas, convenções, guia de estilo ou regulamentações relacionadas à usabilidade.

Portanto, a usabilidade visa a busca da eficiência ao realizar as tarefas presentes na *WebApp*.

### 3.1.4 Eficiência

Sabe-se que um software bem concebido, bem arquitetado e bem executado possui alta qualidade. É fácil trabalhar com ele, mantê-lo e melhorá-lo para suprir as

demandas.

Dessa forma, segundo a norma NBR ISO/IEC 9126, a eficiência é vista como a capacidade do produto de software de apresentar desempenho apropriado, relativo à quantidade de recursos usados, sob condições especificadas.

Cabe ressaltar, que este atributo de qualidade apresenta as seguintes subcaracterísticas:

- **Comportamento em relação ao tempo:** capacidade do produto de software de fornecer tempos de resposta e processamento.

- **Utilização de recursos:** capacidade do produto de software de usar tipos e quantidades apropriados de recursos, quando o software executa suas funções sob condições estabelecidas.

- **Conformidade relacionada à eficiência:** capacidade do produto de software de estar de acordo com normas e convenções relacionadas à eficiência.

Portanto, a eficiência está relacionada com os recursos necessários para os usuários interagirem com o sistema e alcançarem seus objetivos.

### 3.1.5 Manutenibilidade

Sabe-se que um gerente de projeto deve também avaliar a qualidade enquanto o projeto avança. Ou seja, deve medir erros e defeitos obrigando que a modificação seja analisada, implementada, testada, documentada e integrada.

Dessa forma, segundo a norma NBR ISO/IEC 9126, a manutenibilidade é vista como capacidade do produto de software ser modificado. As modificações podem incluir correções, melhorias ou adaptações do software devido as mudanças no ambiente e nos seus requisitos funcionais.

Cabe ressaltar, que este atributo de qualidade apresenta as seguintes subcaracterísticas:

- **Analisabilidade:** capacidade do produto de software de permitir o diagnóstico de deficiências ou causas de falhas no software, ou a identificação de partes a serem modificadas.

- **Modificabilidade:** capacidade do produto de software de permitir que uma modificação especificada seja implementada.

- **Estabilidade:** capacidade do produto de software de evitar efeitos inesperados decorrentes de modificações no software.

- **Testabilidade:** capacidade do produto de software de permitir que o software, quando modificado, seja validado.

- **Conformidade relacionada à manutenibilidade:** capacidade do produto de software de estar de acordo com normas relacionadas à manutenibilidade.

Portanto, a manutenibilidade diz respeito à facilidade com que o mesmo pode ser modificado para satisfazer requisitos do usuário ou ser corrigido quando deficiências são detectadas.

### 3.1.6 Portabilidade

Sabe-se, que os softwares são desenvolvidos visando as arquiteturas de hardware e software dominantes no mercado, no entanto, manter uma equipe de desenvolvimento e testes para arquiteturas menos utilizadas é inviável para a maioria das empresas.

Dessa forma, segundo a norma NBR ISO/IEC 9126, a portabilidade é vista como a capacidade do produto de software de ser transferido de um ambiente para outro.

Cabe ressaltar, que este atributo de qualidade apresenta as seguintes subcaracterísticas:

- **Adaptabilidade:** capacidade do produto de software de ser adaptado para diferentes ambientes especificados, sem necessidade de aplicação de outras ações ou meios além daqueles fornecidos para essa finalidade pelo software considerado.

- **Capacidade para ser instalado:** capacidade do produto de software para ser instalado em um ambiente especificado.

- **Coexistência:** capacidade do produto de software de coexistir com outros produtos de software independentes, em um ambiente comum, compartilhando recursos comuns.

- **Capacidade para substituir:** capacidade do produto de software de ser usado em substituição a outro produto de software especificado, com o mesmo propósito e no mesmo ambiente. Este atributo interage tanto com adaptabilidade quanto com a capacidade para ser instalado.

- **Conformidade relacionada à portabilidade:** capacidade do produto de software de estar de acordo com normas ou convenções relacionadas à portabilidade.

Portanto, estes atributos de qualidade e suas subcaracterísticas são fundamentais para verificar a eficácia<sup>9</sup> da *WebApp* que foi criada. Sendo por sua vez, preciso realizar testes visando medir a qualidade de interação entre o usuário e a *WebApp*, denominada de *avaliação de usabilidade*.

### 3.2 Avaliação de Usabilidade

Sabe-se que ao desenvolver uma *WebApp*, é necessário pensar na usabilidade e, sendo assim, quanto mais cedo se puder detectar os problemas, melhor será. Com isso, entra em prática a avaliação de usabilidade, que consiste em utilizar métodos de inspeção visando identificar problemas num website quando o projeto da interface tiver sido estabelecido. Um dos métodos de avaliação de usabilidade mais utilizado é a *avaliação heurística*.

#### 3.2.1 Avaliação Heurística

A avaliação heurística é um método de inspeção de usabilidade que foi introduzido no início da década de 90 por Jakob Nielsen e Rolf, tendo como característica ser um método relativamente barato e rápido para encontrar problemas em interfaces de usuários.

Visto que, todas as *WebApps* servem a um propósito, mas como saber se ela funciona? Se o desenvolvedor programou a *WebApp*, logo de cara vai saber como fazer uma busca ou comprar um produto nela. Mas e um usuário comum? Neste caso, entra em prática a avaliação heurísticas, que consiste em um método de avaliação de usabilidade no qual especialistas (geralmente, entre 3 e 5) procuram identificar problemas numa interface com o usuário, através da análise e interpretação de um conjunto de princípios ou heurísticas. Ou seja, é nada mais do que a análise da interação homem computador (IHC).

Nestas avaliações, um especialista age como o usuário-alvo e tenta completar uma série de tarefas pré-determinadas na *WebApp* desenvolvida. Por exemplo, tentar comprar um produto, tentar fazer um pagamento de um produto ou até mesmo encontrar informações importantes (como telefone, endereço ou e-mail para

---

<sup>9</sup> É a capacidade de resolução de problemas associado ao grau de realização dos objetivos. O sistema deve fazer o que eu espero que faça.

contato).

Dessa forma, um conjunto de heurísticas<sup>10</sup> para análise é utilizada, neste caso, proposto por J. Nielsen & R. Mack [NIELSEN, 1994] que inclui:

- **Visibilidade do estado do sistema:** o sistema deve sempre manter os usuários informados sobre o que está acontecendo, através de retorno apropriado dentro de tempo razoável. Por exemplo, tempo de resposta do sistema, visto que, o utilizador tende 1,0 segundo a distrai-se; e o contexto espacial, isto é, as páginas devem estar claramente identificadas (títulos).

- **Correspondência entre o sistema e o mundo real:** consiste em falar a linguagem do utilizador, ou seja, usar palavras e conceitos do mundo do utilizador e não usar termos técnicos específicos.

- **Controle e liberdade por parte do usuário:** usuários frequentemente escolhem funções do sistema por engano e irão precisar de uma “saída de emergência” claramente marcada para sair do estado indesejado sem ter que passar por um diálogo extenso. Suporte as funções “desfazer” e “refazer”.

- **Consistência e padrões:** os usuários não devem ter que imaginar quando diferentes palavras, situações ou ações significam a mesma coisa. Siga convenções da plataforma.

- **Prevenção de erros:** melhor ainda que boas mensagens de erro sejam um projeto cuidadoso que evita um problema antes que ele aconteça. Ou elimine condições com tendência a erros ou verifique-as e apresente ao usuário uma opção de confirmação antes que eles se comprometam com a ação.

- **Reconhecer em lugar de memorizar:** minimize a carga da memória do usuário fazendo objetos, ações e opções visíveis. O usuário não deve ter que lembrar informação de uma parte do diálogo para outra. Com isso, instruções para uso do sistema devem ser visíveis.

- **Flexibilidade e eficiência no uso:** consiste em fornecer “aceleradores” permitindo acesso rápido a várias funções da *WebApp*. Por exemplo: os formulários devem possuir um botão “Apagar” que permite limpar o conteúdo de todos os seus campos.

- **Estética e design minimalista:** neste caso, os diálogos não devem conter informação irrelevante. Uma vez que, demasiado de texto e imagens numa páginas,

---

<sup>10</sup> Conhecidas como heurísticas de Nielsen, que visam cercar todos os problemas de uma interface.



distraem os utilizadores. Com isso, os textos devem ser concisos.

- **Ajudar os usuários a reconhecer, diagnosticar e recuperar os erros:** mensagens de erro devem ser expressas em linguagem comum (sem códigos), indicar precisamente o problema, e sugerir construtivamente uma solução.

- **Dar ajuda e documentação:** ainda que seja melhor se um sistema possa ser usado sem documentação, pode ser necessário prover ajuda e documentação. Qualquer informação desse tipo deve ser fácil de procurar, focada na tarefa do usuário.

Dessa forma, os problemas de usabilidade encontrados e apontados consistem de violações ou não conformidades com essa série de regras. Sendo por sua vez, preciso definir e categorizar os pontos da usabilidade a serem analisados em um website.

### 3.3 Pontos da Usabilidade

Sabe-se que a chave para a satisfação de um usuário de um website é a simplicidade: deve-se chegar a um website e conseguir utilizá-lo com simplicidade e eficiência. No entanto, não é isso que sempre ocorre, pois muitas vezes se perde tempo em páginas confusas e que às vezes não nos levam a lugar nenhum. Neste caso, entra em prática os pontos de usabilidade nas *WebApps*, que visam listar e categorizar os problemas que podem ser encontrados em um website. Essa análise é feita dividindo-se os pontos de usabilidade em: acessibilidade, identidade, navegabilidade e conteúdo.

#### 3.3.1 Acessibilidade

Ao se desenvolver um website, deve-se levar em conta a acessibilidade. Ou seja, deve ser construído com a semântica correta de tags HTML para que as pessoas consigam navegar sem problemas podendo aprender, compreender, navegar e interagir. Além disto, envolve o fato de mesmo estando acessível, ele precisa ser muito rápido. Afinal, os usuários na internet estão cada vez menos pacientes.

Dessa forma, os seguintes problemas de acessibilidade podem ser visto em uma *WebApp*:

- **O tempo de carregamento do site é razoável?** Sabe-se que um conteúdo de qualidade e um design atraente são importantes, mas se a *WebApp* demora muito a carregar, poderá se perder visitas antes mesmo que elas vejam o que se têm a oferecer. Com isso, a velocidade com que as páginas de um website carregam são um dos pontos mais críticos de uma boa experiência de utilização.

- **Contraste entre o texto e o fundo é adequado?** Como Steve Jobs disse, design vai além de como as pessoas parecem. Portanto, de nada vale investir em um design que seja “bonito” se você dificultar a experiência do usuário. Uma vez que, é importante usar cor de fundo que não atrapalhe a leitura de seu conteúdo.

- **Tamanho da fonte/espaçamento é de fácil leitura?** Antigamente, o padrão era utilizar a fonte Arial tamanho 12. No entanto, os monitores ganharam mais pixels, se alargaram e manter uma fonte de tamanho 12 é ruim para a leitura de artigos. Com isso, pobre legibilidade aumenta a frustração, que conduz ao abandono da *WebApp*. Além disso, o espaçamento entre os elementos da página (imagens, ícones, textos, etc.) é fundamental, visto que, se tudo estiver amontoado sem uma separação clara entre os elementos, ler e navegar na *WebApp* se tornará uma tarefa bem difícil.

- **Flashes são utilizados com moderação?** Graças ao *Flash*, hoje é possível criar *WebApps* totalmente animadas e com uma rotatividade de conteúdo muito maior do que antigamente. No entanto, o problema no uso do *Flash* é o abuso de seus recursos, por oferecer inúmeros recursos de animações o *Flash* acaba encorajando o uso exagerado dos mesmos.

- **As imagens têm tags ALT apropriadas?** Poucos sabem o que é uma ALT tag. Ela nada mais é do que um texto que diz do que se trata uma imagem.

- **O local tem página personalizada not-found/404?** Sabe-se que a maioria dos websites para por aí e não mostram ao leitor possibilidades para ele voltar para a homepage ou para a área de artigos. Sendo por sua vez, uma página muitas vezes esquecida pelos desenvolvedores web.

### 3.3.2 Identidade

Ter uma identidade visual, que passe uma mensagem de qualidade e segurança aos seus usuários é fundamental para o sucesso de uma *WebApp*, além de transmitir sua essência e filosofia. Ou seja, o visual do website precisa agradar

seus visitantes, assim que entram em sua página inicial, pois o visual é a primeira coisa que os usuários reparam em um website, uma página que tenha um leiaute pobre certamente passará uma imagem ruim para seus visitantes, mesmo tendo um ótimo conteúdo o website poderá perder muitos visitantes por causa de um visual que não o agrada.

Neste caso, entra em prática a identidade, que consiste na imagem que a *WebApp* passa ao usuário no primeiro contato e tendo que representá-la como um serviço de qualidade.

Dessa forma, os seguintes problemas de identidade podem ser visto em uma *WebApp*:

- **Logotipo da empresa é de forma destacada?** Sabe-se, que a maior parte das *WebApps* tem o seu logotipo no canto superior esquerdo das suas páginas. Se o colocar em outros locais, poderá perder identidade porque os seus leitores poderão não reconhecer que aquele é o seu website.

- **Tagline faz propósito de a empresa ficar claro?** Evite jargão de marketing ao definir a tagline da *WebApp*. Uma vez que, ela representa uma pequena frase que faz lembrar-se de uma marca e que fica perto do logotipo.

- **A página inicial é digestível em 5 segundos?** Ao se acessar uma página inicial de uma *WebApp*, tem-se em torno de 5 segundos antes que um usuário deixe seu website. Visto que, se ele não descobrir rapidamente do que se trata o site, ele irá deixá-lo.

- **Há um caminho livre para informações “sobre” a empresa?** Hoje em dia, os usuários fazem questão de saber quem está por trás do site. Uma vez que, o anonimato na web está com os dias contados, sendo por sua vez, preciso deixar claro quem produz o site e como ele pode ajudar o usuário.

- **Há um caminho livre para informações de contato?** Sabe-se, que a internet nos permite entrar em contato com pessoas do mundo inteiro, interessadas no que se tem para compartilhar. Entretanto, se não colocar uma área de contato no seu website, ninguém entrará em contato com você e muitas oportunidades de negócios, parceiras, ideias serão perdidas.

### 3.3.3 Navegabilidade

A navegabilidade traduz-se na facilidade proporcionada ao utilizador na sua

visita ao longo da informação e das páginas do site. Ou seja, para qualquer usuário, qualquer que seja o objetivo e o contexto, é fundamental que consiga navegar conscientemente (sabendo onde está e para onde vai). Neste caso, entra em prática a navegabilidade, que é vista como a propriedade, ou capacidade, que a interface do website possui de facilitar ao usuário chegar ao seu destino da maneira o mais eficiente possível. O lema aqui é: menos clique e informações necessárias sempre à mão e numa sequência lógica.

Dessa forma, os seguintes problemas de navegabilidade podem ser visto em uma *WebApp*:

- **A navegação principal é facilmente identificável?** Quase todas as *WebApps* teve um menu principal desde que os primeiros navegadores chegou ao mercado. Se você tiver duas ou mais áreas de navegação, que fique claro por que eles são diferentes.

- **Menus de navegação são claros e concisos?** Sabe-se, que o menu de navegação está presente em sites há um longo tempo. No entanto, o problema nesse item pode ocorrer de duas formas: (1) o site não possui um menu de navegação; e (2) o site possui vários menus de navegação (confundindo usuários).

- **Logotipo da empresa está ligada à homepage?** Isso pode parecer menor, mas as pessoas esperam logos para conectar-se a homepages, e quando isso não acontece, a confusão se segue. Há muitas *WebApps* que têm o seu logotipo no local correto, mas depois este não serve para nada a não ser para identificar a página em que estar.

- **Os links são consistentes e fáceis de identificar?** No ramo de internet marketing é comum ver sites usando vermelho para os cabeçalhos e vermelho para links. Ao colocar os dois itens com a mesma cor, o usuário fica confuso, não sabendo exatamente onde ele pode e não pode clicar. Além disso, outro erro comum é colocar links com cores diferentes. Se um link é azul e outro link é verde você também está criando confusão para o usuário.

- **Pesquisa no site é de fácil acesso?** Por vezes, os seus visitantes não conseguem encontrar o que querem e precisam fazer uma pesquisa para encontrar aquele artigo que procuram. Porém, alguns sites não possuem campo de pesquisa ou os colocam em um lugar de difícil acesso.

### 3.3.4 Conteúdo

Após garantir a acessibilidade de uma *WebApp*, reforçar sua identidade e mostrar como funciona a navegabilidade, o conteúdo do website precisa estar de acordo com o seu tema, algo que responda às expectativas de seus visitantes.

Encher o website com informações sem valor na tentativa de mostrar para seus visitantes que o website tem conteúdo não vai fazer com que o website agrade seus visitantes, eles simplesmente ignorarão o texto e caso o usuário leia o texto da página, provavelmente perceberá que seu conteúdo é vago e assim será criada uma imagem negativa do website, fazendo com que saiam e não voltem a visitar a página.

“Papo bobo” é como papo furado – sem conteúdo, basicamente apenas uma forma de ser sociável. Porém, a maioria dos usuários Web não tem tempo para papo furado; eles querem ir direto ao que interessa. Você pode – e deve – eliminar tanto “papo bobo” quanto possível. (KRUG, 2008, p.46).

Neste caso, entra em prática o conteúdo, visto que, o conteúdo é o rei. Se não quer que o reino a desmorone, o conteúdo precisa ser consistente, organizado e fácil percorrer.

Dessa forma, os seguintes problemas de conteúdo podem ser visto em uma *WebApp*:

- **Estilos e cores são consistentes?** Não apenas links, mas todas as cores do seu site precisam ser pensadas como um todo e não em particular. Criar uma “salada mista” de cores é ter a certeza de confundir seus usuários. Isso ocorre porque tornando todas as cores do seu site predominantes, você está fazendo com que nenhuma delas se destaque.

- **Ênfase (negrito, etc.) é usado com moderação?** Sabe-se, que a cognição humana tenta chamar a atenção para tudo e acaba, por consequência, não chamando atenção para nada.

- **Anúncios e pop-ups são discretos?** A maioria dos sites necessita de anúncios para pagar suas contas ou para rentabilizar o tempo que o editor leva para entregar um conteúdo de graça. Entretanto, assim como cores e ênfases nos textos, os anúncios também devem ser utilizados com cautela, já que o excesso de anúncios poderá fazer o usuário sair correndo para outro site. Além disso, o uso de pop-ups é um dos erros mais detestados pelos usuários da web, que muitas vezes

apenas pelo fato de saber que a página abre janelas pop-ups quando acessada, evitam visitá-la.

- **Títulos de páginas HTML são explicativos?** Sabe-se, que os títulos devem ser descritivo, único, e não encravado cheio de palavras-chave. Uma vez que, os títulos das páginas são a primeira coisa que os usuários veem, e se esses títulos não fazem sentido, eles não continuam na *WebApp*.

Portanto, os pontos da usabilidade visam categorizar os principais problemas que uma *WebApp* possa apresentar durante a manipulação de sua interface.

## 4 ESTUDO DE CASO – USABILIDADE DE WEBAPPS

Este estudo de caso visa apontar recomendações de melhorias de usabilidade em diferentes *WebApps*, visto que, cada *WebApp* tem seu foco. Sendo assim, essa análise será feita com somente *WebApps* ocidentais utilizando o *Google Chrome*. E, por conseguinte, essa análise será feita dividindo-se os pontos de usabilidade em: acessibilidade, identidade, navegabilidade e conteúdo. Onde serão agrupadas por meio de pesquisas as *WebApps* que se adequem mais naquele ponto analisado. Dessa forma, o objetivo é verificar a qualidade das *WebApps*.

Cabe ressaltar, que não foi utilizada nenhuma métrica de usabilidade devido ao seu aspecto subjetivo.

### 4.1.1 Acessibilidade

Dentre os problemas da acessibilidade, foram analisados os seguintes: tempo de carregamento do site; tamanho da fonte/espacamento dos elementos e a utilização da ferramenta *Flash*. Cabe ressaltar, que foi escolhido apenas 3 (três) problemas essenciais que podem comprometer a acessibilidade da *WebApp*.

• **O tempo de carregamento do site é razoável?** Para ilustrar este problema, foi analisado 16 (dezesesseis) websites de notícias, visto que, são websites que fazem uso de diferentes conteúdos e informações. Cujo objetivo, é verificar o tempo de carregamento do website, conforme a Figura 4.1.

Para realização da análise deste problema foram analisados os seguintes critérios e definições:

- homepage extensa: consiste em conter muitas informações, sendo por sua vez, desnecessárias e desorganizadas.
- imagens pesadas: consiste em ter imagens que demoram muito para carregar, comprometendo o desempenho do site. Que por sua vez, pode apresentar às vezes uma página de erro indicando que houve um erro durante a exibição da página.
- publicidade em excesso na homepage: prejudica o carregamento, uma vez que muitos banners são em *Flash* ou apesar de serem imagens, são bastante pesados.
- vídeos na homepage: prejudica o carregamento, tornando-o extremamente

pesado. Além disso, configurar os vídeos para a opção *Auto play* também poderá ter efeitos negativos.



Figura 4.1: Websites de notícias<sup>11</sup>.

Tendo em vista estes critérios, para análise foi atribuído a seguinte convenção: (1) ruim – a *WebApp* segue a definição do critério analisado; (2) regular – a *WebApp* apresenta algumas melhorias quanto ao critério analisado, isto é, o tempo de carregamento da página leva em média 10 segundos; (3) bom – a *WebApp* apresenta uma melhoria considerável quanto ao critério analisado, isto é, o tempo de carregamento da página leva em média 5 segundos e (4) muito bom – a *WebApp* não segue a definição do critério analisado.

<sup>11</sup> Websites de notícias analisados: Folha de S. Paulo, G1, R7, MSN Brasil (linha 1), 2015; Estadão, Terra, UOL, Yahoo! Notícias (linha 2), 2015; Jovem Pan Online, Carta Maior, Veja, Band.com.br (linha 3), 2015; e EBC, Abril.com, Pop, revista Época (linha 4), 2015.



Tabela 4.1 - Síntese de análise dos critérios que afetam o carregamento de um website.

Critérios analisados/ Websites	Folha de S. Paulo	G1	R7	MSN Brasil	Estadão	Terra Notícias	UOL	Yahoo! Notícias	Jovem Pan Online	Carta Maior	Veja	Band.co.br	EBC	Abril.com	Pop	Revista Época
Homepage Extensa	2	2	1	2	1	1	1	2	2	3	3	3	3	3	3	1
Imagens Pesadas	3	3	1	3	3	3	3	2	3	4	3	4	4	4	3	2
Publicidade em excesso na homepage	2	3	1	4	3	4	3	3	3	4	3	4	4	3	4	2
Vídeos na homepage	4	3	1	2	3	3	3	2	3	4	4	4	4	4	4	2
Percentual (%)	6,1	6,1	2,2	6,1	5,6	5,6	5,6	5,0	6,1	8,4	7,3	8,4	8,4	7,8	7,8	3,9

De acordo com os percentuais da tabela 4.1, pode-se notar que 15 (quinze) sites tiveram carregamento considerável não prejudicando sua usabilidade, mas, o website do R7 foi o que obteve o pior desempenho de carregamento. Portanto, para resolver este problema, é preciso seguir as seguintes recomendações (conforme a Tabela 4.2):

Tabela 4.2 - Síntese de melhorias para os critérios analisados quanto ao carregamento do website R7.

Critérios analisados	Recomendações
Homepage extensa	Organize a homepage contendo apenas informações necessárias de cada sessão, tirando a repetição exacerbada.
Imagens pesadas	Realize a otimização das imagens visando controlar a qualidade e o peso com que a imagem será salva. Geralmente, faça o upload de imagens no formato JPEG, que é um tipo de imagens que são comprimidas.
Publicidade em excesso na homepage	Diminua a quantidade de publicidade no site e distribua melhor os banners restantes.
Vídeos na homepage	Diminua a quantidade de vídeos na homepage, com isso, diminui-se a quantidade de <i>requests</i> <sup>12</sup> feito pelos vídeos.

• **Tamanho da fonte/espaçamento é de fácil leitura?** Para ilustrar este problema, foi analisado 16 (dezesseis) websites de notícias e/ou portais, visto que, são websites que fazem uso de texto para passar a informação desejada. Cujo objetivo, é verificar a legibilidade dos textos na *WebApp*, conforme a Figura 4.1.

<sup>12</sup> Envia uma solicitação de pedido para visualizar o vídeo ao servidor da internet.

Para realização da análise deste problema, foram analisados os seguintes critérios e definições:

- não diferenciação do tipo de notícia na homepage: as notícias principais e secundárias parecem ser apenas um tipo de notícia sendo difíceis de ser diferenciadas.

- não espaçamento entre os elementos das páginas: consiste em ter um aglomerado de elementos da página (imagens, ícones, textos, etc.) garantindo uma não “respiração” da página.

- sem destaque para a notícia principal: a notícia principal parece ser uma notícia secundária, tornando-a difícil de ser diferenciada das outras.

Tendo em vista estes critérios, para análise foi atribuído a seguinte convenção: (1) ruim – a *WebApp* segue a definição do critério analisado; (2) regular – a *WebApp* apresenta algumas melhorias quanto ao critério analisado; (3) bom – a *WebApp* apresenta uma melhoria considerável quanto ao critério analisado e (4) muito bom – a *WebApp* não segue a definição do critério analisado.

Tabela 4.3 - Síntese de análise dos critérios que afetam a legibilidade de um website.

Critérios Analisados/ Websites	Folha de S. Paulo	G1	R7	MSN Brasil	Estadão	Terra Notícias	UOL	Yahoo! Notícias	Jovem Pan Online	Carta Maior	Veja	Band.co.br	EBC	Abril.com	Pop	Revista Época
Não diferenciação do tipo de notícia na homepage	3	3	4	4	4	4	4	4	4	4	2	3	3	3	4	4
Não espaçamento entre os elementos das páginas	4	3	3	4	4	4	3	3	3	4	4	3	4	3	4	4
Sem destaque para a notícia principal	3	3	3	4	3	3	3	4	4	4	2	3	3	3	4	4
Percentual (%)	6,0	5,3	6,0	7,1	6,1	6,1	6,0	6,1	6,1	7,1	4,8	5,3	6,0	5,3	7,1	7,1

De acordo com os percentuais da tabela 4.3, pode-se notar que 15 (quinze) sites tiveram uma legibilidade considerável não prejudicando sua usabilidade, mas, o website da *Veja* foi o que obteve a pior legibilidade. Portanto, para resolver este problema, é preciso seguir as seguintes recomendações (conforme a Tabela 4.4):

Tabela 4.4 - Síntese de melhorias para os critérios analisados quanto à legibilidade do website da Veja.

Critérios Analisados	Recomendações
Não diferenciação do tipo de notícia na homepage	Definir um estilo para fonte (Arial ou Verdana) e um tamanho da fonte específico para as notícias principais e secundárias na homepage.
Não espaçamento entre os elementos das páginas	Faça uso de espaços em branco entre os elementos para melhorar a legibilidade dos textos, evitando um aspecto compacto. Estes espaços em branco dirigem a atenção do utilizador para informação realmente importante.
Sem destaque para a notícia principal	Faça uso de fonte superior a tamanho 12 e em negrito, associando-a uma imagem.

• **Flashes são utilizados com moderação?** Para ilustrar este problema, foi analisado 16 (dezesesseis) websites institucionais, visto que, são websites que fazem o uso de animações, efeitos especiais e *slide shows* para divulgação da empresa e de seus produtos, conforme a Figura 4.2.

Figura 4.2: Websites institucionais<sup>13</sup>.

<sup>13</sup> Websites analisados: Brahma, Devassa, Guaraná Antarctica, Bauducco (linha 1), 2015; Coca-Cola, Fanta, Adidas, Nestlé (linha 2), 2015; Itaipava, Nike, Pepsi, Bob's (linha 3), 2015; e Martini, MC Donalds, Heineken, Skol (linha 4), 2015.

Dentre os 16 (dezesesseis) websites analisados, conforme a Figura 4.2. Pode-se notar que a maioria utiliza a ferramenta *Flash* parcialmente, visando destacar apenas seus produtos, conforme a análise da Figura 4.3.

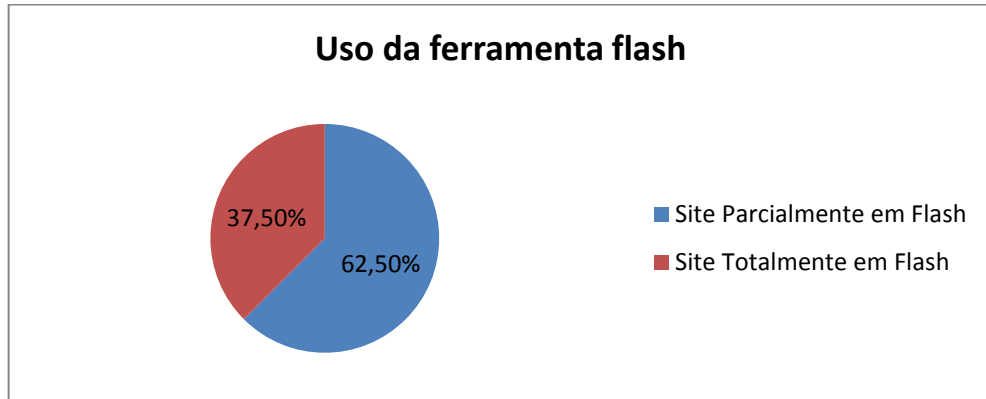


Figura 4.3: Gráfico que representa o percentual de uso da ferramenta flash nos websites analisados.

Para realização da análise deste problema (conforme a Figura 4.3), foram analisados os seguintes critérios e definições:

- peso do site: a relação peso e velocidade de carregamento do site são fundamentais para uma boa usabilidade. Sendo que, em sites em *Flash*, é normal ter que esperar o site inteiro carregar e ainda há a necessidade de plug-in para visualizar.
- uso de botões escondidos: devido ao layout colorido alguns botões passam despercebidos por parecerem apenas mais uma parte do layout, forçando seus visitantes a procurá-los para navegar em suas páginas.
- uso exagerado de animações e/ou efeitos: consiste em ter animações e efeitos exagerados sem fazer jus ao propósito da *WebApp*.

Tendo em vista estes critérios, para análise foi atribuído a seguinte convenção: (1) ruim – a *WebApp* segue a definição do critério analisado; (2) regular – a *WebApp* apresenta algumas melhorias quanto ao critério analisado, isto é, em média 75% do website têm conteúdos em flash; (3) bom – a *WebApp* apresenta uma melhoria considerável quanto ao critério analisado, isto é, em média 50% do website têm conteúdos em flash e (4) muito bom – a *WebApp* não segue a definição do critério analisado.

Tabela 4.5 - Síntese de análise dos critérios de uso de flashes em um website.

Critérios Analisados/ Websites	Brahma	Devassa	Guaraná Antarctica	Bauducco	Coca-Cola	Fanta	Adidas	Nestlé	Itaipava	Nike	Pepsi	Bob's	Martini	MC Donalds	Heineken	Skol
Peso do site	2	1	3	2	4	3	3	4	1	4	2	4	1	3	4	1
Uso de botões escondidos	3	2	3	4	4	4	3	3	2	3	4	2	2	2	3	2
Uso exagerado de animações e/ou efeitos	1	4	3	4	4	3	3	3	2	3	3	3	1	4	3	3
Percentual (%)	4,4	5,2	6,6	7,4	8,8	7,4	6,6	7,4	3,7	7,4	6,6	6,6	3,0	6,6	7,4	4,4

De acordo com os percentuais da tabela 4.5, pode-se notar que 15 (quinze) sites não tiveram um uso exagerado de *Flash* prejudicando sua usabilidade, mas, o website Martini foi o que obteve o pior uso desta ferramenta. Portanto, para resolver este problema, é preciso seguir as seguintes recomendações (conforme a Tabela 4.6):

Tabela 4.6 - Síntese de melhorias para os critérios analisados quanto uso da ferramenta flash no website Martini.

Critérios Analisados	Recomendações
Peso do site	Evitar fazer o site inteiro em <i>Flash</i> , isto é, use somente nas partes essenciais como divulgação de serviços. E nas outras partes, faça uso de HTML visando torná-lo mais fácil de atualizar, o mais acessível possível e otimizado.
Uso de botões escondidos	Dá um layout diferente aos botões em flash, com isso, o usuário poderá identificá-los.
Uso exagerado de animações e/ou efeitos	Reduza o uso exagerado de animações, ou seja, use animações apenas para valorizar a comunicação em situações específicas. E por sua vez, nas animações/efeitos, faça uso dos <i>pre-loaders</i> – são pequenas animações que são carregadas rapidamente no começo do arquivo <i>Flash</i> e indicam que o arquivo está sendo carregado.

#### 4.1.2 Identidade

Dentre os problemas de identidade, foram analisados os seguintes: logotipo de forma destacada, página inicial apresentável e local para informações de contato a empresa. Cabe ressaltar, que foi escolhido apenas 3 (três) problemas essenciais,

que podem comprometer a identidade da *WebApp*.

• **Logotipo da empresa é de forma destacada?** Para ilustrar este problema, foi analisado 16 (dezesseis) websites. Cujo objetivo, é verificar se o logotipo está de forma destacada nas páginas da *WebApp*, conforme a Figura 4.4.

Para realização da análise deste problema, foram analisados os seguintes critérios e definições:

- logotipo não destacável na homepage: consiste em ter um logotipo sem destaque na homepage, fazendo apenas parecer um elemento qualquer.
- logotipo sem ser no canto superior esquerdo: consiste em ter um logotipo em qualquer parte da *WebApp*, menos no canto superior esquerdo. Visto que, na maioria das vezes a observação começa da esquerda para direita em um site.
- logotipo não destacável nas páginas secundárias: consiste em ter um logotipo sem destaque nas páginas secundárias, fazendo apenas parecer um elemento qualquer.



Figura 4.4: Websites diversos<sup>14</sup>.

<sup>14</sup> Websites analisados: Editora FTD, Correios, G1, Coca-Cola (linha 1), 2015; IBGE, Deekdisc, iG, Jovem Pan FM (linha 2), 2015; Rio Anil Shopping, Casas Bahia, Rádio MIX FM, MTV (linha 3), 2015; e O Boticário, Esporte Interativo, Volkswagen, Nike (linha 4), 2015.

Tendo em vista estes critérios, para análise foi atribuído a seguinte convenção: (1) ruim – a *WebApp* segue a definição do critério analisado; (2) regular – a *WebApp* apresenta algumas melhorias quanto ao critério analisado; (3) bom – a *WebApp* apresenta uma melhoria considerável quanto ao critério analisado e (4) muito bom – a *WebApp* não segue a definição do critério analisado.

Tabela 4.7 - Síntese de análise dos critérios de uso do logotipo em um website.

Critérios Analisados/ Websites	Editora FTD	Correios	G1	Coca-Cola	IBGE	Deekdisc	iG	Jovem Pan FM	Rio Anil Shopping	Casas Bahia	Rádio MIX FM	MTV	O Boticário	Esporte Interativo	Volkswagen	Nike
Logotipo não destacável na homepage	4	4	3	4	4	4	3	4	4	3	3	4	3	4	3	2
Logotipo sem ser no canto superior esquerdo	4	4	1	4	4	4	4	4	4	4	4	4	4	4	4	3
Logotipo não destacável nas páginas secundárias	4	4	2	4	4	4	2	4	4	3	3	4	3	4	3	2
Percentual (%)	7,0	7,0	3,5	7,0	7,0	7,0	5,2	7,0	7,0	5,8	5,8	7,0	5,8	7,0	5,8	4,1

De acordo com os percentuais da tabela 4.7, pode-se notar que 15 (quinze) sites obedeceram ao uso do logotipo na *WebApp*, mas, o website do G1 foi o que obteve o pior uso, prejudicando sua identidade. Portanto, para resolver este problema, é preciso seguir as seguintes recomendações (conforme a Tabela 4.8):

Tabela 4.8 - Síntese de melhorias para os critérios analisados quanto uso do logotipo no website G1.

Critérios Analisados	Recomendações
Logotipo não destacável na homepage	Criar um logotipo que passe a imagem daquilo que é a <i>WebApp</i> para os usuários. E também, deve ser maior e mais destacada do que os itens a seu redor.
Logotipo sem ser no canto superior esquerdo	Colocar o logotipo da <i>WebApp</i> sempre no canto superior esquerdo, de modo a chamar de imediato a atenção quando os usuários entrarem na <i>WebApp</i> .
Logotipo não destacável nas páginas secundárias	Criar um logotipo que permaneça de forma destacada nas páginas secundárias.

- **A página inicial é digestível em 5 segundos?** Para ilustrar este problema,

foi analisado 16 (dezesesseis) websites. Cujo objetivo, é verificar se a homepage explica claramente o que é a *WebApp*, quais as informações e serviços estão disponíveis, conforme a Figura 4.4.

Para realização da análise deste problema, foram analisados os seguintes critérios e definições:

- a homepage não passa os valores da *WebApp*: consiste em ter um aglomerado de informações e serviços/produtos que não fazem jus a *WebApp* fazendo com que o usuário vá atrás da resposta.
- informações desorganizadas: consiste em ter um aglomerado de informações repetidas e desorganizadas relacionadas ao foco da *WebApp*.

Tendo em vista estes critérios, para análise foi atribuído a seguinte convenção: (1) ruim – a *WebApp* segue a definição do critério analisado; (2) regular – a *WebApp* apresenta algumas melhorias quanto ao critério analisado, isto é, a homepage tem em média 50% dos seus conteúdos organizados; (3) bom – a *WebApp* apresenta uma melhoria considerável quanto ao critério analisado, isto é, a homepage tem em média 75% dos seus conteúdos organizados e (4) muito bom – a *WebApp* não segue a definição do critério analisado.

Tabela 4.9 - Síntese de análise dos critérios de uma homepage apresentável em um website.

Critérios Analisados /Websites	Editora FTD	Correios	G1	Coca-Cola	IBGE	Deedisc	iG	Jovem Pan FM	Rio Anil Shopping	Casas Bahia	Rádio MIX FM	MTV	O Boticário	Esporte Interativo	Volkswagen	Nike
Homepage não passa os valores da <i>WebApp</i>	4	4	4	4	2	4	3	4	4	3	4	4	4	4	3	4
Informações desorganizadas	4	4	3	4	1	4	3	4	4	3	4	4	4	4	3	3
Percentual	6,9	6,9	6	6,9	2,6	6,9	5,2	6,9	6,9	5,2	6,9	6,9	6,9	6,9	5,2	6

De acordo com os percentuais da tabela 4.9, pode-se notar que 15 (quinze) sites apresentaram uma homepage apresentável, mas, o website do IBGE foi o que obteve a pior apresentação, prejudicando sua identidade. Portanto, para resolver este problema, é preciso seguir as seguintes recomendações (conforme a Tabela 4.10):



Tabela 4.10 - Síntese de melhorias para os critérios analisados quanto à homepage do website IBGE.

Critérios Analisados	Recomendações
Homepage não passa os valores da <i>WebApp</i>	Explicar claramente o qual é o foco da <i>WebApp</i> , isto é, o seu objetivo, suas informações principais e seus produtos. Não “poluindo” visualmente.
Informações desorganizadas	A homepage é a “vitrine” da <i>WebApp</i> , ou seja, organize as informações/produtos por categorias, fazendo-as mais fáceis de serem observadas pelo usuário.

• **Há um caminho livre para informações de contato?** Para ilustrar este problema, foi analisado 16 (dezesseis) websites. Cujo objetivo, é verificar se há uma forma de contato na *WebApp*, conforme a Figura 4.4.

Para realização da análise deste problema, foi analisado o seguinte critério e definição:

- forma de comunicação do visitante com a *WebApp*: deve-se ter uma forma de contato entre o visitante e a empresa melhorando o feedback, podendo ser por meio de telefone, chat, formulário eletrônico e redes sociais (*Twitter*, *Facebook*, *Google+*).

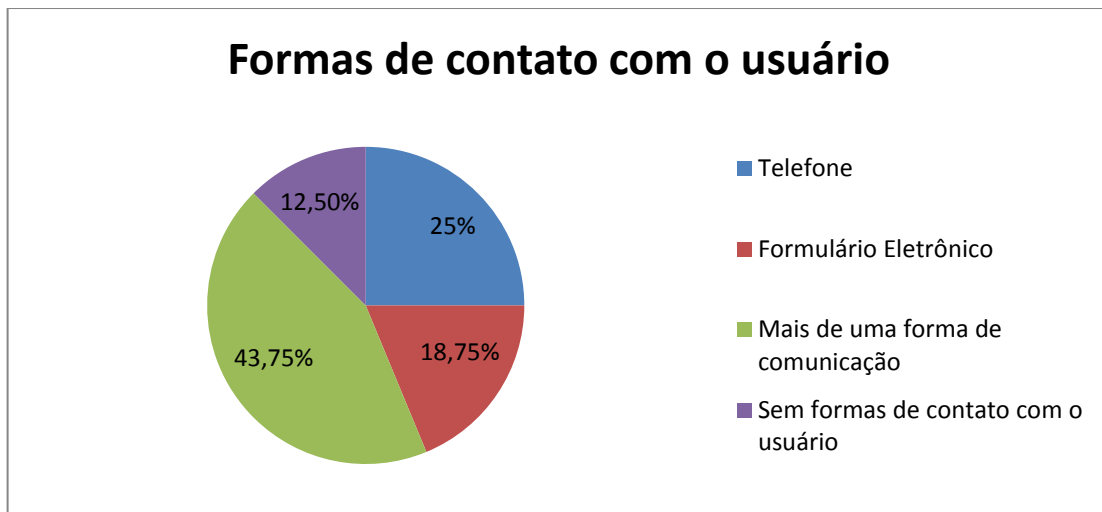


Figura 4.5: Gráfico que representa o percentual de uso da área de contato nos websites analisados.

Dentre os 16 (dezesseis) websites analisados, conforme a Figura 4.4. Pode-se notar que a maioria disponibiliza algum espaço em que o usuário entre em contato com a empresa, não prejudicando sua identidade (conforme a Figura 4.5). Para resolver este problema, basta deixar clara a forma de contato, por exemplo: “Ligue agora”, “Inicie um chat agora” ou “Fale conosco”. A ideia é não deixar o visitante sair da *WebApp* sem que tenha algum feedback com o visitante.

#### 4.1.3 Navegabilidade

Dentre os problemas de navegabilidade, foram analisados os seguintes: menus de navegação são claros e concisos, logotipo ligado a homepage, área de pesquisa no site. Cabe ressaltar, que foi escolhido apenas 3 (três) problemas essenciais, que podem comprometer a navegabilidade da *WebApp*.

• **Menus de navegação são claros e concisos?** Para ilustrar este problema, foi analisado 16 (dezesesseis) websites. Cujo objetivo, é verificar se o menu de navegação provê funcionalidades claras e objetivas a *WebApp*, conforme a Figura 4.6.

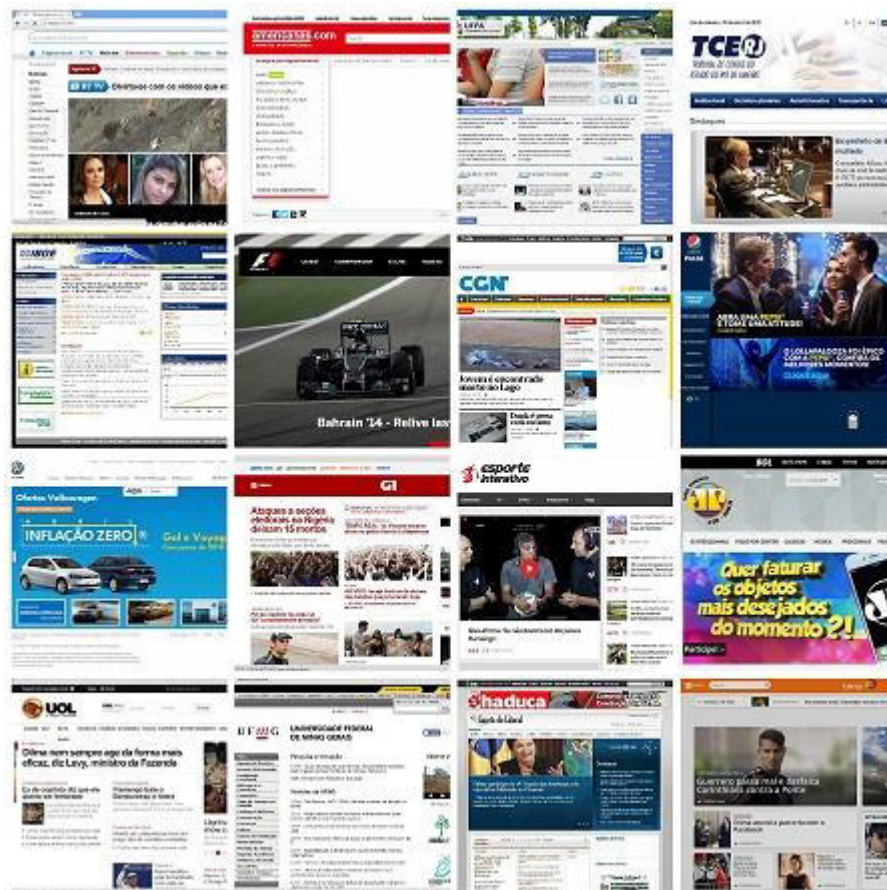


Figura 4.6: Websites diversos<sup>15</sup>.

<sup>15</sup> Websites analisados: R7, Americanas, UFPA, TCE/RJ (linha 1), 2015; IBGE, Fórmula 1, CGN, Pepsi (linha 2), 2015; Volkswagen, G1, Esporte Interativo, Jovem Pan FM (linha 3), 2015; e UOL, UFMG, Gazeta do Litoral, Terra (linha 4), 2015.

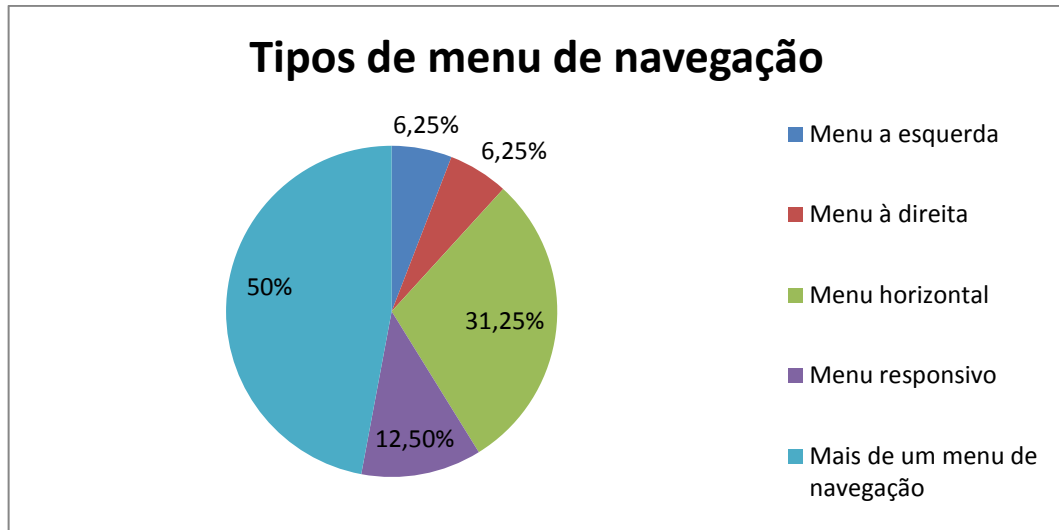


Figura 4.7: Gráfico que representa o percentual de uso de menus de navegação nos websites analisados.

Dentre os 16 (dezesseis) websites analisados, conforme a Figura 4.6. Pode-se notar que a maioria das *WebApps* utilizam mais de um menu de navegação, conforme a análise da Figura 4.7. Isso se deve a combinação mista entre o menu no topo com menus à direita e esquerda ou dois menus horizontais. Esse fato se deve, ao grande volume de informação contida na *WebApp*, com isso, é possível separar os menus de navegação em “primários”, aqueles que possuem maior importância, e “secundários”. Outro ponto a se destacar, conforme a Figura 4.7, é o uso de menu horizontal, isto é, o conteúdo da *WebApp* é ampliado possibilitando um grande movimento lateral e vertical no site.

Para realização da análise deste problema (conforme a Figura 4.7), foram analisados os seguintes critérios e definições:

- menu de navegação não objetivo: consiste em não ter as informações referentes da *WebApp*.
- desorganização dos itens no menu de navegação: consiste em ter um número exagerado de itens em cada categoria do menu de navegação da *WebApp*.
- divergência de estruturas de navegação: consiste em ter menus de navegação em locais diferentes nas páginas da *WebApp*, com cores diferentes, e com itens com nomes e em ordens diferentes, mesmo aqueles que se referem a assuntos iguais (apesar de levarem a páginas diferentes).

Tendo em vista estes critérios, para análise foi atribuído a seguinte convenção: (1) ruim – a *WebApp* segue a definição do critério analisado; (2) regular – a *WebApp* apresenta algumas melhorias quanto ao critério analisado, isto é, 50%

dos conteúdos estão organizados no menu; (3) bom – a *WebApp* apresenta uma melhoria considerável quanto ao critério analisado, isto é, 75% dos conteúdos estão organizados no menu e (4) muito bom – a *WebApp* não segue a definição do critério analisado.

Tabela 4.11 - Síntese de análise dos critérios de uma estrutura de navegação em um website.

Critérios Analisados/ Websites	R7	Americanas	UFPA	TCE/RJ	IBGE	Fórmula 1	CGN	Pepsi	Volkswagen	G1	Esporte Interativo	Jovem Pan FM	UOL	UFMG	Gazeta do Litoral	Terra
Menu de navegação não objetivo	1	4	3	4	3	4	4	4	4	4	4	3	3	3	4	2
Desorganização dos itens no menu de navegação	1	3	4	3	3	3	4	4	2	4	4	3	3	2	4	3
Divergência de estrutura de navegação	2	4	4	4	1	4	4	4	4	4	4	4	2	2	4	4
Percentual (%)	2,5	6,9	6,9	6,9	4,4	6,9	7,5	7,5	6,2	7,5	7,5	6,2	5,0	4,4	7,5	5,6

De acordo com os percentuais da tabela 4.11, pode-se notar que 15 (quinze) sites apresentaram uma boa estrutura de navegação, mas, o website do R7 foi o que obteve a pior navegação entre as páginas. Portanto, para resolver este problema, é preciso seguir as seguintes recomendações (conforme a Tabela 4.12):

Tabela 4.12 - Síntese de melhorias para os critérios analisados quanto à estrutura de navegação do website R7.

Critérios Analisados	Recomendações
Menu de navegação não objetivo	Categorizar as informações e organizar o menu de navegação de acordo com o foco da <i>WebApp</i> .
Desorganização dos itens no menu de navegação	Categorizar e diferenciar os itens de cada categoria, e, por conseguinte se houver agrupar os subitens de forma clara em cada item da categoria.
Divergência de estruturas de navegação	Manter a estrutura do menu de navegação da homepage nas páginas secundárias, isto é, sempre coloca-lo na mesma posição, com cores iguais e com mesmos nomes (caso se remetam aos mesmos itens da homepage).

- **Logotipo da empresa está ligado à homepage?** Para ilustrar este problema, foi analisado 16 (dezesseis) websites. Cujo objetivo, é verificar se o logotipo, podendo ser um símbolo ou o nome da empresa, está ligado à homepage

da *WebApp*, conforme a Figura 4.8.

Para realização da análise deste problema, foram analisados os seguintes critérios e definições:

- não atualização do conteúdo da homepage: o logotipo da homepage não permite a atualização do conteúdo através de um click sobre si próprio. Forçando o usuário, a atualizá-la por meio da tecla F5.
- logotipo das páginas secundárias não é linkável à homepage: o logotipo presente nas páginas secundárias não possui um link associado à homepage.

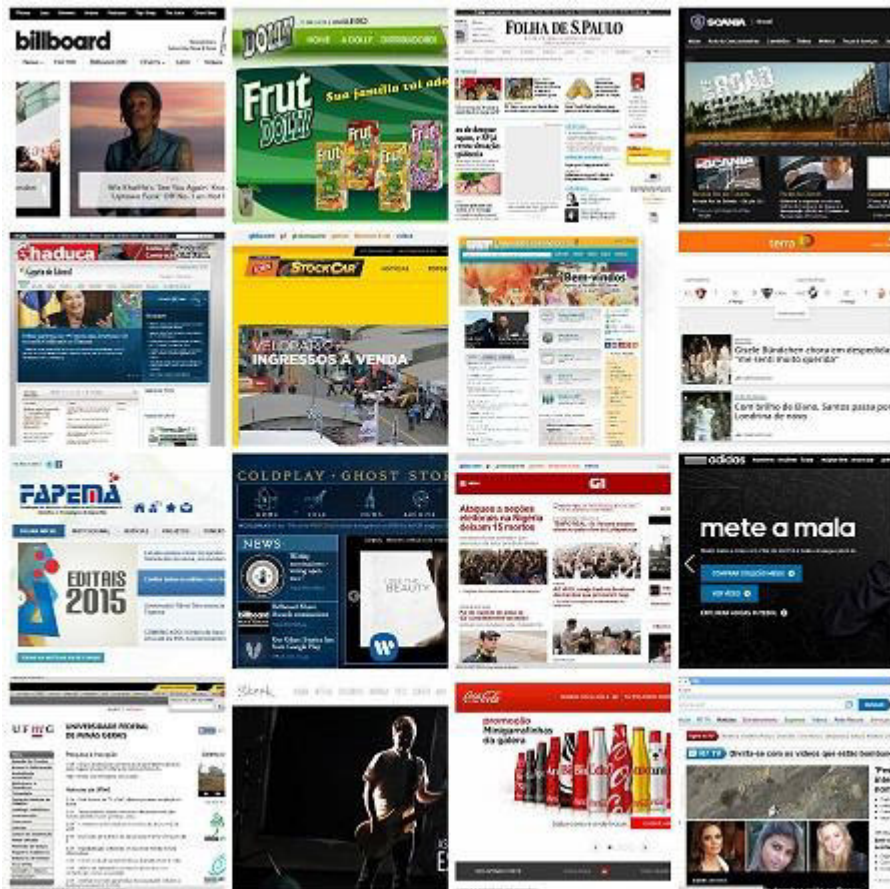


Figura 4.8: Websites diversos<sup>16</sup>.

Tendo em vista estes critérios, para análise foi atribuído a seguinte convenção: (1) ruim – a *WebApp* segue a definição do critério analisado; (2) regular – a *WebApp* apresenta algumas melhorias quanto ao critério analisado; (3) bom – a *WebApp* apresenta uma melhoria considerável quanto ao critério analisado e (4) muito bom – a *WebApp* não segue a definição do critério analisado.

<sup>16</sup> Websites analisados: Billboard, Guaraná Dolly, Folha de S. Paulo, Scania (linha 1), 2015; Gazeta do Litoral, Stock Car, USP, Terra (linha 2), 2015; FAPEMA, Coldplay, G1, Adidas (linha 3), 2015; e UFMG, Skank, Coca-Cola, R7 (linha 4), 2015.

Tabela 4.13 - Síntese de análise dos critérios de uso de um logotipo em um website.

Critérios Analisados/ Websites	Billboard	Guaraná Dolly	Folha de S. Paulo	Scania	Gazeta do Litoral	Stock Car	USP	Terra	FAPEMA	Coldplay	G1	Adidas	UFMG	Skank	Coca-Cola	R7
Não atualização do conteúdo da homepage	4	1	4	4	1	4	4	4	4	4	4	4	4	1	4	4
Logotipo das páginas secundárias não é linkável à homepage	4	4	4	4	1	4	4	4	4	4	4	4	4	4	1	4
Percentual (%)	7,0	4,4	7,0	7,0	1,7	7,0	7,0	7,0	7,0	7,0	7,0	7,0	7,0	4,4	4,4	7,0

De acordo com os percentuais da tabela 4.13, pode-se notar que 15 (quinze) sites apresentaram um bom uso do logotipo, mas, o website do Jornal Gazeta do Litoral foi o que obteve o pior uso do logotipo, prejudicando sua navegabilidade. Portanto, para resolver este problema, é preciso seguir as seguintes recomendações (conforme a Tabela 4.14):

Tabela 4.14 - Síntese de melhorias para os critérios analisados quanto ao uso do logotipo no website Gazeta do Litoral.

Critérios Analisados	Recomendações
Não atualização do conteúdo da homepage.	Permitir que o logotipo da homepage seja clicável, permitindo a atualização de conteúdo.
Logotipo das páginas secundárias não é linkável à homepage.	Permitir que o logotipo presente nas páginas secundárias possua um link associado à homepage.

• **Pesquisa no site é de fácil acesso?** Para ilustrar este problema, foi analisado 16 (dezesesseis) websites. Cujo objetivo, é verificar o uso da área de pesquisa na *WebApp*, conforme a Figura 4.8.

Para realização da análise deste problema, foram analisados os seguintes critérios e definições:

- campo de pesquisa não é visível na *WebApp*: consiste em ter um campo de pesquisa “escondido”, fazendo com que o usuário procure-o na *WebApp*.
- divergência de estruturas de navegação: consiste em ter um campo de pesquisa de diferentes partes nas páginas da *WebApp* e com tamanhos diferentes.

Tendo em vista estes critérios, para análise foi atribuído a seguinte

convenção: (1) ruim – a *WebApp* segue a definição do critério analisado; (2) regular – a *WebApp* apresenta algumas melhorias quanto ao critério analisado; (3) bom – a *WebApp* apresenta uma melhoria considerável quanto ao critério analisado e (4) muito bom – a *WebApp* não segue a definição do critério analisado.

Tabela 4.15 - Síntese de análise dos critérios de uso do campo de pesquisa em um website.

Critérios Analisados/ Websites	Billboard	Guaraná Dolly	Folha de S. Paulo	Scania	Gazeta do Litoral	Stock Car	USP	Terra	FAPEMA	Coldplay	G1	Adidas	UFMG	Skank	Coca-Cola	R7
Campo de pesquisa não é visível na WebApp	3	1	3	3	3	1	4	4	3	3	3	4	4	3	1	2
Divergência de estruturas de navegação	3	1	3	3	3	1	4	4	3	3	3	4	4	3	1	2
Percentual (%)	6,7	2,2	6,7	6,7	6,7	2,2	8,9	8,9	6,7	6,7	6,7	8,9	8,9	6,7	2,2	4,4

De acordo com os percentuais da tabela 4.15, pode-se notar que 13 (treze) sites apresentaram um bom uso da barra de pesquisa, mas, os websites do Guaraná Dolly, Stock Car e Coca-Cola foram o que obtiveram o pior uso desta ferramenta, prejudicando sua navegabilidade. Portanto, para resolver este problema, é preciso seguir as seguintes recomendações (conforme a Tabela 4.16):

Tabela 4.16 - Síntese de melhorias para os critérios analisados quanto ao uso da barra de pesquisa nos websites do Guaraná Dolly, Stock Car e Coca-Cola.

Critérios Analisados	Recomendações
Campo de pesquisa não é visível na WebApp	É fundamental que exista uma ferramenta de pesquisa nas páginas da <i>WebApp</i> , ou seja, coloque-a na parte superior da <i>WebApp</i> distinguindo-a dos elementos ao redor (utilize uma cor diferente), e associe com o nome “Pesquisa” ou com o ícone de pesquisa ao lado.
Divergência de estruturas de navegação	Manter a estrutura da barra de pesquisa da homepage nas páginas secundárias, isto é, sempre colocá-la na mesma posição das páginas.

#### 4.1.4 Conteúdo

Dentre os problemas de conteúdo, foram analisados os seguintes: o uso de

anúncios e pop-ups e os títulos de páginas da *WebApp*. Cabe ressaltar, que foi escolhido apenas 2 (dois) problemas essenciais, que podem comprometer o conteúdo da *WebApp*.

- **Anúncios e pop-ups são discretos?** Para ilustrar este problema, foi analisado 16 (dezesesseis) websites. Cujo objetivo, é verificar o uso de anúncios e pop-ups nas páginas da *WebApp*, conforme a Figura 4.9.

Para realização da análise deste problema, foram analisados os seguintes critérios e definições:

- uso exagerado de anúncios nas páginas: consiste em usar um número exagerado de anúncios nas páginas comprometendo a sua visibilidade.
- uso do mesmo pop-up em diversas páginas: consiste em usar o mesmo pop-up em diversas páginas da *WebApp* fazendo com que a navegação fique cansativa e nada dinâmica.
- uso de pop-ups não alinhados ao conteúdo: consiste em colocar pop-ups não relacionados ao conteúdo da *WebApp* fazendo com que o público alvo desista da *WebApp*.



Figura 4.9: Websites diversos<sup>17</sup>.

<sup>17</sup> Websites analisados: NovoMundo, CityLar, Claro, R7 (linha 1), 2015; G1, NET, Riachuelo, RD1 (linha 2), 2015; Oi, IBGE, Embratel, UOL (linha 3), 2015; e Band.com.br, Folha de S. Paulo, Vivo, Casas Bahia (linha 4), 2015.



Tendo em vista estes critérios, para análise foi atribuído a seguinte convenção: (1) ruim – a *WebApp* segue a definição do critério analisado; (2) regular – a *WebApp* apresenta algumas melhorias quanto ao critério analisado, isto é, anúncios ocupam em média 75% da página; (3) bom – a *WebApp* apresenta uma melhoria considerável quanto ao critério analisado, isto é, anúncios ocupam em média 50% da página e (4) muito bom – a *WebApp* não segue a definição do critério analisado.

Tabela 4.17 - Síntese de análise critérios de uso de anúncios e pop-ups em um website.

Critérios Analisados/ Websites	NovoMundo	CityLar	Claro	R7	G1	NET	Riachuelo	RD1	Oi	IBGE	Embratel	UOL	Band.com.br	Folha de S. Paulo	Vivo	Casas Bahia
Uso exagerado de anúncios nas páginas	3	3	4	2	4	3	3	3	4	4	4	3	4	1	3	2
Uso do mesmo pop-up em diversas páginas	4	4	3	4	4	4	4	2	4	4	4	4	4	4	3	4
Uso de pop-ups não alinhados ao conteúdo	4	4	3	4	4	4	4	2	4	4	4	4	4	3	4	4
Percentual (%)	6,4	6,4	5,8	5,8	7,0	6,4	6,4	4,0	7,0	7,0	7,0	6,4	7,0	4,6	5,8	5,8

De acordo com os percentuais da tabela 4.17, pode-se notar que 15 (quinze) sites apresentaram um bom uso de pop-ups e anúncios em suas páginas, mas, o website do RD1 foi o que obteve o pior uso, prejudicando seu conteúdo. Portanto, para resolver este problema, é preciso seguir as seguintes recomendações (conforme a Tabela 4.18):

Tabela 4.18 - Síntese de melhorias para os critérios analisados quanto o uso de anúncios e pop-ups no website do RD1.

Critérios Analisados	Recomendações
Uso exagerado de anúncios nas páginas	Evite “poluir” as páginas com anúncios, ou seja, distribua os anúncios entre as páginas da <i>WebApp</i> .
Uso do mesmo pop-up em diversas páginas	É saber em quais páginas o pop-up será utilizado.
Uso de pop-ups não alinhados ao conteúdo	Colocar pop-ups relacionados ao conteúdo da <i>WebApp</i> para harmonizar a plataforma, e logo, atrair o público-alvo.

• **Títulos de páginas HTML são explicativos?** Para ilustrar este problema, foi analisado 16 (dezesseis) websites. Cujo objetivo, é verificar se os títulos das páginas da *WebApp* são descritivos ao conteúdo apresentado na página, conforme a Figura 4.9.

Para realização da análise deste problema, foram analisados os seguintes critérios e definições:

- uso de títulos não descritivos nas páginas: consiste em usar títulos completamente diferentes ao conteúdo apresentado pela página.
- uso de títulos repetidos ou clichês: consiste em acrescentar títulos para cada página em um website como “produtos baratos para venda”, por exemplo, torna impossível para os usuários distinguir uma página de outra.

Tendo em vista estes critérios, para análise foi atribuído a seguinte convenção: (1) ruim – a *WebApp* segue a definição do critério analisado; (2) regular – a *WebApp* apresenta algumas melhorias quanto ao critério analisado; (3) bom – a *WebApp* apresenta uma melhoria considerável quanto ao critério analisado e (4) muito bom – a *WebApp* não segue a definição do critério analisado.

Tabela 4.19 - Síntese de análise dos critérios de uso de títulos nas páginas em um website.

Critérios Analisados/ Websites	NovoMundo	CityLar	Claro	R7	G1	NET	Riachuelo	RD1	Oi	IBGE	Embratel	UOL	Band.com.br	Folha de S. Paulo	Vivo	Casas Bahia
Uso de títulos não descritivos nas páginas	4	4	4	4	4	4	4	4	4	2	4	4	4	4	2	4
Uso de títulos repetidos ou clichês	4	4	4	4	4	4	4	4	4	1	4	4	4	4	1	4
Percentual (%)	6,7	6,7	6,7	6,7	6,7	6,7	6,7	6,7	6,7	2,5	6,7	6,7	6,7	6,7	2,5	6,7

De acordo com os percentuais da tabela 4.19, pode-se notar que 14 (quatorze) sites apresentaram um bom uso de nomear os títulos de suas páginas, mas, os website do IBGE e da Vivo foram o que obtiveram o pior uso, prejudicando seu conteúdo. Portanto, para resolver este problema, é preciso seguir as seguintes recomendações (conforme a Tabela 4.20):

Tabela 4.20 - Síntese de melhorias para os critérios analisados quanto ao uso de títulos das páginas nos websites do IBGE e da Vivo.

<b>Critérios Analisados</b>	<b>Recomendações</b>
Uso de títulos não descritivos nas páginas	Crie um título que descreva de maneira objetiva o conteúdo que o visitante irá encontrar.
Uso de títulos repetidos ou clichês	Às vezes, é útil ter alguns termos descritivos no título, mas não há razão para ter as mesmas palavras ou frases aparecendo várias vezes. Portanto, é fundamental ter títulos descritivos e distintos em cada página.

Portanto, tendo em vista estes aspectos sobre a usabilidade nas *WebApps*, através das melhorias propostas neste capítulo, será possível cada mais desenvolver websites fazendo com que os usuários realizem uma tarefa específica de maneira fácil e rápida.

## 5 CONCLUSÃO

Este trabalho investigou a qualidade nas *WebApps* através da análise de diversos pontos de usabilidade, cujo objetivo é propor melhorias na capacidade das *WebApps* em ser compreendida, aprendida, operada e atraente. Para isso, foi realizado uma pesquisa em diversas *WebApps* com intuito de analisar diversos pontos de usabilidade e, com isso, propor melhorias nos aspectos analisados.

O passo inicial consistiu em determinar quais problemas seriam analisados dentro de cada ponto de usabilidade, ou seja, para evitar uma repetição durante a análise, foram escolhidos apenas aqueles problemas essenciais nas *WebApps* durante a interação usuário e *WebApp*. E também, aqueles que podem ser observados facilmente em diversas *WebApps*. Logo em seguida, foram escolhidas as *WebApps* de cada problema, isto é, foi feita uma vasta pesquisa de *WebApps* que apresentasse o problema a ser analisado. Durante essa fase inicial, a dificuldade encontrada foi realizar essa pesquisa de *WebApps* com intuito de verificar quais problemas estavam presentes nas páginas.

Uma etapa, desenvolvida em paralelo, foi à criação de critérios dentro daqueles problemas analisados. Estes critérios foram criados de forma que fosse capaz de verificar se a *WebApp* cumpria ou não com a definição do critério, caso cumprisse com a definição, este problema deverá ser melhorado dentro da *WebApp* aperfeiçoando sua usabilidade. Uma característica importante a se observar, é que alguns problemas dentro daquele ponto apresentaram *WebApps* específicas, isto é, que apresentam o mesmo foco. Isso se deve ao fato de que aquele problema pode ser mais facilmente observável naquele tipo de *WebApp* específica. Em contra partida, outros problemas apresentaram *WebApps* com diferentes focos, ou seja, aquele problema não é observável em apenas um tipo específico de *WebApp*. A dificuldade desta etapa, foi verificar quais *WebApps* apresentavam um foco específico dentro do ponto analisado e, quais poderiam não ser observável apenas naquele ponto.

Durante a fase de análise, foram feitos vários testes de forma que fosse verificada a eficiência de cada critério. Ou seja, a cada análise foi atribuído um índice que indica seu grau de desempenho quanto ao critério analisado. A dificuldade desta etapa, é que durante os testes, foi possível perceber que algumas *WebApps* apresentaram uma lentidão de suas páginas. Isso se deve, a

desorganização e ao uso exagerado de diversos tipos de informações em apenas uma página da *WebApp*. Ou seja, para que esse fato não prejudicasse a análise, foi estimado 5 (cinco) segundos como toleráveis para o carregamento da página. Logo em seguida, foi atribuído um índice quanto a especificação do critério analisado e assim sucessivamente para todas *WebApps*. Com isso, a *WebApp* que apresentar a menor porcentagem dentre as diversas *WebApps*, será proposto as possíveis melhorias para aquele problema na *WebApp* melhorando sua usabilidade.

De maneira geral, a metodologia apresentada se mostrou bastante funcional de forma que trabalhos futuros envolvendo qualidade nas *WebApps* possam ser melhorados. Utilizando a metodologia apresentada, poder-se-á desenvolver aplicações web cada vez mais eficientes quanto à usabilidade, por exemplo: construção de sites dinâmicos ou estáticos, a fim de que as pessoas possam atingir objetivos específicos com eficácia, eficiência e satisfação para realização de uma tarefa específica e importante.

## REFERÊNCIAS

- BREVE, F. A. **Engenharia para web**. Disponível em: <[http://www.fabriciobreve.com/trabalhos/engenharia\\_web.php#\\_Toc10877457](http://www.fabriciobreve.com/trabalhos/engenharia_web.php#_Toc10877457)>. Acesso em: 20 jan. 2015.
- CARVALHO, V. **10 Dicas de usabilidade para web designers**. Disponível em: <<https://mestrejquery.wordpress.com/2013/07/28/10-dicas-de-usabilidade-para-web-designers/>>. Acesso em: 12 mar. 2015.
- GATTI, C. A. M.; WERNECK, B. M. V. **Requisitos de Qualidade de Aplicações Web**. Rio de Janeiro, v. 17, p. 15 – 03, mar. 2004.
- GONÇALVES, J. **Web 2.0 – a primeira evolução da internet**. Disponível em: <<http://hiperbytes.com.br/geral/web-2-0-a-primeira-evolucao-da-internet/>>. Acesso em: 03 jan. 2015.
- GONÇALVES, K. M. **Usabilidade de Software: Estudo de Recomendações Básicas para Verificação do Nível de Conhecimento dos Alunos dos Cursos de Design Gráfico e Sistemas de Informação da UNESP/BAURU**. Dissertação (Mestrado). Universidade Estadual Paulista. Faculdade de Arquitetura, Artes e Comunicação, Bauru, 2008.
- IMMICH, S. L. **Avaliação Heurística da Usabilidade do Website do IBGE – Instituto Brasileiro de Geografia e Estatística**. Trabalho de Conclusão de Curso. Universidade Federal do Rio Grande do Sul, 2011.
- JAMIL, V. **Usabilidade e seu Impacto em Aplicações Web**. Disponível em: <<http://www.scriptcase.com.br/blog/usabilidade-aplicacoes-web/>>. Acesso em: 03 abr. 2015.
- LEE, H. M. L. **Qualidade de Websites e sua Usabilidade**. Trabalho de Conclusão de Curso. Faculdade de Tecnologia da Zona Leste, 2009.
- LIMA, K. **A Evolução da Web 1.0 x 2.0 x 3.0**. Disponível em: <<https://agrecom.wordpress.com/2014/09/09/a-evolucao-da-web-1-0-x-2-0-x-3-0/>>. Acesso em: 02 jan. 2015.
- LUIZA, V. **Análise dos portais G1 e R7**. Disponível em: <<http://pt.slideshare.net/jonathanprateat/anlise-de-usabilidade-sobre-g1-e-r7>>. Acesso em: 27 abr. 2015.
- MANTEIGA, S. M. **O Cone de Incerteza em Projetos**. Disponível em: <<http://sandromanteiga.com/o-cone-de-incerteza-do-prazo-de-projetos/>>. Acesso em: 21 jan. 2015.
- MATOS, A.; BRITTO, T. **Engenharia Web**. Apostila. Universidade do Sagrado Coração Bauru, 2007.

MICHELLE, N. **Web Semântica: em busca da web inteligente**. Disponível em: <<http://pt.slideshare.net/nairamichelle/web-semntica-em-busca-da-web-inteligente>>. Acesso em: 10 jan. 2015.

MORGADO, P. **Gerência de riscos em desenvolvimento de software**. Disponível em: <<http://www.devmedia.com.br/gerencia-de-riscos-em-desenvolvimento-de-software/28506>>. Acesso em: 21 jan. 2015.

MOULIN, R. **O que é e como fazer uma avaliação heurística**. Disponível em: <<http://www.designinterativo.etc.br/arquitetura-de-informacao/o-que-e-e-como-fazer-uma-avaliacao-heuristica>>. Acesso em: 02 mar. 2015.

NETO, D. C. A. **Metodologias de desenvolvimento de aplicações Web – Parte 5**. Disponível em: <<http://www.devmedia.com.br/metodologias-de-desenvolvimento-de-aplicacoes-web-parte-05/9820>>. Acesso em: 03 fev. 2015.

NIELSEN, J.; LORANGER, H. (1957). **Usabilidade na Web** - Projetando Websites com Qualidade. Trad. Sob a direção de Edson Fumankiewicz e Carlos Schafranski. Rio de Janeiro: Campus, 2007.

PRESSMAN, S. R. **Engenharia Web**. Trad. Sob a direção de Daniel Vieira; revisão técnica Delano Medeiros Beder, Marcos Lordello Chaim, Paulo César Masiero. Rio de Janeiro, LTC, 2007.

OSORIO, L. **Aplicações web – conceitos**. Disponível em: <<http://pt.slideshare.net/lmcosorio/3-aplicacoes-web-conceitos>>. Acesso em: 20 fev. 2015.

REBELO, B. I. **Interação entre homem e computador**. Apostila. Centro Universitário UNIEURO, 2009.

ROCKER, P. **A Origem e Evolução da Internet**. Disponível em: <[http://pt.slideshare.net/Pepe\\_Rocker/a-origem-e-evolucao-da-internet](http://pt.slideshare.net/Pepe_Rocker/a-origem-e-evolucao-da-internet)>. Acesso em: 08 jan. 2015.

VINHAIS, A. **10 Dicas de Usabilidade e Navegabilidade para o seu Site**. Disponível em: <<http://escolablog.com/dicas-2/10-dicas-de-usabilidade-e-navegabilidade-para-o-seu-site/>>. Acesso em: 10 mar. 2015.

ZANONI, C. **Testes em Aplicações Web**. Disponível em: <<http://pt.slideshare.net/cynthiazanoni/testes-em-aplicacoes-web>>. Acesso em: 29 jan. 2015.