

**UNIVERSIDADE DE CAXIAS DO SUL – UCS
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

MAURÍCIO GARDINI

**UMA PROPOSTA DE RECOMENDAÇÕES PARA DESIGN DE
INTERFACES *WEB* MÓVEIS**

CAXIAS DO SUL

2015

MAURÍCIO GARDINI

**UMA PROPOSTA DE RECOMENDAÇÕES PARA DESIGN DE
INTERFACES *WEB* MÓVEIS**

Orientadora: Prof^a. Dr^a. Elisa Boff

CAXIAS DO SUL

2015

Este trabalho é dedicado a todos aqueles que acreditam que conhecimento nunca é demais.

Agradecimentos

Agradeço aos meus pais, Valtoir e Carmen, por terem me dado a estrutura que eu precisei para atingir meus objetivos e por terem me ensinado a ser correto e honesto, acima de tudo; ao meu irmão, Alessandro, que, juntamente de meus pais, forma a minha família, que é a base de tudo; à Luciane, por todo o amor, companheirismo e apoio; à minha professora orientadora, Elisa Boff, por sua sabedoria, atenção e paciência.

Agradeço também aos professores e colegas de curso e de centro, pelas vossas amizades, pelos conhecimentos transmitidos e pelos momentos de descontração proporcionados.

“...Not all those who wander are lost ...”
(J. R. R. Tolkien)

Resumo

Por muito tempo, a maior parte dos usuários da *internet* usava os *desktops* como principais dispositivos. Por tal razão, o foco do estudo do *design* de interação do usuário e do desenvolvimento de *websites* foi nestes dispositivos.

A popularização de *smartphones* e *tablets*, porém, vem trazendo mudanças na forma pela qual as pessoas acessam a *internet*, fazendo com que muitos passem a usar este tipo de dispositivo como a principal forma de se navegar. E, em parte graças à quão recentemente este êxodo de usuários *desktop* para os dispositivos móveis ocorreu, ainda existem poucos materiais que tem como foco o desenvolvimento estes dispositivos.

O objetivo deste trabalho é criar um guia de recomendações para *design* de interfaces *web* móveis. Para se alcançar este objetivo, são estudadas as disciplinas que abrangem o *Design* de Experiência do Usuário. Também, abordada-se com mais detalhes diretrizes que foram identificadas por outros autores que visam o *design* de interfaces com boa Usabilidade e Acessibilidade. A filosofia *Mobile First* também é explorada, de forma a identificar as motivações e particularidades de se começar a desenvolver interfaces a partir dos dispositivos móveis. Após, realiza-se a implementação de um protótipo com base no guia. Por fim, realiza-se a avaliação heurística do protótipo criado, de forma a validar o guia.

Palavras-chave: Desenvolvimento *Web* Móvel, Dispositivos Móveis, Usabilidade, Acessibilidade, *Mobile First*, *Design* de Experiência do Usuário

Abstract

For a long time, most of the internet users used desktops as their main devices. For that reason, the focus of both the user interaction design study and of websites development was in those devices.

The popularization of smartphones and tablets, however, is changing the way people access the internet, and many people are using this kind of device as the main way to browse the web. And, partially thanks to how recently this user exodus from desktop to mobile devices has happened, still there are a few materials which focus on the development in those devices.

The goal of this work is to create a guide with interface design recommendations targeted to mobile web interfaces. To reach this goal, the subjects that cover the User Experience Design are studied. Also, the guidelines that were identified by other authors that endorses the interface design with good Usability and Aecessibility are covered with more detail. The Mobile First philosophy also is explored, intending to identify the motivations and particularities of creating interfaces starting by mobile devices. After that, a prototype is implemented based on the guide. Finally, the heuristic evaluation of the prototype is performed, with the goal to validate the guide.

Keywords: Mobile Web Development, Mobile Devices, Usability, Aecessibility, Mobile First, User Experience Design

Lista de ilustrações

Figura 1 – Proficiência e eficiência de uso com relação ao tempo	52
Figura 2 – Total de tempo de uso de mídia digital gasto por plataforma	65
Figura 3 – Previsão de tráfego de dados oriundos de <i>internet</i> móvel por mês em 2019	66
Figura 4 – Facilidade da realização do toque em diferentes áreas da tela quando se está segurando o telefone apenas com a mão direita	70
Figura 5 – Tela da página inicial.	96
Figura 6 – Tela da navegação.	97
Figura 7 – Tela de Entrar/Criar conta.	97
Figura 8 – Tela de Busca.	98
Figura 9 – Tela do Catálogo.	99
Figura 10 – Tela de Detalhes.	99
Figura 11 – Tela da Cesta de Compras.	100
Figura 12 – Tela da Conta do usuário.	100
Figura 13 – Tela de fechamento de pedido.	101
Figura 14 – Diagrama de atividades da realização de um pedido.	102
Figura 15 – <i>Screenshot</i> da página inicial.	104
Figura 16 – <i>Screenshot</i> da página de Entrar/Criar conta em três variações: sem erros de preenchimento, com erros de preenchimento no formulário Entrar e com erros de preenchimento no formulário Criar conta.	106
Figura 17 – <i>Screenshot</i> da Navegação em duas variações: como ela é apresentada caso não haja usuário autenticado e como ela é apresentada para um usuário autenticado.	107
Figura 18 – <i>Screenshot</i> da página de Busca em três variações: com resultados e sem a exibição dos filtros, com resultados e com a exibição dos filtros e sem resultados.	109
Figura 19 – <i>Screenshot</i> da página do Catálogo em duas variações: sem a exibição das categorias e filtros e com a exibição das categorias e filtros.	110
Figura 20 – <i>Screenshot</i> da página de Detalhes em quatro variações: como ela é apresentada quando o produto está em estoque caso não haja usuário autenticado e caso haja, e como ela é apresentada quando o produto está esgotado caso não haja usuário autenticado e caso haja.	112
Figura 21 – <i>Screenshot</i> da página da listagem de avaliações de um produto.	114
Figura 22 – <i>Screenshot</i> da página de detalhes de uma avaliação de um produto.	116
Figura 23 – <i>Screenshot</i> da página da Cesta de Compras em duas variações: com itens e vazia.	117

Figura 24 – <i>Screenshot</i> da página da Conta do usuário autenticado, na seção do Perfil, em duas variações: sem a exibição das seções e com a exibição das seções.	119
Figura 25 – <i>Screenshot</i> da página da Conta do usuário autenticado, durante a ação de edição do perfil em duas variações: sem erros de preenchimento e com erros de preenchimento.	120
Figura 26 – <i>Screenshot</i> da página da Conta do usuário autenticado, durante a ação de alteração da senha em duas variações: sem erros de preenchimento e com erros de preenchimento.	122
Figura 27 – <i>Screenshot</i> da página da Conta do usuário autenticado, durante a ação de remoção da conta em duas variações: sem erros de preenchimento e com erros de preenchimento.	123
Figura 28 – <i>Screenshot</i> da página da Conta do usuário autenticado, na seção da Lista de Endereços, em duas variações: com endereços e vazia.	125
Figura 29 – <i>Screenshot</i> da página da Conta do usuário autenticado, na seção da Lista de Endereços, durante a ação de adição de um novo endereço, em duas variações: sem erros de preenchimento e com erros de preenchimento.	127
Figura 30 – <i>Screenshot</i> da página da Conta do usuário autenticado, na seção da Lista de Endereços, durante a ação de remoção de um endereço.	128
Figura 31 – <i>Screenshot</i> da página da Conta do usuário autenticado, na seção das Listas de Desejos, em duas variações: com itens e vazia.	130
Figura 32 – <i>Screenshot</i> da página da Conta do usuário autenticado, na seção das Listas de Desejos, durante a ação de criação de uma nova lista de desejos, em duas variações: sem erros de preenchimento e com erros de preenchimento.	131
Figura 33 – <i>Screenshot</i> da página da Conta do usuário autenticado, na seção das Listas de Desejos, quando exibindo os detalhes de uma lista de desejos em específico, em duas variações: com itens e vazia.	132
Figura 34 – <i>Screenshot</i> da página da Conta do usuário autenticado, na seção das Listas de Desejos, durante a ação de remoção de um item de uma lista de desejos.	134
Figura 35 – <i>Screenshot</i> da página da Conta do usuário autenticado, na seção das Listas de Desejos, durante a ação de remoção de uma lista de desejos. .	135
Figura 36 – <i>Screenshot</i> da página da Conta do usuário autenticado, na seção do Histórico de Pedidos, em duas variações: sem pedidos e com pedidos. .	136
Figura 37 – <i>Screenshot</i> da página da Conta do usuário autenticado, na seção do Histórico de Pedidos, quando exibindo os detalhes de um pedido. . . .	138

Figura 38 – <i>Screenshot</i> da página de fechamento de pedido, durante o passo de seleção de endereço de entrega, em duas variações: sem erros de preenchimento e com erros de preenchimento.	139
Figura 39 – <i>Screenshot</i> da página de fechamento de pedido, durante o passo de seleção da forma de pagamento.	141
Figura 40 – <i>Screenshot</i> da página de fechamento de pedido, durante o passo da exibição do resumo do pedido.	142
Figura 41 – <i>Screenshot</i> da página de fechamento de pedido, durante o passo da confirmação do sucesso na realização do pedido.	144
Figura 42 – <i>Screenshot</i> da página de erro 404.	145

Lista de abreviaturas e siglas

ARIA	Accessible Rich Internet Applications
CEP	Código de Endereçamento Postal
CSS	Cascading Style Sheets
CSS3	Cascading Style Sheets 3
FAQ	Frequently Asked Questions
GPS	Global Positioning System
GUI	Graphical User Interface
HTML	HyperText Markup Language
NUI	Natural User Interface
NFC	Near Field Communication
PDF	Portable Document Format
QR Code	Quick Response Code
SMIL	Synchronized Multimedia Integration Language
WAI	Web Aecessibility Initiative
WCAG	Web Content Aecessibility Guidelines
W3C	World Wide Web Consortium
URL	Uniform Resource Locator
XHTML	eXtensible HyperText Markup Language

Sumário

1	INTRODUÇÃO	21
1.1	Objetivos	21
1.2	Estrutura do Texto	22
2	DESIGN DE EXPERIÊNCIA DO USUÁRIO	23
2.1	Design Visual	23
2.1.1	Princípios do <i>design</i>	23
2.1.2	Componentes do <i>design</i>	26
2.2	Arquitetura da Informação	27
2.2.1	Sistemas de Organização	27
2.2.1.1	Esquemas de Organização	28
2.2.1.2	Estruturas de Organização	29
2.2.2	Sistemas de Rotulação/Nomenclatura	29
2.2.2.1	Rótulos textuais	30
2.2.2.2	Rótulos icônicos	31
2.2.3	Sistemas de Navegação	31
2.2.3.1	Embutidos	32
2.2.3.2	Suplementares	32
2.2.3.3	Abordagens avançadas	33
2.2.4	Sistemas de Busca	33
2.2.4.1	Foco da busca	34
2.2.4.2	Algoritmos de busca	34
2.2.4.3	Estrutura de apresentação dos resultados	34
2.2.4.4	A Interface de Busca	34
2.3	Design de Interação	35
2.3.1	A Trajetória da Pesquisa e Desenvolvimento de Interface	35
2.3.2	Princípios do <i>Design</i> de Interação	37
2.3.3	Padrões do <i>Design</i> de Interação	38
2.4	Engenharia da Usabilidade	38
2.4.1	Ergonomia	39
2.4.1.1	Condução	39
2.4.1.2	Carga de Trabalho	40
2.4.1.3	Controle Explícito	40
2.4.1.4	Adaptabilidade	41
2.4.1.5	Gerenciamento de Erros	41

2.4.1.6	Consistência	42
2.4.1.7	Significado dos Códigos	42
2.4.1.8	Compatibilidade	42
2.5	Engenharia do Protótipo	42
2.6	Considerações sobre o Capítulo	43
3	USABILIDADE E ACESSIBILIDADE	45
3.1	Acessibilidade	45
3.1.1	WCAG 2.0	45
3.1.2	Técnicas para a aplicação do WCAG	47
3.1.2.1	Técnicas Gerais	47
3.1.2.2	Técnicas de HTML e XHTML	50
3.1.2.3	Técnicas de CSS	50
3.2	Usabilidade	51
3.2.1	Heurísticas de Usabilidade	53
3.2.2	Diretrizes de <i>Web Design</i> e Usabilidade Baseadas em Pesquisa	55
3.2.2.1	Processo de Design e Avaliação	55
3.2.2.2	Otimização da Experiência do Usuário	56
3.2.2.3	Acessibilidade	56
3.2.2.4	<i>Hardware</i> e <i>Software</i>	57
3.2.2.5	Página Inicial	57
3.2.2.6	<i>Layout</i> da Página	57
3.2.2.7	Navegação	58
3.2.2.8	Rolagem da Tela e Paginação	59
3.2.2.9	Cabeçalhos, Títulos e Rótulos	59
3.2.2.10	<i>Links</i>	59
3.2.2.11	Aparência do Texto	59
3.2.2.12	Listas	60
3.2.2.13	<i>Widgets</i>	60
3.2.2.14	Gráficos, Imagens e Multimídia	61
3.2.2.15	Escrevendo Conteúdo para a <i>Internet</i>	61
3.2.2.16	Busca	61
3.3	Avaliação de Usabilidade	62
3.3.1	Avaliação através da análise de especialistas	62
3.3.2	Avaliação através da análise de usuários	62
3.4	Considerações sobre o Capítulo	63
4	MOBILE FIRST	65
4.1	Limitações	67
4.2	Organização	68

4.3	Ações do Usuário	69
4.4	Entrada de Dados	70
4.5	Layout	71
4.6	Considerações sobre o Capítulo	71
5	GUIA DE RECOMENDAÇÕES PARA DESIGN DE INTERFACES WEB MÓVEIS	73
6	DESENVOLVIMENTO	81
6.1	Levantamento de Requisitos	81
6.2	Requisitos Funcionais	82
6.3	Casos de Uso	83
6.3.1	UC-01: Cadastrar novo cliente	83
6.3.2	UC-02: Autenticar cliente	84
6.3.3	UC-03: Consultar pedidos	85
6.3.4	UC-04: Gerenciar dados pessoais	86
6.3.5	UC-05: Adicionar produto ao carrinho de compras	86
6.3.6	UC-06: Alterar a quantidade de um produto já presente no carrinho de compras	87
6.3.7	UC-07: Remover um produto do carrinho de compras	87
6.3.8	UC-08: Finalizar um pedido	88
6.3.9	UC-09: Adicionar um possível endereço de entrega	89
6.3.10	UC-10: Modificar um possível endereço de entrega	89
6.3.11	UC-11: Excluir um possível endereço de entrega	90
6.3.12	UC-12: Avaliar um produto	91
6.3.13	UC-13: Adicionar produto à lista de desejos	91
6.3.14	UC-14: Remover um produto da lista de desejos	92
6.3.15	UC-15: Buscar por um produto	92
6.4	Implementação	93
6.4.1	Componentes	93
6.4.1.1	Python	93
6.4.1.2	Django	93
6.4.1.3	Oscar	93
6.4.2	Customização	94
6.4.3	Identidade Visual	94
6.4.4	Limitações	95
6.5	Telas	96
6.6	Diagrama de atividades	101
7	VALIDAÇÃO E RESULTADOS	103
7.1	Modelo	103

7.2	Avaliação dos cenários	104
7.2.1	Página inicial	104
7.2.2	Entrar/Criar conta	105
7.2.3	Navegação	107
7.2.4	Busca	108
7.2.5	Catálogo	110
7.2.6	Detalhes	111
7.2.7	Avaliações de um produto	114
7.2.8	Avaliações de um produto - Avaliação	116
7.2.9	Cesta de Compras	117
7.2.10	Conta - Perfil	118
7.2.11	Conta - Editar Perfil	120
7.2.12	Conta - Alterar senha	121
7.2.13	Conta - Remover conta	123
7.2.14	Conta - Lista de Endereços	125
7.2.15	Conta - Lista de Endereços - Adicionar novo endereço	126
7.2.16	Conta - Lista de Endereços - Remover endereço	128
7.2.17	Conta - Listas de Desejos	129
7.2.18	Conta - Listas de Desejos - Criar uma nova lista de desejos	131
7.2.19	Conta - Listas de Desejos - Lista de desejos	132
7.2.20	Conta - Listas de Desejos - Remover item de uma lista	133
7.2.21	Conta - Listas de Desejos - Remover lista	135
7.2.22	Conta - Histórico de pedidos	136
7.2.23	Conta - Histórico de pedidos - Pedido	137
7.2.24	Fechamento de pedido - Seleção de endereço de entrega	139
7.2.25	Fechamento de pedido - Seleção da forma de pagamento	141
7.2.26	Fechamento de pedido - Resumo do pedido	142
7.2.27	Fechamento de pedido - Confirmação do pedido	143
7.2.28	Erro 404 (Página não encontrada)	145
8	CONSIDERAÇÕES FINAIS	147
	Referências	149

1 Introdução

A *internet* já é algo que está inserido completamente na vida de boa parte do mundo e, a cada ano, cresce o número de pessoas que tem acesso à ela (ITU, 2014). Porém, tem-se notado que ela já não é mais acessada da mesma forma que era acessada até pouco tempo atrás.

De uns anos para cá, verificou-se uma crescente adesão ao acesso à *internet* através de dispositivos móveis, como os *smartphones* e *tablets*, o que por contrapartida ocasiona o decréscimo de acessos vindos de *desktops* e *notebooks* (COMSCORE, 2014).

O problema desta nova realidade é que a forma a qual o desenvolvimento *web* é realizado ainda hoje por muitos desenvolvedores já não é a mais apropriada. Segundo Wroblewski (2011), por muitos anos a maior parte dos times de desenvolvimento iniciava o planejamento de produtos e serviços tendo como foco os dispositivos *desktop*, deixando que a versão para dispositivos móveis fosse algo secundário ou que sequer era lembrado.

A fim de se obter uma versão de *website* móvel adequada, devemos tratar a programação do comportamento nos dispositivos móveis como prioridade, e não mais como algo a ser deixado em segundo plano. E, apesar desta prática ainda não ser amplamente difundida, nota-se a existência de esforços para tentar mudar esta realidade. A concepção da filosofia *Mobile First*, por exemplo, alicerça-se exatamente nisso: a ideia é que o planejamento do *website* comece exatamente pelo dispositivo mais limitado (Seja em tamanho de tela ou em recursos) e que o planejamento dos demais seja feito com base neste primeiro.

Porém, apenas iniciar o desenvolvimento pelo ambiente *mobile* — que normalmente é o ambiente mais simples — não garante bons resultados por si só. Para isso, devemos identificar o que exatamente é necessário para que tenhamos uma experiência de uso agradável nestes dispositivos.

Diversos materiais e diretrizes foram criadas tratando de aspectos de usabilidade e acessibilidade para interação com interfaces de uma forma geral mas, para que se consiga atingir um resultado mais satisfatório nos dispositivos móveis, seria interessante que fossem verificados os aspectos mais apropriados especificamente para eles.

1.1 Objetivos

O principal objetivo deste trabalho é analisar princípios de *design* de interfaces e propor um guia de recomendações específico para o *design* de interfaces *web* móveis. Como objetivos secundários, tem-se:

1. Realizar um levantamento bibliográfico sobre o *design* de experiência e o desenvolvimento *web*, contemplando assuntos como usabilidade e acessibilidade, bem como o estudo da filosofia *Mobile First*, buscando identificar boas práticas e técnicas de como empregá-las;
2. Desenvolver um protótipo com base nas recomendações levantadas no guia;
3. Avaliar se o protótipo desenvolvido com base no guia obtém um bom desempenho no que diz respeito à sua usabilidade.

1.2 Estrutura do Texto

Este trabalho está estruturado da seguinte forma:

- O Capítulo 2 traz uma visão geral do estudo do *Design* da Experiência do Usuário, elencando-se as competências pelas quais ele é composto e ilustrando a multidisciplinariedade que é característica deste estudo;
- O Capítulo 3 traz especificamente os conceitos de Acessibilidade e Usabilidade, elencando também as principais diretrizes identificadas nos estudos destas disciplinas. Também serão abordadas as formas de avaliação da Usabilidade;
- O Capítulo 4 trata sobre a filosofia *Mobile First*, que argumenta que deve-se iniciar o desenvolvimento de um *website* a partir dos dispositivos móveis;
- O Capítulo 5 apresenta o guia de recomendações para design de interfaces *web* móveis que foi desenvolvido;
- O Capítulo 6 traz considerações a respeito do processo de análise e implementação do protótipo que foi usado para a avaliação do guia;
- O Capítulo 7 apresenta a avaliação heurística do protótipo criado, a fim de se validar o guia criado;
- O Capítulo 8 traz as considerações finais a respeito do trabalho, além de discorrer sobre as contribuições do projeto e trabalhos futuros a serem realizados com base nele.

2 *Design* de Experiência do Usuário

O *Design* de Experiência do Usuário é o processo de projetar todos os aspectos que envolvem a interação de um usuário com um sistema, com o objetivo de melhorar a experiência do usuário com tal sistema.

Segundo Psomas (2007), o *Design* de Experiência do Usuário é composto de cinco competências: *Design* Visual, Arquitetura de Informação, *Design* de Interação, Engenharia da Usabilidade e Engenharia do Protótipo. A estrutura deste capítulo se baseará em uma adaptação desta classificação.

2.1 *Design* Visual

Quando aplica-se qualquer tipo de *design*, isto significa que planeja-se o desenvolvimento de algo em vez de que isso fosse deixado ao acaso. E é exatamente disso que se trata o *design* visual: a organização visual de um certo elemento de forma a se obter uma comunicação efetiva com este usando uma série de princípios como guia para se obter isso (LAUER; PENTAK, 2012).

Segundo Evans e Thomas (2013), o *design* visual possui certos princípios e elementos que, da mesma forma que as palavras em uma linguagem, funcionam como se fossem um vocabulário visual.

2.1.1 Princípios do *design*

Os princípios do *design* são “abstrações generalizáveis, destinadas a orientar os *designers* a pensar sobre aspectos diferentes em seus *designs*. (...) Tendem a ser escritos de maneira prescritiva, sugerindo aos *designers* o que utilizar e o que evitar na construção de uma interface” (PREECE; ROGERS; SHARP, 2005).

Os mais comuns princípios são:

- *Affordance*: É o relacionamento entre um objeto físico e um usuário, a definição de quais ações são possíveis. Também pode-se dizer, de uma forma mais simples, que o objeto em questão “permite” algo.

Um exemplo interessante citado por Norman (2013) é o do vidro: ele permite transparência visual, permitindo que se possa ver através do mesmo; porém, não permite transparência física, e isso pode ser considerado um *anti-affordance*.

É importante que tanto as *affordances* quanto as *anti-affordances* sejam perceptualmente óbvios; caso contrário, o usuário encontrará dificuldades na hora de interagir

com o objeto. Nos casos onde exista dificuldades em se visualizar alguma *affordance* ou *anti-affordance* em específico, pode-se recorrer ao uso de *signifiers* para ajudar nesta tarefa.

- *Signifiers*: São símbolos, sinais perceptíveis de que ações podem ser feitas e de como elas devem ser feitas.

Novamente, um bom exemplo citado por Norman (2013) é o de uma entrada composta de duas portas. Caso não haja indicação alguma, não há como saber se elas devem ser empurradas ou puxadas e o usuário terá que verificar por tentativa e erro sobre o que deverá ser feito. Uma forma de solucionar esta falha de design é inserir *signifiers* em cada porta — neste caso, uma placa em cada porta indicando a ação que a mesma suporta seria suficiente.

- Mapeamento: Exprime a relação entre os controles e os efeitos que eles terão no objeto-alvo.

Por exemplo, ao considerarmos um controle-remoto e as ações de ajuste de volume, podemos verificar que na maior parte das vezes o botão de aumento de volume fica em cima do botão que diminui o volume, exatamente porque existe um mapeamento dos botões com a direção de suas ações. Caso os botões estivessem posicionados de outra forma, teríamos um *layout* menos intuitivo.

- *Feedback*: Trata-se de um retorno de informações após a realização de uma ação, de forma a informar o *status* do sistema e de indicar possíveis próximas ações.

Quando apertamos o botão de liga/desliga em um rádio de um carro, por exemplo, pode-se verificar que o painel se acende logo após, indicando que a ação realizada surtiu algum efeito; caso isso não acontecesse, o usuário poderia se sentir desorientado, o que poderia levar o mesmo a tentar apertar o botão novamente na tentativa de ter sucesso.

- Restrições: Trata-se das formas que se tem de delimitar as ações possíveis de serem realizadas em um dado momento.

Quando consideramos o contexto de uma tela exibindo uma lista de ações, Preece, Rogers e Sharp (2005) citam o quão comum é sombrear quais estão desativadas, restringindo ao usuário o que pode ser feito naquele momento.

- Visibilidade: Ao combinarmos de forma apropriada os conceitos de *affordances*, *signifiers*, restrições, mapeamentos e *feedback*, damos ao usuário condições de saber como lidar com um produto, possibilitando que ele verifique o que o produto faz, como ele funciona e quais ações são possíveis (NORMAN, 2013). À este entendimento do produto como um todo damos o nome de visibilidade.

- Modelos conceituais: É uma explicação de como algo funciona, mas de uma forma bem simplificada.

Norman (2013) cita o exemplo dos ícones de pastas e arquivos que são apresentados na tela de um computador. Na realidade, não existem pastas ou arquivos dentro de um computador, e estes são apenas conceitualizações criadas de forma a facilitar o uso do dispositivo.

Evans e Thomas (2013) também citam alguns princípios, sendo estes relacionados aos princípios citados por Preece, Rogers e Sharp (2005):

- Unicidade: busca trazer a ideia de harmonia entre os elementos de um *design* através do controle da variedade. O objetivo é que, ao se olhar para algo, enxergue-se primeiramente o *design* como um todo antes de que se identifique as partes que o compõem. A falha em se alcançar a unicidade de um *design* praticamente significa que o mesmo falhou em ser bem-sucedido.
- Variedade: tem como objetivo realizar combinações de elementos visuais que aparentemente não tem muito em comum, com o objetivo de se trazer engajamento visual do usuário. Funciona em conjunto com o princípio de Unicidade.
- Hierarquia: ter uma hierarquia é definir uma ordem de importância dos elementos visuais.
- Predomínio: trata-se de um princípio complementar ao de Hierarquia, já que, basicamente, um elemento que é hierarquicamente superior a outro possui dominância sobre este.
- Proporção: é o tamanho relativo de um elemento com relação ao espaço que ele ocupa.
- Equilíbrio: é o arranjo visual dos elementos em um cenário. Existem 4 tipos de equilíbrio visual:
 - Simétrico: dado um eixo central, organiza-se os elementos de forma semelhante ou igual em ambos os lados, alcançando o equilíbrio através da similaridade.
 - Assimétrico: consiste em se alcançar o equilíbrio visual usando partes com elementos em quantidades ou tamanhos diferentes, ou seja, através do contraste.
 - Radial: em vez de ter como base um eixo central, como no equilíbrio simétrico, parte-se de um ponto central, fazendo com que os elementos sejam posicionados ao redor deste.

- Cristalográfico: consiste em uma distribuição uniforme de itens por todo o espaço de forma que exista uma distribuição igual de elementos, independente do lugar onde esteja o foco visual.

Além destes princípios, também existem os princípios de apoio, que afetam os relacionamentos internos de um *design*. São os seguintes:

- Escala: refere-se ao tamanho de um elemento com relação ao tamanho de outros elementos do *design*. Este conceito acaba sendo complementado pelo princípio da proporção.
- Ênfase: trata-se da utilização de um ponto focal para atrair a atenção do usuário para um certo elemento. Pode-se alcançar tal objetivo alterando-se o tamanho, cor ou outras características visuais.
- Movimento: é a coreografia visual de um conjunto de elementos quando organizados em uma ordem apropriada.
- Ritmo: é a repetição de elementos iguais ou semelhantes em um *design*, de forma a dar movimento em uma composição visual.
- Proximidade: trata-se da relação do posicionamento entre elementos dentro de um espaço.
- Repetição: define-se como um padrão regular de posicionamento de elementos semelhantes ou iguais.

2.1.2 Componentes do *design*

Além dos princípios do *design*, o design visual também possui seus componentes, que são:

- Forma: No *design* visual, uma forma trata-se de algo que possui uma largura e uma altura, formando uma figura. O formato da forma ajuda a passar a mensagem do sentimento que se deseja transmitir com um design.
- Espaço: define-se espaço como o local onde as formas serão posicionadas, bem como também o que as formas ocupam ao existirem.
- Linha: trata-se de um conjunto de pontos que possui um certo formato; ou, segundo Evans e Thomas (2013), trata-se do caminho do movimento de um ponto. Linhas também podem ser apresentadas de forma implícita, como ao se alinhar um conjunto de formas.

- Tamanho: são as dimensões físicas de um elemento.
- Cor: uma cor é uma descrição do conjunto de matizes para se representar um certo pigmento.
- Textura: descreve características de uma superfície.
- Tipografia: é um elemento visual de dupla funcionalidade, tendo como objetivo passar uma mensagem verbal, bem como também de passar uma mensagem visual, composto de letras e sinais de pontuação.

2.2 Arquitetura da Informação

A Arquitetura da Informação tem como objetivo realizar a estruturação, organização e rotulação do conteúdo a ser apresentado em um *website*, de forma a tornar mais fácil a busca de informações e a realização de tarefas.

Segundo Morville e Rosenfeld (2007), pode-se dividir os componentes da arquitetura da informação em quatro tipos de sistemas: organização, rotulação, navegação e busca.

2.2.1 Sistemas de Organização

Quando se organiza um conjunto de quaisquer coisas, tem-se como objetivo ter domínio sobre elas, seja para que se facilite o processo de explicá-las ou de entendê-las. E, dentro da área da Arquitetura da Informação, o objetivo é basicamente este: os sistemas de organização tem como meta facilitar a localização de respostas para as perguntas que as pessoas fazem através da categorização e estruturação da informação.

Antigamente, esta tarefa era concentrada nas mãos de bibliotecários, já que a informação se encontrava em sua maior parte nos livros; porém, com o crescimento da *internet* e a possibilidade de que qualquer pessoa possa publicar seu próprio conteúdo livremente, cada vez mais esta responsabilidade vem sendo passada para os desenvolvedores e usuários de sistemas e *websites*. Porém, ao se fazer isso, verifica-se a grande dificuldade de se encontrar métricas e técnicas para praticar a organização de forma efetiva.

Assim como uma mesma palavra pode ter diferentes significados dependendo do contexto em que ela é utilizada, um mesmo tópico pode ser ambíguo quanto à sua classificação, podendo ser incluído em mais de uma categoria dentro de um *website*, o que pode vir a confundir ou impedir que um usuário em busca deste conteúdo o encontre.

Também deve-se considerar que, em se tratando de *websites*, existe uma grande variedade no jeito como são estruturados. Um *website* de notícias não é estruturado da mesma forma que um *blog*, e ambos são diferentes da forma que um *website* institucional é

estruturado. Logo, acaba sendo difícil — e até errôneo — encontrar um único formato que contemple tamanha heterogeneidade.

Os Sistemas de Organização são compostos de Esquemas de Organização e de Estruturas de Organização (MORVILLE; ROSENFELD, 2007).

2.2.1.1 Esquemas de Organização

Um esquema de organização descreve a forma a qual um conjunto de informações é categorizado. Durante o dia-a-dia, lida-se naturalmente com esquemas de organização: a agenda telefônica de um celular, por exemplo, utiliza um dos tipos de esquema, o exato; já um *website* de compras baseia-se em outro tipo, o ambíguo.

Um esquema do tipo exato possui um critério de organização absoluto, que garante que exista apenas uma posição onde cada item pode estar. Desta forma, este tipo de organização faz com que seja mais rápido de se achar um certo item.

Pode-se citar, como exemplos de esquemas de organização exatos:

- Alfabético: Muito usado em agendas telefônicas e enciclopédias.
- Cronológico: Usado em calendários, livros de história e em guias televisivos.
- Geográfico: Usado em *websites* de viagem.

Já um esquema do tipo ambíguo é aquele que exprime categorias que são subjetivas e inexatas. É mais difícil de ser mantido mas, por muitas vezes, é mais útil.

Como exemplo de esquemas de organização ambíguos, pode-se citar:

- Assunto: Usado em jornais e em livros acadêmicos.
- Tarefa: Usado em *softwares* de edição de texto e planilhas eletrônicas, nos quais as funcionalidades são agrupadas de acordo com o tipo de tarefa sobre as quais elas atuam.
- Público: Usado por *websites* corporativos, que possuem um *subsite* específico para pessoa física e outro para pessoa jurídica.
- Metáfora: Usado principalmente no *design* de interface, este esquema de organização tem como objetivo relacionar o conteúdo com conceitos que são familiares para o usuário.
- Híbridos: Misturam dois ou mais esquemas de organização. Muito usado por *websites* de compras, com o objetivo de tentar oferecer mais de uma forma para o usuário encontrar o que precisa comprar.

2.2.1.2 Estruturas de Organização

A escolha de uma estrutura é o que definirá como o usuário irá navegar pela informação. Pode-se escolher uma delas, ou optar-se por se usar duas ou mais de forma complementar. Como estruturas de organização, cita-se: hierarquia, modelo de banco de dados e hipertexto.

A estrutura de organização de hierarquia, também conhecida como a abordagem de cima para baixo, organiza as informações em categorias, que por sua vez podem possuir subcategorias, e assim por diante, de forma semelhante a uma árvore genealógica. É a estrutura de organização mais usada e, por muitas vezes, a mais bem aceita. A sua utilização traz uma sensação de familiaridade, já que grande parte das coisas com as quais lidamos no dia-a-dia são organizadas desta forma.

Já a estrutura de organização de modelo de banco de dados, também conhecida como a abordagem de baixo para cima, estrutura a informação em registros, sendo que cada um deles corresponde a um conjunto de informações de uma entidade em específico. Os registros podem se relacionar com os outros através de chaves, de forma a trazer uma completude da informação sendo requisitada, mas sem perder a organização da mesma. É o formato de estruturação usado por grande parte dos sistemas de banco de dados.

Também existe a estrutura de organização de hipertexto, que trata-se de um tipo de estruturação mais novo, sendo ele composto de dois tipos de entidades: as informações a serem ligadas e as ligações entre estas informações. É uma estrutura que permite uma grande flexibilidade de utilização mas que, ao mesmo tempo, pode vir a trazer confusão por causa desta característica. Por este motivo, ele normalmente acaba sendo usado em estruturas complementares de outras estruturas maiores.

2.2.2 Sistemas de Rotulação/Nomenclatura

A rotulação é um conceito que está mais inserido nas vidas das pessoas do que se imagina. Ao vincular-se uma certa palavra a um conceito ou coisa, está-se rotulando-o, dando-se um nome a ele. E dar um nome à alguma coisa não é nada mais do que criar um atalho mental para quando precisa-se falar dela: em vez de explicar-se todas as propriedades desta coisa, simplesmente menciona-se o nome dela.

No estudo da Arquitetura da Informação, a aplicação é a mesma: quando existe um conjunto de informações e precisa-se representá-lo de alguma forma que não seja ao se escrever todas as informações de fato, o que precisa-se é uma palavra ou conjunto de palavras que represente de forma adequada o que se está tentando resumir. E um rótulo é exatamente isto.

Além dos rótulos textuais, que são os mais utilizados, também existem os rótulos icônicos. Ambos serão detalhados em seguida.

2.2.2.1 Rótulos textuais

Pode-se dividir os rótulos textuais em quatro categorias: *links* contextuais, cabeçalhos, opções de sistemas de navegação e termos de índice (MORVILLE; ROSENFELD, 2007).

Os *links* contextuais são os tipos de rótulos de mais fácil criação, já que são inseridos no meio de um bloco de texto ou em um documento, o que acaba fazendo que sua criação seja feita conforme o necessário, sem prévio planejamento. Porém, esta mesma falta de planejamento, que se mostra benéfica pois possibilita uma maior liberdade na hora de escolher o texto do rótulo, tem seus problemas.

Este tipo de rótulo depende de contexto e, se este último falha em ser bem alicerçado, o *link* perde seu significado. Se o autor do texto tinha em mente um certo público na hora de escrever o texto e, por consequência, criou os rótulos seguindo a mesma linha (por exemplo, baseando-se que o leitor teria conhecimento de uma certa área ao ler o texto), o rótulo em si pode passar pouco significado ao leitor, provocando um baixo engajamento deste.

Os rótulos também podem ser usados como uma forma de instigar curiosidade do leitor. Ao utilizar rótulos misteriosos ou que transmitem uma mensagem incompleta, o autor pode vir a tentar fazer com que o leitor sinta vontade de clicar no *link* e ver onde ele vai parar. Vale notar, porém, que tal uso de *link* contextual exige que o leitor já tenha um certo grau de confiança no autor, o que, em caso negativo, pode trazer resultados indesejados.

Já os cabeçalhos tem como objetivo sumarizar o conteúdo de um texto que vem logo após deles. Eles devem ser criados com um cuidado maior do que os *links* contextuais, já que um cabeçalho malfeito pode desorientar totalmente um leitor em busca de uma informação específica.

Este tipo de rótulo também ajuda a indicar hierarquia entre os blocos de texto: por exemplo, cabeçalhos com maior tamanho de fonte indicam categorias mais amplas e cabeçalhos com menor tamanho de fonte indicam subcategorias dentro de categorias.

Finalmente, quando acompanhado de números, os cabeçalhos podem indicar passos de um processo maior: um número menor indica um passo a ser realizado antes de um passo com um número maior, o que dá maior segurança ao usuário na hora de navegar.

Os rótulos que tem como objetivo figurar como opções de sistemas de navegação devem receber extremo cuidado ao serem criados, tanto por causa da utilização do mesmo quanto por causa da exposição que ele terá.

Um rótulo de um sistema de navegação normalmente aparecerá de forma consistente em uma página da *web*, sendo uma das formas de apresentação mais comuns a barra de

navegação, que acompanha o usuário em qualquer página que ele visitar. A falta de cuidado na hora de criar este rótulo terá um impacto grande para os usuários deste *website*.

Finalmente, os rótulos de termos de índice são um dos mais versáteis, podendo ser utilizados para descrever um *website*, uma página, um texto, um conceito, dentre outros. Também, trata-se de um formato bem contemporâneo de se descrever informações: as *hashtags*, muito usadas em *websites* como o Twitter e o Tumblr, são tipos de termos de índice.

Os termos de índice podem estar visíveis para o usuário, como em nuvens de *tags* ou em índices de *websites*; porém, eles podem estar invisíveis também, como quando são usados na tag HTML <META>. Nesta segunda forma, os termos de índice funcionam como palavras-chave, ajudando mecanismos de busca a encontrar tal página, mesmo que estas palavras não estejam incluídas no texto.

2.2.2.2 Rótulos icônicos

Nos rótulos icônicos, usamos imagens ao invés de textos na hora de representar uma informação. Porém, é necessário que se tenha muito cuidado na hora de criar um rótulo deste tipo, já que uma imagem pode ter significados diferentes quando apresentada para duas pessoas; e também, dependendo do ícone, a imagem deste pode não transmitir significado algum para o usuário.

De uma forma geral, o uso de rótulos icônicos não é muito recomendado por *webdesigners*. Assim, o uso deste tipo de rótulo acaba existindo somente quando ele é muito pontual; quando o uso do ícone é acompanhado de um texto que ajuda a reforçar o significado do ícone; ou ainda, quando o público-alvo prefere este tipo de apresentação, como é o caso das crianças.

2.2.3 Sistemas de Navegação

Sistemas de navegação são aqueles usados com o intuito de possibilitar que o usuário percorra um conjunto de informações de forma bem-sucedida, fazendo com que ele consiga saber em qual informação ele se encontra e quais são os caminhos possíveis para outras informações que ele tem.

Os sistemas de navegação mais comuns podem ser divididos em dois grupos: embutidos e suplementares. Além desses, existem alguns sistemas que não usados com tanta frequência, que serão tratados em uma seção específica (MORVILLE; ROSENFELD, 2007).

2.2.3.1 Embutidos

Cada um dos sistemas de navegação tenta resolver um certo problema, mas gerando certas dificuldades para se alcançar isso. Não é anormal que se use simultaneamente os três sistemas que serão citados a seguir.

Os sistemas de navegação globais são aqueles que são apresentados em cada página de um *website*, normalmente no topo da página. Exatamente por causa da natureza onipresente deste sistema de navegação, deve-se ter cuidado extra na hora do planejamento do mesmo, já que ele terá grande significância (positiva ou negativa) na usabilidade do *website*.

Já um sistema de navegação do tipo local trata-se de um sistema que é normalmente complementar aos sistemas de navegação global, tendo como objetivo tentar trazer opções de navegação secundárias referentes à opção de navegação global escolhida. Com certa frequência, ele encontra-se posicionado no lado esquerdo da página.

Finalmente, para casos onde precisa-se inserir relacionamentos que não se encaixam de forma apropriada nos sistemas de navegação global e local, muitas vezes podemos usar de sistemas de navegação contextuais, que normalmente localizam-se juntamente ou de forma muito próxima ao conteúdo da página. O objetivo deste tipo de sistema em específico acaba sendo de complementar os demais sistemas de navegação.

2.2.3.2 Suplementares

Os sistemas de navegação suplementares tem como objetivo principal o de trazer meios alternativos de se acessar informações e executar tarefas.

Os mapas dos *websites* são indicados para *websites* que possuem uma arquitetura hierárquica. De forma semelhante a um índice de conteúdo de um livro, os mapas dos *websites* provém uma visão mais ampla do conteúdo existente nos mesmos.

Os índices dos *websites*, ao contrário dos mapas dos *websites*, apresentam o conteúdo do *website* de forma não-hierárquica, usando-se de palavras-chave ou frases.

Os sistemas da navegação por guias podem ser utilizados quando se deseja apresentar um tutorial de como realizar uma certa ação dentro de um *website*. Também, pode ser indicado em casos onde o *website* em questão passou por uma reformulação visual, tomando a forma de uma visita guiada. Normalmente a navegação em sistemas deste tipo é linear, já que deseja-se mostrar ou ensinar algo ao usuário de forma efetiva.

Os assistentes e configuradores funcionam de forma semelhante aos guias, tentando trazer uma forma de navegação linear e de fácil compreensão, mas com a diferença que a quantidade de características customizáveis neste tipo de sistema de navegação é bem maior.

Um sistema de navegação por busca dá grande autonomia ao usuário, permitindo que ele navegue pelas informações de um *website* da forma que lhe convir. Devido à importância deste tipo de sistema de navegação, o mesmo será detalhado na seção 2.2.4.

2.2.3.3 Abordagens avançadas

É contínua a procura por novas formas de proporcionar uma navegação agradável para o usuário. Porém, já que estas formas ainda não são tão consolidadas quanto as supracitadas, recomenda-se que somente aventure-se a utilizar estas novas formas após o domínio de aplicação de sistemas de navegação mais clássicos.

Verifica-se, por exemplo, que é cada vez mais frequente a utilização de sistemas de personalização para tentar entregar ao usuário páginas que sejam adequadas ao seu padrão de compra ou gosto pessoal. Porém, existem chances de que a mesma seja inadequada por não considerar certos parâmetros que são importantes para o usuário.

Também pode-se oferecer ao usuário o poder da customização, possibilitando que ele adeque a apresentação da página às suas necessidades. Todavia, deve-se considerar o fato de que nem todos os usuários tem interesse em realizar customizações na sua página, e por tal razão esta funcionalidade pode acabar subutilizada.

Ainda seguindo a linha de pensamento de se dar poderes ao usuário, existem várias formas de se utilizar a chamada navegação social, que é criada a partir de ações realizadas pelas pessoas que visitam o *website*. Um menu de navegação de *links* mais populares, por exemplo, é populado mediante a realização de um *ranking* da contagem de visitas por página.

2.2.4 Sistemas de Busca

Mesmo que tenha existido uma boa implementação de um ou mais sistemas de navegação, a maior parte das informações existentes em um *website* não estarão presentes nos mesmos, e elas deverão ser disponibilizadas de outra forma. Uma forma muito comum de se prover acesso e filtro sobre um conjunto de informações é a de utilizar um sistema de busca.

Deve-se notar, todavia, que para se implementar um sistema de busca que vá ter sucesso em realizar a sua função, não basta apenas prover uma caixa de busca para o usuário digitar o que bem entender e torcer para que ele encontre o que deseja. Certos itens devem ser considerados na hora de se definir a anatomia de um sistema de busca, como o foco do que será buscado e como se apresentará o resultado da busca, itens estes que serão descritos com mais detalhes nas seções a seguir.

2.2.4.1 Foco da busca

Inicialmente, ao se decidir por usar um sistema de busca, pode ser tentador indexar todo o conteúdo das páginas do website, tornando todo o texto que as compõem pesquisável. Porém, podem existir certas áreas da página que não contém informações relevantes ao usuário e que devem ser deixadas de fora — como, por exemplo, os menus de navegação. Logo, é importante que haja certo cuidado na hora de definir o que vai ser buscável ou não.

Também, dependendo do tamanho do *website* e da quantidade de conteúdo que ele possui, pode ser interessante agrupar o conteúdo em zonas de busca, possibilitando que os usuários realizem uma busca em uma ou mais zonas específicas, restringindo o escopo da mesma.

2.2.4.2 Algoritmos de busca

Existem muitas formas de se pesquisar por algo, sendo que cada uma delas possui seu respectivo algoritmo com seus pontos fortes e fraquezas.

A maior parte dos algoritmos de busca utilizam-se da técnica de reconhecimento de padrões onde, dada uma ou mais palavras a serem buscadas, tenta-se encontrar o mesmo padrão dentro do conteúdo a ser verificado pela busca.

Além dos algoritmos, também existem alguns artifícios que são usados na hora de compor a busca em si, como os corretores de digitação, que se encarregam de corrigirem pequenos erros gerados na hora de digitar a palavra a ser buscada; e as ferramentas fonéticas, que expandem a busca para que ela verifique por palavras que soem parecidas com a digitada.

2.2.4.3 Estrutura de apresentação dos resultados

Depois de encontrado o que o usuário buscava, é importante também escolher a melhor forma de apresentar esta informação para o mesmo. Questões como a quantidade de resultados por página e qual ordenação de resultados será escolhida devem ser respondidas tendo em mente o público-alvo da aplicação.

2.2.4.4 A Interface de Busca

A reunião do estudo dos três itens anteriores culminará na interface de busca do sistema. Cabe à ela informar ao usuário que tipos de operadores são possíveis de serem usados, quais filtros podem ser configurados antes mesmo dele realizar qualquer busca e, feita a busca, informar ao usuário o que exatamente e onde foi buscada a informação que ele desejava.

2.3 *Design* de Interação

Segundo Cooper, Reimann e Cronin (1993, tradução nossa), podemos definir *Design* de Interação como sendo “a prática de se projetar produtos digitais, ambientes, sistemas e serviços interativos.”

O *design* de interação combina aspectos teóricos e técnicas do *design* tradicional, de usabilidade e de engenharia, aliados a métodos e práticas próprios. Com isso, dentro dele existe a preocupação com a forma, algo que é compartilhado pelas outras disciplinas da área do *design*; porém, o que é mais importante para ele é algo que as disciplinas de *design* tradicional não exploram com frequência: o *design* do comportamento. (COOPER; REIMANN; CRONIN, 1993)

O principal objetivo do *design* de interação é trazer satisfação das necessidades e desejos da pessoa que irá interagir com um produto. Logo, antes de se ter um produto que seja visualmente agradável, é mais importante que o produto possua uma usabilidade adequada para o usuário.

2.3.1 A Trajetória da Pesquisa e Desenvolvimento de Interface

Durante os anos 50, o *hardware* era o foco principal no desenvolvimento da computação. Tal foco era tão grande que a interação com a máquina era direta: os usuários, sendo em sua maior parte engenheiros e programadores, realizavam as tarefas desejadas interagindo diretamente com o *hardware*, trabalhando com sistemas numéricos binários, octais ou hexadecimais e lidando diretamente com registradores e posições de memória.

Dado este cenário onde o *hardware* possuía um papel tão central, basicamente haviam três formas de se melhorar a interface do usuário: primeiramente, podia-se criar *hardwares* mais confiáveis e eficientes; em segundo lugar, havia a possibilidade de se melhorar a apresentação ergonômica dos componentes de *hardware* (reorganizando os mesmos ou identificando-os apropriadamente, por exemplo); ou então, podia-se livrar o usuário da necessidade de saber sobre o *hardware* na hora de programar.

Foi então que foram criadas as linguagens de programação de alto nível, que acabavam por facilitar a programação, já que proporcionavam ao desenvolvedor a possibilidade de criação de tarefas sem a necessidade de se ter familiaridade com o *hardware*. Vale notar, porém, que isso não foi caracterizado como uma melhoria de interface do usuário na época, já que praticamente não haviam outros usuários senão programadores e engenheiros. Mesmo assim, uma melhoria na interface do programador era benéfica, pois acabava resultando numa melhoria da eficiência do mesmo.

Várias outras melhorias foram sendo feitas na interface do programador com o passar do tempo, mais especificamente durante os anos 60 e também na metade dos anos 70:

a criação de montadores, compiladores, depuradores e sistemas operacionais, por exemplo, acabou fazendo que a tarefa de programar fosse menos dispendiosa; já o surgimento da operação multitarefa e da memória virtual deu mais recursos na mão do desenvolvedor. Mas a melhoria que acabou sendo mais significativa para este cenário foi a criação dos terminais interativos.

Com o advento dos terminais interativos no final dos anos 70, pessoas que até então não tinham tido contato algum com um computador passaram a fazer uso diário do mesmo. Isso mudou um pouco o significado da expressão “interface do usuário”, já que agora os usuários de computadores não eram compostos apenas por programadores. Agora, apesar de ainda se dar atenção à criação de melhorias do ambiente de desenvolvimento, o foco no que tange a interface do usuário deixou de ser em ajudar os “programadores-usuários”, passando a se dar prioridade em ajudar programadores a desenvolverem melhores interfaces para usuários que não programam. Neste momento, nasce o *design* de interface.

Ao se transferir a atenção dos estudos para estes usuários que não programavam — usuários estes que passaram a ser chamados de “usuários finais” —, uma nova gama de áreas de desenvolvimento e pesquisa se abriu. Isso acabou beneficiando indiretamente todos os usuários de terminais interativos, independentemente da familiaridade dos mesmos com programação.

Além de se realizar a aplicação de fatores ergonômicos já existentes, características particulares dos computadores fizeram com que fossem trazidos à tona problemas que iam além dos processos motores e perceptuais, o que acabou por incentivar estudos nas áreas de aprendizagem motora, memória semântica, dentre outras. E assim, de certa forma, surgiu a disciplina da Interface Humano-Computador.

A partir daí, o que aconteceu foi um aprofundamento da pesquisa na interação humano-computador. Tentou-se buscar um foco cognitivo mais profundo, o que resultou no desenvolvimento de interfaces que tentavam modelar os objetivos e planos do usuário; e também, da criação de sistemas que armazenavam ações passadas do usuário de forma a coletar informações que possibilitassem a criação de uma espécie de “diálogo” com o mesmo.

Pelos meados dos anos 90, um novo desafio aparecia: o suporte de grupos nos *softwares*. A tarefa era de relativa dificuldade, porque os grupos acabavam possuindo membros de diferentes cargos, habilidades e preferências e que, logo, tinham diferentes expectativas. Para tentar amenizar tal dificuldade, diferentes abordagens foram usadas, dentre elas: a “pesquisa contextual”, na qual os trabalhadores eram entrevistados dentro de seus contextos trabalhistas; e a de “*design* participativo”, onde os *designers* formavam uma espécie de parceria com os usuários do sistema (GRUDIN, 1990).

2.3.2 Princípios do *Design* de Interação

Os princípios do *design* de interação são diretrizes que tratam problemas de comportamento, forma e conteúdo. A razão da utilização de tais diretrizes tem como fim criar produtos que proporcionem uma boa experiência de uso aos usuários e que cumpram as funções desejadas por eles.

Princípios são tipicamente baseados em valores, e Cooper, Reimann e Cronin (1993) citam que um conjunto destes foi desenvolvido por Robert Reimann, Hugh Dubberly, Kim Goodwin, David Fore e Jonathan Korman. Segundo estes valores, deve-se criar soluções de *design* que sejam:

- **Éticas:** os produtos fazem certas coisas, e os *designers* de interação deste produto devem tentar garantir que estas coisas sejam boas. De uma forma geral, um produto não deve fazer mal a alguém ou, ao menos, deve tentar minimizar tal mal, seja este mal de natureza física, psicológica ou de qualquer outro tipo. Porém, apenas não fazer mal não é suficiente para que um produto seja bem sucedido, de tal forma que um produto também deve tentar melhorar algum aspecto do processo ou ação que este engloba.
- **Apropriadas:** os produtos criados devem entender quais os objetivos do usuário com a utilização do sistema, bem como as suas limitações durante a sua utilização. Os *designers* de interação, então, devem identificar quais os pontos fortes e fracos de seus usuários, de forma a conseguir criar um produto que proporcione ainda mais desenvoltura de uso nos pontos em que o eles possuem mais facilidade e que os ajude nos pontos em que eles possuem mais dificuldades.
- **Realistas:** os produtos desenvolvidos devem ser criados de acordo com todo o conjunto de variáveis que compõem a construção de um produto, e não apenas com um grupo delas. Logo, é interessante que os *designers* de interação tenham um diálogo contínuo com as outras equipes que fazem parte do desenvolvimento, de forma a verificar se o produto que será construído é de fato o que é necessário para o momento, ou se alterações devem ser feitas, de forma a deixar a proposta de produto mais realista e bem definida.
- **Elegantes:** os produtos criados devem ter uma graciosidade tanto visual quanto funcional, sendo agradáveis para o olhar e ao mesmo tempo precisos e práticos. Tais objetivos podem ser alcançados ao se utilizar de três práticas: procurar pela solução mais simples possível, buscar pela coesão e unidade das partes que integram o produto e promover o estímulo da cognição e da emoção do usuário enquanto ele usa o produto.

2.3.3 Padrões do *Design* de Interação

Os padrões do *Design* de Interação tratam-se de formalizações do conhecimento de *design* específicos para um certo contexto ou problema que acontece com frequência. Este tipo de prática tem como principais objetivos a redução do tempo e esforço ao se iniciar novos projetos e de melhorar a qualidade dos produtos.

Pode-se dividir estes padrões em diferentes tipos:

- Posturais: definem a postura do sistema com relação ao usuário.

Quando trata-se de sistemas *desktop*, por exemplo, um sistema pode ser do tipo soberano, que monopoliza a atenção do usuário por longos períodos de tempo; transitório, quando ele é invocado apenas quando necessário, desaparecendo logo após executar sua tarefa; ou do tipo *daemon*¹, quando o sistema realiza suas ações em segundo plano, não requerendo muitas interações com o usuário.

- Estruturais: resolvem problemas relativos à organização da informação e de elementos funcionais na tela.

Este tipo de padrão, apesar de possuir pouca documentação, é muito utilizado. Pode-se citar como exemplos destes padrões a forma como a informação é mostrada em *browsers*, com uma barra de navegação em cima e todo o resto do espaço destinado para o conteúdo; e em clientes de *e-mail* que, em sua maior parte, possuem uma barra lateral para índice, uma área na parte da cima da tela para manipular os *e-mails* e uma área na parte inferior para o corpo do *e-mail* em si.

Ambos os casos denotam padrões que, apesar de não documentados, acabaram sendo identificados como boas formas de se organizar o conteúdo e são amplamente utilizados.

- Comportamentais: tratam de problemas que concernem certas interações com elementos funcionais ou informativos, elementos estes também chamados de *widgets*.

2.4 Engenharia da Usabilidade

A Engenharia da Usabilidade tem por objetivo assegurar que exista um alto grau de usabilidade em uma interface através do emprego de conceitos e métodos. Ela tem como conhecimentos de base a Ergonomia, a Semiótica, a Psicologia Cognitiva, a Sociologia e a Engenharia de Software.

¹ Um processo que fica no plano de fundo e que processa pedidos para certos serviços, requerendo pouca ou nenhuma interação do usuário. Um exemplo deste tipo de processo é o que gerencia a fila de impressão.

A Ergonomia é o estudo da interação do ser humano com outros elementos ou sistemas. A Semiótica trata-se da ciência que estuda não somente os signos e seus significados, mas também do processo da produção de significados de qualquer coisa. A Psicologia Cognitiva é o estudo do processo de aquisição do conhecimento, considerando a forma como o cérebro realiza a percepção, associação, raciocínio, aprendizado e recordação de toda informação que entra em contato com um indivíduo. A Sociologia estuda o comportamento humano em função do meio no qual ele habita, considerando também os grupos nos quais ele se insere e as relações interpessoais que ele realiza. A Engenharia de Software é uma área da computação que tem como fim estruturar a especificação, desenvolvimento e manutenção de softwares.

Para fins práticos, neste trabalho será enfocada a Ergonomia.

2.4.1 Ergonomia

Define-se por Ergonomia como o estudo da interação do ser humano com outros elementos ou sistemas, de forma a melhor entendê-la; também, é composto da aplicação de teorias, métodos e princípios oriundos destes estudos, com o objetivo de otimizar o bem estar do ser humano levando em conta suas necessidades e limitações.

De forma a facilitar o diagnóstico de problemas de usabilidade e padronizar o formato de avaliação de interface, foram criados os critérios ergonômicos. Porém, era importante que tais critérios tivessem base na análise de interface e não dos testes com o usuário, de forma a dar rapidez e reduzir custos de avaliação; que pudessem ser usados por profissionais que não fossem especialistas na área de fatores humanos; e que fossem de natureza explícita e padronizada (BASTIEN; SCAPIN, 1993).

Baseados nestes ensejos, Bastien e Scapin (1993) definiram oito critérios ergonômicos: Condução, Carga de Trabalho, Controle Explícito, Adaptabilidade, Gerenciamento de Erros, Consistência, Significado dos Códigos e Compatibilidade.

2.4.1.1 Condução

Este critério trata das formas pelas quais pode-se guiar e informar o usuário conforme ele interage com um computador. É dividido em quatro critérios menores:

- **Presteza:** Este critério compreende as formas pelas quais se tenta induzir que o usuário realize uma certa ação, bem como as formas de se indicar quais são as próximas ações possíveis e de qual o estado atual no qual o usuário se encontra. Pode-se citar como exemplo destas o uso de máscaras para campos de data e a inclusão de um título para cada página.

- Agrupamento/Distinção de Itens: Trata-se da organização visual de uma informação com relação às outras, de forma a demonstrar o relacionamento, a situação e a hierarquia das mesmas. Pode ser feito por localização, onde o posicionamento do item é que dita o grupo ao qual ele pertence — quando se cria uma lista com hierarquia, por exemplo; ou por formato, onde características gráficas do item indicam em qual grupo o item está contido — quando se agrupa campos de um formulário, por exemplo.
- *Feedback* Imediato: O critério do *Feedback* Imediato trata das respostas às ações dos usuários. Elas devem ser rápidas e devem trazer informações relativas à ação realizada. Como exemplo, ao se clicar em um botão que realizará um processo demorado, convém mostrar ao usuário mensagens relativas à sua complitude.
- Legibilidade: Este critério preocupa-se com as características do texto que compõe a informação a ser apresentada na tela no que tange a facilidade da leitura do mesmo. Pode-se citar, como característica que impacta na legibilidade, o contraste entre um texto e o plano de fundo do texto já que, caso o texto tenha uma cor muito próxima da cor de fundo, a legibilidade será prejudicada.

2.4.1.2 Carga de Trabalho

Este critério trata de elementos de interface que tem como objetivo diminuir a carga cognitiva do usuário. É dividido em dois critérios menores:

- Brevidade: Tem como objetivo limitar a quantidade de itens ou passos a serem armazenados pela mente do usuário ao realizar uma certa ação. Pode-se obter tal feito ao se trabalhar na concisão da informação — evitando que o usuário tenha que digitar os zeros à esquerda em um campo numérico, por exemplo; ou ao se trabalhar na limitação da quantidade de ações a ser realizada pelo usuário — por exemplo, ao carregar as informações de um endereço ao se prover um CEP.
- Densidade da informação: Este critério visa verificar se a quantidade de informação mostrada na tela realmente precisa ser exibida naquele momento, ou se falta informação na mesma, já que ambas as situações podem aumentar a chance de ocorrer erros por parte do usuário.

2.4.1.3 Controle Explícito

Este critério trata tanto do controle que o usuário tem sobre o processamento de suas ações pelo sistema como pelo próprio processamento do sistema após a realização de ações do usuário. É dividido em dois critérios menores:

- **Ação Explícita do Usuário:** dita que o computador somente pode realizar as ações que o usuário pediu e somente quando ele pediu. Por exemplo, para se iniciar um processamento qualquer, deve-se requerer que o usuário aperte algum botão (ou item de interação semelhante) que autorize que este processo se inicie.
- **Controle do Usuário:** define que o usuário deve estar sempre em controle do sistema. Por exemplo, se um processamento está em andamento, deve-se possibilitar que o usuário cancele o mesmo de alguma forma, caso desejar.

2.4.1.4 Adaptabilidade

O critério da Adaptabilidade refere-se à capacidade do sistema em se adaptar de acordo com as necessidades e preferências do usuário. É dividido em dois critérios menores:

- **Flexibilidade:** dita a capacidade do sistema em se adaptar às necessidades do usuário, levando em conta seus hábitos e estratégias de trabalho. Por exemplo, dentre uma certa gama de possíveis informações, pode-se dar a liberdade ao usuário para que ele mesmo defina quais informações são úteis para ele e que devem ser mostradas na tela.
- **Nível de experiência do usuário:** define as formas pelas quais é possível de que um sistema leve em conta a experiência do usuário. Por exemplo, usuários novatos podem realizar uma certa ação seguindo um passo a passo, enquanto usuários mais experientes podem realizar a mesma ação através de um atalho de teclado.

2.4.1.5 Gerenciamento de Erros

Este critério trata dos meios de se prevenir ou reduzir a chance de se ocorrer erros, bem como formas de se recuperar deles. É dividido em três critérios menores:

- **Proteção de Erros:** refere-se aos meios de se detectar e prevenir erros ou quaisquer ações com consequências negativas. Por exemplo, caso o usuário tenha clicado em um botão que descarta algum item, deve-se pedir por confirmação dele, já que o clique pode ter sido não intencional.
- **Qualidade das Mensagens de Erro:** trata da legibilidade, clareza e relevância das mensagens de erro em si. Seguindo este critério, deve-se criar mensagens de erro o mais específicas e breves possível, por exemplo.
- **Correção de Erros:** refere-se aos meios pelos quais o usuário pode corrigir seus próprios erros. Por exemplo, ao digitar uma informação inválida durante o preenchimento de um formulário, deve-se reapresentar o mesmo, requisitando que o usuário corrija apenas aquela informação.

2.4.1.6 Consistência

O critério da Consistência consiste na manutenção de certos padrões visuais quando tratamos de itens dentro de um mesmo contexto; e da diferenciação destes padrões quando trata-se de itens de contextos diferentes. Por exemplo, em uma página de notícias, o título de cada notícia deve estar sempre no mesmo lugar.

2.4.1.7 Significado dos Códigos

Este critério trata do relacionamento de uma palavra ou sigla com o seu significado. A utilização de uma palavra ou sigla a qual o usuário não saiba o significado pode acabar ocasionando erros ou enganos. Um exemplo disso é a utilização dos valores “Sim” e “Não” ao invés de “0” e “1”, trazendo muito mais clareza para o usuário.

2.4.1.8 Compatibilidade

O critério da Compatibilidade refere-se à correspondência dos padrões do usuário com os padrões do sistema ou da tarefa a qual ele irá realizar. Por exemplo, um campo de data deve ser exibido no formato apropriado à língua ou país do usuário.

2.5 Engenharia do Protótipo

A Engenharia de Protótipo trata da prototipagem do produto como um todo.

Um protótipo trata-se de uma versão de testes de um produto, sendo esta normalmente incompleta ou com funcionalidade limitada. A ideia é que, ao se utilizar prototipagem, consiga-se verificar com maior antecedência a existência de algum problema, já que a prototipagem se dá de forma iterativa e durante o processo de desenvolvimento do produto. Já o modelo de desenvolvimento *waterfall*, um dos mais tradicionais, só dá ao usuário uma interface para ser testada quando o produto já está perto de ser concluído e quando modificações são difíceis e custosas de serem feitas.

A ideia por trás da prototipagem é a redução de tempo e custos, e tal feito só pode ser alcançado com a simplificação do que queremos planejar, tentando obter uma versão simplificada do objetivo final do projeto.

De acordo com Nielsen (1993), existem três tipos de prototipagem:

- Vertical: Trata-se de um protótipo que é alcançado ao se desconsiderar grande parte das funcionalidades. Isso é feito de forma que se possa focar em poucas características, mas que se possa explorá-las de forma profunda e com detalhes. A ideia é tentar fazer que estas características sejam apresentadas da forma mais próxima do ideal possível.

- Horizontal: Trata-se de um protótipo no qual a funcionalidade é deixada de lado. No lugar desta, preocupa-se em mostrar ao usuário uma visão geral de todo o sistema. A preocupação aqui é de tentar validar o sistema de uma forma mais abrangente e ver como ele se comporta num todo.
- Cenário: Neste tipo, existe uma redução tanto no número de funcionalidades quanto no detalhamento das mesmas. O objetivo deste tipo de protótipo é verificar como o usuário reage ao protótipo caso ele siga uma sequência de ações pré-determinadas, ou seja, de acordo com um cenário em específico.

Também, quanto ao grau de fidelidade, pode-se separar os protótipos em dois grupos (WALKER; TAKAYAMA; LANDAY, 2002):

- Baixa fidelidade: São protótipos normalmente feitos no papel, sendo feitos à mão ou compostos de colagens de imagens. Sua confecção é mais rápida, o que faz com que eles sejam altamente indicado para as fases iniciais de um projeto, onde nem todas as funcionalidades e requerimentos foram definidos ainda. Porém, seu visual mais cru pode passar uma impressão de amadorismo, e o fato dele ser normalmente todo no papel traz menor interatividade à pessoa que está testando o protótipo.
- Alta fidelidade: São protótipos normalmente feitos com os mesmos tipos de ferramenta que seriam usados na confecção do site real, ou seja, são feitos em computadores, usando HTML, CSS e JavaScript. Este tipo de protótipo permite que se alcance um maior grau de semelhança com o produto final. Porém, produzir protótipos deste tipo é mais trabalhoso, e isso pode inibir que designers façam grandes alterações uma vez que a primeira versão esteja pronta.

Convém observar que, independente do grau de fidelidade utilizado no protótipo, seu papel se cumprirá. Segundo um estudo de Virzi, Sokolov e Karis (1996), protótipos de alta e baixa fidelidade foram igualmente efetivos para encontrar problemas de usabilidade. Porém, mesmo que os protótipos de baixa fidelidade sejam igualmente efetivos e menos trabalhosos de serem feitos, não se deve descartar a utilização de protótipos de alta fidelidade, já que estes proporcionam um maior grau de interatividade, algo que pode ser essencial para a avaliação do usuário em certos casos.

2.6 Considerações sobre o Capítulo

A área de Design de Experiência do Usuário é muito ampla, englobando diversas áreas do conhecimento humano. Tal característica implica que muitos princípios tenham que ser consideradas para que se crie uma experiência satisfatória para o usuário.

De forma a tentar facilitar o reconhecimento e a absorção destas disciplinas, diversos estudiosos trataram de criar divisões e classificações, identificando os componentes ou conceitos de cada área abordada, o que vem a facilitar que futuras aplicações destas áreas sejam realizadas. Porém, por muitas vezes, os conceitos e práticas destas áreas acabam se cruzando, o que pode provocar certa confusão de quem estiver estudando o Design de Experiência do Usuário.

O estudo do Design Visual, por se relacionar com a aparência, trata de conceitos que são mais óbvios e fáceis de serem compreendidos. Porém, por lidar com abstrações generalizáveis, muitas vezes é relativo o sucesso da aplicação das mesmas, já que um mesmo *design* pode vir a provocar diferentes sensações e reações em duas pessoas diferentes.

O estudo da Arquitetura da Informação possui caráter mais classificatório, identificando os componentes que são usados na composição de uma interface e, logo, será melhor utilizado por pessoas que desejam estudar os componentes que formam uma interface de forma mais detalhada.

O estudo do Design de Interação traz tanto uma preocupação com a forma mas, mais importante que isso, traz uma preocupação com o comportamento do que é criado. Seus princípios são mais sugestivos, e devem ser usados de forma a guiar o *design* como um todo.

O estudo da Engenharia da Usabilidade possui alto grau de importância no desenvolvimento de uma interface, já que a aplicação dos conceitos de seus conhecimentos de base poderá oferecer uma visão mais detalhada dos problemas que um usuário poderá se deparar ao utilizá-la. Os critérios ergonômicos identificados por Bastien e Scapin (1993) são ainda hoje objeto de estudo e reflexão, tendo grande importância na área de Design de Experiência do Usuário.

O estudo da Engenharia de Protótipo traz uma disciplina que tem um caráter muito prático: o de avaliar a efetividade das interfaces. Sua aplicação, independentemente do tipo de protótipo escolhido, traz benefícios aos criadores de produtos.

3 Usabilidade e Acessibilidade

Considerar aspectos de Usabilidade ao se desenvolver um *website* pode ser o diferencial para que o mesmo obtenha sucesso em realizar seu propósito, bem como aumentar o grau de engajamento do público alvo. Também, a fim de proporcionar uma melhor experiência de navegação a um público que pode apresentar ou não necessidades especiais, as interfaces de software devem considerar, desde o início do processo de desenvolvimento, aspectos de Acessibilidade.

As seções seguintes tratarão dos conceitos de Acessibilidade e Usabilidade, bem como as formas de avaliação desta última. Dar-se-á um foco maior às práticas e técnicas que possam ser aplicadas para dispositivos móveis, já que este é o foco deste trabalho.

3.1 Acessibilidade

A Acessibilidade concerne o design de produtos e serviços com a preocupação de que pessoas com deficiência consigam usá-los com tanta facilidade e eficiência quanto uma pessoa sem deficiência. Seu emprego é de suma importância, pois promove a igualdade de acesso e oportunidade para todos. Mais especificamente para o caso de *websites* e demais *softwares*, possibilita que as pessoas com deficiências possam usá-los de forma autônoma.

Ou seja, a Acessibilidade se preocupa em desenvolver uma solução que ajude pessoas com deficiências visuais, auditivas, físicas, cognitivas, neurológicas ou de fala. Mas, indiretamente, tal preocupação acaba ajudando também pessoas sem deficiência alguma, como os idosos, que tem suas habilidades reduzidas com o passar do tempo, ou uma pessoa que quebrou o braço e tem que lidar com essa limitação por um certo tempo.

3.1.1 WCAG 2.0

A WAI (*Web Accessibility Initiative*) é uma iniciativa da W3C (*World Wide Web Consortium*) que tem como objetivo desenvolver diretrizes e técnicas que promovam a acessibilidade na *web*. E, mais especificamente voltado para o conteúdo e o desenvolvimento de um *website* em si, a WAI mantém o WCAG (Diretrizes de Acessibilidade para Conteúdo da *Web*, ou *Web Content Accessibility Guidelines*), que atualmente encontra-se na versão 2.0.

O WCAG possui doze diretrizes, divididas em quatro princípios, que tem como objetivo tornar um *website* mais acessível. Também, baseada nestes princípios da WCAG, a W3C também mantém uma lista de técnicas que podem ser utilizadas para se alcançar tais princípios.

Segundo a W3C (2008), para que um *website* seja acessível, ele deve ser:

- Perceptível: deve-se apresentar a informação e os componentes de interface de uma forma que o usuário consiga visualizá-la e ter consciência do que se trata.

Este princípio engloba quatro diretrizes:

- Deve-se fornecer alternativas textuais para conteúdos não-textuais.
 - Deve-se fornecer alternativas de visualização para toda a mídia baseada em tempo.
 - Deve-se criar conteúdo que possa ser apresentado em mais de uma forma sem perder seu sentido ou estrutura.
 - Deve-se formatar o conteúdo de forma que ele seja facilmente discernível do restante dos elementos apresentados na tela
- Operável: deve-se fazer que os componentes de interface sejam construídos de uma forma que o usuário possa interagir com eles e realizar operações sobre eles.

Este princípio engloba quatro diretrizes:

- Deve-se possibilitar que todas as funcionalidades do *website* possam ser acessadas através de um teclado.
 - Deve-se prover tempo suficiente para que os usuários leiam e utilizem o conteúdo apresentado.
 - Deve-se criar conteúdo de forma que se tenha certeza de que o mesmo não causará convulsões em função de piscadas de tela com alta frequência em pouco tempo.
 - Deve-se dar formas de que o usuário consiga entender onde ele se encontra, navegar para onde deseja e encontrar informações que ele precisa.
- Compreensível: deve-se apresentar a informação e os componentes de interface de uma forma que o usuário consiga entender a informação, bem como a forma que deve-se operar a interface.

Este princípio engloba três diretrizes:

- Deve-se fazer que o conteúdo seja legível e compreensível.
- Deve-se criar as páginas de forma que elas se comportem de forma previsível e consistente.
- Deve-se ajudar a prevenir e corrigir os erros do usuário.

- Robusto: deve-se criar um conteúdo que seja robusto o suficiente para que ele seja interpretado de forma apropriada independentemente da interface que será utilizada para exibi-lo.

Este princípio engloba uma diretriz:

- Deve-se criar um *website* de forma a maximizar a compatibilidade com as tecnologias atuais e futuras.

3.1.2 Técnicas para a aplicação do WCAG

A W3C também lista, em um documento separado, algumas técnicas de como aplicar os princípios ditados pelo WCAG. Ao contrário do WCAG, que é um documento estável, esta lista é atualizada anualmente, de forma a tentar se atualizar com as melhores práticas e realizar mudanças em metodologias e ferramentas.

Porém, a existência de um item nesta lista não implica que o mesmo deva ser utilizado em todas as situações, cabendo ao programador analisar se o caso de uso o qual ele está tratando no momento comporta cada item em específico.

A W3C (2015) lista técnicas nos seguintes escopos: geral, HTML e XHTML, CSS, programação do lado do cliente, programação do lado do servidor, SMIL, texto puro, ARIA, Flash, Silverlight, PDF e falhas comuns. Para fins de praticidade, este trabalho apenas abordará alguns destes escopos, prumando-os para o mundo *mobile*, o escopo deste trabalho.

3.1.2.1 Técnicas Gerais

As técnicas gerais são aquelas que tem um âmbito mais global, e devem ser consideradas independente de haverem técnicas específicas para a tecnologia com a qual se está lidando.

É importante, por exemplo, que os títulos, cabeçalhos e rótulos usados em uma página passem de fato alguma mensagem, ou seja, eles devem ser descritivos.

Também é interessante oferecer formas alternativas à navegação usual para que o usuário alcance o conteúdo almejado. Pode-se prover um glossário, um mapa do *website*, um índice ou uma função de busca para tal fim. Uma trilha de navegação (*breadcrumb trail*) também poderia servir parcialmente tal fim, embora ela seja mais usada para indicar onde o usuário se encontra no *website*. Ainda, pode-se prover links para páginas relacionadas em cada página, ou ainda fazer que a página inicial tenha um link para os principais conteúdos do site. Por fim, é interessante que se adicione um link no topo de cada página do website que direcione o usuário para a página inicial (Este tipo de link normalmente é incluído no logo da empresa ou do *website*, mas isso nem sempre pode ficar claro para o usuário).

É recomendado que se ordene os elementos em uma ordem significativa, posicionando-os em uma ordem que represente sequências e relacionamentos dentro do conteúdo. E, uma vez que se tenha optado por uma certa ordem em específico, deve-se usá-la de forma consistente em cada vez que tais elementos forem apresentados. O mesmo vale para rótulos e nomes que, além de serem significativos, devem manter sua consistência quando reutilizados.

Cuidados também devem ser observados durante a criação de *links*, já que estes são a base do funcionamento da Internet. Uma prática recomendada é que o texto do *link*, em combinação com a frase na qual ele está inserido, indique seu propósito de existência. A utilização de rótulos de *link* do tipo “Clique aqui” e afins não são recomendadas, já que o comportamento que se espera naturalmente de um *link* é de que o mesmo seja clicável.

Quando for necessário requisitar do usuário a entrada de dados através de um formulário, deve-se fornecer um ambiente que proporcione segurança à ele. Para alcançar tal objetivo, é interessante prover exemplos de como preencher um campo em específico, bem como máscaras que formatem a entrada de acordo com o formato desejado. Também, deve-se indicar em rótulos adjacentes à qual informação cada campo se refere, e um botão deve fazer a função de enviar o formulário (Tal ação não pode ser feita de forma automática). Por fim, ao se enviar tal formulário, na existência de erros de preenchimento, deve-se informar os campos que contém inconsistências, juntamente de uma explicação textual de qual erro o usuário cometeu; e, em caso de sucesso, deve-se informar ao usuário tal acontecimento.

Um *website* deve ser desenvolvido para todos, porém as pessoas com deficiências de uma forma geral poderão ter certas dificuldades se certos cuidados não forem tidos.

Pessoas daltônicas, por exemplo, tem dificuldades para distinguir ou enxergar certas cores, logo é importante que a distinção de um tipo de informação de outro nunca ocorra apenas através da diferença de cor, mas também através de alguma diferença no texto. Também, é interessante que exista um contraste considerável entre a cor do texto e a cor do plano de fundo, de forma a proporcionar uma melhor leitura por pessoas com este tipo de deficiência visual.

Deficientes visuais, obviamente, não podem depender da visão para adquirir o conteúdo que desejam. Logo, é indispensável que o conteúdo sempre esteja apresentado de uma forma a qual as tecnologias assistivas usadas por eles consigam ler. Por isso, cada imagem deve ter seu atributo **alt** respectivo preenchido com uma representação textual da mesma. O mesmo deve ocorrer para qualquer outro elemento não-textual que possui algum conteúdo no *website*, os quais devem ter uma representação textual indicando sua proposta. E, de forma análoga, se um conteúdo não-textual não contiver alguma informação interessante para o leitor (Caso seja uma imagem meramente decorativa, por exemplo), é interessante que não se forneça uma alternativa não-textual.

Também existem algumas práticas voltadas para os deficientes auditivos. Deve-se incluir legendas em conteúdos audiovisuais, bem como não utilizar apenas de sons para informar alguma informação importante para a navegação (Por exemplo, tocar um som quando o formulário foi preenchido de maneira errada).

Cuidados especiais devem ser tidos ao se criar conteúdo que pisca de alguma forma, já que os mesmos podem afetar pessoas que possuem epilepsia. De uma forma geral, deve-se garantir que nada pisque mais do que três vezes em qualquer período de um segundo.

Mesmo pessoas sem deficiências podem ter dificuldades, como são os casos das pessoas que possuem algum problema de visão. Para estes casos, pode ser interessante que o *site* ofereça uma forma de mudar o tamanho do texto da página, e que o conteúdo e a funcionalidade da página não se perca ao se realizar tal mudança.

Além disso, o *website* não pode incomodar ou ficar no caminho do usuário enquanto ele está tentando buscar por alguma informação. Por isso, não é recomendada a utilização de páginas que atualizem automaticamente (salvo em raras exceções e quando explicitamente mencionado), e no lugar recomenda-se oferecer uma forma do próprio usuário atualizar o conteúdo quando ele quiser. O mesmo vale para a abertura de novas janelas e abas a partir de um *link*, já que o próprio *browser* oferece tal funcionalidade ao usuário, ficando a cargo dele saber quando ele quer fazer isso ou não. Por fim, sons e vídeos não devem ser tocados automaticamente.

Cuidados também devem ser tomados durante o processo de desenvolvimento em si. É interessante que haja uma separação entre a estrutura de apresentação e a informação em si, para que se possa customizar a apresentação conforme necessário sem comprometer o conteúdo. Também, deve-se criar uma marcação HTML apropriada, obtida ao usar-se elementos HTML semânticos e páginas *web* válidas.

A criação do conteúdo a ser colocado no *website* também tem seus cuidados. Ao se incluir o primeiro uso de uma abreviação, por exemplo, pode-se apresentar o mesmo juntamente de sua forma expandida, de forma a informar o usuário sobre a abreviação que será usada a partir daí. Novas palavras, palavras em outras línguas ou palavras que fogem do âmbito usual do usuário devem ser apresentadas juntamente de suas definições, ou ainda, com um *link* para a definição em um dicionário. Também, é interessante que se provenha ilustrações e fotos relativos ao conteúdo abordado, de forma a fazer que as idéias apresentadas sejam mais fáceis de serem entendidas.

Por fim, uma recomendação importante em se tratando de dispositivos móveis é de que se evite que se tenha que realizar qualquer deslocamento horizontal da tela durante a utilização do *site*, especialmente ao se ler algum conteúdo.

3.1.2.2 Técnicas de HTML e XHTML

Estas técnicas são aquelas que devem ser aplicadas quando se está criando a marcação do *website*. Muitas das técnicas mencionadas dentre as técnicas gerais são reiteradas ou complementadas nas técnicas de HTML e XHTML.

É indicado que se identifique apropriadamente cada item da página. Por isso, deve-se prover um título para cada página usando a **tag title**, bem como se complementar o texto de um *link* usando o atributo **title** e se utilizar elementos **label** para associar rótulos textuais com controles de formulário. Especificamente com relação a este último, pode-se utilizar o atributo **title** para identificar controles de formulário quando o elemento **label** não pode ser utilizado.

É importante também utilizar uma marcação semântica e correta: para cabeçalhos, deve-se usar os elementos **h1**, **h2**, **h3**, **h4**, **h5** e **h6**; para listas ou grupos de *links*, deve-se usar os elementos **ol**, **ul** e **dl**; e para conteúdo tabular, deve-se utilizar a marcação para tabelas.

De forma a beneficiar portadores de deficiência visual, é importante que se utilize o atributo **alt** em elementos **img**. Porém, quando se desejar que os mesmos sejam ignorados pelas Tecnologias Assistivas, pode-se utilizar os atributos **alt** e **title** com valor nulo.

Da mesma forma, para se indicar controles de formulário requeridos, é interessante que se utilize o próprio texto do *label* ou o elemento **legend**, de forma que alguém usando uma Tecnologia Assistiva consiga verificar tal requerimento.

Outras práticas que devem ser utilizadas neste contexto são: a utilização do atributo **summary** em um elemento **table**, para se dar uma idéia das informações contidas no mesmo; a utilização dos elementos **fieldset** e **legend**, para prover uma descrição para grupos de controles de formulário; e o uso das *tags* **dfn** e **abbr** para indicar a definição de uma palavra e abreviação, respectivamente.

Recomenda-se também que, além de identificar os *links* com textos que descrevam seus propósitos, que se crie uma ordem de tabulação lógica entre os *links*, controles de formulário e objetos. Tal característica não ajuda só os portadores de deficiência visual, como também as pessoas que navegam utilizando-se do teclado.

Por fim, para que se tenha uma marcação HTML válida, é necessário garantir que a abertura e fechamento de *tags* seja realizada, que os atributos **id** sejam únicos numa página da *web* e que elementos não contenham atributos duplicados.

3.1.2.3 Técnicas de CSS

Estas técnicas são aquelas que devem ser aplicadas quando se está criando a formatação da aparência do *website*.

Algumas destas técnicas se referem ao fato de que muitas das vezes onde poderia-se utilizar imagens para decorar ou embelezar algo, o CSS pode ser usado no lugar. São os casos das bordas e dos elementos estéticos do *layout*, por exemplo, que podem ser feitos em CSS puro muitas das vezes; e da inclusão de imagens decorativas, que pode ser feita via CSS, de forma a evitar a inclusão de marcação HTML desnecessária, que poderia vir a prejudicar a experiência de usuários de leitores de tela. Outro fator estético que deve ser ajustado por CSS e não através da utilização de imagens é o espaçamento, que deve ser configurado ao se utilizar dos atributos **margin** e **padding**.

Ainda na parte que tange os portadores de deficiência visual, pode-se utilizar o CSS para esconder uma parte do *link* do texto, texto este que estará visível para usuário de uma Tecnologia Assistiva, mas que estará invisível para os outros usuários.

De forma a fazer que o conteúdo do *website* se redimensione de forma proporcional quando o tamanho da fonte é ajustada, é interessante que se defina tanto o tamanho de fontes como de elementos que vão conter texto usando a unidade **em**.

3.2 Usabilidade

Segundo Health e Services (2015), a usabilidade mede a qualidade da experiência de um usuário quando interage com um produto ou sistema. Em geral, a usabilidade se refere à facilidade que os usuários têm em aprender a utilizar um produto para alcançar seus objetivos, e a quão satisfeitos ficam com o processo.

Conforme dito em Nielsen (1993), a usabilidade não é uma propriedade monodimensional no que se refere a interface do usuário, sendo ela facetada em cinco atributos:

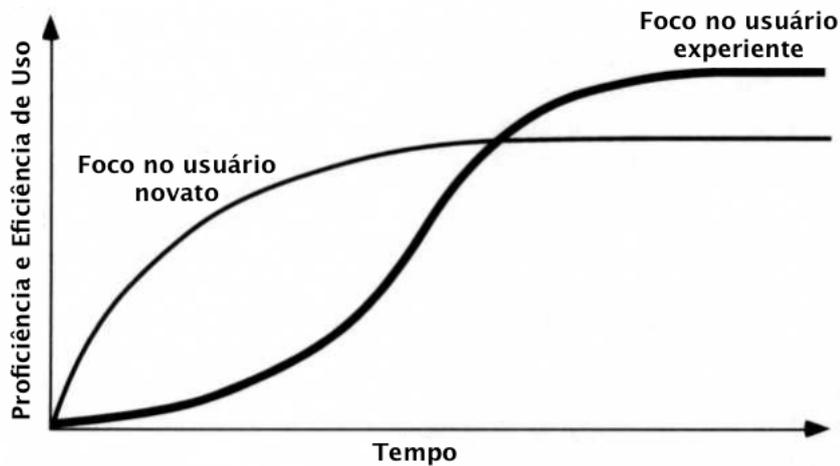
- Capacidade de aprendizagem (*Learnability*): exprime o quão fácil é de se aprender a usar o sistema. É um dos atributos mais importantes, já que grande parte dos primeiros contatos que as pessoas tem com um sistema são para se aprender a usá-lo.

Como mostra a Figura 1, um sistema que tem como foco a facilidade de aprendizado, ou seja, que se preocupa com o usuário novato, deve proporcionar ao usuário um certo grau de proficiência já com pouco tempo de uso de tal sistema.

Existem alguns casos onde um sistema com uma interface que é difícil de se aprender a usar é algo aceitável, normalmente quando estes sistemas tem como foco usuários mais experientes. Nestes casos, o sistema deve prover uma ótima produtividade para compensar tal fator.

- Eficiência de Uso: refere-se a quão rápido um usuário experiente pode conseguir executar tarefas, uma vez que tenha atingido o ápice de sua curva de aprendizado, conforme mostrado na Figura 1.

Figura 1 – Proficiência e eficiência de uso com relação ao tempo



Fonte: Nielsen (1993)

Alguns usuários podem demorar anos até atingir tal nível de performance, enquanto outros vão aprender aquilo que julgam como suficiente para si e estagnar seus aprendizados a partir daí, nunca atingido tal patamar.

- Capacidade de memorização (*Memorability*): descreve o quão fácil é de lembrar de como fazer algo no sistema após ter-se aprendido a fazer tal tarefa.

Tal propriedade atinge os usuários casuais de um sistema, sendo os usuários casuais aqueles que, ao contrário dos usuários novatos, já usaram tal sistema algumas vezes antes. Porém, apesar de já terem usado o sistema e aprendido a realizar tarefas no mesmo antes, eles devem relembrar como fazer tais tarefas.

Um sistema que é fácil de ser memorizado também é benéfico para pessoas que deixam de usar o sistema por um tempo determinado ou que saíram de férias e que, ao voltarem a usar o sistema, devem se lembrar de como o sistema funcionava.

- Erros: qualquer sistema é suscetível à erros, logo, o importante é proporcionar uma experiência de uso na qual o usuário execute a menor quantidade possível dos mesmos.

Na eventualidade de um erro acontecer, o mesmo deve ser idealmente algo o qual o próprio usuário consiga se recuperar o mais rápido possível, evitando o acontecimento de erros catastróficos, erros estes que normalmente não possibilitam que o usuário tenha algum poder sobre a situação e, inclusive, levam à perda da informação do mesmo.

- Satisfação: refere-se à quão agradável é de se usar o sistema.

O nível de satisfação ao se usar um sistema pode se basear em diferentes características do mesmo: enquanto em um cliente de *e-mail* a satisfação pode ser ditada por quão facilmente a tarefa de ler os e-mails é realizada, em um jogo de computador a satisfação pode ser ditada pela quantidade de diversão que a pessoa tem enquanto joga tal jogo.

3.2.1 Heurísticas de Usabilidade

Os princípios de usabilidade normalmente funcionam como base para a avaliação de protótipos e sistemas existentes. São eles que fornecem a base para a avaliação heurística. (PREECE; ROGERS; SHARP, 2005)

Nielsen (1995) desenvolveu os dez princípios fundamentais de usabilidade, chamados de “heurísticas” por se tratarem mais de princípios básicos, com significados amplos, do que regras de usabilidade, mais específicas.

1. Visibilidade do *status* do sistema: O sistema deve sempre informar o usuário do que está acontecendo, de que procedimentos estão sendo realizados e em que estado se encontra o sistema no momento.

Exemplo: se o usuário clicou em um *link* e será necessário se realizar um processamento, pode-se exibir um texto, indicando o que está acontecendo, e uma barra de progresso, indicando quando tempo falta para o processamento ser realizado.

2. Compatibilidade do sistema com o mundo real: O sistema deve se comunicar com o usuário em uma linguagem que o mesmo entenda, usando palavras, frases e conceitos familiares a ele. Também, a informação deve ser ordenada de forma natural e lógica, analogamente ao mundo real.

Exemplo: um usuário visitante de um *site* de notícias, de uma forma geral, não estará acostumado com termos da área de tecnologia e, logo, o seu uso não será indicado, dando lugar à explicações mais simples.

3. Controle e liberdade do usuário: O sistema deve suportar que, na eventualidade de um usuário escolher a função errada, que ele consiga sair deste estado indesejado de forma simples, para que ele então possa escolher a função correta. Em suma, a existência de funções de “Desfazer” e “Refazer” contempla tal ensejo.

Exemplos: o uso de trilhas de navegação (*breadcrumb trails*) permite que o usuário volte para níveis de navegação acima do que ele está no momento, permitindo que ele refaça sua ação, caso necessário. Botões com ações do tipo “Desfazer” e “Remover” e um *link* em cada página que direcione para a página inicial do *website* também se encaixam no caso mencionado.

4. Consistência e padrões: O sistema deve usar as mesmas palavras ao se referir às mesmas coisas em diferentes situações. Também, os usuários já utilizaram outros sistemas, logo o sistema criado deve seguir certa concordância terminológica e funcional com os demais, de forma a trazer mais familiaridade para o usuário.

Exemplo: ao utilizar-se de *links* com a mesma cor que o restante do conteúdo e sem nenhuma outra formatação que o distinga de demais textos, impossibilita-se que o usuário identifique o que é um *link*, algo que ele já está acostumado a discernir através da cor ou da formatação.

5. Prevenção de erros: O sistema deve tentar eliminar condições que sejam propícias à geração de erros.

Exemplo: em um formulário, deve-se informar ao usuário quais campos são obrigatórios, bem como o formato de preenchimento dos mesmos, de forma a evitar que, durante a validação de tal formulário, seja gerado um erro.

6. Reconhecimento em vez de memorização: O sistema deve tentar diminuir a carga de memória do usuário. Isso pode ser feito ao se disponibilizar, na tela corrente, toda a informação que o usuário precisa para conseguir realizar a ação designada àquela tela.

Exemplo: conforme o usuário está digitando um termo para realizar uma consulta em um campo de busca, este campo pode sugerir possíveis termos de busca que o usuário possa estar querendo buscar.

7. Flexibilidade e eficiência de uso: O sistema deve tentar agilizar a interação de um usuário experiente ao utilizar aceleradores, que não comprometem a experiência de um usuário novato, mas permitem que tal usuário experiente realize ações com mais rapidez.

Exemplo: o uso de nuvens de *tags* e a disponibilização de *links* com as páginas recentemente visitadas podem agilizar a navegação de um usuário mais experiente.

8. Estética e design minimalista: O sistema deve exibir somente informações que sejam relevantes e necessárias, já que cada informação ou elemento irrelevante na tela pode estar roubando o espaço de algo mais importante.

Exemplo: o uso de widgets que exibem a hora ou mostram a data atual, coisas que o usuário tem acesso através do seu sistema operacional, pode estar roubando um espaço que poderia estar sendo usado para exibir uma informação mais útil ou, ainda, poluindo o visual do *website*.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros: O sistema, na existência de erros, deverá apresentar mensagens em linguagem clara e objetiva,

indicando qual foi o problema ou situação incomum que aconteceu e sugerindo ao usuário formas de como se recuperar de tal erro.

Exemplo: durante o preenchimento de um formulário, caso o usuário esqueça de preencher um campo obrigatório, o sistema deverá exibir uma mensagem informando o usuário do que ele esqueceu de fazer.

10. Ajuda e documentação: O sistema deverá prover ajuda e documentação, focadas nas ações que o usuário precisa realizar, e escritas em um formato que seja fácil de ser pesquisado e seguido. Tais artefatos não precisam estar necessariamente separados do restante do sistema, sendo mais interessante, inclusive, que eles estejam integrados no mesmo.

Exemplo: o uso de FAQs (*Frequently Asked Questions*, ou “Questões Mais Perguntas”) pode ser interessante para solucionar as dúvidas mais recorrentes do usuário. Além disso, o uso de textos de ajuda juntamente com os campos de um formulário também pode ser uma forma de auxiliar o usuário.

3.2.2 Diretrizes de *Web Design* e Usabilidade Baseadas em Pesquisa

A Divisão de Comunicações Digitais do HHS (*U.S. Department of Health and Human Services*) mantém o Usability.gov, um compêndio de boas práticas e diretrizes voltadas à experiência do usuário. Um de seus artefatos mais importantes é intitulado Diretrizes de *Web Design* e Usabilidade Baseadas em Pesquisa (*The Research-Based Web Design & Usability Guidelines*), um guia de 209 diretrizes, devidamente quantificadas e revisadas, que podem ser aplicadas para que se criem *websites* melhores e mais usáveis.

As diretrizes do Health e Services (2006) são divididas em uma série de escopos. Para fins de praticidade, este trabalho apenas abordará alguns destes escopos, com vistas ao desenvolvimento *mobile*, foco deste trabalho.

Da mesma forma que o WCAG, a existência de uma diretriz não implica que a mesma deva ser usada em todas as situações. Fica a cargo do programador verificar se o caso de uso o qual ele está tratando no momento comporta cada diretriz em questão.

3.2.2.1 Processo de Design e Avaliação

O documento da HHS destaca a importância que de se definir os objetivos que tentarão ser alcançados com a confecção do *website*, já que os objetivos definem muitas coisas, como o público-alvo, o conteúdo e a aparência do mesmo. Dito isto, também verificou-se que o desenvolvimento de tal *website* deve ter foco principal na performance, preocupando-se primeiramente com o conteúdo e a navegação, e deixando a aparência de lado em um primeiro momento, já que a aparência tende a ter uma influência menor do que o conteúdo e a navegação na performance do usuário.

Ainda levando em consideração o que foi falado anteriormente, acima de tudo, deve-se garantir que o conteúdo mostrado no *website* tenha relevância, já que verificou-se que o conteúdo é a parte mais crítica de um *website*, sendo mais importante, inclusive, do que a navegação, o design visual e a funcionalidade. Para verificar qual conteúdo é relevante, é importante saber qual será o tipo de usuário que visitará o *website*, bem como as suas necessidades e expectativas. A fim de descobrir tais informações, pode ser interessante envolver os usuários na hora de se coletar informações.

3.2.2.2 Otimização da Experiência do Usuário

O *website* deve ser mantido de forma a trazer credibilidade ao leitor do mesmo. Para se alcançar tal objetivo, várias práticas podem ser realizadas, como a citação de fontes de artigos publicados no *website*, a atualização frequente de seu conteúdo, a disponibilização de um arquivo contendo o conteúdo mais antigo e a recomendação de fontes e materiais externos.

Também, deve-se realizar esforços para que o usuário somente realize no *website* tarefas que de fato são de seu mérito realizar, deixando a realização de cálculos e afins para o computador. Relacionado a isso, as telas devem ser criadas de forma a tentar diminuir o máximo possível a quantidade de informações que o usuário deve lembrar, seja apresentando em uma tela só todas as informações necessárias para o usuário ou fazendo que o computador lembre das informações pelo usuário.

O tempo de carregamento das páginas também é algo que requer atenção, e deve-se tentar reduzir o máximo possível o tamanho das páginas, de forma a reduzir o tempo de espera do usuário para que ele possa interagir com a página. Porém, caso seja necessário fazê-lo esperar por algum motivo (Por causa de algum processamento que está sendo realizado, por exemplo), convém informar o usuário do que está acontecendo.

A abertura de janelas, *pop-ups* ou abas sem que o usuário realize uma ação que motive tal comportamento é algo que não é visto com bons olhos e deve ser evitado a todo custo.

O *website* deve ser criado de forma simples e objetiva, de uma forma que o usuário não precise de assistência para navegar por ele. Porém, para algum caso no qual ele tenha dificuldades, deve-se prover alguma forma de ajuda para ele, ajuda esta que deve ser escrita usando terminologias que sejam familiares para o usuário.

3.2.2.3 Acessibilidade

De uma forma geral, uma página deve ser criada de uma forma que pessoas com ou sem deficiências consigam ler seu conteúdo e extrair as informações que desejam. Para que tal objetivo seja alcançado, algumas práticas devem ser tomadas, como a provisão de

alternativas textuais para elementos não-textuais (no caso de imagens, por exemplo, deve-se prover o atributo **alt** de uma imagem), a criação de formulários que permitam a navegação utilizando-se apenas de um teclado e a garantia de que eventos e comportamentos gerados através de *scripts* (*JavaScript*, normalmente) não interfiram negativamente na experiência de uso de um usuário de alguma Tecnologia Assistiva. Porém, caso a página tenha muitos recursos audiovisuais e uma experiência igual não possa ser fornecida a ambos, convém se criar uma versão paralela da página com equivalentes textuais dos recursos audiovisuais.

A aparência de um *website* deve funcionar como uma forma de deixar a informação contida nele mais atraente, mas em nenhum momento a sua existência deve ser crucial para a sua correta interpretação. Por isso, não deve-se usar apenas a cor como forma de discernir uma informação de outra, sendo mais apropriada a utilização de marcações (negrito, itálico e sublinhado, por exemplo) para tal fim. Da mesma forma, o não carregamento da folha de estilos do *website* não deve prejudicar a experiência de uso do mesmo.

3.2.2.4 *Hardware e Software*

Pode parecer interessante, de um ponto de vista inclusivo, de se tentar desenvolver *websites* para toda e qualquer plataforma existente. Mas, dado que tal feito é inviável, deve-se realizar o desenvolvimento levando em conta os navegadores mais comuns, mas sem esquecer de considerar suas peculiaridades.

Ainda, deve-se levar em conta quais são as velocidades de conexão mais usadas pelos seus usuários, bem como quais são as resoluções de tela mais comuns usadas em seus dispositivos, a fim de se desenvolver *websites* mais apropriados para seus contextos.

3.2.2.5 *Página Inicial*

Com relação à página inicial, deve-se mostrar as principais opções de acesso dentro do *website*, evitando-se o emprego de longos textos em prol da objetividade. Também, é indicado que a página inicial tenha um tamanho (vertical) diminuto, possibilitando a visualização de toda a informação contida na mesma de forma rápida.

O usuário, também, deve conseguir acessar a página inicial de qualquer página que se estiver. Por tal razão, convém se colocar um *link* para a página inicial em cada página do *website*

3.2.2.6 *Layout da Página*

Ao se pensar no *layout* de uma página, deve-se criar uma organização de conteúdo que não passe uma impressão de desorganização e poluição visual. Para se alcançar tal objetivo, deve-se otimizar a densidade informacional, tomando-se o devido cuidado de não acumular muitas informações em um pequeno espaço da tela, o que dificulta que as

mesmas sejam encontradas pelo usuário. Além disso, a utilização de espaços em branco com moderação pode ajudar a diminuir a densidade informacional de um espaço, além de dar um maior destaque ao elemento em questão.

A consistência da localização de certos elementos, como os menus de navegação e campos de busca, deve ser levada em conta. Tal fator passa mais segurança ao usuário, já que o mesmo sabe que tais elementos estarão sempre no mesmo lugar, independentemente da página que ele visitar. Da mesma forma, um alinhamento de elementos consistente, tanto verticalmente quanto horizontalmente, proporciona uma leitura mais agradável ao usuário, facilitando que ele ache a informação que procura.

Relevante também são os tamanhos verticais das páginas, já que cada tipo de página tem um tamanho mais indicado: páginas iniciais e de navegação devem ser o mais curtas possível, em prol da objetividade, enquanto páginas que requeirão longos tempos de leitura podem ter tamanho maior. Já o tamanho das linhas em um página depende mais da necessidade do leitor do *website*: se a intenção é se obter uma velocidade de leitura maior, deve-se utilizar linhas mais compridas; porém, se a intenção é que o *website* tenha mais aceitação, recomenda-se linhas mais curtas.

O uso de *layouts* fluidos também é recomendado, já que ele proporciona que usuários com telas maiores tenham experiências de uso ainda mais confortáveis.

3.2.2.7 Navegação

Quanto à navegação, seus elementos primários devem ser apresentados do lado esquerdo da tela. Caso haja mais de um tipo de navegação (Uma barra lateral e uma superior, por exemplo), é importante agrupá-las e diferenciá-las apropriadamente. E, caso deseje-se utilizar abas como uma forma de navegação, as mesmas devem estar nomeadas com termos que as descrevam efetivamente, bem como estilizadas de forma a parecer com abas de fato.

Porém, de nada adianta o usuário conseguir navegar na página se ele não tiver consciência de onde ele se encontra no momento. Logo, é importante que exista um *feedback* para ele quanto à sua localização. O uso de trilhas de navegação (*breadcrumbs*) é uma das possíveis soluções.

Como alternativas de navegação para grandes páginas, pode-se usar índices e mapas do *site*. Ou, ainda, pode-se criar páginas específicas para navegação, mas deve-se ter cuidado de mantê-las com um tamanho pequeno.

Por fim, novamente a prática de se abrir janelas é desencorajada aqui, já que ela pode vir a causar desorientação do usuário, já que ele não poderá voltar para onde estava antes a partir da nova janela que foi aberta.

3.2.2.8 Rolagem da Tela e Paginação

A rolagem horizontal deve ser eliminada dos *websites*, já que ela traz muito incômodo para usuários que estão tentando ler um texto, por exemplo. Já a rolagem vertical é de uso comum, mas seu uso deve ser observado.

Quando deseja-se ler um texto longo, recomenda-se o uso de páginas com uma longa rolagem vertical. Inclusive, uma marcação textual bem feita pode permitir que os usuários consigam identificar os itens que procuram enquanto estão rolando a tela rapidamente. Porém, caso o tipo de conteúdo seja outro, pode ser mais interessante usar paginação, já que permite que o usuário pule rapidamente para o conteúdo que deseja sem ficar rolando a tela.

3.2.2.9 Cabeçalhos, Títulos e Rótulos

Os textos dos cabeçalhos, títulos e rótulos devem ser claros e descritivos. No caso específico dos cabeçalhos, deve-se tomar cuidado para que eles sejam únicos, a fim de evitar confusão, e que seu uso seja realizado na ordem apropriada: **h1** antes de **h2**, **h2** antes de **h3**, e assim por diante.

3.2.2.10 *Links*

Na hora de se criar *links*, é importante que seus textos tenham algum significado e, caso apropriado e possível, que seus textos sejam os mesmos de suas páginas de destino. Também, na medida do possível recomenda-se evitar usar imagens como *links*, já que as mesmas podem ter mais de uma interpretação, além de serem inacessíveis por usuários usando leitores de tela caso não tenha-se definido o atributo **alt**.

Ao realizar a estilização CSS, é importante se estilize os *links* de uma forma que eles realmente pareçam *links*, de forma a identificá-los facilmente ao olhar uma página. Deve-se cuidar, também, que o estilo referente aos *links* já visitados seja definido, de forma que esta característica do *link* fique clara para o usuário.

De forma a trazer mais informação ao usuário, pode ser interessante de se criar *links* que apontem para conteúdos relacionados ou que dêem suporte ao abordado na página. Mas, na existência de *links* externos, estes devem ser distinguidos dos demais de alguma forma.

3.2.2.11 Aparência do Texto

Uma boa formatação de texto é essencial para que se exista uma boa experiência de leitura em um *website*. Recomenda-se o uso de textos na cor preta (Ou similares) e de cores de contraste alto como planos de fundo. Quanto ao seu tamanho e fonte, recomenda-se o uso de fontes familiares e de tamanho mínimo de 12 pontos.

Ainda, é importante que exista uma consistência visual na formatação do texto, usando-se sempre as mesmas cores e fontes para os mesmos casos. Caso deseje-se enfatizar ou destacar algo, alterações no estilo e tamanho da fonte poderão cumprir tal papel, assim como o uso do negrito e do itálico.

Também, recomenda-se o uso misto de maiúsculas e minúsculas em longos textos, reservando o uso exclusivo de maiúsculas apenas para casos onde deseja-se dar um destaque pontual.

3.2.2.12 Listas

Ao criar listas, caso já se saiba quais itens da lista o usuário será mais propenso a selecionar, pode-se posicioná-los no topo da lista, de forma a deixar a navegação do usuário mais eficiente. Pode-se fazer o mesmo caso já se saiba de antemão quais são os itens mais relevantes da lista.

É interessante, também, que apresente-se o que cada lista representa, precedendo-a com um título ou breve descrição.

Quanto à formatação, recomenda-se usar o estilo apropriado para cada tipo: listas numeradas quando existe uma ordem ou listas não-numeradas, caso contrário. Além disso, o uso de espaços em branco e de cores de fundo diferentes pode ser interessante para se facilitar a leitura dinâmica destas listas. Por fim, recomenda-se capitalizar a primeira letra da primeira palavra de cada item em uma lista.

3.2.2.13 *Widgets*

Ao se usar *widgets*, é importante que os campos sejam identificados de forma consistente e clara, e seus rótulos devem ser posicionados próximos de seus respectivos campos. Além disso, ao se usar os *widgets* em formulários, é importante discernir quais são os campos requeridos e quais são opcionais, preferencialmente de uma forma que seja inteligível para usuários de leitores de tela.

De forma a minimizar a quantidade de dados a serem informados pelo usuário, alguns campos já podem vir pré-populados com informações padrão. Ou ainda, caso o dado de algum campo possa ser preenchido mediante o preenchimento de um segundo campo, isto deve ser realizado (Por exemplo, o preenchimento do campo de CEP implica no preenchimento dos campos relativos ao endereço). Também, caso o preenchimento de um campo tenha relação com uma certa unidade de medida, a mesma deve ser explicitada na sua descrição.

No caso de se haver várias opções de escolha mutuamente exclusivas em um campo, recomenda-se o uso de *radio buttons*. Caso o ensejo seja de se oferecer uma forma de se realizar seleção múltipla, deve-se usar *checkboxes*. Ainda, caso o dado a ser preenchido

seja muito comprido, pode-se dividir o campo em campos menores (Por exemplo, o preenchimento de um código de barras).

3.2.2.14 Gráficos, Imagens e Multimídia

O uso de gráficos, imagens e multimídia em um *website* deve ser feito com parcimônia, somente sendo realizado quando o mesmo vier a trazer algum benefício para o conteúdo abordado.

Quando optar-se por se usar uma imagem como plano de fundo, deve observar-se que a mesma seja simples, ainda mais se a mesma for ser usada atrás de um texto. E, quando optar-se por se usar uma imagem como um *link*, é interessante que a mesma venha acompanhada de algum rótulo ou título.

Ainda, preferencialmente recomenda-se usar miniaturas de imagens que, quando clicadas, levam para a imagem em tamanho original. Tal prática, além de permitir uma pré-análise da imagem antes do usuário abri-la de fato, ainda ajuda a diminuir o gasto de banda do mesmo.

3.2.2.15 Escrevendo Conteúdo para a *Internet*

Alguns cuidados devem ser tomadas ao se escrever para a Internet. Deve-se, na medida do possível, evitar a utilização de jargões, optando-se, daí, por palavras mais familiares e comuns, de forma a promover um maior engajamento do público. Além disso, a definição de acrônimos e abreviações é essencial para que o usuário entenda do que se está falando.

De forma a otimizar a compreensão da leitura, é interessante que se limite o número de palavras nas sentenças (Para um máximo de doze palavras), bem como o número de sentenças nos parágrafos (Para um máximo de seis sentenças).

3.2.2.16 Busca

Ao se prover um sistema de busca, é interessante que o campo de busca esteja presente em todas as páginas, sempre no mesmo lugar. A busca implementada deve, preferencialmente, buscar no *website* todo, e não deve fazer discernimento entre maiúsculas e minúsculas.

O sistema de busca provido deve permitir que se busque por um número pequenos de palavras. Também, pode ser conveniente para o usuário que lhe seja mostrado exemplos de como realizar uma busca, bem como formas de melhorá-las.

3.3 Avaliação de Usabilidade

A avaliação trata-se da verificação feita em prol de se garantir que o que foi desenvolvido comporta-se de acordo com o esperado e indicado nos requisitos do usuário.

Esta verificação não é algo que deve ocorrer em apenas um momento do processo de desenvolvimento, mas sim algo que deve aplicado durante todo o ciclo de vida, de forma que a avaliação consiga prover informações que impliquem em modificações no design. Desta forma, impede-se que problemas maiores ocorram ao se promover modificações no design enquanto ele ainda está prematuro.

Dix et al. (2004) divide os vários tipos de avaliação em dois grupos: as avaliações feitas através de análise de especialistas e as feitas através da análise de usuários.

3.3.1 Avaliação através da análise de especialistas

Estas formas de avaliação dependem da existência de um *designer* ou de um especialista em fatores humanos. Elas podem ser usadas em qualquer estágio do processo de desenvolvimento e não requerem o envolvimento do usuário.

As principais formas de avaliação desse grupo são:

- Passo a passo cognitivo: esta forma de avaliação consiste em, dada uma sequência de ações de uma determinada tarefa, verificar se, uma vez estando em uma ação, existe algum problema de usabilidade que poderia impedir que o usuário identificasse a próxima ação a ser realizada.

O passo a passo cognitivo tem o propósito de verificar o quão simples o sistema é de ser utilizado, verificando com quanta facilidade se aprende a utilizá-lo apenas se baseando na exploração do mesmo.

- Avaliação heurística: é uma forma de avaliação que foi desenvolvida por Jakob Nielsen e Rolf Molich e é baseada em um conjunto de heurísticas, podendo ser estas as próprias heurísticas de Nielsen, ou alguma outra que se julgar necessária. Pode ser aplicada tanto em uma especificação de *design* como em um protótipo.

Esta avaliação vai iterando sobre as heurísticas escolhidas, tendo como objetivo identificar violações de cada uma delas e definindo o grau de severidade de cada uma delas, o que pode indicar problemas de usabilidade.

3.3.2 Avaliação através da análise de usuários

Estas formas de avaliação contam com a participação do usuário. Elas tendem a ocorrer em estados mais avançados do processo de desenvolvimento, quando se tem ao menos algum protótipo com o qual o usuário pode interagir.

As principais formas de avaliação desse grupo são:

- Métodos empíricos (Avaliação através de experimentos): trata-se de uma forma de avaliação que pode ser usada para muitos fins. Consiste no uso de um experimento controlado, realizado com o fim de se obter evidências empíricas que possam corroborar em prol de uma hipótese.
- Técnicas observacionais: esta forma de avaliação consiste na observação do usuário enquanto ele realiza uma certa tarefa. A partir disso, realiza-se notas do que o usuário está fazendo, de forma a verificar se o que ele está fazendo está de acordo com o que foi planejado quando a interface foi criada. Para se ter uma informação ainda mais apurada com relação à forma de pensamento do usuário, frequentemente pede-se que o usuário fale em voz alta o que está se passando na cabeça dele, explicando o porquê de cada ação que ele está realizando. Ainda, após a realização da tarefa, pode-se realizar uma revisão do que foi registrado pela pessoa que está executando o experimento juntamente com o usuário, com o fim de se esclarecer ainda mais alguns detalhes do processo.
- Técnicas questionativas: aqui, questiona-se diretamente o usuário sobre a interface, tendo como objetivo conseguir a opinião do usuário da forma mais direta possível. É de fácil e barata aplicação. Pode ser administrado em forma de entrevista, que é mais flexível por permitir que o entrevistador possa investigar certos assuntos mais a fundo caso achar necessário; e o questionário, que é mais direto e possui questões fechadas.

3.4 Considerações sobre o Capítulo

A Usabilidade indica a facilidade de um usuário para utilizar um certo produto. A consideração de aspectos de Usabilidade ao se desenvolver tal produto pode ser um dos fatores diferenciais entre o sucesso e o fracasso do mesmo.

A Acessibilidade concerne o desenvolvimento de produtos preocupando-se com pessoas com deficiências. Sua existência é essencial para que haja igualdade de acesso e oportunidade para todos.

Tais disciplinas, porém, precisam estar em constante atualização já que, à medida que novos dispositivos e tecnologias são criadas, estes devem passar a serem considerados nos estudos.

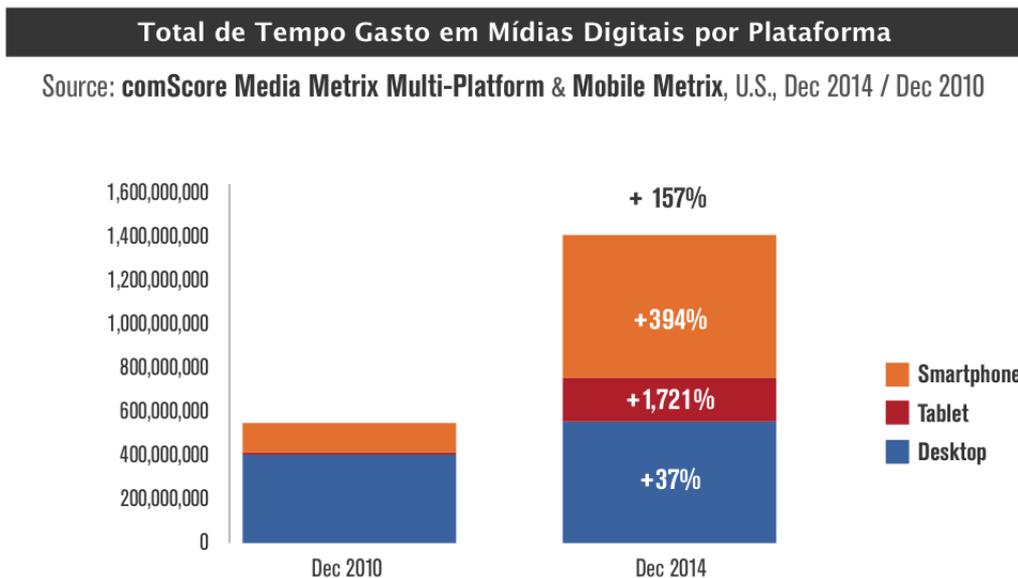
A avaliação da Usabilidade tem como objetivo verificar se as ações que foram executadas em prol de uma melhor Usabilidade de fato foram efetivas em seu objetivo.

Este trabalho proporá um guia de boas práticas para o desenvolvimento *web* móvel e, para verificar se o mesmo é bem sucedido em realizar seu objetivo, será utilizada a avaliação heurística como forma de avaliação da Usabilidade.

4 *Mobile First*

De acordo com o comScore (2015), as plataformas móveis deixaram de ser apenas mais um meio de consumir conteúdo para passarem a ser a principal forma.

Figura 2 – Total de tempo de uso de mídia digital gasto por plataforma



Fonte: comScore (2015)

Como mostra a Figura 2, já se utiliza *smartphones* e *tablets* com maior frequência do que os *desktops*. E tal êxodo de utilização é justificado principalmente por três fatores: o lançamento do iPhone, o barateamento do custo dos dispositivos móveis e o aumento da cobertura das redes móveis.

Até antes do advento do iPhone, acessar a internet através de um dispositivo móvel era uma tarefa que exigia paciência do usuário: seus navegadores demoravam para carregar o conteúdo e os teclados T9 que vinham nestes dispositivos dificultavam a digitação de URLs e queries de busca, bem como a navegação em si. Com isso, a utilização de *internet* móvel era modesta, e em grande parte das vezes não se justificava criar sites voltados para estes dispositivos.

Foi então que, em 29 de Junho de 2007, foi lançado o iPhone, e com ele vieram mudanças tanto na forma de mostrar as informações para o usuário — a primeira versão do Safari, o navegador do iPhone, já se mostrava muito mais robusta que os outros navegadores de dispositivos móveis existentes até então — como na forma de navegar e de

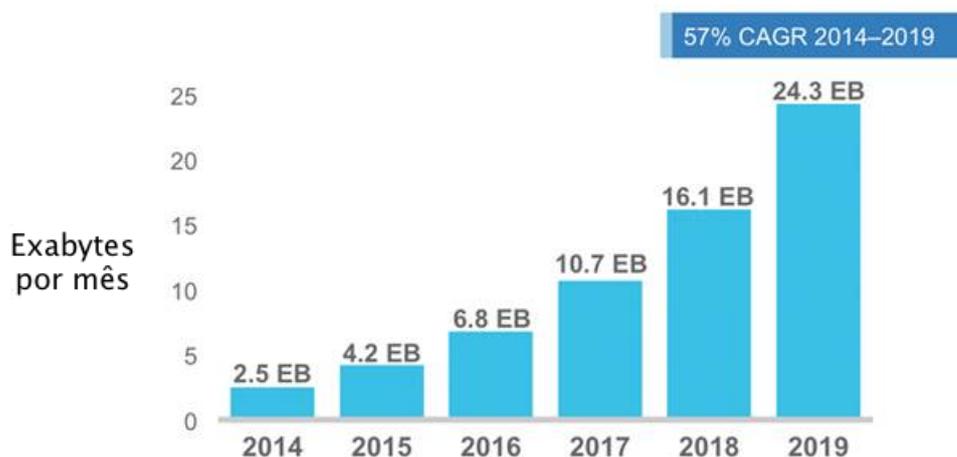
prover entrada de dados — o uso do toque por toda a extensão da tela e a concepção de um teclado virtual QWERTY¹ completo facilitavam a interação do usuário.

Em 2008, surge o primeiro celular com a plataforma Android, utilizando-se de características semelhantes ao iPhone: um navegador de utilização agradável e uma grande ênfase na interação através de toque (apesar deste primeiro celular ainda vir com teclado físico).

A adoção da plataforma Android foi lenta inicialmente, mas seus dispositivos atualmente compõem a maior fração de mercado de dispositivos móveis. E, grande parte graças à sua popularização, com isso pessoas que não teriam dinheiro para adquirir um *notebook* e um plano de internet conseguiram comprar um *smartphone* e adquirir um pacote de operadora móvel com *internet*.

Com o aumento do número de *smartphones*, também cresceu o número de usuários de *internet* móvel. Para suprir esta demanda, as operadoras de celular tiveram tanto que aumentar a cobertura de suas redes como conseguir proporcionar redes com maior velocidade. E, aparentemente, esta tendência não mudará tão cedo, já que se prospecta um tráfego de dados em redes móveis dez vezes maior ao de hoje em 2019, conforme Figura 3. (CISCO, 2015)

Figura 3 – Previsão de tráfego de dados oriundos de *internet* móvel por mês em 2019



Fonte: Cisco (2015)

Dadas estas informações, percebe-se o grande mercado que os dispositivos móveis representam hoje, mercado este que está em constante crescimento e que não parece dar sinais de enfraquecimento tão breve. Deixar de considerar estes dispositivos na hora de desenvolver *websites* é arriscar perder muitos possíveis consumidores, clientes e leitores. E,

¹ Tipo de teclado no qual as seis primeiras teclas da primeira fileira de teclas alfabéticas são as das letras Q, W, E, R, T e Y, nesta ordem.

considerada a grande expressividade do número de usuários destes dispositivos, a ideia de se começar desenvolvendo um website a partir de sua versão *mobile* já não soa mais tão absurda.

A filosofia do *Mobile First* alicerça-se exatamente nisso: a ideia é que o planejamento do *website* comece pelo dispositivo mais limitado (seja em tamanho de tela ou em recursos) e que o planejamento dos demais seja feito com base neste primeiro. As vantagens de tal prática são o maior foco no conteúdo que realmente importa para o usuário e o incentivo em se inovar na forma de se estruturar uma página, grande parte graças às limitações implicadas pelo desenvolvimento de um *website* para uma tela menor.

De acordo com Wroblewski (2011), existem certas limitações a serem consideradas por quem vai desenvolver utilizando a filosofia *Mobile First*. Também, ao se utilizar tal filosofia, deve-se levar em conta alguns fatores relativos à organização, entrada de dados, ações do usuário e *layout*.

4.1 Limitações

Para desenvolvedores que estão acostumados a desenvolver para resoluções de 1024×768 ou maiores, ter que se acostumar a passar a considerar a pequena resolução de 320×480 (e suas variações) como o ponto de início para o desenvolvimento de um *website* pode ser bem difícil.

Porém, com a diminuição do tamanho da área útil, obriga-se a manter o foco em se incluir na tela apenas o conteúdo que é de suma relevância para o usuário. Para fazer isso, exige-se que se faça um trabalho para identificar quais as funcionalidades e recursos que o usuário mais precisa ao visitar o *website*. Feito tal trabalho, além de se usar o resultado para construir a versão *mobile* do *website*, pode-se inclusive considerar modificações no site para *desktop*, já que o foco no que realmente interessa ao usuário é algo interessante de ser alcançado independente da plataforma.

Também, deve-se levar em conta que as redes de *internet* móvel não tem a mesma velocidade e estabilidade das redes cabeadas. Por tal razão, Wroblewski (2011, tradução nossa) diz que “qualquer coisa que puder ser feita para melhorar a performance em plataformas móveis deve ser feita”. Isso inclui:

- Juntar e minificar arquivos de CSS e JavaScript
- Evitar a utilização de bibliotecas de JavaScript e de sistemas de *grid* de CSS, caso eles sejam usados pontualmente — nestes casos, é melhor desenvolver a funcionalidade por conta própria, de forma a ter menos código para ser requerido no carregamento da página

- Usar *sprites* de imagens para combinar várias imagens em uma só, de forma a diminuir o número de requisições para o servidor
- Usar as propriedades do CSS3, como os gradientes e arredondamentos de cantos, para substituir a necessidade de se carregar imagens para desempenhar semelhantes funções

Novamente, a diminuição no carregamento de recursos para a plataforma *mobile* também é benéfica para os dispositivos *desktop*, já que tal diminuição implica numa redução do tempo de carregamento dos *websites* também para estes dispositivos.

Outra informação mostrada por Wroblewski (2011) é de que existem grandes chances de uma pessoa que está usando um dispositivo móvel estar dividindo a atenção dele com alguma outra tarefa, de tal forma que ele metaforicamente imagina um usuário deste tipo de dispositivo como “um olho e um polegar”, já que este usuário provavelmente estará com foco parcial no celular e segurando-o com apenas uma mão, usando apenas o polegar para controlá-lo.

De certa forma relacionado com a informação supracitada, um usuário de um dispositivo *mobile* tende a usá-lo em curtos intervalos de tempo, tendo como objetivo realizar uma função em específico ou resgatar uma informação em particular. Tarefas mais dispendiosas ainda acabam sendo realizadas em dispositivos *desktop* pela maioria das pessoas.

Tais informações indicam que o programador deve se focar em tentar proporcionar a experiência de uso mais simples, direta e agradável possível, de forma que o usuário consiga a informação que precisa sem precisar se focar muito na tarefa e sem levar muito tempo.

4.2 Organização

A organização do conteúdo e das ações possíveis é de suma importância para que um *site* esteja adequado para o mundo *mobile*.

Segundo Wroblewski (2011), existem alguns tipos de interação possíveis de serem realizados em um dispositivo móvel:

- Procurar por algo
- Explorar ou fazer algo como passatempo
- Verificar o status de algo
- Editar ou criar algo

Manter em mente a existência destes distintos tipos de interação pode ajudar na estruturação adequada do *website*, proporcionando uma melhor experiência do usuário.

É importante também que se mantenha o foco em se apresentar conteúdo acima de tudo, deixando a navegação em segundo plano, de forma a tentar trazer a informação que o usuário precisa o mais rápido possível.

Porém, isso não significa que a navegação deva ser desprezada. Mesmo em segundo plano, a navegação deve ser criada de forma inteligente e sucinta, de forma a não tomar muito espaço da já reduzida tela dos dispositivos móveis.

Pode ser interessante, por exemplo, utilizar menus de navegação na parte inferior da página, já que sua posição proporciona um acesso mais facilitado para pessoas que estão utilizando o celular com uma mão só; porém, menus com ações contextuais são mais interessantes de serem posicionados no topo da página, já que sua posição proporciona acesso imediato.

Mesmo assim, acima de tudo deve-se pensar em se manter a simplicidade. A adição de menus deve ser limitada ao mínimo possível e ao que for contextualmente adequado.

4.3 Ações do Usuário

Sendo o toque a principal forma de interação dos dispositivos móveis atuais, deve-se verificar se o *website* sendo criado está preparado para receber interações deste tipo.

Por exemplo, pode-se pensar que, ao se mover um *layout* feito com foco no *desktop* para um dispositivo móvel, o que deve-se fazer basicamente é diminuir tudo, de forma que os itens caibam na tela. Porém, já que a interação com a tela *touchscreen* é normalmente o dedo (em raros casos, uma caneta óptica), que é algo muito maior do que o cursor do *mouse*, deve-se fazer exatamente o contrário: quanto maior e mais bem espaçados os controles com os quais o usuário vai interagir, melhor.

Também, lembrando um fator apontado anteriormente, deve-se considerar que, conforme Figura 4, grande parte das pessoas usa o celular com uma mão só e, logo, as ações do usuário devem ser posicionadas de forma a se beneficiar disso.

Desta forma, pode-se fazer que ações primárias sejam posicionadas no meio ou na parte de baixo da tela, enquanto ações mais destrutivas sejam posicionadas no topo da tela.

Outro ponto a ser considerado é a quantidade de gestos possíveis de serem utilizados através do toque, como o toque em forma de pinça, muito usado para se realizar o zoom em uma foto. Tais particularidades do mundo *mobile* podem proporcionar que, ao se criar sites para estas plataformas, abandone-se um pouco o uso de GUIs, as *Graphical User Interfaces* em prol das NUIs, as *Natural User Interfaces*. Ainda falando do exemplo do zoom da foto,

Figura 4 – Facilidade da realização do toque em diferentes áreas da tela quando se está segurando o telefone apenas com a mão direita



Fonte: Wroblewski (2011)

já é natural para muitas pessoas realizar o *zoom* de uma foto usando o toque em forma de pinça, dispensando o uso de elementos GUI, como botões e menus, para realizar esta tarefa.

Porém, assim como os dispositivos móveis proporcionam algumas funcionalidades a mais, elas acabam não tendo algumas outras que existem em dispositivos *desktop*. Um exemplo disso são os estados de *hover* (quando se coloca o *mouse* sobre alguma coisa) e *focus* (que indica o item em foco no momento), que são inexistentes nos celulares e *tablets*. Logo, todas as ações que estavam mapeadas para estes estados devem ser remapeadas de forma que a experiência de uso em dispositivos móveis não seja prejudicada.

4.4 Entrada de Dados

A *internet* não é feita apenas do consumo de conteúdo. Os usuários também gostam de contribuir com suas opiniões, experiências e conhecimentos. E, aliado ao fato dos celulares e *tablets* estarem com as pessoas onde quer que elas vão, estes dispositivos podem ser ideais para este tipo de tarefa se houver uma adequação na estrutura e nas formas as quais possibilita-se que o usuário introduza informação.

Como o espaço horizontal dos dispositivos móveis é limitado, o uso de rótulos ao

lado de seus respectivos campos não é indicado. No lugar disso, indica-se usar rótulos em cima ou dentro dos campos.

Na hora de se pedir que o usuário informe alguma sequência de dados, pode-se alimentar certos campos com valores padrão, de forma a tentar diminuir o esforço do usuário ao preencher o formulário.

Quando for oportuno, pode-se aplicar máscaras em campos que tem um formato de entrada pré-definido, guiando o usuário e evitando maiores erros de preenchimento.

Cabe notar também que, graças às funcionalidades exclusivas dos dispositivos móveis, pode-se requerer a entrada de informações do usuário de formas que não seriam possíveis em um dispositivo *desktop*, como através de uma foto, da leitura de um *QR Code* ou de uma *tag* NFC. Tais formas de entrada podem diminuir um pouco a morosidade e o incômodo o qual a entrada de dados em um dispositivo móvel pode trazer.

4.5 *Layout*

No mundo *mobile*, a única coisa que se pode ter certeza é a da mudança. A cada mês, novos modelos de celular são lançados, trazendo novas funcionalidades e usando resoluções diferentes das que existiam até então, criando novos desafios ao desenvolvedor. Algumas práticas, porém, podem ajudar a diminuir certas dificuldades.

O uso da *meta tag* “*viewport*”, por exemplo, indica ao navegador do dispositivo como controlar as dimensões da página a ser renderizada, fazendo que a mesma seja apresentada em uma proporção apropriada.

Aliado a isso, é interessante que se utilize de *design* responsivo. Unindo *layouts* fluidos, *media queries* de CSS3 e JavaScript, pode-se adaptar as páginas de forma mais significativa quando estas são expostas à situações onde apenas a configuração do *viewport* não é suficiente (quando se altera a orientação da tela, por exemplo).

Porém, mesmo considerando o *design* responsivo, algumas vezes pode ser apropriado desenvolver uma experiência de uso específica para certos dispositivos, de forma a aproveitar certas características particulares dos mesmos. Por exemplo, um aplicativo para uma televisão terá tanto dispositivos de entrada como tamanho de tela muito diferentes dos de um celular e, por isso, talvez seja indicado desenvolver um *layout* específico para a televisão, para que se possa prover uma experiência de uso adequada.

4.6 Considerações sobre o Capítulo

Diversas estatísticas corroboram a favor do fato de que a venda e a utilização de dispositivos móveis encontram-se em uma crescente, e tal fato indica que os projetos de

desenvolvimento *web* também devem seguir nesta direção, com foco nestes dispositivos.

A filosofia *Mobile First* promove a idéia de que o desenvolvimento *web* deva começar pelos dispositivos móveis, e a razão de tal prática baseia-se tanto nos dados estatísticos como em critérios mais práticos, já que começar a desenvolver focando-se em dispositivos mais limitados tende a gerar produtos finais mais funcionais, esteticamente limpos e rápidos.

5 Guia de Recomendações para Design de Interfaces *Web* Móveis

Como apresentou-se no Capítulo 3, existem diversas recomendações para se desenvolver uma interface que tenha uma melhor Usabilidade. Porém, como foi discutido no Capítulo 4, os dispositivos móveis são muito diferentes dos demais dispositivos.

Os dispositivos móveis, ao contrário dos *desktop*, são, na maior parte do tempo, usados enquanto o usuário o segura, muitas vezes com apenas uma das mãos. Por isso, a interação com o *website* deve ser simplificada, de forma que o esforço para se realizar uma ação seja, na maior parte dos casos, mínimo.

Também deve-se levar em conta o tamanho da tela destes dispositivos. Com dimensões que correspondem à pequenas frações do tamanho das telas de dispositivos *desktop*, a área útil para se exibir informações e realizar interações é bem diminuta, o que necessariamente que cada elemento que será colocado na tela seja avaliado quanto à sua utilidade.

Por outro lado, as suas particularidades também podem trazer vantagens. Algumas funcionalidades são exclusivas dos dispositivos móveis, como os acelerômetros, GPSs e até a capacidade de se realizar ligações, e tais particularidades podem ser usadas a fim de se prover uma melhor experiência de uso para o usuário.

Dada a quantidade de especificidades deste tipo de dispositivo, propõe-se o desenvolvimento de um guia de recomendações específico para o design de interfaces *web* móveis. Este guia tem suas recomendações baseadas no conteúdo estudado em W3C (2008) e Health e Services (2006), além de propor recomendações baseadas na experiência do autor. A divisão das diretrizes que compõem este guia foram feitas com base nas heurísticas de Nielsen.

1. Visibilidade do *status* do sistema

- Cada página deve possuir um título, de forma a informar ao usuário onde ele se encontra, de forma absoluta. O título deve ser descritivo e usar palavras que se encontram no domínio do usuário;
- A navegação deve ser visível em todas as telas e aparecer sempre nos mesmos locais: se existirem poucos itens de navegação, pode-se posicioná-la em cima, de forma a não perder espaço horizontal; caso contrário, a mesma pode ficar escondida, sendo exibida após se apertar o botão do menu (botão este posicionado no canto superior esquerdo), e escondendo-se após a realização de alguma

ação. A navegação deve informar tanto para onde o usuário pode ir a partir da página atual como sinalizar onde ele se encontra no momento;

- Quando o usuário tiver que esperar por algum processamento, deve-se fazer o uso de *spinning wheels*¹, e barras de carregamento, de forma a informar ao usuário que algo está sendo realizado;
- Campos, botões e *widgets* em geral devem possuir uma estilização apropriada, que permita que se possa identificar quando os mesmos estão habilitados, selecionados ou em foco;
- Os *links* devem ter a pseudo-classe CSS “:visited”, que indica que um *link* já foi visitado, definida;
- Os *links* internos e externos devem ser discernidos de alguma forma;
- Caso um formulário anteriormente enviado tenha sido identificado como inválido, a razão da invalidade de algum campo ou de todo o formulário deve ser informada juntamente de seus respectivos elementos;
- Caso algum processamento seja realizado, o sucesso ou falha do mesmo deve ser informado;
- Caso aplicável e apropriado, deve-se trazer a identificação de datas, autores e fontes dos conteúdos publicados;
- Na existência de funcionalidades que possam ser ativadas pelo *swipe*², como em carrosséis de imagens ou de menus de navegação, devem existir indicações visuais da existência de tais funcionalidades.

2. Compatibilidade do sistema com o mundo real

- A identidade visual do *website* deve ser adequada ao público-alvo;
- As siglas, abreviaturas e palavras incomuns devem ser apresentadas acompanhadas de suas definições ou de *links* que informem as definições;
- Deve-se usar padrões, convenções e nomenclaturas já estabelecidas, de forma a não causar confusão;
- O vocabulário usado deve ser apropriado para o público-alvo;
- A informação deve ser organizada e ordenada de uma forma coerente com as necessidades do usuário, a fim de que se aumente seu engajamento.

3. Controle e liberdade do usuário

¹ Elemento gráfico animado existente de forma a indicar que alguma ação está sendo executada

² A ação de se arrastar um ou mais dedos de uma extremidade da tela em direção à outra.

- Deve-se manter um *link* para a página inicial em todas as páginas, de forma que o usuário possa retornar para a mesma caso precise tomar novas ações;
- O *website* deve possuir alguma forma de navegação, possibilitando que o usuário possa ir para outras páginas, caso desejar;
- Recomenda-se o uso de breadcrumbs quando houver uma hierarquia de conteúdo, de forma que o usuário possa visualizar onde ele se encontra no momento com relação ao restante do conteúdo;
- Caso aplicável, deve-se prover formas pelas quais o usuário possa desfazer uma ação que realizou ou que está sendo realizada;
- Caso aplicável, deve-se inserir formas pelas quais o usuário possa voltar de onde veio e/ou sair de onde se encontra;
- Todo o processamento ou procedimento que for realizado pelo *website* deve ser originado a partir de uma outra ação explícita do usuário;
- Deve-se permitir que o usuário navegue, edite e revise livremente as informações que ele proveu em um formulário ou em uma série de formulários.

4. Consistência e padrões

- Os textos de menus devem, idealmente, corresponder ao títulos das páginas para onde eles apontam;
- As convenções de nomenclatura usadas devem ser mantidas através de todo o *website* e, quando existirem convenções do mercado, deve-se adotá-las na medida do possível;
- Os alinhamentos textuais devem seguir padrões, e tais padrões devem ser mantidos por todo o *website*. Sugere-se usar alinhamento à esquerda ou justificado para textos e à esquerda ou centralizado para títulos em geral;
- Deve-se manter a mesma estrutura e navegação em todo o *website*;
- Deve-se seguir a mesma identidade visual (cores e tipografia) em todo o *website*;
- Todo o conteúdo deve estar disponível para o usuário independentemente da orientação (retrato ou paisagem) que ele estiver utilizando;
- Os *links* devem estar apropriadamente estilizados, sendo identificados não apenas por meio de uma cor, como pela formatação textual. Também, eles devem seguir um padrão visual em todo o projeto (página inicial e páginas subordinadas);
- A marcação HTML do *site* deve ser válida, usando elementos HTML semânticos e *tags* com aberturas e fechamentos. Algumas práticas que implementam esta recomendação são: uso de *tags* h1–h6 seguindo a ordem hierárquica, uso da

tag table para dados tabulares, a singularidade do atributo “*id*” entre as *tags* existentes e a inexistência de atributos repetidos em uma *tag*.

5. Prevenção de erros

- Os formulários devem informar ao usuário quais campos são obrigatórios. Além disso, caso aplicável, também deve ser informado o formato de preenchimento dos campos, bem como as dimensões mínimas e máximas dos valores inseridos neles;
- Os botões e itens que podem receber interação do usuário deverão ser estilizados em tamanhos grandes o suficiente para que possam sofrer interações de usuários com diferentes tamanhos de dedos;
- Os botões que realizam processamentos devem ser desabilitados ao serem clicados, e assim devem permanecer até que o processamento acabe;
- Os botões que realizam ações destrutivas devem ser colocados em pontos de acesso mais difícil, de forma que os mesmos não sejam pressionados acidentalmente;
- As ações consideradas destrutivas devem ser precedidas de um pedido de confirmação do usuário;
- Os rótulos devem ser claros e informativos;
- Deve-se evitar o uso de tecnologias que exijam que o usuário baixe alguma extensão ou *software* adicional para executá-la, como o Flash.

6. Reconhecimento em vez de memorização

- Deve-se disponibilizar na tela todas as informações necessárias para a realização da ação proposta pela mesma;
- Deve-se posicionar os elementos mais importantes na parte de cima da página, na tentativa de evitar que o usuário tenha que rolar a página para acessar as informações e funções mais comuns;
- Não se recomenda criar páginas que sejam apenas de navegação ou que contenham formulários que sejam muito longas;
- Deve-se usar espaços em branco para distinguir, ressaltar e organizar elementos;
- Não se recomenda usar apenas cores para distinguir uma informação da outra, recomendando-se usar uma formatação textual em conjunto;
- Deve-se exibir os rótulos dos campos juntamente de seus respectivos campos. O posicionamento ideal dos rótulos é em cima dos campos, já que o espaço horizontal nos dispositivos móveis é limitado;

- Deve-se explicitar a unidade de medida do campo quando o seu valor for medido pela mesma;
- Os campos e botões desabilitados devem estar apropriadamente estilizados;
- Deve-se usar imagens para ilustrar e acompanhar conceitos, sem esquecer de se preencher o atributo “alt” com uma representação textual das mesmas;
- O texto dos links deve indicar de forma clara para onde o usuário será direcionado ao pressioná-lo;

7. Flexibilidade e eficiência de uso

- Todo esforço para se deixar o *website* o mais leve possível deve ser feito, de forma que ele seja carregado com facilidade através de conexões mais lentas e/ou instáveis;
- Deve-se implementar uma busca no *website*, busca esta que deve estar sempre acessível e presente em todas as páginas, de forma que o usuário consiga encontrar o que precisa de forma autônoma. O campo da busca deve ser apresentado idealmente no tamanho máximo permitido pela largura do dispositivo, permitindo a escrita de buscas com muitas palavras;
- Os textos muito grandes devem ser divididos em partes. Cada parte deve ter, em seu final, uma navegação para as demais partes. A navegação destas partes deve possibilitar que o usuário acesse instantaneamente cada uma delas sem ter que passar pelas anteriores;
- Deve-se poder dispensar janelas modais informativos ao se clicar em uma área livre (fora do modal) da tela, tirando a necessidade de se apertar em um botão para fechá-las. De qualquer forma, o botão que fecha a janela modal sempre deve estar presente;
- Um endereço de *e-mail*, ao ser pressionado, deve abrir o cliente de *e-mail* do usuário e iniciar a composição de uma nova mensagem direcionada para tal endereço;
- Um telefone, ao ser pressionado, deve requisitar ao usuário se o mesmo deseja ligar para tal número;
- Um endereço, ao ser pressionado, deve direcionar o usuário para um mapa de navegação posicionado em referido endereço;
- Os vídeos devem ser carregados pausados, de forma a preservar a banda do usuário;
- Na medida do possível, o *website* deve tentar diminuir a quantidade de informações que o usuário tem que digitar, já que tal tarefa não é muito agradável

de ser realizada em um dispositivo móvel. Desta forma, pode-se tentar sugerir informações baseadas em outras informações que se possui (através da geolocalização, por exemplo) ou se calcular informações a partir de uma primeira (preencher o endereço a partir do preenchimento do CEP, por exemplo);

- Todo elemento não-textual que contiver informação estruturante e fundamental para o usuário deve possuir uma alternativa textual;
- Não se deve utilizar apenas de sons para se notificar qualquer coisa ao usuário (por exemplo, indicar que o formulário não foi preenchido corretamente tocando um som);
- Deve-se evitar a criação de páginas que atualizam automaticamente. Caso seja interessante que o conteúdo da página seja atualizado por algum motivo, deve-se oferecer tal funcionalidade através de um botão;
- Deve-se criar uma ordem de tabulação lógica entre os elementos do *website*;
- Caso o *website* possua *login* de usuário, deve existir um botão ao lado do campo de senha que permita que a senha seja exibida de forma clara, para que o usuário possa revisar o que foi digitado;
- A navegação deve estar ordenada de forma a otimizar o acesso do público-alvo do *website*;
- Caso o *website* possua uma versão específica para dispositivos móveis e outra para dispositivos *desktop*, deve-se carregar automaticamente a versão específica para dispositivos móveis. Idealmente, a versão para dispositivos móveis deve ter todas as funcionalidades que a versão para dispositivos *desktop* tem. A versão para dispositivos móveis deve prover um link para a versão para dispositivos *desktop* e vice-versa.

8. Estética e design minimalista

- Deve-se minimizar a quantidade de informações na tela, priorizando-se mostrar apenas a informação necessária para a realização da tarefa atual;
- Deve-se minimizar a densidade informacional, de forma a facilitar a localização de uma informação no texto. Indica-se que as sentenças sejam limitadas à um máximo de doze palavras, e que cada parágrafo seja limitado a um máximo de seis sentenças;
- Ao se usar ícones, deve-se garantir que exista uma harmonia visual entre os mesmos, bem como que os mesmos sejam visualmente e conceitualmente distintos;
- Deve-se usar contrastes grandes entre as cores do texto e do plano de fundo, de forma a facilitar a leitura quando o usuário está em ambientes com muita ou pouca iluminação;

- Os rótulos e títulos devem ser curtos, mas objetivos e descritivos;
- Ao se utilizar imagens e vídeos, deve-se incluir uma breve legenda explicando do que tais elementos se tratam. No caso dos vídeos, a legenda deve incluir a duração do vídeo;
- Deve-se evitar o uso de imagens e animações apenas para fins estéticos, já que elas podem estar roubando um espaço que poderia estar sendo usado para exibir uma informação mais útil ou, ainda, poluindo o visual do *website*;
- Deve-se evitar que se tenha que realizar qualquer deslocamento horizontal da tela durante a utilização do *website*, especialmente ao se ler algum conteúdo;
- Deve-se procurar utilizar uma única fonte em todo o *website*, utilizando variações de tamanho e formatação para diferenciar os diferentes contextos e casos;
- Caso uma imagem possua uma versão maior do que o tamanho da tela, deve-se apresentá-la no tamanho da tela e possibilitar o acesso ao tamanho original da mesma ao pressioná-la ou através de um link situado próximo à mesma.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

- Na existência de qualquer erro, a mensagem de erro deverá ser apresentada em uma linguagem simples, não contendo nada enigmático (algum código de erro interno da aplicação, por exemplo) e explicitando claramente o que aconteceu de errado, bem como a forma pela qual o usuário pode se recuperar de tal erro;
- Na existência de um erro indicando que a página não foi encontrada (404), deve-se prover sugestões de páginas afins ou um campo de busca, permitindo que ele encontre a página correta (caso ela tenha sido movida) ou páginas que se relacionem ao conteúdo desta;
- Na existência de erros de preenchimento de formulário, os campos que estão com inconsistência devem ser destacados e acompanhados de uma mensagem descrevendo a inconsistência.

10. Ajuda e documentação

- Em uma seção específica de ajuda, a mesma deve ser claramente acessível através de algum menu ou submenu, ordenada conforme a ordem de ações que o usuário realizará na página e pesquisável através de uma busca específica para a mesma;
- O uso de mapas de *site* pode ajudar o usuário a se localizar em *websites* que possuem muito conteúdo;
- O uso de FAQs (*Frequently Asked Questions*, ou “Questões Mais Perguntadas”) pode ser interessante para solucionar as dúvidas mais recorrentes do usuário;

- Caso algum campo de um formulário possua detalhes muito específicos de preenchimento, pode-se exibir uma ajuda juntamente com o campo;

A fim de se verificar a efetividade do Guia de Recomendações para *Design* de Interfaces *Web* Móveis, desenvolver-se-á um protótipo, aplicando as recomendações citadas. O protótipo a ser realizado será um *website* de *e-commerce*, que trata-se de um ambiente rico em informações e com vários tipos de interação diferentes, o que será vantajoso para se tentar aproveitar ao máximo as recomendações verificadas. Após implementação, o protótipo será avaliado através da avaliação heurística.

6 Desenvolvimento

Este capítulo apresenta a análise e implementação do protótipo de *e-commerce*, que foi desenvolvido com base no guia de recomendações para *design* de interfaces *web* móveis produzido no Capítulo 5.

6.1 Levantamento de Requisitos

A fim de se verificar quais funcionalidades o protótipo deveria ter, realizou-se uma pesquisa em sistemas afins. Então, a partir destes, elencou-se quais os requisitos necessários.

De forma a considerar um escopo apropriado ao protótipo proposto, restringiu-se a pesquisa apenas para *websites* de *e-commerce* que vendem exclusivamente livros. A partir desta definição, escolheu-se os seguintes *websites*:

- Amazon.com.br (<https://www.amazon.com.br/>): trata-se da versão localizada para o Brasil do maior *website* de varejo *online* dos Estados Unidos da América. Porém, enquanto a versão americana da loja comercializa uma grande variedade de itens, como livros, *e-books*, CDs, eletrônicos, brinquedos e até alimentos, a versão brasileira somente comercializa livros e *e-books*.

O *website* da Amazon.com.br possui, como características, a existência de uma busca, o cadastro e *login* de um cliente, a possibilidade de se adicionar produtos a um carrinho de compras virtual, a possibilidade de se adicionar produtos à uma lista de desejos e a possibilidade de se adicionar avaliações aos produtos.

Também, pode-se gerenciar as informações do cliente, o que possibilita a alteração dos dados pessoais do mesmo e a consulta dos pedidos já realizados, bem como a possibilidade de se adicionar e gerenciar múltiplos endereços e múltiplas listas de desejos.

- Martins Fontes Paulista (<http://www.martinsfontespaulista.com.br/>): a Martins Fontes Paulista é uma livraria localizada na cidade de São Paulo, no estado de São Paulo, que foi fundada nos anos 60. Sua loja *online* comercializa exclusivamente livros físicos.

O *e-commerce* da Martins Fontes Paulista possui, como características, a existência de uma busca, o cadastro e *login* de um cliente, a possibilidade de se adicionar produtos a um carrinho de compras virtual e a possibilidade de se adicionar produtos à uma lista de desejos.

Também, pode-se gerenciar as informações do cliente, o que possibilita a alteração dos dados pessoais do mesmo e a consulta dos pedidos já realizados, bem como a possibilidade de se adicionar e gerenciar múltiplos endereços.

- Cia. dos Livros (<http://www.ciadoslivros.com.br/>): a Cia. dos Livros é uma livraria com sede localizada na cidade de Jandira, no estado de São Paulo. Sua loja *online* comercializa livros físicos e quadrinhos.

O *e-commerce* da Cia. dos Livros possui, como características, a existência de uma busca, o cadastro e *login* de um cliente, a possibilidade de se adicionar produtos a um carrinho de compras virtual, a possibilidade de se adicionar produtos à uma lista de desejos e a possibilidade de se adicionar avaliações aos produtos.

Também, pode-se gerenciar as informações do cliente, o que possibilita a alteração dos dados pessoais do mesmo e a consulta dos pedidos já realizados, bem como a possibilidade de se adicionar e gerenciar múltiplos endereços e múltiplas listas de desejos.

De uma forma geral, algumas características são inerentes de sistemas de *e-commerce*, características estas também presentes nos *e-commerces* de livros pesquisados. Tais características são: a existência de uma busca no catálogo, o cadastro e *login* de um cliente (e as atividades relacionadas a isso, como o gerenciamento do perfil e a visualização de pedidos) e a adição de produtos a um carrinho de compras virtual (e as atividades relacionadas a isso, como a edição da quantidade de um certo produto no carrinho e a remoção do mesmo).

Algumas características, apesar de não serem indispensáveis em um *e-commerce*, mostraram-se presentes em todos os *websites* analisados. Tais características são a adição de produtos à uma lista de desejos e o gerenciamento de múltiplos endereços de entrega.

Também, todos os sistemas pesquisados ofereceram a possibilidade de navegação no acervo de livros por meio de das categorias. Apesar de cada um deles disponibilizar tal funcionalidade de forma diferente, entendeu-se que a existência de tal funcionalidade é interessante para o protótipo que será criado.

Por fim, apesar de uma característica ter sido exclusiva de apenas alguns dos *e-commerces* pesquisados, ela se considerou desejável e interessante de ser apresentadas no protótipo. Tal característica é a possibilidade de se adicionar avaliações aos produtos.

6.2 Requisitos Funcionais

A partir da pesquisa realizada, criaram-se os seguintes requisitos funcionais:

- RF-01: O sistema deve possibilitar a navegação pelos livros por meio de suas categorias

- RF-02: O sistema deve oferecer a busca pelos livros existentes no catálogo
- RF-03: O sistema deve oferecer um cadastro para novos clientes
- RF-04: O sistema deve permitir que um visitante possa se autenticar no sistema
- RF-05: O sistema deve permitir que um cliente autenticado possa visualizar os seus pedidos
- RF-06: O sistema deve permitir que um cliente autenticado possa gerenciar seu perfil
- RF-07: O sistema deve permitir que um produto seja adicionado ao seu carrinho de compras
- RF-08: O sistema deve permitir que a quantidade de um produto adicionado no seu carrinho de compras possa ser alterada
- RF-09: O sistema deve permitir que um produto seja removido do carrinho de compras
- RF-10: O sistema deve permitir que um cliente autenticado possa finalizar um pedido
- RF-11: O sistema deve permitir que um cliente autenticado possa adicionar um endereço
- RF-12: O sistema deve permitir que um cliente autenticado possa editar um endereço
- RF-13: O sistema deve permitir que um cliente autenticado possa excluir um endereço
- RF-14: O sistema deve permitir que um cliente autenticado possa avaliar um produto
- RF-15: O sistema deve permitir que um cliente autenticado possa adicionar um produto a sua lista de desejos
- RF-16: O sistema deve permitir que um cliente autenticado possa remover um produto de sua lista de desejos

A próxima seção apresenta os Casos de Uso que detalham os requisitos funcionais levantados.

6.3 Casos de Uso

6.3.1 UC-01: Cadastrar novo cliente

- **Descrição:** O sistema deve permitir que um visitante crie seu cadastro no sistema.

- **Atores:** Visitante
- **Requisitos funcionais atendidos:** RF-03
- **Pré-condições:** Não há
- **Pós-condições:** Um cliente foi cadastrado
- **Ações possíveis:**
 1. O visitante preenche os campos Endereço de e-mail, Senha e Confirme a senha, já que todos eles são obrigatórios
 2. O visitante clica no botão Cadastrar
 3. O sistema verifica se os campos obrigatórios foram preenchidos
 - 3.1. O sistema verificou que um ou mais campos obrigatórios não foram preenchidos
 - 3.2. O sistema retorna ao passo 1, mostrando novamente o formulário e trazendo os dados que o visitante preencheu, mas informando junto aos campos irregulares que os mesmos não se encontram preenchidos
 4. O sistema verifica se os campos Senha e Confirme a senha conferem
 - 4.1. O sistema verificou que os campos Senha e Confirme a senha não conferem
 - 4.2. O sistema retorna ao passo 1, mostrando novamente o formulário e trazendo os dados que o visitante preencheu, mas informando que os campos Senha e Confirme a senha devem ser iguais
 5. O sistema verifica se o endereço de e-mail informado ainda não foi utilizado no cadastro de algum cliente
 - 5.1. O sistema verificou que o endereço de e-mail informado já foi utilizado no cadastro de algum cliente
 - 5.2. O sistema retorna ao passo 1, mostrando novamente o formulário e trazendo os dados que o visitante preencheu, mas informando que o endereço de e-mail informado já foi utilizado por outro cliente
 6. O sistema persiste os dados informados

6.3.2 UC-02: Autenticar cliente

- **Descrição:** O sistema deve permitir que um visitante se autentique no sistema.
- **Atores:** Visitante
- **Requisitos funcionais atendidos:** RF-04

- **Pré-condições:** O visitante deve possuir um cadastro de cliente
- **Pós-condições:** O visitante deve estar autenticado com a sua respectiva conta de cliente
- **Ações possíveis:**
 1. O visitante preenche os campos de Endereço de e-mail e Senha, sendo ambos de preenchimento obrigatório
 2. O visitante clica no botão Entrar
 3. O sistema verifica se os campos obrigatórios foram preenchidos
 - 3.1. O sistema verificou que um ou mais campos obrigatórios não foram preenchidos
 - 3.2. O sistema retorna ao passo 1, mostrando novamente o formulário e trazendo os dados que o visitante preencheu, mas informando junto aos campos irregulares que os mesmos não se encontram preenchidos
 4. O sistema verifica se a combinação de endereço de e-mail e senha está correta
 - 4.1. O sistema verificou que a combinação de endereço de e-mail e senha está incorreta
 - 4.2. O sistema retorna ao passo 1, mostrando novamente o formulário e trazendo os dados que o visitante preencheu, mas informando que a combinação de endereço de e-mail e senha informados está incorreta
 5. O sistema autentica o visitante como cliente e redireciona-o para a tela inicial do sistema

6.3.3 UC-03: Consultar pedidos

- **Descrição:** O sistema deve permitir que um cliente consulte seus pedidos.
- **Atores:** Cliente
- **Requisitos funcionais atendidos:** RF-05
- **Pré-condições:** O visitante deve estar autenticado com a sua respectiva conta de cliente
- **Pós-condições:** Os pedidos do cliente são listados na tela
- **Ações possíveis:**
 1. O cliente clica no link Historico de pedidos para listar seus pedidos

2. O sistema lista todos os pedidos do cliente
3. O cliente seleciona o pedido desejado, clicando sobre ele
4. O sistema exibe o pedido selecionado

6.3.4 UC-04: Gerenciar dados pessoais

- **Descrição:** O sistema deve permitir que um cliente gerencie o seu perfil
- **Atores:** Cliente
- **Requisitos funcionais atendidos:** RF-06
- **Pré-condições:** O visitante deve estar autenticado com a sua respectiva conta de cliente
- **Pós-condições:** Os dados do perfil atualizados são exibidos na tela
- **Ações possíveis:**
 1. O visitante preenche o campo de Endereço de e-mail, sendo este de preenchimento obrigatório, e os campos de Primeiro nome e Último nome, de preenchimento facultativo
 2. O visitante clica no botão Salvar
 3. O sistema verifica se o campo obrigatório foi preenchido
 - 3.1. O sistema verificou que o campo obrigatório não foi preenchido
 - 3.2. O sistema retorna ao passo 1, mostrando novamente o formulário e trazendo os dados que o visitante preencheu, mas informando junto ao campo irregular que o mesmo não se encontra preenchido
 4. O sistema exibe os dados do perfil atualizados na tela

6.3.5 UC-05: Adicionar produto ao carrinho de compras

- **Descrição:** O sistema deve permitir que um cliente/visitante adicione um produto ao seu carrinho de compras
- **Atores:** Cliente/Visitante
- **Requisitos funcionais atendidos:** RF-07
- **Pré-condições:** Não há
- **Pós-condições:** O produto deve estar presente no carrinho de compras do cliente/visitante

- **Ações possíveis:**

1. O cliente/visitante clica no botão Adicionar ao carrinho
2. O sistema adiciona o produto ao carrinho de compras e redireciona o usuário para o carrinho de compras

6.3.6 UC-06: Alterar a quantidade de um produto já presente no carrinho de compras

- **Descrição:** O sistema deve permitir que um cliente/visitante altere a quantidade de um produto já presente em seu carrinho de compras

- **Atores:** Cliente/Visitante

- **Requisitos funcionais atendidos:** RF-08

- **Pré-condições:** O produto deve estar presente no carrinho de compras do cliente/visitante

- **Pós-condições:** O produto deve estar presente no carrinho de compras do cliente/visitante com a sua respectiva quantidade alterada

- **Ações possíveis:**

1. O cliente/visitante muda o valor do campo Quantidade do produto desejado e clica no botão Atualizar
2. O sistema verifica se a quantidade desejada existe em estoque
 - 2.1. O sistema verificou que a quantidade requisitada do produto não existe em estoque
 - 2.2. O sistema retorna ao passo 1, mostrando novamente o formulário e trazendo os dados que o cliente/visitante preencheu, mas informando junto ao produto que teve a quantidade alterada que a quantidade requisitada não existe em estoque
3. O sistema retorna ao passo 1, mostrando novamente o formulário e trazendo os dados que o cliente/visitante preencheu, com a quantidade do produto atualizada

6.3.7 UC-07: Remover um produto do carrinho de compras

- **Descrição:** O sistema deve permitir que um cliente/visitante remova um produto de seu carrinho de compras

- **Atores:** Cliente/Visitante
- **Requisitos funcionais atendidos:** RF-09
- **Pré-condições:** O produto deve estar presente no carrinho de compras do cliente/visitante
- **Pós-condições:** O produto não deve estar presente no carrinho de compras do cliente/visitante
- **Ações possíveis:**
 1. O cliente/visitante clica no link Remover
 2. O sistema retorna ao passo 1, mostrando novamente o formulário, com o produto requisitado para remoção removido do Carrinho de compras

6.3.8 UC-08: Finalizar um pedido

- **Descrição:** O sistema deve permitir que um cliente autenticado possa finalizar um pedido
- **Atores:** Cliente
- **Requisitos funcionais atendidos:** RF-10
- **Pré-condições:** O cliente deve estar logado no sistema; deve haver ao menos um produto no carrinho de compras
- **Pós-condições:** O pedido do cliente logado deve estar finalizado
- **Ações possíveis:**
 1. O cliente clica no botão Finalizar o pedido agora.
 2. O sistema direciona o cliente para a tela de seleção de endereços.
 3. O cliente seleciona um dos endereços existentes clicando no botão Envie para este endereço equivalente ao endereço desejado.
 4. O cliente confirma o pagamento do pedido (não implementado neste projeto, sendo realizado apenas de forma simulada).
 5. O sistema direciona o cliente para a tela de resumo do pedido realizado.
 6. O cliente clica em Realizar pedido.
 7. O sistema direciona o cliente para a tela de confirmação de realização do pedido.

6.3.9 UC-09: Adicionar um possível endereço de entrega

- **Descrição:** O sistema deve permitir que um cliente logado possa adicionar um possível endereço de entrega à sua lista de endereços.
- **Atores:** Cliente
- **Requisitos funcionais atendidos:** RF-11
- **Pré-condições:** O cliente deve estar logado no sistema; o cliente deve estar na tela de Livro de Endereços
- **Pós-condições:** O novo endereço deverá estar cadastrado na lista de endereços
- **Ações possíveis:**
 1. O cliente clica no link Adicionar novo endereço
 2. O sistema direciona o cliente para o formulário de novo endereço
 3. O cliente preenche os campos de Nome, Sobrenome, Endereço, Cidade, CEP e País, sendo estes de preenchimento obrigatório, e os campos de Título, Complemento, Bairro, Estado, Telefone e Instruções, de preenchimento facultativo
 4. O cliente clica no botão Salvar
 5. O sistema verifica se os campos obrigatórios foram preenchidos
 - 5.1. O sistema verificou que os campos obrigatórios não foram preenchidos
 - 5.2. O sistema retorna ao passo 3, mostrando novamente o formulário e trazendo os dados que o cliente preencheu, mas informando junto aos campos irregulares que os mesmos não se encontram preenchidos
 6. O sistema volta para a tela de Livro de Endereços, informando que o novo endereço foi adicionado

6.3.10 UC-10: Modificar um possível endereço de entrega

- **Descrição:** O sistema deve permitir que um cliente logado possa modificar um possível endereço de entrega de sua lista de endereços.
- **Atores:** Cliente
- **Requisitos funcionais atendidos:** RF-12
- **Pré-condições:** O cliente deve estar logado no sistema; o cliente deve estar na tela de Livro de Endereços

- **Pós-condições:** O endereço afetado pelas alterações deverá estar devidamente alterado na lista de endereços
- **Ações possíveis:**
 1. O cliente clica no link Editar do endereço que deseja alterar
 2. O sistema direciona o cliente para o formulário de edição de endereço
 3. O cliente preenche os campos de Nome, Sobrenome, Endereço, Cidade, CEP e País, sendo estes de preenchimento obrigatório, e os campos de Título, Complemento, Bairro, Estado, Telefone e Instruções, de preenchimento facultativo
 4. O cliente clica no botão Salvar
 5. O sistema verifica se os campos obrigatórios foram preenchidos
 - 5.1. O sistema verificou que os campos obrigatórios não foram preenchidos
 - 5.2. O sistema retorna ao passo 3, mostrando novamente o formulário e trazendo os dados que o cliente preencheu, mas informando junto aos campos irregulares que os mesmos não se encontram preenchidos
 6. O sistema volta para a tela de Livro de Endereços, informando que o endereço foi alterado

6.3.11 UC-11: Excluir um possível endereço de entrega

- **Descrição:** O sistema deve permitir que um cliente logado possa excluir um possível endereço de entrega de sua lista de endereços.
- **Atores:** Cliente
- **Requisitos funcionais atendidos:** RF-13
- **Pré-condições:** O cliente deve estar logado no sistema; o cliente deve estar na tela de Livro de Endereços
- **Pós-condições:** O endereço afetado pelas alterações deverá ter sido devidamente removido da lista de endereços
- **Ações possíveis:**
 1. O cliente clica no link Apagar do endereço que deseja remover
 2. O sistema direciona o cliente para a tela de confirmação de exclusão do endereço
 3. O cliente clica no botão Apagar
 4. O sistema volta para a tela de Livro de Endereços, informando que o endereço foi removido

6.3.12 UC-12: Avaliar um produto

- **Descrição:** O sistema deve permitir que um cliente avalie um produto
- **Atores:** Cliente
- **Requisitos funcionais atendidos:** RF-14
- **Pré-condições:** O visitante deve estar autenticado com a sua respectiva conta de cliente; o cliente deve estar na página do produto que ele deseja avaliar
- **Pós-condições:** O produto terá a sua nova avaliação.
- **Ações possíveis:**
 1. O cliente clica no link Avalie este produto
 2. sistema exibirá o formulário para a criação da avaliação
 3. O cliente preenche os campos Título, Nota e Corpo, todos estes obrigatórios
 - 3.1. O sistema verificou que os campos obrigatórios não foram preenchidos
 - 3.2. O sistema retorna ao passo 2, mostrando novamente o formulário e trazendo os dados que o cliente preencheu, mas informando junto aos campos irregulares que os mesmos não se encontram preenchidos
 4. O sistema volta para a página do produto, informando que a avaliação do produto foi criada

6.3.13 UC-13: Adicionar produto à lista de desejos

- **Descrição:** O sistema deve permitir que um cliente adicione um produto à sua lista de desejos
- **Atores:** Cliente
- **Requisitos funcionais atendidos:** RF-15
- **Pré-condições:** O visitante deve estar autenticado com a sua respectiva conta de cliente; o cliente deve estar na página do produto que ele deseja adicionar à lista de desejos
- **Pós-condições:** O produto deve estar presente na lista de desejos do cliente
- **Ações possíveis:**
 1. O cliente clica no botão Adicionar à lista de desejos
 2. O sistema adiciona o produto à lista de desejos e redireciona o cliente para a página do produto

6.3.14 UC-14: Remover um produto da lista de desejos

- **Descrição:** O sistema deve permitir que um cliente remova um produto de sua lista de desejos
- **Atores:** Cliente
- **Requisitos funcionais atendidos:** RF-16
- **Pré-condições:** O visitante deve estar autenticado com a sua respectiva conta de cliente; o produto deve estar presente na lista de desejos cliente; o cliente deve estar na tela da Lista de desejos
- **Pós-condições:** O produto não deve estar presente na lista de desejos do cliente
- **Ações possíveis:**
 1. O cliente clica no link Remover
 2. O sistema pergunta se o usuário tem certeza de que quer remover o item
 3. O usuário confirma a ação
 4. O sistema retorna ao passo 1, mostrando novamente a lista de desejos, com o produto requisitado para remoção removido da lista de desejos

6.3.15 UC-15: Buscar por um produto

- **Descrição:** O sistema deve permitir que um cliente/visitante busque por um produto.
- **Atores:** Cliente/Visitante.
- **Requisitos funcionais atendidos:** RF-02
- **Pré-condições:** Não há
- **Pós-condições:** O resultado da busca deve estar sendo mostrado para o cliente/visitante
- **Ações possíveis:**
 1. O cliente/visitante preenche o campo de busca com a (s) palavra (s)-chave desejada (s) e clica no botão Buscar
 2. O sistema apresenta ao cliente/visitante o resultado da busca

6.4 Implementação

Nesta seção, será detalhada a implementação do protótipo do *e-commerce* desenvolvido com base no guia de recomendações para *design* de interfaces *web* móveis.

O processo de implementação foi prumado com foco no desenvolvimento para dispositivos móveis, exatamente como a filosofia *Mobile First* diz, com ênfase em celulares.

6.4.1 Componentes

Para a implementação deste protótipo, algumas linguagens e *frameworks* foram utilizados. Em seguida, apresentar-se-á estes elementos.

6.4.1.1 Python

Python (<https://www.python.org>) é uma linguagem de programação de altíssimo nível orientada a objetos, de tipagem forte e dinâmica, interpretada e interativa. Ela foi criada por Guido van Rossum em 1990, e possui um modelo de desenvolvimento comunitário e aberto. Sua especificação é mantida pela *Python Software Foundation*.

A escolha desta linguagem de programação para a produção do protótipo deve-se ao fato dela ter uma sintaxe clara e simples, além de possuir muitos módulos prontos para o uso e diversos recursos que visam facilitar a vida do desenvolvedor.

6.4.1.2 Django

Django (<https://www.djangoproject.com/>) é um *framework* para desenvolvimento web. Ele é totalmente escrito em Python e utiliza o padrão MVC (*Model View Controller*). Atualmente, trata-se de um projeto de código aberto, sendo publicado sob a licença BSD em 2005.

A escolha pela utilização do *framework* deu-se pelo grau de abstração que o mesmo possui, o que proporciona que o desenvolvimento *web* seja o mais direto e produtivo possível.

6.4.1.3 Oscar

Oscar (<http://oscarcommerce.com/>) é um *framework* de *e-commerce* feito para a utilização com o Django. Ele implementa as funções mais básicas de um site de *e-commerce*. Trata-se de um projeto de código aberto, sendo publicado sob a licença BSD.

A escolha pela utilização do *framework* ocorreu pelo fato do mesmo implementar muitas das funções mais básicas de um *e-commerce*. Graças à esta robusta implementação dos recursos necessários para o protótipo, não houveram mudanças na modelagem do Oscar e, portanto, pode-se dizer que a modelagem do projeto segue a modelagem do Oscar.

Também escolheu-se utilizar o Oscar graças ao alto grau de customização que ele permite, proporcionando que se realizassem as adequações funcionais e estéticas necessárias de forma rápida e direta.

6.4.2 Customização

O protótipo criado, apesar de utilizar-se do *framework* Oscar, foi fruto de um alto grau de customização do mesmo.

De forma a contemplar a necessidade de se apresentar o *website* de forma agradável em um dispositivo móvel e apesar de ter-se utilizado dos *templates* do Oscar como ponto inicial do projeto, todos os *templates* utilizados no protótipo criado foram fortemente customizados. Além disso, houve a criação de um *template* novo, de forma a contemplar especificamente o formulário de “Avise-me”, que é exibido quando um produto está esgotado.

Da mesma forma, também utilizou-se das folhas de estilos do Oscar como ponto inicial mas, como já era esperado, elas também sofreram um alto grau de customização. De forma a contemplar as telas menores que estes dispositivos possuem, foi frequente a utilização de *media queries* para a correta identificação das dimensões do dispositivo atual e a correta adequação da interface para o mesmo.

Também, de forma mais pontual, houve a inclusão de certas funcionalidades através de Javascript, de forma a deixar a experiência de uso mais agradável.

Por fim, apesar da modelagem do Oscar ser adequada para o protótipo, necessitou-se customizar algumas aplicações do *framework*, a fim de que o comportamento do *e-commerce* fosse de acordo com o esperado e desejado: customizou-se a aplicação “basket”, de forma que esta passasse a redirecionar o usuário para a Cesta de Compras após a adição de um produto; e das aplicações de “checkout” e “customer”, para que os formulários de endereço do protótipo fossem exibidos em um formato mais adequado para um usuário brasileiro.

6.4.3 Identidade Visual

A fim de se obter um visual agradável e harmonioso, algumas decisões foram feitas no que diz respeito à identidade visual do protótipo:

- Todas as páginas do *website* são compostas da mesma estrutura: o menu superior, fixo, no topo da página; o conteúdo que, quando necessário, desliza sob a barra superior; e o rodapé, no fim da página;
- O protótipo não utiliza formas muito quadradas e, logo, os elementos que compõem a interface (botões, caixas de mensagens, etc.) possuem as bordas arredondadas;

- Todos os *links* são devidamente formatados com o uso de negrito e sublinhado, de forma que os mesmos sejam facilmente discerníveis do restante dos elementos. Também, *links* externos são discernidos dos demais através de um ícone que é posicionado ao lado dos mesmos;
- Com relação às cores, evitou-se a utilização de muitas delas, primando-se pela utilização de variações de marrom e cinza, branco e preto, salvos alguns detalhes bem específicos;
- Os botões do protótipo são estilizados de duas formas: com fundo colorido (marrom) e fonte branca, quando trata-se de um botão com uma função considerada mais importante no formulário; e com fundo branco, fonte colorida (marrom) e borda colorida (marrom), quando o botão em questão realiza uma função mais secundária;
- Filtros e outros elementos que não são usados com tanta frequência pelo usuário são ocultados na tela, sendo exibidos ao se apertar um botão;

6.4.4 Limitações

O guia de recomendações criado no Capítulo 5 incluiu recomendações amplas para desenvolvimento de interfaces móveis em geral. Algumas recomendações, porém, não puderam ser inteiramente contempladas neste protótipo devido à, principalmente, restrições causadas pelo curto tempo para o desenvolvimento do mesmo. Pode-se citar, como recomendações não inteiramente contempladas:

- “Na existência de um erro indicando que a página não foi encontrada (404), deve-se prover sugestões de páginas afins ou um campo de busca, permitindo que ele encontre a página correta (caso ela tenha sido movida) ou páginas que se relacionem ao conteúdo desta”. Embora a página de 404 existente no protótipo informe corretamente o que está acontecendo e direcione-o para algumas possíveis soluções de seu problema, a sugestão de páginas afins ou que se relacionem ao conteúdo da página com erro demandaria alguma implementação extra no *backend*, algo para o qual não houve tempo hábil;
- “A marcação HTML do *site* deve ser válida, usando elementos HTML semânticos e *tags* com aberturas e fechamentos. Algumas práticas que implementam esta recomendação são: uso de *tags* h1–h6 seguindo a ordem hierárquica, uso da *tag* *table* para dados tabulares, a singularidade do atributo “*id*” entre as *tags* existentes e a inexistência de atributos repetidos em uma *tag*”. Embora tenha-se tentado criar marcação HTML válida durante o processo de desenvolvimento, parte do código existente foi herdado do *framework* Oscar e, como não foi feita a verificação de sua

validade, prefere-se não se assumir que a marcação HTML do protótipo em sua totalidade encontra-se válida.

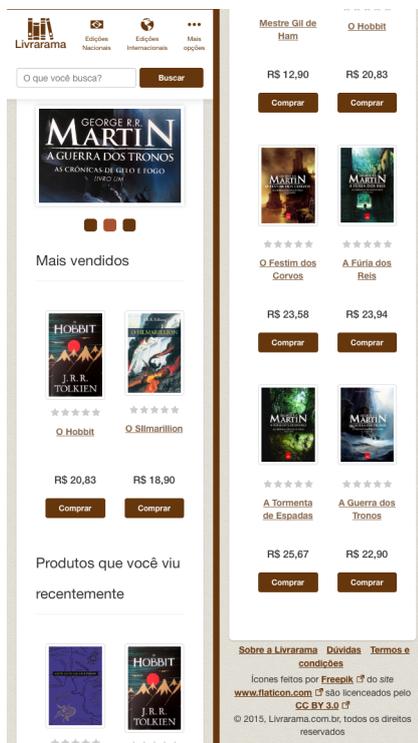
6.5 Telas

De forma a demonstrar visualmente como o protótipo desenvolvido ficou, apresentar-se-ão algumas *screenshots* de telas do mesmo.

Todas as telas aqui mostradas possuem um menu superior fixo, que contém a navegação para a página inicial, botões para se acessar as categorias e um botão para se acessar funções usadas com menos frequência, como a autenticação e a consulta ao carrinho de compras. Além disso, todas as telas também possuem um rodapé fixo, contendo *links* para informações institucionais e para a ajuda.

Também, toda vez que um item é apresentado em alguma categoria, busca ou seção, o mesmo é representado através de uma imagem da capa do item, seguido da nota média das avaliações dos usuários, do título do item, o preço do mesmo e, por fim, de um botão que permite que o usuário compre o produto.

Figura 5 – Tela da página inicial.



Fonte: produzido pelo autor.

A tela da página inicial do protótipo, Figura 5, possui um *slider* de imagens de ofertas, lançamentos ou produtos, seguida de uma seção que exibe os itens mais vendidos e de uma seção que exibe os itens vistos recentemente.

A navegação, Figura 6, quando ativada, exibe o nome do usuário que está autenticado no *e-commerce* no momento (ou “Anônimo” caso não haja usuário autenticado), seguido de um *link* para se visualizar a cesta de compras, um *link* para se acessar a conta do usuário e um *link* para se sair do *e-commerce*.

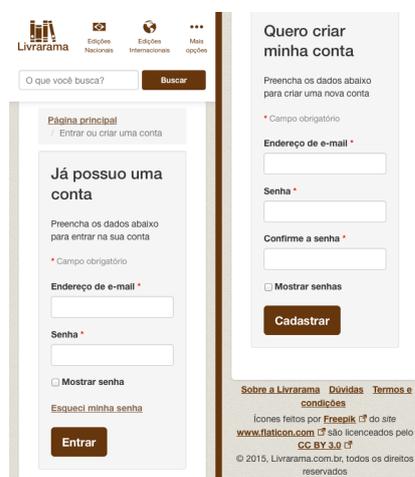
Figura 6 – Tela da navegação.



Fonte: produzido pelo autor.

Na tela de entrar/criar conta, Figura 7, temos, em sua parte superior, o formulário de autenticação, que é usado com mais frequência. Logo abaixo deste, localiza-se o formulário de registro.

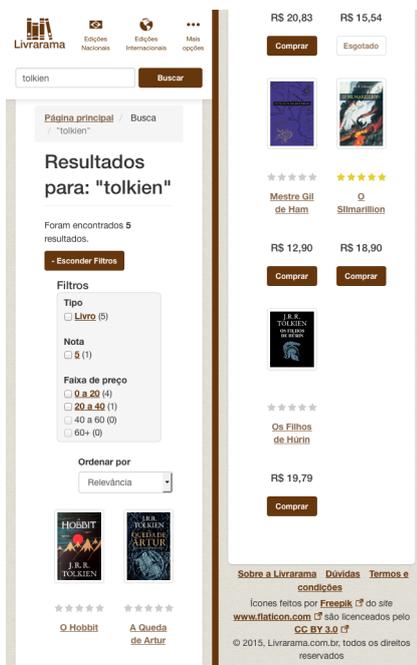
Figura 7 – Tela de Entrar/Criar conta.



Fonte: produzido pelo autor.

A tela de busca, Figura 8, possui, na parte superior, um botão que exibe os filtros com os quais o usuário pode melhorar a sua busca e, logo abaixo, os resultados da busca em si.

Figura 8 – Tela de Busca.



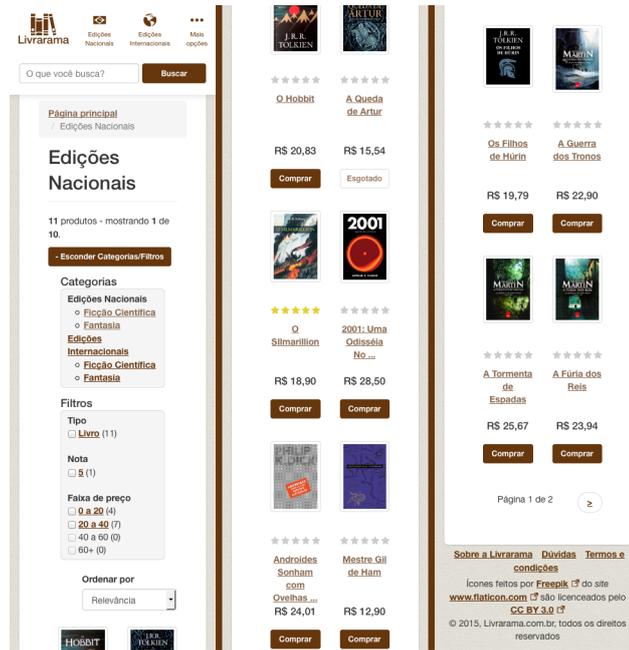
Fonte: produzido pelo autor.

Na tela do catálogo, Figura 9, exibida ao se clicar em um dos *links* das categorias, temos, primeiramente, um botão que exibe as categorias e filtros os quais o usuário pode aplicar a fim de facilitar sua navegação, seguido dos itens que fazem parte da categoria em questão.

A tela de detalhes, Figura 10, exibida ao se clicar em um item, possui uma imagem da capa do item, seguida do título do item, de seu preço e da nota média das avaliações do usuário. A seguir, existe um botão para se adicionar o item à cesta, um botão para se avaliar um produto e outro botão para se adicionar o item à lista de desejos. Por fim, ainda existem na tela as informações do produto si (como o ano, autor e a edição), as avaliações que foram feitas pelos clientes para o produto em questão e uma seção que exibe os itens vistos recentemente.

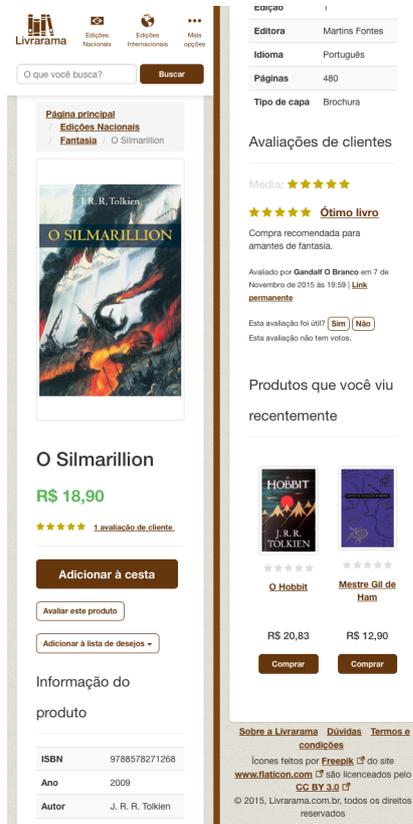
Na tela da cesta de compras, Figura 11, para cada produto existente nela, exibe-se uma miniatura da capa do item, o título do mesmo, a quantidade de exemplares daquele item que se deseja comprar, bem como um botão que permite atualizar esta quantidade, um *link* para se remover o item da cesta, o valor unitário do item e o seu valor total. Mais abaixo, exibe-se o valor total da cesta, o valor do frete e o valor total do pedido. Por fim,

Figura 9 – Tela do Catálogo.



Fonte: produzido pelo autor.

Figura 10 – Tela de Detalhes.



Fonte: produzido pelo autor.

exibe-se um botão que permite fechar o pedido e um segundo botão que permite que se continue comprando.

Figura 11 – Tela da Cesta de Compras.



Fonte: produzido pelo autor.

A tela da conta do usuário, Figura 12, possui, na parte superior, um botão que exibe *links* para as seções da conta, sendo elas o perfil, o histórico de pedidos, a lista de endereços e a lista de desejos. Abaixo destes *links*, exibe-se o conteúdo referente à seção da conta que se está navegando no momento.

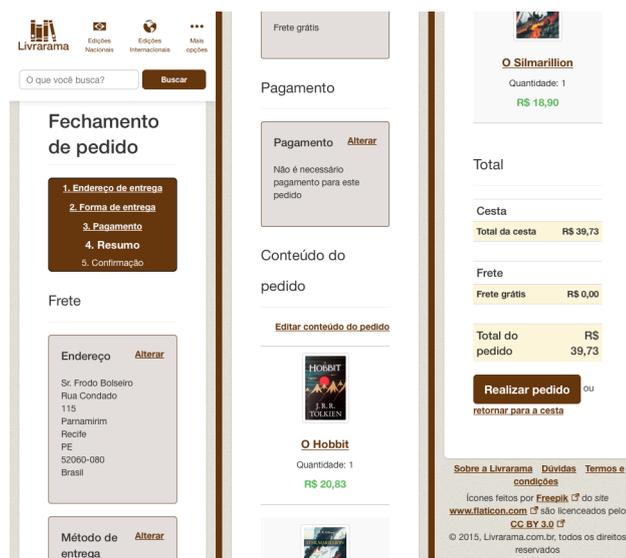
Figura 12 – Tela da Conta do usuário.



Fonte: produzido pelo autor.

Na tela de fechamento do pedido, Figura 13, na parte superior, exibe-se uma navegação dos passos que compõem o fechamento de um pedido, sendo estes a escolha do endereço de entrega, a escolha da forma de entrega, a escolha da forma de pagamento, a exibição de um resumo do pedido e a confirmação da realização do pedido. Abaixo desta navegação, exibe-se o conteúdo referente ao passo que se está no momento.

Figura 13 – Tela de fechamento de pedido.



Fonte: produzido pelo autor.

6.6 Diagrama de atividades

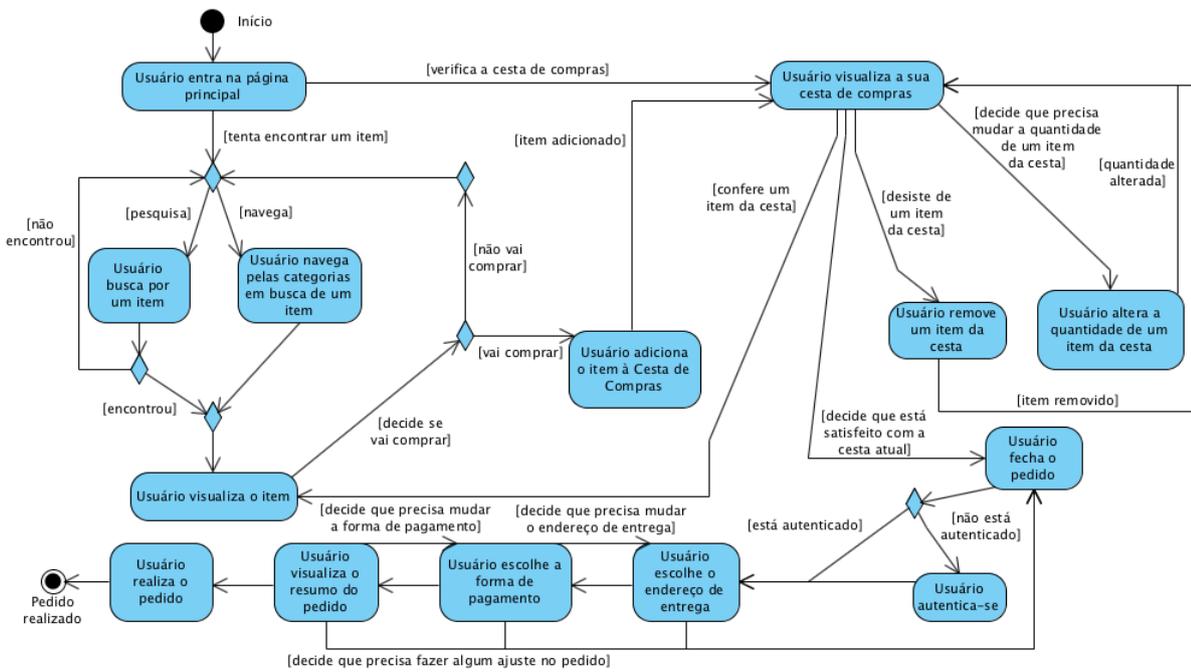
A Figura 14 descreve como o processo de realização de um pedido, a interação mais importante no protótipo, desdobra-se.

O processo inicia quando o visitante/cliente entra no *website* pela página principal procurando por algum item. Ele pode o fazer usando a busca ou através das categorias do menu de navegação. Ao encontrar o item, ele o visualiza, decidindo se o comprará. Ao decidir-se por comprá-lo, o visitante/cliente é redirecionado para a cesta de compras, onde ele pode conferir o que ele está comprando no momento, bem como alterar a quantidade de um item ou removê-lo da cesta.

As ações acima se repetem até que o visitante/cliente decida por fechar o pedido. A partir daí, caso quem esteja visitando o *website* seja um visitante, requer-se que o mesmo se autentique com sua conta de cliente ou que crie uma nova.

O processo de fechamento de pedido começa na escolha do endereço de entrega, seguida logo após pela escolha da forma de pagamento. Ao fim, o cliente verifica o resumo do pedido e confirma o mesmo.

Figura 14 – Diagrama de atividades da realização de um pedido.



Fonte: produzido pelo autor.

7 Validação e Resultados

Este capítulo apresenta a validação do guia de recomendações para design de interfaces web móveis por meio da avaliação heurística do protótipo criado.

A avaliação heurística consiste na avaliação de cada tela existente no sistema de forma individual. Caso haja relevância suficiente de uma parte isolada, a mesma será avaliada individualmente.

Então, considerando cada um dos cenários acima, avaliou-se o mesmo sob a ótica de cada uma das dez heurísticas de Nielsen, verificando se o cenário analisado está de acordo com a heurística em questão. Caso esteja, passa-se para a próxima heurística e repete-se o procedimento; caso contrário, deve-se identificar que parte ou ação no cenário em questão que possui o problema, a razão da violação da heurística, a gravidade da violação, a sugestão de uma possível uma solução para o problema e, caso necessário, algum comentário sobre o problema.

7.1 Modelo

O modelo que será utilizado na avaliação heurística consiste na seguinte estrutura: para cada cenário, iterar-se-á sobre cada uma das 10 heurísticas, perguntando:

1. Este cenário está de acordo com a heurística?
 - 1.1. Sim
 - 1.1.1. Passa-se para a próxima heurística
 - 1.2. Não
 - 1.2.1. Identifica-se o local ou ação que possui o problema
 - 1.2.2. Descreve-se por que a heurística está sendo violada, face ao perfil de usuários e tarefas pressupostos
 - 1.2.3. Julga-se a gravidade do problema:
 - 0: Não é considerado, totalmente, um problema de usabilidade;
 - 1: Problema apenas estético, não sendo necessário consertá-lo a menos caso exista tempo extra disponível no projeto;
 - 2: Problema menor de usabilidade, o conserto deste problema deverá ser baixa prioridade;
 - 3: Problema maior de usabilidade, sendo importante consertá-lo;

- 4: Catástrofe de usabilidade, sendo obrigatório consertá-lo antes do produto ser divulgado;

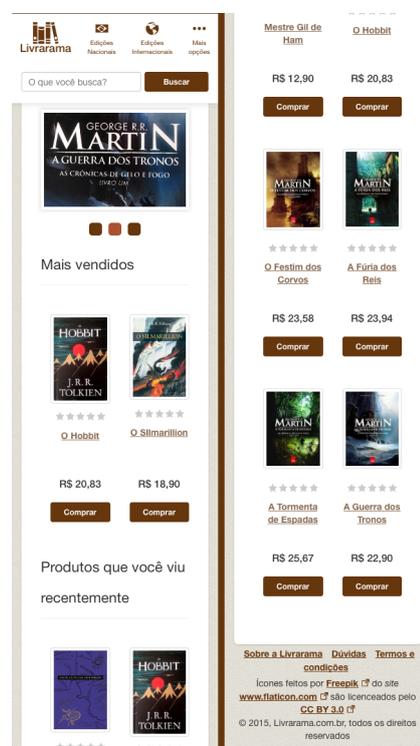
1.2.4. Se possível, sugere-se solução para o problema

1.2.5. Anota qualquer comentário adicional que julgar relevante

7.2 Avaliação dos cenários

7.2.1 Página inicial

Figura 15 – *Screenshot* da página inicial.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuário

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.2 Entrar/Criar conta

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuário

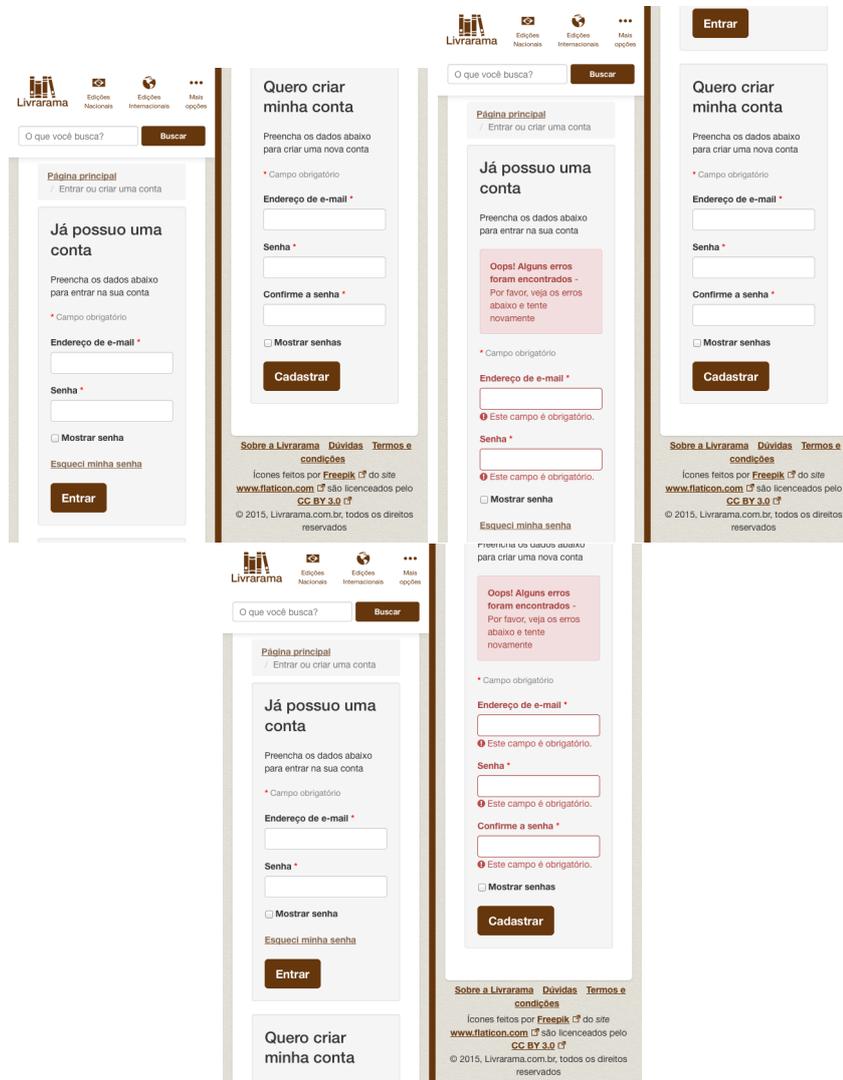
3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

Figura 16 – *Screenshot* da página de Entrar/Criar conta em três variações: sem erros de preenchimento, com erros de preenchimento no formulário Entrar e com erros de preenchimento no formulário Criar conta.



Fonte: produzido pelo autor.

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

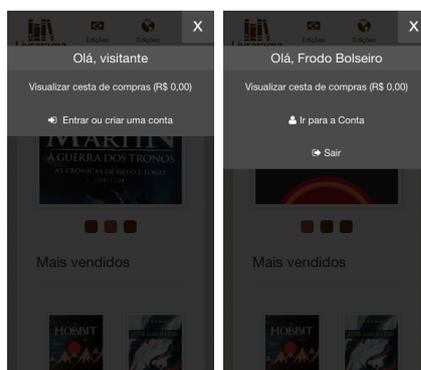
9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.3 Navegação

Figura 17 – *Screenshot* da Navegação em duas variações: como ela é apresentada caso não haja usuário autenticado e como ela é apresentada para um usuário autenticado.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuario

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.4 Busca

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuario

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

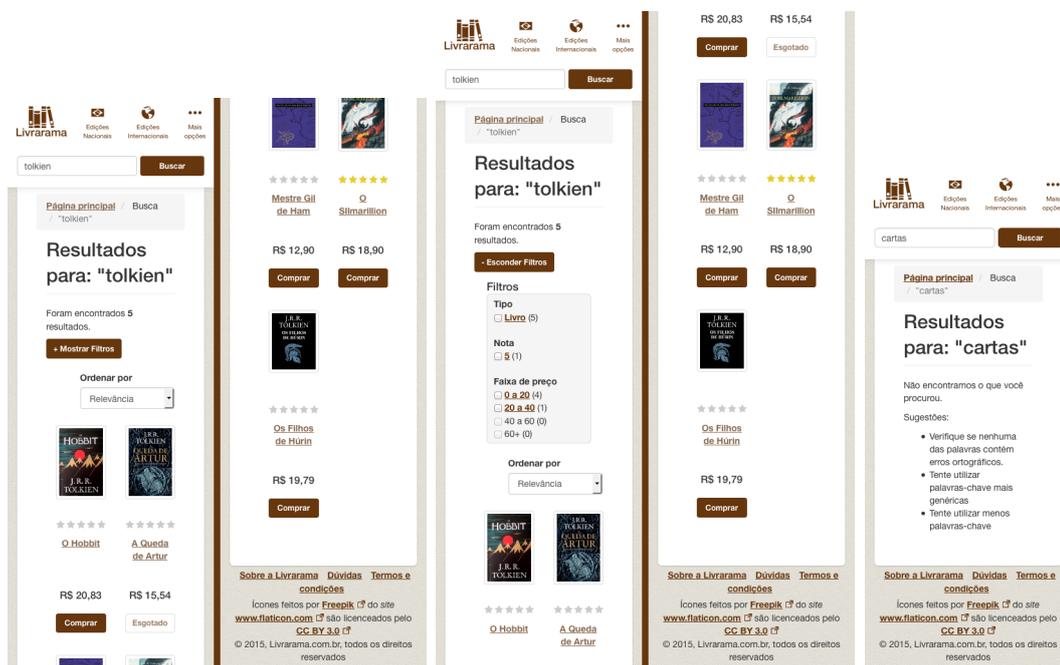
5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

Figura 18 – *Screenshot* da página de Busca em três variações: com resultados e sem a exibição dos filtros, com resultados e com a exibição dos filtros e sem resultados.



Fonte: produzido pelo autor.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Não.

9.1.1. Localização na qual o problema existe ou situação que leva o problema a acontecer: quando realiza-se uma busca e ela não retorna resultados.

9.1.2. Descrição do problema: quando realiza-se uma busca e ela não retorna resultados, apesar de se sugerir formas através das quais o usuário pode alterar a sua busca, o sistema em si não sugere alternativas às palavras-chave que o usuário usou, como correções de uma eventual grafia errada ou sinônimos das palavras pesquisadas.

9.1.3. Gravidade: 2. O problema encontrado pode gerar certo incômodo ao usuário, mas não impedirá ele de realizar a ação.

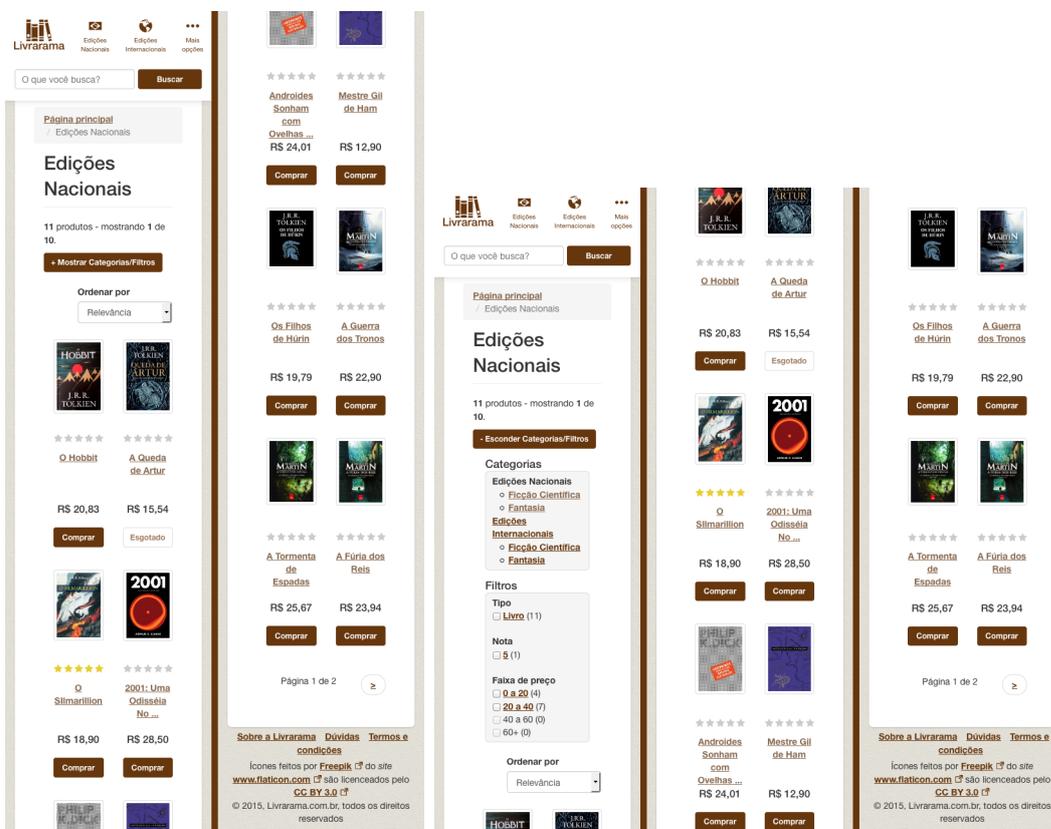
9.1.4. Possível solução: poderia-se implementar um sistema que sugerisse buscas com palavras semelhantes às buscadas para o usuário.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.5 Catálogo

Figura 19 – *Screenshot* da página do Catálogo em duas variações: sem a exibição das categorias e filtros e com a exibição das categorias e filtros.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

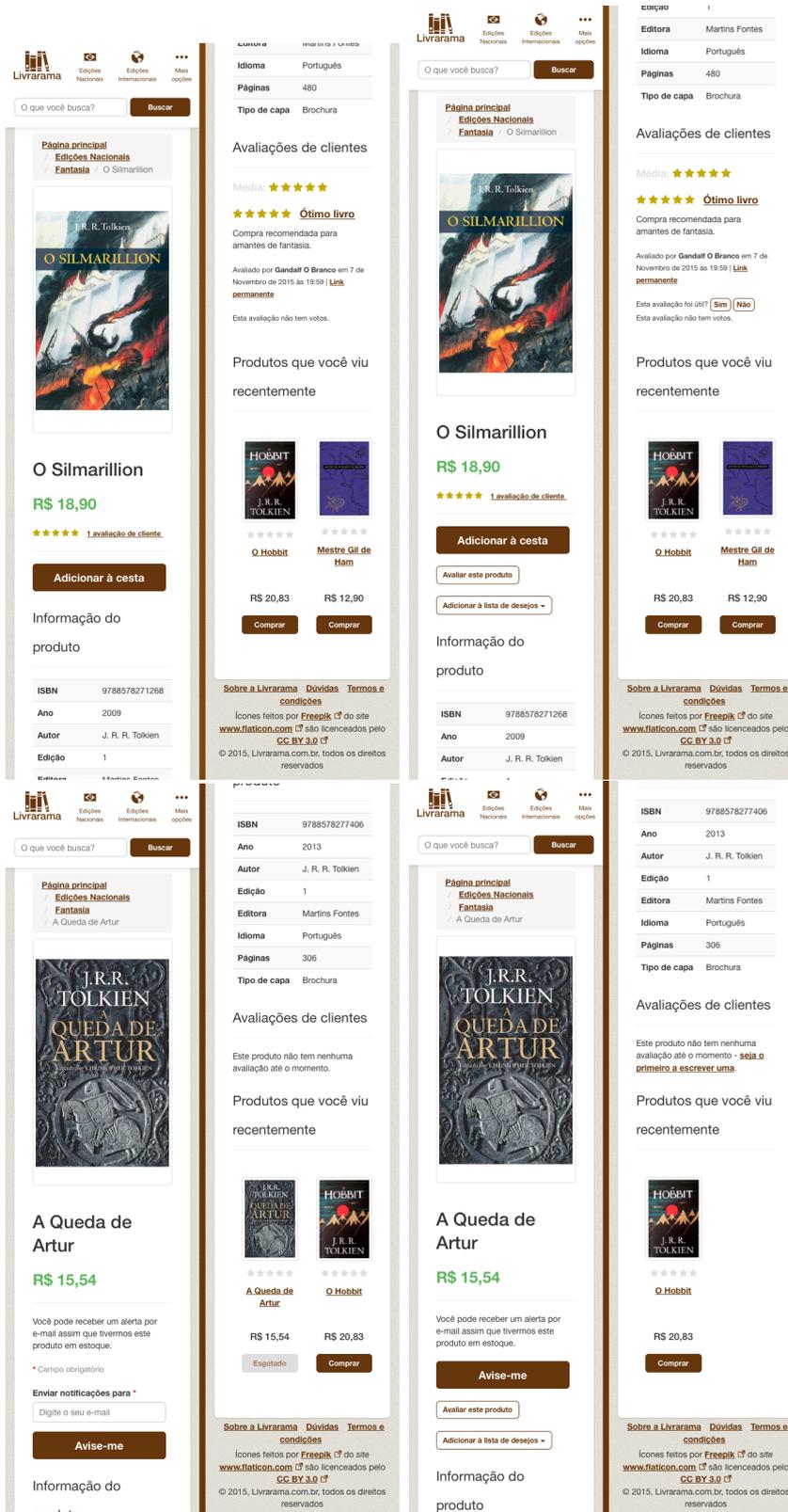
2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuário

- 3.1. Esta tela está de acordo com a heurística? Sim.
- 4. Consistência e padrões
 - 4.1. Esta tela está de acordo com a heurística? Sim.
- 5. Prevenção de erros
 - 5.1. Esta tela está de acordo com a heurística? Sim.
- 6. Reconhecimento em vez de memorização
 - 6.1. Esta tela está de acordo com a heurística? Sim.
- 7. Flexibilidade e eficiência de uso
 - 7.1. Esta tela está de acordo com a heurística? Sim.
- 8. Estética e design minimalista
 - 8.1. Esta tela está de acordo com a heurística? Sim.
- 9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros
 - 9.1. Esta tela está de acordo com a heurística? Sim.
- 10. Ajuda e documentação
 - 10.1. Esta tela está de acordo com a heurística? Sim.

7.2.6 Detalhes

Figura 20 – Screenshot da página de Detalhes em quatro variações: como ela é apresentada quando o produto está em estoque caso não haja usuário autenticado e caso haja, e como ela é apresentada quando o produto está esgotado caso não haja usuário autenticado e caso haja.



1. Visibilidade do status do sistema

- 1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

- 2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuário

- 3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

- 4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

- 5.1. Esta tela está de acordo com a heurística? Não.

- 5.1.1. Localização na qual o problema existe ou situação que leva o problema a acontecer: ao se clicar no botão “Avaliar este produto”.

- 5.1.2. Descrição do problema: quando clica-se no botão “Avaliar este produto”, a tela do usuário desliza até o formulário de preenchimento da avaliação, escondido na interface até então. Porém, o deslizamento acontece de forma não apropriada, pois o usuário acaba sendo direcionado para a metade do formulário, e não para o início do mesmo.

- 5.1.3. Gravidade: 2. O problema encontrado pode gerar certo incômodo ao usuário, mas não impedirá ele de realizar a ação. Além disso, a avaliação de produtos é uma ação que é realizada com menos frequência que outras ações, como a de compra.

- 5.1.4. Possível solução: fazer que o deslizamento da tela passe a acontecer para o início do formulário.

- 5.1.5. Comentários: o comportamento identificado parece ter ocorrido por causa da forma pela qual escolheu-se implementar o menu de navegação do site, já que o conteúdo do site parece ser jogado para baixo da mesma. Provavelmente, para resolver este problema, será necessário mudar a forma de implementação do deslizamento interno, deixando-se de usar links internos e passando-se a usar Javascript, por exemplo.

6. Reconhecimento em vez de memorização

- 6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.7 Avaliações de um produto

Figura 21 – *Screenshot* da página da listagem de avaliações de um produto.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

- 1.1. Esta tela está de acordo com a heurística? Sim.
2. Compatibilidade do sistema com o mundo real
 - 2.1. Esta tela está de acordo com a heurística? Sim.
3. Controle e liberdade do usuário
 - 3.1. Esta tela está de acordo com a heurística? Sim.
4. Consistência e padrões
 - 4.1. Esta tela está de acordo com a heurística? Sim.
5. Prevenção de erros
 - 5.1. Esta tela está de acordo com a heurística? Sim.
6. Reconhecimento em vez de memorização
 - 6.1. Esta tela está de acordo com a heurística? Sim.
7. Flexibilidade e eficiência de uso
 - 7.1. Esta tela está de acordo com a heurística? Sim.
8. Estética e design minimalista
 - 8.1. Esta tela está de acordo com a heurística? Sim.
9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros
 - 9.1. Esta tela está de acordo com a heurística? Sim.
10. Ajuda e documentação
 - 10.1. Esta tela está de acordo com a heurística? Sim.

Figura 22 – *Screenshot* da página de detalhes de uma avaliação de um produto.



Fonte: produzido pelo autor.

7.2.8 Avaliações de um produto - Avaliação

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuário

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

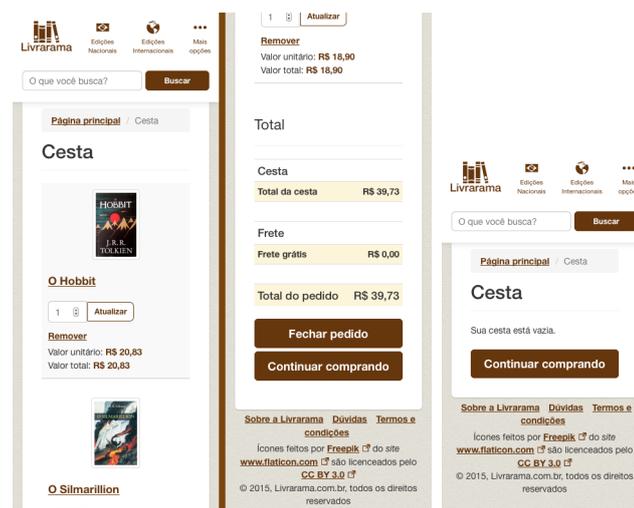
9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.9 Cesta de Compras

Figura 23 – *Screenshot* da página da Cesta de Compras em duas variações: com itens e vazia.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuario

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.10 Conta - Perfil

1. Visibilidade do status do sistema

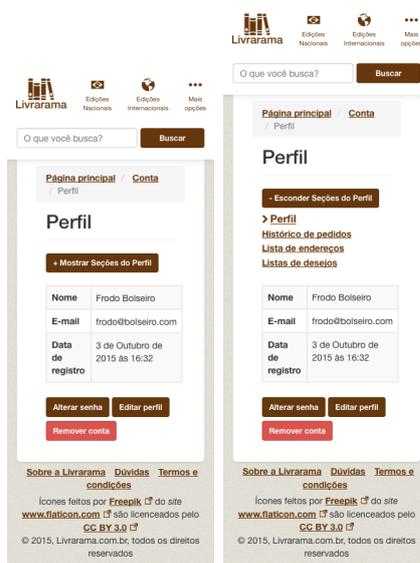
1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuario

Figura 24 – *Screenshot* da página da Conta do usuário autenticado, na seção do Perfil, em duas variações: sem a exibição das seções e com a exibição das seções.



Fonte: produzido pelo autor.

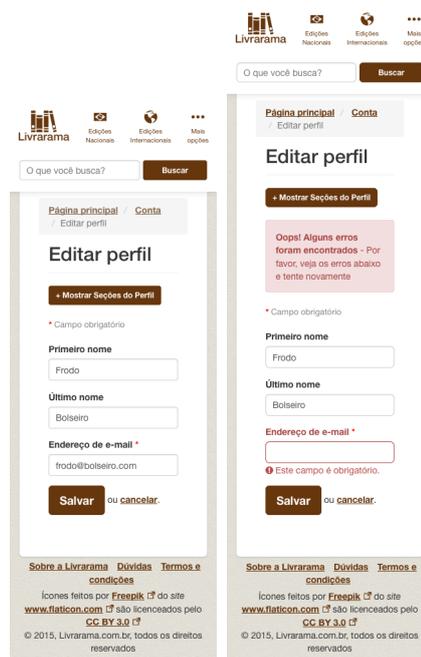
- 3.1. Esta tela está de acordo com a heurística? Sim.
4. Consistência e padrões
 - 4.1. Esta tela está de acordo com a heurística? Sim.
5. Prevenção de erros
 - 5.1. Esta tela está de acordo com a heurística? Sim.
6. Reconhecimento em vez de memorização
 - 6.1. Esta tela está de acordo com a heurística? Sim.
7. Flexibilidade e eficiência de uso
 - 7.1. Esta tela está de acordo com a heurística? Sim.
8. Estética e design minimalista
 - 8.1. Esta tela está de acordo com a heurística? Sim.
9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros
 - 9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.11 Conta - Editar Perfil

Figura 25 – *Screenshot* da página da Conta do usuário autenticado, durante a ação de edição do perfil em duas variações: sem erros de preenchimento e com erros de preenchimento.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuário

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.12 Conta - Alterar senha

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuario

3.1. Esta tela está de acordo com a heurística? Sim.

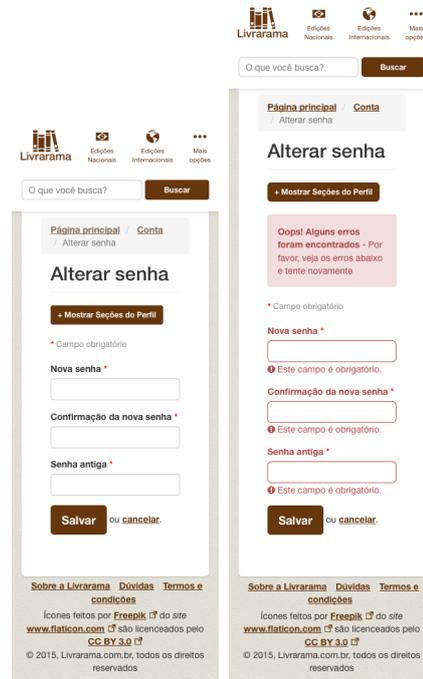
4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Não.

Figura 26 – *Screenshot* da página da Conta do usuário autenticado, durante a ação de alteração da senha em duas variações: sem erros de preenchimento e com erros de preenchimento.



Fonte: produzido pelo autor.

- 5.1.1. Localização na qual o problema existe ou situação que leva o problema a acontecer: os campos do formulário existente na página.
- 5.1.2. Descrição do problema: o formulário desta página possui três campos do tipo “password”, e a implementação natural destes campos é que o conteúdo existente nos mesmos fique oculto. Porém, existe a possibilidade de que o usuário cometa erros de digitação durante o preenchimento de um destes campos.
- 5.1.3. Gravidade: 3. Apesar da ação de alteração de troca de senha ser realizada com pouca frequência, a implementação natural dos campos do tipo do tipo “password” é de que, após o envio do formulário, o valor dos mesmos não deve ser recuperado. Logo, o usuário teria que digitar os valores dos três campos novamente, o que será uma ação frustrante para o mesmo.
- 5.1.4. Possível solução: da mesma forma do que foi implementado nos formulários de login e registro, adicionar um campo que exiba os valores destes campos.

6. Reconhecimento em vez de memorização

- 6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

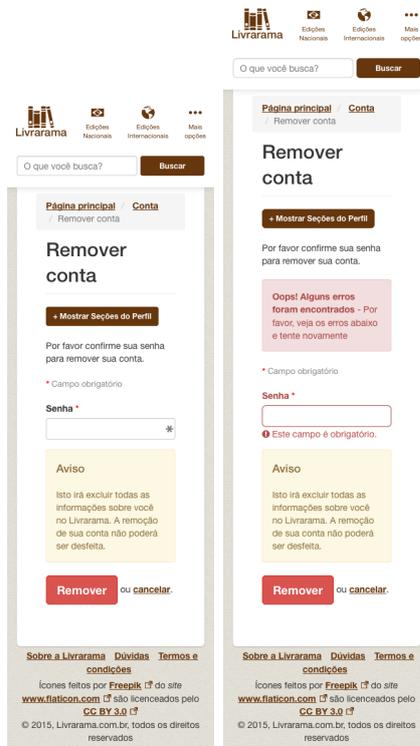
9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.13 Conta - Remover conta

Figura 27 – *Screenshot* da página da Conta do usuário autenticado, durante a ação de remoção da conta em duas variações: sem erros de preenchimento e com erros de preenchimento.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

- 1.1. Esta tela está de acordo com a heurística? Sim.
2. Compatibilidade do sistema com o mundo real
 - 2.1. Esta tela está de acordo com a heurística? Sim.
3. Controle e liberdade do usuario
 - 3.1. Esta tela está de acordo com a heurística? Sim.
4. Consistência e padrões
 - 4.1. Esta tela está de acordo com a heurística? Sim.
5. Prevenção de erros
 - 5.1. Esta tela está de acordo com a heurística? Não.
 - 5.1.1. Localização na qual o problema existe ou situação que leva o problema a acontecer: o campo do formulário existente na página.
 - 5.1.2. Descrição do problema: o formulário desta página possui um campo do tipo “*password*”, e a implementação natural deste campo é que o conteúdo existente nos mesmos fique oculto. Porém, existe a possibilidade de que o usuário cometa erros de digitação durante o preenchimento de um deste campo.
 - 5.1.3. Gravidade: 2. A ação de alteração de remoção de conta é realizada (provavelmente) apenas uma vez pelo usuário, e ele só terá que preencher novamente um campo, caso ele cometa erros.
 - 5.1.4. Possível solução: da mesma forma do que foi implementado nos formulários de login e registro, adicionar um campo que exiba o valor deste campo.
6. Reconhecimento em vez de memorização
 - 6.1. Esta tela está de acordo com a heurística? Sim.
7. Flexibilidade e eficiência de uso
 - 7.1. Esta tela está de acordo com a heurística? Sim.
8. Estética e design minimalista
 - 8.1. Esta tela está de acordo com a heurística? Sim.
9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

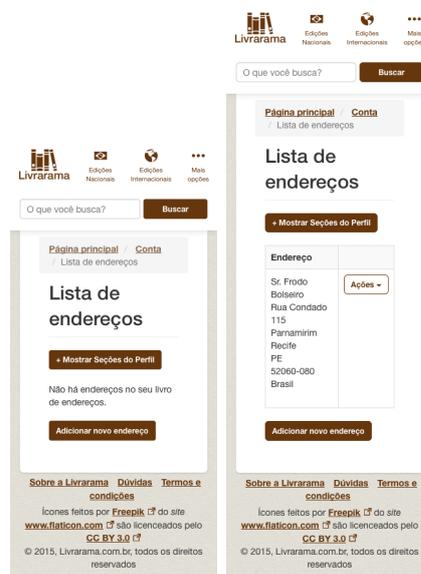
9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.14 Conta - Lista de Endereços

Figura 28 – *Screenshot* da página da Conta do usuário autenticado, na seção da Lista de Endereços, em duas variações: com endereços e vazia.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuário

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.15 Conta - Lista de Endereços - Adicionar novo endereço

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuário

3.1. Esta tela está de acordo com a heurística? Não.

3.1.1. Localização na qual o problema existe ou situação que leva o problema a acontecer: o campo de CEP no formulário.

3.1.2. Descrição do problema: quando o usuário termina de preencher o campo de CEP com seus oito dígitos, o *website* automaticamente realiza a busca do endereço equivalente ao CEP informado, de forma a preencher os campos do formulário para o usuário. Apesar de tal implementação ir a favor da heurística de Flexibilidade e Eficiência de Uso, o fato da ação de busca do CEP não ser realizada pelo usuário vai contra a presente heurística.

Figura 29 – *Screenshot* da página da Conta do usuário autenticado, na seção da Lista de Endereços, durante a ação de adição de um novo endereço, em duas variações: sem erros de preenchimento e com erros de preenchimento.

The image displays two versions of the 'Adicionar novo endereço' (Add new address) form in the Livrarama user account interface. Both versions show the form's layout, including fields for Title, Name, Surname, CEP, Address, Number, and Neighborhood, along with a 'Telefone' (Phone) field and an 'Instruções' (Instructions) text area. The left version shows the form with all fields filled correctly. The right version shows the form with red error messages indicating that several fields are mandatory but empty.

Fonte: produzido pelo autor.

3.1.3. Gravidade: 3. Apesar da ação de adição de um endereço não ser algo que o usuário realizará com frequência, o fato dos campos serem preenchidos de forma automática pode causar surpresa e até um certo aborrecimento, caso o usuário já tenha preenchido o restante dos campos antes de preencher o campo de CEP.

3.1.4. Possível solução: colocar ao lado do campo de CEP um botão, identificado como “Buscar CEP”, para que o usuário possa realizar a ação de forma explícita e consciente.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.16 Conta - Lista de Endereços - Remover endereço

Figura 30 – *Screenshot* da página da Conta do usuário autenticado, na seção da Lista de Endereços, durante a ação de remoção de um endereço.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuario

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.17 Conta - Listas de Desejos

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

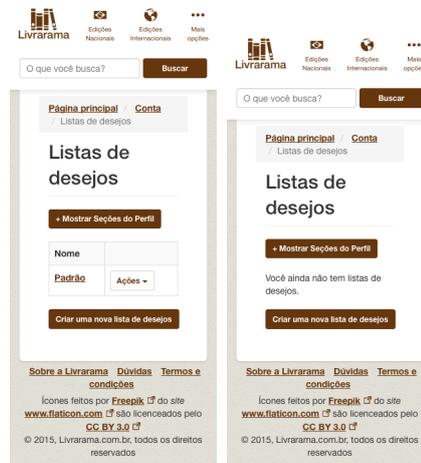
2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuario

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

Figura 31 – *Screenshot* da página da Conta do usuário autenticado, na seção das Listas de Desejos, em duas variações: com itens e vazia.



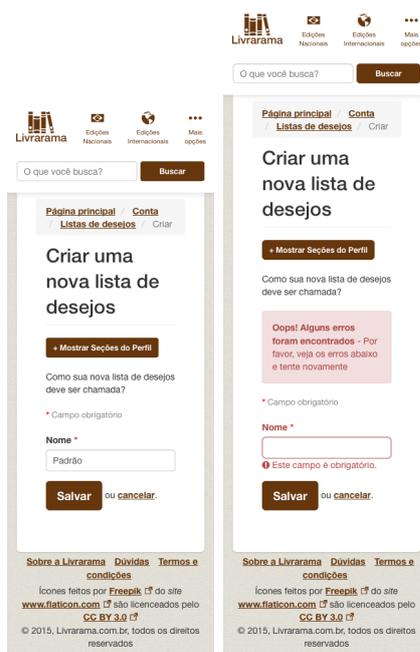
Fonte: produzido pelo autor.

- 4.1. Esta tela está de acordo com a heurística? Sim.
5. Prevenção de erros
 - 5.1. Esta tela está de acordo com a heurística? Sim.
6. Reconhecimento em vez de memorização
 - 6.1. Esta tela está de acordo com a heurística? Sim.
7. Flexibilidade e eficiência de uso
 - 7.1. Esta tela está de acordo com a heurística? Sim.
8. Estética e design minimalista
 - 8.1. Esta tela está de acordo com a heurística? Sim.
9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros
 - 9.1. Esta tela está de acordo com a heurística? Sim.
10. Ajuda e documentação
 - 10.1. Esta tela está de acordo com a heurística? Sim.

7.2.18 Conta - Listas de Desejos - Criar uma nova lista de desejos

1. Visibilidade do status do sistema
 - 1.1. Esta tela está de acordo com a heurística? Sim.
2. Compatibilidade do sistema com o mundo real
 - 2.1. Esta tela está de acordo com a heurística? Sim.
3. Controle e liberdade do usuário
 - 3.1. Esta tela está de acordo com a heurística? Sim.
4. Consistência e padrões
 - 4.1. Esta tela está de acordo com a heurística? Sim.
5. Prevenção de erros
 - 5.1. Esta tela está de acordo com a heurística? Sim.
6. Reconhecimento em vez de memorização

Figura 32 – *Screenshot* da página da Conta do usuário autenticado, na seção das Listas de Desejos, durante a ação de criação de uma nova lista de desejos, em duas variações: sem erros de preenchimento e com erros de preenchimento.



Fonte: produzido pelo autor.

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

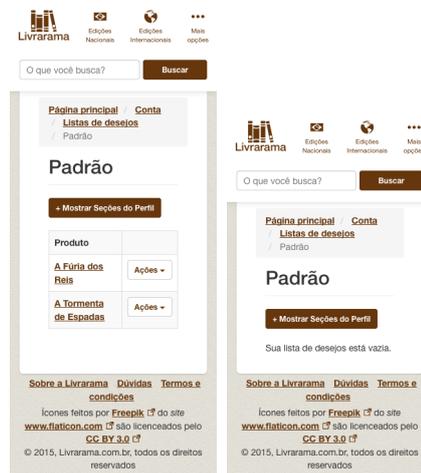
9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.19 Conta - Listas de Desejos - Lista de desejos

Figura 33 – *Screenshot* da página da Conta do usuário autenticado, na seção das Listas de Desejos, quando exibindo os detalhes de uma lista de desejos em específico, em duas variações: com itens e vazia.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuario

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.20 Conta - Listas de Desejos - Remover item de uma lista

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuario

3.1. Esta tela está de acordo com a heurística? Sim.

Figura 34 – *Screenshot* da página da Conta do usuário autenticado, na seção das Listas de Desejos, durante a ação de remoção de um item de uma lista de desejos.



Fonte: produzido pelo autor.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.21 Conta - Listas de Desejos - Remover lista

1. Visibilidade do status do sistema
 - 1.1. Esta tela está de acordo com a heurística? Sim.
2. Compatibilidade do sistema com o mundo real
 - 2.1. Esta tela está de acordo com a heurística? Sim.
3. Controle e liberdade do usuário
 - 3.1. Esta tela está de acordo com a heurística? Sim.
4. Consistência e padrões
 - 4.1. Esta tela está de acordo com a heurística? Sim.
5. Prevenção de erros
 - 5.1. Esta tela está de acordo com a heurística? Sim.
6. Reconhecimento em vez de memorização
 - 6.1. Esta tela está de acordo com a heurística? Sim.
7. Flexibilidade e eficiência de uso

Figura 35 – *Screenshot* da página da Conta do usuário autenticado, na seção das Listas de Desejos, durante a ação de remoção de uma lista de desejos.



Fonte: produzido pelo autor.

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

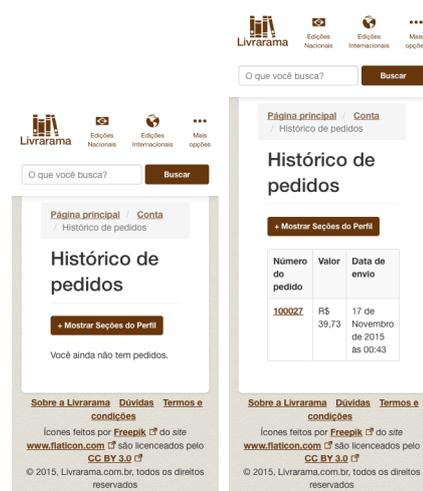
9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.22 Conta - Histórico de pedidos

Figura 36 – *Screenshot* da página da Conta do usuário autenticado, na seção do Histórico de Pedidos, em duas variações: sem pedidos e com pedidos.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuário

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.23 Conta - Histórico de pedidos - Pedido

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

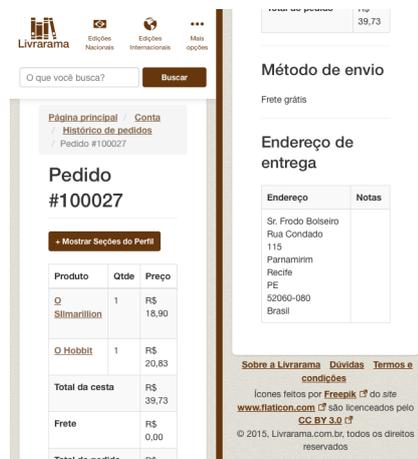
3. Controle e liberdade do usuário

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Não.

Figura 37 – *Screenshot* da página da Conta do usuário autenticado, na seção do Histórico de Pedidos, quando exibindo os detalhes de um pedido.



Fonte: produzido pelo autor.

- 4.1.1. Localização na qual o problema existe ou situação que leva o problema a acontecer: o cabeçalho da tabela que mostra o resumo dos valores do pedido.
 - 4.1.2. Descrição do problema: ao contrário das demais tabelas existentes no *website*, o cabeçalho desta tabela não está num tom acinzentado.
 - 4.1.3. Gravidade: 1. O problema identificado é apenas estético e não dificultará o entendimento do usuário.
 - 4.1.4. Possível solução: colorir o cabeçalho com a cor acinzentado usado no cabeçalho das demais tabelas.
5. Prevenção de erros
 - 5.1. Esta tela está de acordo com a heurística? Sim.
 6. Reconhecimento em vez de memorização
 - 6.1. Esta tela está de acordo com a heurística? Sim.
 7. Flexibilidade e eficiência de uso
 - 7.1. Esta tela está de acordo com a heurística? Sim.
 8. Estética e design minimalista
 - 8.1. Esta tela está de acordo com a heurística? Sim.
 9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

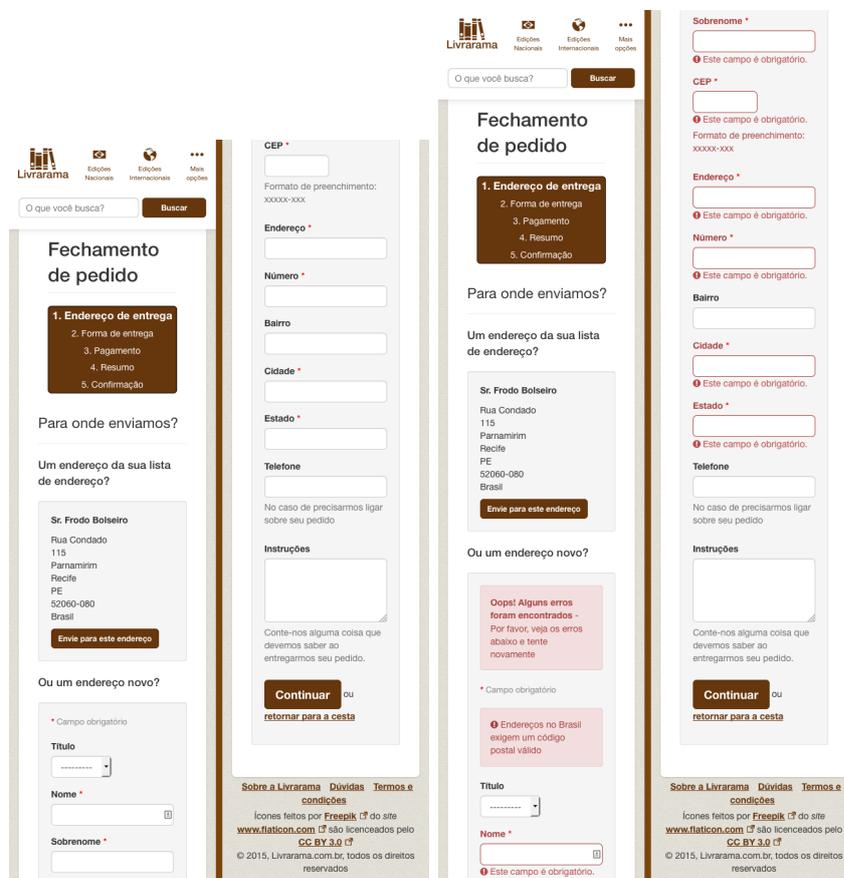
9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.24 Fechamento de pedido - Seleção de endereço de entrega

Figura 38 – *Screenshot* da página de fechamento de pedido, durante o passo de seleção de endereço de entrega, em duas variações: sem erros de preenchimento e com erros de preenchimento.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuario

3.1. Esta tela está de acordo com a heurística? Não.

3.1.1. Localização na qual o problema existe ou situação que leva o problema a acontecer: o campo de CEP no formulário.

3.1.2. Descrição do problema: quando o usuário termina de preencher o campo de CEP com seus oito dígitos, o *website* automaticamente realiza a busca do endereço equivalente ao CEP informado, de forma a preencher os campos do formulário para o usuário. Apesar de tal implementação ir a favor da heurística de Flexibilidade e Eficiência de Uso, o fato da ação de busca do CEP não ser realizada pelo usuário vai contra a presente heurística.

3.1.3. Gravidade: 3. Apesar da ação de adição de um endereço não ser algo que o usuário realizará com frequência, o fato dos campos serem preenchidos de forma automática pode causar surpresa e até um certo aborrecimento, caso o usuário já tenha preenchido o restante dos campos antes de preencher o campo de CEP.

3.1.4. Possível solução: colocar ao lado do campo de CEP um botão, identificado como “Buscar CEP”, para que o usuário possa realizar a ação de forma explícita e consciente.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

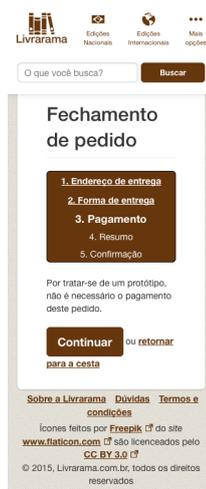
9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.25 Fechamento de pedido - Seleção da forma de pagamento

Figura 39 – *Screenshot* da página de fechamento de pedido, durante o passo de seleção da forma de pagamento.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

2. Compatibilidade do sistema com o mundo real

2.1. Esta tela está de acordo com a heurística? Sim.

3. Controle e liberdade do usuário

3.1. Esta tela está de acordo com a heurística? Sim.

4. Consistência e padrões

4.1. Esta tela está de acordo com a heurística? Sim.

5. Prevenção de erros

5.1. Esta tela está de acordo com a heurística? Sim.

6. Reconhecimento em vez de memorização

6.1. Esta tela está de acordo com a heurística? Sim.

7. Flexibilidade e eficiência de uso

7.1. Esta tela está de acordo com a heurística? Sim.

8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

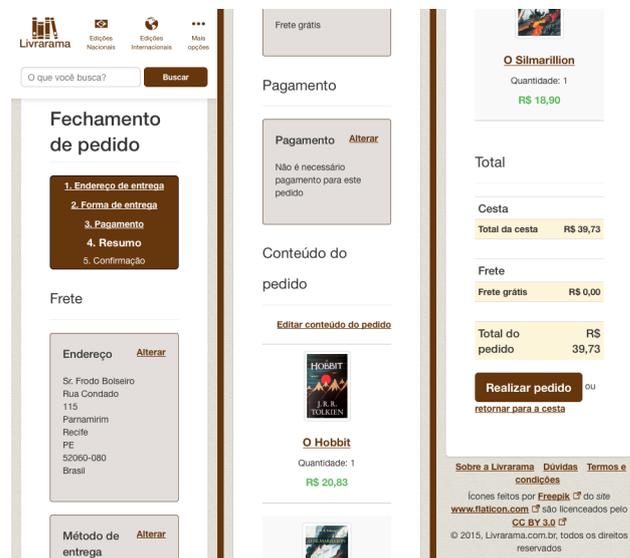
9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

7.2.26 Fechamento de pedido - Resumo do pedido

Figura 40 – *Screenshot* da página de fechamento de pedido, durante o passo da exibição do resumo do pedido.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema

1.1. Esta tela está de acordo com a heurística? Sim.

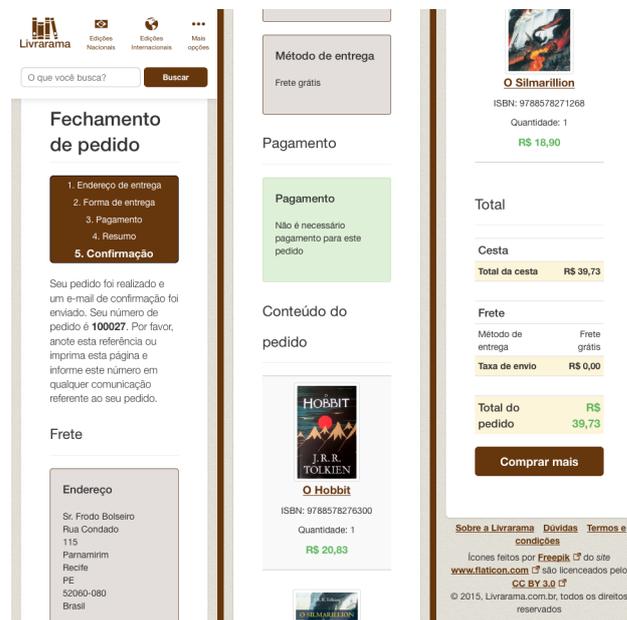
2. Compatibilidade do sistema com o mundo real

- 2.1. Esta tela está de acordo com a heurística? Sim.
- 3. Controle e liberdade do usuario
 - 3.1. Esta tela está de acordo com a heurística? Sim.
- 4. Consistência e padrões
 - 4.1. Esta tela está de acordo com a heurística? Sim.
- 5. Prevenção de erros
 - 5.1. Esta tela está de acordo com a heurística? Sim.
- 6. Reconhecimento em vez de memorização
 - 6.1. Esta tela está de acordo com a heurística? Sim.
- 7. Flexibilidade e eficiência de uso
 - 7.1. Esta tela está de acordo com a heurística? Sim.
- 8. Estética e design minimalista
 - 8.1. Esta tela está de acordo com a heurística? Sim.
- 9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros
 - 9.1. Esta tela está de acordo com a heurística? Sim.
- 10. Ajuda e documentação
 - 10.1. Esta tela está de acordo com a heurística? Sim.

7.2.27 Fechamento de pedido - Confirmação do pedido

- 1. Visibilidade do status do sistema
 - 1.1. Esta tela está de acordo com a heurística? Sim.
- 2. Compatibilidade do sistema com o mundo real
 - 2.1. Esta tela está de acordo com a heurística? Sim.
- 3. Controle e liberdade do usuario
 - 3.1. Esta tela está de acordo com a heurística? Sim.

Figura 41 – *Screenshot* da página de fechamento de pedido, durante o passo da confirmação do sucesso na realização do pedido.



Fonte: produzido pelo autor.

4. Consistência e padrões
 - 4.1. Esta tela está de acordo com a heurística? Sim.
5. Prevenção de erros
 - 5.1. Esta tela está de acordo com a heurística? Sim.
6. Reconhecimento em vez de memorização
 - 6.1. Esta tela está de acordo com a heurística? Sim.
7. Flexibilidade e eficiência de uso
 - 7.1. Esta tela está de acordo com a heurística? Sim.
8. Estética e design minimalista
 - 8.1. Esta tela está de acordo com a heurística? Sim.
9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros
 - 9.1. Esta tela está de acordo com a heurística? Sim.
10. Ajuda e documentação
 - 10.1. Esta tela está de acordo com a heurística? Sim.

7.2.28 Erro 404 (Página não encontrada)

Figura 42 – *Screenshot* da página de erro 404.



Fonte: produzido pelo autor.

1. Visibilidade do status do sistema
 - 1.1. Esta tela está de acordo com a heurística? Sim.
2. Compatibilidade do sistema com o mundo real
 - 2.1. Esta tela está de acordo com a heurística? Sim.
3. Controle e liberdade do usuário
 - 3.1. Esta tela está de acordo com a heurística? Sim.
4. Consistência e padrões
 - 4.1. Esta tela está de acordo com a heurística? Sim.
5. Prevenção de erros
 - 5.1. Esta tela está de acordo com a heurística? Sim.
6. Reconhecimento em vez de memorização
 - 6.1. Esta tela está de acordo com a heurística? Sim.
7. Flexibilidade e eficiência de uso
 - 7.1. Esta tela está de acordo com a heurística? Sim.
8. Estética e design minimalista

8.1. Esta tela está de acordo com a heurística? Sim.

9. Ajudar os usuários a reconhecer, diagnosticar e se recuperar de erros

9.1. Esta tela está de acordo com a heurística? Sim.

10. Ajuda e documentação

10.1. Esta tela está de acordo com a heurística? Sim.

A avaliação acima, portanto, analisou 28 cenários sob a ótica de 10 heurísticas e, após tal análise, encontraram-se apenas 7 problemas de usabilidade, sendo que nenhum deles foi considerado de gravidade 4 (três problemas de gravidade 3, três problemas de gravidade 2 e 1 problema de gravidade 1).

Identificou-se que poucas heurísticas foram desrespeitadas, já que o desenvolvimento do protótipo foi prumado pelo cumprimento das mesmas. Eventuais problemas de interface que foram identificados durante a fase de prototipação, que foi alvo de vários refinamentos, foram corrigidos antes da avaliação final.

Diante da avaliação acima, conclui-se que as heurísticas foram atendidas em sua maioria, e que o guia teve sucesso em realizar o que ele se propôs.

8 Considerações Finais

Já é numericamente evidente que o êxodo dos usuários de dispositivos *desktop* para dispositivos móveis está acontecendo, e que deve-se dar uma maior importância para o desenvolvimento de *websites* para estes dispositivos com características tão particulares.

Diversas diretrizes foram criadas tendo como foco o desenvolvimento de interfaces gráficas num geral e, em alguns casos, interfaces *desktop*, mas o estudo e desenvolvimento de diretrizes específicas para dispositivos móveis é algo que ainda pode ser muito explorado.

Em vista disso, com base nos fundamentos estudados foi desenvolvida uma proposta de um guia de recomendações para *design* de interfaces *web* móveis. A fim de se verificar sua efetividade, desenvolveu-se um protótipo de um *e-commerce* e, em um segundo momento, realizou-se a avaliação heurística deste protótipo. Após a realização da avaliação heurística, verificou-se a efetividade do guia criado para o fim ao qual ele se propôs.

O guia desenvolvido consegue ajudar o desenvolvedor durante o processo de desenvolvimento, para que o mesmo produza um *website* o mais apropriado possível para um dispositivo móvel e pontuando algumas características que devem ser observadas ao se produzir para este tipo de dispositivo.

Com o desenvolvimento deste protótipo, constatou-se que seguir o que a filosofia *Mobile First* diz, ou seja, começar a desenvolver um *website* pelos dispositivos móveis, é vantajoso para o processo de desenvolvimento de um *website* de forma geral, e não apenas para dispositivos móveis. Ao se usar o protótipo criado em um dispositivo *desktop*, verificou-se que, salvos alguns aspectos estéticos, o *website* encontra-se totalmente funcional e possui relativo apelo visual.

O referencial teórico utilizado no desenvolvimento do guia, proveniente de fontes com notoriedade na área de IHC e fruto de diversas revisões, foi de suma importância para que o guia cumprisse seu objetivo.

Considera-se que este trabalho contribuiu com a criação de um guia de recomendações específico para o desenvolvimento web mobile, algo que era inédito até então. Este guia pode ser usado por desenvolvedores para orientá-los em suas decisões estéticas e funcionais ao se desenvolver um *website* voltado para dispositivos móveis.

Também, cabe mencionar que deseja-se disponibilizar o guia de forma gratuita e aberta, o que facilitará o alcance da informação.

Como trabalhos futuros, acredita-se que seria promissora a organização das boas práticas por estágio de desenvolvimento, de forma que o desenvolvedor consiga ter maior foco na aplicação das mesmas.

Referências

- BASTIEN, J. M. C.; SCAPIN, D. L. Ergonomic criteria for the evaluation of human-computer interfaces. RT-0156, 1993. Disponível em: <<https://hal.inria.fr/inria-00070012/en/>>. Acesso em: 10 mai 2015. Citado 2 vezes nas páginas 39 e 44.
- CISCO. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2014 – 2019. 2015. Disponível em: <http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white_paper_c11-520862.pdf>. Acesso em: 22 mai 2015. Citado na página 66.
- COMSCORE. The U.S. Mobile App Report. 2014. Disponível em: <<http://www.comscore.com/Insights/Presentations-and-Whitepapers/2014/The-US-Mobile-App-Report>>. Acesso em: 22 mar 2015. Citado na página 21.
- COMSCORE. U.S. Digital Future in Focus 2015. 2015. Disponível em: <<http://www.comscore.com/Insights/Presentations-and-Whitepapers/2015/2015-US-Digital-Future-in-Focus>>. Acesso em: 21 mai 2015. Citado na página 65.
- COOPER, A.; REIMANN, R.; CRONIN, D. About Face 3: The Essentials of Interaction Design. 1. ed. Indianápolis: Wiley Publishing, 1993. Citado 2 vezes nas páginas 35 e 37.
- DIX, A. et al. Human-Computer Interaction. England: Pearson Education Limited, 2004. 834 p. ISBN 978-0-13-046109-4. Citado na página 62.
- EVANS, P.; THOMAS, M. A. Exploring The Elements Of Design. 3. ed. Clifton Park: Delmar Cengage Learning, 2013. Citado 3 vezes nas páginas 23, 25 e 26.
- GRUDIN, J. The computer reaches out: The historical continuity of interface design. Proceedings of the ACM CHI'90, p. 261–268, 1990. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.128.6801&rep=rep1&type=pdf>>. Acesso em: 16 abr 2015. Citado na página 36.
- HEALTH, U. S. D. of; SERVICES, H. The Research-Based Web Design & Usability Guidelines. Washington: U. S. Government Printing Office, 2006. Disponível em: <http://www.usability.gov/sites/default/files/documents/guidelines_book.pdf>. Acesso em: 06 jun 2015. Citado 2 vezes nas páginas 55 e 73.
- HEALTH, U. S. D. of; SERVICES, H. Usability Evaluation Basics. Washington: U. S. Government Printing Office, 2015. Disponível em: <<http://www.usability.gov/what-and-why/usability-evaluation.html>>. Acesso em: 16 jun 2015. Citado na página 51.
- ITU. Measuring the Information Society Report. Geneva: International Telecommunication Union, 2014. ISBN 978-92-61-14661-0. Disponível em: <<http://www.itu.int/en/ITU-D/Statistics/Pages/publications/mis2014.aspx>>. Acesso em: 18 mar. 2015. Citado na página 21.
- LAUER, D. A.; PENTAK, S. Design Basics. 8. ed. Boston: Clark Baxter, 2012. Citado na página 23.

MORVILLE, P.; ROSENFELD, L. Information Architecture for the World Wide Web. 3. ed. Sebastopol: O'Reilly Media, Inc., 2007. Citado 4 vezes nas páginas 27, 28, 30 e 31.

NIELSEN, J. Usability Engineering. 1. ed. San Diego: Academic Press, 1993. ISBN 0-12-528406-9. Citado 3 vezes nas páginas 42, 51 e 52.

NIELSEN, J. 10 Usability Heuristics for User Interface Design. [S.l.], 1995. Disponível em: <<http://www.nngroup.com/articles/ten-usability-heuristics/>>. Acesso em: 20 abr 2015. Citado na página 53.

NORMAN, D. The Design Of Everyday Things. Indianápolis: Basic Books, 2013. Citado 3 vezes nas páginas 23, 24 e 25.

PREECE, J.; ROGERS, Y.; SHARP, H. Design de Interação: Além da interação homem-computador. São Paulo: Bookman Companhia Editora, 2005. Citado 4 vezes nas páginas 23, 24, 25 e 53.

PSOMAS, S. The Five Competencies of User Experience Design. [S.l.], 2007. Disponível em: <<http://www.uxmatters.com/mt/archives/2007/11/the-five-competencies-of-user-experience-design.php>>. Acesso em: 19 abr 2015. Citado na página 23.

VIRZI, R. A.; SOKOLOV, J. L.; KARIS, D. Usability problem identification using both low- and high-fidelity prototypes. Proceedings of the ACM CHI'96, p. 236–243, 1996. Disponível em: <<https://faculty.washington.edu/socha/css572winter2012/usability%20problem%20identification%20p236-virzi.pdf>>. Acesso em: 25 mai 2015. Citado na página 43.

W3C. Web Content Accessibility Guidelines (WCAG) 2.0. 2008. Disponível em: <<http://www.w3.org/TR/WCAG20/>>. Acesso em: 02 jun 2015. Citado 2 vezes nas páginas 46 e 73.

W3C. Techniques for WCAG 2.0. 2015. Disponível em: <<http://www.w3.org/TR/WCAG20-TECHS/>>. Acesso em: 02 jun 2015. Citado na página 47.

WALKER, M.; TAKAYAMA, L.; LANDAY, J. A. High-fidelity or low-fidelity, paper or computer? choosing attributes when testing web prototypes. Proceedings of the Human Factors and Ergonomics Society Annual Meeting, v. 46, 2002. Disponível em: <http://dub.washington.edu:2007/projects/fidelity/pubs/Walker_HFES_2002.pdf>. Acesso em: 25 mai 2015. Citado na página 43.

WROBLEWSKI, L. Mobile First. New York: A Book Apart, 2011. 130 p. ISBN 978-1-937557-02-7. Citado 4 vezes nas páginas 21, 67, 68 e 70.