

---

Usabilidade da interface de dispositivos móveis:  
heurísticas e diretrizes para o design

*Olibário José Machado Neto*

---



SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: \_\_\_\_\_

# Usabilidade da interface de dispositivos móveis: heurísticas e diretrizes para o design

**Olibário José Machado Neto**

***Orientadora:* Profa. Dra. Maria da Graça Campos Pimentel**

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências - Ciências de Computação e Matemática Computacional. *EXEMPLAR DE DEFESA*

**USP – São Carlos  
Abril de 2013**

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi  
e Seção Técnica de Informática, ICMC/USP,  
com os dados fornecidos pelo(a) autor(a)

JM113u José Machado Neto, Olibário  
u Usabilidade da interface de dispositivos móveis:  
heurísticas e diretrizes para o design / Olibário  
José Machado Neto; orientadora Maria da Graça Campos  
Pimentel. -- São Carlos, 2013.  
118 p.

Dissertação (Mestrado - Programa de Pós-Graduação  
em Ciências de Computação e Matemática  
Computacional) -- Instituto de Ciências Matemáticas  
e de Computação, Universidade de São Paulo, 2013.

1. Avaliação heurística. 2. Usabilidade de  
interfaces. 3. Computação móvel. I. da Graça Campos  
Pimentel, Maria, orient. II. Título.

# Agradecimentos

---

---

A Deus pelo Seu incomensurável amor por mim, provado na cruz em que Seu filho santo morreu a minha morte para que eu, imerecidamente, vivesse a vida Dele.

Aos meus pais Marta e Sebastião, por terem sido tão presentes, mesmo estando tão distantes fisicamente. Minhas oportunidades e conquistas são frutos dos sonhos que vocês sonham comigo.

Aos meus avós, pelo amor e pelas constantes orações.

Agradeço à minha orientadora Maria da Graça C. Pimentel pelas ótimas ideias, pela ajuda para preparar a Metodologia de Pesquisa deste trabalho, pelas oportunidades de ministração de minicursos no Instituto e por outros desafios impostos a mim. Sua experiência em pesquisa e orientação me deram uma segurança ímpar para a realização deste mestrado.

Ao CnPQ, pelo aporte financeiro.

Aos meus amigos “aparterrâneos”, obrigado por terem sido uma ótima família.

Aos meus tios Val e Maria, por sempre me receberem de braços abertos em casa e por terem me ajudado a me manter forte mesmo estando longe de tantos que amo.

Agradeço aos meus amigos da Primeira Igreja Batista de São Carlos, aos das equipes de voleibol da USP e de São Carlos e aos do Laboratório de Pesquisa, pelo companheirismo.



# Resumo

---

---

Do ponto de vista do usuário, a interface é uma das partes mais importantes dos sistemas computacionais, porque por meio dela o usuário vê, ouve e sente. Essa relevância motiva pesquisadores da área de Interação Humano-Computador a estudarem maneiras de se criarem interfaces com *design* focado em usabilidade. A avaliação da usabilidade de interfaces visa verificar se elas atendem aos requisitos do usuário de forma que as funcionalidades do sistema sejam realizadas de modo efetivo, eficiente e que satisfaça as expectativas do usuário. Tendo em vista que o ciclo de desenvolvimento de *software* costuma ser longo, avaliações da usabilidade de diferentes versões de interfaces devem ser realizadas no decorrer do processo, como forma de minimizar erros e reduzir custos de produção do sistema. Uma das avaliações de usabilidade mais conhecidas é a avaliação heurística, criada por Jacob Nielsen, que se destaca pelo baixo custo e rapidez de aplicação. Nela, especialistas avaliam as interfaces e os diálogos do sistema com base em um conjunto de regras gerais, as heurísticas, que lhes permitem identificar problemas de usabilidade. Apesar de respeitadas e amplamente usadas, as heurísticas de Nielsen foram criadas sem foco em interfaces de dispositivos móveis, muito difundidos atualmente. Por meio deste trabalho, verificou-se que as heurísticas de Nielsen têm limitações para encontrarem problemas de usabilidade em interfaces de dispositivos móveis. Por conta disso, propôs-se um conjunto de heurísticas para avaliação de interfaces de dispositivos móveis e se definiu um conjunto de diretrizes para o desenvolvimento dessas interfaces. A validação das heurísticas propostas indicou que elas foram mais efetivas que as de Nielsen para encontrarem problemas de usabilidade considerados pelos especialistas como catastróficos ou de baixa gravidade.

**Palavras-chaves:** Avaliação heurística, *Design* de Interfaces, Avaliação de Usabilidade de Interfaces de Dispositivos móveis.



# Abstract

---

---

From the user's point of view, the interface is one the most important part of computer systems, because everything he sees, hears and feels are contained therein. This relevance motivates researchers of Human-Computer Interaction to study ways to create interfaces with design focused on usability. The usability evaluation of interfaces aims to determine whether the interfaces meet the requirements of the system so that its functionalities are carried out effectively, efficiently and satisfying the user's expectations. Considering that the software development lifecycle is often long, usability evaluations of different versions of interfaces should be made during the process, in order to minimize errors and reduce production costs of the final system. Heuristic evaluation, created by Jacob Nielsen, is one of the most used usability evaluation methods, because of it low cost and ease of implementation. In this evaluation method, experts evaluate interfaces and system's dialogues based on a set of general rules, called heuristics, which enable them to identify usability problems. Although respected and widely used, Nielsen's heuristics were not created having mobile devices' interfaces in mind. Through this work, it was verified that Nielsen's heuristics have limitations in finding usability problems in mobile devices' interfaces. Because of this, we proposed a set of heuristics for evaluating interfaces for mobile devices and defined a set of guidelines for the development of these interfaces. The validation of the proposed heuristics indicated that they were more effective than Nielsen's to find usability problems considered by experts as catastrophic or of low gravity.

**Keywords:** Heuristic Evaluation, Interface Design, Usability Evaluation of Mobile Devices' Interfaces.



# Sumário

---

---

Resumo . . . . .	vii
Abstract . . . . .	ix
Sumário . . . . .	xii
Lista de Figuras . . . . .	xiv
Lista de Tabelas . . . . .	xvi
Lista de Quadros . . . . .	xvii
<b>1 Introdução</b>	<b>1</b>
1.1 Contextualização . . . . .	1
1.2 Identificação do problema . . . . .	5
1.3 Questão de pesquisa . . . . .	7
1.4 Objetivos . . . . .	7
1.5 Resultados . . . . .	7
1.6 Estrutura do trabalho . . . . .	8
<b>2 Recursos para construção de interfaces com usabilidade</b>	<b>9</b>
2.1 Diretrizes ( <i>Guidelines</i> ) . . . . .	9
2.1.1 Diretrizes para criação de interfaces de dispositivos computacionais em geral	10
2.1.2 Diretrizes para criação de interfaces de dispositivos móveis . . . . .	13
2.2 Heurísticas . . . . .	18
2.3 Outros princípios de <i>design</i> . . . . .	24
2.4 Considerações finais . . . . .	28
<b>3 Metodologia de Pesquisa</b>	<b>29</b>
3.1 Elaboração das heurísticas . . . . .	31
3.1.1 Análise de aplicativos para Android . . . . .	31
3.1.2 Acerca dos aplicativos inspecionados . . . . .	32
3.1.3 <i>Brainstorming</i> com especialistas . . . . .	33
3.2 Validação das heurísticas . . . . .	33
3.2.1 Avaliações heurísticas . . . . .	33
3.2.2 Testes com usuários . . . . .	37
3.2.3 Avaliações em outros dispositivos móveis . . . . .	38
3.3 Elaboração das <i>guidelines</i> (diretrizes) . . . . .	38
3.4 Considerações finais . . . . .	40

<b>4 Resultados obtidos</b>	<b>41</b>
4.1 Proposição das heurísticas . . . . .	41
4.1.1 Análise de aplicativos para Android . . . . .	41
4.1.2 <i>Brainstorming</i> com especialistas . . . . .	48
4.2 Validação das heurísticas . . . . .	50
4.2.1 Avaliações heurísticas do aplicativo de anotações multimídia . . . . .	50
4.2.2 Avaliação com usuários finais . . . . .	65
4.2.3 Avaliação heurística do UOL Notícias . . . . .	73
4.3 Criação das diretrizes . . . . .	75
4.4 Considerações finais . . . . .	89
<b>5 Conclusão</b>	<b>91</b>
<b>A Arquivo de registros do usuário novato</b>	<b>103</b>
<b>B Arquivo de registros do usuário intermediário</b>	<b>107</b>
<b>C Ilustrações dos componentes de interface para dispositivos móveis</b>	<b>111</b>
<b>Referências bibliográficas</b>	<b>102</b>

# Lista de Figuras

---

---

2.1	Exemplo de questão fechada de múltipla escolha que permite mais de uma resposta. Elaborada eletronicamente por meio de caixas de múltipla escolha. . . . .	10
2.2	Exemplo de componente de texto contendo informações como dicas ao usuário no próprio componente. . . . .	11
2.3	Aplicativo “agenda” com interface fazendo analogia a uma agenda de papel [Apple, 2010]. . . . .	16
2.4	Tela criada com elementos com pelo menos 48dp de altura e de largura. O esquema gráfico está à esquerda, enquanto que a interface final está à direita [Google, 2012a].	17
2.5	Exemplo de mensagem de erro bem-humorada. . . . .	23
2.6	Exemplo de interface desmistificada [Moraveji e Soesanto, 2012]. . . . .	24
2.7	Exemplo bem sucedido de aplicação do princípio CARP de alinhamento a uma página <i>Web</i> . . . . .	27
2.8	Exemplo bem sucedido de aplicação do princípio CARP de repetição a uma página <i>Web</i> . . . . .	27
2.9	Processo de design. [Dix et al., 2004] . . . . .	28
3.1	Dispositivos usados na fase de uso de aplicativos baseados em Android. À esquerda, <i>smartphone</i> I5500 da Samsung. À direita, <i>tablet</i> XOOM da Motorola. . . . .	31
3.2	Tela inicial do aplicativo de anotação multimídia em vídeos. . . . .	34
3.3	Tela para seleção de usuário que irá navegar pelo aplicativo de anotação multimídia.	35
3.4	Tela para seleção de vídeo a ser anotado. . . . .	35
3.5	Tela em que o usuário escolhe o autor do vídeo a ser anotado. . . . .	36
3.6	Tela principal da aplicação de anotação de vídeos. . . . .	36
3.7	Formato das diretrizes para criação de interfaces de dispositivos móveis. . . . .	39
4.1	Relação entre o número de problemas encontrados por cada grau de severidade. . .	63
4.2	Percentual de problemas encontrados com cada grau de severidade. . . . .	64
4.3	Quantidade de problemas associados a cada heurística de Nielsen. . . . .	64
4.4	Quantidade de problemas associados a cada heurística para avaliação de interfaces de dispositivos móveis. . . . .	65
4.5	Código-fonte responsável pela criação e gravação do arquivo de registros de ações de usuários do aplicativo de anotações multimídias. . . . .	66
4.6	Trecho de um dos arquivos gravados no cartão SD do dispositivo móvel. . . . .	67

4.7	Quantidade de erros cometidos por cada usuário para realizar cada atividade a ele requerida.Trecho de um dos arquivos gravados no cartão SD do dispositivo móvel. .	68
4.8	Tempo gasto por cada usuário para realizar cada atividade a ele requerida.Trecho de um dos arquivos gravados no cartão SD do dispositivo móvel. . . . .	68
C.1	Barra de Abas. A) Android; B) iOS; C) Windows Phone. . . . .	112
C.2	Barra de Atividades. A) Android; B) iOS. . . . .	112
C.3	Barra de navegação. A) Android; B) iOS. . . . .	113
C.4	Barra de progresso. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	113
C.5	Barra de status. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	113
C.6	Barra deslizante de ajuste. A) Android; B) Blackberry; C) iOS; D) Windows Phone. .	113
C.7	Barra inferior. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	114
C.8	Botão. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	114
C.9	Caixa de múltipla escolha. A) Android; B) Blackberry; C) iOS; D) Windows Phone. .	114
C.10	Caixa de seleção. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	114
C.11	Campo de busca. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	114
C.12	Campo de texto. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	115
C.13	Diálogo. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	115
C.14	Girador de progresso. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	115
C.15	Link. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	115
C.16	Lista. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	116
C.17	Menu popup. A) Android; B) Blackberry. . . . .	116
C.18	Radio button. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	116
C.19	Seletor de arquivos. A) Blackberry. . . . .	116
C.20	Seletor de data. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	117
C.21	Seletor de hora. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	117
C.22	Seletor numérico. A) Android; B) iOS; C) Windows Phone. . . . .	117
C.23	Submenu. A) Android; B) Windows Phone. . . . .	117
C.24	Switch. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	118
C.25	Tabela. A) Android; B) Blackberry; C) iOS; D) Windows Phone. . . . .	118

# Lista de Tabelas

---

---

3.1	Relação entre aplicativos avaliados e principais funcionalidades que cada um deles oferece. . . . .	32
4.1	Categorização dos problemas encontrados nas interfaces do Facebook. . . . .	41
4.2	Categorização dos problemas encontrados nas interfaces do Foursquare. . . . .	43
4.3	Categorização dos problemas encontrados nas interfaces do Gmail. . . . .	43
4.4	Categorização dos problemas encontrados nas interfaces do Twitter. . . . .	44
4.5	Associação das categorias encontradas às heurísticas de Nielsen. . . . .	45
4.6	Primeira versão das heurísticas para avaliação da usabilidade de interfaces de dispositivos móveis. . . . .	47
4.7	Segunda versão das heurísticas para avaliação de usabilidade de interfaces de dispositivos móveis. . . . .	49
4.8	Heurísticas revisadas de Nielsen traduzidas por Rocha e Baranauskas [2003]. . . .	51
4.9	Avaliação heurística pelo primeiro especialista, com base nas heurísticas de Nielsen.	52
4.10	Avaliação heurística pelo segundo especialista, com base nas heurísticas de Nielsen.	52
4.11	Avaliação heurística pelo terceiro especialista, com base nas heurísticas de Nielsen.	53
4.12	Avaliação heurística pelo quarto especialista, com base nas heurísticas de Nielsen.	53
4.13	Avaliação heurística pelo quinto especialista, com base nas heurísticas de Nielsen. .	54
4.14	Avaliação heurística pelo primeiro especialista, com base nas heurísticas para dispositivos móveis. . . . .	55
4.15	Avaliação heurística pelo segundo especialista, com base nas heurísticas para dispositivos móveis. . . . .	56
4.16	Avaliação heurística pelo terceiro especialista, com base nas heurísticas para dispositivos móveis. . . . .	56
4.17	Avaliação heurística pelo quarto especialista, com base nas heurísticas para dispositivos móveis. . . . .	57
4.18	Avaliação heurística pelo quinto especialista, com base nas heurísticas para dispositivos móveis. . . . .	58
4.19	Categorização dos problemas encontrados nas interfaces do Twitter. . . . .	60
4.20	Sumarização das quantidades de problemas encontrados com cada grau de severidade, por cada avaliador que usou as heurísticas de Nielsen. . . . .	62

4.21	Sumarização das quantidades de problemas encontrados com cada grau de severidade, por cada avaliador que usou as heurísticas para interfaces de dispositivos móveis. . . . .	63
4.22	Resultado da avaliação heurística do UOL Notícias utilizando-se o iPhone 4S . . . .	73
4.23	Resultado da avaliação heurística do UOL Notícias utilizando-se o Motorola Blur . .	74
4.24	Resultado da avaliação heurística do UOL Notícias utilizando-se o Motorola XOOM.	74
4.25	Mapeamento da terminologia de componentes usada pelas empresas Google, BlackBerry, Apple e Microsoft. . . . .	75

# Lista de Quadros

---

---

4.1	Barra de Abas. . . . .	76
4.2	Barra de Atividades. . . . .	77
4.3	Barra de Navegação. . . . .	77
4.4	Barra de Progresso. . . . .	78
4.5	Barra de Status. . . . .	78
4.6	Barra Deslizante de Ajuste. . . . .	79
4.7	Barra Inferior. . . . .	79
4.8	Botão. . . . .	79
4.9	Caixa de múltipla escolha. . . . .	80
4.10	Caixa de seleção. . . . .	80
4.11	Campo de busca. . . . .	81
4.12	Campo de texto. . . . .	81
4.13	Diálogo. . . . .	82
4.14	Girador de progresso. . . . .	82
4.15	Link. . . . .	83
4.16	Lista. . . . .	84
4.17	Menu Popup. . . . .	84
4.18	Radio Button. . . . .	85
4.19	Seletor de Arquivos. . . . .	85
4.20	Seletor de data. . . . .	86
4.21	Seletor Numérico. . . . .	86
4.22	Submenu. . . . .	87
4.23	Switch. . . . .	87
4.24	Tabela. . . . .	88
4.25	Texto (Rótulo). . . . .	88



---

# Introdução

---

Neste capítulo são apresentadas a motivação para a pesquisa e a descrição do problema abordado, os objetivos do trabalho e a estruturação do restante do documento.

## 1.1 Contextualização

A evolução das tecnologias de informação (TIs) contribuiu grandemente para a agregação de valor a processos e serviços [Torres, 1995]. Por conta disso, as TIs se tornaram fundamentais para aperfeiçoar os processos organizacionais em geral [Goodman e Lawless, 1994] e as estratégias competitivas de empresas e pessoas [Lai e Mahapatra, 1997]. O computador pessoal popularizou-se e, com ele, cresceu o número de pessoas que acessam a internet.

Paralelamente ao crescimento e mudanças constantes da internet, aumentou também o número de dispositivos capazes de acessar a rede de computadores. Hoje é comum ter internet no *notebook*, no *netbook* e em dispositivos móveis, como *smartphones*, *tablets* e PDAs.

Essa popularização da Internet, aliada ao interesse de se usarem aparelhos móveis em lugares diversos enquanto se realizam tarefas cotidianas, contribuiu para que os dispositivos evoluíssem tecnologicamente. Pode-se dizer que eles se tornaram tecnologias multimídia poderosas, capazes de disponibilizar diferentes tipos de conteúdo a boas taxas de processamento. Entretanto, apesar de cada vez mais onipresente no dia-a-dia das pessoas que os utilizam, ainda há carência de estudos de *design* de interface específicos para tais dispositivos.

O *design* de interfaces de sistemas interativos é uma tarefa tão relevante que se tornou uma das subáreas da Interação Humano-Computador (IHC) que, por sua vez, visa estudar, planejar e entender como pessoas e dispositivos computacionais podem interagir de forma que as necessidades delas sejam contempladas da forma mais efetiva possível [Galitz, 2003]. De modo geral, essa efetividade de interação é obtida quando o usuário percebe o sistema e consegue se comunicar com ele da forma mais natural possível. Evidentemente, para interagem com o usuário, dispositivos computacionais devem dispor de meios captadores de reações sinestésicas dos humanos, como tato, visão ou audição. Esse meio de captação em sistemas computacionais é chamado de “interface” e, para que ela possa maximizar a comunicação entre humanos e

computadores, é necessário que ela possua boa usabilidade.

Gatica-Perez e Montoliu [2010] observam que a evolução dos dispositivos móveis tem feito com que eles se tornem dispositivos multimídia naturais por serem capazes de adquirir, acessar, gerenciar e transmitir múltiplos tipos de mídia como vídeo, imagens, áudio e mapas. Os autores identificam a demanda por pesquisas associadas a interfaces com usuários que lhes garantam usabilidade e por estudos relativos à análise de comportamentos de usuários, para desenvolvimento de tais interfaces, como realizado por Bigonha et al. [2010].

O conceito de usabilidade é antigo e não se aplica apenas ao contexto computacional. Segundo Galitz [2003], Bennett [1979] foi o primeiro pesquisador a usar o termo “usabilidade”, referindo-se a ela como a efetividade com que o usuário realiza suas atividades. Nos anos seguintes, Shackel [1981] propôs uma definição um pouco mais formal para o termo. Finalmente, Shackel [1991] divulgou uma definição que, segundo Galitz [2003], é simples e pertinente: algo apresenta grande usabilidade quando humanos conseguem usá-lo com facilidade e efetividade, sendo facilidade uma métrica de avaliação subjetiva e efetividade o desempenho humano ao usá-lo.

Existem variadas definições de “usabilidade” na Literatura Científica. Dentre elas, são famosas as elaboradas pela Organização Internacional de Padronização (ISO), por elas parecerem mais abrangentes que as demais. O padrão ISO [1998] define usabilidade como um objetivo descrito em alto nível de abstração: a capacidade de um produto poder ser usado por usuários específicos para atingir metas específicas com efetividade, eficiência e satisfação, com base em um contexto de utilização específico. Apesar de a definição ser abrangente, segundo Seffah et al. [2001] o relacionamento entre os parâmetros efetividade, eficiência e satisfação e os objetivos da usabilidade é nebuloso, difícil de ser definido. A ISO/IEC [2001], por sua vez, especifica usabilidade em um contexto estritamente computacional como a capacidade de o *software* ser entendido, aprendido e usado de forma que o usuário se sinta atraído pelo sistema. Entretanto, essa definição não expõe claramente as métricas segundo as quais se devem concentrar os esforços de *design*, falhando do ponto de vista prático porque “atração” não é uma métrica fácil de ser aferida [Kunjachan, 2011].

Shneiderman e Plaisant [2009] foram capazes de especificar um conjunto de métricas que podem ser aferidas do ponto de vista prático para facilitar que os objetivos de eficiência e satisfação no contexto de usabilidade de dispositivos computacionais sejam atingidos. Tais fatores são os seguintes:

- **Tempo de aprendizagem.** Tempo necessário para que um usuário típico do sistema aprenda as ações relevantes que devem ser executadas, a fim de que sejam realizadas as principais atividades disponibilizadas pelo sistema.
- **Desempenho (*performance*).** Tempo necessário para que o usuário execute as tarefas principais do sistema.
- **Taxa de erros cometidos pelo usuário.** Contagem do número de erros que os usuários cometem para executar cada tarefa fundamental do sistema e respectiva listagem desses erros. Apesar de o tempo gasto pelo usuário para identificar um erro e em seguida se recuperar dele também ser contabilizado na métrica de desempenho, a taxa de erros é uma preocupação capital que precisa ser tratada de forma separada.
- **Sedimentação do conhecimento por experiência.** Visa analisar a facilidade para que um usuário típico execute as funcionalidades principais do sistema uma hora, um dia ou uma semana depois de tê-las realizado pela primeira vez. Esta métrica tem um relacionamento

estreito com a medida “tempo de aprendizagem” e é diretamente afetada pela frequência com a qual o usuário interage com as interfaces do sistema.

- **Satisfação subjetiva.** Métrica para analisar o quanto os usuários gostaram de utilizar as interfaces do sistema. Essa pergunta pode ser respondida por meio de entrevistas com os usuários ou com questionários contendo questões na forma de escalas de satisfação ou mesmo na forma aberta, em que os usuários se sentem completamente livres para responderem com as próprias palavras.

Evidentemente, é impossível criar um sistema com interfaces que permitam excelentes resultados em todas as métricas expostas, porque elas se influenciam: se a taxa de erros for muito pequena, provavelmente o desempenho será afetado; se o tempo de aprendizagem não for um fator crítico, é possível ganhar desempenho pelo uso de abreviações, macros ou atalhos [Shneiderman e Plaisant, 2009]. Efetividade e satisfação normalmente são obtidas quando todas as métricas forem planejadas com bom senso, ou seja, de acordo com as necessidades do sistema e dos usuários.

De fato, atualmente os *designers* de interface desenvolvem sua capacidade criativa de acordo com as funcionalidades que o sistema deve disponibilizar e de acordo com os diferentes grupos de usuários que o utilizarão. Diferentes conjuntos de recomendações para desenvolvimento de interfaces foram desenvolvidos [Shneiderman e Plaisant, 2009] [Gong e Tarasewich, 2011] [Bertini et al., 2006] [Moraveji e Soesanto, 2012], mas se nota que eles são completamente dependentes do contexto da aplicação, do ponto de vista de suas funcionalidades e de seus usuários [Shneiderman e Plaisant, 2009]. De qualquer forma, é inegável que, do ponto de vista do usuário, a interface é a parte mais importante dos sistemas computacionais, porque nela está contido tudo que o usuário consegue fazer. Isso significa que, apesar de não tornar o sistema funcional por si só, a interface é, para a maioria dos usuários, o próprio sistema em si. Todas as demais camadas de código-fonte necessárias para fazerem o sistema funcionar são invisíveis a quem utiliza sistemas computacionais.

As interfaces evoluíram de terminais baseados em comandos de texto para as interfaces gráficas atuais, amplamente difundidas principalmente por conta da *World Wide Web* e da popularização da internet. De acordo com a Agência de Telecomunicações das Nações Unidas, o número de pessoas que acessam a rede passou de 1,6 bilhão para mais de dois bilhões entre os anos de 2010 e 2011, um crescimento expressivo que permitiu a consolidação das atuais interfaces gráficas como os principais meios de interação do usuário com o computador.

Alguns entusiastas acreditam que os dispositivos computacionais e suas interfaces evoluirão a ponto de se tornarem invisíveis, embutidas no próprio ambiente do usuário de forma que a interação seja imperceptível [Weiser, 1993]. Embora os conservadores afirmem que essa evolução é utópica, nota-se que o uso de dispositivos móveis para interagir com usuários em diferentes ambientes têm auxiliado que as interfaces se tornem imperceptíveis, porque os dispositivos são capazes de identificar algumas informações acerca do ambiente e dos usuários automaticamente [Krumm, 2010], respondendo a eventos diversos. De qualquer forma, existe um grande interesse em desenvolver interfaces mais inteligentes e com mais usabilidade. Neste trabalho, o *design* de interfaces será tratado com foco em usabilidade e no contexto de dispositivos móveis.

As principais motivações para o desenvolvimento de interfaces com boa usabilidade são a existência de inúmeras interfaces com *design* pobre, que não facilitam a interação e que precisam de melhorias, e o grande benefício que interfaces robustas e elegantes proporcionam aos usuários, em termos das métricas de usabilidade apresentadas. Shneiderman e Plaisant [2009] enumeram alguns ambientes que necessitam de interfaces com boa usabilidade para o contexto

de dispositivos eletrônicos em geral. Algumas das aplicações levantadas por esses autores podem ser estendidas ao contexto de aplicativos para dispositivos móveis como motivações deste estudo, e seguem:

- **Uso comercial e industrial.** Aplicativos típicos de uso comercial e industrial incluem sistemas bancários, sistemas de controle de estoque, sistemas controle de inventários, de controle de reservas de hotéis, dentre outros. Nesses casos, geralmente o orçamento é o fator mais limitante e os gastos com treinamento de pessoal são normalmente altos, de forma que as interfaces devam ser desenvolvidas com foco em facilidade de aprendizagem. Além disso, é comum que as interfaces para esses tipos de sistema necessitem de bom desempenho, por conta do número grande de transações que são processadas em pouco tempo.
- **Sistemas para lares, escritórios e para entretenimento.** Aplicativos desta seção incluem e-mail, jogos, sistemas de gerenciamento de arquivos, sistemas de propósito educativo e mecanismos de buscas. Normalmente, facilidade de aprendizagem, baixa taxa de erros e satisfação subjetiva são fatores de usabilidade importantes a serem considerados no *design* dessas interfaces, principalmente porque tais aplicativos costumam não ter restrições de usuários e normalmente existem concorrentes no mercado. Usuários incapazes de executar as funcionalidades desses sistemas com facilidade tendem a procurar aplicativos concorrentes ou a parar de usar o aplicativo.
- **Interfaces colaborativas, criativas ou exploratórias.** Interfaces exploratórias incluem navegadores *Web*, simuladores de experimentos e sistemas de apoio a negócios. Sistemas destinados ao estímulo da capacidade criativa incluem interfaces para aplicativos artísticos e de *design* arquitetônico. Normalmente, interfaces que se enquadram nessa seção são operadas por usuários com grande domínio das funcionalidades realizadas, mas com pouco conhecimento dos conceitos computacionais envolvidos. Interfaces colaborativas permitem que duas ou mais pessoas operem uma mesma funcionalidade, simultaneamente ou não.
- **Interfaces de caráter sociopolítico.** Aplicativos dessa seção geralmente são criados por entidades governamentais e incluem sistemas de votação popular, de monitoramento de saúde de pacientes ou de criação de perfis para investigações criminais. Desses, nota-se que o uso para fins médicos está mais difundido para o caso de aplicativos de dispositivos móveis, por conta de a segurança e a privacidade ainda serem problemas capitais no caso desses dispositivos.

A usabilidade de interfaces de dispositivos móveis é muito peculiar, porque eles possuem limitações que inexistem no caso de computadores pessoais. As limitações mais evidentes são de caráter físico. Por exemplo, as telas dos dispositivos móveis são consideravelmente menores que as dos computadores pessoais. No caso, é oportuno lembrar que dois fatores devem ser considerados quando se cita a tela de um dispositivo como fator importante para usabilidade de interfaces: o comprimento da diagonal da tela do aparelho, que define o número de polegadas que ela possui, e a resolução da tela, ou seja, a quantidade de pixels existentes em proporção à área da tela.

Em seus estudos sobre desenvolvimento de aplicativos para dispositivos móveis, Oinas-Kukkonen e Kurkela [2003] constataram que os aplicativos bem sucedidos possuem *design* focado não só na limitação física da tela, mas também em outras limitações destes aparelhos, tais como bandas de acesso menores, eventuais gargalos com processamento e memória e diferenças

entre os métodos de entrada de dados, quando comparados aos computadores pessoais. Além disso, existem preocupações igualmente importantes que fazem parte de outros contextos, como o social e o cognitivo.

No contexto social, as limitações estão centradas principalmente na diversidade cultural das pessoas de diferentes regiões, mas que eventualmente utilizam as mesmas interfaces. Pessoas de diferentes locais geográficos tendem a ter expectativas diferentes acerca dos aplicativos que utilizam. O mesmo ocorre quando se comparam pessoas com idades muito discrepantes (crianças em relação a idosos, por exemplo). Finalmente, o próprio propósito da aplicação pode gerar em um determinado usuário uma expectativa diferente da gerada para outro usuário [Galitz, 2003].

Do ponto de vista cognitivo, existem evidências científicas que comprovam que o cérebro humano é extremamente sensível a interferências, que fazem com que o usuário se distraia enquanto interage com um dispositivo móvel. Isso significa que, ao interagir com um aplicativo, o usuário perde parte da atenção quando eventos externos ocorrem e essa desconcentração influencia no tempo de aprendizagem e na velocidade com que as funcionalidades são executadas [Gazzalley, 2012]. Pode-se notar esse fenômeno com o exemplo de um passageiro em um metrô, que precisa prestar atenção às informações que acabaram de serem passadas a ele pela cabine de controle de estações enquanto ele utiliza um aparelho *smartphone*.

Todas as características citadas com relação ao modo com que usuários interagem com dispositivos móveis servem como grandes motivadores de pesquisas relacionadas ao *design* de interfaces. De maneira genérica, pode-se afirmar que a disseminação de dispositivos móveis diversos representa um grande desafio aos pesquisadores da área de Interação Humano-Computador, porque os usuários desejam que seus aplicativos sejam capazes de serem acessados de forma prática nos vários aparelhos que eles utilizam, mesmo que os sistemas computacionais e os ambientes de utilização desses aparelhos mudem constantemente. É com base nesse anseio do usuário que este projeto de pesquisa é útil. A seguir, a identificação do problema a ser tratado será realizada.

## 1.2 Identificação do problema

Apesar de o tema “usabilidade” ter-se intensificado com a preocupação de se criarem interfaces melhores para computadores em geral, nota-se que a Literatura Científica não conta com muitos estudos que vinculam as principais contribuições sobre o *design* de interfaces com usabilidade para dispositivos móveis, tomando-se como base os princípios e guias consolidados para o *design* de interfaces de dispositivos em geral. Os estudos referentes ao contexto de *design* de interfaces para aparelhos móveis parecem, portanto, desconexos de algumas conclusões teóricas consolidadas cientificamente. Esse desvínculo poderia ter gerado estudos que não identificassem problemas de usabilidade da forma que poderiam.

Historicamente, Galitz [2003] foi o primeiro estudioso a propor ideias gerais de como se construir interfaces gráficas com usabilidade [Shneiderman e Plaisant, 2009]. Nesse estudo, princípios importantes de *design* foram levados em consideração, mas Galitz [2003] não expõe claramente um conjunto de regras a serem analisadas pelos envolvidos no processo de *design*. Posteriormente, os próprios Shneiderman e Plaisant [2009] propuseram um novo guia de como se realizar a mesma tarefa, que resultaram em uma lista de recomendações para interessados em criar interfaces com usabilidade. Apesar de a lista proposta ser mais aplicável do ponto de vista prático, Shneiderman e Plaisant [2009] não consideraram fatores psicológicos que influenciam o usuário quando ele utiliza dispositivos computacionais. Ademais, nenhum desses três autores levou em consideração o *design* de interfaces para dispositivos móveis.

Tanto Shneiderman e Plaisant [2009] quanto Galitz [2003] apresentam guias de *design* que parecem influenciados por estudos anteriores realizados por Nielsen (1994), responsável pelo conjunto de heurísticas para avaliação de usabilidade de interfaces de dispositivos computacionais mais conhecido pelos pesquisadores de Interação Humano-Computador [Moraveji e Soesanto, 2012], e seus estudos são base para outras contribuições no âmbito de *design* de interface. Embora esses estudos sejam relevantes, eles compartilham a mesma limitação existente nos estudos de Shneiderman e Plaisant [2009] e de Galitz [2003]: *design* de interface sem foco em dispositivos móveis. A avaliação de interfaces é uma atividade indispensável ao processo de desenvolvimento de sistemas interativos, porque ela permite que os *designers* verifiquem se a interface do sistema atende aos requisitos do usuário de modo que as funcionalidades sejam realizadas de modo efetivo. Na avaliação heurística, um conjunto de especialistas avalia se a interface está de acordo com um conjunto de regras, capazes de identificar potenciais problemas de usabilidade [Nielsen, 1994].

Aparentemente Bertini et al. [2006] foram os poucos autores a tentarem propor heurísticas para avaliação de interfaces de dispositivos móveis. Eles criaram esse conjunto de heurísticas com base no conhecimento de três pesquisadores experientes da área de Interação Humano-Computador, os quais identificaram problemas de usabilidade que consideravam importantes de serem tratados em interfaces de dispositivos móveis. Tais problemas foram colocados em uma planilha, que foi discutida entre os três na forma de um *brainstorming*. O resultado foi um conjunto de oito heurísticas para avaliação de interfaces de dispositivos móveis geradas a partir das heurísticas de [Nielsen, 1994], as quais foram analisadas e aprovadas por mais oito especialistas da área. Apesar de as heurísticas serem focadas em interfaces de contexto de dispositivos móveis, os resultados das avaliações não foram enumerados explicitamente no estudo e não se fez uma descrição detalhada do aplicativo avaliado. Além disso, a metodologia usada não considera alguns conceitos teóricos para o design, como as regras básicas de *design* elaboradas por Williams [2005] ou teorias de características de interface que incomodam psicologicamente o usuário [Moraveji e Soesanto, 2012].

Com propósitos semelhantes aos de Bertini et al. [2006], Gong e Tarasewich [2011] revisitaram os estudos de Shneiderman e Plaisant [2009] e os adaptaram para criarem um novo conjunto de regras que, segundo aqueles autores, seriam pertinentes para o contexto de interfaces de dispositivos móveis. Contudo, a metodologia utilizada para chegarem a tais diretrizes não foi encontrada nas fontes pesquisadas, de modo que o valor científico do estudo ficou comprometido. Os autores também não descreveram nenhum experimento científico que validasse as heurísticas propostas por eles.

É importante ressaltar que existem sistemas operacionais específicos para dispositivos móveis e que as empresas detentoras de alguns desses sistemas divulgam em seus *websites* oficiais diretrizes para a elaboração de interfaces de forma que os elementos de interação sejam utilizados de maneira adequada, a fim de maximizar a usabilidade do aplicativo final. As empresas pesquisadas para esta dissertação que divulgam essas *guidelines* são: Apple, criadora do sistema iOS; Blackberry, criadora do sistema Blackberry; Google, criador do sistema Android, e Microsoft, criadora do sistema Windows Phone.

A divulgação dessas *guidelines* por meio dessas grandes empresas evidencia que a criação de interfaces com adequação aos princípios básicos de usabilidade conhecidos transcendeu o contexto científico e atingiu o contexto comercial. Contudo, as documentações são baseadas nos componentes de interfaces existentes em cada sistema operacional que, apesar de terem propósitos semelhantes entre os sistemas, possuem terminologias distintas. Em alguns casos, até a localização dos componentes de interfaces variam de sistema para sistema.

Essas diferenciações dificultam que pessoas sem conhecimento técnico na linguagem de programação usada no sistema operacional sigam tais recomendações. Por conta disso, esta dissertação propõe um conjunto de diretrizes, o qual poderá ser usado por *designers* sem conhecimentos técnicos em Computação criarem protótipos de interfaces.

Além dessa motivação, outro fator que norteou o trabalho foi a dificuldade de encontrar referências científicas que disponibilizassem diretrizes ou heurísticas específicas para interfaces de dispositivos móveis. Portanto, são objetivos deste trabalho criar um conjunto de *guidelines* genérico que complemente as recomendações oficiais das empresas detentoras dos sistemas operacionais existentes, e propor um conjunto de heurísticas para avaliação de usabilidade de interfaces de dispositivos móveis. Em particular, o conjunto de heurísticas é baseado nas heurísticas tradicionais já consolidadas, nas *guidelines* para criação de interfaces e em princípios de usabilidade que, até onde foi possível verificar, não são citados nos estudos relacionados na Literatura Científica. Todos esses tópicos teóricos serão explanados no Capítulo 2.

Finalmente, cabe ressaltar que o conceito de usabilidade é inevitavelmente vinculado às funcionalidades do sistema, seja ele elaborado para computadores pessoais tradicionais, baseado na *Web* ou desenvolvido para dispositivos móveis, porque ela abrange não somente a interface, mas também vários outros fatores, como utilidade e confiabilidade [Nielsen, 1993]. Por conta disso, é comum encontrar referências com dicas de como tratar a usabilidade de sistemas de forma atrelada às funcionalidades. As questões a serem respondidas neste projeto podem ser verificadas no subcapítulo 1.3.

### 1.3 Questão de pesquisa

Este trabalho de pesquisa pretende responder à seguinte questão:

- As heurísticas de Nielsen são deficientes em identificar problemas de usabilidade de aplicações de dispositivos móveis?

As atividades descritas na Seção 3.1.2 indicaram que a resposta à pergunta acima é “sim”. Por conta disso, o mestrando identificou a oportunidade de estender as heurísticas de Nielsen para elaborar heurísticas para a avaliação de interfaces de dispositivos móveis.

### 1.4 Objetivos

Após verificar que as heurísticas de Nielsen não eram facilmente associadas a determinadas categorias de problemas de usabilidade encontrados em quatro aplicativos de dispositivos móveis inspecionados, este trabalho teve por objetivo propor um conjunto de heurísticas para avaliação de interfaces de dispositivos móveis por meio do uso de tais categorias de problemas e das heurísticas já existentes. O objetivo secundário do trabalho foi definir um conjunto de diretrizes para o desenvolvimento dessas interfaces, por meio da análise de recomendações para o *design* de interfaces disponibilizadas por empresas detentoras de sistemas de dispositivos móveis.

### 1.5 Resultados

Os resultados indicaram que, apesar de as heurísticas propostas terem se baseado nas heurísticas de Nielsen, elas encontraram um número maior de problemas quando comparadas às heurísticas de Nielsen. O número de problemas considerados pelos avaliadores como de baixa prioridade também foi maior. Além disso, o número de problemas considerados catastróficos foi mais que o dobro do encontrado pelas heurísticas de Nielsen.

## **1.6 Estrutura do trabalho**

O restante deste documento está estruturado da seguinte forma: no Capítulo 2 são explicados os principais resultados acerca do tema abordado neste trabalho, os quais serão usados como base para a criação dos documentos citados como objetivos na Seção 1.4. Em seguida, no Capítulo 3, será abordada a Metodologia de Pesquisa utilizada. No Capítulo 4, descrever-se-ão os resultados obtidos ao longo do estudo. O Capítulo 5 contém as conclusões do projeto de pesquisa, que visam responder às questões sumarizadas na Subseção 1.3. Por fim, são apresentadas as Referências Bibliográficas utilizadas.

Neste trabalho, o termo “mestrando” será usado para referir-se ao aluno que realizou este trabalho.

---

# Recursos para construção de interfaces com usabilidade

---

Este capítulo contém informações que servem de subsídio para pessoas interessadas na criação de interfaces com boa usabilidade. Trata-se de um apanhado de ideias que são aceitas na Literatura Científica para auxiliar neste processo de criação e se destina a estudantes e jovens pesquisadores que desejem iniciar uma pesquisa nessa área e a demais interessados em criação de interfaces, tais como *webdesigners*, desenvolvedores de *software* e publicitários. Os estudos não são necessariamente independentes, no sentido de que um conjunto de recomendações pode ter-se baseado noutras recomendações elaboradas por outrem. Por completude, todos os estudos relacionados utilizados neste trabalho serão expostos neste capítulo. Entretanto, algumas recomendações expostas tornaram-se obsoletas. Uma ressalva será realizada nas Seção 2.4 para elucidar tais recomendações.

## 2.1 Diretrizes (*Guidelines*)

Diretrizes são conjuntos de regras para o *design* de interfaces, que permitem que os envolvidos nesse processo de *design* tenham uma documentação para prevenir e corrigir erros de usabilidade que já são conhecidos por conta da experiência de outrem ou por conta de conhecimentos oriundos da Psicologia [Preece, 1994]. Estas regras representam, portanto, uma linguagem comum que facilita o trabalho em equipe e uniformiza as interfaces sendo criadas, em termos de terminologias, de sequências de ações necessárias para se completar determinadas tarefas e de aparências [Shneiderman e Plaisant, 2009]. Apesar de ainda se criarem *guidelines* para o *design* de interfaces, essa é uma prática relativamente antiga entre os pesquisadores da área de Interação Humano-Computador. Em 1995, Mullet e Sano [1995] criaram um conjunto de recomendações que se acredita serem as primeiras regras voltadas para o *design* de criação de interfaces gráficas (ou simplesmente GUIs, do Inglês Graphical User Interfaces) em dispositivos computacionais, de modo geral. Mais tarde, Lynch e Horton [2001] criaram um novo conjunto de regras para esse mesmo tipo de interface, as quais foram atualizadas por Ga-

Na sua opinião, quais são os motivos de alguns estudantes não respeitarem o professor na sala de aula?

Falta de disciplina em casa;

Falta de cobrança do professor;

Falta de rigor da legislação brasileira com os alunos;

Maus exemplos dados pelos programas de TV;

Influência dos videogames;

Outro:

checkboxes

**Figura 2.1:** Exemplo de questão fechada de múltipla escolha que permite mais de uma resposta. Elaborada eletronicamente por meio de caixas de múltipla escolha.

litz quatro anos depois. Hoje, as interfaces gráficas ainda desempenham um papel importante na interação com usuários, já que a maioria dos formulários de sistemas atuais é baseada em componentes gráficos, principalmente por conta da expansão da *World Wide Web*.

Provavelmente, as *guidelines* mais conhecidas para o *design* de interfaces foram propostas por Shneiderman e Plaisant [2009]. Embora elas não tenham sido elaboradas especificamente para a criação de interfaces de dispositivos móveis, elas são usadas como base para que pesquisadores criem novas *guidelines* para avaliarem interfaces de aparelhos móveis [Gong e Tarasewich, 2011].

### 2.1.1 Diretrizes para criação de interfaces de dispositivos computacionais em geral

O conjunto de diretrizes elaborado por Shneiderman e Plaisant [2009] permitem que se previnjam erros de usabilidade de interfaces de dispositivos computacionais, de modo geral. Em seu estudo, esses autores organizaram as recomendações de acordo com quatro interesses, considerados por eles de grande importância para o processo de *design*. São eles: navegabilidade da interface, organização do conteúdo exibido, captura da atenção do usuário e facilitação de entrada de dados.

A navegabilidade da interface é um fator de preocupação importante para o processo de *design* porque muitos usuários apresentam problemas para navegar por determinadas interfaces computacionais [Koyani et al., 2003]. Com o propósito de melhorar a navegação pela interface por parte do usuário, Shneiderman e Plaisant [2009] criaram o seguinte conjunto de recomendações:

- **Padronize a sequência de tarefas.** Funcionalidades semelhantes devem ser executadas seguindo-se uma mesma sequência de atividades, de maneira análoga.
- **Assegure que os links da página sejam descritivos.** Quando um link é criado, o texto descritivo associado a ele deve explicar com precisão a ação a ser realizada caso ele seja ativado.
- **Use cabeçalhos distintos para funcionalidades distintas.** Cabeçalhos devem ser associados ao conteúdo sendo apresentado. Conteúdos diferentes devem ter cabeçalhos diferentes.
- **Use caixas de múltipla escolha (*checkboxes*) para escolhas que possam apenas duas possíveis respostas.** Escolhas referentes a perguntas que permitem apenas duas respostas claras e distintas devem ser exibidas na forma de *checkboxes*, como se mostra na Figura 2.1.
- **Permita que o usuário imprima a página adequadamente.** Desenvolva as interfaces em medidas adequadas para a impressão, para que todos os componentes da tela impressa apareçam corretamente no papel.



Figura 2.2: Exemplo de componente de texto contendo informações como dicas ao usuário no próprio componente.

- **Use imagens pequenas como forma de pré-visualizar as imagens no tamanho original.** Em interfaces em que a visualização da imagem no tamanho original não seja necessária, providencie uma imagem menor ao usuário.

A organização da informação a ser exibida é um ponto de atenção porque conteúdos mal exibidos acarretam em interpretações erradas dos usuários. Além disso, a disposição dos componentes pode comprometer a visualização deles pelo usuário e associação de cada informação a componentes diversos da interface. As recomendações de Shneiderman e Plaisant [2009] para melhoria da organização da informação derivam de contribuições feitas bem antes, por Smith e Mosier [1986], e seguem:

- **Consistência dos dados sendo exibidos.** A terminologia, abreviações, formatos, cores e capitalizações de termos devem ser documentados na forma de um dicionário de significados, para que todos os *designers* saibam como e quando utilizá-los.
- **Facilidade de assimilação da informação pelo usuário.** A linguagem deve ser familiar ao usuário, o formato de exibição de dados deve permitir que os dados sejam facilmente assimilados. Estão inclusas nessa recomendação espaçamento correto, alinhamento à esquerda de palavras, alinhamento de casas decimais de números em ponto flutuante, dentre outros aspectos.
- **Minimização da carga de memória do usuário.** Deve-se evitar que o usuário tenha que se lembrar de informações entre interfaces diferentes da aplicação. Cada atividade deve ser planejada de forma que o usuário a complete em um pequeno número de passos. Em casos de grandes quantidades de informações, deve-se separar a atividade em partes independentes, cada qual contendo um conjunto pequeno de informações às quais o usuário deve prestar atenção.
- **Compatibilidade entre a entrada de dados e o conteúdo exibido.** Elementos de informação exibidos ao usuário devem associar-se de forma precisa e sem ambiguidades com os componentes de entrada de dados. Eventualmente, elementos de entrada de texto podem exibir informações ao usuário nos próprios espaços de entrada, como se pode notar na Figura 2.2.
- **Flexibilidade do acesso e uso das informações.** Os usuários devem ser capazes de assimilar as informações das interfaces de diferentes formas, para que eles mesmos escolham o modo com que deverão realizar determinadas atividades. Ordenações de linhas e colunas de tabelas são bons exemplos de como tornar dados tabulares flexíveis ao usuário.

Informações são exibidas ao usuário ao longo de toda a interação com interfaces do sistema. Isso torna necessário que elementos de cada interface sejam exibidos de forma distintiva, simplesmente para captar a atenção de quem interage e mantê-lo estimulado até que as funcionalidades sendo realizadas finalizem [Wickens e Hollands, 2000]. É possível chamar a atenção do

usuário por meio de várias técnicas, enumeradas a seguir na forma de *guidelines* [Shneiderman e Plaisant, 2009]:

- **Intensidade dos elementos.** Varie a intensidade de elementos na tela. Entretanto, é recomendável que apenas dois níveis de intensidade sejam usados em uma mesma tela.
- **Marcação.** Sublinhe elementos, coloque-nos bordas, aponte-os com setas ou com indicadores que não fazem parte do texto, como asteriscos, numerais e outros símbolos.
- **Tamanho.** Use até quatro tamanhos diferentes em uma mesma tela. Evidentemente, os elementos de maior tamanho chamarão mais atenção.
- **Escolha das fontes.** Use até três fontes diferentes para escrever mensagens.
- **Efeito de vídeo inverso.** Inverta a cor do fundo com as do componente.
- **Efeito pisca-pisca.** Use esse efeito em componentes pequenos e em locais secundários da tela.
- **Cores.** Use até quatro cores diferentes, exceto quando o propósito da interface depende do uso de um número maior de cores. Imagens coloridas não são consideradas nessa contagem.
- **Áudio.** Use sons sutis para indicar respostas positivas às ações do usuário e sons mais chamativos para alertar sobre condições adversas.

O último fator de preocupação identificado por Shneiderman e Plaisant [2009] é o de facilitação de entrada de dados. Esses autores acreditam que a entrada de dados seja uma das atividades que mais demanda tempo dos usuários e que, caso seja realizada de forma incorreta, causa frustrações que podem fazer com que o usuário desista de usar o sistema. As *guidelines* elaboradas para amenizar os erros nesse contexto são:

- **Consistência das transações que exigem entrada de dados.** As condições de entrada de dados podem mudar ao longo das interfaces, mas as sequências de ações necessárias para que o usuário realize a tarefa devem ser semelhantes no que se referem a delimitadores, abreviações, espaçamentos, dentre outros fatores.
- **Minimização do número de ações a serem realizadas pelo usuário.** As interfaces devem ser criadas para que os usuários finalizem a entrada dos dados necessários com um número mínimo de interações, a fim de que ele não se canse. Substituir ações de digitação de frases por comandos de cliques em opções e sentenças pré-definidas são formas de atingir esse objetivo. Contudo, essa pode ser uma solução ruim caso o usuário precise mover a mão para outro dispositivo que esteja distante. Em casos de usuários com boa perícia da tecnologia sendo utilizada, a digitação de 6 a 8 caracteres é priorizada em relação às atividades de cliques de mouse ou *joystick*, por exemplo. Em casos que uma mesma informação deve ser fornecida pelo usuário em diferentes locais da interface, o sistema deve copiar a primeira resposta dada para a localização corrente do formulário.

As *guidelines* de minimização da carga de memória do usuário, compatibilidade entre a entrada de dados e o conteúdo exibido e flexibilidade do acesso e uso das informações devem ser repensados no fator de facilitação de entrada de dados, porque eles também estão fortemente relacionados a ele. A documentação por *guidelines*, embora sirva como uma boa base para *designers* criarem interfaces que previnam erros com base nas experiências de outrem, deve ser

analisada minuciosamente de acordo com os usuários e com as funcionalidades a serem realizadas no sistema computacional, para que eventuais conflitos sejam desfeitos [Preece, 1994]. A regra de consistência, por exemplo, é importante do ponto de vista de facilidade de aprendizagem, mas pode ser considerada irrelevante para usuários avançados do sistema [Preece, 1994].

A identificação do perfil do usuário é uma tarefa a ser confrontada com as *guidelines* propostos porque os indivíduos possuem suas peculiaridades, que podem variar inclusive entre pessoas de uma mesma região ou cultura. Por exemplo, algumas pessoas preferem visualizar dados tabulados a enxergá-los sob uma perspectiva gráfica; outras preferem lidar com palavras a lidar com dados numéricos.

Além do conhecimento sobre as preferências dos usuários que devem ser suportadas pela interface para promover flexibilidade de uso, é importante que os *designers* conheçam o grau de perícia do usuário com a aplicação sendo desenvolvida. Existem usuários que não sabem usar menus com dados apresentados em hierarquias, por exemplo. Um usuário idoso que esteja enviando um email para o seu neto pela primeira vez terá um nível mínimo de perícia de utilização dos componentes de interface, de forma que ela deverá permitir que ele tenha fácil acesso a um conjunto de instruções que o auxiliem a utilizar o sistema. Para usuários novatos, o número de ações para completar atividades deve ser muito pequeno e os formulários devem ser preenchidos de forma gradual com instruções, para que ele se acostume com a forma pela qual deverá fornecer os dados requeridos, a qual deverá ser mantida ao longo de todas as interfaces da funcionalidade [McGrenere et al., 2002].

A definição das funcionalidades a serem suportadas pelo sistema é importante por conta de as funcionalidades definirem o conjunto de tarefas a serem realizadas pelo usuário. Portanto, as funcionalidades permitem que a consistência de navegação seja planejada pelos *designers* antes de eles criarem efetivamente a interface. Por meio de entrevistas elaboradas com os usuários, podem-se identificar as funcionalidades mais importantes e se criarem atalhos para resolvê-las. Somente as funcionalidades menos utilizadas poderão exigir uma sequência maior de interações pelo usuário e, com base no conhecimento de usuários e das funcionalidades, os *designers* poderão definir o modo pelo qual se interagirá com a interface (uso de menus, uso da linguagem natural, comandos de voz, preenchimento de formulários e por manipulação direta<sup>1</sup>).

Finalmente, é válido destacar que as regras citadas devem ser analisadas de acordo com as peculiaridades de *hardware* de cada dispositivo computacional. Além de as configurações computacionais variarem bastante, os dispositivos de entrada e saída de dados podem ser distintos entre dispositivos, de modo que uma guideline praticável para um conjunto de aparelhos se torne impraticável para outro conjunto.

### 2.1.2 Diretrizes para criação de interfaces de dispositivos móveis

As pesquisas de usabilidade de interfaces para dispositivos móveis estão alguns passos atrás daquelas relacionadas às interfaces de computadores pessoais, por fatores diversos: a interação em dispositivos móveis é muito peculiar e diferente da interação com computadores pessoais [Robertson et al., 2005]; o mercado de dispositivos móveis aqueceu após o mercado de computadores pessoais já estar consolidado [Ji et al., 2006]; dispositivos móveis possuem características e limitações físicas distintas [Kunjachan, 2011].

Todos os estudos analisados ao longo deste projeto de mestrado que divulgam *guidelines* para criação de interfaces de dispositivos móveis com usabilidade foram publicados pelos fabricantes desses próprios dispositivos móveis. Essas documentações estão integralmente disponíveis em

---

<sup>1</sup>Manipulação direta é uma técnica em que o *designer* consegue criar uma interface que tenha grande similaridade com o contexto de uso no ambiente real. Geralmente se obtém esse efeito quando se consegue mapear graficamente as representações de objetos em ações, por meio de metáforas visuais.

páginas oficiais desses fabricantes na Internet e uma síntese delas será feita neste subcapítulo. Os demais estudos usados como bases teóricas desta dissertação, no contexto de interfaces com usabilidade em aparelhos móveis, são direcionados principalmente à avaliação dessas interfaces, por meio de princípios de *design* e de heurísticas, que serão descritos posteriormente, na Subseção 2.2. As *guidelines* encontradas com o propósito de auxiliar o *design* de interfaces de dispositivos móveis são listadas a seguir:

1. **Destaque a principal atividade da aplicação e garanta os subsídios necessários para que o usuário complete qualquer tarefa.** Quando o foco da aplicação é estabelecido e mantido na tarefa principal da aplicação, o nível de satisfação do usuário tende a ser maior [Apple, 2010]. Em cada tela exibida, verifique se toda informação sendo exibida é necessária no momento para que o usuário finalize a atividade sendo realizada [Apple, 2010] [Microsoft, 2013]. Em caso negativo, verifique se haverá algum momento em que a informação se tornará crítica. Em caso afirmativo, exiba a informação [Apple, 2010]. Toda informação necessária para completar uma atividade deve estar visível na tela [Blackberry, 2012]. Além disso, a estrutura do conteúdo deve ser agrupada em porções pequenas e homogêneas de informação [Grasso e Roselli, 2005].
2. **Invista os maiores esforços nos fatores da aplicação mais importantes do ponto de vista do usuário.** Em um jogo de videogame, os usuários não estão interessados em gerenciar contas pessoais ou em gerar e compartilhar conteúdos, mas sim em interagir com o jogo com baixos tempos de resposta por meio de um roteiro que os fascinem e por meio de cenários atrativos. Conteúdos que prendem a atenção de usuário podem ser destacados por meio de efeitos de esmaecimento de controles que não estão sendo usados no momento [Apple, 2010]. Por exemplo, se o usuário estiver vendo uma imagem no seu dispositivo móvel, os botões de interação com aplicação podem desaparecer após certo tempo sem atividade do usuário, e reaparecerem somente quando a tela do dispositivo é tocada.
3. **Pense no *design* da interface como uma atividade a ser preenchida de cima para baixo.** Em *smartphones*, o topo da tela é o local mais visualizado pelo usuário, porque a interação com o aparelho ocorre enquanto o usuário o segura: com a mão não dominante, enquanto interage com um dos dedos da outra mão; com mão dominante enquanto interage com o polegar dessa mão; entre as duas mãos, enquanto interage com ambos os polegares [Hayhoe, 2001]. Por conta disso, as informações genéricas mais importantes devem ser dispostas no topo de modo que, à medida que o usuário desliza a tela para baixo, ele encontra informações mais específicas [Apple, 2010]. Não disponibilize as funcionalidades mais frequentemente usadas na parte inferior da tela, tampouco em locais que não estejam visíveis sem que o usuário deslize a tela para baixo [Blackberry, 2012]. Idealmente, toda a informação deve ficar visível na tela de uma só vez (Grasso e Roselli, 2005). A quantidade de operações de deslizamento para realizar tarefas deve ser limitada, para que o usuário não se sinta incomodado [Hayhoe, 2001].
4. **Disponibilize um caminho lógico para o usuário.** Usuários gostam de saber em que ponto estão da aplicação, por meio de caminhos apresentados na tela. Opções de “voltar” são sempre importantes e, sempre que possível, permita que apenas uma sequência lógica seja usada para que uma tela seja acessada [Apple, 2010]. Os relacionamentos entre telas diferentes devem ser feitos por meio de transições visíveis e áreas com propósitos diferentes devem ter aparências distintas [Google, 2012a].

5. **Torne a interação fácil e óbvia.** O usuário deve entender imediatamente o que deve ser feito com o aplicativo por meio da interface. O uso de cores pode ser benéfico, mas o usuário tende a assimilar que cores iguais indicam ações análogas. Datas devem ser exibidas corretamente de acordo com o fuso horário do usuário, para evitar confusões e interpretações erradas [Google, 2012a]. Facilitar o entendimento de texto em dispositivos móveis implica em exibir frases curtas [Apple, 2010]. Use interações padrões do dispositivo para executar funcionalidades que sejam consistentes com tais interações, como a operação de deslizamento lateral para mudar fotos de um álbum. [Blackberry, 2012]. Se possível, explore o uso de áudio para substituir conteúdos textuais extensos [Dimakopoulos e Magoulas, 2009], mas dê ao usuário a opção de desligar o áudio no momento que preferir [Hayhoe, 2001].
6. **Facilite a entrada de dados.** A entrada de dados costuma demandar muito tempo de interação, de forma que, caso o usuário tenha que fornecer uma grande quantidade de dados antes que qualquer ação relevante ocorra, é provável que ele se desestime. Balanceie a quantidade de dados requeridos com o que a aplicação pode fornecer por meio da interface. Utilize o componente correto para cada tipo de informação a ser coletada, para minimizar o tempo de interação e evitar erros do usuário [Apple, 2010]. Os elementos usados na interface devem sugerir ao usuário como eles devem ser usados [Luchini et al., 2002].
7. **Estimule a conectividade e o comportamento colaborativo.** Sempre que oportuno, permita que o usuário compartilhe informações com outras pessoas, tais como a localização atual, opiniões, resultados, dentre outras. A interface pode permitir que o dispositivo se comunique com outros dispositivos móveis [Apple, 2010]. O *design* da interface deve levar em consideração as capacidades tecnológicas dos aparelhos [Grasso e Roselli, 2005] [Hayhoe, 2001].
8. **Torne a interface mais realista possível.** Usuários se sentem mais motivados a interagir com interfaces que se assemelham com artefatos do mundo real. Por exemplo, o aplicativo “agenda” mostrado na Figura 2.3 é de fácil entendimento por parecer com uma agenda de papel. Pense nos objetos do mundo real como oportunidades de comunicação simples e efetiva com o usuário [Apple, 2010]. Sempre que possível, permita que ações sejam assimiladas por meio de metáforas, para facilitarem o entendimento [Blackberry, 2012]. Objetos reais são mais divertidos de serem manipulados pelo usuário, o que torna a interação mais natural por reduzir o esforço cognitivo necessário para realizar as atividades [Google, 2012a]. Um exemplo de metáfora na Figura 2.3 é o botão com o símbolo “+”, que sugere a adição de um novo contato à agenda eletrônica.
9. **Dê suporte à mudança de orientação.** As pessoas possuem diferentes preferências de orientação, que podem se alterar de acordo com a tela com a qual elas interagem. Por conta disso, crie interfaces que mantenham foco nas atividades e nas informações mais importantes, independentemente da orientação do dispositivo móvel (retrato ou paisagem). Em casos que o aplicativo deve funcionar apenas em uma orientação, exibe a interface nessa orientação, mesmo que o dispositivo esteja noutra orientação, para que o usuário entenda que ele deve girar o aparelho para interagir com a interface. A mudança de orientação nem sempre deve significar um simples redimensionamento de conteúdo. Às vezes, ela deve acarretar em uma total reorganização do conteúdo exibido, para atender as expectativas do usuário.

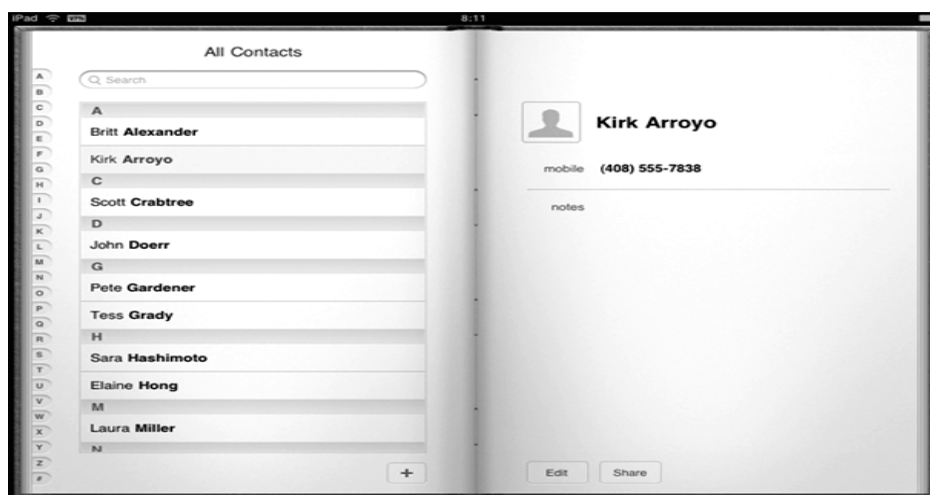
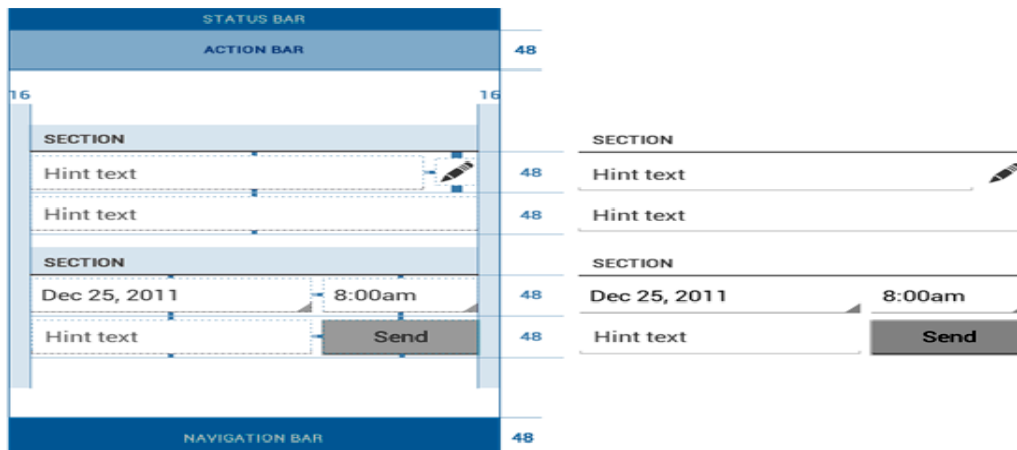


Figura 2.3: Aplicativo “agenda” com interface fazendo analogia a uma agenda de papel [Apple, 2010].

10. **Mantenha o usuário ciente de qualquer ação.** Jamais termine um aplicativo sem avisar o usuário ou sem que ele tenha escolhido explicitamente encerrar a interação, porque ele tenderá a imaginar que a aplicação parou de funcionar [Apple, 2010]. Casos excepcionais em que ações devem ser abortadas devem ser explicados ao usuário por meio de uma interface clara e elegante que não só explane o problema, mas também sugira ações para corrigi-lo [Blackberry, 2012]. Diálogos e informações de alerta devem ser exibidos apenas quando o usuário seleciona uma opção que inicia uma funcionalidade com problemas [Apple, 2010].
11. **Dê controle ao usuário.** A interface deve exibir automaticamente o máximo de informação possível, para evitar que o usuário forneça seus dados dispendiosamente [Apple, 2010] [Google, 2012a]. Entretanto, deixe claro que o usuário está no controle no aplicativo, permitindo que ele explicitamente suas preferências de utilização e que ele desfça operações realizadas recentemente [Blackberry, 2012].
12. **Crie uma página de ajuda.** É importante que haja interfaces específicas para opções de ajuda ao usuário. Os problemas abordados nessa seção devem ser facilmente encontrados por meio de buscas. A interface deve permitir a exibição adequada dos passos necessários para corrigir os problemas [Blackberry, 2012]. Organize os tópicos de ajuda em uma tabela de conteúdos, a partir da qual o usuário possa selecionar um tópico de interesse [Hayhoe, 2001].
13. **Aposte em um design minimalista.** Elabore uma interface simples, mas de fácil entendimento, por meio de um uso balanceado de elementos e de cores. Use gráficos e animações para melhorar o entendimento [Blackberry, 2012], mas os evite no momento da inicialização do aplicativo [Apple, 2010]. Manter a interface simples facilita que diferentes dispositivos acessem o conteúdo de forma menos discrepante [Blackberry, 2012].
14. **Use imagens e gráficos em alta definição e editados profissionalmente.** Dedique um bom tempo de trabalho para a criação do ícone do aplicativo, que aparecerá ao longo de toda a interação, porque ele é a imagem que ficará na mente do usuário associada à aplicação [Apple, 2010]. Edite imagens e gráficos para que gerem impacto visual [Apple, 2010] e surpreenda o usuário com a beleza das animações e imagens utilizadas [Google, 2012a]. Efeitos gráficos inesperados costumam surpreender o usuário e fazer com que ele pense que está usando um aplicativo mais robusto do que a média [Google, 2012a].

15. **Use componentes na medida adequada.** Os componentes que precisam ser acessados com toque precisam ter no mínimo 48dp de altura e de largura, para acomodar o dedo do usuário adequadamente [Google, 2012a]. Segundo Apple [2010], os componentes para dispositivos no sistema iOS devem ter 44x44 pontos<sup>2</sup>. Um exemplo de *design* com base nas medidas propostas por Google [2012a] pode ser visualizado na Figura 2.4.



**Figura 2.4:** Tela criada com elementos com pelo menos 48dp de altura e de largura. O esquema gráfico está à esquerda, enquanto que a interface final está à direita [Google, 2012a].

A preocupação de empresas detentoras de sistemas operacionais para dispositivos móveis divulgarem *guidelines* para construção de interfaces demonstra que essas documentações são importantes para que aplicativos sejam competitivos e desenvolvidos de forma atrativa aos usuários. Entretanto, nota-se que cada conjunto divulgado tem suas particularidades por conta de os sistemas operacionais serem diferentes, no que se refere aos componentes de interfaces. As empresas Apple, Google (empresa responsável pelo Android), Microsoft (responsável pelo Windows Phone), Blackberry e Nokia (responsável pelo sistema operacional Symbian) possuem *guidelines* para uso dos componentes gráficos existentes, os quais possuem nomes e *layouts* diferentes, que dificultam o entendimento da documentação por pessoas que não possuem conhecimentos técnicos de cada sistema operacional. Um mapeamento desses componentes poderá ser contemplado no Capítulo 4 deste trabalho de pesquisa, a fim de que um conjunto de *guidelines* unificado para *design* de interfaces possa ser divulgado.

Por serem imperativos, as *guidelines* são diretos e fáceis de serem interpretados, ainda que difíceis de serem seguidos. Outra limitação deles é a existência de fatores subjetivos, de forma que não se sabe, por exemplo, o grau de sucintez das informações a serem exibidas ou o grau de exuberância das imagens. O bom senso e experiência do *designer* são fundamentais para que esses parâmetros sejam elucidados.

*Guidelines* têm uma relação estreita com heurísticas de avaliação de usabilidade de interfaces e, muitas vezes, são escritas com base em heurísticas. Entretanto, as heurísticas são elaboradas como subsídio para avaliar uma interface já pronta ou esquematizada (no papel, por exemplo) e só podem ser usadas por especialistas, ao contrário das *guidelines*, que podem ser usadas por qualquer interessado em criar interfaces com usabilidade. No subcapítulo seguinte, serão realizadas uma síntese do funcionamento da avaliação heurística e um levantamento das principais heurísticas de usabilidade utilizadas atualmente.

<sup>2</sup>As documentações consultadas não revelam como essas empresas chegaram a esses números.

## 2.2 Heurísticas

A computação enfrenta grandes desafios relativos a metodologias de *design* e de avaliação de interfaces que auxiliem no processo de desenvolvimento de *software* de qualidade [Bertini et al., 2006]. Em particular, os métodos de avaliação de interfaces propõem técnicas diversas por meio das quais se podem medir a usabilidade, sendo que algumas já parecem consolidadas [Kjeldskov e Stage, 2004]. Dessas, podem-se citar: avaliações de desempenho ao se executar atividades [Gonçalves et al., 2011]; avaliações empíricas de tempo [Arhipainen e Tähti, 2003] [Balagtas-Fernandez e Hussmann, 2009] ou de número de teclas acessadas ao longo do tempo [MacKenzie e Zhang, 1999]; entrevistas com usuários [Jones e Marsden, 2006]; questionários distribuídos aos usuários [Väättäjä e Roto, 2010] [Bradburn et al., 2004]; avaliações por meio de percursos cognitivos no contexto de atividade com o usuário [Blackmon et al., 2002]; observações gravadas de interações realizadas em ambientes reais de utilização do sistema [Matera et al., 2006]; simulações [Hagen et al., 2005]; avaliações de regras subjetivas por meio de especialistas [Bertini et al., 2006], dentre outras.

Avaliações de usabilidade de interfaces são processos que visam garantir, por meio da análise dessas interfaces, que o sistema funciona adequadamente e satisfaz as expectativas dos usuários e os requisitos de *software* elicitados nas fases iniciais do processo de desenvolvimento do software. Os objetivos dessas avaliações são examinar a experiência do usuário ao utilizar as interfaces, avaliar a acessibilidade das funcionalidades disponibilizadas e identificar problemas de *design* [Sharp et al., 2007].

Tipicamente, problemas de *design* identificados precocemente são mais baratos e mais fáceis de serem corrigidos do que os identificados em longo prazo [Bertini et al., 2006]. Por conta disso, avaliações periódicas de usabilidade devem ser realizadas ao longo do processo de desenvolvimento, cada vez que uma nova versão da interface tiver que ser lançada (por conta de mudanças de requisitos de sistema, de mudanças de usuário ou de adição de funcionalidades). A maioria das avaliações de usabilidade conhecidas é realizada na presença dos usuários finais da aplicação. Nesse modelo de avaliação, podem-se identificar parâmetros importantes sobre o modo pelo qual o usuário final utiliza o sistema, porque os experimentos são conduzidos no ambiente real de utilização da interface. Contudo, eles costumam demandar muito tempo para serem planejados e os custos de realização dessas avaliações costumam ser maiores [Kantore, 2011]. Em casos de questionários serem usados na avaliação, ainda existe o problema de formulação correta das questões, para que as respostas obtidas sejam relevantes para o objetivo da avaliação [Rothwell, 1993] [Brace, 2004] [Siniscalco e Auriat, 2005]. Apesar de os estudos sobre avaliações de interfaces continuarem em aberto, há um consenso de que um dos métodos de avaliação mais baratos e rápidos, e que obtêm resultados relevantes, é a avaliação heurística [Bonifácio et al., 2010].

A avaliação heurística é realizada por meio de um conjunto pequeno de especialistas de *design* que, separadamente, avaliam a interface confrontando-as com regras, conhecidas como “heurísticas”, para identificar eventuais erros de *design* que comprometam a usabilidade. Acredita-se que de três a cinco avaliadores especialistas sejam capazes de identificar de 75% a 80% dos problemas de usabilidades de interfaces computacionais [Nielsen e Molich, 1990] [Nielsen, 1994]. Provavelmente o conjunto de heurísticas mais conhecido pelos estudiosos de métodos de avaliação de usabilidade de interfaces seja o elaborado por Nielsen [1994], e segue:

1. **Visibilidade do status do sistema.** O usuário deve estar completamente informado do que está acontecendo, por meio de *feedback* imediato da interface.
2. **Compatibilidade do sistema com o mundo real.** A terminologia deve ser adequada à

linguagem do usuário e não orientada ao sistema. As informações devem ser organizadas de acordo com o modelo mental do usuário. Segundo Helander [1997], o modelo mental se refere à expectativa que um usuário possui em relação ao comportamento do computador.

3. **Controle e liberdade do usuário.** Disponibilize “saídas de emergência” ao usuário, para que ele possa desfazer ou refazer ações, a fim de que ele se situe em um ponto recente da interação.
4. **Consistência e padrões.** Nunca identifique uma mesma ação por ícones ou metáforas diferentes. Elementos similares devem ser usados para propósitos semelhantes, assim como funcionalidades semelhantes devem possuir uma sequência de ações semelhantes.
5. **Prevenção de erros.** Idealmente, interfaces não precisam de mensagens de erro por serem capazes de prevenir que erros ocorram. Ações definitivas podem ter um tratamento anterior para que o usuário as confirme por meio de checkboxes, por exemplo.
6. **Reconhecimento ao invés de lembrança.** O usuário não deve precisar memorizar o que está realizando. Permita que a interface atue como um meio de dialogar com o usuário, em tempo de execução.
7. **Flexibilidade e eficiência de uso.** O sistema deve ser fácil de ser operado por usuários novatos, mas também robusto o suficiente para permitir eficiência de uso a usuário avançados. Teclas de atalho e de comandos por voz podem ser alternativas para tornarem a interface flexível. Noutras palavras, flexibilidade implica em permitir que uma mesma funcionalidade seja realizada por comandos distintos.
8. **Estética e design minimalista.** As informações devem ser sucintas e não devem informar mais do que os usuários necessitam para realizar a funcionalidade corrente. Os diálogos do sistema precisam ser diretos e naturais e devem aparecer nos momentos adequados.
9. **Ajuda para usuário identificar, diagnosticar e corrigir erros.** As mensagens de erros devem ser claras e simples e não podem intimidar o usuário. Ao contrário, devem estimulá-lo ao oferecer formas de corrigir o erro.
10. **Ajudas (Help) e documentação.** Um bom *design* evita que o usuário tenha que usar opções de ajuda com frequência. Entretanto, é fundamental que o sistema possua telas específicas de ajuda, para orientar o usuário em casos de dúvidas.

Nitidamente, existe uma dependência estreita entre as heurísticas criadas por Nielsen [1994] e as diretrizes divulgadas por empresas detentoras de sistemas operacionais para dispositivos móveis. Entretanto, é válido ressaltar que as heurísticas apresentadas não foram criadas tendo-se a usabilidade de interfaces para computação móvel em mente.

A heurística que trata a compatibilidade do sistema com o mundo real cita que o *design* deve ser elaborado com base no modelo mental do usuário. O modo de pensar do ser humano é objeto de estudo de muitas áreas de pesquisa, não só da Interação Humano-Computador, e fundamental para se entender como se deve desenvolver interfaces com foco no usuário. Rocha e Baranauskas [2003] explicam que humanos compartilham muitas características físicas e psicológicas, mas são bastante heterogêneos em termos de qualidades como habilidades cognitivas e motivação e acrescentam que essas diferenças individuais têm importância fundamental no *design* da interface de um sistema computacional.

Segundo Sutcliffe [1995], modelos mentais podem tanto ser físicos como conceituais. Os primeiros descrevem o relacionamento de objetos no mundo real em termos de distribuição espacial de eventos em um período. Podem ser visualizados, especialmente se o problema envolve raciocínio espacial. Os modelos conceituais, por sua vez, existem em diferentes manifestações. São expressões linguísticas superficiais e em uma linguagem interna que, embora baseada na linguística, representa uma abstração futura. Modelos conceituais seriam, portanto, uma espécie de linguagem mental interna que representa valores reais sobre objetos e suas relações. Norman [1988] sugere que a forma dos modelos mentais difere entre pessoas e depende de estilos cognitivos pessoais e essa é a principal dificuldade ao se tratar de processos mentais para *design* de interfaces com usuários.

?) destacam que, em particular, o termo ora se refere ao modelo que o usuário tem do sistema, outrora ao modelo que o projetista tem do sistema, e outras vezes ao modelo que o projetista ou o sistema tem do usuário. Portanto, modelo mental do usuário compreende o modelo do sistema, formado pelos usuários, através de experiências e interações com o sistema e a partir de sua imagem do sistema.

Segundo Bertini et al. [2006], o conjunto de heurísticas elaborado por Jacob Nielsen é utilizado como forma de precaver erros de usabilidade e satisfazer requisitos de qualidade de interfaces em vários domínios de aplicação. Por conseguinte, é esperado que tais heurísticas também sejam usadas como base científica para avaliação de interfaces de dispositivos móveis. Entretanto, os autores ressaltam que, por conta de a interação nesses dispositivos ser muito peculiar, o ideal é que se usem heurísticas apropriadas para eles. Por conta dessa necessidade, Bertini et al. [2006] elaboraram regras próprias para esse fim. O conjunto de heurísticas elaborado por Bertini et al. [2006] como refinamento das regras de Nielsen e adaptação delas para o contexto de interfaces de computadores móveis contém oito regras, e segue:

1. **Visibilidade do status do sistema e facilidade de encontrar o dispositivo móvel.** O sistema deve sempre manter o usuário informado sobre o que está ocorrendo. Além disso, o sistema deve dar prioridade a mensagens relativas a aspectos críticos do sistema, como capacidade da bateria, condições do ambiente de utilização e informações de conectividade. Tendo em vista que dispositivos móveis são perdidos com certa facilidade, medidas de encriptação de dados devem ser consideradas e o sistema deve prover formas de o usuário encontrar o dispositivo, caso ele tenha sido esquecido fora de lugar.
2. **Compatibilidade entre o sistema e o mundo real.** Permita que o usuário entenda a informação sendo exibida de forma correta, por meio de uma disposição de elementos em ordem natural e lógica. Sempre que possível, o sistema deve permitir identificar condições ambientes locais e informações de uso automaticamente e exibi-las de forma adequada ao usuário.
3. **Consistência e mapeamento.** O modelo conceitual que o usuário possui acerca da relação entre função e interação deve ser consistente com o contexto de utilização. É crucial que haja um mapeamento adequado entre ação a ser realizada e modo de realizar esta mesma ação no mundo real.
4. **Boa ergonomia e design minimalista.** Dispositivos móveis devem ser fáceis de manusear com apenas uma das mãos e ser resistentes a degradação por ações do ambiente, como umidade. Além disso, nenhuma informação desnecessária deve ser exibida ao usuário.
5. **Facilidade de entrada de dados, legibilidade e capacidade de assimilação.** Os dispositivos móveis devem prover modos simples para que o usuário informe dados de entrada,

preferencialmente sem que o usuário precise usar as duas mãos para executar tal tarefa. A tela deve possuir todas as informações visíveis ao usuário, independentemente das condições de luminosidade do ambiente. Idealmente, o usuário deve ser capaz de assimilar a informação sendo exibida imediatamente.

6. **Flexibilidade, eficiência de uso e personalização.** Permita que os usuários personalizem as ações de acordo com as necessidades deles. Sempre que possível, o sistema deve ser capaz de sugerir ao usuário formas de personalizar ações que porventura sejam benéficas em algum contexto de utilização.
7. **Convenções estéticas, sociais e de privacidade.** Leve em consideração aspectos emocionais e estéticos dos usuários que utilizarão o dispositivo. Assegure que as informações do usuário serão mantidas com segurança e privacidade. As interações devem respeitar convenções sociais dos usuários.
8. **Gerenciamento de erros realístico.** Proteja o usuário dos erros de interação. Se não for possível fazê-lo, permita que o usuário identifique o erro, o diagnostique e, se possível, o corrija. Mensagens de erros devem ser claras e sucintas. Se o erro for irreversível, certifique-se que o usuário entenderá a condição em que ele ocorreu.

Como se percebe, as heurísticas de Bertini não consideram apenas aspectos de usabilidade da interface dos dispositivos, mas também aspectos sistêmicos e de *hardware*, o que difere o trabalho deles do proposto nesta dissertação. Um ponto importante a destacar é o aspecto humano presente na heurística número 7, o qual é citado pelos autores no estudo, mas não aprofundado.

Um trabalho interessante e que aborda heurísticas com base em fatores humanos foi realizado por Moraveji e Soesanto [2012], no qual os autores aliam conhecimentos de anos de estudos empíricos acerca de fatores que estressam o usuário quando ele interage com uma interface de dispositivo móvel aos estudos de Nielsen [1994]. O resultado é um conjunto de dez heurísticas que viabilizam que especialistas avaliem interfaces com base em dois propósitos: usabilidade e potencialidade estressora. Os autores ainda salientam que este é provavelmente o único trabalho com esse propósito que trata de uma metodologia que pode ser seguida na ausência de usuários finais.

Por conta de estresse ser muitas vezes um fator subjetivo, algumas interfaces parecem difíceis de serem avaliadas sob a perspectiva de fatores estressores ao usuário. Entretanto, estudos da Psicologia identificaram alguns padrões que auxiliam a identificação desses fatores, os quais foram sumarizados por Lupien et al. [2007], resultando em quatro características de elementos estressores, as quais serão numeradas de C1 a C4 neste documento.

A primeira característica (C1) levantada por Lupien et al. [2007] é a de que elementos estressores parecem imprevisíveis, incertos ou não familiares em algum contexto. A segunda (C2) deriva de estudos realizados por Henry e Grim [1990] e diz respeito à sensação de que a pessoa está perdendo o controle da situação. A característica seguinte (C3), identificada por Dienstbier [1989], é a propensão de algo causar perda ou dano ao indivíduo, ou seja, o indivíduo perde o sentimento de posse dele por algo. Por fim (C4), Lupien et al. [2007] citam a capacidade de o elemento representar uma ameaça à identidade ou à autoestima do indivíduo, quando julgada do ponto de vista social [Dickerson, 2004].

Ao usar as quatro características estressoras sumarizadas por Lupien et al. [2007] para criar as dez heurísticas para avaliação de interfaces de usuário, Moraveji e Soesanto [2012], assim

como Bertini et al. [2006], acabam sobrepondo algumas heurísticas de Nielsen [1994]. Entretanto, é nítida a preocupação dos autores em elaborarem regras que mitiguem os aspectos estressores das interfaces. As dez regras levantadas por Moraveji e Soesanto [2012] são acompanhadas da listagem das características estressores que elas combatem, e seguem:

1. **Capacidade de controlar interrupções (combate C1 e C2).** Interrupções imprevisíveis comprometem o controle do usuário sobre a interface. Esse é um fator tão importante de ser considerado que existe um ramo da Interação Homem-Computador que estuda especificamente os problemas causados por interrupções [Iqbal e Horvitz, 2007]. Essa heurística visa verificar se a interface dispõe de formas de bloquear, controlar, ou desabilitar temporariamente interrupções durante apresentações, ligações telefônicas, ou durante outras funcionalidades que demandam atenção do usuário. Opções como “Não mostrar essa mensagem novamente” ou “Lembrar-me dessa preferência no futuro” são boas formas de o usuário controlar interrupções.
2. **Não sobrecarregar o usuário (combate C2 e C4).** Grandes quantidades de dados são comuns em aplicativos usados por muitos usuários ou que possuam dados armazenados em servidores *Web*. Nesses casos, o estresse do usuário está em sentir que ele não é capaz de lidar com a quantidade de informação, de modo que a interação parece que não terá fim. O usuário pode imaginar que não está usando o aplicativo da forma correta, que não está conseguindo se comunicar com outros usuários (em casos de programas colaborativos) ou que não ofereceu a quantidade de dados suficiente para o aplicativo funcionar. Ademais, o usuário pode se sentir sobrecarregado por telas com muitas opções para serem visualizadas e/ou preenchidas. Em aplicativos de email, por exemplo, deve-se evitar mostrar várias mensagens não lidas de uma só vez.
3. **Considerar a percepção de tempo dos seres humanos (combate C1 e C2).** A barra de progresso indica ao usuário o tempo a transcorrer de uma determinada atividade, do ponto de vista do sistema. Entretanto, a percepção de tempo humana não é linear, porque o indivíduos tendem a imaginar que o tempo passa mais devagar quanto mais eles esperam por algo [Harrison et al., 2007]. Esse comportamento da barra de progressos pode gerar imprevisibilidade e falta de controle do momento em que o usuário poderá usar o sistema. Uma forma de amenizar esse problema é permitir que processos mais longos iniciem o processamento mais cedo, em paralelo com outras interações mais leves. Outra alternativa é distrair o usuário com elementos lúdicos ou informações adicionais que sejam interessantes.
4. **Usar entonação apropriada (combate C1 e C4).** Usuários simpatizam com aplicativos com comportamento baseados em polidez e reciprocidade [Nass et al., 1994]. Quando o sistema não é capaz de gerar esse comportamento, os usuários tendem a se sentirem constrangidos pelo tom das mensagens. Por conta disso, é importante que os *designers* se valham de comunicação que se aproxime da humana, em termos de entonação e de emoção. Alguns exemplos incluem uso de mensagens bem-humoradas e que fazem apologia ao aplicativo, como se pode visualizar na Figura 2.5. Pedidos com polidez devem ser preferidos às mensagens imperativas que indicam obrigatoriedade de ações.
5. **Prover *feedback* positivo à entrada do usuário e eventos de interação (combate C1 e C4).** *Feedback* negativo pode causar estresse no sentido de que os usuários podem sentir que não são capazes de fornecer ao sistema as informações das quais ele necessita. Esse tipo de *feedback* tende a negligenciar a forma natural de conversação entre indivíduos, o



Figura 2.5: Exemplo de mensagem de erro bem-humorada.

que pode causar estranheza. O sistema pode informar o usuário de que o problema ocorrido é comum, com mensagens como “Outras pessoas cometeram este mesmo equívoco. Sem problemas.” e mostrar mensagens positivas quando o usuário completa uma determinada tarefa, como “Obrigado por preencher nosso formulário.”.

6. **Estimular interações sociais (combate C4).** Dispor de elementos simples como botões de “curtir” ou “retwitter” ajudam a usuários quebrar barreiras de interação social, por serem diretas e estarem embutidas na própria interface.
7. **Amenizar a pressão ao usuário em relação ao tempo (combate C1, C2 e C4).** A maneira pela qual o usuário se sente estressado por conta da pressão imposta pelo tempo é bem documentada por Maule e Hockey [1993], mas ainda não foi aplicada a dispositivos computacionais de modo geral, tampouco para dispositivos móveis. Ao contrário da heurística 3, que trata do tempo enquanto o usuário aguarda, essa heurística considera o tempo enquanto o usuário interage por meio da interface. Usuários tendem a se incomodar e perder o controle da situação quando são pressionados por tempo para realizarem as atividades. Em contextos competitivos, eles podem se frustrar ao pensarem que estão realizando as tarefas mais demoradamente do que outros usuários.
8. **Usar elementos naturalmente calmantes (combate C1).** Chamar a atenção ou gerar fascínio ao usuário de forma involuntária por meio de elementos naturais tende a fazer com que ele se sinta bem, porque os níveis de atividade fisiológicas do indivíduo são alterados de forma positiva [Ulrich et al., 1991]. Essa heurística valida se a interface embute elementos naturais (sons, imagens ou animações) de forma que o usuário as assimile e as utilize com facilidade.
9. **Esclarecer as ações ao usuário (combate C1 e C2).** Em cada interface navegada, o usuário espera poder resolver uma determinada atividade que, caso esteja desabilitada, pode gerar irritação e estresse. Nesses casos em que o sistema não pode disponibilizar uma determinada funcionalidade, a interface deve esclarecer ao usuário o motivo pelo qual a funcionalidade não pode ser realizada, ou sugerir formas alternativas de o usuário realizar a atividade.
10. **Desmistificar a interface.** Usuários podem se estressar quando interfaces o apresentam uma miríade de escolhas a fazer de uma única vez. Além disso, sugerir ao usuário que ele procure ajuda no sistema pode prejudicar a autoestima dele. De modo a complementar a

heurística de “Ajuda e Documentação” proposta por Nielsen [1994], o sistema deve desmistificar os elementos da interface ao usuário, antes que ele tenha que recorrer à opção de Ajuda, como pode ser visto na Figura 2.6.

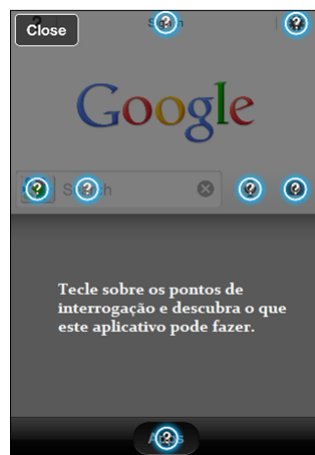


Figura 2.6: Exemplo de interface desmistificada [Moraveji e Soesanto, 2012].

O ideal seria que os especialistas envolvidos na avaliação tivessem conhecimentos de aspectos fisiológicos humanos, que muitas vezes não são dominados por profissionais da Computação. De qualquer forma, alguns aspectos dos estudos elaborados pelos autores podem ser de grande importância para a criação de heurísticas mais completas para avaliação de interfaces de dispositivos móveis. Espera-se, nos Resultados desta dissertação, que as heurísticas propostas aliem os pontos positivos de cada heurística apresentada neste subcapítulo com os demais princípios de usabilidade consolidados na Literatura Científica em Interação Humano-Computador, as quais serão apresentadas no próximo subcapítulo.

### 2.3 Outros princípios de *design*

Princípios de *design* são, a rigor, conjuntos de *guidelines* genéricos para guiar o *design*. Segundo Preece [1994], *guidelines* podem ser tanto regras específicas a serem seguidas, quanto conjuntos genéricos de princípios a serem aplicados nos mais diversos domínios. Normalmente, regras específicas para criação de interfaces com usabilidade derivam de princípios de *design* mais genéricos, apesar de a relação entre as heurísticas apresentadas na subseção 2.2 e os princípios de *design* consagrados em Interação Humano-Computador não ser muito especificada em cada heurística. Neste subcapítulo serão apresentados alguns dos princípios de *design* mais conhecidos nessa área de pesquisa.

Dix et al. [2004] dividem os princípios de *design* de sistemas interativos, que incluem não só a interface com o usuário, mas também aplicativos comerciais de *software* e dispositivos computacionais [Santos, 2008], em três categorias: aprendizibilidade, flexibilidade e robustez. Para cada categoria, os autores enumeram um conjunto de princípios menos genéricos que as compõem. O princípio de aprendizibilidade se refere à capacidade de o sistema interativo permitir que o usuário aprenda a utilizá-lo em um primeiro momento, para que ele seja capaz de explorar todas as funcionalidades do sistema com relativa facilidade. Como o tempo de aprendizagem varia de acordo com o usuário, normalmente a aprendibilidade é analisada de acordo com a eficiência de execução de atividades pelo usuário [Tullis e Albert, 2010]. Os cinco princípios específicos que compõem a aprendizibilidade são:

1. **Previsibilidade.** O desejo maior dos usuários de sistemas interativos é obter do sistema

as respostas que eles desejam em relação à ação realizada por eles. Geralmente, esse objetivo pode ser alcançado por meio de um planejamento de quais atividades contemplarão um determinado anseio dos usuários. Em geral, as pessoas são capazes de prever ações de acordo com determinadas vivências que elas tiveram no mundo real. Estendendo-se esse comportamento humano ao contexto de interação entre pessoas e sistemas interativos, diz-se que sistemas com boa previsibilidade são os que deixam claro ao usuário as consequências das ações realizadas por eles.

2. **Capacidade de sintetização.** Depois que o usuário executa uma determinada ação, é indispensável (e crítico) que o sistema o retorne algum *feedback*. Existem várias formas de se realizar esse *feedback*, mas é imprescindível que ele seja confiável e efetivo no sentido de que o usuário perceba claramente que mudanças ocorreram no sistema.
3. **Familiaridade.** Esse princípio diz respeito à capacidade de o sistema criar elos entre objetos ou situações exibidas ao usuário com objetos ou situações conhecidas por ele no mundo real.
4. **Consistência.** O uso de elementos de entrada/saída deve ser análogo para situações análogas. O mesmo vale para o comportamento de todo o sistema para situações parecidas.
5. **Generalização.** Esse princípio se refere à capacidade de o usuário tomar porções pequenas do sistema e generalizá-las, para deduzir funcionalidades mais amplas. É possível que em uma aplicação para desenhar quadrados, o usuário generalize a opção de desenho para tentar desenhar retângulos, por exemplo.

O conceito de flexibilidade está relacionado à multiplicidade de formas com que o sistema e o usuário trocam informações, ou seja, o usuário deve dispor de caminhos alternativos para chegar a um mesmo resultado. Dix et al. [2004] enumeram os cinco princípios a seguir como formadores do princípio de flexibilidade:

1. **Iniciativa de diálogo.** É evidente que todo sistema deve possuir alguma forma de interagir com o usuário, a qual caracteriza a forma pela qual eles se comunicam. O cerne desse princípio está em decidir se o sistema irá iniciar esse diálogo, ou se essa tarefa será realizada pelo usuário. Não existe recomendação formal que opte por uma abordagem em detrimento da outra. Entretanto, é fundamental que o usuário seja capaz de abandonar, suspender ou continuar uma atividade do sistema a partir de qualquer ponto. É comum que abordagens híbridas de iniciativa de diálogo sejam adotadas em sistemas interativos, para satisfazer requisitos de funcionalidades e garantir flexibilidade de uso.
2. **Multithreading.** Relaciona-se à capacidade de o usuário realizar mais de uma tarefa concomitantemente. Geralmente o termo mais usado em Interação Humano-Computador para designar esse comportamento é “multimodalidade”. Usuários podem realizar diferentes tarefas em locais diferentes, abrindo abas distintas para operações distintas, por exemplo, ou realizá-las em uma mesma tela de um mesmo dispositivo.
3. **Migração de atividades.** Permitir que usuários realizem tarefas rotineiras pode fazer com que eles se desestimulem por perderem a concentração, o que aumenta a probabilidade de erros ao longo da interação. A migração de atividades se relaciona à capacidade de o sistema realizar atividades que corriqueiramente são de responsabilidade do usuário. Esse comportamento economiza tempo e diminui a probabilidade de erros. Novamente, é

necessário que se estime quão migrável pode ser uma determinada funcionalidade, já que muitas vezes não é possível dispensar totalmente as ações realizadas pelo usuário.

4. **Substitutividade.** Esse conceito se relaciona à possibilidade de usar termos equivalentes para valores de entrada/saída como, por exemplo, permitir visualização de temperatura tanto em graus Celsius quanto em Fahrenheit.
5. **Personalização.** Muitos usuários desejam que as interfaces sejam capazes de se adaptarem às suas necessidades. Essa adaptação pode ocorrer tanto por parte do usuário (adaptabilidade) quanto por parte do sistema (adaptatividade). As customizações de sistema mais comuns atualmente são a possibilidade de escolha de características pessoais ou de criação de atalhos para funcionalidades.

O princípio de robustez enumerado por Dix et al. [2004] se refere ao nível de suporte oferecido ao usuário para que ele obtenha sucesso na interação. Esse princípio foi dividido em quatro princípios menores:

1. **Observabilidade.** O usuário deve ser capaz de identificar o estado atual da aplicação rapidamente por meio da representação perceptível da interface. Para tanto, as informações contidas na interface devem ser intuitivas e relevantes.
2. **Recuperabilidade.** Capacidade de o usuário se recuperar de situações de erros ou desfazê-los. O sistema deve permitir que o usuário efetue uma operação de correção sempre que um erro ocorrer. Além disso, o usuário deve ser capaz de voltar a uma situação anterior ao erro. As mensagens de erro devem ser concisas, informativas, específicas e construtivas.
3. **Capacidade de resposta.** Para efeito de estabilidade, o sistema deve garantir o oferecimento de respostas rápidas ao usuário, a fim de que ele não se sinta confuso enquanto realiza uma determinada atividade. Além disso, espera-se que o sistema notifique o usuário de alguma forma, após ele ter solicitado uma determinada ação. Esse comportamento é importante para que o usuário tenha a garantia de que o sistema está cuidando da solicitação por ele enviada.
4. **Conformidade de realização de atividades.** Refere-se ao grau de suporte que o sistema oferece ao usuário em relação às tarefas que ele pode realizar e à maneira pela qual ele pretende realizá-las.

Em um contexto mais genérico, não se limitando apenas ao *design* de sistemas interativos, Williams [2005] elaborou um conjunto de princípios que ele considera básicos em qualquer obra de design, aos quais ele abreviou como CARP: **c**ontraste, **a**linhamento, **r**epetição e **p**roximidade.

O princípio de contraste prega que elementos importantes devem ter contraste em relação aos demais, principalmente em relação ao fundo. O olho humano tem tendência a repousar sobre elementos que estão deslocados do padrão e esse pode ser um fator importante de se explorar quando se elabora uma interface. O contraste de um elemento pode ocorrer por meio da alteração do seu tamanho, da adição de textura, da mudança da sua orientação ou da sua forma, da colorização ou da alteração do posicionamento padrão entre outros.

O princípio de alinhamento atenta para que elementos visuais não sejam dispostos de qualquer forma, mas sim de maneira alinhada. Elementos análogos colocados de forma alinhada facilitam a compreensão porque permitem que o usuário visualize os elementos de forma linear, de forma mais confortável aos olhos. Um exemplo de interface com bom alinhamento de elementos pode ser verificado na Figura 2.7.

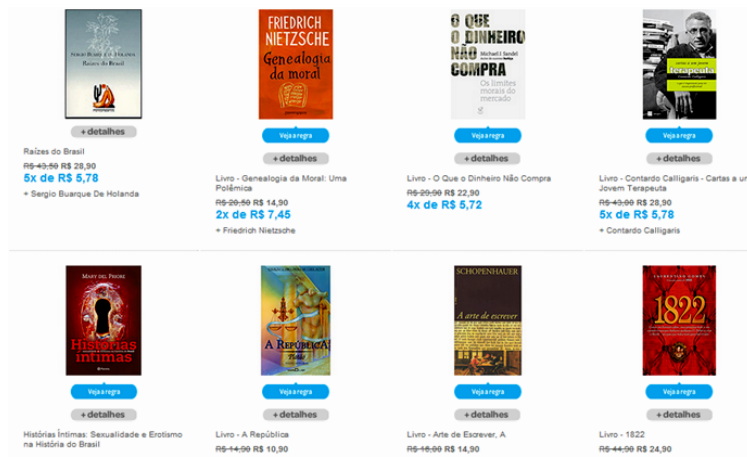


Figura 2.7: Exemplo bem sucedido de aplicação do princípio CARP de alinhamento a uma página Web.

A repetição é utilizada no *design* para criar uma âncora visual. Ela é responsável pela facilidade de se mudar de um elemento para outro, em uma determinada interface. Além disso, elementos repetidos sugerem que eles fazem parte de um mesmo grupo, o que facilita a interpretação. Os blocos com título do texto e resumo, imagens à direita e links no cabeçalho de cada bloco da interface presente na Figura 2.8 formam um exemplo de repetição bem sucedido, porque o usuário consegue identificar os elementos principais de cada mensagem da mesma forma. Por esse exemplo, nota-se que o conceito de repetição tem ligação com o princípio de consistência de interfaces, abordado por Dix et al. [2004].

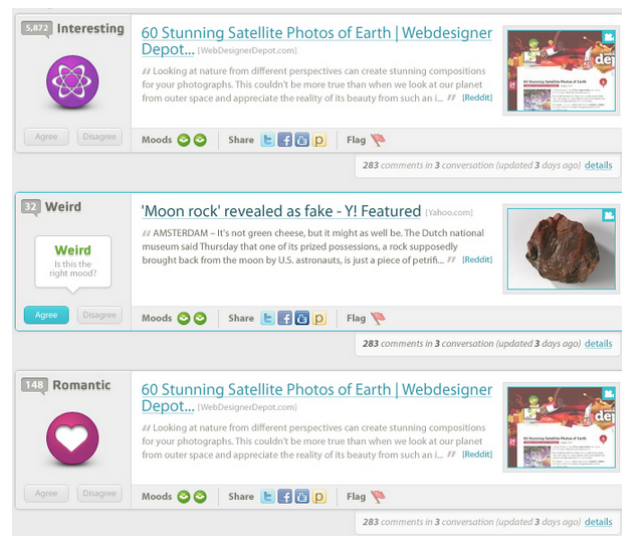


Figura 2.8: Exemplo bem sucedido de aplicação do princípio CARP de repetição a uma página Web.

Por fim, o princípio de proximidade visa garantir que elementos relacionados estejam próximos, ao passo que a distância entre elementos sem relação deve ser maior. Esse princípio ajuda a manter a unidade e ordem do *layout*.

Novamente, os princípios devem ser aplicados em conformidade com o contexto de aplicação. O contraste não pode, por exemplo, interromper o processo natural de leitura de um texto por meio da alteração do tamanho das letras utilizadas em determinadas palavras. O *design* sempre deve levar em consideração as restrições dos usuários e do dispositivo a ser utilizado. A complexidade do processo de criação de interfaces deve ser gerenciada ao longo do processo

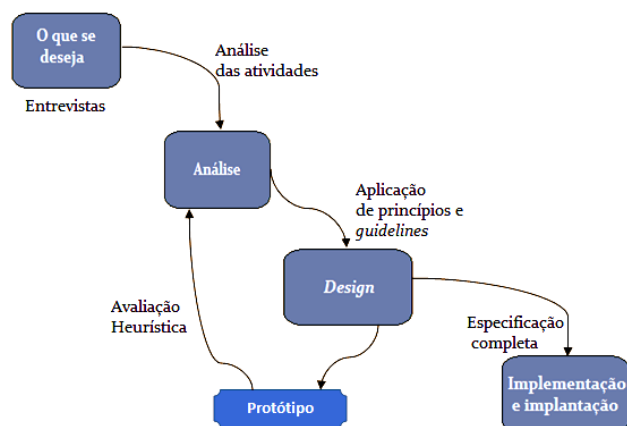


Figura 2.9: Processo de design. [Dix et al., 2004]

de desenvolvimento, por meio de protótipos com interfaces por meio das quais se avalia a usabilidade. Uma sumarização do processo de *design* elaborada por Dix et al. [2004] pode ser visualizada na Figura 2.9.

## 2.4 Considerações finais

Algumas das técnicas para destacar elementos da interface citadas por Shneiderman e Plaisant [2009] podem ser consideradas controversas atualmente, porque a aplicação delas pode atrair a atenção do usuário para pontos secundários da interface. São os casos do efeito de vídeo inverso e do pisca-pisca. Inverter as cores do elemento com o fundo da interface pode causar uma grande estranheza ao usuário e gerar falhas de consistência na interface. O efeito pisca-pisca em elementos pouco sutis da interface causam irritação ao usuário e deve ser evitado.

Shneiderman e Plaisant [2009] explicam que a entrada de dados é uma das tarefas que mais demanda tempo ao usuário, mas essa não é uma informação válida para todos os casos. Alguns aplicativos precisam de poucos dados para funcionarem corretamente e a entrada de dados é uma tarefa rápida de ser realizada. Da mesma forma, a entrada de dados tem relação direta com a experiência do usuário.

As heurísticas propostas neste trabalho têm relação com as recomendações propostas neste capítulo, mas nenhum mapeamento explícito entre as heurísticas e as recomendações é realizado.

---

## Metodologia de Pesquisa

---

Neste capítulo explicam-se as atividades realizadas para que os resultados da pesquisa fossem obtidos. A metodologia de pesquisa inicial para a execução deste trabalho foi exploratória. Trata-se de uma modalidade de pesquisa em que o pesquisador visa aprofundar seus conhecimentos sobre o tema estudado através de leituras de artigos, periódicos e livros relacionados [Mattar, 1999].

A leitura desses estudos é muito importante para a formação teórica do pesquisador e permite que ele enriqueça seu potencial crítico-científico. Tendo em vista que estudos são divulgados ao longo dos anos, é importante que o estudante esteja ciente das atualizações referentes a trabalhos relacionados à pesquisa dele. Logo, a primeira atividade da Metodologia de Trabalho é também a mais extensa e compreende essa análise bibliográfica. Os estudos mais importantes acerca do tema pesquisado estão reportados no Capítulo 2.

Por meio da leitura dos trabalhos relacionados, nota-se que a criação de heurísticas e de diretrizes para construção de interfaces com usabilidade para aplicativos móveis é uma contribuição importante para a Interação Humano-Computador. Entretanto, apenas um conhecimento teórico atualizado acerca do tema de usabilidade dessas interfaces não é suficiente para que se comece a planejar novas heurísticas. A experiência de uso de programas em dispositivos móveis é igualmente importante e serve como base prática para confrontar as teorias aprendidas. Por conta disso, realizou-se um período de experimentação de quatro aplicativos baseados em Android, por um período de um mês, para identificar alguns problemas comuns de usabilidade das interfaces desses aplicativos. A experimentação desses aplicativos foi realizada por inspeção em dois dispositivos móveis distintos, a fim de encontrar problemas de usabilidade e de categorizá-los. Algumas dessas categorias não eram claramente identificadas nas heurísticas de Nielsen e justificavam a criação de novas heurísticas para a avaliação de interfaces de dispositivos móveis.

Em métodos de inspeção de usabilidade os avaliadores examinam aspectos relacionados a usabilidade de uma interface com objetivo de encontrar problemas de usabilidade. Os avaliadores podem ser especialistas em usabilidade, consultores de desenvolvimento de *software*, especialistas em um determinado padrão de interface, usuários finais, dentre outros [Rocha e Baranauskas, 2003].

A versão de heurísticas criada após a etapa de inspeção foi discutida juntamente com outros cinco especialistas de usabilidade em uma sessão de *brainstorming*, a fim de verificar possíveis melhorias do conjunto por meio de modificações das descrições ou de inclusão de princípios de usabilidade que não haviam sido considerados para a proposição das heurísticas iniciais. Em seguida uma nova sessão de *brainstorming* foi marcada com os mesmos especialistas para que uma segunda versão das heurísticas fosse criada.

A segunda versão de heurísticas foi validada por meio de avaliações heurísticas de um aplicativo criado para o sistema operacional Android por cinco especialistas de usabilidade. Para analisar a efetividade dessas avaliações, esse mesmo aplicativo foi avaliado por meio de avaliações heurísticas que usaram as heurísticas de Nielsen, por outros cinco especialistas.

A ordem das atividades realizadas ao longo deste projeto de pesquisa é a seguinte:

1. Inspeção de quatro aplicativos baseados no sistema operacional Android, para identificar problemas de usabilidade, verificar que as heurísticas de Nielsen não permitem identificar todos esses problemas, e propor um conjunto inicial de heurísticas;
2. *Brainstorming* com especialistas em usabilidade para refinar o conjunto inicial de heurísticas e propor melhorias para uma segunda versão;
3. Nova sessão de *brainstorming* com os mesmos especialistas em usabilidade para refinar a segunda versão de heurísticas;
4. Avaliação heurística de um aplicativo de dispositivo móvel baseado no sistema operacional Android em um tablet de 10,1", utilizando as heurísticas tradicionais de Nielsen;
5. Avaliação heurística desse mesmo aplicativo utilizando a segunda versão de heurísticas para avaliação de usabilidade de interfaces de dispositivos móveis, proposta após a segunda sessão de *brainstorming*;
6. Análise dos resultados das avaliações heurísticas para decidir por melhorias no conjunto de heurísticas proposto;
7. Teste desse mesmo aplicativo baseado em Android com usuários finais utilizando o tablet usado pelos especialistas, para confrontar os problemas desses usuários com os problemas de usabilidade encontrados nas avaliações heurísticas realizadas;
8. Avaliação heurística de outro aplicativo para dispositivo móvel em diferentes dispositivos móveis, utilizando as heurísticas para avaliação de usabilidade de interfaces de dispositivo móvel;
9. Criação de diretrizes genéricas para *design* de interfaces de dispositivos móveis com base em seus componentes de entrada.

A listagem de tarefas propostas permitiu dividir o trabalho do projeto em três grandes etapas: proposição das heurísticas para avaliação de interfaces de dispositivos móveis com base no fato de as heurísticas de Nielsen terem se mostrado falhas para este propósito, validação dessas heurísticas, e criação das diretrizes para *design* de interfaces de dispositivos móveis. O detalhamento das atividades realizadas, separados de acordo com cada etapa do processo, será dado nas próximas seções.

### 3.1 Elaboração das heurísticas

#### 3.1.1 Análise de aplicativos para Android

Inicialmente, o mestrando atuou como usuário de quatro aplicativos famosos da loja virtual de produtos para Android ao longo de quinze dias, a fim de listar as dificuldades encontradas acerca da usabilidade da interface. Os aplicativos escolhidos foram instalados no início de agosto de 2012 em dois dispositivos baseados em Android: um *smartphone* Samsung Galaxy 5 I5500 com tela de 2.8" e um *tablet* Motorola XOOM com tela de 10.1", ambos com tecnologia sensível ao toque. Esses dispositivos podem ser visualizados na Figura 3.1.



**Figura 3.1:** Dispositivos usados na fase de uso de aplicativos baseados em Android. À esquerda, *smartphone* I5500 da Samsung. À direita, *tablet* XOOM da Motorola.

Os aplicativos escolhidos para análise foram Facebook, Twitter, Gmail e Foursquare, por conta de o propósito de cada um deles divergir e por conta da popularidade que eles possuem. Na ocasião da instalação, todos esses aplicativos estavam entre os 10 mais instalados por meio da loja de aplicativos Android, o Google Play [Google, 2012b].

A avaliação da usabilidade dos aplicativos foi realizada por inspeção, a partir de listagens das funcionalidades principais de cada aplicativo testado. Essa listagem foi obtida das páginas *Web* oficiais dos aplicativos. A relação entre cada aplicativo e as funcionalidades principais que ele oferece pode ser visualizada na Tabela 3.1. Evidentemente, cada funcionalidade listada abrange uma série de passos necessários para que ela seja realizada. Analogamente, é possível que na mesma tela da funcionalidade existam opções de realizar atividades menores.

O método de inspeção utilizado é conhecido como “simulação de uso”, também chamado de “revisão por especialista”. Nesse método, especialistas simulam o comportamento de usuários com menos experiência a fim de antecipar-se aos problemas de usabilidade que possam encontrar ou de identificá-los à medida que eles aparecem [Preece, 1994].

A primeira semana da quinzena de avaliação caracterizou-se pela utilização dos aplicativos no tablet. É válido destacar que o avaliador já tinha conhecimento acerca do funcionamento das versões *Web* do Facebook, do Gmail e do Twitter antes de testá-los nas suas respectivas versões para Android. Contudo, nenhum dos aplicativos havia sido usado pelo especialista no contexto de dispositivos móveis, de forma que o primeiro dia da análise fora usado apenas para navegação pelos aplicativos, a fim de que se aprendesse a usar as funcionalidades principais. O mestrando já tinha utilizado celulares e *tablets* sensíveis ao toque no momento da avaliação. À medida que problemas de usabilidade eram encontrados, eles eram anotados como forma de documentação, pelo próprio avaliador.

Na segunda semana, as mesmas atividades foram realizadas por meio dos aplicativos instalados no smartphone. A documentação com os problemas encontrados no *tablet* foram revisitados para validar se eles também ocorriam nas interfaces do smartphone.

Ao fim das duas avaliações, obteve-se um documento único com os problemas encontrados, o qual foi analisado de modo a categorizar problemas de usabilidade encontrados. As categorias de problemas encontrados por meio da inspeção por simulação dos aplicativos foram confrontadas com as heurísticas de Nielsen para identificar que havia dificuldade de associar algumas dessas categorias a heurísticas de Nielsen.

### 3.1.2 Acerca dos aplicativos inspecionados

O Facebook é uma rede social que, de acordo com sua página oficial na Internet, conecta mais de um bilhão de usuários desde dezembro de 2012. O objetivo inicial dessa rede social era criar um ambiente colaborativo de envio e recebimento de mensagens, mas ela se expandiu e hoje basicamente qualquer conteúdo multimídia pode ser compartilhado pela rede, além de ser possível criar e navegar por grupos com propósitos específicos e por páginas de eventos. O aplicativo conta ainda com uma opção de “chat” para troca de mensagens instantâneas entre usuários conectados.

O Twitter se define em sua página Web oficial como uma rede de informações em tempo real, por meio da qual os usuários se conectam com as últimas histórias, ideias, opiniões e notícias sobre o que eles achem mais interessantes. Ao contrário do que ocorre no Facebook, os usuários não têm um sistema prático de trocas de mensagens e as mensagens publicadas têm tamanho limitado de caracteres (Twitter Inc, 2013).

O Gmail, por sua vez, é um serviço de correio eletrônico que permite que várias contas sejam gerenciadas simultaneamente, por meio de notificações do próprio aplicativo. É o serviço de email mais popular da loja do Google Play [Google, 2012b].

Por fim, escolheu-se o aplicativo do Foursquare, rede social que permite que pessoas compartilhem informações diversas sobre locais. Essas informações incluem espaço físico, endereço, recomendações dos serviços, custos dos serviços prestados pelo estabelecimento, dentre outros. O ponto forte desse aplicativo é o uso de recursos de localização geográfica do dispositivo (por meio de GPS ou pela Internet) para sugerir locais próximos ao usuário que lhe oferecem serviços dos quais ele gosta. Por meio desse sistema de localização, ele permite reunir pessoas geograficamente próximas.

Como se percebe, os propósitos dos aplicativos são variados: a primeira dá acesso a uma rede social abrangente que permite compartilhar basicamente conteúdo; a segunda é uma rede social que permite compartilhar mensagens rápidas; a terceira é um servidor de emails; a última dá acesso a uma rede social que permite reunir pessoas fisicamente.

**Tabela 3.1:** Relação entre aplicativos avaliados e principais funcionalidades que cada um deles oferece.

<b>Aplicativo</b>	<b>Principais funcionalidades</b>
<b>Facebook</b>	Adicionar um amigo; curtir a postagem de outrem; compartilhar a postagem de outrem; comentar uma postagem; visualizar arquivos de um grupo; visualizar fotos de um amigo; adicionar fotos a um álbum; visualizar postagens no próprio mural; visualizar postagens no mural de outra pessoa; buscar pessoas; usar serviço de chat.
<b>Twitter</b>	Seguir um usuário; visualizar seu próprio mural de informações; visualizar o mural de informações de um usuário que você segue; visualizar o mural de um usuário qualquer; escrever uma mensagem para seus seguidores; escrever uma mensagem para uma pessoa que você segue; buscar informações acerca de uma <i>hashtag</i> ; visualizar assuntos mais comentados na rede.

Continua na página seguinte. . .

**Tabela 3.1 – Continuação.**

<b>Aplicativo</b>	<b>Principais funcionalidades</b>
<b>Gmail</b>	Visualizar conteúdo dos emails; marcar email como não-lido; remover email; arquivar email; buscar email; encaminhar um email; criar filtros.
<b>Foursquare</b>	Realizar <i>check-in</i> , para informar a própria localização; adicionar amigos; adicionar lugares; adicionar comentários a lugares; visualizar pessoas nos arredores; visualizar lugares nos arredores.

### 3.1.3 *Brainstorming* com especialistas

As categorias de problemas encontrados por meio da inspeção por simulação dos aplicativos motivaram a proposição de heurísticas para a avaliação de interfaces de dispositivos móveis, por conta das dificuldades de se associarem determinadas categorias às heurísticas de Nielsen. As associações categoria/heurística realizadas sem dificuldades resultaram em heurísticas para interfaces de dispositivos móveis parecidas com as heurísticas de Nielsen equivalentes. Por outro lado, as categorias isoladas deram origem a heurísticas específicas com descrições específicas. O resultado desse processo foi um conjunto inicial de heurísticas, o qual foi discutido na primeira sessão de *brainstorming*.

Uma sessão de *brainstorming* é caracterizada pela reunião de um grupo de até 20 pessoas, na qual todos os envolvidos sabem os objetivos da discussão e os problemas a serem resolvidos. Trata-se de uma discussão informal em que todas as pessoas apresentam sugestões e as discutem, a fim de que se obtenham soluções para o problema acordado. O método é muito usado em ambientes corporativos, mas podem ser estendidos ao âmbito educacional [Coyne e Coyne, 2011] deste trabalho.

Os especialistas envolvidos nas sessões de *brainstorming* para a criação das heurísticas tinham conhecimento dos principais estudos de usabilidade conhecidos. Três deles eram doutorandos e dois eram mestrandos no mesmo laboratório de pesquisa do mestrando.

Os primeiros 15 minutos da primeira sessão de discussão foram utilizados na exposição dos resultados das avaliações obtidas pelo uso dos quatro aplicativos nos dois dispositivos móveis citados. Cada heurística proposta foi explicada e discutida entre os membros da sessão para se decidir pela validade dela no contexto desejado. Algumas heurísticas foram fundidas, outras foram mudadas e algumas simplesmente foram eliminadas. Ao fim dessa sessão, os especialistas ajudaram o avaliador com sugestões leitura de novos artigos.

Os princípios abordados nos artigos sugeridos foram usados para elaborar a segunda versão das heurísticas. Em seguida, uma nova sessão de *brainstorming* foi agendada com os mesmos especialistas presentes à primeira sessão. Analogamente ao que já havia sido discutido na primeira sessão, as heurísticas sofreram novas mudanças, que as deixaram preparadas para serem avaliadas com outros aplicativos por outros especialistas. Por “preparadas”, entende-se que as heurísticas foram levadas a um patamar no qual a pesquisa bibliográfica acerca do tema se tornara exaustiva e nenhum problema adicional pudera ser identificado pelos especialistas que participaram da discussão.

Com o fim da segunda sessão de *brainstorming*, finaliza-se a primeira etapa desta Metodologia de Pesquisa.

## 3.2 Validação das heurísticas

### 3.2.1 Avaliações heurísticas

Essa etapa se caracteriza pela aplicação de avaliações heurísticas em aplicativos de dispositivos móveis, para validar a adequação (ou não) das heurísticas propostas para avaliação de

usabilidade de interfaces desses dispositivos. Foram escolhidos dois aplicativos para serem avaliados, sendo que um deles foi desenvolvido no Laboratório de Sistemas *Web* e Multimídia do Instituto de Ciências Matemáticas e de Computação (ICMC), da Universidade de São Paulo em São Carlos, ao passo que o outro foi obtido da loja virtual de aplicativos para Android.

O aplicativo desenvolvido no ICMC tem por objetivo realizar anotações multimídia em vídeo. Ele funciona da seguinte maneira:

1. Identifica-se o usuário que irá navegar pelo aplicativo. Os nomes que aparecem para o usuário são relativos às contas do Google sincronizadas com o dispositivo.
2. Escolhe-se o vídeo que se deseja visualizar.
3. Identificam-se os autores dos quais se desejam ver as anotações desse vídeo.
4. Anota-se o vídeo.

As anotações de um autor são disponibilizadas a outrem caso o autor compartilhe suas anotações por meio do aplicativo. Ao selecionar a opção de compartilhamento, o autor envia um arquivo com suas anotações a um ou mais destinatários, a fim de que eles possam visualizá-las em seus próprios dispositivos móveis.

As anotações realizadas pelo usuário podem ser textuais, de áudio ou de tinta eletrônica. Para realizar uma anotação textual, o usuário tecla sobre uma área de texto, que ativa o teclado do sistema operacional para que a mensagem seja digitada. A anotação de tinta é feita passando-se o dedo na área do próprio vídeo, mas precisa ser habilitada teclando-se sobre o botão “Ink Annotation is Off” que, após a habilitação, passa a se chamar “Ink Annotation is On”. O áudio ambiente é gravado como anotação depois de o usuário habilitar essa funcionalidade no aplicativo, teclando o botão “Audio Annotation is Off”. Todo o aplicativo está em Inglês e não se pode trocar o idioma. As telas desse sistema podem ser visualizadas nas Figuras 3.2, 3.3, 3.4, 3.5 e 3.6.

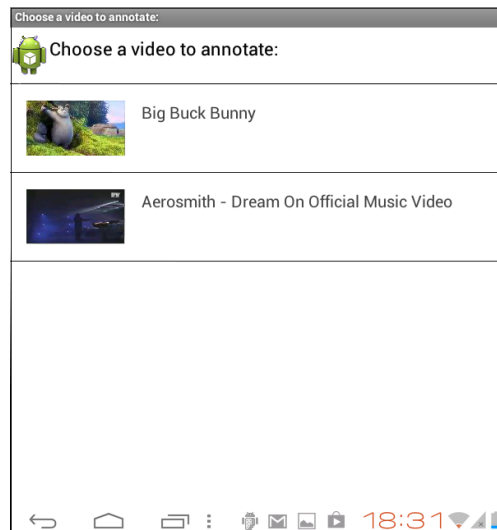


**Figura 3.2:** Tela inicial do aplicativo de anotação multimídia em vídeos.

A avaliação heurística do aplicativo de anotações multimídia foi realizada por dez especialistas em usabilidade, todos com conhecimento do sistema operacional Android e com experiência com dispositivos touchscreen. Cinco especialistas avaliaram o aplicativo usando o conjunto de heurísticas tradicionais de Nielsen, enquanto que os outros cinco o avaliaram usando o conjunto de heurísticas proposto para interfaces de dispositivos móveis. Apenas dois especialistas nunca



**Figura 3.3:** Tela para seleção de usuário que irá navegar pelo aplicativo de anotação multimídia.



**Figura 3.4:** Tela para seleção de vídeo a ser anotado.

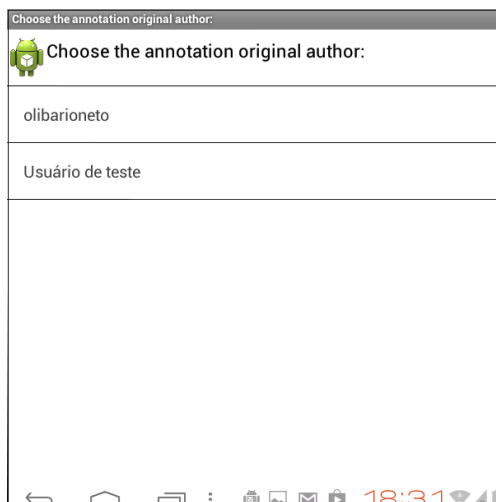
havia usado um tablet. Desses dois, um avaliou o sistema usando as heurísticas de Nielsen e o outro o avaliou com novas heurísticas.

As sessões de avaliação foram agendadas individualmente com cada especialista. Antes de iniciar a avaliação, explicava-se sobre o aplicativo a ser avaliado, deixava-se o especialista navegar pelas interfaces e, então, uma tabela contendo as heurísticas a serem usadas e suas respectivas descrições era oferecida ao especialista. Os problemas reportados pelo especialista ao longo da navegação pelas telas do sistema, bem como as heurísticas com as quais esses problemas se relacionavam, eram anotados pelo mestrando em uma tabela.

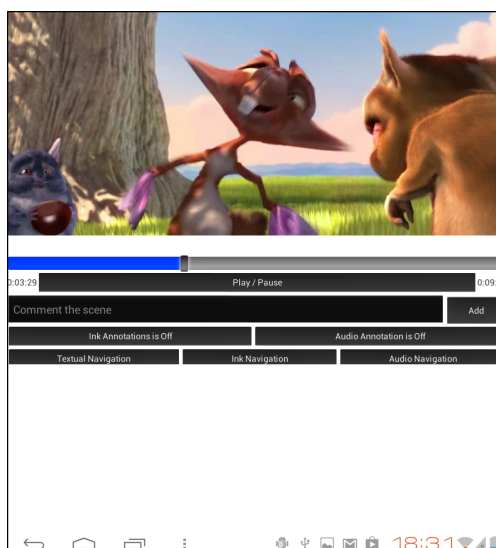
Os especialistas ficaram livres para executarem a avaliação no tempo que quisessem. Caso eles não tivessem passado por todas as telas no aplicativo, o mestrando intervinha e os mostrava os fluxos de atividades que remanesciam inexplorados, para garantir que todas as interfaces seriam visualizadas pelo especialista.

Todos os problemas reportados pelo especialista foram anotados pelo mestrando. Após cada avaliação, o especialista era convidado pelo mestrando a responder, para cada problema encontrado, o grau de severidade associado a ele. O grau de severidade é uma escala de 0 a 4 elaborada por Nielsen [1994], que visa aferir a gravidade dos problemas encontrados de acordo com as seguintes descrições:

1. Eu não concordo que isso seja um problema de usabilidade;
2. Problema cosmético: deve ser corrigido apenas em caso de haver tempo extra no projeto para tal;
3. Problema de usabilidade menor: a correção desse problema é uma tarefa de baixa priori-



**Figura 3.5:** Tela em que o usuário escolhe o autor do vídeo a ser anotado.



**Figura 3.6:** Tela principal da aplicação de anotação de vídeos.

dade;

4. Problema de usabilidade maior: é importante que esse problema seja corrigido e tratado com grande prioridade;
5. Catástrofe de usabilidade: é obrigatório corrigir o problema antes que o produto seja liberado.

Segundo Nielsen [1994], não se pode ocupar o especialista questionando-o acerca dos graus de severidade de cada problema encontrado durante a avaliação heurística, porque ele tende a perder a concentração no processo e encontra menos erros do que encontraria caso não tivesse que se preocupar em indicar as severidades dos problemas. Por conta disso que se optou por perguntar os graus de severidade apenas depois de o especialista ter terminado a avaliação heurística.

Os resultados obtidos nas avaliações foram separados de acordo com o conjunto de heurísticas utilizado, a fim de que eles fossem comparados. Esse processo de comparação e os resultados obtidos serão expostos no próximo capítulo.

### 3.2.2 Testes com usuários

A análise dos resultados das heurísticas foi importante para identificar se elas poderiam ser melhoradas. Completado esse procedimento, dois experimentos foram realizados com dois usuários da aplicação, que tentaram executar as principais funcionalidades do aplicativo sem nunca tê-lo visto antes.

Um dos usuários escolhido era novato e o outro era intermediário. O usuário novato escolhido nunca havia usado dispositivo móvel sensível ao toque. A única tela com essa tecnologia que ele conhecia era a do terminal de autoatendimento bancário que ele utilizava. Além disso, esse usuário nunca havia usado *tablets* ou smartphones. O usuário intermediário, por sua vez, possuía um *smartphone* sensível ao toque equipado com o sistema operacional Windows Phone e já havia utilizado *smartphones* e *tablets* equipados com Android. Ambos os usuários tinham fluência em Inglês, idioma no qual o aplicativo foi escrito.

O objetivo desses experimentos com usuários é calcular o desempenho e a taxa de erros cometidos pelos usuários ao executar cada uma das funcionalidades principais, a fim de confrontar os problemas encontrados pelos usuários com os problemas levantados pelos especialistas nas avaliações heurísticas. As ações dos usuários foram escritas em um arquivo no cartão de memória do dispositivo, para que as métricas sejam calculadas com exatidão. Além disso, as sessões experimentais foram gravadas em vídeo, para que se identificassem erros que o arquivo não reporta e para que se tivesse noção exata do tempo que o usuário levou para finalizar uma tarefa, já que cada tarefa era ditada pelo avaliador e o tempo desse ditado não pôde ser identificado pelo arquivo na memória do dispositivo, devendo ser descontado da interação para cálculo do desempenho do usuário.

O arquivo de gravação contém informações textuais referentes às ações realizadas pelo usuário em cada momento da interação, tais como cliques sobre botões, realização de anotações, cancelamento de ações, etc. O procedimento para gravação dessas informações, assim como de outras informações necessárias para o cálculo das métricas, foi implementado pelo mestrando e será mostrado no Capítulo 4, assim como a estruturação dos arquivos usados.

O ambiente de experimentação dos usuários foi formado apenas por uma cadeira, na qual o usuário se sentava enquanto tentava realizar as funcionalidades que lhe eram ditadas pelo avaliador, que também realizava a filmagem. O aplicativo já estava aberto na página inicial para que o usuário pudesse utilizá-lo. Antes de se iniciar o experimento, a única informação que o usuário recebeu foi a seguinte:

*“O aplicativo que você irá utilizar tem o propósito de realizar anotações de tinta, de áudio e de texto em um vídeo que você escolher. Por anotação de tinta, entenda um desenho que você irá adicionar a uma determinada cena desse vídeo. Anotações de áudio são gravações de áudio que você poderá fazer em uma determinada cena. Uma anotação de texto se refere a uma legenda textual que você poderá adicionar a uma cena do vídeo escolhido.”*

Ciente do propósito do aplicativo, o usuário era orientado a tentar realizar cada funcionalidade que lhe era requisitada, a saber:

1. Escolher um vídeo qualquer;
2. Inserir uma legenda em uma cena do vídeo;
3. Desenhar sobre uma cena do vídeo;
4. Realizar uma gravação de áudio em uma cena do vídeo;
5. Visualizar a cena que contém a legenda que foi inserida;

6. Editar a legenda feita à cena do vídeo;
7. Inserir um novo desenho sobre uma cena;
8. Realizar uma nova gravação de áudio;
9. Ouvir um dos áudios gravados;
10. Apagar o último desenho realizado;
11. Apagar a legenda inserida;
12. Apagar um dos áudios gravados.

### 3.2.3 Avaliações em outros dispositivos móveis

O mestrando não teve contato com o aplicativo de anotações no momento da criação das heurísticas. Contudo, entende-se que a fundamentação científica das novas heurísticas se tornaria mais forte caso outro aplicativo fosse avaliado em diferentes dispositivos, inclusive de diferentes sistemas computacionais. O estudo de Bertini et al. [2006], por exemplo, revelou que as heurísticas propostas por esses autores revelaram mais problemas que as heurísticas de Nielsen, mas não se consegue concluir se as heurísticas foram elaboradas com foco no propósito do aplicativo avaliado, o que certamente teria acrescentado viés aos resultados.

Por conta disso, terminada a etapa de experimentação com o usuário, passou-se para a fase de realização de uma avaliação heurística usando as heurísticas novas para analisar as interfaces de um aplicativo externo ao laboratório de pesquisa do ICMC, consolidado em versões para dispositivos móveis. O aplicativo escolhido para essa análise foi o UOL Notícias, criado pelo portal de notícias brasileiro UOL como uma alternativa do usuário para acompanhar as principais notícias do Brasil e do mundo. A escolha dele para o experimento se justifica pela relevância do portal no Brasil e pelo fato de ele possuir versões nos sistemas operacionais iOS, Android e Windows Phone. A popularidade do aplicativo é tamanha que sua versão para Android já foi instalada em mais 500 mil dispositivos equipados com esse sistema operacional [Google, 2012b].

O UOL Notícias foi avaliado com as heurísticas novas nos *smartphones* iPhone 4S com tela de 3,5" e Motorola Blur com tela de 3,7" e no *tablet* Motorola XOOM de 10,1". Após a etapa de avaliação heurística do UOL Notícias, iniciou-se a etapa de criação de diretrizes para a criação de interfaces com bom uso dos componentes existentes para interagir com o usuário. Esse processo será discutido no subcapítulo 3.3.

## 3.3 Elaboração das *guidelines* (diretrizes)

As *guidelines* foram elaboradas utilizando-se as documentações oficiais presentes nas páginas *Web* das empresas detentoras dos sistemas operacionais Android, Blackberry, iOS e WindowsPhone.

Cada uma dessas empresas disponibiliza um conjunto de *guidelines* para o *design* de aplicativos com base no uso adequado dos componentes de interface existentes nos respectivos sistemas operacionais, a fim de que o usuário disponha do máximo de usabilidade possível. Por exemplo, para o *design* de um componente de diálogo com o usuário, a documentação do Android explicita as seguintes recomendações: evite usar barras de títulos em diálogos; mostre ao usuário perguntas completas acerca da atividade a ser elucidada.

É interessante que as empresas produtoras dos sistemas operacionais mais usados da atualidade se preocupem em documentar por meio de recomendações o que se deve e o que não se

deve fazer ao escolher um componente para uma interface. Contudo, nota-se que a terminologia usada para designar esses componentes varia sensivelmente de acordo com cada fabricante, o que dificulta que pessoas sem conhecimento específico em um sistema operacional use os componentes corretos para criar uma interface com o usuário. Além disso, alguns elementos de um dado sistema operacional simplesmente não existem em outro sistema, o que pode levar um *designer* a criar interfaces inconsistentes ou impossíveis de serem implementadas.

Por conta desses problemas, uma análise das documentações supracitadas foi realizada. Para cada elemento abordado em uma documentação, procurou-se o seu equivalente na documentação dos demais sistemas operacionais. Por exemplo, botões são chamados de “Buttons” em todas as referências encontradas. Entretanto, barras de progresso possuem nomes diferentes de acordo com os sistemas operacionais.

Após a etapa de mapeamento dos componentes, realizou-se a fusão das diretrizes para cada componente, para gerar as diretrizes gerais. Houve casos de alguns componentes que só possuíam recomendações em um ou em alguns sistemas, mas não em todos. Nesses casos, o componente foi considerado de forma que apenas as recomendações existentes fossem usadas para as diretrizes finais.

Para facilitar o entendimento do documento de diretrizes gerais, optou-se por expor cada recomendação de acordo com um único formato. Dessa forma, o documento final ficou com uma estrutura bem definida e fácil de ser consultada.

Alguns componentes possuem potencialidades maiores em um dado sistema operacional do que em outro. Isso significa que eventualmente é impossível criar interfaces idênticas para todos os sistemas operacionais consultados. Por conta disso, acrescentou-se ao documento final, para cada componente de interface que possuísse distinções de comportamentos entre sistemas, uma seção intitulada “Casos específicos”, na qual as peculiaridades desse componente são expostas, assim como as diretrizes que são intrínsecas a ele. Essa distinção possibilita que o *designer* tome ciência de formas de potencializar a usabilidade da interface de acordo com o sistema operacional para o qual ele deseja criar as interfaces. As diretrizes para um determinado componente seguem o formato da Figura 3.7.

<b>Componente 1</b>
<b>Diretrizes comuns para usabilidade</b>
Diretriz 1
Diretriz 2
⋮
Diretriz n
<b>Casos específicos</b>
<b>Sistema operacional 1</b>
Diretriz 1
Diretriz 2
⋮
Diretriz n
<b>Sistema operacional 2</b>
Diretriz 1
Diretriz 2
⋮
Diretriz n
<b>Componente 2</b>
...

Figura 3.7: Formato das diretrizes para criação de interfaces de dispositivos móveis.

Tendo em vista que neste ponto do estudo os resultados das avaliações heurísticas já são conhecidos, novas diretrizes foram elaboradas para serem usadas de acordo com cada compo-

nente, a fim de enriquecer o documento final. O documento resultante desse processo pode ser contemplado no Capítulo 4. Por fim, ressalva-se que as documentações individuais de cada componente de cada sistema operacional estudado não foram embutidas no capítulo teórico deste trabalho, porque elas o estenderiam desnecessariamente, já que todas elas seriam reexibidas no Capítulo 4.

### **3.4 Considerações finais**

A categorização de problemas de usabilidade levou em consideração os problemas relacionados à interface encontrados pelo mestrando, que possui limitações e potencialidades que outros avaliadores eventualmente não teriam. Quando avaliações de usabilidade são realizadas, o próprio avaliador pode representar um agente de viés aos resultados, por conta de vários aspectos que são difíceis de serem analisados no momento dessas avaliações, como alterações de humor, cansaço, conhecimento técnico acerca do assunto, experiência em avaliação de aplicativos análogos, etc. Apesar de o mestrando ter-se preparado por meio de um forte embasamento teórico obtido pela metodologia exploratória de pesquisa e de os fatores de cansaço e de humor não terem afetado os resultados obtidos por conta do longo período de dias de experimentação, não é possível afirmar que as categorias obtidas são as melhores possíveis. Provavelmente, a colaboração de outro especialista pudesse resultar na identificação de outras categorias. De qualquer forma, os resultados que serão apresentados no Capítulo 4 evidenciaram que as heurísticas propostas foram eficazes para a identificação de problemas de usabilidade de interfaces de dispositivos móveis.

## Resultados obtidos

Neste capítulo são mostrados os resultados obtidos ao longo da execução das atividades expostas na Metodologia de Pesquisa (capítulo 3). Para facilitar o entendimento, as informações estão estruturadas de forma análoga à estruturação da Metodologia utilizada.

### 4.1 Proposição das heurísticas

#### 4.1.1 Análise de aplicativos para Android

Os aplicativos analisados neste estudo para a verificação da inadequação das heurísticas de Nielsen para a avaliação de usabilidade de interfaces de dispositivos móveis foram Facebook, Foursquare, Gmail e Twitter para que se encontrassem problemas de usabilidade que deram origem às heurísticas a propostas neste trabalho. Ao longo da interação com cada aplicativo, os problemas encontrados foram documentados e associados a uma categoria de problema. Essas informações estão documentadas nas Tabelas 4.1, 4.2, 4.3 e 4.4. As letras que aparecem à frente da descrição de cada problema indicam o dispositivo no qual ele foi verificado. Nesse sentido, a letra “S” indica que o problema foi identificado no *smartphone* Galaxy I5500, ao passo que a letra “T” indica que ele foi verificado enquanto se usava o aplicativo no *tablet* Motorola XOOM.

**Tabela 4.1:** Categorização dos problemas encontrados nas interfaces do Facebook.

FACEBOOK	
Problema	Categoria
Na tela “Find friends”, o botão “skip” que aparece antes da mensagem principal, no canto superior direito, é tão pequeno que pode passar despercebido. (S)	Facilidade de acesso à funcionalidade; Visibilidade da informação presente na tela.
Na tela “Find freinds”, o botão “skip”, no canto superior direito, aparece antes da mensagem principal, que é importante. O usuário pode pressionar o botão sem ter lido a mensagem. (S, T)	Disposição do componente de interface; visibilidade da informação presente na tela.
	Continua na página seguinte. . .

**Tabela 4.1 – Continuação.**

<b>Problema</b>	<b>Categoria</b>
Na tela principal do usuário, as opções de navegação visíveis são “About”, “Photos” e “Friends”, obrigando que o usuário deslize o dedo sobre os itens para descobrir que existem outros disponíveis, mas que não estão sendo mostrados. (S)	Visibilidade das interações possíveis.
As fotos publicadas no mural do usuário são ampliadas quando ele toca sobre elas, mas não se pode mover a imagem para ver pontos específicos, o que eventualmente impossibilita a visualização de detalhes da imagem. (S, T)	Visibilidade da informação presente na tela (no caso, a imagem).
É necessário que as letras das mensagens no celular sejam menores, por conta do tamanho da tela. Contudo, quando a mensagem representa um “link” textual para outra página, tal como ocorre na <i>Web</i> quando se usa computadores pessoais, o usuário tem dificuldade de assimilar que a mensagem tem funcionalidade de um “link”, porque não diferenciação do texto com as mensagens meramente informativas. (S)	Facilidade de acesso à funcionalidade.
Geralmente são colocadas quatro miniaturas de fotos na tela do celular, tornando impossível que o usuário as identifique com clareza. Em paisagem, elas são visualizadas corretamente, mas em paisagem elas se tornam ilegíveis. (S)	Visibilidade da informação presente na tela; aproveitamento de espaço de acordo com a orientação.
O menu de opções lateral contém muitas opções em listas separadas por categorias. Acontece que cada categoria contém 3 subcategorias à mostra, fazendo com que os últimos registros sejam encontrados com dificuldade, apenas depois de várias operações de rolagem. Os submenus poderiam estar ocultos e aparecerem apenas quando o usuário tocasse sobre os respectivos menus. (S)	Facilidade de acesso à funcionalidade.
A notificação de nova mensagem é adequadamente discreta e acompanhada de uma vibração do aparelho. Contudo, caso a notificação seja de uma conversa que está acontecendo, a conversa não é atualizada automaticamente, causando dúvidas. Além disso, a notificação some em instantes, permitindo que o usuário nem leia a mensagem. (S)	Facilidade de acesso à funcionalidade; <i>Feedback</i> visível e fácil de ser interpretado; prevenção de erros.
A opção de carregar uma foto do sistema operacional ou da máquina definitivamente não se parece com um botão, porque ele se confunde com o restante da mensagem. (S)	Facilidade de acesso à funcionalidade; adequação do componente à funcionalidade.
O aplicativo informa quantas atualizações o usuário ainda não viu, mas quando o usuário visualiza uma delas, todas as outras ficam desmarcadas, como se ele já as tivesse visto. O usuário precisa memorizar quais atualizações ele já viu, para executar corretamente essa funcionalidade. (S, T)	<i>Feedback</i> fácil de ser interpretado e com escopo local (apenas na atualização que foi vista); Minimização da carga de memória do usuário.
Ao comentar sobre uma atualização do mural, o teclado do sistema oculta a atualização, fazendo com que o usuário perca o contato visual com a mensagem sobre a qual ele está comentando. (S)	Visibilidade das informações presentes na tela; bom aproveitamento do espaço da tela.
Documentos postados em grupos não possuem barras de rolagem. Esse comportamento impede que o usuário estime o tamanho do documento. (S, T)	<i>Feedback</i> fácil de ser interpretado.
Documentos postados no grupo não permitem aumento do tamanho da letra. (S, T)	<i>Feedback</i> fácil de ser interpretado.
	Continua na página seguinte. . .

**Tabela 4.1 – Continuação.**

<b>Problema</b>	<b>Categoria</b>
O <i>feedback</i> do sistema quando se curte uma mensagem é muito discreto, porque o botão sequer muda de cor e nenhuma mensagem aparece. (S, T)	<i>Feedback</i> fácil de ser interpretado; facilidade de acesso à funcionalidade.
As fotos do mural aparecem picotadas e o “zoom” permitido não é capaz de exibir o que falta da imagem. (S, T)	Visibilidade das informações presentes na tela.
No tablet, sobra espaço em todas as bordas do aplicativo. Mesmo assim, todas as imagens do mural aparecem cortadas. (T)	Bom aproveitamento do espaço da tela.
É necessário realizar uma operação de deslizamento com o dedo sobre o comentário para que a opção de remoção apareça. O usuário tem que adivinhar que esta interação é possível. (S, T)	Visibilidade das interações possíveis; Facilidade de acesso à funcionalidade.
Existem páginas que possuem barra de rolagem e páginas que não as possuem. (S, T)	Padronização da interface.
Tela de mensagens mostra uma miniatura da foto para cada mensagem, ao invés de mostrar uma única vez para um grupo de mensagens feitas pelo mesmo usuário. (S)	<i>Design</i> minimalista; bom aproveitamento de espaço.
O campo para escrever comentário a respeito de uma mensagem é muito distante da mensagem e fora de onde o usuário está observando. (T)	Bom aproveitamento de espaço

**Tabela 4.2:** Categorização dos problemas encontrados nas interfaces do Foursquare.

<b>FOURSQUARE</b>	
<b>Problema</b>	<b>Categoria</b>
Campo de senha não permite que o usuário a visualize. (S, T)	Facilidade da entrada de dados.
O botão “Learn more in our policy” deveria ser mais chamativo. Apesar de não ter um bom tamanho, ele não possui bordas e o contraste em relação ao fundo é muito pequeno. (S, T)	Facilidade de acesso à funcionalidade; facilidade de entrada de dados.
Opção de “logout” extremamente difícil de ser encontrada. (S, T)	Facilidade de acesso à funcionalidade.
As solicitações de amizade não podem ser aceitas todas de uma vez. Opções análogas devem aparecer em componentes de múltipla escolha para facilitar a atividade. (S, T)	Adequação do componente à funcionalidade.
O problema anterior é agravado porque a página é atualizada completamente toda vez que o usuário toca sobre um dos botões de “Accept”, para aceitar as amizades. Isso implica que, quando o usuário tem mais de uma solicitação para aprovar, ele precisa encontrar o ponto em que estava a cada atualização. (S, T)	Facilidade de entrada de dados; <i>Feedback</i> sem escopo local (atualiza-se toda a página, ao invés de se atualizar apenas o componente selecionado).
O coração para sinalizar a operação análoga ao “Like” do Facebook não é uma boa metáfora. (S, T)	Linguagem do usuário; compatibilidade entre componente e funcionalidade.

**Tabela 4.3:** Categorização dos problemas encontrados nas interfaces do Gmail.

<b>GMAIL</b>	
<b>Problema</b>	<b>Categoria</b>
O campo “escrever email” é muito pequeno. (S)	Facilidade de entrada de dados.
	Continua na página seguinte. . .

**Tabela 4.3 – Continuação.**

<b>Problema</b>	<b>Categoria</b>
Falta informar o usuário que ele realizou determinada tarefa, com uma mensagem de rodapé. (S)	<i>Feedback</i> fácil de ser interpretado.
O nome do remetente muitas vezes não cabe na tela e, mesmo quando o usuário tecla sobre o nome para visualizá-lo, o nome não é exibido por completo. (S)	Visibilidade de toda a informação da tela; ajuda e documentação.
Telas que aparecem com mensagens de texto muito pequenas precisam de uma opção de aumento ou de personalização pelo usuário. (S)	Personalização; Visibilidade de toda informação da tela.
Os botões para alternar entre mensagens são muito próximos e podem ser tocados erroneamente. (T)	Aproveitamento de espaço da tela.
Botão “Inbox” acima do cabeçalho da mensagem para direcionar para a caixa de entrada, mas na verdade é um indicador do filtro ao qual a mensagem pertence. (T)	Adequação do componente à funcionalidade; clareza e objetividade da mensagem.
Usuário não sabe a funcionalidade do botão “mute”, porque não há explicação sobre isso. (S, T)	Ajuda e documentação; <i>feedback</i> fácil de ser interpretado.
Após executar a funcionalidade “mute”, o usuário só consegue revertê-la por alguns segundos seguintes. Depois a opção desaparece. (S, T)	Prevenção de erros; recuperação do estado anterior do sistema.
Ao mostrar as mensagens mais antigas, a tela desorganiza a ordem de mensagens lidas/não-lidas que aparecia anteriormente. As mensagens ficam embaralhadas. (S, T)	Padronização da interface; consistência da interface.
Quando se abre uma mensagem recebida, aparecem três botões acima da mensagem. Os dois primeiros exibem exatamente a mesma tela. Não entendi a diferença entre eles. (S, T)	Adequação do componente à funcionalidade; <i>design</i> minimalista; ajuda e documentação.
Além do movimento de deslizamento de dedo para exibir mensagem seguinte, aplicativo poderia disponibilizar dois botões, para evitar que o usuário volte para tela principal várias vezes, caso ele não perceba a possibilidade dessa interação. (S, T)	Clareza da interação; facilidade de entrada de dados; minimização da carga de memória do usuário.
Ao arquivar uma mensagem, o botão de desfazer aparece no rodapé da página sem muito contraste e com transparência em relação ao fundo. (S,T)	<i>Feedback</i> fácil de ser interpretado; prevenção de erros; retomada ao estado anterior do sistema.
O botão que volta para a caixa de entrada é um desenho que não é autoexplicativo. O usuário é direcionado àquela tela, mas tem dificuldade para retomar a mensagem que estava lendo. (S, T)	Prevenção de erros; retomada ao estado anterior do sistema; adequação do componente à funcionalidade.
O botão “Show all labels” não mostra nada quando os rótulos (labels) já estão na tela. O usuário pode não saber o que o aplicativo chama de “label”. Portanto, seria válido que um destaque sobre os elementos fosse realizado. (T)	<i>Feedback</i> fácil de ser interpretado.

**Tabela 4.4:** Categorização dos problemas encontrados nas interfaces do Twitter.

<b>TWITTER</b>	
<b>Problema</b>	<b>Categoria</b>
Na tela de login, o botão “sign up” fica no canto superior direito. Nessa posição, parece que ele não está relacionado ao conteúdo central, como ocorre com o botão de login abaixo do conteúdo, como deve ser. (S, T)	Clareza do mapeamento entre componente e informação.
	Continua na página seguinte. . .

**Tabela 4.4 – Continuação.**

<b>Problema</b>	<b>Categoria</b>
O botão de busca do Twitter não busca informações da página corrente, mas sim de todo o twitter, mesmo que eu esteja na tela que mostra o mural de outra pessoa. (S, T)	Clareza do mapeamento entre componente e informação; disposição dos componentes da interface.
Quando se marca uma pessoa em uma mensagem, a lista de dicas de pessoas a serem marcadas é impossível de manusear porque só possui duas linhas e o dispositivo não é capaz de interpretar a operação de deslizamento de dedo com tão pouco espaço. (S, T)	Facilidade de entrada de dados.
A metáfora do pássaro com o símbolo “+” à frente não significa nada ao usuário novato. Ele pode teclar sobre esse botão sem querer e seguir uma pessoa que ela não gostaria de seguir. (S, T)	Adequação do componente ao usuário; Adequação do componente à funcionalidade.
A tela “Retweeted by” tem o botão do canto direito muito distante do nome ao qual ele se associa, no canto esquerdo. A noção de alinhamento fica prejudicada. (T)	Clareza do mapeamento entre componente e informação; bom aproveitamento de espaço da tela.
O botão ao lado do botão “Follow” abre um popup que não possui botão para voltar. (S, T)	Prevenção de erros; minimização da carga de memória do usuário.
Quando um usuário não possui nenhuma lista de pessoas que seguem, o aplicativo deveria informar que não há o que mostrar, ao invés de mostrar uma tela branca sem interação alguma como o usuário. (S, T)	<i>Feedback</i> fácil de ser interpretado; prevenção de erros.
Algumas informações solicitam que o usuário escolha um aplicativo externo para que a mensagem seja lida. (S, T)	Aplicação deve ser autocontida.

Os resultados das Tabelas 4.1 a 4.4 foram agrupados para se identificarem as categorias distintas de problemas encontrados. Ao todo, 19 categorias foram identificadas.

Evidentemente, não é sensato que um conjunto com tantas heurísticas seja considerado para uma avaliação, porque os especialistas levariam muito tempo para associar os problemas das interfaces às heurísticas. Além disso, algumas das categorias possuem similaridades, que podem causar redundância. Tendo em vista que, em uma avaliação heurística, cada regra é acompanhada por uma descrição textual, conveio que essas redundâncias fossem suprimidas por meio do agrupamento das heurísticas e redação precisa das descrições. As categorias distintas, resultantes das análises das Tabelas 4.1 a 4.4, estão associadas às heurísticas de Nielsen que as identificam na Tabela 4.5 a seguir. Por meio dessa tabela, nota-se que algumas categorias não puderam ser associadas às heurísticas de Nielsen (células em branco). Outras foram associadas com dificuldade, de forma que a associação foi aproximada. Essas dificuldades indicam que existem limitações das heurísticas de Nielsen para a avaliação de interfaces de dispositivos móveis.

**Tabela 4.5:** Associação das categorias encontradas às heurísticas de Nielsen

<b>Categoria</b>	<b>Heurística de Nielsen</b>
Aproveitamento de espaço na tela, de acordo com a orientação.	6. Reconhecimento ao invés de lembrança (Associação aproximada).
Consistência da interface.	4. Consistência e padrões.
Padronização da interface.	4. Consistência e padrões.
Visibilidade da informação presente na tela.	6. Reconhecimento ao invés de lembrança.

Continua na página seguinte. . .

**Tabela 4.5 – Continuação.**

<b>Categoria</b>	<b>Heurística de Nielsen</b>
Adequação do componente à funcionalidade.	
Clareza do mapeamento entre componente e informação.	
Disposição do componente de interface.	8. Estética e <i>design</i> minimalista.
Clareza e objetividade da mensagem.	2. Compatibilidade do sistema com o mundo real.
Linguagem do usuário.	2. Compatibilidade do sistema com o mundo real.
Prevenção de erros.	9. Ajudar o usuário a reconhecer, diagnosticar e corrigir erros.
Recuperação do estado anterior do sistema.	3. Controle e liberdade do usuário. 9. Ajudar o usuário a reconhecer, diagnosticar e corrigir erros.
Facilidade de entrada de dados.	6. Reconhecimento ao invés de lembrança (Associação aproximada).
Facilidade de acesso à funcionalidade	6. Reconhecimento ao invés de lembrança.
Visibilidade das interações possíveis	
<i>Feedback</i> fácil de ser interpretado e com escopo local.	1. Visibilidade do <i>status</i> do sistema.
Aplicação deve ser autocontida.	
Ajuda e documentação.	10. Ajuda e documentação.
Minimização da carga de memória do usuário.	5. Prevenção de erros. 6. Reconhecimento ao invés de lembrança.
Personalização	7. Flexibilidade e eficiência de uso.

O mapeamento realizado entre categoria e heurística de Nielsen revelou que algumas categorias se relacionavam às mesmas heurísticas. Por conta disso, notou-se a necessidade de agrupar algumas das categorias obtidas. Da mesma forma, algumas categorias compartilharam as mesmas soluções para a resolução de problemas associados a elas e, por conta disso, também foram agrupadas para a proposição de uma mesma heurística. Os resultados desses agrupamentos estão expostos a seguir. À frente de cada categoria obtida, identifica-se a heurística para interfaces de dispositivos móveis à qual ela foi associada, de forma abreviada (a letra “H” é abreviação para “heurística”).

1. Aproveitamento de espaço na tela, de acordo com a orientação (H1);
2. Consistência da interface (H2);
3. Padronização da interface (H2);
4. Visibilidade da informação presente na tela (H3);
5. Adequação do componente à funcionalidade (H4);
6. Clareza do mapeamento entre componente e informação (H4);
7. Disposição do componente de interface (H4);
8. Clareza e objetividade da mensagem (H5);
9. Linguagem do usuário (H5);
10. Prevenção de erros (H6);

11. Recuperação do estado anterior do sistema (H6);
12. Facilidade de entrada de dados (H7);
13. Facilidade de acesso à funcionalidade (H8);
14. Visibilidade das interações possíveis (H8);
15. *Feedback* fácil de ser interpretado e com escopo local (H9);
16. Aplicação deve ser autocontida (H10);
17. Ajuda e documentação (H11);
18. Minimização da carga de memória do usuário (H12);
19. Personalização (H13).

Agrupadas as categorias encontradas, apenas se redigiram as respectivas descrições das heurísticas propostas. Esse processo deu origem à primeira versão das heurísticas para avaliação de usabilidade de interfaces de dispositivos móveis, presentes na Tabela 4.6.

**Tabela 4.6:** Primeira versão das heurísticas para avaliação da usabilidade de interfaces de dispositivos móveis.

<b>Heurística</b>	<b>Descrição</b>
1. Bom aproveitamento do espaço da tela, de acordo com a orientação.	O <i>design</i> deve ser realizado de forma que os itens não fiquem muito distantes, nem muito colados. Espaçamentos de margens não podem ser grandes, para melhorar a visibilidade das informações.
2. Padronização dos componentes da interface.	A aplicação deve manter os componentes no mesmo lugar ao longo de toda a interação, para facilitar a aprendizagem do usuário ao estimular a memória de curto prazo. A metáfora de cada componente ou funcionalidade deve ser única ao longo da aplicação.
3. Visibilidade de toda a informação existente.	Todas as informações devem ser visíveis e legíveis, mesmo em retrato e em paisagem. Isso também vale para mídias, que devem de ser vistas ou executadas na íntegra.
4. Adequação do componente à funcionalidade.	O usuário deve saber exatamente o que ele deve colocar como entrada a um componente, sem que haja ambiguidades ou dúvidas. Metáforas de funcionalidades devem ser compreendidas sem dificuldades.
5. Adequação da mensagem.	A aplicação deve falar a linguagem do usuário. As instruções para executar as funcionalidades devem ser claras e objetivas.
6. Prevenção de erros e retomada rápida ao último estado estável.	O sistema deve ser capaz de se antecipar a uma situação que leve a algum erro por parte do usuário com base em alguma atividade já realizada pelo usuário. Quando um erro ocorrer, a aplicação deve avisar o usuário prontamente e retornar ao último estado estável do aplicativo.
7. Facilidade de entrada de dados.	A forma com que o usuário fornece os dados pode se basear em tecnologias assistivas, mas a aplicação deve sempre mostrar com legibilidade o que está sendo digitado, para que o usuário tenha total controle da situação. O usuário deve conseguir fornecer os dados requeridos de forma prática.
8. Facilidade de acesso a todas as funcionalidades.	As funcionalidades principais da aplicação devem ser realizadas com maior facilidade possível, preferencialmente em apenas uma interação. Nenhuma funcionalidade deve ser difícil de encontrar na interface da aplicação. Todos os componentes de entrada devem ser assimilados com facilidade.

Continua na página seguinte. . .

**Tabela 4.6 – Continuação.**

<b>Heurística</b>	<b>Descrição</b>
9. <i>Feedback</i> fácil de ser notado e com escopo local.	O <i>feedback</i> deve ser fácil de ser notado, para que não haja dúvidas de que a operação foi realizada ou ela ainda está em andamento. Atualizações locais na página devem ser priorizadas, para evitar recarregamento e perda do ponto em que o usuário estava.
10. Aplicação deve ser autocontida.	A aplicação deve conter todas as funcionalidades e informações embutidas nela, sem a necessidade de encaminhar o usuário para programas externos.
11. Ajuda e documentação.	O aplicativo deve possuir opção de Ajuda para especificar os problemas comuns e as formas de solucioná-los. Os assuntos considerados nessa opção devem ser fáceis de serem encontrados.
12. Minimização da carga de memória do usuário.	A interface não pode exigir que o usuário se recorde de ações realizadas anteriormente. Tudo que o usuário precisa para finalizar uma atividade deve estar contido na interface sendo visualizada.
13. Personalização.	A interface deve permitir opções de configurações pessoais dos usuários, como aumentos do tamanho das letras, disposição de elementos na tela, etc.

#### 4.1.2 *Brainstorming* com especialistas

A primeira atividade da primeira sessão de *brainstorming* com cinco especialistas foi uma exposição do conteúdo pesquisado. Esses estudos foram os mesmos expostos no Capítulo 2, com exceção dos trabalhos de aspectos estressores de interfaces [Moraveji e Soesanto, 2012] e de aspectos de *design* criados por Williams [2005], que não foram expostos.

As discussões subsequentes à exposição da teoria usada para a primeira versão das heurísticas resultaram em algumas alterações importantes dessa versão inicial. As heurísticas foram analisadas individualmente, na ordem em que aparecem na Tabela 4.6.

Na primeira heurística, decidiu-se que a descrição seria mudada para “Bom aproveitamento do espaço da tela” e que a instrução de adequação entre as visões em retrato e em paisagem seriam transferidas para a coluna de descrição da heurística.

O nome da segunda heurística foi mudado para “Consistência e padrões da interface”, para que se criasse uma concordância perfeita com a heurística de Nielsen que trata desse mesmo aspecto. Além disso, adicionou-se à descrição a informação de que funcionalidades análogas devem ser realizadas de formas análogas, para facilitação do aprendizado.

A terceira heurística foi mudada para “Visibilidade e acesso fácil a toda informação existente”, para enfatizar que a visualização de componentes não é suficiente para a boa usabilidade da interface.

Na sexta heurística, acrescentou-se à descrição a informação de que, caso o aplicativo não consiga retomar ao último estado estável, ele deve passar o controle ao usuário, para que ele decida o que fazer.

A oitava heurística teve a sentença “Todos os componentes de entrada devem ser assimilados com facilidade” retirada da descrição, porque se concluiu que essa instrução estava implícita na heurística “Adequação do componente à funcionalidade”. Ademais, o nome da heurística foi alterado para “Facilidade de acesso às funcionalidades”.

A “preferência por um *feedback* local” foi retirada da descrição da heurística 9 e mantida apenas em sua respectiva descrição.

A heurística 10 foi descartada. Os envolvidos no *brainstorming* destacaram que aplicativos de dispositivos móveis corriqueiramente precisam de permissão do usuário para se comunicarem com aplicativos externos e que esse é um comportamento típico da própria interação, especialmente em casos de aplicativos gratuitos que possuem referências a conteúdos publicitários.

A heurística 13 foi descartada, porque se considerou que as informações pertinentes a ela poderia ser acrescida à heurística “Visibilidade e acesso fácil de toda informação existente”.

As heurísticas 4, 5, 7, 11 e 12 se mantiveram inalteradas após a primeira reunião.

Uma decisão dos especialistas afetou a criação da segunda versão das heurísticas. Segundo eles, seria imprescindível que os estudos de Moraveji e Soesanto [2012] e de Williams [2005] acerca de estressores de interfaces e de boas práticas para o design, respectivamente, fossem considerados para a nova versão.

A segunda sessão de *brainstorming* se caracterizou pela discussão dos aspectos importantes de cada uma das pesquisas sugeridas pelos especialistas, que foram acrescidas pelo mestrando após a primeira reunião. Após essa discussão, decidiu-se acrescentar a informação de que uma mesma funcionalidade poderia ser acessada e executada de mais de uma forma, para facilitar a usabilidade por usuários mais experientes, à heurística “Facilidade de acesso às funcionalidades”. Ao fim das discussões, obteve-se a Tabela 4.7, a qual foi usada para as avaliações heurísticas pelos especialistas, na etapa de validação. Nessa tabela, as partes negritadas das descrições correspondem aos aspectos teóricos dos estudos sugeridos pelos especialistas e que complementaram a versão anterior das heurísticas.

**Tabela 4.7:** Segunda versão das heurísticas para avaliação de usabilidade de interfaces de dispositivos móveis.

Heurística	Descrição
1. Bom aproveitamento do espaço da tela	Independentemente da orientação do dispositivo, o <i>design</i> deve ser realizado de forma que os itens não fiquem muito distantes, nem muito juntos. <b>Elementos relacionados devem estar próximos e os sem relacionamento devem estar mais afastados.</b> Interfaces não devem estar carregadas com muitos elementos.
2. Consistência e padrões da interface.	A aplicação deve manter os componentes no mesmo lugar e na mesma configuração ao longo de toda a interação, para facilitar a aprendizagem. Funcionalidades análogas devem possuir interações análogas, por meio de atividades parecidas. As características de cada componente (seu tamanho, fonte, cor, etc.) devem permanecer os mesmos em toda a aplicação.
3. Visibilidade e acesso fácil a toda informação existente.	Todas as informações devem ser visíveis e legíveis, tanto em retrato quanto em paisagem. O usuário não deve se esforçar para encontrar ou entender qualquer informação sendo transmitida. Isso também vale para mídias, que devem de ser vistas ou executadas na íntegra. <b>Os elementos da interface devem possuir contraste e elementos de um mesmo grupo de informações devem ter alinhamento adequado.</b>
4. Adequação entre o componente e sua funcionalidade.	O usuário deve saber exatamente o que ele deve colocar como entrada a um componente, sem que haja ambiguidades ou dúvidas. Metáforas de funcionalidades devem ser compreendidas sem dificuldades.
5. Adequação de mensagem à funcionalidade e ao usuário.	A aplicação deve falar a linguagem do usuário e as instruções para executar as funcionalidades devem ser claras e objetivas. <b>A leitura deve ser natural e a linguagem não deve ser invasiva no sentido de obrigar o usuário a fazer algo.</b>

Continua na página seguinte. . .

Tabela 4.7 – Continuação.

Heurística	Descrição
6. Prevenção de erros e retomada rápida ao último estado estável.	O sistema deve ser capaz de se antecipar a uma situação que leve a algum erro por parte do usuário com base em alguma atividade já realizada pelo usuário. Quando um erro ocorrer, a aplicação deve avisar o usuário prontamente e retornar ao último estado estável. <b>Em casos em que o retorno ao último estado seja difícil, o sistema pode transferir o controle para o usuário, para que este decida o que fazer (para onde ir).</b>
7. Facilidade de entrada de dados.	A forma com que o usuário fornece os dados pode se basear em tecnologias assistivas (dispositivos que se conectam ao dispositivo móvel para garantirem acessibilidade a usuários), mas a aplicação deve sempre mostrar claramente o que está sendo solicitado, por meio de texto, áudio, vídeo etc., para que o usuário tenha total controle da situação.
8. Facilidade de acesso às funcionalidades.	As funcionalidades principais da aplicação devem ser realizadas com maior facilidade possível, preferencialmente em apenas uma interação. Além disso, Elas devem ter evidência na interface. As funcionalidades mais frequentes podem ser realizadas por mais de um caminho ou por meio de atalhos. Nenhuma funcionalidade deve ser difícil de encontrar na interface da aplicação.
9. <i>Feedback</i> imediato e fácil de ser notado.	O <i>feedback</i> deve ser fácil de ser notado, para que não haja dúvidas de que a operação foi realizada ou está em andamento. Atualizações locais na página devem ser priorizadas, para evitar recarregamento e perda do ponto em que o usuário estava. <b>Mensagens que aparecem muitas vezes devem ter opção de serem ocultadas pelo usuário. Barras de progresso demoradas devem permitir que o usuário continue executando outras atividades. Feedbacks positivos devem ser visíveis, mas não exigir interação redundante com o usuário, para não estressá-lo.</b>
10. Ajuda e documentação.	O aplicativo deve possuir opção de Ajuda para especificar os problemas comuns e as formas de solucioná-los. Os assuntos considerados nessa opção devem ser fáceis de serem encontrados.
11. Minimização da carga de Memória do usuário.	Aplicações devem permitir que o usuário obtenha a informação de que precisa com facilidade, sem exigir que o usuário memorize passos anteriores para completar uma atividade.

## 4.2 Validação das heurísticas

### 4.2.1 Avaliações heurísticas do aplicativo de anotações multimídia

As heurísticas foram validadas por meio de avaliações heurísticas com 10 especialistas, sendo que cinco deles utilizaram as heurísticas tradicionais de Nielsen para validação de interfaces interativas e os demais usaram as novas heurísticas propostas na Seção 4.1.2.

Os resultados referentes às avaliações heurísticas do aplicativo de anotações multimídia desenvolvido no ICMC estão estruturados em tabelas deste subcapítulo. Cada tabela possui um cabeçalho que identifica o avaliador. Além disso, relacionam-se o problema encontrado, as heurísticas com as quais ele foi relacionado e o grau de severidade. Os primeiros resultados a serem mostrados compreendem os especialistas que usaram os princípios de Nielsen para avaliarem o aplicativo de anotações. Antes de cada sessão de avaliação, cada especialista recebeu uma tabela com as 10 heurísticas de Nielsen, para que elas pudessem ser relacionadas aos problemas encontrados. A tabela mostrada a esses avaliadores é idêntica à Tabela 4.8 e representa uma transcrição das heurísticas revisadas de Nielsen [1994], tal qual se encontra na obra de Rocha e Baranauskas [2003].

**Tabela 4.8:** Heurísticas revisadas de Nielsen traduzidas por Rocha e Baranauskas [2003].

<b>Heurística</b>	<b>Descrição</b>
1. Visibilidade do <i>status</i> do sistema.	Sistema precisa manter os usuários informados sobre o que está acontecendo, fornecendo um <i>feedback</i> adequado dentro de um tempo razoável.
2. Compatibilidade do sistema com o mundo real.	Sistema precisa falar a linguagem do usuário, com palavras, frases e conceitos familiares ao usuário, ao invés de termos orientados ao sistema. Seguir convenções do mundo real, fazendo com que a informação apareça em uma ordem natural e lógica.
3. Controle e liberdade do usuário.	Usuários frequentemente escolhem por engano funções do sistema e precisam ter claras saídas de emergência para sair do estado indesejado sem ter que percorrer um extenso diálogo. Prover funções <i>undo</i> e <i>redo</i> .
4. Consistência e padrões.	Usuários não precisam adivinhar que diferentes palavras, situações ou ações significam a mesma coisa. Seguir convenções de plataforma computacional.
5. Prevenção de erros.	Melhor que uma boa mensagem de erro é um design cuidadoso que previne o erro antes que ele aconteça.
6. Reconhecimento ao invés de lembrança.	Tornar objetos, ações e opções visíveis. Usuário não deve lembrar informações de uma para outra parte do diálogo. Instruções para uso do sistema devem estar visíveis e facilmente recuperáveis quando necessário.
7. Flexibilidade e eficiência de uso.	Usuários novatos se tornam peritos com o uso. Prover aceleradores de forma a aumentar a velocidade da interação. Permitir a usuários experientes “cortar caminho” em ações frequentes.
8. Estética e <i>design</i> minimalista.	Diálogos não devem conter informações irrelevantes e raramente necessárias. Qualquer unidade de informação extra no diálogo irá competir com unidades relevantes de informação e diminuir sua visibilidade relativa.
9. Ajudar o usuário a reconhecer, diagnosticar e corrigir erros.	Mensagens de erros devem ser expressas em linguagem clara, indicando precisamente o problema e construtivamente sugerindo uma solução.
10. Ajuda e documentação.	Embora seja melhor um sistema que possa ser usado sem documentação, é necessário prover “Help” e documentação. Essas informações devem ser fáceis de encontrar, focalizadas na tarefa do usuário e não muito extensas.

Os perfis dos cinco especialistas que utilizaram a Tabela 4.8 para avaliarem o aplicativo de anotações são os seguintes:

- Avaliador 1: especialista que estuda anotações multimídia em vídeos. Possui celular sensível ao toque e equipado com Android e já utilizou *tablet* em disciplinas da Pós-graduação.
- Avaliador 2: especialista com experiência em desenvolvimento de aplicativos Android para dispositivos móveis. Utiliza celular sensível ao toque com Android e utilizou *tablet* em disciplinas da Pós-graduação.
- Avaliador 3: especialista com experiência profissional em avaliações heurísticas de programas interativos para computadores. Utiliza celular sensível ao toque com sistema operacional Android. Entretanto, nunca havia utilizado *tablets*.
- Avaliador 4: especialista com experiência em desenvolvimento de aplicativos Android para dispositivos móveis. Utiliza celular sensível ao toque com Android e utilizou *tablet* em disciplinas da Pós-graduação.
- Avaliador 5: utiliza com frequência dispositivos móveis sensíveis ao toque, tanto com Android quanto com iOS. É programador esporádico de aplicativos simples para Android.

Os resultados obtidos por esses avaliadores podem ser verificados nas Tabelas 4.9, 4.10, 4.11, 4.12 e 4.13.

**Tabela 4.9:** Resultados da avaliação heurística do aplicativo de anotações realizada pelo primeiro especialista, com base nas heurísticas de Nielsen.

<b>AVALIADOR 1</b>		
<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Não entendi porque a conta do usuário foi perguntada novamente.	8	2
Ao invés das opções On e Off, poderia haver alguma outra instrução mais visível.	1	2
Seria bom anotar com outras cores de tinta.	7	2
Aplicativo devia possuir os controles tradicionais do playback (play/-pause), ao invés de um botão simples.	6	3
Na tela de navegação por tinta, a opção de remover está como “clear”, mas logo em seguida ela aparece como “remove”, na caixa de diálogo.	4	2
Na navegação por texto, o usuário pode não saber que é possível remover a anotação, porque essa funcionalidade é acessada apenas depois que o usuário clica em “Edit”.	4	2
Os nomes dos comandos devem ser iguais em todas as telas. Numa tela está “Edit”, noutra está “Clear”, noutra está “Remove”.	4	2
No modo paisagem, os controles somem. Poderia haver uma opção para ativá-los nesse modo de visualização.	7	3

**Tabela 4.10:** Avaliação heurística pelo segundo especialista, com base nas heurísticas de Nielsen.

<b>AVALIADOR 2</b>		
<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Texto da barra de título se repete na instrução das telas de escolha de usuário, logo em seguida.	4	1
A tela de escolha de vídeo não mostra as miniaturas de todos os vídeos.	9	2
Tela de escolha de autor está estranha, porque eu já escolhi o usuário anteriormente.	2, 4	3
Os botões estão difíceis de serem reconhecidos. A interface poderia ser mais enxuta.	8	2
Falta <i>feedback</i> avisando que meu áudio foi gravado.	1	3
A anotação de tinta não sumiu da tela, mesmo depois que escolhi “Ink Annotation is off”.	3, 4, 5	3
As opções de editar e remover poderiam estar na mesma tela, porque a opção de remoção aparece de surpresa, depois de o usuário selecionar “Edit”.	4	2
Poderia haver uma forma de ver todas as anotações de um determinado momento. Ver uma de cada vez dá muito trabalho.	7	2
Falta uma opção de ajuda ao usuário. Eu gostaria de saber se é possível apagar a anotação de tinta, por exemplo.	10	4
As anotações não estão sendo persistidas. Quando volto para a tela inicial e tento rever o vídeo, as anotações não estão lá.	4	4
Falta mostrar qual usuário eu selecionei, na tela principal do vídeo.	1	2

Continua na página seguinte. . .

**Tabela 4.10 – Continuação.**

<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Não anotei nada e, mesmo assim, o aplicativo me deixa compartilhar anotações.	5	3
Ao voltar da tela de compartilhamento, o diálogo perguntando se quero compartilhar o conteúdo reaparece.	2, 3, 4	2

**Tabela 4.11:** Avaliação heurística pelo terceiro especialista, com base nas heurísticas de Nielsen.

<b>AVALIADOR 3</b>		
<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Falta informação do caminho que já percorri, para que eu saiba onde estou na aplicação.	1	3
As mensagens temporárias estão muito discretas, difíceis de serem enxergadas.	5	2
A tela de contexto carece de um botão para voltar.	4	2
Mensagem indicando que a anotação foi editada aparece quase no rodapé, onde o usuário não está olhando.	9	3
<i>Feedback</i> indicando que áudio está gravando está muito ruim. A mudança on/off no botão não é suficiente. Poderia aparecer um ícone avisando o usuário ou até mesmo um popup.	1	3
Aplicação precisa de uma opção de ajuda, para tirar dúvidas do usuário.	10	4
Se eu for pra tela inicial do Android e voltar pra aplicação, o vídeo recomeça, ao invés de continuar de onde eu havia parado. Poderia haver uma opção de salvar o estado do trabalho.	1, 10	3
Não vejo a utilidade da tela de escolha de usuário, já que ela sincroniza com a conta do Google.	8	1
Tela de escolha de autor muito parecida com a tela de usuário. Não sei se prosseguir ou se voltei pra tela anterior. Novamente, poderia haver um caminho percorrido pelo usuário.	3, 5	3
Não entendo como se cria um autor novo, e o aplicativo não me esclarece isso.	4	3
Os comandos de anotação desaparecem quando estou no modo paisagem.	5	3
A tela com o botão de “start” está esquisita. Pra quê ela serve? Esse botão isolado nem parece botão.	1, 4, 10	2

**Tabela 4.12:** Avaliação heurística pelo quarto especialista, com base nas heurísticas de Nielsen.

<b>AVALIADOR 4</b>		
<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
O <i>feedback</i> depois de a anotação de áudio ser realizada não é suficiente. Poderia aparecer uma mensagem informando que a anotação foi realizada.	1	3
Caso eu deseje fazer multianotações, o aplicativo não facilita essa funcionalidade. É difícil identificar quais anotações estão disponíveis.	1	3
O espaço poderia ser mais bem aproveitado.	8	1

Continua na página seguinte. . .

**Tabela 4.12 – Continuação.**

<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
A anotação de tinta não desaparece, mesmo depois de ser removida na tela de navegação.	5	4
Os botões de anotação de tinta e de áudio poderiam ser ícones diferenciados, ao invés de botões.	4	4
Os botões de navegação poderiam estar agrupados.	1	3
Poderia haver a possibilidade de discriminar quais anotações foram feitas por quais autores.	3	2
O botão de navegação de “Ink Navigation” e “Ink Annotation” são muito parecidos. Errei algumas vezes.	8	4
Ao compartilhar um conteúdo, a caixa de diálogos poderia conter os botões “yes” e “no”, ao invés de “ok” e “cancel”. Na tela seguinte, a mensagem poderia ser outra, ao invés de “Choose Application”.	4	2
Mensagem temporária do sistema operacional poderia aparecer sempre centralizada e chamar mais atenção.	9	4
Os botões de funcionalidades somem quando estou no modo paisagem. Poderia haver uma documentação desse comportamento.	10	3

**Tabela 4.13:** Avaliação heurística pelo quinto especialista, com base nas heurísticas de Nielsen.

<b>AVALIADOR 5</b>		
<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Botões estão muito próximos.	3	3
Botões de anotação de tinta e de áudio deixam usuário confuso quanto aos seus <i>status</i> (ligado/desligado).	1	2
Tela de vídeos não possui informação do local onde esses vídeos estão armazenados, dificultando a usabilidade por parte do usuário inexperiente.	7	3
O vídeo não pausa para que eu faça a anotação de tinta.	5	3
Falta dar opção ao usuário de visualizar múltiplos autores.	3	1
Falta de Padrão no tamanho dos botões com funcionalidades semelhantes, afetando a visualização dos mesmos.	4	2
Modo paisagem poderia possuir barra vertical com botões de visualização de anotações de tinta, de áudio e de texto.	7	1
Falta de padrão quanto ao nome dos botões de remoção das anotações em tinta e áudio.	4	1
Usuário não tem opção de escolher quais anotações ele gostaria de visualizar.	3	1
Falta opção de visualizar informações básicas dos vídeos, tais como: número de anotações de texto, áudio e tinta que o mesmo possui e shares.	1	2
Mostrar nome do usuário ativo que está gerando as anotações.	1	2

A seguir são expostos os perfis dos cinco avaliadores que realizaram a avaliação heurística do aplicativo de anotações multimídias utilizando as heurísticas propostas neste trabalho, para avaliação de interfaces de dispositivos móveis.

- Avaliador 1: especialista possui celular sensível ao toque e equipado com Android e já utilizou *tablet* em disciplinas da Pós-graduação.

- Avaliador 2: especialista com experiência em desenvolvimento de aplicativos Android para dispositivos móveis. Utiliza celular sensível ao toque com Android e utilizou *tablet* em disciplinas da Pós-graduação.
- Avaliador 3: especialista com experiência em desenvolvimento de aplicativos Android para dispositivos móveis. Não possui celular nem tablet, mas utilizou muitas vezes os dois dispositivos em disciplinas da Pós-graduação e em minicursos de programação básica para Android ministrados por ele.
- Avaliador 4: possui *smartphone* sensível ao toque equipado com Android. É programador esporádico de aplicativos simples para Android.
- Avaliador 5: especialista com experiência profissional em avaliações heurísticas de programas interativos para computadores. Utiliza celular sensível ao toque com sistema operacional Android. Entretanto, nunca havia utilizado tablets.

Os resultados obtidos por esses avaliadores são apresentados nas Tabelas 4.14, 4.15, 4.16, 4.17 e 4.18.

**Tabela 4.14:** Resultados da avaliação heurística do aplicativo de anotações realizada pelo primeiro especialista, utilizando as heurísticas para dispositivos móveis.

<b>AVALIADOR 1</b>		
<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Botão start desnecessário na primeira tela.	Não identificada.	1
Botão play/pause difícil de entender porque não possui a metáfora conhecida com os desenhos de play e de pause.	3, 5	3
Botão play/pause muito próximo dos demais.	1	2
Letra dos botões muito pequena.	3, 7	2
A tela de navegação deveria parar a execução do vídeo.	2	3
Informações de contexto devem permitir que o usuário volte à tela principal apenas por tocar qualquer parte que não seja a caixa de informação. Clicar no botão voltar do sistema operacional não é intuitivo.	5, 6	1
Botões “Share” e “Context Information” têm propósitos diferentes e deveriam estar destacados de forma diferente dos demais botões.	1	2
Botões com funcionalidades análogas parecem desconexos na interface. Anotar texto e ver anotações de texto deveriam estar próximos, um em frente ao outro, por exemplo. Idem para as demais anotações.	2	3
Anotação de tinta não sai do vídeo, mesmo depois que a anotação de tinta está desabilitada. Botão não está fazendo o que deveria.	4, 5, 10	4
Anotação de áudio deveria ter um <i>feedback</i> após o clique, algo como “Talk”.	7, 10	2
Botão “Cancel” na tela de navegação deveria se chamar “Exit”, porque o usuário não começou a executar nenhuma ação neste ponto da interação.	5	2
Ao ir para a tela anterior, selecionar o mesmo autor do mesmo vídeo, as anotações desaparecem. A aplicação deveria voltar ao estado anterior.	6	4
Username deveria ser considerado o autor, em minha opinião. A tela de autor parece desnecessária.	8	3

Continua na página seguinte. . .

**Tabela 4.14 – Continuação.**

<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Em paisagem, o vídeo aparece com os botões de navegação de vídeo com as metáforas conhecidas, ao contrário do que ocorrem em retrato.	2	3
Na tela de navegação, o formato da hora considera horas, minutos e segundos, mas no player o formato aparece apenas com minutos e segundos.	2	1
Como não há informação dizendo que em paisagem o app não funciona da mesma forma que em retrato, o usuário não sabe dessa distinção.	2, 5, 10	2

**Tabela 4.15:** Resultados da avaliação heurística do aplicativo de anotações realizada pelo segundo especialista, utilizando as heurísticas para dispositivos móveis.

<b>AVALIADOR 2</b>		
<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Botão Play/pause largo demais.	1	1
Botões agrupados, quando há muito espaço livre na tela.	1	2
Não há opção de Help.	10	2
Botões de navegação poderiam ser agrupados em um único botão.	3, 8	2
Lista de navegação não dá ideia ao usuário de que ela pode ser rolada.	9	2
A tela de visualização de informações de contexto carece de um botão “Ok”, para voltar para a tela principal.	8	2
Para remover, é necessário clicar em editar, e isso dificulta o acesso.	8	3
A tela de navegação de tinta tem opção “clear”, diferentemente do que acontece nas outras telas.	2, 5	3
Após compartilhar uma anotação, a caixa de diálogo aparece mais de uma vez.	9	3
Anotar texto poderia deixar o vídeo tocando, assim como ocorre nas demais anotações.	2	1
Falta opção de salvar as anotações, porque elas somem depois que se volta uma tela.	7, 11	4
Falta opção “clear all”, para limpar todos os campos preenchidos.	7	2
Para cada vídeo, deve-se colocar o autor novamente. Essa informação poderia ser salva.	8, 11	3

**Tabela 4.16:** Resultados da avaliação heurística do aplicativo de anotações realizada pelo terceiro especialista, utilizando as heurísticas para dispositivos móveis.

<b>AVALIADOR 3</b>		
<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Botão “start” está em uma posição não usual. Ele poderia estar ou no topo ou mais embaixo, com mais destaque.	1, 11	1
É estranho pedir autor quando já se solicitou o usuário. A diferença de papéis não é clara. Tela não deveria existir ou uma mensagem esclarecendo poderia ser providenciada.	4, 5, 10	3
Os comandos em retrato não são possíveis de serem vistos em paisagem, mesmo quando se toca no vídeo.	2	4

Continua na página seguinte. . .

**Tabela 4.16 – Continuação.**

<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Na tela principal, sobra muito espaço abaixo dos botões. 1 2 O botão “Add” aparentemente adiciona mensagem em branco na cena, porque os botões são deslocados quando a anotação de texto aparece.	10	4
Marcação de tinta não desaparece. Não sei o que o aplicativo espera que eu faça.	6, 7, 9	4
Botão “cancel” na tela de navegação está com nome impróprio. Poderia ser “close”.	5	1
Usuário tem que memorizar qual é o botão de “voltar” na tela de informações de contexto, porque ele carece de um botão para voltar.	11	2
Botões estão menores que deveriam. Apertei botão “share” sem querer, por ele estar próximo demais do botão “Textual navigation”.	1	2
Dois botões poderiam ocupar a mesma largura na tela principal. Olhar share/context em relação a ink annotation/áudio annotation. Eles estão desalinhados.	2	1
Quando se aperta em “áudio annotation”, o usuário deve adivinhar que tem que falar alguma coisa, porque não há instrução.	9	3
Compartilhar conteúdo replica mensagem de diálogo.	2	2
O vídeo deveria pausar enquanto o usuário compartilha o conteúdo.	2, 9	1
A marcação de tinta é gravada de tempos em tempos, mas essa funcionalidade deveria ocorrer de uma só vez, quando o usuário tirasse o dedo da tela.	1, 6	3
Botões na tela de navegação para edição/remoção possuem nomes diferentes em cada funcionalidade.	2	1
As informações anotadas não são salvas quando se seleciona o vídeo novamente.	10	4
Símbolo básico play/pause (universal) poderia utilizados, ao invés do botão comum.	11	1
Ausência de botão “Close Application” na aplicação.	2, 8, 10, 11	2

**Tabela 4.17:** Resultados da avaliação heurística do aplicativo de anotações realizada pelo quarto especialista, utilizando as heurísticas para dispositivos móveis.

<b>AVALIADOR 4</b>		
<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
A tela de navegação exibe mensagem para que o usuário selecione o instante a ser navegado mesmo quando não há nenhuma anotação feita.	2, 4, 6	2
O botão de “Add” parece adicionar qualquer anotação, mas ele só adiciona a anotação de texto.	1, 6	3
Falta uma mensagem avisando que a gravação de áudio está ocorrendo.	3, 7, 8, 9, 10, 11	3
Seria interessante saber quanto tempo tem a anotação de áudio, na tela de navegação de anotação de áudio.	3	1
Poderia haver uma forma de controlar o volume da anotação de áudio em relação ao volume do vídeo, porque o usuário pode querer que sua voz tenha mais evidência que o som do vídeo.	7, 10	1
Como há muito espaço livre abaixo dos botões principais, pensei que a anotação deveria ser feita nesse espaço, e não no próprio vídeo.	1	2

Continua na página seguinte. . .

**Tabela 4.17 – Continuação.**

<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Em paisagem, os comandos não aparecem.	3	4
A anotação de tinta no vídeo não desaparece.	2, 6	4
Falta botão “sair”.	5, 6, 8	2
Os botões deveriam estar mais alinhados e aproveitar melhor o espaço da tela.	1	2
Deveria haver opção de sair na tela de contextos.	5, 6, 8	2
“Informação de contexto” é um termo inadequado ao usuário.	4	2
Os botões poderiam estar próximos de acordo com a funcionalidade que eles possuem.	1	3
No momento de compartilhar um texto no Orkut, após clicar no botão “share”, apareceu uma tela preta.	6, 9, 11	4
Ao clicar em voltar, depois da tela preta, a mensagem de compartilhar anotação reapareceu.	6	4
Ao clicar em voltar, depois que a tela preta apareceu, o vídeo volta no início.	11	2
O nome “navigation” não é muito intuitivo. Poderia ser “annotation list”.	4, 5	2
A partir de um número de anotações com tintas, é impossível anotar formas curvas.	6, 7	4
Um usuário comum não sabe o que é um arquivo XML, no momento em que escolhe compartilhar o conteúdo.	5	2

**Tabela 4.18:** Resultados da avaliação heurística do aplicativo de anotações realizada pelo quinto especialista, utilizando as heurísticas para dispositivos móveis.

<b>AVALIADOR 5</b>		
<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Botão “start” está em uma posição não usual. Poderia estar embaixo.	2	1
Uma tela inteira só com um botão “start”. Por que não inicia o aplicativo de uma vez?	1	2
Fiquei perdido quando a segunda tela pedindo usuário apareceu.	4	
Usuário deve apertar duas vezes em “Edit” para poder editar anotação de texto. (Uma vez na tela de navegação, outra vez no diálogo).	4	2
Poderia permitir que o usuário editasse a cena no próprio campo de texto, ao passar pela legenda desejada.	7	3
Em paisagem, não há como fazer anotações.	2, 8	4
Senti falta de opção de pausar o vídeo clicar sobre ele.	4	1
Com o vídeo parado, fiz anotação de tinta e fui para a tela de navegação. Removi a anotação, ela saiu da tela de navegação, mas não saiu do vídeo, na tela principal.	9	4
Falta uma ajuda nesse aplicativo. Estou confuso.	10	4
Há duas formas de realizar anotações: cena a cena, com o vídeo parado; e ao longo da execução do vídeo. Entretanto, a anotação de texto só é permitida com o vídeo parado. Poderia ser um comportamento análogo.	7, 8	2
Geralmente players possuem o tempo no formato <tempo corrente>/<tempo final>, ao invés do formato usado no sistema.	2, 9	1

Continua na página seguinte. . .

**Tabela 4.18 – Continuação.**

<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Ao compartilhar as anotações de texto, a interface não deixa claro se todas as anotações realizadas serão compartilhadas ou se será apenas uma.	5, 9	3
A tela de informação de contexto não permite voltar para a tela anterior. Em todas as outras telas, eu pude voltar.	2	4
Poderia haver um <i>feedback</i> ao clicar em fazer anotação de áudio. Algo do tipo “Iniciando gravação em 3..2. . . 1”. Esse problema ocorre também para a anotação de tinta, porque o usuário pode pensar que deve anotar onde aparece o teclado, ou nem perceber que pode anotar.	7, 9	3
Ao terminar o vídeo, ele poderia parar.	2	1
Play/pause são separados por barra, enquanto que On e Off são alterados no próprio botão.	2	1
Como há espaço na tela, os botões poderiam estar mais espaçados. Botões de navegação poderiam estar mais distantes dos demais.	1	1
Usuário não entende o que o aplicativo considera informação de contexto. Apaguei todas as anotações e eles continuam aparecendo. Coloquei anotação nova e ela não apareceu, ou seja, o comportamento está confuso.	5	2
Ao longo do caminho até a tela principal, poderia haver um <i>feedback</i> das opções preenchidas anteriormente.	9, 11	2
O aplicativo tenta exibir as informações de contexto mesmo quando não há nenhuma.	5	1
A barra de controle do aplicativo exibe a mesma informação sendo exibida na antes da lista de opções, nas telas de seleção de usuário. Poderia ser uma informação diferente.	2	1
Para quê serve o logo, se ele não aparece em todas as telas? Analogamente ao logo, a barra de informações do aplicativo desaparece na tela principal.	2	1

Os resultados de cada avaliador que usou as heurísticas tradicionais de Nielsen foram agrupados. Evidentemente, alguns problemas foram encontrados por mais de um avaliador, embora as descrições textuais deles não tenham sido idênticas. Nestes casos, o problema foi considerado apenas uma vez. Esse mesmo processo de agrupamento foi realizado com os resultados obtidos pelos especialistas que utilizaram as heurísticas propostas para interfaces de dispositivos móveis. Esses dois agrupamentos permitiram identificar os problemas que puderam ser identificados pelos dois conjuntos de heurísticas. Ao todo, os 10 avaliadores identificaram 75 problemas de usabilidade distintos, os quais estão reportados na Tabela 4.19. Nessa tabela, cada problema descrito é acompanhado das letras “N”, “M” ou “NM”. A letra “N” indica que o problema foi encontrado apenas pela heurística de Nielsen, ao passo que a letra “M” caracteriza um problema encontrado apenas por avaliadores que usaram as heurísticas para interfaces de dispositivos móveis. A sigla “NM”, por sua vez, indica que o problema foi encontrado por ambas as heurísticas.

Tabela 4.19: Categorização dos problemas encontrados nas interfaces do Twitter.

<b>Problema</b>	<b>Descrição</b>
N1	Funcionalidade faltante: ver todas as anotações de um instante qualquer. Ver uma a uma dá trabalho.
N2	Funcionalidade faltante: discriminar quais anotações foram feitas por quais autores.
N3	Funcionalidade faltante: mudar cor da anotação de tinta.
N4	Na tela de navegação por tinta, a opção de remover está como “clear”, mas logo em seguida ela aparece como “remove”, na caixa de diálogo.
N5	Funcionalidade faltante: realizar multianotações (habilitar todas ao mesmo tempo).
N6	Mensagem indicando que a anotação foi editada aparece quase no rodapé, onde o usuário não está olhando.
N7	As fotos miniaturas deveriam pegar um frame significativo do vídeo.
N8	Os botões estão difíceis de serem reconhecidos.
N9	Sistema não informa o usuário de que o áudio foi gravado.
N10	O sistema não informa o usuário o que selecionei na tela principal.
N11	O aplicativo deixa a opção de compartilhamento habilitada mesmo quando o usuário ainda não realizou nenhuma anotação.
N12	Se eu for pra tela inicial do Android e voltar pra aplicação, o vídeo recomeça, ao invés de continuar de onde eu havia parado. Poderia haver uma opção de salvar o estado do trabalho.
N13	Não vejo a utilidade da tela de escolha de usuário, já que ela sincroniza com a conta do Google.
N14	Os botões de navegação poderiam estar agrupados.
N15	Ao compartilhar um conteúdo, a caixa de diálogos poderia conter os botões “sim” e “não”, ao invés de “ok” e “cancel”. Na tela seguinte, a mensagem poderia ser outra, ao invés de “Choose Application”.
N16	Tela de vídeos não possui informação do local onde esses vídeos estão alocados, dificultando a usabilidade por parte do usuário inexperiente.
N17	O vídeo não está pausando para criação da anotação de tinta.
N18	Dar opção ao usuário de visualizar múltiplos autores.
N19	Falta opção de visualizar informações básicas dos vídeos, tais como: numero de anotações de texto, áudio e tinta e de compartilhamentos que o mesmo possui.
N20	Na tela de navegação por tinta, a opção de remover está como “clear”, mas logo em seguida ela aparece como “remove”, na caixa de diálogo.
M1	Botão play/pause muito próximo dos demais.
M2	Letra nos botões está muito pequena.
M3	A tela de navegação deveria parar a execução do vídeo.
M4	Botões de Share e de Infos de contexto têm propósitos diferentes e deveriam estar destacados de forma diferente dos demais botões.
M5	Botão “Cancel” na tela de navegação deveria se chamar “Exit”, porque o usuário não começou a executar nenhuma ação neste ponto da interação.
M6	Em paisagem, o vídeo aparece com os botões de navegação de vídeo com as metáforas conhecidas, ao contrário do que ocorrem em retrato.
M7	Na tela de navegação, o formato da hora considera horas, minutos e segundos, mas no player o formato aparece apenas com minutos e segundos.
M8	Botão play/pause está muito largo.
M9	Botões de navegação poderiam ser agrupados em um único botão.
M10	Lista de navegação não dá ideia ao usuário de que ela pode ser rolada.

Continua na página seguinte. . .

**Tabela 4.19 – Continuação.**

<b>Problema</b>	<b>Descrição</b>
M11	Falta opção de “clear all”, para limpar todos os campos preenchidos.
M12	Para cada vídeo, deve-se colocar o autor novamente. Essa informação poderia ser salva.
M13	O botão “Add” aparentemente adiciona mensagem em branco na cena, porque os botões são deslocados quando a anotação de texto aparece.
M14	Dois botões poderiam ocupar a mesma largura na tela principal. Olhar share/context em relação a ink annotation/áudio annotation. Eles estão desalinhados.
M15	O vídeo deveria pausar enquanto o usuário compartilha o conteúdo.
M16	A marcação de tinta é gravada de tempos em tempos, mas essa funcionalidade deveria ocorrer de uma só vez, quando o usuário tirasse o dedo da tela.
M17	Ausência de botão “Close Application” na aplicação.
M18	A tela de navegação exibe mensagem para que o usuário selecione o instante a ser navegado mesmo quando não há nenhuma anotação feita.
M19	O botão de “Add” parece adicionar qualquer anotação, mas ele só adiciona a anotação de texto.
M20	Seria interessante saber quanto tempo tem a anotação de áudio, na tela de navegação de anotação de áudio.
M21	Funcionalidade faltante: poderia haver uma forma de controlar o volume da anotação de áudio em relação ao volume do vídeo, porque o usuário pode querer que sua voz tenha mais evidência que o som do vídeo.
M22	Como há muito espaço livre abaixo dos botões principais, pensei que a anotação deveria ser feita nesse espaço, e não no próprio vídeo.
M23	“Context information” é um termo inadequado ao usuário.
M24	No momento de compartilhar um texto no Orkut, após clicar no botão “share”, apareceu uma tela preta.
M25	Ao clicar em voltar, depois que a tela preta apareceu, o vídeo volta no início.
M26	O nome “navigation” na tela que lista as anotações não é muito intuitivo.
M27	A partir de um número de anotações com tintas, é impossível anotar formas curvas.
M28	Um usuário comum não sabe o que é um arquivo XML, no momento em que escolhe compartilhar o conteúdo.
M29	Botão “start” está em uma posição não-usual. Poderia estar embaixo.
M30	Usuário deve apertar duas vezes em “Edit” para poder editar anotação de texto. (Uma vez na tela de navegação, outra vez no diálogo).
M31	Funcionalidade faltante: Senti falta de opção de pausar o vídeo clicar sobre ele.
M32	Há duas formas de realizar anotações: cena a cena, com o vídeo parado; e ao longo da execução do vídeo. Entretanto, a anotação de texto só é permitida com o vídeo parado. Poderia ser um comportamento análogo.
M33	Geralmente players possuem o tempo no formato <tempo corrente>/<tempo final>, ao invés do formato usado no sistema.
M34	Ao compartilhar as anotações de texto, a interface não deixa claro se todas as anotações realizadas serão compartilhadas ou se será apenas uma.
M35	O vídeo poderia parar quando terminasse de ser executado.
M36	Play/pause são separados por barra, enquanto que On e Off são alternados no próprio botão.
M37	O aplicativo tenta exibir as informações de contexto mesmo quando não há nenhuma.
M38	Para quê serve o logo, se ele não aparece em todas as telas? Analogamente ao logo, a barra de informações do aplicativo desaparece na tela principal.
NM1	Tela inicial poderia ser desconsiderada.

Continua na página seguinte. . .

**Tabela 4.19 – Continuação.**

<b>Problema</b>	<b>Descrição</b>
NM2	Deveria ter usado os controles tradicionais do playback (play/pause) ao invés de um botão simples.
NM3	A tela de contexto carece de um botão para voltar.
NM4	Botões com funcionalidades análogas parecem desconexos na interface.
NM5	A anotação de tinta não desapareceu da tela.
NM6	Dificuldade de entender se as funcionalidades de anotações de áudio e de vídeo estão prontas para uso.
NM7	As anotações não estão sendo persistidas.
NM8	Dificuldade de entender a distinção de usuários.
NM9	O espaço da tela poderia ser mais bem aproveitado.
NM10	Falta Ajuda para que o usuário saiba o que é e o que não é possível realizar com o aplicativo.
NM11	A opção de remoção está escondida. Poderia ser mais evidente.
NM12	Os rótulos dos botões de remoção não estão iguais nas telas de navegação.
NM13	Ao voltar da tela de compartilhamento, o diálogo perguntando se quero compartilhar o conteúdo reaparece.
NM14	No modo paisagem, os controles desaparecem.
NM15	Deveria ter usado os controles tradicionais do playback (play/pause) ao invés de um botão simples.
NM16	Não sei o que já preenchi quando estou em uma tela qualquer.
NM17	Texto da barra de título se repete na instrução das telas de escolha de usuário.

A Tabela 4.19 permite concluir que 20 problemas foram encontrados exclusivamente pelas heurísticas de Nielsen (26,67% dos 75 problemas encontrados), ao passo que 38 problemas foram identificados exclusivamente pelas heurísticas novas (50,66% do total). Dezesete problemas foram encontrados por ambas as heurísticas e representam 22,67% do total.

Os resultados dos avaliadores em relação aos graus de severidade associados a cada problema estão dispostos nas Tabelas 4.20 e 4.21. Na primeira delas constam as informações obtidas por meio das avaliações utilizando as heurísticas de Nielsen. Os resultados encontrados por meio da avaliação com as novas heurísticas estão na segunda tabela.

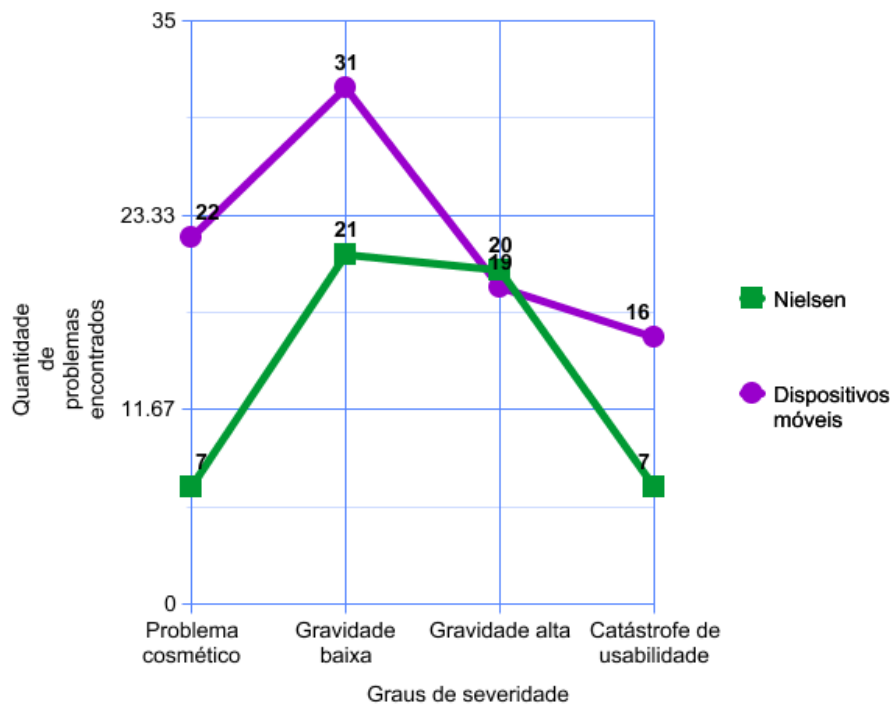
**Tabela 4.20:** Sumarização das quantidades de problemas encontrados com cada grau de severidade, por cada avaliador que usou as heurísticas de Nielsen.

<b>Quantidade de problemas encontrados</b>				
<b>Avaliador</b>	<b>Grau 1</b>	<b>Grau 2</b>	<b>Grau 3</b>	<b>Grau 4</b>
1	0	6	2	0
2	1	6	4	2
3	1	3	7	1
4	1	2	4	4
5	4	4	3	0
<b>Total</b>	7	21	20	7
<b>Média (%)</b>	12,73	38,18	36,36	12,73
<b>Desvio padrão</b>	2,42	1,60	1,67	2,49

**Tabela 4.21:** Sumarização das quantidades de problemas encontrados com cada grau de severidade, por cada avaliador que usou as heurísticas para interfaces de dispositivos móveis.

Quantidade de problemas encontrados				
Avaliador	Grau 1	Grau 2	Grau 3	Grau 4
1	3	6	5	2
2	2	6	4	1
3	6	5	3	4
4	2	9	3	5
5	9	5	4	4
<b>Total</b>	22	31	19	16
<b>Média (%)</b>	25,00	35,23	21,59	18,18
<b>Desvio padrão</b>	2,72	1,47	0,75	1,47

Do ponto de vista absoluto, os resultados obtidos pelas heurísticas novas foram consideravelmente melhores que os obtidos pelas heurísticas tradicionais, com exceção de problemas com gravidade alta (3), que tiveram 20 problemas em Nielsen e 19 nas novas heurísticas. Essa comparação pode ser visualizada na Figura 4.1. Comparativamente aos valores percentuais obtidos, ou seja, às contribuições que cada grau de severidade teve em relação aos problemas de cada abordagem de heurísticas, nota-se que as heurísticas tradicionais foram menos efetivas para encontrarem problemas catastróficos. Contudo, os problemas considerados catastróficos tiveram maior contribuição percentual em relação ao total nas heurísticas novas, como se constata na Figura 4.2.



**Figura 4.1:** Relação entre o número de problemas encontrados por cada grau de severidade.

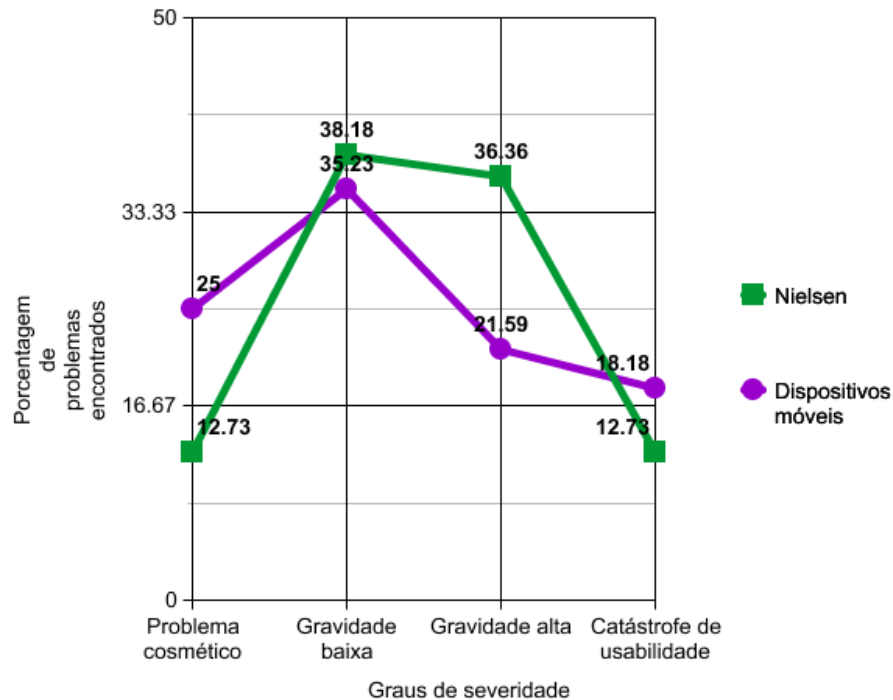


Figura 4.2: Percentual de problemas encontrados com cada grau de severidade.

Outro resultado que se pode obter das avaliações heurísticas realizadas é a quantidade de problemas encontrados com cada heurística, de acordo com cada modalidade utilizada (tradicional ou para dispositivos móveis). Pelas Figuras 4.3 e 4.4, percebe-se que a heurística de Consistência e Padrões foi a que mais encontrou problemas nos experimentos realizados.

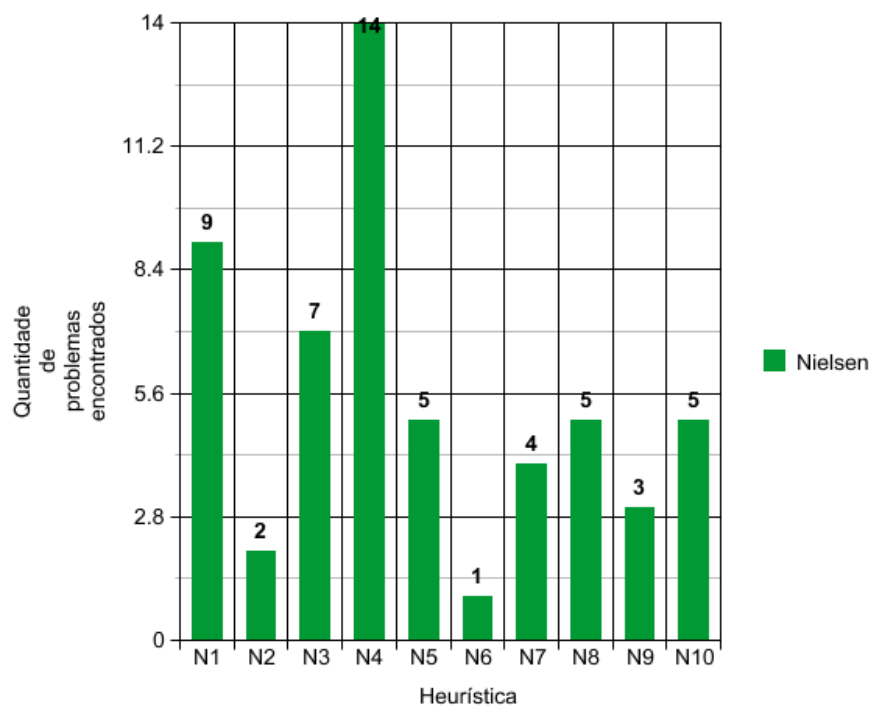
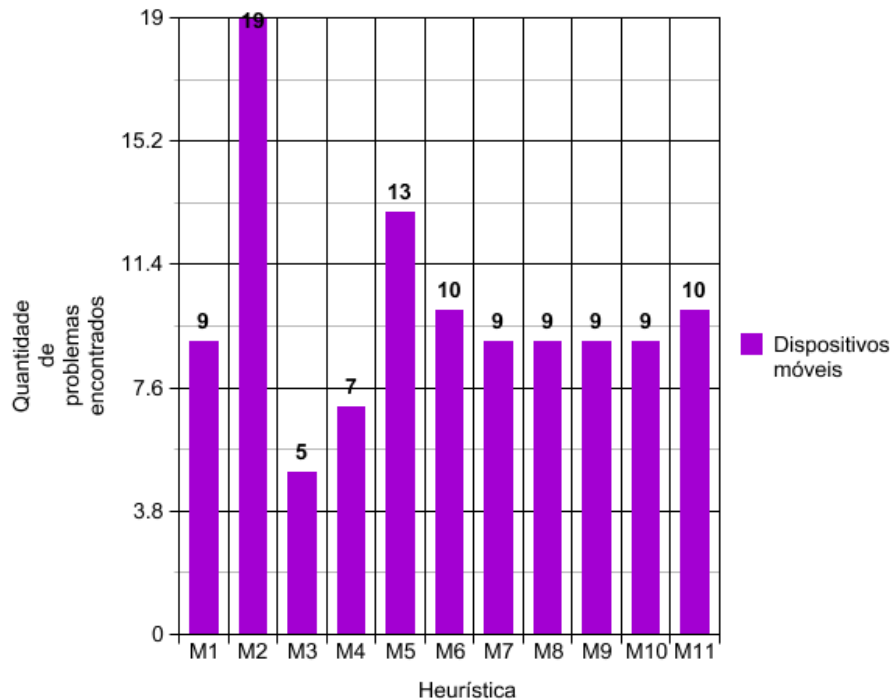


Figura 4.3: Quantidade de problemas associados a cada heurística de Nielsen.



**Figura 4.4:** Quantidade de problemas associados a cada heurística para avaliação de interfaces de dispositivos móveis.

Há de se ressaltar que o primeiro avaliador que usou as heurísticas para interfaces de dispositivos móveis não conseguiu associar o problema a nenhuma das heurísticas. Contudo, todos os demais especialistas que encontraram esse mesmo problema utilizando o mesmo conjunto de heurísticas conseguiram fazer essa associação. De qualquer forma, tendo em vista que o problema se refere ao botão “start” ser desnecessário, é pertinente esclarecer na descrição da heurística “Bom aproveitamento do espaço da tela” que apenas os componentes necessários para realizar a funcionalidade devem ser exibidos.

#### 4.2.2 Avaliação com usuários finais

A avaliação do aplicativo para anotação multimídia pelos dois usuários finais teve registro automático das atividades realizadas por eles ao longo das interações. Este registro foi implementado em Java juntamente com o código-fonte da aplicação. As informações são salvas em um arquivo do tipo HTML (*Hyper Text Markup Language*) no cartão SD do *tablet* Motorola XOOM usado nos experimentos. O trecho de código-fonte responsável pela criação deste arquivo e pela gravação dos dados está apresentado na Figura 4.5. Por questões didáticas, as linhas do código-fonte foram numeradas.

Na linha 1 da Figura 4.5, especifica-se o cabeçalho da função responsável pela criação do arquivo e gravação dos dados nele. O primeiro parâmetro da função é o nome do arquivo a ser criado. O segundo parâmetro é a mensagem a ser gravada. Como as mensagens são referentes às interações do usuário enquanto ele utiliza o aplicativo, várias solicitações à função da linha 1 foram feitas, com mensagens diferentes.

A linha 2 é responsável pela identificação do diretório em que o cartão SD do dispositivo móvel se encontra e pela associação deste diretório ao arquivo intitulado “file” dessa mesma linha. Identificado o diretório de gravação, o arquivo é efetivamente criado com o nome desejado, na linha 3.

Na linha 6 cria-se uma instância do gravador dos dados do arquivo. O segundo parâmetro

```

1 public static void saveToLog(String filename, String msg){
2     File file = Environment.getExternalStorageDirectory();
3     File userFile = new File(file, filename);
4     FileWriter writer;
5     try {
6         writer = new FileWriter(userFile, true);
7         writer.append(msg+getNow());
8         writer.flush();
9         writer.close();
10    } catch (IOException e) {
11        e.printStackTrace();
12        Log.e("Error while saving " +
13            "data to the SD card.", msg);
14    }
15 }
16
17 public static String getNow(){
18     SimpleDateFormat dateFormat =
19         new SimpleDateFormat("HH:mm:ss zzzz");
20     return dateFormat.format(new Date());
21 }
22

```

**Figura 4.5:** Código-fonte responsável pela criação e gravação do arquivo de registros de ações de usuários do aplicativo de anotações multimídias.

da função dessa linha especifica se os dados a serem escritos no arquivo serão acrescentados aos dados que já existem nele. Como a função `saveToFile()` é chamada diversas vezes ao longo da utilização do aplicativo, é importante que os dados gravados em cada instante não sejam apagados para gravação de novos dados.

A gravação da mensagem ocorre na linha 7, juntamente com a hora em que a mensagem foi gravada em horas, minutos, segundos e milissegundos. As linhas 8 e 9 seguintes são responsáveis pela liberação do gravador de dados do arquivo, para liberação de memória e garantia de que os dados foram gravados adequadamente. A formatação da hora de gravação é feita pela função `getNow()`, na linha 17.

Note-se que a função da linha 1 não exige que o arquivo tenha formato HTML. Contudo, todos os arquivos usados neste trabalho foram escritos nesse formato. Na Figura 4.6, pode-se verificar como se estruturam esses arquivos HTML. O primeiro caractere de cada linha do arquivo identifica a tela em que a interação ocorreu. No caso do aplicativo de anotações, caixas de diálogo podem aparecer quando o usuário interage na tela principal. Esses diálogos são representados por números em forma de subitens. A segunda informação de cada linha é a ação executada pelo usuário, seguida pelo instante em que o usuário a realizou. As ações possíveis de serem executadas são as seguintes:

- ADD: pressionamento o botão “Add”;
- Audio\_Ann\_Disabled: finalização de gravação de áudio;
- Audio\_Ann\_Enabled: indicação de que o áudio está pronto para ser gravado;
- AUDIO\_ANNOTATION: realização de anotação de áudio;
- CANCEL: cancelamento da operação anterior (retorno à tela anterior);
- CHOOSE\_AUTHOR: escolha do autor do vídeo;

```

1 START: 01:54:27 Brasilia Time
2 CHOOSE_USERNAME 0: 01:54:28 Brasilia Time
3 CHOOSE_VIDEO 0: 01:54:30 Brasilia Time
4 CHOOSE_AUTHOR 0: 01:54:32 Brasilia Time
5 EditText_TOUCH: 01:54:39 Brasilia Time
5 EditText_TOUCH: 01:54:39 Brasilia Time
5 EditText_TOUCH: 01:54:39 Brasilia Time
5 EditText_TOUCH: 01:54:39 Brasilia Time
5 EditText_FOCUSED 01:54:39 Brasilia Time
5 ADD: 01:54:46 Brasilia Time
5 EditText_UnFOCUSED 01:54:46 Brasilia Time
5 TEXT_ANNOTATION: 01:54:46 Brasilia Time

5 TA 01:54:49 Brasilia Time
  5.1 Edit:01:54:51 Brasilia Time
    5.2 Edit: 01:54:56 Brasilia Time
      5.3 OK: 01:54:58 Brasilia Time

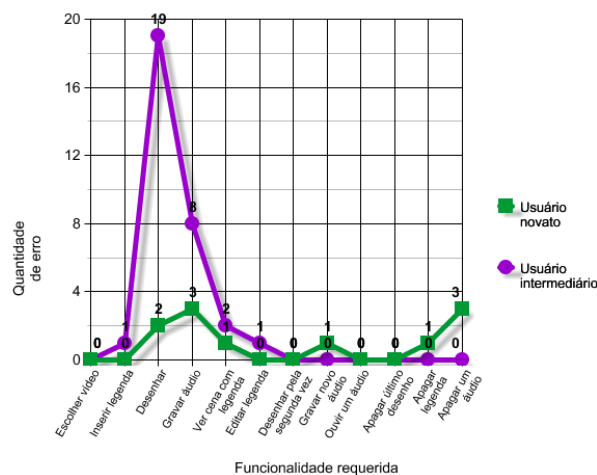
```

Figura 4.6: Trecho de um dos arquivos gravados no cartão SD do dispositivo móvel.

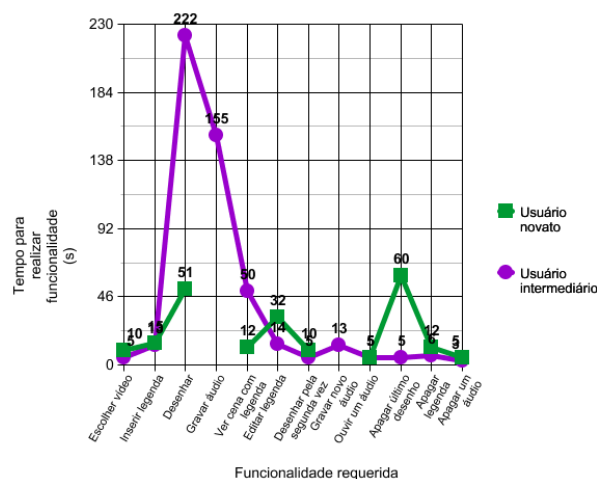
- CHOOSE\_USERNAME: escolha do nome do usuário associado à conta do Google;
- CHOOSE\_VIDEO: escolha do vídeo a ser anotado;
- CONTEXT\_INFO: visualização de informações sobre as anotações realizadas;
- EditText\_FOCUSED ou EditText\_TOUCH: seleção da caixa de texto para realizar anotação de texto;
- EditText\_UnFOCUSED: término da utilização da caixa de texto;
- Ink\_Ann\_Disabled: desativação da anotação de texto;
- Ink\_Ann\_Enabled: indicação de que a anotação de tinta eletrônica está pronta para uso;
- INK\_ANNOTATION: realização da anotação de vídeo;
- Navigate to: busca do usuário por um trecho do vídeo por meio da barra de busca;
- Pushed PAUSE: pausa do vídeo;
- Pushed PLAY: liberação do vídeo, que estava pausado;
- START: pressionamento do botão start;
- TEXT\_ANNOTATION: realização de anotação de texto.

Na Figura 4.7, podem-se confrontar as quantidades de erros que cada usuário cometeu para realizar cada funcionalidade requerida ou até desistir de realizá-la. Em seguida, na Figura 4.8, realiza-se um comparativo do tempo que cada usuário demorou para realizar cada uma das atividades ou até ele desistir de realizá-la.

Os arquivos de registros dos usuários novato e intermediários estão incluídos nos Apêndices A e B, respectivamente. Pelo Gráfico 5, nota-se que o usuário intermediário realizou um número bem maior de erros que ao usuário novato quando as atividades lhes eram solicitadas pela primeira vez. A Figura 4.8 permite concluir que o usuário intermediário também demora



**Figura 4.7:** Quantidade de erros cometidos por cada usuário para realizar cada atividade a ele requerida. Trecho de um dos arquivos gravados no cartão SD do dispositivo móvel.



**Figura 4.8:** Tempo gasto por cada usuário para realizar cada atividade a ele requerida. Trecho de um dos arquivos gravados no cartão SD do dispositivo móvel.

consideravelmente mais tempo para realizar as funcionalidades pela primeira vez, exceto no caso da escolha de vídeos, que contém apenas um fluxo de interação no aplicativo e praticamente não permite que o usuário erre.

É necessário ressaltar que o usuário novato simplesmente não conseguiu realizar gravações de áudio e desistiu de realizar essa funcionalidade, ao passo que o usuário intermediário conseguiu realizar todas as funcionalidades requeridas. Os gráficos permitem concluir que, apesar de o usuário intermediário ter demorado mais tempo para realizar as atividades pela primeira vez, ele gastou menos tempo para realizá-las pela segunda vez, em comparação com o usuário novato. Isso é um indício de que o usuário intermediário aproveita melhor as interações iniciais com o aplicativo para aprender por experiência.

Pela análise dos vídeos dos experimentos, é notável o motivo pelo qual o usuário intermediário erra mais e demora mais tempo para realizar as funcionalidades pela primeira vez: enquanto o usuário novato visualizou os componentes da tela e leu os comandos antes de realizar cada ação, o usuário intermediário simplesmente pressionou os botões aleatoriamente até encontrar a tela que apresentava corretamente a funcionalidade que ele desejava realizar. O experimento permitiu, portanto, concluir que o usuário intermediário assumiu que ele conseguiria operar

a interface antes mesmo de conhecê-la, apoiando-se no fato de que ele poderia voltar para a tela anterior cada vez que uma tela indesejada aparecesse por consequência da escolha de um componente errado.

Tendo em vista que os experimentos com usuários foram realizados como forma de validar as avaliações heurísticas conduzidas pelos especialistas, a especificação dos erros cometidos por cada usuário ao longo das interações se torna necessária. Para fins de comparação com os resultados esperados no cenário ideal, as ações corretas para concluir cada atividade requisitada serão apresentadas seguidas dos erros cometidos por cada usuário.

### 1. Escolher vídeo:

#### (a) Cenário ideal:

- i. Pressionar o botão “start”;
- ii. Selecionar algum elemento da lista de usuários;
- iii. Selecionar vídeo na lista de vídeo.

(b) Erros cometidos pelo usuário novato: não houve.

(c) Erros cometidos pelo usuário intermediário: não houve.

### 2. Inserir legenda:

#### (a) Cenário ideal:

- i. Pressionar o campo de texto;
- ii. Digitar a legenda;
- iii. Pressionar a tecla “Enter” ou o botão “Add”.

(b) Erros cometidos pelo usuário novato: não houve.

(c) Erros cometidos pelo usuário intermediário:

- i. Pressionar o botão “Ink Navigation is Off”.

### 3. Desenhar sobre uma cena:

#### (a) Cenário ideal:

- i. Pressionar o botão “Ink Navigation is Off”;
- ii. Passar o dedo sobre o próprio vídeo;
- iii. Pressionar o botão “Ink Navigation is On” (etapa não obrigatória, mas desejável).

(b) Erros cometidos pelo usuário novato:

- i. Pressionar o botão “Video Context Information”;
- ii. Pressionar o botão “Ink Navigation”.

(c) Erros cometidos pelo usuário intermediário:

- i. Pressionou o botão “Ink Navigation”;
- ii. Tentou desenhar passando o dedo sobre o espaço preto livre na tela “Ink Navigation”, entre a mensagem no cabeçalho e o botão “Cancel” no rodapé;
- iii. Pressionou a mensagem estática que solicita que ele selecione uma miniatura de figura, na tela “Ink Navigation”;
- iv. Repetiu o erro ii;
- v. Pressionou o botão “Add”;

- vi. Pressionou o botão “Video Context Information”;
- vii. Repetiu o erro i;
- viii. Repetiu o erro v;
- ix. Repetiu o erro vi;
- x. Pressionou o botão “Share”;
- xi. Repetiu o erro i;
- xii. Repetiu o erro ii. Em seguida, pediu para tentar mais tarde e pulou para a próxima atividade;
- xiii. Repetiu o erro i;
- xiv. Repetiu o erro v;
- xv. Repetiu o erro i;
- xvi. Repetiu o erro x;
- xvii. Pressiona o campo de texto. Diz novamente que quer tentar mais tarde;
- xviii. Repetiu o erro i;
- xix. Repetiu o erro ii. Em seguida, realiza a atividade.

#### 4. Gravar áudio sobre uma cena:

##### (a) Cenário ideal:

- i. Pressionar o botão “Audio Annotation is Off”;
- ii. Falar a mensagem;
- iii. Pressionar o botão “Audio Annotation is On”.

##### (b) Erros cometidos pelo usuário novato:

- i. Pressionou o botão “Video Context Information”;
- ii. Pressionou o botão “Audio Navigation”;
- iii. Pressiona o botão de informações sobre o dispositivo (bateria, wi-fi, data e hora, etc.) para procurar alguma informação que o ajude. Em seguida desiste.

##### (c) Erros cometidos pelo usuário intermediário:

- i. Pressionou o botão “Add”;
- ii. Pressionou o botão “Audio Navigation”;
- iii. Pressionou o botão “Ink Annotation is On”;
- iv. Pressionou o campo de texto;
- v. Pressionou o botão de comando de voz do teclado, imaginando que o áudio poderia ser gravado por esse botão;
- vi. Repetiu erro ii. Em seguida, pediu para realizar a atividade mais tarde;
- vii. Repetiu o erro iv;
- viii. Repetiu o erro v. Em seguida, finalizou a atividade corretamente.

#### 5. Visualizar cena que contém a legenda inserida:

##### (a) Cenário ideal:

- i. Pressionar o botão “Textual Navigation”;
- ii. Pressionar sobre a imagem correspondente à legenda ou sobre a própria legenda.

##### (b) Erro cometido pelo usuário novato:

- i. Pressionou o botão “Edit”, mas percebeu o erro rapidamente. Em seguida, finaliza com sucesso.

(c) Erros cometidos pelo usuário intermediário:

- i. Pressionou o botão “Ink Annotation”;
- ii. Tentou visualizar a legenda assistindo ao vídeo e procurando a legenda pela barra de progresso do próprio vídeo. Em seguida, finaliza a atividade com sucesso.

**6. Editar legenda:**

(a) Cenário ideal:

- i. Pressionar o botão “Ink Navigation”;
- ii. Pressionar o botão “Edit”;
- iii. Pressionar o novo botão “Edit” que aparece na caixa de diálogo;
- iv. Digitar o novo texto;
- v. Pressionar o botão “Ok”.

(b) Erros cometidos pelo usuário novato: não houve.

(c) Erro cometido pelo usuário intermediário:

- i. Pressionou o botão que o enviava para a cena que continha a legenda. Em seguida, realiza corretamente a funcionalidade.

**7. Desenhar sobre uma cena, pela segunda vez:**

(a) Erros cometidos pelo usuário novato: não houve.

(b) Erros cometidos pelo usuário intermediário: não houve.

**8. Gravar um novo áudio:**

(a) Erro cometido pelo usuário novato:

- i. Pressiona o botão de informações sobre o dispositivo (bateria, wi-fi, data e hora, etc.) para procurar alguma informação que o ajude. Em seguida desiste novamente e se irrita.

(b) Erros cometidos pelo usuário intermediário: não houve.

**9. Ouvir um dos áudios gravados:**

(a) Cenário ideal:

- i. Pressionar o botão “Audio Navigation”;
- ii. Pressionar sobre a mensagem de áudio ou sobre a imagem equivalente.

(b) Erros cometidos pelo usuário novato: não houve.

(c) Erros cometidos pelo usuário intermediário: não houve.

**10. Apagar o último desenho:**

(a) Cenário ideal:

- i. Pressionar o botão “Ink Navigation”;
- ii. Pressionar o último botão “Clear” da lista;
- iii. Pressionar o botão “Ok” na caixa de diálogo.

- (b) Erros cometidos pelo usuário novato: não houve.
- (c) Erros cometidos pelo usuário intermediário: não houve.

#### 11. Apagar uma das legendas:

- (a) Cenário ideal:
  - i. Pressionar o botão “Textual Navigation”;
  - ii. Pressionar o botão “Edit”;
  - iii. Pressionar o botão “Remove”;
- (b) Erro cometido pelo usuário novato:
  - i. Pressionou o botão “Share” sem querer. Logo depois completou com o sucesso a atividade.
  - ii. Erros cometidos pelo usuário intermediário: não houve.

#### 12. Apagar uma das gravações de áudio:

- (a) Cenário ideal:
  - i. Pressionar o botão “Audio Navigation”;
  - ii. Pressionar o botão “Remove”;
  - iii. Pressionar o botão “Ok”.
- (b) Erros cometidos pelo usuário novato:
  - i. Usuário tenta iniciar a atividade, um pouco irritado. Pressiona corretamente o botão “Audio Navigation”, mas acha que errou e volta para a tela principal;
  - ii. Pressiona o botão “Audio Annotation is On”;
  - iii. Pressiona a descrição do áudio e navega para a cena que o contém. Em seguida, realiza corretamente a atividade.
- (c) Erros cometidos pelo usuário intermediário: não houve.

Como se percebe pelo Gráfico 4.7, as atividades relacionadas à gravação de áudio foram problemáticas para ambos os usuários, porque nenhum dos dois percebeu que o áudio estava sendo gravado logo depois de eles terem pressionado o botão correto. O usuário novato realizou duas gravações de áudio involuntariamente quando lhe foi solicitada essa atividade. Contudo, pela ausência de aviso do sistema, ele pensou que nenhuma mensagem havia sido gravada. Quando lhe foi solicitado que ouvisse um áudio gravado, ele percebeu pela tela de navegação que havia gravações feitas. Essa conclusão o permitiu remover o último áudio, operação que ele realizou sem erros, mas ficou na dúvida se o áudio era de fato o último, porque não havia informação sobre a hora da gravação.

Desenhar sobre uma cena foi uma atividade interrompida pelo usuário intermediário por duas vezes até que ele a acertasse.

Ao ser solicitado para realizar o segundo desenho, o usuário novato reclamou que o primeiro desenho que ele havia feito não havia saído da tela. Além disso, ele reclamou que os botões da tela principal estavam muito próximos.

Analisando-se os erros cometidos pelos usuários, nota-se que apenas um deles não seria evitado pelos resultados das avaliações heurísticas para interfaces de dispositivos móveis realizadas: quando o usuário intermediário pressionou o botão de comando de voz, ele pensou por um momento que estivesse realizando a funcionalidade de gravação corretamente, mas logo

percebeu que havia se enganado. Apesar de ele ter conseguido se recuperar do estado de erro, nenhum dos especialistas das avaliações heurísticas percebeu que o botão de comando de voz poderia causar essa ambiguidade. No sistema Android, o comando de voz simplesmente permite que o usuário preencha um campo de texto por meio da fala, ao invés preenchê-lo por meio da digitação da mensagem pelo teclado. Tendo em vista que esse problema poderia ter sido identificado pela heurística “Adequação do componente à funcionalidade”, nenhuma alteração das heurísticas foi realizada após os experimentos com os usuários.

#### 4.2.3 Avaliação heurística do UOL Notícias

Os resultados da avaliação heurística realizada com o aplicativo UOL Notícias nos dispositivos iPhone 4S, Motorola Blur e Motorola XOOM estão apresentados nas Tabelas 4.22, 4.23 e 4.24, respectivamente.

**Tabela 4.22:** Resultado da avaliação heurística do UOL Notícias utilizando-se o iPhone 4S

<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Não existe ferramenta de busca de notícias.	3	2
Na tela de Notícias, existe uma barra horizontal que especifica o tipo de notícia que estamos acessando. Nela para trocarmos o tipo visualizado, devemos clicar sobre ele, não existindo a possibilidade de arraste horizontal (prática muito comum em celulares touchscreen).	3	1
Na aba “ler depois” não existe a possibilidade de remover mais de uma notícia ao mesmo tempo, sendo necessário realizar a operação para cada notícia existente.	8	3
Informações referentes ao vídeo na aba Vídeos estão muito pequenas, dificultando a visualização das mesmas.	3	2
Lista de cidades está com informações mal ordenadas, dificultando a busca do usuário.	3	2
Banner localizado em cima da barra de guias está dificultando a visualização das informações.	3	2
Não existe a possibilidade de visualização em modo paisagem das fotos apresentadas na guia Fotos.	3	3
Não existe a possibilidade de visualização em modo tela-cheia dos vídeos apresentados na guia Vídeos.	3	3
Na tela inicial, as informações estão mal orientadas, dificultando sua visualização.	3	3

**Tabela 4.23:** Resultado da avaliação heurística do UOL Notícias utilizando-se o Motorola Blur

<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
As notícias estão muito próximas.	1	2
Não existe um padrão de interação na tela inicial. A notícia em destaque é mudada rolando a tela horizontalmente, ao contrário do que ocorre com as demais notícias.	2	2
Os nomes das cidades são carregados todos de uma vez. Poderiam ser carregados aos poucos, à medida que o usuário procura por elas.	3	3
Os nomes das cidades não estão ordenados no padrão que o usuário conhece.	3	3
A barra de progresso fica girando sem que se exiba uma mensagem para o usuário. Conexões lentas podem levar o usuário a pensar que o aplicativo travou.	6	4
A barra de <i>status</i> do dispositivo desaparece quando o vídeo é solicitado para ser carregado.	9	4
Não encontrei o botão “ler depois” que o aplicativo cita nas notícias.	2	3
O aplicativo não dá opção de selecionar mais de uma notícia offline de uma só vez.	8	4
Falta a possibilidade de buscar notícia offlines no aplicativo. 8 3 O vídeo não está aparecendo. Só consigo ouvir o som, porque a imagem não aparece.	6	4
As abas não são alternadas pelo movimento de deslizamento lateral.	3	2

**Tabela 4.24:** Resultado da avaliação heurística do UOL Notícias utilizando-se o Motorola XOOM.

<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Interface está inconsistente, porque a tela de visualização da notícia é diferente quando se pressiona no texto, em relação a quando se pressiona a própria imagem.	2	2
Elementos da tela estão muito desalinhados. A impressão que se tem é de que os elementos estão jogados na tela.	3	2
A interface carece de flexibilidade. Nada é personalizável e as potencialidades do dispositivo não são exploradas.	5	3
Informação sobre o vídeo que está sendo visualizado está ilegível.	3	2
As notícias estão difíceis de serem encontradas, porque não encontrei nenhuma possibilidade de buscá-las.	8	3
Usuário precisa acessar várias telas para visualizar um conteúdo qualquer, como texto, vídeo, etc.. Nesse processo, ele precisa lembrar o que preencheu no início da interação.	11	4
Na tela de fotos, o desenho do botão de miniaturas é pouco intuitivo, pouco esclarecedor.	4	1
As notícias estão divididas por data, mas quando se rola a tela, o usuário mal percebe as datas que aparecem, porque elas são pequenas e sem contraste.	3	2
Em retrato, as informações da imagem parecem estar desvinculadas dela, porque estão muito distantes.	2	1
A informação da foto em que o usuário está é muito pequena em relação ao tamanho da tela. Passa despercebida.	3	3

Continua na página seguinte...

**Tabela 4.24 – Continuação.**

<b>Problema</b>	<b>Heurística</b>	<b>Severidade</b>
Na tela inicial, as fotos não contêm a data, como ocorrem nas notícias. Falta padronização das informações exibidas.	2	1

Pelas Tabelas 4.22, 4.23 e 4.24, contabilizam-se 31 problemas de usabilidade, sendo que todos eles puderam ser associados a alguma heurística para avaliação de interfaces de dispositivos móveis. Essa análise individual permitiu concluir que as heurísticas propostas puderam ser usadas para a avaliação de interfaces de dispositivos de diferentes resoluções e tamanhos de tela e de diferentes sistemas operacionais. Os estudos encontrados na Literatura Científica que propunham heurísticas para o contexto móvel não consideraram dispositivos com características tão discrepantes nas avaliações realizadas.

### 4.3 Criação das diretrizes

A primeira atividade realizada para a criação do documento unificado de diretrizes para o *design* de interfaces de dispositivos móveis, com foco nos componentes de interface disponíveis, foi o mapeamento desses componentes com base na terminologia de cada empresa detentora de cada documento. Esse mapeamento foi realizado por meio da leitura dos documentos disponibilizados por essas empresas para dispositivos equipados com Android, Blackberry, iOS e Windows Phone e enumeração de cada elemento de interface citado ou ilustrado em cada documento. A representação visual do componente associada ao seu funcionamento permitiu afirmar que um dado elemento de uma documentação é análogo a outro elemento de outra documentação.

O mapeamento dos componentes de interface de cada sistema operacional está sumarizado na Tabela 4.25. Por meio dele, concluiu-se que a terminologia diferenciada não é o único problema que os interessados no *design* de interfaces para esses sistemas operacionais enfrentam, porque existem componentes que não estão disponíveis em todos os sistemas analisados, como se percebem nas células em branco da Tabela 4.25.

Os elementos de interfaces da Tabela 4.25 possuem nomes em Inglês, porque todas as documentações encontradas estavam escritas neste idioma. Por conta disso, optou-se pela criação de um nome em Português para caracterizá-lo no documento unificado, exceto em casos em que a terminologia do elemento era a mesma para todos os sistemas e para os quais a tradução era difícil de ser realizada.

**Tabela 4.25:** Mapeamento da terminologia de componentes usada pelas empresas Google, Blackberry, Apple e Microsoft.

<b>Componente</b>	<b>Android</b>	<b>Blackberry</b>	<b>iOS</b>	<b>Windows Phone</b>
Barra de abas	Top Bar/ Tab Bar		Tab Bar	Pivot Control
Barra de atividades	Main Action Bar		Navigation Bar	
Barra de navegação	Navigation Bar		Navigation Bar	
Barra de progresso	Progress Bar	Progress Indicator	Progress View	Progress Bar
Barra de Status	Status Bar	Status Bar	Status Bar	Status Bar
Barra deslizante de ajuste	Seekbar/ Slider	Slider	Slider	Slider
Barra inferior	Bottom Bar	Toolbar	Toolbar	Application Bar
Botão	Button	Button	Button	Button
Caixa de múltipla escolha	Spinner	Drop-down List	Simulado com TableView	Simulado com List-Picker

Continua na página seguinte. . .

**Tabela 4.25 – Continuação.**

<b>Componente</b>	<b>Android</b>	<b>Blackberry</b>	<b>iOS</b>	<b>Windows Phone</b>
Caixa de seleção	Checkbox	Checkbox	Checkbox	Simulado com Toggle Switch
Campo de busca	Search View	Search Field	Search Bar	Autocomplete Box
Campo de texto	Textfield	Textfield	Textfield	TextBox
Diálogo	Dialog	Dialog Box	Alert	Message Box
Girador de progresso	Progress Bar	Activity Indicator	Activity indicator	
Link	Checked Textview	Hyperlink	Link	Hyperlink Button
Lista	List View	List	Picker	ListBox
Menu popup	Popup Menu	Pop-up menu		
Radio Button	Radio Button	Radio Button	Radio Button	Radio Button
Seletor de arquivos		Filepicker		
Seletor de data	Datepicker	Datepicker	Datepicker	Datepicker
Seletor de hora	Timepicker	Timepicker	Timepicker	Timepicker
Seletor numérico	Number Picker		Stepper	Numeric Up Button
Submenu	Submenu			Context Menu
Switch	Toggle Button	Simulado com Labeled Switch	Switch	Toggle Switch
Tabela	Simulado com Grid	Table	Table View	Simulado com Grid
Texto (Rótulo)	TextView	Label	Label	Label

A associação dos componentes da Tabela 4.25 permitiu a criação das diretrizes para os componentes que serão mostrados nos Quadros 4.1 a 4.25 a seguir. As ilustrações de cada componente presente neste documento podem ser encontradas no Apêndice C.

<p><b>BARRA DE ABAS</b></p> <p>UTILIDADE: permitir que o usuário alterne sobre diferentes visões ou modos de exibição do aplicativo.</p> <p>DIRETRIZES GERAIS:</p> <p>Não coloque o título do aplicativo na barra de abas;</p> <p>Se você não usar uma barra de atividades, coloque o logo do aplicativo à esquerda das abas da barra de abas.</p> <p>Não disponibilize controles ao usuário por meio das abas. Elas devem apenas alternar modos de exibição, alternar entre tarefas, etc., mas não devem disponibilizar funcionalidades do aplicativo, como busca ou cadastro de pessoas.</p> <p>Mantenha todas as abas na tela, mesmo que não haja visão a ser mostrada por uma delas em um determinado contexto. Se necessário, desabilite a aba, mas nunca a remova da interface.</p> <p>Não mude a localização ou orientação das abas quando o dispositivo for alterado da posição retrato para paisagem (e vice-versa).</p> <p>Tente não usar mais do que 4 abas, porque os usuários podem se perder quando muitas abas são disponibilizadas.</p> <p>Use até duas palavras para descrever cada aba.</p> <p>DIRETRIZES ESPECÍFICAS:</p> <p>ANDROID:</p> <p>O Android disponibiliza dois tipos de abas: fixas e roláveis. Use abas fixas quando o aplicativo possui no máximo 3 visões. Caso contrário, use abas roláveis.</p>
--

**Quadro 4.1:** Barra de Abas.

**BARRA DE ATIVIDADES**

**UTILIDADE:** identificar o aplicativo e prover ações ou modos de navegação ao usuário.

**DIRETRIZES GERAIS:**

Coloque o logo do aplicativo à esquerda na barra de atividades.

À direita do logo, identifique a tela em que o usuário está por meio de um título.

Mantenha a barra de atividades consistente ao longo de todas as telas do aplicativo.

Disponibilize na barra de atividades apenas funcionalidades que são fundamentais à interação, com verbos autoexplicativos, como *Buscar* ou *Compartilhar*.

Se você quiser permitir que o usuário alterne entre diferentes visões, coloque as visões em uma caixa de múltipla escolha na barra de atividades.

**DIRETRIZES ESPECÍFICAS:****ANDROID:**

Porções da barra de atividades que não cabem na tela são colocadas em uma área adicional identificada por "...". Nunca deixe que uma funcionalidade importante esteja disponível nesse menu adicional.

Em paisagem, a barra inferior é unida automaticamente à barra de atividades, para melhor aproveitamento de espaço. Analise a interface para evitar que funcionalidades importantes sejam colocadas no menu adicional.

**Quadro 4.2:** *Barra de Atividades.*

**BARRA DE NAVEGAÇÃO**

**UTILIDADE:** permitir a navegação por diferentes telas e gerenciar o conteúdo de telas do aplicativo.

**DIRETRIZES GERAIS:**

Use o título da tela atual como título da barra de navegação.

A fonte-padrão provê o máximo de facilidade de leitura, mas isso não impede que ela seja mudada.

Neste caso, certifique-se que o texto da barra de navegação está legível.

Evite preencher a barra de navegação com muitos controles adicionais, mesmo que haja espaço suficiente.

A barra de navegação pode ser formada por diferentes componentes. Use cada componente de acordo com a documentação que ele possui.

**DIRETRIZES ESPECÍFICAS:****iOS:**

A barra de navegação deve conter um título, um botão para voltar para a tela anterior e um botão para controlar o conteúdo da tela sendo usada.

Se você usar a barra de atividades para colocar botões em abas, não use título. Use apenas os botões que formam tais abas.

Se apropriado, adapte a aparência da barra de navegação de acordo com seu aplicativo. Por exemplo, você pode fornecer a imagem customizada do seu plano de fundo ou a tonalidade da barra e também especificar a translucidez. Em alguns casos, pode ser uma boa idéia exibir uma imagem de fundo redimensionada. Tenha certeza de que a aparência da customização da sua barra de navegação é consistente com a aparência e estilo do seu aplicativo. Se você usar a barra de navegação translúcida, por exemplo, não a combine com uma barra de ferramentas opaca.

Analise os elementos da barra de navegação tanto em retrato quanto em paisagem, porque a barra sobre redimensionamento de acordo com a mudança da orientação.

**Quadro 4.3:** *Barra de Navegação.*

**BARRA DE PROGRESSO**

**UTILIDADE:** informar ao usuário de que o sistema está executando uma tarefa e quando essa tarefa irá finalizar.

**DIRETRIZES COMUNS:**

Use a barra de progresso em atividades para as quais se pode estimar o tempo de finalização.

Informe o usuário sobre o progresso de qualquer atividade que demore mais que dois segundos.

Disponibilize ao usuário informações importantes sobre o progresso da tarefa sendo realizada. Por exemplo, caso a instalação de um sistema esteja ocorrendo no dispositivo, mostre ao usuário o percentual da instalação que já foi completado.

Em casos de atividades longas realizadas em mais de uma fase, separe as atividades em barras de progresso distintas, cada qual com uma descrição relevante, tal como ‘‘Baixando o aplicativo’’ ou ‘‘Instalando’’.

Use verbos no gerúndio para indicar atividades em andamento. Por exemplo, ‘‘Conectando’’.

Use verbos no particípio passado para indicar que a atividade parou. Por exemplo, ‘‘Cancelado’’.

**DIRETRIZES ESPECÍFICAS:****BLACKBERRY:**

Permita que o usuário omita o componente da tela, caso ele queira.

**Quadro 4.4:** *Barra de Progresso.*

**BARRA DE STATUS**

**UTILIDADE:** exibe informações importantes sobre o dispositivo, como data, hora e nível de carga da bateria.

**DIRETRIZES GERAIS:**

Não oculte a barra de status a menos que seu aplicativo seja um jogo ou exiba elementos em tela cheia.

Oculte a barra de status e todos os demais elementos da interface quando o usuário deseja ampliar um elemento específico para visualizá-lo em tela cheia. Por exemplo, caso o usuário queira visualizar uma imagem em tela cheia, exiba apenas a imagem e disponibilize os comandos quando o usuário colocar o dedo sobre a tela.

Não customize a barra de status.

A barra de status pode ser formada por diferentes componentes. Use cada componente de acordo com a documentação que ele possui.

Analise os elementos da barra de navegação tanto em retrato quanto em paisagem, porque a barra sobre redimensionamento de acordo com a mudança da orientação.

**DIRETRIZES ESPECÍFICAS:****iOS:**

Escolha a cor da barra de status dentre as seguintes possíveis: cinza (cor padrão), preto opaco e preto translúcido.

**Quadro 4.5:** *Barra de Status.*

**BARRA DESLIZANTE DE AJUSTE**

UTILIDADE: permitir que o usuário ajuste de valores dentro uma escala de valores possíveis.

Diretrizes gerais:

Coloque imagens como ícones em cada lado da barra para mostrar ao usuário os extremos da escala.

Em caso de os valores extremos serem numéricos, prefira usar um seletor numérico à barra deslizante de ajuste.

**Quadro 4.6:** *Barra Deslizante de Ajuste.*

**BARRA INFERIOR**

UTILIDADE: permitir que os usuários acessem com facilidade funcionalidades frequentes de um aplicativo.

DIRETRIZES COMUNS:

Use a barra inferior para exibir ao usuário ações frequentes do aplicativo ou da tela com a qual ele está interagindo.

Coloque mensagens positivas ou referentes a ações menos desastrosas à esquerda, antes dos demais.

Por exemplo, opções ‘‘Salvar’’

e ‘‘Enviar’’ devem aparecer à esquerda da opção ‘‘Remover’’.

Não seja redundante. Se uma ação estiver explícita no conteúdo principal da tela, não a replique na barra inferior.

Use no máximo 5 itens caso a tela seja exibida em retrato. Use até 7 itens caso a tela esteja exibida em paisagem.

A barra inferior pode ser formada por diferentes componentes. Use cada componente de acordo com a documentação que ele possui.

Analise os elementos da barra de navegação tanto em retrato quanto em paisagem, porque a barra sobre redimensionamento de acordo com a mudança da orientação.

DIRETRIZES ESPECÍFICAS:

BLACKBERRY:

Apenas os dispositivos Storm Series e Torch 9800 dispõem de barras inferiores.

Use ícones com tamanho aproximado de 33x33 pixels.

iOS:

Personalize a aparência da barra de ferramentas para que ela combine com a aparência geral da sua aplicação. Use ícones com tamanho médio de 44x44 pixels.

**Quadro 4.7:** *Barra Inferior.*

**BOTÃO**

UTILIDADE: São usados para selecionar itens ou opções.

DIRETRIZES COMUNS:

Não inclua símbolos de marcas em botões.

Use verbos para indicar as ações dos botões e seja sucinto na escolha.

Não use títulos de botões que não tenham significado relevante para a funcionalidade sendo realizada.

A mensagem que especifica o botão nunca deve possuir mais do que suas palavras.

Sempre use a fonte padrão do dispositivo, a menos que você esteja referenciando fontes específicas de marcas ou de convenções formais (sociais, etnográficas, etc.) para um dado contexto.

**Quadro 4.8:** *Botão.*

**CAIXA DE MÚLTIPLA ESCOLHA**

**UTILIDADE:** permite que o usuário selecione um item de uma lista de itens mutuamente exclusivos.

**DIRETRIZES COMUNS:**

Use caixas de múltipla escolha quando se têm dois ou mais elementos que podem ser escolhidos e quando o espaço na tela for uma limitação do design.

Caso espaço na tela não seja um problema, prefira usar “radio buttons”.

O valor-padrão, que aparece no componente antes de o usuário pressionar sobre ele, deve ser o valor mais provável de ser selecionado pelo usuário.

Destaque o item que está sendo selecionado para o usuário.

Nunca exija que o usuário selecione uma resposta que ele eventualmente não queira. Disponibilize opções “Outro” ou “Nenhum”, por exemplo.

Opção de escape como “Outro” e “Nenhum” devem ser o primeiro item da lista, logo após o valor-padrão.

Evite textos muito longos para os valores do campo, porque eles podem ser truncados em telas menores, dificultando a compreensão.

Não use opções dicotômicas, como “Sim” e “Não”, em campos de múltipla escolha. Use caixas de seleção para esse propósito.

**Quadro 4.9:** Caixa de múltipla escolha.

**CAIXA DE SELEÇÃO**

**UTILIDADE:** permitir que o usuário escolha dentre duas opções binárias, facilmente diferenciáveis.

Normalmente, caixas de seleção são usadas em grupo, para que o usuário selecione mais de uma resposta a uma mesma pergunta.

**DIRETRIZES COMUNS:**

Não inicie uma atividade quando o usuário selecionar uma caixa de seleção. Por exemplo, não abra uma nova tela após a seleção ocorrer.

Alinhe as caixas de seleção verticalmente.

Agrupe as caixas de seleção de maneira lógica, colocando os itens relacionados próximos e os itens mais frequentemente usados primeiro.

Associe cada caixa de seleção a mensagens formatadas em uma ou duas linhas, de forma padronizada por toda a interface. Essa padronização ajuda o usuário a localizar as informações que lhe interessam.

A descrição da caixa de seleção deve ter um antônimo facilmente identificado. Por exemplo, não use uma caixa de seleção associada à palavra “Retrato” para indicar se um componente será usado em retrato ou em paisagem, porque a diferenciação desses termos não é clara para todos os usuários.

**Quadro 4.10:** Caixa de seleção.

**CAMPO DE BUSCA**

**UTILIDADE:** buscar aplicativos a partir da área de trabalho ou buscar informações de um aplicativo em uso. Trata-se de um tipo especial de campo de texto.

**DIRETRIZES COMUNS:**

O campo de busca pode ser usado tanto parte do conteúdo principal da tela, quanto em menus ou barras de atividades.

O conteúdo que poderá ser buscado deverá ser relevante para o escopo da busca. Informações irrelevantes não devem aparecer como resultados da busca.

Se possível, exiba os resultados mais relevantes no topo da lista de resultados.

Nos resultados da busca, negrite as palavras que correspondem aos caracteres digitados pelo usuário.

Caso queira utilizar um rótulo antes do campo de busca, coloque dois pontos (:) ao final do rótulo.

Você pode incluir um texto no campo de busca, para indicar ao usuário a ação a ser realizada (por exemplo, "Buscar") ou para indicá-lo o contexto da busca (por exemplo, "Google").

**DIRETRIZES ESPECÍFICAS:****iOS:**

Você pode acrescentar à barra de busca do iOS um botão de favoritos. Esse botão funciona como um atalho para informações que os usuários querem encontrar mais facilmente.

Você pode escolher uma imagem para fundo da caixa de pesquisa, ou escolher uma cor de fundo dentre as seguintes possibilidades: azul, preto e preto translúcido. Em caso da escolha de imagem, certifique-se de que ela é redimensionável de acordo com a orientação.

**Quadro 4.11:** *Campo de busca.*

**CAMPO DE TEXTO**

**UTILIDADE:** permitir digitação de texto para entrada de dados.

**DIRETRIZES COMUNS:**

Use um campo de texto com suporte à funcionalidade que se deseja. Por exemplo, campos de email devem ser associados a campos de texto próprios para preenchimento de email, que incluem caracteres "@" e ".com" facilmente acessados. Além dos campos de texto comuns (para preenchimento de nomes, por exemplo), existem campos de texto específicos para os seguintes tipos de dados: data, hora, número, senha, número de telefone e endereços `\textit{Web}`.

Caso o espaço na tela seja uma limitação, coloque uma dica no próprio campo de texto, que desaparece quando o usuário seleciona o campo. Nesse caso, use uma dica concisa, com a primeira letra em maiúsculo.

Os rótulos que identificam campos de texto devem ter dois pontos (:) como último caractere.

Caso prefira, coloque imagens como ícones no lado esquerdo do campo de texto, como substituição do rótulo convencional. As imagens devem conter metáforas facilmente identificáveis.

Associe o campo de texto a um botão "Limpar" sempre que necessário.

Certifique-se de que o texto sendo fornecido pelo usuário está legível no componente.

**Quadro 4.12:** *Campo de texto.*

**DIÁLOGO**

**UTILIDADE:** mostrar ao usuário informações necessárias para completar uma atividade. informar ao usuário mensagens emergenciais ou que esclareçam o estado da atividade sendo realizada. avisar usuário sobre condições e situações adversas.

**DIRETRIZES COMUNS:**

Use botões para confirmar ou cancelar ações do usuário. Não use links ou outros componentes.

Use até três opções de respostas em diálogos. Caso sejam necessárias mais opções, opte por uma nova tela. Idealmente, os alertas possuem dois botões.

Permita que o diálogo desapareça quando o usuário pressionar a tecla “voltar” do dispositivo.

Centralize o componente de diálogo na tela.

Use diálogos que ocupem no máximo *90% da altura da tela do dispositivo*.

Use botões de confirmação e relativos a ações menos drásticas antes dos demais. Por exemplo, use a opção “Salvar” antes de “Remover” ou “Cancelar”.

Centralize os botões do diálogo.

Caso seja necessário incluir uma caixa de seleção ao diálogo, alinhe a caixa de seleção à mensagem do diálogo. A caixa de diálogo deve estar pré-selecionada, a menos que ela referencie uma ação crítica do aplicativo.

Seja sucinto, mas use mensagens completas.

Fale a linguagem do usuário. Por exemplo, prefira a frase “O arquivo não foi salvo porque não há espaço disponível no cartão de memória.” à frase “Erro de gravação de dados”.

Use mensagens positivas e nunca culpe o usuário. Em casos de erros, foque nas atividades que o usuário pode fazer para sair do estado de erro.

**Não**

Use reticências (**\ldots**) apenas para indicar o progresso de atividades, ao final dos rótulos.

Certifique-se de minimizar o número de diálogos da sua aplicação. Eles interrompem a interação e devem ser usados apenas para exibir informações realmente úteis em contextos isolados.

Tente manter a mensagem curta o suficiente para ser exibida em uma única linha.

Evite o uso de “você”, “seu”, “eu” ou “meu” tanto quanto possível, porque as pessoas podem se sentir constrangidas ou inibidas por essas mensagens.

Caso a mensagem seja um título ou uma frase afirmativa e curta, use a primeira letra de cada palavra em caixa alta. Por exemplo: “Atividade De Remoção”.

Caso a mensagem seja uma pergunta, coloque apenas a primeira letra da primeira palavra em caixa alta e acrescente ponto de interrogação ao final.

Use a palavra “tocar” ao invés de “toque” ou “clique” ou “escolha” para descrever a ação de seleção.

Teste a aparência do diálogo nas duas orientações possíveis, porque em paisagem a altura é limitada em relação ao que se nota em retrato.

Evite botões com significados binários em diálogos (botão “Sim” e botão “Não”, por exemplo). Prefira botões com as ações possíveis, como “Salvar” e “Cancelar”.

**Quadro 4.13:** *Diálogo.*

**GIRADOR DE PROGRESSO**

UTILIDADE: informar ao usuário de que o sistema está executando uma tarefa.

**DIRETRIZES COMUNS:**

Use o girador de progresso quando você lida com uma atividade para a qual não se consegue estimar o tempo de finalização.

Use o girador de progresso quando é mais importante mostrar ao usuário que a atividade que ele solicitou está em andamento do que mostrá-lo quando a atividade irá finalizar.

Informe o usuário sobre o progresso de qualquer atividade que demore mais que dois segundos.

Associe o componente a uma mensagem relevante ao usuário em casos de atividades complexas, que podem demorar um tempo considerável para serem executadas.

**DIRETRIZES ESPECÍFICAS:****BLACKBERRY:**

Permita que o usuário omita o componente da tela caso ele queira.

**Quadro 4.14:** *Girador de progresso.*

**LINK****UTILIDADE:****Diretrizes gerais:**

Use a primeira letra do link em caixa alta (maiúscula), a menos que o link esteja no fim de uma sentença.

Não use links no meio de uma mensagem. Faça da frase inteira um link ou use-o usá-lo no final da sentença. Por exemplo, prefira frases como “Para maiores informações, use o seguinte endereço: <link>” a “Use o endereço <link> para maiores informações”.

Use contraste no link em relação ao fundo e em relação ao restante da frase (em casos em que eles sejam parte da frase).

Não use símbolos de marcas em links.

Não use mais do que duas palavras para identificar um link.

Nunca use dois ou mais links próximos uns do outro, porque a diferenciação entre eles é difícil de ser notada visualmente.

Nunca desabilite um link, a menos que o aplicativo esteja esperando que uma atividade do sistema seja completada.

**Quadro 4.15:** *Link.*

**LISTA**

**UTILIDADE:** exibir uma listagem de elementos, da qual o usuário seleciona um elemento para ser usado em uma funcionalidade. A lista exibe os elementos em linhas.

**Diretrizes gerais:**

Rótulos de listas devem aparecer à esquerda da lista, terminado com o caractere dois pontos (:).

Use uma lista quando os usuários têm ideia de que valores irão encontrar, porque em listas grandes a maioria dos valores só é visualizada quando a tela é rolada.

Use divisores horizontais em listas para especificar o fim de uma seção, de um grupo de elementos.

Mantenha o mesmo alinhamento de elementos em todas as linhas da lista.

Certifique-se de que os itens da lista estão legíveis tanto em retrato quanto em paisagem.

Use fontes com bom contraste e sem serifa.

Não use efeitos de iluminação ou animações em itens de listas. Use iluminação apenas para indicar o item que foi selecionado pelo usuário.

**DIRETRIZES ESPECÍFICAS:**

**BLACKBERRY:**

Use botões “Anterior” e “Próximo” em casos de listas muito grandes.

Permita que os usuários naveguem pelos elementos da lista por meio das teclas “N” (next) e “P” (previous), no caso de listas de aplicativos em Inglês.

**iOS:**

Use uma tabela para exibir muitos valores, porque as listas no iOS são uma generalização dos componentes seletores, que têm altura e largura fixas, fazendo com que poucos elementos sejam exibidos por vez. Usando tabelas os usuários tendem a encontrar a informação que desejam mais rapidamente.

**Quadro 4.16:** *Lista.*

**MENU POPUP**

**UTILIDADE:** é uma maneira fácil de acessar as ações mais comuns acerca de um item da interface.

**Diretrizes gerais:**

Use menus popup apenas se eles forem agregar valor à funcionalidade.

Inclua no menu popup somente as ações mais comuns acerca do componente.

Inclua um ícone e um rótulo textual para identificar cada item do menu.

Opção de Ajuda deve estar distante das funcionalidades principais.

**DIRETRIZES ESPECÍFICAS:**

**BLACKBERRY:**

Você pode criar menus popup com nove ações, cinco ações ou três ações.

Os ícones do menu devem possuir em média 33x33 pixels de tamanho.

O item mais importante do menu deve estar com foco na interface. Os demais itens devem ser ordenados dos mais acessados para os menos acessados.

**Quadro 4.17:** *Menu Popup.*

**RADIO BUTTON**

**UTILIDADE:** indica um conjunto de itens mutuamente exclusivos, mas relacionados entre si.

**DIRETRIZES COMUNS:**

Use radio buttons para exibir duas ou mais opções ao usuário, quando o espaço da tela não for um fator limitante para a interface. Em caso de espaço ser uma limitação, prefira usar uma lista.

O conteúdo de radio buttons não deve mudar de acordo com o contexto. Portanto, assegure que os valores relativos a eles serão estáticos ao longo de toda interação.

Não inicie uma atividade nova quando o usuário selecionar uma opção em um radio button. Por exemplo, não abra uma nova tela.

Alinhe radio buttons verticalmente.

Agrupe os radio buttons de acordo com a relação que eles possuem entre si.

Prefira deixar os itens mais comuns antes dos demais.

Use rótulos do lado direito dos radio buttons.

Coloque as mensagens dos radio buttons com a primeira letra em maiúsculo.

Não use ponto final nos rótulos de radio buttons.

Use no máximo 8 radio buttons em um único grupo. Caso sejam necessários mais opções, prefira uma lista.

Nunca use radio buttons para exibir resultados de buscas a banco de dados, a menos que se saiba que a busca nunca retornará mais do que 8 valores.

Não use radio buttons para iniciar ações na interface, como desabilitar componentes ou mostrar um diálogo.

**Quadro 4.18:** *Radio Button.*

**SELETOR DE ARQUIVOS**

**UTILIDADE:** oferecer ao usuário uma opção prática de navegar pelos arquivos de aplicativos BlackBerry.

**Diretrizes:**

Mostre ao usuário itens de acordo com o contexto buscado. Por exemplo, se o usuário estiver navegando por um diretório que contém imagens, mostre miniaturas das imagens que existem nesse diretório.

Permita que os usuários naveguem a partir de um diretório-padrão, para que eles encontrem o arquivo que desejam com mais facilidade.

**Quadro 4.19:** *Seletor de Arquivos.*

**SELETOR DE DATA**

**UTILIDADE:** permitir que o usuário selecione com facilidade um dia, mês ou ano de uma data específica.

**DIRETRIZES COMUNS:**

Use valores que serão usados no aplicativo. Por exemplo, se você sabe que dias da segunda quinzena de um mês não serão selecionados, exclua-os dos valores possíveis.

Seletores de data devem ser usados preferencialmente em diálogos, porque eles ocupam muito espaço da tela.

**Seletor de hora**

**UTILIDADE:** permitir que o usuário selecione uma hora, minuto e segundo de um instante qualquer.

**DIRETRIZES COMUNS:**

Limite os valores a serem preenchidos se não for necessário saber com exatidão a hora do evento.

Por exemplo, organize os minutos de 10 em 10 minutos.

Seletores de data devem ser usados preferencialmente em diálogos, porque eles ocupam muito espaço da tela.

**Quadro 4.20:** *Seletor de data.*

**SELETOR NUMÉRICO**

**UTILIDADE:** incrementar ou decrementar o valor de um número.

**Diretrizes gerais:**

Use seletores numéricos quando os usuários querem selecionar dentre poucos valores numéricos possíveis.

Não use seletores numéricos caso os números possíveis variem constantemente de acordo com o contexto.

**DIRETRIZES ESPECÍFICAS:****iOS:**

No iOS, o seletor não é automaticamente associado a um número sendo exibido, porque ele só possui os botões “-” e “+”. Por isso, deixe claro que valor está sendo modificado por meio de destaque no elemento sendo modificado.

**Quadro 4.21:** *Seletor Numérico.*

**SUBMENU**

**UTILIDADE:** permitir que o usuário encontre itens frequentemente usados ou importantes, que são um subconjunto de um item do menu.

Diretrizes gerais:

Use submenus para reduzir o número de itens que aparecem no menu. Por exemplo, se o usuário precisar rolar a tela para visualizar os itens de um menu, agrupe os itens do menu e coloque opções em submenus.

Agrupe itens relacionados em submenus. Por exemplo, deixe a opção “Ordenar por” no menu principal e ofereça opções “Data”, “Nome” e “Assunto” no submenu.

Use pelo menos três itens em um submenu.

Caso sejam necessários mais do que seis itens em um submenu, separe-os por linhas separadoras.

Coloque os itens mais frequentemente usados primeiro e os separe dos demais com linhas de separação.

Ações importantes e gerais não devem ser colocadas em submenus.

Evite usar submenus de submenus. Mantenha apenas um nível de diferença em relação ao menu principal

Prefira usar verbos para descrever as ações no menu e substantivos nos submenus.

Nunca repita o texto no menu no submenu. Por exemplo, jamais coloque “Ordenar por” no menu e “Ordenar por data”, “Ordenar por nome”, etc. no submenu.

Coloque a primeira letra do menu e do submenu em maiúscula, mesmo que o submenu complete a frase iniciada no menu. Por exemplo, coloque “Enviar” no menu e “Por email” no submenu.

Evite usar o termo “Mais” como item de um menu. Geralmente, é difícil saber o que consta nos submenus de itens identificados por esse termo.

**Quadro 4.22:** *Submenu.*

**SWTCH**

**UTILIDADE:** fornecer uma maneira visual de se alternar entre dois estados mutuamente exclusivos.

Diretrizes gerais:

Escolha dois valores que tenham significados opostos. Por exemplo, “Ligado” e “Desligado”.

Você pode usar um switch para alternar o estado de outros componentes da interface. Dependendo da escolha, alguns elementos podem aparecer e outros podem ser desabilitados, por exemplo.

**Quadro 4.23:** *Switch.*

**TABELA**

**UTILIDADE:** apresentar informações em colunas e linhas. Em iOS, as tabelas só possuem uma única coluna.

**DIRETRIZES COMUNS:**

Use cabeçalhos para ajudar o usuário a navegarem por tabelas muito grandes.

Use tabelas para exibir grandes ou pequenas quantidades de informações.

Esclareça ao usuário qual item foi selecionado por ele, por meio de destaque da linha selecionada e /ou início de uma nova ação.

Caso muitas informações necessitem ser carregadas, disponibilize algumas informações textuais iniciais imediatamente ao usuário e carregue os demais itens ao poucos.

Não use tamanhos variáveis para as linhas da tabela.

Use textos sucintos para evitar que eles sejam truncados.

**DIRETRIZES ESPECÍFICAS:****iOS:**

Use índices flutuantes para auxiliar o usuário a localizar itens em tabelas grandes. Esses itens são geralmente letras que aparecem flutuando na extremidade direita da tabela e que permitem acesso direto a elementos que iniciem com a letra selecionada pelo usuário. Nesse caso, não use elementos no canto direito da tabela, para que eles não sejam sobrepostos pelos índices.

**Quadro 4.24:** *Tabela.*

**TEXTO (RÓTULO)**

**UTILIDADE:** Exibir sentenças textuais ao usuário. Rótulos são textos usados identificar um componente da interface ou para simplesmente mostrar uma mensagem.

**DIRETRIZES COMUNS:**

O texto deve ser fácil de entender, conciso e claro.

Rótulos que aparecem à esquerda de componentes devem possuir dois pontos (:) como último caractere.

Se você deseja usar diferentes fontes na interface, prefira fontes de uma mesma família, porque elas possuem similaridades entre si e tendem a criar interfaces mais atrativas.

Evite usar negrito para dar ênfase a textos.

Não use letras pequenas.

Evite usar itálico e sublinhado. Esses efeitos podem deixar os textos difíceis de serem lidos.

Certifique que o texto estará visível mesmo em ambientes com pouca luminosidade.

Dê contraste ao texto em relação ao fundo, para aumentar a legibilidade.

Pontue rótulos de listas, campos de texto e campos de busca com dois pontos (:).

Não use dois pontos quando o rótulo for um título.

Não use aspas simples, nem use pontos de exclamação em textos.

Não escreva sentenças em letra maiúscula, porque elas podem constranger ou inibir o usuário.

Coloque a primeira letra em caixa alta para referenciar sentenças e rótulos de caixas de seleção e de radio buttons.

Evite usar abreviações, a menos que o espaço seja um fator limitante e a abreviação seja facilmente interpretada.

Use uma terminologia com a qual o usuário está familiarizado.

Evite jargões, termos técnicos ou coloquialismos. Fale a linguagem do usuário.

Evite usar símbolos para substituírem textos. Por exemplo, não utilize “&” no lugar de “e”.

**Quadro 4.25:** *Texto (Rótulo).*

As diretrizes presentes nos Quadros 4.1 e 4.25 acima certamente auxiliam na tomada de decisões de *design* de interfaces, mas elas não apresentam recomendações de acordo com contexto do aplicativo que se deseja desenvolver. Usuários interessados em jogos, por exemplo, desejarão dispor de interfaces com elementos que facilitem o entretenimento, os quais podem ser obtidos pelo uso de conteúdos multimídia associados aos componentes de interface comuns. Além disso, o processo de *design* envolve a avaliação de protótipos para que as escolhas corretas sejam tomadas ao se liberar o produto final. De qualquer forma, as diretrizes ajudam a direcionar o processo de criação de cada interface, para posterior avaliação.

Um ponto importante a destacar com relação às diretrizes é que os componentes se relacionam de formas diversas em interfaces. Textos podem ser associados a praticamente todo componente, por exemplo. Logo, a utilização da documentação de um componente deve levar em consideração a documentação de todos os demais componentes que se associam a ele.

#### **4.4 Considerações finais**

O mapeamento entre uma determinada categoria de problema e a heurística de Nielsen equivalente foi realizado com base nas dificuldades encontradas pelo próprio mestrando. Tendo em vista que a associação de problemas a heurísticas é um processo que gera resultados que variam de acordo com cada avaliador, os resultados expostos na Tabela 4.5 poderiam ter sido diferentes caso outros avaliadores tivessem sido recrutados para associarem as categorias às heurísticas.

Bertini et al. [2006] propuseram um conjunto de heurísticas para avaliação de dispositivos móveis, considerando aspectos como privacidade e ergonomia desses dispositivos, os quais não foram considerados neste trabalho. Apesar dessa diferença, os dois trabalhos possuem propósitos semelhantes e seria válido realizar avaliações heurísticas do aplicativo de anotações multimídia com as heurísticas de Bertini et al. [2006], a fim de comparar os resultados dessas avaliações com os descritos neste Capítulo.



---

## Conclusão

---

A inspeção por simulação realizada com os quatro aplicativos em Android selecionados identificou 19 categorias de problemas, sendo que algumas delas não puderam ser facilmente relacionadas a alguma heurística de Nielsen. Como exemplos, podem-se citar as categorias , “aproveitamento do espaço da tela, de acordo com a orientação”, “adequação do componente à funcionalidade” e “visibilidade das interações possíveis”.

As avaliações heurísticas deste trabalho evidenciaram que as heurísticas para avaliação de usabilidade de interfaces de dispositivos móveis, levando-se em consideração apenas a camada de visualização, é capaz de identificar uma grande quantidade de problemas de usabilidade em interfaces de dispositivos móveis. Assim como os resultados expostos por Bertini et al. [2006], as heurísticas propostas foram capazes de encontrar uma quantidade maior de problemas em relação à abordagem de Nielsen [1994], com o diferencial de que os perfis dos avaliadores deste trabalho foram claramente expostos e escolhidos de forma a equilibrar os dois grupos de avaliadores.

Apesar de compartilhar o mesmo resultado em termos de quantidade de problemas, este trabalho obteve resultados sensivelmente diferentes dos de Bertini et al. [2006] em relação aos problemas cosméticos. As heurísticas de Bertini et al. [2006] não foram capazes de identificar muitos problemas cosméticos, ao passo que, em nosso estudo, o número de problemas cosméticos obtido foi mais que o triplo que o obtido pelas heurísticas de Nielsen (vide Gráfico 4.1). Além disso, nos experimentos de Bertini et al. [2006], as heurísticas propostas por eles encontraram um número menor de problemas catastróficos que as heurísticas tradicionais, ao passo que as heurísticas elaboradas neste trabalho encontraram mais que o dobro de problemas catastróficos em relação à abordagem tradicional. Contudo, ambos os estudos concluíram que problemas graves (de graus 3 a 4) são mais bem encontrados utilizando-se Nielsen, quando analisados em termos percentuais em relação à quantidade total de problemas encontrados por cada conjunto de heurísticas.

Pelos resultados obtidos neste estudo, sugere-se que as heurísticas propostas são mais adequadas que as de Nielsen para encontrarem problemas de menor severidade (graus 1 e 2). Os problemas catastróficos também foram mais efetivamente encontrados, ao passo que o número

de problemas de grau 3 encontrados por cada conjunto de heurísticas analisado foi praticamente o mesmo. Tendo em vista que o processo de desenvolvimento de *software* é composto por etapas que evoluem o sistema com o passar do tempo, pode ser uma abordagem interessante a grupos de desenvolvimento realizar avaliações heurísticas com as heurísticas propostas neste trabalho em etapas iniciais de desenvolvimento para encontrarem um número maior de problemas, bem como os problemas catastróficos, e executar a avaliação tradicional em etapas posteriores, caso desejem encontrar problemas graves que não tiverem sido encontrados anteriormente. Contudo, pelos resultados obtidos, sugere-se que o número de problemas novos encontrados pelas avaliações de Nielsen não seria expressivo.

Os resultados das avaliações utilizando-se as heurísticas propostas foram expressivos, mas não foram ideais, porque alguns problemas só foram detectados utilizando-se as heurísticas de Nielsen. Por conta disso, levantou-se possibilidade de se reestruturarem as heurísticas com base nesses problemas. Contudo, observou-se que todos os problemas encontrados por Nielsen poderiam ter sido associados em pelo menos uma heurística para avaliação de interfaces de dispositivos móveis.

Em relação ao número de heurísticas associadas a um único problema, notou-se uma variação maior no caso das novas heurísticas, ou seja, para um dado problema, normalmente mais heurísticas novas foram associadas a ele que heurísticas de Nielsen. Essa característica agrega ao novo conjunto de heurísticas uma vantagem e uma limitação em relação ao conjunto tradicional: especialistas tendem a levar menos tempo para associar um problema a uma heurística, mas provavelmente as heurísticas tradicionais são mais bem elaboradas no sentido de não darem margem à associação de muitas heurísticas a um único problema.

As heurísticas propostas neste trabalho podem ser usadas para avaliar a interface de dispositivos móveis de diversos tamanhos de tela e, em nossos experimentos, todas as dificuldades encontradas pelos usuários finais que usaram o aplicativo criado no Laboratório de Sistemas Web e Multimídia Interativos do ICMC poderiam ter sido identificadas pelas heurísticas propostas. Além disso, apenas uma dificuldade não teria sido suprimida caso uma nova versão do aplicativo que contemplasse as correções dos problemas encontrados por tais heurísticas tivesse sido criada.

As diretrizes propostas reúnem as recomendações de *design* dos componentes básicos de interface que se encontram nos aplicativos de dispositivos móveis diversos. O formato do documento facilita a leitura e a compreensão de cada componente, explanando as especificidades de cada item de acordo com os sistemas operacionais estudados. O mestrando acredita que não haja documentação semelhante disponível, especialmente no idioma Português.

Como um possível trabalho futuro, pode-se citar a inclusão de novos parâmetros humanos relativos à forma com a qual usuários interagem com dispositivos computacionais de forma geral, e com dispositivos móveis de forma específica, que estão em constante evolução e possuem a dificuldade da interdisciplinaridade, já que eles dependem de resultados da Psicologia, Sociologia, entre outras áreas. Da mesma forma, os componentes de interface podem mudar com o passar dos anos, tornando-se necessária a revisitação das diretrizes expostas neste trabalho.

Outra atividade futura seria a realização de avaliações heurísticas com o aplicativo de anotações multimídia descrito na Seção 3.2.1 com as heurísticas propostas por Bertini et al. [2006], a fim de comparar os resultados obtidos com essas avaliações com os obtidos neste trabalho.

As avaliações heurísticas deste trabalho foram realizadas separadamente em dias distintos com cada especialista, mas os resultados obtidos foram analisados numa mesma versão do aplicativo e em paralelo de acordo com cada grupo de 5 especialistas. Seria interessante analisar o impacto de se realizarem avaliações heurísticas usando as heurísticas propostas em versões

iniciais de interfaces e as heurísticas de Nielsen em versões posteriores e vice-versa.

Este trabalho levou em consideração aspectos intrínsecos à camada de visualização de software. Existem, porém, pesquisas que visam à criação de heurísticas que abrangem outras questões, como economia de energia e integração com tecnologias assistivas para suporte a acessibilidade, por exemplo.



# Referências bibliográficas

---

- Apple (2010). iOS UI Element Usage Guidelines. Technical report. [Accessed March 2013]. Available from: [http://developer.apple.com/library/ios/#documentation/UserExperience/Conceptual/MobileHIG/UIElementGuidelines/UIElementGuidelines.html#//apple\\_ref/doc/uid/TP40006556-CH13-SW1](http://developer.apple.com/library/ios/#documentation/UserExperience/Conceptual/MobileHIG/UIElementGuidelines/UIElementGuidelines.html#//apple_ref/doc/uid/TP40006556-CH13-SW1).
- Arhippainen, L. and Tähti, M. (2003). Empirical Evaluation of User Experience in Two Adaptive Mobile Application Prototypes. pages 27–34. ACM. [Accessed March 2013]. Available from: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.99.7147&rep=rep1&type=pdf>.
- Balagtas-Fernandez, F. and Hussmann, H. (2009). Evaluation of User-Interfaces for Mobile Application Development Environments. In *Proceedings of the 13th International Conference on Human-Computer Interaction. Part I: New Trends*, pages 204–213, Berlin, Heidelberg. Springer-Verlag. [Accessed March 2013]. Available from: [http://dx.doi.org/10.1007/978-3-642-02574-7\\_23](http://dx.doi.org/10.1007/978-3-642-02574-7_23).
- Bennett, J. L. (1979). The Commercial Impact of Usability in Interactive Systems. *Human-Computer Communication*, volume 2.
- Bertini, E., Gabrielli, S., and Kimani, S. (2006). Appropriating and Assessing Heuristics for Mobile Computing. In *Proceedings of the Working Conference on Advanced Visual Interfaces, AVI '06*, pages 119–126, New York, NY, USA. [Accessed March 2013]. Available from: <http://doi.acm.org/10.1145/1133265.1133291>.
- Bigonha, C., Cardoso, T. N. C., Moro, M. M., Almeida, V. A. F., and Gonçalves, M. A. (2010). Detecting Evangelists and Detractors on Twitter. In *Simpósio Brasileiro de Sistemas Multimídia e Web*.
- Blackberry (2012). Blackberry Smartphones: UI Guidelines. Technical report. [Accessed March 2013]. Available from: [http://docs.blackberry.com/en/developers/deliverables/17964/BlackBerry\\_Smartphones-UI\\_Guidelines-T893501-980426-0721013746-001-6.0-US.pdf](http://docs.blackberry.com/en/developers/deliverables/17964/BlackBerry_Smartphones-UI_Guidelines-T893501-980426-0721013746-001-6.0-US.pdf).
- Blackmon, M. H., Polson, P. G., Kitajima, M., and Lewis, C. (2002). Cognitive Walkthrough for the Web. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Changing our World, Changing Ourselves*, CHI '02, pages 463–470, New York, NY, USA. ACM. [Accessed March 2013]. Available from: <http://dx.doi.org/10.1145/503376.503459>.
- Bonifácio, B., Viana, D., Vieira, S., Araújo, C., and Conte, T. (2010). Aplicando técnicas de inspeção de usabilidade para avaliar aplicações móveis. In *Proceedings of the IX Symposium on*

- Human Factors in Computing Systems*, IHC '10, pages 189–192, Porto Alegre, Brazil. Brazilian Computer Society. [Accessed March 2013]. Available from: <http://dl.acm.org/citation.cfm?id=1999593.1999615>.
- Brace, I. (2004). *Questionnaire Design: How To Plan, Structure And Write Survey Material For Effective Market Research*. Kogan Page Business Books, Creative Print and Design, United Kingdom.
- Bradburn, N., Sudman, S., and Wansink, B. (2004). *Asking Questions: The Definitive Guide to Questionnaire Design – For Market Research, Political Polls, and Social and Health Questionnaires*. Research Methods for the Social Sciences. Wiley. [Accessed March 2013]. Available from: <http://books.google.com.br/books?id=YXKbTx2j9i4C>.
- Cooper, A. (1995). *About Face: The Essentials of User Interface Design*. John Wiley & Sons. Available from: <http://books.google.com.br/books?id=htxQAAAAMAAJ>.
- Coyne, S. and Coyne, K. (2011). *Seven Steps to Better Brainstorming*. Accessed March 2013. Available from: [http://www.mckinseyquarterly.com/Seven\\_steps\\_to\\_better\\_brainstorming\\_2767](http://www.mckinseyquarterly.com/Seven_steps_to_better_brainstorming_2767).
- Dickerson, S. (2004). Acute Stressors and Cortisol Responses: A Theoretical Integration and Synthesis of Laboratory Research. *Psychological Bulletin*, 130(3):355–391.
- Dienstbier, R. A. (1989). Arousal and Physiological Toughness: Implications for Mental and Physical Health. *Psychiatry and Psychology Commons*, 96(1):84–100. [Accessed March 2013]. Available from: <http://www.biomedsearch.com/nih/Arousal-physiological-toughness-implications-mental/2538855.html>.
- Dimakopoulos, D. N. and Magoulas, G. D. (2009). Interface Design and Evaluation of a Personal Information Space for Mobile Learners. *International Journal of Mobile Learning and Organisation*, 3(4):440–463. [Accessed March 2013]. Available from: <http://dx.doi.org/10.1504/IJML0.2009.027458>.
- Dix, A., Finlay, J. E., Abowd, G. D., and Beale, R. (2004). *Human-Computer Interaction*. Prentice-Hall, Upper Saddle River, NJ, USA, 3rd edition.
- Galitz, W. O. (2003). *The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques*. John Wiley & Sons, New York, NY, USA, 2nd edition.
- Gatica-Perez, D. and Montoliu, R. (2010). Discovering Human Places of Interest from Multimodal Mobile Phone Data. In *Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia*, MUM '10, pages 12:1–12:10, New York, NY, USA. ACM. Available from: <http://doi.acm.org/10.1145/1899475.1899487>.
- Gazzalley, A. (2012). How Mobile Tech Can Influence Our Brain. [Part of Complete Coverage on Our Mobile Society. Special to CNN. Online: Accessed on 03-2013].
- Gonçalves, V. P., Neris, V. P. A., Morandini, M., Nakagawa, E. Y., and Ueyama, J. (2011). Uma Revisão Sistemática sobre Métodos de Avaliação de Usabilidade Aplicados em Software de Telefones Celulares. In *Proceedings of the 10th Brazilian Symposium on Human Factors in Computing Systems and the 5th Latin American Conference on Human-Computer Interaction*, IHC e CLIHC '11, pages 197–201, Porto Alegre, Brazil. Brazilian Computer Society. [Accessed March 2013]. Available from: <http://dl.acm.org/citation.cfm?id=2254436.2254470>.

- Gong, J. and Tarasewich, P. (2011). Guidelines for Handheld Mobile Device Interface Design. In *Proceedings of the 2004 DSI Annual Meeting*. Accessed March 2013. Available from: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.87.5230>.
- Goodman, R. and Lawless, M. (1994). *Technology and Strategy: Conceptual Models and Diagnostics*. Oxford University Press, United Kingdom. Available from: <http://books.google.com.br/books?id=ii3SmAEACAAJ>.
- Google (2012a). Design: Android Developers. Technical report. [Accessed March 2013]. Available from: <http://developer.android.com/design/index.html>.
- Google (2012b). Google Play. [Accessed March 2013]. Available from: <https://play.google.com/store>.
- Grasso, A. and Roselli, T. (2005). Guidelines for Designing and Developing Contents for Mobile Learning. In *Proceedings of the IEEE International Workshop on Wireless and Mobile Technologies in Education, WMTE '05*, pages 123–127, Washington, DC, USA. IEEE Computer Society. [Accessed March 2013]. Available from: <http://dx.doi.org/10.1109/WMTE.2005.27>.
- Hagen, P., Robertson, T., Kan, M., and Sadler, K. (2005). Emerging Research Methods for Understanding Mobile Technology Use. In *Proceedings of the 17th Australia conference on Computer-Human Interaction: Citizens Online: Considerations for Today and the Future, OZ-CHI '05*, pages 1–10, Narrabundah, Australia. [Accessed March 2013]. Available from: <http://dl.acm.org/citation.cfm?id=1108368.1108417>.
- Harrison, C., Amento, B., Kuznetsov, S., and Bell, R. (2007). Rethinking the progress bar. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology, UIST '07*, pages 115–118, New York, NY, USA. ACM. [Accessed March 2013]. Available from: <http://doi.acm.org/10.1145/1294211.1294231>.
- Hayhoe, G. (2001). From Desktop to Palmtop: Creating Usable Online Documents for Wireless and Handheld Devices. In *Professional Communication Conference, 2001. IPCC 2001. IEEE International*, pages 1–11.
- Helander, M. (1997). *Handbook of Human-Computer Interaction*. North-Holland, 2nd edition. Available from: <http://books.google.com.br/books?id=2-FQAAAAMAAJ>.
- Henry, J. P. and Grim, C. (1990). Psychosocial Mechanisms of Primary Hypertension. *J Hypertens*, 8(9):783–793. [Accessed March 2013]. Available from: <http://www.biomedsearch.com/nih/Psychosocial-mechanisms-primary-hypertension/2172367.html>.
- Iqbal, S. T. and Horvitz, E. (2007). Disruption and recovery of computing tasks: Field study, analysis, and directions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '07*, pages 677–686, New York, NY, USA. ACM. [Accessed March 2013]. Available from: <http://doi.acm.org/10.1145/1240624.1240730>.
- ISO (1998). ISO 9241-11: Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs) - Part 11 : Guidance on Usability. Technical report, International Organization for Standardization, Geneva.
- ISO/IEC (2001). ISO/IEC 9126. Software Engineering – Product Quality. Technical report, International Organization for Standardization, Geneva.

- Ji, Y. G., Park, J. H., Lee, C., and Yun, M. H. (2006). A Usability Checklist for the Usability Evaluation of Mobile Phone User Interface. *International Journal of Human-Computer Interaction*, 20(3):207–231. [Accessed March 2013]. Available from: [http://www.leaonline.com/doi/abs/10.1207/s15327590ijhc2003\\_3?journalCode=ijhc](http://www.leaonline.com/doi/abs/10.1207/s15327590ijhc2003_3?journalCode=ijhc).
- Jones, M. and Marsden, G. (2006). Mobile Interaction Design. *Proceedings of the 7th international conference on Human computer interaction with mobile devices services MobileHCI 05*, 1:369. [Accessed March 2013]. Available from: <http://portal.acm.org/citation.cfm?doid=1085777.1085872>.
- Kantore, A. (2011). *User-Interface Evaluation Metrics for a Typical M-Learning Application*. The School of Information Communication and Technology. Nelson Mandela Metropolitan University. [Accessed March 2013]. Available from: <http://books.google.com.br/books?id=YXKbTx2j9i4C>.
- Kjeldskov, J. and Stage, J. (2004). New Techniques for Usability Evaluation of Mobile Systems. *International Journal of Human-Computer Studies*, 60:599–620.
- Koyani, S. J., Bailey, R. W., and Nall, J. R. (2003). *Research-Based Web Design & Usability Guidelines*. Computer Psychology. Available from: <http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0974996904>.
- Krumm, J. (2010). Processing Sequential Sensor Data. In *Ubiquitous Computing Fundamentals*, pages 353–380. Chapman & Hall/CRC, Boca Raton, FL, 1st edition.
- Kunjachan, C. (2011). Evaluation of Usability on Mobile User Interface. University of Washington, Bothell, USA.
- Lai, V. S. and Mahapatra, R. K. (1997). Exploring the Research in Information Technology Implementation. *Information & Management*, 32(4):187–201. [Accessed February 2013]. Available from: [http://dx.doi.org/10.1016/S0378-7206\(97\)00022-0](http://dx.doi.org/10.1016/S0378-7206(97)00022-0).
- Luchini, K., Quintana, C., Krajcik, J., Farah, C., Nandihalli, N., Reese, K., Wieczorek, A., and Soloway, E. (2002). Scaffolding in the Small: Designing Educational Supports for Concept Mapping on Handheld Computers. In *CHI '02 Extended Abstracts on Human Factors in Computing Systems*, CHI '02, pages 792–793, New York, NY, USA. ACM. [Accessed January 2013]. Available from: <http://dx.doi.org/10.1145/506443.506600>.
- Lupien, S. J., Maheu, F., Tu, M., Fiocco, A., and Schramek, T. E. (2007). The effects of stress and stress hormones on human cognition: Implications for the field of brain and cognition. *Brain and Cognition*, 65(3):209–237. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/17466428>.
- Lynch, P. and Horton, S. (2001). *Web Style Guide: Basic Design Principles for Creating Web Sites*. Yale University Press. [Accessed March 2013]. Available from: <http://books.google.com.br/books?id=gGbCQgAACAAJ>.
- MacKenzie, I. S. and Zhang, S. X. (1999). The Design and Evaluation of a High-Performance Soft Keyboard. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, CHI '99, pages 25–31, New York, NY, USA. ACM. [Accessed March 2013]. Available from: <http://doi.acm.org/10.1145/302979.302983>.

- Matera, M., Rizzo, F., Carughi, G. T., and Milano, P. (2006). Web Usability: Principles and Evaluation Methods. In Mendes, E. and Mosley, N., editors, *Web engineering*, pages 144–166. Springer. [Accessed March 2013]. Available from: <http://www.springerlink.com/index/hp5t836gxxv33m47.pdf>.
- Mattar, F. (1999). *Pesquisa de Marketing: Metodologia e Planejamento*. Pesquisa de marketing. Atlas. Available from: <http://books.google.com.br/books?id=Jcr1PgAACAAJ>.
- Maule, A. and Hockey, G. (1993). State, Stress, and Time Pressure. In *Time Pressure and Stress in Human Judgment and Decision Making*, pages 83–101. Springer US. [Accessed March 2013]. Available from: [http://dx.doi.org/10.1007/978-1-4757-6846-6\\_6](http://dx.doi.org/10.1007/978-1-4757-6846-6_6).
- McGrenere, J., Baecker, R. M., and Booth, K. S. (2002). An Evaluation of a Multiple Interface Design Solution for Bloated Software. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '02, pages 164–170, New York, NY, USA. ACM. [Accessed March 2013]. Available from: <http://doi.acm.org/10.1145/503376.503406>.
- Microsoft (2013). User Experience Design Guidelines for Windows Phone. Technical report. [Accessed March 2013]. Available from: [http://msdn.microsoft.com/en-us/library/windowsphone/design/hh202915\(v=vs.92\).aspx](http://msdn.microsoft.com/en-us/library/windowsphone/design/hh202915(v=vs.92).aspx).
- Moraveji, N. and Soesanto, C. (2012). Towards Stress-Less User Interfaces: 10 Design Heuristics Based on the Psychophysiology of Stress. In *Extended Abstracts on Human Factors in Computing Systems*, CHI EA '12, pages 1643–1648, New York, NY, USA. ACM. [Accessed March 2013]. Available from: <http://doi.acm.org/10.1145/2212776.2223686>.
- Mullet, K. and Sano, D. (1995). *Designing Visual Interfaces: Communication Oriented Techniques*. Communication oriented techniques. SunSoft Press. Available from: <http://books.google.com.br/books?id=LCYPAQAAMAAJ>.
- Nass, C., Steuer, J., and Tauber, E. R. (1994). Computers are Social Actors. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '94, pages 72–78, New York, NY, USA. ACM. [Accessed March 2013]. Available from: <http://doi.acm.org/10.1145/191666.191703>.
- Nielsen, J. (1993). *Usability Engineering*. Morgan Kaufmann Publishers, San Francisco, CA, USA.
- Nielsen, J. (1994). Heuristic evaluation. In Nielsen, J. and Mack, R. L., editors, *Usability Inspection Methods*, pages 25–62. John Wiley & Sons, New York, NY, USA. [Accessed March 2013]. Available from: <http://dl.acm.org/citation.cfm?id=189200.189209>.
- Nielsen, J. and Molich, R. (1990). Heuristic evaluation of User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '90, pages 249–256, New York, NY, USA. ACM. [Accessed March 2013]. Available from: <http://doi.acm.org/10.1145/97243.97281>.
- Norman, D. (1988). *The Design of Everyday Things*. Number 842 in The Design of Everyday Things. Bantam Doubleday Dell Publishing Group. Available from: <http://books.google.com.br/books?id=b09jQgAACAAJ>.
- Oinas-Kukkonen, H. and Kurkela, V. (2003). Developing Successful Mobile Applications. In *Computer Science and Technology*. Available from: <http://en.scientificcommons.org/42316819>.

- Preece, J. (1994). *Human-Computer Interaction. Concepts And Design*. Ics Series. Addison-Wesley Pub. Co. Available from: <http://books.google.com.br/books?id=p99QAAAAMAAJ>.
- Robertson, G., Czerwinski, M., Baudisch, P., Meyers, B., Robbins, D., Smith, G., and Tan, D. (2005). The large-display user experience. *IEEE Computer Graphics and Applications*, 25(4):44–51. [Accessed March 2013]. Available from: <http://dx.doi.org/10.1109/MCG.2005.88>.
- Rocha, H. V. and Baranauskas, M. C. C. (2003). *Design e Avaliação de Interfaces Humano-Computador*. EdUnicamp. Available from: [http://pan.nied.unicamp.br/download\\_livro/livrodownload.html](http://pan.nied.unicamp.br/download_livro/livrodownload.html).
- Rothwell, A. (1993). *Questionnaire Design*. De Montfort University. Kogan Press. Available from: <http://books.google.com.br/books?id=rnZOMwEACAAJ>.
- Santos, P. J. N. (2008). Usabilidade em sistemas de realidade virtual: Estudos com utilizadores. Master's thesis, Universidade de Aveiro, Portugal. [Accessed March 2013]. Available from: <http://hdl.handle.net/10773/2059>.
- Seffah, A., Kececi, N., and Donyaee, M. (2001). QUIM: A Framework for Quantifying Usability Metrics in Software Quality Models. In *Asia-Pacific Conference on Quality Software*, pages 311–318.
- Shackel, B. (1981). The Concept of Usability. In *Proceedings of IBM Software and Information Usability Symposium*, pages 1–30, New York, NY, USA. IBM Corporation.
- Shackel, B. (1991). Usability–context, Framework, Definition, Design and Evaluation. In *Human Factors for Informatics Usability*, pages 21–37. Cambridge University Press, New York, NY, USA. [Accessed March 2013]. Available from: <http://dl.acm.org/citation.cfm?id=117829.117833>.
- Sharp, H., Rogers, Y., and Preece, J. (2007). *Interaction Design: Beyond Human-Computer Interaction*. Wiley, 2nd edition. [Accessed March 2013]. Available from: <http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0470018666>.
- Shneiderman, B. and Plaisant, C. (2009). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley Longman, Boston, MA, USA, 5th edition.
- Siniscalco, M. T. and Auriat, N. (2005). *Questionnaire Design*. Sabine Lebeau, International Institute for Educational Planning/UNESCO.
- Smith, S. L. and Mosier, J. N. (1986). Guidelines for Designing User Interface Software. Technical report, The MITRE Corporation Bedford, Massachusetts, USA. [Prepared for Deputy Commander for Development Plans and Support Systems, Electronic Systems Division, AFSC, United States Air Force, Hanscom Air Force Base, Massachusetts].
- Sutcliffe, A. (1995). *Human-Computer Interface Design*. Macmillan Computer Science Series. Macmillan Press. Available from: <http://books.google.com.br/books?id=gOVQAAAAMAAJ>.
- Torres, N. (1995). *Competitividade Empresarial com a Tecnologia de Informação*. Makron Books, São Paulo. Available from: <http://books.google.com.br/books?id=QBCiAQAACAAJ>.
- Tullis, T. and Albert, W. (2010). *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics*. Morgan Kaufmann series in interactive technologies. Elsevier Science. Available from: <http://books.google.com.br/books?id=KsjpuMJ6T-YC>.

- Ulrich, R. S., Simons, R. F., Losito, B. D., Fiorito, E., Miles, M. A., and Zelson, M. (1991). Stress Recovery during Exposure to Natural and Urban Environments. *Journal of Environmental Psychology*, 11(3). Available from: <http://www.sciencedirect.com/science/article/B6WJ8-4GK8NPT-1/2/aca5972eb250c78fd991457187117cc2>.
- Vääätäjä, H. and Roto, V. (2010). Mobile Questionnaires for User Experience Evaluation. In *Extended Abstracts on Human Factors in Computing Systems*, CHI EA '10, pages 3361–3366, New York, NY, USA. ACM. [Accessed March 2013]. Available from: <http://doi.acm.org/10.1145/1753846.1753985>.
- Weiser, M. (1993). Some Computer Science Issues in Ubiquitous Computing. *Communications of the ACM*, 36(7):75–84. [Accessed March 2013]. Available from: <http://doi.acm.org/10.1145/159544.159617>.
- Wickens, C. D. and Hollands, J. G. (2000). *Engineering Psychology and Human Performance*, volume 27. Prentice Hall. [Accessed March 2013]. Available from: [http://webfiles.ita.chalmers.se/~mys/HumanAspects/WickensHollands/0\\_Wickens\\_Index\\_Preface.pdf](http://webfiles.ita.chalmers.se/~mys/HumanAspects/WickensHollands/0_Wickens_Index_Preface.pdf).
- Williams, R. (2005). *The Non-designers Design Book: Design and Typographic Principles for the Visual Novice*. Non Designer's Design Book. Peachpit Press. Available from: <http://books.google.com.br/books?id=8IlqSMtWck0C>.



---

## Arquivo de registros do usuário novato

---

1 START: 13:30:07 Brasilia Time  
2 CHOOSE\_USERNAME 1: 13:30:19 Brasilia Time  
3 CHOOSE VIDEO 1: 13:30:30 Brasilia Time  
4 CHOOSE AUTHOR 1: 13:30:40 Brasilia Time  
5 EditText\_TOUCH 13:30:53 Brasilia Time  
5 ADD: 13:31:04 Brasilia Time  
5 TEXT ANNOTATION: 13:31:04 Brasilia Time  
5 INK\_Ann\_Enabled: 13:31:26 Brasilia Time  
    5 CONTEXT\_INFO 13:31:31 Brasilia Time  
  
IA: 13:31:42 Brasilia Time  
5.1 Cancel: 13:31:53 Brasilia Time  
  
5 INK ANNOTATION: 13:31:57 Brasilia Time  
5 INK ANNOTATION: 13:31:58 Brasilia Time 5 INK ANNOTATION: 13:31:59 Brasilia Time  
5 INK ANNOTATION: 13:32:00 Brasilia Time  
5 INK ANNOTATION: 13:32:01 Brasilia Time  
5 INK ANNOTATION: 13:32:01 Brasilia Time  
5 Audio\_Ann\_Enabled: 13:32:14 Brasilia Time  
5 CONTEXT\_INFO 13:32:28 Brasilia Time  
  
AA13:32:32 Brasilia Time  
5.1 Navigate to108 at: 13:32:38 Brasilia Time  
  
5 EditText\_TOUCH 13:32:59 Brasilia Time  
5 ADD: 13:33:24 Brasilia Time  
5 TEXT ANNOTATION: 13:33:24 Brasilia Time  
5 INK ANNOTATION: 13:34:04 Brasilia Time

5 INK ANNOTATION: 13:34:05 Brasilia Time

AA13:34:28 Brasilia Time

5.1 Cancel: 13:34:31 Brasilia Time

5 Audio\_Ann\_Disabled: 13:34:35 Brasilia Time

5 AUDIO ANNOTATION: 13:34:35 Brasilia Time

AA13:34:52 Brasilia Time

5.1 Navigate to79 at: 13:35:01 Brasilia Time

5 INK ANNOTATION: 13:35:01 Brasilia Time

AA13:35:02 Brasilia Time

5.1 Remove: 13:35:10 Brasilia Time

5.2 OK: 13:35:13 Brasilia Time

AA13:35:13 Brasilia Time

5.1 Cancel: 13:35:28 Brasilia Time

AA13:35:41 Brasilia Time

5.1 Navigate to79 at: 13:35:43 Brasilia Time

5 TA 13:36:03 Brasilia Time

5.1 Edit:13:36:06 Brasilia Time

5.2 Cancel: 13:36:09 Brasilia Time

5.1 Navigate to 10 at: 13:36:10 Brasilia Time

5 TA 13:36:20 Brasilia Time

5.1 Navigate to 10 at: 13:36:22 Brasilia Time

5 SHARE: 13:36:28 Brasilia Time

5 TA 13:36:31 Brasilia Time

5.1 Edit:13:36:33 Brasilia Time

5.2 Remove: 13:36:37 Brasilia Time

5.3 OK: 13:36:38 Brasilia Time

5 TA 13:36:39 Brasilia Time

5.1 Cancel: 13:36:41 Brasilia Time

IA: 13:36:47 Brasilia Time

5.1 Clear:13:36:49 Brasilia Time

5.2 OK:13:36:50 Brasilia Time

5 TA 13:36:53 Brasilia Time

5.1 Edit: 13:36:56 Brasilia Time

5.2 Edit: 13:36:59 Brasilia Time

5.3 OK: 13:37:01 Brasilia Time

IA: 13:36:50 Brasilia Time

5.1 Cancel: 13:37:06 Brasilia Time

4 CHOOSE AUTHOR 0: 13:37:08 Brasilia Time



---

## Arquivo de registros do usuário intermediário

---

1 START: 20:34:07 Brasilia Time  
2 CHOOSE\_USERNAME 1: 20:34:10 Brasilia Time  
3 CHOOSE VIDEO 1: 20:34:13 Brasilia Time  
4 CHOOSE AUTHOR 1: 20:34:16 Brasilia Time  
5 INK\_Ann\_Enabled: 20:34:53 Brasilia Time  
5 EditText\_TOUCH 20:34:56 Brasilia Time  
5 EditText\_ENTER ADD: 20:35:03 Brasilia Time  
5 ADD: 20:35:03 Brasilia Time  
5 TEXT ANNOTATION: 20:35:03 Brasilia Time

IA: 20:35:18 Brasilia Time  
5.1 Cancel: 20:35:31 Brasilia Time

IA: 20:35:36 Brasilia Time  
5.1 Cancel: 20:35:41 Brasilia Time

5 ADD: 20:35:45 Brasilia Time  
5 TEXT ANNOTATION: 20:35:45 Brasilia Time  
5 CONTEXT\_INFO 20:35:54 Brasilia Time

IA: 20:36:04 Brasilia Time  
5.1 Cancel: 20:36:08 Brasilia Time

5 ADD: 20:36:13 Brasilia Time  
5 CONTEXT\_INFO 20:36:21 Brasilia Time  
5 SHARE: 20:36:43 Brasilia Time

IA: 20:36:47 Brasilia Time

5 Audio\_Ann\_Enabled: 20:37:15 Brasilia Time  
5 ADD: 20:37:20 Brasilia Time

AA20:37:26 Brasilia Time  
5.1 Cancel: 20:37:34 Brasilia Time

5 INK\_Ann\_Disabled: 20:37:41 Brasilia Time  
5 EditText\_TOUCH 20:37:42 Brasilia Time

AA20:38:27 Brasilia Time  
5.1 Cancel: 20:38:35 Brasilia Time

5 INK\_Ann\_Enabled: 20:39:32 Brasilia Time  
5 SeekBar: 136202277357820:39:33 Brasilia Time  
5 Audio\_Ann\_Disabled: 20:39:33 Brasilia Time  
5 AUDIO ANNOTATION: 20:39:33 Brasilia Time  
5 SeekBar: 136202277379220:39:33 Brasilia Time  
5 Pushed PLAY: 13620227750220:39:37 Brasilia Time

5 TA 20:40:00 Brasilia Time  
5.1 Navigate to 39 at: 20:40:02 Brasilia Time

5 TA 20:40:13 Brasilia Time  
5.1 Edit:20:40:16 Brasilia Time  
5.2 Remove: 20:40:18 Brasilia Time  
5.3 OK: 20:40:20 Brasilia Time

5 TA 20:40:20 Brasilia Time  
5.1 Cancel: 20:40:23 Brasilia Time

IA: 20:40:27 Brasilia Time  
5.1 Cancel: 20:40:29 Brasilia Time  
5 ADD: 20:40:32 Brasilia Time  
5 ADD: 20:40:34 Brasilia Time

IA: 20:40:39 Brasilia Time  
5.1 Cancel: 20:40:42 Brasilia Time

5 Pushed PLAY: 136202286446220:41:04 Brasilia Time  
5 SHARE: 20:41:07 Brasilia Time  
5 EditText\_TOUCH 20:41:20 Brasilia Time  
5 Audio\_Ann\_Enabled: 20:41:47 Brasilia Time

AA20:41:49 Brasilia Time  
5.1 Cancel: 20:41:53 Brasilia Time

5 EditText\_TOUCH 20:41:57 Brasilia Time

5 Pushed PLAY: 136202293641320:42:16 Brasilia Time  
5 SeekBar: 136202293802620:42:18 Brasilia Time  
5 Audio\_Ann\_Disabled: 20:42:18 Brasilia Time

IA: 20:42:26 Brasilia Time

5.1 Cancel: 20:42:28 Brasilia Time

5 INK\_Ann\_Enabled: 20:42:39 Brasilia Time  
5 INK\_Ann\_Disabled: 20:42:45 Brasilia Time  
5 INK ANNOTATION: 20:42:45 Brasilia Time

IA: 20:42:47 Brasilia Time

5.1 Cancel: 20:42:50 Brasilia Time

5 INK\_Ann\_Enabled: 20:42:53 Brasilia Time  
5 INK\_Ann\_Disabled: 20:43:15 Brasilia Time  
5 INK ANNOTATION: 20:43:15 Brasilia Time  
AA20:43:22 Brasilia Time  
5.1 Navigate to1804 at:20:43:25 Brasilia Time

5 TA 20:43:34 Brasilia Time

5.1 Navigate to 48 at: 20:43:36 Brasilia Time

5 TA 20:43:42 Brasilia Time

5.1 Edit: 20:43:47 Brasilia Time

5.2 Edit: 20:43:49 Brasilia Time

5.3 OK: 20:43:50 Brasilia Time

IA: 20:44:03 Brasilia Time

5.1 Clear: 20:44:05 Brasilia Time

5.2 Cancel: 20:44:07 Brasilia Time

5.1 Cancel: 20:44:08 Brasilia Time

AA20:44:16 Brasilia Time

5.1 Remove: 20:44:18 Brasilia Time

5.2 Cancel: 20:44:19 Brasilia Time

5.1 Cancel: 20:44:19 Brasilia Time

IA: 20:44:21 Brasilia Time

5.1 Cancel: 20:44:21 Brasilia Time

4 CHOOSE AUTHOR 0: 20:44:22 Brasilia Time  
3 CHOOSE VIDEO 1: 20:44:22 Brasilia Time  
2 CHOOSE\_USERNAME 1: 20:44:22 Brasilia Time  
1 START: 20:44:23 Brasilia Time



---

## Ilustrações dos componentes de interface para dispositivos móveis

---

**N**A Seção 4.3 foram descritas recomendações de uso dos componentes de interface para o *design* de interfaces para dispositivos móveis com foco em usabilidade. Um paralelo das diferentes terminologias usadas por cada empresa que disponibiliza essas recomendações foi realizado, mas não se mostrou o aspecto visual de cada componente, de acordo com cada empresa. Esse apêndice contém ilustrações de cada componente, para que o interessado no *design* dessas interfaces consiga elaborar *layouts* que possam ser implementados na tecnologia escolhida.

Cada ilustração de componente é acompanhada por uma letra, que indica o sistema operacional referente a ele.

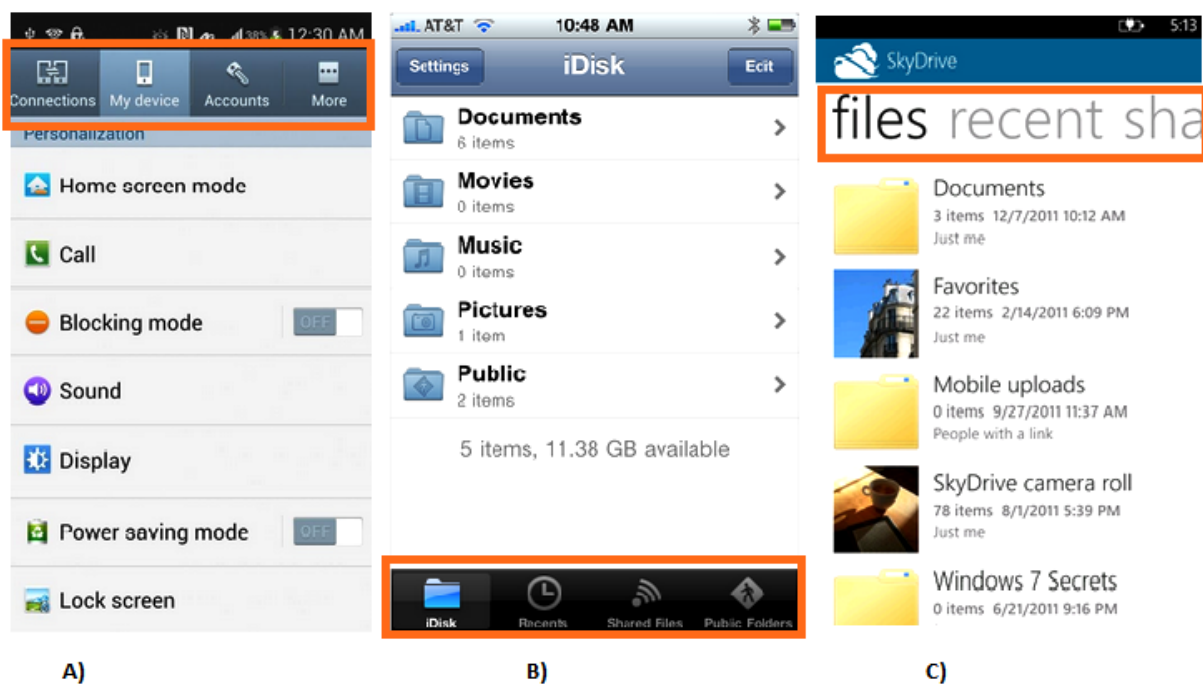


Figura C.1: Barra de Abas. A) Android; B) iOS; C) Windows Phone.

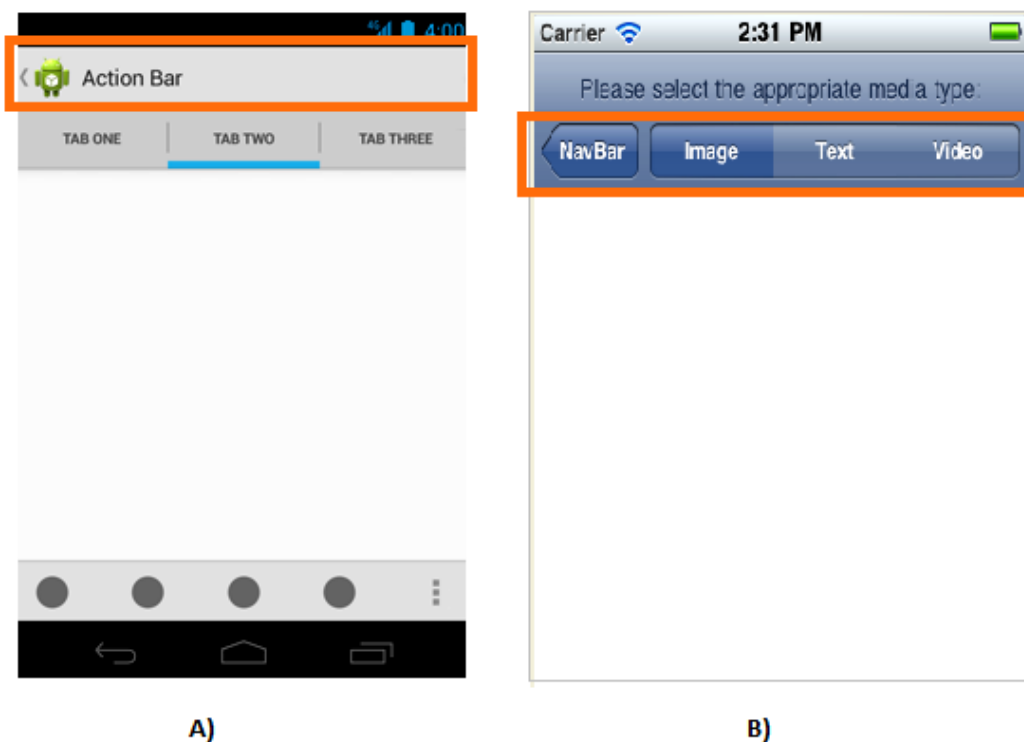


Figura C.2: Barra de Atividades. A) Android; B) iOS.

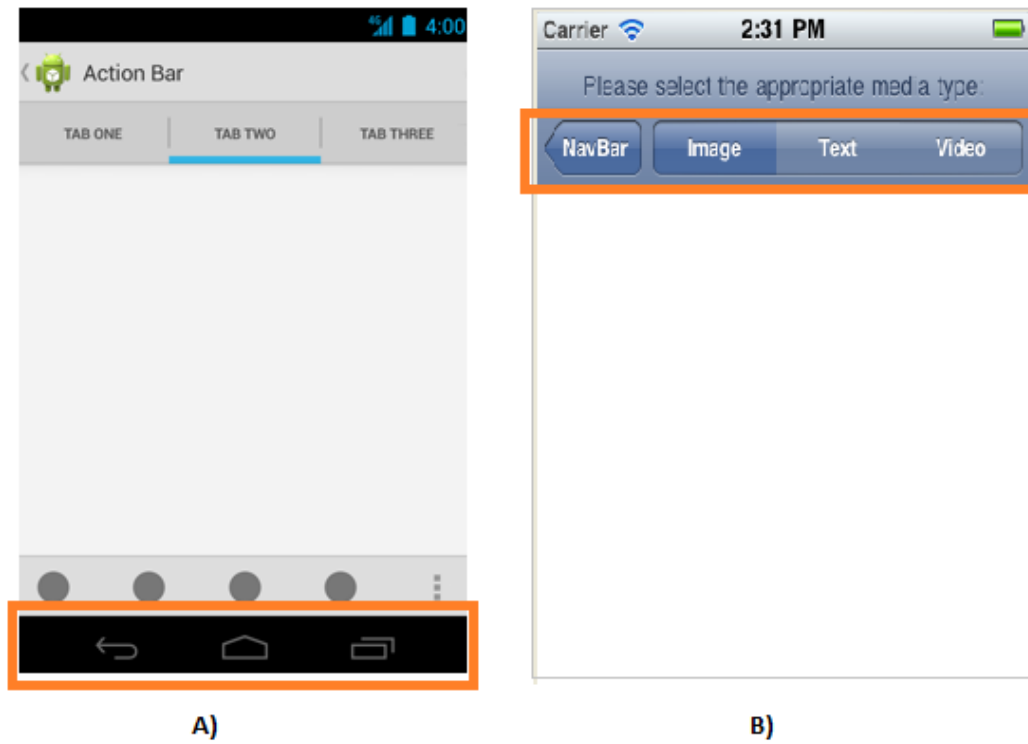


Figura C.3: Barra de navegação. A) Android; B) iOS.



Figura C.4: Barra de progresso. A) Android; B) Blackberry; C) iOS; D) Windows Phone.



Figura C.5: Barra de status. A) Android; B) Blackberry; C) iOS; D) Windows Phone.



Figura C.6: Barra deslizante de ajuste. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

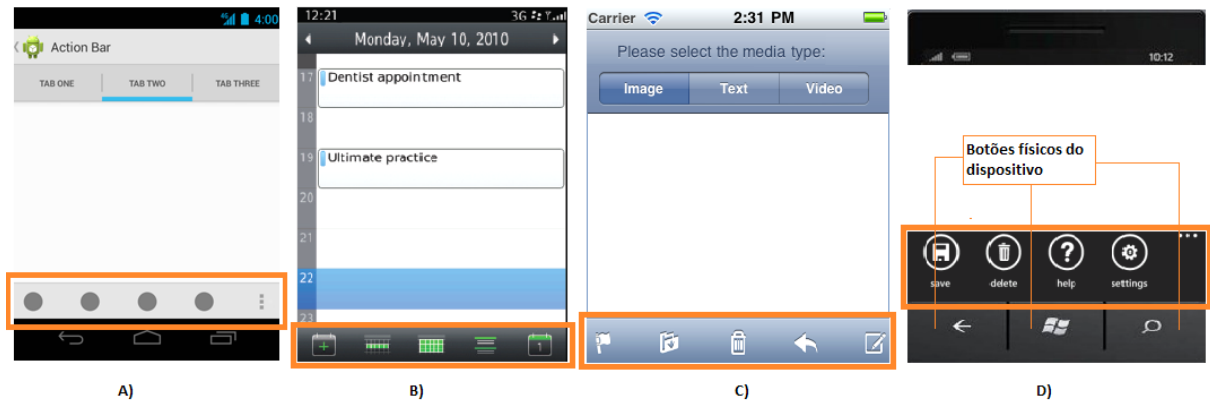


Figura C.7: Barra inferior. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

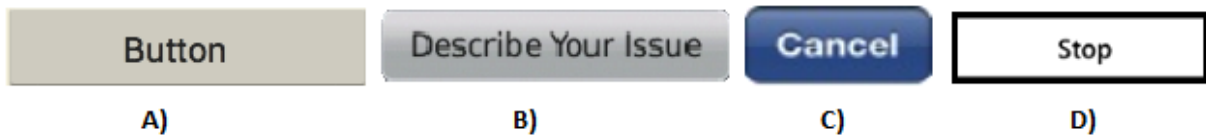


Figura C.8: Botão. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

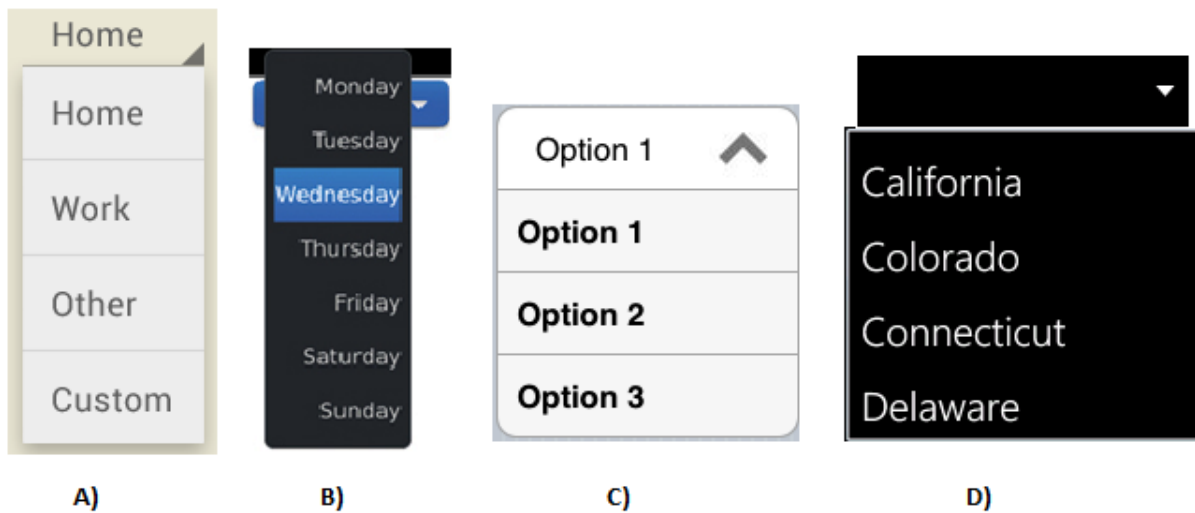


Figura C.9: Caixa de múltipla escolha. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

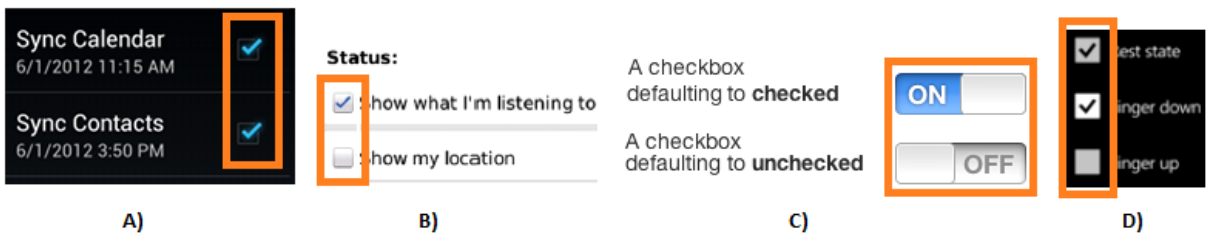


Figura C.10: Caixa de seleção. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

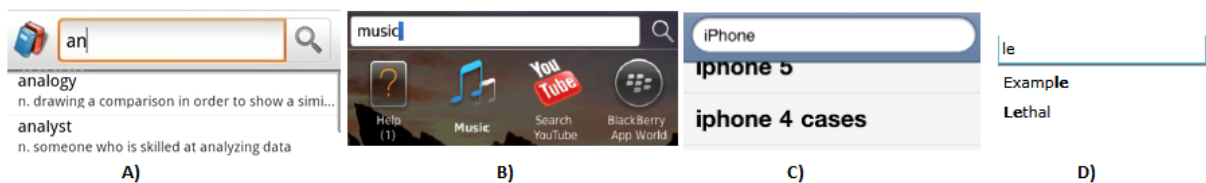


Figura C.11: Campo de busca. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

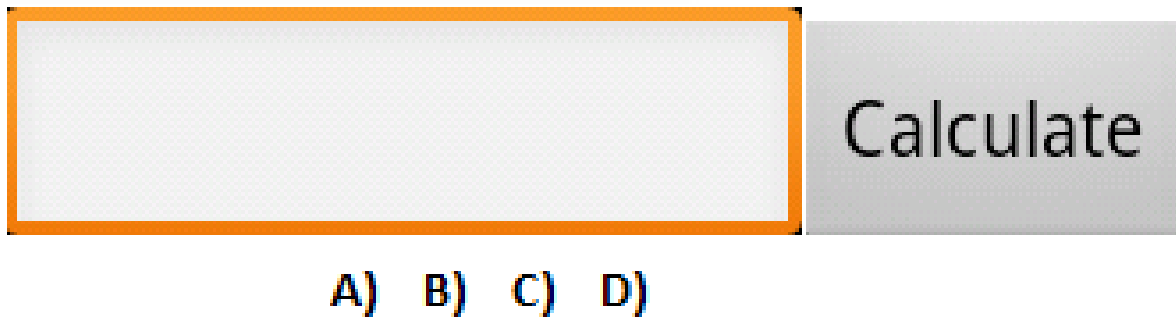


Figura C.12: Campo de texto. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

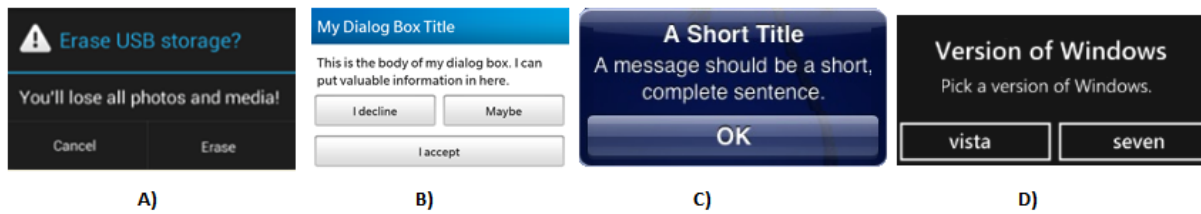


Figura C.13: Diálogo. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

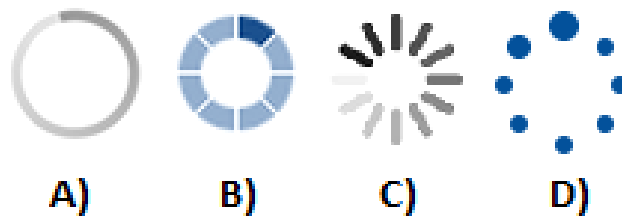


Figura C.14: Girador de progresso. A) Android; B) Blackberry; C) iOS; D) Windows Phone.



Figura C.15: Link. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

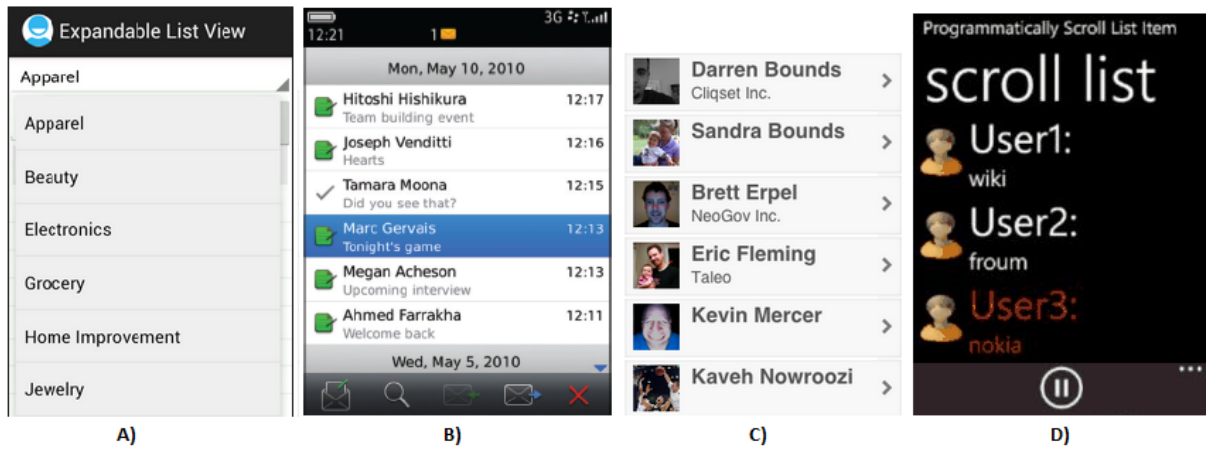


Figura C.16: Lista. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

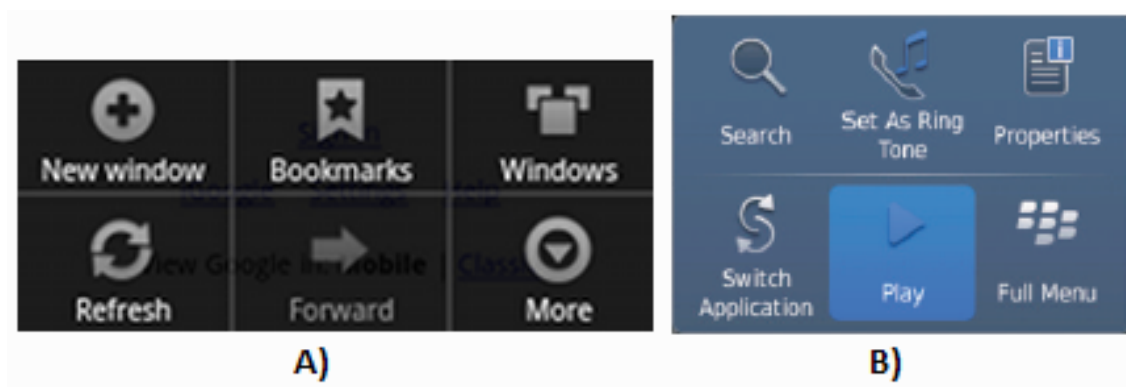


Figura C.17: Menu popup. A) Android; B) Blackberry.

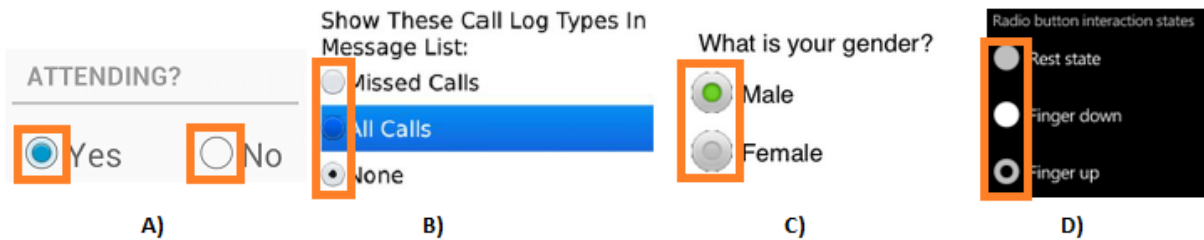


Figura C.18: Radio button. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

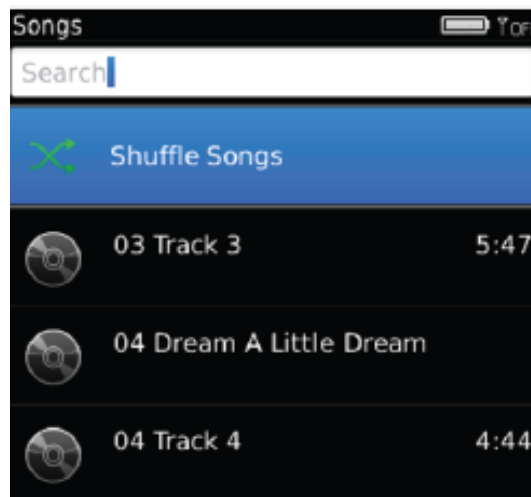


Figura C.19: Seletor de arquivos. A) Blackberry.



Figura C.20: Seletor de data. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

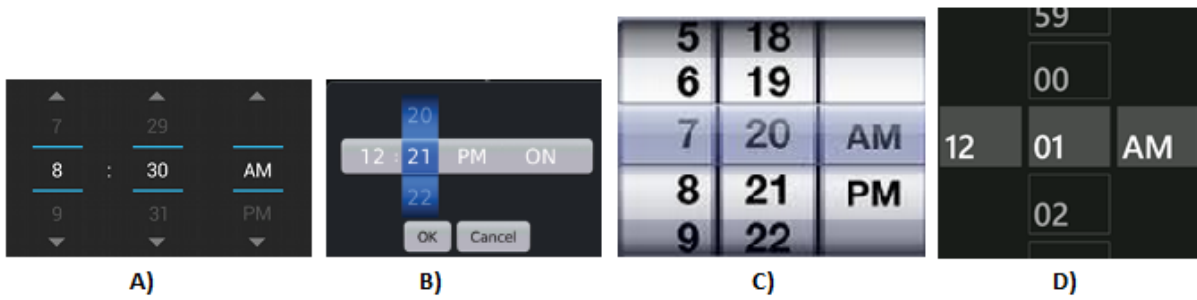


Figura C.21: Seletor de hora. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

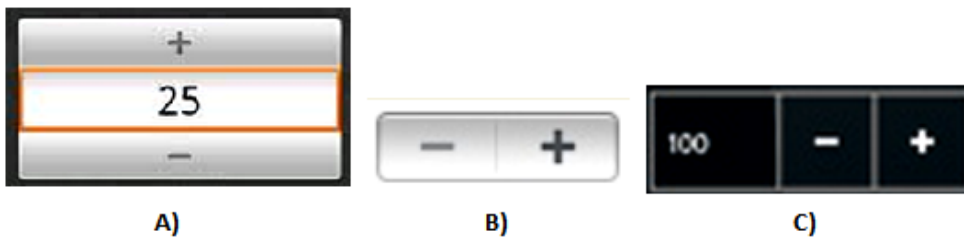


Figura C.22: Seletor numérico. A) Android; B) iOS; C) Windows Phone.

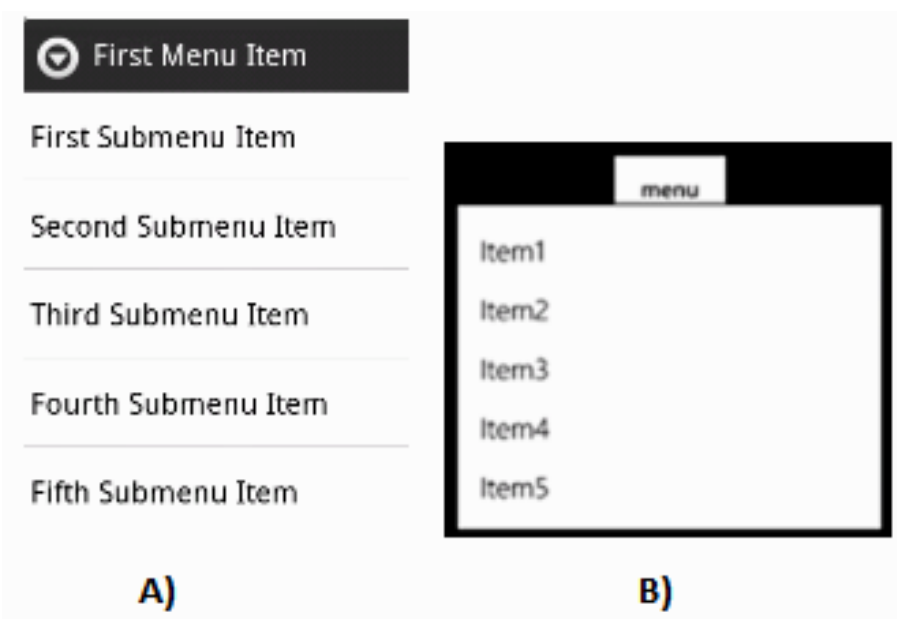


Figura C.23: Submenu. A) Android; B) Windows Phone.



Figura C.24: Switch. A) Android; B) Blackberry; C) iOS; D) Windows Phone.

Product	Sales	%
Chai	\$25.000	↓ -3
Chang	\$73.000	↑ 15
Aniseed Syrup	\$16.000	↑ 2
Chef Anton's...	\$114.000	↑ 31
Chef Anton's...	\$21.000	↓ -7
Grandma's B...	\$6.000	↓ -17
Uncle Bob's O...	\$6.000	↑ 4
Northwoods...	\$127.000	↑ 31
Keyboard	\$6.000	↓ -25

BlackBerry Devices	
8100 (Pearl)	4.3 2006 keyboard/trackball
8220 (Pearl Flip)	4.6 2008 keyboard/trackball
8300 (Curve)	4.5 2008 keyboard/trackball
8330 (Curve)	4.6 2008 keyboard/trackball
8700g	4.1 2005 keyboard

F	Q
Sarah Fahrzad	A
Rachael Falworth	B
Jose Fargo	C
Jim Ferris	D
Kimmie Fong	E
Lynn Foote	F
Pete Gardner	G
Monique Gaspard	H
	I
	J
	K
	L
	M
	N
	O
	P
	R
	S
	T
	U
	V
	W
	X
	Y
	Z
	''

First	Last
Charlie	Orsted
Karl	Neiman
Ed	Griswold
Mark	Frömmier
Zeb	Lehman
Larry	Ulam
Rich	Neiman
Karl	Paulson
Ben	Trask
Steve	Neiman
Vic	Myers
Herb	Lehman
Fred	Bishop-
Ben	Griswold
Noah	Trask

Figura C.25: Tabela. A) Android; B) Blackberry; C) iOS; D) Windows Phone.