
Interação humano-computador: técnicas para desenvolvimento de sistemas mobiles

Henrique Leal Tavares*

Carlos Eduardo da Trindade Ribeiro**

Marco Aurélio Migliorini Antunes***

Daiane de Lima Antunes****

Juliana Migliorini Antunes*****

Regina Célia Baptista Belluzzo*****

RESUMO

Esse artigo tem como objetivo demonstrar e exemplificar as diferentes maneiras de criação e aperfeiçoamento na interação do usuário com sistemas *mobiles*, detalhando boas práticas, conceitos e normas para que o público que tiver acesso à aplicação sentir-se “bem” usando

*Programador de software e tecnólogo em Análise de Sistemas pela FATEC. Email: henrique.tavares@fatec.sp.gov.br.

**Mestre em Televisão Digital pela UNESP Bauru, especialista em Sistemas para Internet pela UNIVEM - Marília/SP. Email: eduardoribeiro@etec.sp.gov.br

** Mestre em Televisão Digital pela FAAC Unesp Bauru e docente da

*** Mestre em Televisão Digital pela FAAC Unesp Bauru e docente das Faculdades Integradas de Bauru - FIB. Email: prof_marco@terra.com.br.

****Mestranda Programa de Pós-Graduação em Mídia e Tecnologia da Faculdade de Arquitetura, Artes e Comunicação (FAAC) da UNESP/Bauru. Email: daiane_s_l@yahoo.com.br.

*****Graduada em Pedagogia pela Faculdade de Filosofia Ciências e Letras Carlos Queiroz. Especialista em Educação Especial e Inclusiva pela FAAG de Agudos(2011). Docente de Educação Básica I da Prefeitura de São Pedro e Professora Orientadora de Aprendizagem da SESI - Santa Cruz do Rio Pardo. Email: ju.migliorini.antunes@gmail.com.

*****Pós-Doutorado em Ciências Humanas na Universidade Estadual Paulista Júlio de Mesquita Filho (UNESP). Email: rbelluzzo@gmail.com.

o aplicativo, tornando-o de fácil utilização e funcional. O estudo foi realizado por meio de consultas bibliográficas, criações de protótipos e aprofundamento de conceitos na interação humano computador aplicado aos dispositivos móveis.

Palavras-chave: Aplicativos. Interação. Dispositivos móveis.

1. INTRODUÇÃO

É fato que os *smartphones* estão conquistando um enorme espaço no mercado. No Brasil, não é diferente, segundo uma notícia da revista Exame: “Nas projeções do estudo, o Brasil terá um *smartphone* ou *tablet* por cada habitante, com um total de 208 milhões, para o biênio 2017-2018. ”

Com isso, surge uma nova modalidade de desenvolvimento *mobile* (móvel) que não se limita a *smartphones*, mas abrange dispositivos, como: *tablets*, relógios inteligentes e qualquer outra tecnologia vestível ou não.

O termo *computação vestível*, ou *tecnologia vestível* refere-se a uma nova abordagem de computação, redefinindo a interação humano-máquina, em que os aplicativos estão diretamente conectados com o usuário; em termos gerais, o usuário estaria “vestindo seu aplicativos”. (CUETO, 2014)

Outra pesquisa, desta vez desenvolvida por Simon Khalaf, engenheiro de produtos da Yahoo Flurry, apresenta que o uso de aplicativos cresceu, em média, 58% no ano de 2015, tornando alto o número de usuários que passaram a utilizar aplicativos, comparado com o dos últimos anos, que tiveram crescimentos de apenas 10 a 20 por cento (SIMON, 2016).

Porém, com este crescimento exponencial de *smartphones* e aplicativos, cresce também o olhar crítico dos usuários perante essas aplicações e suas respectivas funcionalidades e facilidades de uso, como aponta o estudo realizado pela *Forrester Consulting*, ao revelar que 60% dos usuários de aplicações *mobiles* consideram *design* e performance essenciais para usabilidade e interação, tirando até o foco principal do aplicativo de suas prioridades.

2. PRINCIPAIS FATORES DE UM BOM APP (APLICAÇÃO)

Existem vários padrões de desenvolvimento do *front-end* de uma aplicação *mobile*, que englobam tipos e tamanhos de fontes, telas de *splash*, tempo de resposta, cores e afins.

Desenvolvedores *front-end* são os que cuidam de todas as interfaces gráficas de uma aplicação. O termo, geralmente, é mais aplicado para o desenvolvimento

Web, em que são empregadas linguagens de marcações e estilos, como *HTML*, *CSS* e *jQuery* e, para auxiliar na realização desta tarefa, *frameworks* também são muito utilizados, como: *Ionic*, *Bootstrap* e *Wordpress*. (DEV MEDIA, 2012)

Segundo o blog *One Day Testing*, 47% dos usuários reportaram problemas com a performance do aplicativo; por isto, é necessário não só atenção ao visual, mas também ao seu desempenho enquanto executado em primeiro ou em segundo plano.

Para sabermos como desenvolver soluções inteligentes, intuitivas e satisfazer o usuário, faz-se necessária a captura de dados para ter conhecimento do que não fazer primeiramente.

Pesquisas feitas pela *Compuware* revelam que 48% dos usuários que utilizaram aplicações que não corresponderam às suas expectativas jamais utilizariam o *app* novamente e nem o indicariam; e 31% dos usuários de *apps* estudados dizem que a taxa de carregamento de determinada funcionalidade do aplicativo não deve ultrapassar dos 2 segundos, considerando 60% o tempo de resposta primordial para atender de forma positiva as expectativas dos usuários (COMPUWARE, 2015).

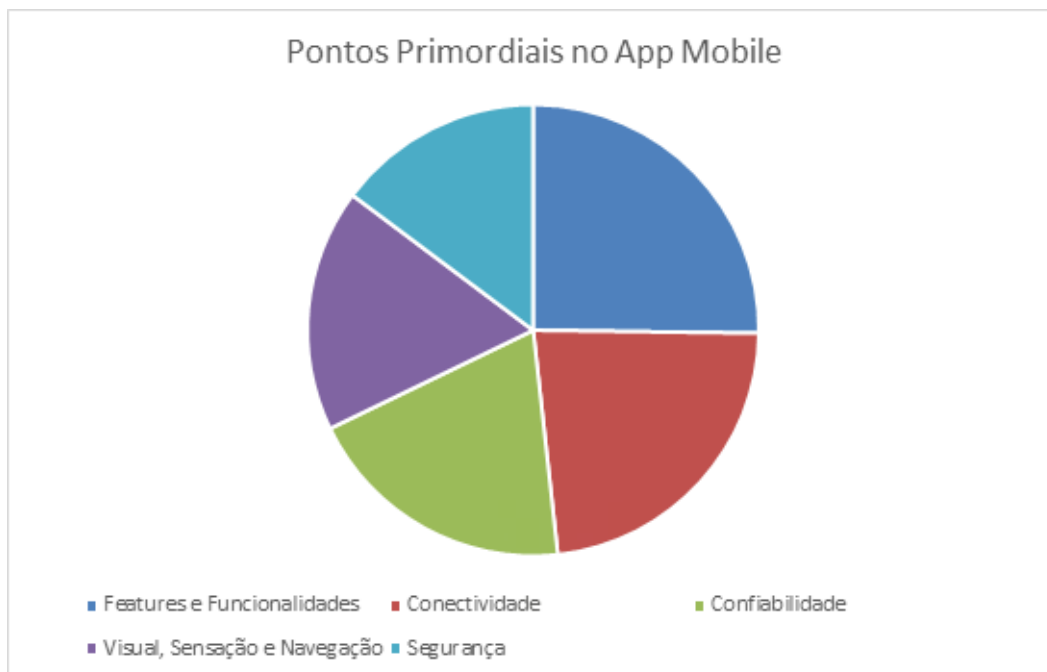
A satisfação é uma resposta do usuário na interação com o produto. Permite medir a extensão pela qual os usuários estão livres de desconforto e as atitudes em relação ao uso do produto. A satisfação pode ser especificada e medida pela avaliação subjetiva em escalas de desconforto experimentado, gosto pelo produto, satisfação com o uso do produto, aceitação da carga de trabalho quando realizam diferentes tarefas ou a extensão com os quais objetivos particulares de usabilidade foram alcançados. Outras medidas de satisfação podem incluir o número de comentários positivos e negativos registrados durante o uso do produto (FILARDI e TRAINA, p. 177, 2008).

Uma das melhores formas de prototipagem para verificar a expectativa do usuário é utilizar *wireframes*, que permitem a ele recolher mais requisitos do sistema a ser desenvolvido e, conseqüentemente, melhorará a interação e a satisfação do usuário futuramente.

Um *wireframe* é um guia visual básico de uma interface de usuário, ou seja, é uma sugestão de estrutura de uma tela. O objetivo do *wireframe* não é indicar se um botão será redondo ou quadrado, ou se a cor da aba vai ser verde ou azul, mas sim indicar o local mais correto para o botão. Desta forma, tanto o cliente, quanto os próprios desenvolvedores poderão visualizar como ficará uma interface do usuário (VALNEY, 2016).

Abaixo segue um gráfico sobre os pontos primordiais que os usuários consideram nas aplicações *mobile*:

Gráfico 1: Pontos Primordiais no App Mobile



Fonte: Neka (2016)

2. TÉCNICAS BÁSICAS PARA UM VISUAL AGRADÁVEL

Levando-se em consideração medidas de 2” a 5” para smartphones e 6” a 11” para tablets, foram estipuladas algumas boas normas pela *Return Path*, revista que desenvolveu um estudo sobre aceitação de aplicativos no mercado consumidor. Dentre essas normas estão:

- **Largura:** É recomendável que a largura tenha entre 320 e 550px. Se existirem muitas informações, evite usar várias colunas, procure usar uma única para que o texto seja melhor exibido na tela.
- **Fonte:** Use fontes com tamanho determinado em “px”. Defina o tamanho do corpo do texto com pelo menos 13px, ou use valores relativos, como: médium, large, x-large, xx-large. Não é recomendável usar os tamanhos small, x-small e xx-small.
- **Call To Action:** O CTA que utilizar, como por exemplo, a palavra *comprar*, esta deverá ser chamativa e grande o suficiente para o usuário poder lê-la sem ter a necessidade de utilizar zoom e tocar a função com o polegar. Também não é recomendável deixar *links* muito próximos.
- **Redes Sociais:** O usuário que interage com e-mails via *smartphone*,

ou *tablet* normalmente utiliza bastante as redes sociais; então, é provável que ele compartilhe, recomende ou “curta” os seus produtos. Sempre inclua estes *links* para compartilhamento nas redes sociais.

- **Navegação:** Navegação adequada, considerando o modelo mental por meio de janelas, menus, caixas de diálogos e painéis de controle em formato compreensível.
- **Interação e Feedback:** *Input* efetivo e *feedback* do *output* de informações para assegurar ao usuário que uma ação está em processamento.
- **Visualização de Informações:** Apresentação de informações por tabelas, gráficos, mapas e diagramas. Uma vez que a tela destes dispositivos ainda é pequena em comparação à dos computadores comuns (mesmo se tratando de tablets), é preciso se valer de componentes confiáveis que são capazes de apresentar uma boa quantidade de informação de modo compacto, conciso, de fácil visualização e acessível.

A revista também deixa claro ser de vital importância o teste em vários dispositivos, com dimensões e tecnologias nativas diferentes responsivas ou não (RETURNPATH, 2013).

3. FERRAMENTAS PARA DESENVOLVIMENTO DE INTERFACES MOBILES

Atualmente existem diversas plataformas que nos proporcionam ambientes propícios para o desenvolvimento *front-end*, *back-end* quanto *full-stack*. Abaixo seguirão algumas ferramentas que poderemos usar a nosso favor quando formos colocar a ideia de um aplicativo *mobile* em fase de teste ou prática.

Desenvolvedor *front-end* é o responsável que projeta as interfaces de uma aplicação, são eles que trabalham com HTML e o CSS, por exemplo, para construir as páginas web. Desenvolvedor *back-end* é o responsável por dinamizar a aplicação através das linguagens de programação (JAVA, C#, DELPHI, PHP), cada um dos papéis é muito importante e eles possuem dependência um do outro. Afinal de contas o que adianta um layout chamativo se a aplicação é estática, ou se o site é construído utilizando as melhores práticas de programação, contudo a sua usabilidade é fraca? (DEVMEDIA, p. 122, 2012)

Ser um desenvolvedor *full-stack* não implica, necessariamente, ser um desenvolvedor experiente, ou um desenvolvedor *web*. Basta o programador ter conhecimento razoável em tecnologias *back-end* e *front-end*, de modo que se sinta confortável com ambas as metodologias (VIDAL, 2016).

Prototyper App Design: É uma das mais poderosas ferramentas para criar protótipos interativos sem escrever códigos. Tem versões grátis e paga e está disponível para Mac e Windows. Nele, podemos ver como será um *webapp*, mas também para protótipo *apps iOS* ou *Android*.

É uma ferramenta muito fácil de usar e intuitiva. Importando imagens feitas com Photoshop ou Fireworks, em seguida pode ser feito um protótipo válido. Basta ir adicionando interatividade para as imagens já importadas. (NEKA, p. 39, 2015)

Para um protótipo simples, se a pessoa for um *freelancer*, ou, simplesmente, não quiser desenvolver, usar a versão gratuita será mais que suficiente.

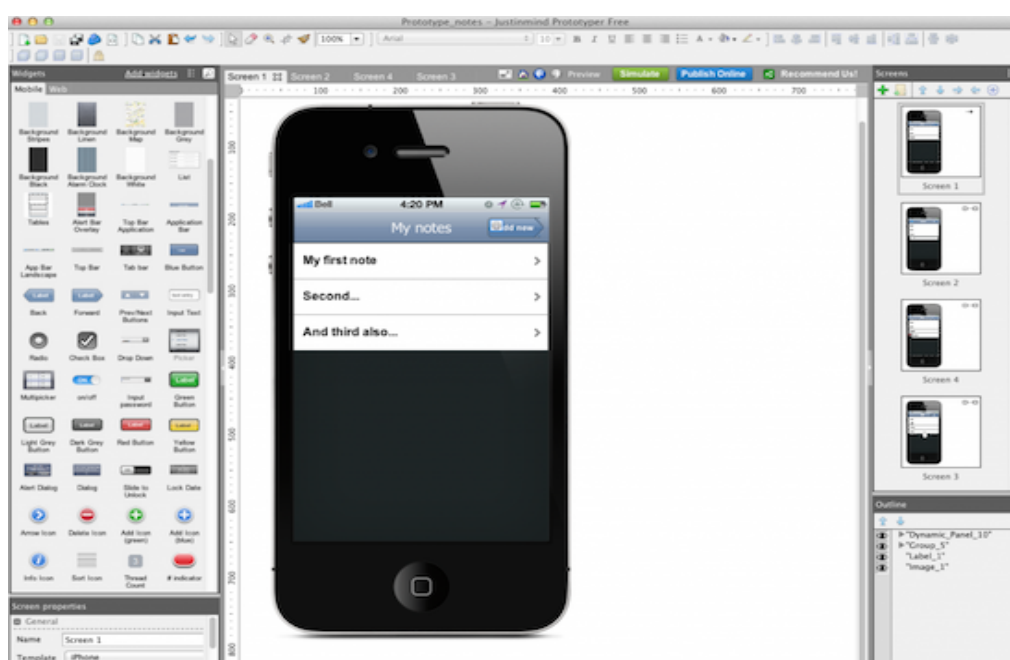


Figura 1: Prototyper App Design
Fonte: Compuware (2015)

NinjaMock: É uma oferta de ferramenta *on-line* em que os itens que incorporados oferecem um acabamento de desenho à mão livre. Em termos de *design*, este não é o mais adequado para criar qualquer tipo de aplicativo, especialmente se for levado em conta as mais recentes tendências e o gosto dos desenvolvedores de *apps* para o *flat design*.

A principal vantagem do NinjaMock é ser totalmente gratuito, embora um dos principais problemas encontrados, por exemplo, reside no fato de a interação do seu protótipo não se estender para além de ligações entre telas e telas.

Flat Design é uma tendência recente, que defende interfaces simples e claras, entre outros detalhes que veremos de seguida. [...] a cor dos elementos mais dominantes no Flat Design, assistindo-se a uma predominância das cores claras e retro (tais como salmão, violeta, verde e azul (ASCENSÃO, p.118, 2014).

O *Flat Design* é simples por natureza, deixando os textos e cores falarem por si.



Figura 2: NinjaMock
Fonte: Compuware (2015)

Ionic Creator: é um *framework* para desenvolvimento de aplicações para dispositivos móveis, de rápido e fácil desenvolvimento. Ele gera componentes que são customizados facilmente para tornar o desenvolvimento *front-end* uma tarefa mais simples, desta forma, gerando códigos em HTML5, CSS3 e JavaScript, tecnologias essenciais e que se adaptam facilmente a mudanças seja pelo próprio Ionic, ou por outro *framework*.

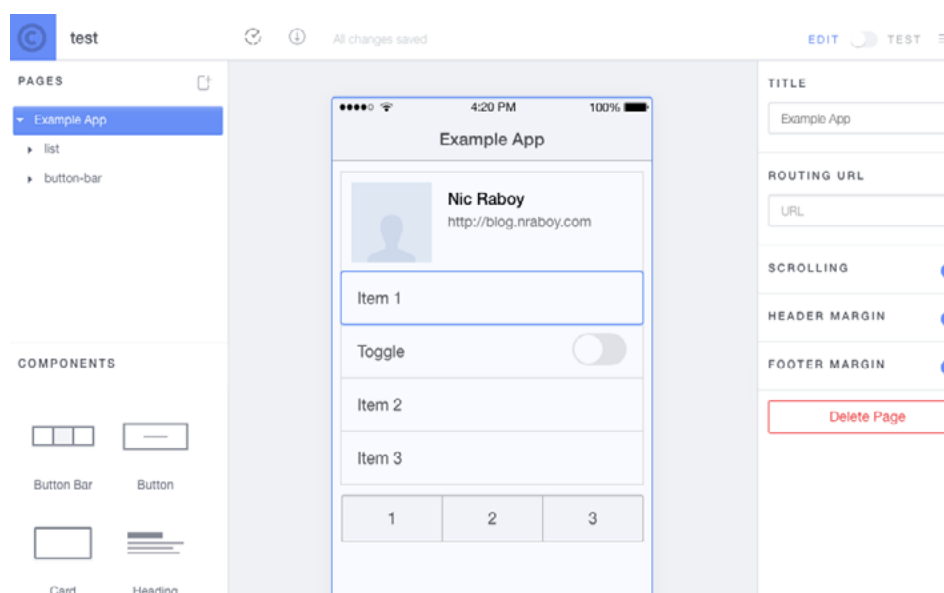


Figura 3: Ionic Creator
Fonte: Compuware (2015)

4. CONSIDERAÇÕES FINAIS

Indubitavelmente as aplicações *mobile* chegaram ao mercado para ficar e caíram no gosto do consumidor rapidamente.

No projeto que foi desenvolvido para teste e argumentação deste artigo, foram utilizadas diversas opções que os leques de funcionalidades das ferramentas acima citadas oferecem, tornando-se, assim, um protótipo menos complexo pela clareza de seu código fonte.

O estudo sobre *flat design* está presente na maioria dos aplicativos de sucesso e só tende a melhorar para contribuir com a interação do usuário na interface de nossa aplicação, tornando-se uma experiência mais agradável e intuitiva.

Cabe reforçar que são diversas as técnicas de atração de usuário para o seu aplicativo; porém, as que estão no topo das mais utilizadas e com melhores resultados são as que utilizam o *flat design* para criação da interface, que busca limpeza e objetividade nas telas, pois toda forma de abstração de complexidade deve ser considerada, e o processo, cada vez mais claro, significa muito para a boa interação do usuário com a máquina.

É recomendável que cada desenvolvedor tenha sua própria especialidade, um time específico para desenvolver o *front-end* da aplicação, pois os mesmos terão o conhecimento necessário e maior propriedade para tornar a aplicação mais legível e aplicar os padrões de boas práticas; da mesma maneira que devem existir desenvolvedores *back-end*, utilizando as melhores metodologias de desenvolvimento, abstraindo a aplicação de erros que possam implicar a perda de usuários.

REFERÊNCIAS BIBLIOGRÁFICAS

ASCENSÃO, C. P., **O que é flat design? 2014**. Disponível em: < <http://www.gestordeconteudos.com/Webdesign/Oque%C3%A9FlatDesign/tabid/4368/Default.aspx> >. Acesso: 15 set. 2015.

COMPUWARE, **Mobile Apps: What Consumers Really Need and Want, 2015**. Disponível em: < https://info.dynatrace.com/rs/compuware/images/Mobile_App_Survey_Report.pdf >. Acesso: 25 ago 2016.

DEVMEDIA, **O que é desenvolvedor front-end e back-end? 2012**. Disponível em: < <http://www.devmedia.com.br/> >. Acesso: 05 maio 2016.

FILARDI, A. L.; TRAINA, A. J. M., **Montando questionários para medir a satisfação do usuário: avaliação de interface de um sistema que utiliza técnicas de recuperação de imagens por conteúdo**. Porto Alegre, Artmed, 2008.

KHALAF, S. **Media, Productivity and Emojis Give Mobile Another Stunning Growth Year**. Disponível em: < <http://flurrymobile.tumblr.com/post/136677391508/stateofmobile2015> >. Acesso: 06 out. 2016.

LEBLANC, S. G., **A windows graphic user interface (GUI) for the five-scale model for fast BRDF simulations**. Disponível em: < <http://www.tandfonline.com/doi/abs/10.1080/02757250009532423> >. Acesso: 31 maio 2016.

NEKA, E. **Yeeply, App Design: Como fazer um protótipo de app. FlurryMobile. 2015**. Disponível em: < <https://pt.yeeply.com/blog/app-design-como-fazer-um-prototipo-de-app/> >. Acesso: 15 abr. 2016.

RETURNPATH. **Consumers Refusing To Cut The Cord. 2013**. Disponível em: < <https://returnpath.com/newsroom/consumers-refusing-cut-cord-email-marketers-see-multiscreen-mobile-interactions-entering-holiday-season/> >. Acesso: 10 mar. 2016.

VALNEY, M. **O que é um Wireframe e como criar a navegação do meu APP? 2016**. Disponível em: < <https://mariovalney.com/aula-9-o-que-e-um-wireframe-e-como-criar-a-navegacao-do-meu-app/> >. Acesso: 19 mar.2016.

VIDAL, T. **O que é um Full-Stack Developer? 2016**. Disponível em: < <https://thiagoacvidal.com/2015/07/16/full-stack-developer/> >. Acesso: 23 mar.2016.