

ANÁLISE DA UTILIZAÇÃO DE FRAMEWORK FRONT-END EM SISTEMA WEB ADAPTATIVO: UM ESTUDO DA PERSPECTIVA DO DESENVOLVEDOR

FRONT-END FRAMEWORK IN ADAPTIVE WEB SYSTEM USAGE ANALYSIS: A DEVELOPER PERSPECTIVE STUDY

ANÁLISIS DEL USO DEL FRAMEWORK FRONT-END EN UN SISTEMA WEB ADAPTATIVO: UN ESTUDIO DE PERSPECTIVA PARA EL DESARROLLADOR

31

Paulo Henrique Vieira Cândido¹
Danilo Camargo Bueno²
Carlos Henrique da Silva Santos³

Resumo: Este trabalho traz como contribuições aos desenvolvedores de aplicações Web sob análises UI e UX, criando relações e demonstrando a implementação de componentes do *Bootstrap*, por ser um dos *Framework Front-end* mais populares no mercado. Essas análises foram realizadas considerando o desenvolvimento de *Software as a Service* (SaaS) como, por exemplo, ao considerar diferentes tipos de menu e adaptação de tela. Os resultados demonstraram simplicidade no uso desses recursos tanto durante a codificação quanto na análise da interação humano-computador.

Palavras-chave: Aplicação Web. User Interface (UI). User Experience (UX). Responsividade.

Abstract: This work contributes to UI and UX Web application designer, generating relationships and demonstrating the implementation of *Bootstrap* components, because it is one of the most popular *Front-End* frameworks on the market. The analytics were considered the development of software as a service (SaaS), such as the development of different types of approaches of menu and screen adaptation. The results demonstrated the simplicity in twofold resources for codification and the human-computer interaction analysis.

Keywords: Web Application. User Interface (UI). User Experience (UX). Responsivity.

¹ Graduando em Licenciatura em Física. IFSP, campus Itapetininga. E-mail: paulo.vieira@aluno.ifsp.edu.br.

² Mestre em Ciência da Computação. Professor no IFSP, campus Itapetininga. E-mail: danilo.bueno@ifsp.edu.br

³ Doutor em Engenharia Elétrica. Professor no IFSP, campus Itapetininga. E-mail: carlos.santos@ifsp.edu.br

Resumen: Este trabajo aporta contribuciones a los desarrolladores de aplicaciones web en el análisis de UI y UX, creando relaciones y demostrando la implementación de los componentes de Bootstrap, siendo uno de los front-end más populares en el mercado. Estos análisis se realizaron considerando el desarrollo de Software como Servicio (SaaS), por ejemplo, al considerar diferentes tipos de menú y adaptación de pantalla. Los resultados demostraron simplicidad en el uso de estos recursos tanto durante la codificación como en el análisis de la interacción humano-computadora.

Palabras-clave: Aplicación Web. User Interface (UI). User Experience (UX). Capacidad de Respuesta.

Envio 18/07/2019

Revisão 19/07/2019

Aceite 25/10/2019

Introdução

A evolução da Internet tem como ponto de partida a criação da *World Wide Web* (WWW) por Tim Berners Lee em 1989. Desde então há cerca de aproximadamente 1,7 bilhões de páginas disponíveis (análise em julho de 2018), sendo esta uma estimativa em constante mudança (Internetlivestats, 2019).

Por outro lado, têm-se também de considerar a evolução das telas dos diferentes dos dispositivos computacionais que acessam a WWW, assim como a variabilidade de recursos visuais que vêm sendo disponibilizados para todos esses (Jakobson; Reuben, 2015).

Esse crescimento faz notar a importância da velocidade no desenvolvimento de aplicações Web e das notáveis formas de adaptação das interfaces já disponíveis atualmente no mercado de desenvolvimento (Souza; Alvarenga, 2004).

Ao se considerar a maturidade, manutenibilidade e segurança a ser provida pelas páginas a serem desenvolvidas para a composição de sistemas mais robustos, deve-se considerar o emprego de padrões de desenvolvimento de projetos, onde os recursos a serem utilizados pode ser o principal fator de organização dos projetos. Portanto, para atender esses requisitos de sistema e variável expansão de recursos para a WWW, considera-se como essencial o desenvolvimento de diferentes técnicas e ferramentas que acelerem a implementação das interfaces. (Otto; Thornton, 2017; Pagani, 2011).

Neste contexto, desponta-se como necessário estudar e evoluir os *Frameworks Front-end* (FeF), os quais integram diferentes componentes, *containers*, objetos e outras estruturas de dados que encapsulam variados recursos de interface audio-visual para potencializar a usabilidade estudada na área de Interação Humano-Computador (Mendoza; Gu, 2018), (Kaluz; et Al, 2015), (Harb et Al., 2011).

Para tanto, este trabalho tem como objetivo geral uma revisão da literatura sobre a adaptação de interfaces e o uso de FeFs, tomando-se como base o *Bootstrap* por ser um dos mais explorados no mercado, para a implementação de um aplicativo Web tomado como estudo de caso para analisar os recursos apresentados. Além disso, esse trabalho também traz como contribuição a continuidade dos estudos apresentados por Cândido (2017) e Bueno (2016), os quais são abordados em mais detalhes na seção de Referencial Teórico.

Assim, tem-se como especificidades de deste trabalho uma revisão sobre a definição dos padrões e componentes do FeF *Bootstrap* que integram a implementação do aplicativo aqui apresentado, com descrições ao longo processo de desenvolvimento. Também é considerada a codificação e a forma de abordagem do FeF para os padrões analisados como objetivos específicos deste artigo.

Da perspectiva do desenvolvedor, neste artigo considera-se a demanda de conhecimento necessário para que eles possam atuar no mercado, tendo como foco principal o desenvolvimento de Aplicações Web, considerando sua Adaptabilidade. Para isso, considera-se adaptabilidade como sendo a capacidade de adaptação da interface ao usuário e aos diferentes dispositivos (Otto; Thornton, 2018; Wang et Al, 2017).

Ainda, por esta área ser bastante ampla, as análises dos recursos consideram as perspectivas de *User Experience* (UX) e *User Interface* (UI), trazendo também características pertinentes da implementação dos componentes utilizados, tomando-se como base sua documentação, com o intuito de nortear o entendimento e explorar a individualidade de cada um destes (Matiola, 2015; Cândido, 2017).

Para a apresentação este trabalho está organizado em uma seção referencial teórico, que traz as definições de FeF, Padrões de Design da Interface do Usuário, Adaptação de Interfaces, *User Experience* (UX) e *User Interface* (UI). Na sequência, uma seção desenvolvimento que descreve as etapas de implementação dos componentes dos *Frameworks* e o seu comportamento adaptativo no aplicativo em questão. Por fim, as considerações finais e trabalhos futuros são apresentados na Seção Considerações Finais.

Referencial Teórico

Esta seção apresenta de forma progressiva conceitos e definições de *Frameworks Front-end* (FeF), Padrões de Projeto de Interface do Usuário, Adaptação de Interfaces, *User Interface* (UI) e *User Experience* (UX).

Iniciando com o ferramental base tecnicamente conhecido por *Framework*, o qual pode ser entendido como um conjunto de componentes que formam uma estrutura básica para o desenvolvimento de algo. De forma análoga a um arcabouço, um *Framework* é composto por elementos que resolveram problemas comuns enfrentados anteriormente. Portanto, pode-se

considerar que os *Frameworks* são recursos computacionais para o desenvolvimento de aplicações Web, que disponibilizam ferramentas que auxiliam na solução de problemas de natureza semelhante a experiências prévias (Pagani, 2011; Maldonado; Et Al., 2012; Bueno, 2017; Cândido, 2017).

Esses *Frameworks* para o desenvolvimento Web são usualmente classificados em (i) FeF, que são compostos por elementos relacionados ao desenvolvimento de interfaces, sendo estes desenvolvidos em HTML, CSS e Javascript; e (ii) *Frameworks Back-end* que lidam com o que acontece do lado do servidor, geralmente compostos por linguagens como Python, Ruby ou PHP (Chan, 2016; Pagani, 2011). A Figura 1 é uma representação gráfica para facilitar a compreensão da integração desses frameworks com exemplos de tecnologias.

Figura 1: Desenvolvimento Front-end e Back-end



Fonte: O Autor (2019).

Há diversos FeF disponíveis no mercado, sendo alguns listados nos trabalhos de Jain (2016) e Bueno (2017), sendo analisados por Cândido (2017), com destaque principal a três FeF, o *Bootstrap*, *Materialize CSS* e o *Foundation*. Seguindo as conclusões de Cândido (2017), torna-se interessante os aspectos do *Bootstrap*, visto que nas conclusões do autor este se mostrou mais conciso e melhor estruturado para o desenvolvimento de aplicações Web.

O *Bootstrap*, que em tradução livre para o Português seria alça de inicialização, o qual surgiu da necessidade da padronização de componentes utilizados internamente pelos

desenvolvedores e funcionários do *Twitter*⁴. Em 2011 foi publicado por Mark Otto e Jacob Thornton no Github e segue atualmente na sua versão 4.1 (Otto; Thornton, 2017).

Padrões de Projeto de Interface do Usuário

O desenvolvimento das interfaces para usuários, quando bem elaboradas e parametrizadas, podem seguir os Padrões de Projeto de Interface do Usuário - *User Interface Design Patterns* (UIDP). O UIDP podem ser definidos, segundo a *Interaction Design Foundation* (2018), e em tradução livre para o Português como sendo “descrições das melhores práticas no design da interface do usuário. São soluções gerais e reutilizáveis para problemas comuns” (*Foundation*, 2018). Também é interessante considerar a definição de Toxboe (2016) dada por:

“São soluções recorrentes que resolvem problemas comuns de projeto. Os padrões de design são pontos de referência padrão para o designer de interface com o usuário experiente.” (Toxboe, 2016)[tradução livre para o Português]

36

Tomando-se como base as definições apresentadas, compreende-se que a UIDP surge como um facilitador para os desenvolvedores, visando facilitar e agilizar o trabalho dos desenvolvedores de interfaces, tomando-se como base a redução de erros ou falhas comuns em experiências anteriores.

Essa afirmação é aderente à de Toxboe (2016), que destaca que os padrões de UIDP surgem do enfrentamento de problemas pelo desenvolvedor durante o desenvolvimento da aplicação, portanto, cada padrão possui determinados elementos para se constituir como padrão.

Para isso, UIDP é organizada por *Foundation* (2018) em elementos para i) problema que deve ser reconhecido como uma possível, ou confirmada, dificuldade encontrada na usabilidade da aplicação em determinado ii) contexto de uso, sendo este baseado em um ou mais iii) princípios de design. Tendo uma iv) solução comprovada, que determina apenas o

⁴ <https://twitter.com/>

núcleo do problema e como o resolver, sendo esta unida do v) porquê utilizar o padrão juntamente com vi) exemplos e a sua forma de vii) implementação.

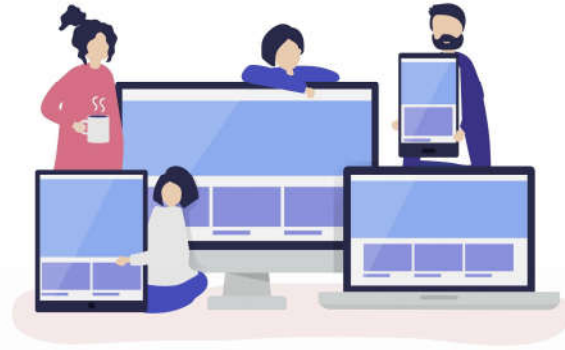
Além disso, as UIDP usualmente são organizados em bibliotecas, sendo classificados por categorias de acordo com os problemas em comum que eles têm como foco resolver. Para exemplificar, Cândido (2017) disponibilizou de forma gratuita e aberta um compilado de padrões, utilizados para demonstração de comportamento e implementação através do *GitHub* sob o endereço <https://paulhenrique.github.io/frameworks/>.

Adaptação de Interfaces

Na Ciência da Computação há uma área de estudos denominada Interação Humano-Computador (IHC), a qual avalia os recursos e as formas de interação do humano com o computador. Além disso, nesta área há a especificidade de estudos nomeados por Adaptação de Interfaces, que tem como premissa básica analisar as diferentes interfaces apresentadas em por variados tipos dispositivos, esta, por sua vez, subdivide-se em Interfaces Adaptativas e Interfaces Adaptáveis (Rosa; Silva, 2013; Bueno, 2017).

As Interfaces Adaptáveis moldam-se ao comportamento e preferências da utilização de do usuário. Esses sistemas das escolhas deste para que o usuário esteja no controle da aplicação, sendo que quando este muda suas opções, a interface se adapta a fim de suprir as suas escolhas (Bueno, 2017, Gullà et. al, 2015).

Figura 3: Interfaces adaptativas;



Fonte: O Autor (2019).

Já as Interfaces Adaptativas adaptam-se ao dispositivo em que estão sendo utilizadas, ou seja, a recursos e comportamentos ligados ao seu funcionamento. Esta categoria subdivide-se em outras duas: a i) Multimodalidade, que é como o usuário interage com a interface, seja por diferentes movimentos na tela, por voz ou movimentos e ii) Sensibilidade ao contexto, que envolve a capacidade de adaptação da interface ao contexto em que a aplicação está sendo executada e o qual o usuário está inserido, ou seja, circunstâncias de baixa iluminação, de muitos ruídos e de movimento, por exemplo (Harb et al., 2011; Bueno, 2017, Gullà et. al, 2015).

Nesse contexto surge o conceito de *Responsive Web Design* (RWD) (Figura 3), que envolve a capacidade que a aplicação tem de reconhecer o contexto do tamanho da tela do dispositivo e adaptar-se à este, estando preparada para diferentes tamanhos de tela e tendo como tecnologias básicas o Javascript, HTML e CSS (Harb et al., 2011; Jain, 2015; Bueno, 2017, Gullà et. al, 2015).

User Interface (UI) e User Experience (UX)

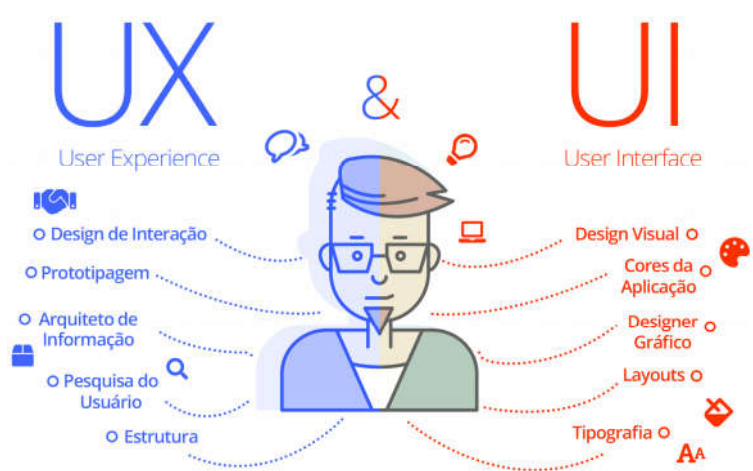
A *User Experience (UX)* e *User Interface (UI)*, que em tradução livre para o Português considera-se a Experiência do Usuário e Interação ou Interface do Usuário, respectivamente, estudam as interfaces quanto as experiências dos usuários e aos recursos de interface disponibilizados, sendo termos que vêm se popularizando nos últimos anos (Matiola, 2015).

Para facilitar a descrição dos tópicos abordados em cada uma dessas áreas, a Figura 4 apresenta uma representação simplificada associando palavras-chave as suas respectivas áreas.

Essa representação é importante devido a proximidade dessas duas áreas, o que muitas vezes torna os termos confusos.

Iniciando com a UI Matiola (2015), é voltada ao projeto da interface do usuário, focando-se na interação do usuário interage com a interface, sendo esta de um dispositivo ou software.

Figura 4: As características de UX e UI.



Fonte: O Autor (2019).

A Udacity (2018) descreve os projetistas de UI como sendo os responsáveis pela aparência das interfaces, sendo:

“[...] escolher as cores e a tipografia a ser usada no site, o formato e o posicionamento dos botões, menus e links, garantir que cores semelhantes tenham funções semelhantes e que tanto a versão desktop quanto a versão mobile do serviço sejam reconhecíveis como edições diferentes de um mesmo serviço, por exemplo.” (Udacity, 2018).

Portanto, ao se considerar que UI está associada aos aspectos visuais da interface do sistema e que, por meio dessas observações, pode-se também verificar as experiências relatadas

pelos usuários por meio da UX, tornando-se uma associação necessária para avaliação de requisitos de interface.

Para tanto, considera-se o que Gonzalez (2017) definiu, que UX é “[...] o estudo comportamental dos usuários, com o objetivo de entregar uma melhor experiência de uso nos produtos sendo eles digitais ou não [...]”. Udacity (2018) já aponta que o projetista de UX tenta imaginar como seria o comportamento do usuário ao utilizar a interface, podendo recorrer a pesquisas com os usuários para entendê-los.

Seguindo as definições de Udacity (2018), Gonzalez (2017) e Mäntylä (2015), e a representação gráfica da Figura 4, pode-se abstrair que UI está contido em UX, sendo considerados importantes para avaliar a usabilidade do sistema.

Estudos de Casos de Uso de Frameworks sob as Perspectivas de UX e UI

Para o estreitamento dos casos de uso e considerando-se a adequação de análise deste trabalho, toma-se como base exemplos de análises de frameworks os trabalhos de Bueno (2017), Bueno e Zaina (2016) e Cândido (2017) considerando os FeFs quanto a sua popularidade e propostas apresentadas pelos autores para trabalhos futuros.

Bueno (2017) propõe uma abordagem híbrida para adaptação de interfaces, a qual chamou de *HyMobWeb*. Ele aborda conceitos relacionados a adaptação de interfaces, considerando as interfaces adaptáveis, interfaces adaptativas e suas subcategorias: Multimodalidade e Sensibilidade ao contexto.

Ainda, Bueno (2017) segue as premissas descritas por Jain (2015) para selecionar o FeF de pesquisa sua popularidade e utilização em aplicações Web. Também segue o percurso que considerou pesquisas que destacaram as dificuldades dos usuários em utilizar aplicações não adaptadas aos dispositivos móveis para o desenvolvimento do *HyMobWeb*.

O segundo trabalho tomado como estudo de caso foi realizado Cândido (2017), em que ele apresentou os problemas de interação na adaptação de interface Web para dispositivos móveis nos FeF. Para isso, ele utilizou os FeFs Bootstrap, Materialize e Foundation e seguiu análises com base em UIDP. Isso resultou em observações acerca das características individuais

de cada FeF e seu comportamento quanto a Multimodalidade e Sensibilidade ao contexto, tendo como principal área os dispositivos móveis.

No ano de 2016 Bueno e Zaina (2016) apresentaram resultados de seus estudos exploratórios que realizaram com a adaptação Web de FeF para dispositivos móveis utilizando como base os usuários finais e o FeF *Bootstrap*, apontando soluções como *Responsive Web Design* (RWD). Eles conseguiram identificar problemas na adaptação de elementos de interação proporcionadas pelo FeF. Para isso, eles aplicaram um questionário baseado no método SAM (*Self Assessment Manikin*), que é uma técnica de avaliação que visa medir a emoção, motivação e dominância do usuário durante a experimentação do aplicativo.

Esse conjunto de estudos de caso, com suas respectivas experiências e análises descritas de maneira clara e cientificamente detalhada, possibilitou um interessante ponto de partida para os estudos apresentados neste trabalho.

Metodologia de Análise UIDP do FeF: Definindo recursos e roteiro avaliativo.

41

A realização das análises torna necessário um planejamento e organização do fluxo de implementação, para isso serão definidos na atual seção a (i) metodologia e definição dos UIDP que serão implementados e a (ii) metodologia que será utilizada para as análises bem como a (iii) organização dos fatores utilizados na análise.

Metodologia e Definição dos UIDP

Para definição dos UIDP que serão utilizados nas implementações decidiu-se por observar os trabalhos de Bueno (2018), Jain (2016) e Cândido (2017) onde, em meio a outras descrições, os autores analisam o *Bootstrap*, objeto de estudo aqui também. As três pesquisas mostram características do FeF proveitosas a esta análise.

Bueno (2018) e Jain (2016) trazem uma classificação inerente a grande utilização do *Bootstrap*, como FeF mais utilizado em 2016 e 2017, para tanto a utilização de seus

componentes é de grande proveito para essa análise, mas para sua interpretação de forma concisa e clara é necessária uma ordem de relevância de componentes.

Neste contexto, torna-se muito interessante a utilização de bibliotecas de UIDP, o *Bootstrap* disponibiliza na sua documentação diferentes categorias com diversos componentes, os quais serão trazidos à análise com suas relações em UIDP.

Observando a necessidade da utilização de uma biblioteca e também de que já houveram trabalhos que realizaram classificações do mesmo gênero, ou seja, que identificaram componentes do *Bootstrap* relevantes à análise, decidiu-se aproveitar da classificação realizada por Cândido (2017) para a escolha dos UIDP do *Bootstrap* que serão implementados e analisados aqui.

Como metodologia de escolha dos UIDPs se definiu um roteiro de análise que respeitasse: a (i) pertinência do UIDP para a usabilidade do aplicativo, o (ii) contexto necessário para implementação, a (iii) necessidade do padrão para a interface.

Cândido (2017) aponta 6 UIDPs do *Bootstrap*: *Module Tabs*, *Breadcrumbs*, *Carousel*, *Vertical Dropdown Menu*, *Calendar Picker* e *Alternating Row Colors*. Utilizando o roteiro de análises com os UIDPs do autor e análise obteve-se a Tabela 1, a organização desta é dada em parâmetros de análise de 1 a 5 baseados em Toxboe (2016), onde 5 o UIDP torna-se de grande importância para o parâmetro e 1 quando o padrão não possui grande relevância para o parâmetro.

Tabela 1: Análise de relevância dos UIDPs para implementação.

UIDP/Parâmetro	i	ii	iii
<i>Module Tabs</i>	4	2	2
<i>Breadcrumbs</i>	2	3	3
<i>Carousel</i>	2	1	1
<i>Vertical Dropdown Menu</i>	4	5	5
<i>Calendar Picker</i>	2	2	3
<i>Alternating Row Colors</i>	4	5	5

Fonte: O Autor (2019).

Perante os dados apresentados, decidiu-se utilizar dos UIDPs *Vertical Dropdown Menu* e *Alternating Row Colors* devido a grande pertinência apresentada por ambos para a usabilidade e estruturação da interface do aplicativo.

43

Metodologia de Análise dos UIDPs

Partindo da definição dos UIDPs que serão implementados torna-se necessário um processo de implementação destes no aplicativo, portanto, torna-se necessário, como metodologia, (i) identificar os correspondentes componentes aos UIDPs na documentação do *Bootstrap*, (ii) descrever de forma breve o comportamento e a importância deste na interface em questão, (iii) demonstrar a implementação através de códigos HTML e (iv) analisar de forma comparativa e descritiva o comportamento do padrão e (v) os resultados obtidos com essa implementação.

Implementação e Análise dos UIDPs

A implementação dos UIDPs foi realizada em um aplicativo de visualização de dados científicos, portanto, a importância da praticidade, simplicidade e fácil localização do usuário na interface é de extrema importância e, para o desenvolvedor, perpassa por diferentes processos até a conclusão da interface e sua devida experimentação.

Tendo em vista a análise da implementação dos UIDPs do FeF *Bootstrap*, e não de sua estruturação inicial, durante o decorrer das análises considerar-se-á que o leitor possua conhecimento dos templates iniciais e a estrutura mínima essencial para sua utilização, que pode ser encontrada na documentação oficial do FeF sob o endereço: <https://getbootstrap.com/docs/4.0/getting-started/introduction/>.

Serão tratados nas seguintes subseções as características dos padrões *Vertical Dropdown Menu* e *Alternating Row Colors* de acordo com suas definições por Toxboe (2016), respeitando a metodologia descrita na Seção Metodologia de Análise dos UIDPs, ordenando-se por i) correspondências na documentação - sendo esta a oficial disponibilizada pelo *Bootstrap* em: <https://getbootstrap.com/docs/4.0> - (ii) Comportamento na interface, (iii) demonstração de implementação, (iv) análise comparativa/descritiva e (v) resultados obtidos.

Ao final da análise dos dois componentes serão descritos os comportamentos esperados e os encontrados a partir das perspectivas UX e UI, possíveis falhas e propostas às necessidades observadas durante o desenvolvimento e utilização do aplicativo em fase de testes.

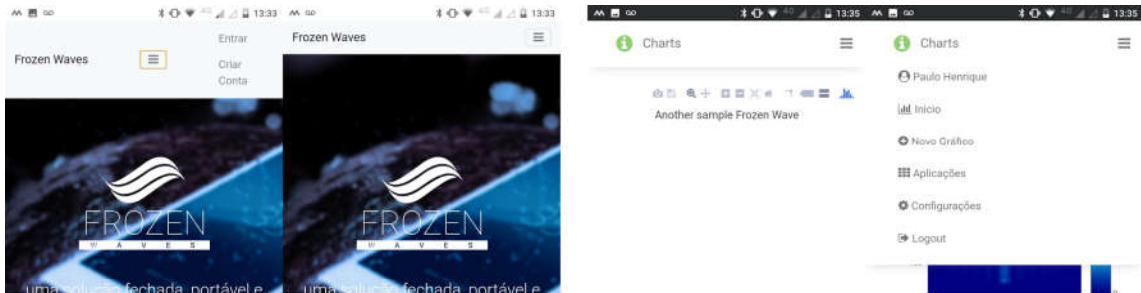
44

UIDP *Vertical Dropdown Menu*

i) Correspondências na Documentação: O UIDP *Vertical Dropdown Menu* pode ser encontrado em diferentes categorias dentro da documentação do *Bootstrap*, sendo caracterizado como uma variação da *Navbar*, uma variação de *Button* e possuindo uma categoria específica para botões, a *Dropdown*.

ii) Comportamento na Interface: O *Vertical Dropdown Menu* surge para o auxílio na disposição de itens de menu que precisam de espaço vertical e o mesmo pode se tornar um problema. Na interface do aplicativo utilizou-se deste UIDP para auxílio da navegação mobile do usuário em todo o sistema, para isso se aproveitou do componente correspondente *Navbar* do *Bootstrap*. A Figura 5 demonstra o funcionamento deste, na Figura 5 (a) na página inicial e Figura 5 (b) na página de utilização do aplicativo.

Figura 5: Comportamento do *Vertical Dropdown Menu*.



(a)

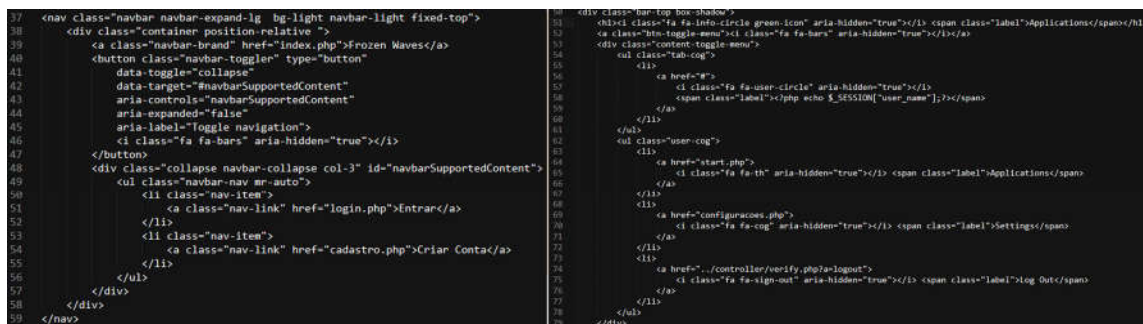
(b)

Fonte: O Autor (2019).

45

iii) **Demonstração de implementação:** A forma de implementação é descrita na documentação oficial do *Bootstrap*, que mostra de forma concisa como esta deve ser feita. Na Figura 6 (a) é possível ver como foi a implementação em HTML com as marcações das classes padrões para página inicial do aplicativo, na Figura 6 (b) a implementação do padrão com diferentes classes definidas e já configuradas pelo autor.

Figura 6: Implementação do UIDP Vertical Dropdown Menu.



(a)

(b)

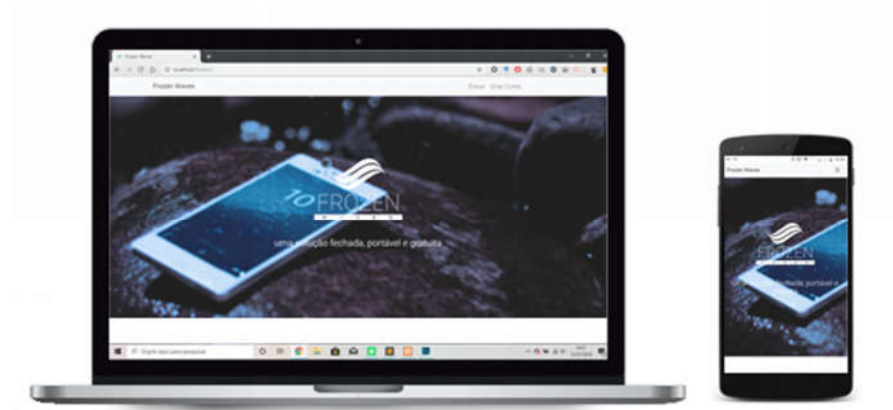
Fonte: O Autor (2019).

iv) Análise comparativa/descritiva: É importante notar o que é inerente ao comportamento do UIDP, este é configurado pelos códigos na Figura 6, a estrutura é baseada em um elemento *div* identificado por diferentes classes, responsáveis pelo comportamento visual. É possível observar na Figura 6 (b) a utilização de uma *Unordered List* (*ul*) – uma lista não ordenada, as classes que podem ser observadas são *navbar-nav*, que é relacionada ao comportamento e exibição da sua estrutura em forma de barra e a classe *mr-auto*, que é relacionada a posição do menu referente à *navbar*, ou seja, o posicionamento que esta fica em relação ao elemento onde está incluída.

v) Resultados obtidos: Após a implementação na interface tornou-se notável o impacto para a navegação dentro do aplicativo, a usabilidade ficou mais clara devido a facilidade na troca de páginas. Também é importante observar que devido aos recursos utilizados durante a implementação não é necessário utilizar de dois menus mas apenas um, que se a tela do dispositivo de acordo com a configuração do *grid* do *Bootstrap*. A Figura 7 mostra a relação deste comportamento de adaptatividade.

46

Figura 7: Relação interface *Desktop* e *Mobile*.



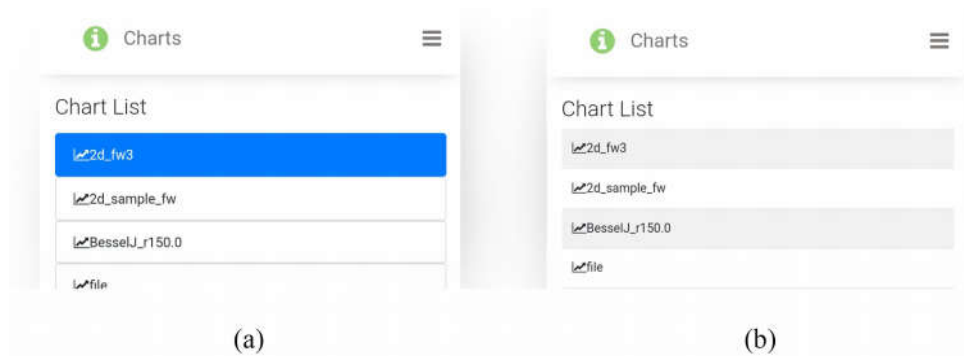
Fonte: O Autor (2019).

UIDP *Alternating Row Colors*

i) Correspondências na Documentação: O UIDP *Alternating Row Colors*, considerando sua definição por Toxboe (2016), pode ser encontrado em uma subcategoria de *Content*, a *Tables* que contém os componentes relacionados a exibição de dados em tabelas. O componente que corresponde ao *Alternating Row Colors* é o *Striped rows*.

ii) Comportamento na Interface: O *Alternating Row Colors* surge como um auxílio para o usuário se localizar em uma lista ou tabela com vários itens alternando as cores das linhas da lista. No aplicativo há uma lista dos itens já criados pelo usuário que permite a seleção para visualização. A Figura 8 (a) demonstra a implementação original utilizada anteriormente no aplicativo para esta seleção, a Figura 8 (b) mostra a utilização do UIDP para realização da tarefa.

Figura 8: (a) Lista padrão do Bootstrap; (b) Lista utilizando o *Alternating Row Colors*.



Fonte: O Autor (2019)

iii) Demonstração de Implementação: A forma de implementação é descrita na documentação oficial mas é interessante notar que não é necessária a utilização de todos os elementos descritos para alcançar o objetivo do UIDP. Na Figura 9 (a) é possível ver o código respectivo a implementação visualizada anteriormente na Figura 8 (a) enquanto na Figura 9 (b) é possível observar a implementação da Figura 8 (b).

Figura 9: (a) Implementação de lista simples; (b) Implementação do *Alternating Row Colors*.

<pre> 15 <div class="col-12 col-md-4 d-none"> 16 <h4 class="thin">Chart List</h4> 17 <ul class="list-group mt-3 list-chart"> 18 <?php 19 \$charts = new Charts; 20 \$data = \$charts->create_list_charts(); 21 foreach (\$data as \$entry): 22 \$item["src"] = \$entry; 23 \$item["title"] = pathinfo(\$entry, PATHINFO_FILENAME); 24 ?> 25 <a class="btn-PreviewGraph" data-file="<?> \$item["src"] ?>"> 26 <li class="list-group-item list-graph"> 27 <i class="fa fa-line-chart"></i><?> \$item["title"] ?> 28 29 30 <?php endforeach; ?> 31 32 </div> </pre>	<pre> 33 <div class="col-12 col-md-4"> 34 <h4 class="thin">Chart List</h4> 35 <table class="table table-striped"> 36 <tbody> 37 <?php 38 \$charts = new Charts; 39 \$data = \$charts->create_list_charts(); 40 foreach (\$data as \$entry): 41 \$item["src"] = \$entry; 42 \$item["title"] = pathinfo(\$entry, PATHINFO_FILENAME); 43 ?> 44 <tr> 45 <a class="btn-PreviewGraph" data-file="<?> \$item["src"] ?>"> 46 <td class="list-graph"> 47 <i class="fa fa-line-chart"></i><?> \$item["title"] ?> 48 </td> 49 50 </tr> 51 <?php endforeach; ?> 52 </tbody> 53 </table> 54 </div> </pre>
--	--

(a)

(b)

Fonte: Autor (2019).

iv) **Análise comparativa/descritiva:** É possível notar pelos processos de implementação e pela visualização do resultado na interface uma grande variação, na Figura 9 (b) é utilizada a estrutura de tabela enquanto na Figura 9 (a) é uma estrutura de lista, que pode ser mais comum em estruturas de navegação. Esta diferença pode ser de grande impacto na estrutura de todo o sistema, ao utilizar a estrutura da Alternating Row Colors há uma facilidade e simplicidade muito maior, visto que não há necessidade de muitas classes para se suprir a necessidade, enquanto ao utilizar a estrutura de lista com links há a necessidade de uma intervenção do desenvolvedor através de CSS, ou Javascript, para modificação do comportamento padrão do componente.

v) **Resultados obtidos:** Após a implementação do UIDP foi possível notar a facilidade de identificação do item pelo usuário, este se torna mais simples devido a variação das cores, o que torna mais simples a percepção do item que se quer escolher.

Análise da perspectiva UX e UI

Durante a análise dos UIDPs foram notados alguns pontos que podem ser interessantes, devido o escopo do trabalho. Estes serão observados nesta seção e serão expostos a partir das perspectivas UX e UI.

UIDP Vertical Dropdown Menu: Ao observar o funcionamento de toda a categoria *Navigation* de Toxboe (2016) a utilização do UIDP *Vertical Dropdown Menu* possui características extremamente importantes para o funcionamento de todo o aplicativo. O *Bootstrap* utiliza de uma abordagem que pode ser considerada simples mas sua implementação

pode ser confusa a usuários iniciantes, visto que estes podem não perceber a troca da disposição dos itens e a relação de classes expostas na documentação oficial. Apesar de a implementação do UIDP, assim como mencionado na Seção UIDP *Vertical Dropdown Menu* e Seção UIDP *Alternating Row Colors* ser descrita de forma simples, a documentação não deixa claro a necessidade do prévio conhecimento do desenvolvedor em outros componentes, o componente *grid* é um exemplo, ele surge em diversos momentos na documentação e isto pode ser um possível problema para o desenvolvedor.

Da perspectiva UI o componente mostra-se eficaz, Cândido (2017) demonstra isso de forma sucinta, mas para que haja a real adaptatividade do aplicativo é necessário que o desenvolvedor tome de devida atenção durante o desenvolvimento e realize diversos testes com diferentes dispositivos e telas de diferentes tamanhos. Durante as primeiras experiências com o padrão foram observadas falhas em dispositivos de diferentes tamanhos de tela devido a falta de informação na área documentação.

UIDP *Alternating Row Colors*: Por perceber a necessidade de trabalhar com informações e dados já preenchidos por parte do usuário, a categoria *Dealing with Data* surge como um grande facilitador. O *Alternating Row Colors* se mostra como grande auxílio para exibição desses dados, mas o componente disponibilizado pelo *Bootstrap* falha na perspectiva UX, o usuário pode não identificar o item que está ativo e, por padrão, não há a disponibilidade de um recurso que demonstre qual a classe ativa. Isso retoma a discussão anterior, há outros componentes no FeF e que estão descritos na documentação oficial mas não são relacionados à essas necessidades, isso pode se tornar grande empecilho para o desenvolvedor que não conhece dessas informações.

Da perspectiva UI o componente oferece adaptatividade notável, a documentação mostra classes, melhor descritas nos trabalhos anteriores, que demonstram uma adaptatividade em telas de diferentes tamanhos, apesar da aparente possível complicação com larguras de tela, o *Bootstrap* faz com que o componente se ajuste a largura da tela e faz aparecer uma barra rolante, deixando, o que se refere em CSS como *overflow*, o que transborda desse limite, com um *scroll*, que permite a visualização de todas as informações.

Conclusão

Este trabalho se objetivou inicialmente em uma revisão da literatura sobre a adaptação de interfaces e o uso de FeFs, tomando-se como base o *Bootstrap* para a implementação de um aplicativo Web, este também trouxe como objetivo contribuir para com determinada continuidade dos estudos apresentados por Cândido (2017) e Bueno (2016).

Neste sentido, este trabalho apresentou um estudo exploratório acerca da adaptação de interfaces em FeF explicando o funcionamento de UIDP's e as perspectivas UI e UX. Este apresentou-se como continuação de trabalhos anteriores, realizados por Bueno; Zaina (2016), com foco na capacidade adaptativa do FeF *Bootstrap*, e Cândido (2017), que analisou 3 FeF, *Zurb Foundation*, *Materialize CSS* e *Bootstrap*, e suas respectivas capacidades adaptativas, utilizando da classificação de Toxboe (2016), que resultou em 6 UIDPs para análise de suas características e possíveis problemas de interação. Destes, 2 foram selecionados para o presente trabalho, utilizando da pertinência para usabilidade, contexto necessário para implementação e necessidade destes para a interface, para utilização em uma proposta de análise de implementação e utilização do FeF *Bootstrap* em um aplicativo Web a partir das perspectivas UI e UX por parte do desenvolvedor.

A análise utilizou da busca por correspondências na documentação, observação do comportamento na interface, bem como a demonstração da implementação e análise comparativa/descritiva até os resultados obtidos dos UIDPs para uma observação sucinta, que se atentasse de forma concisa às perspectivas UI e UX com o intuito de esclarecer e nortear a escolha e estudo do desenvolvedor ao utilizar o *Bootstrap*. Todavia, é importante notar que a análise se limita por conta do público alvo, neste caso o próprio autor e as experiências acerca da área de estudos. Também é necessário levar em conta o escopo da pesquisa, que se limitou a dois UIDPs, visto que eram inerentes ao aplicativo utilizado. Portanto, não é possível a generalização das observações aqui realizadas pelo autor bem como a consolidação destas para os outros componentes do *Bootstrap*.

O trabalho corroborou para as pesquisas anteriores de forma a rever observações já apontadas por Cândido (2017) e demonstrar características exploradas por Bueno; Zaina (2016), trazendo uma análise UI e UX não utilizada pelos trabalhos anteriores. Para trabalhos futuros torna-se necessário realizar análises acerca do que já fora experimentado aqui da

perspectiva do usuário final da aplicação, bem como uma análise dos outros FeF já demonstrados anteriormente ou até mesmo a adição de novos FeF emergentes desde seu desenvolvimento.

Os resultados obtidos neste trabalho, possibilitaram parcerias com outros grupos de pesquisa para o desenvolvimento de aplicativos Web em desenvolvimento, com especial atenção a modelagem, simulação e visualização de dispositivos e efeitos fotônicos voltados às telecomunicações.

Referências

AGHAEI, S.; NEMATBAKHSI, M. A.; FARSANI, H. K. **Evolution of the world wideweb: From web 1.0 to web 4.0**. International Journal of Web & Semantic Technology, Academy & Industry Research Collaboration Center (AIRCC), v. 3, n. 1, p. 1, 2012.

BUENO, D. C. **HyMobWeb: uma abordagem para a adaptação híbrida de interfaces Web móveis sensíveis ao contexto e com suporte à multimodalidade**. 2018. 173 f. Dissertação (Mestrado) - Curso de Programa de Pós-graduação em Ciência da Computação, Ciências Exatas e da Terra:ciência da Computação, Universidade Federal de São Carlos, Sorocaba, 2017. Disponível em: <<https://repositorio.ufscar.br/handle/ufscar/9157>>. Acesso em: 13 dez. 2018.

BUENO, D. C.; ZAINA, L. M. **A comparative exploratory study on mobile web adaptation in front-end frameworks from the perspective of end-users**. In: ACM. Proceedings of the 15th Brazilian Symposium on Human Factors in Computer Systems. [S.l.], 2016. p. 30.

CÂNDIDO, P. H. V. **Problemas de interação na adaptação de interface Web móvel em Frameworks Front-end**. Itapetininga, 2017. 51 f. Trabalho Final de Curso Técnico (Técnico em Manutenção e Suporte em Informática) -- Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Câmpus Itapetininga, Itapetininga, 2017. Disponível em: <<https://drive.ifsp.edu.br/s/CoQpCEY3zC00jw6>>. Acesso em: 19 fev. 2018.

CARNEIRO, C. M. P. d. S. et al. **Frameworks de aplicações orientadas a objetos: uma abordagem interativa e incremental.** Universidade Salvador, 2003

CHAN, I. **O que é front-end e back-end** Programaria, 2016. Disponível em: <<https://www.programaria.org/o-que-e-front-end-e-back-end/>>. Acesso em: 13 de Dezembro de 2018.

FIGUEIREDO, C. F. P. **A sensibilidade ao contexto na utilização de aplicações móveis.** Dissertação (Mestrado) — Universidade de Aveiro, 2011

FOUNDATION, Interaction Design . Fundação. **Padrões de Design da Interface do Usuário (UI): Sua definição constantemente atualizada de Padrões de Design da Interface do Usuário (UI) e coleta de conteúdo e literatura sobre tópicos.** Disponível em: <<https://www.interaction-design.org/literature/topics/ui-design-patterns>>. Acesso em: 14 dez. 2018.

GONZALEZ, Guilherme; AELA.IO. **Qual a diferença entre UX e UI?** 2017. Medium Corporation. Disponível em: <<https://medium.com/aela/qual-a-diferen%C3%A7a-entre-ux-e-ui-1ea8a4b4f747>>. Acesso em: 13 dez. 2018.

GULLÀ, Francesca et al. **Method to design adaptable and adaptive user interfaces.** In: International Conference on Human-Computer Interaction. Springer, Cham, 2015. p. 19-24.
INTERNETLIVESTATS. **InternetLiveStats.** 2019. Disponível em:<<http://www.internetlivestats.com/>>. Acesso em: 04 de Julho de 2019

JAIN, N. **Review of different responsive css front-end frameworks.** Journal of Global Research in Computer Science, v. 5, n. 11, p. 5–10, 2015.

JAKOBSON, Gabriel; REUBEN, Steven. **Presentation of document history in a web browsing application.** U.S. Patent n. 9,135,228, 15 set. 2015.

KALUZ, Martin et al. A flexible and configurable architecture for automatic control remote laboratories. **IEEE Transactions on Learning Technologies**, v. 8, n. 3, p. 299-310, 2015.

MALDONADO, J. C. et al. **Padrões e frameworks de software**. Notas Didáticas, Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo, ICMC/USP, São Paulo, SP, Brasil, 2002.

MATIOLA, Willian. **O que é UI Design e UX Design?** 2015. Design Culture, Um projeto UNCOD PRO™. Disponível em: <<https://designculture.com.br/o-que-e-ui-design-e-ux-design>>. Acesso em: 13 dez. 2018.

MENDOZA, Abner; GU, Guofei. Mobile application web api reconnaissance: Web-to-mobile inconsistencies & vulnerabilities. In: **2018 IEEE Symposium on Security and Privacy (SP)**. IEEE, 2018. p. 756-769.

OTTO, M.; THORNTON, J. **Bootstrap is the most popular HTML, CSS, and JS framework for developing responsive, mobile first projects on the web**. 2017. Disponível em: <<https://getbootstrap.com/docs/3.3/>>. Acesso em: 13 de Dezembro de 2018.

PAGANI, T. **Quando utilizar (ou não) frameworks CSS**. 2011. Disponível em: <<https://tableless.com.br/quando-utilizar-ou-nao-frameworks-css/>>. Acesso em: 13 dez 2018.

ROSA, D.; SILVA, T. L.; **Adaptação de interfaces para dispositivos móveis com HTML5**. Anais do Eati - Encontro Anual de Tecnologia da Informação e Semana Acadêmica de Tecnologia da Informação, Frederico Westphalen - Rs, v. 1, n. 3, p.249-252, nov. 2013. Anual. Universidade Federal de Santa Maria (UFSM). Disponível em: <<http://www.eati.info/eati/2013/assets/anais/artigo249.pdf>>. Acesso em: 13 dez. 2018.

SOUZA, R. R.; ALVARENGA, L. **A web semântica e suas contribuições para a ciência da informação**. *Ciência da Informação*, SciELO Brasil, v. 33, n. 1, 2004. Disponível em: <<http://courses.iicm.tugraz.at/iaweb/surveys/ws2011/g3-survey-resp-web-design.pdf>>. Acesso em: 17 de Agosto de 2017

TOXBOE, A. **UI-Patterns - User Interface Design Patterns Library**. 2016. Disponível em:

<<http://ui-patterns.com/>>. Acesso em: 14 de Dezembro de 2018.

UDACITY, Brazil (Comp.). **UX e UI: quais são as maiores diferenças entre esses campos?** 2018.

Universidade do Vale do Silício. Disponível em: <<https://br.udacity.com/blog/post/ux-ui-diferencas>>.

Acesso em: 13 dez. 2018.

WANG, A. et al. **Um framework front-end moderno e responsivo baseado em Material Design**.

2017. Disponível em: <<http://materializecss.com/>>. Acesso em: 14 de Dezembro de 2018.