

PWA-EU: uma abordagem para o desenvolvimento de aplicações PWA baseadas em EUD

Giulia de Andrade Cardieri
Universidade Federal de São Carlos
Sorocaba-SP
giulia.cardieri@gmail.com

Luciana Martinez Zaina
Universidade Federal de São Carlos
Sorocaba-SP
lzaina@ufscar.br

ABSTRACT

Interfaces adaptativas propõem a mudança de seu comportamento de acordo com determinadas características de uso, como preferências do usuário e contextos de uso. *Progressive Web App* (PWA) é uma nova técnica de desenvolvimento de aplicações móveis proposta pela Google, que mistura tecnologia de recursos de desenvolvimento *web* e nativo. Essas aplicações são inicialmente apresentadas como *web* e progressivamente, com o crescimento da relação do usuário e a aplicação, tornam-se mais completas e parecidas com aplicações nativas. Contudo, essa abordagem não dá a liberdade para o usuário atuar como um agente ativo e participar de escolhas na adaptação da interface durante a interação. Este projeto de mestrado tem como objetivo propor a abordagem PWA-EU, uma extensão da arquitetura de uma aplicação PWA, considerando a perspectiva do usuário final utilizando *meta-design*, uma abordagem de *End-user development*. Como contribuição deste trabalho, pode-se destacar a inclusão do usuário como agente ativo no processo de adaptação e explorar esse novo modelo de desenvolvimento que são as PWAs.

ACM Classification Keywords

H.5.2 Information Interfaces and Presentation: User Interfaces

Author Keywords

mobile devices; progressive web app; adaptative interfaces; meta-design; end-user development

INTRODUÇÃO

Um dos desafios atuais na área de tecnologia é desenvolver interfaces de aplicações interativas para diferentes dispositivos. Aplicações *web* são populares, sendo que em 2016 a quantidade de usuários dessas aplicações cresceu 82% em relação a 2014. Enquanto o de aplicações nativas e híbridas cresceu 45% [6]. Além disso, cada navegador e dispositivo possuem suas próprias limitações que devem ser consideradas pelos desenvolvedores e designers de uma aplicação. Essa grande variedade de formas para acessar uma aplicação traz a necessidade de adaptar interfaces com as preferências dos usuários e

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Copyright 2018 SBC.

IHC 2018, Anais Estendidos do XVII Simpósio Brasileiro sobre Fatores Humanos em Sistemas Computacionais
Outubro 22-26, 2018, Belém, Brasil
WTD-IHC

diferentes dispositivos, com variados tamanhos, dimensões e contextos de uso. A partir disso, vários trabalhos vem sendo propostos [1, 4, 5, 9, 14, 15, 16, 18, 19].

Uma das propostas mais utilizadas e conhecidas é a do *Web Design Responsivo* (RWD - *Responsive Web Design*). Essa abordagem parte do princípio que o desenvolvimento e design de uma aplicação *web* deve se adaptar ao comportamento do usuário e do ambiente. Essa adaptação é feita de forma automática a partir da largura da tela, orientação e plataforma do dispositivo, utilizando *media queries* de folhas de estilo em cascata (CSS - *Cascading Style Sheets*) [3]. Porém, existem lacunas, pois fatores como multimodalidade e preferências do usuário ainda não são tratadas nessa proposta [9].

Progressive Web App (PWA) é uma nova abordagem, proposta pela Google, para desenvolver aplicações e mistura recursos da tecnologia de aplicações *web* e nativa. Essas aplicações são inicialmente apresentadas como uma aplicação *web* utilizando RWD, porém, progressivamente, considerando as interações contínuas do usuário, tornam-se mais completas e parecidas com aplicações nativas [17]. Os PWAs são uma tecnologia recente e não há grande quantidade de estudos científicos sobre a experiência dos usuários utilizando essa nova forma de desenvolver. Empresas como Forbes, Lancôme, OLX e Twitter já migraram suas aplicações *web* para PWA [10].

End-user development (EUD) tem foco em permitir que usuários não primariamente interessados em *softwares* possam criar, modificar ou estender um sistema de acordo com suas necessidades. O *meta-design* é uma abordagem da área de EUD onde o usuário final atua ativamente no processo de criação de um sistema. Seu principal objetivo é criar um ambiente para empoderar usuários a participar ativamente no desenvolvimento de aplicações, sem ficar restrito ao seu uso [8]. Nesse cenário, as preferências e necessidades dos usuários são atendidas com a modificação e evolução do sistema criado através da intervenção do próprio usuário [19].

Este projeto de mestrado tem como objetivo propor a abordagem PWA-EU, uma extensão da arquitetura PWA, incluindo o *meta-design* para permitir que as preferências e necessidades dos usuários sejam consideradas na interface. A criação da abordagem PWA-EU foi motivada pela pequena quantidade de estudos sobre o uso de aplicações PWA. Acredita-se que os resultados desse projeto poderão contribuir para a inclusão de usuários como agentes ativos no processo de adaptação, questão pouco explorada na área de RWD e PWAs.

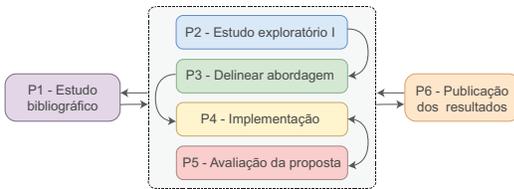


Figura 1. Visão geral do ciclo de pesquisa

Trabalho	Formas de adaptação	Plataforma
UIFlex [19]	MM, SC, PU	Web
HyMobWeb [5]	MM, SC	Web mobile
Bueno [4]	MM, SC	Web Mobile
Tukuchiy [1]	SC, PU	Desktop
M3I [15]	MM, SC	Mobile
Ghiani [9]	MM, SC	Web mobile
W3Touch [16]	Não há	Web mobile
Manca [14]	MM, SC	Web mobile
Chameleon [18]	SC, PU	Mobile
PWA-EU	PU	Web Mobile, Mobile

Tabela 1. Comparação das características dos trabalhos relacionados

A abordagem PWA permite que usuários naveguem de forma mais rápida, confiável e com uma experiência de usuário mais imersiva em relação a aplicações com RWD [12]. Ao incluir EUD através de *meta-design* na arquitetura dessas aplicações, espera-se torná-las mais flexíveis na perspectiva do usuário. Essa flexibilidade permite que os usuários acessem opções de aparência, formas de interação e exibição de conteúdo, atendendo necessidades não identificadas previamente por designers e desenvolvedores.

O estudo de caso I (ECI) apresentou uma análise qualitativa sobre a experiência do usuário em aplicações PWA, *web mobile* e nativa Android, analisando duas perspectivas. No ponto de vista do usuário, concluiu-se que uma boa experiência foi vivenciada mesmo com dificuldades relacionadas a elementos de interface. Já na visão do especialista em IHC, identificou-se problemas comuns na interação dos usuários com elementos de interface. Em geral, pode-se concluir que uma experiência de usuário positiva foi obtida nas distintas aplicações, mesmo com problemas de interação. Como trabalhos futuros, pretende-se dar continuidade a análise da experiência dos usuários finais em aplicações PWA através do estudo de caso II (ECII). Analisando os benefícios da participação ativa do usuário no *design* da interface.

TRABALHOS RELACIONADOS

Na linha de adaptação de interfaces destacam-se os trabalhos [1, 4, 5, 9, 14, 15, 16, 18]. Os trabalhos apresentados na Tabela 1 propõem diferentes formas para adaptar a interface, considerando fatores como multimodalidade (MM), sensibilidade aos contextos de uso (SC) e preferências do usuário (PU) em diferentes plataformas como *web*, *web mobile* e *mobile*.

UIFlex [19] é uma ferramenta para adaptação da interface de *websites* de acordo com o perfil do usuário utilizando princípios de *meta-design*. A adaptação é realizada em duas etapas, a primeira modela as interações do usuário através da criação de um perfil de interação, criado a partir de um formulário sobre suas preferências, e sugestões de autoridades (W3C,

Mozilla Developer Network). A segunda etapa adapta as interfaces injetando códigos *JavaScript* (JS), CSS e HTML nas páginas visitadas, a partir do perfil de interação do usuário.

HyMobWeb [5] é uma abordagem híbrida na perspectiva de desenvolvedores para adaptação de interfaces *web* em dispositivos móveis, baseadas em sensibilidade ao contexto e multimodalidade. A adaptação é realizada através da extensão de *frameworks front-end*, adicionando pontos de adaptação, utilizando uma *Domain Specific Language* desenvolvida pelos autores no código da interface. Uma estratégia híbrida foi adotada para combinar adaptações estáticas, quando as alterações são realizadas durante a implementação, e dinâmicas, onde as modificações ocorrem em tempo de execução.

Apesar de apresentar diversas formas de solucionar problemas na área de adaptação de interfaces, os trabalhos citados acima têm limitações. A maioria dos autores propõe técnicas, *frameworks* e algoritmos inteligentes para gerar interfaces adaptadas automaticamente [1, 4, 5, 9, 14, 15, 16, 18]. Já as propostas [4, 5, 9, 16, 14] consideram apenas aspectos tecnológicos e dimensões de contexto de uso em suas adaptações. Em alguns casos as preferências do usuário são consideradas como uma das bases do processo de adaptação [1, 18]. Contudo, não concedem liberdade para que o usuário atue como um agente ativo realizando suas escolhas. A proposta UIFlex [19] considera algumas preferências dos usuários e permite sua participação ativa para alterar adaptações realizadas automaticamente pelo sistema. Porém, UIFlex está limitado a um ambiente *web*, sem permitir seu uso em dispositivos móveis. Além disso, UIFlex não permite que o usuário altere a aparência de elementos de interface, forma de exibição de conteúdo e o uso de modalidades. Já M3I [15] é um *framework* que auxilia desenvolvedores a adicionar diferentes modalidades e contextos de uso durante a fase de desenvolvimento. A perspectiva do usuário final não é considerada pelo *framework*, contudo, uma de suas demonstrações de uso permite a participação do usuário final como co-designer da aplicação.

Uma lacuna ainda não explorada é um mecanismo de adaptação de interfaces com conceitos de EUD através de *meta-design*. Nessa proposta o usuário participaria ativamente das modificações realizadas, como a alteração da aparência de elementos de interface, formato de exibição do conteúdo e formas de interação. Uma questão pouco explorada em plataformas *web mobile*, especialmente em PWAs.

METODOLOGIA

A metodologia proposta para alcançar o objetivo deste projeto inclui um ciclo dividido em seis passos, ilustrados pela Figura 1 e detalhados abaixo.

P1 - Estudo bibliográfico: revisão de trabalhos dos principais temas envolvidos neste projeto, procurando encontrar tendências, limitações e novas descobertas nesse campo.

P2 - Estudo exploratório I: planejamento, execução e análise do estudo de caso I (ECI), relacionado a aplicações PWA.

P3 - Delinear abordagem: elaboração da abordagem proposta a partir dos resultados da análise do estudo exploratório I e revisão bibliográfica.

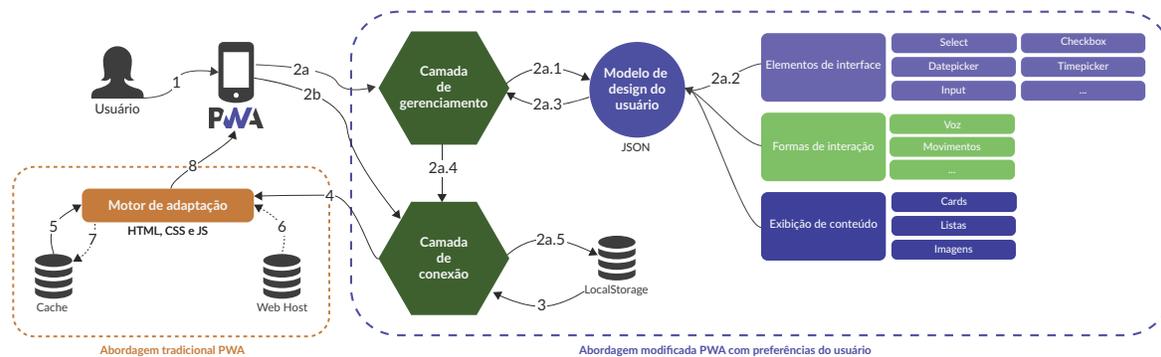


Figura 2. PWA-EU - abordagem proposta

P4 - Implementação: planejamento e desenvolvimento de prova de conceito.

P5 - Avaliação da proposta: planejamento, execução e análise do estudo de caso II (ECII).

P6 - Publicação dos resultados: escrita de publicações do resultado final deste projeto.

ABORDAGEM PWA-EU

O método tradicional de desenvolvimento de um PWA tem como base a divisão do conteúdo entre uma arquitetura de *shell* e o conteúdo dinâmico. Essa arquitetura consiste em carregar somente o esqueleto da interface do usuário da aplicação, composta por arquivos HTML, CSS e JavaScript (JS) e conteúdo estático como imagens e fontes. Esse esqueleto é armazenado em *cache* no navegador do usuário e recuperado toda vez que o PWA é acessado através de *Service Workers* (SWs). SWs são *scripts* em JS executados pelo navegador em segundo plano com um ciclo de vida completamente separado da página *web*. Essa tecnologia é necessária para o desenvolvimento de PWAs e permite o armazenamento de arquivos no navegador local utilizando *cache*, garantindo acesso aos documentos quando não há conexão com a internet [11]. Com isso, o usuário não precisa baixar novas versões, pois a atual será automaticamente baixada do servidor web quando houver acesso à internet. Ao obter uma nova versão, os SWs armazenam em *cache* e apagam a versão anterior.

Na Figura 2 a abordagem PWA-EU é apresentada. Ela adapta a arquitetura de desenvolvimento tradicional de um PWA para armazenar as preferências do usuário localmente no navegador, seguindo o conceito de EUD e *meta-design*. As modificações ocorrem após interações do usuário com a aplicação PWA (1). A seguir, dois caminhos podem ser tomados:

(i) Se o usuário alterou suas preferências em relação a elementos de interface, formas de interação e exibição de conteúdo, a camada de gerenciamento é chamada (2a). Essa camada é criada com códigos JS e é responsável por receber dois tipos de requisições da aplicação PWA: (i) receber e enviar modificações no modelo de design do usuário e (ii) enviar o conteúdo do novo modelo de usuário para a camada de conexão. O modelo de design do usuário tem o formato *JavaScript Object Notation* (JSON) e define e combina as preferências de design do usuário, incluindo aparência de elementos de interface, formas de interação, exibição de conteúdo como dados e imagens. Esse modelo permite salvar esses dados para acessos futuros, seguindo o conceito de *meta-design* e EUD. O *Meta-design*

proporciona que usuários atuem como *co-designers* da aplicação, atendendo um número maior de suas necessidades para viabilizar uma experiência mais agradável. A seguir, a camada de gerenciamento envia uma modificação ao modelo de design do usuário (2a.1). Elementos de interação específicos, formato de exibição de conteúdo e formas de interação são enviados ao modelo de design do usuário (2a.2). O JSON com as preferências do usuário é recebido pela camada de gerenciamento (2a.3) e enviado a camada de conexão (2a.4). A camada de conexão são códigos em JS que armazenam localmente no navegador o JSON definido no modelo de design do usuário. Para isso, a propriedade *localStorage* do HTML5 é utilizada. Além disso, essa camada também envia esse JSON para o motor de adaptação. O passo (2a.5) ocorre quando essa camada armazena o JSON utilizando *localStorage*.

(ii) No segundo caso, o usuário não alterou suas preferências e a aplicação chama diretamente a camada de conexão (2b). Para ambos os casos, a camada de conexão recebe o JSON a partir do *localStorage* (3) e o envia para o motor de adaptação (4). O motor de adaptação utiliza JS e CSS para modificar o esqueleto da aplicação, criado com HTML, CSS e JS, considerando as preferências do usuário definidas no JSON do modelo de design do usuário. O motor também é responsável por obter um novo esqueleto da aplicação do servidor web, quando há conexão com a internet, armazenar o esqueleto em *cache* e recuperá-lo. Os passos 5, 6, 7 e 8 fazem parte da abordagem tradicional de desenvolvimento de uma aplicação PWA, descrita no início desta seção.

AVALIAÇÃO

Uma avaliação da perspectiva da arquitetura de software da abordagem proposta não será realizada. Somente uma prova de conceito será avaliada, pois esta abordagem está inserida no contexto de IHC com foco em EUD e *meta-design*.

O estudo de caso I (ECI) foi realizado, seguindo as diretrizes estabelecidas por Lazar et. al [13] e teve como objetivo analisar aspectos de experiência do usuário com diferentes elementos de interface em três plataformas: PWA, *web mobile* e nativa Android. A partir dos dados coletados, foi realizada uma análise qualitativa com base em duas diferentes perspectivas. Inicialmente, o *feedback* dos participantes foi explorado na perspectiva do usuário, através do método *Self-Assessment Manikin* (SAM) [2]. A segunda perspectiva selecionada foi a de um especialista em interação humano-computador (IHC), onde o especialista analisou as expressões faciais dos participantes

através das 10 heurísticas de emoção [7] com a finalidade de identificar as emoções vivenciadas pelos usuários e seus gatilhos, encontrados através da técnica Open Coding [20]. O artigo "*Analyzing user experience in mobile web, native and progressive web applications: a user and HCI specialist perspectives*", será publicado no IHC 2018 e detalha o ECI.

O estudo de caso II (ECII) está na fase de planejamento e será executado com o objetivo de avaliar a proposta desenvolvida sob a perspectiva do usuário. Uma aplicação PWA será desenvolvida baseando-se na abordagem PWA-EU com objetivo de permitir a participação ativa de usuários nas escolhas de design durante suas interações. Uma comparação será realizada entre dois grupos de participantes, os primeiros não atuarão como *co-designers*, e o segundo grupo poderá alterar características de aparência e formas de interação na aplicação. Essa avaliação será realizada de forma qualitativa, utilizando métricas similares ao ECI. Contudo, espera-se analisar não só as interações com elementos de interface, mas também suas interações com diferentes formatos de exibição de conteúdo e formas de interação, como ditação por voz, reconhecimento de voz e movimentos como *swipe*. Além disso, será feita uma análise sobre os efeitos da participação do usuário como *co-designer* na aplicação.

CONSIDERAÇÕES FINAIS

Este projeto de mestrado apresenta a abordagem PWA-EU, uma adaptação da arquitetura de aplicações PWA para incluir conceitos de *meta-design*, uma abordagem de EUD. Na PWA-EU, usuários participam ativamente das escolhas de *design* da aplicação durante sua execução.

AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

REFERÊNCIAS

1. Luísa Barrera-Leon, Nadia Mejia-Molina, Angela Carrillo-Ramos, Leonardo Flórez-Valencia, and Jaime A. Pavlich-Mariscal. 2016. Tukuchiy: A dynamic user interface generator to improve usability. *International Journal of Web Information Systems* 12, 2 (2016), 150–176.
2. M Bradley and Peter J Lang. 1994. Measuring Emotion: The Self-Assessment Manikin and the Semantic Differential. *Journal of Behavior Therapy and Experimental Psychiatry* 25, 1 (1994), 49–59.
3. Jay Bryant and Mike Jones. 2012. Responsive Web Design. In *Pro HTML5 Performance*. Vol. 67. Apress, Berkeley, CA.
4. Danilo C. Bueno and Luciana M. Zaina. 2016. Um estudo exploratório comparativo sobre adaptação Web móvel em frameworks front-end na perspectiva dos usuários finais. In *IHC '16*. São Paulo, Brazil.
5. Danilo C. Bueno and Luciana M. Zaina. 2018. HyMobWeb : A hybrid adaptation of context-sensitive Web interfaces with multimodality support in mobile devices. *SBC Journal on Interactive Systems* (2018). (In press).
6. Comscore. 2016. The 2016 U.S. Mobile App Report. (2016). <https://goo.gl/Fi9UjT> [Acessado: 14 de Setembro de 2017].
7. Eva de Lera and Muriel Garreta-Domingo. 2007. Ten Emotion Heuristics: Guidelines for assessing the user's affective dimension easily and cost-effectively. In *BCS-HCI '07*, Vol. 2. 7–10.
8. Gerhard Fischer. 2009. End-user development and meta-design: Foundations for cultures of participation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 5435 LNCS (2009), 3–14.
9. Giuseppe Ghiani, Marco Manca, Fabio Paternò, and Claudio Porta. 2014. Beyond responsive design: Context-dependent multimodal augmentation of web applications. In *MobiWIS '14*, Vol. 8640 LNCS. 71–85.
10. Google Developers. 2015. All showcase tagged: progressive-web-apps. (2015). <https://goo.gl/6RkdEd> [Acessado: 1 de Fevereiro de 2018].
11. Google Developers. 2018a. Introduction to Progressive Web App Architectures. (2018). <https://goo.gl/ARBx1U> [Acessado: 17 de Abril de 2018].
12. Google Developers. 2018b. Progressive Web Apps. (2018). <https://goo.gl/m2155N> [Acessado: 10 de Maio de 2018].
13. Jonathan Lazar, Jinjuan H. Feng, and Harry Hochheiser. 2010. *Research Methods in Human-Computer Interaction*. Wiley Publishing.
14. Marco Manca, Fabio Paternò, Carmen Santoro, and Lucio Davide Spano. 2013. Generation of multi-device adaptive multimodal web applications. In *MobiWis '13*, Vol. 8093 LNCS. 218–232.
15. Andreas Möller, Stefan Diewald, Luis Roalter, and Matthias Kranz. 2014. Supporting Mobile Multimodal Interaction with a Rule-Based Framework. In *Mensch und Computer 2014*.
16. Michael Nebeling, Maximilian Speicher, and Moira Norrie. 2013. W3touch: Metrics-based Web Page Adaptation for Touch. In *CHI'2013*. Paris, France, 2311.
17. Petele. 2016. Your First Progressive Web App. (2016). <https://goo.gl/52y6qo> [Acessado: 26 de Setembro de 2017].
18. Paulo Pombinho, Ana P. Afonso, and Maria Beatriz Carmo. 2011. Chameleon - A context adaptive visualization framework for a mobile environment. In *IV 2011*. 151–157.
19. Mailson Q. Proença and Vânia P. A. Neris. 2017. UIFlex : a meta-design solution for the web. In *IHC '17*, Vol. 2. 461–470.
20. A. Strauss and J. M. Corbin. (1998). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory* (2 ed.). Vol. 4. Thousand Oaks.