

APONTAMENTO A CERCA DO PROGRESSIVE WEB APPS

Patrícia Esteves TRINDADE, (UNESP)¹

Letícia Passos AFFINI (UNESP)²

Resumo: O presente trabalho tem o objetivo de apresentar e discutir conceitos acerca da abordagem *Progressive Web Apps* (PWA), utilizada no desenvolvimento de aplicativos para dispositivos móveis. A tecnologia PWA surgiu em 2015, com apenas alguns recursos e está em constante aprimoramento. Utilizou-se como metodologia a pesquisa bibliográfica, tendo como fonte mais expressiva o *site* da *Google Developers*. Entende-se PWA como uma tecnologia que agrega os recursos disponíveis nos aplicativos nativos com os recursos da *web*, melhorando a experiência do usuário e reduzindo a utilização do hardware. Observou-se, a partir de coleta de dados no *site* da Nomura Tower, queda no número de *downloads* de aplicativos nas lojas *Apple* e *Play Store*, como também no investimento massivo da *Google* e *Microsoft* no desenvolvimento de recursos PWA; a partir destas informações, infere-se que o recurso PWA pode ser considerado uma tecnologia expressiva e emergente para o mercado.

Palavras-chave: *Progressive web apps*; Inovação; Tecnologia.

Abstract: This paper aims to present and discuss concepts about the Progressive Web Apps (PWA) approach used in the development of mobile applications. PWA technology emerged in 2015 with only a few features and is constantly improving. The bibliographical research methodology was used as a methodology, having as the most expressive source the site of Google Developers. PWA is understood as a technology that aggregates the features available in native applications with the web resources, improving the user experience and reducing the use of hardware. Data collection on the Nomura Tower site saw a drop in the number of app downloads in the Apple and Play Store stores, as well as the massive investment by Google and Microsoft in the development of PWA resources; from this information, it is inferred that the PWA resource can be considered an expressive and emerging technology for the market.

Keywords: *Progressive web apps*; Innovation; Technology.

INTRODUÇÃO

Atualmente, no Brasil há cerca de cento e vinte milhões de usuários de Internet, e grande parte dos acessos é realizada por *smartphones* (CETIC.BR, 2018). O avanço na tecnologia móvel fez com que os *smartphones* obtivessem prioridade entre os usuários, pois os mesmos proporcionam troca de mensagens, possibilitam assistir a

¹ Mestranda no Programa de Pós-Graduação em Mídia e Tecnologia da Universidade Estadual Paulista “Júlio de Mesquita Filho” (UNESP), FAAC, Bauru, integrante do GrAAu – Grupo de Análise Audiovisual. pe.trindade@unesp.br ORCID 0000-0002-1218-1625

² Professora do Programa de Pós-graduação em Mídia e Tecnologia da Universidade Estadual Paulista “Júlio de Mesquita Filho” (UNESP), FAAC, Bauru. Líder do Grupo de Análise Audiovisual - GrAAu, cadastrado no CNPq. leticia.affini@unesp.br. ORCID 0000-0003-4688-289X

vídeos, programas, séries e conversar por chamadas de voz ou vídeo. Essa preferência como meio de comunicação e interatividade foi impulsionada pela utilização de aplicativos, que ajudam os usuários em atividades corriqueiras do dia a dia, de forma rápida e fácil. Pode-se considerar que as empresas do ramo da telefonia móvel têm buscado atualizar e oferecer aos seus usuários comodidade na utilização dos serviços disponibilizados, a partir de qualquer local, por meio de *smartphones* ou outro tipo de dispositivo móvel com conexão à internet.

Os aplicativos móveis (*Apps*) são um *software* projetado para ser executado especificamente em um dispositivo móvel, como um *smartphone* ou *tablet*. Os *apps* podem ser instalados por meio de lojas *on-line* específicas, como é o caso da *App Store*, *Play Store*, *Microsoft Store*, entre outras diversas lojas. Há uma infinidade de fornecedores que disponibilizam aplicativos para *download* nas lojas virtuais, para todo tipo e gênero de aplicação. Pelo fato do mercado de dispositivos móveis ser ramificado, há uma gama de plataformas de desenvolvimento de sistemas operacionais móveis, gerando uma variedade de aplicativos específicos para cada plataforma e que podem ser executados sob uma arquitetura específica. Entre os sistemas operacionais *mobile* mais utilizados, o que mais se destaca é o *Android*, da *Google*, que domina 89% (oitenta e nove por cento) do mercado brasileiro de *smartphones*, seguido pelo *IOS*, da *Apple*, com 7% (sete por cento) (MOBILETIME, 2018). Existem outras plataformas de sistemas operacionais, porém a porcentagem de utilização é tão irrisória que não serão abordadas no presente trabalho. Em resumo, isto significa que uma aplicação desenvolvida para o sistema *Android* não funciona nos sistemas da *Apple* ou do *Windows*, e vice e versa; para cada sistema operacional é necessário uma nova aplicação. Essa multiplicidade torna imprescindível a necessidade de se reescrever o mesmo aplicativo diversas vezes, codificado de maneira diferente, para que possa ser executado sob a sua arquitetura específica. Isso dificulta a tarefa de desenvolver um aplicativo que atenda a uma grande parcela de usuários e, conseqüentemente, ao mercado, pois como cada sistema operacional tem a sua própria linguagem para desenvolvimento, a capacidade de portabilidade de código se torna baixa, aumentando o prazo e o custo para o desenvolvimento de um único aplicativo para diferentes plataformas [FERNANDES, 2017].

Os usuários de *smartphones* têm, como fator em comum, o desejo de estar sempre conectados aos assuntos de seu interesse e as aplicações móveis surgiram como uma inovação para facilitar a sua rotina. Ao atender os requisitos de conectividade e eficiência, esperados nesse tipo de experiência móvel, os aplicativos se tornaram uma das melhores maneiras para se estabelecer uma conexão entre usuários e suas demandas. Entretanto, muitos aplicativos são instalados e utilizados apenas uma vez, visto que, nos *sites* de busca, não há como consultar as ferramentas neles disponíveis, pois esses são encontrados somente em suas respectivas lojas *on-line*. Nas lojas da *Apple Store* e da *Play Store* há milhões de aplicativos, contudo o número de *downloads* de novos *apps* começa a desacelerar, concentrando-se, em sua maioria, nas principais redes sociais.

Tal reflexão é necessária, já que geralmente os usuários utilizam a internet para suprir as suas necessidades, como entretenimento, serviços, relacionamentos interpessoais, pesquisas educacionais, e normalmente o fazem por meio de um *smartphone*. Como muitas empresas divulgam o seu *app* com a exigência de sua instalação, vários usuários, que estavam interessados em apenas um serviço, deixam de utilizá-lo, por vários motivos, entre os quais se destacam a insuficiência de espaço de armazenamento, a limitação de banda larga de conexão de internet, além da burocracia do preenchimento de dados. Dessa forma, observa-se um número baixo de *downloads*, o que demonstra a dificuldade de instalação e manutenção de um aplicativo pelos usuários, pois grande parte dos *smartphones* é constituída por modelos simples, de baixo custo e com recursos escassos, o que faz com que somente os aplicativos funcionais e prioritários sejam mantidos.

Isso não significa o fim do desenvolvimento de aplicativos, muito menos a extinção da *App Store* ou da *Play Store*, mas uma melhor abordagem para o mundo *mobile*, na maioria dos casos, com a utilização de uma tecnologia chamada *Progressive Web App (PWA)*. Segundo a Wikipédia, *Progressive Web App (PWA)* é um termo usado para denotar aplicativos que usam as mais recentes tecnologias da *web*. Os aplicativos da *web* progressivos são páginas *web* (ou *sites*) tecnicamente regulares, mas podem aparecer para o usuário como aplicativos tradicionais, ou aplicativos móveis nativos. Esse novo tipo de aplicativo tenta combinar os recursos oferecidos pela maioria dos navegadores modernos com os benefícios da experiência móvel.

O *Progressive Web Apps* foi desenvolvido pelo time da *Google Developers*, com destaque para a *designer* Frances Berriman e o engenheiro do *Google Chrome* Alex Russell, em 2015. Segundo Russell:

These apps aren't packaged and deployed through stores, they're just websites that took all the right vitamins. They keep the web's ask-when-you-need-it permission model and add in new capabilities like being top-level in your task switcher, on your home screen, and in your notification tray. Users don't have to make a heavyweight choice up-front and don't implicitly sign up for something dangerous just by clicking on a link. Sites that want to send you notifications or be on your home screen have to *earn* that right over time as you use them more and more. They progressively become "apps" (RUSSEL, Alex. Infrequently Note. Disponível em: <<https://infrequently.org/2015/06/progressive-apps-escaping-tabs-without-losing-our-soul/>> Acesso em: 09 de agosto de 2018.

Os *PWAs* são definidos por um conjunto de conceitos e palavras-chave, incluindo: conectividade progressiva (para qualquer usuário, independente do browser); responsivo (elaborado para qualquer dispositivo *desktop* ou *mobile*); *app-like* semelhante a um aplicativo – (o usuário se sente em um aplicativo nativo); atual (não há necessidade de fazer atualizações, como está tudo na *web*, quando abrir o *app* a nova versão já será carregada); descobrível - *SEO Friendly*³ – (os mecanismos de busca conseguem encontrar o conteúdo dos aplicativos); reengajável (através de *push notifications*⁴, o usuário pode ser constantemente engajável); instalável (podem ser

³ É um *site* que respeita uma série de padrões considerados como "boas práticas" pelos **buscadores** – e, assim, ele se torna mais "amigável" a estes **mecanismos de busca**. Disponível em: <http://www.intermidias.com.br/seo-otimizacao-de-sites/seo-friendly/>, acessado em 06/10/2018.

⁴ As mensagens que são enviadas ao dispositivo do usuário, não limitando-se a computadores e dispositivos móveis, de maneira direta são chamadas de *push notification*. Disponível em: <https://blog.lahar.com.br/midias-sociais/o-que-e-push-notification/>, acessado em 06/10/2018.

adicionados à *home screen* do dispositivo móvel); e linkável (mais fácil de compartilhar conteúdo ao enviar o *link* para alguém).

Estas podem ser consideradas as contribuições do PWA para a unificação da experiência móvel, na qual os aplicativos da *web* podem ser instalados e distribuídos sem as lojas, experiências *off-line* avançadas, sincronizações periódicas em segundo plano. Os *service workers* oferecem a base técnica para todos esses recursos. O *service worker* é um *scrip*⁵ que o navegador executa em segundo plano, separado da página da *Web*, e possibilita recursos que não precisam de uma página da *Web* ou de interação do usuário.

METODOLOGIA

Observando esse cenário, este trabalho tem o objetivo de apresentar conceitos sobre a abordagem *Progressive Web Apps* para o desenvolvimento de aplicações para dispositivos móveis. O intuito é trazer apontamentos sobre a utilização dos recursos disponíveis que permitem criar aplicativos diferenciados, que proporcionem uma ótima experiência ao usuário final. A conceituação da abordagem *PWA* foi motivada pela pequena quantidade de estudos sobre o uso de aplicações *web* progressivas.

O presente estudo baseou-se em análise descritiva, com revisão bibliográfica, tendo como base de dados livros, artigos científicos e *sites* na área de tecnologia e informação como fonte de publicações eletrônicas, visto que o assunto abordado é recente e há carência de materiais já publicados; portanto, as documentações oficiais, disponibilizadas pelas equipes responsáveis pela ferramenta *Google Developers*, representam parcela considerável do material consultado.

FORMAS DE DESENVOLVIMENTO

⁵ É um texto com uma série de instruções para serem seguidas

Sabe-se que os *smartphones* estão conquistando um enorme espaço no mercado, e grande parte desse sucesso deve-se aos aplicativos *mobile*, que modificaram o uso dos aparelhos móveis e criaram novas possibilidades, explorando funções do *hardware* de forma inovadora. Nesse sentido, o mercado de aplicativos *mobile* não para de crescer e, com isso, várias formas de desenvolver aplicações móveis, por meio de aplicações nativas, híbridas e aplicações *web* progressiva.

APLICAÇÕES NATIVAS

Antes de falar sobre o objetivo principal deste artigo (*Progressive Web Apps*) é necessário conhecer as formas para a construção de aplicações móveis. Os aplicativos nativos são elaborados para uma determinada plataforma e o seu desenvolvimento é realizado em linguagens de programação específicas. São instalados diretamente no dispositivo e podem ser acessados através de ícones fixados na tela do dispositivo móvel; os mesmos são instalados através de uma loja de aplicativos e podem aproveitar todas as funcionalidades do sistema operacional do dispositivo, como: câmera, GPS, acelerômetro, bússola, lista de contatos e afins. Também é possível aproveitar o uso de gestos, sistemas de notificação nativos, como o *push notification*, e funcionar sem conexão com a internet, caso o conteúdo esteja embarcado.

As aplicações nativas são caracterizadas pela sua rapidez e por oferecer uma experiência de utilização otimizada, por possuir melhor acesso, integração ao *hardware* e à plataforma nativa do aparelho. Integram-se com funções avançadas e a outras *APPS*. Podem manipular o funcionamento do aparelho e até substituir ou complementar funções nativas. Como uma *app* nativa, precisa ser instalada e há a necessidade de disponibilizar o aplicativo em uma loja de fabricante, o que é um processo lento e burocrático. Se a aplicação for disponibilizada para várias plataformas, será necessário desenvolver *apps* distintas, pois não há reaproveitamento de código, o que encarece o projeto.

Destaca-se, também, o uso casual dos aplicativos; um *app* exige um compromisso, uma instalação e, muitas vezes, o usuário não precisa e nem está disposto

a isso, tem interesse somente em uma determinada informação num dado momento. *Apps* são desenvolvidas para usuários fiéis, no entanto, estudos mostram que a maioria das *apps* instaladas são abertas poucas vezes.

APLICAÇÕES HÍBRIDAS

O diferencial das aplicações híbridas se dá na forma como o código é desenvolvido para a aplicação móvel. Aplicações híbridas são executadas dentro de um ambiente de processo nativo, na plataforma do dispositivo, assim como os aplicativos nativos. Os aplicativos híbridos são desenvolvidos com tecnologia *web*, *HTML*, *CSS* e o *JavaScript*, que utilizam funcionalidades desses sistemas, chamado de *WebView*, para apresentar o código *web* como aplicação responsiva para qualquer plataforma, bem como é executado por um navegador em tela cheia, que não possui barra de endereços ou os demais controles, normalmente imbuídos no navegador.

Ao utilizarem a tecnologia *web* partilhada por várias plataformas, as aplicações híbridas são mais eficazes em termos de custo e têm a vantagem adicional de reutilizarem o mesmo código para diversas plataformas. Possui manutenção mais fácil e mais acessível em termos de tempo e custo, principalmente em casos de aplicações multiplataforma. Os aplicativos são distribuídos em lojas oficiais, e passam pelo processo burocrático que as aplicações nativas. A atualização para as novas versões pode ser demorada, pois há a necessidade de adaptação das novas tecnologias.

PROGRESSIVE WEB APPS (PWA)

Como o desenvolvimento de aplicativos e sites *web* está cada vez mais similar, vemos, muitas vezes, *sites* que lançam aplicativos com os mesmos serviços da plataforma *online* ou desenvolvem aplicativos *web* para atender usuários que os acessam por meio de dispositivos móveis. A evolução dos *smartphones*, que permite softwares cada vez mais avançados e aplicações com novos serviços e recursos,

transformando estes tipos de dispositivos em oportunidade de entretenimento, acesso à informação e solução de problemas, integrando-se, assim, ao cotidiano das pessoas e facilitando diversas tarefas do dia a dia. Contudo, esse mercado é disputado por diferentes plataformas de desenvolvimento, com várias soluções de mercado, necessitando, dessa maneira, de profissionais especializados nas mais diferentes linguagens de programação, com tempo e custo de desenvolvimento elevados; todos esses fatores são pontos problemáticos para o desenvolvimento de *apps*.

O desenvolvimento *Web* pode ser uma solução adequada para os problemas citados acima, já que possui plataforma única e portátil. As *Web App* são, basicamente, *sites* com *layout* e algumas funcionalidades feitas especificamente para dispositivos móveis, tornando a experiência do usuário parecida a de um aplicativo nativo (TAVARES, 2016). Com a evolução da *web* nos últimos anos, é possível utilizar alguns recursos de *hardware*, como a câmera do computador e tirar fotos por meio do *browser*. As APIs da *web* estão em constante evolução, permitindo desenvolver aplicações cada vez mais complexas somente com o uso de *HTML*, *CSS* e *JavaScript*. Todo esse avanço da *web* permite levar essas funcionalidades para a plataforma *mobile*, e é aqui que entram os PWAs.

Progressive Web App (PWA) é uma nova abordagem, proposta pela *Google*, para desenvolver aplicações que combinam recursos da tecnologia de aplicações *web* e nativa. São aplicações *web* que possuem comportamento e aparência semelhantes a um aplicativo nativo. São inicialmente apresentadas como uma aplicação *web*, porém, progressivamente e considerando as interações contínuas do usuário, tornam-se mais completas e parecidas com as aplicações nativas. Fornecem funcionalidades adicionais ao dispositivo em que é instalado, como: funcionar *offline*, notificações *push*, sistema de gerência de memória, carregamento mais rápido, visual e velocidade de *apps* nativos, entre outros. Possuem *layout* mais amigável que *sites* responsivos, podem ser confundidos com aplicativos nativos, pois é possível ter uma experiência em tela cheia sem necessitar a visibilidade da barra de endereços, fazendo com que o usuário final não consiga distinguir um PWA de um aplicativo nativo.

Tal abordagem permite que os usuários naveguem de forma rápida, confiável e imersiva, fatos que podem ser considerados como os três pilares de uma aplicação *web*

progressiva. Confiável, pois permite carregamento instantâneo, mesmo sem conexão, utiliza *Proxy* escrito em *JavaScript*, responsável por controlar o *cache* e as requisições do navegador e, ao pré-armazenar em *cache* os principais recursos, elimina-se a dependência da rede, garantindo uma experiência instantânea e confiável. Rápido, indo além do carregamento e interagindo, de maneira significativa, com as aplicações criadas, com respostas ágeis às interações do usuário. Imersiva/Atraente, com experiências muito próximas às dos aplicativos nativos, com interfaces mais atraentes, podendo ser exibidos em tela normal ou *full screen*, e engajamento do usuário por meio de notificações *push*. Os *PWAs* são instaláveis, sem a necessidade de download em uma loja de aplicativos, e pode-se inserir um ícone à tela do *smartphone* e a página será carregada quando o ícone for tocado.

CARACTERÍSTICAS DO *PWA*

Segundo o *Google*, um *PWA* deve possuir as seguintes características:

- **Progressivo:** funciona para todos os usuários, independentemente da escolha do navegador, porque é construído com aprimoramento progressivo, como um princípio central.
- **Responsivo:** compatível com qualquer formato: desktop, celular, *tablet* ou o que mais surgir.
- **Independente de conectividade:** aprimorado com *Service Worker*, para trabalhar *offline*, ou em redes de baixa qualidade.
- **Semelhante a aplicativos (App-like):** parece com aplicativos para os usuários, com interações e navegação de estilo de aplicativos, pois é compilado no modelo de *shell* de aplicativo.
- **Atual:** sempre atualizado, graças ao processo de atualização do *Service Worker*.
- **Seguro:** fornecido via HTTPS para evitar invasões e garantir que o conteúdo não seja adulterado.

- **Descobrível:** pode ser identificado como “aplicativo” graças aos manifestos W3C e ao escopo de registro do *Service Worker*, que permitem que os mecanismos de pesquisa os encontrem.

- **Reengajável:** facilita o reengajamento com recursos como notificações *push*.

- **Instalável:** permite que os usuários adicionem os aplicativos que consideram úteis em suas telas iniciais sem precisar acessar uma loja de aplicativos.

- **Linkável:** compartilhado facilmente, informando apenas a URL do *site*, e não requer instalação complexa.

REQUISITOS

Para se iniciar um projeto *web* utilizando os conceitos do PWA, são necessários três requisitos: Conexão HTTPS, *Web App Manifest* e *Service Worker*. Ressalta-se que, dependendo do suporte do navegador, os aplicativos podem nem sempre ser capazes de usar o *app manifest* ou o *service worker*. Entretanto, como um PWA deve melhorar progressivamente os aplicativos com essas tecnologias, os tais aplicativos ainda devem funcionar mesmo que um navegador não tenha o apoio deles. Com uma visão futurista, empresas como *Google*, *Microsoft* e *Apple* caminham em direção às aplicações progressivas da *web*, ou numa vertente onde as aplicações *web* serão de suma importância no seguimento *mobile*. Pesquisas apontam investimento massivo da *Microsoft* na tecnologia PWA, assim também como da *Google*, com suporte progressivo em todas as plataformas, no *Chrome* e no *iOS*, e, mais recentemente, o navegador padrão da *Apple*, o Safari, já divulgou alguns sinais sobre desenvolvimento e suporte ao PWA.

CONEXÃO HTTPS

HTTPS é a sigla em inglês de *Hyper Text Transfer Protocol Secure*, que em português significa “Protocolo de transferência de hipertexto seguro”; é uma implementação do protocolo HTTP sobre uma camada adicional de segurança e utiliza o

protocolo SSL/TLS. Essa camada adicional permite que os dados sejam transmitidos por meio de uma conexão criptografada e que se verifique a autenticidade do servidor e do cliente por meio de certificados digitais (WIKIPEDIA)⁶.

Para Kayce Basques, desenvolver da *Google*⁷, todos os *sites* devem ser protegidos pelo protocolo HTTPS, mesmo que não utilizem informações sigilosas, pois, além de fornecer segurança e integridade de dados, fundamentais para o *site* e as informações pessoais dos usuários, tal protocolo é uma exigência de muitos recursos dos navegadores, especialmente os que utilizam aplicações *web* progressivas. Basques destaca, ainda, que o HTTPS evita a adulteração da comunicação entre o *website* e os navegadores, impede a interceptação dos dados entre *sites* e usuários, pois acredita que toda informação desprotegida pode revelar comportamentos e identidades de usuários. Ações inovadoras e avançadas da *web*, como tirar fotos ou gravar áudio, permitem experiências *offline*, como as dos aplicativos com os *Service Workers* ou criar aplicativos progressivos, exigem permissão explícita do usuário antes de serem executadas e, com isso, muitas APIs antigas estão sendo atualizadas para exigir permissão para serem executadas, e o protocolo HTTPS é um componente essencial dos fluxos de trabalho de permissão, finaliza Basques.

WEB APP MANIFEST

O Manifesto do Aplicativo *Web*, conhecido como *Web App Manifest*, é um arquivo texto JSON simples, que fornece informações sobre o aplicativo (como nome, autor, ícone e descrição) e como ele deve se comportar, quando “instalado” no dispositivo móvel. O manifesto informa detalhes para *sites* instalados na tela inicial de um dispositivo, proporcionando aos usuários um acesso mais rápido e uma experiência

⁶ Disponível em: https://pt.wikipedia.org/wiki/Hyper_Text_Transfer_Protocol_Secure, acessado em 24 de outubro de 2018.

⁷ Disponível em: <https://developers.google.com/web/fundamentals/security/encrypt-in-transit/why-https>, acessado em 24 de outubro de 2018

mais rica. Com os arquivos de manifesto, pode-se especificar se um aplicativo da *web* deve ser aberto no modo de tela cheia, possuir uma cor personalizada na barra de endereços e até definir ícones que podem ser inseridos na tela inicial dos dispositivos. Os *web app manifest* fazem parte de uma coleção de tecnologias da *Progressive Web Apps* que permite a sua instalação, na tela inicial de um dispositivo, sem precisar estar em uma loja de aplicativos, além de recursos como trabalhar *offline* e receber notificações *push*.

Entende-se JSON – JavaScript Object Notation – como um modelo para armazenamento e transmissão de informações no formato texto. Apesar de muito simples, tem sido bastante utilizado por aplicações *Web* devido à sua capacidade de estruturar informações de uma forma mais compacta do que a conseguida pelo modelo XML⁸, tornando mais rápido a análise dessas informações. Dados representados em JSON podem ser acessados por qualquer linguagem de programação, por meio de APIs específicos. Isto explica o fato de empresas que possuem grandes volumes de dados para transmissão adotarem o JSON.

SERVICE WORKER

Um *Service Worker* é um *script* que o navegador executa em segundo plano, separado da página da *web*, possibilitando recursos que não precisam de uma página ou de interação do usuário. Atualmente, os *service workers* já incluem recursos como notificações *push* e sincronização em segundo plano. Um dos seus principais recursos é a capacidade de interceptar e tratar solicitações de rede, incluindo o gerenciamento programático de um *cache* de respostas. Essencialmente, comportam-se como um servidor *Proxy*, situado entre uma aplicação *web*, o navegador e a rede (quando esta estiver disponível), o que possibilita criar experiências *offline* eficientes, interceptar requisições de rede e atualizar os repositórios que residem no servidor. Portanto, sem um *service worker* é quase impossível oferecer uma boa experiência com o PWA.

⁸ XML – Extensible Markup Language – seu propósito principal é a facilidade de compartilhamento de informações por intermédio da internet. Disponível em: <https://pt.wikipedia.org/wiki/XML>, acessado em 25 de outubro de 2018

As notificações por *push*, da *web*, são oportunas, relevantes e precisas, consideradas primordiais pelos desenvolvedores e, agora, inseridas nas aplicações *web*. Elas permitem que os usuários aceitem atualizações e proporcionam conteúdos pertinentes, com o intuito de engajamento do usuário.

A sincronização em segundo plano, *background sync*, é uma nova API da *Web* que permite adiar ações até que o usuário tenha conectividade estável; isso garante que a solicitação realizada pelo usuário seja realmente enviada.

CONSIDERAÇÕES FINAIS

Com o crescimento do mercado de *smartphones*, observa-se uma tendência e crescimento no desenvolvimento de aplicações *mobile*. Há uma inquietação nas empresas que desenvolvem aplicativos para *smartphones* com redução de custo, tempo e complexidade. Com a evolução do desenvolvimento *web*, e novos recursos suportados pelos navegadores, os *Progressive Web Apps* estão ganhando espaço no mercado de aplicativos *mobile* e podem ser uma solução para os desafios na elaboração de aplicativos.

Um *Progressive Web Apps* é um *site* que pode ser feito para parecer com e funcionar como um aplicativo nativo em um *smartphone*, *tablet*, *laptop* ou *desktop*. Inicia como uma simples aba no navegador e se torna, progressivamente, um aplicativo à medida que o usuário se engaja e interage com ele, até o momento em que o adiciona à tela inicial do dispositivo; nesse momento, passa a adquirir funções que anteriormente eram exclusivas de aplicativos nativos, como: geolocalização, notificações *push*, uso *offline*, câmera, entre outros. Possui serviços como o *service workers*, que executam *scripts* em segundo plano, e permite a execução e o armazenamento *offline*.

Conforme apresentado, verifica-se que o PWA traz benefícios para desenvolvedores e empresas, sendo que utiliza linguagem de programação simples e abrangente, como HTML, CSS e *JavaScript*, o que permite desenvolvimento e manutenção com baixo custo e menor tempo, sem contar a portabilidade para outros

navegadores e plataformas, bem como sua disponibilização em diversos tipos e formatos de dispositivos, assegurados pelo *design* responsivo.

Outra vantagem para as aplicações *web* progressivas é que a sua tecnologia permite criar um *app* sem a necessidade de passar pelo processo de aprovação das lojas de aplicativos, e o usuário não precisa se comprometer a instalar um aplicativo antes mesmo de saber suas funcionalidades, já que no PWA este procedimento é completamente invisível e gradual, engajando o usuário e aprimorando sua performance a cada interação com a aplicação. Os PWA também são relativamente seguros, pois em sua instalação não acessam o *hardware* dos sistemas; essa permissão é explícita e concedida pelo usuário, quando há a necessidade de acesso a determinados recursos.

Dentro do cenário apresentado, percebe-se que a abordagem *PWA* representa um grande avanço no desenvolvimento de aplicativos móveis, tanto pelo uso dos recentes recursos de tecnologias quanto pela portabilidade de plataforma, porém todas as tecnologias conceituadas são novas, estão em crescimento contínuo e são estruturadas pela evolução da *web*, o que pode causar receio nos desenvolvedores em sua utilização. Em contrapartida, observam-se empresas gigantescas, como *Google* e *Microsoft*, investindo massivamente em aplicações *web* progressivas, por considerarem essa tecnologia um novo passo para a evolução da *web*.

REFERÊNCIAS

ATER, T. **Building Progressive Web Apps**. Sebastopol: O'Reilly Media Inc, 2017

FERNANDES, Gabriel et al. **Projeto e desenvolvimento de aplicações para dispositivos móveis híbridas a partir de tecnologias para Web: um estudo de caso em jogos digitais**. Disponível em: <<https://seer.cesjf.br/index.php/cesi/article/view/1263>> Acesso em: 18 de maio de 2018.

MURAROLLI, P. L; GIROTTI, M. T. **Inovações Tecnológicas nas Perspectivas Computacionais**. São Paulo: Biblioteca 24 Horas, 2015.

TAVARES, Henrique Leal. **Introdução a Desenvolvimento de Aplicações Híbridas**. Disponível em

<http://www.fatecgarca.edu.br/revista/Volume6/artigos_v6/artigo17.pdf>. Acesso em 29 de março de 2018.

LOPES, Sérgio. **A Web Mobile – Programe para um mundo de muitos dispositivos**. Casa do Código, 2013.

TOLEDO, Jan Miszura Toledo; DEUS, Gilcimar Divino de. **Desenvolvimento em Smartphones - Aplicativos Nativos e Web**. Disponível em <<http://www.cpgls.pucgoias.edu.br/7mostra/Artigos/agrarias.pdf>>. Acesso em 05 de novembro de 2018.

SILVA, Marcelo Mora da; SANTOS, Marilde Terezinha Prado. **Paradigmas de Desenvolvimento de Aplicativos para Celulares**. ISSN 2316-2872. T.I.S. São Carlos, v.3, n.2, p.162-170, maio-ago 2014. Disponível em <<file:///C:/Users/User/Documents/2018/Mestrado/Orienta%C3%A7%C3%A3o/Projeto/86-329-1-PB.pdf>>. Acesso em 29 de outubro de 2018.

CARDIERI, Giulia de Andrade; ZAINA, Luciana Martinez. **PWA-EU: uma abordagem para o desenvolvimento de aplicações PWA baseadas em EUD**. Disponível em <<file:///C:/Users/User/Documents/2018/Mestrado/Orienta%C3%A7%C3%A3o/Projeto/4220-697-4217-1-10-20181023-fala%20sobre%20pwa.pdf>>. Acesso em 22 de outubro de 2018.

Google Developers. 2018a. **Progressive Web App**. (2018). Disponível em <<https://developers.google.com/web/progressive-web-apps/>>. Acesso em 22 de março de 2018.