

**APLICATIVOS HÍBRIDOS: DESENVOLVIMENTO DE APLICATIVOS UTILIZANDO
TECNOLOGIAS WEB**

HYBRID APPS: DEVELOPING APPLICATIONS USING WEB TECHNOLOGIES

Alessandro Rocha Martins

Lucas da Conceição

Romantiezer Beloni Pereira¹

André Bessa da Silva²

¹Graduandos em Sistemas de Informação na Faculdade Multivix em Cachoeiro de Itapemirim/ES.

²Mestrando em Inteligência Computacional na Universidade Cândido Mendes UCAM CAMPOS -RJ – Professor da Faculdade Multivix em Cachoeiro de Itapemirim/ES.

RESUMO

A diversidade de dispositivos móveis e seus sistemas operacionais, e a necessidade de desenvolver mais de uma vez o aplicativo para cada sistema operacional distinto é desafiador, de modo a ser necessário possuir equipes com tais conhecimentos específicos, além do custo considerável por projeto. Como solução, o desenvolvimento híbrido, junto de um framework, possibilita o alcance de várias plataformas com um único código fonte e, em grande parte, por via das tecnologias web, como o HTML5, CSS3 e Javascript. Os principais frameworks para o desenvolvimento híbrido são o Phonegap, Ionic e Xamarim, dos quais, cada um têm suas particularidades. Portanto, antes de realizar a escolha é fundamental que seja observado pontos como o nível de conhecimento do profissional com a ferramenta e quais recursos o aplicativo necessitará acessar no smartphone. Sendo assim, o intuito deste trabalho é orientar que há outras possibilidades de desenvolvimento de aplicativo que, não tão diferente da abordagem nativa, que é o desenvolvimento de aplicativos utilizando tecnologias da própria plataforma como o Java e o Swift. A híbrida tem ganhando espaço no mercado por ser uma alternativa que permite desenvolver aplicativos com um desempenho satisfatório, com menor custo e alcançando diversas plataformas, como Android e iOS.

Palavras-chave: Aplicativo móvel. Desenvolvimento híbrido. Dispositivo móvel. Framework.

ABSTRACT

The diversity of mobile devices and their operating systems, the need to more than once develop the application for each distinct operating system is challenging, so it is necessary to have teams with such specific knowledge in addition to the considerable cost per project. As a solution, the hybrid development, together with a framework, allows the reach of several platforms with a single source code and, in large part, through web technologies such as HTML5, CSS3 and Javascript. The main frameworks for hybrid development are Phonegap, Ionic and Xamarim, of which each has its own particularities. Therefore, before making the choice, it is essential that you observe points such as the level of knowledge of the professional with the tool and what resources the application will need to access on the smartphone. Therefore, the purpose of this paper is to guide that there are other possibilities of application development that, not so different from the native approach, is the development of

applications using technologies of the platform itself as Java and Swift. The hybrid has been gaining ground in the market as an alternative to developing applications with satisfactory performance at a lower cost and reaching various platforms such as Android and iOS.

Keywords: Mobile app. Hybrid development. Mobile device. Framework.

1 INTRODUÇÃO

De maneira geral, os aplicativos são construídos para sua linguagem nativa, ou seja, seu sistema operacional específico como o Android que utiliza o Java e o iOS que utiliza o Swift, por exemplo, de forma a ser necessário desenvolvê-lo mais de uma vez, caso seja de interesse disponibilizá-lo em mais de uma plataforma. Sendo assim, o custo para desenvolver uma aplicação móvel pode torna-se inviável por demandar maior conhecimento, ter infraestrutura apropriada, licenças para desenvolvimento entre outros, assim como destaca Gabellini (2016).

Diante de tal desafio, surge a abordagem de desenvolvimento híbrido, a qual traz como proposta a construção de aplicativos com menor custo e capazes de acessar os principais recursos dos smartphones, a título de exemplo o GPS e a câmera, de igual forma se fosse desenvolvido em uma linguagem nativa. Portanto, esses aplicativos são gerados a partir das tecnologias web que, integradas a um framework, permitem então o acesso aos tais recursos.

Gasparotto (2015) afirma que essa abordagem ganha importância por reduzir custos do projeto. Tendo como motivação a redução de custos, o reaproveitamento do código fonte é o fator principal, otimizando o tempo de produção escrevendo apenas uma vez o código e claro, diminui os encargos financeiros com uma equipe menor, já que não é necessário ter vários desenvolvedores para linguagens distintas.

Por conseguinte, o objetivo da pesquisa é poder contribuir com a comunidade de desenvolvedores mostrando que, é possível desenvolver um aplicativo com menor custo financeiro, em menor tempo, agregando uma ótima experiência de uso para os usuários. Sendo assim, fortalecendo a abordagem híbrida, à qual é uma opção,

principalmente por ser capaz de realizar muitas funções desempenhadas pelos aplicativos desenvolvidos de linguagens nativas.

Conforme destacam Silva e Santos (2014), há uma importância em realizar estudos sobre todo o processo de maneira a ser possível determinar a melhor prática para a construção do aplicativo, se será em linguagem nativa ou híbrido, levando em consideração diversos pontos importantes para todo o contexto de desenvolvimento, vislumbrando uma possível necessidade de acesso a uma determinada função do aparelho.

2 DESENVOLVIMENTO HÍBRIDO

Conforme Lynch (2014), o PhoneGap foi desenvolvido pela Nitobi Software. O objetivo proposto no projeto era fazer com que os aplicativos fossem desenvolvidos de uma maneira não nativa, fazendo uso das tecnologias da web: linguagem de marcação de hipertexto (HTML), folha de estilo em cascata (CSS) e JavaScript, podendo então acessar os recursos nativos da plataforma por intermédio de uma API Javascript.

Consoante ao entendimento de Charland e LeRoux (2011), o PhoneGap foi apresentado oficialmente por Eric Oesterle, Rob Ellis e Brock Whitten em um evento em São Francisco, chamado iPhone Dev Camp, para plataforma iPhone OS SDK e só mais tarde foi implementado para o Android, BlackBerry, Windows Phone e para as outras plataformas que o mesmo suporta. Um tempo após ser apresentado, a empresa Apple confirmou a plataforma como segura para o desenvolvimento iOS por estar de acordo com as diretrizes da empresa.

Em conformidade com Lynch (2014), a Adobe Systems comprou a Nitobi Software em 2011. Logo em seguida, o código fonte do Phonegap foi doado para a Apache Software Foundation para começar um projeto chamado Apache Cordova, que é o mesmo projeto, porém, open source, no caso, é aberto à comunidade. O PhoneGap é uma versão comercial da adobe. O Apache Cordova é usado em muitos projetos pelos desenvolvedores e serviu como base para desenvolver outros frameworks voltado aos aplicativos híbridos, como por exemplo, o Ionic.

3 APLICATIVOS MÓVEIS

Segundo os dizeres de Silva, Pires e Carvalho Neto (2015), os aplicativos foram desenvolvidos com o propósito de serem executados através de dispositivos móveis, a exemplo os tablets e smartphones, que tenham recurso de instalação e capacidade para armazenar e executá-los. Pinho e Oliveira (2014) afirmam que existem diversos tipos de aplicativos disponíveis, porém é necessário que o usuário possua um dispositivo móvel capaz de executá-los.

Os dispositivos móveis atuais são aparelhos capazes de executar e armazenar esses aplicativos. Essa capacidade de execução é oriunda de um hardware agregado ao aparelho que o concede um ótimo desempenho. É comum encontrar aparelhos que possuem processadores potentes, além de uma quantidade satisfatória de memória RAM e armazenamento interno, possibilitando então a instalação dos aplicativos, bem como sua utilização e, para que se torne possível utilizar o aplicativo, é necessário realizar o download do mesmo.

Para a realização do download de um aplicativo é necessário a conexão com uma fonte de internet. Os aplicativos são liberados para downloads via lojas virtuais, existindo uma para cada sistema operacional. A loja virtual da Apple é a Apple Store e só são utilizáveis para os dispositivos com o sistema operacional mobile iOS, enquanto que os dispositivos Android utilizam a loja Play Store, pertencente ao Google. O Windows Phone Store é a loja da Microsoft e foram desenvolvidos para dispositivos com este sistema operacional.

Uma parte considerável destes aplicativos são liberados de forma gratuita. Outros, por sua vez, são bloqueados e liberados apenas em casos específicos, quando o usuário realiza sua aquisição, por exemplo. Existem algumas empresas que atuam nessa área de desenvolvimento de aplicativos móveis e adotaram políticas para a liberação dos seus produtos e, quando liberados gratuitamente, os recursos são limitados, funcionando somente com funções básicas. Caso o usuário queira ter direito a todas as funções ele precisará fazer a aquisição do aplicativo na loja.

3.1 Tipos de Aplicativos Móveis

Taurion (2013) reitera a existência de três tipos de aplicativos que podemos desenvolver, cada um com suas capacidades e restrições, nos quais são: os aplicativos nativos, web app e os híbridos. De acordo com Stangarone (2016), a escolha entre os tipos de desenvolvimento dependerá de cada situação, de forma a considerar alguns pontos cruciais, bem como o tempo esperado para entrega e quais recursos necessários e/ou fundamentais que esses aplicativos irão precisar acessar e, quais as habilidades que a equipe possui.

Pinho e Oliveira (2014) afirmam que o desenvolvimento de um aplicativo nativo é lento e carece de esforço considerável da equipe. Nessa prática, é preciso realizar a construção do mesmo aplicativo para todas as plataformas de dispositivos móveis (sistema operacional móvel) mais usadas e desejadas pelos usuários, garantindo assim maior alcance de dispositivos.

Conforme Taurion (2013), os aplicativos nativos são aqueles construídos para uma plataforma mobile específica, como por exemplo o Android e o iOS. Estes, devem ser desenvolvidos utilizando a linguagem de programação própria da plataforma. Isto é, se for para Android, utiliza Java, para iOS, Swift (antes, Objective-C). Com isso, teremos acesso completo a todos os recursos disponíveis no dispositivo, bem como câmeras, acelerômetro, GPS e outros.

Stark e Jepson (2012, p. 16) afirmam que “um aplicativo de web é basicamente um website que é otimizado especificamente para uso em um smartphone”, enquanto que nos dizeres de Taurion (2013), os aplicativos de web são acessados pelo navegador, assim como nos desktops, porém para os dispositivos móveis é necessário ser construído de modo que eles reconheçam a tela dos dispositivos e se ajustem a ela, pois a tela que visualizamos nos dispositivos serão diferentes.

De acordo com Pinho e Oliveira (2014) todos os dispositivos móveis atuais possuem um navegador, desta forma, eles podem ser acessados de qualquer lugar desde que o dispositivo esteja conectado à internet. Outras características marcantes dessa abordagem são os fatos de não serem instalados e não serem escritos na linguagem específica da plataforma, como afirmam Stark e Jepson (2012).

Esses aplicativos rodam dentro do navegador e, por motivo de segurança, não possuem acesso direto aos recursos da plataforma nativa, como câmera, geolocalização, animações 3D com aceleração na Graphics Processing Unit (GPU) e outros. Vale ressaltar que existem muitos recursos específicos nos dispositivos móveis os quais só são acessados por meio das plataformas nativas, de maneira que se torna impossível acessá-los através de uma Web Apps (LOPES, 2013).

Conforme anuncia Stangarone (2012), boa parte dos aplicativos possuem uma parte híbrida. Dentre estes, aplicativos de famosas redes sociais como Facebook e o LinkedIn, como também o da poderosa Netflix foram desenvolvidos nessa metodologia. Ela utiliza a mesma interface para todas as plataformas móveis, porém o streaming do aplicativo é desenvolvido de modo nativo. Visualmente observando, não há diferença entre nativo e híbrido para o usuário final.

Segundo Silva, Pires e Carvalho Neto (2015), esses aplicativos utilizam um controle de visualização da web (Web View) que é um tipo de navegador onde o aplicativo será executado. A Web View funciona em modo de tela cheia e não possui os componentes de um navegador, como a barra de endereço e outros controles, possui apenas o que é preciso para que o HTML5, CSS3 e JavaScript funcione.

Segundo Stangarone (2012), em muitas situações é possível utilizar os aplicativos híbridos ao invés dos nativos e, assim, a empresa economiza tempo e dinheiro. Budiu (2013) sustenta a ideia de que os aplicativos híbridos são multiplataforma, ou seja, é possível utilizá-los em mais de um sistema operacional móvel, diminuindo assim os custos de desenvolvimento no qual é produzido somente uma vez.

O desempenho é inferior se comparado às aplicações nativas e a interface é mais limitada, não sendo possível construir aplicativos que utilizem gráficos pesados como os jogos, mas, por ser multiplataforma, têm conquistado seu espaço no mercado. Sendo assim, o foco no desenvolvimento é maior e permite à equipe trabalhar e realizar atualizações periódicas, já que o código só será alterado uma única vez, facilitando, portanto, todo o processo (CAMPAGNOLI, 2015).

De acordo com Taurion (2013), é fundamental realizar a escolha da abordagem a ser utilizada antes de iniciar a codificação. Isso se dá pelo fato de que, cada abordagem irá conter suas vantagens e desvantagens, isto é, irá depender de cada situação em específico. Todavia, a abordagem híbrida pode ser interessante caso seu objetivo seja alcançar muitas plataformas mobile, e claro, se houver interesse em adquirir um produto com menor custo financeiro e de tempo, além de oferecer o melhor dos aplicativos nativos da linguagem e web.

Tabela 1 – Comparação aplicativos móveis.

	Nativo	Híbrido	Web móvel
Habilidades necessárias para alcançar Android e iOS:	Swift Java C#	HTML CSS Javascript Framework	HTML CSS JavaScript
Distribuição:	Loja de aplicativos	Loja de aplicativos	Internet
Velocidade de desenvolvimento:	Devagar	Moderada	Rápida
Custo de desenvolvimento:	Alto	Moderado	Baixo
Custo de manutenção:	Alto	Moderado	Baixo
Desempenho gráfico:	Alto	Moderado	Moderado
Desempenho do aplicativo:	Rápido	Moderado	Moderado
Melhor usado para:	Jogos/aplicativos que necessitam de alto desempenho gráfico	Aplicativos que necessitam de acesso total ao dispositivo e que não precisam de alto desempenho	Aplicativos que não necessitam de um alto desempenho, e precisam apenas de acesso parcial ao dispositivo

Fonte: Adaptada de Stangarone (2016).

4 TECNOLOGIAS USADAS PARA DESENVOLVER APLICATIVOS HÍBRIDOS

Os aplicativos desenvolvidos de forma híbrida operam a partir do browser do sistema operacional móvel, utilizando tecnologia web (HTML5, CSS3 e JavaScript) em boa parte de sua composição. Vale ressaltar que esses aplicativos, diferente dos nativos, não possuem acesso direto aos recursos dos dispositivos, sendo necessário o uso de um framework, o qual irá possibilitar o uso dos recursos em questão (MENDES; GARBAZZA; TERRA, 2014).

4.1 HTML, CSS e JavaScript

Segundo os ensinamentos de Feitosa (2012), o HTML é uma linguagem de marcação de hipertexto e é utilizada por todos os navegadores, independentemente da plataforma. Pinho e Oliveira (2014) afirmam que a linguagem HTML é constituída por tags que definem o que é um parágrafo, um vídeo, imagem, títulos dentre outros. Em outras palavras, ele é o responsável pela estrutura de uma página web, de maneira que o navegador vai definir como mostrar determinado conteúdo da página de forma que dê mais sentido a cada elemento.

De acordo com Stark e Jepson (2012), os navegadores tem um estilo padrão para apresentar as tags, porém, básicos, sendo necessário a utilização de estilos mais avançados para posicionar e controlar os elementos em tela, definindo como será apresentado. Usando o CSS3, é possível determinar o visual do site adicionando cores, alterando fontes e tamanho, adicionando imagens de fundo e modificando todo o layout, permitindo assim apresentar um design mais atrativo para as páginas web desenvolvidas.

Segundo Barros e Santos (2008), o CSS3 é responsável pelo design das páginas e o faz através dos seletores que permitem formatar uma tag. A sintaxe do CSS3 é composta por três partes: o seletor, a propriedade e o valor. O seletor pode ser a própria tag, uma classe ou um identificador. A propriedade é o atributo que o seletor vai receber, podendo ser uma ou mais propriedades. E, por último, o valor que é a configuração da propriedade.

O Javascript é uma linguagem de script que também é utilizada em uma página HTML e é através dela que se torna possível atualizar o conteúdo da página sem a necessidade de atualiza-la por inteiro com um refresh, basta realizar uma requisição no servidor de forma assíncrona com Ajax. Outra funcionalidade bem comum é para realizar a validação de campos de formulários (STARK; JEPSON, 2012).

4.2 Frameworks: Phonegap, Ionic e Xamarin

Segundo Pinho e Oliveira (2014), no mundo de desenvolvimento de aplicativos móveis, o maior desafio é alcançar as diversas plataformas existentes no mercado, já que cada uma possui uma linguagem própria. Desta forma, impossibilita que um aplicativo desenvolvido para uma plataforma nativa seja compatível com todas as outras.

4.2.1 Phonegap

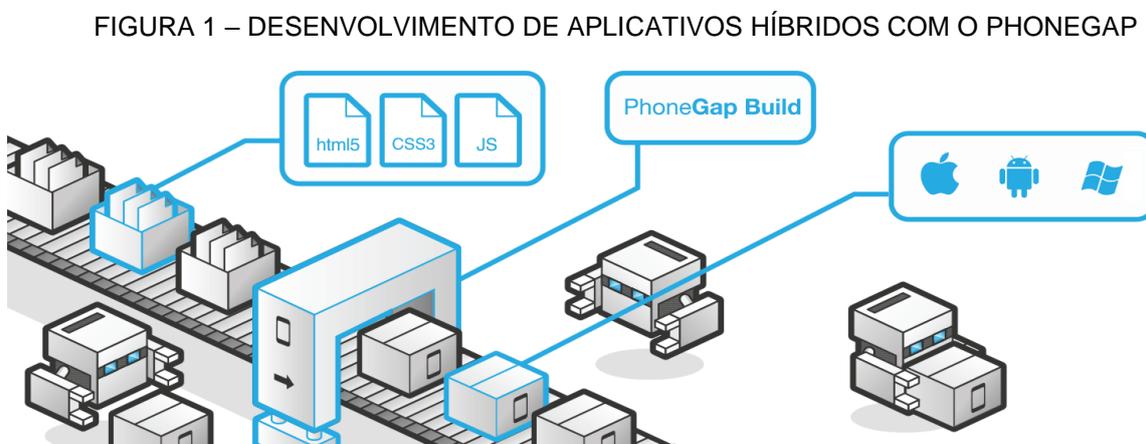
Conforme Santos Júnior (2015), o PhoneGap é um framework que possibilita o desenvolvimento de aplicações móveis através das tecnologias web (HTML5, CSS3 e Javascript). Ele contém um conjunto de API's para acessar funções nativas dos dispositivos através do Javascript e o resultado disso é um aplicativo híbrido que funciona em diversas plataformas, como o Android, iOS, Windows Phone e outros.

Para Souza e Lemes (2014), os aplicativos desenvolvidos com o Phonegap utilizam uma Web Browser View, que é um tipo de navegador web que ocupa 100% da largura e altura do dispositivo para mostrar todo o conteúdo visual para o usuário. Porém, diferente dos navegadores padrão, como por exemplo o Google Chrome e Firefox, ele não possui uma barra de ferramentas, barra de rolagem e nenhuma outra característica de um navegador padrão.

Os aplicativos híbridos não utilizam a estrutura de UI nativa da plataforma e, também não usam simplesmente navegadores para a visualização de todo o layout, mas sim uma Webview, tornando possível desenvolver um aplicativo para todas as plataformas, pelo fato de que todos os sistemas operacionais que o PhoneGap suporta faz uso da mesma Webview em seu código nativo. O Android por exemplo, utiliza a classe android Webkit WebView e o iOS a classe UIWebView.

O resultado de uma aplicação desenvolvida com o Phonegap é um arquivo binário que pode ser distribuído para os seus respectivos sistemas operacionais, assim como destaca Trice (2012). Para os aplicativos Android, é gerado um arquivo APK (Android Package), e para o iOS é gerado um arquivo IPA (iOS Application archive), sendo os mesmos gerados nativamente e distribuídos em suas respectivas lojas virtuais.

Em suma, as principais ações do PhoneGap em seu processo funcional é receber o HTML5, CSS3 e o Javascript, encapsular e gerar o arquivo de acordo com a plataforma escolhida, sendo possível também o acesso às funções nativas através do JavaScript por intermédio de APIs. A figura 1 representa todo esse processo de desenvolvimento por meio do uso do framework.



Fonte: PhoneGap, 2018.

4.2.2 Ionic

Desenvolvido pela Drifty Co, o Ionic framework é um kit de desenvolvimento de software (SDK) open source para desenvolvimento de aplicações mobile híbridas por meio do HTML5, CSS3 e Javascript. Este possui uma gama de funções e componentes que permitem criar aplicativos de alto desempenho (VENTEU; PINTO, 2018). Com o framework, é possível criar aplicações multiplataformas e utilizar as funcionalidades nativas dos dispositivos através do Apache Cordova, assim como destaca Felizardo (2018).

Por ser construído sobre o AngularJS e utilizar o Apache Cordova, responsável pela webview, ele é capaz de utilizar as tecnologias web e as renderizar. Wahlbrinck e Boniati (2017) declaram que, com as diretrizes herdadas do AngularJS, o Ionic pode reusar seu código HTML e sua lógica de marcação, os quais são alguns de seus recursos.

O objetivo do framework é possibilitar uma interface amigável a aplicação, visando uma boa aparência e experiência de uso do aplicativo por parte do usuário. Outro ponto crucial é a simplificação de processos por meio de suas ferramentas,

contribuindo com a garantia de maior eficiência para o projeto, de modo a assegurar a agilidade com o desenvolvimento.

4.2.3 Xamarin

Segundo Quaiato (2016), o Xamarin é uma suíte de ferramentas disponibilizadas para desenvolvedores de aplicativos mobile. Essa plataforma permite a construção de aplicativos através da linguagem de programação C# (C Sharp) e utilizar os recursos desta linguagem. Provedi (2016) afirma que o Xamarin permite o desenvolvimento multiplataforma por meio de um único código, sendo necessário desenvolver apenas a interface para as plataformas desejadas posteriormente. A plataforma funciona com a utilização da linguagem C# para desenvolver o código fonte, enquanto que a aplicação gerada será de forma nativa por conseguir explorar os recursos da linguagem desejada.

Com o código desenvolvido é possível utilizá-lo para gerar aplicativos para outros sistemas operacionais, porém, o compartilhamento de código não se dá por total. Gasparotto (2017) estima que é reaproveitado cerca de 75% do código construído, agregando os acessos aos recursos como também a interface de usuário. Portanto, assim como afirma Quaiato (2016), o reaproveitamento do código vem com toda a camada de negócios, ficando então pendente somente a codificação da chamada à API específica e sua interface de usuário.

A plataforma contribui com a produção das aplicações através de suas ferramentas integradas e têm-se como destaque o Xamarin Test Cloud e Xamarin Insight. O uso do xamarin test cloud é para o processo de testes, assegurando a qualidade do produto antes mesmo de iniciar seu desenvolvimento. Porém, esta fase de testes pode se tornar inviável caso seja necessário realizar testes físicos, ou seja, testes reais com hardwares e sistemas operacionais distintos, de maneira a necessitar investir em diversos dispositivos para tal (GASPAROTTO, 2015).

A contribuição do Xamarin Test Cloud está com a possibilidade de disponibilizar, na nuvem de testes, o teste de interface de usuário, o qual pode ser realizado através de milhares de dispositivos reais. Para que seja possível essa disponibilização para testes, a equipe necessita estar cadastrada na base de dados do Test Cloud e então

escolher a plataforma de interesse. Para disponibilizá-lo, é necessário escolher os dispositivos desejados para a realização dos testes.

Uma segunda ferramenta bastante importante é o Xamarin Insight, o qual é, de acordo com Vyedin (2014), um sistema para monitoramento do aplicativo em tempo real, buscando encontrar e relatar problemas que, de alguma forma, estão atrapalhando a experiência de uso do usuário. Porém, o Xamarin Insight foi descontinuado e o projeto foi assumido pelo Visual Studio App Center, o qual manteve algumas funções antigas.

5 METODOLOGIA

Um trabalho a ser elaborado ganha importância a partir da busca por conhecimentos úteis ao avanço da ciência, assim como dizem Gerhardt e Silveira (2009), que definem estes trabalhos como de natureza básica, de forma a fomentar a comunidade a buscar e ampliar os conhecimentos relacionados ao assunto abordado e de interesse mútuo.

O presente trabalho configura-se uma pesquisa bibliográfica por ser baseado em estudos relacionados à temática do desenvolvimento multiplataforma, e correlacionados, já feitos anteriormente ao tempo presente. Portanto, foi realizado pesquisas em fontes que possui foco no assunto abordado, possibilitando então estruturar e definir o que deveria ser estudado e buscar quais informações existem sobre o assunto pesquisado.

Sendo assim, o principal objetivo da pesquisa foi explorar o assunto de desenvolvimento híbrido, de maneira a buscar maiores conhecimentos sobre o mesmo. Para buscar esse conhecimento foram utilizadas fontes que possibilitaria uma rápida aquisição das informações necessárias para um melhor entendimento e ser possível abordá-lo com maior destreza.

Gil (2002) afirma que, através da pesquisa bibliográfica, torna possível ao pesquisador obter um conhecimento maior se comparado a uma feita diretamente por ele, sendo por vantagem então o levantamento de um número maior de informações, pois seria bastante trabalhoso saber distinguir todas as diferenças entre os tipos de desenvolvimento de aplicativos, se não fosse pela pesquisa bibliográfica.

Porém, a pesquisa bibliográfica pode prejudicar a qualidade e a confiança do estudo que está sendo feito, já que existem muitas fontes com informações não confiáveis e sem qualidade, de maneira a comprometer o seu resultado final. Portanto, para diminuir essa possibilidade, é preciso que os pesquisadores analisem bem as informações reunidas e utilizem diferentes fontes de dados, garantindo assim a qualidade das informações obtidas em seu estudo.

6 CONSIDERAÇÕES FINAIS

O presente artigo teve como objetivo mostrar a possibilidade de desenvolver aplicativos móveis para diferentes sistemas operacionais mobile utilizando tecnologias web. Nesse estudo, foram abordados conteúdos oriundos de livros e artigos científicos relacionados à temática, pesquisas na internet, de forma a possibilitar apresentar as principais ferramentas para melhor aproveitamento e agilidade no desenvolvimento.

Deve-se notar como o avanço tecnológico se tornou um fator positivo para que os desenvolvedores possam escolher a melhor maneira para trabalhar, agregando assim, não só valor ao seu produto, mas também tornando o projeto mais viável, por meio de ferramentas que contribuam com a minimização de custos financeiros, e assim sendo mais rentável e de fácil elaboração, e claro, tornando o projeto lucrativo.

Neste trabalho o objetivo foi apresentar alguns frameworks de desenvolvimento que dão suporte à abordagem híbrida, de maneira que empresas ou profissionais liberais, busquem uma outra alternativa para o desenvolvimento de aplicações. Sendo assim, foi levantado ferramentas já conceituadas no mercado e com uma comunidade ativa de desenvolvedores.

7 REFERÊNCIAS

BARROS, Isabelle Guimarães M. O. de; SANTOS, Carlos Felipe Araujo dos. Apostila de Introdução ao CSS. Niterói: Universidade Federal Fluminense, 2008. 21 p. (Grupo PET-Tele). Disponível em: <<https://www.telecom.uff.br/pet/petws/downloads/tutoriais/css/css2k80912.pdf>>. Acesso em: 15 nov. 2018.

BUDIU, Raluca. **Mobile: Native Apps, Web Apps, and Hybrid Apps**. 2013. Disponível em: <<https://www.nngroup.com/articles/mobile-native-apps/>>. Acesso em: 18 jun. 2018.

CAMPAGNOLI, Julian de Alessandro. **PhoneGap e Cordova**: Como criar aplicativos mobile híbridos e offline. 2015. Disponível em: <<https://www.devmedia.com.br/phonegap-e-cordova-como-criar-aplicativos-mobile-hibridos-e-offline/32361>>. Acesso em: 18 jun. 2018.

CHARLAND, Andre; LEROUX, Brian. Mobile application development. **Communications Of The Acm**, Newyork, v. 54, n. 5, p.49-53, 5 maio 2011. Association for Computing Machinery (ACM). <http://dx.doi.org/10.1145/1941487.1941504>.

FEITOSA, Eduardo. **O que é o HTML5**. 2012. Disponível em: <<https://www.devmedia.com.br/o-que-e-o-html5/25820>>. Acesso em: 18 jun. 2018.

FELIZARDO, André. **Ionic Framework**: Desenvolvendo Mobile Apps. 2018. Disponível em: <<http://www.andrefelizardo.com.br/blog/ionic-framework-desenvolvendo-mobile-apps/>>. Acesso em: 08 nov. 2018.

GABELLINI, Laryssa. **Quanto custa um aplicativo?** 2016. Disponível em: <<https://usemobile.com.br/quanto-custa-um-aplicativo/>>. Acesso em: 27 out. 2018.

GASPAROTTO, Henrique Machado. **Desenvolvimento multiplataforma com Xamarin**. 2015. Disponível em: <<https://www.devmedia.com.br/space/henrique-machado-gasparotto>>. Acesso em: 15 out. 2018.

GASPAROTTO, Henrique Machado. **Xamarin, Ionic e Cordova**: Conheça o que são e as principais diferenças. 2017. Disponível em: <<https://www.devmedia.com.br/xamarin-ionic-e-cordova-conheca-o-que-sao-e-as-principais-diferencas/37690>>. Acesso em: 08 nov. 2018.

GERHARDT, Tatiana Engel; SILVEIRA, Denise Tolfo. **Métodos de pesquisa**. Porto Alegre: Editora da Ufrgs, 2009. 120 p. Disponível em: <<http://www.ufrgs.br/cursopgdr/downloadsSerie/derad005.pdf>>. Acesso em: 21 nov. 2018.

GIL, Antônio Carlos. **Como elaborar projetos de pesquisa**. 4. ed. São Paulo: Atlas S.A., 2002. 176 p.

LOPES, Sergio. **A web mobile**: programe para um mundo de dispositivos. São Paulo: Casa do Código, 2013. 308 p.

LYNCH, Max. **A última palavra em Cordova e PhoneGap**. 2014. Disponível em: <<https://blog.ionicframework.com/what-is-cordova-phonegap/>>. Acesso em: 11 nov. 2018.

MENDES, Mariana Ribeiro; GARBAZZA, Itagildo Edmar; TERRA, Daniela Costa. Desenvolvimento híbrido versus desenvolvimento nativo de aplicativos móveis.

Semana de Ciência e Tecnologia do IFMG, Bambuí, p. 1-5, out. 2014. Disponível em:

<https://www.bambui.ifmg.edu.br/jornada_cientifica/2014/resumos/Info/Desenvolvimento%20h%C3%ADbrido%20versus%20desenvolvimento%20nativo%20de%20apl.pdf>. Acesso em: 21 jul. 2018.

PHONEGAP. **Adobe PhoneGap Build**. Disponível em:
<<https://build.phonegap.com/>>. Acesso em: 11 nov. 2018.

PINHO, Gabriel Paes; OLIVEIRA, José Fernando Rodrigues. **Desenvolvimento de aplicativos híbridos multiplataforma para dispositivos móveis**. 2014. 68 f. TCC (Graduação) - Curso de Tecnologia em Análise e Desenvolvimento de Sistemas, Instituto Federal de Educação, Ciência e Tecnologia do Pará, Itaituba, 2014.

PROCEDI, Lisandro. **Avaliação do framework Xamarin. Forms para desenvolvimento de aplicativos móveis multiplataforma, criando uma aplicação real**. 2016. 53 f. Monografia (Especialização) - Curso de Ciência da Computação, Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2016.

QUAIATO, Vinicius. **O que é Xamarin?** 2016. Lambda3. Disponível em:
<<https://www.lambda3.com.br/2016/10/o-que-e-xamarin/>>. Acesso em: 16 out. 2018.

SANTOS JÚNIOR, Gesmar de Paula. **Development geolocation system in augmented reality for mobile multiplatform**. 2015. 63 f. Dissertação (Mestrado em Engenharias) - Universidade Federal de Uberlândia, Uberlândia, 2015.

SILVA, Leandro Luquetti B. da; PIRES, Daniel Facciolo; CARVALHO NETO, Silvio. Desenvolvimento de Aplicações para Dispositivos Móveis: Tipos e Exemplo de Aplicação na plataforma iOS. **II Workshop de Iniciação Científica em Sistemas de Informação**, Goiânia, p. 25-30, maio. 2015. Disponível em:
<<http://www.lbd.dcc.ufmg.br/colecoes/wicsi/2015/004.pdf>>. Acesso em: 18 jun. 2018.

SILVA, Marcelo Moro da; SANTOS, Marilde Terezinha Prado. Os Paradigmas de Desenvolvimento de Aplicativos para Aparelhos Celulares. **Departamento da Computação**: Universidade Federal de São Carlos (UFSCar, São Carlos, v. 3, n. 2, p.162-170, maio 2014.

SOUZA, Leonardo Moreira de; LEMES, Túlio Henrique Seixas. **Um estudo sobre o PhoneGap e seu desempenho ante a linguagem nativa do Android**. 2014. 55 f. TCC (Graduação) - Curso de Sistemas de Informação, Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, 2014. Disponível em:
<<http://bsi.uniriotec.br/tcc/textos/201412SouzaLemes.pdf>>. Acesso em: 21 nov. 2018.

STANGARONE, Joe. **Mrc's Cup of Joe Blog**. 2012. Disponível em:
<<https://www.mrc-productivity.com/blog/2012/02/why-hybrid-mobile-apps-are-so-popular-right-now/>>. Acesso em: 18 jun. 2018.

STANGARONE, Joe. **Mrc's Cup of Joe Blog**. 2016. Disponível em: <<https://www.mrc-productivity.com/blog/2016/06/the-mobile-app-comparison-chart-hybrid-vs-native-vs-mobile-web/>>. Acesso em: 18 jun. 2018.

STARK, Jonathan; JEPSON, Brian. **Construindo Aplicativos Android com HTML, CSS e Javascript**: Criando Aplicativos Nativos com Ferramentas Baseadas nos Padrões Web. São Paulo: Novatec, 2012. 200 p.

TAURION, Cezar. **O desafio de desenvolver apps para o mundo móvel – Parte 02**. 2013. Disponível em: <<https://imasters.com.br/mobile/o-desafio-de-desenvolver-apps-para-o-mundo-movel-parte-02/>>. Acesso em: 18 jun. 2018.

TRICE, Andrew. **PhoneGap explicado visualmente**. 2012. Disponível em: <<https://phonegap.com/blog/2012/05/02/phonegap-explained-visually/>>. Acesso em: 11 nov. 2018.

VENTEU, K. C.; PINTO, G. S. **DESENVOLVIMENTO MÓVEL HÍBRIDO**. Revista Interface Tecnológica, v. 15, n. 1, p. 86-96, 30 jun. 2018.

VYEDIN, Nina. **Introducing Xamarin Insights**: Real-time Monitoring for Your Apps. 2014. Disponível em: <<https://blog.xamarin.com/monitoring-your-apps-with-xamarin-insights/>>. Acesso em: 16 out. 2018.

WAHLBRINCK, Kamile A.; BONIATI, Bruno B. Aplicações Mobile Híbridas: Um Estudo de Caso do Framework Ionic para Construção de um Diário de Classe. **Anais do Eati**: Encontro Anual de Tecnologia da Informação, Frederico Westphalen, v. 7, n. 1, p.69-76, nov. 2017. Anual.