

Estudo de Frameworks de Desenvolvimento Multiplataformas para Dispositivos Móveis

Rômulo Vanzin^{1,2}, Bruno Batista Boniati² e Teresinha Letícia da Silva²

¹Pos-graduação em Gestão de Tecnologia da Informação - Colégio Agrícola de Frederico Westphalen – Universidade Federal de Santa Maria.

²Colégio Agrícola de Frederico Westphalen – Universidade Federal de Santa Maria
Caixa Postal 54 – 98.400-000 – Frederico Westphalen – RS
romulofw@gmail.com, {bruno,leticia}@cafw.ufsm.br

Resumo. *O presente trabalho apresenta um estudo de diferentes Frameworks de Desenvolvimento Multiplataforma para Dispositivos Móveis. Esta é uma parte inicial de um projeto que tem por objetivo implementar uma aplicação de coleta de dados utilizando frameworks de desenvolvimento mobile multiplataforma, com o intuito de elencar as principais características destes frameworks, possibilitando assim, que se tenha o embasamento necessário para optar ou não dentre estas tecnologias de desenvolvimento em projetos para dispositivos móveis.*

1. Introdução

O grande crescimento da utilização de dispositivos móveis nos últimos anos, nos remete também, ao aumento significativo do número de aplicativos desenvolvidos para tais dispositivos. A popularização destes equipamentos deve-se à redução do custo (permitindo sua aquisição pela maioria da população) e ao aumento da capacidade de armazenamento e processamento destes aparelhos. Neste contexto, pode-se dizer que existem duas plataformas que dominam o mercado de dispositivos móveis, a plataforma Android (Google) e a plataforma iOS (Apple) [UOL Notícias, 2012].

As empresas que necessitam migrar ou desenvolver seus aplicativos para a plataforma *mobile* não podem estar atreladas ou dependentes a exclusivamente uma plataforma. Com a evolução do mercado *mobile* as empresas deverão produzir aplicativos que possibilitem, por exemplo, que um usuário da plataforma Android tenha o mesmo aplicativo para a plataforma iOS e vice versa.

Uma possível alternativa para tal situação são os *frameworks* de desenvolvimento multiplataforma. Tais *frameworks* fornecem vastas possibilidades de desenvolvimento para suprir esta dificuldade, e também, proporcionam a solução de outros problemas, entre os quais o tempo de desenvolvimento, migração de aplicativos para as diferentes plataformas, formação de equipes de desenvolvimentos distintas, produção de códigos fontes em diversas linguagens, custo em licenças de software para desenvolvimento *mobile* e treinamento das equipes de desenvolvimento.

Este artigo descreve os principais *frameworks* de desenvolvimento *mobile*, ao longo da seção 2 serão abordados alguns *frameworks* nativos e alguns *frameworks* multiplataforma ressaltando as suas principais características.

2. Frameworks de desenvolvimento mobile

Até pouco tempo atrás, havia duas correntes de desenvolvimento *mobile*: a programação para iOS e a programação para Android. Com o crescimento do mercado, começaram a surgir *frameworks* multiplataforma, cuja promessa é abstrair a API (*Application*

Programming Interface) nativa do aparelho, e possibilitar a criação de projetos multiplataforma, de forma rápida e com baixo investimento.

Existem *frameworks* de desenvolvimento *mobile* para três tipos de aplicações: nativa, *web* e híbrida. As aplicações nativas são aquelas desenvolvidas usando a linguagem de programação específica de uma plataforma, fornecendo acesso a todos os recursos de hardware e aumentando a usabilidade, no entanto, necessitam de um maior tempo de desenvolvimento e são restritas a plataforma escolhida. As aplicações *web* são dependentes de acesso a internet, não possuem acesso direto ao *hardware* e são executadas dentro do navegador *web* do dispositivo (dificultando questões de usabilidade), mas são portáteis e multiplataforma. As aplicações híbridas são uma mistura das aplicações nativas e *web*, buscando o melhor dos dois mundos: tempo médio de desenvolvimento e usabilidade adequada.

2.1 Frameworks Nativos

A seguir serão discriminados os principais *frameworks* para desenvolvimento de aplicações nativas. Em cada caso serão citadas brevemente algumas características relevantes como a plataforma de execução e a linguagem de utilizada.

2.1.1 Android SDK

Utilizado para o desenvolvimento de aplicações para a plataforma Android. O Android SDK é gratuito e é executado em tanto em MS Windows, Mac ou Linux e utiliza várias IDEs para desenvolvimento, como Eclipse e NetBeans com a linguagem Java ou C++ [Duarte Jr, 2012].

2.1.2 iOS SDK

SDK oficial da Apple para desenvolvimento de aplicações móveis para plataformas como iPhone, iPod e iPad. Os usuários destas plataformas possuem objetivos bem definido quanto a usabilidade e aparência. Ele roda somente em Mac e utiliza o Xcode como IDE e utiliza a linguagem Objective-C [Duarte Jr, 2012].

2.1.3 Java Micro Edition

Primeiro *framework* utilizado para desenvolvimento de aplicações para dispositivos móveis, foi desenvolvido pela Sun Microsystems e atualmente é mantido pela Oracle. O JME já foi largamente utilizado para celulares, e hoje em dia está em segundo plano devido à grande quantidade de *smartphones* que utilizam sistemas operacionais mais robustos. Mesmo assim, é plataforma dominante entre os celulares mais simples e com recursos limitados. Ele roda em Windows, Mac e Linux e utiliza várias IDEs para desenvolvimento como Eclipse e NetBeans com a linguagem Java [Duarte Jr, 2012].

2.1.4 Windows Phone SDK

Plataforma SDK, utilizada por desenvolvedores.NET, é uma ferramenta que permite a criação de aplicativos utilizando as tecnologias Microsoft. Ela é utilizada para projetar e criar aplicativos para Windows Phone. Esta ferramenta gera código para Framework.NET e suporta linguagens Visual Basic .NET, C#, C++, e J#. Trabalha com tecnologias como Visual Studio, Silverlight e .NET, WPF, Silverlight e XAML [Kamada, 2012].

2.2 Frameworks Multiplataforma

Outra categoria de *frameworks* para desenvolvimento *mobile* são os multiplataforma. Estes, são projetados para criar aplicações multiplataforma sem a necessidade de reescrever o código para cada plataforma. Tem como vantagens o código fonte único, a possibilidade de utilizar interfaces gráficas nativas da plataforma e o acesso aos recursos de *hardware*. A seguir são citados os principais *frameworks* multiplataforma disponíveis.

2.2.1 Corona SDK

Criado em 2008 pela empresa californiana CoronaLabs, o Corona SDK é um kit de desenvolvimento que utiliza a linguagem de programação Lua para a criação de jogos e aplicativos para dispositivos móveis. É constituído basicamente de um simulador de dispositivos e um servidor, que gera os códigos e entrega o programa pronto para o seu desenvolvedor [Pereira, 2011]. Provê centenas de APIs para desenvolvimento multiplataforma (iOS, Android, Kindle Fire e Nook Color) e roda sobre Windows e Mac. Possui uma versão *trial* ilimitado, porém, quando se deseja publicar a aplicação nas lojas de aplicativos o preço de licenciamento é bastante alto [Pereira, 2011].

2.2.2 PhoneGap

O Phonegap é uma tecnologia *open source* que tem como principal característica desenvolver aplicativos do tipo *web*, que podem ser funcionais nos ambientes *online* ou *off-line*. Para suprir as necessidades das linguagens nativas de cada plataforma, ele utiliza os padrões *web* como o HTML5, o CSS3 e o Javascript, tratando de uma forma bastante simples o seu desenvolvimento [Gunter, 2011]. O PhoneGap provê um *container web* no qual é possível construir aplicações multiplataforma (iOS, Android, BlackBerry, WebOS, Windows Phone, Symbian e Bada) usando apenas o básico da *web*: HTML5+JS. Ele provê APIs Javascript para que se tenha acesso aos recursos de hardware do *smartphone* e exige apenas que os dispositivos tenham navegadores *web* que suportem esses recursos. Utiliza como IDE de desenvolvimento o Adobe DreamWeaver [Duarte Jr, 2012].

2.2.3 Mono Framework

O Mono Framework é o framework .NET portado para o mundo *open source* multiplataforma. Embora o objetivo inicial do projeto Mono, iniciado pela Novell e hoje mantido pela mexicana Xamarin, era portar aplicações .NET para Linux, hoje pode-se contar com subdivisões do framework como MonoDroid (Android), MonoTouch (iOS) e MonoGame, que é a versão multiplataforma do XNA (framework para jogos) da Microsoft. Usa a plataforma .NET e suas linguagens para construir aplicações e o Visual Studio ou Mono Develop como ferramenta de desenvolvimento [Duarte Jr, 2012].

2.2.4 Titanium SDK

O Titanium usa Javascript para desenvolvimento de suas aplicações para iOS, Android e dispositivos que suportem HTML5 em geral. Seus criadores da Appcelerator desenvolveram mais de 5000 APIs para acesso aos recursos de hardware das plataformas e prometem 70% a mais de produtividade sobre desenvolvimento Java e Objective-C [Revista Mobile Magazine, 2012].

3. Considerações finais

Pode-se concluir que o cenário *mobile* está em constante crescimento, apresentando desafios significativos no desenvolvimento de aplicativos. O crescente número de sistemas operacionais, bem como a combinação de dispositivos, tamanhos físicos e padrões transformaram o desenvolvimento de aplicações utilizando *frameworks* nativos uma estratégia de custo muito alto. Surgem então, como alternativa, um crescente mercado de soluções multiplataforma e ferramentas de desenvolvimento de mobilidade. Esses *frameworks* tem em sua perspectiva a facilidade no desenvolvimento de aplicações, visto que, utilizando um único ambiente, pode-se desenvolver para diferentes plataformas.

Porém, esta tecnologia ainda é indicada para projetos de menor porte ou para projetos que tenham como padrões as funcionalidades já existentes nos *frameworks* e que sejam compatíveis com as plataformas desejadas. Assim, não existe uma solução única e padronizada para todos os tipos de aplicações, cada solução precisa de uma análise detalhada de todos os requisitos que devem ser implementados visando definir a solução mais adequada.

Como continuidade deste trabalho pretende-se implementar uma aplicação de coleta de dados utilizando *frameworks* de desenvolvimento *mobile* multiplataforma, com o objetivo de elencar as principais características destes *frameworks*, possibilitando assim, que se tenha o embasamento necessário para optar ou não dentre estas tecnologias de desenvolvimento em projetos para dispositivos móveis.

Referências

- Duarte Jr., L. F. **Conheça os Frameworks de Desenvolvimento Mobile**. Disponível em: <<http://www.luiztools.com/post/Conheca-os-Frameworks-de-Desenvolvimento-Mobile.aspx>> Acessado em: 20 set. 2012.
- Guinter, M. **Desenvolvimento Mobile Multiplataforma com PhoneGap**. Disponível em: <<http://www.mobiltec.com.br/blog/index.php/desenvolvimento-mobile-multiplataforma-com-phonegap/>> Acessado em: 15 ago. 2012.
- Kamada, T. et al. **Análise das Plataformas de Desenvolvimento Mobile aplicados na Área Educacional, usando Android e Windows Phone. Estudo de Caso: Aplicativo Planetas no Windows Phone**. Revista Novas Tecnologias na Educação, CINTED - UFRGS, V. 10 n.1, julho 2012.
- Pereira, R. R.; Escobar, M. S. **Framework para construção de aplicações com interfaces gráficas adaptáveis para dispositivos móveis**. Trabalho de Conclusão, Universidade Luterana do Brasil (Ulbra) – Curso de Sistemas de Informação – Campus Canoas, 2011.
- Revista Mobile Magazine 41: **Titanium Mobile: Aplicações Multiplataforma** Disponível em: <<http://www.devmedia.com.br/titanium-mobile-aplicacoes-multiplataforma-revista-mobile-magazine-41/24055>>. Acessado em: 20 set. 2012.
- Uol Notícias: **Android e iOS representam 85% do mercado de smartphones no 2º tri, diz consultoria IDC**. Disponível em: <<http://tecnologia.uol.com.br/noticias/redacao/2012/08/08/android-e-ios-representam-85-do-mercado-de-smartphones-do-2-tri-diz-consultoria-idc.htm>>, Acessado em: 20 set. 2012.