

Um estudo dos Sistemas Operacionais Android e iOS para o desenvolvimento de aplicativos

Vinícius Rafael Lobo de Mendonça¹, Thiago Jabur Bittar^{1,2}, Márcio de Souza Dias¹

¹Departamento de Ciência da Computação – Universidade Federal de Goiás (UFG)
CEP: 75704-020 – Catalão – GO – Brazil

²Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo (USP)
CEP: 13566-590 – São Carlos – SP – Brazil

vrlm.bytes@gmail.com, thiagojabur@gmail.com, marciosouzadias@gmail.com

Abstract. *Developing applications for mobile platforms is a labor market are little known by beginners and advanced programmers. This paper provides a comparative study of the major operating systems for mobile devices, Android and iOS. The goal is that the developer can compare the two mobile platforms and choose the one that best fits your profile. The mobile operating systems are compared in the following aspects: Liberty Development, Security, Application Availability, Fragmentation, Portability, Multithread and Open Source.*

Resumo. *O desenvolvimento de aplicativos para plataformas móveis é um mercado de trabalho pouco conhecido pelos programadores sejam iniciantes ou avançados. Este artigo fornece um estudo comparativo dos principais sistemas operacionais para dispositivos móveis: Android e iOS. O objetivo é que o desenvolvedor possa comparar as duas plataformas móveis e escolher aquela que encaixe melhor em seu perfil. Os sistemas operacionais móveis são comparados nos seguintes aspectos: Liberdade de Desenvolvimento, Segurança, Disponibilização de Aplicativos, Fragmentação, Portabilidade e Código Aberto.*

1. Introdução

O surgimento dos Sistemas Operacionais Móveis é um processo recente e consequência do aparecimento dos smartphones. Os *smartphones* são dispositivos móveis capazes de realizar grandes variedades de funções: filmar e reproduzir vídeos, acessar a internet banda larga por Wi-fi ou 3G, GPS, enviar e receber e-mails, ler e editar documentos em vários formatos etc [Fling 2009].

Os primeiros celulares comerciais foram lançados na década de 80, não sendo nada portáteis, custavam preços altos e possuíam apenas a capacidade de realizar ligações. No decorrer dos anos os celulares além de diminuir de tamanho acabaram agregando novas funções. Hoje os celulares realizam uma grande variedade de funções e são encontrados a preços cada vez mais acessíveis. Assim os aparelhos atingiram um novo estágio: os *smartphones* [Fling 2009]. Os celulares inteligentes como também são chamados, possuem um Sistema Operacional para gerenciamento de todos componentes do dispositivo móvel. Tais são classificados como Sistema Operacional para Dispositivos Móveis.

Na seção 2 é definido o conceito de Sistema Operacional Móvel e uma breve introdução sobre o Android e iOS. Na seção 3 é discutida a liberdade de desenvolvimento

e são mostradas as ferramentas que o desenvolvedor necessita para iniciar o desenvolvimento de aplicativos. a seção 4 é aprofundado o conceito de fragmentação. Na seção 5 é discutido o aspecto da portabilidade das aplicações. Na seção 6 é relacionado o estudo do código aberto nos sistemas operacionais. Na seção 7 aproveita-se para discutir a disponibilização de aplicativos nas lojas. Na seção 8 é discutido o aspecto da segurança em ambos os sistemas. A conclusão do artigo é realizada na seção 9.

2. Sistemas Operacionais Móveis

Um sistema computacional moderno é composto de vários componentes: um ou vários processadores, disco rígido, impressoras, teclado, monitor etc. Sistemas operacionais são softwares complexos e responsáveis pelo gerenciamento dos componentes fundamentais para funcionamento de um computador. Além de gerenciar, ele fornece uma interface gráfica que torna mais fácil a vida do usuário por facilitar sua interação com hardware. Os Sistemas Operacionais são classificados em vários tipos: Servidor, Grande Porte, Embarcado, Tempo Real etc. [Tanenbaum 2008]. Além desses tipos citados pode se incluir os Sistemas Operacionais para Dispositivos móveis.

Eles recebem essa classificação por executarem em aparelhos como *smartphones* e *tablets*. Esses sistemas não podem ser comparados com sistemas embarcados. Sistemas embarcados são mais simples e desenvolvidos para um objetivo em particular [Tanenbaum 2008]. Os dispositivos móveis oferecem novas formas de interação com o usuário (sensores, GPS, acelerômetro, teclados virtuais, *widjets* etc.) tornando seu desenvolvimento completamente diferente de projetar aplicações para computadores [Wasserman 2010]. Os principais sistemas operacionais móveis são: Android, iOS, Symbian, Windows Mobile e BlackBerry. O Android e iOS são as principais plataformas do mercado [Goadrich and Rogers 2011], no intuito de realizar uma melhor comparação foca-se no estudo nas duas plataformas citadas.

2.1. Android

O Android é um Sistema Operacional Móvel Open Source desenvolvido inicialmente pela Google e possui uma arquitetura baseada na versão 2.6 do kernel Linux para o controle das principais tarefas do sistema como segurança, gerenciamento de memória, gerenciamento de processos, pilha de rede e modelo de driver [GoogleInc 2011]. O kernel atua como uma camada de abstração entre o hardware e o resto da pilha de software. Posteriormente a responsabilidade do desenvolvimento foi transferida para *Open Handset Alliance (OHA)*.

Open Handset Alliance é um consórcio de grandes empresas com objetivo de popularizar e melhorar os dispositivos móveis. O Google é um dos membros do consórcio, continua responsável por controlar importantes etapas do desenvolvimento do sistema como a gerencia do produto e a engenharia de processos [OpenHeadsetAlliance 2011]. A arquitetura do Sistema Operacional Android é composta por cinco camadas[GoogleInc 2011]:

- *Applications*: A primeira camada e a mais perto do usuário é a dos aplicativos. Ela é composta pelos aplicativos nativos do Sistema Operacional Android como cliente de e-mail, programa de SMS, calendário, mapas, navegador etc.
- *Applications Frameworks*: Nesta camada encontramos os componentes pelo gerenciamento das Activitys, gerenciamento das Views, gerenciamento de Janelas,

Provedores de Conteúdos e etc. Estes componentes são manipulados pelos desenvolvedores durante o desenvolvimento.

- **Runtime:** A camada responsável pela execução dos aplicativos. Os aplicativos são escritos na linguagem Java, quando compiladas eles geram bytescodes Dalvik ao contrário de bytescodes JVM porque o sistema não possui Máquina Virtual Java (JVM). No Android está presente a Máquina Virtual Dalvik (DVM) desenvolvida para dispositivos móveis, ela é uma versão otimizada da JVM por possuir menos instruções.
- **Libraries:** Nesta camada encontramos diversas bibliotecas como a biblioteca C padrão, SQLite(Banco de Dados), OpenGL(Renderização 3D) etc.
- **Kernel:** A última camada e a de mais baixo nível é do kernel. O kernel atua como uma camada de abstração entre hardware e as camadas superiores, permitindo acesso a recursos como áudio, vídeo e protocolos de rede.



Figura 1. Arquitetura do Sistema Android [GoogleInc 2011]

2.2. iOS

IOs é a abreviatura para iPhone Operation System sendo desenvolvido pela Apple. O sistema foi baseado no Sistema Operacional MAC OS X e projetado para atender às necessidades de aparelhos móveis desenvolvidos pela Apple. O sistema realiza uma abstração entre a comunicação do hardware com o aplicativo. A arquitetura do iOS é composta por quatro camadas [AppleInc 2011]:

- **CocoaTouch:** Camada equivalente à camada de frameworks do Android. Esta camada fornece ferramentas e infraestrutura para implementar eventos e aplicações para a interface do iPhone.
- **Media:** A camada responsável por fornecer recursos áudio e vídeos. A tecnologia mais avançada para experiência multimídia e são encontradas bibliotecas como : OpenGL, QuartzCore etc.
- **Core Services:** A camada que fornece os serviços fundamentais do sistema como AdressBook, Core Location, CFNetwork, Security, SQLite etc.
- **Core OS:** Nesta camada que se encontra o kernel do sistema. Além do kernel, encontramos os drivers e as interfaces básicas do sistema.



Figura 2. Arquitetura do Sistema iOS [AppleInc 2011]

3. Liberdade de Desenvolvimento

O desenvolvimento de aplicações móveis é uma área que vem atraindo a atenção dos desenvolvedores. Os aplicativos móveis, em geral, são softwares relativamente pequenos e que necessitam de uma pequena equipe com um ou dois programadores responsáveis pelas etapas de desenvolvimento (concepção, projeto, implementação etc.). Além da vantagem de pequenas equipes de desenvolvimento, os sistemas operacionais móveis permitem a comunicação entre os aplicativos tornando assim o trabalho de programação menos trabalhoso.

Os aplicativos iOS são desenvolvidos na linguagem de programação Objective C que possui características da linguagem Smalltalk e da linguagem C. O programador iOS precisa preocupar com o gerenciamento de memória decorrente da linguagem de programação utilizada que é bem próxima da família de linguagem C. O desenvolvimento iOS é restrito porque o ambiente de desenvolvimento é possível apenas em computadores Mac rodando Mac OS X. A quantidade de Macs é inferior ao número de computadores que possuem outros SOs instalados [Goadrich and Rogers 2011].

O Ambiente de Desenvolvimento Integrado (*IDE*) utilizado é o XCode, ambiente padrão da Apple para o gerenciamento de projetos relacionados ao Sistema Operacional Mac OS X. O programador também deverá realizar download e instalar o kit de desenvolvimento de software (*SDK*), para configurar o *IDE* e iniciar o desenvolvimento. O *SDK* é composto por várias ferramentas como emulador para simulação de um dispositivo e o construtor de interface.

O desenvolvimento do Android ao contrário do iOS é multiplataforma podendo ser desenvolvido em qualquer dos principais sistemas operacionais: Windows(XP ou superior), Mac OS X (10.5.8 ou superior) e sistemas Linux(rodando com kernel 2.6 ou superior). Isto proporciona um maior alcance de programadores e a possibilidade de desenvolvimento independente do sistema. Além da flexibilidade de Sistema Operacional, o programador pode usar o *IDE* de sua preferência para o desenvolvimento. Por exemplo: Netbeans, Eclipse etc.

O Eclipse é o ambiente de desenvolvimento recomendado pelo Google devido a existência de um *plug-in* que facilita a vida do programador Android. Com uso do *plug-in*, Android Development Tools (*ADT*), os desenvolvedores podem utilizar essa *IDE* para simplificar a criação de projetos e proporcionar vários benefícios como acesso rápido ao

SDK, e um ambiente de depuração integrado [Goadrich and Rogers 2011] . A linguagem utilizada na programação dos aplicativos é o Java em conjunto com a linguagem de marcação Extensible Markup Language(*XML*) . *XML* é utilizado para projetar as interfaces gráficas de usuário (*GUI*). A abstração da comunicação entre as duas linguagens é feita por meio do arquivo *R.java* existente no projeto da aplicação.

4. Fragmentação

A fragmentação é a medida utilizada para determinar o nível de desintegração de um sistema operacional quanto variedade de versões e dispositivos móveis suportados. O Android é um sistema operacional apoia uma variedade de dispositivos móveis com diferentes resoluções de tela, hardware, tamanho físico, profundidade de cor e métodos de entradas [Charland and LeRoux 2011].

O sistema então é proprietário de um grande nível de fragmentação, o programador precisa levar em conta esses vários fatores na hora do desenvolvimento, ao disponibilizar seu aplicativo para *download* um usuário pode negá-lo porque seu aplicativo não funcionou perfeitamente ou simplesmente porque nem executou. Outra condição que precisa ser levado em conta na hora do desenvolvimento é a variedade de versões do Sistema. Por exemplo um aplicativo desenvolvido utilizando o *SDK* 2.1, ele executará apenas na versão 2.1 ou superiores do Android.

O sistema iOS não possui tanta fragmentação quando comparado a outros sistemas operacionais móveis sendo executado em três dispositivos móveis diferentes (IPod Touch, iPhone e iPad) número bem inferior quando comparado variedade de dispositivos que possuem o Android instalado[2010]. A pouca quantidade de aparelhos é resultado da política da Apple em focar na fabricação de aparelhos com excelente *design* e qualidade.

5. Portabilidade

A fragmentação de um Sistema Operacional é um obstáculo durante o desenvolvimento. Qual versão ? Qual resolução de tela ? Entre outras perguntas habitam na mente do desenvolvedor de durante a programação do seu aplicativo. A variedade de versões e dispositivos móveis executando Android mais a ausência de método que simplifique o desenvolvimento, contribui para tornar o trabalho do programador oneroso.

Apesar do lançamento da versão 3.0 e superiores específica para tablets, os aplicativos desenvolvidos para smartphones com Android continuam a demonstrar problemas quando executados em dispositivos com telas maiores desconfigurando assim *layout* das aplicações.

A Google, empresa responsável pelo sistema, possui a intenção de lançar uma versão que agrega melhores recursos da versão para *smartphones*(2.0 ou superiores) com a versão para *tablets*: a versão 4.0. O objetivo principal é solucionar o problema da fragmentação e facilitar portabilidade de aplicativos entre diferentes dispositivos [GoogleInc 2011].

O iOS desfruta de uma interface de comunicação entre o *hardware* e o *software*. Essa interface possibilita na medida do possível que a aplicação execute corretamente em diferentes dispositivos, simplificando o trabalho do desenvolver que não precisa preocupar com a fragmentação, mesmo que pequena, do sistema [Wasserman 2010].

6. Código Aberto

A principal vantagem do Android em relação ao iOS é ser baseado em Linux, sistema operacional de código aberto [GoogleInc 2011]. O kernel Linux permite que várias aplicações possam ser executadas simultaneamente, permitindo que aplicações executem em *background* sem que o usuário perceba. O sistema Operacional Android pode ser considerado um sistema *multithread* porque além dos aplicativos nativos o sistema permite que aplicativos de terceiros executem ao mesmo tempo.

No sistema iOS não permite que um aplicativo de terceiro execute em *background* ao minimizar uma aplicação está na verdade finalizando o processo relacionado. O usuário ao acessar novamente a aplicação não encontrará no último estado de utilização, mas sim no estado inicial. Os aplicativos nativos são componentes do sistema operacional, enquanto aplicativos de terceiros são publicados na loja de aplicativos pelos desenvolvedores. A plataforma móvel *open source* é a possibilidade de customização, o desenvolvedor possui acesso ao código do sistema operacional podendo assim realizar modificações no sistema como alterar telas, ícones, *design* da lista de contatos etc.

O desenvolvedor iOS não possui acesso ao código fonte do sistema operacional, não podendo assim realizar alterações. O código aberto permite também que a comunidade de desenvolvedores examinem o código fonte e detectem falhas ao longo do tempo contribuindo para melhorar o sistema operacional [Nachenberg 2011].

7. Disponibilização dos Aplicativos

A distribuição de softwares no Sistemas Operacionais ocorre de forma centralizada, grandes e pequenos desenvolvedores disponibilizam seus aplicativos na loja de aplicativos [Gilbert et al. 2011]. Os aplicativos estão disponíveis para download por meio de lojas aplicativos específicas para cada plataforma móvel, sendo disponibilizados pelos desenvolvedores de forma paga ou gratuita. A loja de aplicativos do Android é chamada de Android Market, os desenvolvedores que desejarem disponibilizar seus aplicativos na loja deverão realizar um cadastro como desenvolver e pagar uma taxa única para assim realizar sua inscrição. Aceito o seu cadastro, o desenvolvedor ao enviar seu aplicativo deverá fornecer informações relacionados como versões do sistema suportada, o tipo do aplicativo e a forma de pagamento [GoogleInc 2011].

O Google não fatura nada sobre cada aplicativo pago vendido, 30% é da operadora móvel e os 70% restante é do desenvolvedor. A divisão de lucros pode ser alterada de acordo com os objetivos traçados pelo programador. Por exemplo, o desenvolvedor fornece parte do seu lucro para operadora que acaba por fazer publicidade do seu aplicativo no mercado aplicativo. O processo de cadastro de desenvolvedor iOS é semelhante ao desenvolvedor do Android mas a divisão de lucros é diferente. A operadora não fatura nada com a venda de aplicativos, os 30% acabam ficando com a Apple.[AppleInc 2011]

8. Segurança

A segurança dos sistemas operacionais é processo iniciado na aceitação da disponibilização dos aplicativos para o usuário. A quantidade de aplicativos aceitos mostram que estes testes realizados não asseguram que o usuário esteja instalando um aplicativo seguro em seu aparelho, então é necessário que ele tome preocupações como evitar usufruir de aplicativos disponíveis na internet [Gilbert et al. 2011].

O desenvolvedor durante a programação do seu projeto fornece as permissões necessárias para funcionamento do seu aplicativo, seja ele desenvolvido no Android ou no iOS. Por exemplo, se um aplicativo acessa dados pessoais ou informações sigilosas para os quais não foi concedida a permissão, estamos diante de um caso de violação de segurança.

A violação ocorre quando um aplicativo realiza alguma ação além das permissões definidas pelo desenvolvedor [Gilbert et al. 2011]. As permissões de uso dos recursos do sistema pelo aplicativo não é oculta para os usuários. Depois realizar o download da aplicação, o usuário ao instalar será notificado das permissões e ao concordar com a instalação, estará assim aceitando todos os termos. O número crescente de usuários das plataformas móveis acabou chamando à atenção dos *hackers*. Termos como antivírus, *malware* e *hackers* que anteriormente estavam relacionados apenas ao mundo dos computadores pessoais acabaram tornando frequentes no mundo dos dispositivos móveis.

A segurança no Android é herdada do kernel Linux que permite cada aplicativo execute dentro de sua própria máquina virtual, isolada assim das outras aplicações. Tal isolamento impede que uma ameaça se espalhe pelo sistema de forma direta, mas ainda existe o fato que uma aplicação pode requerer comunicação com outra aplicação para executar uma ação como acessar a caixa de entrada do e-mail [Nachenberg 2011].

A possibilidade da realização do download de aplicativos em computadores e posteriormente realizar a transferência dos aplicativos para seu dispositivo móvel. O Android no intuito de proteger o sistema não permite a instalação de aplicativos de fontes desconhecidas, mas esta opção pode ser alterada pelo usuário mas ele é notificado dos possíveis problemas futuros decorrentes deste ato.

O iOS libera o acesso ao mercado de aplicativos apenas pelos aparelhos. O processo de certificação da aplicação na sua loja de aplicativos persiste entre 1 ou 2 semanas, sendo aceito a etapa seguinte é a criação da assinatura digital pelo desenvolvedor. A assinatura digital é a garantia que autor do aplicação foi aprovado pela Apple e que seu aplicativo não pode ser adulterado. No caso de violação de segurança facilita a identificação do culpado. O processo de aceitação de aplicativos na Android Market permitindo os desenvolvedores gerem seus próprios certificados de assinatura, essa liberdade pode ser aproveitada pelos hackers para criação de perfis falsos [Nachenberg 2011].

9. Conclusão

Os *smatphones* revolucionaram o mercado de desenvolvimento. Antes da chegada dos *smartphones*, os dispositivos móveis eram relativamente esquecidos pelos desenvolvedores devido às várias limitações de seu *hardware* e *software*. A chegada dos *smartphones* acabou modificando esse quadro e introduzindo aparelhos poderosos no mercado. Além de *hardware* poderoso, eles portavam sistemas operacionais classificados como móveis e novas formas de interação com o usuário.

No mercado existem vários sistemas operacionais móveis mas neste artigo o foco era nos dois principais: iOS e Android. Na tabela(ver Tabela1) os dois sistemas operacionais são confrontados em propriedades altercadas em tópicos anteriores do trabalho, para que o desenvolvedor possa a partir dessas características escolher pelo sistema que melhor encaixa em seu conjunto de habilidades.

Tabela 1. Comparação do Android e o iOS

Característica	Android	iOS
Liberdade de Desenvolvimento	X	
Emulador e Debugador no SDK	X	X
Facilidade de Portabilidade		X
Código Aberto	X	
Construtor de Interface	X	X
IDE	Eclipse, Netbeans etc.	XCode
Linguagem de programação	Java com XML	Objective-C
Nível de Fragmentação	Alto	Baixo
Nível de Segurança	Baixo	Alto

Referências

- (2010). Introduction to processing on android devices. In *ACM SIGGRAPH ASIA 2010 Courses*, SA '10, pages 25:1–25:137, New York, NY, USA. ACM.
- AppleInc (2011). ios overview. http://developer.apple.com/library/ios/referencelibrary/GettingStarted/URL_iPhone_OS_Overview/_index.html//apple_ref/doc/uid/TP40007592, Acessado em: 13 de maio de 2011.
- Charland, A. and LeRoux, B. (2011). Mobile application development: Web vs. native. *Queue*, 9:20:20–20:28.
- Fling, B. (2009). Mobile design and development. pages 01–12, 1005 Gravenstein Highway North, Sebastopol, CA 95472. O'Reilly Media Inc.
- Gilbert, P., Chun, B.-G., Cox, L. P., and Jung, J. (2011). Vision: automated security validation of mobile apps at app markets. In *Proceedings of the second international workshop on Mobile cloud computing and services*, MCS '11, pages 21–26, New York, NY, USA. ACM.
- Goadrich, M. H. and Rogers, M. P. (2011). Smart smartphone development: ios versus android. In *Proceedings of the 42nd ACM technical symposium on Computer science education*, SIGCSE '11, pages 607–612, New York, NY, USA. ACM.
- GoogleInc (2011). What is android? <http://developer.android.com/guide/basics/what-is-android.html>, Acessado em 06 de maio de 2011.
- Nachenberg, C. (2011). Examining the security approaches employed in apple's ios and google's android. Mountain View, CA 94043 USA. Symantec Corporation, Symantec Corporation.
- OpenHeadsetAlliance (2011). Overview. http://www.openhandsetalliance.com/oha_overview.html, Acessado em 05 de maio de 2011.
- Tanenbaum, A. S. (2008). Sistemas operacionais modernos. pages 1 – 15. Pearson Prentice Hall. tradução Ronaldo A.L. onçalves, Luís A. Consularo ; revisão técnica Regina Borges de Araujo.
- Wasserman, A. I. (2010). Software engineering issues for mobile application development. In *Proceedings of the FSE/SDP workshop on Future of software engineering research*, FoSER '10, pages 397–400, New York, NY, USA. ACM.