

# **Plataforma Android**

**Elder Elisandro Semberger<sup>1</sup>, Ivonei Freitas<sup>1</sup>, Ramiro Vani<sup>2</sup>**

<sup>1</sup>UNIOESTE - Universidade Estadual do Oeste do Paraná  
Rua Universitária, 2069. Jardim Universitário.  
Caixa Postal 711 - CEP 85819-110 – Cascavel, PR  
[elderres@gmail.com](mailto:elderres@gmail.com)  
[ivonei@hotmail.com](mailto:ivonei@hotmail.com)

<sup>2</sup>UNIVEL - União Educacional de Cascavel  
Av. Tito Muffato, 2317. Bairro Santa Cruz.  
CEP: 85.806-080 – Cascavel, PR  
[ramirovani@hotmail.com](mailto:ramirovani@hotmail.com)

*Resumo.* resumo.

## **1. Introdução**

O mercado de celulares vem crescendo de maneira bastante rápida. Em 2005, havia dois bilhões de usuários [1], já no final de 2008 este número chegou a quatro bilhões de pessoas que usam aparelho celular [2][3].

Diferentemente do inicio da proliferação destes dispositivos, é crescente o número de consumidores que buscam mais suporte e novas funcionalidades, como telas grandes, variados recursos multimídia, conectividade, jogos e aplicativos específicos. Esse crescimento é consequência do aumento de usuários que buscam também em dispositivos móveis o lazer (jogos), a comodidade (ver extrato da conta bancária, previsão do tempo, fazer compras, controlar equipamentos em suas residências), ou ainda pela necessidade, como enviar um pedido ao fornecedor ou um orçamento a um cliente [3].

Para acompanhar tal crescimento, é importante que empresas fabricantes de dispositivos e aplicativos para móveis busquem ou desenvolvam novas tecnologias a fim de atender a demanda e conquistar a preferência de operadoras e usuários.

Nesse contexto de inovações encontra-se a Plataforma Android.

Este artigo objetiva apresentar a motivação para o surgimento desta nova plataforma, as características básicas de funcionamento, como “preparar” a IDE Eclipse para implementar aplicativos compatíveis e, finalmente, uma aplicação exemplificando o uso da mesma, em detalhes.

## **2. A Plataforma Android**

Quando era esperado que a Google, uma das maiores empresas de tecnologia do mundo, estivesse próxima de lançar um telefone celular veio a surpresa com o lançamento de algo que vai além de um simples dispositivo: Uma plataforma completa para dispositivos móveis, batizada de Android.

Android surgiu da parceria da Google com a *Open Handset Alliance (OHA)*, aliança onde figuram os principais atores do mercado móvel mundial, conforme a Tabela 1 (atualizada em 01 de junho de 2009):

Operadoras Móveis	Empresas de Software	Empresas de Comercialização	Industria de Semicondutores	Fabricantes de Celulares
China Mobile	Ascender	Aplix	Audience	HTC
China Unicom	Corporation	Noser	Broadcom Corporation	LG
KDDI	eBay	Engineering	Intel Corporation	Motorola
Corporation	Esmertec	The Astonishing	Marvell Technology	Samsung
NTT DoCoMo	Google	Tribe	Group	Electronics
Sprint Nextel	LivingImage	Wind River	Nvidia Corporation	ASUSTek
T-Mobile	NMS	Systems	Qualcomm	Garmin
Telecom Italia	Communications	Omron Software	SiRF Technology	Huawei
Telefónica	Nuance	Teleca	Holdings	Technologie
Vodafone	Communications		Synaptics	s
Softbank	PacketVideo		Texas Instruments	Sony
Ericsson	SkyPop		AKM Semiconductor	Ericsson
	SONiVOX		ARM	Toshiba
	Borqs		Atheros Communications	Acer
			EMP	

**Tabela 1.** Membros da *Open Handset Alliance* [4]

## 2.1 O projeto Android

Como supracitado, o projeto Android trata-se de uma plataforma completa para dispositivos móveis, e inclui Sistema Operacional (Kernel GNU Linux – versão 2.6), *Middleware* e aplicações.

Conta também com um SDK que disponibiliza as ferramentas e APIs necessárias para o desenvolvimento na plataforma Android, usando a linguagem JAVA [5].

As principais funcionalidades da plataforma são [3][5]:

- **Framework de desenvolvimento de aplicações:** Reutilização de código e facilidade de acesso a recursos exclusivos e manutenção
- **Nova máquina virtual (dalvik):** Criada e otimizada para dispositivos móveis e suas limitações
- **Navegador web integrado:** Baseado no projeto *open source webkit* [6] – o mesmo do iPhone e Nokia série 60
- **Biblioteca de gráficos otimizada para dispositivos móveis:** Exclusiva biblioteca para gráficos 2d e 3d baseada na especificação OpenGL ES 1.0 [7], com aceleração por hardware como opcional
- **SQLite:** armazenamento de dados estruturados
- **Supporte multimídia:** compatibilidade com os principais formatos existentes, entre eles mpeg4, h.264, mp3, aac, amr, jpg, png e gif

- **Telefonia com tecnologia GSM:** As aplicações podem manipular operações telefônicas, caso o fabricante permita esse acesso
- **Bluetooth, EDGE, 3G e WiFi:** foco nas principais tecnologias de transmissão de dados sem fio, também depende da permissão do fabricante para acesso
- **Câmera e GPS:** Ter o celular como uma ferramenta para interação com redes sociais, também dependente da permissão do fabricante para acesso
- **Ambiente de desenvolvimento com plugin para Eclipse [8]:** Inclui emulador, ferramentas para *debug* e supervisão de memória e desempenho

## 2.2 Arquitetura

A arquitetura da plataforma Android é dividida em quatro camadas: Kernel GNU Linux, bibliotecas, *framework* para aplicações e as próprias aplicações – além da porção *runtime*, necessária para a execução dos aplicativos no dispositivo. A Figura 1 ilustra tais camadas conforme a posição em que se encontram.

A camada mais baixa da arquitetura, Kernel Linux, é a responsável por gerenciar os processos, *threads*, arquivos, pastas além de redes e *drivers* dos dispositivos. É responsável por gerenciar todos os processos e a memória.



Figura 1. A arquitetura da Plataforma Android

## 2.3 Máquina virtual

A base de desenvolvimento do Android é Java. Mas no seu sistema operacional não existe uma Máquina Virtual Java (JVM) e sim outra máquina virtual, a Dalvik, otimizada para interpretar códigos em dispositivos móveis [3].

Ao desenvolver aplicações para o Android, utiliza-se a linguagem Java e todos os recursos que esta oferece. Mas, ao compilar o bytecode (.class), este é convertido para o formato específico da máquina virtual Dalvik, o .dex (*Dalvik Executable*), que representa a aplicação do Android compilada [3][5].

De modo análogo aos aplicativos em Java, depois da aplicação Android ter sido compilada, juntamente com outras bibliotecas e possíveis imagens utilizadas é gerado um arquivo único, neste caso com a extensão .apk (Android Package File), que é a aplicação pronta para ser instalada [3].

Se o ambiente de desenvolvimento utilizado for o Eclipse [8], toda essa compilação e geração do arquivo .apk acontece automaticamente.

#### 2.4. Código aberto e livre

O Google Android é a primeira plataforma para aplicações móveis completamente livre e de código aberto (*open-source*) [3].

Para operadoras, fabricantes e usuários avançados isso é muito vantajoso, pois é possível utilizar gratuitamente o sistema operacional além de facilidade em personalizá-lo.

A licença de uso é a *Apache Software Foundation* (ASF) [ref], que permite que alterações efetuadas no código-fonte não sejam obrigatoriamente compartilhadas.

O código fonte do Android está disponível no site do projeto: <http://source.android.com>.

### 3. Configurando o Ambiente

Esta seção aborda apresenta os procedimentos necessários para instalação das ferramentas e *plugins* necessários para desenvolver para Android.

#### 3.1. Requisitos de Software e Sistema

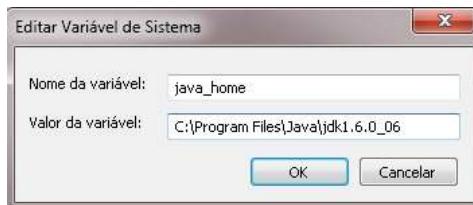
O SDK é suportado nos seguintes sistemas operacionais:

- Windows XP ou Vista
- Mac OS X 10.4.8 ou posterior (somente X86)
- Qualquer distribuição Linux

Os ambientes de desenvolvimento suportados são:

- Eclipse 3.3 ou 3.4, utilizando o *plugin Android Development Tools* (ADT) para integrar o emulador ao Eclipse
- JDK5 ou JDK6 (Apenas a JRE não é o suficiente)

É preciso configurar corretamente a variável de ambiente Java\_Home, apontando para o caminho do JDK, conforme Figura 2:



**Figura 2.** Configuração Java\_Home

variável de ambiente

#### 3.2. Instalação do Eclipse

Para instalação do Eclipse é necessário baixar o arquivo compactado [8].

O Eclipse não é um arquivo instalável. Depois de baixado, deve-se descompactar em qualquer diretório. Neste arquivo há o executável chamado *eclipse.exe*.

### 3.3. Instalação do SDK

Assim como o Eclipse, o SDK [9] não é um arquivo instalável. Ao baixar, basta descompactá-lo em qualquer diretório. A versão utilizada neste artigo foi a 1.1.

### 3.4. Instalação do plugin para o Eclipse

A instalação do plugin para o ambiente Eclipse é feito da seguinte forma:

1. Inicie o Eclipse;
2. Menu Help > Software Updates
3. Em *Software Updates and Add-ons*, selecione Available Software e clique no botão Add Site, insira a URL <http://dl-ssl.google.com/android/eclipse/site.xml> e clique em OK.
4. Marque os *plugins* do Android e clique em *install*, como mostra a Figura 3.

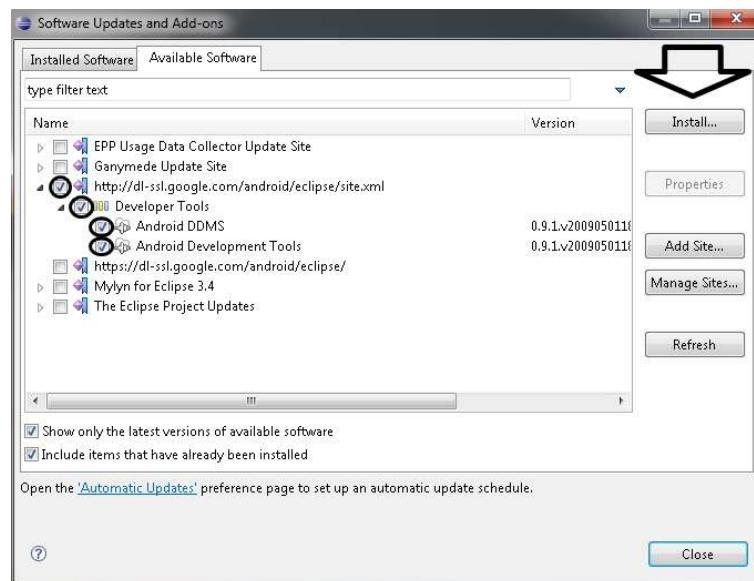
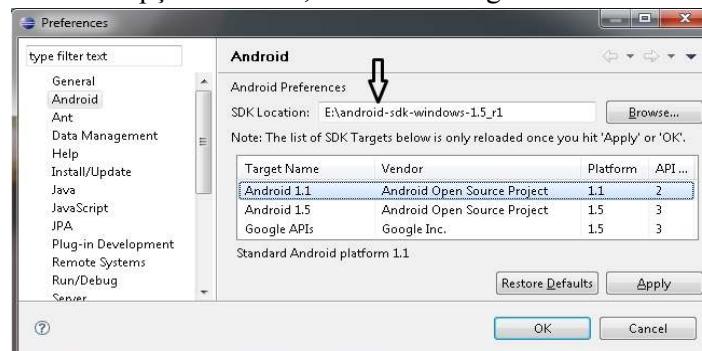


Figura 3. Instalação do plugin do Android para o Eclipse

Basta reiniciar o Eclipse e o plugin do Android já estará disponível.

### 3.5. Configurando o Eclipse

É necessário apontar o caminho do SDK. Para fazer isso, basta clicar em Window > Preferences e selecionar a opção Android, conforme a Figura 4 mostra:



**Figura 4.** Configuração do caminho do SDK

Após confirmar estas configurações, o ambiente Eclipse está pronto para criar aplicações Android.

## 4. Uma aplicação Android

Antes de mostrar uma aplicação Android, é preciso conhecer e entender a anatomia de aplicações para esta plataforma, já que estas não seguem os formatos conhecidos de aplicações Java ou de um Midlet para Java ME [11].

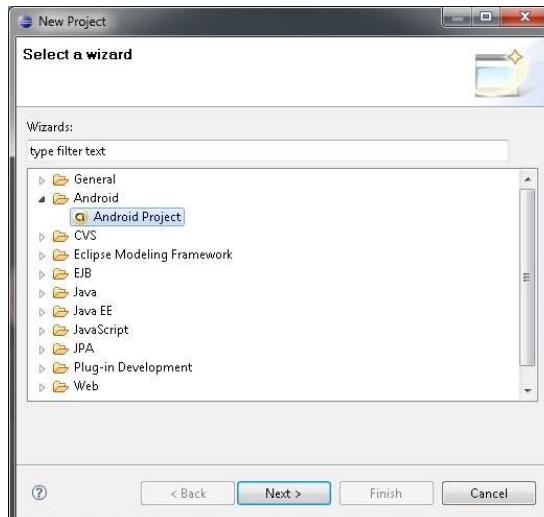
A aplicação é dividida em quatro módulos [3] [5] [11]:

- **activity**: É o módulo mais comum de uma aplicação Android. Refere-se a uma tela da aplicação e é implementada como uma única classe que deriva da classe base *activity*. Essa classe deve exibir uma interface com o usuário e tratar os eventos a ela relacionados
- **intent receiver**: Usado quando se deseja que a aplicação reaja a algum evento externo (o telefone tocar, Internet disponível ou um alarme). Este módulo não exibe interface com o usuário, mas pode através da classe *notificationmanager* avisar o usuário que tal evento ocorreu
- **service**: Um *service* é o código que é executado durante toda a aplicação, sem a necessidade de uma interface com o usuário (tocadores de música ou players de vídeo). Após o usuário escolher qual música deseja ouvir ele não quer que sua música pare por causa de algum outro evento
- **content provider**: Uma aplicação pode armazenar dados através de uma base SQLite, por exemplo [3][10]. Este módulo permite que as diversas aplicações no dispositivo compartilhem informações

A aplicação qualquer não precisa necessariamente implementar todos os módulos, mas provavelmente ela terá uma combinação entre dois ou mais deles.

### 4.1 Criando a Aplicação Android

Para criar uma aplicação Android basta clicar em File > New > Project, na próxima janela, escolha Android e depois clique em *next* (Figura 5).



**Figura 5.** Criação de um projeto Android

Deve-se inserir os dados conforme segue e clique em *finish* conforme segue, e seu projeto estará criado:

- **Project Name:** ListaCompras. Este é o nome do projeto que vai ser criado no Eclipse
- **Package Name:** br.univel.exemplos. Nome do pacote que contém a Activity (Tela principal). O pacote é muito importante no Android porque é utilizado como identificador para a aplicação
- **Activity Name:** ListaDeCompras. Nome da activity principal do projeto. Uma activity representa uma tela e é o ponto inicial de uma aplicação
- **Application Name:** SuperMercado. É apenas o título da aplicação

O conteúdo do arquivo `strings.xml` deve estar da forma como indica o Código XML 1:

---

#### Código XML 1 strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="titulo">Lista de Compras</string>
    <string name="app_name">SuperMercado</string>
</resources>
```

O arquivo `main.xml`, que é a tela da aplicação, deverá estar com setado com os parâmetros conforme mostrado no Código XML 2 `main.xml`:

---

#### Código XML 2 main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/titulo" />
    <CheckBox
        android:text="LEITE EM PÓ - - - R$ 8.90"
        android:id="@+id/checkLeitePo"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" >
    </CheckBox>
    <CheckBox
        android:text="LEITE EM CX - - - R$ 1.90"
        android:id="@+id/checkLeiteCx"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" >
    </CheckBox>
    <Button
        android:text="OK"
        android:id="@+id/btnOK" />
</LinearLayout>
```

---

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content" >
    </Button>
    <TextView
        android:text="Itens Comprados"
        android:id="@+id/TextView01"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" >
    </TextView>
    <TextView
        android:text=""
        android:id="@+id/comprados"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" >
    </TextView>
</LinearLayout>
```

---

Ao inserir uma variável no arquivo `main.xml`, uma referência é criada para ela no arquivo `R.java`. Esse arquivo funciona como uma espécie de interface entre o código java e o código xml das telas. Esse arquivo jamais deve ser alterado manualmente.

O código JAVA do aplicativo exemplo pode ser visto no Algoritmo 1:

---

### **Algoritmo 1 ListaDeCompras**

```

package br.exemplo.univel;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.TextView;

public class ListaDeCompras extends Activity {
    int comprados;
    double total;
    @Override
    public void onCreate(Bundle icicle) {
        super.onCreate(icicle);
        setContentView(R.layout.main);
        final CheckBox leitepo = (CheckBox) findViewById(R.id.checkLeitePo);
        final CheckBox leitecx = (CheckBox) findViewById(R.id.checkLeiteCx);
        final Button button = (Button) findViewById(R.id.btnOK);
        final TextView tx = (TextView) findViewById(R.id.comprados);

        leitepo.setOnClickListener(new Button.OnClickListener() {
            public void onClick(View arg0) {
                if (leitepo.isChecked()){
                    comprados++;
                    total = total + 8.90;
                    tx.setText(comprados + " | " + "Total" + " | R$ " + total );
                } else {
                    comprados--;
                    total = total - 8.90;
                    tx.setText(comprados + " | " + "Total" + " | R$ " + total );
                }
            }
        });

        leitecx.setOnClickListener(new Button.OnClickListener() {
            public void onClick(View arg0) {
                if (leitecx.isChecked()){
                    comprados++;
                    total = total + 1.90;
                    tx.setText(comprados + " | " + "Total" + " | R$ " + total );
                } else {

```

---

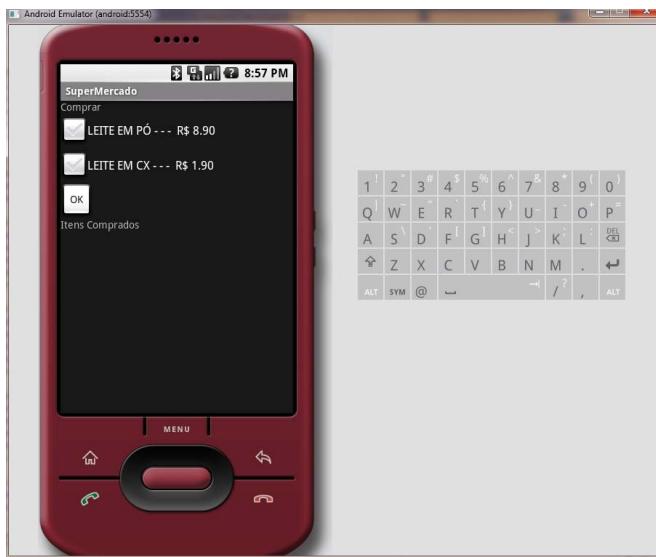
```

        comprados--;
        total = total - 1.90;
        tx.setText(comprados + " | " + "Total" + " | R$ " + total );
    }
});

button.setOnClickListener(new Button.OnClickListener(){
    public void onClick(View arg0) {
        if (leitepo.isChecked() && leitecx.isChecked()){
            tx.setText("Compra terminada \n" + comprados + " " + "Itens \n" +
                    "R$" + " " + total + " " + "gasto" );
        }else{
            tx.setText("Você ainda não comprou tudo");
        }
    }
});
}
}

```

Conforme o código e demais configurações supra mostradas, ao ser executada a aplicação, poderá ser visto no emulador a aplicação rodando, conforme a Figura 6:



**Figura 6.** Emulador rodando o aplicativo supra exemplificado para a plataforma Android.

## 5. Referências

- [1]Artigo do Portal Terra. <http://tecnologia.terra.com.br/interna/0,,OI672724-EI4796,00.html>. Acesso em 04/06/2009.
- [2] Artigo Info Abril. <http://info.abril.com.br/aberto/infonews/092008/25092008-21.shtml>. Acesso em 04/06/2009.
- [3] LECHETA, R. **Aprenda a criar aplicações para dispositivos móveis com o Android SDK**. São Paulo, Novatec, 2009.
- [4] OPEN HANDSET ALLIANCE. <http://www.openhandsetalliance.com/> Home>Members. Acesso em 06/06/2009.
- [5] EDUARDO SPINOLA. Android – A nova plataforma móvel. Artigo publicado na DevMedia (link nos favoritos)
- [6] The WebKit Open Source Project. <http://webkit.org/>. Acesso em 07/06/2009.

- [7] The Standard for Embedded Accelerated 3D Graphics. Kronos Group. [http://www.khronos.org/opengles/1\\_X/](http://www.khronos.org/opengles/1_X/). Acesso em 02/06/2009.
- [8] Projeto Eclipse. <http://www.eclipse.org/>. Acesso em 02/06/2009/
- [9] Download do SDK. [http://developer.android.com/sdk/download.html?v=android-sdk-windows-1.5\\_r1.zip](http://developer.android.com/sdk/download.html?v=android-sdk-windows-1.5_r1.zip). Download em 05/05/2009.
- [10] Peña J. B. **Estudio de La Plataforma Android**. Dissertação de Mestrado. Orientador Antoni Oller. 24/07/2008. Universitat Politècnica de Cataluya
- [11] Mangas D. C. **Desenvolupament d'aplicacions lliures pera telèfons mòbils usant la plataforma Android**. Dissertação de Mestrado. Orientador Antoni Soto Riera. 22/01/2009.