

Desenvolvimento de aplicativos para Android

Bruno França

bfmonteiro@gmail.com

23 de Agosto de 2012

Sumário

Introdução

Conceitos

Hello, World

Menus

Exercício

Conclusão



Android

- ▶ Linux

Android

- ▶ Linux
- ▶ Open source

Android

- ▶ Linux
- ▶ Open source
- ▶ Google e Open Handset Alliance

Apresentação do Google Play [SAIBA MAIS](#)

Pesquisar



LOJA DE APlicATIVOS ANDROID

MEUS APlicATIVOS ANDROID

JOGOS OLÍMPICOS
2012

Aplicativos oficiais

POPULARES

CATEGORIAS

SUGESTÕES DA EQUIPE

NOSSAS SUGESTÕES PARA TABLET

Principais aplicativos pagos

- 1** Where's My Water?
DISNEY
- 2** Smart Tools - Ferramentas
SMART TOOLS CO.
- 3** PES 2012 Pro Evolution Soc...
KONAMI DIGITAL ENTERTAINMENT...
- 4** Angry Birds Space Premium
ROVIO MOBILE LTD.



Eurosport.com

EUROSPORT

★★★★★ (14.666)

INSTALAR



Shoot The Zombirds

INFINITE DREAMS

★★★★★ (15.071)

INSTALAR



KAYAK

KAYAK.COM

★★★★★ (42.421)

INSTALAR



JAZZ: Trump's Journey

BULKYPIX

★★★★★ (1.644)

INSTALAR



Endomondo Sports T...

ENDOMONDO

ESCOLHA DOS EDIT...



Brightwood Adventur...

KIWI, INC.

★★★★★ (23.911)



Workout Trainer

SKIMBLE INC.

★★★★★ (28.550)



Defender II

DROIDHEN

★★★★★ (91.762)

Google Search

Google Inc.

PRINCIPAL DESENVOLVIMENTO



★★★★★ (150.493)

INSTALADO

Este aplicativo é compatível com seu Vivo Motorola MB860.

Outros aplicativos deste desenvolvedor

**Gmail**

GOOGLE INC.

★★★★★ (390.376)

Gratuito

**Maps**

GOOGLE INC.

ESCOLHA DO EDITOR

Gratuito

**Street View In Google Maps**

GOOGLE INC.

★★★★★ (336.555)

Gratuito

**Voice Search**

GOOGLE INC.

★★★★★ (161.109)

Gratuito

[Veja mais >](#)

VISÃO GERAL

COMENTÁRIOS DOS USUÁRIOS

O QUE HÁ DE NOVO

PERMISSÕES

Permissões

ESTE APPLICATIVO TEM ACESSO A:

SUAS CONTAS

USAR AS CREDENCIAIS DE AUTENTICAÇÃO DE UMA CONTA

Permite que o aplicativo solicite tokens de autenticação.

GERENCIAR A LISTA DE CONTAS

Permite que um aplicativo execute operações como adição e remoção de contas e exclusão de senhas.

SUA LOCALIZAÇÃO

LOCAL APROXIMADO (COM BASE NA REDE)

Acessa fontes de localização aproximada, como o banco de dados de rede do celular, para determinar uma localização aproximada do tablet, quando disponível. Aplicativos maliciosos podem usar esse recurso para determinar aproximadamente onde você está. Acessa fontes de localização aproximada, como o banco de dados do celular, para determinar uma localização aproximada do celular, quando disponível. Aplicativos maliciosos podem usar esse recurso para determinar aproximadamente onde você está.

LOCALIZAÇÃO PRECISA (GPS)

Acessa fontes de localização finas, como o Sistema de Posicionamento Global no tablet, quando disponível. Aplicativos maliciosos podem usar esse recurso para determinar onde você está, e esse uso pode consumir energia da bateria. Acessa fontes de localização finas, como o Sistema de Posicionamento Global no celular, quando disponível. Aplicativos maliciosos podem usar esse recurso para determinar onde você está, e esse uso pode consumir energia da bateria.

COMUNICAÇÃO DA REDE

ACESSO TOTAL À INTERNET

Permite que o aplicativo crie sockets de rede.

SUAS INFORMAÇÕES PESSOAIS

LER DADOS DO CONTACTO

Android: motivação

- ▶ Plataforma mais popular em smartphones¹
- ▶ Aplicações nativas e de terceiros são escritas com as mesmas APIs e rodam no mesmo ambiente
- ▶ Desenvolvimento sem custos e multiplataforma
- ▶ Documentação, exemplos e código fonte disponíveis

¹<http://www.prweb.com/releases/2012/5/prweb9514037.htm>

Por onde começamos?

SDK

(Android Software Development Kit)

SDK

(Android Software Development Kit)

- ▶ API
- ▶ Ferramentas
- ▶ Emulador
- ▶ Documentação
- ▶ Código fonte
- ▶ Exemplos
- ▶ ...



Developers

Design

Develop

Distribute



Jelly Bean now available!

Android 4.1 (Jelly Bean) builds on what's great about Android with improvements to performance and user experience.

New APIs are also available that allow you to build richer and more interactive notifications, transfer larger payloads through NFC, discover services over Wi-Fi, and much more.

[More about Jelly Bean](#)[About Android](#)[Get the SDK](#)[Open Source](#)[Support](#)[Legal](#)

Except as noted, this content is licensed under [Creative Commons Attribution 2.5](#). For details and restrictions, see the [Content License](#).

API



Android APIs API level: 16 +

android.view.animation

android.view.inputmethod

android.view.textservice

android.webkit

android.widget

dalvik.bytecode

dalvik.system

java.awt.font

java.beans

java.io

java.lang

java.lang.annotation

java.lang.ref

java.lang.reflect

java.math

java.util

TextSwitcher

TextView

TextView.SavedState

TimePicker

Toast

ToggleButton

TwoLineListitem

VideoView

ViewAnimator

ViewFlipper

ViewSwitcher

ZoomButton

ZoomButtonsController

ZoomControls

public class

Toast

extends Object

java.lang.Object

↳ android.widget.Toast

Summary: Constants | Ctors | Methods | Inherited Methods | [Expand All]

Since: API Level 1

Class Overview

A toast is a view containing a quick little message for the user. The toast class helps you create and show those.

When the view is shown to the user, appears as a floating view over the application. It will never receive focus. The user will probably be in the middle of typing something else. The idea is to be as unobtrusive as possible, while still showing the user the information you want them to see. Two examples are the volume control, and the brief message saying that your settings have been saved.

The easiest way to use this class is to call one of the static methods that constructs everything you need and returns a new Toast object.

Developer Guides

For information about creating Toast notifications, read the [Toast Notifications](#) developer guide.

Summary

Constants

int	LENGTH_LONG	Show the view or text notification for a long period of time.
-----	-----------------------------	---

int	LENGTH_SHORT	Show the view or text notification for a short period of time.
-----	------------------------------	--

Public Constructors

Toast(Context context)
--

Use Tree Navigation



API

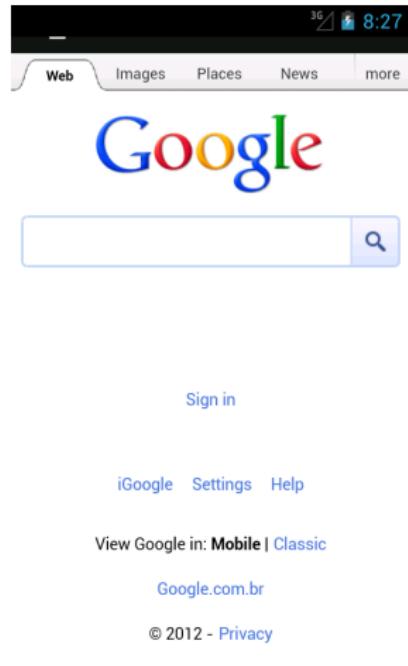
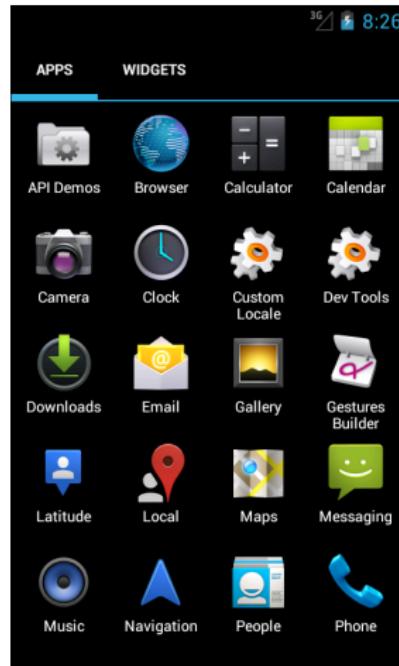
```
Toast.makeText(context, "Hello", Toast.LENGTH_LONG).show();
```

```
Toast.makeText(context, "Hello", Toast.LENGTH_LONG).show();
```



Emulador

Emulador



Ferramentas

Android SDK Manager

SDK Path: /home/bruno/local/opt/android-sdk-linux_x86

Packages

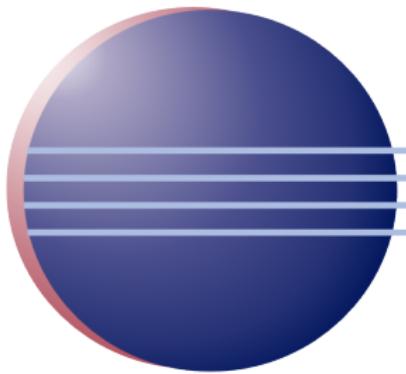
Name	API	Rev.	Status
Tools			
Android SDK Tools	20	Installed	
Android SDK Platform-tools	12	Installed	
Android 4.1 (API 16)			
Documentation for Android SDK	16	1	Installed
SDK Platform	16	1	Installed
Samples for SDK	16	1	Installed
ARM EABI v7a System Image	16	1	Installed
Google APIs	16	2	Installed
Sources for Android SDK	16	1	Installed
Android 4.0.3 (API 15)			
SDK Platform	15	3	Installed

Show: Updates/New Installed Obsolete Select [New](#) or [Updates](#) [Install 1](#)

Sort by: API level Repository [Deselect All](#) [Delete](#)

Done loading packages.

```
bruno@bruno-LN:~/tmp$ android create project -n HelloWorld -t android-16 -p HelloWorld -k com.helloworld -a HelloActivity
```



+

ADT

(Android Development Tools)



Sumário

Introdução

Conceitos

Hello, World

Menus

Exercício

Conclusão

Activities, Views, Resources e Intents

Activities

- ▶ São as telas da aplicação

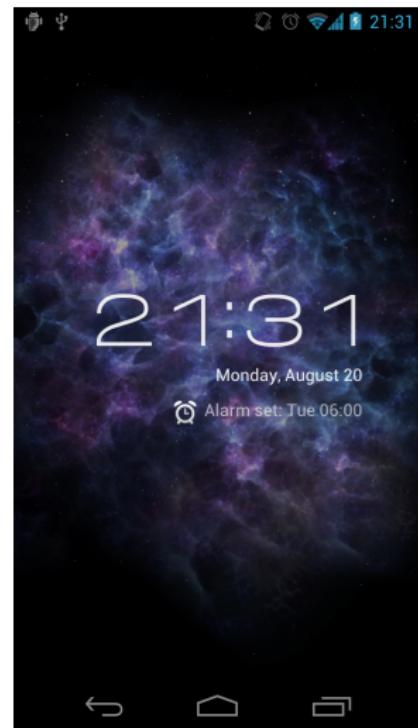
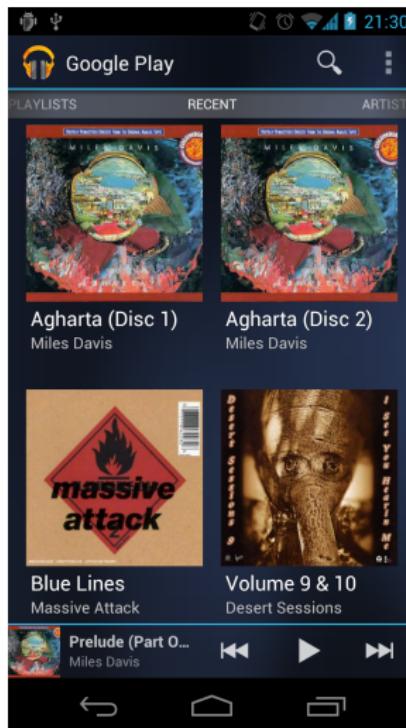
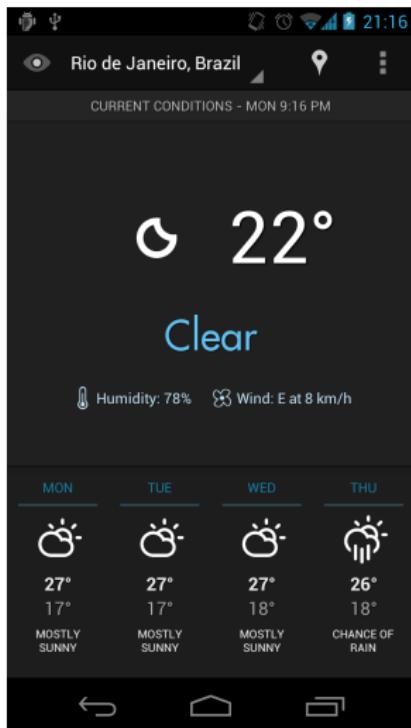
Activities

- ▶ São as telas da aplicação
- ▶ Aplicação: conjunto de activities vagamente ligadas umas às outras

Activities

- ▶ São as telas da aplicação
- ▶ Aplicação: conjunto de activities vagamente ligadas umas às outras
- ▶ Herdam da classe **android.app.Activity**

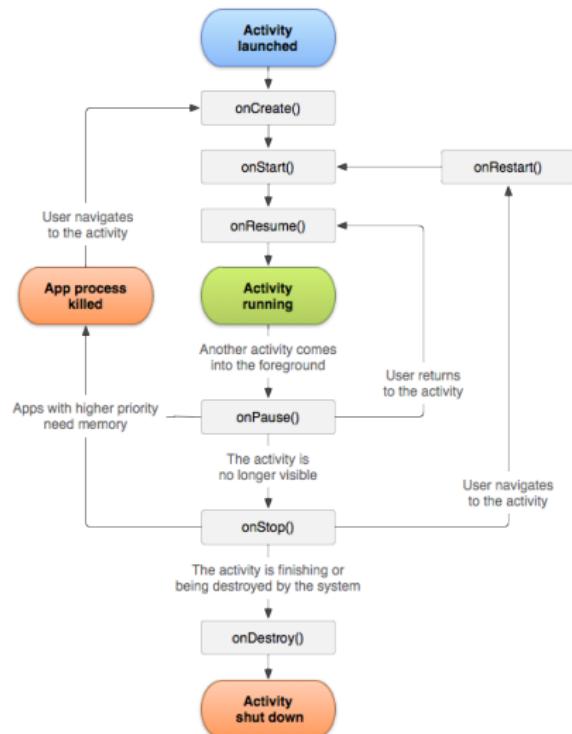
Activities: exemplos



Activities: exemplo de código

```
package com.foo;  
import android.app.Activity;  
import android.os.Bundle;  
  
public class FooActivity extends Activity  
{  
    /** Called when the activity is first created. */  
    @Override  
    public void onCreate(Bundle savedInstanceState)  
    {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);  
    }  
}
```

Ciclo de vida



Activities: características

- ▶ Ponto de interação com o usuário
- ▶ Activities iniciam outras activities
- ▶ Apenas uma activity está ativa por vez
- ▶ Activities exibem layouts e views

Views

- ▶ Blocos visuais básicos, como botões, imagens, textos etc...

Views

- ▶ Blocos visuais básicos, como botões, imagens, textos etc...
- ▶ Herdam da classe **android.view.View**

Views

- ▶ Blocos visuais básicos, como botões, imagens, textos etc...
- ▶ Herdam da classe **android.view.View**
- ▶ São agrupadas em layouts

Views

- ▶ Blocos visuais básicos, como botões, imagens, textos etc...
- ▶ Herdam da classe **android.view.View**
- ▶ São agrupadas em layouts
- ▶ Em geral, layouts e views são especificados em XML

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:orientation="vertical" >
    <EditText>
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:hint="@string/to" />
    <EditText>
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:hint="@string/subject" />
    <EditText>
        android:layout_width="fill_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:gravity="top"
        android:hint="@string/message" />
    <Button>
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:layout_gravity="right"
        android:text="@string/send" />
</LinearLayout>
```

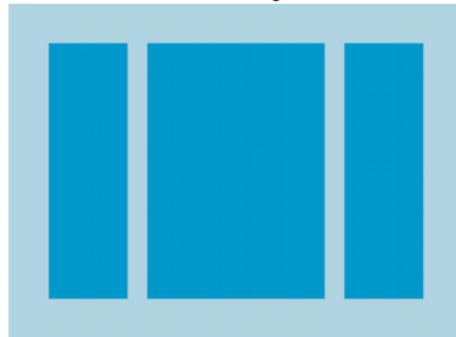
1,1

All



Layout: exemplos

Linear Layout:



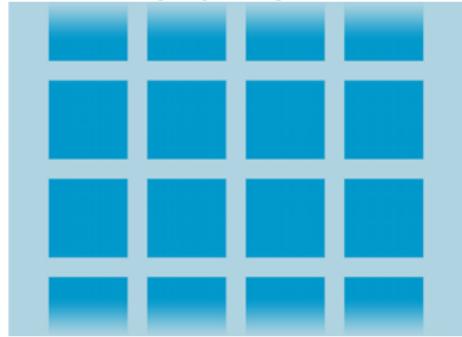
Relative Layout:



Web View:



Grid View:



Views: mutabilidade

- ▶ Layouts e views são acessíveis do código Java

Views: mutabilidade

- ▶ Layouts e views são acessíveis do código Java
- ▶ **findViewById**

Views: propriedades

Em XML:

```
<TextView  
    android:id="@+id/my_textview"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Hello" />
```

Em Java:

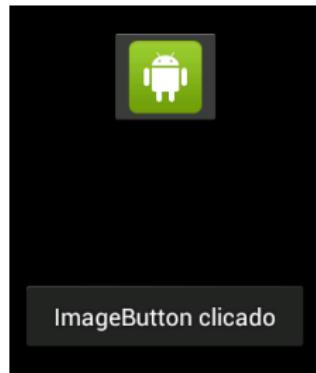
```
TextView tv = (TextView) findViewById(R.id.my_textview);  
tv.setText("Hello");
```

Resultado:

Hello

Views: listeners

```
ImageButton ib = (ImageButton) findViewById(R.id.my_imgbutton);  
ib.setOnClickListener(new OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Context context = MainActivity.this;  
        Toast.makeText(context, "ImageButton clicado", Toast.LENGTH_LONG).show();  
    }  
});
```



Views: modificando visibilidade

```
TextView1  
TextView2  
TextView3
```

```
findViewById(R.id.textview2).setVisibility(View.GONE); -
```

```
TextView1  
TextView3
```

```
findViewById(R.id.my_imgbutton);  
getString(R.string.hello);  
  
setContentView(R.layout.main);
```

```
findViewById(R.id.my_imgbutton);
```

```
getString(R.string.hello);
```

```
setContentView(R.layout.main);
```



Resources

- ▶ Só código Java não é suficiente. Ex: figuras, layouts (XML), strings etc

Resources

- ▶ Só código Java não é suficiente. Ex: figuras, layouts (XML), strings etc
- ▶ Podem ser acessadas pelo código Java através da classe **R**

Resources

- ▶ Só código Java não é suficiente. Ex: figuras, layouts (XML), strings etc
- ▶ Podem ser acessadas pelo código Java através da classe **R**
- ▶ Também podem ser referenciadas em resources XML

Resources

```
bruno at amsterdam in /tmp/foo
$ tree res
res
├── drawable-hdpi
│   └── ic_launcher.png
├── drawable-ldpi
│   └── ic_launcher.png
└── drawable-mdpi
    └── ic_launcher.png
layout
└── main.xml
values
└── strings.xml

5 directories, 5 files
```

strings.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">MinhaAplicação</string>
    <string name="hello">Olá</string>
</resources>
```

Na activity:

```
String hello = getString(R.string.hello);
Toast.makeText(this, hello, Toast.LENGTH_LONG).show();
```

A classe R

- ▶ Gerada automaticamente

A classe R

- ▶ Gerada automaticamente
- ▶ Contém referências para os resources presentes na pasta **res**

A classe R

- ▶ Gerada automaticamente
- ▶ Contém referências para os resources presentes na pasta **res**

```
String hello = getString(R.string.hello);  
Toast.makeText(this, hello, Toast.LENGTH_LONG).show();
```

Referenciando resources no XML

```
<ImageButton  
    android:id="@+id/refresh_button"  
    android:onClick="onRefreshClick"  
    android:src="@drawable/ic_action_refresh" />
```

```
<TextView  
    android:id="@+id/result"  
    android:text="@string/success" />
```

Intents

- ▶ São mensagens trocadas entre os componentes do sistema

Intents

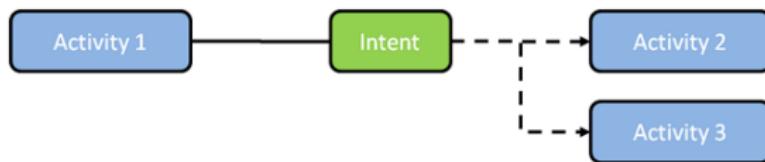
- ▶ São mensagens trocadas entre os componentes do sistema
- ▶ Usados para qualquer tipo de comunicação

Intents

- ▶ São mensagens trocadas entre os componentes do sistema
- ▶ Usados para qualquer tipo de comunicação
- ▶ Ex: início de tarefas, transferência de dados e notificação de componentes

Intents

- ▶ São mensagens trocadas entre os componentes do sistema
- ▶ Usados para qualquer tipo de comunicação
- ▶ Ex: início de tarefas, transferência de dados e notificação de componentes



Intents

Explícitos: indicam explicitamente o destinatário

```
Intent i = new Intent(this, FooActivity.class);  
startActivity(i);
```

Intents

Explícitos: indicam explicitamente o destinatário

```
Intent i = new Intent(this, FooActivity.class);  
startActivity(i);
```

Implícitos: especificam uma ação e seu argumento

```
Uri uri = Uri.parse("http://www.reddit.com");  
Intent i = new Intent(Intent.ACTION_VIEW, uri);  
startActivity(i);
```

Intents para transferência de dados

BarActivity.java:

```
Intent intent = new Intent(this, FooActivity.class);  
intent.putExtra("chave", "valor");  
startActivity(intent);
```

FooActivity.java:

```
public class FooActivity extends Activity {  
  
    @Override  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
  
        Bundle extras = getIntent().getExtras();  
        String value = extras.getString("chave");  
    }  
}
```

Intents como notificações

Com o consentimento do usuário, aplicações podem ser notificadas sobre diversos eventos

Intents como notificações

Com o consentimento do usuário, aplicações podem ser notificadas sobre diversos eventos

```
</activity>
<receiver android:name="com.android.music.MediaButtonIntentReceiver">
    <intent-filter>
        <action android:name="android.intent.action.MEDIA_BUTTON" />
        <action android:name="android.media.AUDIO_BECOMING_NOISY" />
    </intent-filter>
</receiver>

```

```
@Override
public void onReceive(Context context, Intent intent) {
    String intentAction = intent.getAction();
    if (AudioManager.ACTION_AUDIO_BECOMING_NOISY.equals(intentAction)) {
        Intent i = new Intent(context, MediaPlaybackService.class);
        i.setAction(MediaPlaybackService.SERVICECMD);
        i.putExtra(MediaPlaybackService.CMDNAME, MediaPlaybackService.CMDPAUSE);
        context.startService(i);
    } else if (Intent.ACTION_MEDIA_BUTTON.equals(intentAction)) {
```

Charge By Shake

Free Apps Dev



★★★★★ (2.882)

INSTALAR

Este aplicativo é compatível com seu Vivo Motorola MB860.

Outros aplicativos deste desenvolvedor

**Atire Bolha**

FREE APPS DEV

★★★★★ (246)

Gratuito

**Shake To Lock Screen**

FREE APPS DEV

★★★★★ (1.848)

Gratuito

**Simply Compass**

FREE APPS DEV

★★★★★ (106)

Gratuito

**Lovely Tiny Notes**

FREE APPS DEV

★★★★★ (267)

Gratuito

[Veja mais >](#)

Quem viu este aplicativo também viu

[VISÃO GERAL](#)
[COMENTÁRIOS DOS USUÁRIOS](#)
[O QUE HÁ DE NOVO](#)
[PERMISSÕES](#)

Permissões

ESTE APPLICATIVO TEM ACESSO A:

SUA LOCALIZAÇÃO

LOCAL APROXIMADO (COM BASE NA REDE)

Acessa fontes de localização aproximada, como o banco de dados de rede do celular, para determinar uma localização aproximada do tablet, quando disponível. Aplicativos maliciosos podem usar esse recurso para determinar aproximadamente onde você está. Acessa fontes de localização aproximada, como o banco de dados rede do celular, para determinar uma localização aproximada do celular, quando disponível. Aplicativos maliciosos podem usar esse recurso para determinar aproximadamente onde você está.

LOCALIZAÇÃO PRECISA (GPS)

Acessa fontes de localização finas, como o Sistema de Posicionamento Global no tablet, quando disponível. Aplicativos maliciosos podem usar esse recurso para determinar onde você está, e esse uso pode consumir energia da bateria. Acessa fontes de localização finas, como o Sistema de Posicionamento Global no celular, quando disponível. Aplicativos maliciosos podem usar esse recurso para determinar onde você está, e esse uso pode consumir energia da bateria.

COMUNICAÇÃO DA REDE

ACESSO TOTAL À INTERNET

Permite que o aplicativo crie sockets de rede.

SUAS INFORMAÇÕES PESSOAIS

GRAVAR HISTÓRICO E FAVORITOS DO NAVEGADOR

Permite que o aplicativo modifique o histórico do navegador ou favoritos armazenados em seu tablet. Aplicativos maliciosos podem usar isso para apagar ou modificar os dados de seu navegador. Permite que o aplicativo modifique o histórico do navegador ou favoritos armazenados em seu celular. Aplicativos maliciosos podem usar isso para apagar ou modificar os dados de seu navegador.

LER HISTÓRICO E FAVORITOS DO NAVEGADOR

Permite que o aplicativo leia todos os URLs visitados pelo navegador e todos os favoritos do navegador.

CHAMADAS TELEFÔNICAS

[LER ESTADO E IDENTIDADE DO TELEFONE](#)

Activities como funções

Intents são úteis para requisitar funcionalidades

Activities como funções

Intents são úteis para requisitar funcionalidades

```
Intent intent = new Intent(android.content.Intent.ACTION_SEND);  
intent.setType("text/plain");  
  
intent.putExtra(Intent.EXTRA_SUBJECT, "Algum assunto");  
intent.putExtra(Intent.EXTRA_TEXT, "Corpo da mensagem");  
  
startActivity(intent);
```

Intent filters

Como uma activity se declara capaz de realizar uma certa tarefa?

Intent filters

Como uma activity se declara capaz de realizar uma certa tarefa?

```
<uses-permission android:name="android.permission.INTERNET" />
```

```
<activity
    android:name=".ActivityTest"
    android:label="@string/app_name" >
    <intent-filter>
        <action android:name="android.intent.action.SEND" />
        <category android:name="android.intent.category.DEFAULT" />
        <data android:mimeType="text/plain" />
    </intent-filter>
</activity>
```

Sumário

Introdução

Conceitos

Hello, World

Menus

Exercício

Conclusão

Hello, Android!

Sumário

Introdução

Conceitos

Hello, World

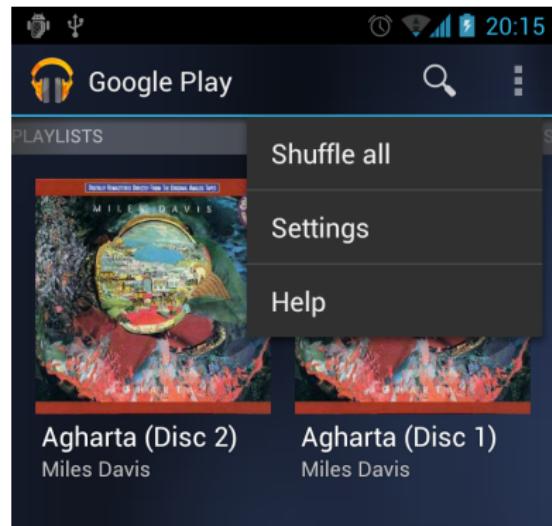
Menus

Exercício

Conclusão

Menus

Apresentam ações que podem ser realizadas dentro de uma activity



Menus: exemplo

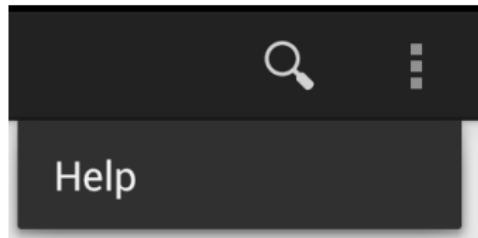
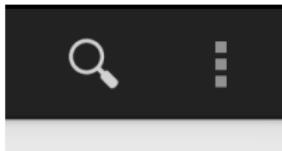
res/menu/some_activity_menu.xml:

```
<?xml version="1.0" encoding="utf-8"?>—  
<menu xmlns:android="http://schemas.android.com/apk/res/android">—  
    <item android:id="@+id/menu_search" —  
        android:icon="@drawable/ic_action_search" —  
        android:title="@string/search" —  
        android:showAsAction="ifRoom"/>—  
    <item android:id="@+id/menu_help" —  
        android:title="@string/help" —  
        android:showAsAction="never" />—  
</menu>—
```

Menus: exemplo

res/menu/some_activity_menu.xml:

```
<?xml version="1.0" encoding="utf-8"?>—  
<menu xmlns:android="http://schemas.android.com/apk/res/android">—  
    <item android:id="@+id/menu_search" —  
        android:icon="@drawable/ic_action_search" —  
        android:title="@string/search" —  
        android:showAsAction="ifRoom"/>—  
    <item android:id="@+id/menu_help" —  
        android:title="@string/help" —  
        android:showAsAction="never" />—  
</menu>—
```



Menus: callbacks

```
@Override  
public boolean onCreateOptionsMenu(Menu menu) {  
    MenuInflater inflater = getMenuInflater();  
    inflater.inflate(R.menu.activity_hello, menu);  
    return true;  
}  
  
@Override  
public boolean onOptionsItemSelected(MenuItem item) {  
    switch (item.getItemId()) {  
        case R.id.menu_search:  
            showSearch();  
            return true;  
        case R.id.menu_help:  
            showHelp();  
            return true;  
        default:  
            return super.onOptionsItemSelected(item);  
    }  
}
```

Introdução

Conceitos

Hello, World

Menus

Exercício

Conclusão

Conversor de temperatura

Sumário

Introdução

Conceitos

Hello, World

Menus

Exercício

Conclusão

TODO

- ▶ Fragments
- ▶ Adapters
- ▶ Threads e Serviços
- ▶ Broadcast Receivers
- ▶ Notificações
- ▶ Preferências
- ▶ Content Providers
- ▶ SQLite
- ▶ HTTP
- ▶ ...

{ github.com bitbucket.org } /qrntt/sc2-android