

Google Android



Componentes Gráficos

Google Android

No Android, por padrão já existe uma separação entre interface gráfica (View) e Controlador (Controller).

View -> Resources ->



Controller -> Activities ->

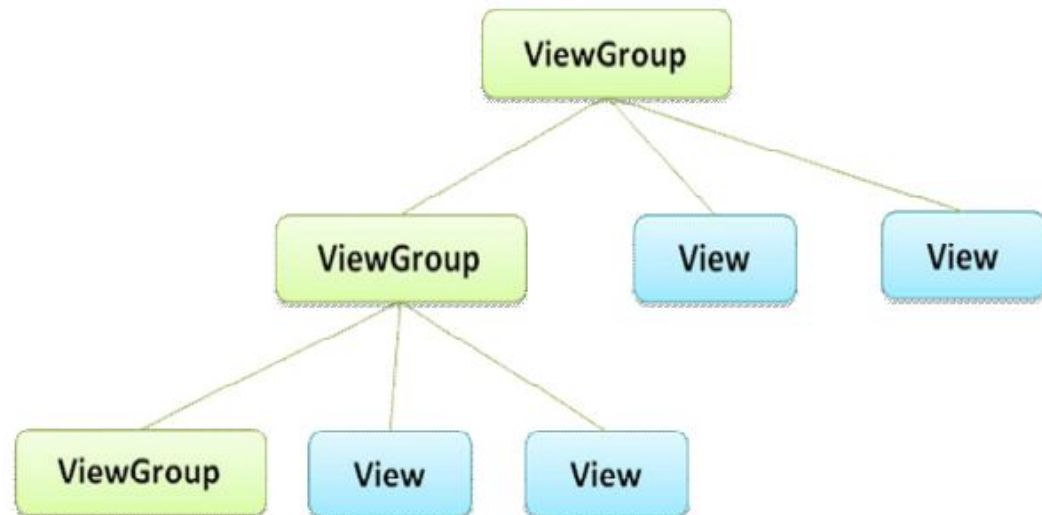


Google Android

Layouts

- ✓ Um tipo de View cuja a principal proposta é organizar outros controles na tela. São usados para exibir controles filhos como textos, caixas de texto, imagens, botões e etc. na tela.
- ✓ São definidos no arquivo de recursos /res/layout

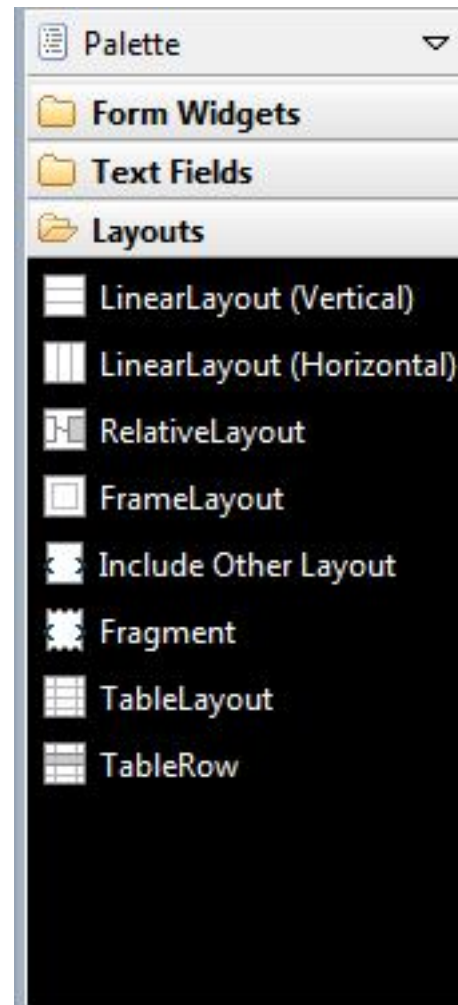
- ✓ Classe base
`android.view.ViewGroup`
que representa um Container
de `android.view.View`



Google Android

Tipos de Layout

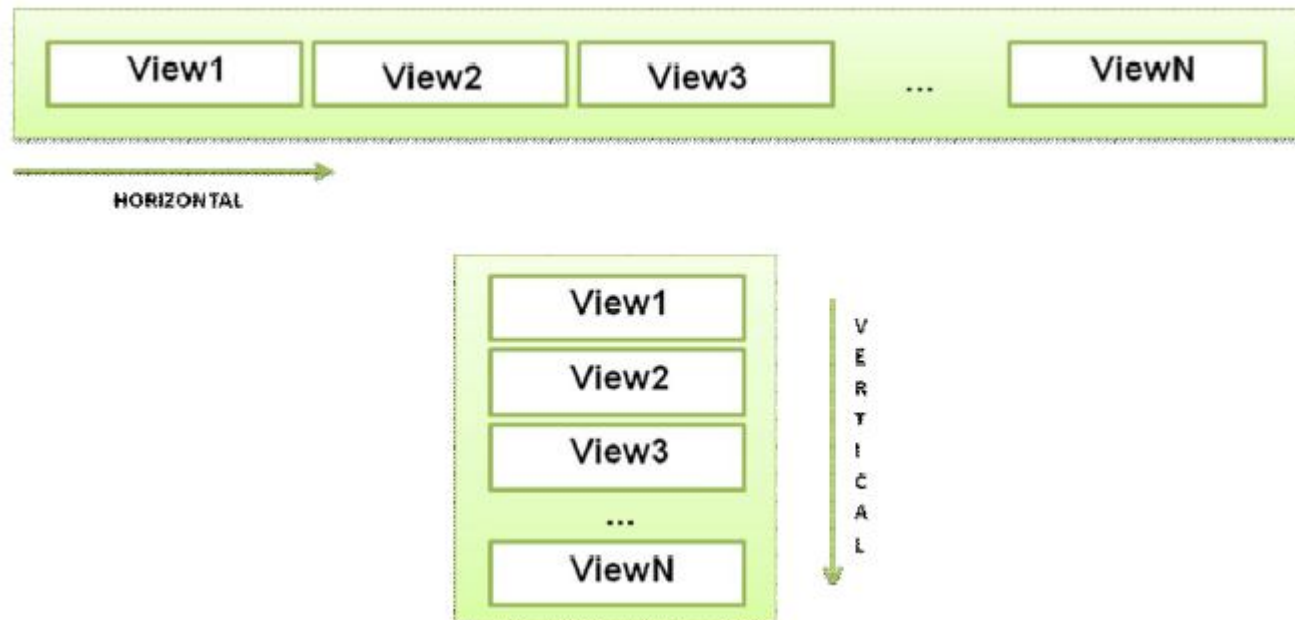
- ✓ LinearLayout
- ✓ FrameLayout
- ✓ RelativeLayout
- ✓ GridLayout
- ✓ TableLayout
- ✓ AbsoluteLayout



Google Android

Linear Layout

Utilizado para dispor os componentes em uma única direção (android:layout_orientation)



Google Android

Linear Layout

```
android:layout_width="fill_parent"  
android:layout_height="fill_parent"
```



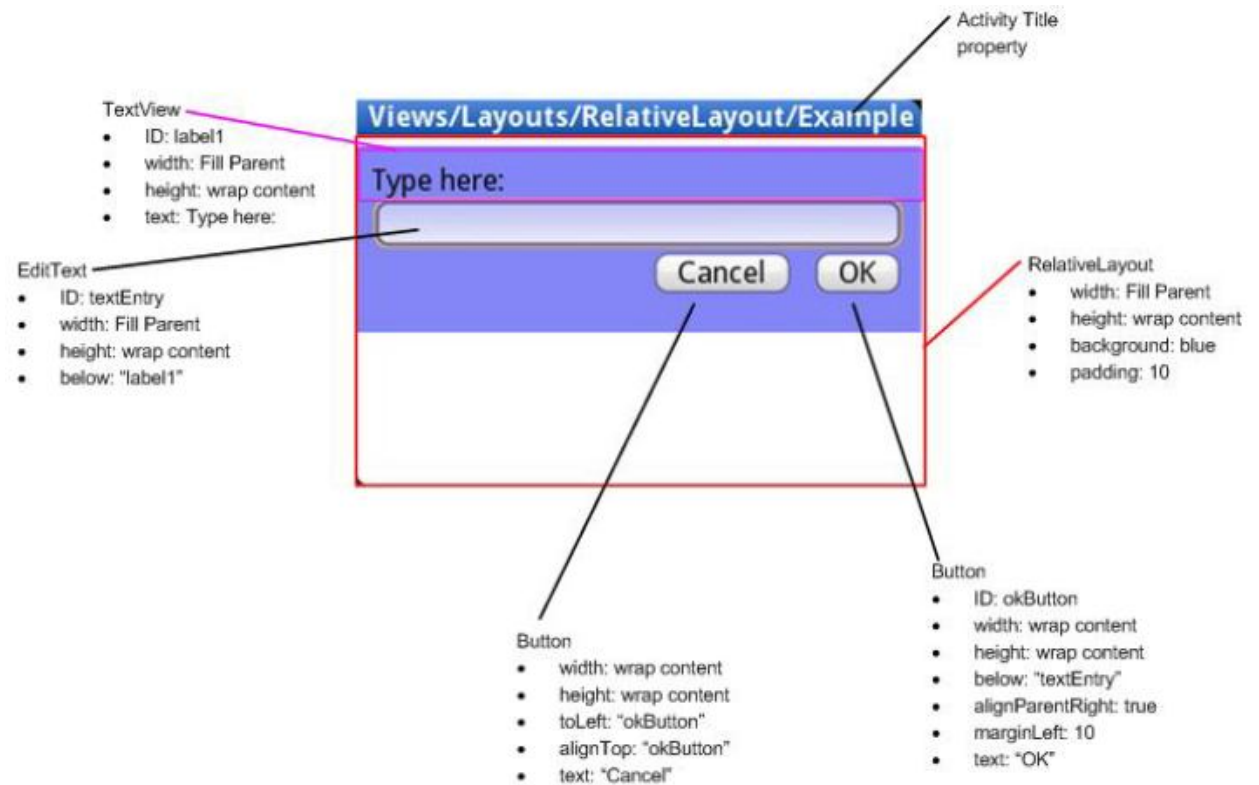
Layouts Project



Google Android

Relative Layout

Os componentes são ajustados através de relacionamentos entre si ou ao seu pai especificado pelo ID.



Layouts Project

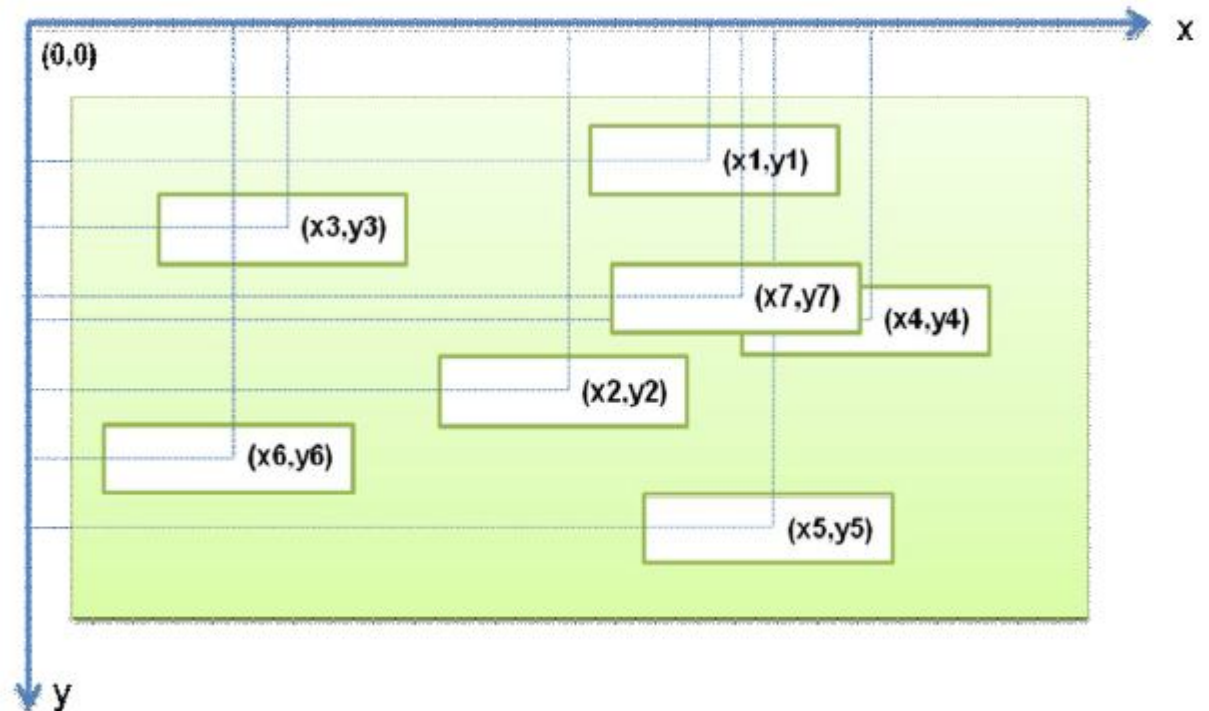


Google Android

Absolute Layout

Organiza os componentes de forma a implementar um plano cartesiano, no qual as suas posições x e y devem ser definidas para que estes possam ser posicionados de forma absoluta.

`android:layout_x="45px"`
`android:layout_y="87px"`



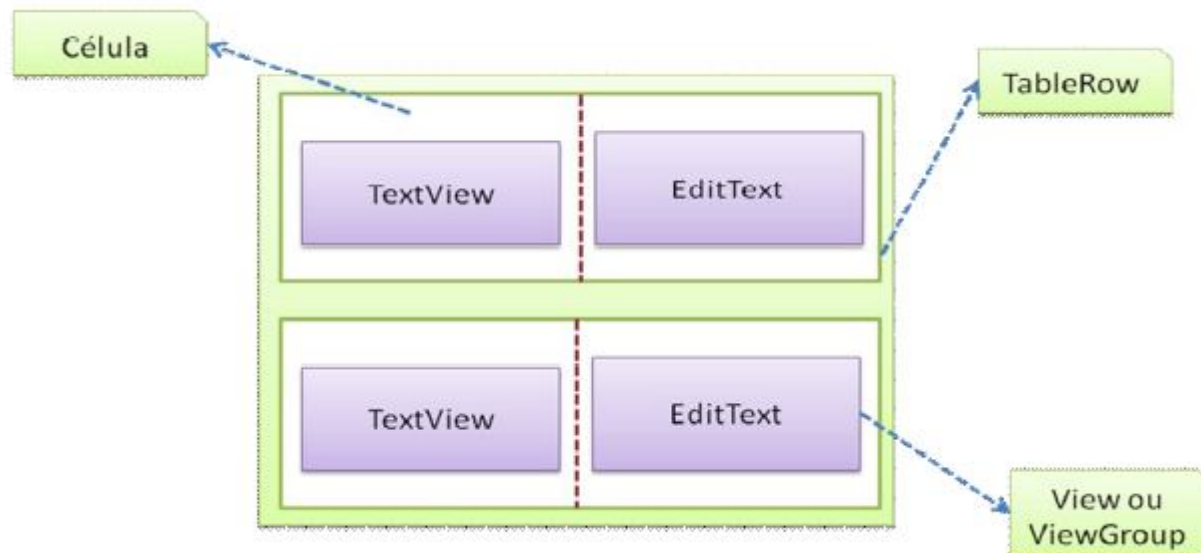
Google Android

Table Layout

Ordena as Views ou GroupViews em linhas e colunas. Cada Linha é representada pelo componente TableRow.



Layouts Project



Google Android

android.view.View

Classe que representa a base para componentes Android.

<View

android:id="@+id/id"

android:layout_width="fill_parent"

android:layout_height="wrap_content"/>

View findViewById (int id);



Google Android

android.widget.TextView

Mostra um texto para o usuário.

<TextView

```
android:id="@+id/text1"  
android:layout_width="fill_parent"  
android:layout_height="wrap_content"  
android:text="@string/hello" />
```

```
TextView t = (TextView) findViewById(R.id.text1);
```



GraphicComponents Project



text_view.xml



TextViewActivity.java



Google Android

android.widget.EditText

Uma implementação de TextView que permite um TextView ser editável.



GraphicComponents Project

<EditText

```
    android:id="@+id/editText1"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content" >
```

```
    <requestFocus />
```

```
</EditText>
```



edit_text_view.xml



Google Android

android.widget.EditText

android:inputType="textPassword"

android:inputType="phone"

android:inputType="textMultiLine"



```
EditText edtTxt1 = (EditText) findViewById(R.id.editText1);
```

```
Log.i("tag", "Value Txt: " + edtTxt1.getText());
```



EditTextActivity.java



Google Android

android.widget.Button

<Button

```
android:id="@+id/button1"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:text="@string/btn1" />
```



button.xml



ButtonActivity.java

```
Button button = (Button) findViewById(R.id.btn1);
```

▶

Google Android

Listeners e Buttons

```
Button button = (Button) findViewById(R.id.btn1);  
button.setOnClickListener(new View.OnClickListener() {  
    public void onClick(View v) {  
        // Perform action on click  
    }  
});
```



ButtonActivity.java



Google Android

android.widget.Toast

Pequeno container para mensagens rápidas mostradas ao usuário.

- ✓ Não há interação com o usuário;
- ✓ Usado para informações não essenciais; (Informação que melhora o uso da app para o usuário).

`makeText(Context context, CharSequence text, int duration)`

`makeText(Context context, int resId, int duration)`

`show()`

`int LENGTH_LONG`

`int LENGTH_SHORT`



Google Android

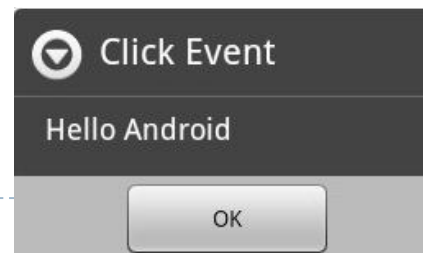
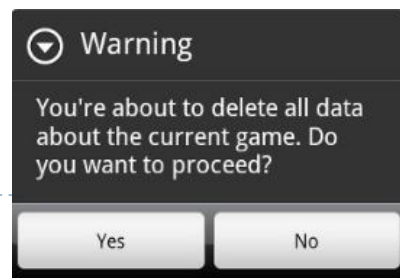
android.app.Dialog

Classe base para todos os tipos de Dialogs em Activities.

- ✓ Interação com o usuário;
- ✓ Permite Dialogs customizados.

Elementos Básicos de um Dialog

- ✓ Título;
- ✓ Mensagem;
- ✓ Botões de Coleta de Informações (Yes, No, Cancel, etc).



Google Android

android.app.AlertDialog.Builder

Dialogs de alerta em uma aplicação Android.

```
AlertDialog.Builder builder = new AlertDialog.Builder(this);  
builder.setTitle("Alerta");  
builder.setIcon(android.R.drawable.btn_star);  
builder.setMessage("Mensagem");  
builder.show();
```



AlertDialogs Project



AlertDialogActivity.java



Google Android

Listeners para AlertDialog.Builder

```
builder.setPositiveButton(android.R.string.ok,  
    new DialogInterface.OnClickListener() {  
        public void onClick(DialogInterface dialog, int which) {  
            Toast.makeText(getApplicationContext(),  
                "Clicked OK!", Toast.LENGTH_SHORT).show();  
            return;  
        }  
    });
```

```
builder.setNegativeButton(android.R.string.cancel,  
    new DialogInterface.OnClickListener() {  
        public void onClick(DialogInterface dialog, int which) {  
            Toast.makeText(getApplicationContext(),  
                "Clicked Cancel!", Toast.LENGTH_SHORT)  
                .show();  
            return;  
        }  
    });
```

```
builder.setNeutralButton(android.R.string.ok, listener);
```



AlertDialogActivity.java



Google Android

Outros métodos em Dialogs

```
setMultiChoiceItems(CharSequence[], boolean[] ,  
    DialogInterface.OnMultiChoiceClickListener);
```

```
setSingleChoiceItems(CharSequence[], int ,  
    DialogInterface.OnMultiChoiceClickListener);
```

- ✓ Customizar o próprio Dialog
- ✓ DatePicker

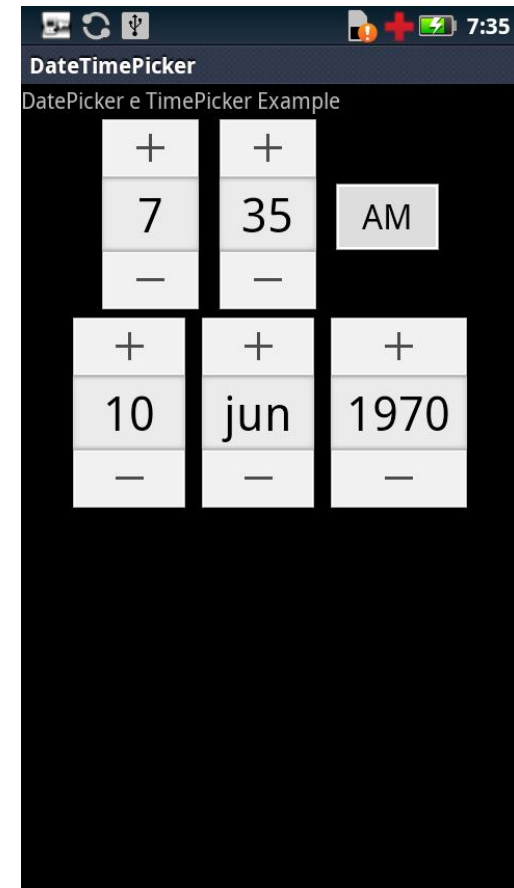


Google Android

DatePicker, TimePicker e DialogsPicker

Representa um calendário em Android.

- ✓ Calendar



Google Android

DatePicker e TimePicker

<TimePicker

```
    android:id="@+id/timePicker1"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="center" />
```

<DatePicker

```
    android:id="@+id/datePicker1"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="center"/>
```

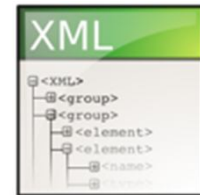


android:maxDate
android:minDate

format: mm/dd/yyyy



DateTimePicker Project



date_time_picker.xml



Google Android

DatePicker

```
DatePicker datePicker = (DatePicker)  
    findViewById(R.id.datePicker1);
```

```
datePicker.getMonth();  
datePicker.getDayOfMonth();  
datePicker.getYear();
```



DateTimePickerActivity.java



Google Android

DatePicker

`updateDate(int year, int month, int dayOfMonth)`

`datePicker.updateDate(2012, 5, 10);`



DateTimePickerActivity.java



Google Android

DatePicker

```
init(int year, int monthOfYear, int dayOfMonth,  
    DatePicker.OnDateChangeListener  
    onDateChangeListener)
```

```
datePicker.init(2010, 5, 13, mDatePickerSetListener);
```



DateTimePickerActivity.java



Google Android

DatePicker

```
DatePicker.OnDateChangeListener mDatePickerSetListener = new  
    DatePicker.OnDateChangeListener() {
```

```
@Override
```

```
public void onDateChanged(DatePicker view, int year, int monthOfYear,  
int dayOfMonth) {
```

```
}
```

```
};
```



DateTimePickerActivity.java



Google Android

TimePicker

```
TimePicker timePicker = (TimePicker) findViewById(R.id.timePicker1);
```

```
timePicker.setIs24HourView(true);
```

```
timePicker.getCurrentHour();
```

```
timePicker.getCurrentMinute();
```

```
timePicker.setCurrentHour(9);
```

```
timePicker.setCurrentMinute(26);
```



DateTimePickerActivity.java



Google Android

TimePicker

`setOnTimeChangeListener(TimePicker.OnTimeChangedListener onTimeChangedListener)...`



Google Android

TimePicker

```
timePicker.setOnTimeChangeListener(new  
    TimePicker.OnTimeChangeListener() {
```

```
    @Override
```

```
    public void onTimeChanged(TimePicker view, int hourOfDay, int minute) {  
    // TODO Auto-generated method stub  
    }  
});
```



DateTimePickerActivity.java



Google Android

DialogsPicker

✓ DatePickerDialog



✓ TimePickerDialog



Google Android

DatePickerDialog

- ✓ Construtor



DatePickerDialog(**Context** context,
DatePickerDialog.OnDateSetListener callBack, **int**
year, **int** monthOfYear, **int** dayOfMonth)



Google Android

DatePickerDialog

```
Calendar calendario = Calendar.getInstance();  
int ano = calendario.get(Calendar.YEAR);  
int mes = calendario.get(Calendar.MONTH);  
int dia = calendario.get(Calendar.DAY_OF_MONTH);
```



DateTimePickerActivity.java

```
DatePickerDialog dialog = new DatePickerDialog(this,  
                                                mDateSetListener, ano , mes, dia);  
dialog.show();
```



Google Android

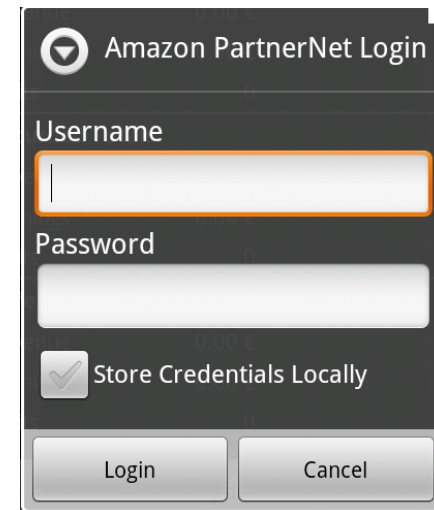
Dialog Customizado

- ✓ View customizada

```
Dialog dialog = new Dialog(this);  
dialog.setTitle("Simple Dialog!");  
dialog.setView(view);
```

- ✓ Classe que extends Dialog

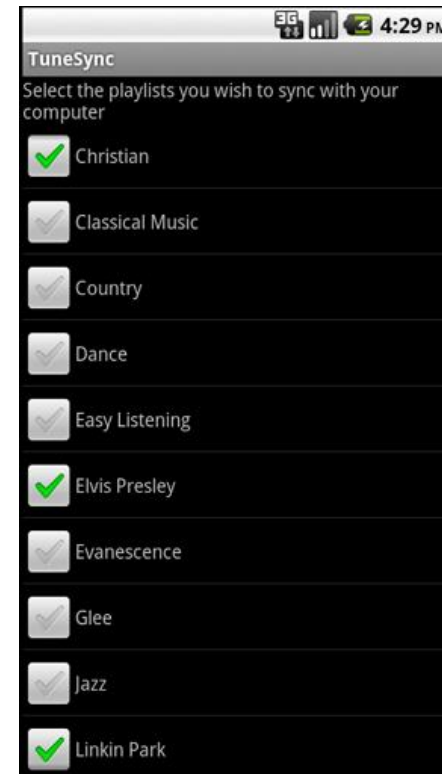
```
MyDialog d = new MyDialog(this);  
//seus métodos
```



Google Android

RadioButton, RadioGroup e CheckBox

Componentes que controlam configurações, opções e etc. em uma Activity.



Google Android

android.widget.RadioButton

Componente que representa um botão marcável na tela.

<RadioButton

```
    android:id="@+id/radioAlone"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:checked="true"  
    android:text="Utilizar onDemand" />
```



GraphicComponents Project



controls.xml



Google Android

android.widget.RadioGroup

Representa um grupo de Radios, onde já é implementado o SingleChoice.

<RadioGroup

```
    android:id="@+id/sexos"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content">
```

<RadioButton

```
    android:id="@+id/radioFeminino"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:checked="true"  
    android:text="Feminino" />
```

<RadioButton

```
    android:id="@+id/radioMasculino"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Masculino" />
```

</RadioGroup>



controls.xml



Google Android

android.widget.RadioGroup

```
final RadioGroup rgpSexos = (RadioGroup) findViewById(R.id.sexos);
```

```
Button btnChecks = (Button) findViewById(R.id.btnGroup);  
btnChecks.setOnClickListener(new View.OnClickListener() {
```

```
    @Override
```

```
    public void onClick(View v) {
```

```
        if (rgpSexos.getCheckedRadioButtonId() == R.id.radioFeminino) {
```

```
            Toast.makeText(getApplicationContext(), "Feminino Seleccionado!",  
                Toast.LENGTH_LONG).show();
```

```
        } else if (rgpSexos.getCheckedRadioButtonId() == R.id.radioMasculino) {
```

```
            Toast.makeText(getApplicationContext(), "Masculino Seleccionado!",  
                Toast.LENGTH_LONG).show();
```

```
        }
```

```
    }
```

```
});
```



ControlActivity.java



Google Android

android.widget.CheckBox

Representa um check na tela, chamado também de flag.

<CheckBox

android:id="@+id/chkConfig"

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:text="Exibir sempre esta mensagem" />



controls.xml



Google Android

android.widget.CheckBox

```
final CheckBox chkConfig = (CheckBox) findViewById(R.id.chkConfig);  
  
if (chkConfig.isChecked()){  
    Toast.makeText(getApplicationContext(),  
        "Configuração Ativa!", Toast.LENGTH_SHORT).show();  
}
```



ControlActivity.java



Google Android

android.widget.ImageView

Exibe uma imagem na tela.

<ImageView

android:id="@+id/myImage"

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:src="@drawable/ic_launcher" />



controls.xml



Google Android

android.widget.ImageView

```
final ImageView image = (ImageView) findViewById(R.id.myImage);  
image.setImageResource(R.drawable.logo);
```



ControlActivity.java

✓ Password Verify

Outros usos

```
Bitmap bMap = BitmapFactory.decodeFile("/sdcard/my_image.png");  
image.setImageBitmap(bMap);
```



Google Android

android.widget.ListView



ListView Project

Exibe os elementos de uma lista em um padrão comum de aplicações mobile.

<ListView

android:id="@+id/listView1"

android:layout_width="match_parent"

android:layout_height="wrap_content" >

</ListView>



list_view.xml



Google Android

android.widget.ListView

```
ListView listView = (ListView) findViewById(R.id.listView1);
```

```
String[] values = new String[] { "Android", "iPhone", "WindowsMobile",  
    "Blackberry", "WebOS", "Ubuntu", "Windows7", "Max OS X",  
    "Linux", "OS/2" };
```

```
// 1 – Contexto
```

```
// 2 – Layout da Linha
```

```
// 3 – ID da View na qual o dado será escrito
```

```
// 4 – Valores
```

```
ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,  
    android.R.layout.simple_list_item_1, android.R.id.text1, values);
```

```
listView.setAdapter(adapter);
```



ListViewActivity.java



Google Android

android.widget.ListView

```
ListView listView = (ListView) findViewById(R.id.listView1);
```

```
ArrayList<String> lista = new ArrayList<String>();
```

```
lista.add("1 - Aécio");
```

```
lista.add("2 - João");
```

```
lista.add("3 - Maria");
```

```
lista.add("4 - Juliana");
```

```
listView.setAdapter(new ArrayAdapter(this, R.layout.data_list, lista););
```



ListViewActivity.java



Google Android

android.widget.ListView

✓ R.layout.data

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<TextView xmlns:android="http://schemas.android.com/apk/res/android"  
    android:layout_width="fill_parent" android:layout_height="fill_parent"  
    android:textSize="20sp" android:padding="15dp">
```

```
</TextView>
```



data_list.xml



Google Android

✓ Adapter

ListView recupera informações e as exibe através de um Adaptador.

✓ android.widget.BaseAdapter

Classe base de Adapters para implementação de ListViews e Spinners customizados.

@Override

```
public View getView(int position, View convertView, ViewGroup parent)
```



Google Android

ListView e Listeners

OnItemClickListener

@Override

```
public void onItemClick(AdapterView<?> parent, View view,  
    int position, long id)
```



Google Android

ListView e Listeners

```
listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
  
    @Override  
    public void onItemClick(AdapterView<?> parent, View view, int position,  
        long id) {  
        String value = (String) parent.getAdapter().getItem(position);  
        Toast.makeText(getApplicationContext(), "Click ListItem Number " + position +  
            " Value: "+value,  
            Toast.LENGTH_LONG).show();  
    }  
  
});
```



ListViewActivity.java



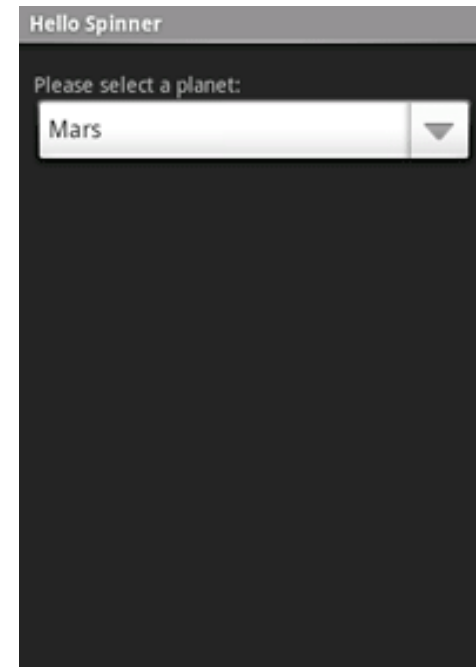
Google Android

android.widget.Spinner

Widget similar a uma list (ComboBox) para selecionar itens.

<Spinner

```
    android:id="@+id/spinner"  
    android:layout_width="fill_parent"  
    android:layout_height="wrap_content"  
    android:prompt="@string/planet_prompt" />
```



Google Android

android.widget.Spinner

✓ strings.xml

```
<string name="planet_prompt">Choose a planet</string>
<string-array name="planets_array">
  <item>Mercury</item>
  <item>Venus</item>
  <item>Earth</item>
  <item>Mars</item>
  <item>Jupiter</item>
  <item>Saturn</item>
  <item>Uranus</item>
  <item>Neptune</item>
</string-array>
```



Spinner Project



spinner.xml



Google Android

android.widget.Spinner

```
Spinner spinner = (Spinner) findViewById(R.id.spinner);
```

```
ArrayAdapter<CharSequence> adapter =  
ArrayAdapter.createFromResource(this, R.array.planets_array,  
    android.R.layout.simple_spinner_item);
```

```
spinner.setAdapter(adapter);
```



SpinnerActivity.java



Google Android

Listener e Spinner

```
spinner.setOnItemSelectedListener(new MyOnItemSelectedListener());
```

```
public class MyOnItemSelectedListener implements OnItemSelectedListener {
```

```
    public void onItemSelected(AdapterView<?> parent,
        View view, int pos, long id) {
        Toast.makeText(parent.getContext(), "The planet is " +
            parent.getItemAtPosition(pos).toString(),
            Toast.LENGTH_LONG).show();
    }
```

```
    public void onNothingSelected(AdapterView parent) {
        // Do nothing.
    }
}
```



MyOnItemSelectedListener.java



Google Android

Menu

@Override

public boolean onCreateOptionsMenu(Menu menu)

@Override

public boolean onOptionsItemSelected(int featureId, MenuItem item)



Google Android

Menu



GraphicComponents Project

```
private static final int ADD_MENU = 1;
```

```
@Override
```

```
public boolean onCreateOptionsMenu(Menu menu) {
```

```
    menu.add(0, ADD_MENU, 0,
```

```
        R.string.add_options_menu).setIcon(android.R.drawable.ic_menu_add);
```

```
    return true;
```

```
}
```



GraphicComponentsActivity.jav



Google Android

Menu

@Override

```
public boolean onOptionsItemSelected(int featureId, MenuItem item) {
```

```
    if (item.getItemId() == ADD_MENU) {
```

```
        Intent it = new Intent(this, MyActivity.class);
```

```
        startActivity(it);
```

```
        return true;
```

```
    }
```

```
    return false;
```

```
}
```



GraphicComponentsActivity.jav

