

Trabalho

App_Compra

Prof. Alexei Bueno

ENUNCIADO

- Você foi contratado para desenvolver um aplicativo que auxilie os usuários a anotarem os **itens** que realizarão compras nos supermercados.
- O aplicativo deverá permitir criar uma lista de itens **(assim como fizemos a lista de tarefas)**
- Ao separar determinado item o usuário deverá ser capaz de excluí-lo da lista **(assim como as tarefas do aplicativo anterior)**

A seguir irei lembrar os
principais itens que
fizemos no
App_Lista_Tarefa

1º Registrando o BD interno:



```
manifest
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="br.com.philadelpho.app_lista_tarefas_3ds">

    <meta-data android:name="DATABASE" android:value="tarefa.db"/>
    <meta-data android:name="DOMAIN_PACKAGE_NAME" android:value="br.com.philadelpho.app_lista_tarefas_3ds"/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="App_Lista_Tarefas_3DS"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".TelaPrincipal">
```

Irá se chamar **app_compra**

2º Importando biblioteca Sugar:

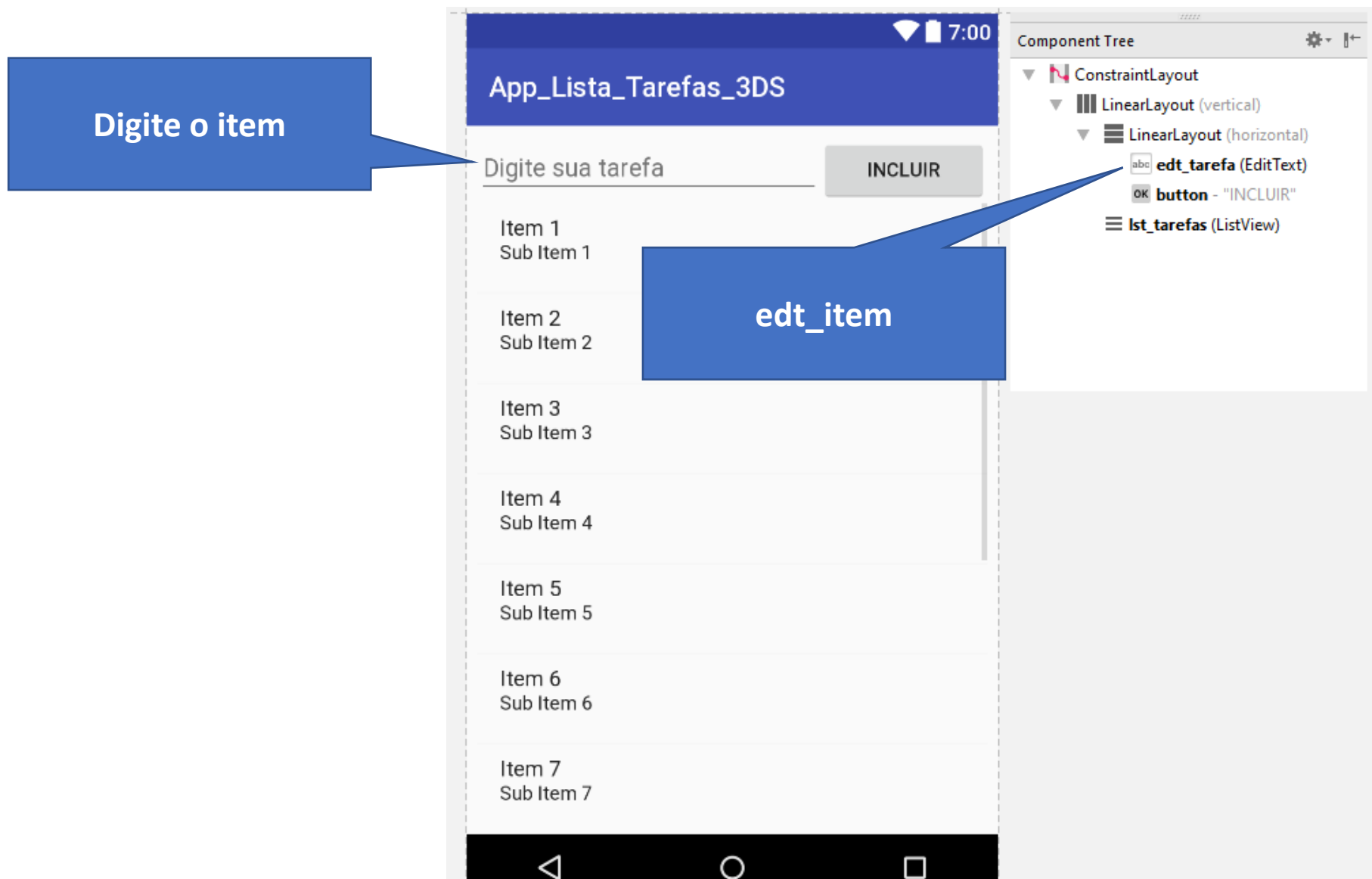


3º Criando a classe de BD

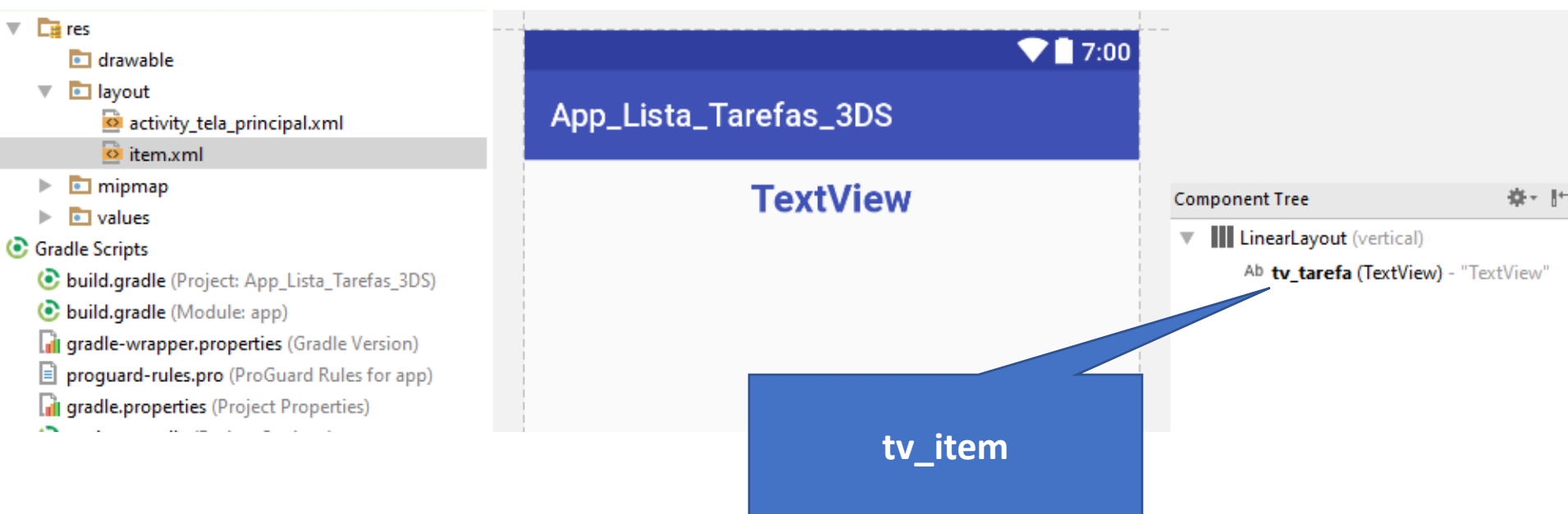
```
public class Tarefa extends SugarRecord {  
    Long id;  
    String tarefa;  
  
    public Tarefa() {  
  
    }  
  
    public Tarefa(String tarefa) {  
        this.tarefa = tarefa;  
    }  
}
```

A classe irá se chamar **Item**,
tendo como atributos **id** e
produto

4º Interface da TelaPrincipal:



5º Crio o “item” para a listagem:



Rotina de inclusão e listagem:

```
public void incluir_clique(View v){
    String nome_tarefa = edt_tarefa.getText().toString();

    new Tarefa(nome_tarefa).save();

    edt_tarefa.setText("");
    Toast.makeText(this, "Incluído com sucesso!", Toast.LENGTH_SHORT).show();
    listar();
}
```

Ao invés de tarefa é
item

```
//Lista o banco
private void listar(){
    //Lista tudo, na sequencia do cadastro
    //List<Tarefa> tarefas = Tarefa.listAll(Tarefa.class);

    List<Tarefa> tarefas = Tarefa.findWithQuery(Tarefa.class, "SELECT * FROM Tarefa ORDER BY tarefa");

    ArrayAdapter<Tarefa> adaptador = new ArrayAdapter<>(this,
        R.layout.item, R.id.tv_tarefa, tarefas);

    lst_tarefas.setAdapter(adaptador);
}
```

Ao invés de tarefa é
item

Obs.: coloquei duas maneiras de listar, sem ordenação e pela biblioteca e com Select, utilizando do Order By

Rotina para **excluir** item:

Ao invés de **tarefa** é
item

```
private void excluir(final Tarefa tarefa) {  
    AlertDialog.Builder alerta = new AlertDialog.Builder(this);  
    alerta.setTitle("Atenção!");  
    alerta.setMessage("Deseja excluir "+tarefa.getTarefa()+"?");  
    //Sim  
    alerta.setPositiveButton("Sim", new DialogInterface.OnClickListener() {  
        @Override  
        public void onClick(DialogInterface dialog, int which) {  
            tarefa.delete();  
            listar();  
        }  
    });  
    //Não  
    alerta.setNegativeButton("Não", null);  
    alerta.show();  
}
```

Ao clicar em um **item**, chamo rotina para excluir (e processos iniciais):

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_tela_principal);

    edt_tarefa = (EditText)findViewById(R.id.edt_tarefa);
    lst_tarefas = (ListView)findViewById(R.id.lst_tarefas);

    //Conecta no banco
    SugarContext.init(this);

    listar();

    //Clique no listview
    lst_tarefas.setOnItemClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
            //Pego o objeto
            Tarefa tarefa = (Tarefa) lst_tarefas.getItemAtPosition(position);
            excluir(tarefa);
        }
    });
}
```

Ao invés de **tarefa** é
item