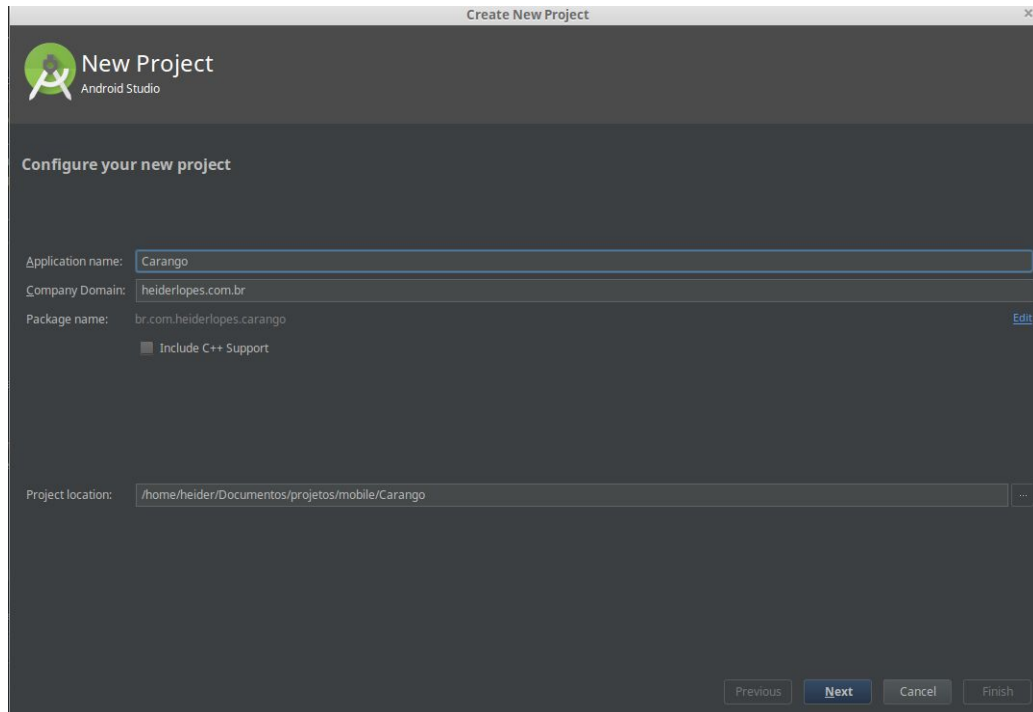


## Carango APP

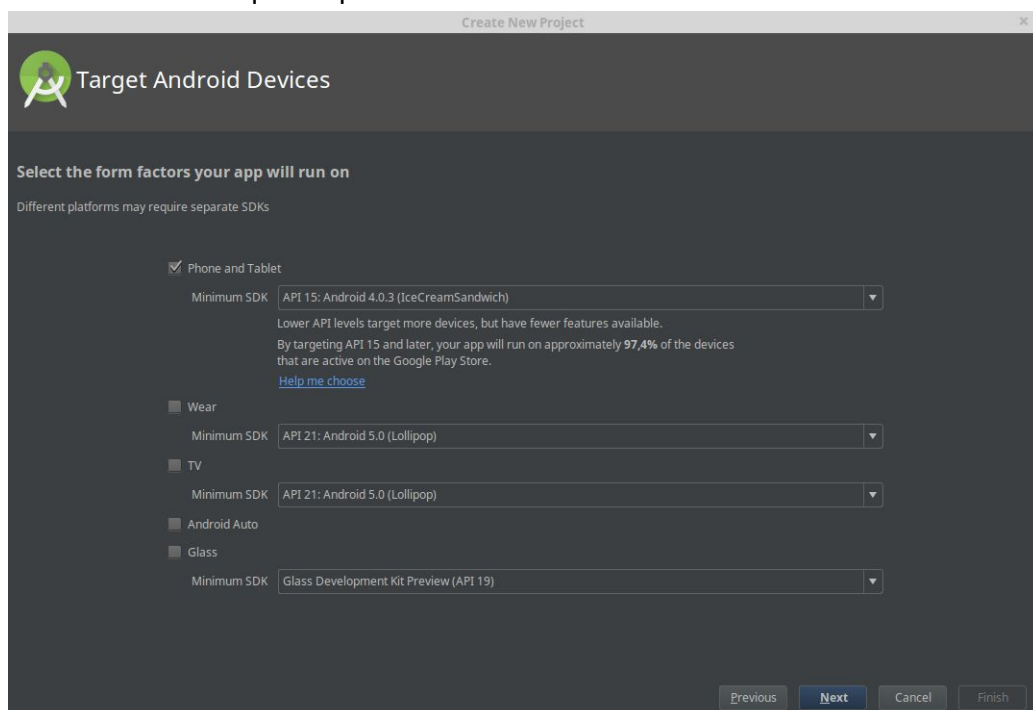
Descrição do nosso aplicativo

**Criando nosso aplicativo passo a passo.**

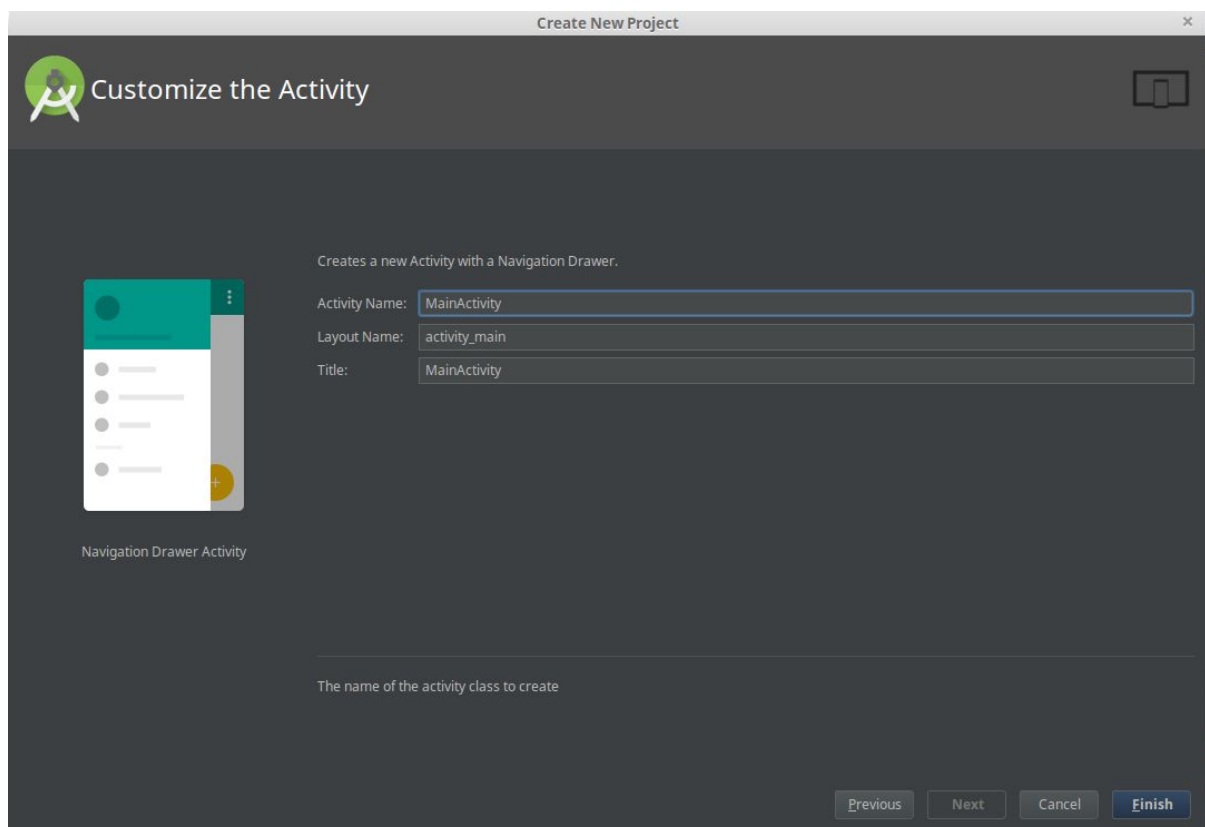
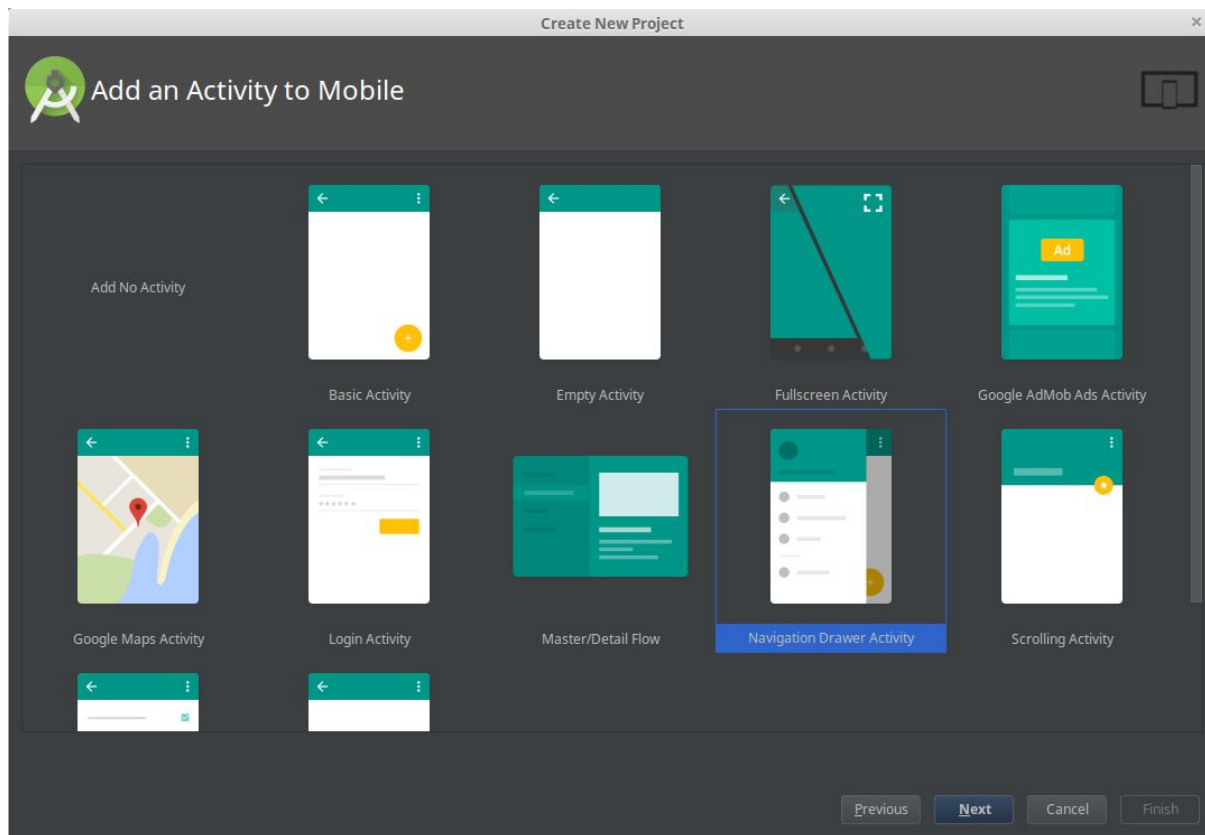
Abrir o Android Studio e criar o projeto abaixo:



Defina a versão mínima que o aplicativo deverá rodar:



Selecione o “template” da primeira Activity que será criada. Para este exemplo, utilizaremos o Navigation Drawer Activity



Dê o nome da activity e clique em **Finish**.

#### **Observação:**

O Navigation view possui 2 propriedades simples para configurar o menu lateral. São elas:

**app:headerLayout="@layout/nav\_header\_main"**: utilizada para definir o layout do cabeçalho do menu. É possível criar o layout que você quiser aqui.

**app:menu="@menu/activity\_main\_drawer"**: que define um arquivo de menu (formato XML) para compor as opções do menu lateral.

Note que o nosso **NavigationView** foi encontra-se no arquivo activity\_main e possui essas duas propriedades.

#### **Observação 2**

Caso você já tenha um projeto criado, para inserir o NavigationView, devemos importar a biblioteca:

**'com.android.support.design:XX.X.X'**, onde XX.X.X é o mesmo número da versão da biblioteca **'com.android.support.appcompat-V7'**

#### **Por exemplo:**

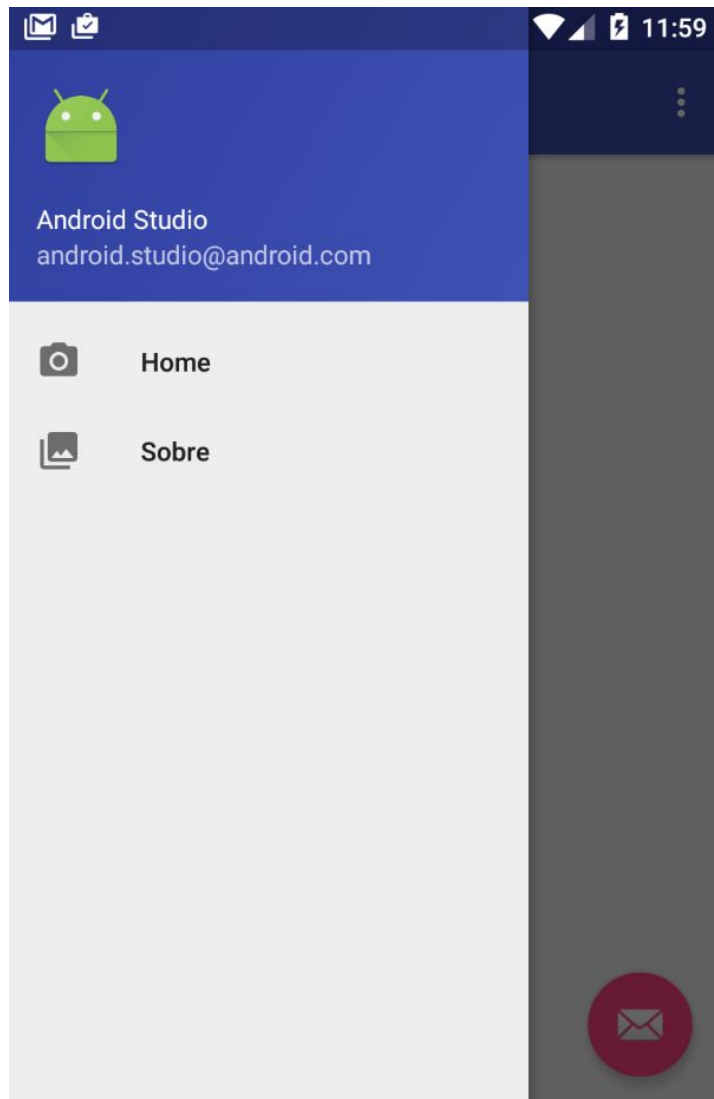
compile 'com.android.support:appcompat-v7:24.2.1'  
compile 'com.android.support:design:24.2.1'

#### **Hora de Praticar**

1º Abra o arquivo **nav\_header\_main.xml** e altere de acordo com suas preferências.  
(Dicas para praticar: Troque a cor de background, altere a imagem, troque as cores do texto)

2º Abra o arquivo **activity\_main\_drawer.xml** e altera o menu para que contenha as seguintes opções: Home e Sobre.

**Exemplo de como ficará a tela:**



Pronto, nosso menu está previamente configurado, agora iremos implementar a nossa Home.

## TabLayout

A Material Design Support Library fornece uma classe chamada `TabLayout` a qual podemos utilizar para criar Tabs.

## Configurando nosso Style

Abra o **styles.xml** e altere o parent do style `AppTheme` para `Theme.AppCompat.Light.NoActionBar` com isso podemos utilizar nossa toolbar customizada.

## Inserindo o TabLayout

Abra o arquivo **content\_main.xml** e nele iremos inserir o nosso **TabLayout**, por exemplo:

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:background="#dfdfdf"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:layout_marginTop="?attr/actionBarSize">
    <android.support.design.widget.TabLayout
        android:id="@+id/tabs"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/colorPrimary"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true" />

    <android.support.v4.view.ViewPager
        android:id="@+id/viewpager"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@+id/tabs"/>

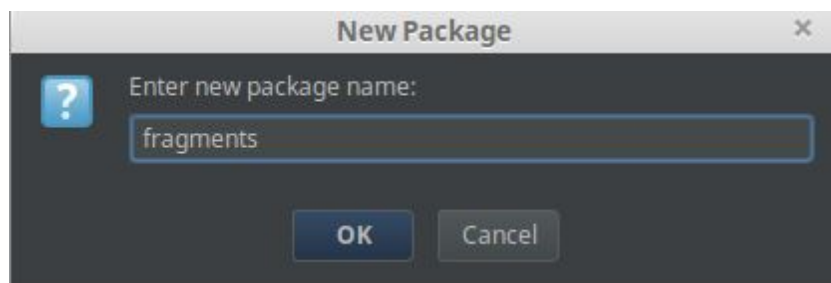
</RelativeLayout>
```

## Criando as telas que irão aparecer no home

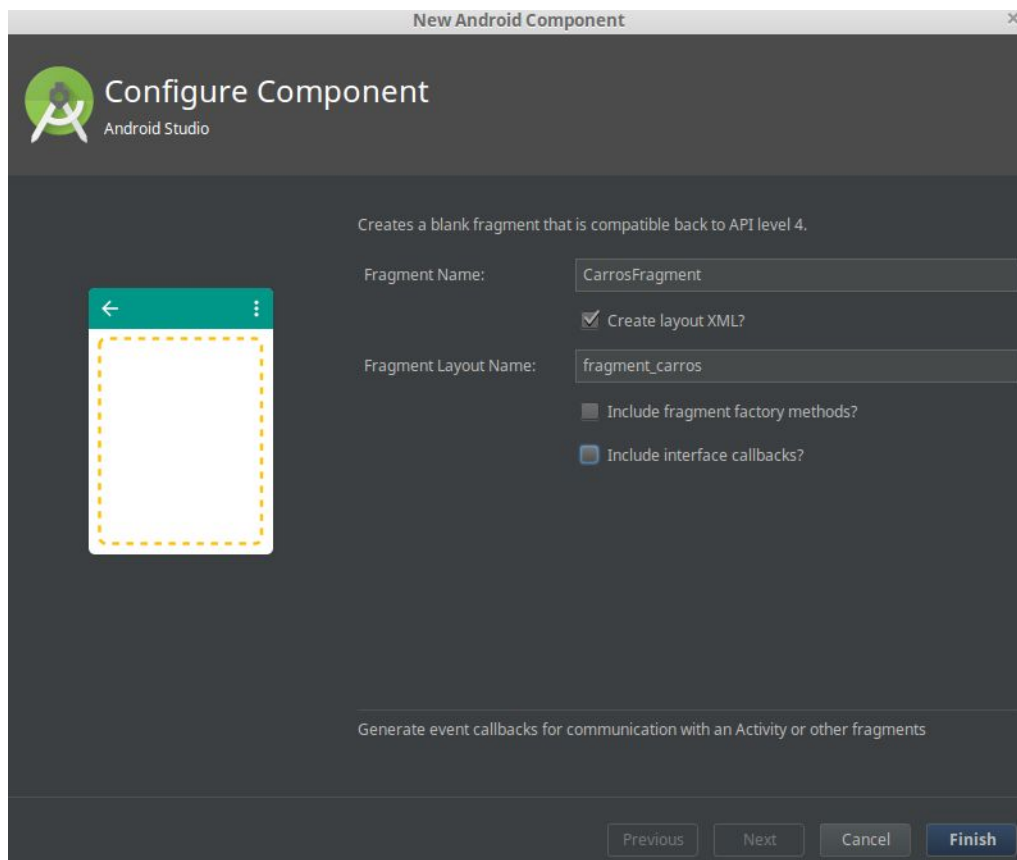
## Fragments

A partir da versão HONEYCOMB (Android 3.0) surgiu o Fragment pensando na modularização, na construção de User Interface sofisticados para telas maiores e ajudar a expandir a sua aplicação entre telas pequenas e grandes. Caso prático: imagine que a interface do usuário não é mais apenas uma Activity, mas sim um fragmento (pedaço) que você pode chamar em várias Activities de acordo com a necessidade, por exemplo: banners de propaganda. Esse é um conceito base dos Fragments. A gerência deles são feitas através do FragmentManager.

Crie um novo package chamado de **Fragment** (botão direito no pacote do projeto → New → Package)



Em seguida, crie um novo fragment (conforme imagem abaixo) dentro desse pacote (botão direito sobre o pacote → New → Fragment → Fragment Blank)



## Adicionando nosso Fragment

Nós temos o ViewPager inserido no nosso layout, conforme consta em um dos passos anteriores, o próximo passo agora é criarmos um Adapter para conseguirmos colocar nossos fragments em no ViewPager, para isso, crie um novo pacote chamado de **adapter**, dentro desse novo pacote crie uma classe chamada **TabsAdapter**.

```
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentManager;
import android.support.v4.app.FragmentPagerAdapter;

import br.com.heiderlopes.carango.fragments.CarrosFragment;

public class TabsAdapter extends FragmentPagerAdapter {

    public static final int TOTAL_TABS = 2;
    public TabsAdapter(FragmentManager fm) {
        super(fm);
    }

    //Nesse método definimos qual fragment deverá ser exibido no ViewPager
```

```

@Override
public Fragment getItem(int position) {

    Bundle args = new Bundle();

    if (position == 0) {
        args.putInt("tipo", 1);
    } else if (position == 1) {
        args.putInt("tipo", 2);
    }

    Fragment f = new CarrosFragment();
    f.setArguments(args);
    return f;
}

//Retorna o total de tabs que temos criada
@Override
public int getCount() {
    return TOTAL_TABS;
}
}

```

Com o nosso TabAdapter criado, devemos agora programar a nossa Activity. Para isso, abra-a e crie um método chamado **initTabLayout**, conforme código abaixo:

```

private void initTabLayout() {
    TabLayout tabLayout = (TabLayout) findViewById(R.id.tabs);

    tabLayout.addTab(tabLayout.newTab().setText("Clássicos"));
    tabLayout.addTab(tabLayout.newTab().setText("Esportivos"));

    tabLayout.setTabGravity(TabLayout.GRAVITY_FILL);

    final ViewPager viewPager = (ViewPager) findViewById(R.id.viewpager);
    final TabsAdapter adapter = new TabsAdapter (getSupportFragmentManager());
    viewPager.setAdapter(adapter);
    viewPager.addOnPageChangeListener(new
    TabLayout.TabLayoutOnPageChangeListener(tabLayout));

    tabLayout.setOnTabSelectedListener(new TabLayout.OnTabSelectedListener() {
        @Override
        public void onTabSelected(TabLayout.Tab tab) {
            viewPager.setCurrentItem(tab.getPosition());
        }
    });

    @Override

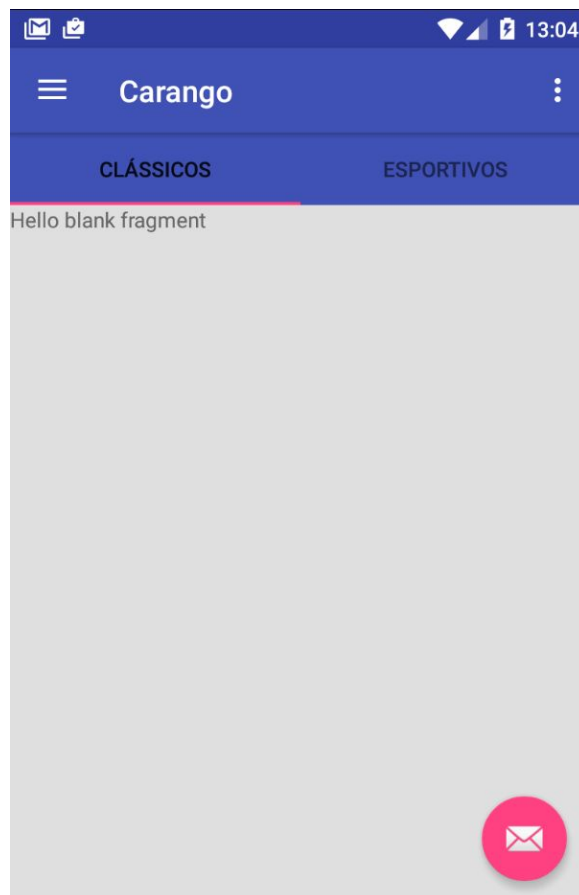
```



```
public void onTabUnselected(TabLayout.Tab tab) {  
  
}  
  
@Override  
public void onTabReselected(TabLayout.Tab tab) {  
  
}  
});  
}
```

Chame esse método dentro do **onCreate** de nossa Activity.

Até o momento já temos nossa aplicação da seguinte forma:



### Mostrando os carros

Iremos mostrar os carros em uma lista, para isso, utilizaremos o RecyclerView.

### Adicionando as dependências

Abra o arquivo build.gradle e adicione a seguinte linha de código no dependências:

compile 'com.android.support:recyclerview-v7:XX.X.X'

onde XX.X.X é o mesmo número da versão utilizada nas bibliotecas com.android.support

## Criando o layout

Abra o **fragment\_carros.xml** e adicione um RecyclerView

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="14dp">

    <android.support.v7.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:cacheColorHint="@android:color/transparent"
        android:clipToPadding="false"
        android:divider="@null"
        android:dividerHeight="0dp"
        android:listSelector="@android:color/transparent"
        android:scrollbarStyle="outsideOverlay"
        android:scrollbars="vertical" />

</FrameLayout>
```

Para mostrarmos os carros, iremos criar um layout que irá representar cada uma das linhas. Crie um arquivo chamado **item\_carro.xml** que contenha: Imagem, Nome, Descricao.

No layout abaixo estamos utilizando o cardview, para isso, devemos realizar o import da biblioteca:

Abra o arquivo build.gradle e adicione a seguinte linha de código no dependências:

compile 'com.android.support:cardview-v7:XX.X.X'

onde XX.X.X é o mesmo número da versão utilizada nas bibliotecas com.android.support

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:card_view="http://schemas.android.com/apk/res-auto">
```

```
android:id="@+id/card_view"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_margin="6dp"
android:clickable="true"
android:foreground="?attr/selectableItemBackground"
card_view:cardBackgroundColor="#FFFFFF"
card_view:cardCornerRadius="2dp"
card_view:cardElevation="6dp">
```

#### <LinearLayout

```
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="vertical">
```

#### <FrameLayout

```
android:layout_width="match_parent"
android:layout_height="wrap_content">
```

#### <FrameLayout

```
android:layout_width="wrap_content"
android:layout_height="wrap_content">
```

<!-- Foto do carro -->

#### <ImageView

```
android:id="@+id/img"
android:layout_width="350dp"
android:layout_height="150dp"
android:padding="16dp"
android:scaleType="centerCrop" />
```

<!-- Barra de progresso enquanto carrega a foto -->

#### <ProgressBar

```
android:id="@+id/progressImg"
style="@android:style/Widget.ProgressBar.Large"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_gravity="center|center_vertical"
android:layout_marginRight="6dp"
android:gravity="center|center_vertical"
android:visibility="invisible" />
```

#### </FrameLayout>

#### <TextView

```
android:id="@+id/text"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_gravity="right|bottom"
android:layout_marginTop="20dp"
android:background="#AA00000C"
android:padding="10dp"
```

```

        android:text="Carro"
        android:textColor="#fafafa"
        android:textSize="22sp" />
    </FrameLayout>

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:background="@android:color/darker_gray" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"

        >

        <TextView
            android:layout_weight="0.1"
            android:id="@+id/desc"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ellipsize="end"
            android:maxLines="3"
            android:padding="16dp"
            android:text="Descricao do carro" />

    </LinearLayout>
</LinearLayout>

</android.support.v7.widget.CardView>

```

Com o layout que irá representar cada linha, devemos agora criar um adapter para conseguirmos adicionar no nosso RecyclerView.

### Criando nosso modelo para representar o carro

Agora temos que criar um modelo que irá representar um carro. Crie um novo pacote chamado **model**, e dentro dele crie uma classe chamada **Carro**. Ela deverá conter: nome, desc, foto. Com os seus respectivos getters e setters.

## Criando nosso adapter

No pacote adapter, crie uma nova classe chamada **CarroListAdapter.java**

Como iremos mostrar a imagem do carro através da URL, utilizaremos a biblioteca Picasso para trazer esses dados. Para isso, abra o build.gradle e adicione a seguinte dependência:

```
package br.com.heiderlopes.carango.adapter;

import android.content.Context;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.ProgressBar;
import android.widget.TextView;

import com.squareup.picasso.Picasso;

import java.util.List;

import br.com.heiderlopes.carango.R;
import br.com.heiderlopes.carango.model.Carro;

public class CarroListAdapter extends
RecyclerView.Adapter<CarroListAdapter.CarrosViewHolder> {

    private final List<Carro> carros;
    private final Context context;

    private CarroOnClickListener carroOnClickListener;

    public CarroListAdapter(Context context, List<Carro> carros, CarroOnClickListener
carroOnClickListener) {
        this.context = context;
        this.carros = carros;
        this.carroOnClickListener = carroOnClickListener;
    }

    @Override
    public int getItemCount() {
        return this.carros != null ? this.carros.size() : 0;
    }

    @Override
    public CarrosViewHolder onCreateViewHolder(ViewGroup viewGroup, int viewType) {
        // Infla a view do layout
        View view = LayoutInflater.from(context).inflate(R.layout.item_carro, viewGroup,
false);
```

```

        return new CarrosViewHolder(view);
    }

    @Override
    public void onBindViewHolder(final CarrosViewHolder holder, final int position) {
        Carro c = carros.get(position);
        holder.tNome.setText(c.getNome());
        holder.desc.setText(c.getDesc());

        Picasso.with(context)
            .load(c.getFoto())
            .placeholder(R.mipmap.ic_launcher)
            .error(R.mipmap.ic_launcher)
            .into(holder.img);

        holder.progress.setVisibility(View.VISIBLE);

        if (carroOnClickListener != null) {
            holder.itemView.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View v) {
                    carroOnClickListener.onClickCarro(holder.itemView, position); // A variável
position é final
                }
            });
        }
        holder.progress.setVisibility(View.GONE);
    }

    //Acao a ser executada ao clicar em um item da lista - Devera ser implementada na
Activity ou Fragment
    public interface CarroOnClickListener {
        public void onClickCarro(View view, int idx);
    }

    //Acao a ser executada ao clicar no botao options - Devera ser implementada na Activity
ou Fragment
    public interface OptionsOnClickListener {
        public void onClickOptions(View view, int idx);
    }

    // ViewHolder com as views
    public static class CarrosViewHolder extends RecyclerView.ViewHolder {
        public TextView tNome;
        ImageView img;
        ProgressBar progress;
        TextView desc;

        public CarrosViewHolder(View view) {
            super(view);

```

```
// Cria as views para salvar no ViewHolder
tNome = (TextView) view.findViewById(R.id.text);
img = (ImageView) view.findViewById(R.id.img);
progress = (ProgressBar) view.findViewById(R.id.progressImg);
desc = (TextView) view.findViewById(R.id.desc);
    }
}
}
```

## Consumindo os dados da Web

O primeiro passo é importarmos uma biblioteca que irá nos auxiliar no consumo dos dados. Neste exemplo iremos utilizar a Retrofit. Abra o arquivo build.gradle e adicione a dependência:

compile 'com.squareup.retrofit2:retrofit:2.1.0'

```
compile 'com.squareup.retrofit2:converter-gson:2.0.2'
```

O próximo passo é criarmos uma interface que será utilizada para chamada do nosso serviço. O serviço que iremos utilizar será:

<http://www.heiderlopes.com.br/carros/tipo/{tipo}>

Então vamos lá, crie um novo package chamado de **api**, e dentro dele uma interface chamada **CarroAPI**.

Nossa interface terá o seguinte código:

```
import java.util.List;

import br.com.heiderlopes.carango.model.Carro;
import retrofit2.Call;
import retrofit2.http.GET;
import retrofit2.http.Path;

public interface CarroAPI {
    @GET("/carros/tipo/{tipo}")
    Call<List<Carro>> findBy(@Path("tipo") String tags);
}
```

## Abra o CarrosFraagment

```
import android.content.Intent;
import android.os.Bundle;
import android.support.annotation.Nullable;
import android.support.design.widget.Snackbar;
import android.support.v4.app.Fragment;
import android.support.v7.widget.DefaultItemAnimator;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import java.util.ArrayList;
import java.util.List;

import br.com.heiderlopes.carango.CarroDetalheActivity;
import br.com.heiderlopes.carango.R;
import br.com.heiderlopes.carango.adapter.CarroListAdapter;
import br.com.heiderlopes.carango.api.CarroAPI;
import br.com.heiderlopes.carango.model.Carro;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
import retrofit2.Retrofit;
import retrofit2.converter.gson.GsonConverterFactory;

public class CarrosFragment extends Fragment implements Callback<List<Carro>> {
    protected RecyclerView recyclerView;
    private List<Carro> carros;
    private LinearLayoutManager mLayoutManager;
    private String tipo;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            this.tipo = getArguments().getString("tipo");
        }
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {
        View view = inflater.inflate(R.layout.fragment_carros, container, false);
        recyclerView = (RecyclerView) view.findViewById(R.id.recyclerView);
        mLayoutManager = new LinearLayoutManager(getActivity());
```



```

        recyclerView.setLayoutManager(mLayoutManager);
        recyclerView.setItemAnimator(new DefaultItemAnimator());
        recyclerView.setHasFixedSize(true);
        return view;
    }

    @Override
    public void onActivityCreated(@Nullable Bundle savedInstanceState) {
        super.onActivityCreated(savedInstanceState);
        loadCarros();
    }

    private void loadCarros() {
        carros = new ArrayList<>();
        Retrofit retrofit = new Retrofit.Builder()
            .baseUrl("http://www.heiderlopes.com.br")
            .addConverterFactory(GsonConverterFactory.create())
            .build();

        // prepara a chamada no Retrofit 2.0
        CarroAPI carroAPI = retrofit.create(CarroAPI.class);
        Call<List<Carro>> call = carroAPI.findBy(tipo);
        //chamada assincrona
        call.enqueue(this);
    }

    //Listener utilizado ao clicar na linha do recyclerview
    private CarroListAdapter.CarroOnClickListener onClickCarro() {
        return new CarroListAdapter.CarroOnClickListener() {
            @Override
            public void onClickCarro(View view, int idx) {
                Carro c = carros.get(idx);
                Intent intent = new Intent(getContext(), CarroDetalheActivity.class);
                intent.putExtra("carro", c);
                startActivity(intent);
            }
        };
    }

    //Chamada caso de de sucesso na requisicao
    @Override
    public void onResponse(Call<List<Carro>> call, Response<List<Carro>> response) {
        recyclerView.setAdapter(new CarroListAdapter(getContext(), response.body(),
onClickCarro()));
    }

    //Chamada caso de erro na requisicao
    @Override
    public void onFailure(Call<List<Carro>> call, Throwable t) {
        Snackbar.make(recyclerView, t.getMessage(), Snackbar.LENGTH_INDEFINITE)
            .setAction("Action", null).show();
    }

```

```
}  
}
```

Crie uma nova Activity chamada **CarroDetalheActivity**, usando como template **EmptyActivity**.

O layout utilizado será:

```
<android.support.design.widget.CoordinatorLayout  
    xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:app="http://schemas.android.com/apk/res-auto"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent">  
  
    <android.support.design.widget.AppBarLayout  
        android:layout_width="match_parent"  
        android:layout_height="250dp"  
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">  
  
        <android.support.design.widget.CollapsingToolbarLayout  
            android:id="@+id/collapsing_toolbar"  
            android:layout_width="match_parent"  
            android:layout_height="match_parent"  
            app:contentScrim="?attr/colorPrimary"  
            app:layout_scrollFlags="scroll|exitUntilCollapsed">  
  
                <ImageView  
                    android:id="@+id/imagen"  
                    android:layout_width="match_parent"  
                    android:layout_height="match_parent"  
                    android:scaleType="centerCrop"  
                    android:src="@drawable/audi_spyder"  
                    app:layout_collapseMode="parallax"/>  
  
                <android.support.v7.widget.Toolbar  
                    android:id="@+id/toolbar"  
                    android:layout_width="match_parent"  
                    android:layout_height="?attr/actionBarSize"  
                    app:layout_collapseMode="pin"/>  
            </android.support.design.widget.CollapsingToolbarLayout>  
        </android.support.design.widget.AppBarLayout>  
        <android.support.v4.widget.NestedScrollView  
            android:layout_width="match_parent"  
            android:layout_height="match_parent"  
            app:layout_behavior="@string/appbar_scrolling_view_behavior">
```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:paddingTop="24dp">
    <android.support.v7.widget.CardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="16dp">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:padding="16dp">
            <TextView
                android:id="@+id/desc"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:text="Texto"/>
            </LinearLayout>
        </android.support.v7.widget.CardView>
    </LinearLayout>
</android.support.v4.widget.NestedScrollView>
</android.support.design.widget.CoordinatorLayout>

```

## Código da activity

```

import android.graphics.drawable.Drawable;
import android.os.Bundle;
import android.support.design.widget.CollapsingToolbarLayout;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.widget.ImageView;
import android.widget.TextView;

import java.io.IOException;
import java.io.InputStream;

import br.com.heiderlopes.carango.model.Carro;

public class CarroDetalheActivity extends AppCompatActivity {

    private ImageView imagem;
    private TextView desc;
    private CollapsingToolbarLayout collapsingToolbar;
    private Toolbar toolbar;

    @Override

```

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_carro_detalhe);

    toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);

    collapsingToolbar = (CollapsingToolbarLayout) findViewById(R.id.collapsing_toolbar);
    imagem = (ImageView) findViewById(R.id.imagem);
    desc = (TextView) findViewById(R.id.desc);

    if(getIntent() != null) {
        try{
            Carro carro = (Carro) getIntent().getSerializableExtra("carro");
            InputStream ims = getAssets().open(carro.getFoto());
            Drawable d = Drawable.createFromStream(ims, null);
            imagem.setImageDrawable(d);
            collapsingToolbar.setTitle(carro.getNome());
            desc.setText(carro.getDesc());
        } catch (IOException ioe) {
            ioe.printStackTrace();
        }
    }
}

```

Próximos passos:

Crie a tela de Sobre e abra a partir do NavigationView