

FUNDAÇÃO INSTITUTO DE EDUCAÇÃO DE BARUERI

Unidade Jardim Belval

Programação de Aplicativos 3

PRA3

Tutorial Android para Navegação de Janelas Internas

versão 2.0

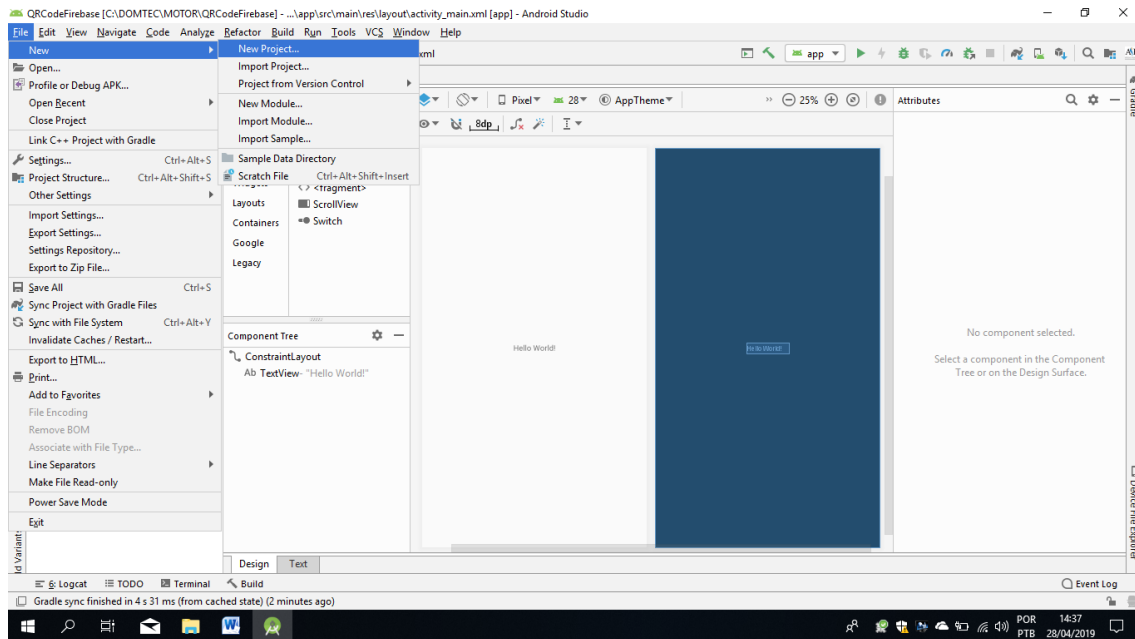
Prof. Adriano Domingues

2020

AULA 02 – Navigation Drawer

1. CRIAR PROJETO:

a. FILE > NEW > NEW PROJECT...



a. Altere as opções da primeira janela e clique no Finish:

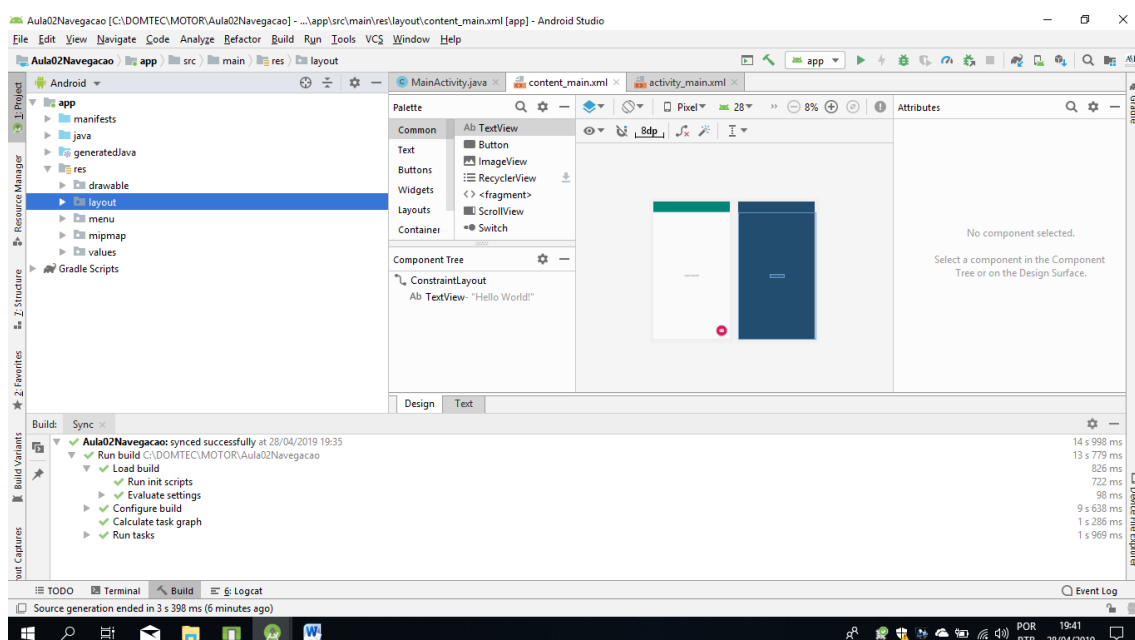
- i. Application Name: Aula02Navegacao
- ii. Company Domain: br.com.itb. aula02navegacao
- iii. Location: Z:\... (sua pasta na Z:)

b. API 22

c. ESCOLHA A ATIVIDADE DE EXEMPLO: Navigation Drawer Activity

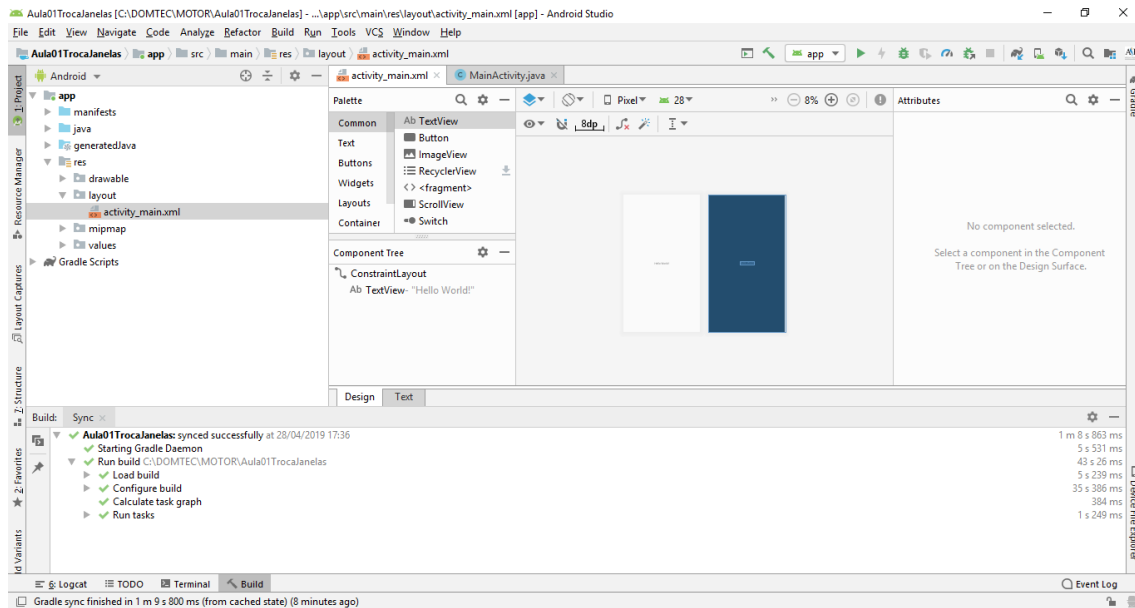
- iv. Não altere nada na última janela e clique em Finish

b. Aguarde o carregamento do projeto:



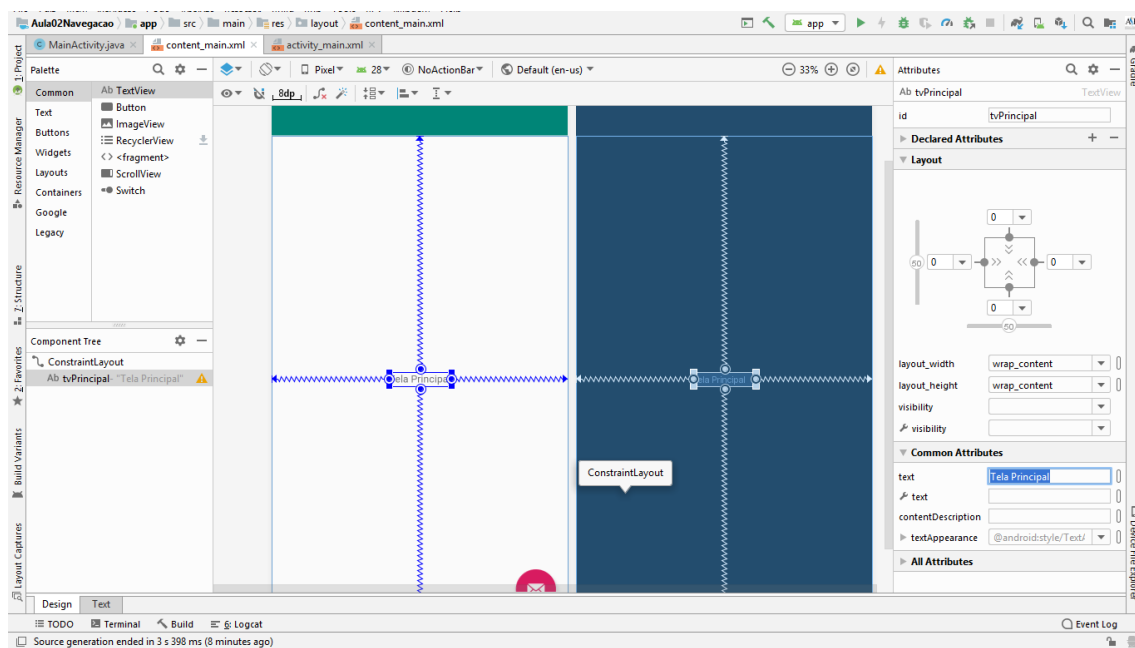
c. Abra a edição da tela content_main.xml

i. Clique duas vezes em Android > app > res > layout > content_main.xml

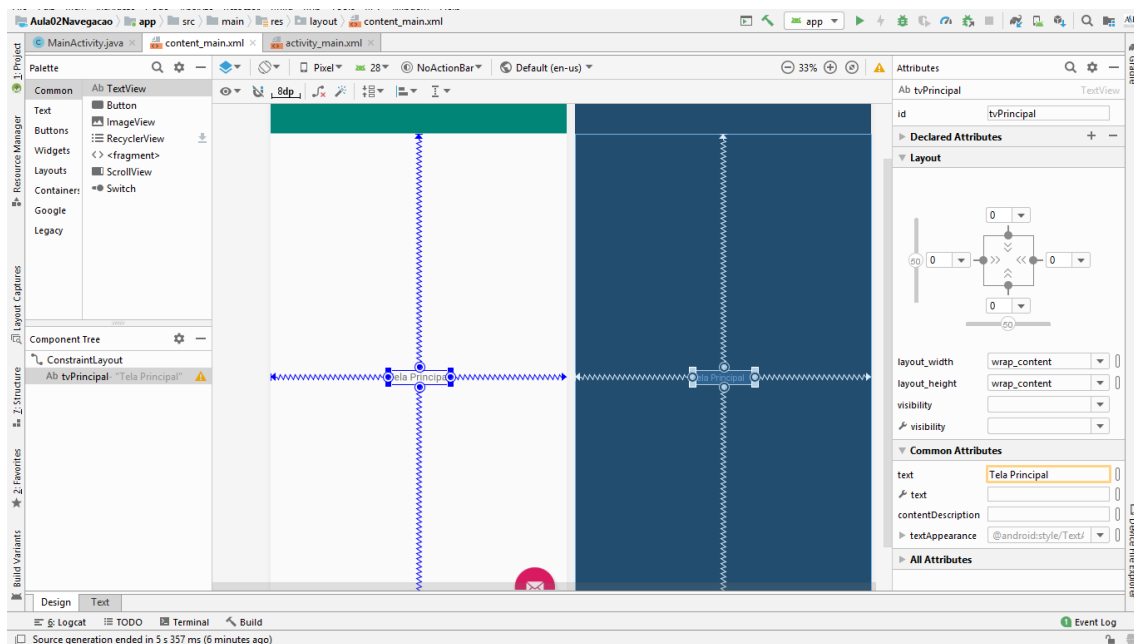


d. Clique no menu View > Entre Full Screen

i. Ajuste a tela para melhor visualização e edição da janela



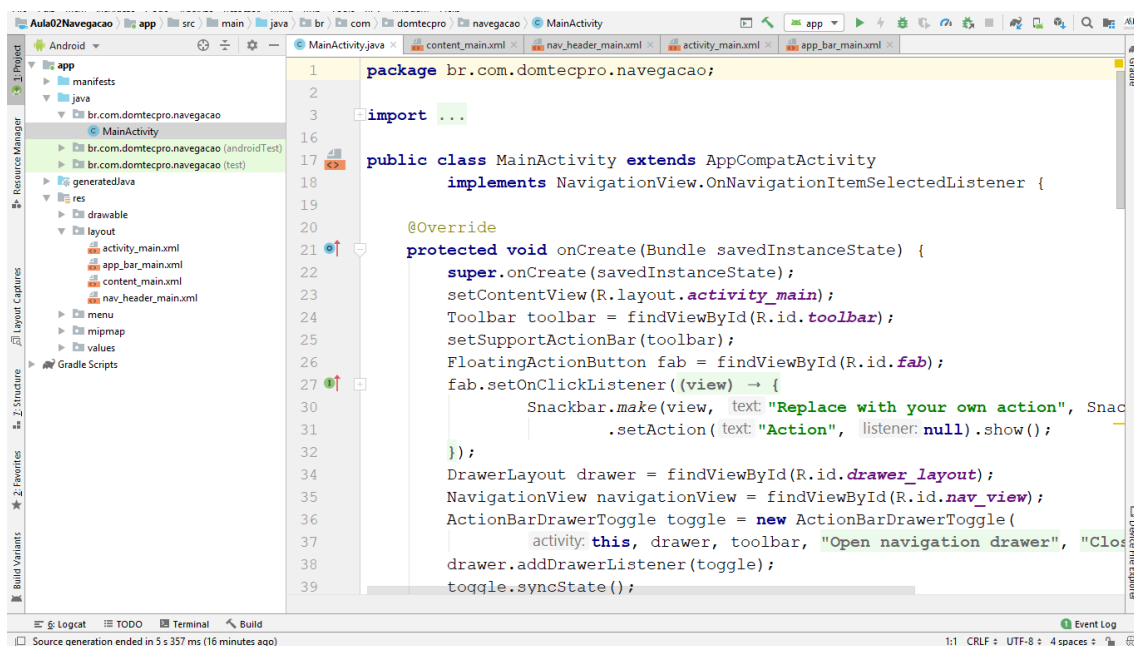
- e. Altere as propriedades do objeto TextView da janela de conteúdo:
- i. id: tvPrincipal
 - ii. text: Tela Principal



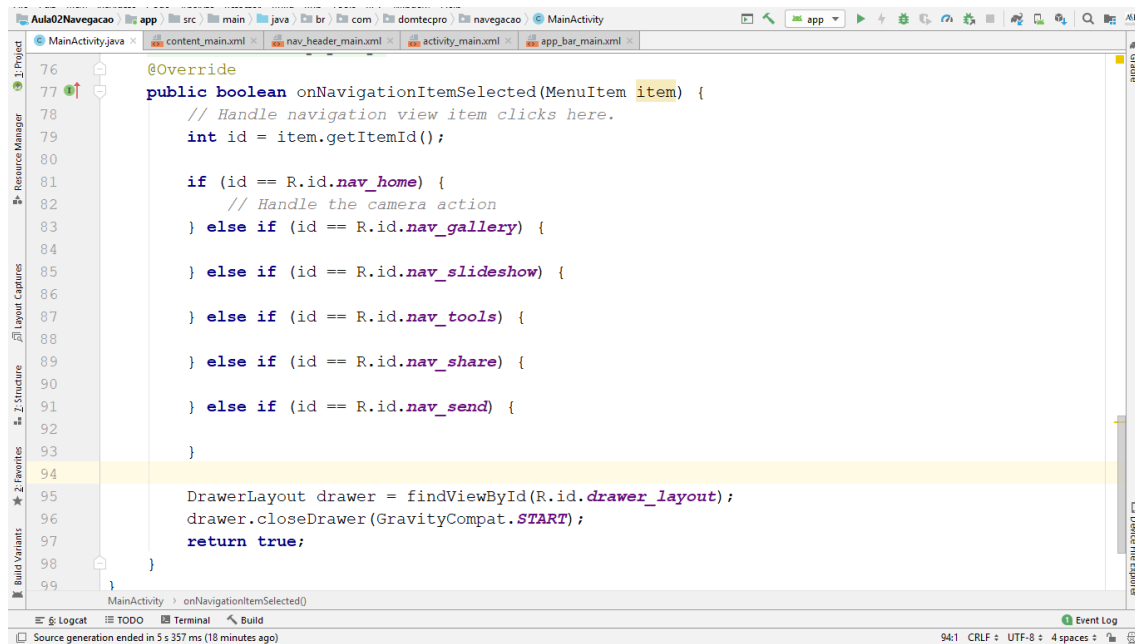
Obs.: As margens definidas como **0** centralizam o objeto em relação às dimensões da janela.

Os parâmetros definidos como **wrap_content** indica que as margens internas do objetos são definidas pelo tamanho da palavra escrita para apresentação.

- f. Clique em Android > app > java > br.com.itb.navegacao
- i. Abra a classe MainActivity.java deste projeto.



- g. Desça até a linha 77 no método
onNavigationItemSelectedListener(MenuItem item)



- i. A imagem acima retrata o código original, vamos adicionar a chamada da câmera do celular a partir da linha 82.
ii. Abaixo o código com a chamada simples da câmera:

```

@Override
public boolean onNavigationItemSelectedListener(MenuItem item) {
    // Handle navigation view item clicks here.
    int id = item.getItemId();

    if (id == R.id.nav_home) {
        // Handle the camera action
        Intent it = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
        startActivity(it);
    } else if (id == R.id.nav_gallery) {

    } else if (id == R.id.nav_slideshow) {

    } else if (id == R.id.nav_tools) {

    } else if (id == R.id.nav_share) {

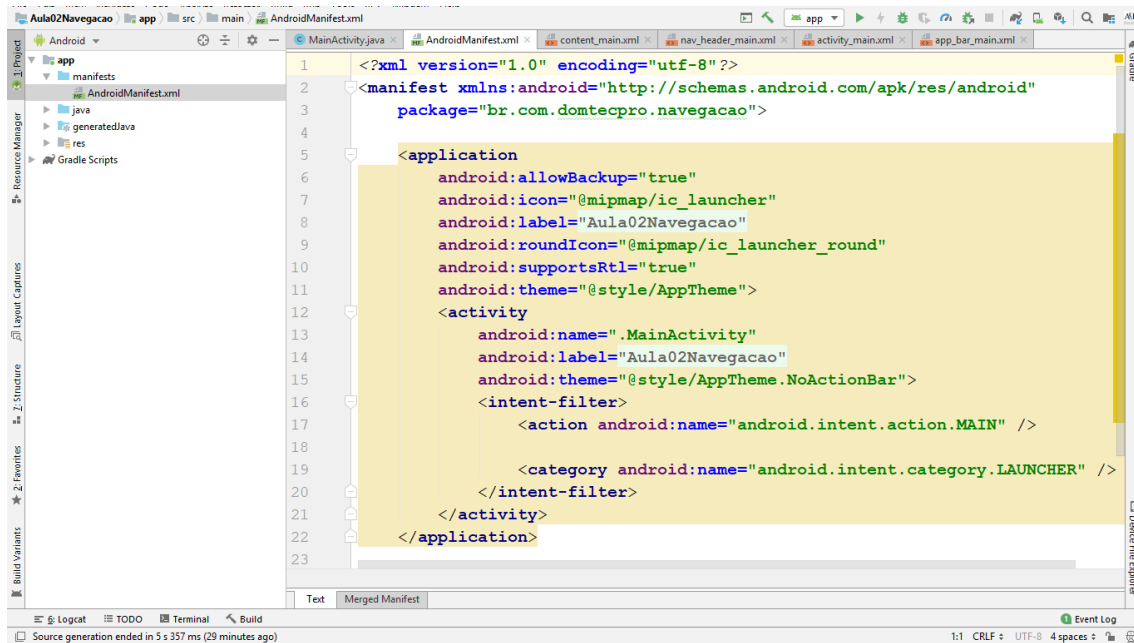
    } else if (id == R.id.nav_send) {

    }

    DrawerLayout drawer = findViewById(R.id.drawer_layout);
    drawer.closeDrawer(GravityCompat.START);
    return true;
}

```

- h. Antes de efetuarmos os testes, devemos solicitar a permissão para a utilização da câmera:
- i. Abra: Android > app > manifests > AndroidManifest.xml

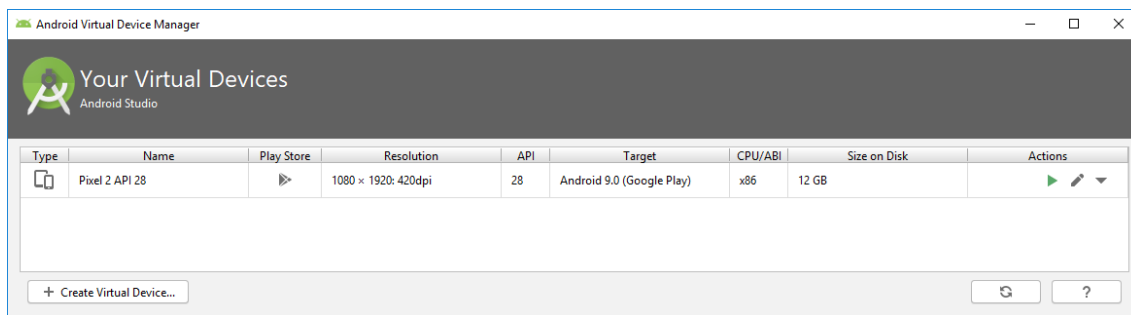


- ii. A partir da linha 4 vamos adicionar as solicitações de permissões para acesso à câmera e ao armazenamento externo (SDCARD).
- iii. Assim fica o código do AndroidManifest.xml

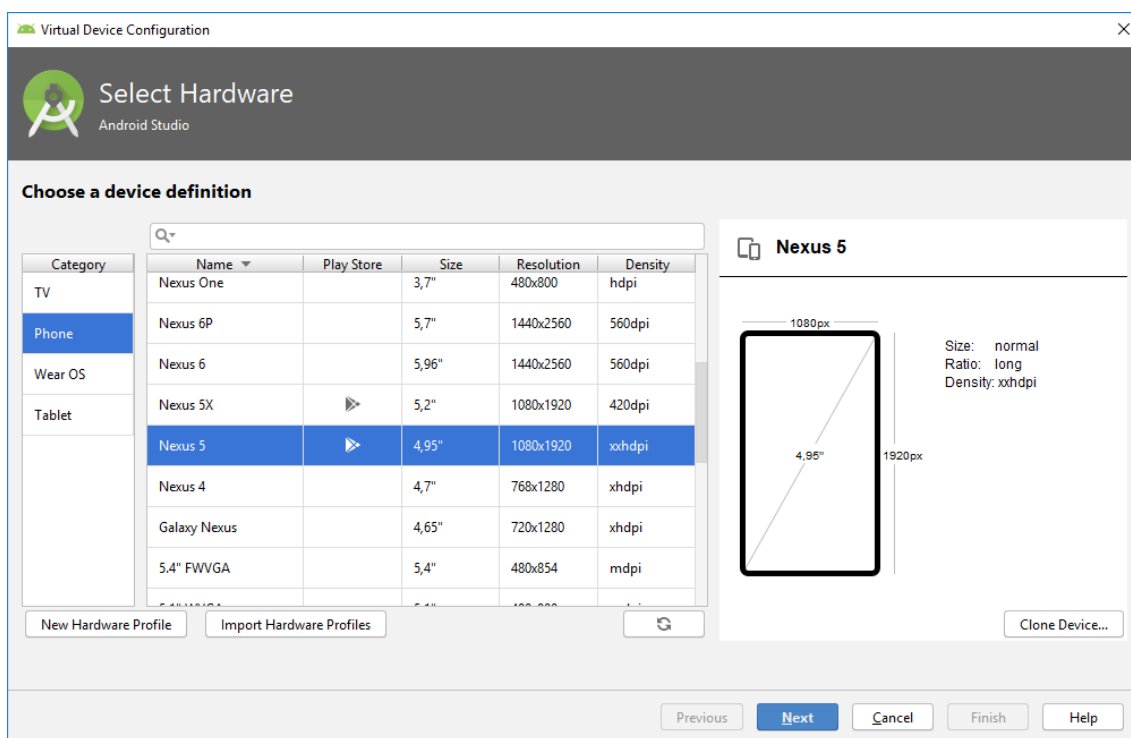
```
<permission android:name="android.permission.CAMERA" />
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
<uses-permission
  android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission
  android:name="android.permission.WRITE_EXTERNAL_STORAGE" />

<uses-feature
  android:name="android.hardware.camera"
  android:required="true" />
<uses-feature android:name="android.hardware.camera.autofocus" />
```

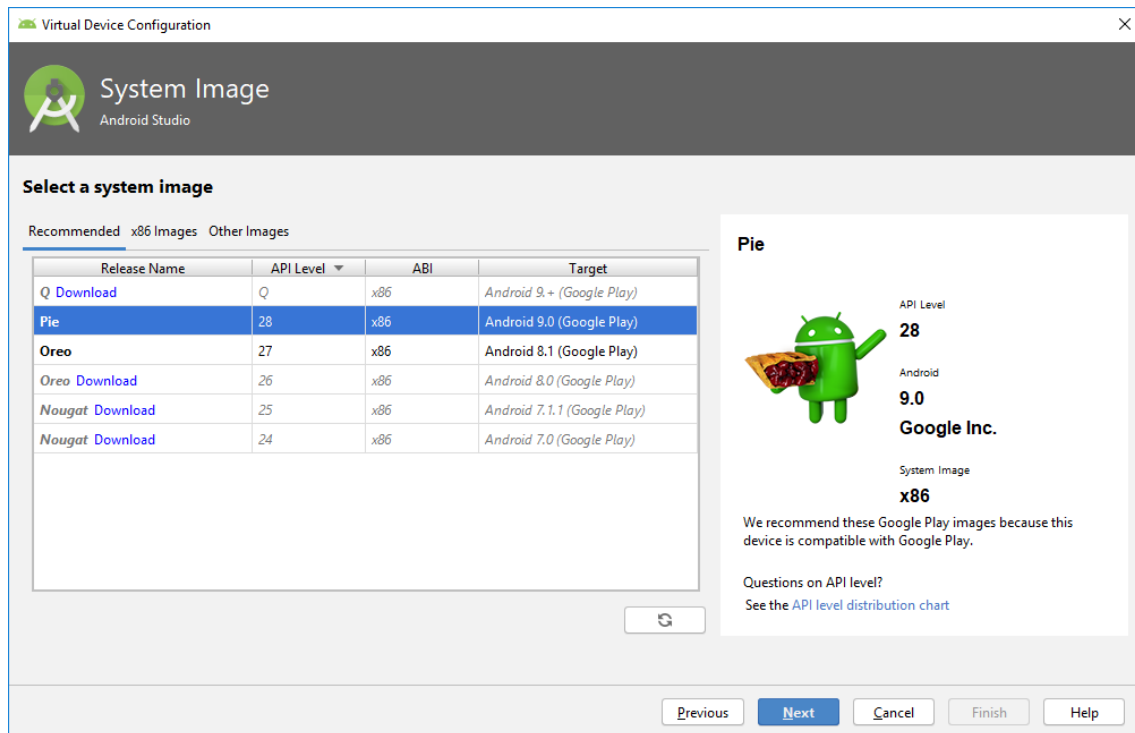
Clique em Tools > AVD Manager e crie sua máquina virtual clicando em Create Virtual Device:



Selecione o aparelho, isto é a dimensão da tela desejada e clique em Next:

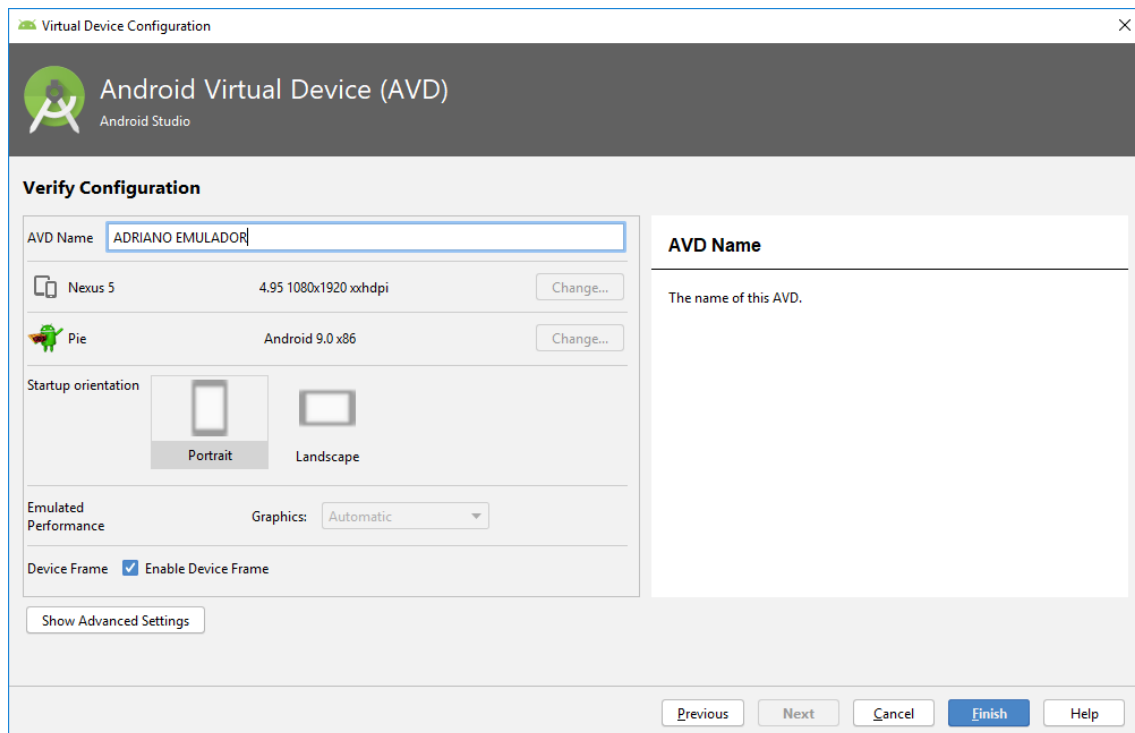


Selecione a imagem de sistema operacional da máquina virtual e clique em Next:

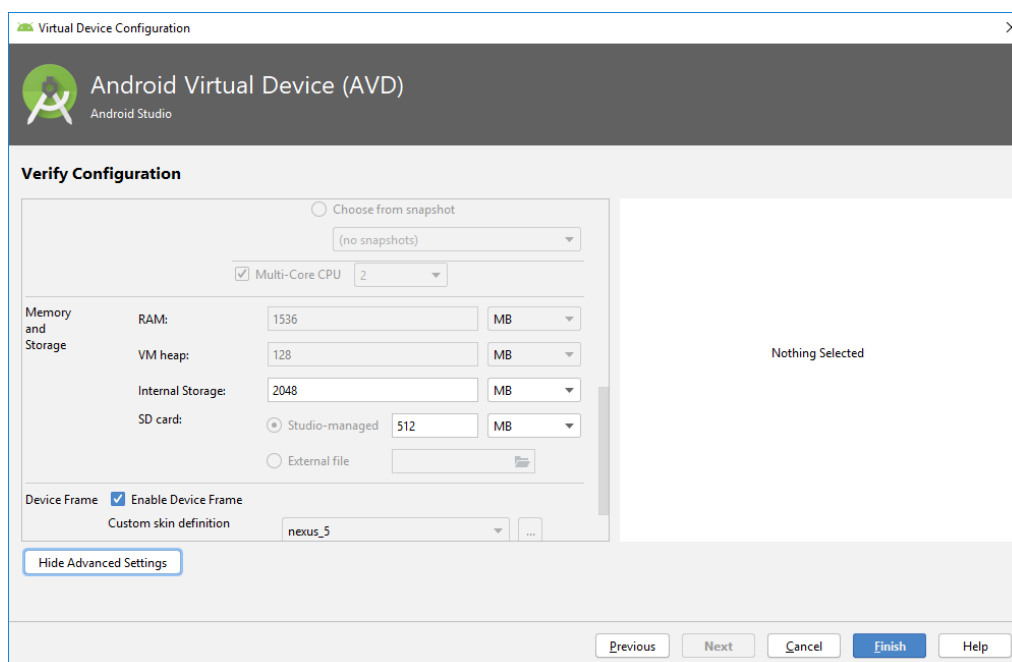


Altere o AVD Name para <SEU NOME> EMULADOR:

Se preferir remova o clique na opção Enable Device Frame se preferir obter apenas a emulação da tela do celular.



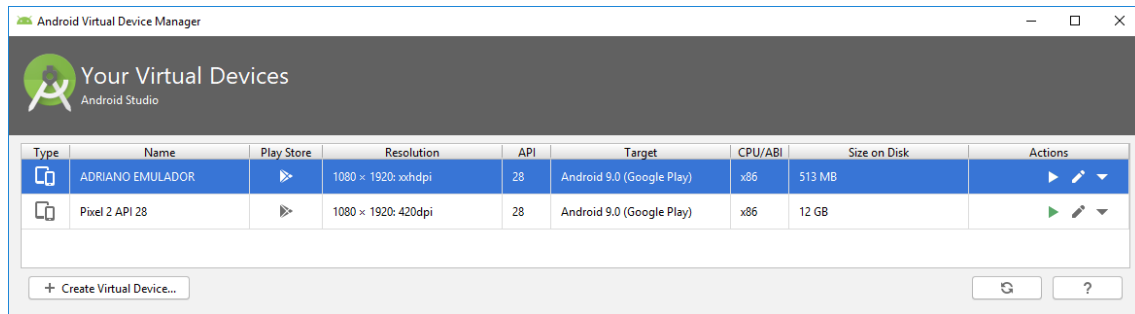
Clique no botão chamado Show Advanced Settings para ajustar a memória do emulador:



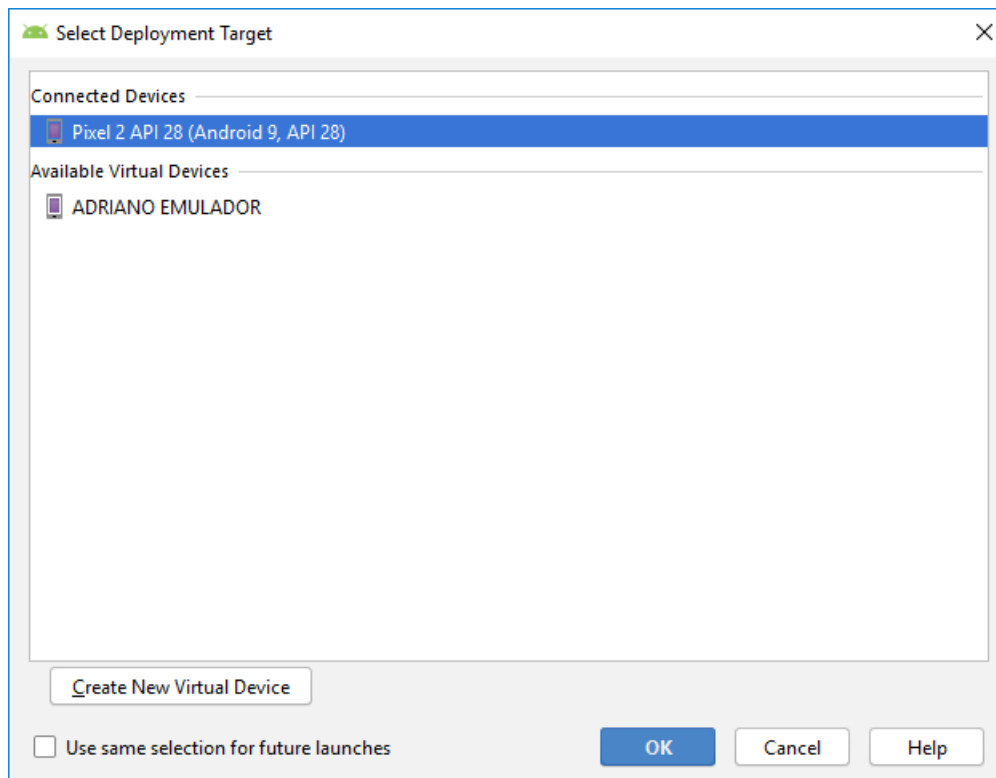
Altere a propriedade RAM para 200Mb e a propriedade VM heap para 32Mb.

Clique em Finish ao terminar os ajustes de memória.

Execute o emulador criado, clicando no ícone de execução (play), para editar a máquina virtual basta clicar na caneta ao lado do play:

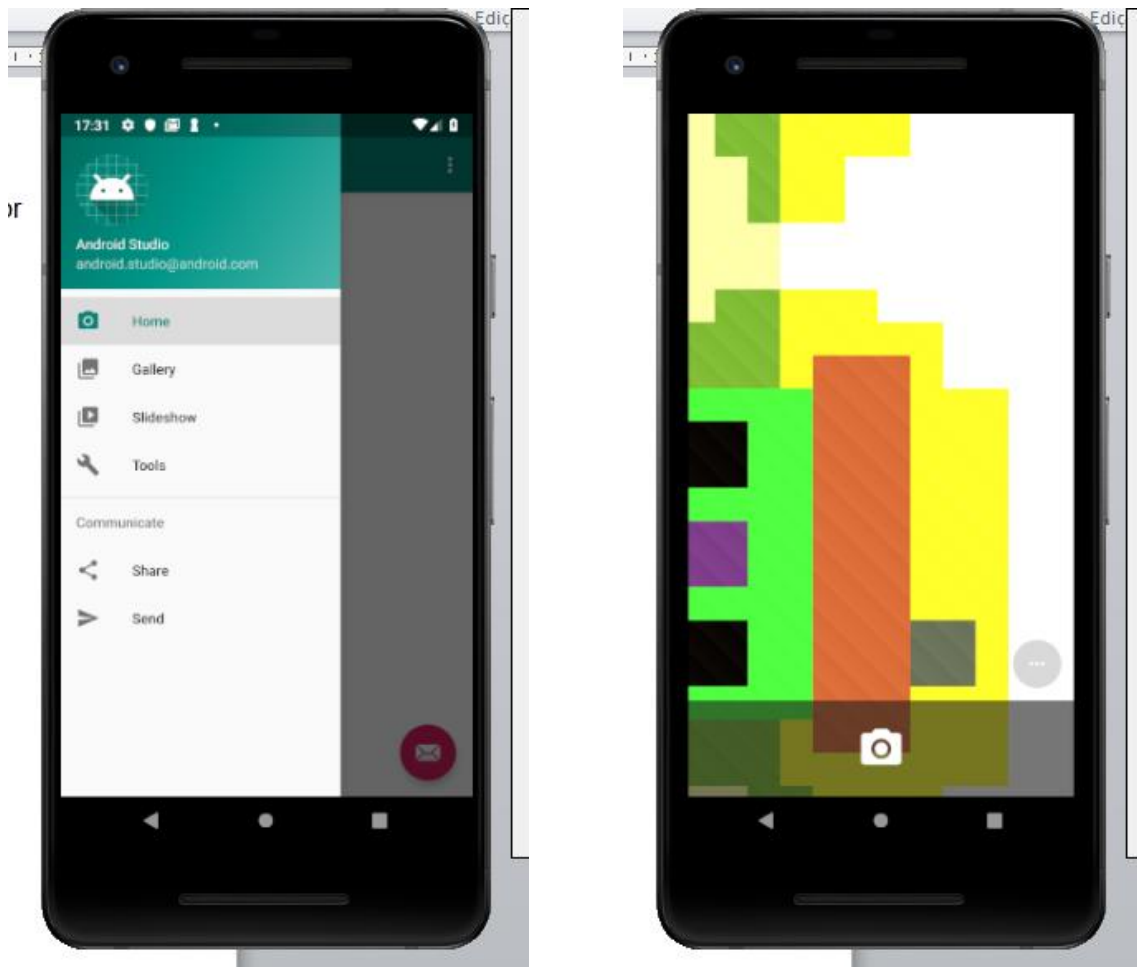


Clique no menu Run > Run app... (Shift+F10)



Escolha o emulador e clique em OK (dê preferência ao aparelho ou emulador em execução em Connected Devices)

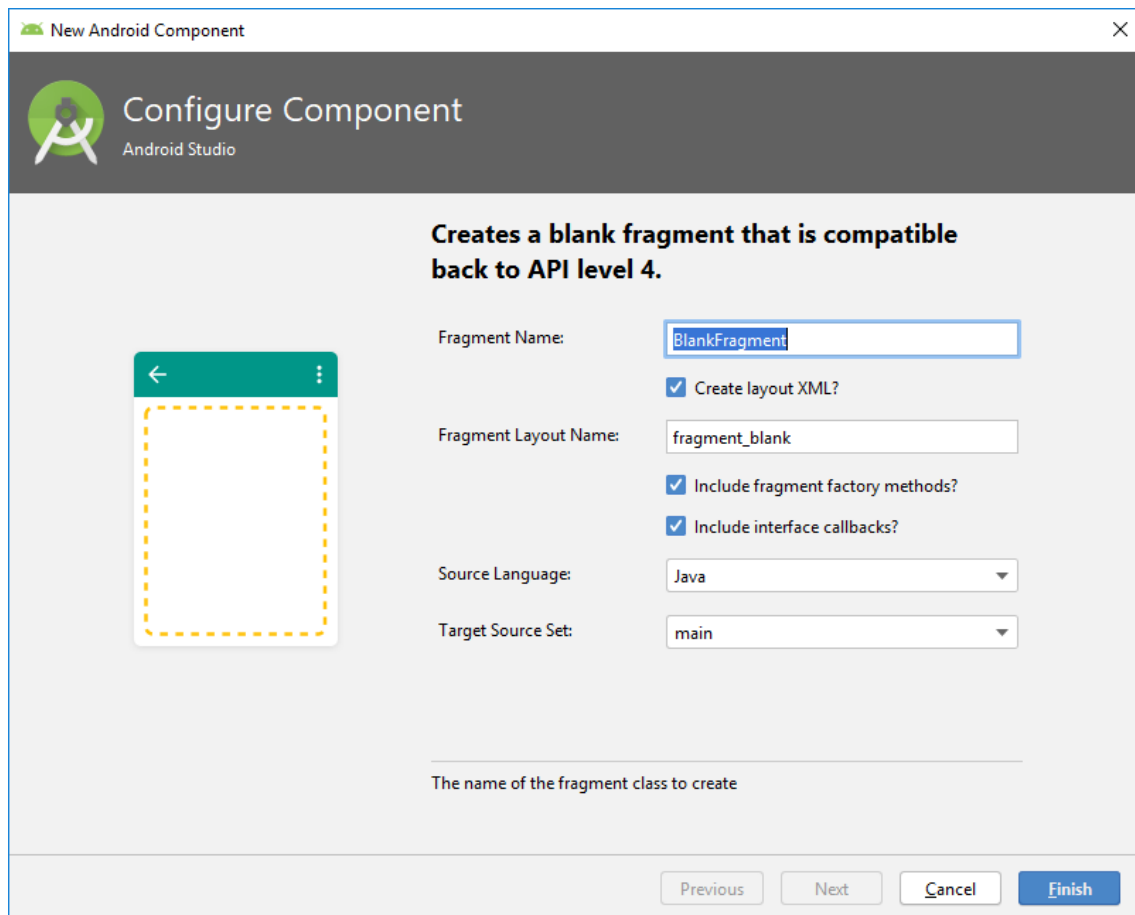
Clique no menu lateral em Home para acionar a câmera:



Obs.: Talvez a câmera não funcione em algumas versões ou aparelhos, vamos abordar estas questões no tutorial de Camera.

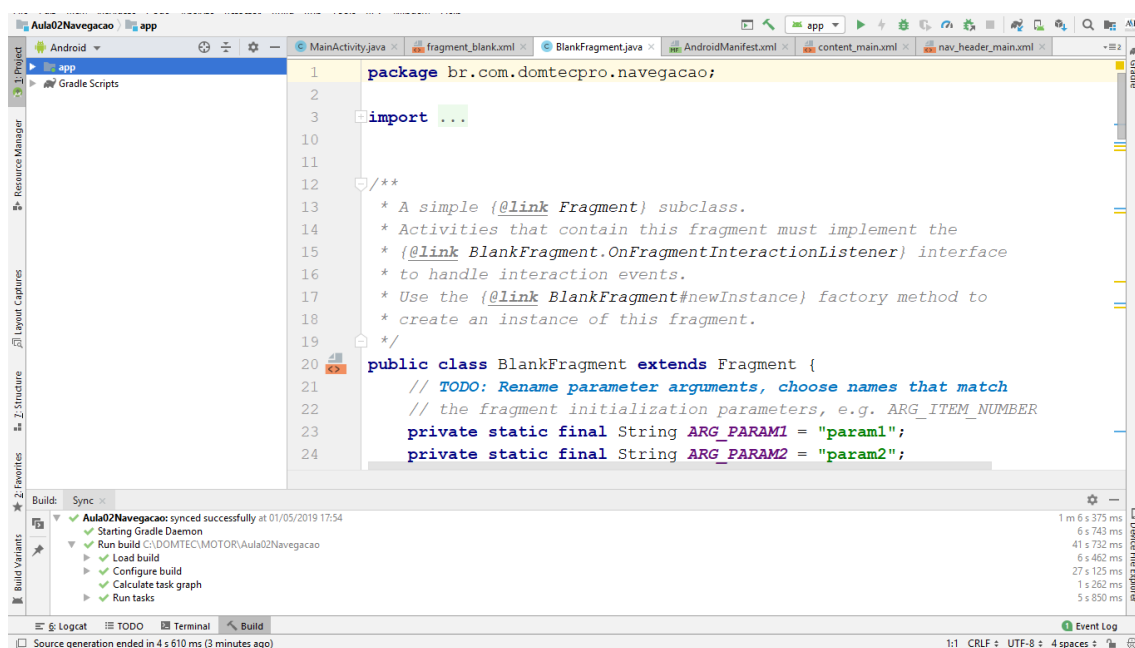
Agora vamos adicionar um fragmento ao projeto, e programar para este fragmento aparecer na Atividade principal do projeto.

- i. Clique em Project > Android > app
 - i. Clique em File > New > Fragment > Fragment (Blank)



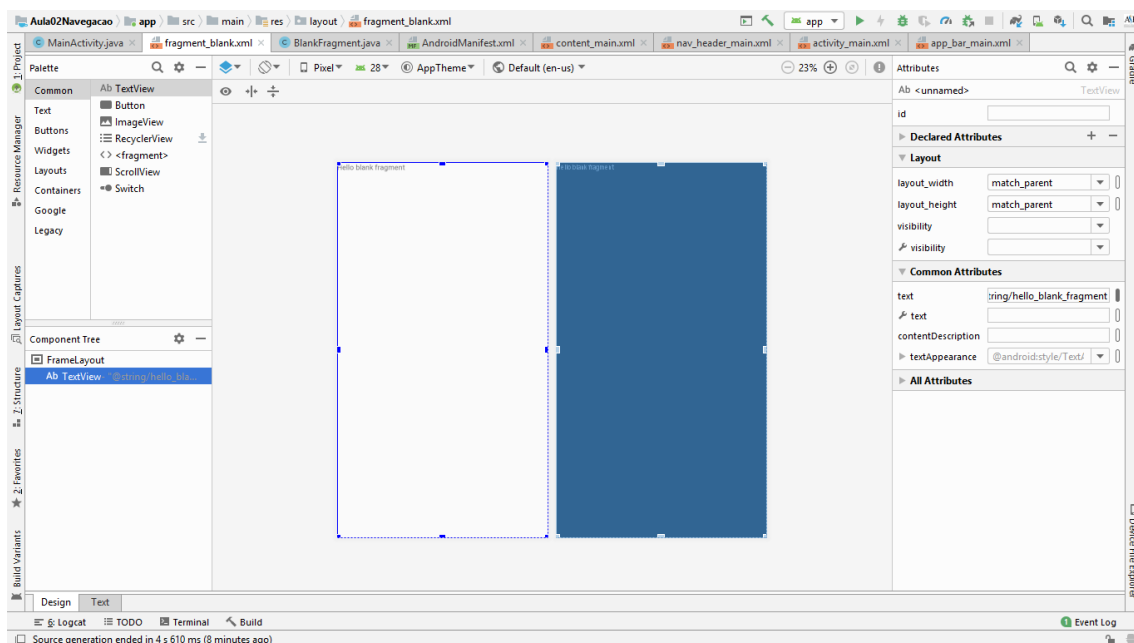
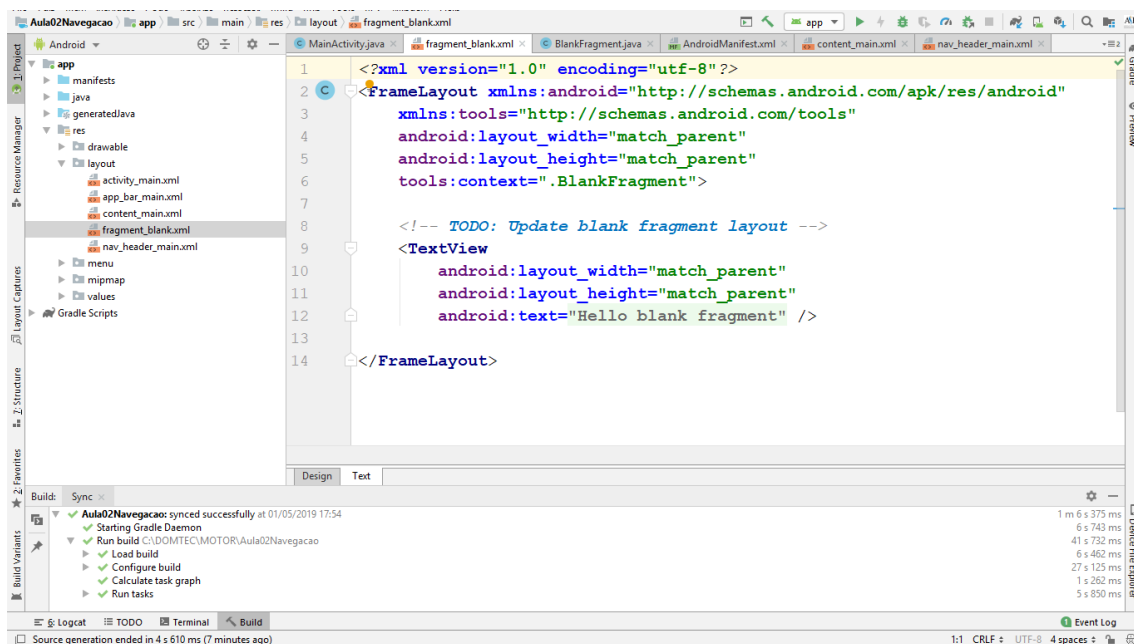
Não altere nada e clique em Finish.

Após o carregamento das telas no projeto aparecerá:



Abra a janela que será controlada por este fragmento:

Clique em Project > Android > add > res > layout > fragment_blank.xml

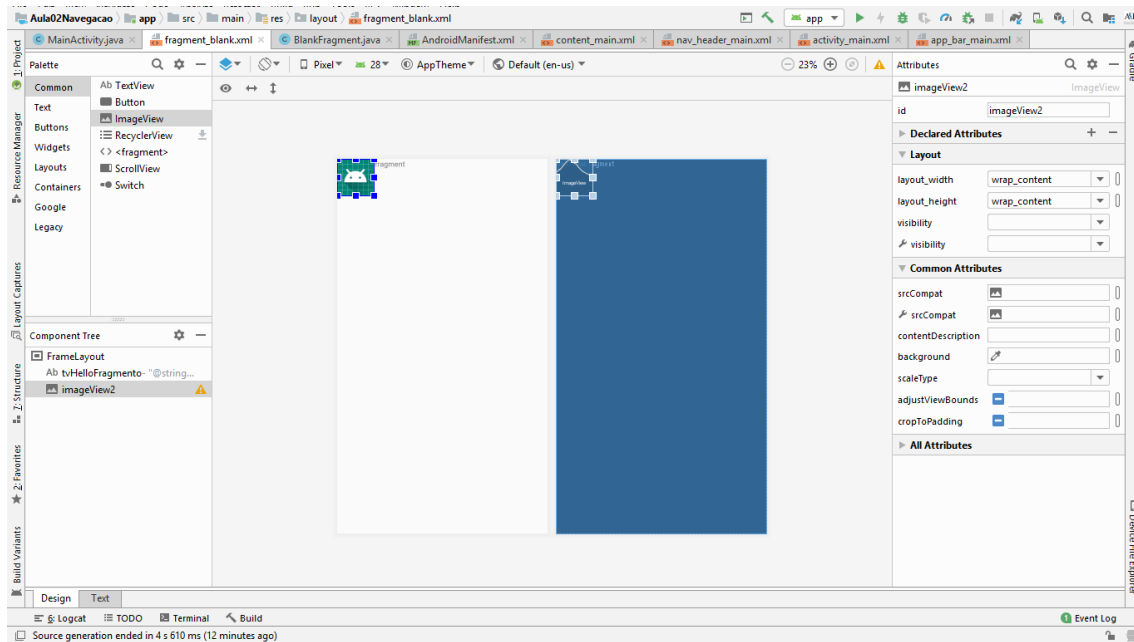


Vamos ajustar a janela acima, onde o TextView esta ocupando totalmente a tela.

Troque os valores dos atributos do TextView deste fragmento:

id: tvHelloFragmento

Adicione um ImageView ao fragment_blank.xml:

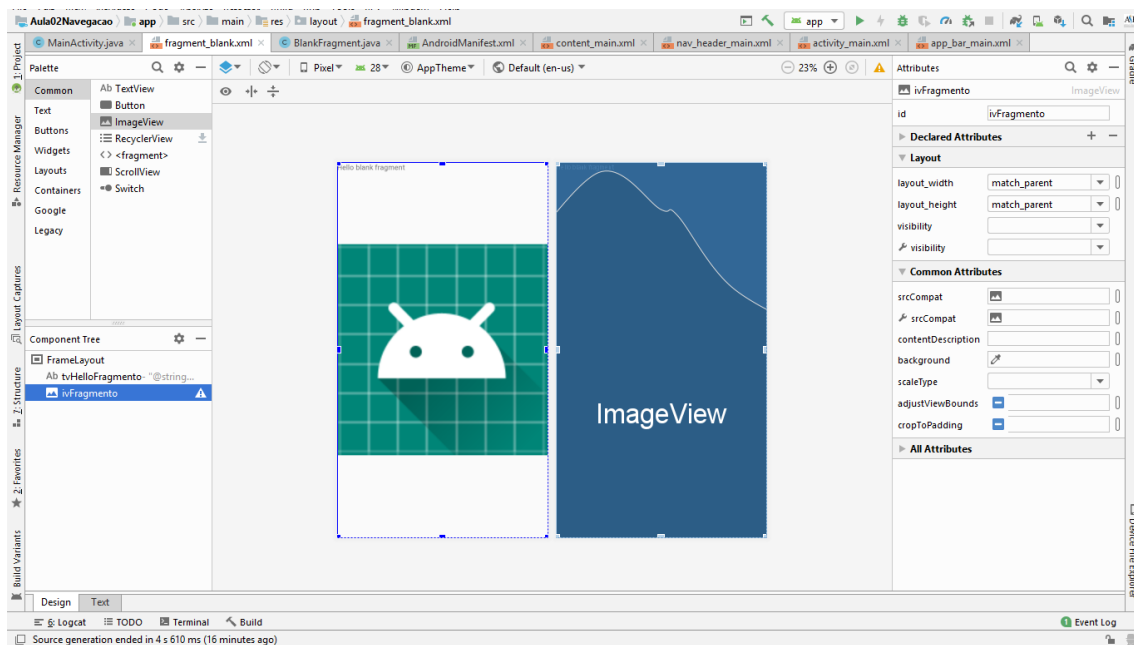


Altere os valores das propriedades:

id: ivFragmento

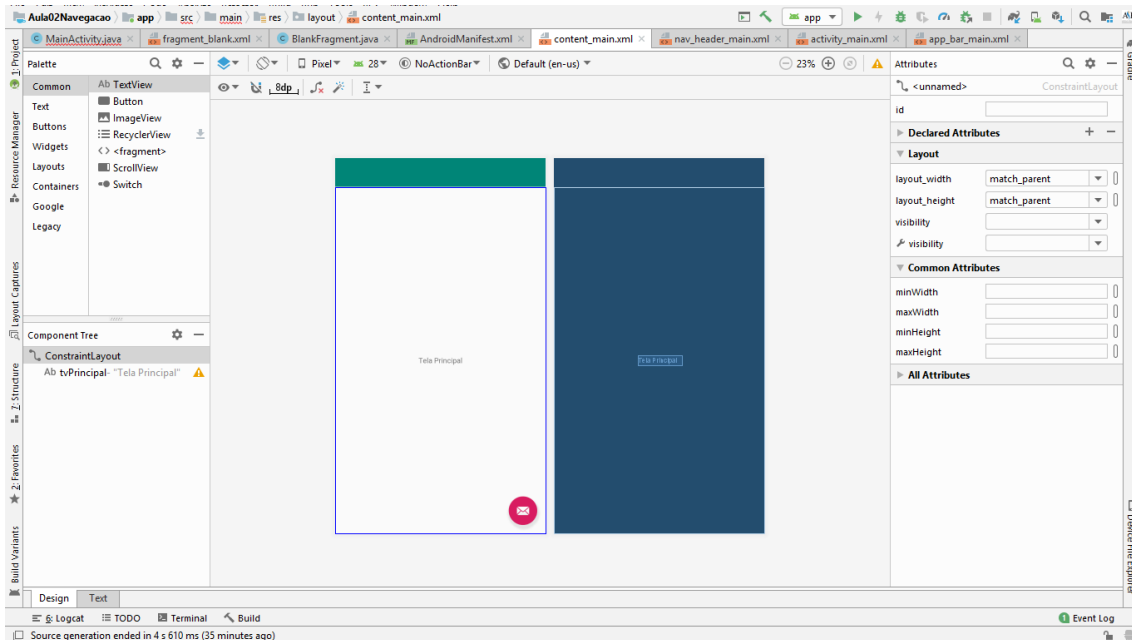
layout_width: match_parent

layout_height: match_parent

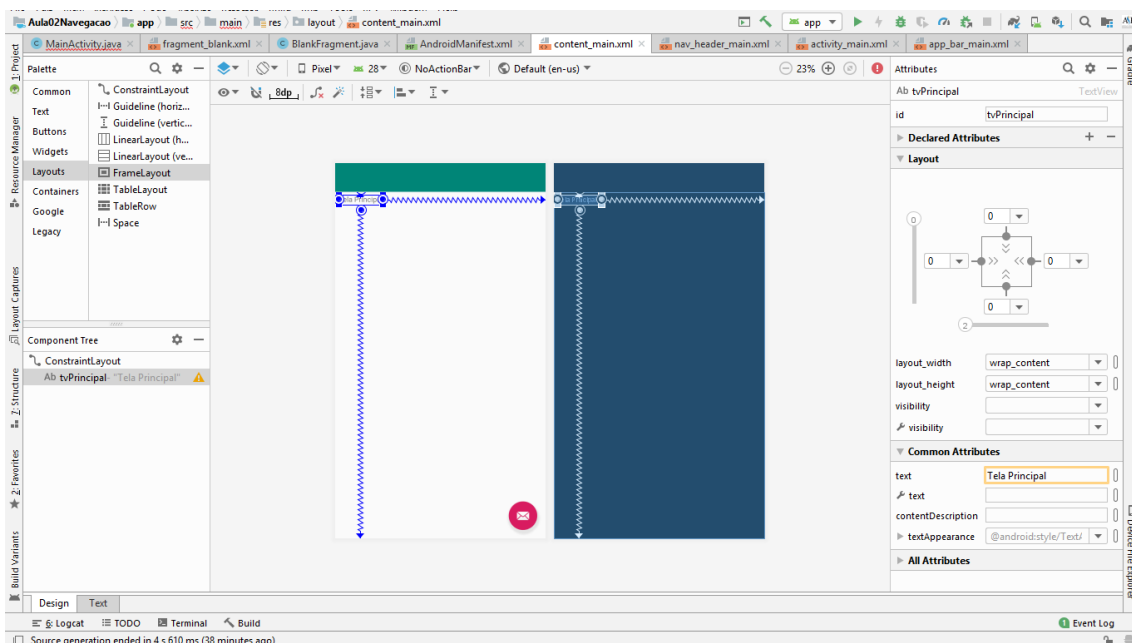


Antes da programação, vamos adicionar um objeto `FrameLayout` ao `content_main.xml`:

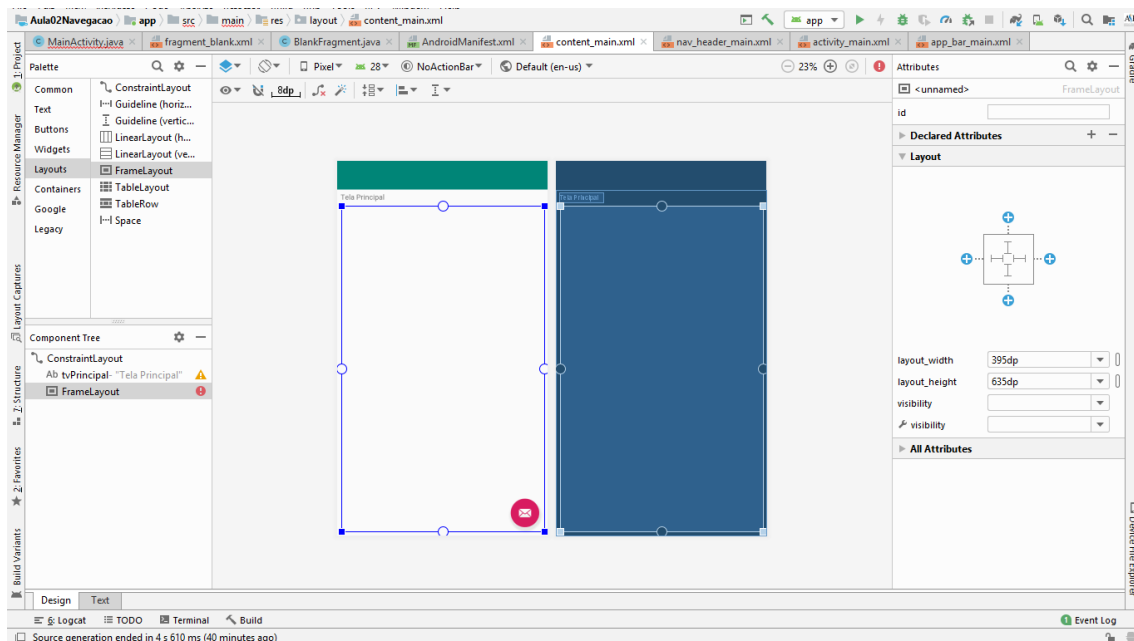
Abra a janela `content_main.xml` clicando em `Project > Android > app > res > layout > content_main.xml`



Altere o posicionamento da `TextView tvPrincipal` como abaixo demonstrado:



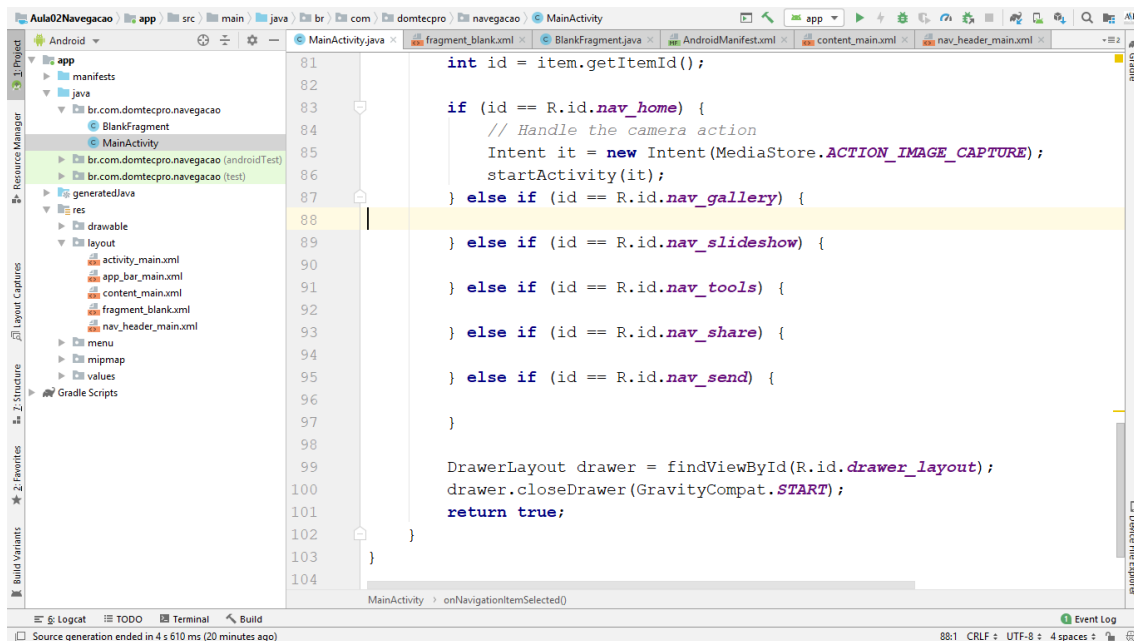
Adicione um objeto `FrameLayout` nesta tela. Ao adicionar, tente arrastar o objeto exatamente para o centro da tela:



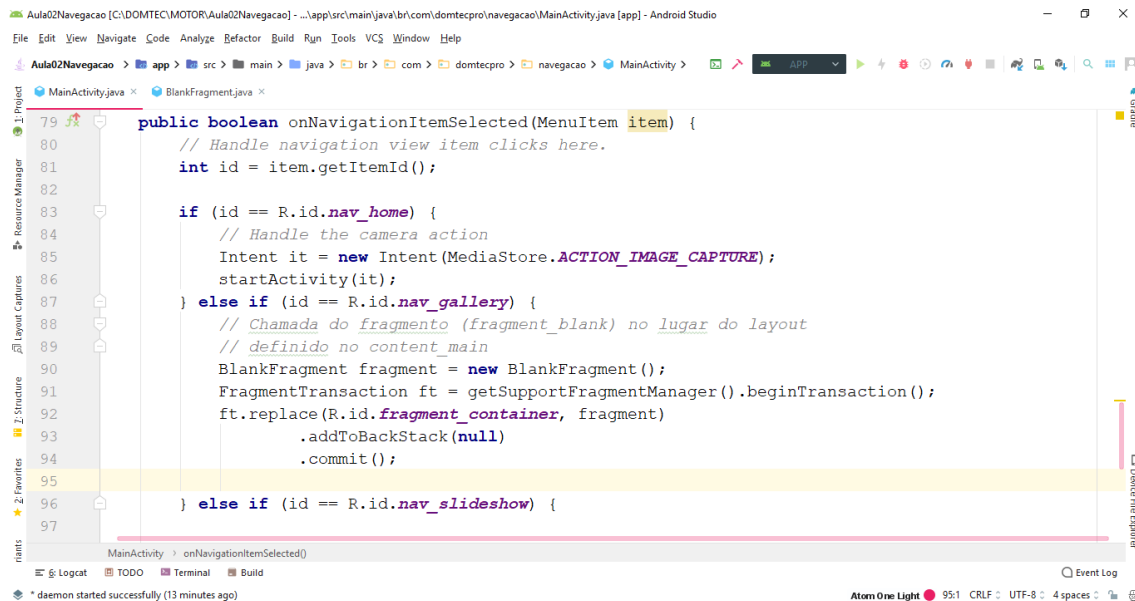
Faça a ancoragem, clicando nos sinais de + definido as margens do objeto.

Altere a propriedade id do FrameLayout para **fragment_container**.

Volte para a classe MainActivity.java



Altere o código para chamar o `fragment_blank.xml`, quando for acionada a opção `gallery` do menu:



Esse pedaço de código deve ser colado na opção **gallery** (`R.id.nav_gallery`) do menu lateral:

```
// Chamada do fragmento (fragment blank) no lugar do FrameLayout
// definido no content_main
BlankFragment fragment = new BlankFragment(); //fragmento criado
// Objeto responsável pela troca de fragmentos
FragmentTransaction ft =
getSupportFragmentManager().beginTransaction();
// Objeto trocando (replace) o fragmento informado (BlankFragment) no
// espaço definido no FrameLayout do content_main.xml
ft.replace(R.id.fragment_container, fragment)
    .addToBackStack(null) // adiciona ícone de retorno
    .commit(); //confirma a alteração do fragmento
```

Volte para o topo da classe `MainActivity` e adicione na assinatura da classe o código:

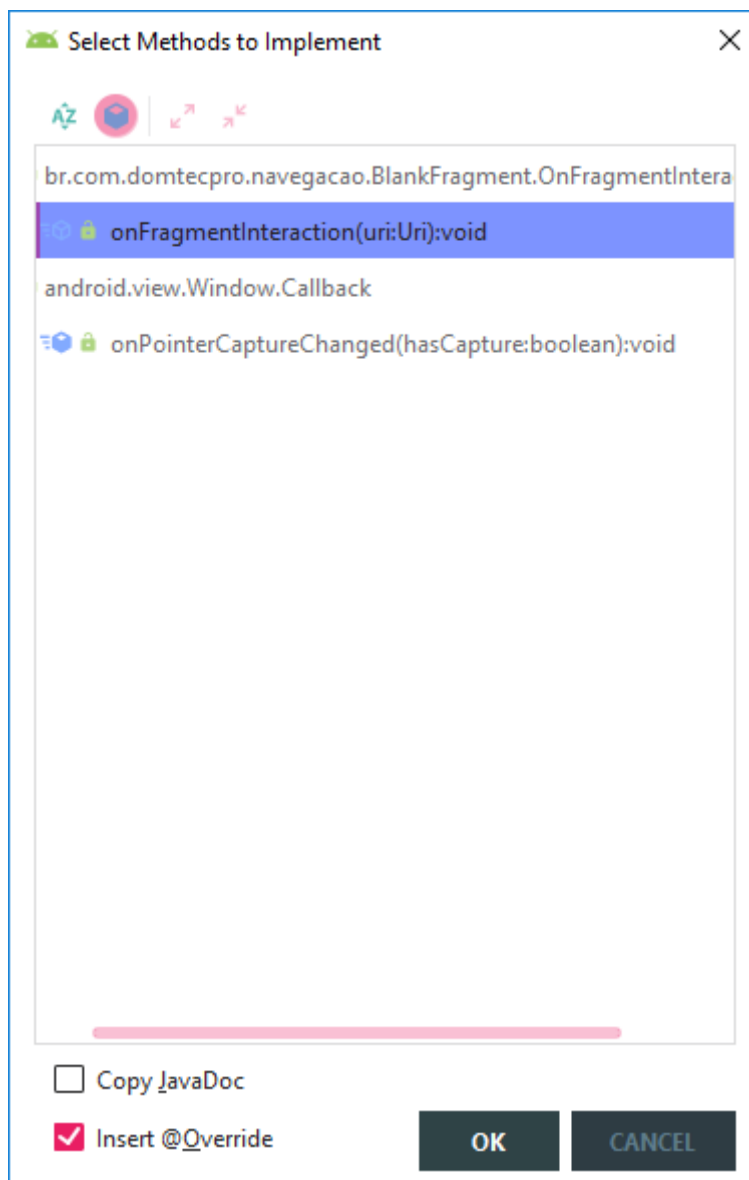
```
, BlankFragment.OnFragmentInteractionListener
```

A assinatura deve ficar assim:

```
public class MainActivity extends AppCompatActivity
    implements NavigationView.OnNavigationItemSelectedListener,
    BlankFragment.OnFragmentInteractionListener {
```

Toda a assinatura ficará marcada com uma linha vermelha, clique uma vez com o mouse, e do lado esquerdo clique na lâmpada vermelha e escolha a opção **Implement methods**.

Esta janela irá aparecer:



Escolha a opção `onFragmentInteraction(uri:Uri):void`

E clique em **OK**

Teste o aplicativo, veja que depois que escolhemos a **gallery** o fragmento não desaparece ao clicarmos em outras opções, vamos corrigir isso.

Vamos fazer desaparecer o **FrameLayout fragment_container**, quando clicarmos em outras opções:

Obs.: As alterações no código estão marcadas de amarelo.

```
@Override
public boolean onNavigationItemSelected(MenuItem item) {
    // Handle navigation view item clicks here.
    int id = item.getItemId();
    // Estabelecer o vínculo entre o FrameLayout da tela e o objeto
    java criado
    FrameLayout fragmentContainer =
    findViewById(R.id.fragment_container);

    if (id == R.id.nav_home) {
        // Handle the camera action
        Intent it = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
        startActivity(it);
        // Determinar visibilidade do FrameLayout fragment_container
        fragmentContainer.setVisibility(View.INVISIBLE);

    } else if (id == R.id.nav_gallery) {
        // Chamada do fragmento (fragment_blank) no lugar do
        FrameLayout
        // definido no content_main
        BlankFragment fragment = new BlankFragment(); //fragmento
        criado
        // Objeto responsável pela troca de fragmentos
        FragmentTransaction ft =
        getSupportFragmentManager().beginTransaction();
        // Objeto trocando (replace) o fragmento informado
        (BlankFragment) no
        // espaço definido no FrameLayout do content main.xml
        ft.replace(R.id.fragment_container, fragment)
            .addToBackStack(null) // adiciona ícone de retorno
            .commit(); //confirma a alteração do fragmento
        // Determinar visibilidade do FrameLayout fragment_container
        fragmentContainer.setVisibility(View.VISIBLE);

    } else if (id == R.id.nav_slideshow) {
        // Determinar visibilidade do FrameLayout fragment_container
        fragmentContainer.setVisibility(View.INVISIBLE);
    } else if (id == R.id.nav_tools) {
        // Determinar visibilidade do FrameLayout fragment_container
        fragmentContainer.setVisibility(View.INVISIBLE);
    } else if (id == R.id.nav_share) {
        // Determinar visibilidade do FrameLayout fragment_container
        fragmentContainer.setVisibility(View.INVISIBLE);
    } else if (id == R.id.nav_send) {
        // Determinar visibilidade do FrameLayout fragment_container
        fragmentContainer.setVisibility(View.INVISIBLE);
    }

    DrawerLayout drawer = findViewById(R.id.drawer_layout);
    drawer.closeDrawer(GravityCompat.START);
    return true;
}
```

Referência Bibliográfica

- [1] <http://developer.android.com>. Acessado em 28/04/2019.
- [2] <https://stackoverflow.com/questions/17398640/android-support-v4-app-fragmenttransaction-required>. Acessado em 01/05/2019.
- [3] <https://developer.android.com/training/basics/fragments/communicating.html>. Acessado em 01/05/2019.
- [4] <https://developer.android.com/guide/components/fragments.html>. Acessado em 01/05/2019.