

Ficha do Aplicativo

Aplicativo Requisições HTTP

Resumo

Este aplicativo conecta-se a um servidor remoto através de requisições HTTP (Hiper Text Transfer Protocol).

Objetivos de Aprendizagem

1. Compreender os conceitos de conexão entre o aplicativo e um servidor remoto, utilizando requisições HTTP
2. Utilizar o conjunto de bibliotecas OkHTTP3
3. Utilizar a classe Picasso para carga e tratamento de imagens
4. Configurar dependências no Gradle
5. Liberar permissão de uso da internet no aplicativo, usando o AndroidManifest
6. Enviar requisições assíncronas a um servidor, obter e tratar a resposta a essas requisições
7. Carregar imagens a partir do Servidor, usando a classe Picasso
8. Carregar um Spinner a partir de dados obtidos no servidor, usando Adapter
9. Carregar um ListView a partir de dados obtidos no servidor, usando Adapter

Dinâmica do Aplicativo


A tela inicial do aplicativo contém quatro botões que permitirão:

- a) Carregar uma página web a partir de uma url fornecida
- b) Carregar um arquivo texto disponível em um servidor remoto
- c) Carregar um spinner a partir das linhas de dados contidas em um arquivo texto no servidor. Essas linhas conterão nomes de imagens que poderão ser carregadas a partir da escolha de um item desse spinner
- d) Carregar um ListView a partir de um arquivo texto disponível no servidor, no formato CSV (Comma Separated Values). Este ListView será personalizado com a criação de um layout e um Adapter específicos.

Lista de Activities do Aplicativo

Nome	Layout
MainActivity.java	activity_main.xml
ObtemPagina.java	activity_obtem_pagina.xml
ObtemArqTexto.java	activity_obtem_arqtexto.xml
ObtemImagem.java	activity_obtem_imagem.xml
ObtemDados.java	activity_obtem_dados.xml

Drawables deste Aplicativo

Nome do arquivo	Dimensões	Tipo	Imagem
naocarregado.png	567 x 567	Imagem PNG	

Referências conceituais

Para um conteúdo sobre requisições HTTP acesse: <https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Messages>

Para saber mais sobre a biblioteca de Classes OkHttp acesse: <https://square.github.io/okhttp>

Para saber mais sobre a biblioteca de Classes Picasso acesse: <https://square.github.io/picasso>

Resources

strings.xml

```
<resources>
  <string name="app_name">Requisições HTTP</string>
  <string name="btnpagina">Obter Página</string>
  <string name="btntexto">Obter Texto</string>
  <string name="btnimagem">Obter Imagem</string>
  <string name="btndados">Obter Dados</string>
  <string name="btnexecreqpag">Executa Requisição - Carrega página</string>
  <string name="btnexecreqtxt">Executa Requisição - Obtém texto</string>
  <string name="btnexecreqimg">Executa Requisição - Monta Spinner</string>
  <string name="btnexecreqdados">Executa Requisição - Carrega Dados</string>

  <string name="rotulo_reqpag">Página obtida</string>
  <string name="rotulo_reqtxt">Texto obtido do servidor</string>
  <string name="rotulo_reqimg">Spinner carregado a partir do servidor</string>
  <string-array name="vazio">
    <item>...</item>
  </string-array>
</resources>
```

colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <color name="purple_200">#FFB86FC</color>
  <color name="purple_500">#FF6200EE</color>
  <color name="purple_700">#FF3700B3</color>
  <color name="teal_200">#FF03DAC5</color>
  <color name="teal_700">#FF018786</color>
  <color name="black">#FF000000</color>
  <color name="white">#FFFFFFFF</color>
  <color name="corFundo">#FFF5A2</color>
  <color name="corCxTexto">#FFFAD3</color>
  <color name="corTexto">#2E0D70</color>
</resources>
```

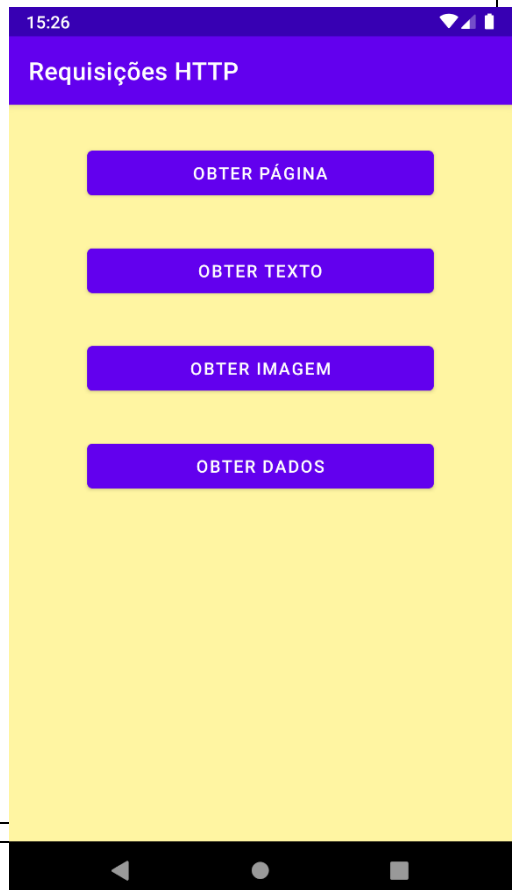
Detalhamento das Activities

Layout: activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:background="@color/corFundo"
  tools:context=".MainActivity">

  <Button
    android:id="@+id/btnObterPagina"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="64dp"
    android:layout_marginTop="32dp"
    android:layout_marginEnd="64dp"
    android:onClick="btnObterPaginaOnClick"
    android:text="@string/btnpagina"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="1.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

  <Button
    android:id="@+id/btnObterTexto"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="64dp"
    android:layout_marginTop="32dp"
    android:layout_marginEnd="64dp"
    android:onClick="btnObterTextoOnClick"
    android:text="@string/btntexto"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="1.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/btnObterPagina" />
```



```

<Button
    android:id="@+id/btnObterImagem"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="64dp"
    android:layout_marginTop="32dp"
    android:layout_marginEnd="64dp"
    android:onClick="btnObterImagemOnClick"
    android:text="@string/btnimagem"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="1.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/btnObterTexto" />

<Button
    android:id="@+id/btnObterDados"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="64dp"
    android:layout_marginTop="32dp"
    android:layout_marginEnd="64dp"
    android:onClick="btnObterDadosOnClick"
    android:text="@string/btndados"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="1.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/btnObterImagem" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

Código: MainActivity.java

```

public class MainActivity extends AppCompatActivity {

    public static final String URL_RAIZ = "http://www.clictec.com.br/ilp506/reqhttp/";

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

    public void btnObterPaginaOnClick (View view) {
        Intent intent = new Intent(this, ObtemPagina.class);
        startActivity(intent);
    }

    public void btnObterTextoOnClick (View view) {
        Intent intent = new Intent(this, ObtemArqTexto.class);
        startActivity(intent);
    }

    public void btnObterImagemOnClick (View view) {
        Intent intent = new Intent(this, ObtemImagem.class);
        startActivity(intent);
    }

    public void btnObterDadosOnClick (View view) {
        Intent intent = new Intent(this, ObtemDados.class);
        startActivity(intent);
    }

}

```

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/corFundo"
    tools:context=".ObtemPagina">

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="8dp"
        android:text="URL"
        android:textColor="@color/corTexto"
        app:layout_constraintBaseline_toBaselineOf="@+id/edtURL"
        app:layout_constraintStart_toStartOf="parent" />

    <EditText
        android:id="@+id/edtURL"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="16dp"
        android:ems="10"
        android:hint="Nome do Arquivo"
        android:inputType="text"
        android:text="http://www.clictec.com.br"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toEndOf="@+id/textView1"
        app:layout_constraintTop_toTopOf="parent" />

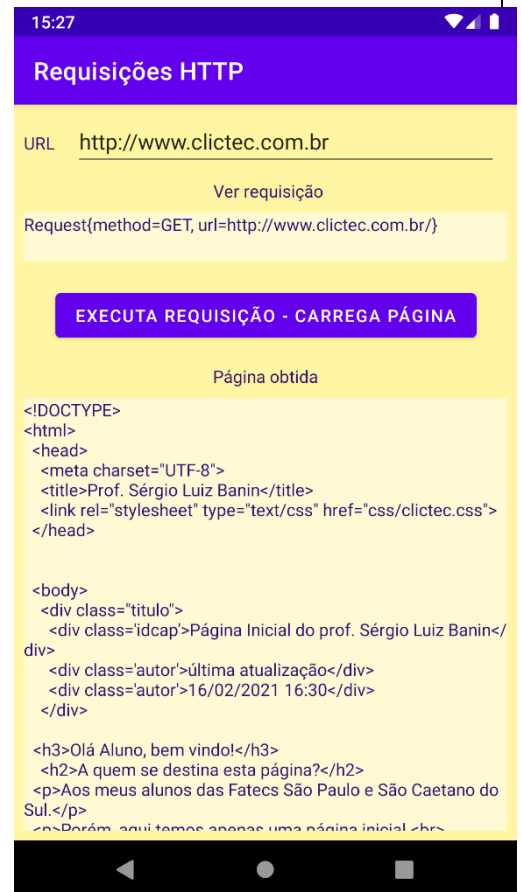
    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:text="Ver requisição"
        android:textColor="@color/corTexto"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.507"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/edtURL" />

    <TextView
        android:id="@+id/txtVerRequisicao"
        android:layout_width="0dp"
        android:layout_height="40dp"
        android:layout_marginStart="8dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="8dp"
        android:background="@color/corCxTexto"
        android:textColor="@color/corTexto"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView2"
        tools:text="txtVerRequisicao" />

    <Button
        android:id="@+id/btnCarregaPagina"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="96dp"
        android:onClick="btnCarregaPaginaOnClick"
        android:text="@string/btnexecreqpag"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.507"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/edtURL" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="@string/rotulo_reqpag"

```



```

    android:textColor="@color/corTexto"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/btnCarregaPagina" />

```

```

<TextView
    android:id="@+id/txtVerPagina"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_marginStart="8dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="8dp"
    android:layout_marginBottom="8dp"
    android:background="@color/corCxTexto"
    android:textColor="@color/corTexto"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView3"
    tools:text="txtVerPagina" />

```

```

</androidx.constraintlayout.widget.ConstraintLayout>

```

Código: ObtemPagina.java

```

public class ObtemPagina extends AppCompatActivity {

    private OkHttpClient reqHTTP;

    private EditText edtURL;
    private TextView txtVerRequisicao;
    private TextView txtVerPagina;

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

        reqHTTP = new OkHttpClient();

        edtURL = findViewById(R.id.edtURL);
        txtVerRequisicao = findViewById(R.id.txtVerRequisicao);
        txtVerRequisicao.setMovementMethod(new ScrollingMovementMethod());
        txtVerPagina = findViewById(R.id.txtVerPagina);
        txtVerPagina.setMovementMethod(new ScrollingMovementMethod());
    }

    public void btnCarregaPaginaOnClick(View view) {
        executaRequisicao();
    }

    private void executaRequisicao() {
        String url = edtURL.getText().toString().toLowerCase();
        Request request = new Request.Builder()
            .url(url)
            .build();
        txtVerRequisicao.setText(request.toString());

        Call call = reqHTTP.newCall(request);
        call.enqueue(new Callback() {
            @Override
            public void onFailure(@NotNull Call call, @NotNull IOException e) {
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        txtVerPagina.setText("Falha de conexão\n" + e.getMessage());
                    }
                });
            }
        });

        @Override
        public void onResponse(@NotNull Call call, @NotNull Response response) throws IOException {
            runOnUiThread(new Runnable() {
                @Override
                public void run() {
                    try {
                        txtVerPagina.setText(response.body().string());
                        txtVerPagina.setScrollY(0);
                    }
                }
            });
        }
    }
}

```



```

app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/textView5" />

```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

Código: ObtemArqTexto.java

```

public class ObtemArqTexto extends AppCompatActivity {

    private OkHttpClient reqHTTP;

    private EditText edtArqTexto;
    private TextView txtTextoObtido;

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

        reqHTTP = new OkHttpClient();

        edtArqTexto = findViewById(R.id.edtArqTexto);
        txtTextoObtido = findViewById(R.id.txtTextoObtido);
        txtTextoObtido.setMovementMethod(new ScrollingMovementMethod());
    }

    public void btnObtemTextoOnClick(View view) {
        executaRequisicao();
    }

    private void executaRequisicao() {
        String url = MainActivity.URL_RAIZ + edtArqTexto.getText().toString().toLowerCase();
        Request request = new Request.Builder()
            .url(url)
            .build();

        Call call = reqHTTP.newCall(request);
        call.enqueue(new Callback() {
            @Override
            public void onFailure(@NotNull Call call, @NotNull IOException e) {
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        txtTextoObtido.setText("Falha de conexão - " + e.getMessage());
                    }
                });
            }
        });

        @Override
        public void onResponse(@NotNull Call call, @NotNull Response response) throws IOException {
            runOnUiThread(new Runnable() {
                @Override
                public void run() {
                    try {
                        txtTextoObtido.setText(response.body().string());
                    } catch (IOException e) {
                        txtTextoObtido.setText("Erro no tratamento da mensagem - " + e.getMessage());
                    }
                }
            });
        }
    });
}
}

```

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/corFundo"
    tools:context=".ObtemImagem">

    <TextView
        android:id="@+id/textView6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="8dp"
        android:text="Qual arquivo carregar?"
        android:textColor="@color/corTexto"
        app:layout_constraintBaseline_toBaselineOf="@+id/edtNomeImagens"
        app:layout_constraintStart_toStartOf="parent" />

    <EditText
        android:id="@+id/edtNomeImagens"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="16dp"
        android:ems="10"
        android:hint="Nome do Arquivo"
        android:inputType="text"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toEndOf="@+id/textView6"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/btnExecutaRequisicao"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:onClick="btnExecutaRequisicaoOnClick"
        android:text="@string/btnexecreqimg"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/edtNomeImagens" />

    <TextView
        android:id="@+id/textView7"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="@string/rotulo_reqimg"
        android:textColor="@color/corTexto"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/btnExecutaRequisicao" />

    <Spinner
        android:id="@+id/spnImagens"
        android:layout_width="0dp"
        android:layout_height="30dp"
        android:layout_marginStart="16dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="16dp"
        android:background="@color/corCxTexto"
        android:entries="@array/vazio"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView7" />

    <ImageView
        android:id="@+id/imgCarregada"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:layout_marginBottom="16dp"
        app:layout_constraintBottom_toTopOf="@+id/txtStatus"
        app:layout_constraintEnd_toEndOf="parent"

```




```

app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/spnImagens"
app:srcCompat="@drawable/naocarregado" />

<TextView
    android:id="@+id/txtStatus"
    android:layout_width="0dp"
    android:layout_height="32dp"
    android:layout_marginStart="16dp"
    android:layout_marginEnd="16dp"
    android:layout_marginBottom="4dp"
    android:textSize="12sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

Código: ObtemImagem.java

```

public class ObtemImagem extends AppCompatActivity {

    private OkHttpClient reqHttp;

    private EditText edtNomeImagens;
    private TextView txtStatus;
    private Spinner spnImagens;
    private ImageView imgCarregada;

    private List<String> listaNomesImgs;

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

        reqHttp = new OkHttpClient();

        edtNomeImagens = findViewById(R.id.edtNomeImagens);
        txtStatus = findViewById(R.id.txtStatus);
        spnImagens = findViewById(R.id.spnImagens);
        imgCarregada = findViewById(R.id.imgCarregada);
    }

    public void btnExecutaRequisicaoOnClick(View view) {
        executaRequisicao();
    }

    private void executaRequisicao() {
        String url = MainActivity.URL_RAIZ + edtNomeImagens.getText().toString().toLowerCase();
        Request request = new Request.Builder()
            .url(url)
            .build();
        Call call = reqHttp.newCall(request);
        call.enqueue(new Callback() {
            @Override
            public void onFailure(@NotNull Call call, @NotNull IOException e) {
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        txtStatus.setText("Falha de conexão - " + e.getMessage());
                    }
                });
            }

            @Override
            public void onResponse(@NotNull Call call, @NotNull Response response) throws IOException {
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        try {
                            montaSpnImagens(response.body().string());
                        }
                        catch (IOException e) {
                            txtStatus.setText("Falha durante carga dos dados.");
                        }
                    }
                });
            }
        });
    }
}

```

```

}

private void montaSpnImagens(String nomesImgs) {
    //this.nomesImgs = nomesImgs;
    txtStatus.setText(nomesImgs);
    listaNomesImgs = Arrays.asList(nomesImgs.split("\n"));
    ArrayAdapter<String> adaptador = new ArrayAdapter<>(
        this,
        android.R.layout.simple_spinner_item,
        listaNomesImgs
    );
    spnImagens.setAdapter(adaptador);
    setSpnImagensOnItemSelected();
}

private void setSpnImagensOnItemSelected() {
    spnImagens.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
        @Override
        public void onItemSelected(AdapterView<?> adapterView, View view, int position, long l) {
            carregaImagem(position);
        }

        @Override
        public void onNothingSelected(AdapterView<?> adapterView) {
            imgCarregada.setImageResource(R.drawable.nao_carregado);
            txtStatus.setText("");
        }
    });
}

private void carregaImagem(int i) {
    String nomeImagem = listaNomesImgs.get(i);
    String url = MainActivity.URL_RAIZ + nomeImagem;
    txtStatus.setText(url);
    Picasso.get()
        .load(url)
        .memoryPolicy(MemoryPolicy.NO_CACHE)
        .networkPolicy(NetworkPolicy.NO_CACHE)
        .into(imgCarregada);
}
}
}

```

Layout: activity_obtem_dados.xml

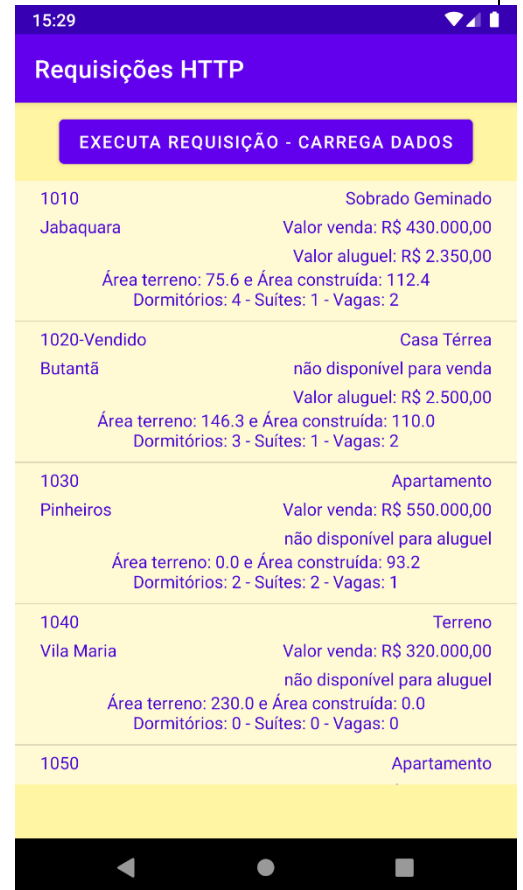
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/corFundo"
    tools:context=".ObtemDados">

    <Button
        android:id="@+id/btnCarregaDados"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:onClick="btnCarregaDadosOnClick"
        android:text="@string/btnexecreqdados"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <ListView
        android:id="@+id/lvDados"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:layout_marginTop="8dp"
        android:layout_marginBottom="8dp"
        android:background="@color/corCxTexto"
        app:layout_constraintBottom_toTopOf="@+id/txtStatus"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/btnCarregaDados" />

    <TextView
        android:id="@+id/txtStatus"
        android:layout_width="0dp"
        android:layout_height="32dp"
        android:layout_marginStart="16dp"
        android:layout_marginEnd="16dp"
        android:layout_marginBottom="4dp"
        android:textSize="12sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```



Código: ObtemDados.java

```
public class ObtemDados extends AppCompatActivity {

    private OkHttpClient reqHttp;

    private ListView lvDados;
    private TextView txtStatus;

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

        reqHttp = new OkHttpClient();

        lvDados = findViewById(R.id.lvDados);
        txtStatus = findViewById(R.id.txtStatus);
    }

    public void btnCarregaDadosOnClick(View view) {
        String url = MainActivity.URL_RAIZ + "dados.txt";
        Request request = new Request.Builder()
            .url(url)
            .build();
        Call call = reqHttp.newCall(request);
        call.enqueue(new Callback() {
            @Override
            public void onFailure(@NotNull Call call, @NotNull IOException e) {
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
```



```

    android:gravity="end"
    android:textColor="#4700D6"
    app:layout_constraintEnd_toStartOf="@+id/guideline2"
    app:layout_constraintStart_toEndOf="@+id/txtIdImovel"
    app:layout_constraintTop_toTopOf="parent"
    tools:text="txtTipoImovel" />

<TextView
    android:id="@+id/txtBairroImovel"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="4dp"
    android:textColor="#4700D6"
    app:layout_constraintStart_toStartOf="@+id/guideline1"
    app:layout_constraintTop_toBottomOf="@+id/txtIdImovel"
    tools:text="txtBairroImovel" />

<TextView
    android:id="@+id/txtVlrVendaImovel"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="4dp"
    android:gravity="end"
    android:textColor="#4700D6"
    app:layout_constraintEnd_toStartOf="@+id/guideline2"
    app:layout_constraintStart_toEndOf="@+id/txtIdImovel"
    app:layout_constraintTop_toBottomOf="@+id/txtTipoImovel"
    tools:text="txtVlrVendaImovel" />

<TextView
    android:id="@+id/txtVlrAluguelImovel"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="4dp"
    android:gravity="end"
    android:textColor="#4700D6"
    app:layout_constraintEnd_toStartOf="@+id/guideline2"
    app:layout_constraintStart_toEndOf="@+id/txtIdImovel"
    app:layout_constraintTop_toBottomOf="@+id/txtVlrVendaImovel"
    tools:text="txtVlrAluguelImovel" />

<TextView
    android:id="@+id/txtComplemento"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:paddingBottom="8dp"
    android:gravity="center"
    android:text="txtComplemento"
    android:textColor="#4700D6"
    app:layout_constraintEnd_toStartOf="@+id/guideline2"
    app:layout_constraintStart_toStartOf="@+id/guideline1"
    app:layout_constraintTop_toBottomOf="@+id/txtVlrAluguelImovel"
    tools:text="txtComplemento" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

Classe: DadosAdapter.java

```

public class DadosAdapter extends BaseAdapter {

    private final List<String> dados;
    private final Activity atividade;

    public DadosAdapter(List<String> dados, Activity atividade) {
        this.dados = dados;
        this.atividade = atividade;
    }

    @Override
    public int getCount() {
        return dados.size();
    }

    @Override
    public Object getItem(int i) {
        return dados.get(i);
    }
}

```

```

}

@Override
public long getItemId(int i) {
    return i;
}

@Override
public View getView(int i, View view, ViewGroup viewGroup) {
    String s;
    double d;
    if (view == null) {
        LayoutInflater inflater = atividade.getLayoutInflater();
        view = inflater.inflate(R.layout.faixa_dado, viewGroup, false);
    }
    s = dados.get(i);
    List<String> imovel = Arrays.asList(s.split(";"));

    TextView txt;
    txt = view.findViewById(R.id.txtIdImovel);
    txt.setText(imovel.get(0));
    txt = view.findViewById(R.id.txtBairroImovel);
    txt.setText(imovel.get(1));
    txt = view.findViewById(R.id.txtTipoImovel);
    txt.setText(imovel.get(2));
    txt = view.findViewById(R.id.txtVlrVendaImovel);
    d = Double.parseDouble(imovel.get(8)); // valor venda
    if (d != 0)
        txt.setText(String.format(Locale.getDefault(), "Valor venda: R$ %, .2f", d));
    else
        txt.setText("não disponível para venda");
    txt = view.findViewById(R.id.txtVlrAluguelImovel);
    d = Double.parseDouble(imovel.get(9)); // valor aluguel
    if (d != 0)
        txt.setText(String.format(Locale.getDefault(), "Valor aluguel: R$ %, .2f", d));
    else
        txt.setText("não disponível para aluguel");
    txt = view.findViewById(R.id.txtComplemento);
    s = "Área terreno: " + imovel.get(3) + " e Área construída: " + imovel.get(4) + "\n";
    s = s + "Dormitórios: " + imovel.get(5) + " - Suítes: " + imovel.get(6) + " - Vagas: " + imovel.get(7);
    txt.setText(s);

    return view;
}
}

```

Notas Técnicas

Permissão para acessar a Internet e trafegar caracteres não criptografados

Este aplicativo precisa de permissão de uso de internet. Isto deve ser informado no arquivo AndroidManifest, através da cláusula de permissão conforme indicado a seguir.

Além disso, vamos acessar um endereço na internet através do protocolo HTTP, sem recurso de segurança. Por padrão o Android não permite esse tipo de acesso dos apps a endereços web. Porém, é possível alterar esse padrão e passar a permitir tal tipo de acesso. Essa alteração também é feita no AndroidManifest.xml e também está indicada a seguir. Durante a aula será demonstrado que quando o acesso é feito ao servidor através do protocolo HTTPS (S de Secure) essa liberação não é necessária.

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.ilp506.requisicoeshhttp">

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

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/Theme.RequisicoesHTTP"
        android:usesCleartextTraffic="true">
        <activity android:name=".ObtemDados" />
        <activity android:name=".ObtemPagina" />
        <activity android:name=".ObtemImagem" />
        <activity android:name=".ObtemArqTexto" />
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

Esta linha concede permissão de acesso à Internet

usesCleartextTraffic = "true" permite trafegar texto não criptografado

Declaração de dependências no Gradle

Neste aplicativo usaremos dois conjuntos de classes externas à plataforma Android, são elas:

- OkHttp3 – que será utilizada para executar as requisições ao servidor
- Picasso – que será utilizada para trabalhar com as imagens que serão carregadas do servidor

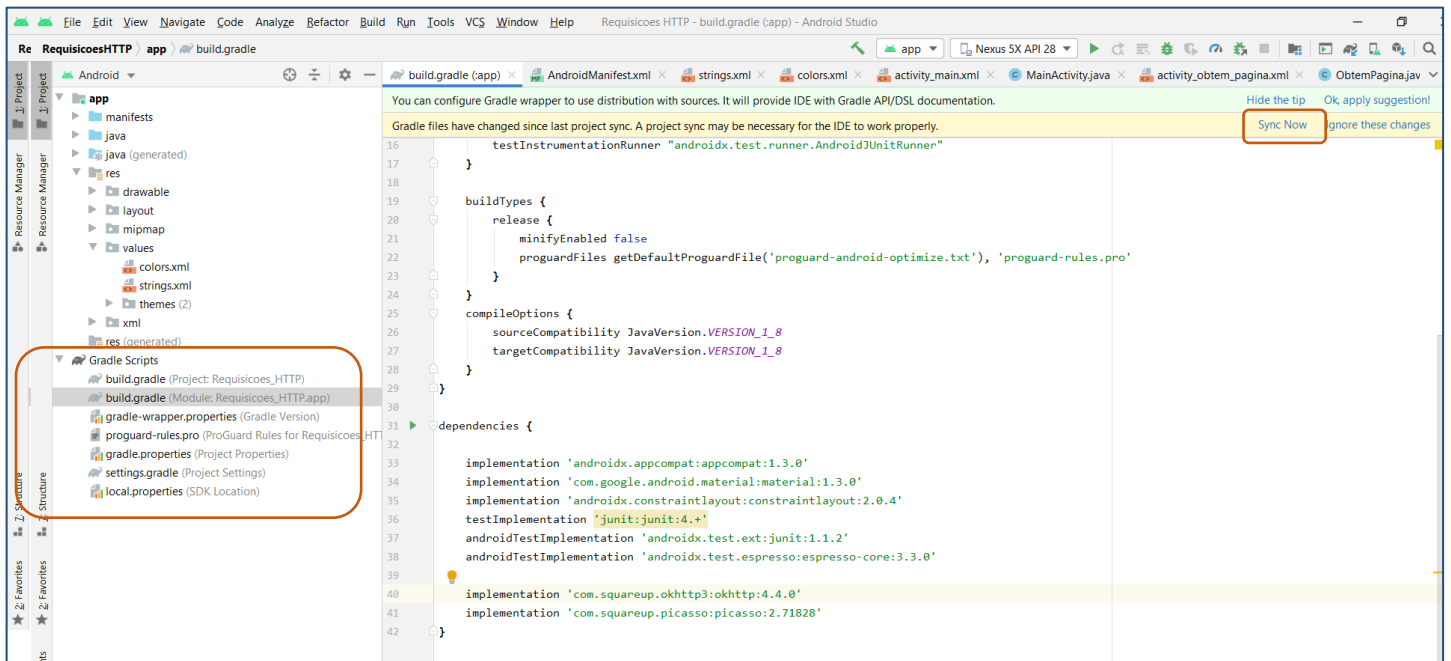
Ambas são projetos Open Source, modernas, simples de usar e seguras. Por esses motivos são amplamente usadas no mundo dos aplicativos mobile para Android.

Para possibilitar seu uso é necessário informar isso ao Android Studio. Essa informação será vinculada ao projeto (e não ao IDE) e deve ser especificada no arquivo build.gradle na seção *dependencies*, usando a diretiva *implementation*, conforme mostrado a seguir.

Para abrir o arquivo build.gradle abra o painel do projeto e expanda as opções sob o item Gradle Scripts. O segundo item da lista é o build.gradle do módulo, dê um duplo clique sobre o mesmo. O arquivo será aberto à direita. Localize a seção de dependências e acrescente estas duas linhas:

```
implementation 'com.squareup.okhttp3:okhttp:4.4.0'
implementation 'com.squareup.picasso:picasso:2.71828'
```

Após digitar as dependências clique em Sync Now que apareça no topo da tela e está destacado na imagem.



build.gradle do módulo

```

plugins {
    id 'com.android.application'
}

android {
    compileSdkVersion 30
    buildToolsVersion "30.0.2"

    defaultConfig {
        applicationId "com.ilp506.requisicoeshttp"
        minSdkVersion 22
        targetSdkVersion 30
        versionCode 1
        versionName "1.0"

        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
    }

    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
        }
    }
    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }
}

dependencies {

    implementation 'androidx.appcompat:appcompat:1.3.0'
    implementation 'com.google.android.material:material:1.3.0'
    implementation 'androidx.constraintlayout:constraintlayout:2.0.4'
    testImplementation 'junit:junit:4.+'
    androidTestImplementation 'androidx.test.ext:junit:1.1.2'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.3.0'

    implementation 'com.squareup.okhttp3:okhttp:4.4.0'
    implementation 'com.squareup.picasso:picasso:2.71828'
}

```

← Estas duas linhas especificam as dependências