|  |  |
| --- | --- |
|  | **UNIVERSIDAD DON BOSCO**  **FACULTAD DE ESTUDIOS TECNOLÓGICOS**  **COORDINACIÓN DE COMPUTACIÓN Y MÓVILES** |
| **Ciclo II** | **Desarrollo de Avanzado de Software para móviles** Guía de Laboratorio No. 1 Networking Android - Retrofit |

* **I Objetivos**
* Conocer Retrofit para realizar peticiones.
* Acceder a webservices y utilizar sus respectivos endpoints.
* **II Desarrollo**

**Para esta guía hará uso de la api 21 como api mínima.**

Agregar las dependencias:

Agregar las siguientes dependencias a su proyecto:

|  |
| --- |
| compile **'com.github.bumptech.glide:glide:3.6.1'** compile **'com.squareup.retrofit2:retrofit:2.3.0'** compile **'com.squareup.retrofit2:converter-gson:2.3.0'**  *//La version de design depende de su compile version* compile **'com.android.support:design:27.1.1'** |

Chiste.java

|  |
| --- |
| **package** sv.edu.udb.guia01;  **import** java.io.Serializable; **import** java.util.Date;   **public class** Chiste **implements** Serializable{  **private int id**;  **private int id\_autor**;  **private** String **nombre**;  **private** String **texto**;  **private** String **image**;  **private** Date **creation\_at**;  **private** Date **updated\_at**;    **public int** getId() {  **return id**;  }   **public void** setId(**int** id) {  **this**.**id** = id;  }   **public int** getId\_autor() {  **return id\_autor**;  }   **public void** setId\_autor(**int** id\_autor) {  **this**.**id\_autor** = id\_autor;  }   **public** String getNombre() {  **return nombre**;  }   **public void** setNombre(String nombre) {  **this**.**nombre** = nombre;  }   **public** String getTexto() {  **return texto**;  }   **public void** setTexto(String texto) {  **this**.**texto** = texto;  }   **public** String getImage() {  **return image**;  }   **public void** setImage(String image) {  **this**.**image** = image;  }   **public** Date getCreation\_at() {  **return creation\_at**;  }   **public void** setCreation\_at(Date creation\_at) {  **this**.**creation\_at** = creation\_at;  }   **public** Date getUpdated\_at() {  **return updated\_at**;  }   **public void** setUpdated\_at(Date updated\_at) {  **this**.**updated\_at** = updated\_at;  } } |

Autor.java

|  |
| --- |
| **package** sv.edu.udb.guia01;  **import** com.google.gson.annotations.SerializedName;  **import** java.util.Date; **import** java.util.List;  **public class** Autor {  **private int id**;  **private** String **nombre**;  **private** Date **creation\_at**;  **private** Date **updated\_at**;   @SerializedName(**"chistesautor"**)  **private** List<Chiste> **chisteList**;    **public int** getId() {  **return id**;  }   **public void** setId(**int** id) {  **this**.**id** = id;  }   **public** String getNombre() {  **return nombre**;  }   **public void** setNombre(String nombre) {  **this**.**nombre** = nombre;  }   **public** Date getCreation\_at() {  **return creation\_at**;  }   **public void** setCreation\_at(Date creation\_at) {  **this**.**creation\_at** = creation\_at;  }   **public** Date getUpdated\_at() {  **return updated\_at**;  }   **public void** setUpdated\_at(Date updated\_at) {  **this**.**updated\_at** = updated\_at;  }   **public** List<Chiste> getChisteList() {  **return chisteList**;  }   **public void** setChisteList(List<Chiste> chisteList) {  **this**.**chisteList** = chisteList;  } } |

Crear la siguiente clase dentro de un paquete denominado voice: SpeakRequest.java

|  |
| --- |
| **package** sv.edu.udb.guia01;  **import** android.content.Context; **import** android.speech.tts.TextToSpeech; **import** android.util.Log;  **import** java.util.ArrayList; **import** java.util.Locale;  **public class** SpeakRequest **implements** TextToSpeech.OnInitListener {   TextToSpeech **tts** ;  Locale **spanish** = **new** Locale(**"es"**, **"ES"**);  Locale **english** = **new** Locale(**"us"**, **"US"**);  Context **context**;   **public** SpeakRequest(Context context){  **this**.**context** = context;  **tts** = **new** TextToSpeech(context, **this**);  }   @Override  **public void** onInit(**int** status) {   **if**(status == TextToSpeech.***SUCCESS***){  **int** result=**tts**.setLanguage(**spanish**);  **if**(result==TextToSpeech.***LANG\_MISSING\_DATA*** ||  result==TextToSpeech.***LANG\_NOT\_SUPPORTED***){  Log.*e*(**"error"**, **"This Language is not supported"**);  }  }  **else** Log.*e*(**"error"**, **"Initilization Failed!"**);  }    **public void** speak(**final** String text) {  **if**(text==**null**||**""**.equals(text))  {  String errorText = **"No existe contenido asociado"**;  **tts**.speak(errorText, TextToSpeech.***QUEUE\_FLUSH***, **null**);  }**else** {   **while**(isSpeaking()){  *//Don't do anything* }   **new** Thread(**new** Runnable() {   @Override  **public void** run() {  **tts**.speak(text, TextToSpeech.***QUEUE\_ADD***, **null**);  }  }).start();  }  }   **public void** speak(**final** String [] lines){   String text=**""**;   **for** (String tmp : lines) {  text += tmp+**"."**;  }    speak(text);   }    **public void** speak(ArrayList<String> lines){  speak((String [])lines.toArray());  }   **public void** stopSpeak(){  **tts**.stop();  }   **public boolean** isSpeaking(){  **return this**.**tts**.isSpeaking();  }     **public void** onDestroy(){   **if**(**tts**!=**null**) {  **tts**.stop();  **tts**.shutdown();  }   }  } |

Crear la interface: EndPointInterface.java

|  |
| --- |
| **package** sv.edu.udb.guia01;  **import** java.util.List;  **import** retrofit2.Call; **import** retrofit2.http.GET; **import** retrofit2.http.Path;   **public interface** EndPointInterface {   String ***base\_url*** = **"http://34.211.243.185:8080/"**;   @GET(**"autores/"**)  Call<List<Autor>> getAutores();   @GET(**"autores/{id}"**)  Call<Autor> getAutor(@Path(**"id"**) String id);  } |

Editar activity\_main.xml

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**android.support.design.widget.CoordinatorLayout 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"  tools:context="sv.edu.udb.guia01.MainActivity"**>   <**ListView xmlns:android="http://schemas.android.com/apk/res/android"  android:id="@+id/listView"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:stretchMode="columnWidth"  android:gravity="center"** />   <**android.support.design.widget.FloatingActionButton  android:id="@+id/fab"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="bottom|end"  android:layout\_margin="10dp"  app:srcCompat="@android:drawable/ic\_input\_add"** />  </**android.support.design.widget.CoordinatorLayout**> |

Editar MainActivity.java

|  |
| --- |
| **package** sv.edu.udb.guia01;  **import** android.content.Intent; **import** android.support.design.widget.FloatingActionButton; **import** android.support.design.widget.Snackbar; **import** android.support.v7.app.AppCompatActivity; **import** android.os.Bundle; **import** android.util.Log; **import** android.view.View; **import** android.widget.AdapterView; **import** android.widget.ArrayAdapter; **import** android.widget.ListView;  **import** com.google.gson.Gson; **import** com.google.gson.GsonBuilder;  **import** java.util.ArrayList; **import** java.util.List;  **import** retrofit2.Call; **import** retrofit2.Callback; **import** retrofit2.Response; **import** retrofit2.Retrofit; **import** retrofit2.converter.gson.GsonConverterFactory;  **public class** MainActivity **extends** AppCompatActivity {    ListView **listView**;   List<String> **stringListAutores**=**new** ArrayList<>();  List<Autor> **listAutores**;   Gson **gson** = **new** GsonBuilder()  .setDateFormat(**"yyyy-MM-dd"**)  .create();    Retrofit **retrofit** = **new** Retrofit.Builder()  .baseUrl(**"http://34.211.243.185:8080/"**)  .addConverterFactory(GsonConverterFactory.*create*(**gson**))  .build();   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_main***);   **listView** = (ListView) findViewById(R.id.***listView***);   EndPointInterface apiService = **retrofit**.create(EndPointInterface.**class**);    Call<List<Autor>> call = apiService.getAutores();   call.enqueue(**new** Callback<List<Autor>>() {  @Override  **public void** onResponse(Call<List<Autor>> call, Response<List<Autor>> response) {  **listAutores** = response.body();  fillData(**listAutores**);  }   @Override  **public void** onFailure(Call<List<Autor>> call, Throwable t) {  Log.*d*(**"UDBLOG:Error"**,t.getMessage());  }  });    FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.***fab***);  fab.setOnClickListener(**new** View.OnClickListener() {  @Override  **public void** onClick(View view) {  Snackbar.*make*(view, **"Replace with your own action"**, Snackbar.***LENGTH\_LONG***)  .setAction(**"Action"**, **null**).show();  }  });  }   **private void** fillData(List<Autor> autores) {   **for**(Autor autor:autores){  **this**.**stringListAutores**.add(autor.getNombre());  }   ArrayAdapter<String> arrayAdapter = **new** ArrayAdapter<>(**this**,android.R.layout.***simple\_list\_item\_1***,**this**.**stringListAutores**);  **listView**.setAdapter(arrayAdapter);   **listView**.setOnItemClickListener(**new** AdapterView.OnItemClickListener() {  @Override  **public void** onItemClick(AdapterView<?> adapterView, View view, **int** i, **long** l) {  Intent intent = **new** Intent(MainActivity.**this**,GridListActivity.**class**);   intent.putExtra(**"idautor"**,String.*valueOf*(**listAutores**.get(i).getId()));   startActivity(intent);   }  });  }   } |

Editar grid\_item.xml

|  |
| --- |
| *<?***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="200dp"**>   <**ImageView  android:id="@+id/imagenchiste"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:scaleType="centerCrop"** />   <**TextView  android:id="@+id/titulochiste"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_gravity="bottom"  android:background="#efdef5"  android:maxLines="1"  android:padding="16dp"  android:textAppearance="@style/TextAppearance.AppCompat.Subhead"  android:textColor="@android:color/white"** />  </**FrameLayout**> |

Crear ChistesAdapter.java

|  |
| --- |
| **package** sv.edu.udb.guia01;  **import** android.content.Context; **import** android.view.LayoutInflater; **import** android.view.View; **import** android.view.ViewGroup; **import** android.widget.BaseAdapter; **import** android.widget.ImageView; **import** android.widget.TextView;  **import** com.bumptech.glide.Glide;  **import** java.util.List;    **public class** ChistesAdapter **extends** BaseAdapter {  **private** Context **context**;   List<Chiste> **chisteList**;   **private static** String *IMAGE\_PREFIX* = **"http://34.211.243.185:8080/images/"**;   **public** ChistesAdapter(Context context, List<Chiste> chisteList){  **this**.**context** = context;  **this**.**chisteList** = chisteList;  }   @Override  **public int** getCount() {  **return this**.**chisteList**.size();  }   @Override  **public** Object getItem(**int** i) {  **return chisteList**.get(i);  }   @Override  **public long** getItemId(**int** i) {  **return chisteList**.get(i).getId();  }   @Override  **public** View getView(**int** i, View view, ViewGroup viewGroup) {  **if** (view == **null**) {  LayoutInflater inflater = (LayoutInflater) **context** .getSystemService(Context.***LAYOUT\_INFLATER\_SERVICE***);  view = inflater.inflate(R.layout.***grid\_item***, viewGroup, **false**);  }   ImageView imageView = (ImageView) view.findViewById(R.id.***imagenchiste***);  TextView textView = (TextView) view.findViewById(R.id.***titulochiste***);  **final** Chiste chiste = **chisteList**.get(i);  Glide.*with*(imageView.getContext()).load(*IMAGE\_PREFIX*+chiste.getImage()).placeholder(R.mipmap.***ic\_launcher***).into(imageView);  textView.setText(chiste.getNombre());  **return** view;   } } |

**Crear la Actividad GridListActivity**

**Crear la Activitidad DetailActivity**

Editar activity\_grid\_list.xml

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**RelativeLayout 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"  tools:context="sv.edu.udb.guia01.GridListActivity"**>   <**GridView xmlns:android="http://schemas.android.com/apk/res/android"  android:id="@+id/gridview"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:columnWidth="90dp"  android:numColumns="auto\_fit"  android:verticalSpacing="10dp"  android:horizontalSpacing="10dp"  android:stretchMode="columnWidth"  android:gravity="center"** />  </**RelativeLayout**> |

Editar GridListActivity.java

|  |
| --- |
| **package** sv.edu.udb.guia01;  **import** android.content.Intent; **import** android.os.Build; **import** android.support.v4.app.ActivityCompat; **import** android.support.v4.app.ActivityOptionsCompat; **import** android.support.v4.util.Pair; **import** android.support.v7.app.AppCompatActivity; **import** android.os.Bundle; **import** android.util.Log; **import** android.view.View; **import** android.widget.AdapterView; **import** android.widget.GridView;  **import** com.google.gson.Gson; **import** com.google.gson.GsonBuilder;  **import** retrofit2.Call; **import** retrofit2.Callback; **import** retrofit2.Response; **import** retrofit2.Retrofit; **import** retrofit2.converter.gson.GsonConverterFactory;  **public class** GridListActivity **extends** AppCompatActivity **implements** AdapterView.OnItemClickListener{   **private** GridView **gridView**;  **private** ChistesAdapter **adaptadorChistes**;   **private** Autor **autor**;    Gson **gson** = **new** GsonBuilder()  .setDateFormat(**"yyyy-MM-dd"**)  .create();     Retrofit **retrofit** = **new** Retrofit.Builder()  .baseUrl(EndPointInterface.***base\_url***)*//La interface tiene el base url  //.addCallAdapterFactory(RxJava2CallAdapterFactory.create())* .addConverterFactory(GsonConverterFactory.*create*(**gson**))  .build();   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_grid\_list***);    **gridView** = (GridView) findViewById(R.id.***gridview***);   EndPointInterface apiService = **retrofit**.create(EndPointInterface.**class**);   String idautor= getIntent().getStringExtra(**"idautor"**);   Call<Autor> call = apiService.getAutor(idautor);   call.enqueue(**new** Callback<Autor>() {  @Override  **public void** onResponse(Call<Autor> call, Response<Autor> response) {  Autor autor = response.body();   fillData(autor);  }     @Override  **public void** onFailure(Call<Autor> call, Throwable t) {  Log.*d*(**"UDBLOG:Error"**,t.getMessage());  }  });     }   **private void** fillData(Autor autor) {  **this**.**autor** = autor;  **adaptadorChistes**= **new** ChistesAdapter(GridListActivity.**this**,autor.getChisteList());  **this**.**gridView**.setAdapter(**adaptadorChistes**);  **this**.**gridView**.setOnItemClickListener(**this**);  }    @Override  **public void** onItemClick(AdapterView<?> adapterView, View view, **int** i, **long** l) {  Chiste chiste = **this**.**autor**.getChisteList().get(i);  Intent intent = **new** Intent(**this**, DetailActivity.**class**);  intent.putExtra(DetailActivity.***EXTRA\_PARAM\_ID***, chiste);   **if** (Build.VERSION.***SDK\_INT*** >= Build.VERSION\_CODES.***LOLLIPOP***) {   ActivityOptionsCompat activityOptions =  ActivityOptionsCompat.*makeSceneTransitionAnimation*(  **this**,  **new** Pair<View, String>(view.findViewById(R.id.***imagenchiste***),  DetailActivity.***VIEW\_NAME\_HEADER\_IMAGE***)  );   ActivityCompat.*startActivity*(**this**, intent, activityOptions.toBundle());  } **else** startActivity(intent);  }   } |

Editar activity\_detail.xml

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**LinearLayout 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:orientation="vertical"**>   <**ImageView xmlns:android="http://schemas.android.com/apk/res/android"  android:id="@+id/imagen\_extendida"  android:layout\_width="match\_parent"  android:layout\_height="0px"  android:layout\_weight="10"** >   </**ImageView**>   <**Button  android:layout\_width="match\_parent"  android:layout\_height="0px"  android:layout\_weight="1"  android:text="Hablar"  android:onClick="hablar"** />  </**LinearLayout**> |

DetailActivity.java

|  |
| --- |
| **package** sv.edu.udb.guia01;  **import** android.support.v7.app.AppCompatActivity; **import** android.os.Bundle; **import** android.view.View; **import** android.widget.ImageView;  **import** com.bumptech.glide.Glide;  **import** sv.edu.udb.guia01.SpeakRequest;  **public class** DetailActivity **extends** AppCompatActivity {   **public static final** String ***EXTRA\_PARAM\_ID*** = **"sv.edu.udb.id"**;  **public static final** String ***VIEW\_NAME\_HEADER\_IMAGE*** = **"imagen\_compartida"**;  **private static** String *IMAGE\_PREFIX* = **"http://34.211.243.185:8080/images/"**;    Chiste **chiste**;  ImageView **imagenExtendida**;  SpeakRequest **speakRequest**;   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_detail***);   **chiste** = (Chiste) getIntent().getSerializableExtra(***EXTRA\_PARAM\_ID***);   **imagenExtendida** = (ImageView) findViewById(R.id.***imagen\_extendida***);   cargarContenido();   **if**(**speakRequest** == **null**) {  **speakRequest** = **new** SpeakRequest(**this**);  }   }    **private void** cargarContenido() {  Glide.*with*(**imagenExtendida**.getContext())  .load(*IMAGE\_PREFIX*+**chiste**.getImage())  .placeholder(R.mipmap.***ic\_launcher***)  .into(**imagenExtendida**);  }   @Override  **protected void** onPause() {  **speakRequest**.onDestroy();  **super**.onPause();  }   **public void** hablar(View view) {  **if**(**speakRequest**.isSpeaking()){  **speakRequest**.stopSpeak();  }  **speakRequest**.speak(**chiste**.getTexto());   } } |

Agregue permisos de internet al Proyecto:

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

Procure incorporar las actividades necesarias

|  |
| --- |
| <**activity android:name=".MainActivity"**>  <**intent-filter**>  <**action android:name="android.intent.action.MAIN"** />   <**category android:name="android.intent.category.LAUNCHER"** />  </**intent-filter**> </**activity**> <**activity android:name=".GridListActivity"**></**activity**> <**activity android:name=".DetailActivity"** /> |

* **III Ejercicios Complementarios**

Cambiar la API mínima del proyecto, y corrija el proyecto para que sea capaz de ejecutarse sin las transiciones de material design.

Obtenga un webservice y acceda a sus endpoints utilizando retrofit.