


Open · Opened 3 days ago by  **Rubén Montero**

Una petición de una lista

Resumen

- Lanzaremos una petición que solicita un *array* JSON
- Mostraremos un *SnackBar* si es exitosa

Descripción

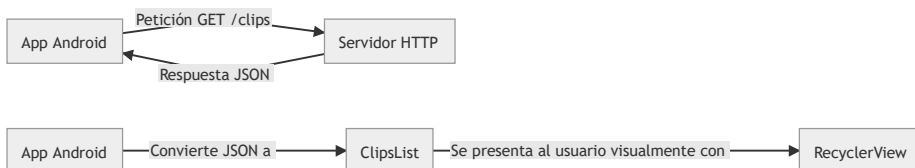
Nuestra aplicación lanza una petición de red que obtiene una respuesta sencilla.

```
{
  "status": "Server is healthy!"
}
```

Ahora vamos a lanzar una petición que obtendrá datos más elaborados. Un *array*:

```
[
  {
    "id": 1,
    "title": "Automatic Dialer (Part II)",
    "videoUrl": "https://raw.githubusercontent.com/rubenmv0/fp/main/simpsons/automatic_dialer_part_ii.mp4"
  },
  {
    "id": 2,
    "title": "Desparasitándose",
    "videoUrl": "https://raw.githubusercontent.com/rubenmv0/fp/main/simpsons/desparasitandome.mp4"
  },
  {
    "id": 3,
    "title": "Glasses",
    "videoUrl": "https://raw.githubusercontent.com/rubenmv0/fp/main/simpsons/glasses.mp4"
  },
  {
    "id": 4,
    "title": "Marcador Automático",
    "videoUrl": "https://raw.githubusercontent.com/rubenmv0/fp/main/simpsons/marcador_automatiko.mp4"
  },
  {
    "id": 5,
    "title": "Tarta",
    "videoUrl": "https://raw.githubusercontent.com/rubenmv0/fp/main/simpsons/tarta.mp4"
  }
]
```

El objetivo final es *presentar* estos datos al usuario.



En esta tarea, sólo vamos a preocuparnos de mandar la petición HTTP.

La tarea

Para empezar, en `MainActivity.java`, **crea** un método privado (dentro de la clase):

```
private void requestCliplist() {
```

```
}

```

...y luego, **invócalo** desde `onCreate`, como se indica a continuación:

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    this.queue = Volley.newRequestQueue(context);

    // Solicitamos una lista de clips
    requestClipList();

    // Aquí lanzaremos una petición HTTP
    // ...

```

(De esta manera, `requestClipList` encapsulará la lógica para enviar la nueva petición HTTP y no haremos crecer excesivamente nuestro `onCreate`).

A continuación, dentro de `requestClipList`, **instancia** un nuevo objeto `JSONArrayRequest`. Los parámetros enviados al método constructor serán:

- `Request.Method.GET`
- `Server.name + "/clips"`
- `null`
- `new Response.Listener{...}`
- `new Response.ErrorListener{...}`

Luego, **añade** dicha petición a la cola de red con `this.queue.add(...);`

SnackBar

Para terminar la tarea, vamos a mostrar un `SnackBar` si la petición a `/clips` se ha completado correctamente. Es un componente *similar* al `Toast`, con estas diferencias:

- Puede permitir al usuario *clickar* para llevar a cabo alguna acción.
- Cuando lo construimos, debemos especificar una `View`. Se pintará *debajo* de dicha `View`.

Para mostrar un `SnackBar`, como primer paso, **añade** `id` al `ConstraintLayout` en `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:id="@+id/main_layout"
    tools:context=".MainActivity">

```

...y luego, desde `MainActivity.java`, **añade** un atributo privado (`ConstraintLayout`) e **inicialízalo** desde `onCreate` empleando `findViewById`. Quedará así:

```
public class MainActivity extends AppCompatActivity {
    private RequestQueue queue;
    private Context context = this;
    private ConstraintLayout mainLayout; // Añadimos un nuevo atributo

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        this.queue = Volley.newRequestQueue(context);
        this.mainLayout = findViewById(R.id.main_layout); // Lo asignamos con findViewById

        // ...

```

(En general es buena práctica almacenar los componentes como atributos privados e inicializarlos sólo una vez en `onCreate`. Esto se debe a que `findViewById` se considera una operación costosa).

Finalmente, otra vez en `MainActivity.java`, **añade** estas dos líneas a tu nuevo `onResponse` (dentro de `requestClipList`):

```
Snackbar.make(mainLayout, "Clips obtained", Snackbar.LENGTH_SHORT).show();
```

¡Listo! Si pruebas la aplicación, deberías ver un `SnackBar` con el mensaje `"Clips obtained"` que demuestra que la petición a <http://raspi:8000/clips> se ha completado con éxito.

Por último

Sube tus cambios al repositorio en un nuevo *commit*.



[Rubén Montero @ruben.montero](#) changed milestone to [%Sprint 2](#) 3 days ago