

PROYECTO PETAGRAM RCE – ASIGNATURA PRIMERA SEMANA DEL CURSO 4 DE LA ESPECIALIZACION DE DESARROLLO DE APLICACIONES MOVILES PARA ANDROID DE COURSERA Y LA UNAM.

Para este proyecto se utilizaron 2 Endpoints del API de Instagram, el de buscar usuarios por su nombre y el de obtener la media reciente de un usuario por su identificador.

GET **/users/search**

https://api.instagram.com/v1/users/search?q=jack&access_token=ACCESS-TOKEN

RESPONSE ▾

Get a list of users matching the query.

REQUIREMENTS

Scope: public_content

PARAMETERS

ACCESS_TOKEN	A valid access token.
Q	A query string.
COUNT	Number of users to return.

GET **/users/{user-id}/media/recent**

https://api.instagram.com/v1/users/{user-id}/media/recent/?access_token=ACCESS-TOKEN

RESPONSE ▾

Get the most recent media published by a user.
The public_content scope is required if the user is not the owner of the access_token.

REQUIREMENTS

Scope: public_content

PARAMETERS

ACCESS_TOKEN	A valid access token.
MAX_ID	Return media earlier than this max_id.
MIN_ID	Return media later than this min_id.
COUNT	Count of media to return.

Aunque la clase RestApi contiene más definiciones de GET solamente se utilizaron los siguientes:

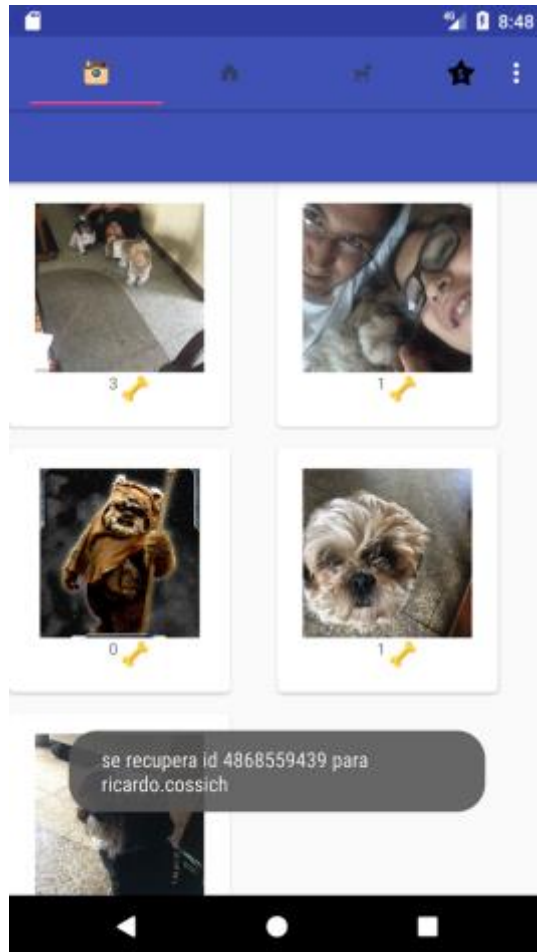
```
@GET(ConstantsRestApi.URL_GET_USERS_QUERY)  
Call<InformacionUsuarioResponse> getUsuariosBusqueda(@Query("q") String usuario_a_buscar);
```

```
@GET(ConstantsRestApi.URL_GET_RECENT_MEDIA_USER2)
Call<MascotaResponse> getRecentMediaUserId(@Path("userid") String usuario_id);
```

Para cada uno de estos se definieron deserializadores, InformacionUsuarioDeserializador y MascotaDeserializador respectivamente.

A continuación, algunos pantallazos para mostrar la funcionalidad de esta asignación:

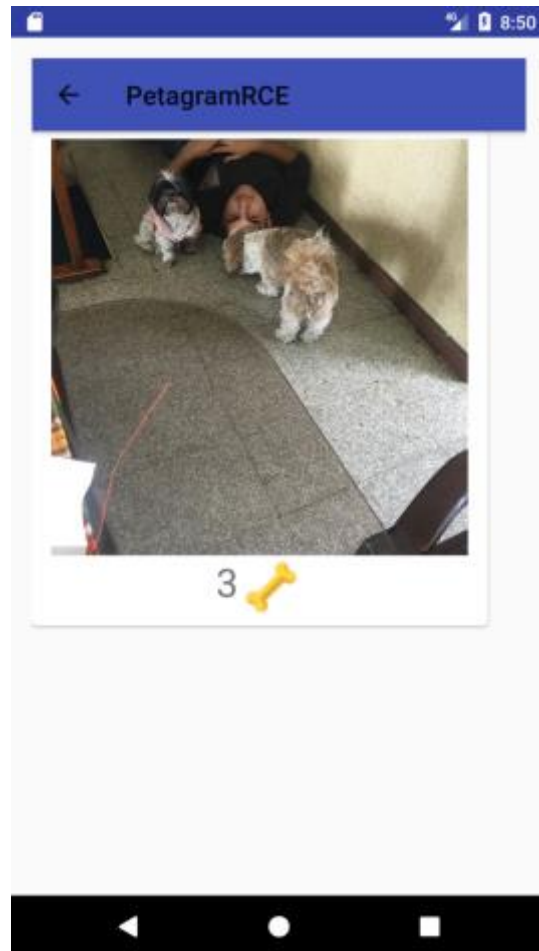
PRIMERO, Vale la pena mencionar que se creó un nuevo Fragmento en la actividad principal para no “perder” el tema de acceso a bases de datos del proyecto anterior. Para este nuevo fragmento “FragmentMascota” se creó un modelo de datos basado en la clase “MascotaInstagram” y en los resultados de objetos JSON “MascotaResponse” e “InformacionUsuarioResponse”. En la parte de presentador y vista se tiene la clase “FragmentMascotaPresenter” que es la que llama a los API de Instagram y luego al adaptador del fragment “fragment_mascota”, llamado “MascotaAdaptador4” que llena el RecyclerView adecuado.



Como se observa en el pantallazo, hay un fragmento con el ícono de Instagram (el de la izquierda), que muestra las fotos recuperadas para mi usuario (Ricardo.cossich), pero lo

hace a través de mi identificador (que se ve en el mensaje Toast). Para recuperar estas imágenes se hizo uso de los dos EndPoints.

Cuando se da un clic a una imagen nos lleva a la pantalla (o actividad de detalle), pantallazo a continuación:



También se pidió que la aplicación tenga una simulación de TimeLine de media reciente de los usuarios que aprobaron el SandBox de Instagram. En mi caso implementé una pantalla donde se muestra la media reciente de un usuario específico. Esta pantalla es accesible desde el menú de la aplicación en la opción “Visualizar TimeLine”, a continuación pantallazo de esta actividad.



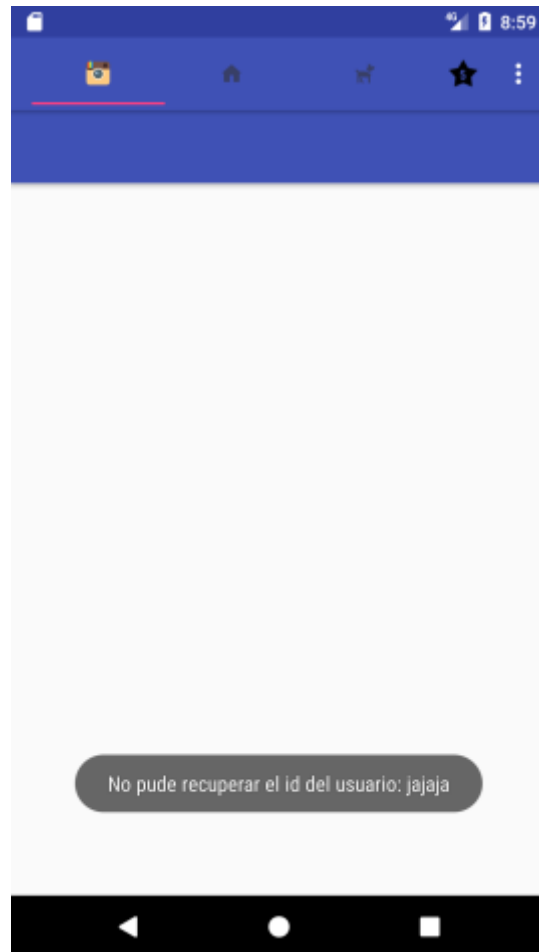
También se solicitó que la aplicación tenga una opción de menú donde se pueda definir la cuenta de Instagram de la que se debe recuperar la media a presentar en la pantalla o actividad principal (fragmento adicional agregado).

Para este efecto lo que se hace es que se llama una nueva actividad donde se pide el nombre de la cuenta y al presionar el botón se vuelve a refrescar la actividad principal pasando como parámetro el texto introducido en el campo de cuenta.

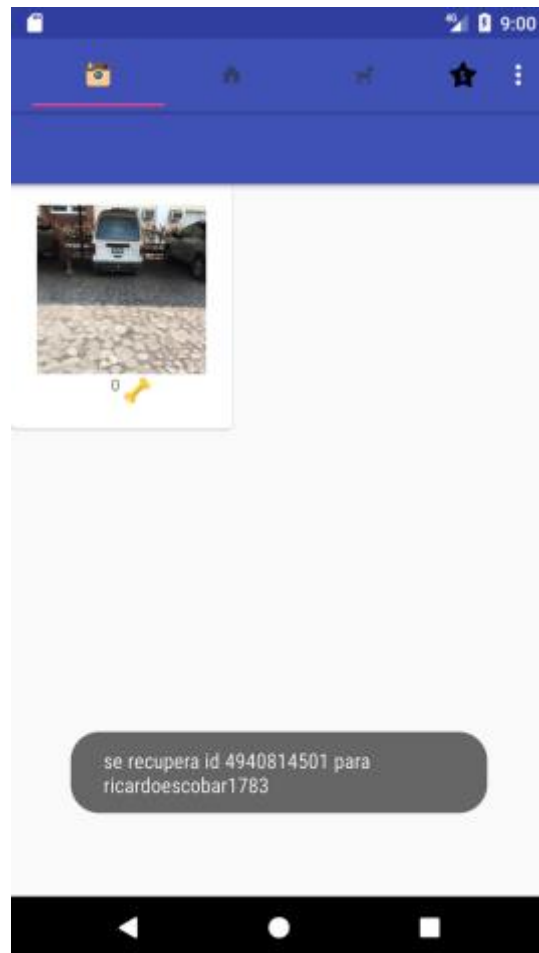
A continuación, pantallazo de la pantalla done se ingresa la cuenta a utilizar de Instagram.



Si se ingresa una cuenta no existente o a la cual no se tiene acceso, aparecerá la pantalla de actividad principal mostrando un mensaje similar al que se muestra en el siguiente pantallazo.



Caso contrario si la cuenta existe y se tiene acceso a la misma, entonces se llama a la actividad principal pasando como parámetro la cuenta, y en el presentador del fragmento se obtiene de primero el identificador del usuario y luego con este identificador se obtiene la media reciente del usuario. A continuación, un pantallazo mostrando el resultado exitoso de una cuenta existente y con permisos para accederla con el TOKEN de mi cuenta.



Se muestra un mensaje donde se despliega la cuenta y el identificador del usuario.

Acá finaliza el documento PDF solicitado para esta asignación.