

En la interfaz EndpointsApi se han utilizado dos Endpoints.

El primero de ellos, que se ha llamado getUserID, devuelve datos de un usuario a partir de su nombre. En concreto, nos devolverá su nombre de usuario, su foto de perfil, y lo más importante, su id. Para configurar la url de este Endpoint se han usado dos "@Query". Estos han servido para dar el formato "clave = cosa a buscar", que en este caso particular es "q = nombre_de_usuario_que_estemos_buscando", "access_token = nuestro_token". Además, si se colocan varios "@Query", automáticamente se coloca un "&" entre ellos, tal y como se requiere en la url. La url definitiva quedaría:
"https://api.instagram.com/v1/users/search?q=jack&access_token=ACCESS-TOKEN"
Donde lo marcado con color azul es lo especificado en los "@Query", y lo marcado en color rojo es lo generado de forma automática.

El segundo Endpoint, que se ha llamado getRecentMedia, se genera de forma similar. En este caso se ha usado "@Path" en lugar de "@Query", ya que lo que se desea es sustituir un valor. El valor a sustituir es el id del usuario, que se obtiene a partir del Endpoint anterior. Haciendo uso de "@Path" se sustituye en la url final el bloque "{id}" por la id especificada en la variable "userid". La cadena final sería:
https://api.instagram.com/v1/users/{user-id}/media/recent/?access_token=ACCESS-TOKEN
donde lo marcado en color azul es sustituido por la id del usuario de acuerdo con la variable "userid".

```
public interface EndpointsApi {

    String urlusuario = "users/search";

    @GET(urlusuario)

    Call<UsuarioResponse> getUserID(
        @Query("q") String userid,
        @Query("access_token") String token
    );

    String urlmedia = "users/{id}/media/recent/?access_token=" +
        ConstantsRestApi.ACCESS_TOKEN;

    @GET(urlmedia)

    Call<MascotaResponse> getRecentMedia(@Path("id") String userid);

}
```