

Tarea 4 - Creando Soporte Android Wear

Para esta tarea se implementó el endpoint de relaciones de Instagram.

```
@GET(ConstantsRestAPI.URL_RELATIONSHIP)
Call<UsuarioFollowResponse> getRelationship(@Path("user-id") String idUser);

@FormUrlEncoded
@POST(ConstantsRestAPI.URL_FOLLOW_USER)
Call<UsuarioFollowResponse> modifyFollowUser(@Path("user-id") String idUser, @Field("action") String action);
```

```
public Gson construyeGsonDeserializadorGetRelationship(){
    GsonBuilder gsonBuilder = new GsonBuilder();
    gsonBuilder.registerTypeAdapter(UsuarioFollowResponse.class, new UsuarioFollowDeserializador());
    return gsonBuilder.create();
}
```

```
public class UsuarioFollowResponse {

    private String codigoRespuesta;
    private String outgoing_status;
    private String target_user_is_private;
    private String incoming_status;

    public UsuarioFollowResponse() {
    }

    public UsuarioFollowResponse(String codigoRespuesta, String incoming_status, String outgoing_status, String target_user_is_private) {
        this.codigoRespuesta = codigoRespuesta;
        this.incoming_status = incoming_status;
        this.outgoing_status = outgoing_status;
        this.target_user_is_private = target_user_is_private;
    }

    public String getCodigoRespuesta() { return codigoRespuesta; }

    public void setCodigoRespuesta(String codigoRespuesta) {
        this.codigoRespuesta = codigoRespuesta;
    }

    public String getIncoming_status() { return incoming_status; }
```

```

public class UsuarioFollowDeserializador implements JsonSerializer<UsuarioFollowResponse> {
    @Override
    public UsuarioFollowResponse deserialize(JsonElement json, Type typeOfT, JsonDeserializationContext context)
        throws JsonParseException, JsonMappingException, IOException {
        Gson gson = new Gson();
        UsuarioFollowResponse usuarioFollowResponse = gson.fromJson(json, UsuarioFollowResponse.class);
        JsonObject usuarioFollowResponseData = json.getAsJsonObject().getAsJsonObject(JsonKeys.LIKE_META);
        String sCode = usuarioFollowResponseData.get(JsonKeys.LIKE_CODE).getAsString();

        JsonObject usuarioFollowData = json.getAsJsonObject().getAsJsonObject(JsonKeys.MEDIA_RESPONSE_ARRAY);
        String sOut = usuarioFollowData.get(JsonKeys.FOLLOW_OUTGOING_STATUS).getAsString();
        String sIn = usuarioFollowData.get(JsonKeys.FOLLOW_INCOMING_STATUS).getAsString();
        String sUserPrivate = usuarioFollowData.get(JsonKeys.FOLLOW_TARGET_USER_PRIVATE).getAsString();

        usuarioFollowResponse.setCodigoRespuesta(sCode);
        usuarioFollowResponse.setOutgoing_status(sOut);
        usuarioFollowResponse.setTarget_user_is_private(sUserPrivate);
        usuarioFollowResponse.setIncoming_status(sIn);

        return usuarioFollowResponse;
    }
}

```

```

private void darFollowUnfollow(String action, final Context context){
    RestAPIAdapter restApiAdapter = new RestAPIAdapter();
    Gson gsonUser = restApiAdapter.construyeGsonDeserializadorGetRelationship();
    EndpointsAPI endpointsAPI = restApiAdapter.establecerConexionRestAPIInstagram(gsonUser);
    Call<UsuarioFollowResponse> usuarioFollowResponseCall = endpointsAPI.modifyFollowUser(metodos.mostrarUsuarioLike());

    usuarioFollowResponseCall.enqueue(new Callback<UsuarioFollowResponse>() {
        @Override
        public void onResponse(Call<UsuarioFollowResponse> call, Response<UsuarioFollowResponse> response) {
            UsuarioFollowResponse usuarioFollowResponse = response.body();
            if(usuarioFollowResponse.getCodigoRespuesta().equals("200")){
                Log.d("ACTION_STATUS", usuarioFollowResponse.getOutgoing_status());
                if(usuarioFollowResponse.getOutgoing_status().equals("follows")){
                    Toast.makeText(context, "Has empezado a seguir a " + metodos.mostrarUsuarioLike(), Toast.LENGTH_SHORT).show();
                }
            } else {
                if(usuarioFollowResponse.getOutgoing_status().equals("requested")){
                    Toast.makeText(context, "Has solicitado seguir a " + metodos.mostrarUsuarioLike(), Toast.LENGTH_SHORT).show();
                } else {
                    Toast.makeText(context, "Has dejado de seguir a " + metodos.mostrarUsuarioLike(), Toast.LENGTH_SHORT).show();
                }
            }
        }
    });
}

@Override

```

Se implementó la notificación para Android Wear, así como las acciones requeridas.

```
public void onReceive(Context context, Intent intent) {
    tipoAccion = intent.getAction();
    metodos = new Metodos(context);

    if (Constantes.KEY_MY_PROFILE.equals(tipoAccion)) {
        Intent intentPerfil = new Intent(context, MascotasLista.class);
        intentPerfil.putExtra("tab", 2);
        intentPerfil.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        context.startActivity(intentPerfil);
        Log.d("BROADCAST_RECEIVER", "Se ejecutó la acción 'Mi Perfil'");
    } else {
        if (Constantes.KEY_FOLLOWS.equals(tipoAccion)) {
            revisarRelacion(context);
            Log.d("BROADCAST_RECEIVER", "Se ejecutó la acción 'Follow / Unfollow'");
        } else {
            if (Constantes.KEY_USER_PROFILE.equals(tipoAccion)) {
                Intent intentUserPerfil = new Intent(context, UsuarioLike.class);
                intentUserPerfil.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
                context.startActivity(intentUserPerfil);

                Log.d("BROADCAST_RECEIVER", "Se ejecutó la acción 'Ver Usuario'");
            }
        }
    }
}

NotificationManager notificationManager = (NotificationManager) context.getSystemService(Context.NOTIFICATION_SERVICE);
notificationManager.cancel(NOTIFY_ID);
```

```
private void lanzarNotificacion(RemoteMessage remoteMessage) {
    Intent intMyProfile = new Intent();
    intMyProfile.setAction("MY_PROFILE");

    Intent intFollows = new Intent();
    intFollows.setAction("FOLLOWS");

    Intent intProfile = new Intent();
    intProfile.setAction("USER_PROFILE");

    PendingIntent pendingIntentMyProfile = PendingIntent.getBroadcast(this, 0, intMyProfile, PendingIntent.FLAG_UPDATE_CURRENT);
    PendingIntent pendingIntentFollows = PendingIntent.getBroadcast(this, 0, intFollows, PendingIntent.FLAG_UPDATE_CURRENT);
    PendingIntent pendingIntentProfile = PendingIntent.getBroadcast(this, 0, intProfile, PendingIntent.FLAG_UPDATE_CURRENT);

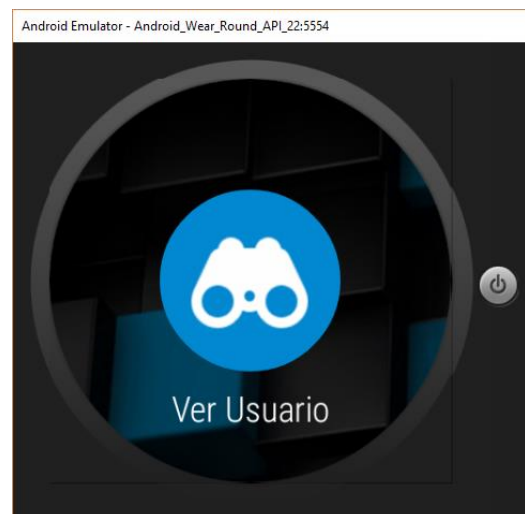
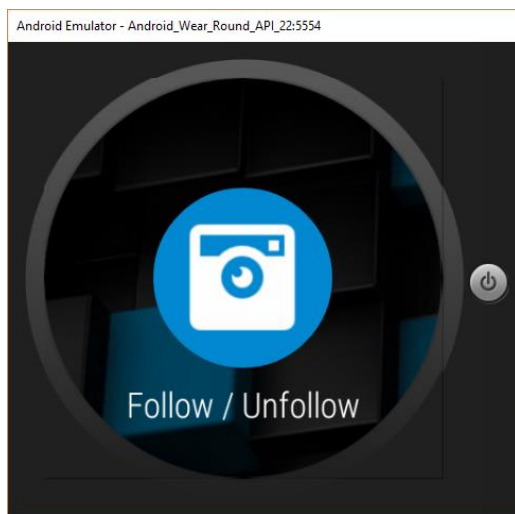
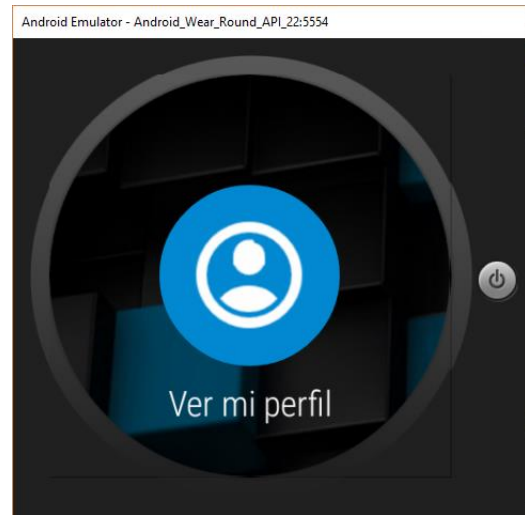
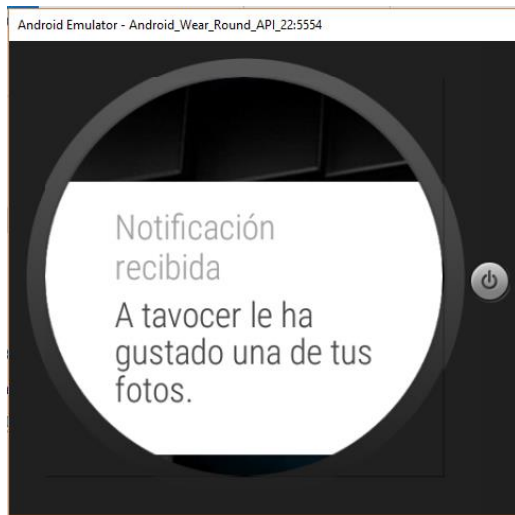
    Uri sonido = RingtoneManager.getDefaultUri(RingtoneManager.TYPE_NOTIFICATION);

    NotificationCompat.Action actionMiPerfil =
        new NotificationCompat.Action.Builder(R.drawable.my_profile_wear, "Ver mi perfil", pendingIntentMyProfile).build();

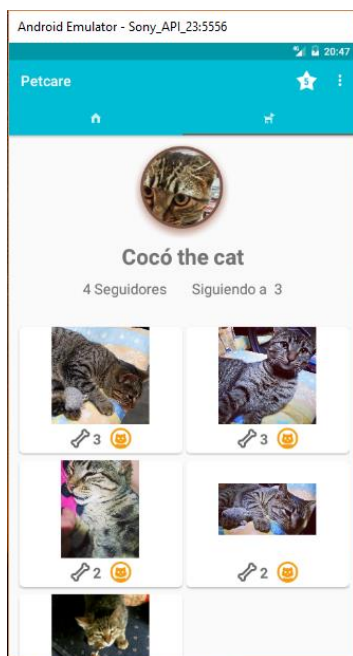
    NotificationCompat.Action actionFollow =
        new NotificationCompat.Action.Builder(R.drawable.follow_wear, "Follow / Unfollow", pendingIntentFollows).build();

    NotificationCompat.Action actionPerfil =
        new NotificationCompat.Action.Builder(R.drawable.profile_wear, "Ver Usuario", pendingIntentProfile).build();

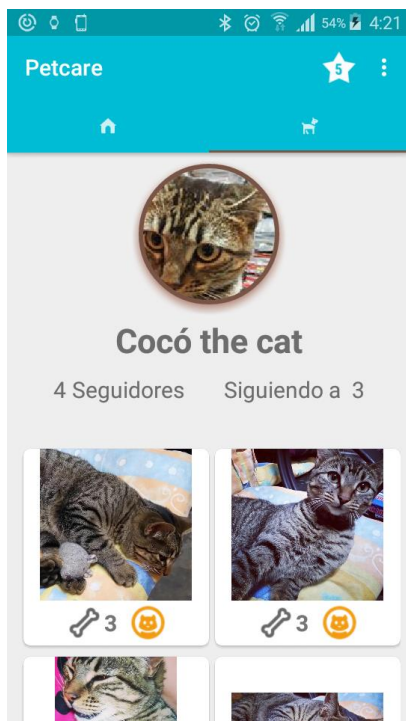
    WearableExtender wearableExtender =
        new WearableExtender();
```



En este ejemplo, se utiliza un emulador con la cuenta tavocer, que está visualizando la cuenta cococatmx:



Se utiliza un celular con la cuenta cococatmx configurada:



Al enviar una notificación, en el emulador del reloj se muestran tres acciones, la primera 'Ver mi perfil', mostrará en el celular la cuenta de cococatmx, que es la que está configurada el celular.

La acción Follow / Unfollow, permitirá poder seguir o dejar de seguir de la cuenta cococatmx a la cuenta tavocer:

Se muestra que se da follow a la cuenta tavocer, la cual ya cuenta con 60 seguidores.

```
antes.mypetcare D/MyFirebaseMsgService: From: 798262235485
antes.mypetcare D/ID_USER_LIKE: 34636213
antes.mypetcare D/USER_LIKE: tavocer
antes.mypetcare D/MyFirebaseMsgService: Message data payload: {iduser=34636213, suser=tavocer}
antes.mypetcare D/MyFirebaseMsgService: Message Notification Body: A tavocer le ha gustado una de tus fotos.
antes.mypetcare D/BROADCAST_RECEIVER: Se ejecutó la acción 'Follow / Unfollow'
antes.mypetcare D/RELATION_STATUS: Follow
antes.mypetcare D/ACTION_STATUS: follows
```



Cuando se da unfollow, la cuenta de tavocer ahora cuenta con 59 seguidores:

```
vantes.mypetcare I/System.out: (HTTPLog)-Static: isSBSettingEnabled false
vantes.mypetcare I/System.out: (HTTPLog)-Static: isSBSettingEnabled false
vantes.mypetcare I/System.out: (HTTPLog)-Static: isSBSettingEnabled false
vantes.mypetcare I/System.out: (HTTPLog)-Static: isSBSettingEnabled false
vantes.mypetcare D/ViewRootImpl: ViewPostImeInputStage ACTION_DOWN
vantes.mypetcare D/MyFirebaseMsgService: From: 798262235485
vantes.mypetcare D/ID_USER_LIKE: 34636213
vantes.mypetcare D/USER_LIKE: tavocer
vantes.mypetcare D/MyFirebaseMsgService: Message data payload: {iduser=34636213, suser=tavocer}
vantes.mypetcare D/MyFirebaseMsgService: Message Notification Body: A tavocer le ha gustado una de tus fotos.
vantes.mypetcare D/BROADCAST_RECEIVER: Se ejecutó la acción 'Follow / Unfollow'
vantes.mypetcare D/RELATION_STATUS: Unfollow
```



La última acción, Ver Usuario mostrará el perfil del usuario que dio like a una foto de cococatmx, en este caso tavocer:

