

PLATAFORMA

NODEJS



END POINT

app/src/main/java/com/coursera/app/pm/mascotitas/restApi/ConstantesRestApi.java

```
ra x MascotaAdapter.java x EndpointsApi.java x ConstantesRestApi.java x FollowDeserializador.java x MascotaDeserializador.java x 5
public static final String KEY_GET_RECENT_MEDIA_USER_ID = "users/{user-id}/media/recent/";
public final static String USER_ID_MEDIA_RECENT = "users/{user-id}/";

//Obtiene la informacion de un usuario
public static final String URL_GET_RECENT_MEDIA_USER = KEY_GET_RECENT_MEDIA_USER + KEY_ACCESS_TOKEN + ACCESS_TOKEN;
//https://api.instagram.com/v1/users/{user-id}/media/recent/?access_token=ACCESS-TOKEN

public static final String URL_GET_RECENT_MEDIA_USER_TEST = KEY_GET_RECENT_MEDIA_USER + KEY_ACCESS_TOKEN + ACCESS_TOKEN_TEST;
//Consulta my server HEROKU
public static final String SERVER_URL_BASE = "https://mighty-forest-19952.herokuapp.com/";
public static final String SERVER_POST_TOKEN = "registrar-usuario";
public static final String KEY_TOQUE_ANIMAL = "toque-animal/{id}/{animal}";

//https://api.instagram.com/v1/users/3505167991/relationship?access_token=3440197046.f096016.6e701e3ec983405bbd6bfef9d665a51
public final static String KEY_SET_FOLLOW = "users/{user-id}/relationship?access_token="+ACCESS_TOKEN_TEST ;
public final static String KEY_GET_FOLLOWS = "follows";

// https://api.instagram.com/v1/users/self/?access_token=ACCESS-TOKEN
public final static String KEY_GET_PROFILE_USER = "users/self/";
```

HEROKU

Personal apps > mighty-forest-19952 ★ Open app More ↻

Overview Resources Deploy Metrics Activity Access Settings

Installed add-ons **\$0.00/month** Configure Add-ons

There are no add-ons for this app
You can add add-ons to this app and they will show here. [Learn more](#)

Dyno formation **\$0.00/month** Configure Dynos

This app is using free dynos

web node index.js **ON**

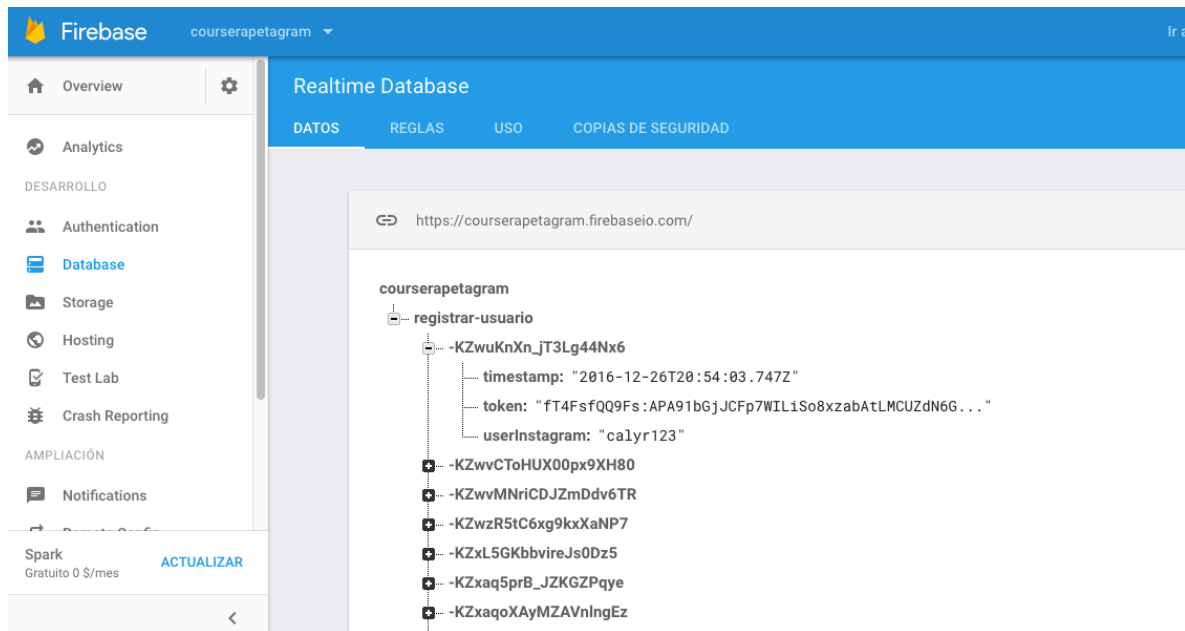
Collaborator activity Manage Access

calyr.software@gmail.com 10 deploys

Latest activity All Activity

- calyr.software@gmail.com: Deployed **2f3a9f4**
8 days ago • v43
- calyr.software@gmail.com: Build succeeded
8 days ago • [View build log](#)
- calyr.software@gmail.com: Deployed **27bc3d1**
8 days ago • v42
- calyr.software@gmail.com: Build succeeded
8 days ago • [View build log](#)
- calyr.software@gmail.com: Deployed **ed1db68**
8 days ago • v41
- calvr.software@email.com: Build succeeded

FIREBASE



TAREA 2

END POINT LIKE

`@POST(ConstantsRestApi.LIKES_URL_MEDIA)`
`Call<LikesMedia> likeImagen(@Path("media_id") String mediaId);`

app/src/main/java/com/coursera/app/pm/mascotitas/restApi/ConstantesRestApi.java

`// Likes media`

`public final static String LIKES_URL_MEDIA = "media/{media_id}/likes"+ KEY_ACCESS_TOKEN + ACCESS_TOKEN;`

ENVIAR LA NOTIFICACION HEROKU WITH FIREBASE

`@GET(ConstantsRestApi.SEND_NOTIFICACION)`
`Call<UsuarioResponse> sendNotification(@Path("token") String token, @Path("usuario_instagram") String usuario, @Path("foto_id") String foto);`

```
va x MascotaAdapter.java x EndpointsApi.java x ConstantesRestApi.java x FollowDeserializador.java x MascotaDeserializador.java

//Obtener los medios recientes por id de usuario
//https://api.instagram.com/v1/users/{user-id}/media/recent/?access_token=ACCESS-TOKEN
public static final String KEY_GET_USERS= "users/{user-id}/media/recent/";
public final static String URL_GET_FOLLOWS_MEDIA_RECENT = KEY_GET_USERS +KEY_ACCESS_TOKEN + ACCESS_TOKEN;

// https://api.instagram.com/v1/users/{user-id}?access_token=ACCESS-TOKEN
public final static String URL_GET_PROFILE_USER_WITH_ID = USER_ID_MEDIA_RECENT + KEY_ACCESS_TOKEN + ACCESS_TOKEN;

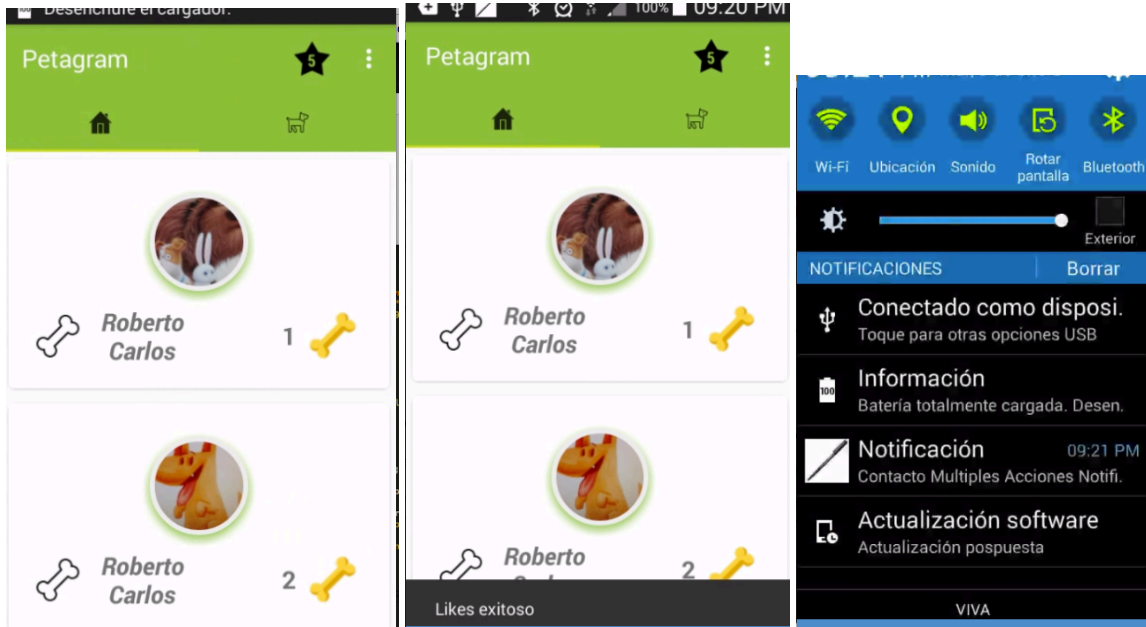
// Likes media
//https://api.instagram.com/v1/media/1394325059561543348_3584052004/likes
public final static String LIKES_URL_MEDIA = "media/{media_id}/likes"+ KEY_ACCESS_TOKEN + ACCESS_TOKEN;

//
public final static String SEND_NOTIFICACION = "send-notification/{token}/{usuario_instagram}/{foto_id}";
}
```

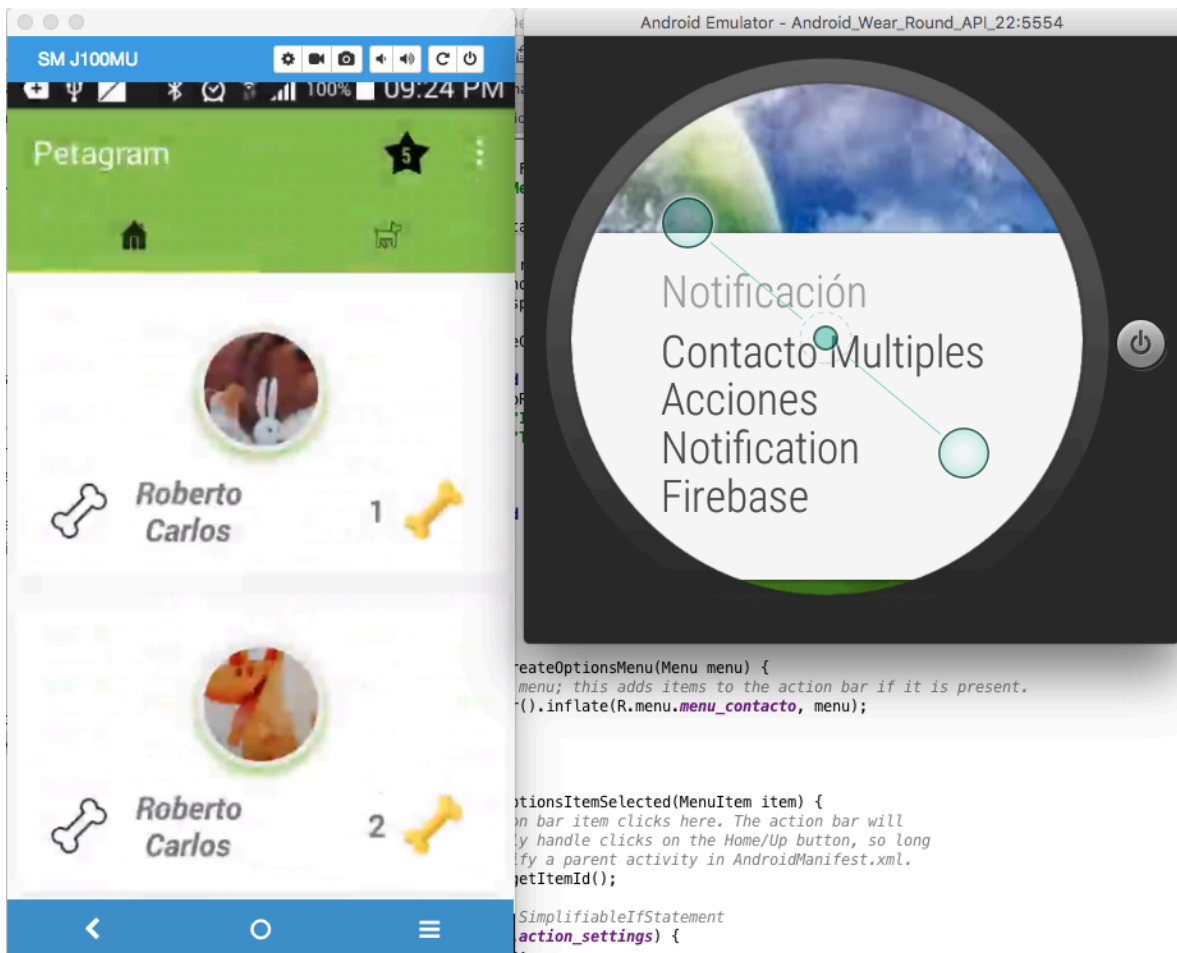
ENVIAR LA NOTIFICACION HEROKU WITH FIREBASE

```
84
85
86 app.get('/:send-notification/:token/:usuario_instagram/:foto_id',function(req,res){
87     var token = req.params.token;
88     var usuario = req.params.usuario_instagram;
89     var foto = req.params.foto_id;
90
91     res.setHeader("Content-Type", "application/json");
92     var respuesta = {
93         token: token
94     };
95     enviarNotificacion(token, "Gracias por dar likes estimado " + usuario + " a la foto " + foto);
96     res.send(JSON.stringify(respuesta));
97
98
99 });
100
101 function enviarNotificacion(token,mensaje){
102     console.log("El token " + token + " " + mensaje);
103     var serverKey = 'AAAAH9Uk36E:APA91bG9JeqpvX_05IMncSdWba5Td7aPo@n-yonaxnQQRf78xga01V_10yuWb51pB1Y1vPe76SNY6blef44YGvkkEF7PQP3mHP1tXLgmR0ooKPGbJH3eM';
104     var fcm = new FCM(serverKey);
105
106     var message = {
107         to: token, // required fill with device token or topics
108         collapse_key: '',
109         data: {},
110         notification: {
111             title: 'Notificacion desde el servidor',
112             body: mensaje,
113             icon: "bone",
114             color: "#000CD4"
115         }
116     };
117     console.log(JSON.stringify(message));
118     //callback style
119     fcm.send(message, function(err, response){
120         if (err) {
121             console.log(JSON.stringify(err));
122             console.log("Something has gone wrong!");
123         } else {
124             console.log("Successfully sent with response: ", response);
125         }
126     });
127 }
```

DAR LIKE Y NOTIFICACION

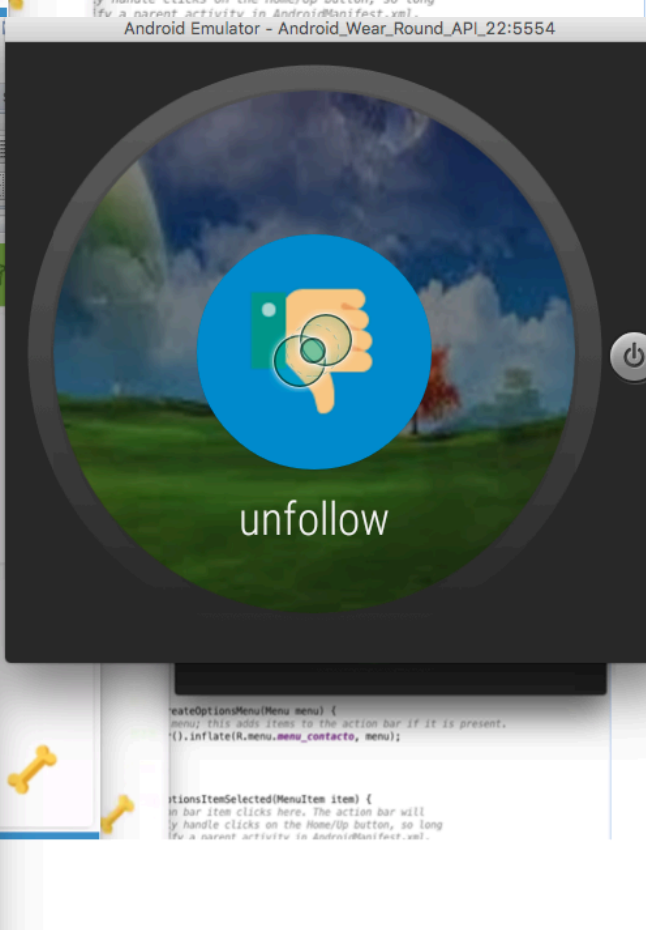
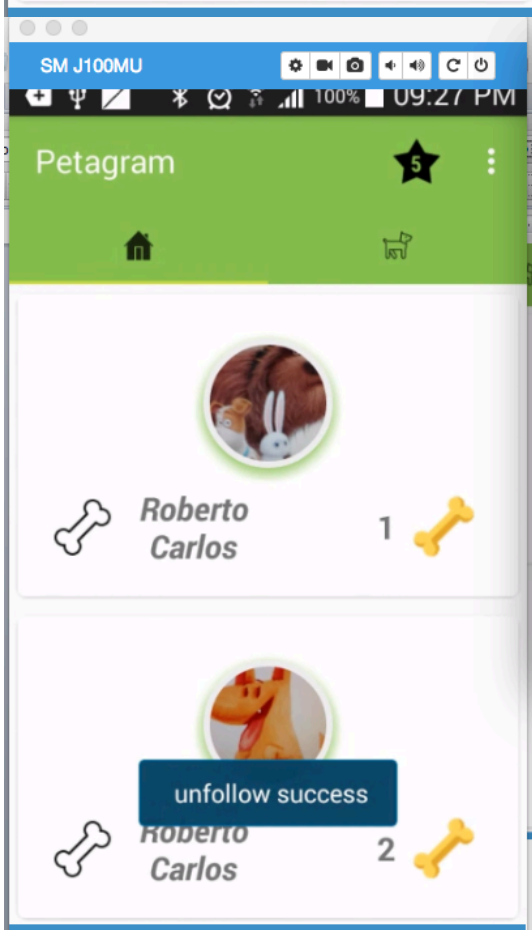
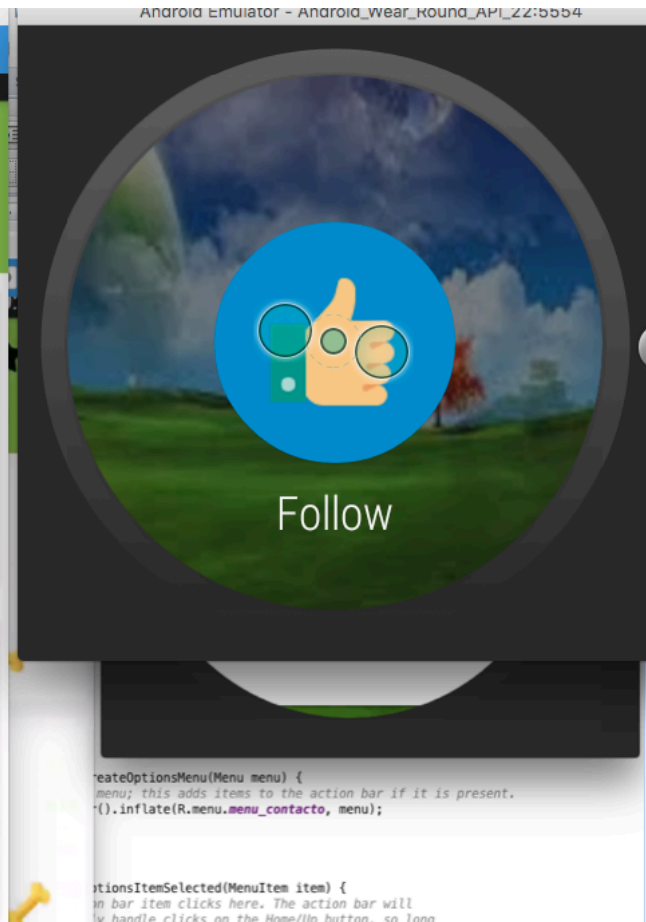
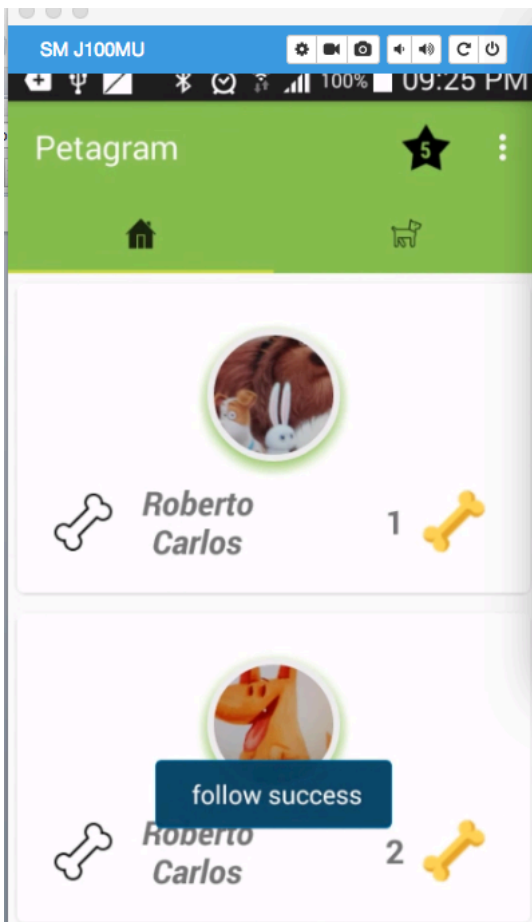


MULTIPLES NOTIFICACIONES ANDROID WEAR

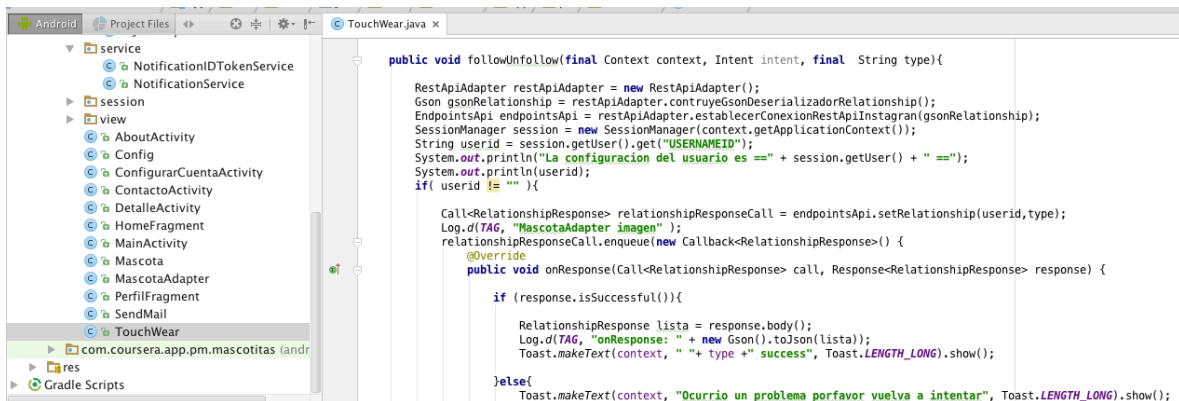


ACCION FOLLOW

UNFOLLOW



API REST recibe un parametro follow y unfollow

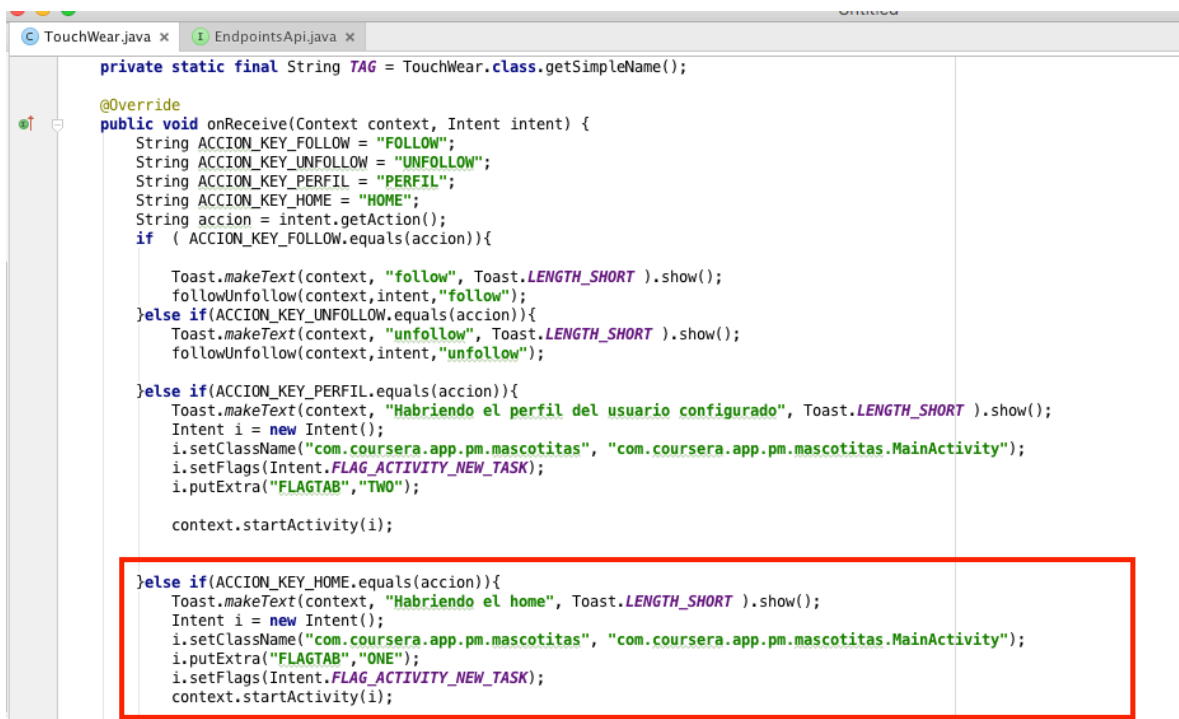


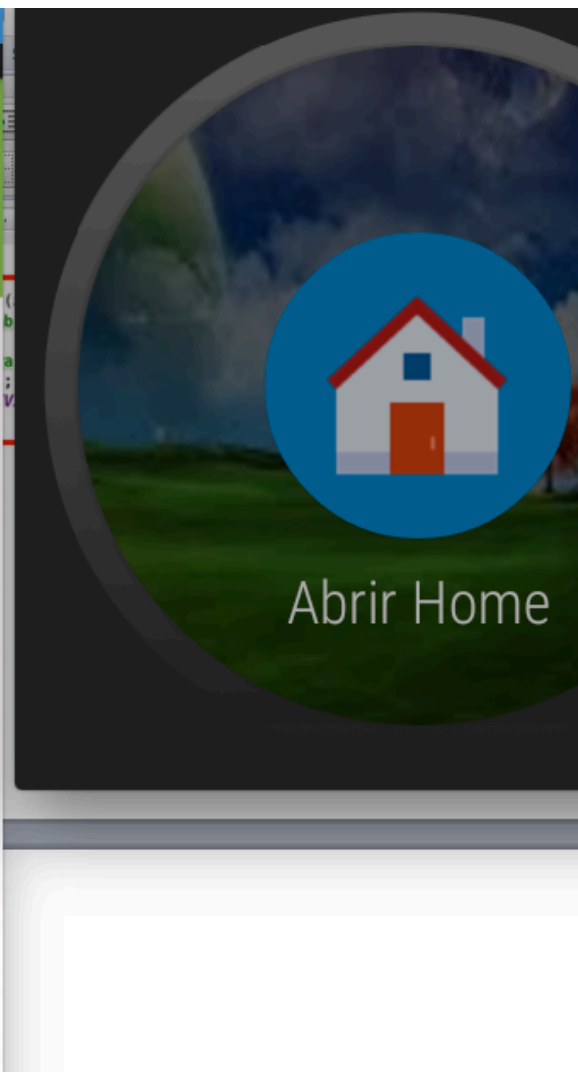
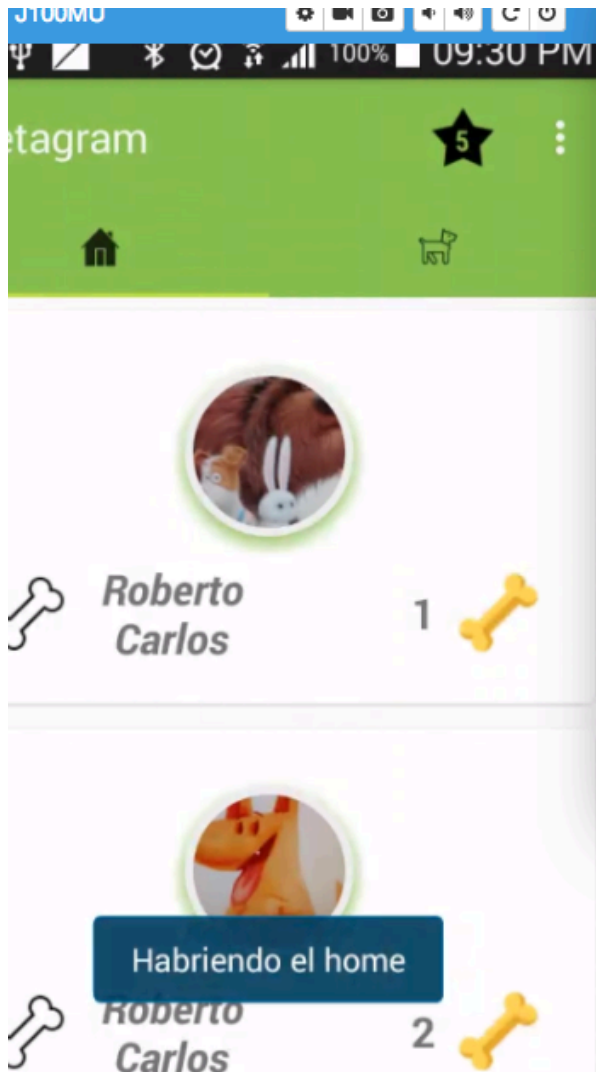
@FormUrlEncoded

@POST(ConstantsRestApi.KEY_SET_FOLLOW)

Call<RelationshipResponse> setRelationship(@Path("user-id") String id, @Field("action") String action);

ACCION HOME





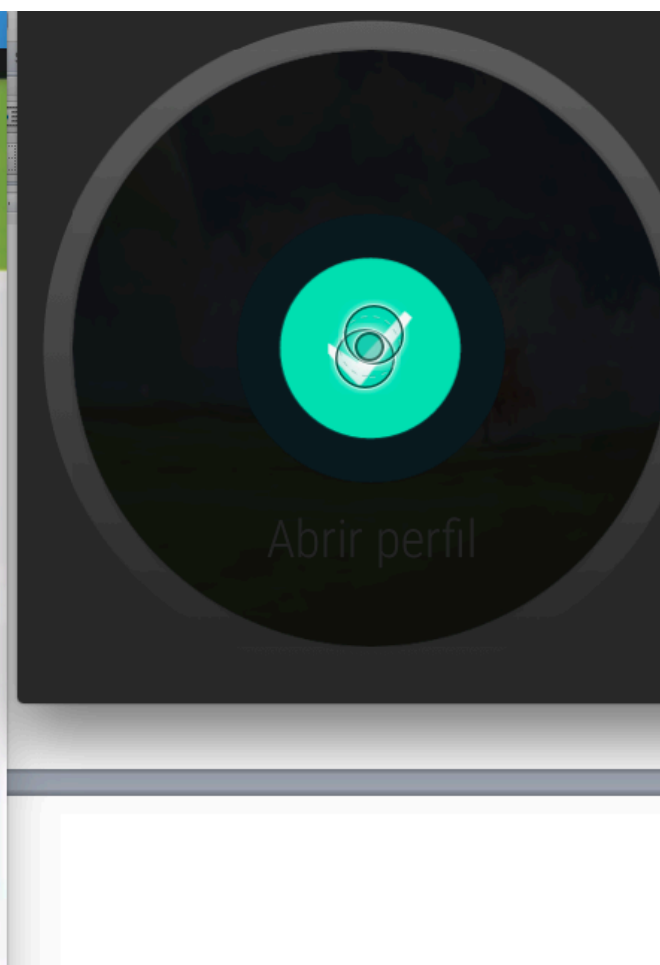
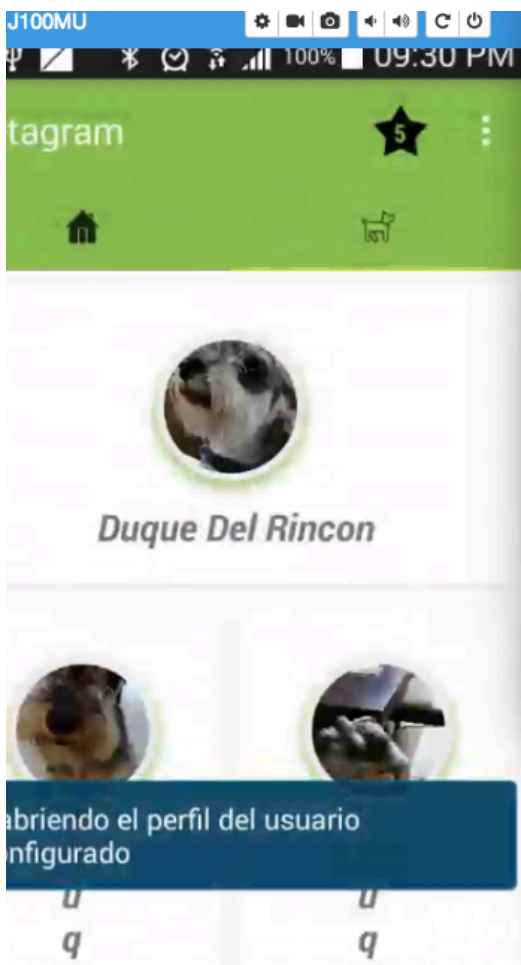
ACCION PERFIL


```

TouchWear.java x EndpointsApi.java x
private static final String TAG = TouchWear.class.getSimpleName();

@Override
public void onReceive(Context context, Intent intent) {
    String ACCION_KEY_FOLLOW = "FOLLOW";
    String ACCION_KEY_UNFOLLOW = "UNFOLLOW";
    String ACCION_KEY_PERFIL = "PERFIL";
    String ACCION_KEY_HOME = "HOME";
    String accion = intent.getAction();
    if (ACCION_KEY_FOLLOW.equals(accion)){
        Toast.makeText(context, "follow", Toast.LENGTH_SHORT).show();
        followUnfollow(context,intent,"follow");
    }else if(ACCION_KEY_UNFOLLOW.equals(accion)){
        Toast.makeText(context, "unfollow", Toast.LENGTH_SHORT).show();
        followUnfollow(context,intent,"unfollow");
    }else if(ACCION_KEY_PERFIL.equals(accion)){
        Toast.makeText(context, "Habriendo el perfil del usuario configurado", Toast.LENGTH_SHORT).show();
        Intent i = new Intent();
        i.setClassName("com.coursera.app.pm.mascotitas", "com.coursera.app.pm.mascotitas.MainActivity");
        i.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        i.putExtra("FLAGTAB", "TWO");
        context.startActivity(i);
    }else if(ACCION_KEY_HOME.equals(accion)){
        Toast.makeText(context, "Habriendo el home", Toast.LENGTH_SHORT).show();
        Intent i = new Intent();
        i.setClassName("com.coursera.app.pm.mascotitas", "com.coursera.app.pm.mascotitas.MainActivity");
        i.putExtra("FLAGTAB", "ONE");
        i.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        context.startActivity(i);
    }
}

```



CODIGO MULTIPLES ACCCIONES

```
import ...  
/**  
 * @author Roberto Carlos Callisaya Mamani  
 * @version 1.0  
 */  
public class NotificationService extends FirebaseMessagingService {  
    public static final String TAG = "NotificationService";  
    public static final int NOTIFICATION_ID = 001;  
    @Override  
    public void onMessageReceived(RemoteMessage remoteMessage) {  
        super.onMessageReceived(remoteMessage);  
        Log.d(TAG, "From" + remoteMessage.getFrom());  
        Log.d(TAG, "Notification Message Body" + remoteMessage.getNotification().getBody());  
  
        Intent i = new Intent(this, MainActivity.class);  
        PendingIntent pendingIntent = PendingIntent.getActivity(this, 0, i, PendingIntent.FLAG_ONE_SHOT);  
  
        Uri sonido = RingtoneManager.getDefaultUri(RingtoneManager.TYPE_ALARM);  
  
        //NOTIFICACION FOLLOW  
        Intent iFollow = new Intent();  
        iFollow.setAction("FOLLOW");  
        PendingIntent pendingIntentFollow = PendingIntent.getBroadcast(this, 0, iFollow, PendingIntent.FLAG_UPDATE_CURRENT);  
  
        android.support.v7.app.NotificationCompat.Action actionFollow =  
            new NotificationCompat.Action.Builder(R.drawable.ic_full_follow, "Follow", pendingIntentFollow)  
                .build();  
  
        //NOTIFICACION UNFOLLOW  
        Intent iUnfollow = new Intent();  
        iUnfollow.setAction("UNFOLLOW");  
        PendingIntent pendingIntentUnfollow = PendingIntent.getBroadcast(this, 0, iUnfollow, PendingIntent.FLAG_UPDATE_CURRENT);  
  
        android.support.v7.app.NotificationCompat.Action actionUnfollow =
```

```
        .build();  
  
        android.support.v7.app.NotificationCompat.WearableExtender wearableExtender =  
            new NotificationCompat.WearableExtender()  
                .setHintHideIcon(true)  
                .setBackground(BitmapFactory.decodeResource(getResources(), R.drawable.backgroundwear))  
                .setGravity(Gravity.CENTER_VERTICAL);  
  
        NotificationCompat.Builder notification = new NotificationCompat.Builder(this)  
            .setSmallIcon(R.drawable.bone)  
            .setContentTitle("Notificación")  
            .setContentText("Contacto Múltiples Acciones Notification Firebase")  
            .setSound(sonido)  
            .setAutoCancel(true)  
            .extend(wearableExtender.addAction(actionFollow).addAction(actionUnfollow).addAction(actionHome).addAction(actionUnfollow));  
  
        NotificationManagerCompat notificationManager = (NotificationManagerCompat.from(this));  
        notificationManager.notify(0, notification.build());
```