

Tarea 3.2: Creando, Recibiendo y Actuando Push Notifications.

Se creó el Endpoint para dar like a un foto en Instagram.

```
@GET(ConstantsRestAPI.URL_USER_ID_RECENT_MEDIA)
Call<MascotaIdResponse> getRecentMediaUserId(@Path("user-id") String idUser);

@GET(ConstantsRestAPI.URL_USER_ID)
Call<UsuarioIdResponse> getUserId(@Path("user-id") String idUser);

@POST(ConstantsRestAPI.URL_USER_PHOTO_LIKE)
Call<MascotaLikeResponse> setLikeUserPhoto(@Path("media-id") String idfoto);
```

```
private String deserializadorMascotaLikeJson(JSONObject mascotaLikeResponseData){
    ArrayList<MascotaLikeResponse> mascotaLikeResponses = new ArrayList<>();

    JSONObject jsonObjectUser = mascotaLikeResponseData.getAsJsonObject(JsonKeys.LIKE_META);
    String sCode = jsonObjectUser.get(JsonKeys.LIKE_CODE).getAsString();

    return sCode;
}
```

```
public Gson construyeGsonDeserializadorUserPhotoLike(){
    GsonBuilder gsonBuilder = new GsonBuilder();
    gsonBuilder.registerTypeAdapter(MascotaLikeResponse.class, new MascotaLikeDeserializador());
    return gsonBuilder.create();
}
```

Node js para usar el API de Instagram:

```
app.post('/', tokenUsuarioFoto, function(request, response){
    var idfoto = request.body.idfoto;
    var idusuario = request.body.idusuario;
    var iddispositivo = request.body.iddispositivo;

    var respuesta = {};
    var usuario = "";
    var db = firebase.database();
    var ref = db.ref(tokenDevicesURI);
    var usuarioencontrado = false;
    var notify = false;
    var susuario = "";

    ref.orderByChild("idusuario").equalTo(idusuario).on("child_added", function(snapshot){
        usuarioencontrado = true;
        usuario = snapshot.val();

        if(iddispositivo == usuario.iddispositivo && idusuarior != usuario.idusuario){
            guardarLikeMascota(idusuario, idfoto, iddispositivo);
        }

        var subdb = db.ref(tokenDevicesURI);
        subdb.orderByChild("iddispositivo").equalTo(iddispositivo).on("child_added", function(snapshot){
            device = snapshot.val();

            susuario = device.susuario;
        });

        enviarNotificacion(usuario.iddispositivo, susuario);
        notify = true;

        respuesta = {
            idusuario: susuario,
            idfoto: idfoto,
            iddispositivo: usuario.iddispositivo,
            notify: notify
        };

        console.log("Se encontró a un usuario.");
    });
});
```

```

    }, function(error){
        console.log("Error al leer: " + error.code);
        respuesta = {
            idusuario: "",
            idfoto: "",
            iddispositivo: "",
            notify: false
        };
    });

    if(usuarioencontrado == false){
        console.log("No se encontró a un usuario.");
        respuesta = {
            idusuario: "Usuario no encontrado",
            idfoto: "Sin foto",
            iddispositivo: "No device found.",
            notify: false
        };
    }

    response.send(JSON.stringify(respuesta));
});

function guardarLikeMascota(idusuario, idfoto, iddispositivo){
    var db = firebase.database();
    var likeMascota = db.ref(tokenUsuarioFoto).push();

    likeMascota.set({
        idusuario: idusuario,
        idfoto: idfoto,
        iddispositivo: iddispositivo
    });
}

```

Endpoint para guardar el like de una foto en firebase

```
public interface EndpointsAPIFirebase {

    @FormUrlEncoded
    @POST(ConstantsRestAPIFirebase.TOKEN)
    Call<UsuarioFirebaseResponse> registrarTokenId(@Field("idusuario") String idusuario,
                                                    @Field("susuuario") String susuuario,
                                                    @Field("iddispositivo") String iddispositivo);

    @FormUrlEncoded
    @POST(ConstantsRestAPIFirebase.TOKEN_FOTO_LIKE)
    Call<UsuarioLikeResponse> registrarLikeUsuario(@Field("idusuario") String idusuario,
                                                    @Field("idfoto") String idfoto,
                                                    @Field("iddispositivo") String iddispositivo);
}
```

```
private boolean notify;

public UsuarioLikeResponse(String id, String iddispositivo, String idfoto, String idusuario, boolean notify) {
    this.id = id;
    this.iddispositivo = iddispositivo;
    this.idfoto = idfoto;
    this.idusuario = idusuario;
    this.notify = notify;
}

public UsuarioLikeResponse() {
}

public String getId() { return id; }

public void setId(String id) { this.id = id; }

public String getIddispositivo() { return iddispositivo; }

public void setIddispositivo(String iddispositivo) { this.iddispositivo = iddispositivo; }

public String getIdfoto() { return idfoto; }

public void setIdfoto(String idfoto) { this.idfoto = idfoto; }

public String getIdusuario() { return idusuario; }

public void setIdusuario(String idusuario) { this.idusuario = idusuario; }

public boolean isNotify() { return notify; }

public void setNotify(boolean notify) { this.notify = notify; }
```

```
public class ConstantesRestAPIFirebase {
    public static final String TOKEN_URL = "https://cryptic-ocean-21392.herokuapp.com/";
    public static final String TOKEN = "registrar-usuario";
    public static final String TOKEN_FOTO_LIKE = "like-usuario/";
}
```

Configuración firebase:

```
app.post('/', + tokenUsuarioFoto, function(request, response){
  var idfoto = request.body.idfoto;
  var idusuario = request.body.idusuario;
  var iddispositivo = request.body.iddispositivo;

  var respuesta = {};
  var usuario = "";
  var db = firebase.database();
  var ref = db.ref(tokenDevicesURI);
  var usuarioencontrado = false;
  var notify = false;
  var susuario = "";

  ref.orderByChild("idusuario").equalTo(idusuario).on("child_added", function(snapshot){
    usuarioencontrado = true;
    usuario = snapshot.val();

    if(iddispositivo == usuario.iddispositivo && idusuarior != usuario.idusuario){
      guardarLikeMascota(idusuario, idfoto, iddispositivo);
    }

    var subdb = db.ref(tokenDevicesURI);
    subdb.orderByChild("iddispositivo").equalTo(iddispositivo).on("child_added", function(snapshot){
      device = snapshot.val();

      susuario = device.susuario;
    });

    enviarNotificacion(usuario.iddispositivo, susuario);
    notify = true;

    respuesta = {
      idusuario: susuario,
      idfoto: idfoto,
      iddispositivo: usuario.iddispositivo,
      notify: notify
    };

    console.log("Se encontró a un usuario.");
  });
```

```
}, function(error){
  console.log("Error al leer: " + error.code);
  respuesta = {
    idusuario: "",
    idfoto: "",
    iddispositivo: "",
    notify: false
  };
});

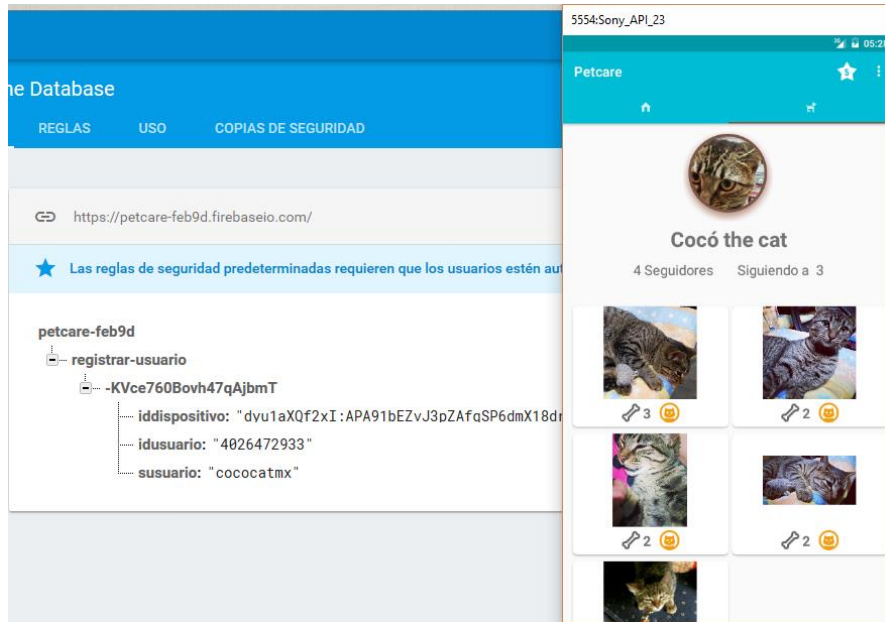
if(usuarioencontrado == false){
  console.log("No se encontró a un usuario.");
  respuesta = {
    idusuario: "Usuario no encontrado",
    idfoto: "Sin foto",
    iddispositivo: "No device found.",
    notify: false
  };
}

response.send(JSON.stringify(respuesta));
});
```

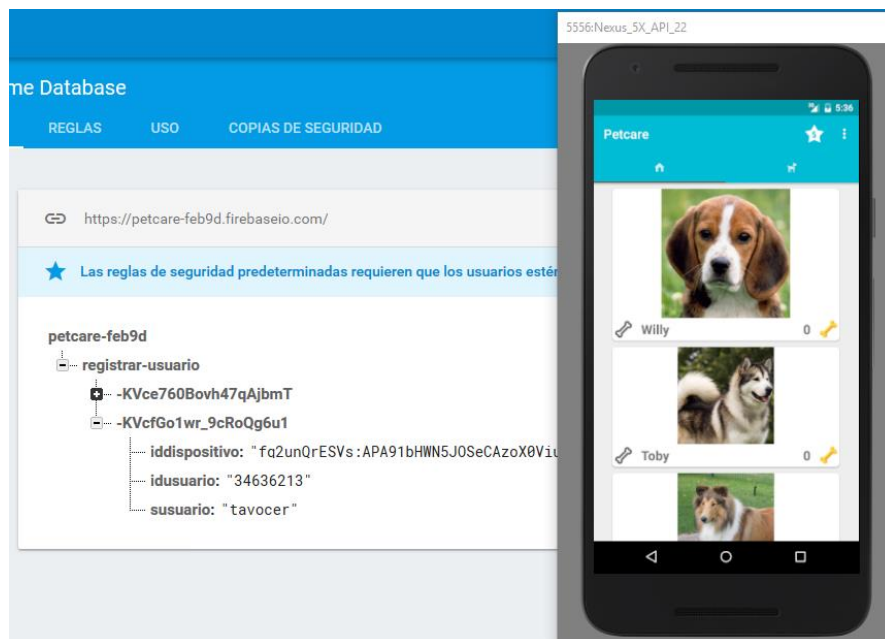
Aplicación funcionando

Se registran primero los emuladores para poder ver el ejercicio de notificaciones.

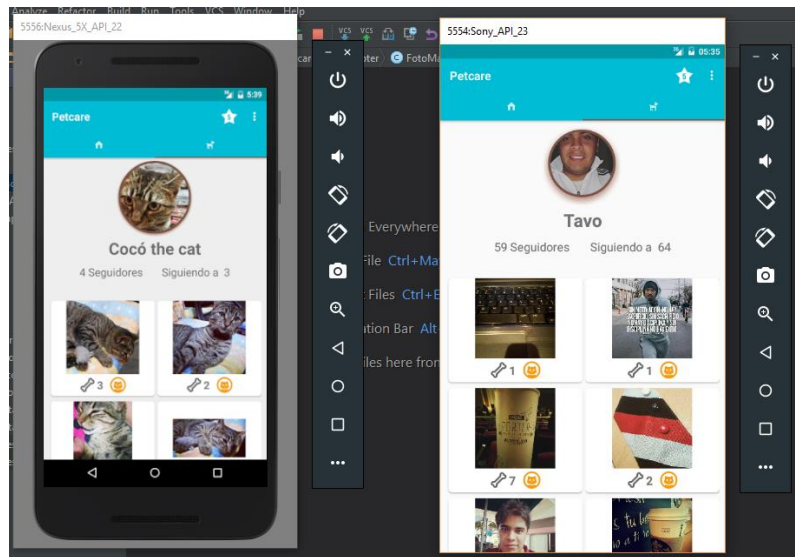
Cococatmx – Sony API 23



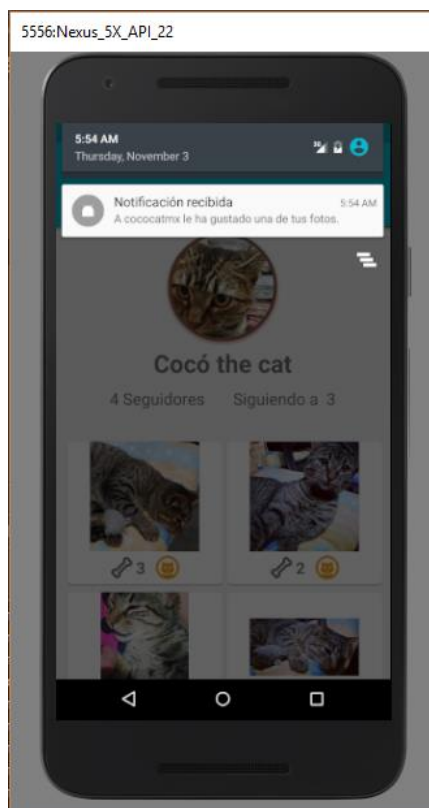
Tavocer – Nexus 5X API 22



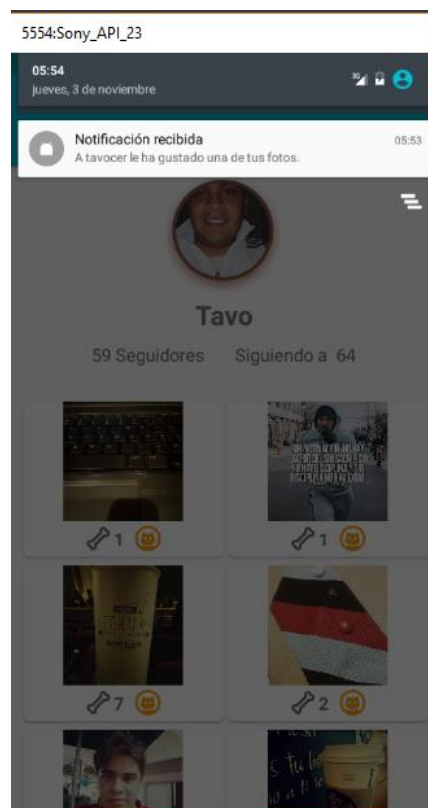
Ahora el emulador con el usuario cococatmx mostrará el perfil de Instagram de tavocer y el otro emulador hará lo mismo con el perfil de cococatmx.



El siguiente paso es dar like en alguna de las fotos de cococatmx



TAVOCER



COCOCATMX

Al dar clic en la notificación recibida, se abrirá la pantalla con el timeline.

Aquí se muestra el log indicando que se envían los likes:

```
11-03 05:43:18.407 26360-26360/mx.octaviocervantes.mypetcare D/NOTIFICACION_FOTO: 1357723650514395641_4026472933
11-03 05:43:18.407 26360-26360/mx.octaviocervantes.mypetcare D/NOTIFY: true
11-03 05:43:42.588 26360-26360/mx.octaviocervantes.mypetcare D/BEOFRE_Receptor: 4026472933
11-03 05:43:42.588 26360-26360/mx.octaviocervantes.mypetcare D/BEOFRE_FOTO: 1357723650514395641_4026472933
11-03 05:43:42.588 26360-26360/mx.octaviocervantes.mypetcare D/DISPOSITIVO_EMITOR: fq2unQrESVs:APA91bHWN5J0SeCAzoX0ViukTmW2
11-03 05:43:43.608 26360-26360/mx.octaviocervantes.mypetcare D/NOTIFICACION_USER: tavocer
11-03 05:43:43.608 26360-26360/mx.octaviocervantes.mypetcare D/NOTIFICACION_DVC: dyu1aXQf2xI:APA91bEZvJ3pZafqSP6dmX18dr-Ux
11-03 05:43:43.608 26360-26360/mx.octaviocervantes.mypetcare D/NOTIFICACION_FOTO: 1357723650514395641_4026472933
11-03 05:43:43.608 26360-26360/mx.octaviocervantes.mypetcare D/NOTIFY: true
```

Base de datos Firebase

