# Configuración endpoint en servidor

Implementación servicio POST Registro dispositivos y cuentas instagram

El servicio se implementó con Heroku y NodeJS

**URL POST** 

https://secret-beyond-94805.herokuapp.com/like\_timeline

datos de entrada

- ✓ Token
- √ id\_usuario\_instagram
- √ id\_foto\_instagram

Servicio GET

https://secret-beyond-94805.herokuapp.com/like-instagram/:id/:userinstagram

Configuración servicio en Heroku con nodeJS

```
var tokenDevicesURI_likeTimeline = "like_timeline";

app.post('/' + tokenDevicesURI_likeTimeline, function(request, response){
    var token = request.body.token;
    var id_usuario_instagram = request.body.id_usuario_instagram;
    var id_foto_instagram = request.body.id_foto_instagram;

var db = firebase.database();
    var tokenDevices = db.ref(tokenDevicesURI_likeTimeline).push();
    tokenDevices.set({
        token,
```

```
id_usuario_instagram: id_usuario_instagram,
               id_foto_instagram: id_foto_instagram
       });
        var path = tokenDevices.toString(); //https://petagram-9e8df.firebaseio.com/registrar-
usuario/-KfX73UtIpfgtOiwC2CF
        var pathSplit = path.split(tokenDevicesURI_likeTimeline + "/");
        var idAutoGenerado = pathSplit[1];
        var respuesta = generarRespuestaRegisterLikes(db, idAutoGenerado);
        response.setHeader("Content-Type", "application/json");
        response.send(JSON.stringify(respuesta)); //imprime en json pero formato string
        //response.send(request.body.token);
});
function generarRespuestaRegisterLikes(db, idAutoGenerado){
        var respuesta = {};
        var usuario = "";
        var ref = db.ref("like timeline");
        ref.on("child_added",function(snapshot, prevChildKey){
               usuario = snapshot.val();
               respuesta = {
                       id: idAutoGenerado,
                       token: usuario.token,
                       id_usuario_instagram: usuario.id_usuario_instagram,
                       id foto instagram: usuario.id foto instagram
               }
       });
        return respuesta;
}
//GET
//https://secret-beyond-94805.herokuapp.com/like-instagram
//id
//userinstagram
app.get("/like-instagram/:id/:userinstagram", function(request, response){
                       = request.params.id;
        var id
        var userinstagram
                               = request.params.userinstagram;
```

```
var db = firebase.database();
        var ref = db.ref("registrar-usuario/" + id);
        var usuario = "";
        var respuesta = {};
        ref.on("value", function(snapshot) {
               //console.log(snapshot.val());
               usuario = snapshot.val();
               var mensaje = "El usuario" + userinstagram + " te dio un like en una de tus fotos
:)";
               enviarNotificacion(usuario.token, mensaje);
               respuesta = {
                       id: id,
                       token: usuario.token,
                       userinstagram: usuario.userinstagram
               };
               response.setHeader("Content-Type", "application/json");
               response.send(JSON.stringify(respuesta));
        }, function (errorObject) {
               console.log("The read failed: " + errorObject.code);
               respuesta = {
                       id: "",
                       token: "",
                       userinstagram: ""
               };
               response.setHeader("Content-Type", "application/json");
               response.send(JSON.stringify(respuesta));
       });
});
function enviarNotificacion(tokenDestinatario, mensaje) {
        var serverKey = 'AAAAddmBPZQ:APA91bEKjB1PVX-gwCYt3EmdtXcAFB_xEoAsytf-
9ZSgDLE3zemZObqk_G7uTWWmw8B9pG3ekrZwO8fOKi4jkGrYLWjxKlkX8qlKo8JXJSifjtHG38qDJxc9f
Xastf8gjgmfUWYwVEiH';
       var fcm = new FCM(serverKey);
```

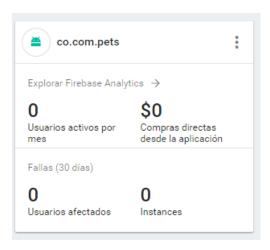
```
var message = {
          to: tokenDestinatario, // required
          collapse_key: ",
          data: {},
          notification: {
            title: 'Notificacion desde Servidor',
            body: mensaje,
            icon: "notificacion",
            sound: "default",
            color: "#00BCD4"
          }
        };
        fcm.send(message, function(err, response){
          if (err) {
            console.log("Something has gone wrong!");
          } else {
            console.log("Successfully sent with response: ", response);
          }
        });
        // fcm.send(message)
  // .then(function(response){
    // console.log("Successfully sent with response: ", response);
  // })
  // .catch(function(err){
    // console.log("Something has gone wrong2!");
    // console.error(err);
  // })
}
Para finalizar se realiza despliegue del servicio a Heroku.
        git add.
        git commit -m "Implementando servicio POST y GET"
        git push heroku master
```

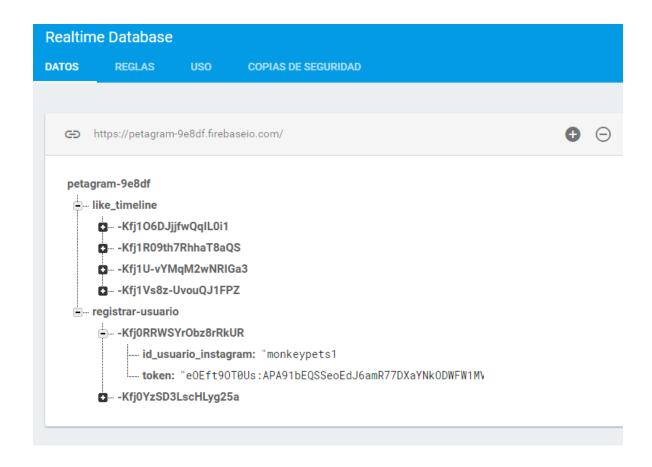
## **Almacenamiento datos**

Se hace uso de la Base de datos en **FIREBASE** 

## Pasos

- 1. Se realiza registro del paquete
  - a. co.com.pets





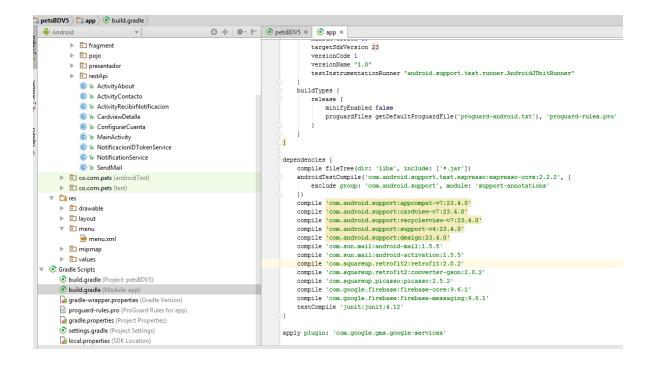
#### Consumiendo servicio POST en heroku

#### Resultado

# Implementado la app en proyecto android

1. configuración librerías

```
petsBDV5 🕞 build.gradle
                                                   // Top-level build file where you can add configuration options common to all sub-p
             i fragment
          ▶ in pojo
                                                                        buildscript {
          ▶ □ presentador
                                                                           repositories {
          jcenter()
             C & ActivityAbout
             © 🚡 ActivityContacto
                                                                            dependencies {
                                                                                classpath 'com.android.tools.build:gradle:2.2.3'
             © 🚡 ActivityRecibirNotificacion
                                                                            classpath 'com.google.gms:google-services:3.0.0'
             © 🚡 CardviewDetalle
             © 🚡 ConfigurarCuenta
                                                                                // NOTE: Do not place your application dependencies here; they belong // in the individual module build.gradle files \,
             © 🖰 MainActivity
             © a NotificacionIDTokenService
             © a NotificationService
             © & SendMail
                                                                        allprojects {
      co.com.pets (androidTest)
                                                                           repositories {
       co.com.pets (test)
                                                                               jcenter()
    ▼ 📑 res
       ▶ layout
                                                                        task clean(type: Delete) {
       ▼ 🛅 menu
                                                                           delete rootProject.buildDir
             menu.xml
       ▶ imipmap
       ▶ 🛅 values
    Gradle Scripts
          build.gradle (Project: petsBl
       ( huild gradle (Module: app)
```



```
🔁 petsBDV5 🕽 📴 app 🕽 📀 build.gradle
                                                            ₩ Android
            ▶ ☐ fragment
                                                                                              targetSdkVersion 23
            versionCode 1
                                                                                              versionName "1.0"
testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
            ▶ ☐ presentador
           ▶ i restApi
               © & ActivityAbout
                                                                                          buildTypes {
               © & ActivityContacto
                                                                                              release {
               © 🚡 ActivityRecibirNotificacion
                                                                                                  minifyEnabled false
               © & CardviewDetalle
                                                                                                  proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
               © & ConfigurarCuenta
               © & MainActivity
               © & NotificacionIDTokenService
               © a NotificationService
                                                                                    dependencies {
               C 🗈 SendMail
                                                                                         compile fileTree(dir: 'libs', include: ['*.jar'])
                                                                                         androidTestCompile('com.android.support.test.espresso:espresso-core:2.2.2', {
   exclude group: 'com.android.support', module: 'support-annotations'
       co.com.pets (androidTest)
       co.com.pets (test)
    ▼ 🛅 res
                                                                                          compile 'com.android.support:appcompat-v7:23.4.0'
       ▶ a drawable
                                                                                         compile 'com.android.support:cardview-v7:23.4.0'
compile 'com.android.support:recyclerview-v7:23.4.0'
        ▶ i layout
                                                                                         compile 'com.android.support:support-v4:23.4.0'
compile 'com.android.support:design:23.4.0'
compile 'com.sun.mail:android-mail:1.5.5'
        ▼ 🛅 menu
               menu.xml
        ▶ imipmap
                                                                                         compile 'com.sun.mail:android-activation:1.5.5'
compile 'com.squareup.retrofit2:retrofit:2.0.2'
        ▶ 🖻 values

▼ Gradle Scripts

                                                                                          compile 'com.squareup.retrofit2:converter-gson:2.0.2'
        build.gradle (Project: petsBDV5)
                                                                                          compile 'com.squareup.picasso:picasso:2.5.2'
        build.gradle (Module: app)
                                                                                          compile 'com.google.firebase:firebase-core:9.6.1'
compile 'com.google.firebase:firebase-messaging:9.6.1'
        gradle-wrapper.properties (Gradle Version)
                                                                                          testCompile 'junit:junit:4.12'
        proguard-rules.pro (ProGuard Rules for app)
        gradle.properties (Project Properties)
        settings.gradle (Project Settings)
                                                                                     apply plugin: 'com.google.gms.google-services'
        local.properties (SDK Location)
```