

Ejercicio de desarrollo API Heroku con Node.js



Endpoint Node.js

```
82 //endpoint para el envio de usuario instagram
83 //https://console.firebase.google.com/u/0/project/petagram4/database/petagram4/data/usuario_i
84 var registrarUsuarioDb="usuario_instagram";
85 app.post('/registrar-usuario', function(request, response) {
86   //var id_dispositivo=request.body.id_dispositivo;
87   // var id_usuario_instagram=request.body.id_usuario_instagram;
88
89   var db=firebase.database();
90   var userDevices=db.ref(registrarUsuarioDb).push();
91   userDevices.set({
92     id_dispositivo:request.body.id_dispositivo,
93     id_usuario_instagram:request.body.id_usuario_instagram
94   });
95   tokenDevicesURI=registrarUsuarioDb
96   var path=userDevices.toString();
97   var pathSplit=path.split(registrarUsuarioDb + "/");
98   var idAutoGenerado=pathSplit[1];
99   //armo la respuesta
100   var respuesta=generarRespuestaAUsuarioInstagram(db,idAutoGenerado);
101   response.setHeader("Content-Type","application/json");
102   response.send(JSON.stringify(respuesta));
103 });
104
105
106 function generarRespuestaAUsuarioInstagram(db,idAutoGenerado){
107
108   var respuesta={};
109   var usuario="";
110   //nombre de la base de datos que generalmetne es el nombre del uri
111   var ref=db.ref(tokenDevicesURI);
112   ref.on("child_added",function(snapshot,prevChildKey){
113     usuario=snapshot.val();
114     respuesta={
115       id:idAutoGenerado,
116       id_dispositivo:usuario.id_dispositivo,
117       id_usuario_instagram:usuario.id_usuario_instagram
118     };
119   });
120 };
121 return respuesta;
122 }
```

Llamada desde Android

```

140 public void EnviarUsuarioApi(){
141     //envio notificacion a servidor heroku con usuario y token
142     RestApiAdapter restApiAdapter=new RestApiAdapter();
143     EndpointsApi endpointsApi=restApiAdapter.establecerConexionRestHerokuRestApi();
144     Call<UsuarioRespuesta> usuarioRespuestaCall=endpointsApi.registrarUsuarioInstagram( id_dispositivo: "123456", id_usuario_instagram: "miUsuarioInstagram");
145
146     usuarioRespuestaCall.enqueue(new Callback<UsuarioRespuesta>() {
147         @Override
148         public void onResponse(Call<UsuarioRespuesta> call, Response<UsuarioRespuesta> response) {
149             UsuarioRespuesta usuarioRespuesta=response.body();
150             //guardo en el log la respuesta.
151             Log.d( tag: "ID_HEROKU", usuarioRespuesta.getId());
152             Log.d( tag: "ID_DISPOSITIVO", usuarioRespuesta.getId_dispositivo());
153             Log.d( tag: "ID_USUARIO", usuarioRespuesta.getId_usuario_instagram());
154         }
155
156         @Override
157         public void onFailure(Call<UsuarioRespuesta> call, Throwable t) {
158             Log.d( tag: "ID_HEROKU", msg: "Error en API");
159         }
160     });
161 }
162

```

Resultado del llamado al endpoint, como se ve se guarda en log las tres variables: ID, Id_dispositivo e Id_usuarioInstagram

La aplicación corre satisfactoriamente implementando retrofit2, heroku, node.js y firebase.

Por un tema de tiempo, no levanta el usuario real de Instagram, se envía uno creado en memoria.