

# PETAGRAM WEB SERVICES

## Semana 3 Curso 4

### Descripción de aplicación

Implementación del api de Instagram para dar like, ademas se insertan los datos del like en el servidor Firebase.

### Descripción del Endpoint

Se configura un servidor por medio de Node.js, el enlace para insertar en el servidor es el siguiente:

<https://glacial-brushlands-83248.herokuapp.com/likes>

Se usa un metodo post para insertar los datos en la base de datos.

```
var pathLikes = "likes";
app.post("/", pathLikes, function(request, response) {

  var idDispositivo = request.body.id_dispositivo;
  var idUsuarioInstagram = request.body.id_usuario_instagram;
  var idMediaInstagram = request.body.id_media_instagram;

  var db = firebase.database();
  var rfRegistroUsuario = db.ref(pathLikes).push();
  rfRegistroUsuario.set({
    id_dispositivo: idDispositivo,
    id_usuario_instagram: idUsuarioInstagram,
    id_media_instagram: idMediaInstagram
  });

  var idFirebaseGenerado = rfRegistroUsuario.key;

  var respuesta = generarRespuestaIdMedia(db, idFirebaseGenerado);
  response.setHeader("Content-Type", "application/json");
  response.send(JSON.stringify(respuesta));

  //response.send(idDispositivo + '/' + idUsuarioInstagram);

});
```

En el método se obtienen los datos del enlace y se insertan en la base de datos, al final solo se genera una respuesta en formato JSON por medio del metodo generar respuestaIdMedia().

```
function generarRespuestaIdMedia(db, idFirebaseGenerado) {

  var respuesta = {};
  var usuario = "";
  var ref = db.ref(pathLikes);
  ref.on("child_added", function(snapshot, prevChildKey) {

    usuario = snapshot.val();
    respuesta = {
      id: idFirebaseGenerado,
```

```

        idDispositivo: usuario.id_dispositivo,
        idUsuarioInstagram: usuario.id_usuario_instagram,
        idMediaInstagram: usuario.id_media_instagram
    };
}

return respuesta;
}

```

El método simplemente modela el tipo de respuesta que obtendremos.

Para obtener las notificaciones, se crea una URL que se consulta por el metodo GET

```

app.get("/like/:idFirebase/:idDispositivo/:usuario", function(request, response)
{
    var idFirebase = request.params.idFirebase;
    var idDispositivo = request.params.idDispositivo;
    var usuario = request.params.usuario;

    var db = firebase.database();
    var ref = db.ref("registrar-usuario/" + idFirebase);
    var user = "";
    var respuesta = {};
    ref.on("value", function(snapshot) {
        console.log(snapshot.val());
        user = snapshot.val();

        var mensaje = usuario + " le ha gustado tu foto";
        enviarNotificacion(user.id_dispositivo, mensaje);
        respuesta = {
            ID_Firebase: idFirebase,
            ID_Dispositivo: user.id_dispositivo,
            Usuario: usuario
        };

        response.send(JSON.stringify(respuesta));
    }, function (errorObject) {
        console.log("The read failed: " + errorObject.code);
        respuesta = {
            id: "",
            token: ""
        };
        response.send(JSON.stringify(respuesta));
    });
});

```

Dentro de este método se llama a la funcion enviarNotificacion pasandole los parametros del Id del dispositivo y un mensaje para la notificación.

```

function enviarNotificacion(tokenDestinatario, mensaje) {

    var serverKey = "AAAAPPOhNZ8:APA91bEJDBaOBuvTayY1-
TI444kIuQveEhTgETtQ6vpVh4uE94MiTkMv33Ix-
ipd_pLS8jIDdO2JJPChih1jyU2r47b7pwOFvtuyqdewv77YUPX2Hkan2t4kpLeDuP4VzhLcZ5_XunLs";
    var fcm = new FCM(serverKey);

    var message = {

```

```

to: tokenDestinatario, // required fill with device token or topics
collapse_key: '',
data: {

},
notification: {
  title: 'Petagram',
  body: mensaje,
  icon: "instagramold",
  sound: "default",
  color: "#00BCD4"
}
};

//callback style
fcm.send(message, function(err, response){
  if (err) {
    console.log("Something has gone wrong!");
  } else {
    console.log("Successfully sent with response: ", response);
  }
});

//promise style
fcm.send(message)
  .then(function(response){
    console.log("Successfully sent with response: ", response);
  })
  .catch(function(err){
    console.log("Something has gone wrong!");
    console.error(err);
  })
}

```

El trabajo lo realiza la función enviarNotificacion que usa la libreria FCM para enviar notificaciones.