

PETAGRAM WEB SERVICES

Semana 3 Curso 4

Descripción de aplicación

Web service en donde se configura un Endpoint para guardar en la base de datos de Firebase el Id del dispositivo que tiene la app y el Id del usuario de instagram.

Descripcion del Endpoint

Se configura un servidor por medio de Node.js, el mismo usado en las practicas, el enlace para el servidor es el siguiente:

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

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

```
var path = "registrar-usuario";  
app.post("/" + path, function(request, response) {  
    var idDispositivo = request.body.id_dispositivo;  
    var idUsuarioInstagram = request.body.id_usuario_instagram;  
    var db = firebase.database();  
    var rfRegistroUsuario = db.ref(path).push();  
    rfRegistroUsuario.set({  
        id_dispositivo: idDispositivo,  
        id_usuario_instagram: idUsuarioInstagram  
    });  
    var idFirebaseGenerado = rfRegistroUsuario.key;  
    var respuesta = generarRespuestaToken(db, idFirebaseGenerado);  
    response.setHeader("Content-Type", "application/json");  
    response.send(JSON.stringify(respuesta));  
});
```

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

```

function generarRespuestaToken(db, idFirebaseGenerado) {
    var respuesta = {};
    var usuario = "";
    var ref = db.ref(path);
    ref.on("child_added", function(snapshot, prevChildKey){
        usuario = snapshot.val();
        respuesta = {
            id: idFirebaseGenerado,
            idDispositivo: usuario.id_dispositivo,
            idUsuarioInstagram: usuario.id_usuario_instagram
        };
    })
    return respuesta;
}

```

El metodo simplemente modela el tipo de respuesta que obtendremos