

Configurando mi propio Endpoint en mi Servidor

Se utilizaron las herramientas (previamente instaladas) heroku, node, npm y git para el manejo del repositorio.

Para el crear el repositorio de Heroku se utilizaron los siguientes comandos:

1. heroku login
2. git clone https://github.com/heroku/node-js-getting-started.git
3. cd node-js-getting-started
4. heroku create
5. git push heroku master
6. heroku ps:scale web=1
7. heroku open

Se instalaron localmente las siguientes dependencias:

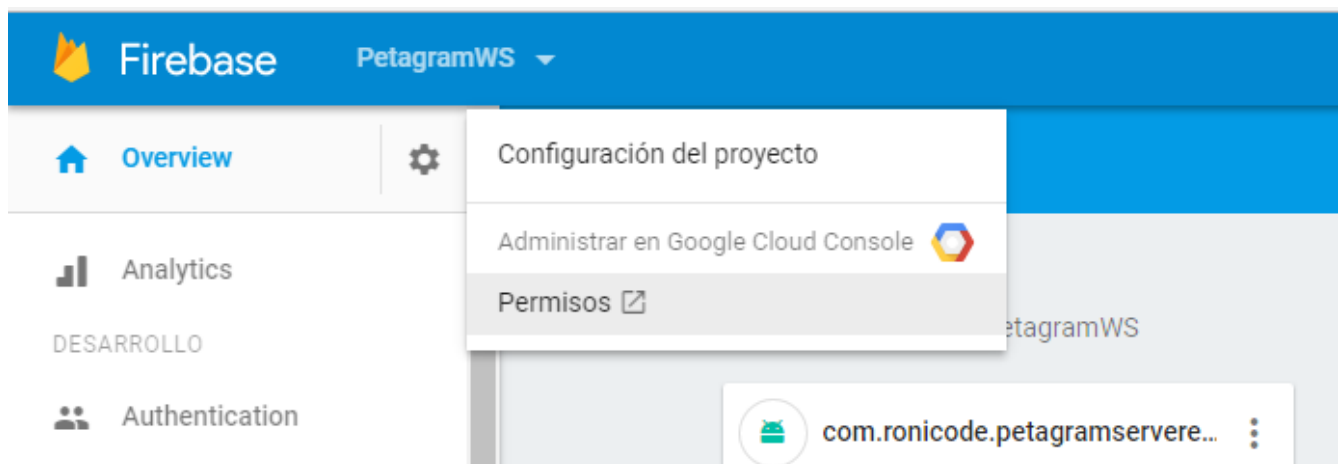
1. npm install
2. npm install --save --save-exact body-parser
3. npm install --save --save-exact firebase

FIREBASE:

URL: <https://petagramws-31c83.firebaseio.com>

ServiceAccount: PetagramWS-be19bd98d917.json

Creando la cuenta de servicio:



IAM y administrac...

Cuentas de servicio

CREAR CUENTA DE SERVICIO

+ IAM

Cuentas de servicio

Etiquetas

Configuración

Claves de cifrado

Proxy de comprobación de i...

Cuentas de servicio del proyecto "PetagramWS"

Una cuenta de servicio representa a una identidad de servicio de Google Cloud, como el i...
 fuera de Google. [Más información](#)

<input type="checkbox"/> Nombre de cuenta de servicio ^	ID de cuenta de servicio
<input type="checkbox"/> App Engine default service account	petagramws-31c83@appspot.gserviceaccount.com
<input type="checkbox"/> petagramserver	petagramserver@petagramws-31c83.iam.gserviceaccount.com

Configuración del servidor:

```

var express = require('express');
var app = express();

app.set('port', (process.env.PORT || 5000));

var bodyParser = require("body-parser");
app.use(bodyParser.json()); //soporte para codificar json
app.use(bodyParser.urlencoded({ extended: true })); //soporte para codificar las url

var firebase = require("firebase");
firebase.initializeApp({
  serviceAccount: "PetagramWS-be19bd98d917.json",
  databaseURL: "petagramws-31c83.firebaseio.com"
});

app.use(express.static(__dirname + '/public'));

// views is directory for all template files
app.set('views', __dirname + '/views');
app.set('view engine', 'ejs');

app.get('/android', function(request, response) {
  response.render('pages/index');
});

```

```

//POST
//https://limitless-cliffs-10593.herokuapp.com/regarstrar-usuario

var uri_servicio = '/regarstrar-usuario';

app.post('/regarstrar-usuario', function(request, response) {

  // Parámetros POST
  var id_dispositivo = request.body.id_dispositivo;
  var id_usuario_instagram = request.body.id_usuario_instagram;

  // Guarda los parametros en la base de datos
  var db = firebase.database();
  var db_Ref = db.ref(uri_servicio).push();
  db_Ref.set({
    id_dispositivo: id_dispositivo,
    id_usuario_instagram: id_usuario_instagram
  });

  // Obtiene el identificador del registro
  var path = db_Ref.toString(); //https://petagramws-31c83.firebaseio.com/regarstrar-usuario
  var pathSplit = path.split(uri_servicio + "/")
  var idAutoGenerado = pathSplit[1];

  // Regresa la respuesta en formato JSON
  var respuesta = generarRespuestaJSON(db, idAutoGenerado);
  response.setHeader("Content-Type", "application/json");
  response.send(JSON.stringify(respuesta));

});

```

```

function generarRespuestaJSON(db, idAutoGenerado) {
  var respuesta = {};
  var usuario = "";
  var db_Ref = db.ref(uri_servicio);
  db_Ref.on("child_added", function(snapshot, prevChildKey) {
    usuario = snapshot.val();
    respuesta = {
      id_registro: idAutoGenerado,
      id_dispositivo: usuario.id_dispositivo,
      id_usuario_instagram: usuario.id_usuario_instagram
    };
  });

  return respuesta;
}

app.listen(app.get('port'), function() {
  console.log('Node app is running on port', app.get('port'));
});

```