Firebase Workshop Session#3 March-2020

Firebase workshop by Cristian Arce & Alfredo Bonilla

Session #3

Angular Project

Clone the base project for **Session#3** from this repo into a local folder

```
$  git clone git@github.com:cristianarceGL/firebase-workshop.git

$  cd firebase-workshop

$  git checkout Session3-Base

$  git pull
```

After pulling the repo, it is mandatory to install all dependencies and run the app

```
$ npm install && npm start
```

After running this, open the browser using the url **http://localhost:4200/**, a login page must be displayed

Firebase Functions

Step 1: Move to **functions** folder and execute

```
$ npm install
```

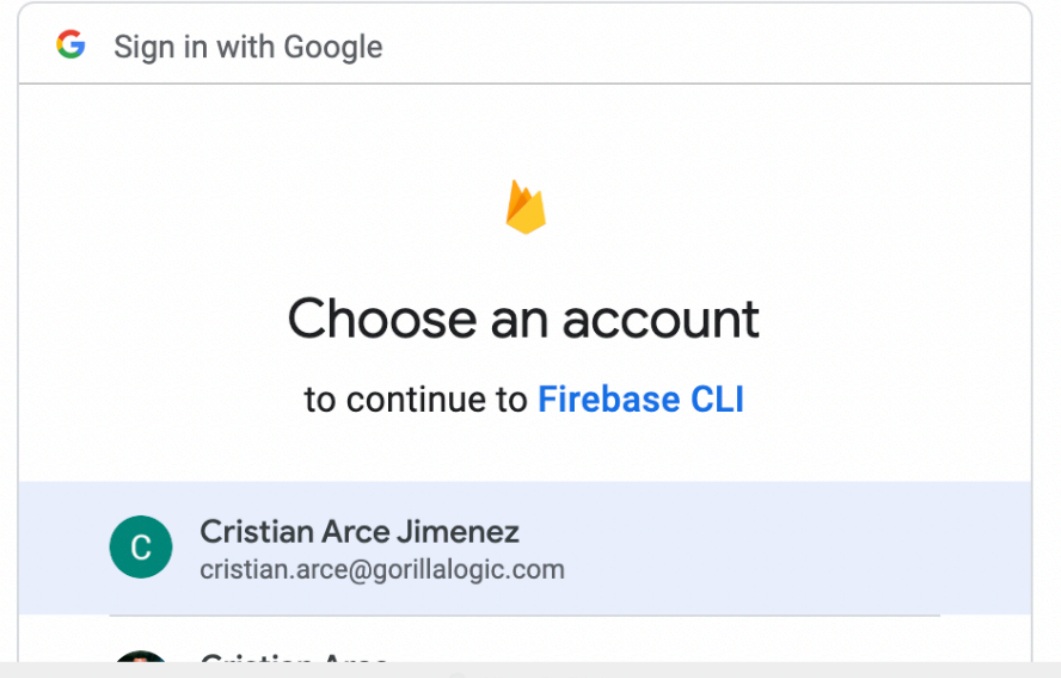
Step 2: It is required to have the latest version of Firebase tools, for install it globally execute

```
$ npm install -g firebase-tools
```

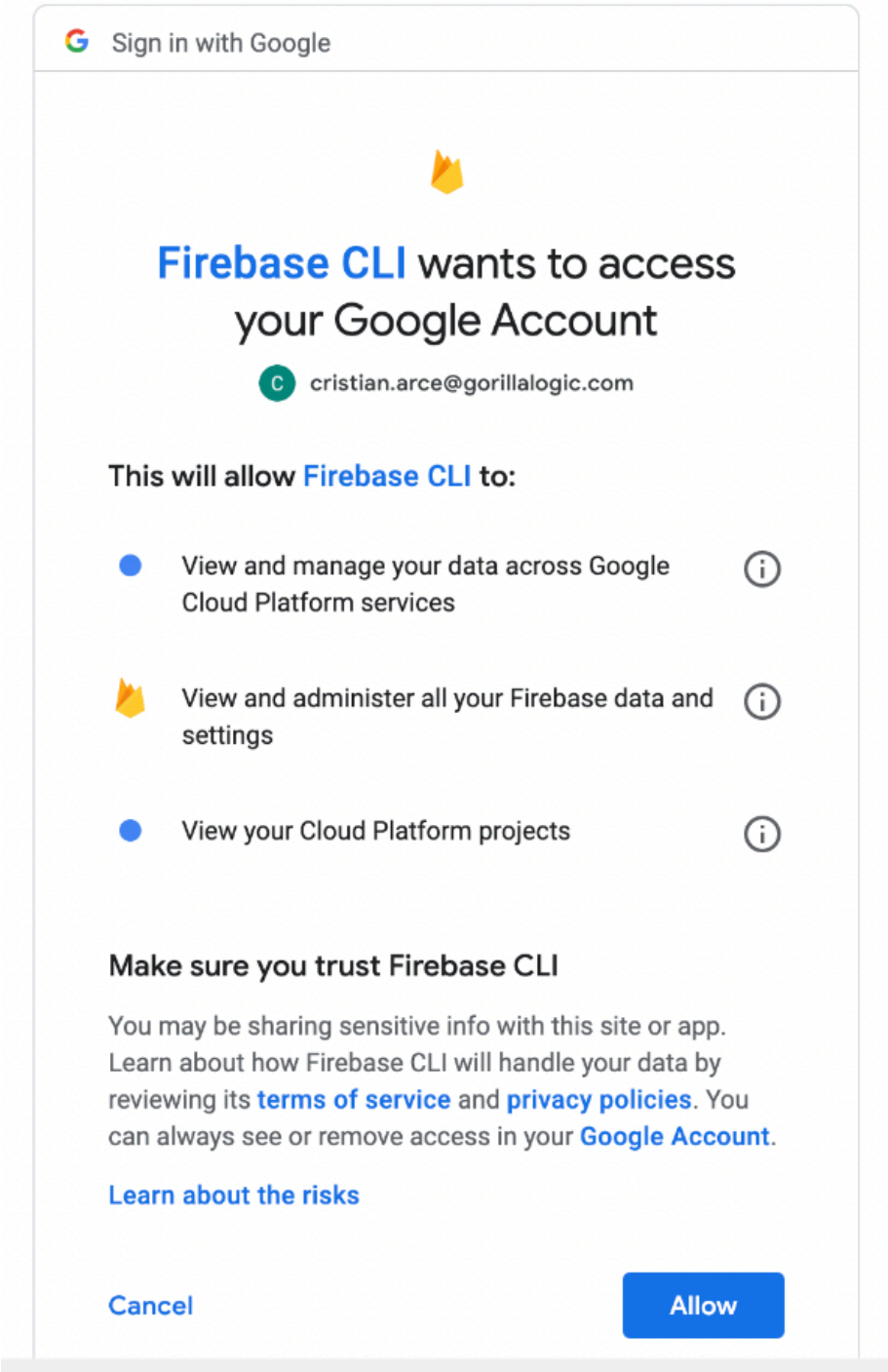
Step 3: It is also required to log into your Firebase profile; for doing this and create a reference in the current project, we execute

```
$ firebase login
```

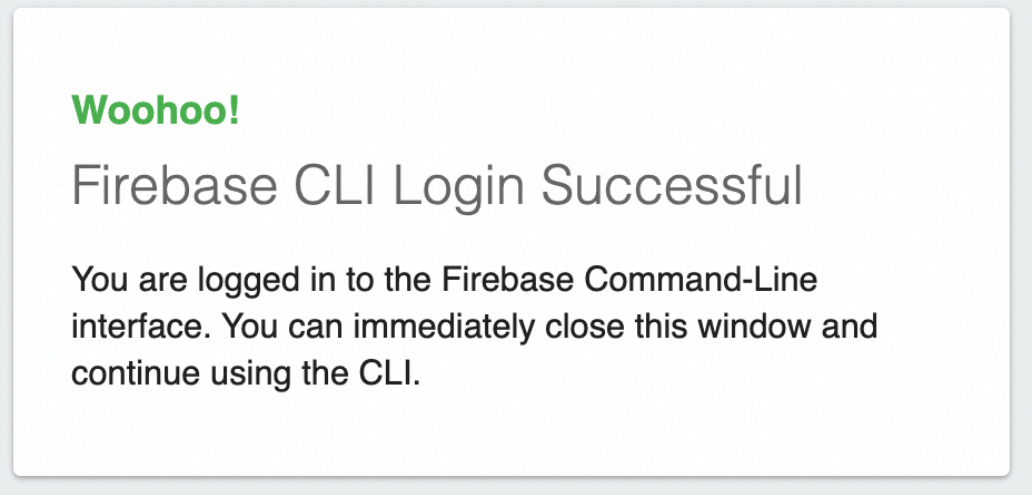
After executing this command, a prompt window will be displayed for choosing a valid **Google Account**



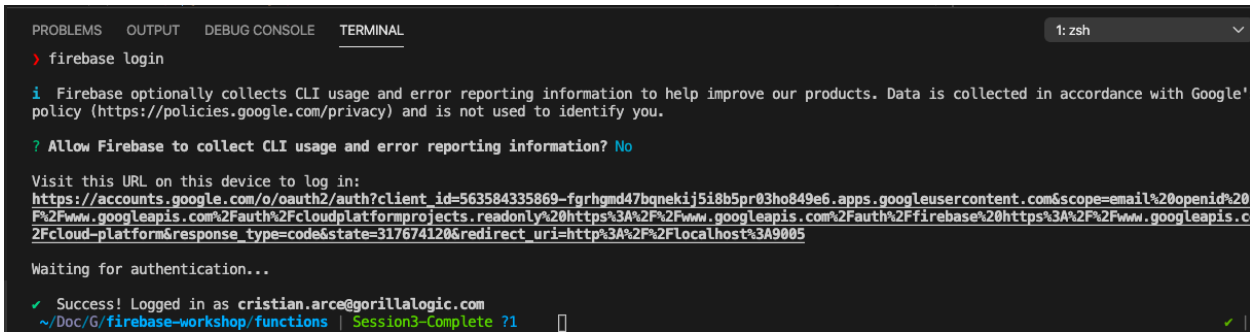
The **Firebase CLI** will show the tools and features it will need to access in the profile



After allowing the access a successful message is displayed



A summary of the process result can be found in terminal



Step 4: Now, we need to initialize **Firebase Cloud Functions**, for that, in the terminal (while being in the new **functions** folder) type

```
$ firebase init functions
```

Note: Select in the 1st question = “**Typescript**”, in other questions select “**No**”

Step 5: In functions/src/index.ts, the content is override using this code below

```
import * as admin from 'firebase-admin';
import * as functions from 'firebase-functions';
import { UserRecord } from 'firebase-
functions/lib/providers/auth';

admin.initializeApp(functions.config().firebase);

// tslint:disable-next-line: no-implicit-
dependencies
const cors = require('cors')({ origin: true });
const moment = require('moment');
const firestoreInstance = admin.firestore();

exports.getDate = functions.https.onRequest((req,
res) => {
    if (req.method !== 'GET') {
        return res.status(403).send('Forbidden!');
    }
    return cors(req, res, () => {
        let format = req.query.format;
        if (!format) {
            format = req.body.format;
        }
        const formattedDate =
moment().format(format);
        res.status(200).send(formattedDate);
    });
});

exports.createUser =
```

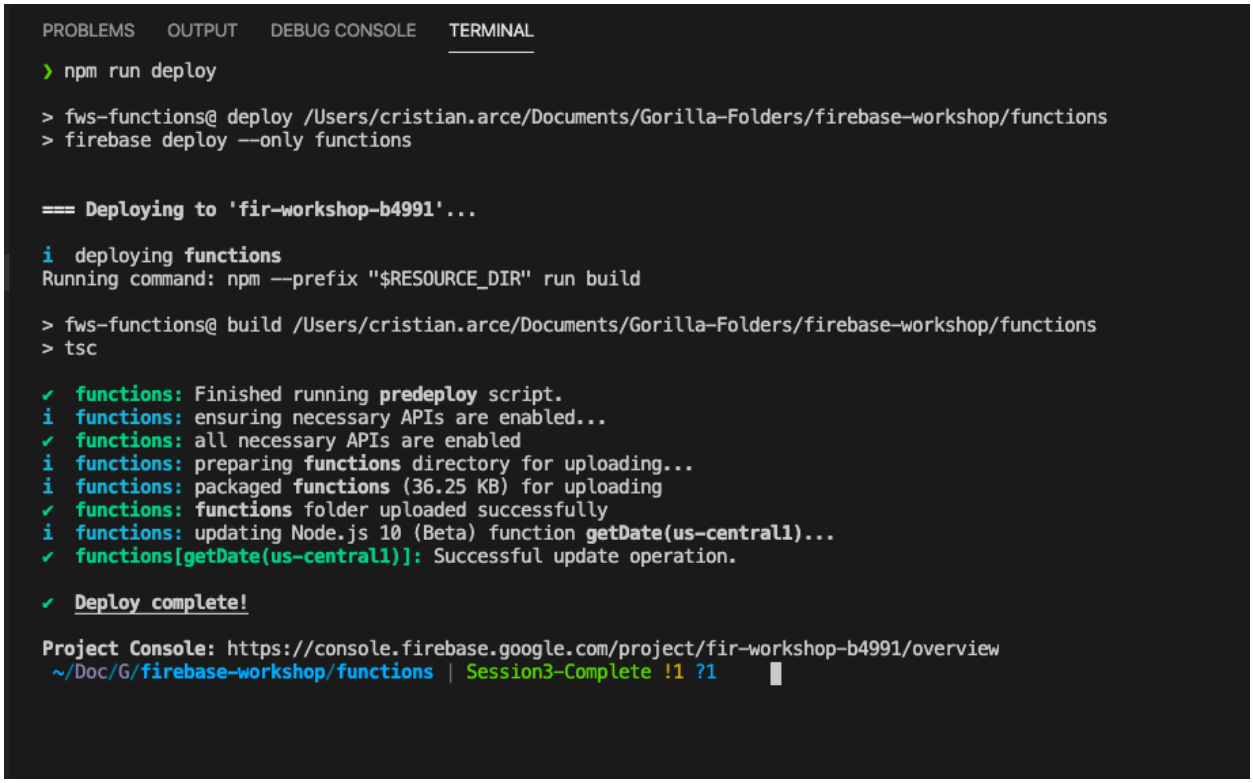
```
functions.auth.user().onCreate((user: UserRecord,
context) => {
    return firestoreInstance
        .collection('user')
        .doc(`${user.uid}`)
        .set({
            id: user.uid,
            uid: user.uid,
            email: user.email,
            address: '**NotDefined**',
            cellphone: '**NotDefined**',
            displayName: '**NotDefined**',
            phoneNumber: '**NotDefined**',
            photoURL: '**NotDefined**',
        });
});

exports.deleteUser =
functions.auth.user().onDelete((user: UserRecord) =>
{
    return firestoreInstance
        .collection('user')
        .doc(`${user.uid}`)
        .delete();
});
```

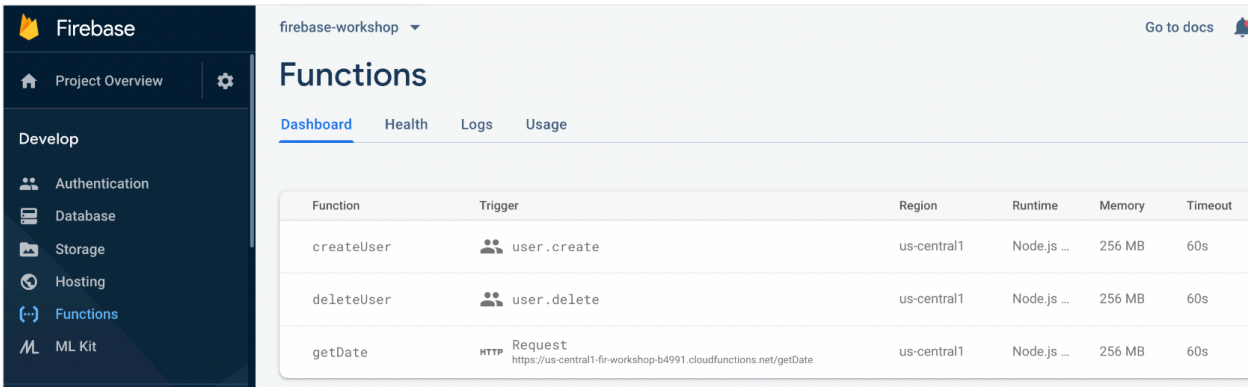
Step 6: It is time to deploy the functions in Firebase, for doing that execute the following command

```
$ npm run deploy
```

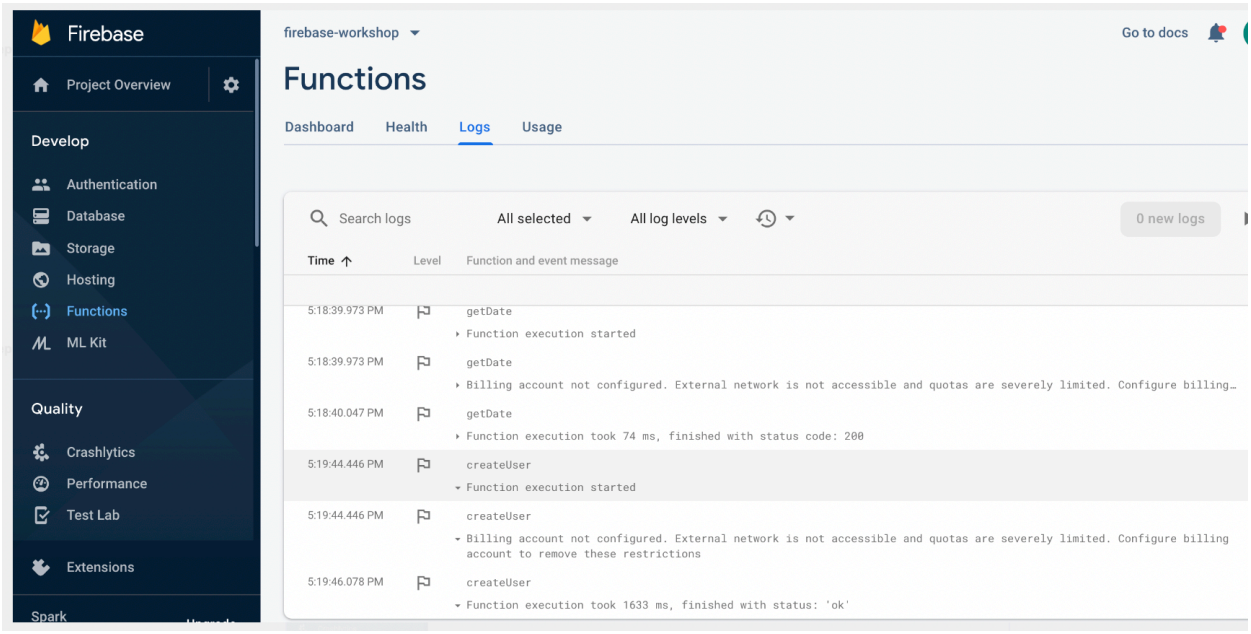
This is the result in terminal



Step 7: After deploying, in the web profile, navigate to **Functions/Dashboard**, there t can be found all the deployed functions for the project



Step 8: Also in **Functions/Logs**, it is possible to track the steps followed in the execution plan of every single function

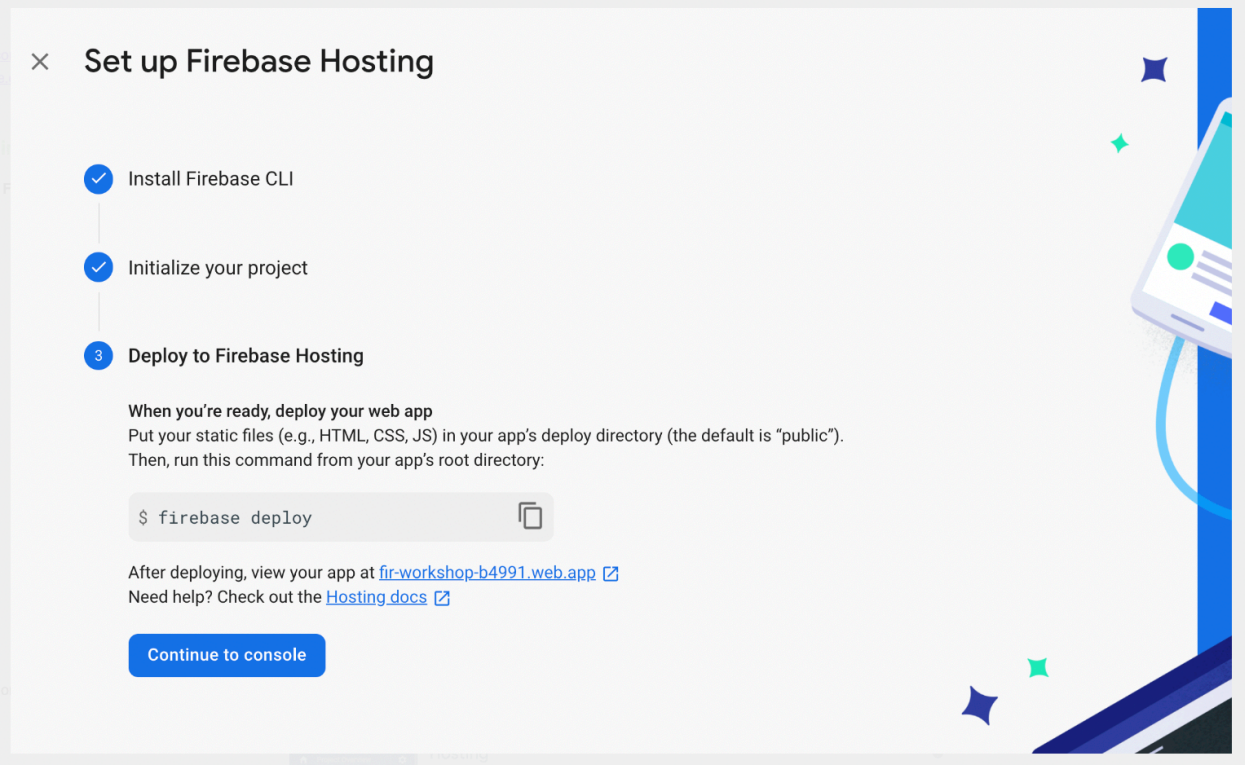


For more information about functions, samples and auth events

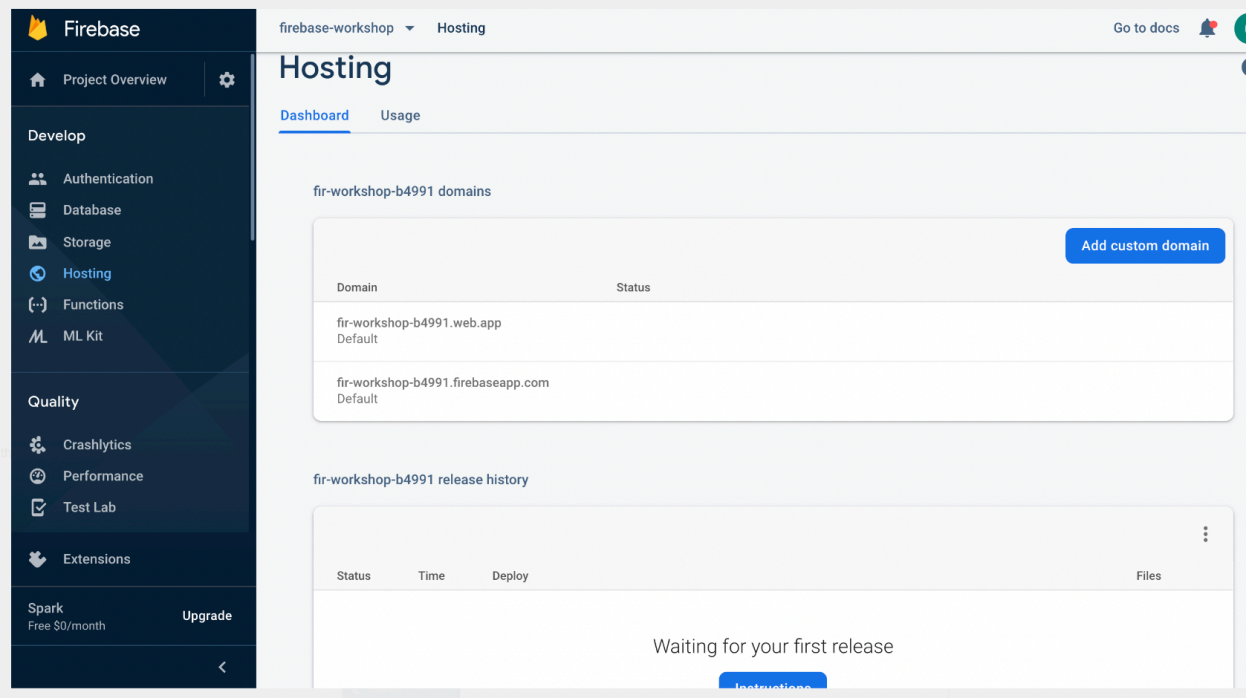
- https://firebase.google.com/docs/functions/use-cases
- https://github.com/firebase/functions-samples
- https://firebase.google.com/docs/functions/auth-events

Firebase Hosting

Step 1: Navigate to **Firebase/Hosting** option, and start the set up using the default values



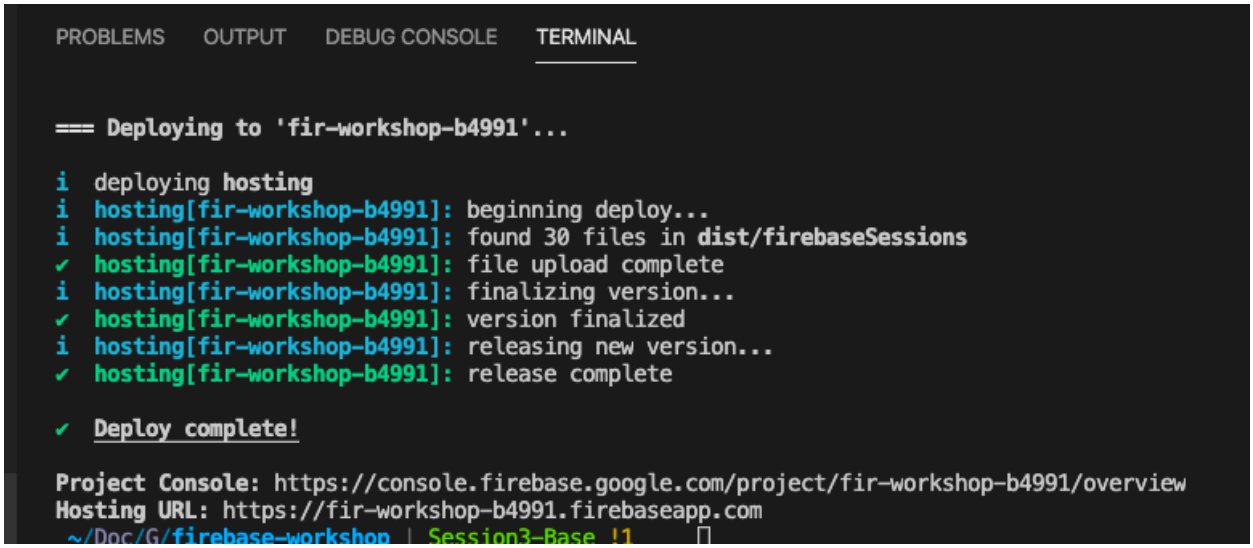
When the setup is done, the **Hosting** is ready to get deploys



Step 2: Move to **root** folder and execute

```
$ firebase deploy --only hosting:target-name
```

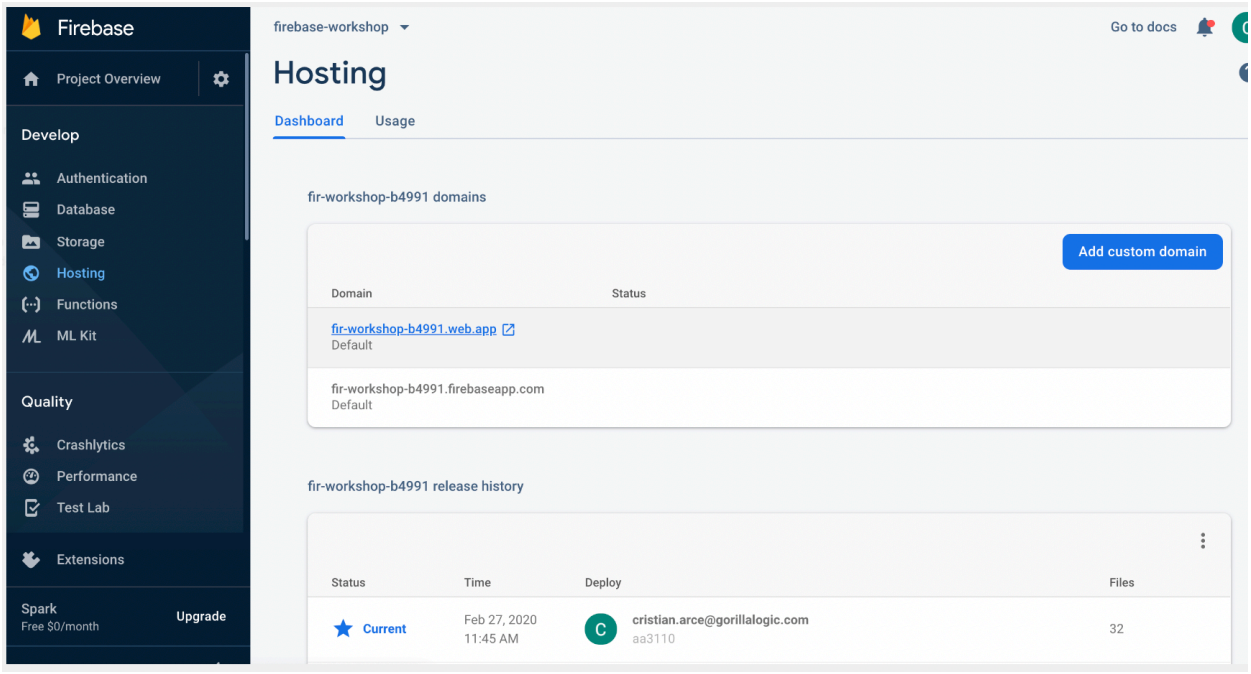
Note: the target-name is the unique alphanumeric identifier of the project, it can be found in **Firebase/Hosting/Dashboard** next to domain



PS: in case of wanting to avoid possible reference error, there is a npm script for installing dependencies, build the app and deploy to Firebase host

```
$ npm run deploy
```

Step 3: After a succeeded deploy, in **Firebase/Hosting/Dashboard** the hosting already give access to a couple domains for public access, there is even an option for adding a custom domain, for this workshop purposes the default ones are good to go


Firebase
Project Overview
Develop
Authentication
Database
Storage
Hosting
Functions
ML Kit
Quality
Crashlytics
Performance
Test Lab
Extensions
Spark
Free $0/month
Upgrade
firebase-workshop
Go to docs
Hosting
Dashboard
Usage
fir-workshop-b4991 domains
Add custom domain
Domain
Status
fir-workshop-b4991.web.app
Default
fir-workshop-b4991.firebaseapp.com
Default
fir-workshop-b4991 release history
Status
Time
Deploy
Files
Current
Feb 27, 2020
11:45 AM
C
cristian.arce@gorillalogic.com
aa3110
32