Deploy an Express API on Firebase hosting

Todd H. Albert, Ph.D.

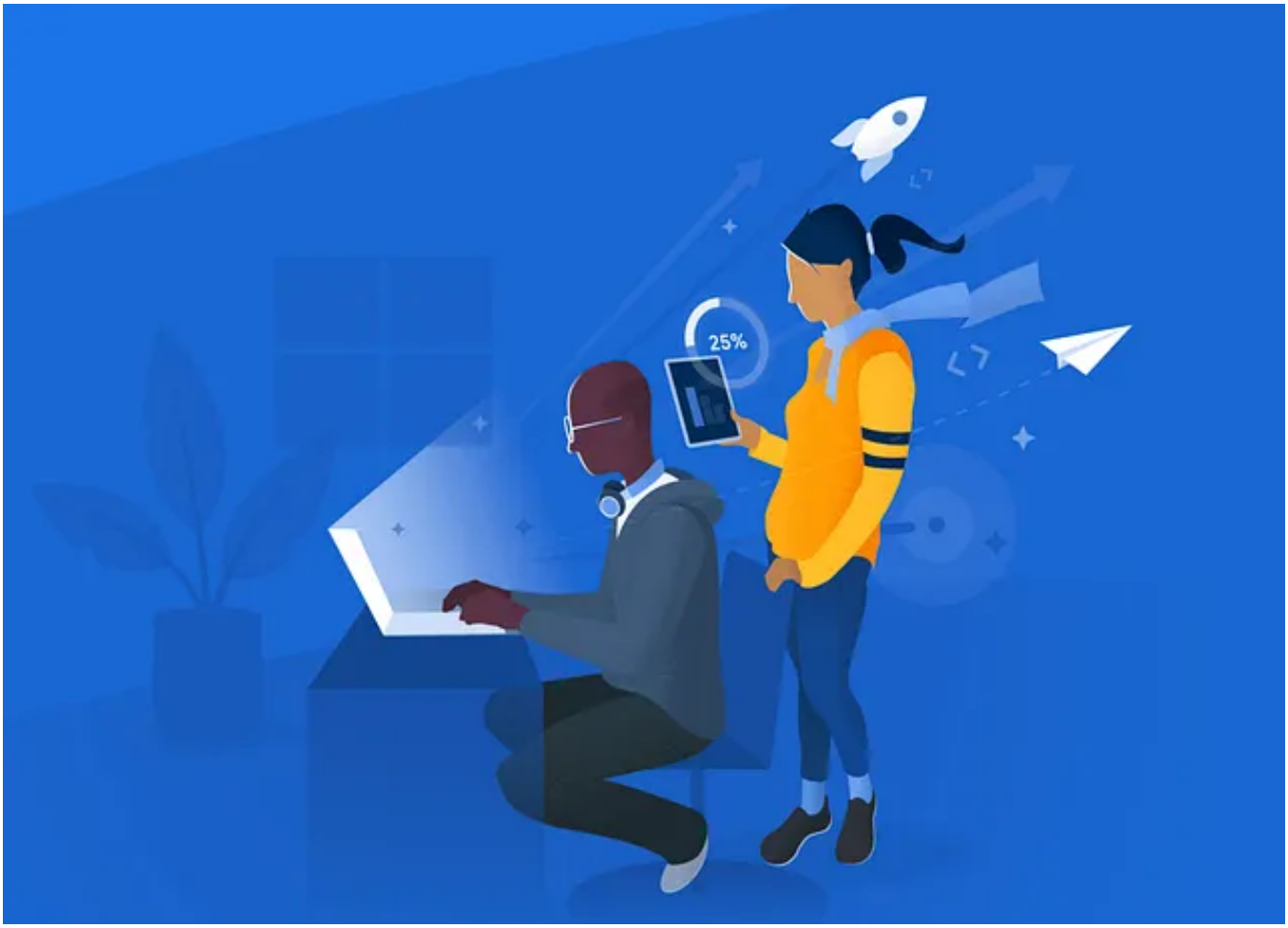
This article is outdated and based on ES5. For an updated version using ES6, checkout out this great article by Gisselle Pombar.

It's funny that when I mention using Firebase (a Google Cloud Platform product) to host my React apps, APIs, and databases, some developers or tech professionals scoff. But I challenge any developer or team to deploy a project to a faster, more stable, and more scalable platform.

In every test, Firebase beats every other cloud platform in speed and cost and it will scale to the size of Google.

And it's relatively easy to use. Goodbye Heroku. Goodbye AWS.

Hello Firebase.



Jonathan Sanchez already published an article on how to deploy your React App to Firebase hosting. This article provides a step-by-step process to create and deploy an Express API on Firebase.

The basic process is that we will use Firebase cloud functions to create a single function, app, which will run our Express API. For all routes, we will setup Firebase hosting to route all requests to this single function. So any requests made on any routes will be directed to our Express API.

First, let's create a Firebase project. We can do this from command line, but let's do it from the Firebase console.

From the console, click **Add Project**. You'll need to create a unique name. I'm using "deploy-api-fb". Be sure to use only lower-case numbers,

dashes, and numbers. Click **Continue**. Disable Analytics and click **Create Project**.

Once the project is created, click **Continue**. We are going to be leveraging Cloud Functions, which Firebase allows us to invoke 2M times per month for Free, however, they do not allow us to deploy cloud functions on a free account.

So, click the **Spark Plan** pill button next to your project name. Under Blaze, select **Select Plan**, connect a billing account, and **Purchase**. Follow the prompts and **Set a budget alert**.

Now, we can do all the rest from command line and our IDE (e.g. VS Code). If you haven't already installed Firebase Command Line Tools, you can do so with

npm i -g firebase-tools

Once installed, run

firebase login

which should open a browser and have you authenticate with Google and grant permissions to your command line to control your Firebase projects.

Let's create a folder for our new API.

mkdir deploy-api

cd deploy-api

Inside this new folder, initialize Firebase hosting

firebase init hosting

Choose **Use existing project** since we created the project in steps 1–2 above.
Find your project on the list using the arrow keys and hit **Enter** to accept.
You can then hit **Enter** three or four times to accept the defaults.

Now initialize Firebase functions
firebase init functions
Hit **Enter** several times to accept all defaults.

We now have a **public** folder for hosting and a **functions** folder for our API. Inside functions, we have our package.json, our node_modules, and our index.js, where we will create our Express API. Go to the functions folder and install Express and CORS. Then go back to the project folder.
cd functions
npm install express cors
cd ..
Launch VS Code and open this project.
code .

Let’s edit functions/index.js
Leave the first line. Import Express. Import CORS. Create the Express App. Use CORS. Create a sample route.

Finally, export the cloud function (we’ll call it “app”) and use firebase-functions library to point https requests to our express app
exports.app = functions.https.onRequest(app);

Now we need to setup hosting to direct all requests to this cloud function.

In the firebase.json file, add the following under hosting:

“rewrites”: [{
“source”: “”,**
“function”: “app”
}],

This tells hosting to redirect all requests (** is wildcard for “all”) to the function called app.

We can test our api locally by running

firebase emulators:start

We can deploy to firebase using

firebase deploy

And that’s it. You’ll get a hosting URL like https://deploy-api-fb.web.app – if you visit https://deploy-api-fb.web.app/test or whatever route you setup, you should see the results.

goto http://127.0.0.1:5001/lcys2025-e3078/us-central1/app/test