**Web Application deployment on Azure PaaS**

Web Application Deployment on Azure uses the following technologies and tools:

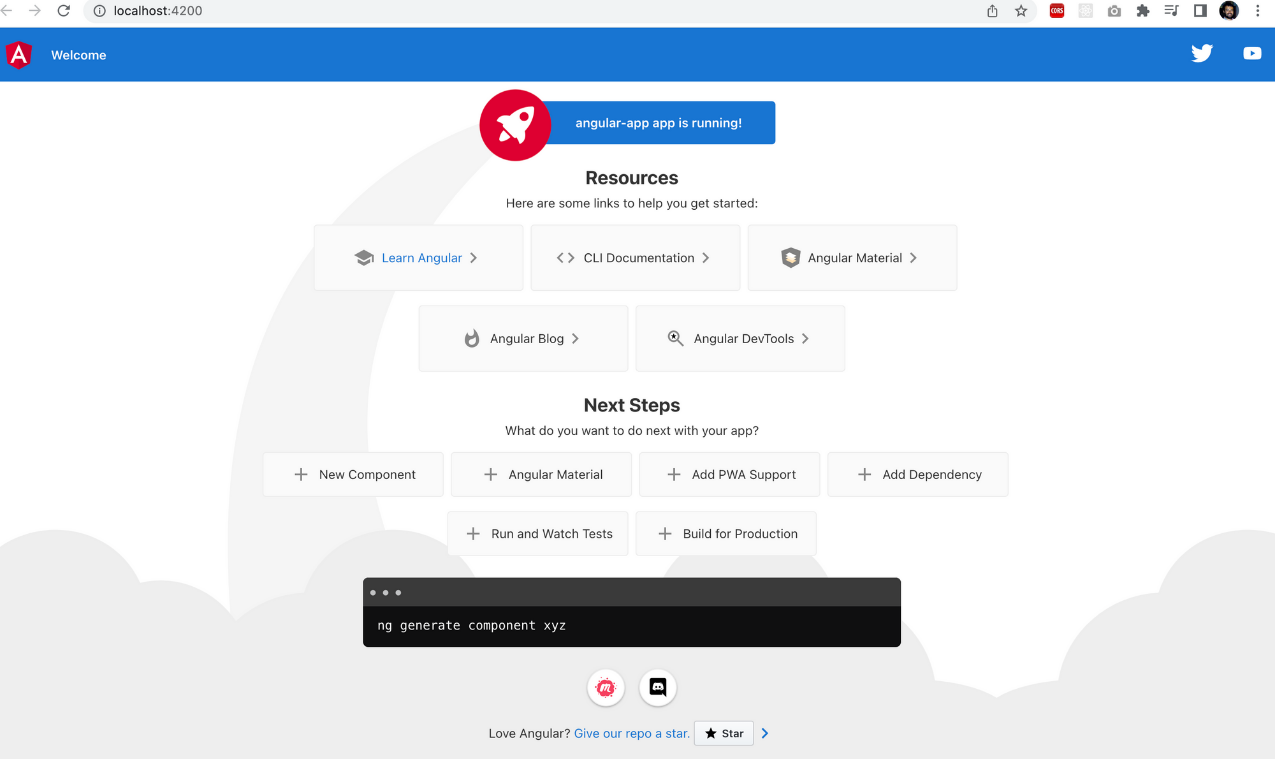
* A GitHub account
* Node.js
* Npm (Node.js package manager. Included with Node installation.)
* Angular CLI (a command-line interface tool to create, develop and maintain your Angular app.)
* Azure account

## STEP 1: Create the Angular App Locally

## create your new Angular app called “angular\_app”:

* ng new angular\_app
* ng serve —-open

The option “--open” tells Angular CLI that you want to open the application with your default browser. This means that you don’t need to copy and paste the URL into your browser’s address bar



**Step 2 – Push the Angular App to Your GitHub Repository**

You should have created a new repository on your GitHub account dedicated to this project.

Once the repository is ready, push your Angular app from your machine to your repository. If you don’t add a README file when creating your repo, you should see the instructions on how to push your code directly on your repository page.

git init

git add .

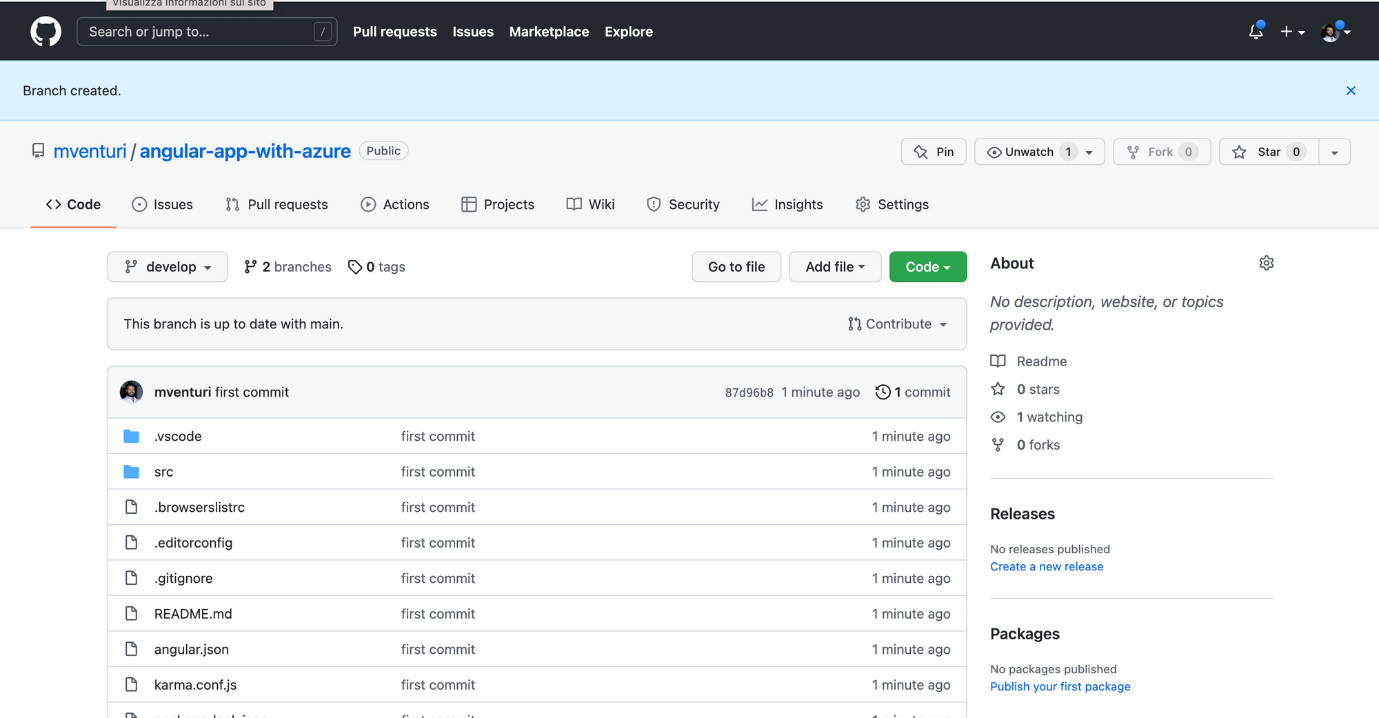
git commit -m "first commit"

git branch -M main

git remote add origin https://github.com/<REPO>.git

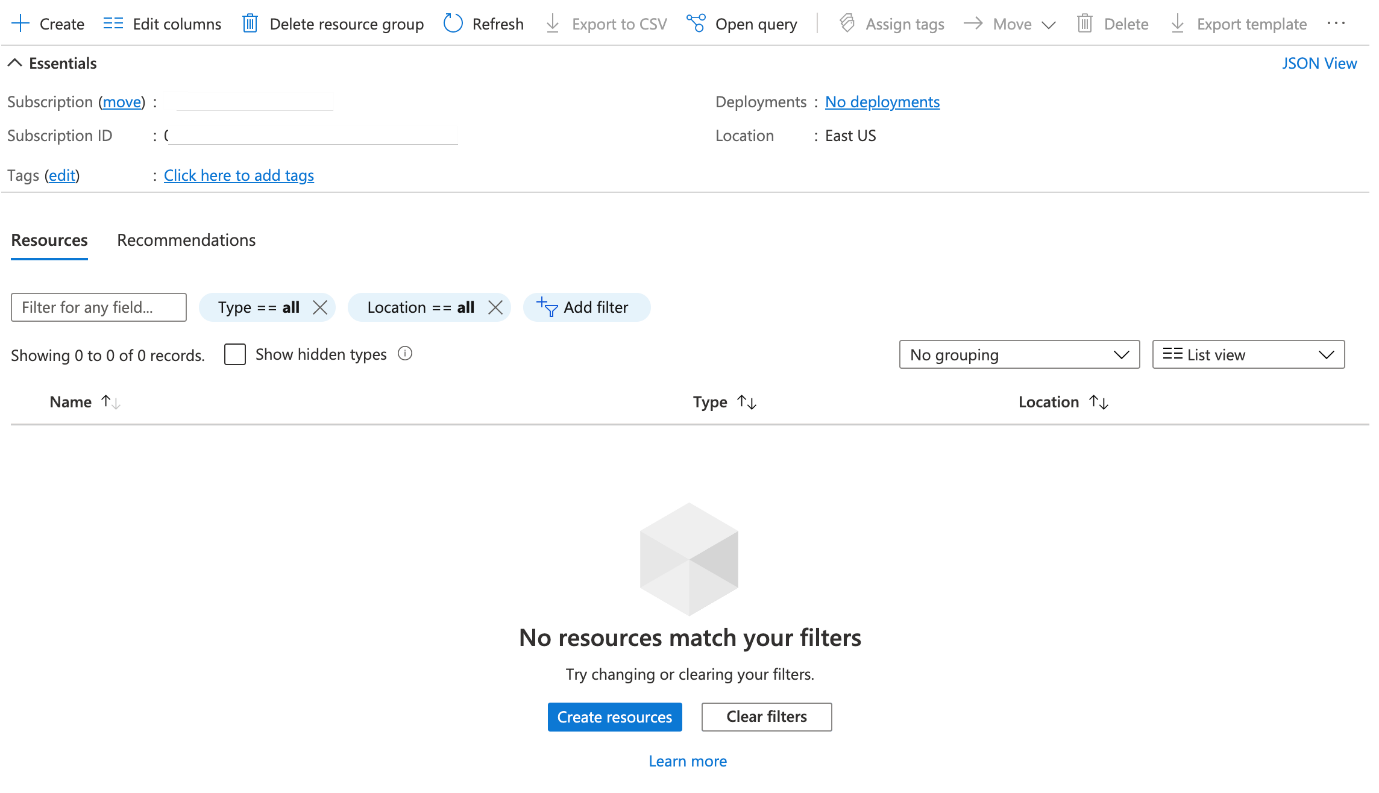
git push -u origin main

Once you run these commands successfully, you should see something like this:



**Step 3 – Time to Switch to Azure**

Now you'll create a brand new Azure Static Web App and connect it to your repository.

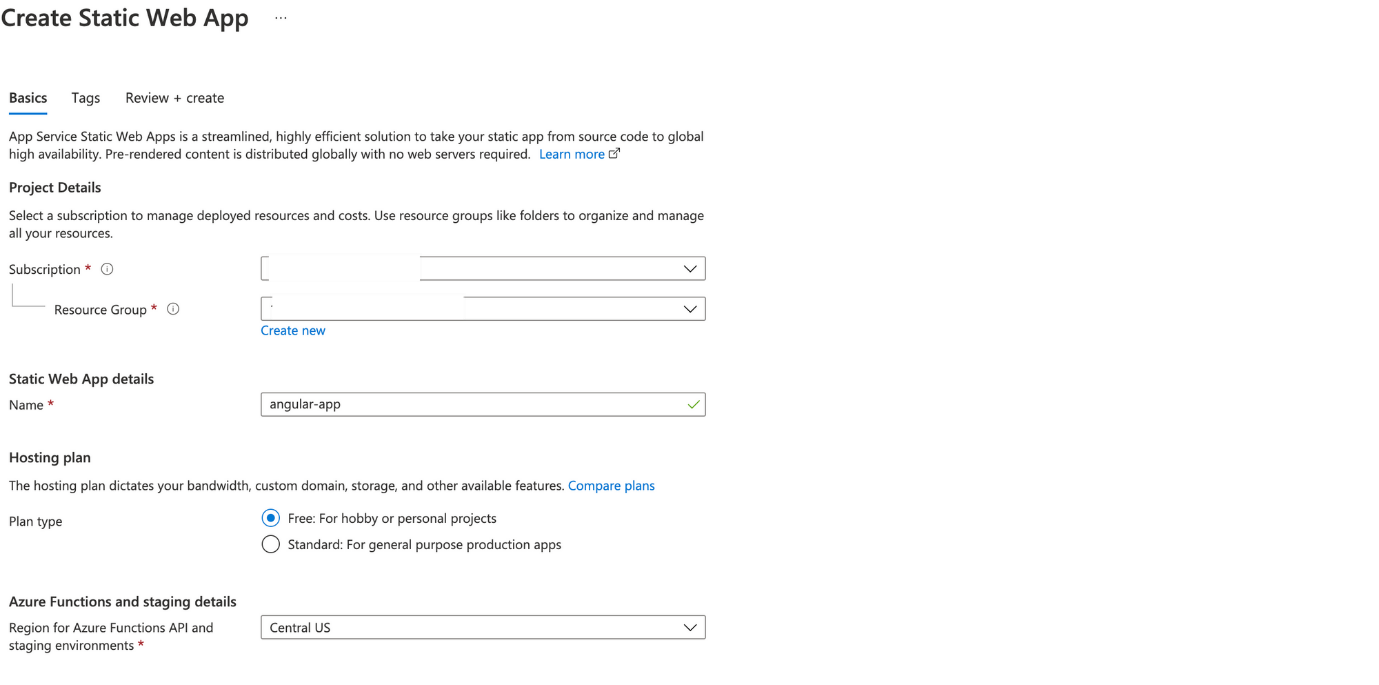


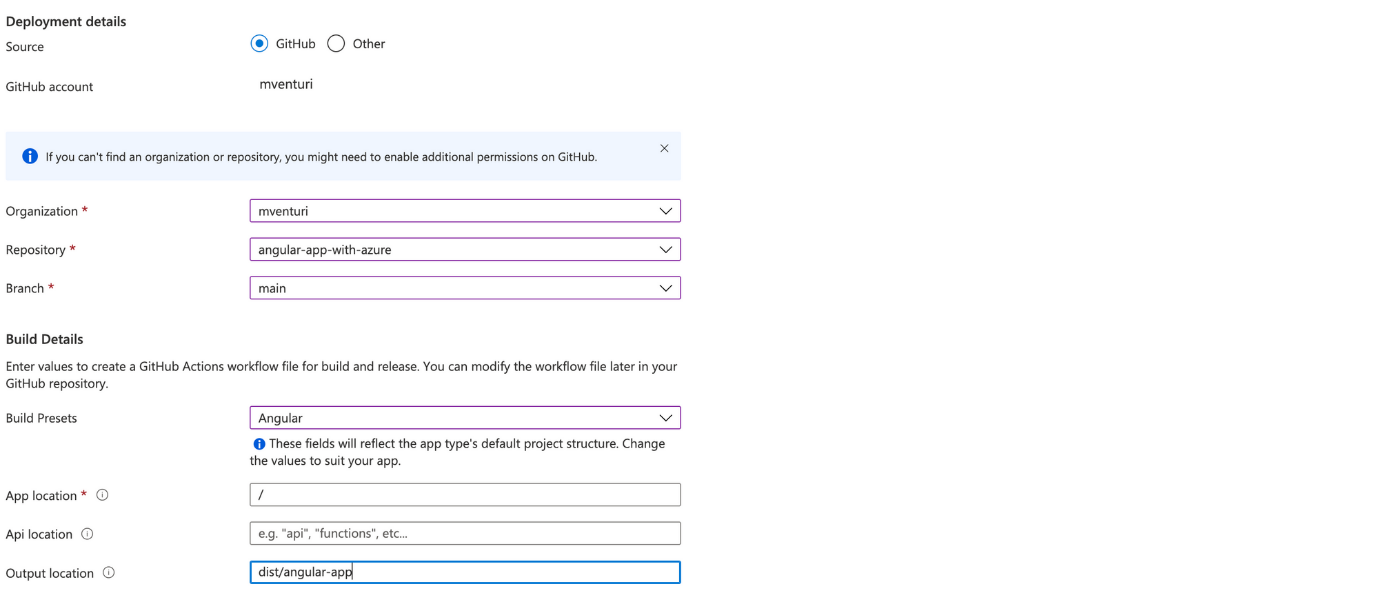
Hit “create” and you'll see the list of resources available on Azure. Filter resources by “Static Web App” and select it.



Then move to Azure’s wizard to create the resource. Let’s see how you should fill it out.

First of all, choose the subscription and resource group. Then go with the Static Web App details: choose the name – “angular-app” – and stick with the Free plan. Then select your region – mine is “Central US”.





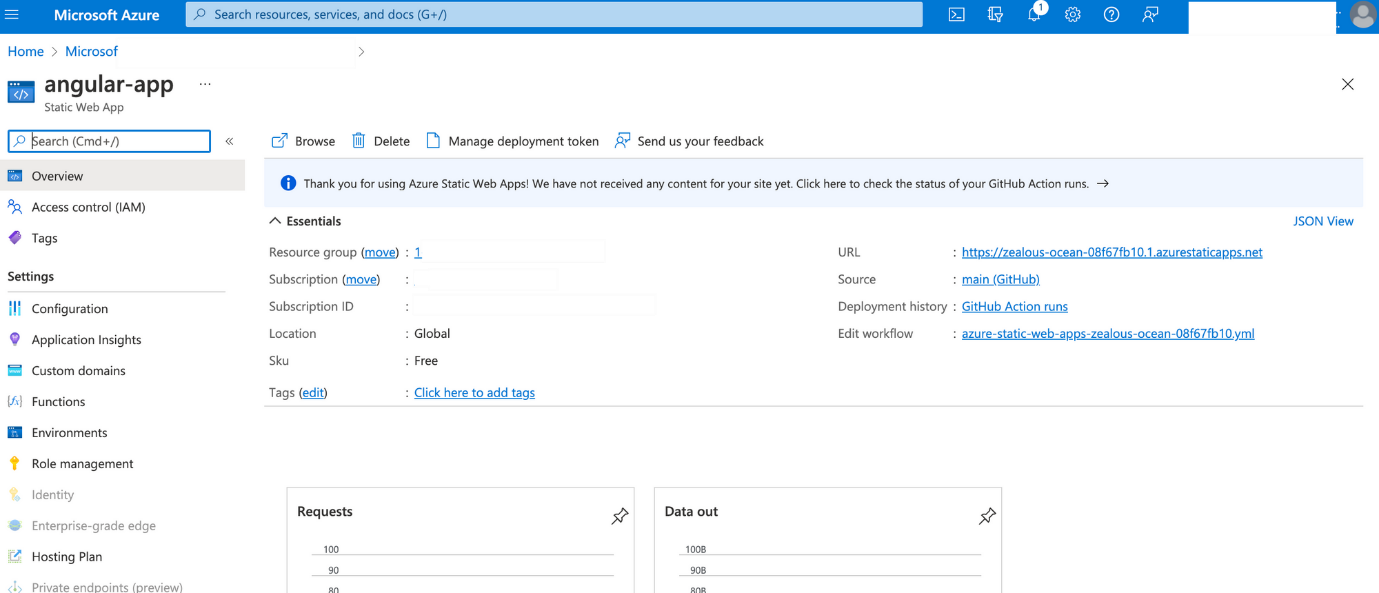
You can synch your GitHub account in the “Deployment Details” section, and then specify the Organization, Repository, and branch you'll get your codebase from.

As I said at the beginning of this tutorial, I choose the “main” branch as the production one.

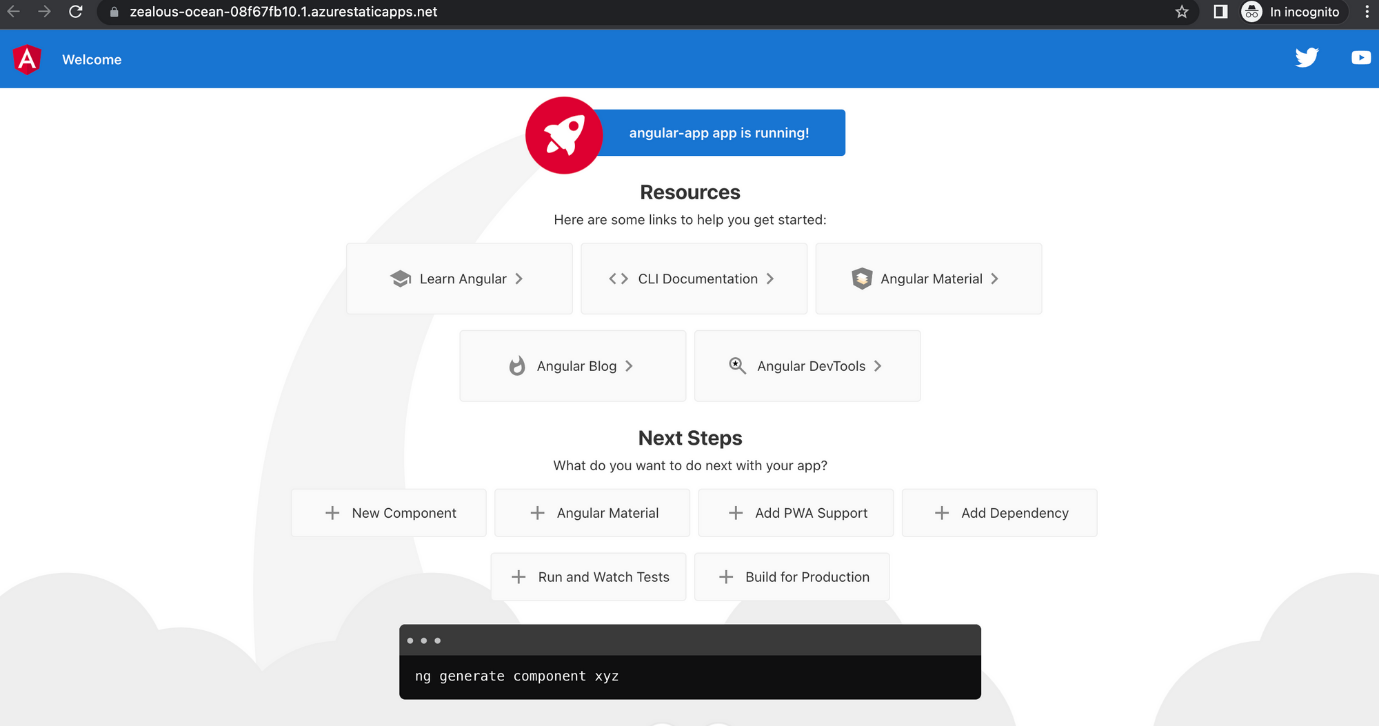
Now let’s focus on the “Build Details”. You'll choose “Angular” as “Build Presets”, and specify that the App is located at the root directory on “App Location”.

Then you'll type the path of the output location (“dist” directory plus name-of-project directory. In my case, it's “dist/angular-app”. This is where Angular CLI locates the build of your project. I spent a lot of time looking for this info and I think it’s good to share it with you).

So, now you're ready to create your Static Web App. Hit “create” and see what happens. This is the overview of my app:



Next, hit the “URL” link and this is what you should see:



**Without using git**

1. For Deploying Angular we need to create an App Service in azure.
2. In Azure the App service is created using the technology stack as used for creating the Angular (Node Version).
3. After Opening Visual Studio Code, the base URLs are changed as per the API URL in Azure.
4. In terminal type “ng build configuration=production”. A “dist” folder will get created.
5. Next Deploy the “dist” folder in Azure App service created for Angular.