Deploying to Azure

This guide walks through how to deploy and host your next Gatsby site on Azure.

Azure is a great option for deploying Gatsby sites. Azure is a large cloud platform with hundreds of services working together to give you serverless, databases, AI, and static website hosting. The Azure Static Web Apps service is meant to be used with static websites. It provides features like hosting, CDN, authentication/authorization, continuous deployment with Git-triggered builds, HTTPS, the ability to add a serverless API, and much more.

Prerequisites

All you need is a Gatsby site living in a GitHub repository!

Git setup

Azure Static Web Apps service currently supports GitHub. In order to use the Azure service, you'll need to version your Gatsby project with Git and push it to GitHub. If you already have a Gatsby site in a GitHub repository, you can skip the steps below.

Create a local Git repository

1. Initialize a Git repository, place yourself in the root directory of your Gatsby app and type `git init`.

2. Next, create a file called `.gitignore` in the root of your project and give it the following content:

   ```
   node_modules
   build
   ```

   The above configuration will prevent the `build` and `node_modules` directories from being added to your repository. The `build` directory changes every time you build. The `node_modules` directory is only needed at build time and can be quite large because of all the libraries it contains.

3. Finally, add the change and commit it.

```
git add .
git commit -m "adding Gatsby project"
```

Create a GitHub repository and push your code

1. Go to GitHub and log in. You should now be at a URL like
   `https://github.com/<your username>?tab=repositories`.

2. Click the `new` button.

3. Name your repository and click `Create repository`.

4. Finally, add your GitHub repository as a remote and push. Type the following commands to accomplish that (replacing `<user>` with your GitHub username):

   ```
   git remote add origin https://github.com/<user>/gatsby-app.git
   git push -u origin master
   ```

Now you are ready to deploy your site!

Deploy to Azure Static Web Apps

1. Sign in to the Azure portal.
2. In the top bar, search for **Static Web Apps**.
3. Select **Static Web Apps**.
4. Click **+ Add** at the top left.

Configuring settings

Next, configure your new app and link it to your GitHub repository.

1. Enter the **Project Details**.

Setting	Value
Subscription	**Your chosen subscription**
Resource Group	*A pre-existing or new resource group*

2. Enter the **Static Web Apps details**.

Setting	Value
Name	Name your app. Valid characters are `a-z` (case insensitive), `0-9`, and `_`.
Region	Select Region closest to you
SKU	**Free**

3. Click the **Sign-in with GitHub** button and authenticate with GitHub.

4. Enter the **Source Control Details**.

Setting	Value
Organization	Select the Organization where you created the repository
Repository	**gatsby-app**
Branch	**master**

5. Click the **Next: Build >** button to edit the build configuration.

Building your site

Next, add configuration details specific to your Gatsby project.

The `App location` is the location of your front end app code. If you have a frontend app and an API in different directories but in the same repository, you will need to point out under which subdirectory your front end app lives. You can leave this blank if you have a Gatsby project at the root.

The default value for `Api location` is `/api`. As we don't have an API in an `/api` directory we can leave this blank and the Static Web App Service will ignore it during build. If you wish you can add an API at a later stage. Refer to the links at the bottom of this doc that describes how to do that.

The really important field to specify is the `App artifact location`, this should point to the folder where your Gatsby project is built, this would be `public` directory.

Setting	Value
App location	/
Api location	**api** or *Leave blank*
App artifact location	**public**

Click the **Review + create** button.

Continue to create the application.

1. Click the **Create** button

2. Once the deployment is complete, click the **Go to resource** button

Review the GitHub Action

At this stage, your Static Web Apps instance is created in Azure but your app is not yet deployed. The GitHub Action that Azure creates in your repository will run automatically to perform the first build and deployment of your app, but it takes a couple of minutes to finish.

You can check the status of your build and deploy action by clicking the blue information banner at the top of your main window.

View your website

Once your GitHub Action finishes building and publishing your web app, you can browse to see your running app.

Click on the *URL* link in the Azure portal to visit your app in the browser.

Congratulations! You've deployed your first app to Azure Static Web Apps!

> Don't worry if you see a web page that says the app hasn't been built and deployed yet. Try refreshing the browser in a minute. The GitHub Action runs automatically when the Azure Static Web Apps is created. So if you see the splash page, the app is still being deployed.

Additional resources

There's much more to learn about Azure Static Web Apps such as working with routes, setting up custom domains, adding a Serverless API, and much more. Below are some useful links:

- Docs: Azure Static Web Apps, overview page
- Docs: Azure Static Web Apps, add Serverless API
- Docs: Azure Static Web Apps, setup Custom domain
- LEARN module: Gatsby and Azure Static Web Apps
- LEARN module: SPA applications + Serverless API and Azure Static Web Apps
- Docs: Azure Static Web Apps, Routing
- Docs: Azure Static Web Apps, Authentication & Authorization
- Quickstart: Azure Static Web Apps + Gatsby