

AWS Amplify Gen 2 Deployment Guide

1. Introduction

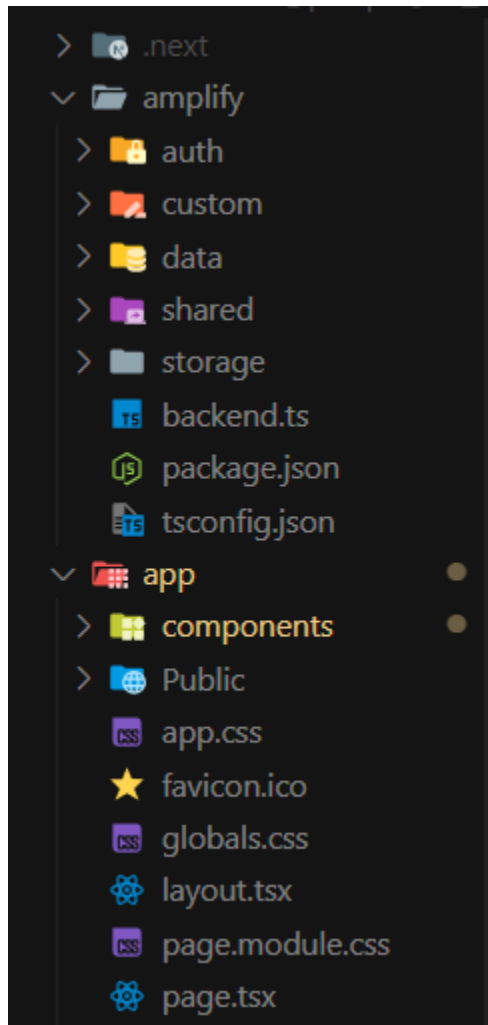
- **Objective:** This guide provides step-by-step instructions to deploy and manage your application using AWS Amplify Gen 2.
- **Overview:** The application is a fullstack TypeScript project utilizing AWS Amplify Gen 2 for backend services with framework NextJs, including authentication by cognito, data storage, and hosting.

2. Prerequisites

- **AWS Account:** Ensure you have an active AWS account with appropriate permissions.
- **GitHub Account:** Required for repository hosting for the frontend code and integration with Amplify.
- **Development Tools:**
 - Node.js (v18 or later)
 - Npm to manage packages
 - Git
 - AWS CLI (configured with your AWS credentials)

3. Project Structure

Your project should have the following structure:



4. Deployment Steps

4.1 Initialize Amplify in Your Project

Run the following command to set up Amplify Gen 2 in your project:

```
npm create amplify@latest
```

This command initializes the `amplify/` directory with necessary backend configurations. [Amplify Documentation](#) and also don't forget the output json file in the backend resources in the amplify service of the app itself.

4.2 Configure Backend Services

- **Authentication:** Define authentication strategies (e.g., email/password, social logins) in `amplify/auth/resource.ts`.
- **Data Models:** Define your data schema using TypeScript in `amplify/data/resource.ts`.
- **Custom changes** like s3 integrations in the custom folder

4.3 Connect to AWS Amplify Console

1. Navigate to the [AWS Amplify Console](#).
2. Click on **"Create App"** and select **"Host web app"**.
3. Choose **GitHub** as your repository provider and authorize Amplify to access your GitHub account.
4. Select your repository and branch (e.g., `main`) to connect.
5. Amplify will detect the build settings automatically. Review and modify if necessary.
6. Click **"Save and Deploy"** to initiate the deployment process.

4.4 Monitor Deployment

Once deployment starts, Amplify will provision backend resources and deploy your frontend. Monitor the progress in the Amplify Console and in case of failure you can monitor the logs for any errors in build and always consider that the amplify build checks more things than the normal amplify like the static files.

5. Post-Deployment

- **Access Application:** After successful deployment, access your application via the provided Amplify domain.
- **Backend Resources:** Manage and monitor backend resources (e.g., authentication, data models) directly from the Amplify Console.
- **Environment Variables:** Set and manage environment variables for different environments (e.g., development, production) within the Amplify Console.
- **Add different branches to deployment:** we created three: one for dev, one for staging and third for production.

6. Updating the Application

- **Code Changes:** Make changes to your frontend or backend code in your local development environment.
- **Commit and Push:** Commit your changes and push them to the connected GitHub branch.
- **Automatic Deployment:** Amplify will automatically detect changes and redeploy the application.