12.03.24

AWS Amplify

Andrew Nguyen
Pavel Braginskiy

Lorem Ipsum

What is AWS Amplify?

- An AWS service for building and deploying full stack applications to the cloud
- What can you build with Amplify? (source)
  - Server-side rendered web applications
  - Single page web apps and static websites
  - Native mobile applications
  - Cross-platform applications
- Interfaces with other AWS services like CloudFront and S3

W

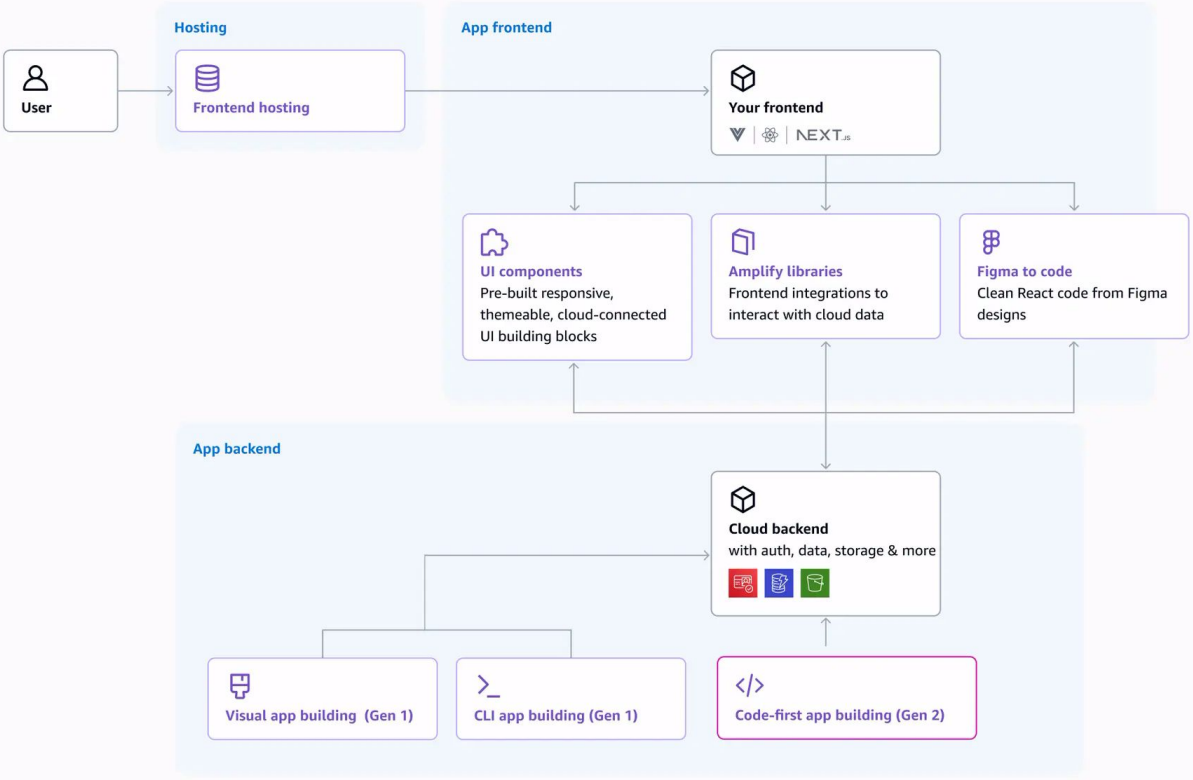


Figure: https://aws.amazon.com/amplify/

History

History of AWS Amplify: Why?

Released in 2017 as an open-source JavaScript library to make it easier to develop cloud-connected mobile and web apps (source).

History of AWS Amplify: Why?

Motivations:

- Server-Side Rendering (SSR)
- Full-stack frameworks such as Next.js, Nuxt that allow developers to handle both front end and back end in a single codebase and deployment.
    - CSR: Client-Side Rendering
    - SSG: Server-Side Generation
    - SEO: Search Engine Optimization

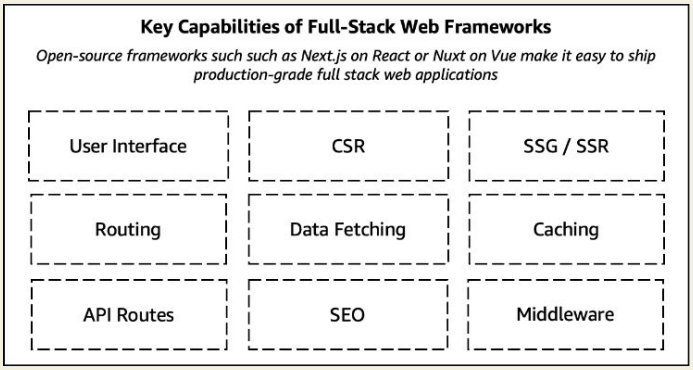


Figure: https://aws.amazon.com/blogs/mobile/evolution-of-full-stack-development-with-aws-amplify/

History of AWS Amplify: How?

- How AWS Amplify has evolved:
  - **11/2017**: Initial release as an open-source JavaScript library
  - **8/2018**: Launch of Amplify CLI
  - **12/2020**: Launch of Amplify Studio, a GUI to build backends
  - **12/2021**: Addition of UI building to Amplify Studio, with Figma-to-React capabilities and form generation.
  - Gen 2 of AWS Amplify introduces infrastructure-as-code tools and AI assistance from Amazon Q Developer
- Driving factors for evolution:
  - Generative AI
  - Faster feedback loop
  - Lower the barrier to entry for full-stack development and hosting
- (Source)

History of AWS Amplify: Who?

- Alternatives:
  - Google Firebase (Platform as a Service)
  - Azure App Service (Platform as a Service)
  - Vercel (Platform as a Service)
  - Heroku (Platform as a Service)
  - SST (Infrastructure as Code)

Features

Features (Front End)

- Supports hosting for popular front end web frameworks
  - React
  - Vue
  - Angular
- Automated scaling backed by AWS Cloudfront cloud delivery network (CDN)
- Automated build and deploy from GitHub
  - Also works with AWS CodeCommit
- Preview deployments
- Monitoring

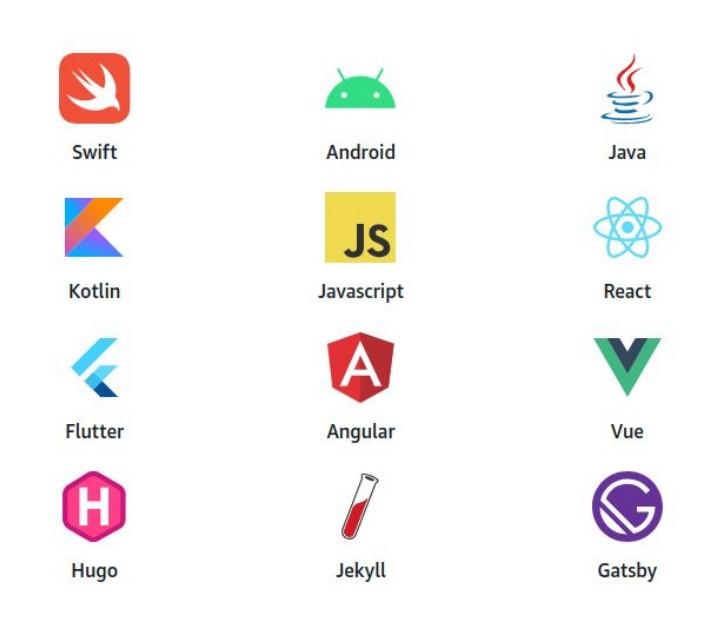


Figure: https://aws.amazon.com/amplify/

Features (Back End)

- Serverless, scales resources as needed (source)
- Under the hood:
  - S3
  - Lambda
  - CloudFront
- Can compose backend with any AWS resource, including DynamoDB, AppSync, etc.
- Simplified authentication (leave this to the experts)
- Low code (build with GUI)

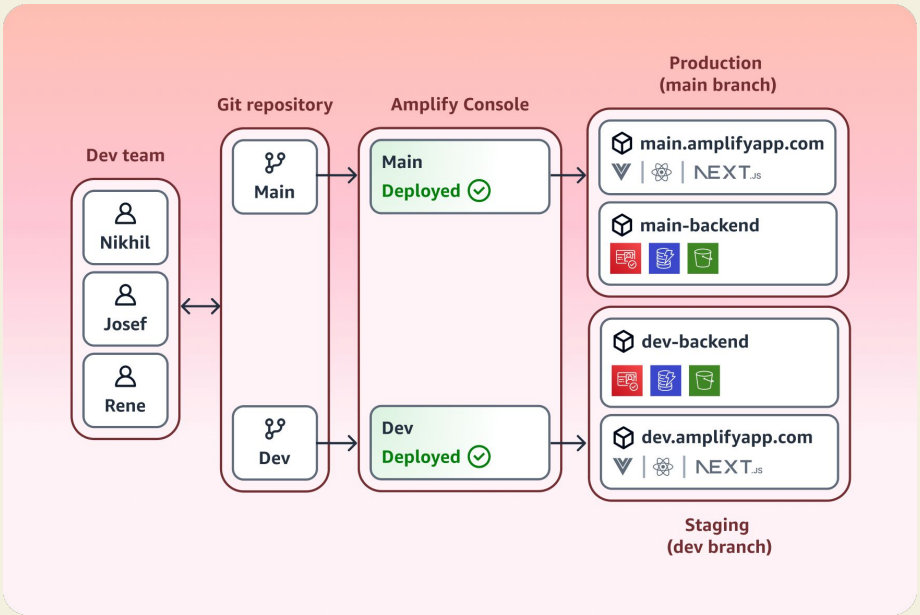


Figure: https://aws.amazon.com/amplify/extensibility/

Use cases

Example Use Case 1: Deploying a Website

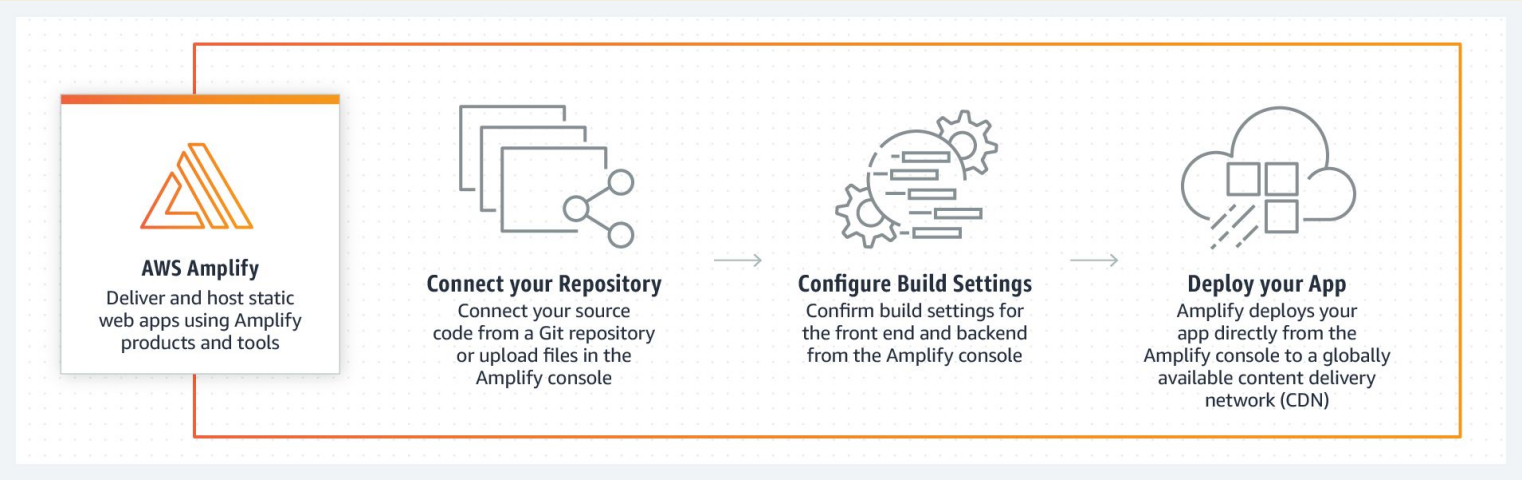


Figure: https://aws.amazon.com/amplify/hosting/

Example Use Case 2: Rapidly Developing Back End Infrastructure

Source: https://aws.amazon.com/amplify/customers/
Case study: https://aws.amazon.com/solutions/case-studies/neimanmarcus-case-study/

Advantages and disadvantages

AWS Amplify Advantages

- Pay as you go pricing
  - Only get charged for the infrastructure you use as opposed to buying into a fixed-price plan or by seat
- Managed solution for cloud infrastructure
  - Don't need to know how to provision and use cloud resources when all you want to do is host a website
- Integrated with AWS
  - Amazon S3 (Storage)
  - CloudFront (Content Delivery Network)
  - Amazon Cognito (Authentication)
  - AWS Lambda (Server-Side Rendering)
- Source

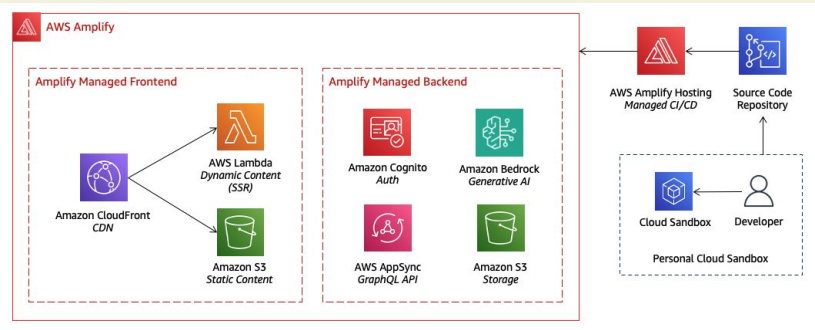
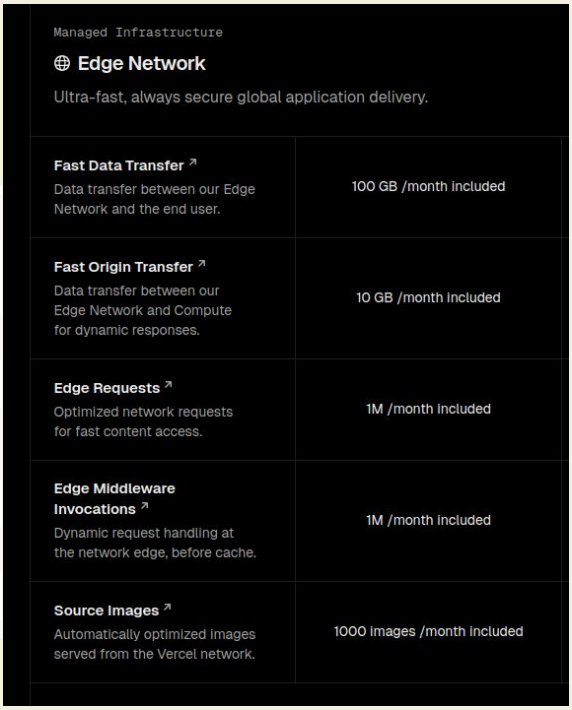


Figure: https://aws.amazon.com/blogs/mobile/evolution-of-full-stack-development-with-aws-amplify/

AWS Amplify Disadvantages

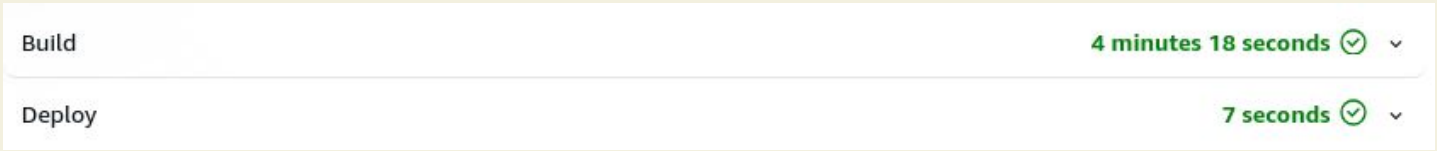
- Uncompetitive free tier vs competing services
  - AWS Amplify free tier (left)
  - Vercel free tier (right)

Deploy an app	**with AWS Free Tier**
	Free for 12 months.
Build and deploy	No cost up to 1,000 build minutes per month
Data storage	No cost up to 5 GB stored on CDN per month
Data transfer out	No cost up to 15 GB per month
Request count (SSR)	No cost up to 500,000 requests per month
Request duration (SSR)	No cost up to 100 GB-hours per month



AWS Amplify Disadvantages

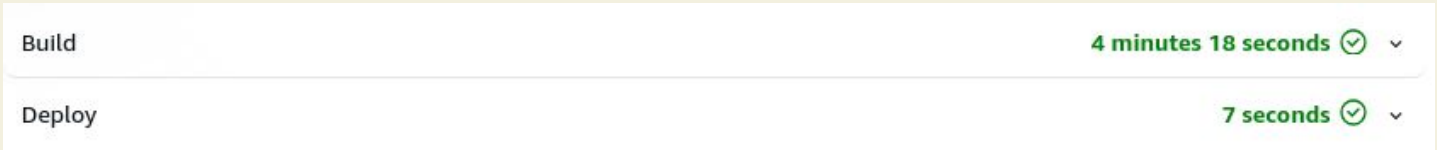
- Slow build times (more on this later)
  - Build and deploy time for demo app as reported by Amplify:



Developer experience

Usability Impressions

- Simple to use, but the build times were quite slow
  - For a continuous iteration workflow this can be very frustrating
  - Build and deploy time for demo app:

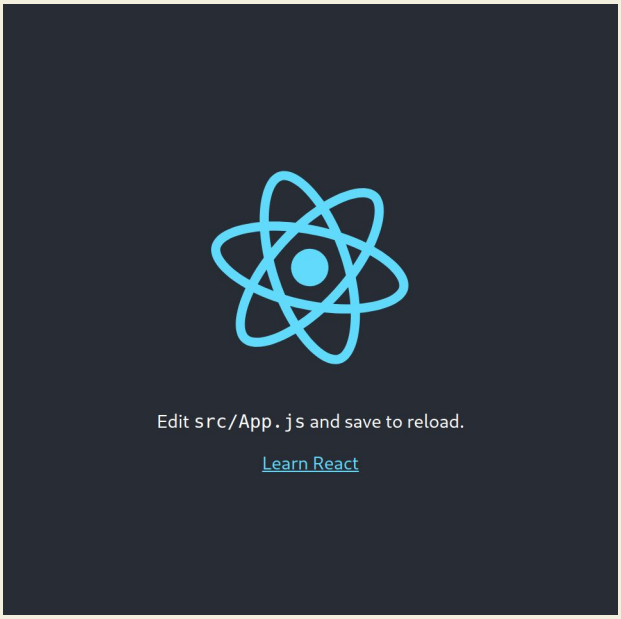


Usability Impressions

Experiment: Build time comparison for create-react-app, a simple React website:

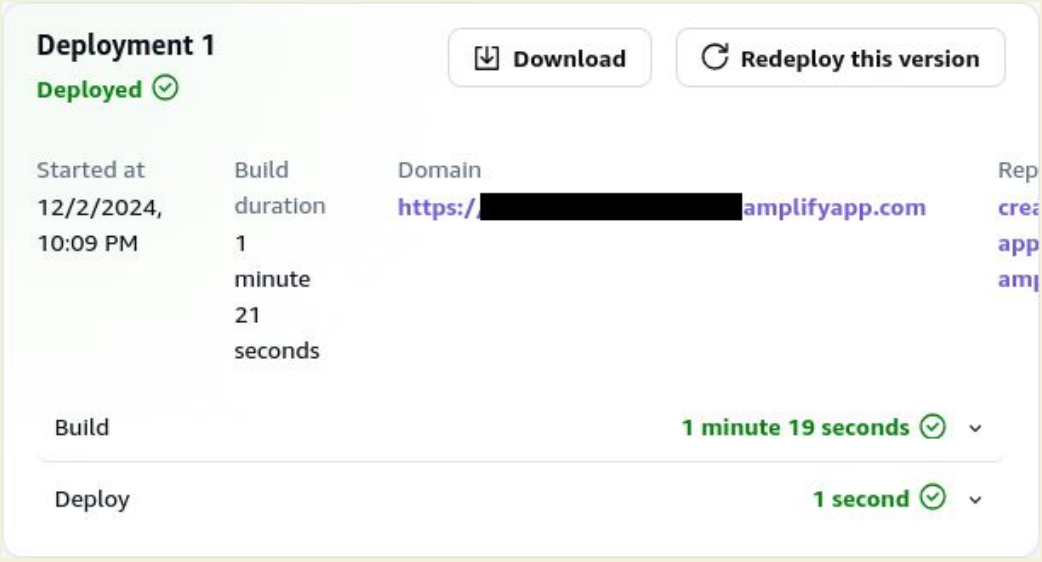
Steps:

- Initialize create-react-app project
  - (`npx create-react-app my-app`)
- Push to GitHub
- Deploy to cloud service with default settings



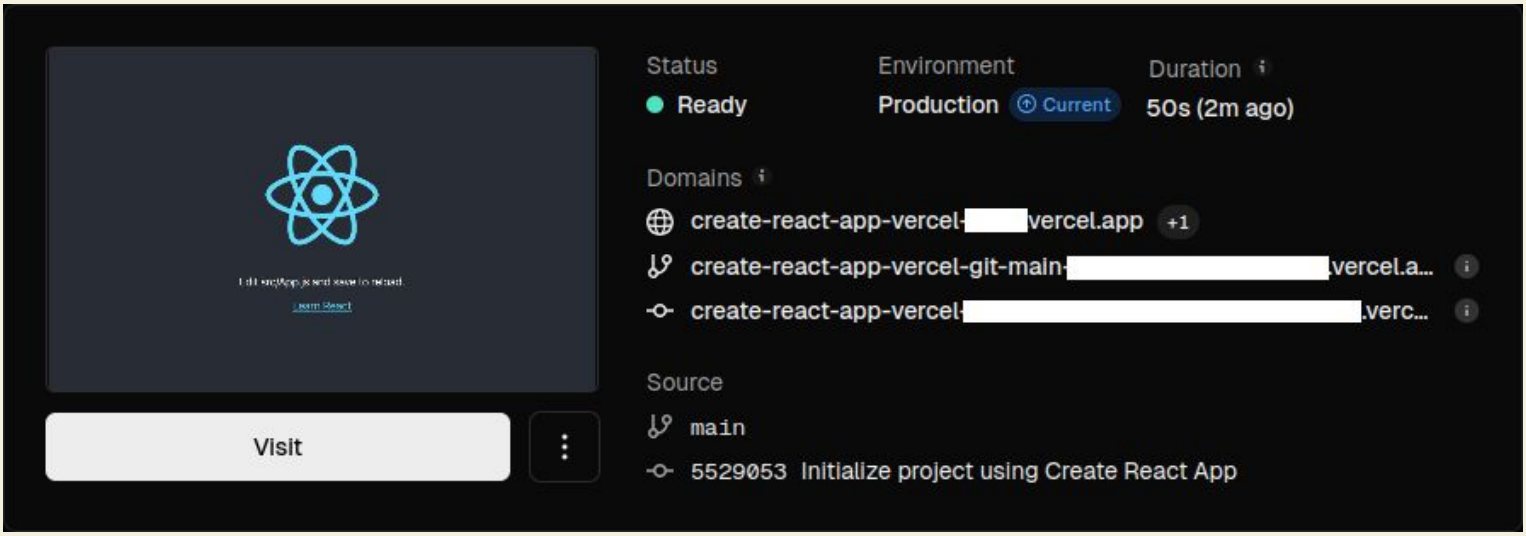
Usability Impressions

AWS Amplify: 1 min, 21 sec



Usability Impressions

Vercel: 50 sec (38% less time!)



Cost

Use case	Pricing	
Work with teams	Always free - no pay-per-seat pricing.	
Deploy an app	**with AWS Free Tier**	**without AWS Free Tier**
	Free for 12 months.	Pay for what you use. Includes multiple sites per project and public SSL certificates at no additional cost.
Build and deploy	No cost up to 1,000 build minutes per month	$0.01 per minute
Data storage	No cost up to 5 GB stored on CDN per month	$0.023 per GB per month (this charge recurs until the app is deleted)
Data transfer out	No cost up to 15 GB per month	$0.15 per GB served
Request count (SSR)	No cost up to 500,000 requests per month	$0.30 per 1 million requests
Request duration (SSR)	No cost up to 100 GB-hours per month	$0.20 per hour (GB-hour)
Build a backend	Amplify app frontends are powered by fully-managed AWS services. Many offer generous Free Tiers to get started and pay as you go pricing thereafter. No monthly minimums - simply pay for what you use.	

From Amplify pricing: https://aws.amazon.com/amplify/pricing/

Cost Analysis: Two Use-Cases

- Hobby Development and Rapid Prototyping
- Production Deployment

Cost Analysis: Rapid Prototyping

Example 1

A startup team with 5 developers have an app that has 300 daily active users. The team commits code 2 times per day.

Monthly build & deploy charges

- Assumptions: Average build time = 3 mins; Number of work days/month = 20
- Total build time per month = num of devs * num of commits/day * num of days * avg. build time = 5*2*20*3 = 600 build mins per month
- **Monthly build & deploy charges = 600*.01 = $6**

Monthly hosting charges

- Assumptions: Web app size = 25 MB, average size of page requested = 1.5 MB
- Monthly GB served = Daily active users * average page size * days = 300 * (1.5/1024) * 30 = 13.18 GB
- Monthly GB stored = web app size * number of monthly builds = (25/1024)*(5*2*20) = 4.88 GB
- **Monthly hosting charges = 13.18*$0.15 + 4.88*$0.023= $1.97 + $0.11 = $2.08**

Total monthly charges

Total charges = Build & deploy charges + Hosting charges = $6+$2.08 = $8.08 per month

Example taken from Amplify pricing: https://aws.amazon.com/amplify/pricing/

Cost Analysis: Rapid Prototyping

Compare to Vercel free tier:

- 100gb/month of data transfer out
- Unlimited build time

Vercel builds the same application in seconds vs Amplify's minutes, which makes being charged for build time in Amplify extremely unappealing.

Amplify doesn't offer much to justify these downsides.

Vercel pricing: https://vercel.com/pricing

Cost Analysis: Production

Example 2

A web app has 10,000 daily active users and is updated 2 times per month.

Monthly build & deploy charges

- Assumptions: Average build time = 3 mins
- Total build time per month = num of updates/month * avg. build time = 2*3 = 6 build mins per month
- **Monthly build & deploy charges = 6*.01 = $0.06**

Monthly hosting charges

- Assumptions: Web app size = 100 MB, average size of page requested = 1.5 MB
- Monthly GB served = Daily active users * average page size * days = 10,000 * (1.5/1024) * 30 = 439.45 GB
- Monthly GB stored = web app size * number of monthly builds = (100/1024)*2 = 0.19 GB
- **Monthly hosting charges = 439.45*$0.15 + 0.19*$0.023= $65.92**

Total monthly charges

Total charges = Build & deploy charges + Hosting charges = $0.06+$65.92 = $65.98 per month

From Amplify pricing: https://aws.amazon.com/amplify/pricing/

Cost Analysis: Production

Compare to EC2

- EC2 charges data transfer at $0.09/GB. For 439.45GB, we pay 39$.
- This leaves 27$/month. We can afford a t3.small instance for 15$.
- t3.small isn't a very powerful instance,
- 10,000 requests per day is around 7 requests per minute, which our instance should be able to handle *easily*.

The biggest cost of Amplify in this use-case is the Data Transfer rate of $0.15/GB, much higher than EC2's. 10,000 requests a day is not that many, and it's clear that as the size of the webpage or the number of requests increases, the cost of data transfer will greatly outpace any other costs.

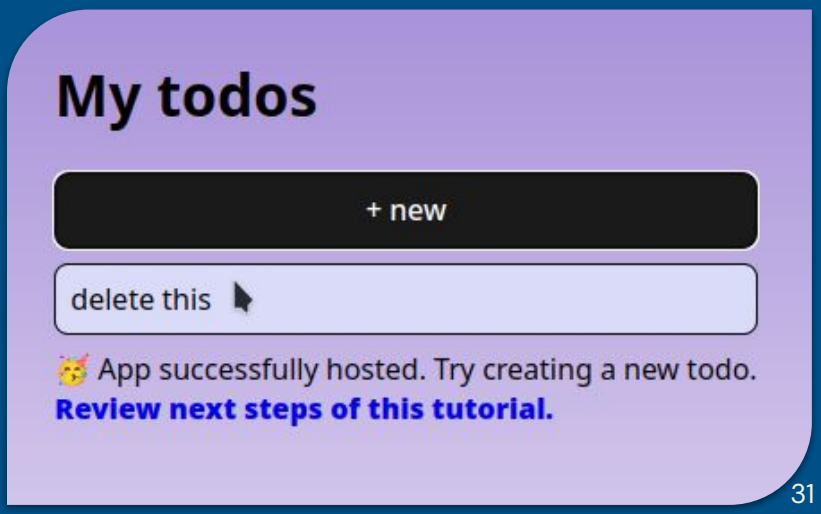
Conclusions

- AWS Amplify provides a simple, managed solution for deploying and hosting a full stack web application
- Good if you want to fully buy into the AWS ecosystem and you are unfamiliar with cloud technologies and just want to get a website spun up
- Ultimately, not very cost-effective for what you get: You may want to consider other services if you're looking for better developer experience or more services under the free tier.

Demonstration

Making a to-do app with Vue and AWS Amplify

Adapted from AWS Amplify Vue Quickstart Guide

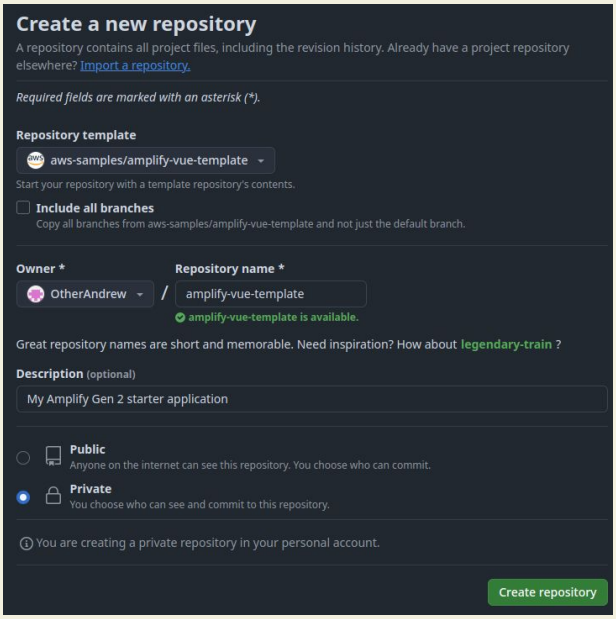


Prerequisites

- AWS Account
- GitHub Account
- Node.js
- Git
- Code editor

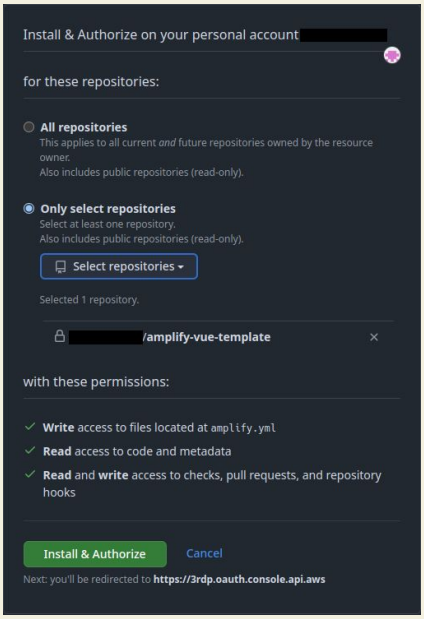
Step 1: Create starter app from template

Template:
https://github.com/new?template_name=amplify-vue-template&template_owner=aws-samples&name=amplify-vue-template&description=My%20Amplify%20Gen%202%20starter%20application



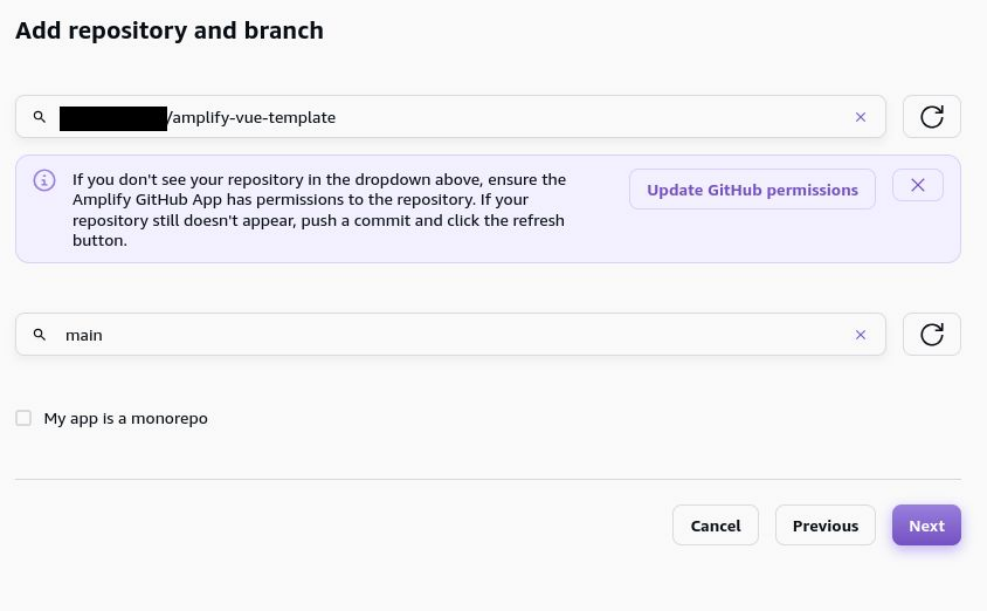
Step 2: Deploy starter app to GitHub

- Deploy to Amplify: https://console.aws.amazon.com/amplify/create/repo-branch
- Select GitHub
- Give AWS permission to read from your GitHub account
- Install and Authorize AWS to read from the repo you just created



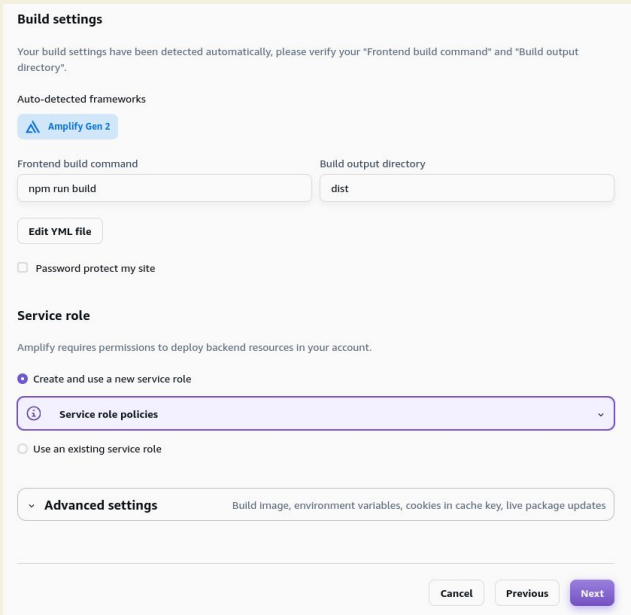
Step 2: Deploy starter app to GitHub

- Select repo for starter app, and choose main branch
- Click “Next”



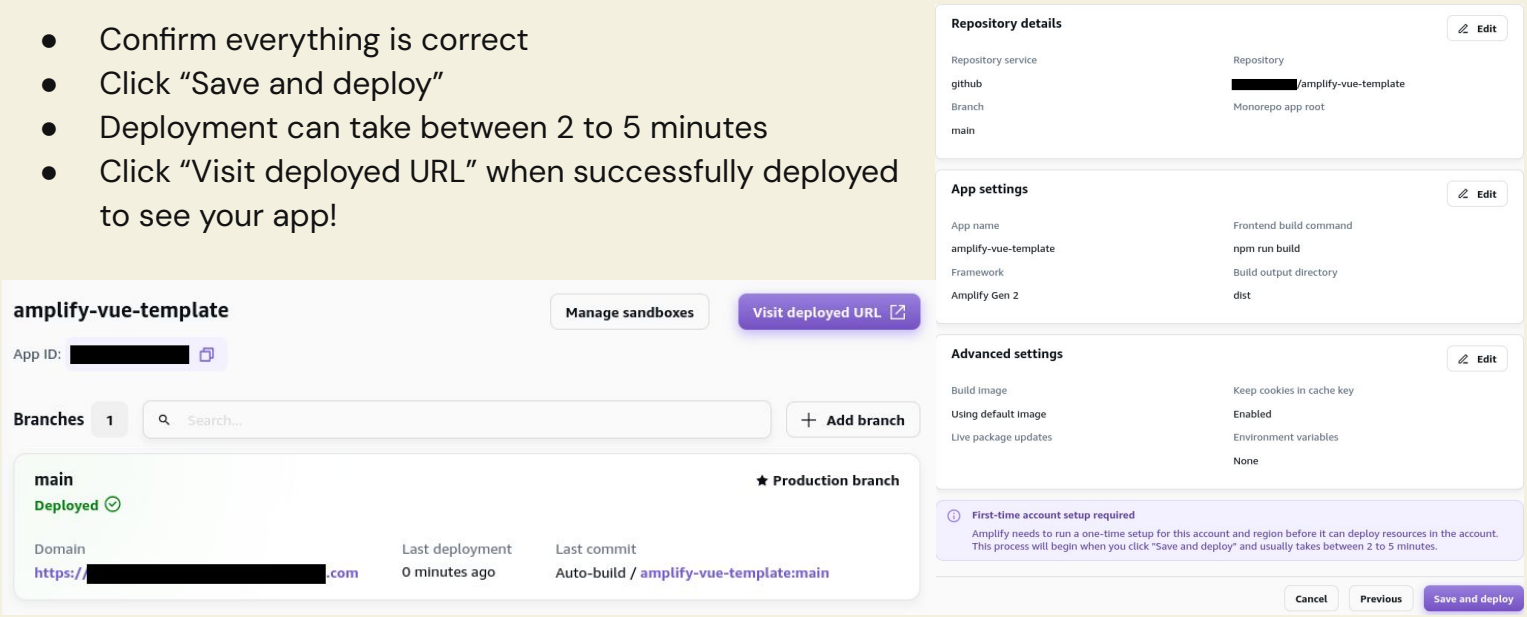
Step 2: Deploy starter app to GitHub

- Use default settings on App Settings screen
- Click “Next”



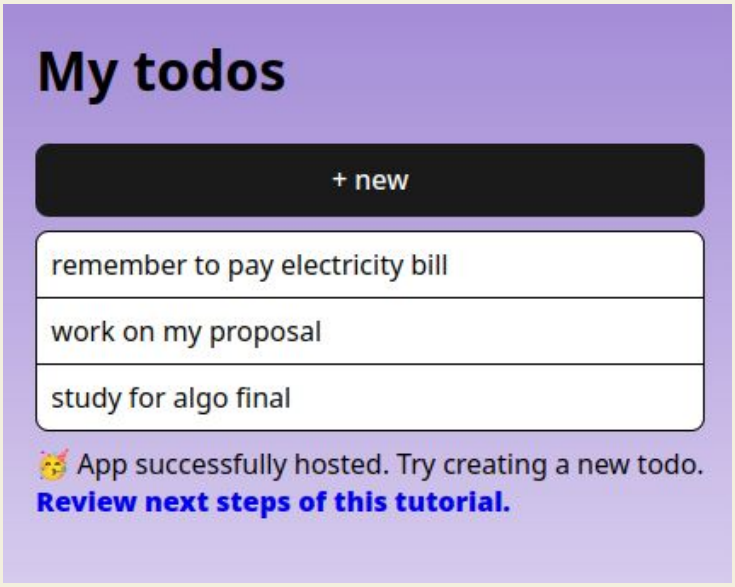
Step 2: Deploy starter app to GitHub

- Confirm everything is correct
- Click “Save and deploy”
- Deployment can take between 2 to 5 minutes
- Click “Visit deployed URL” when successfully deployed to see your app!



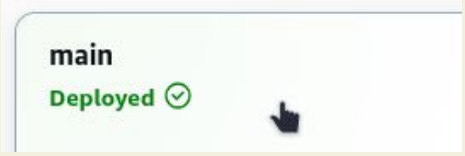
Step 3: Test your new app!

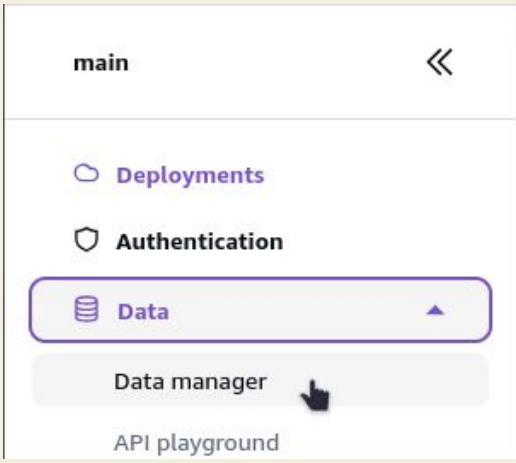
- Add some items

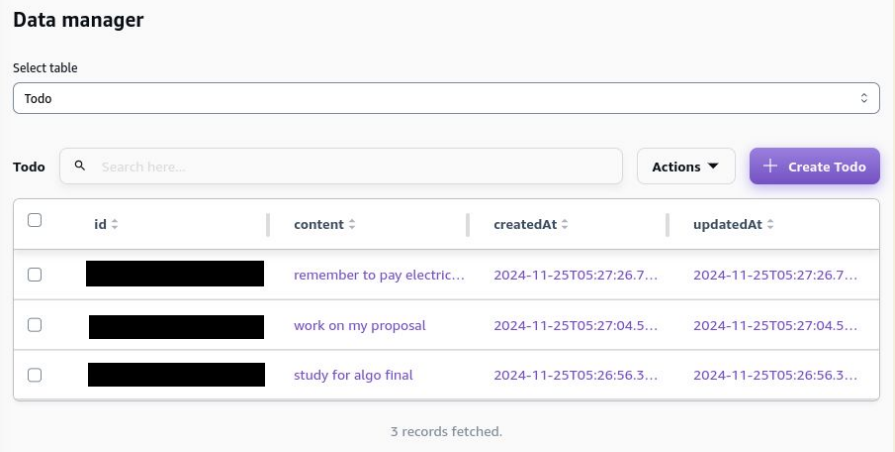


Step 3: Test your new app!

- Click into the main branch in Amplify overview
- Click into “Data manager” under “Data” in the left sidebar to see your todos



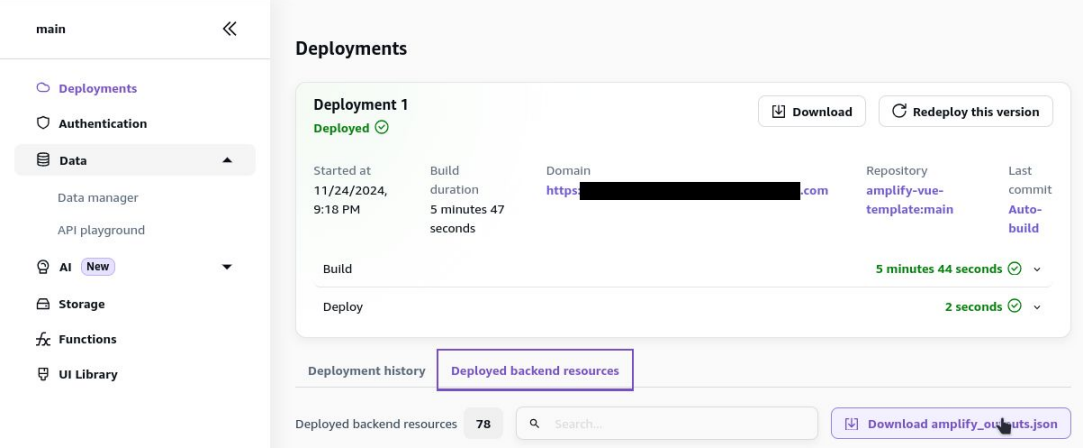




Let’s add a feature.

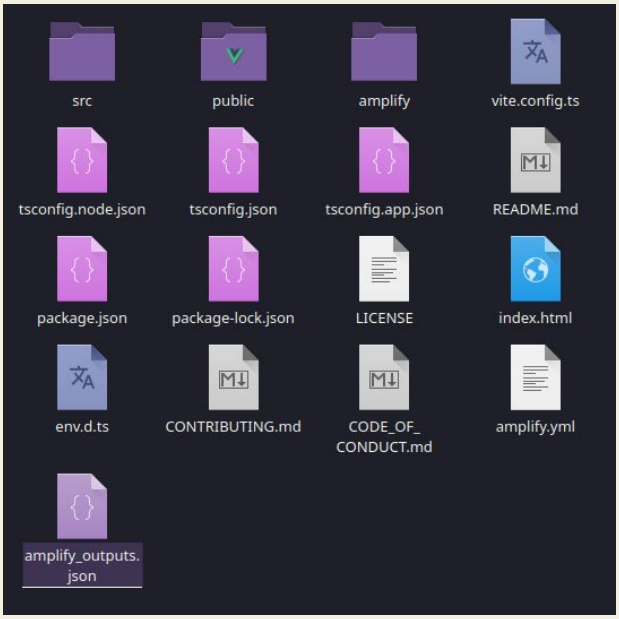
Step 4: Set up local dev environment

- Download amplify_outputs.json
  - Click into the “Deployments” tab in the sidebar
  - Scroll down and click on “Download amplify_outputs.json”



Step 4: Set up local dev environment

- Clone your repo to your local machine
  - `git clone https://github.com/<github-user>/amplify-vue-template.git`
- Navigate into the repo and install dependencies
  - `cd amplify-vue-template && npm install`
- Move the “amplify_outputs.json” file you just downloaded into the repo’s root directory
- Open the project in your favorite code editor



Step 5: Add delete functionality

- In src/components/Todos.vue, add :

```
function deleteTodo(id: string) {
    client.models.Todo.delete({ id })
}
```

Under the createTodo() function.

Step 5: Add delete functionality

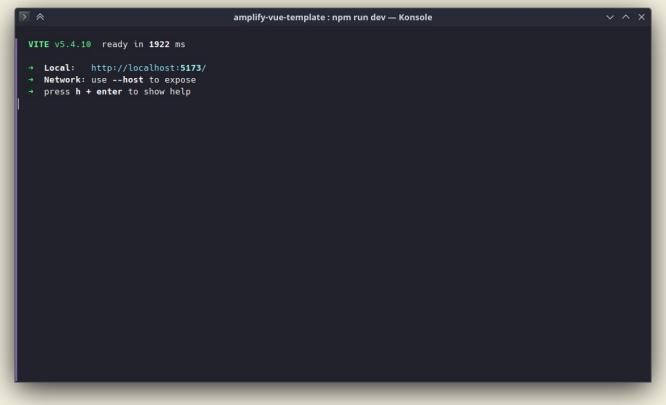
- Add an onclick handler to the list elements
  - Add:

```
@click="deleteTodo(todo.id)"
```

inside the <li> opening tag

Step 6: Test changes locally

- Run `npm run dev`
- Visit local dev server at http://localhost:5173/
- Click on a to-do item to delete it!



Step 7: Commit changes and push

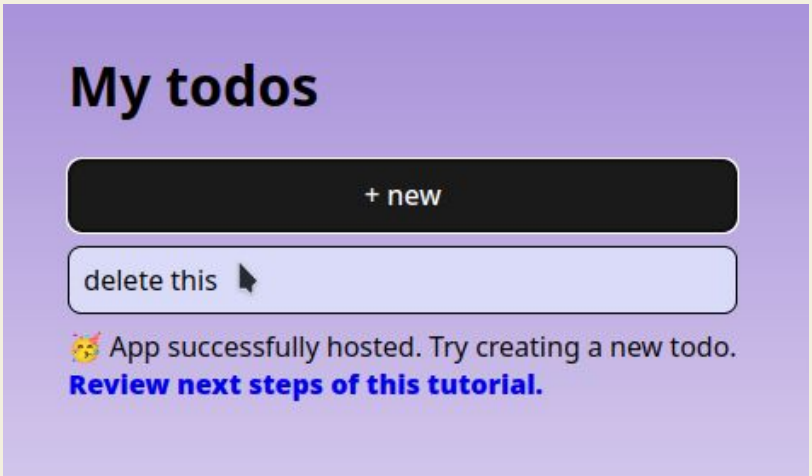
```
git add src/components/Todos.vue

git commit -m "add delete on click functionality to Todos.vue"

git push
```

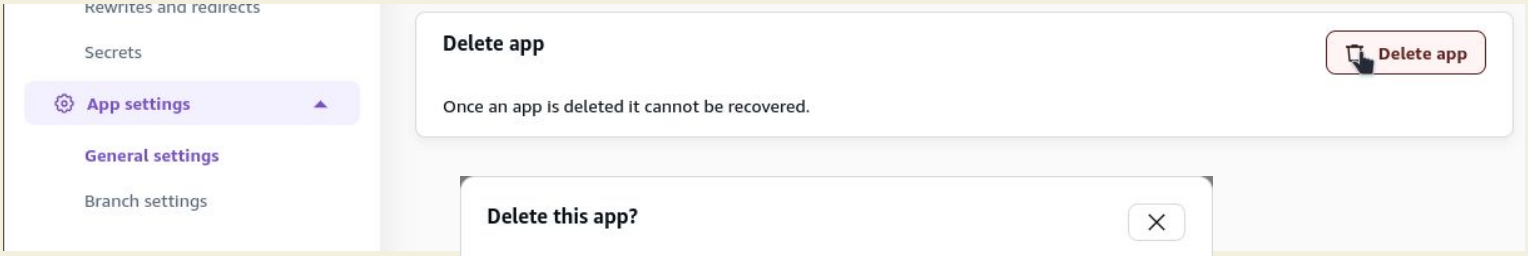
Step 8: See changes on Amplify!

- Amplify will automatically rebuild and redeploy the app on every commit
  - This can take between 2 to 5 minutes

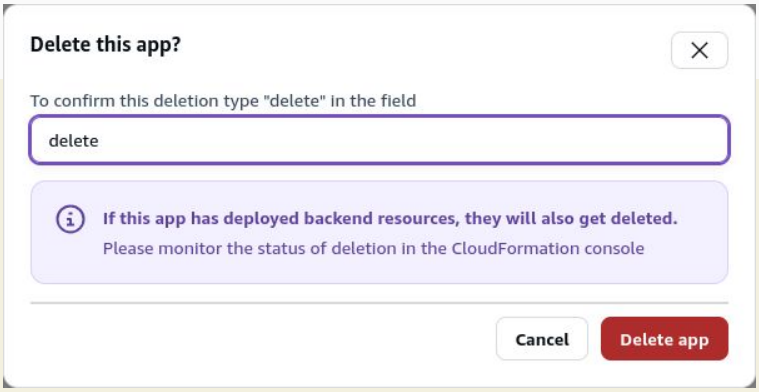


Step 9: Teardown

- Under “App settings” > “General settings” in the sidebar, click on the “Delete app” button



- Confirm deletion:



Questions?