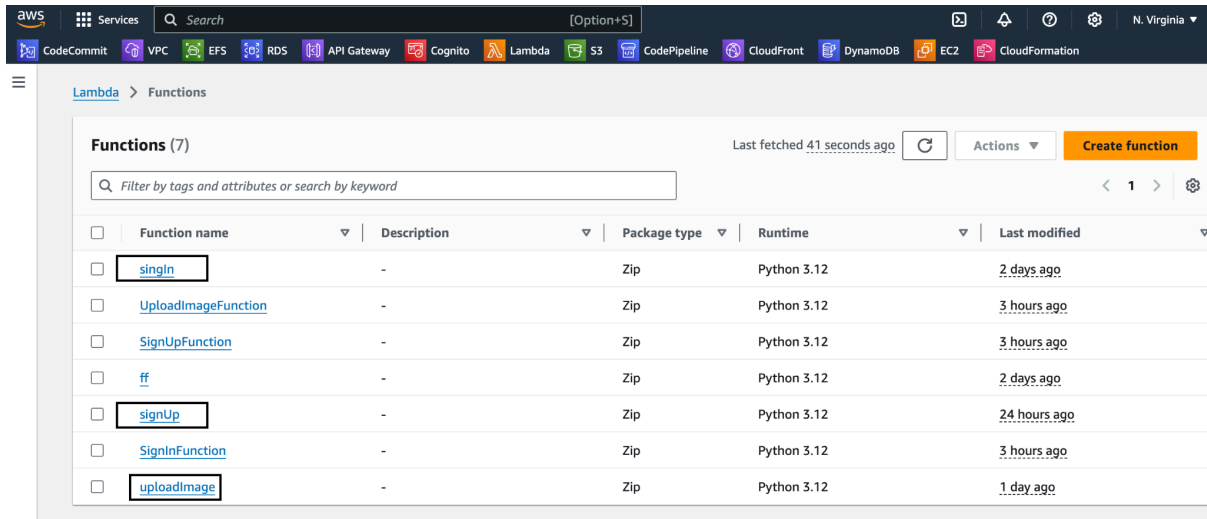# AWS App Development and Deployment

Mayssa Elloumi
melloumi@miu.edu

Step 1: Manual Development and Testing:

Developed Lambda Functions:

test case : lambda signup:

Configure test event   ✕

A test event is a JSON object that mocks the structure of requests emitted by AWS services to invoke a Lambda function. Use it to see the function's invocation result.

To invoke your function without saving an event, modify the event, then choose Test. Lambda uses the modified event to invoke your function, but does not overwrite the original event until you choose Save changes.

Test event action

◯ Create new event	⦿ Edit saved event

Event name

event1.1 ▼	↻	Delete

Event JSON   [Format JSON]

```
1 ▾ {
2       "body": "{\"email\": \"testemail9@example.com\", \"password\": \"testpassword\", \"nar
3   }
```

Cancel   Invoke   Save

☰ aws ⦙⦙⦙ Services   Q Search   [Option+S]   ⊡ 🔔 ❓ ⚙ N. Virginia ▼

CodeCommit   VPC   EFS   RDS   API Gateway   Cognito   Lambda   S3   CodePipeline   CloudFront   DynamoDB   EC2   CloudFormation

☰ ⊘ The test event **event1.1** was successfully saved.

▲ File Edit Find View Go Tools Window   [Test ▾] Deploy   ⤢ ⚙

Q Go to Anything (⌘ P)   ☰   lambda_function ✕   Environment Var ✕   **Execution result**✕   ⊕

▾ 📁 signUp · /   ⚙▾   ▾ Execution results   Status: **Succeeded** Max memory used: 86 MB Time: 598.95 ms
   ⊕ lambda_function.py   **Test Event Name**
                                event1.1

                                Response
                                {
   "statusCode": 200,
                                    "body": "\"User registered successfully\"",
                                    "headers": {
                                        "Access-Control-Allow-Origin": "*",
                                        "Access-Control-Allow-Methods": "POST, GET, OPTIONS",
                                        "Access-Control-Allow-Headers": "Content-Type, Authorization",
                                        "Access-Control-Allow-Credentials": "true"
                                    }
                                }

                                Function Logs
   START RequestId: 4affbc3c-9973-4636-aea2-8025da698e80 Version: $LATEST
                                END RequestId: 4affbc3c-9973-4636-aea2-8025da698e80
                                REPORT RequestId: 4affbc3c-9973-4636-aea2-8025da698e80 Duration: 598.95 ms Billed Duration: 599 ms Memory Size: 128 MB Max Memory Used: 86 MB

                                Request ID
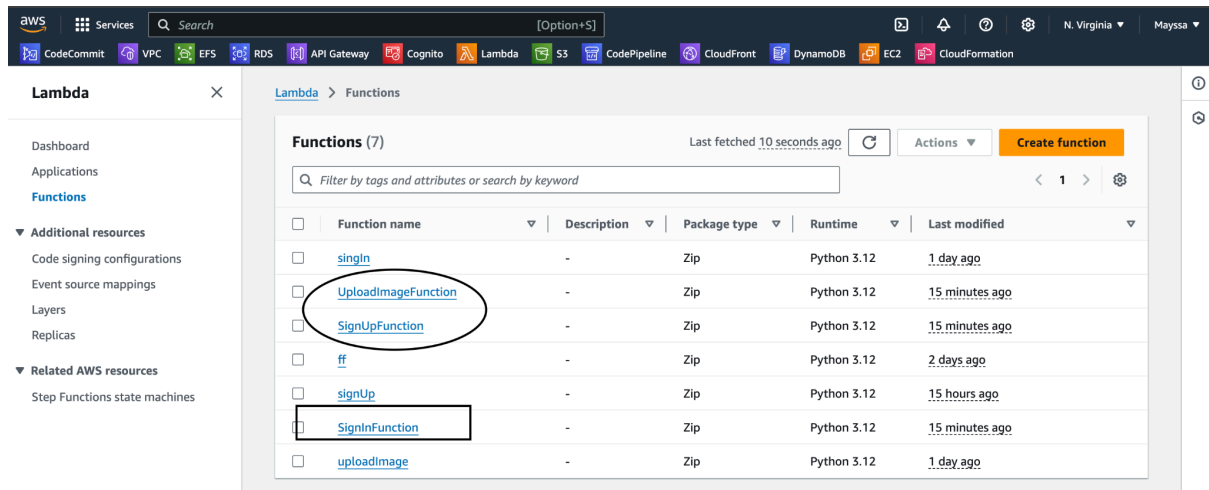   4affbc3c-9973-4636-aea2-8025da698e80

API gateway development:
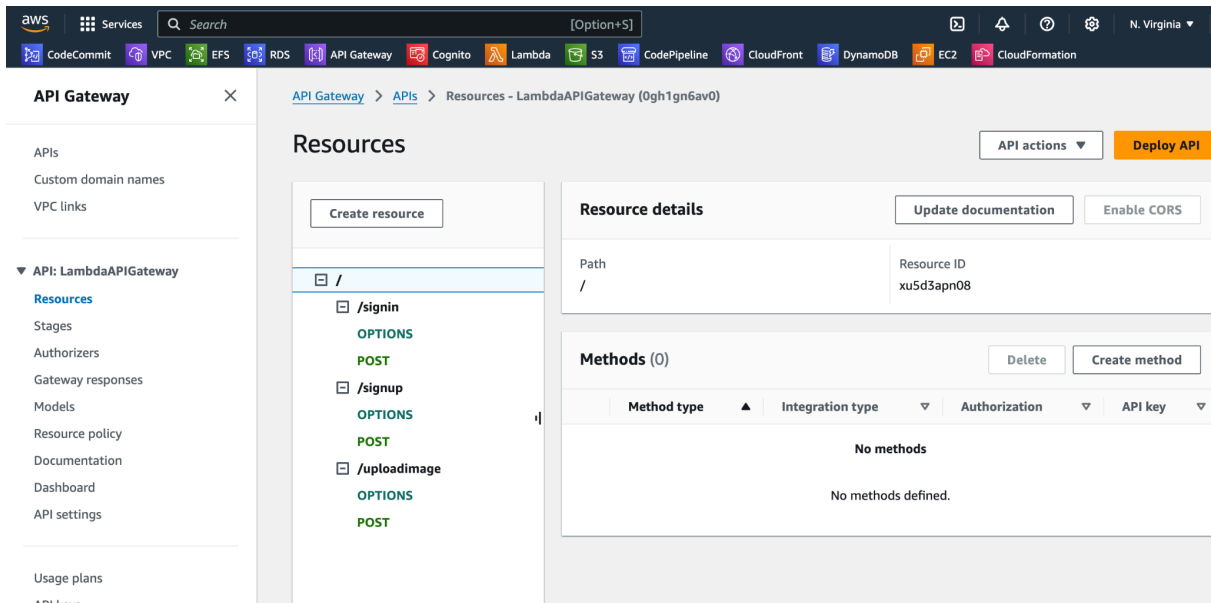
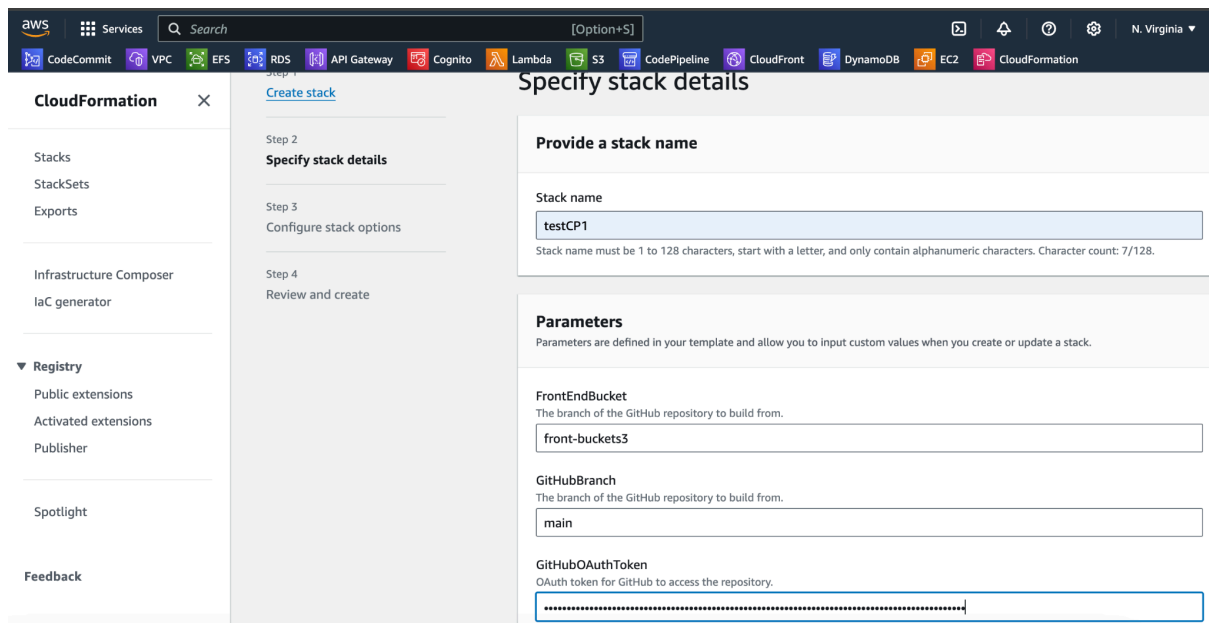After the launch of the yaml file of cloud formation:
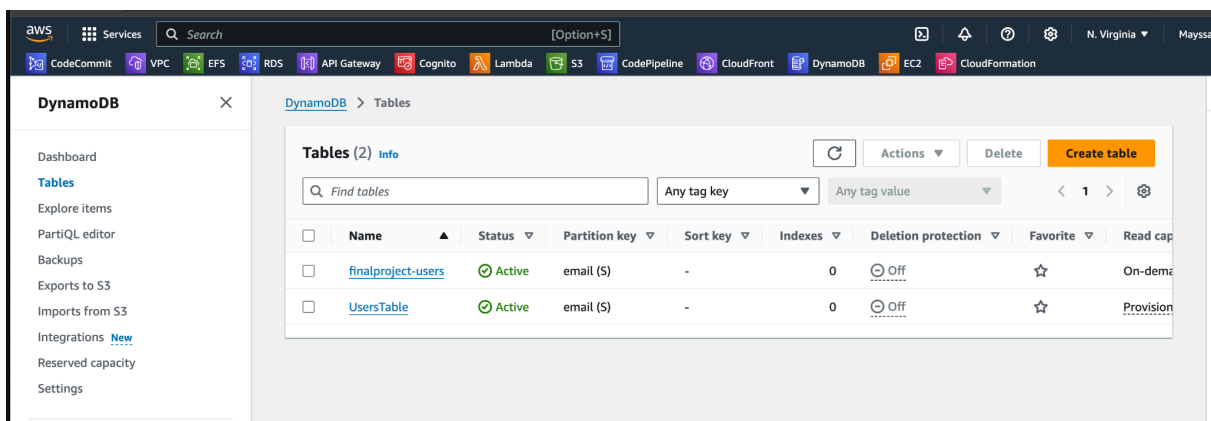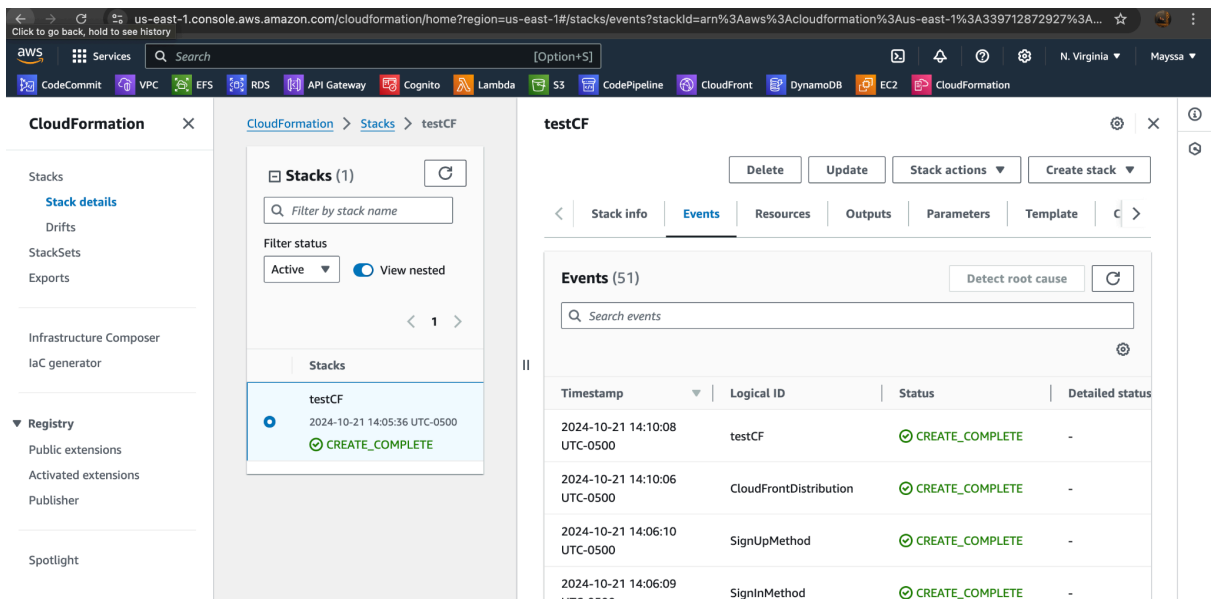
Post YAML file of cloud pipelines:

- Upload Auth token for the Github account (In this case, my account has been hardcoded in the configuration. Can be parameterized if necessary):

Auth Token:
github_pat_11BIK437Q0ivulE8jnereC_S2nXfLl9Tnl9t1Bi8zYZlwN62urjom52s00bBAewH1xl3KZ
ND4DSy9y89tY

- Post Run

Code pipeline dev:

Deployment buckets:

DynamoDB setup:
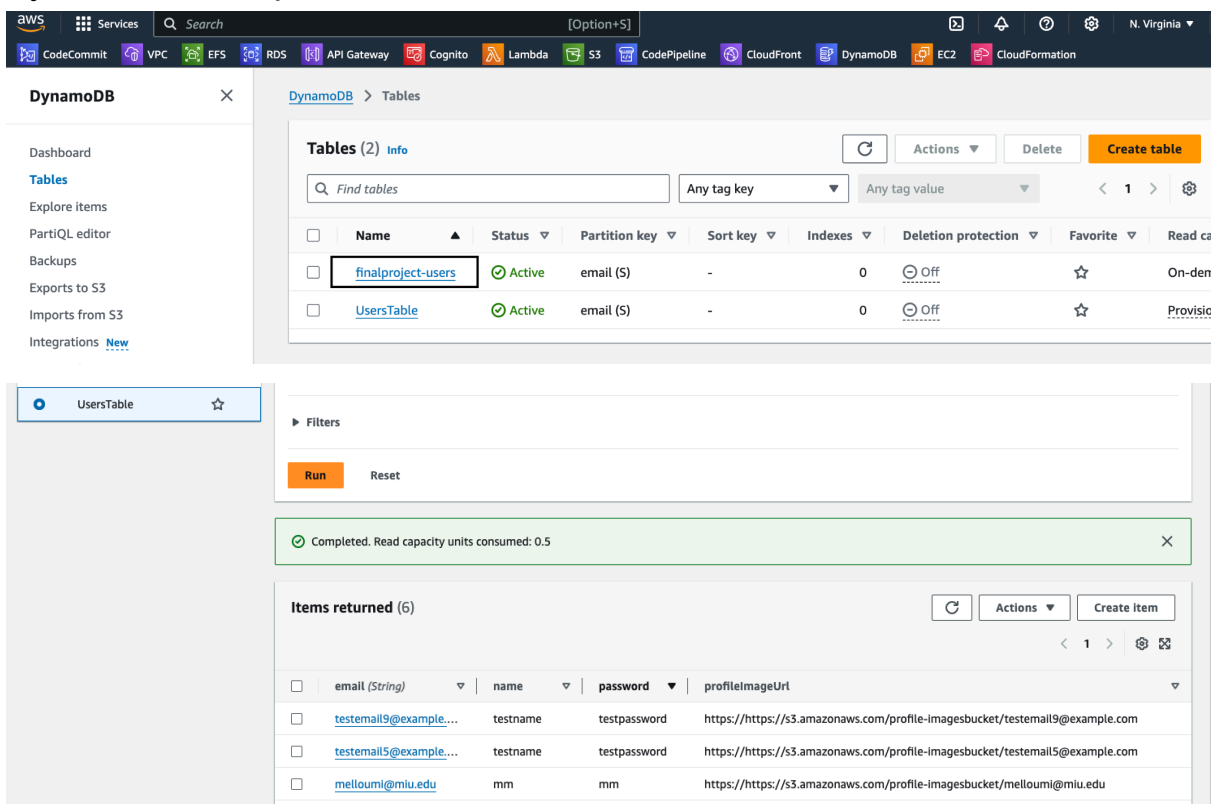
User interface:
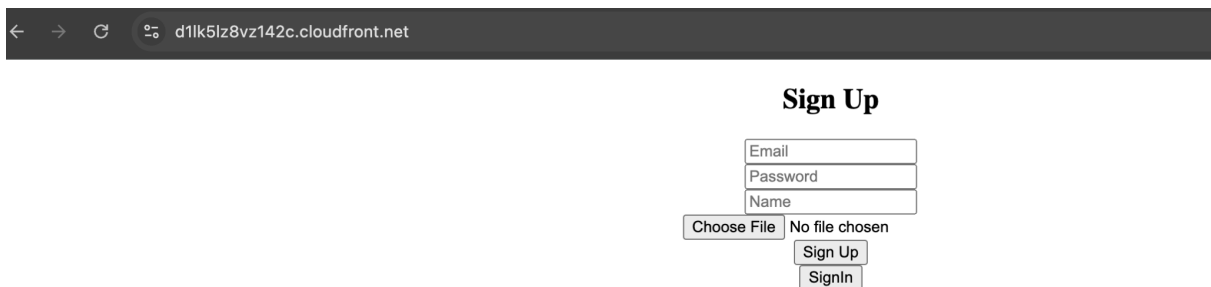
Sign Up

mayssa@gmail.com

••••

mayssa

Choose File MainAfter.webp

Sign Up

SignIn

Click on SignUp to be directed to the Login Page:

Login

Email: Enter your email
Password: Enter your password

Login

Login

Email: mayssa@gmail.com
Password: ••••

Login

Click on the Login button to move to Home page:

Welcome to the Home Page

Submit Logout

Click on Submit to update on DynamoDB:

Welcome to the Home Page

Submit Logout
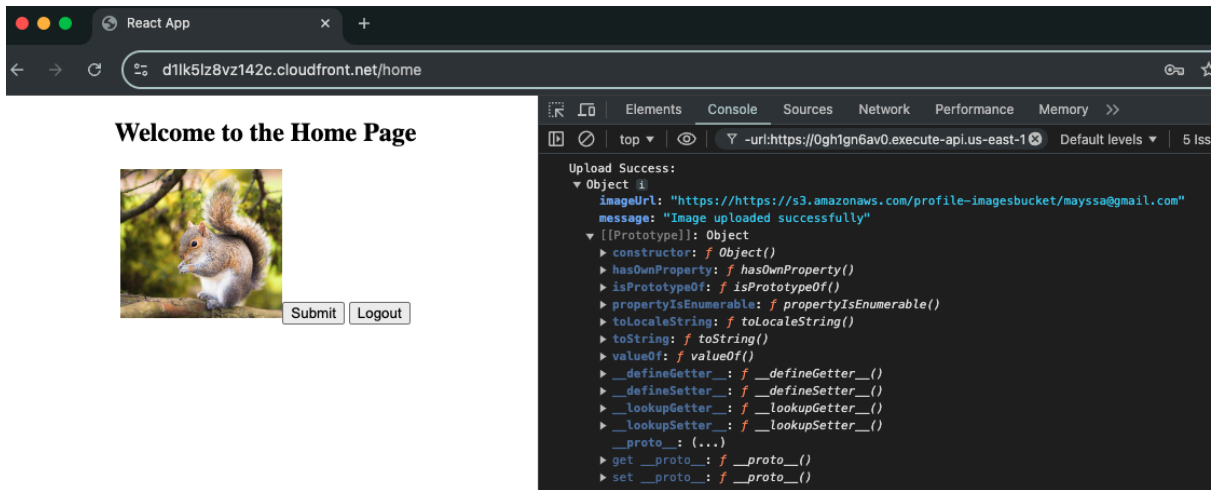
Click on Logout to move to SignIn Page:
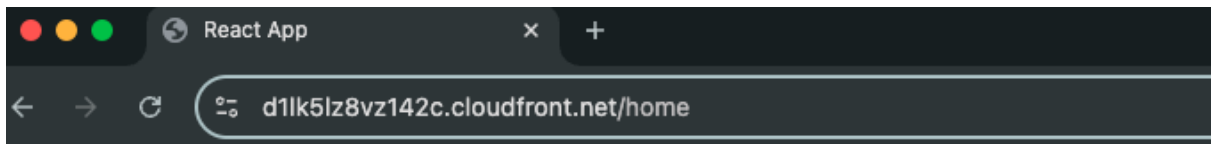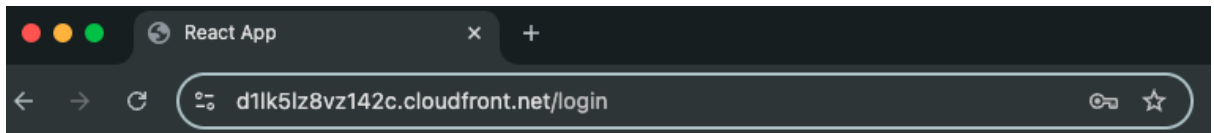
Step 2: Cloud Formation Resource Deployment:

Cloud Pipeline based Frontend React CI/CD:
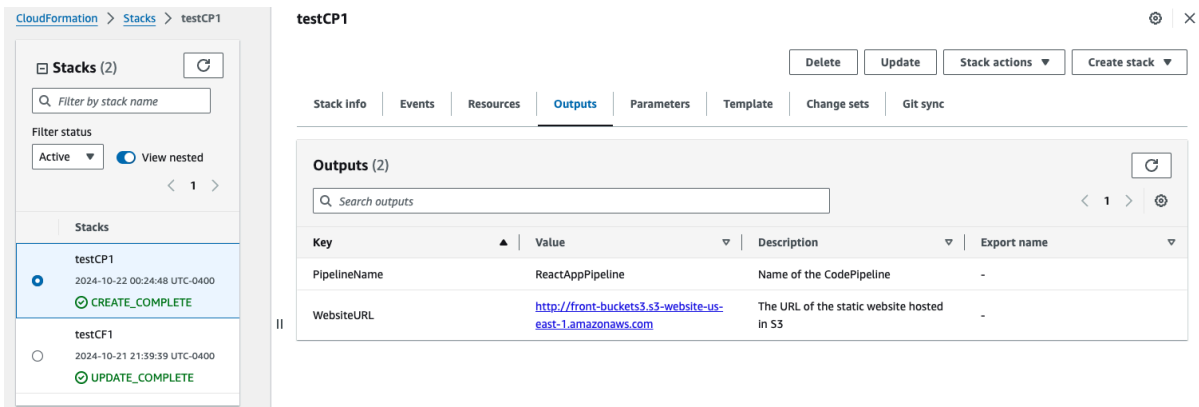
Deployment of backend lambda functions, API gateway, DynamoDB:
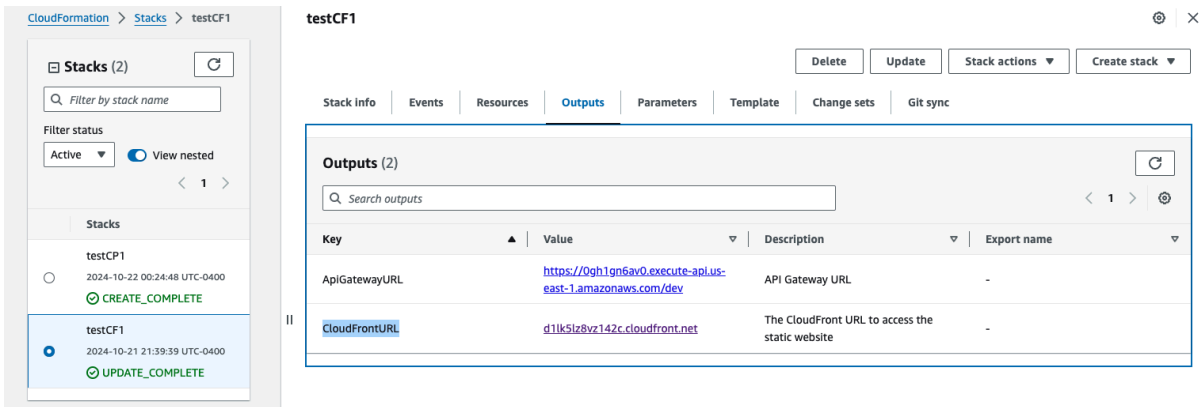
Note: All IAM roles are policies are also included in the YAML template file.

Step 3: CICD Implementation:

Stages:
1. Source:

ReactAppPipeline

Pipeline type: **V1**   Execution mode: **SUPERSEDED**

⊘ **Source** Succeeded

Pipeline execution ID: 5a263c82-e9aa-4c03-a88c-769678fa59e3

GitHub_Source
GitHub (Version 1) ↗

⊘ Succeeded - 22 minutes ago

8d2db918 ↗

View details

8d2db918 ↗ GitHub_Source: cloud_formation file update

2. Build:

⊘ **Build** Succeeded

Pipeline execution ID: 5a263c82-e9aa-4c03-a88c-769678fa59e3

BuildAction
AWS CodeBuild

⊘ Succeeded - 21 minutes ago

View details

8d2db918 ↗ GitHub_Source: cloud_formation file update

Execution Logs:

Action execution details

Action name: BuildAction   Status: Succeeded

```
 90  You can control this with the homepage field in your package.json.
 91
 92  The build folder is ready to be deployed.
 93  You may serve it with a static server:
 94
 95    npm install -g serve
 96    serve -s build
 97
 98  Find out more about deployment here:
 99
100    https://cra.link/deployment
101
102
103  [Container] 2024/10/22 16:11:11.361190 Phase complete: BUILD State: SUCCEEDED
104  [Container] 2024/10/22 16:11:11.361209 Phase context status code:  Message:
105  [Container] 2024/10/22 16:11:11.394612 Entering phase POST_BUILD
106  [Container] 2024/10/22 16:11:11.396951 Phase complete: POST_BUILD State: SUCCEEDED
107  [Container] 2024/10/22 16:11:11.396966 Phase context status code:  Message:
108  [Container] 2024/10/22 16:11:11.481767 Expanding base directory path: build
109  [Container] 2024/10/22 16:11:11.483796 Assembling file list
110  [Container] 2024/10/22 16:11:11.483810 Expanding build
111  [Container] 2024/10/22 16:11:11.485568 Expanding file paths for base directory build
112  [Container] 2024/10/22 16:11:11.485579 Assembling file list
113  [Container] 2024/10/22 16:11:11.485583 Expanding **/*
114  [Container] 2024/10/22 16:11:11.487453 Found 7 file(s)
115  [Container] 2024/10/22 16:11:11.523718 Set report auto-discover timeout to 5 seconds
116  [Container] 2024/10/22 16:11:11.523780 Expanding base directory path:  .
117  [Container] 2024/10/22 16:11:11.525658 Assembling file list
118  [Container] 2024/10/22 16:11:11.525674 Expanding .
119  [Container] 2024/10/22 16:11:11.527407 Expanding file paths for base directory .
120  [Container] 2024/10/22 16:11:11.527418 Assembling file list
121  [Container] 2024/10/22 16:11:11.527422 Expanding **/*
122  [Container] 2024/10/22 16:11:11.691645 Found 11 file(s)
123  [Container] 2024/10/22 16:11:11.691695 Report auto-discover file discovery took 0.167977 seconds
124  [Container] 2024/10/22 16:11:11.692366 Phase complete: UPLOAD_ARTIFACTS State: SUCCEEDED
125  [Container] 2024/10/22 16:11:11.692380 Phase context status code:  Message:
126
```

3. Deploy:

⊘ **Deploy** Succeeded

Pipeline execution ID: 5a263c82-e9aa-4c03-a88c-769678fa59e3

S3Deploy

Amazon S3 ⧉

⊘ Succeeded - 21 minutes ago

View details

8d2db918 ⧉ GitHub_Source: cloud_formation file update

Deployed frontend:

front-buckets3 Info

Objects | Properties | Permissions | Metrics | Management | Access Points

Objects (3) Info

[⟳] | [□ Copy S3 URI] | [□ Copy URL] | [↓ Download] | [Open ⬈] | [Delete] | [Actions ▼] | [Create folder] | [⬆ Upload]

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory ⬈ to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more ⬈

🔍 Find objects by prefix

⟨ 1 ⟩ ⚙

	Name ▲	Type ▽	Last modified ▽	Size ▽	Storage class ▽
☐	📄 asset-manifest.json	json	October 22, 2024, 12:11:50 (UTC-04:00)	369.0 B	Standard
☐	📄 index.html	html	October 22, 2024, 12:11:50 (UTC-04:00)	600.0 B	Standard
☐	📁 static/	Folder	-	-	-