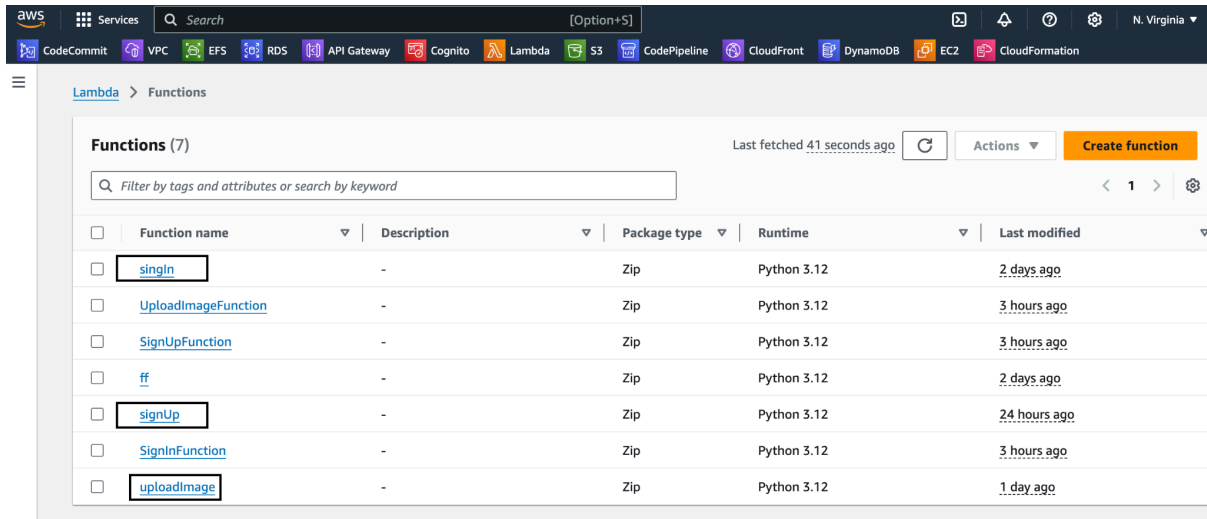


AWS App Development and Deployment

Mayssa Elloumi
melloumi@miu.edu

Step 1: Manual Development and Testing:

Developed Lambda Functions:



The screenshot shows the AWS Lambda console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and a list of services including CodeCommit, VPC, EFS, RDS, API Gateway, Cognito, Lambda, S3, CodePipeline, CloudFront, DynamoDB, EC2, and CloudFormation. Below the navigation bar, the breadcrumb trail shows 'Lambda > Functions'. The main content area displays 'Functions (7)' with a refresh button and a 'Create function' button. A search bar is present with the placeholder text 'Filter by tags and attributes or search by keyword'. Below the search bar is a table listing the functions.

<input type="checkbox"/>	Function name	Description	Package type	Runtime	Last modified
<input type="checkbox"/>	singIn	-	Zip	Python 3.12	2 days ago
<input type="checkbox"/>	UploadImageFunction	-	Zip	Python 3.12	3 hours ago
<input type="checkbox"/>	SignUpFunction	-	Zip	Python 3.12	3 hours ago
<input type="checkbox"/>	ff	-	Zip	Python 3.12	2 days ago
<input type="checkbox"/>	signUp	-	Zip	Python 3.12	24 hours ago
<input type="checkbox"/>	SignInFunction	-	Zip	Python 3.12	3 hours ago
<input type="checkbox"/>	uploadImage	-	Zip	Python 3.12	1 day ago

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\", \"name\": \"testname\"}"  
3 }
```

Cancel

Invoke

Save

The screenshot shows the AWS Lambda console interface. At the top, a green notification bar states: "The test event event1.1 was successfully saved." Below this, the "Execution results" tab is selected, displaying the following information:

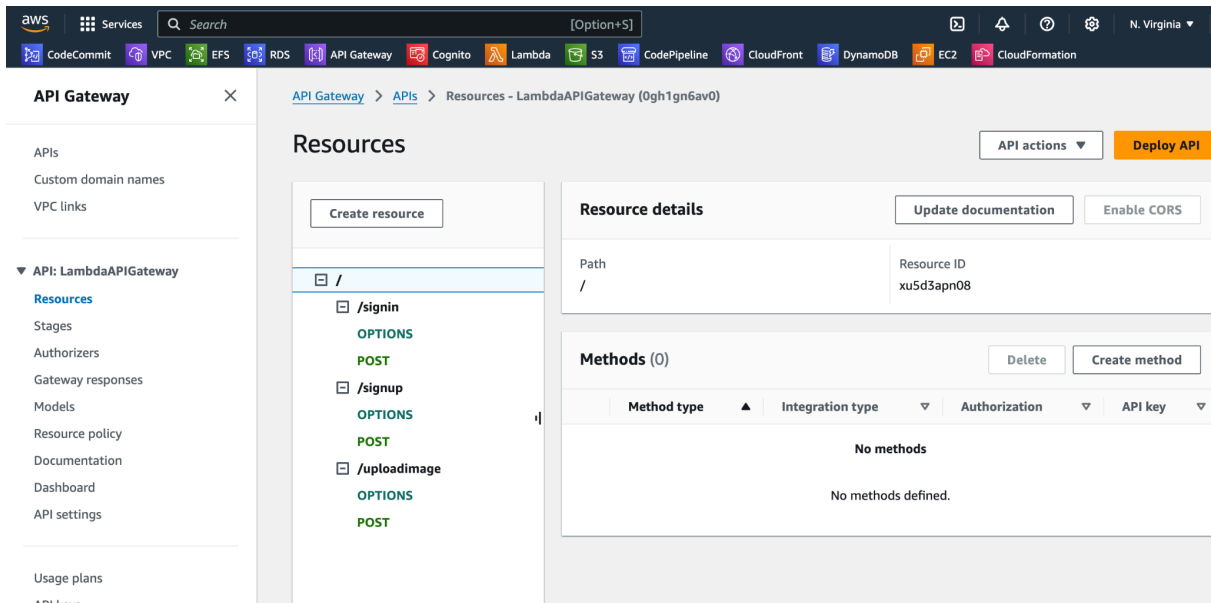
- Test Event Name:** event1.1
- Response:**

```
{  
  "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:**

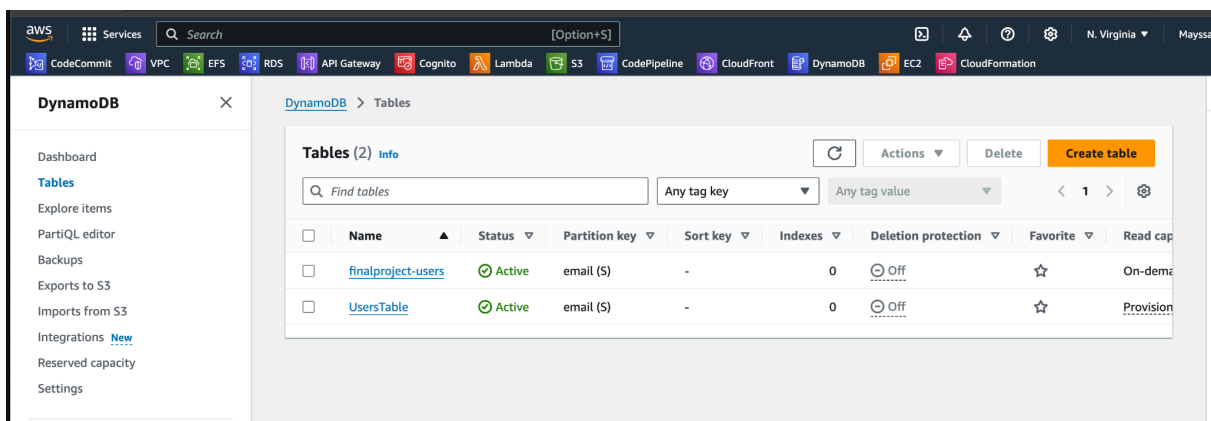
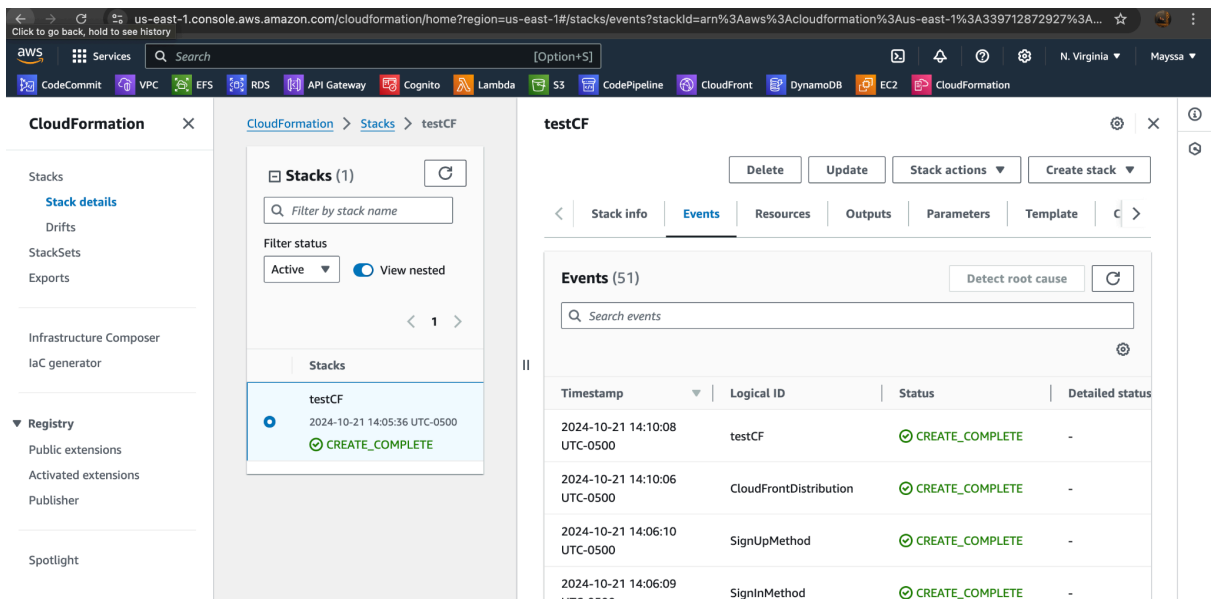
```
START RequestId: 4affbc3c-9973-4636-aea2-8025da698e80 Version: $LATEST  
END RequestId: 4affbc3c-9973-4636-aea2-8025da698e80  
REPORT RequestId: 4affbc3c-9973-4636-aea2-8025da698e80 Duration: 598.95 ms Billed Duration: 599 ms Memory Size: 128 MB Max Memory Used: 86 MB
```
- Request ID:** 4affbc3c-9973-4636-aea2-8025da698e80

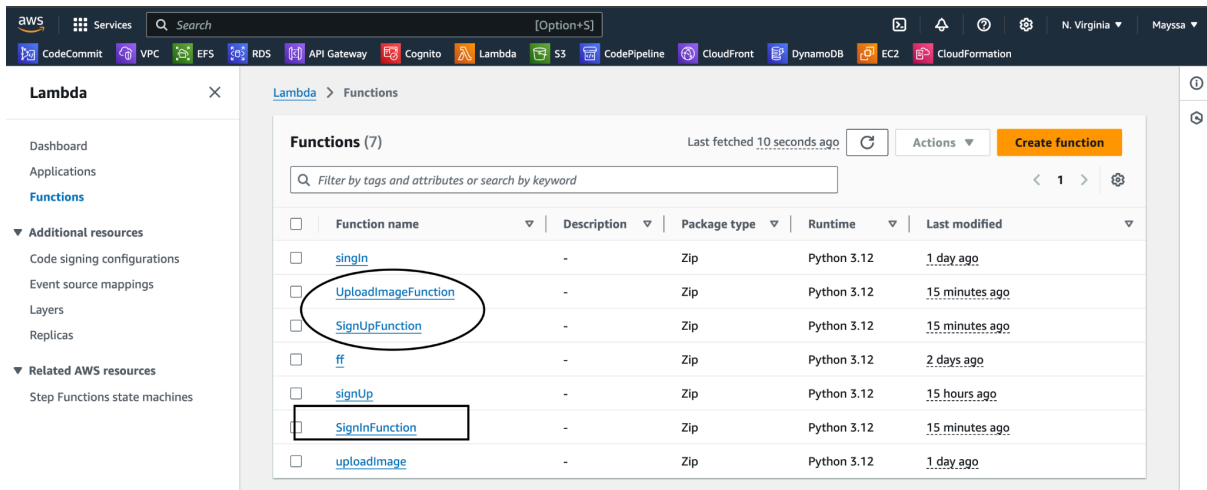
The status is "Succeeded", with a max memory used of 86 MB and a time of 598.95 ms.

API gateway development:



After the launch of the yaml file of cloud formation:



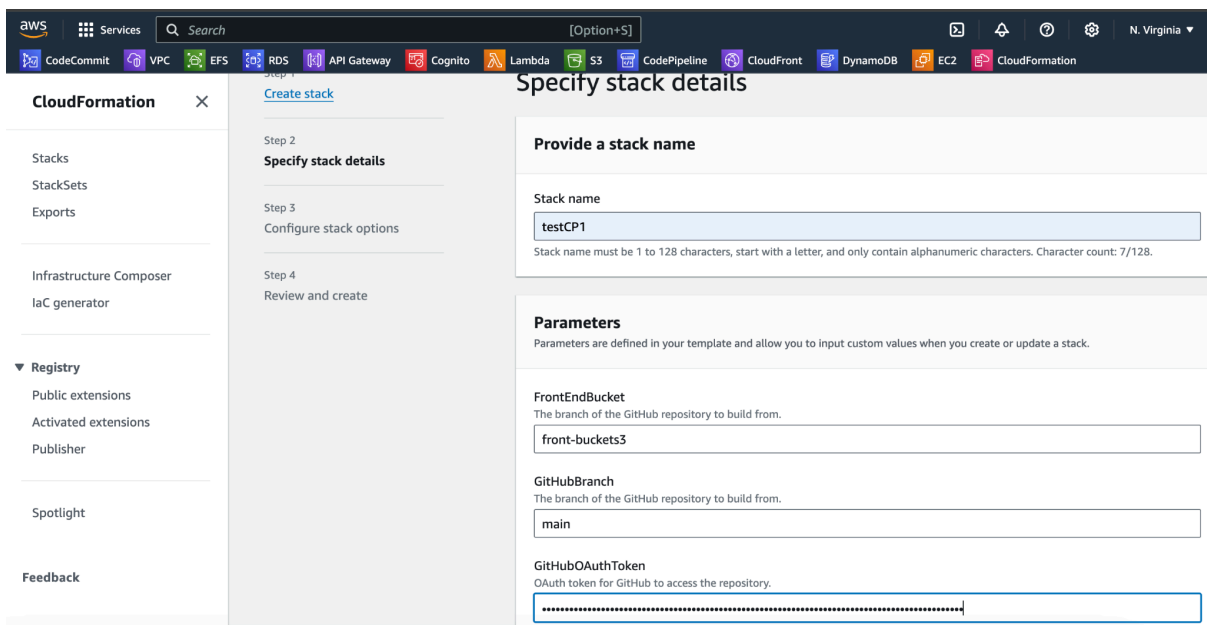


Post YAML file of cloud pipelines:

- 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_11BMKEK5l0tfXTtmKeIX6c_fBJ3VaRYo6X6KifiFjK6CBN5mXIWHZvgr7riEppGTYjED
LKCJCsv3FABht



- Post Run

aws Services Search [Option+S] N. Virginia

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

CloudFormation

Stacks

Stack details

Drifts

StackSets

Exports

Infrastructure Composer

laC generator

Registry

Public extensions

Activated extensions

Publisher

Spotlight

CloudFormation > Stacks > testCP1

Stacks (2)

Filter by stack name

Filter status

Active View nested

Stacks

testCP1

2024-10-21 22:58:42 UTC-0500

CREATE_COMPLETE

testCP1

2024-10-21 20:39:39 UTC-0500

CREATE_COMPLETE

testCP1

Delete Update Stack actions Create stack

Stack info Events Resources Outputs Parameters Template

Overview

Stack ID

arn:aws:cloudformation:us-east-1:339712872927:stack/testCP1/edbe1110-9029-11ef-b08b-0e76f521cad7

Description

CodePipeline for building and deploying a React app to an existing S3 bucket as a static website.

Status

CREATE_COMPLETE

Detailed status

-

Status reason

-

Root stack

-

Parent stack

-

Created time

2024-10-21 22:58:42 UTC-0500

Updated time

Code pipeline dev:

aws Services Search [Option+S] N. Virginia

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

Developer Tools

CodePipeline

Source • CodeCommit

Artifacts • CodeArtifact

Build • CodeBuild

Deploy • CodeDeploy

Pipeline • CodePipeline

Getting started

Pipelines

Settings

Go to resource

Developer Tools > CodePipeline > Pipelines

Introducing the new V2 pipeline type with improved release safety, pipeline triggers, parameterized pipelines, and a new billing model. Learn more

Pipelines Info

Notify View history Release change Delete pipeline Create pipeline

Name Latest execution status Latest source revisions Latest execution started Most recent executions

ReactAppPipeline

(Type: V1 | Execution mode: SUPERSEDED)

Succeeded

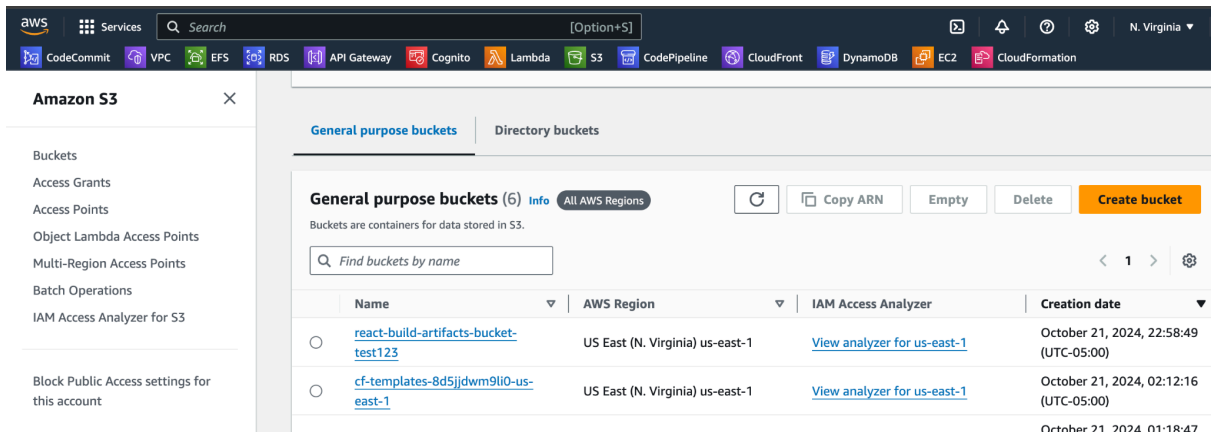
GitHub_Source - 0862a919 updated build file

19 minutes ago

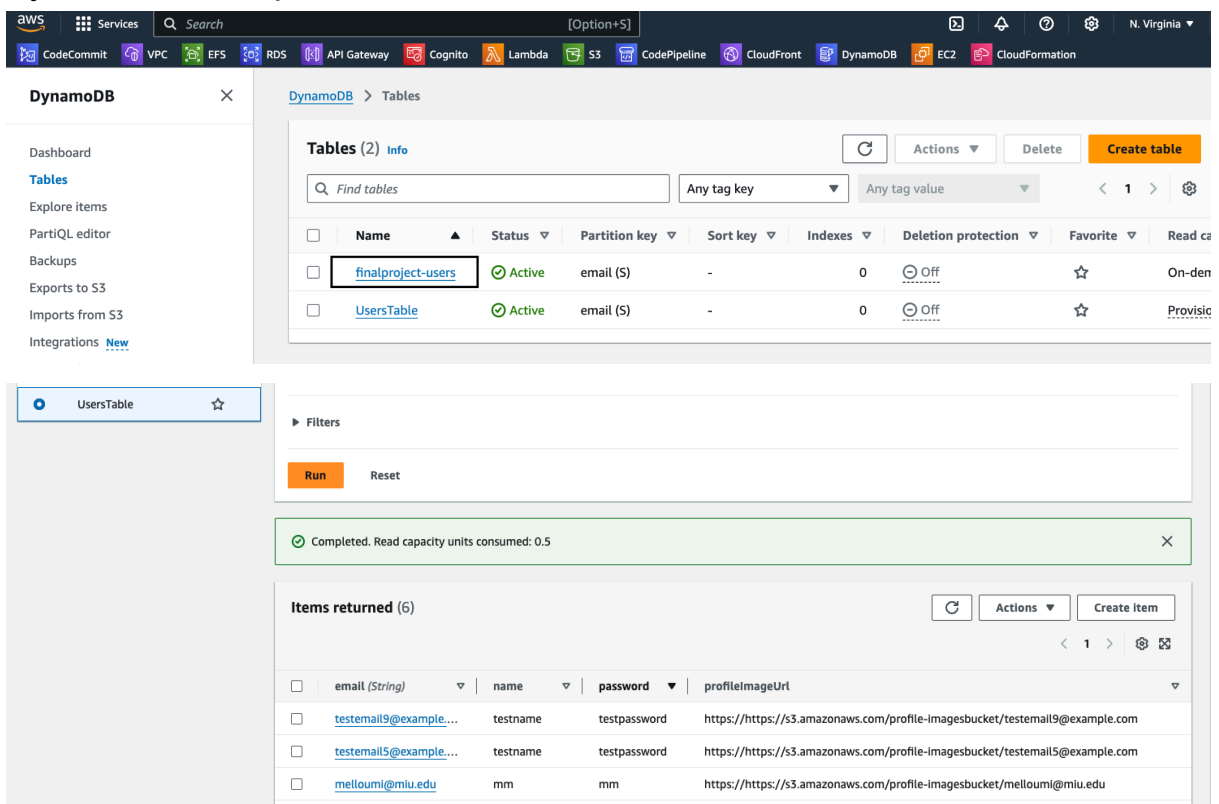
View details

Item selection ReactAppPipeline is not selected

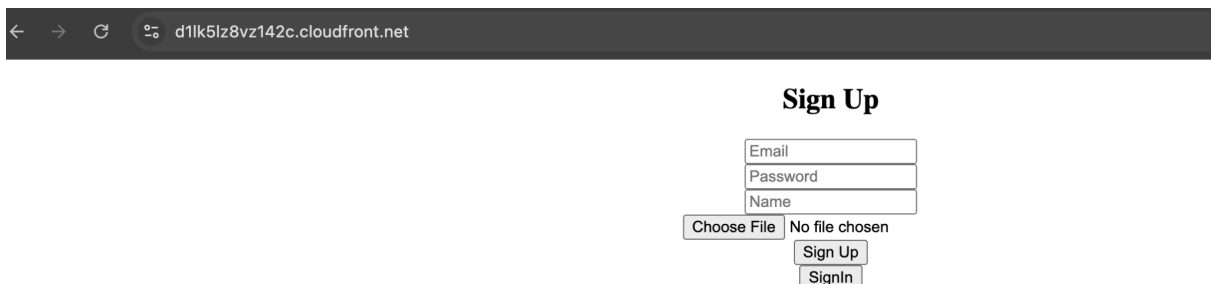
Deployment buckets:



DynamoDB setup:



User interface:



← → ↻ d1k5lz8vz142c.cloudfront.net

Sign Up

Click on SignUp to be directed to the Login Page:

← → ↻ d1k5lz8vz142c.cloudfront.net/login

Login

Email:
Password:

← → ↻ d1k5lz8vz142c.cloudfront.net/login


Login

Email:
Password:

Click on the Login button to move to Home page:

← → ↻ d1k5lz8vz142c.cloudfront.net/home


Welcome to the Home Page

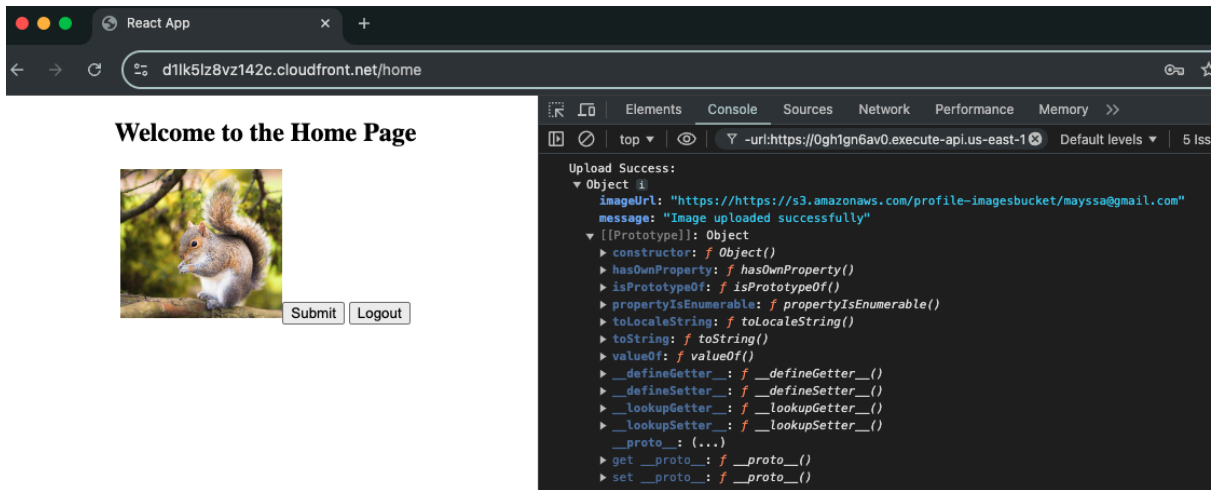


Click on Submit to update on DynamoDB:

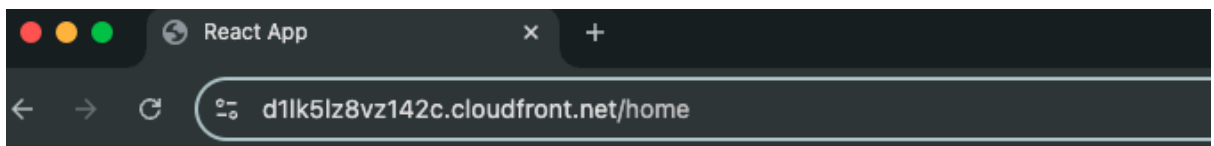
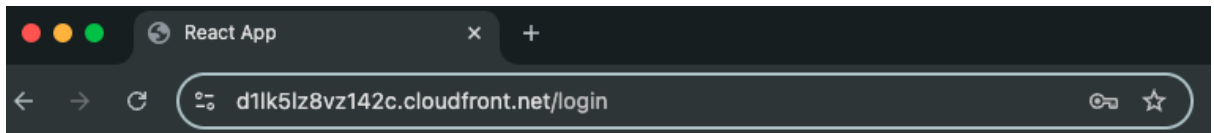
← → ↻ d1k5lz8vz142c.cloudfront.net/home

Welcome to the Home Page





Click on Logout to move to SignIn Page:



Welcome to the Home Page



Step 2: Cloud Formation Resource Deployment:

Cloud Pipeline based Frontend React CI/CD:

CloudFormation > Stacks > testCP1

Stacks (2)

Filter by stack name

Filter status

Active

View nested

Stacks

testCP1

2024-10-22 00:24:48 UTC-0400

CREATE_COMPLETE

testCF1

2024-10-21 21:39:39 UTC-0400

UPDATE_COMPLETE

testCP1

Delete Update Stack actions Create stack

Stack info Events Resources Outputs Parameters Template Change sets Git sync

Outputs (2)

Search outputs

Key	Value	Description	Export name
PipelineName	ReactAppPipeline	Name of the CodePipeline	-
WebsiteURL	http://front-buckets3.s3-website-us-east-1.amazonaws.com	The URL of the static website hosted in S3	-

Deployment of backend lambda functions, API gateway, DynamoDB:

CloudFormation > Stacks > testCF1

Stacks (2)

Filter by stack name

Filter status

Active

View nested

Stacks

testCP1

2024-10-22 00:24:48 UTC-0400

CREATE_COMPLETE

testCF1

2024-10-21 21:39:39 UTC-0400

UPDATE_COMPLETE

testCF1

Delete Update Stack actions Create stack

Stack info Events Resources Outputs Parameters Template Change sets Git sync

Outputs (2)

Search outputs

Key	Value	Description	Export name
ApiGatewayURL	https://0gh1gn6av0.execute-api.us-east-1.amazonaws.com/dev	API Gateway URL	-
CloudFrontURL	d1k5lz8vz142c.cloudfront.net	The CloudFront URL to access the static website	-

Note: All IAM roles are policies are also included in the YAML template file.

Step 3: CI/CD 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:

Objects (3) Info

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

<input type="checkbox"/>	Name ▲	Type ▼	Last modified ▼	Size ▼	Storage class ▼
<input type="checkbox"/>	 asset-manifest.json	json	October 22, 2024, 12:11:50 (UTC-04:00)	369.0 B	Standard
<input type="checkbox"/>	 index.html	html	October 22, 2024, 12:11:50 (UTC-04:00)	600.0 B	Standard
<input type="checkbox"/>	 static/	Folder	-	-	-