

CI/CD Final Project — End-to-End Pipeline on AWS

Prof : Aagam Jhaveri

Team Memebers : Sumanth Reddy K

:Shivansh Kumar

Git Repo Link :

[**https://github.com/SumanthReddyKConestoga/CICDFinalProject**](https://github.com/SumanthReddyKConestoga/CICDFinalProject)

Teams Recording Link :

https://teams.microsoft.com/l/meetingrecap?driveId=b%21pWcxUqrJZUKOJ-tbFthA0rLWcHqKS8JCro6mNzOA0gUTz4tkXHsMQpY2pw_f2dy&driveItemId=01US4QXR6T4PHG74OYVFEISJO_VLE6JUP5U&sitePath=https%3A%2F%2Fstuconestogacon-my.sharepoint.com%2F%3Av%3A%2Fg%2Fpersonal%2Fskonannagari0660_conestogac_on_ca%2FEdPjzm_x2KlliSX_VWTvaP7QB6-pJLHI_INQC4Uc8bZlu2w&fileUrl=https%3A%2F%2Fstuconestogacon-my.sharepoint.com%2F%3Av%3A%2Fg%2Fpersonal%2Fskonannagari0660_conestogac_on_ca%2FEdPjzm_x2KlliSX_VWTvaP7QB6-pJLHI_INQC4Uc8bZlu2w&threadId=19%3A581a6b67-f20c-4be3-a471-4620d6a607aa_86265a36-5959-4da0-a21a-456e1fbe039d%40unq.gbl.spaces&callId=9507f5be-36ee-4e36-94b3-089c46edabc2&threadType=OneOnOneChat&meetingType=Unknown&subType=RecapSharingLink_RecapChiclet

The screenshot shows a GitHub Actions run summary for a CI/CD pipeline. The pipeline was triggered via push 5 hours ago by SumanthReddyKConestoga. The status is Success, and the total duration was 2m 3s. One artifact was produced.

Jobs:

- test-backend
- build-and-push
- tf-validate-and-apply

Run details:

- Usage
- Workflow file

ci-cd.yml on push

Artifacts: Produced during runtime

Name	Size	Digest
backend-coverage	21.9 KB	sha256:df11933e541baadc827cb2417e4ee6bb00a3ddde8dc78895ccb37e51644bdcd



Deploy to AWS | Actions secrets - Sumanth | Actions secrets - Sumanth | ec2-3-81-43-211.compute | SumanthReddyKConestoga | updated - SumanthReddy | + | - | X

github.com/SumanthReddyKConestoga/CICDFinalProject/actions/runs/17015077522/job/48236276035

YouTube Maps Gmail News Translate GitHub - parthpanc... TCS Next step - Goo... Welcome: Sumanth... All Bookmarks

updated #61

test-backend succeeded 5 hours ago in 11s

Search logs

Jobs

- test-backend** (green)
- build-and-push
- tf-validate-and-apply

Run details

Usage

Workflow file

test-backend succeeded 5 hours ago in 11s

Set up job 1s

Run actions/checkout@v4 0s

Run actions/setup-node@v4 1s

Install backend deps 3s

Tests (with coverage) 2s

```

1 ► Run cd backend
12
13 > cl-cd-final-backend@1.0.0 test
14 > cross-env NODE_ENV=test node --experimental-vm-modules ./node_modules/jest/bin/jest.js --runInBand --coverage
15
16 (node:21135) ExperimentalWarning: VM Modules is an experimental feature and might change at any time
17 (Use `node --trace-warnings ...` to show where the warning was created)
18 PASS tests/api/_md_utils.test.js
19 PASS tests/api/test.js
20 PASS tests/bml/test.js
21 -----
22 File  | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #
23 -----|-----|-----|-----|-----|-----
24 All files | 45.74 | 41.37 | 28.57 | 59.6 |
25 dh.js | 4.76 | 59 | 0 | 5.26 | 12-75
26 index.js | 57.53 | 40 | 59 | 64.86 | 22-24,27-28,66-68,84-100,186-116,122
27 -----
28
29 Test Suites: 3 passed, 3 total
30 Tests: 6 passed, 6 total
31 Snapshots: 0 total
32 Time: 1.169 s
33 Ran all test suites.

```

Upload backend coverage 1s

CI/CD (App → ECR → Terraform/EC2)

updated #61

Re-run all jobs

test-backend succeeded 5 hours ago in 11s

Search logs

Jobs

- test-backend** (green)
- build-and-push
- tf-validate-and-apply

Run details

Usage

Workflow file

test-backend succeeded 5 hours ago in 11s

Set up job 1s

Run actions/checkout@v4 0s

Run actions/setup-node@v4 1s

Install backend deps 3s

Tests (with coverage) 2s

Upload backend coverage 1s

```

1 ► Run actions/upload-artifact@v4
16 With the provided path, there will be 11 files uploaded
17 Artifact name is valid!
18 Root directory input is valid!
19 Beginning upload of artifact content to blob storage
20 Uploaded bytes 22388
21 Finished uploading artifact content to blob storage!
22 SHA256 digest of uploaded artifact zip is df11933a541baadcb27c2417e4ee0b000a3dd8d8c78895cb037a518440cd
23 Finalizing artifact upload
24 Artifact backend-coverage.zip successfully finalized. Artifact ID 37812698087
25 Artifact backend-coverage has been successfully uploaded! Final size is 22388 bytes. Artifact ID is 37812698087
26 Artifact download URL: https://github.com/SumanthReddyKConestoga/CICDFinalProject/actions/runs/17015077522/artifacts/37812698087

```

Post Run actions/setup-node@v4 0s

Post Run actions/checkout@v4 0s

Complete job 0s

News for you Severe storm ris...

Deploy to AWS | Actions secrets - Sumanth... | Actions secrets - Sumanth... | ec2-3-81-43-211.compute... | SumanthReddyKConestoga | updated - SumanthReddyK | + | - | X

github.com/SumanthReddyKConestoga/CICDFinalProject/actions/runs/17015077522/job/48236280500

YouTube Maps Gmail News Translate GitHub - parthpanc... TCS Next step - Goo... Welcome: Sumanth... All Bookmarks

SumanthReddyKConestoga / CICDFinalProject Actions Projects Wiki Security Insights Settings

CICD (App → ECR → Terraform/EC2) updated #61

Summary

Jobs

- test-backend
- build-and-push**
- tf-validate-and-apply

Run details

Usage

Workflow file

build-and-push succeeded 5 hours ago in 33s

Re-run all jobs

Search logs

Set up job

Run actions/checkout@v4

Configure AWS credentials

```
1 ▶ Run aws-actions/configure-aws-credentials@v4
2 with:
3   aws-access-key-id: ***
4   aws-secret-access-key: ***
5   aws-region: ***
6   audience: sts.amazonaws.com
7   env:
8     AWS_REGION: ***
9     AWS_ACCOUNT_ID: ***
10    ECR_REGISTRY: ***.dkr.ecr.***.amazonaws.com
11    BACKEND_REPO: cicd-final-backend
12    FRONTEND_REPO: cicd-final-frontend
13    IMAGE_TAG: latest
14 Proceeding with IAM user credentials
```

Login to Amazon ECR

Ensure ECR repositories exist

Build & push backend

Build & push frontend

News for you Severe storm ris...

Deploy to AWS | Actions secrets - Sumanth... | Actions secrets - Sumanth... | ec2-3-81-43-211.compute... | SumanthReddyKConestoga | updated - SumanthReddyK | + | - | X

github.com/SumanthReddyKConestoga/CICDFinalProject/actions/runs/17015077522/job/48236280500

YouTube Maps Gmail News Translate GitHub - parthpanc... TCS Next step - Goo... Welcome: Sumanth... All Bookmarks

SumanthReddyKConestoga / CICDFinalProject Actions Projects Wiki Security Insights Settings

CICD (App → ECR → Terraform/EC2) updated #61

Summary

Jobs

- test-backend
- build-and-push**
- tf-validate-and-apply

Run details

Usage

Workflow file

build-and-push succeeded 5 hours ago in 33s

Re-run all jobs

Search logs

Configure AWS credentials

```
4   aws-access-key-id: ***
5   aws-region: ***
6   audience: sts.amazonaws.com
7   env:
8     AWS_REGION: ***
9     AWS_ACCOUNT_ID: ***
10    ECR_REGISTRY: ***.dkr.ecr.***.amazonaws.com
11    BACKEND_REPO: cicd-final-backend
12    FRONTEND_REPO: cicd-final-frontend
13    IMAGE_TAG: latest
14 Proceeding with IAM user credentials
```

Login to Amazon ECR

```
1 ▶ Run aws-actions/amazon-ecr-logIn@v2
2 with:
3   mask-password: true
4   registry-type: private
5   skip-logout: false
6   env:
7     AWS_REGION: ***
8     AWS_ACCOUNT_ID: ***
9     ECR_REGISTRY: ***.dkr.ecr.***.amazonaws.com
10    BACKEND_REPO: cicd-final-backend
11    FRONTEND_REPO: cicd-final-frontend
12    IMAGE_TAG: latest
13    AWS_DEFAULT_REGION: ***
14    AWS_ACCESS_KEY_ID: ***
15    AWS_SECRET_ACCESS_KEY: ***
16 Logging into registry ***.dkr.ecr.***.amazonaws.com
```

Ensure ECR repositories exist

Build & push backend

Build & push frontend

News for you Severe storm ris...

Deploy to AWS | Actions secrets - Sumanth... | Actions secrets - Sumanth... | ec2-3-81-43-211.compute... | SumanthReddyKConestoga | updated - SumanthReddyK | + | - | X

github.com/SumanthReddyKConestoga/CICDFinalProject/actions/runs/_/48236280500

YouTube Maps Gmail News Translate GitHub - parthpanc... TCS Next step - Goo... Welcome: Sumanth... All Bookmarks

SumanthReddyKConestoga / CICDFinalProject Actions Projects Wiki Security Insights Settings

CICD (App → ECR → Terraform/EC2) updated - SumanthReddyK

Re-run all jobs

Search logs

Set up job

Run actions/checkout@v4

Configure AWS credentials

```
1 ▶ Run aws-actions/configure-aws-credentials@v4
2 with:
3   aws-access-key-id: ***
4   aws-secret-access-key: ***
5   aws-region: ***
6   audience: sts.amazonaws.com
7   env:
8     AWS_REGION: ***
9     AWS_ACCOUNT_ID: ***
10    ECR_REGISTRY: ***.dkr.ecr.***.amazonaws.com
11    BACKEND_REPO: cicd-final-backend
12    FRONTEND_REPO: cicd-final-frontend
13    IMAGE_TAG: latest
14 Proceeding with IAM user credentials
```

Login to Amazon ECR

```
1 ▶ Run aws-actions/amazon-ecr-logIn@v2
2 with:
3   mask-password: true
4   registry-type: private
5   skip-logout: false
6   env:
7     AWS_REGION: ***
8     AWS_ACCOUNT_ID: ***
9     ECR_REGISTRY: ***.dkr.ecr.***.amazonaws.com
10    BACKEND_REPO: cicd-final-backend
11    FRONTEND_REPO: cicd-final-frontend
12    IMAGE_TAG: latest
13    AWS_DEFAULT_REGION: ***
14    AWS_ACCESS_KEY_ID: ***
15    AWS_SECRET_ACCESS_KEY: ***
16 Logging into registry ***.dkr.ecr.***.amazonaws.com
```

Ensure ECR repositories exist

Build & push backend

Build & push frontend

News for you Severe storm ris...

Deploy to AWS | Actions secrets - Sumanthi | Actions secrets - Sumanthi | ec2-3-81-43-211.compute | SumanthReddyKConestoga | updated - SumanthReddy | + | - | X

github.com/SumanthReddyKConestoga/CICDFinalProject/actions/runs/17015077522/job/48236280500

YouTube | Maps | Gmail | News | Translate | GitHub - parthpanc... | TCS Next step - Goo... | Welcome: Sumanth...

All Bookmarks

Summary

Jobs

- test-backend
- build-and-push**
- tf-validate-and-apply

Run details

Usage

Workflow file

build-and-push

succeeded 5 hours ago in 33s

```

37
38 # [internal] load build context
39 # transferring context: 15.498B done
40 # DONE 0.0s
41
42 # [1/1] FROM docker.io/library/nginx:alpine@sha256:2459b38ed006e999c252d374550c91490060bbfb335fa8029bf0e310b095
43 # resolve docker.io/library/nginx:alpine@sha256:2459b38ed006e999c252d374550c91490060bbfb335fa8029bf0e310b095 done
44 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
45 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
46 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
47 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
48 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
49 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
50 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
51 # extracting sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
52 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
53 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
54 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
55 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
56 # extracting sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
57 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
58 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
59 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
60 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
61 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
62 # extracting sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
63 # sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
64 # extracting sha256:0e04ad4084080480d8c5d5989405188879373a056a36534835434085a7f53752575a256a777985a6921654097985 done
65 # sha256:c01ff4086f82493a4b0802ec71bfe092ca25b505f302ec78da9770b97895356cb 9.409B / 16.840B 0.6s done
66 # extracting sha256:c01ff4086f82493a4b0802ec71bfe092ca25b505f302ec78da9770b97895356cb 9.409B / 16.840B 0.6s done
67 # extracting sha256:c01ff4086f82493a4b0802ec71bfe092ca25b505f302ec78da9770b97895356cb 9.409B / 16.840B 0.6s done
68 # extracting sha256:c01ff4086f82493a4b0802ec71bfe092ca25b505f302ec78da9770b97895356cb 9.409B / 16.840B 0.6s done
69 # extracting sha256:c01ff4086f82493a4b0802ec71bfe092ca25b505f302ec78da9770b97895356cb 9.409B / 16.840B 0.6s done
70 # sha256:c01ff4086f82493a4b0802ec71bfe092ca25b505f302ec78da9770b97895356cb 9.409B / 16.840B 0.6s done
71 # extracting sha256:c01ff4086f82493a4b0802ec71bfe092ca25b505f302ec78da9770b97895356cb 9.409B / 16.840B 0.6s done
72 # DONE 1.s
    
```

News for you
Severe storm ris...

Deploy to AWS | Actions secrets - Sumanthi | Actions secrets - Sumanthi | ec2-3-81-43-211.compute | SumanthReddyKConestoga | updated - SumanthReddy | + | - | X

github.com/SumanthReddyKConestoga/CICDFinalProject/actions/runs/17015077522/job/48236280500

YouTube | Maps | Gmail | News | Translate | GitHub - parthpanc... | TCS Next step - Goo... | Welcome: Sumanth...

All Bookmarks

Summary

Jobs

- test-backend
- build-and-push**
- tf-validate-and-apply

Run details

Usage

Workflow file

build-and-push

succeeded 5 hours ago in 33s

```

84 # 1.819 rpm notice
85 # DONE 1.9s
86
87 # [5/5] COPY src ./src
88 # DONE 0.4s
89
90 #!! exporting to image
91 #!! exporting layers
92 #!! exporting layers 0.4s done
93 #!! writing image sha256:d9fd0d95ca0a2540cf948a6a8d48a1e599630fca7e1f30b770b9f09fc60d1c4 done
94 #!! naming to ***.dkr.ecr.***.amazonaws.com/cicd-final-backend:latest done
95 #!! DONE 0.4s
96 # docker push ***.dkr.ecr.***.amazonaws.com/cicd-final-backend:latest
97 The push refers to repository [***.dkr.ecr.***.amazonaws.com/cicd-final-backend]
98 4e9c20baebcd: Preparing
99 41de9713ef08: Preparing
100 34937110bc2f: Preparing
101 34937110bc2f: Waiting
102 2121009a7b07: Preparing
103 f9a4080cb1c1: Preparing
104 001346575d61: Preparing
105 0000019316d1: Preparing
106 f304080cb1c1: Waiting
107 001260575d61: Waiting
108 0000018d16d1: Waiting
109 82140d97a7: Layer already exists
110 f304080cb1c1: Layer already exists
111 001260575d61: Layer already exists
112 0000018d16d1: Layer already exists
113 4a9c20baebcd: Pushed
114 00c09246c031: Pushed
115 34937110bc2f: Pushed
116 41de9713ef08: Pushed
117 latest digest: sha256:f93f45c97b6c781b0fb4f763e95a00003314ec200064a9ca097ea09b4c60a65bd3 size: 1992
    
```

Build & push frontend

Rainy days ahead 22°C

Deploy to AWS | Actions secrets - Sumanth | Actions secrets - Sumanth | ec2-3-81-43-211.compute | SumanthReddyKConestoga | updated - SumanthReddy | + | - | X

github.com/SumanthReddyKConestoga/CICDFinalProject/actions/runs/17015077522/job/48236291815

YouTube Maps Gmail News Translate GitHub - parthpanc... TCS Next step - Goo... Welcome: Sumanth... All Bookmarks

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

CI/CD (App → ECR → Terraform/EC2) updated #61

Re-run all jobs

Summary

Jobs

- test-backend
- build-and-push
- tf-validate-and-apply**

Run details

Usage

Workflow file

tf-validate-and-apply succeeded 5 hours ago in 1m 8s

Set up job

```

1 Current runner version: '2.327.1'
2 ▶ Runner Image Provisioner
3 Hosted Compute Agent
4 Version: 20250812.370
5 Commit: 4a202b75750004e3e907c2150c8cabe342a3da32
6 Build Date: 2025-08-12T16:08:14Z
7 ▶ Operating System
8 Ubuntu
9 24.04.2
10
11 ▶ Runner Image
12 Image: ubuntu-24.04
13 Version: 20250810.1.0
14 Included Software: https://github.com/actions/runner-images/blob/ubuntu24/20250810.1/images/ubuntu/Ubuntu%2024.04-Readme.md
15 Image Release: https://github.com/actions/runner-images/releases/tag/ubuntu24%2F20250810.1
16 GITHUB_TOKEN Permissions
17 Contents: read
18 Metadata: read
19 Secret source: Actions
20 Prepare workflow directory
21 Prepare all required actions
22 Getting action download info
23 Download action repository "actions/checkout@v4" (SHA:00ebab027e200071cdedf949ebbe08a4986955)
24 Download action repository "hashicorp/setup-terraform@3" (SHA:19bd5d4a3c349c3f98e8801555d81cc2e209883d4)
25 Download action repository "aws-actions/configure-aws-credentials@v4" (SHA:13d06a429d7300a6ac190c126e071d42e09343502)
26 Complete job name: tf-validate-and-apply

```

Run actions/checkout@v4

Setup Terraform

```

1 ▶ Run hashicorp/setup-terraform@v3
2 with:
3   terraform_repos: false
4   cli_config_credentials_hostname: app.terraform.io
5   terraform_version: latest
6   env:
7     AWS_REGION: ***
8     AWS_ACCOUNT_ID: ***
9     ECR_REGISTRY: "***.dkr.ecr.***.amazonaws.com"
10    BACKEND_REPO: cicd-final-backend
11    FRONTEND_REPO: cicd-final-frontend
12    IMAGE_TAG: latest
13    /usr/bin/true > - /home/runner/work/_temp/cdb8505d-7e91-4760-87aa-a64bf4d90620

```

Configure AWS credentials

```

1 ▶ Run aws-actions/configure-aws-credentials@v4
2 with:
3   aws-access-key-id: ***
4   aws-secret-access-key: ***
5   aws-region: ***

```

https://github.com/SumanthReddyKConestoga/CICDFinalProject/actions/runs/17015077522/job/48236291815#step:3:4

Rainy days ahead 22°C

Search

ENG US 2:36 AM 2025-08-17

Deploy to AWS | Actions secrets - Sumanth | Actions secrets - Sumanth | ec2-3-81-43-211.compute | SumanthReddyKConestoga | updated - SumanthReddy | + | - | X

github.com/SumanthReddyKConestoga/CICDFinalProject/actions/runs/17015077522/job/48236291815

YouTube | Maps | Gmail | News | Translate | GitHub - parthpanc... | TCS Next step - Goo... | Welcome: Sumanth... | All Bookmarks

Summary

tf-validate-and-apply succeeded 5 hours ago in 1m 8s

Jobs

- test-backend
- build-and-push
- tf-validate-and-apply**

Run details

Usage

Workflow file

tf-validate-and-apply

```

14 [
15   "UserId": "****",
16   "Account": "****",
17   "Arn": "arn:aws:iam::****:root"
18 ]

```

Terraform fmt (auto-fix)

Terraform validate

```

1 ▶ Run cd infra/terraform
2 Initializing modules...
3 - m2 in module/ec2
4 Initializing provider plugins...
5 - Reusing previous version of hashicorp/aws from the dependency lock file
6 - Reusing previous version of hashicorp/random from the dependency lock file
7 - Installing hashicorp/aws v5.100.0...
8 - Installed hashicorp/aws v5.100.0 (signed by HashiCorp)
9 - Installing hashicorp/random v3.7.2...
10 - Installed hashicorp/random v3.7.2 (signed by HashiCorp)
11 Terraform has made some changes to the provider dependency selections recorded
12 in the .terraform.lock.hcl file. Review those changes and commit them to your
13 version control system if they represent changes you intended to make.
14
15 Terraform has been successfully initialized!
16
17 You may now begin working with Terraform. Try running "terraform plan" to see
18 any changes that are required for your infrastructure. All Terraform commands
19 should now work.
20
21 If you ever set or change modules or backend configuration for Terraform,
22 rerun this command to reinitialize your working directory. If you forget, other
23 commands will detect it and remind you to do so if necessary.
24 Success! The configuration is valid.
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39

```

Terraform plan

Rainy days ahead 22°C

Deploy to AWS | Actions secrets - Sumanth | Actions secrets - Sumanth | ec2-3-81-43-211.compute | SumanthReddyKConestoga | updated - SumanthReddy | + | - | X

github.com/SumanthReddyKConestoga/CICDFinalProject/actions/runs/17015077522/job/48236291815

YouTube | Maps | Gmail | News | Translate | GitHub - parthpanc... | TCS Next step - Goo... | Welcome: Sumanth... | All Bookmarks

Summary

tf-validate-and-apply succeeded 5 hours ago in 1m 8s

Jobs

- test-backend
- build-and-push
- tf-validate-and-apply**

Run details

Usage

Workflow file

tf-validate-and-apply

```

29 aws_s3_bucket_public_access_block.storage: Creating...
30 aws_s3_bucket_public_access_block.storage: Creation complete after 0s [id=cld-final-storage-5c921b]
31 aws_im_instance_profile.ec2.profile: Creation complete after 6s [id=ec2-app-profile-20250817012513910000000002]
32 aws_vpc_main: Still creating... [0m00s elapsed]
33 aws_vpc_main: Creation complete after 13s [id=vpc-00314f91b4d456990]
34 aws_internet_gateway.igp: Creating...
35 aws_subnet_public_1: Creating...
36 aws_subnet_public_2: Creating...
37 aws_route_table.public: Creating...
38 aws_security_group.app: Creating...
39 aws_internet_gateway.igp: Creation complete after 0s [id=igw-0714d7b97ff81c4914]
40 aws_route_table.public: Creation complete after 0s [id=rtb-0ce000000000000000000000000000000]
41 aws_route_definition.routetab: Creating...
42 aws_route_definition.routetab: Creation complete after 1s [id=rtrb-0be000000000000000000000000000000]
43 aws_security_group.app: Creation complete after 3s [id=sg-004f31fd6305fe099]
44 aws_subnet_public_1: Still creating... [0m00s elapsed]
45 aws_subnet_public_2: Still creating... [0m00s elapsed]
46 aws_subnet_public_2: Creation complete after 1s [id=subnet-0ff97a2e17428e92e]
47 aws_route_table_association.public_assoc_2: Creating...
48 aws_subnet_public_1: Creation complete after 1s [id=subnet-0641bae9ad16d42]
49 aws_route_table_association.public_assoc_1: Creating...
50 module.ec2.aws_instance.app: Creating...
51 aws_route_table_association.public_assoc_1: Creation complete after 1s [id=rtrbsoc-0af52fd021b84d79d]
52 aws_route_table_association.public_assoc_2: Creation complete after 2s [id=rtrbsoc-0f33950be8409eb09]
53 module.ec2.aws_instance.app: Still creating... [0m00s elapsed]
54 module.ec2.aws_instance.app: Creation complete after 14s [id=i-0ecb710e5d098caea]
55
56 Apply complete! Resources: 17 added, 0 changed, 0 destroyed.
57
58 Outputs:
59
60 app_url = "http://ec2-54-161-137-101.compute-1.amazonaws.com:3000"
61 instance_id = "i-0ecb710e5d098caea"
62 public_dns = "ec2-54-161-137-101.compute-1.amazonaws.com"
63 public_ip = "54.161.137.101"
64 storage_bucket = "cld-final-storage-5c921b"

```

Screenshot of a GitHub Actions pipeline run titled "tf-validate-and-apply". The pipeline has three jobs: "test-backend", "build-and-push", and "tf-validate-and-apply". The "tf-validate-and-apply" job has completed successfully. The logs for this job show the output of a Terraform apply command:

```

55
56 Apply complete! Resources: 17 added, 0 changed, 0 destroyed.
57
58 Outputs:
59
60 app_url = "https://ec2-54-161-137-101.compute-1.amazonaws.com:3000"
61 instance_id = "i-0eb7d0e5609caea"
62 public_dns = "ec2-54-161-137-101.compute-1.amazonaws.com"
63 public_ip = "54.161.137.101"
64 storage_bucket = "cicd-final-storage-5c921b"

```

The pipeline also includes a "Show outputs" section with the following content:

```

1 1 Run echo "App URL: $APP_URL"
2 echo "$App URL: $APP_URL"
3 echo "Public IP: $PUBLIC_IP"
4 shell: /usr/bin/bash -e {0}
5 env:
6 AWS_REGION: ***
7 AWS_ACCOUNT_ID: ***
8 ECR_REGISTRY: ***.dkr.ecr.***.amazonaws.com
9 BACKEND_REPO: cicd-final-backend
10 FRONTEND_REPO: cicd-final-frontend
11 AWS_DEFAULT_REGION: ***
12 AWS_ACCESS_KEY_ID: ***
13 AWS_SECRET_ACCESS_KEY: ***
14 APP_URL: http://ec2-54-161-137-101.compute-1.amazonaws.com:3000
15 PUBLIC_IP: 54.161.137.101
16 App URL: https://ec2-54-161-137-101.compute-1.amazonaws.com:3000
17 Public IP: 54.161.137.101
18
19 Post Configure AWS credentials
20 Post Run actions/checkout@v4
21
22 Complete job

```

Below the pipeline, the Docker Desktop interface shows the "cicdfinalproject" container running. The logs for this container show MySQL startup and configuration messages:

```

2025-08-16 21:46:45 frontend-1 | 2025/08/17 01:46:45 [notice] 1#1: start worker process 37
2025-08-16 21:46:45 frontend-1 | 2025/08/17 01:46:45 [notice] 1#1: start worker process 38
2025-08-16 21:46:45 frontend-1 | 2025/08/17 01:46:45 [notice] 1#1: start worker process 39
2025-08-16 21:46:45 frontend-1 | 2025/08/17 01:46:45 [notice] 1#1: start worker process 40
2025-08-16 21:46:45 frontend-1 | 2025/08/17 01:46:45 [notice] 1#1: start worker process 41
2025-08-16 21:46:33 db-1 | 2025/08/17 01:46:33+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.43-1.c19 started.
2025-08-16 21:46:34 db-1 | 2025/08/17 01:46:34+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
2025-08-16 21:46:34 db-1 | 2025/08/17 01:46:34+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.43-1.c19 started.
2025-08-16 21:46:34 db-1 | '/var/lib/mysql/mysql.sock' -> '/var/run/mysql/mysql.sock'
2025-08-16 21:46:35 db-1 | 2025-08-17T01:46:35.170909Z 0 [Warning] [MY-011668] [Server] The syntax '--skip-host-cache' is deprecated and will be removed in a future release. Please use SET GLOBAL host_cache_size=0 instead.
2025-08-16 21:46:35 db-1 | 2025-08-17T01:46:35.174405Z 0 [System] [MY-011161] [Server] /usr/sbin/mysqld (mysqld 8.0.43) starting as process 1
2025-08-16 21:46:35 db-1 | 2025-08-17T01:46:35.187817Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2025-08-16 21:46:35 db-1 | 2025-08-17T01:46:35.645091Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2025-08-16 21:46:36 db-1 | 2025-08-17T01:46:36.074993Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.
2025-08-16 21:46:36 db-1 | 2025-08-17T01:46:36.075067Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encrypted connections are now supported for this channel.
2025-08-16 21:46:36 db-1 | 2025-08-17T01:46:36.080945Z 0 [Warning] [MY-011810] [Server] Insecure configuration for --pid-file: Location '/var/run/mysql' in the path is accessible to all OS users. Consider choosing a different directory.
2025-08-16 21:46:36 db-1 | 2025-08-17T01:46:36.121366Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8.0.43' socket: '/var/run/mysql/mysql.sock' port: 3306 MySQL Community Server - GPL.
2025-08-16 21:46:36 db-1 | 2025-08-17T01:46:36.121400Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Bind-address: '::' port: 33060, socket: '/var/run/mysql/mysql.sock'

```

The Docker desktop status bar indicates the engine is running, RAM usage is 7.28 GB, CPU usage is 2.69%, and disk usage is 10.98 GB.

Docker Desktop interface showing a running container named **cicdfinalproject-backend-1**. The container ID is `da99867ec478` and it was created from the image `cicdfinalproject-backend:latest`. The status is **Running (5 hours ago)**. The logs show the message `2025-08-16 21:46:45 API listening on 3000`.

Below the Docker Desktop window, a screenshot of a Windows taskbar shows multiple browser tabs open, including GitHub, AWS Lambda, and a terminal window. The terminal window shows a successful Terraform plan execution:

```
tf validate-and-apply
succeeded 5 hours ago in 1m 6s

Terraform plan
451
452
453
454 Saved the plan to: tfplan
455
456 To perform exactly these actions, run the following command to apply:
457   terraform apply "tfplan"

Terraform apply
Show outputs

1 ▶ Run echo "APP_URL: $APP_URL"
2 echo "App URL: $APP_URL"
3 echo "Public IP: $PUBLIC_IP"
4 shell: /usr/bin/bash -e {0}
5 exit
6 AWS_REGION: ***
7 AWS_ACCOUNT_ID: ***
8 ECS_REGISTRY: ***.dkr.ecr.***.amazonaws.com
9 BACKEND_REPO: cicd-final-backend
10 FRONTEND_REPO: cicd-final-frontend
11 IMAGE_TAGS: latest
12 AWS_DEFAULT_REGION: ***
13 AWS_ACCESS_KEY_ID: ***
14 AWS_SECRET_ACCESS_KEY: ***
15 APP_URL: http://ec2-54-161-137-101.compute-1.amazonaws.com:3000
16 PUBLIC_IP: 54.161.137.101
17 App URL: http://54-161-137-101.compute-1.amazonaws.com:3000
18 Public IP: 54.161.137.101

Post Configure AWS credentials
Post Run actions/checkout@v4
Complete job
```

The screenshot shows the GitHub Actions secrets settings page. On the left, a sidebar lists various GitHub features like Branches, Tags, Rules, Actions, Models, Webhooks, Copilot, Environments, Codespaces, and Pages. Under the 'Actions' section, 'Secrets and variables' is expanded, showing 'Actions', 'Codespaces', and 'Dependabot'. Below this, 'Integrations' includes 'GitHub Apps' and 'Email notifications'. The main area is divided into 'Environment secrets' (which displays a message: 'This environment has no secrets.' and a 'Manage environment secrets' button) and 'Repository secrets' (which lists several secrets with their names, last updated times, and edit/delete icons). A green button at the top right says 'New repository secret'.

Name	Last updated
AWS_ACCESS_KEY_ID	18 hours ago
AWS_ACCOUNT_ID	yesterday
AWS_DEFAULT_REGION	19 hours ago
AWS_REGION	19 hours ago
AWS_SECRET_ACCESS_KEY	18 hours ago
DB_PASSWORD	2 days ago
ECR_REPO_URL	19 hours ago
ECR_REPO_URL_FRONTEND	18 hours ago
TF_STATE_BUCKET	2 days ago
TF_STATE_TABLE	2 days ago

The screenshot shows the GitHub repository page for 'CICDFinalProject'. The top navigation bar includes links for Deploy to AWS, SumanthReddyKConestoga, Actions secrets, ec2-3-81-43-211.compute, SumanthReddyKConestoga, and updated - SumanthReddy. The repository name 'SumanthReddyKConestoga / CICDFinalProject' is visible. The main interface shows a 'Code' tab selected, displaying the repository's codebase. A dropdown menu for 'init/scaffold' is open, showing a list of branches: default, init/scaffold, demo/change, demo/pipeline, feat/ci, feat/o-ready, feat/ec2-deploy, feat/ncr-push, feat/tac-skeleton, feat/images, and feat/pipelines. The repository has 14 branches and 0 tags. To the right, there are sections for 'About' (with a note: 'No description, website, or topics provided.'), 'Releases' (with a note: 'No releases published'), 'Packages' (with a note: 'No packages published'), and 'Languages' (showing a breakdown: JavaScript 56.8%, CSS 23.2%, HTML 15.9%, HCL 3.1%, and Dockerfile 1.0%).

Deploy to AWS | Branches - SumanthReddy | Actions secrets - Sumanth... | ec2-3-81-43-211.compute... | SumanthReddyKConestoga | updated - SumanthReddy | + | - | X | [github.com/SumanthReddyKConestoga/CICDFinalProject/branches](#)

YouTube | Maps | Gmail | News | Translate | GitHub - parthpanc... | TCS Next step - Goo... | Welcome: Sumanth... | All Bookmarks

Default

Branch	Updated	Check status	Behind Ahead	Pull request
init/scaffold	3 days ago	✓ 2 / 2	Default	#5

Your branches

Branch	Updated	Check status	Behind Ahead	Pull request
main	5 hours ago	✓ 3 / 3	0 85	#18
feat/ec2-deploy	2 days ago	✓ 4 / 4	0 24	#17
demo/pipeline	2 days ago	✓ 2 / 2	0 20	#16
demo/change	2 days ago	✓ 7 / 7	0 19	#15
feat/pipelines	2 days ago	✓ 6 / 6	0 18	#14

[View more branches >](#)

Active branches

Branch	Updated	Check status	Behind Ahead	Pull request
main	5 hours ago	✓ 3 / 3	0 85	#18
feat/ec2-deploy	2 days ago	✓ 4 / 4	0 24	#17
demo/pipeline	2 days ago	✓ 2 / 2	0 20	#16
demo/change	2 days ago	✓ 7 / 7	0 19	#15
feat/pipelines	2 days ago	✓ 6 / 6	0 18	#14

[View more branches >](#)

ATL - CLE Video highlight | Search | [X](#) [File](#) [Edit](#) [View](#) [Insert](#) [Format](#) [Tools](#) [Help](#) | ENG US | 2:39 AM | 2025-08-17

Deploy to AWS | Workflow runs - Sumanth... | Actions secrets - Sumanth... | ec2-3-81-43-211.compute... | SumanthReddyKConestoga | updated - SumanthReddy | + | - | X | [github.com/SumanthReddyKConestoga/CICDFinalProject/actions](#)

YouTube | Maps | Gmail | News | Translate | GitHub - parthpanc... | TCS Next step - Goo... | Welcome: Sumanth... | All Bookmarks

SumanthReddyKConestoga / CICDFinalProject

Actions | New workflow | Code | Issues | Pull requests | 3 | All workflows

All workflows

Showing runs from all workflows

244 workflow runs

Event	Status	Branch	Actor
updated	5 hours ago	main	⌚ 2m 3s
updated	5 hours ago	main	⌚ 1m 55s
updated	5 hours ago	main	⌚ Failure

Deploy to AWS | Main by SumanthReddyKC... | Actions secrets - Sumanth... | ec2-3-81-43-211.compute... | SumanthReddyKConestoga | updated - SumanthReddy... | + | - | X

github.com/SumanthReddyKCConestoga/CICDFinalProject/pull/18

YouTube | Maps | Gmail | News | Translate | GitHub - parthpanc... | TCS Next step - Goo... | Welcome: Sumanth... | All Bookmarks

Main #18

SumanthReddyKCConestoga commented yesterday

No description provided.

SumanthReddyKCConestoga added 30 commits 3 days ago

- Merge pull request #5 from SumanthReddyKCConestoga/init/scaffold
- cil(images): build images (dry-run) + upload artifacts
- Merge pull request #6 From SumanthReddyKCConestoga/feat/images
- cil(images): push to ECR with AWS creds
- Merge pull request #7 From SumanthReddyKCConestoga/feat/ecr-push
- iac(deploy): ECS/ALB Terraform + CI validate + ECR build (clean history)
- Merge pull request #8 From SumanthReddyKCConestoga/feat/deploy-clean
- deploy: add terraform apply stage (VDP)
- Merge pull request #9 From SumanthReddyKCConestoga/feat/deploy-apply
- cil(deploy): enable deploy with session token
- Merge pull request #10 From SumanthReddyKCConestoga/fix/aws-session-token
- deploy: use latest image tag to avoid ECS pull race
- Merge pull request #11 From SumanthReddyKCConestoga/fix/deploy-latest

Owner: SumanthReddyKCConestoga

Reviewers: SumanthReddyKCConestoga

Suggestions: Copilot

Request: Still in progress? Convert to draft

Assignees: No one—assign yourself

Labels: None yet

Projects: None yet

Milestones: None

Development: None yet

Successfully merging this pull request may close these issues.

Notifications: 1319add Notifications | Unsubscribe

Customize

ATL - CLE Video highlight

Search

CloudShell Feedback

Trending videos Avatar: Fire and...

54.146.129.13 | ec2-54-146-129-13.compute-1.amazonaws.com | updated - Sumanth... | updated - Sumanth... | ec2-54-161-137-205.2025-08-17 | - | X

YouTube | Maps | Gmail | News | Translate | GitHub - parthpanc... | TCS Next step - Goo... | Welcome: Sumanth... | All Bookmarks

Account ID: 0197-7393-0547 - shivansh

United States (N. Virginia)

EC2 Instances > i-0f76bdf8e4b6cf58c

Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores
Auto Scaling
Auto Scaling Groups
Settings

CloudShell Feedback

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ENG US 2:42 AM 2025-08-17

SumanthReddy | Calls | Meeting | Instances | EC2 | 54.146.129.13 | ec2-54-146-129-13 | updated - Sumanth | updated - Sumanth | ec2-54-161-137 | + | - | X

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#InstancesinstanceState=running&sort=desc&platform=details

YouTube Maps Gmail News Translate GitHub - parthpan... TCS Next step - Goo... Welcome: Sumanth...

All Bookmarks Account ID: 0197-7393-0547 shivansh United States (N. Virginia)

EC2 > Instances

Instances (1/4) Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
ec2-app-instance	i-0f76bdf8e4b6cf58c	Running	t3.micro	3/3 checks passed	View alarms +	us-east-1a	ec2-54-146-129-13.co...	54.146.129.13
ec2-app-instance	i-07cc5c13b2b9090fe	Running	t3.micro	3/3 checks passed	View alarms +	us-east-1a	ec2-3-81-43-211.comp...	3.81.43.211
ec2-app-instance	i-0eb710e5d698cae	Running	t3.micro	3/3 checks passed	View alarms +	us-east-1a	ec2-54-161-137-101.co...	54.161.137.101
ec2-app-instance	i-09fafc6324888771b	Running	t3.micro	3/3 checks passed	View alarms +	us-east-1a	ec2-100-24-10-2.com...	100.24.10.2

i-0f76bdf8e4b6cf58c (ec2-app-instance)

Instance ID: i-0f76bdf8e4b6cf58c

Public IPv4 address: 54.146.129.13 [open address]

Instance state: Running

Private IP DNS name (IPv4 only): ip-10-0-10-207.ec2.internal

Instance type: t3.micro

VPC ID: vpc-04f54a0e5990d5d0 (ec2-app-vpc)

Subnet ID: subnet-050df3f91e33a2d73 (ec2-app-public-subnet-1)

Instance ARN: arn:aws:ec2:us-east-1:019773930547:instance/i-0f76bdf8e4b6cf58c

Private IPv4 addresses: 10.0.10.207

Public DNS: ec2-54-146-129-13.compute-1.amazonaws.com [open address]

Elastic IP addresses: -

AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations. [Learn more]

Auto Scaling Group name: -

Managed: false

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG US 243 AM 2025-08-17

SumanthReddy | Calls | Meeting | SecurityGroups | 54.146.129.13 | ec2-54-146-129-13 | updated - Sumanth | updated - Sumanth | ec2-54-161-137 | + | - | X

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#SecurityGroups:

YouTube Maps Gmail News Translate GitHub - parthpan... TCS Next step - Goo... Welcome: Sumanth...

All Bookmarks Account ID: 0197-7393-0547 shivansh United States (N. Virginia)

EC2 > Security Groups

Security Groups (1/11) Info

Name	Security group ID	Security group name	VPC ID	Description	Owner
sg-00277310a162a228es	default	vpc-00277310a162a228es	-	default VPC security group	019773930547
ec2-app-alb-sg	sg-02655d6c950781ed	ec2-app-alb-sg	vpc-04f54a0e5990d5d0	Allow HTTP to ALB	019773930547
-	sg-0aa786ec879db25d	default	vpc-04f54a0e5990d5d0	default VPC security group	019773930547
ec2-app-sg	sg-0dd4f31f46305fb89	ec2-app-sg	vpc-003314f91b4656990	Allow HTTP and SSH	019773930547
-	sg-06779386dd02c9bc9	default	vpc-04c8c6a78fd39af77	default VPC security group	019773930547

sg-0774c393185a0ec62 - ec2-app-sg

Inbound rules (2/2)

Name	Security group rule ID	IP version	Type	Protocol	Port range	Source	Description
-	sgr-04627065b07be5878	IPv4	Custom TCP	TCP	3000	0.0.0.0/0	HTTP (app)
-	sgr-01d1537189d73794b	IPv4	SSH	TCP	22	0.0.0.0/0	SSH from your IP

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG US 242 AM 2025-08-17

SumanthReddy | Calls | Meeting | Load balancers | 54.146.129.13 | ec2-54-146-129 | updated - Sumanth | updated - Sumanth | ec2-54-161-137 | + | - | X

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LoadBalancers:

YouTube Maps Gmail News Translate GitHub - parthpanc... TCS Next step - Goo... Welcome: Sumanth...

All Bookmarks United States (N. Virginia) Account ID: 0197-7393-0547 shivansh

EC2 > Load balancers

Load balancers (1/1)

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Actions Create load balancer

Load balancer: ec2-app-alb

Details Listeners and rules Network mapping Resource map Security Monitoring Integrations Attributes Capacity Tags

Details

Load balancer type	Status	Load balancer IP address type
Application	Active	IPv4
Scheme	Hosted zone	Date created
Internet-facing	Z355X0DTRQ7X7K	August 16, 2025, 16:41 (UTC-04:00)
Load balancer ARN		DNS name info
arn:aws:elasticloadbalancing:us-east-1:019773930547:loadbalancer/app/ec2-app-alb/2c7e56ef0a0e56c		ec2-app-alb-813300711.us-east-1.elb.amazonaws.com (A Record)

CloudShell Feedback 22°C Partly cloudy 244 AM 2025-08-17

SumanthReddy | Calls | Meeting | Target groups | 54.146.129.13 | ec2-54-146-129 | updated - Sumanth | updated - Sumanth | ec2-54-161-137 | + | - | X

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#TargetGroups:

YouTube Maps Gmail News Translate GitHub - parthpanc... TCS Next step - Goo... Welcome: Sumanth...

All Bookmarks United States (N. Virginia) Account ID: 0197-7393-0547 shivansh

EC2 > Target groups

Target groups (1/8)

Details

Name	ARN	Port	Protocol	Target type	Load balancer	VPC ID
ec2a-20250816202759...	arn:aws:elasticloadbalancin...	80	HTTP	Instance	None associated	vpc-06bb45659504cc7c
ec2a-20250816204113...	arn:aws:elasticloadbalancin...	80	HTTP	Instance	ec2-app-alb	vpc-04f54a0e599d0d5d0
ec2b-20250816202759...	arn:aws:elasticloadbalancin...	3000	HTTP	Instance	None associated	vpc-06bb45659504cc7c
ec2b-20250816204113...	arn:aws:elasticloadbalancin...	3000	HTTP	Instance	ec2-app-alb	vpc-04f54a0e599d0d5d0

Target group: ec2a-2025081620411302190000000a

Details

Target type	Protocol : Port	Protocol version	VPC
Instance	HTTP: 80	HTTP1	vpc-04f54a0e599d0d5d0
IP address type	Load balancer		
IPv4	ec2-app-alb		

1 Total targets	<input checked="" type="radio"/> 0 Healthy	<input checked="" type="radio"/> 1 Unhealthy	<input type="radio"/> 0 Unused	<input type="radio"/> 0 Initial	<input type="radio"/> 0 Draining
	0 Anomalous				

Distribution of targets by Availability Zone (AZ)

Select values in this table to see corresponding filters applied to the Registered targets table below.

CloudShell Feedback 22°C Partly cloudy 244 AM 2025-08-17

SumanthReddy/ Calls | Meeting Listener details | 54.146.129.13 | ec2-54-146-129 | updated - Sumanth Reddy | updated - Sumanth Reddy | ec2-54-161-137 | + - | X

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#ELBLListenerV2:loadBalancerArn=arn:aws:elasticloadbalancing:us-east-1:019773930547:loadbalancer/app/ec2-app-alb/2c7e56ef0ac0e56c/93aae51d88891d1

YouTube Maps Gmail News Translate GitHub - parthpanch... TCS Next step - Goo... Welcome: Sumanth Reddy All Bookmarks Account ID: 0197-7393-0547 shivansh United States (N. Virginia)

EC2 > Load balancers > ec2-app-alb > HTTP:80 listener

HTTP:80 Info

Details
A listener checks for connection requests using the protocol and port that you configure. The default action and any additional rules that you create determine how the Application Load Balancer routes requests to its registered targets.

Protocol:Port	Load balancer	Default actions		
HTTP:80	ec2-app-alb	<ul style="list-style-type: none"> Forward to target group ec2-2025081620411302190000000a 1 (100%) Target group stickiness: Off 		
<ul style="list-style-type: none"> Forward to target group ec2-2025081620411302190000000a 1 (100%) Target group stickiness: Off 				

Listener ARN
arn:aws:elasticloadbalancing:us-east-1:019773930547:listener/app/ec2-app-alb/2c7e56ef0ac0e56c/93aae51d88891d1

Rules **Attributes** **Tags**

Listener rules (2) Info
Traffic received by the listener is routed according to the default action and any additional rules. Rules are evaluated in priority order from the lowest value to the highest value.

Priority	Name tag	Conditions (If)	Actions (Then)	ARN	Tags	Actions	
100	-	Path = /api/*	<ul style="list-style-type: none"> Forward to target group ec2-20250816204113021900000009 1 (100%) Target group stickiness: Off 	ARN	0 tags		
Last (default)	Default	If no other rule applies	<ul style="list-style-type: none"> Forward to target group ec2-2025081620411302190000000a 1 (100%) Target group stickiness: Off 	ARN	0 tags		

CloudShell Feedback 22°C Partly cloudy 244 AM 2025-08-17 © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences Account ID: 0197-7393-0547 shivansh United States (N. Virginia)

SumanthReddy/ Calls | Meeting Load balancers | 54.146.129.13 | ec2-54-146-129 | updated - Sumanth Reddy | updated - Sumanth Reddy | ec2-54-161-137 | + - | X

YouTube Maps Gmail News Translate GitHub - parthpanch... TCS Next step - Goo... Welcome: Sumanth Reddy All Bookmarks Account ID: 0197-7393-0547 shivansh United States (N. Virginia)

EC2 > Load balancers

Load balancers (1/1)

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Name	Type	VPC ID	Availability Zones	DNS name	Date created
ec2-app-alb	application	vpc-04f54a0e599e0d5...	2 Availability Zones	ec2-app-alb-813300711.us...	August 16, 2025, 16:41 (UTC-04:00)

Load balancer: ec2-app-alb

Listeners and rules (1) Info
A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

Protocol:Port	Default action	Rules	ARN	Security policy	Default SSL/TLS certificate	mTLS	Trust store
HTTP:80	<ul style="list-style-type: none"> Forward to target group ec2-2025081620411302190000000a 1 (100%) Target group stickiness: Off 	2 rules	ARN	Not applicable	Not applicable	Not applicable	Not applicable

CloudShell Feedback 22°C Partly cloudy 244 AM 2025-08-17 © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences Account ID: 0197-7393-0547 shivansh United States (N. Virginia)

The screenshot displays two separate browser windows for AWS services.

AWS S3 Console:

- Left Sidebar:** Shows navigation links for General purpose buckets, Directory buckets, Table buckets, Vector buckets, Access Grants, Access Points (General Purpose Buckets, FSx file systems), Access Points (Directory Buckets), Object Lambda Access Points, Multi-Region Access Points, Batch Operations, and IAM Access Analyzer for S3.
- Right Content Area:** Shows the "General purpose buckets" section with four buckets listed:

Name	AWS Region	Creation date
cfd-templates-1tmezkk6i2ai3-us-east-1	US East (N. Virginia) us-east-1	July 17, 2025, 20:17:11 (UTC-04:00)
cld-final-storage-4b35f	US East (N. Virginia) us-east-1	August 16, 2025, 21:20:26 (UTC-04:00)
cld-final-storage-5c921b	US East (N. Virginia) us-east-1	August 16, 2025, 21:25:14 (UTC-04:00)
cld-final-storage-d47f5a	US East (N. Virginia) us-east-1	August 16, 2025, 18:25:00 (UTC-04:00)

AWS VPC Console:

- Left Sidebar:** Shows navigation links for EC2 Global View, Virtual private cloud (Your VPCs, Subnets, Route tables, Internet gateways, Egress-only Internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, NAT gateways, Peering connections, Route servers), Security (Network ACLs, Security groups), PrivateLink and Lattice (Getting started, Endpoints), and Endpoint services.
- Right Content Area:** Shows the "Your VPCs" section with five VPCs listed:

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP option set	Main route table
ec2-app-vpc	vpc-04c86a78fd39af77	Available	10.0.0.0/16	-	dopt-04702cd9a68552...	rtb-0d9c4b74ea...
ec2-app-vpc	vpc-035c79dfcc4db1d6	Available	10.0.0.0/16	-	dopt-04702cd9a68552...	rtb-0b2bd4ec...
ec2-app-vpc	vpc-04f54a0e599d0d5d0	Available	10.0.0.0/16	-	dopt-04702cd9a68552...	rtb-0f57a9f558...
-	vpc-0066cc90e85c338bf	Available	172.31.0.0/16	-	dopt-04702cd9a68552...	rtb-0u0339a0...

CICD FINAL PROJECT

Age & BMI Studio • Built by Sumanth&Shivansh

Your name Date of birth Height (cm) Weight (kg)

Sumanth yyyy-mm-dd 175 72 Calculate Reset

years
months
days

Days Hours Minutes Seconds

BMI

Enter height & weight for insights tailored to your age.

Recent Calculations

Name	DOB	Y	M	D	Height	Weight	BMI	Status	When
Sumanth	2000-03-06	25	5	12	175	72	25.38	Overweight	2025-08-17

CICD FINAL PROJECT

Age & BMI Studio • Built by Sumanth&Shivansh

Your name Date of birth Height (cm) Weight (kg)

Sumanth Reddy 2000-03-06 183 85 Calculate Reset

25 years
5 months
12 days

Days 9,295 Hours 223,086 Minutes 13,385,213 Seconds 803,112,832

25.38 BMI Overweight

Move more; manage portions. Age 25.

Recent Calculations

Name	DOB	Y	M	D	Height	Weight	BMI	Status	When
Sumanth Reddy	2000-03-06	25	5	12	183	85	25.38	Overweight	2025-08-17

The screenshot displays two browser tabs side-by-side, both running the "Age & BMI Studio" application.

localhost:8090 Tab:

- Recent Calculations:**

Name	DOB	Y	M	D	Height	Weight	BMI	Status	When
Sumanth Reddy	2000-03-06	25	5	11	183.00	85.00	25.38	Overweight	8/17/2025, 5:55:50 AM
shivansh	2008-02-05	17	6	9	152.00	52.00	22.51	Normal	8/14/2025, 6:47:50 PM
prem bro	1996-01-01	29	7	13	175.00	90.00	29.39	Overweight	8/14/2025, 11:41:40 AM
sumanth	2000-03-06	25	5	8					8/14/2025, 10:50:28 AM
- Recent Calculations:** Shows the same four entries as the table above.

ec2-3-81-43-211.compute-1.amazonaws.com:3000 Tab:

- Form Fields:**
 - Your name: Sumanth
 - Date of birth: yyyy-mm-dd (placeholder)
 - Height (cm): 175
 - Weight (kg): 72
 - Buttons: Calculate, Reset
- Output Area:**
 - Years: [empty input]
 - Months: [empty input]
 - Days: [empty input]
 - Time Components: Days, Hours, Minutes, Seconds (each with a minus sign)
 - Note: Enter height & weight for insights tailored to your age.
- Recent Calculations:** Shows the same four entries as the other tab.

CI/CD Final Project — End-to-End Pipeline on AWS

1) What we built (overview)

Goal: Build and demonstrate a complete CI/CD pipeline that automatically builds, tests, packages, provisions infrastructure, and deploys a 3-tier web app to AWS.

Stack (simple, familiar components):

- **Frontend:** Nginx serving static files; proxies requests under /api/* to the backend.
- **Backend:** Node.js (v18) REST API (listens on 0.0.0.0:3000).
- **Database:** MySQL 8.0.
- **Containers:** Docker for each component.
- **Local orchestration:** Docker Compose (frontend, backend, db).
- **Cloud:** Amazon EC2 (Amazon Linux), Docker + Compose on the instance.
- **Registry:** Amazon ECR (stores frontend and backend images).
- **IaC:** Terraform (EC2, IAM instance profile, Security Group, outputs).
- **CI/CD:** GitHub Actions (tests → build & push to ECR → Terraform validate/plan/apply).
- **Access/ops:** AWS Systems Manager (SSM) for automation and logs; no SSH keys required.

2) Application details (meets “Application Functionality”)

Frontend

- Serves index.html, app.js, style.css.
- Nginx config proxies /api/* to backend:3000.

Backend

- Node.js/Express.
- Example health route: GET /api/health → 200 OK.
- Environment variables: APP_PORT, DB_HOST, DB_USER, DB_PASSWORD, DB_NAME.

Database

- MySQL 8.0.

- Credentials provided via Compose env and .env file.
- Exposed **only internally** on AWS (no public port).

Local run (single command)

```
# from the project root
docker compose -f ./docker-compose.local.yml up -d --build
# open http://localhost:8090
```

3) Infrastructure as Code (meets “Infrastructure Definition”)

Terraform structure (infra/terraform/):

- **Resources:**
 - aws_instance for EC2 (Amazon Linux).
 - aws_iam_role + aws_iam_instance_profile with:
 - AmazonSSMManagedInstanceCore
 - AmazonEC2ContainerRegistryReadOnly
 - aws_security_group allowing inbound **TCP 80** from 0.0.0.0/0.
- **Outputs:**
 - public_ip
 - app_url (e.g., http://<public-ip>)

Key variables declared (examples):

- aws_region (default us-east-1)
- instance_type (default t3.micro)
- app_port (default 3000)
- ecr_repo_url (backend image repo URL)
- image_tag (default latest)

- app_name, vpc_cidr (sane defaults)

Validation gates in CI:

- terraform fmt (we auto-fix in CI for demo smoothness)
- terraform validate
- terraform plan (saved to tfplan)
- terraform apply tfplan

Why this design: short provisioning time, simple public access on port 80, IAM-based access to SSM/ECR (no static keys on the EC2 box).

4) Version control & workflow (meets “Version Control”)

- Repo layout:
 - frontend/ (Dockerfile, nginx.conf)
 - backend/ (Dockerfile, tests)
 - infra/terraform/
 - .github/workflows/deploy.yml
- Branching:
 - main as release branch.
 - Feature branches for changes.
 - **Pull Requests:** ≥ 5 with reviews (screenshots kept).
- Commits:
 - Frequent, small commits per component (≥ 5 per component).

5) CI/CD pipeline (meets “CI/CD Pipeline Implementation”)

Trigger: Pushes to main.

Jobs & stages:

1. Test (backend)

- npm ci
- npm test -- --coverage
- Uploads coverage artifact.

2. Build & Push

- Configure AWS credentials from repo secrets.
- Login to ECR.
- Ensure ECR repos exist (cicd-final-backend, cicd-final-frontend).
- Build Docker images and push:
 - \${ACCOUNT}.dkr.ecr.\${REGION}.amazonaws.com/cicd-final-backend:latest
 - \${ACCOUNT}.dkr.ecr.\${REGION}.amazonaws.com/cicd-final-frontend:latest

3. Terraform (Validate → Plan → Apply)

- terraform fmt -recursive -write=true
- terraform validate
- terraform plan -out=tfplan
- terraform apply tfplan
- Print APP_URL and PUBLIC_IP for demo.

Secrets needed (GitHub → Settings → Secrets and variables → Actions):

- AWS_ACCESS_KEY_ID, AWS_SECRET_ACCESS_KEY
- AWS_ACCOUNT_ID
- AWS_REGION (e.g., us-east-1)

Automated triggers: Every push to main starts the full pipeline.

Bonus (optional scanning):

- Add CodeQL or Trivy step before the build for code / image scanning (we tested this locally; simple to drop in).

6) Monitoring and logs (meets “Monitoring & Logs”)

- **CloudWatch/EC2 Monitoring:** basic CPU and network graphs on the EC2 instance page.
- **Containers/logs:** via SSM Run Command or Session Manager shells:
 - docker compose ps
 - docker compose logs -f frontend
 - docker compose logs -f backend
 - docker compose logs -f db
- **Troubleshooting quickly:**
 - 502 or API not reachable → confirm backend listens on 0.0.0.0:3000; check frontend/nginx.conf proxy.
 - Page not loading → Security Group open on **TCP 80**; verify docker compose up -d completed.
 - DB connection errors → ensure DB_HOST=db and give MySQL 30–60s to pass health checks.

7) Issues encountered and how we fixed them (short, factual)

1. ECR login: “400 Bad Request”

- **Cause:** Password wasn’t piped from AWS CLI to Docker.
- **Fix:**
 - (aws ecr get-login-password --region us-east-1) |
 - docker login --username AWS --password-stdin <account>.dkr.ecr.us-east-1.amazonaws.com

2. Amazon Linux 2023: curl vs curl-minimal conflict

- **Cause:** Installing curl on a host with curl-minimal created a dnf conflict.
- **Fix (in automation):**
 - `dnf -y swap curl-minimal curl || true`
 - `dnf -y install docker unzip`

3. SSM session auto-terminating

- **Cause:** Missing AmazonSSMManagedInstanceCore role or network path to SSM endpoints.
- **Fix:** Attach the role; use **Run Command** for non-interactive provisioning.

4. Terraform: “No valid credential sources found”

- **Cause:** CI job didn’t have AWS credentials.
- **Fix:** Add `aws-actions/configure-aws-credentials@v4` before any terraform command.

5. Terraform: undeclared variables

- **Cause:** Code referenced variables not declared in `variables.tf`.
- **Fix:** Declare all vars with defaults (`app_port`, `ecr_repo_url`, `app_name`, `vpc_cidr`, etc.).

6. Terraform fmt failure (exit code 3)

- **Cause:** Files not formatted.
- **Fix:** `terraform fmt -recursive` locally or let CI run with `-write=true`.

7. GitHub Actions: YAML syntax error

- **Cause:** Indentation/colon mistake.
- **Fix:** Replace with validated workflow YAML.

8. Frontend Docker build: nginx.conf not found

- **Cause:** File name typo (nginx.config); tried to run COPY in shell instead of Dockerfile.
- **Fix:** Place nginx.conf in frontend/; keep COPY in the Dockerfile.

9. Port mismatch

- **Cause:** Local frontend published 8090:80; EC2 expected port 80.
- **Fix:** On EC2 Compose, publish 80:80 only; backend internal.

10. DB connection

- **Cause:** Using 127.0.0.1 for DB inside containers.
- **Fix:** Use Compose DNS name db.

A. Show the repo

- Folders: frontend/, backend/, infra/terraform/, .github/workflows/deploy.yml.
- Branches and PRs (screenshots ready if needed).

B. Run tests locally (30–60s)

```
cd .\backend
```

```
npm ci
```

```
npm test -- --coverage
```

```
cd ..
```

C. Validate IaC locally

```
cd .\infra\terraform
```

```
terraform validate
```

```
cd ..\..
```

D. Trigger CI/CD

```
git add -A
```

```
git commit -m "demo: trigger pipeline"
```

```
git push origin main
```

- Open **GitHub Actions**: narrate stages (test → build/push → terraform).

E. Open deployed app

- EC2 → copy **Public IPv4 DNS** → open in browser (should serve frontend).
- Hit /api/health if available.

F. Show monitoring/logs briefly

- EC2 Monitoring tab (CPU/network).
- Optional: SSM Run Command → docker compose ps.

9) What we would improve next (brief)

- Move from EC2+Compose to **ECS Fargate** (managed scaling) or **EKS**.
- Add **ALB + ACM** for HTTPS and health checks.
- Move DB into **Amazon RDS**.
- Use **OIDC** for GitHub → AWS (no long-lived CI keys).
- Make **code/image scanning** (CodeQL/Trivy) a required gate.
- Add CloudWatch dashboards and alarms.

10) Command quick reference (local + cloud)

Local (start/stop):

```
docker compose -f .\docker-compose.local.yml up -d --build
```

```
start http://localhost:8090
```

```
docker compose -f .\docker-compose.local.yml down
```

CI seed (if needed):

```
$REG=<account>.dkr.ecr.us-east-1.amazonaws.com"

(aws ecr get-login-password --region us-east-1) | docker login --username AWS --
password-stdin $REG

docker build -t $REG/cicd-final-backend:latest .\backend; docker push $REG/cicd-
final-backend:latest

docker build -t $REG/cicd-final-frontend:latest .\frontend; docker push
$REG/cicd-final-frontend:latest
```

Terraform (manual):

```
cd .\infra\terraform

terraform fmt -recursive

terraform validate

terraform plan -out=tfplan

terraform apply -auto-approve tfplan
```

Summary: We delivered a working, repeatable CI/CD pipeline with tests, image builds, IaC validation, and deployment to AWS. The report lists the exact issues we hit and how we resolved them.