

Backend Continuous Deployment

Backend Continuous Deployment #6

Re-run all jobs

Summary

Jobs

- Lint
- Test
- Build

Run details

- Usage
- Workflow file

Manually triggered 3 weeks ago

Status

Total duration

Artifacts

msizafrans 602b295 main

Success

2m 28s

—

backend-cd.yaml

on: workflow_dispatch

Lint

1m 2s

Test

1m 13s

Build

57s



Backend Continuous Deployment

Backend Continuous Deployment #6

Re-run all jobs

Summary

Jobs

Lint

Test

Build

Run details

Usage

Workflow file

Lint

succeeded 3 weeks ago in 1m 2s

Search logs

- > Set up job 0s
- > Checking Out the code from Repo 0s
- > Installing dependency manager pipenv 4s
- > Cache before dependency install 1s
- > Installing dependencies 52s
- ▼ Now Running Lint 0s
 - 1 Run pipenv run lint
 - 2 pipenv run lint
 - 3 shell: /usr/bin/bash -e {0}
- > Post Cache before dependency install 0s
- > Post Checking Out the code from Repo 0s

✓ Backend Continuous Deployment #6

Re-run all jobs



🏠 Summary

Jobs

✓ Lint

✓ Test

✓ Build

Run details

🕒 Usage

📄 Workflow file

Test

succeeded 3 weeks ago in 1m 13s

🔍 Search logs



- > ✓ Set up job 1s
- > ✓ Checking Out the code from Repo 1s
- > ✓ Installing dependency manager pipenv 5s
- > ✓ Cache before dependency install 0s
- > ✓ Installing dependencies 1m 1s
- ▼ ✓ Now Running Tests 1s

```
1 ▶ Run pipenv run test
4 ===== test session starts =====
5 platform linux -- Python 3.10.12, pytest-7.4.3, pluggy-1.3.0 -- /home/runner/.local/share/virtualenvs/backend-S-kel2zF/bin/python
6 cachedir: .pytest_cache
7 rootdir: /home/runner/work/Project-Movie-Picture-Pipeline/Project-Movie-Picture-Pipeline/backend
8 collecting ... collected 3 items
9
10 test_app.py::test_movies_endpoint_returns_200 PASSED [ 33%]
11 test_app.py::test_movies_endpoint_returns_json PASSED [ 66%]
12 test_app.py::test_movies_endpoint_returns_valid_data PASSED [100%]
13
14 ===== 3 passed in 0.14s =====
```

Jobs

✓ Lint

✓ Test

✓ Build

Run details

Usage

Workflow file

Build

succeeded 3 weeks ago in 57s

Search logs



- > ✓ Set up job 1s
- > ✓ Checking Out the code from Repo 0s
- > ✓ Configure AWS credentials 0s
- > ✓ Login to Amazon ECR 1s
- > ✓ Build, tag, and push docker image to Amazon ECR 41s
- > ✓ Setup Authentication against EKS Cluster 7s
- > ✓ Run init.sh 1s

▼ ✓ Deploy the image to the cluster 2s

```
1 ▼ Run cd k8s
2   cd k8s
3   kustomize edit set image backend=***.dkr.ecr.us-east-1.amazonaws.com/backend:latest
4   kustomize build | kubectl apply -f -
5   shell: /usr/bin/bash -e {0}
6   env:
7     AWS_DEFAULT_REGION: us-east-1
8     AWS_REGION: us-east-1
9     AWS_ACCESS_KEY_ID: ***
10    AWS_SECRET_ACCESS_KEY: ***
11    REGISTRY: ***.dkr.ecr.us-east-1.amazonaws.com
12  service/backend created
13  deployment.apps/backend created
```