

Lab Report: CI/CD

Student information

- Student name: Xander Van der Linden
- Student code: 202292316

Assignment description

De opdracht draaide om het opzetten van een volledige Continuous Integration/Continuous Deployment (CI/CD) pipeline met behulp van GitHub Actions en Docker. Het doel was om de webapp met dieren van het vorige lab in een repository te plaatsen, die te testen, en automatisch te deployen in een Docker-container. Hierbij moest je een aantal stappen volgen, zoals:

- Je moest een nieuwe repository aanmaken op GitHub en de sample-applicatie daarin uploaden.
- Een workflow opzetten in GitHub Actions: Hier ging het om het automatiseren van de build- en testprocessen. Je moest meerdere workflows configureren die de code van de repository checkt, Node.js installeert, afhankelijkheden installeert, en vervolgens tests en een linter uitvoert.
- Docker image bouwen en pushen naar Docker Hub: Je moest een Docker image maken van de sample-applicatie, deze lokaal testen en vervolgens naar Docker Hub pushen, zodat anderen het kunnen gebruiken.
- Automatische triggers instellen: Telkens wanneer er een verandering in de code was (zoals een commit of push), moest de CI/CD pipeline automatisch opnieuw draaien, de code opnieuw testen en het Docker image bijwerken als alles succesvol was.
- Dependabot instellen: Je moest Dependabot activeren om automatisch pull requests te maken als een afhankelijkheid verouderd is of een update nodig heeft.

Proof of work done

github.com/xvanderlinden/cicd-sample-app/blob/main/.github/workflows/demo.yml

MessagesYouTubeNetflixHogeschool Gent ...DEP2G6 board - Agi...Data Engineering R...AI, ML & Data Engin...De StandaardHet weerbericht vo...Keras: Deep Learn...Alle bookmarks

xvanderlinden / cicd-sample-app

Type to search

CodeIssuesPull requestsActionsProjectsWikiSecurityInsightsSettings

Files

main

Go to file

.github/workflows

demo.yml

02-dockerlab

README.md

cicd-sample-app / .github / workflows / demo.yml

View Runs

xvanderlinden Added GitHub Actions demo workflow

bc76717 · 1 minute agoHistory

CodeBlame17 lines (17 loc) · 433 BytesCode 55% faster with GitHub Copilot

```
1 ---
2 name: GitHub Actions Demo
3 run-name: ${ github.actor } is testing out GitHub Actions
4 on:
5   push:
6     branches:
7       - main
8 jobs:
9   Explore-GitHub-Actions:
10    runs-on: ubuntu-latest
11    steps:
12      - name: Check out repository code
13        uses: actions/checkout@v3
14      - name: List files in the repository
15        run: |
16          ls ${ github.workspace }
17      - run: echo "This job's status is ${ job.status }."
```

xvanderlinden / cicd-sample-app

Type to search

CodeIssuesPull requestsActionsProjectsWikiSecurityInsightsSettings

Actions

New workflow

All workflows

Showing runs from all workflows

1 workflow run

EventStatusBranchActor

✓ xvanderlinden is testing out GitHub Actions

GitHub Actions Demo #1: Commit bc76717 pushed by xvanderlinden

main

8 minutes ago

15s

xvanderlinden / cicd-sample-app

Type to search

CodeIssuesPull requestsActionsProjectsWikiSecurityInsightsSettings

← GitHub Actions Demo

✓ xvanderlinden is testing out GitHub Actions #1

Re-run all jobs

Summary

Jobs

Explore-GitHub-Actions

Run details

Usage

Workflow file

Triggered via push 9 minutes ago

xvanderlinden pushed · bc76717main

Status

Success

Total duration

15s

Artifacts

-

demo.yml

on: push

✓ Explore-GitHub-Actions

3s

Annotations

1 warning

⚠ Explore-GitHub-Actions

The following actions use a deprecated Node.js version and will be forced to run on node20: actions/checkout@v3. For more info: https://github.blog/changelog/2024-03-07-github-actions-al...

Show more

The screenshot displays the GitHub Actions interface. On the left, the 'Actions' tab is selected, showing a list of workflows under 'Build and Lint'. The main area shows a table of 12 workflow runs. Two runs are visible: 'Added Docker image build step' (succeeded, 34s) and 'linting opgelost met --fix' (succeeded, 19s). Below the table, a detailed log for a 'build' workflow is shown, indicating it succeeded in 33s. The log includes steps like 'Log in to Docker Hub' and 'Build and push Docker image', with the latter showing a series of 'invalid character NUL' errors during the build process.

Evaluation criteria

- [x] Show that you created a GitHub repository for the sample application
- [x] Show the overview of workflow runs in the GitHub Actions tab
- [x] Show that the application image is available on Docker Hub
- [x] Make a change to the sample application, commit and push, and show that the build pipeline is triggered automatically
- [x] Enable dependabot and show that it creates a PR if a dependency is outdated.
- [x] Show that you wrote an elaborate lab report in Markdown and pushed it to the repository
- [x] Show that you updated the cheat sheet with the commands you need to remember

Issues

Ik ondervond heel wat problemen met de yarn lint uit te voeren. Door de oude eslintrc.json te migreren naar een nieuwe eslint.config.mjs met het commando `npx @eslint/migrate-config .eslintrc.json` en het commando `yarn eslint` te gebruiken heb ik dit opgelost. In de Build.yml heb ik dit commando dan ook aangepast.

Reflection

Buiten de problemen met de linter vond ik dit een zeer interessante opdracht aangezien ik nog nooit met github actions had gewerkt. Ik zie zeker het nut in van deze technologie en zal dit waarschijnlijk nog veel gebruiken in de toekomst. Als extra heb ik gezorgd dat de yarn test slaagde.

Resources

<https://eslint.org/docs/latest/use/configure/migration-guide> <https://github.com/features/actions>
<https://github.com/xvanderlinden/cicd-sample-app>