

GitHub: Actions, deploy



Ausgangslage:

GitHub Actions , test, build and deploy Sie kennen bereits das einfache React-Projekt <https://gitlab.com/bbwrl/m346-ref-card-02.git> Sie nutzen CI/CD von GitHub, die GitHub Action um das Projekt nach einem Commit und Push automatisiert zu bauen, testen und als Image auf Docker-Hub! abzuspeichern.

Zeitbedarf:

2 Lektionen

Hilfsmittel:

React, Json Server integriert, basamiq – mockups

Methode/Sozialform:



Lernziele:

- ✓ ReactJS – GitHub Actions
- ✓ Sie können eine CI-CD Pipeline auf GitHub mittels GitHub Action aufsetzen.
- ✓ Sie können ein Projekt automatisiert bauen, testen und ein Image bauen.
- ✓ Sie können das Image automatisiert auf Docker-Hub abspeichern.
- ✓ Sie können die Zugangsdaten für Docker-Hub als Secrets schützen.

Legende:  Einzelarbeit,  Partnerarbeit,  Dokumentation,  Code,  Präsentation

Links: <https://www.youtube.com/watch?v=iJ63nXg-LvQ>

Git Repo: <https://gitlab.com/bbw-m426-group/biztrips-2023.git>

Inhaltsverzeichnis

1	<u>VORBEREITUNGEN</u>	<u>2</u>
2	<u>AUFGABE: CI-CD, GITHUB, REACT, DOCKER HUB:.....</u>	<u>2</u>
3	<u>AUFGABE: VARIANTE CI-CD, GITHUB, REACT, GITHUB CONTAINER REGISTRY:.....</u>	<u>3</u>
4	<u>ZUSATZAUFGABE: BEST PRACTICE:</u>	<u>3</u>
	<u>.....</u>	<u>3</u>

1 Vorbereitungen

Das Ziel ist es die GitHub Actions zu verstehen, dazu gehen wir folgendermassen vor:

- Starten mit dem Quickstart
<https://docs.github.com/en/actions/quickstart>
- (überfliegen, ausprobieren)
- Notieren, was war neu

Tieferes Know How aufbauen:

- <https://docs.github.com/en/actions/learn-github-actions>
 1. Understanding GitHub Actions
 2. Finding and customizing actions
 3. Essential features of GitHub Actions
 4. Expressions
 5. Contexts
 6. Variables
 7. Using starter workflows
 8. Usage limits, billing, and administration
- Deployment:
- Deployment to Amazon ECS
- <https://docs.github.com/en/actions/deployment/deploying-to-your-cloud-provider/deploying-to-amazon-elastic-container-service>

Best practices:

- <https://docs.github.com/en/actions/security-guides/security-hardening-for-github-actions>
- <https://exercism.org/docs/building/github/gha-best-practices>

2 Aufgabe: CI-CD, github, react, docker hub:

Aufgabe:

- Nutzen Sie das oben genannte Projekt.
Wenden Sie CI-CD mit GitHub Action an und berücksichtigen Sie dabei die obigen Ziele.

Vorgehen:

- Suchen Sie ein einfaches Tutorial
zBsp: Google: "github action react dockerhub"
- Setzen Sie die Aufgabe um.

Ergebnis festhalten.

- Halten Sie die Schritte Ihres Vorgehens kurz in einem Markdown im Projekt fest.
- Beschränken Sie sich auf die Essenz, kurz knapp und verständlich.
- Ergänzen Sie ins Markdown Bilder, die Ihr Ergebnis zeigen
Achten Sie darauf, dass Ihre individuelle Lösung eindeutig erkennbar ist.
(Commits, Workflows, Dockerhub, Container, Web-Site, ...)

Erweiterung:

- Nur Veränderungen auf dem main / master branch führen dazu, dass das Image auf docker-hub landet

3 Aufgabe: Variante CI-CD, github, react, github container registry:

Aufgabe

- Alles gleich wie oben ... ausser
- Verwenden Sie die *github container registry*.

Ergebnis festhalten.

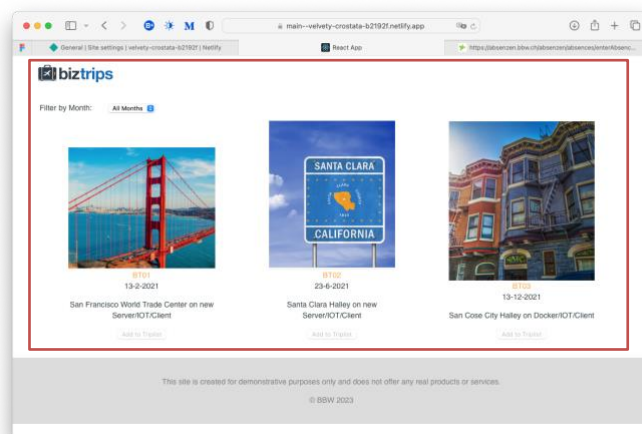
- Wie oben

4 Zusatzaufgabe: Best practice:

Recherchieren Sie nach Best practice im Umgang mit CI-CD, github actions:

- Halten Sie die Essenz in Ihrem markdown fest.

Deployete Site:



artifacts will safe be deployed