

CI-CD, github, react, aws + Versionierung

Einleitung:

Nun soll das Image der Application bei einer Änderung direkt in das Amazon Elastic Container Registry übertragen werden und dort automatisiert zum starten eines neuen Containers führen.

Sie verwenden wiederum das **einfache React-Projekt**

<https://gitlab.com/bbwrl/m346-ref-card-02.git>

Eine weitere Hilfe sind die Dokument aus dem Modul 347, sowie allenfalls das Dokument 12.2_M347_30.4_CI_CD-Pipeline_gitlab_aws.pdf

Ziele:

- Sie können eine CI-CD Pipeline auf GitHub mittels GitHub Action aufsetzen.
- Sie können ein Projekt automatisiert builden, testen und ein Image builden.
- Sie können das Image automatisiert auf **Amazon Elastic Container Registry** abspeichern.
- Sie können die Zugangsdaten für **AWS als Secrets** schützen.
- Sie können AWS so konfigurieren, dass ein geändertes Image zu einem neuen Container führt.
- Sie können die Software strukturiert versionieren und damit die Bezeichnung des Images beeinflussen.

Inhalte

Einleitung:	1
Ziele:	1
Inhalte	1
Aufgabe: Software strukturiert versionieren	1
Aufgabe: CI-CD, github, react, AWS ECR:	2
Aufgabe: Erweiterung auf AWS ECS:	2

Aufgabe: Software strukturiert versionieren

Aufgabe:

- Recherchieren Sie, auf welche Arten Software-Artefakte strukturiert, versioniert werden.
- Wählen Sie eine Art aus und versionieren Sie mit Hilfe von git. zBsp Git-Tag
- Benutzen Sie diesen Tag, um die Bezeichnung des Images zu beeinflussen.

Aufgabe: CI-CD, github, react, AWS ECR:

Aufgabe:

- Nutzen Sie das oben genannte Projekt.
Wenden Sie CI-CD mit GitHub Action an.
 - o Sie können eine CI-CD Pipeline auf GitHub mittels GitHub Action aufsetzen.
 - o Sie können ein Projekt automatisiert builden, testen und ein Image builden.
 - o Sie können das Image automatisiert auf Amazon Elastic Container Registry abspeichern.
 - o Sie können die Zugangsdaten für AWS als Secrets schützen.
- Denken Sie an die Versionierung
 - o Sie können die Software strukturiert versionieren und damit die Bezeichnung des Images beeinflussen.

Vorgehen:

- Unterlagen aus Modul 347 verwenden
- Allenfalls Tutorial suchen
- Aufgabe umsetzen
- Test mit Softwareänderungen und Versions-Tags.

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, ECR)

Aufgabe: Erweiterung auf AWS ECS:

Aufgabe:

- Nun ergänzen Sie noch den AWS Container Service ECS.
 - o Sie können AWS so konfigurieren, dass ein geändertes Image zu einem neuen Container führt.

Ergebnis festhalten.

- Ergänzen Sie ins Markdown Bilder, die Ihr Ergebnis zeigen
Achten Sie darauf, dass Ihre individuelle Lösung eindeutig erkennbar ist
(ECS, Änderung -> Wirkung))