EVENTS

tech&talk Stuttgart

Beyond Gitlab Auto DevOps

Christoph Walter
@christoph_wltr

Online

18.03.2022

codecentric Berlin

tech&talk Stuttgart

Hallo 👋

tech&talk Karlsruhe

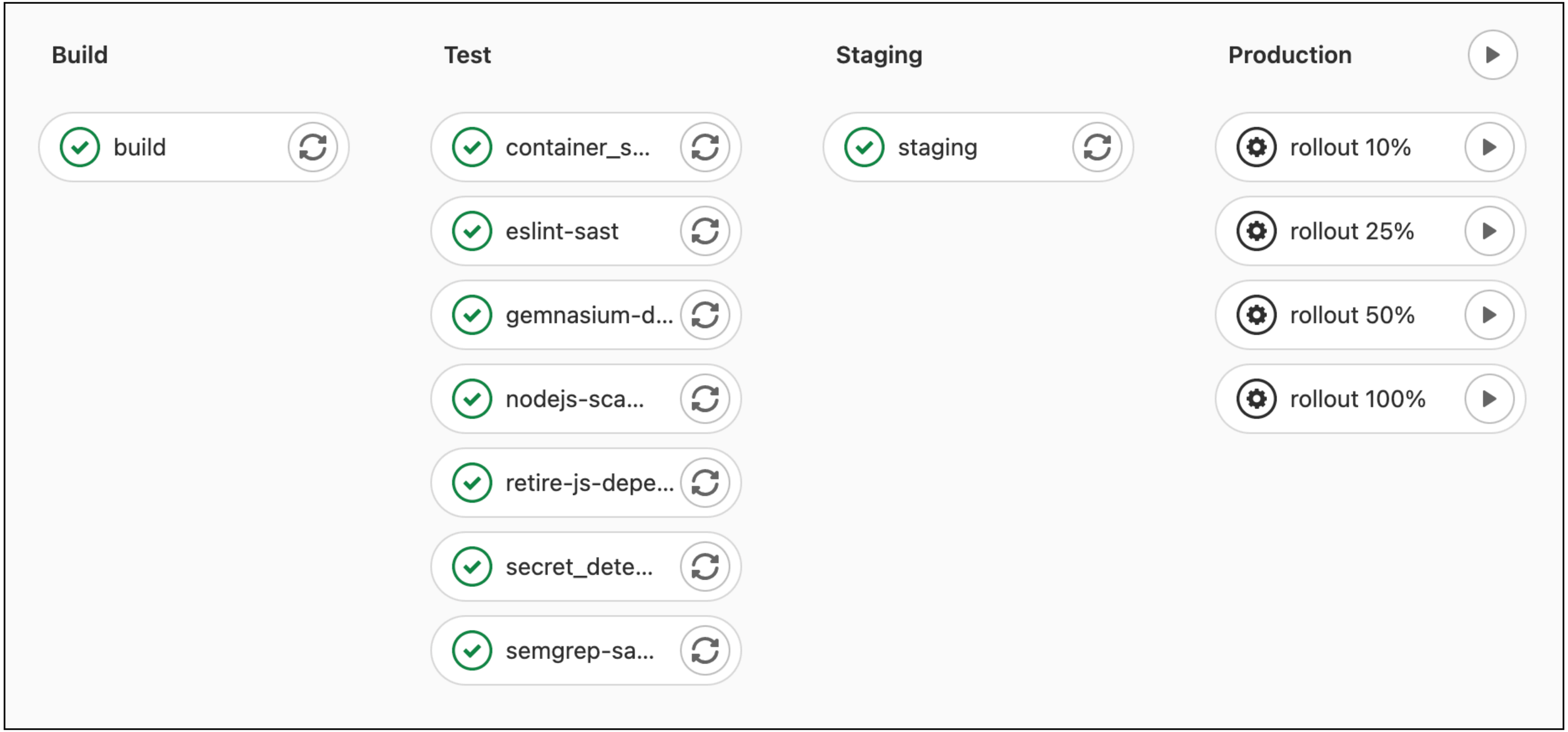
Was ist GitLab Auto DevOps?

„Commit your code and GitLab does the rest. No manual configuration.“

– GitLab Auto DevOps

Getting started

Aller Anfang ist ~~schwer~~ leicht.


Build
build
Test
container_s...
eslint-sast
gemnasium-d...
nodejs-sca...
retire-js-depe...
secret_dete...
semgrep-sa...
Staging
staging
Production
rollout 10%
rollout 25%
rollout 50%
rollout 100%

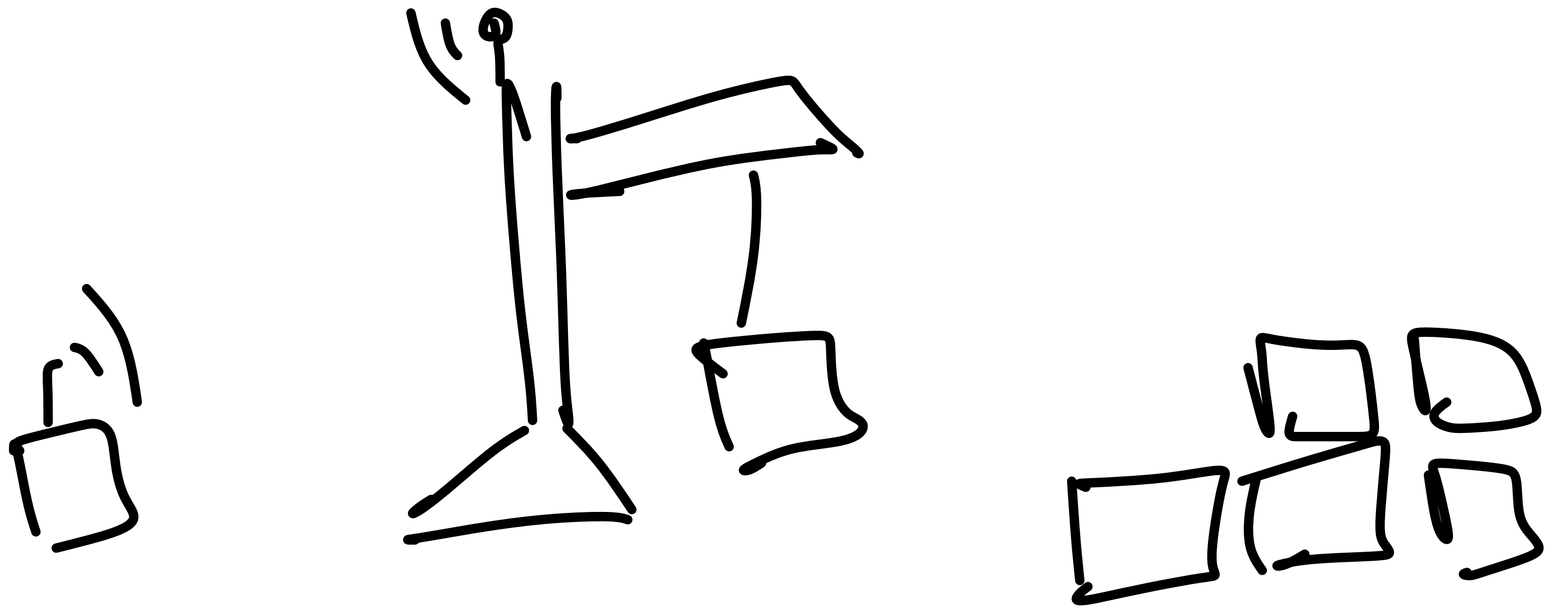
Under the hood

Wie funktioniert das eigentlich?

Configure Auto DevOps

Wenn „Auto“ nicht mehr ausreicht.

CI/CD Variablen



Configure Auto DevOps - CI/CD Variablen

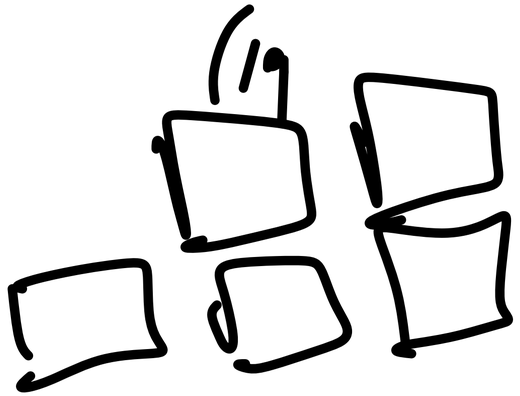
```yaml
# .gitlab-ci.yml
include:
  - template: Auto-DevOps.gitlab-ci.yml

variables:
  TEST_DISABLED: "true"
  BROWSER_PERFORMANCE_DISABLED: "true"
  CODE_QUALITY_DISABLED: "true"
  POSTGRES_ENABLED: "false"
  KUBE_INGRESS_BASE_DOMAIN: "localdev.me"
```

Auto DevOps CI/CD Variables

Type	↑ Key	Value	Protected	Masked	Environments	
Variable	POSTGRES_ENABLED	**********	✕	✕	All (default)	

Umgebungsvariablen



Configure Auto DevOps - Umgebungsvariablen

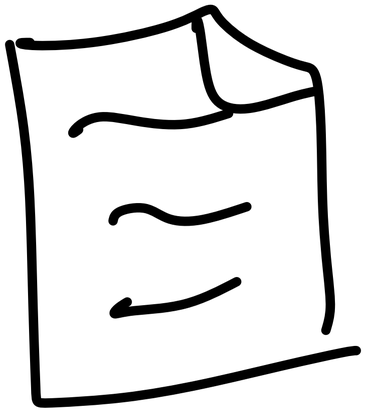
```yaml
                                                    # .gitlab-ci.yml
include:
  - template: Auto-DevOps.gitlab-ci.yml

variables:
  K8S_SECRET_SOME_VARIABLE: "this-is-passed-to-your-container"
```

process.env.SOME_VARIABLE

Type	↑ Key	Value	Protected	Masked	Environments	
Variable	POSTGRES_ENABLED	**********	✕	✕	All (default)	✎
Variable	K8S_SECRET_ACTIVATE_FE...	**********	✕	✕	All (default)	✎
Variable	K8S_SECRET_ACTIVATE_FE...	**********	✕	✕	production	✎

Deployment Konfiguration

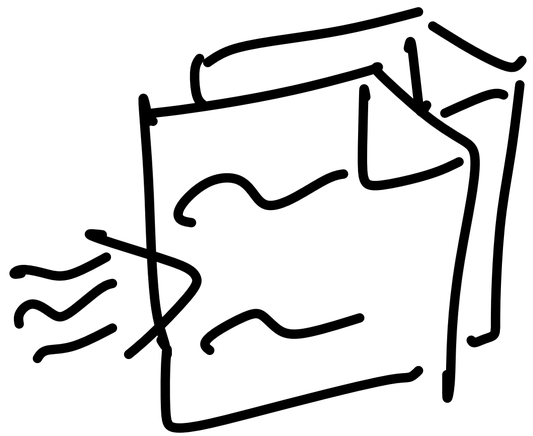


Configure Auto DevOps - Deployment Konfiguration

```
                    # .gitlab/auto-deploy-values.yaml
service:
  internalPort: 80
ingress:
  tls:
    enabled: false
```

values.yaml - Auto Deploy Image

Änderungen am Helm Chart

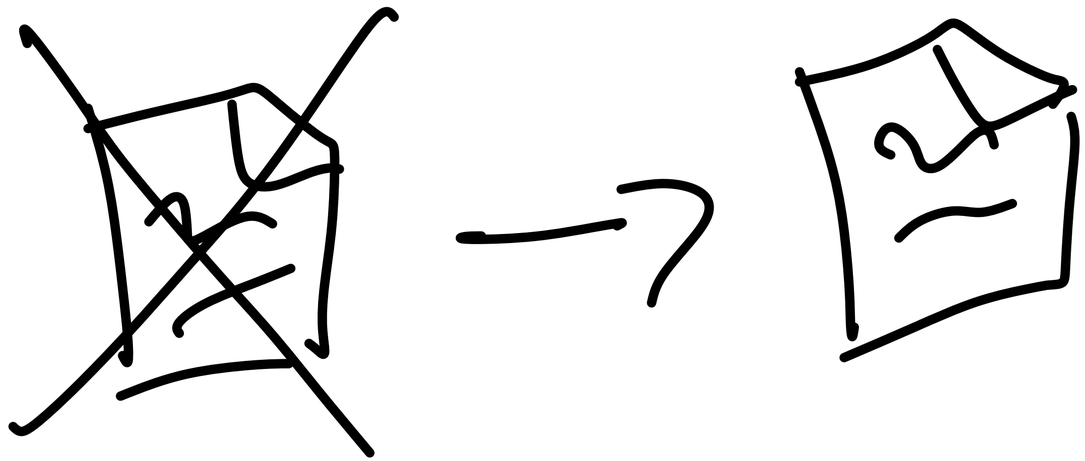


Configure Auto DevOps - Änderungen am Helm Chart

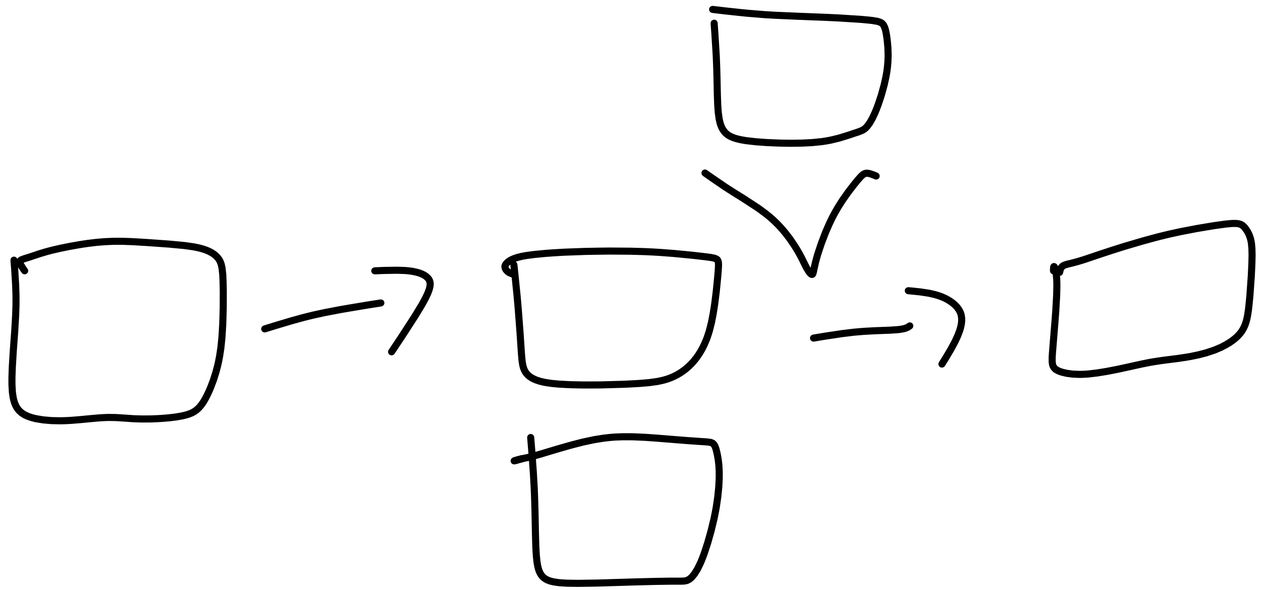
Helm Chart - Auto Deploy Image  /chart

Configure Auto DevOps

Eigenes Helm Chart



Änderungen an der Pipeline



Hands-on

Aber vorher noch Zeit für Fragen.

Workshop Environment

https://christophwalter.net/blog/beyond-gitlab-auto-devops-workshop

Nützlich Links

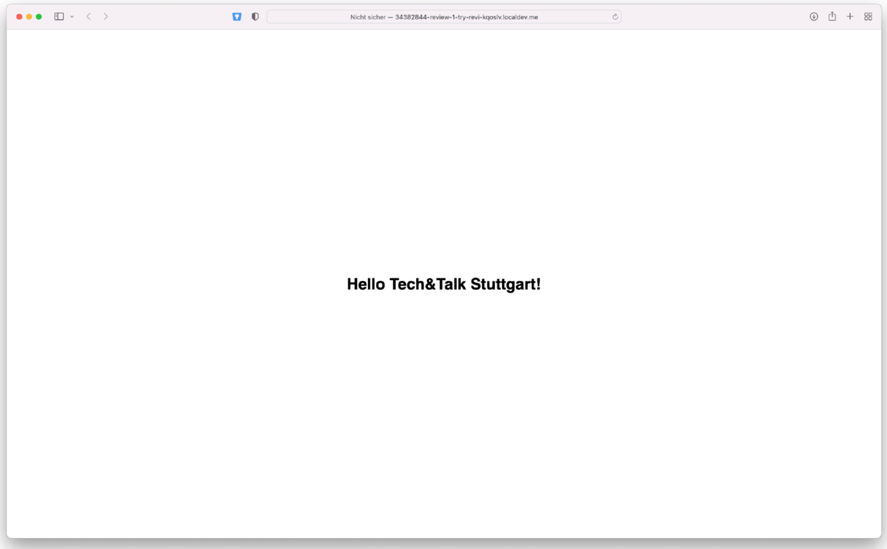
Auto DevOps Pipeline
Auto Deploy Image
Auto Build Image
Auto DevOps CI/CD Variables
Predefined CI/CD Variables

Challenge 1

GitLab Workflow kennenlernen

Challenge 1 - GitLab Workflow kennenlernen

Lege einen neuen Task an, erstelle einen Merge request und ändere den Titel der Demo Anwendung.



Tipp: Die Datei ./index.ejs beschreibt den Inhalt der Webseite.

Challenge 2

Arbeiten mit Umgebungsvariablen

Ändere den Titel der Anwendung auf Basis einer Umgebungsvariable.

Tipp: Mit Hilfe der Template Engine ejs können Umgebungsvariablen ausgegeben werden. Betroffene Dateien: index.js und index.ejs.

Challenge 3

Kennenlernen der CI/CD Pipeline

Mache den Main Branch unter der Subdomain workshop.localdev.me erreichbar.

Tipp: Die URL wird im Deploy Job der Auto DevOps Pipeline definiert.

Challenge 4

Wir brauchen mehr Tests!

Erstelle einen Job der nach jedem Deployment prüft, ob die Anwendung auch erreichbar ist.

Tipp: Dafür brauchen wir einen neuen Job, den passenden Tests (z.B. mit Cypress) und müssen die zu testende URL übergeben.