

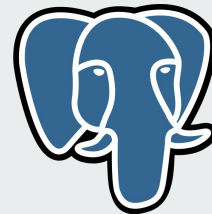
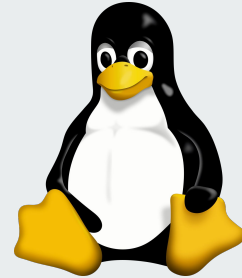


CI/CD für Einsteiger:innen

Der erste Schritt zur Automatisierung

Ole Koring

Softwareentwickler @ Techniker Krankenkasse





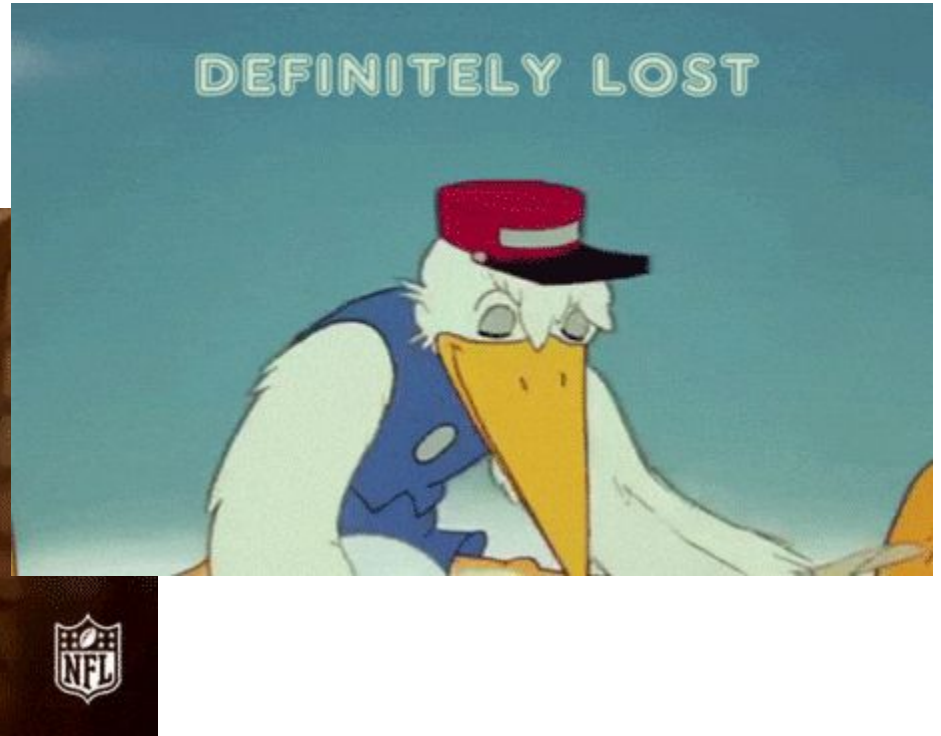
Gliederung

1. Motivation
2. Was ist CI/CD?
3. Beispielprojekt
4. Automatisch testen
5. Automatisch bauen
6. Automatisch deployen

Motivation

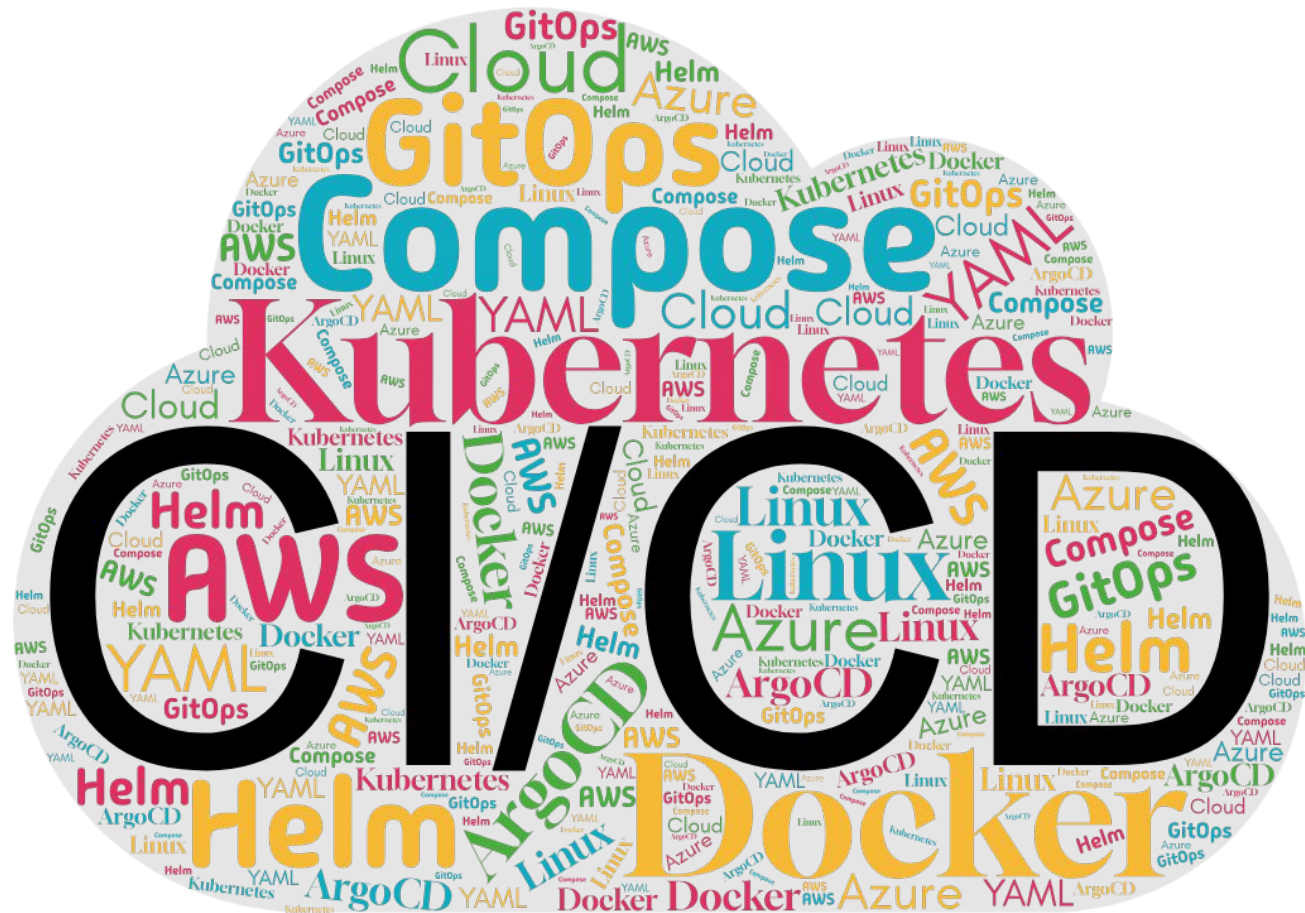


Motivation



Motivation







Was ist CI/CD überhaupt?

- C = Continuous => immer, ständig
- I = Integration => mergen & testen von neuem Code
- D = Delivery => Software ist auslieferbar
- D = Deployment => Software ausliefern

Wir wollen:

- neuen Code ständig integrieren
- mit jedem Stand Tests ausführen (möglichst schnell)
- und den neuen Code Stand ausliefern nach Produktion (oder Testsystem)



CD - Delivery vs Deployment

	Continuous Delivery	Continuous Deployment
Tests ausführen	✓	✓
Release erstellen	✓	✓
Release deployen	manuell	✓



Continuous?

Oft genug, dass

- Änderungen klein sind
- Feedback schnell ist
- Deployment Routine ist

=> kommt drauf an



Was bringt das?

- schnelles Feedback durch häufige Testausführung
- welche Commits brechen die Tests?
- Kund:innen erhalten regelmäßig Updates und Bugfixes
- Kund:innen erhalten nur funktionierende Software (Voraussetzung Tests)
- kleine Releases sind besser überblickbar

Beispielprojekt

- Spring Boot
- Java 21
- Gradle
- PostgreSQL
- Backend only
- hohe Unit Test Abdeckung
- Push direkt auf main
- Merge Requests unterstützen
- GitLab CI/CD
- Docker
- Docker Compose
- Linux Server



<https://gitlab.com/oleko2305/ci-cd-einstieg>



Pipeline

Stage 1

Job 1

Job 2

Job n

Stage 2

Job 1

Job 2

Job n

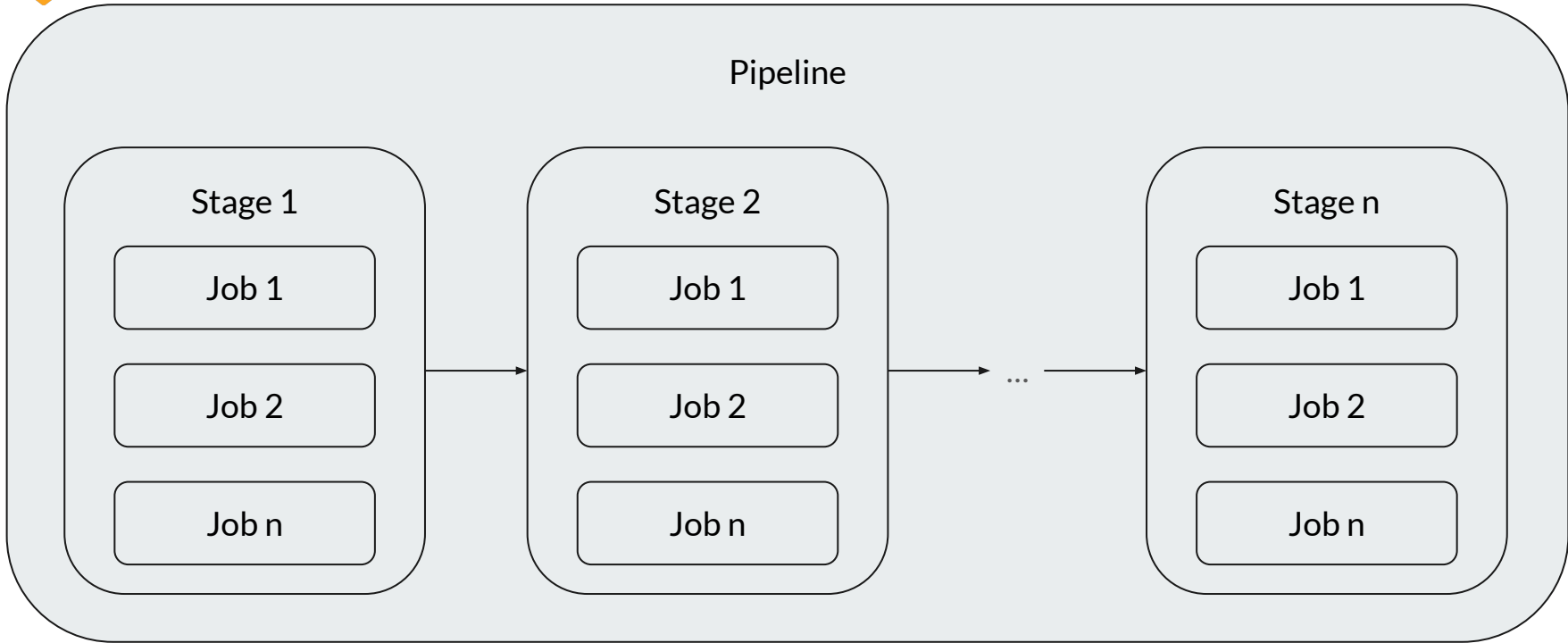
...

Stage n

Job 1

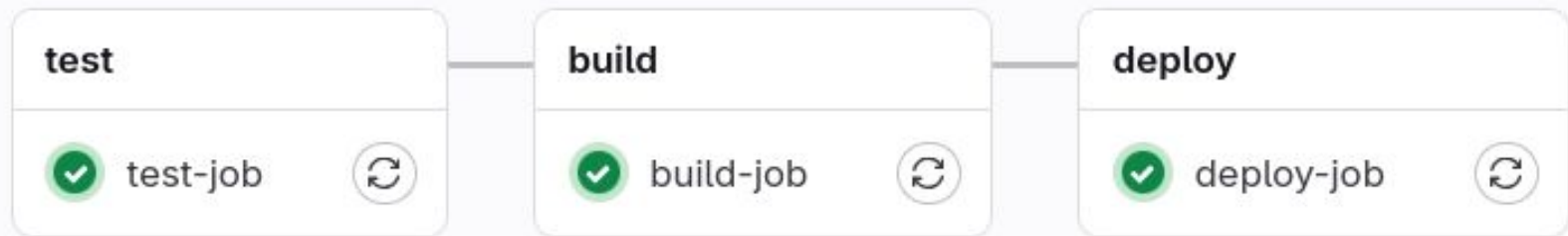
Job 2

Job n

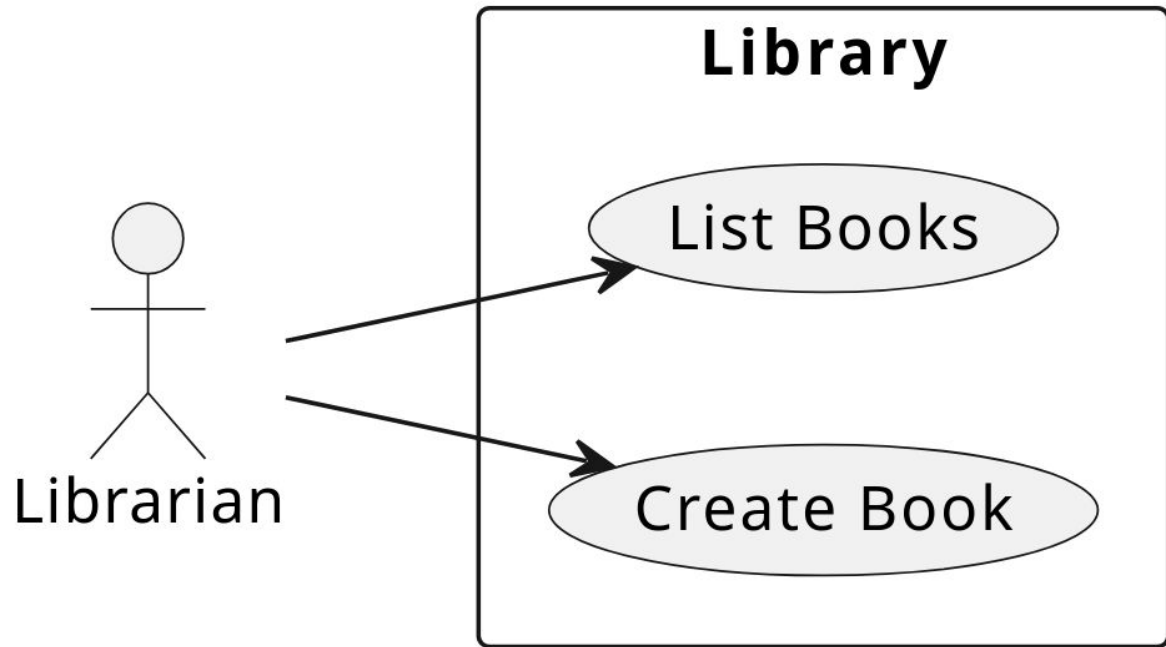




Ziel



Domäne





Project

☒ Gradle - Groovy ☐ Gradle - Kotlin ☐ Maven

Language

☒ Java ☐ Kotlin ☐ Groovy

Spring Boot

☐ 4.0.0 (SNAPSHOT) ☐ 3.5.1 (SNAPSHOT) ☒ 3.5.0 ☐ 3.4.7 (SNAPSHOT)
☐ 3.4.6 ☐ 3.3.13 (SNAPSHOT) ☐ 3.3.12

Project Metadata

Group

Artifact

Name

Description

Package name

Packaging ☒ Jar ☐ War

Java ☐ 24 ☒ 21 ☐ 17

Dependencies

ADD DEPENDENCIES... CTRL + B

Spring Web WEB

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

Spring Data JPA SQL

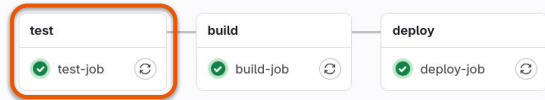
Persist data in SQL stores with Java Persistence API using Spring Data and Hibernate.

H2 Database SQL

Provides a fast in-memory database that supports JDBC API and R2DBC access, with a small (2mb) footprint. Supports embedded and server modes as well as a browser based console application.

PostgreSQL Driver SQL

A JDBC and R2DBC driver that allows Java programs to connect to a PostgreSQL database using standard, database independent Java code.

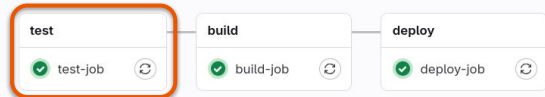


CI - Tests ausführen

Voraussetzungen:

- CI/CD-Umgebung => GitLab
- Testausführung per Skript => `./gradlew test`
- Testreport wird generiert => `./gradlew test`





CI - Tests ausführen

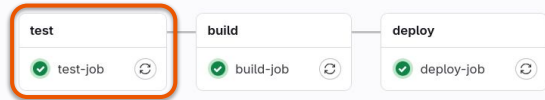


```
> ./gradlew test
OpenJDK 64-Bit Server VM warning: Sharing is only supported for boot loader classes because bootstrap classpath has been appended
2025-09-14T18:28:45.121+02:00 INFO 13384 — [ci-cd-einstieg] [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence unit 'default'
2025-09-14T18:28:45.128+02:00 INFO 13384 — [ci-cd-einstieg] [ionShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown initiated ...
2025-09-14T18:28:45.221+02:00 INFO 13384 — [ci-cd-einstieg] [ionShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown completed.

BUILD SUCCESSFUL in 18s
5 actionable tasks: 5 executed
```

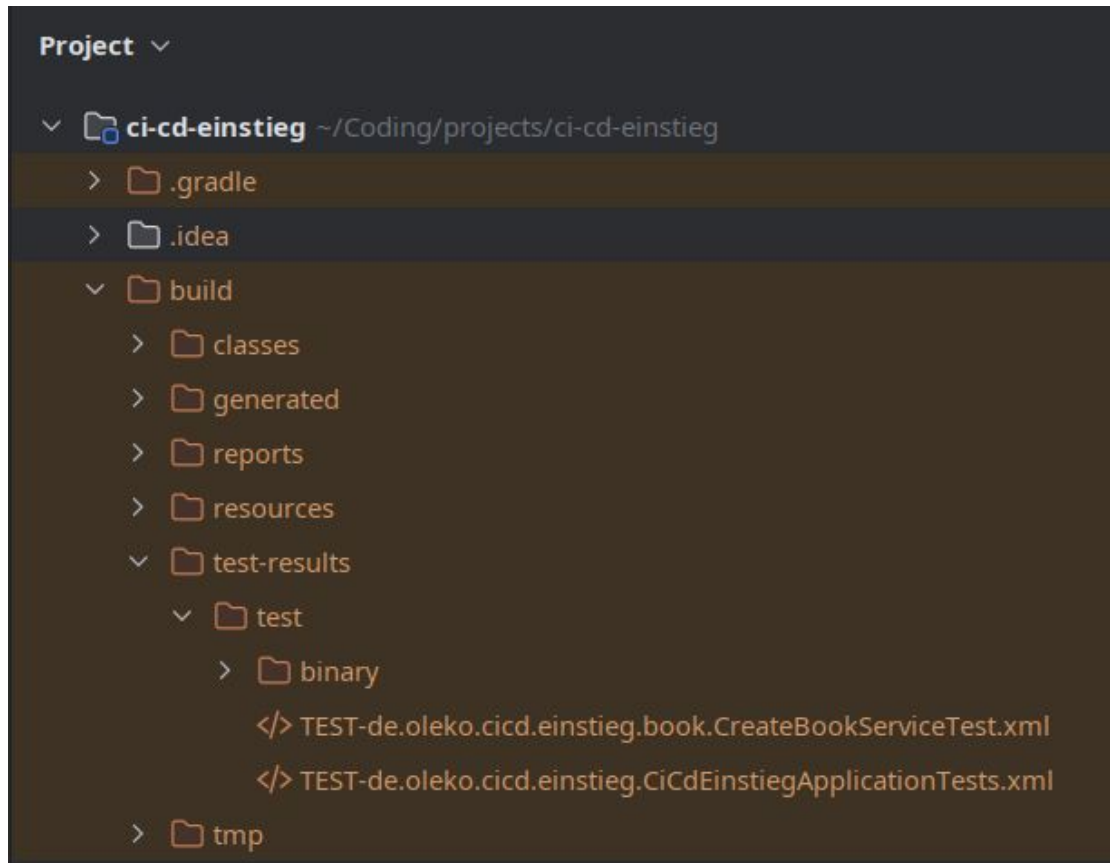
```
> ./gradlew test
OpenJDK 64-Bit Server VM warnin
2025-09-14T18:28:45.121+02:00
2025-09-14T18:28:45.128+02:00
2025-09-14T18:28:45.221+02:00

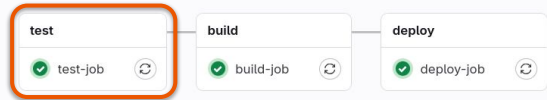
BUILD SUCCESSFUL in 18s
5 actionable tasks: 5 executed
```



CI - Tests ausführen

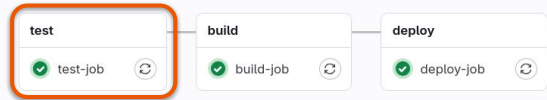
> ./gradlew test





TEST-de.oleko.cicd.einstieg.book.CreateBookServiceTest.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<testsuite name="de.oleko.cicd.einstieg.book.CreateBookServiceTest" tests="3" skipped="0" failures="0" errors="0"
  timestamp="2025-06-02T18:36:37.112Z" hostname="name-of-pc" time="0.07">
  <properties/>
  <testcase name="new_book_is_saved()" classname="de.oleko.cicd.einstieg.book.CreateBookServiceTest" time="0.062"/>
  <testcase name="book_request_params_correct()" classname="de.oleko.cicd.einstieg.book.CreateBookServiceTest" time="0.005"/>
  <testcase name="new_book_is_available()" classname="de.oleko.cicd.einstieg.book.CreateBookServiceTest" time="0.001"/>
  <system-out><![CDATA[]]></system-out>
  <system-err><![CDATA[]]></system-err>
</testsuite>
```



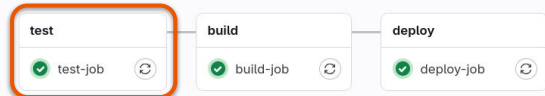
CI - Tests ausführen



.gitlab-ci.yml

```
test-job:
  stage: test
  image: eclipse-temurin:21.0.7_6-jdk
  script:
    - ./gradlew clean test
  artifacts:
    reports:
      junit: '**/build/test-results/**/TEST-*.xml'
  rules:
    - if: $CI_PIPELINE_SOURCE == 'merge_request_event'
    - if: $CI_COMMIT_BRANCH == $CI_DEFAULT_BRANCH
```





ci-cd test

Delete

✓ Passed Ole Koring created pipeline for commit `b5aa4e92` 2 weeks ago, finished 2 weeks ago

For `main`

latest branch 1 job 1.31 1 minute 18 seconds, queued for 1 seconds

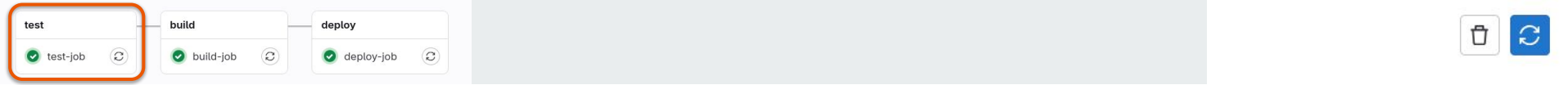
Pipeline Jobs 1 Tests 4

< test-job

4 tests 0 failures 0 errors 100% success rate 668.00ms

Tests

Suite	Name	Filename	Status	Duration	Details
de.oleko.cicd.einstieg.CiCdEinstiegApplicationTests	contextLoads()		✓	610.00ms	View details
de.oleko.cicd.einstieg.book.CreateBookServiceTest	new_book_is_saved()		✓	52.00ms	View details
de.oleko.cicd.einstieg.book.CreateBookServiceTest	book_request_params_correct()		✓	5.00ms	View details
de.oleko.cicd.einstieg.book.CreateBookServiceTest	new_book_is_available()		✓	1.00ms	View details



Code Coverage

mögliche Erweiterung

im Beispielprojekt umgesetzt

Duration: 1 minute 21 seconds

Finished: 5 minutes ago

Queued: 0 seconds

Timeout: 1h (from project) ?

Runner: #12270848 (ns46NMmJ) 2-green.saas-linux-small-amd64.runners-manager.gitlab.com/default

Source: Push

Coverage: 50%

Test summary: 4

Job artifacts ?

These artifacts are the latest. They will not be deleted (even if expired) until newer artifacts are available.

Commit cee1da76

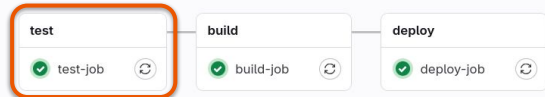
coverage report

Pipeline #1882954896 ✓ Passed for main

test

Related jobs

→ ✓ test-job



Fehlschlagende Tests



```
> ./gradlew test
Starting a Gradle Daemon, 1 incompatible Daemon could not be reused, use --status for details
OpenJDK 64-Bit Server VM warning: Sharing is only supported for boot loader classes because bootstrap classpath has been appended
2025-09-10T13:31:32.014+02:00 INFO 5036 — [ci-cd-einstieg] [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence unit 'default'

> Task :test

CreateBookServiceTest > new_book_is_available() FAILED
    org.opentest4j.AssertionFailedError at CreateBookServiceTest.java:30

2025-09-10T13:31:32.019+02:00 INFO 5036 — [ci-cd-einstieg] [ionShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown initiated...
2025-09-10T13:31:32.105+02:00 INFO 5036 — [ci-cd-einstieg] [ionShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown completed.

> Task :test FAILED

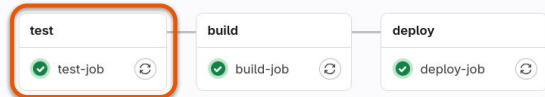
7 tests completed, 1 failed

FAILURE: Build failed with an exception.

* What went wrong:
Execution failed for task ':test'.
> There were failing tests. See the report at: file:///home/ole/Coding/projects/ci-cd-einstieg/build/reports/tests/test/index.html

* Try:
> Run with --scan to get full insights.

BUILD FAILED in 21s
5 actionable tasks: 5 executed
```



Fehlschlagende Tests

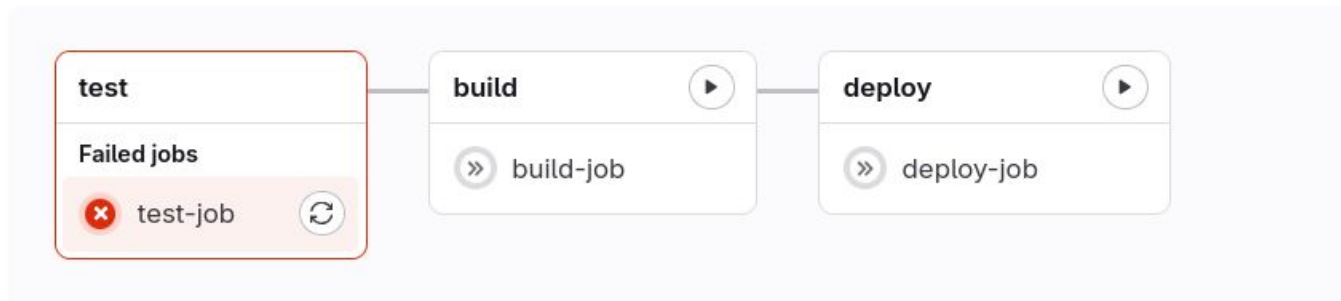
failing test as example

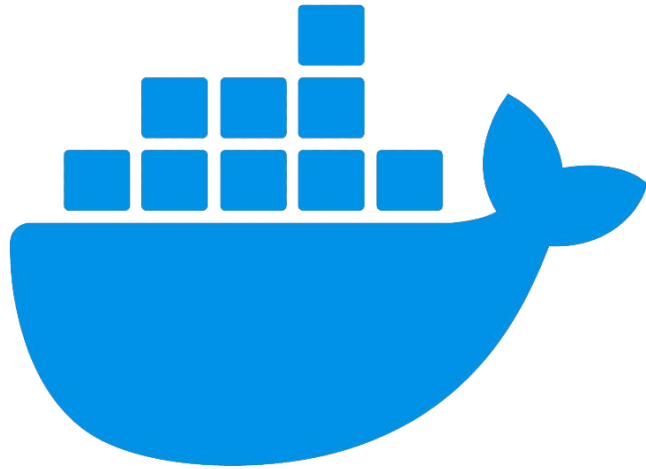
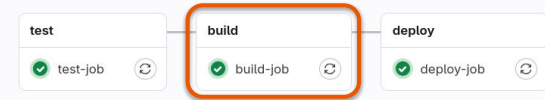
❌ Failed Ole Koring created pipeline for commit `94464766` 1 minute ago, finished 10 seconds ago

For `main`

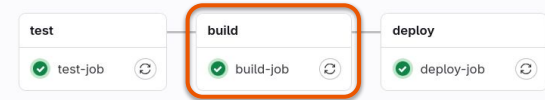
latest branch 3 jobs 1.33 1 minute 19 seconds, queued for 1 seconds

Pipeline Jobs 3 Failed Jobs 1 Tests 7

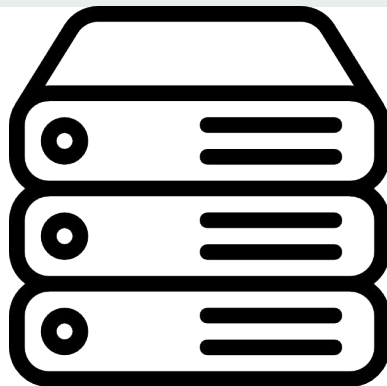




docker®

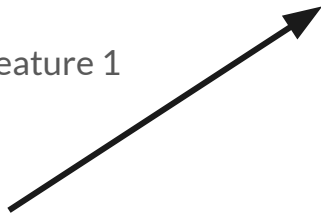


Beispiel

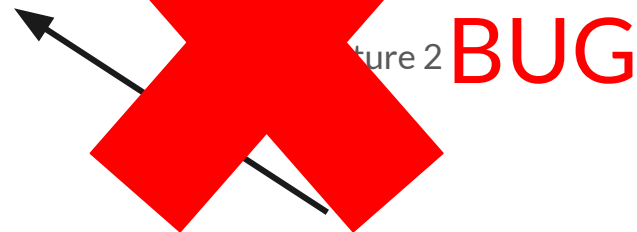


- Ubuntu Server 24.04 LTS
- Java 21 LTS
- Apache Commons Lang 3.12.0

Neues Feature 1



- Mac OS 26 Beta
- Java 24
- Apache Commons Lang 3.18.0

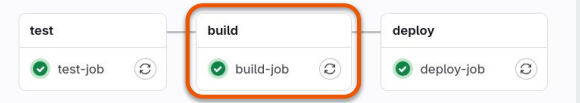


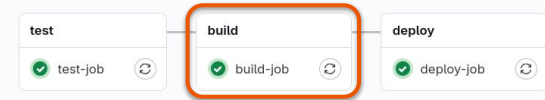
Neues Feature 2

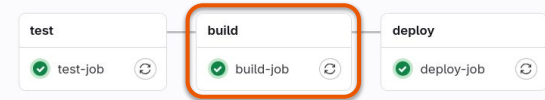


- Windows 11
- Java 22
- Apache Commons Lang 3.17.0

It works on my machine!?!

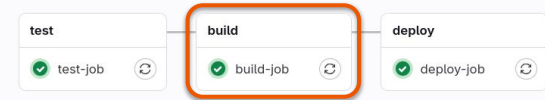






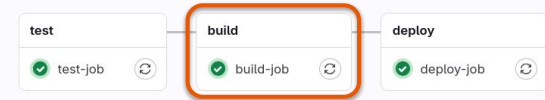
Docker Image

“An image is a **read-only template** with instructions for creating a Docker container. Often, an image is **based on another image**, with some additional customization. For example, you may build an image which is based on the **ubuntu** image, but installs the **Apache web server** and **your application**, as well as the **configuration details** needed to make your application run.”



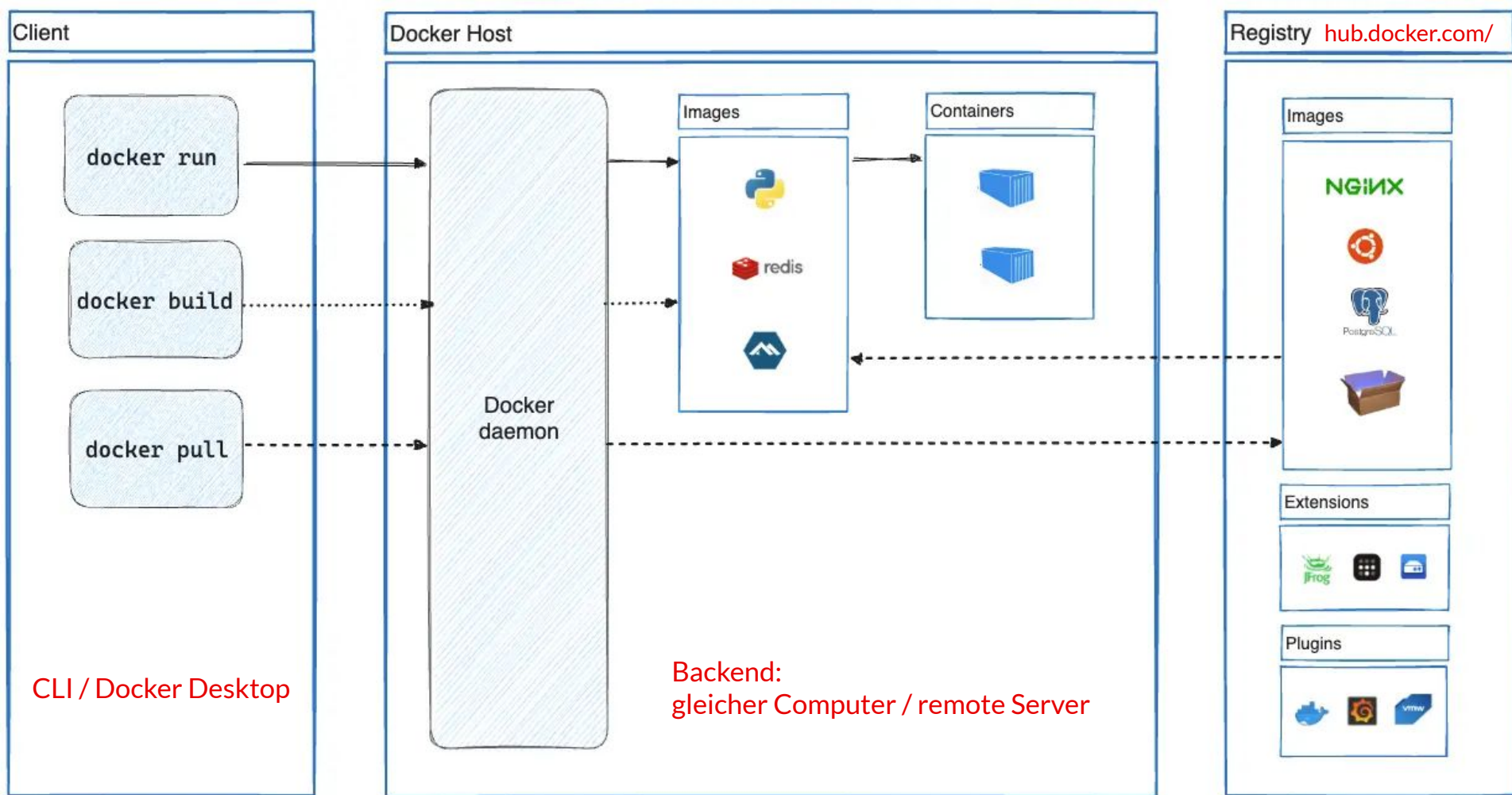
Docker Container

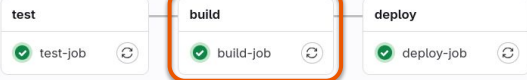
“A container is a **runnable instance of an image**. You can create, start, stop, move, or delete a container using the Docker API or CLI. You can connect a container to one or more **networks**, attach **storage** to it, or even create a new image based on its current state.”



Docker Registry

“A Docker registry **stores Docker images**. Docker Hub is a public registry that anyone can use, and Docker looks for images on Docker Hub by default. You can even run your own private registry.”





.jar-File bauen

> ./gradlew bootJar



CE ci-cd-einstieg main

Project

ci-cd-einstieg ~/Coding/projects/ci-cd-einstieg

> .gradle

> .idea

build

> classes

> generated

> jacoco

libs

ci-cd-einstieg-1.0.0.jar

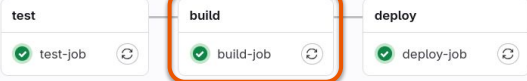
> reports

> resources

> test-results

> tmp

resolvedMainClassName



CD - Image bauen + Registry

Project

C ci-cd-einstieg

🚀 Pinned ▾

Merge requests 0

Issue boards

Branches

Commits

Pipelines

👤 Manage >

📅 Plan >

</> Code >

🚀 Build >

🛡️ Secure >

🔄 Deploy >

🔧 Operate >

📊 Monitor >

📈 Analyze >

⚙️ Settings >

Releases

Feature flags

Package registry

Container registry 🚀

Model registry

Pages

test

✓ test-job ↺

build

✓ build-job ↺

deploy

✓ deploy-job ↺

Container registry



🕒 Cleanup is not scheduled. 🛡️ Container scanning for registry: Off



Updated ▾



There are no container images stored for this project

With the container registry, every project can have its own space to store its Docker images. [More Information](#)

CLI commands

If you are not already logged in, you need to authenticate to the container registry by using your GitLab username and password. If you have [Two-Factor Authentication](#) enabled, use a [personal access token](#) instead of a password.

```
docker login registry.gitlab.com
```



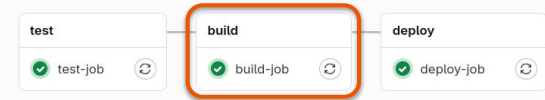
You can add an image to this registry with the following commands:

```
docker build -t registry.gitlab.com/oleko2305/ci-cd-einsti
```



```
docker push registry.gitlab.com/oleko2305/ci-cd-einstieg
```





Dockerfile



```
FROM eclipse-temurin:21.0.7_6-jdk AS builder
```

```
WORKDIR /app
```

```
ADD . /app
```

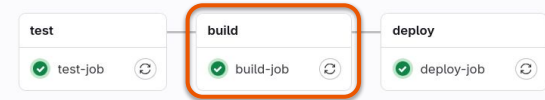
```
RUN ./gradlew bootJar
```

```
FROM eclipse-temurin:21.0.7_6-jre
```

```
COPY --from=builder /app/build/libs/ci-cd-einstieg-1.0.0.jar app.jar
```

```
EXPOSE 8080
```

```
ENTRYPOINT ["java", "-jar", "app.jar"]
```



Build & Push Image



.gitlab-ci.yml



```
build-job:
  stage: build
  image: docker:cli
  services:
    - docker:dind
  before_script:
    - docker login -u "$GITLAB_USER" -p "$GITLAB_PASSWORD" registry.gitlab.com
  script:
    - docker build --platform linux/amd64 -t registry.gitlab.com/oleko2305/ci-cd-einstieg:$CI_COMMIT_SHORT_SHA
    - docker push registry.gitlab.com/oleko2305/ci-cd-einstieg:$CI_COMMIT_SHORT_SHA
```

test

test-job

build

build-job

deploy

deploy-job



ci-cd-einstieg

1 tag 143.02 MiB Cleanup disabled Created Jul 6, 2025 14:40 Last published at Jul 6, 2025 14:40

Filter results

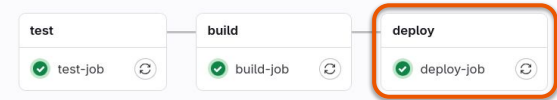
Published

bc331f14 143.03 MiB

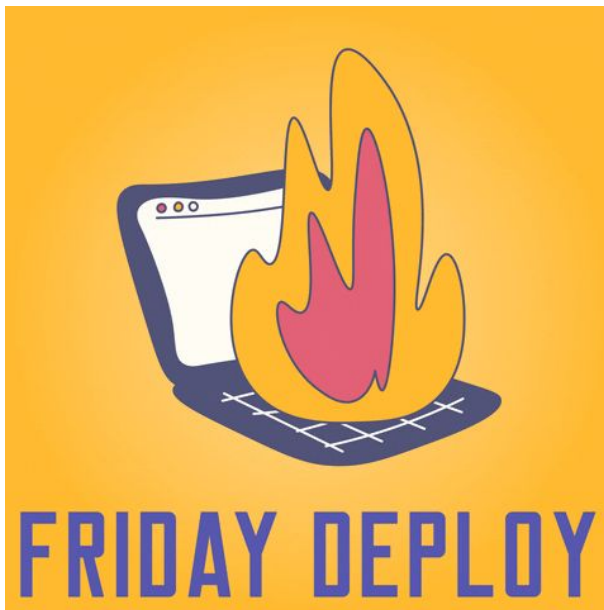
Published 3 minutes ago

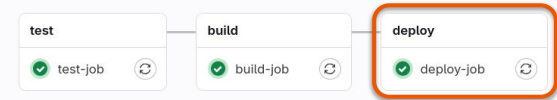
Digest: 96e01ae

- Published to the `oleko2305/ci-cd-einstieg` image repository on July 6, 2025 at 4:40:59 PM GMT+2
- Manifest digest: `sha256:96e01aed5a58e7e1c555f228cedc94a619cd41a95b4262e384de4768a1dc5f01`
- Manifest media type: `application/vnd.docker.distribution.manifest.v2+json`
- Configuration digest: `sha256:42859af6893c0e52a32a078c5ee2b73abdacab162e0bf7299b7ea9caacd50ea6`



CD - Deployment





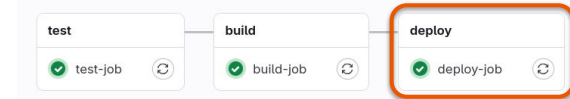
Docker Compose

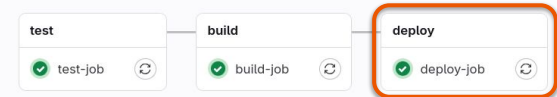
```
services:
  app:
    container_name: ci-cd-backend
    image: registry.gitlab.com/oleko2305/ci-cd-einstieg:${IMAGE_TAG}
    restart: unless-stopped
    environment:
      - POSTGRES_USER=${POSTGRES_USER}
      - POSTGRES_PASSWORD=${POSTGRES_PASSWORD}
    ports:
      - "8080:8080"
    depends_on:
      - db
  db:
    container_name: ci-cd-db
    image: postgres:17.5
    restart: unless-stopped
    volumes:
      - ci-cd-einstieg-db-data:/var/lib/postgresql/data
    environment:
      - POSTGRES_USER=${POSTGRES_USER}
      - POSTGRES_PASSWORD=${POSTGRES_PASSWORD}
      - POSTGRES_DB=ci_cd_einstieg
volumes:
  ci-cd-einstieg-db-data
```



docker-compose.yml

```
services:
  app:
    container_name: ci-cd-backend
    image: registry.gitlab.com/oleko2305/ci-cd-einstieg:${IMAGE_TAG}
    restart: unless-stopped
    environment:
      - POSTGRES_USER=${POSTGRES_USER}
      - POSTGRES_PASSWORD=${POSTGRES_PASSWORD}
    ports:
      - "8080:8080"
    depends_on:
      - db
  db:
    container_name: ci-cd-db
    image: postgres:17.5
    restart: unless-stopped
    volumes:
      - ci-cd-einstieg-db-data:/var/lib/postgresql/data
    environment:
      - POSTGRES_USER=${POSTGRES_USER}
      - POSTGRES_PASSWORD=${POSTGRES_PASSWORD}
      - POSTGRES_DB=ci_cd_einstieg
volumes:
  ci-cd-einstieg-db-data :
```

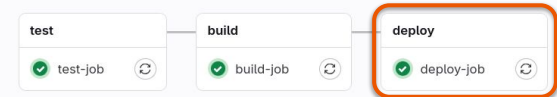




Docker Compose

```
> IMAGE_TAG=bc331f14 POSTGRES_USER=postgres POSTGRES_PASSWORD=password docker compose up -d
[+] Running 3/3
✓ Network ci-cd-einstieg_default Created
✓ Container ci-cd-db Started
✓ Container ci-cd-backend Started
```

```
> docker ps
CONTAINER ID    IMAGE
9eb39def3219   registry.gitlab.com/oleko2305/ci-cd-einstieg:bc331f14
5b01d10953d8   postgres:17.5
```



CD - Deployment

```
deploy-job:
  stage: deploy
  before_script:
    - chmod 400 $SERVER_KEY_FILE
  script:
    - export IMAGE_TAG= $CI_COMMIT_SHORT_SHA
    - export POSTGRES_USER= $POSTGRES_USER
    - export POSTGRES_PASSWORD= $POSTGRES_PASSWORD
    - ssh -o StrictHostKeyChecking=no -i $SERVER_KEY_FILE -p $SERVER_SSH_PORT $SERVER_USER @$SERVER_IP "mkdir -p apps/ci-cd-einstieg"
    - scp -o StrictHostKeyChecking=no -i $SERVER_KEY_FILE -P $SERVER_SSH_PORT docker-compose.yml $SERVER_USER @$SERVER_IP :apps/ci-cd-einstieg/
    - ssh -o StrictHostKeyChecking=no -i $SERVER_KEY_FILE -p $SERVER_SSH_PORT $SERVER_USER @$SERVER_IP "echo 'IMAGE_TAG=${ IMAGE_TAG }' > apps/ci-cd-einstieg/.env"
    - ssh -o StrictHostKeyChecking=no -i $SERVER_KEY_FILE -p $SERVER_SSH_PORT $SERVER_USER @$SERVER_IP "echo 'POSTGRES_USER=${ POSTGRES_USER }' >> apps/ci-cd-einstieg/.env"
    - ssh -o StrictHostKeyChecking=no -i $SERVER_KEY_FILE -p $SERVER_SSH_PORT $SERVER_USER @$SERVER_IP "echo 'POSTGRES_PASSWORD=${ POSTGRES_PASSWORD }' >> apps/ci-cd-einstieg/.env"
    - ssh -o StrictHostKeyChecking=no -i $SERVER_KEY_FILE -p $SERVER_SSH_PORT $SERVER_USER @$SERVER_IP "
      set -e &&
      cd apps/ci-cd-einstieg &&
      echo ' $GITLAB_PASSWORD ' | docker login -u ' $GITLAB_USER ' --password-stdin registry.gitlab.com &&
      docker compose pull &&
      docker compose up -d --remove-orphans"
  rules:
    - if: $CI_COMMIT_BRANCH == $CI_DEFAULT_BRANCH
```



deploy-job:

stage: deploy

before_script:

- chmod 400 \$SERVER_KEY_FILE

script:

- export IMAGE_TAG=\$CI_COMMIT_SHORT_SHA

- export POSTGRES_USER=\$POSTGRES_USER

- export POSTGRES_PASSWORD=\$POSTGRES_PASSWORD

- ssh -o StrictHostKeyChecking=no -i \$SERVER_KEY_FILE -p \$SERVER_SSH_PORT \$SERVER_USER@\$SERVER_IP "mkdir -p apps/ci-cd-einstieg"

- scp -o StrictHostKeyChecking=no -i \$SERVER_KEY_FILE -P \$SERVER_SSH_PORT docker-compose.yml \$SERVER_USER@\$SERVER_IP:apps/ci-cd-einstieg/

- ssh -o ... "echo 'IMAGE_TAG=\${ IMAGE_TAG }' > apps/ci-cd-einstieg/.env"

- ssh -o ... "echo 'POSTGRES_USER=\${ POSTGRES_USER }' >> apps/ci-cd-einstieg/.env"

- ssh -o ... "echo 'POSTGRES_PASSWORD=\${ POSTGRES_PASSWORD }' >> apps/ci-cd-einstieg/.env"

- ssh -o ... "

- set -e &&

- cd apps/ci-cd-einstieg &&

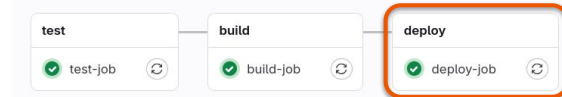
- echo '\$GITLAB_PASSWORD' | docker login -u '\$GITLAB_USER' --password-stdin registry.gitlab.com &&

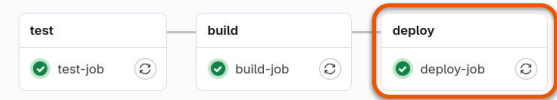
- docker compose pull &&

- docker compose up -d --remove-orphans"

rules:

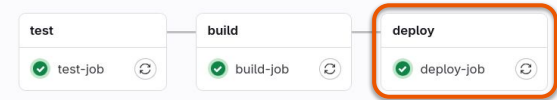
- if: \$CI_COMMIT_BRANCH == \$CI_DEFAULT_BRANCH





Passwörter & Variablen

CI/CD Variables </> 8				Reveal values	Add variable
Key ↑	Value	Environments	Actions		
GITLAB_PASSWORD	Masked Hidden	All (default)			
GITLAB_USER	Masked Expanded	All (default)			
POSTGRES_PASSWORD	Masked	All (default)			
POSTGRES_USER	Masked	All (default)			
SERVER_IP		All (default)			
SERVER_KEY_FILE Private key file for server	File	All (default)			
SERVER_SSH_PORT		All (default)			
SERVER_USER		All (default)			



Passwörter & Variablen



application.properties

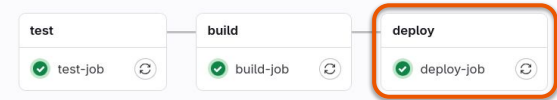
```
# Postgres
```

```
spring.datasource.url=jdbc:postgresql://db:5432/ci_cd_einstieg
```

```
spring.datasource.username=${POSTGRES_USER}
```

```
spring.datasource.password=${POSTGRES_PASSWORD}
```

```
spring.datasource.driver-class-name=org.postgresql.Driver
```



Passwörter & Variablen



docker-compose.yml

```
services:
  app:
    container_name: ci-cd-backend
    environment:
      - POSTGRES_USER=${POSTGRES_USER}
      - POSTGRES_PASSWORD=${POSTGRES_PASSWORD}
    depends_on:
      - db
  db:
    container_name: ci-cd-db
    environment:
      - POSTGRES_USER=${POSTGRES_USER}
      - POSTGRES_PASSWORD=${POSTGRES_PASSWORD}
      - POSTGRES_DB=ci_cd_einstieg
volumes:
  ci-cd-einstieg-db-data:
```





Passwörter & Variablen

`.env`

`IMAGE_TAG=a1b2d3c4`

`POSTGRES_USER=my-postgres-username`

`POSTGRES_PASSWORD=password123!`



CI/CD



<http://ci.olekoring.de:8080/book>

Ole Koring / ci-cd-einstieg / Pipelines / #1982301919

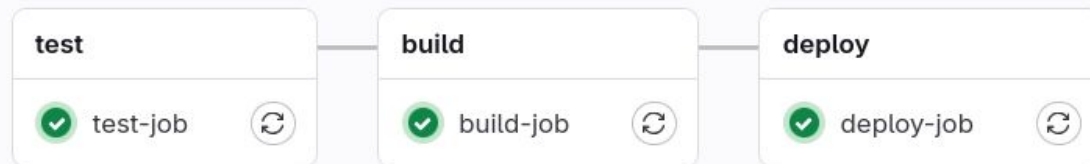
application properties

✓ Passed Ole Koring created pipeline for commit `d435769c` 2 hours ago, finished 2 hours ago

For `main`

latest branch 3 jobs 4.09 4 minutes 5 seconds, queued for 1 seconds

Pipeline Jobs 3 Tests 7





Danke!