

## CICD\_Ex03\_s2310237004\_Lichtenberger

A. Local Containerization	Multi-stage Dockerfile and .dockerignore	Build stage (Maven) → runtime stage (Temurin JRE); reasonable ignore rules	4
	Local build of the image	<code>docker build -t &lt;owner&gt;/&lt;app&gt;:dev .</code> succeeds	2
	Local run of the container	Container outputs expected result	2

Zuerst habe ich ein Dockerfile & .dockerignore erstellt siehe repo.

Local Build und Test:

Commands:

`docker build -t myapp:dev .`

`docker run --rm myapp:dev`

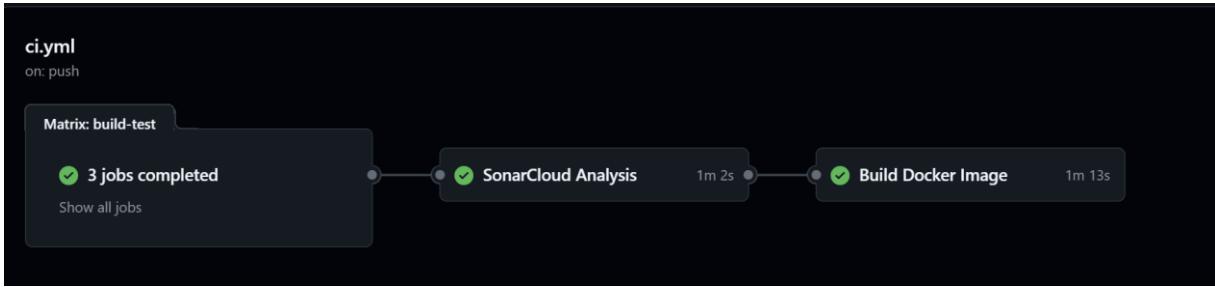
Output:

```
PS C:\Users\Licht\OneDrive\Documents\FH\FH_Semester\5.Semester\CICD\Ex01\cicd-BA-uebung01-Lichtenberger> docker build -t myapp:dev .
[+] Building 0.8s (15/15) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 536B
=> [internal] load metadata for docker.io/library/eclipse-temurin:20-jre
=> [internal] load metadata for docker.io/library/maven:3.9.3-eclipse-temurin-20
=> [internal] load .dockerignore
=> => transferring context: 768
=> [build 1/6] FROM docker.io/library/maven:3.9.3-eclipse-temurin-20@sha256:de70becda2f183567e105530460b6cbefc014b64f27c2003d36d7fc85bc222f
=> => resolve docker.io/library/maven:3.9.3-eclipse-temurin-20@sha256:de70becda2f183567e105530460b6cbefc014b64f27c2003d36d7fc85bc222f
=> [internal] load build context
=> => transferring context: 909B
=> [stage-1 1/3] FROM docker.io/library/eclipse-temurin:20-jre@sha256:0fcc06627fe82ef264674a1ad6ce29c91cfea45151c98fd5e27e15d2d7a966b3
=> => resolve docker.io/library/eclipse-temurin:20-jre@sha256:0fcc06627fe82ef264674a1ad6ce29c91cfea45151c98fd5e27e15d2d7a966b3
=> CACHED [stage-1 2/3] WORKDIR /app
=> CACHED [build 2/6] WORKDIR /app
=> CACHED [build 3/6] COPY pom.xml .
=> CACHED [build 4/6] RUN mvn dependency:go-offline
=> CACHED [build 5/6] COPY src ./src
=> CACHED [build 6/6] RUN mvn clean package -DskipTests && ls -la target/
=> CACHED [stage-1 3/3] COPY --from=build /app/target/*.jar app.jar
=> exporting to image
INFO: Sum(2,3) = 5
Nov 06, 2025 11:20:50 AM com.example.cicd.App main
INFO: sumUp=10, addAll=10
Nov 06, 2025 11:20:50 AM com.example.cicd.App main
INFO: isPalindrome(Anna)? false
Nov 06, 2025 11:20:50 AM com.example.cicd.App main
INFO: safeParseInt(42): 42
Nov 06, 2025 11:20:50 AM com.example.cicd.App main
INFO: safeParseInt(x): 0
```

Tests laufen wie erwartet durch.

B. CI Integration	Separate Docker job with sensible triggers	Placed logically after tests/Sonar; path filters & concurrency avoid duplicate runs	3
	Build image and tag with short SHA	Single-arch build (for <code>load</code> ) or justified alternative; clear tag naming	2
	Upload artifact as <code>.tar</code> named with SHA	Clear artifact name (e.g., <code>docker-image-&lt;sha&gt;.tar</code> )	2
	Clean integration with existing pipeline	Job dependencies ( <code>needs</code> ) correct, no Sonar double-analysis, minimal noise	2
	Readability & maintainability	Good job/step names, consistent artifact/tag naming	2

## Separate Docker job



## Sensible Triggers:

```
name: Docker Image (Artifact)

on:
  push:
    branches: [main]
    paths:
      - Dockerfile
      - .dockerignore
      - pom.xml
      - src/**
      - .github/workflows/**
  pull_request:
    branches: [main]
    paths:
      - Dockerfile
      - .dockerignore
      - pom.xml
      - src/**
      - .github/workflows/**
```

## Build Image + SHA

Artifacts			
Produced during runtime			
Name	Size	Digest	
Lichtiix~cicd-BA-uebung01-Lichtenberger~FUBC5W.dockerbuild	140 KB	sha256:1bdf575195644168586f6247df1489d76baa99c9a225fa5e...	<a href="#">Download</a> <a href="#">Delete</a>
docker-image-d7fef3f	93.4 MB	sha256:f819bb80cc3824d9b62b9fc2f293be7bac776aaef62453e13...	<a href="#">Download</a> <a href="#">Delete</a>

## Upload .tar

```
- name: Upload Docker image artifact
  uses: actions/upload-artifact@v4
  with:
    name: docker-image-${{ steps.vars.outputs.short_sha }}
    path: image-${{ steps.vars.outputs.short_sha }}.tar
    retention-days: 7
```

C. Consume Artifact	Download artifact from Actions & load image locally	report shown in PDF, docker load -i image-<sha>.tar , tag visible in docker images	2
	Run image locally	Container starts successfully; brief observation in PDF	2
	Total		24

## Download & load image

```
PS C:\Users\Licht\OneDrive\Documents\FH\FH_Semester\5.Semester\CICD\Ex01\cicd-BA-uebung01-Lichtenberger\images> docker load -i image-Sab3994.tar
Loaded image: local/app:Sab3994
PS C:\Users\Licht\OneDrive\Documents\FH\FH_Semester\5.Semester\CICD\Ex01\cicd-BA-uebung01-Lichtenberger\images> docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
local/app Sab3994 6f296d08580e Less than a second ago 582MB
myapp dev 7c96fc14f438 20 minutes ago 396MB
PS C:\Users\Licht\OneDrive\Documents\FH\FH_Semester\5.Semester\CICD\Ex01\cicd-BA-uebung01-Lichtenberger\images>
```

## Run image locally in Docker Desktop:

