Oskar Gunnarsson

Takima Devops crash course

1. Docker

1-1. We created a container for a database and another for running a postgres server. We put these containers in a single network. We copied SQL files into our entrypoint directory to have them run when running the container. We ran these and used adminer to have an interface to work in.

1-2. We use a multistage build to be able to combine multiple images for creating the final product. Springboot was used to provide a "simple API" that has a single endpoint.

1-3.

build	Build or rebuild services
bundle	Generate a Docker bundle from the Compose file
config	Validate and view the Compose file
create	Create services

1-4. We have a http front-end which depends on the backend (simpleAPI). We have the simpleapi backend, which depends on the database. Then we have the database. These are all connected to the same network and run on a single port.

1-5. `docker tag exampleId oskarggg/mysimpleapi`

`docker push oskarggg/mysimpleapi`

`docker tag exampleId oskarggg/mydatabase`

`docker push oskarggg/mydatabase`

`docker tag exampleId oskarggg/myhttp`

`docker push oskarggg/mymyhttp`

2. Github actions

2-1.
Testcontainers are temporarily created containers used for testing. When the tests are finished the containers are destroyed.

2-2.

```yaml
name: CI devops 2022
on:
  #to begin you want to launch this job in main and develop
  push:
```

```
    branches: main #TODO
  pull_request:

jobs:
  test-backend:
    runs-on: ubuntu-22.04
    steps:
     #checkout your github code using actions/checkout@v2.5.0
      - uses: actions/checkout@v2.5.0

     #do the same with another action (actions/setup-java@v3) that
enable to setup jdk 17
      - name: Set up JDK 17
        uses: actions/setup-java@v3
        with:
          distribution: 'zulu'
          java-version: '17'

     #finally build your app with the latest command
      - name: Build and test with Maven
        run: mvn -f
/home/runner/work/Devopstd/Devopstd/simple-api-student-main/ clean
verify   #TODO
```

This integrates the application. We then set up a way to publish it on docker. As well as a quality gate using sonar.

3. Ansible

Ansible is used to set up a digital infrastructure.

3-1. The inventory is used to define the managed nodes.