Issues

1. Start webapp with uploades images -> OK

2. Start web app with local build -> OK

Issue solved:

My ubuntu xenial machine had an outdated version of docker-compose which could not handle “version 3”. Latest version 1.20.1 solved the issue.

3. Development environment:

- local MongoDB -> OK

- npm run nodemon-dev-server fails

0 info it worked if it ends with ok

1 verbose cli [ '/usr/bin/node', '/usr/bin/npm', 'run', 'nodemon-dev-server' ]

2 info using npm@5.6.0

3 info using node@v8.10.0

4 verbose run-script [ 'prenodemon-dev-server',

4 verbose run-script 'nodemon-dev-server',

4 verbose run-script 'postnodemon-dev-server' ]

5 info lifecycle webapp@1.0.0~prenodemon-dev-server: webapp@1.0.0

6 info lifecycle webapp@1.0.0~nodemon-dev-server: webapp@1.0.0

7 verbose lifecycle webapp@1.0.0~nodemon-dev-server: unsafe-perm in lifecycle true

8 verbose lifecycle webapp@1.0.0~nodemon-dev-server: PATH: /usr/lib/node\_modules/npm/node\_modules/npm-lifecycle/node-gyp-bin:/home/josef/IdeaProjects/wacm\_ss18/webapp/node\_modules/.bin:/opt/anaconda3/bin:/home/josef/.rbenv/plugins/ruby-build/bin:/home/josef/.rbenv/shims:/home/josef/.rbenv/bin:/home/josef/bin:/home/josef/.local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/usr/bin/robomongo/bin:/snap/bin:/opt/gams/gams24.7\_linux\_x64\_64\_sfx

9 verbose lifecycle webapp@1.0.0~nodemon-dev-server: CWD: /home/josef/IdeaProjects/wacm\_ss18/webapp

10 silly lifecycle webapp@1.0.0~nodemon-dev-server: Args: [ '-c', 'nodemon server.js -c ./config/dev\_config.json' ]

11 info lifecycle webapp@1.0.0~nodemon-dev-server: Failed to exec nodemon-dev-server script

12 verbose stack Error: webapp@1.0.0 nodemon-dev-server: `nodemon server.js -c ./config/dev\_config.json`

12 verbose stack spawn ENOENT

12 verbose stack at ChildProcess.<anonymous> (/usr/lib/node\_modules/npm/node\_modules/npm-lifecycle/lib/spawn.js:48:18)

12 verbose stack at emitTwo (events.js:126:13)

12 verbose stack at ChildProcess.emit (events.js:214:7)

12 verbose stack at maybeClose (internal/child\_process.js:925:16)

12 verbose stack at Process.ChildProcess.\_handle.onexit (internal/child\_process.js:209:5)

13 verbose pkgid webapp@1.0.0

14 verbose cwd /home/josef/IdeaProjects/wacm\_ss18/webapp

15 verbose Linux 4.13.0-37-generic

16 verbose argv "/usr/bin/node" "/usr/bin/npm" "run" "nodemon-dev-server"

17 verbose node v8.10.0

18 verbose npm v5.6.0

19 error file sh

20 error code ELIFECYCLE

21 error errno ENOENT

22 error syscall spawn

23 error webapp@1.0.0 nodemon-dev-server: `nodemon server.js -c ./config/dev\_config.json`

23 error spawn ENOENT

24 error Failed at the webapp@1.0.0 nodemon-dev-server script.

24 error This is probably not a problem with npm. There is likely additional logging output above.

25 verbose exit [ 1, true ]

- npm run build-watch -> OK

IntelliJ -> some issues, can be addressed later

Technology stack

**Node.js**

Open-source, cross-platform JavaScript runtime environment for executing JavaScript code server-side

**npm**

default package manager for Node.js (alternative = yarn)

**MongoDB**

a free and open-source cross-platform document-oriented database. Classified as a NoSQL database program. Uses JSON-like documents with schemas.

**Vue.js**

An open-source progressive JavaScript framework for building user interfaces. Vue can also function as a web application framework capable of powering advanced single-page applications.

Repository: github.com

https://github.com/m-ismar/wacm\_ss18.git

Docker: operating-system-level virtualization (containerization)

Images: waecm-2018-group-13-bsp-1

*executable packages incl. the code, a runtime, libraries, environment variables, config files*

Containers: are runtime instances of images

wacmss18\_webapp\_1

wacmss18\_mongodb\_1

**Dockerfile.web (defines our container)**

# Use an official runtime as a parent image

**FROM node:carbon**

# Set the working directory

**WORKDIR /usr/wacm\_group13**

# Copy ./webapp into the container

**COPY ./webapp .**

# Install all dependencies

**RUN yarn install**

# Make ports available to the world outside this container

**EXPOSE 3000**

**EXPOSE 8443**

# Build and run the application

**CMD [ "npm", "run", "build" ]**

**CMD [ "npm", "start" ]**

Services (containers in production) define how containers should behave

**docker-compose.yml**

version: '3'

services:

webapp:

image: ebuccaneer/waecm-2018-group-13-bsp-1-web

expose:

- "8080"

ports:

- "8080:8080"

mongodb:

image: mongo:3.6.2

ports:

- "27017:27017"

**docker-compose-local.yml**

version: '3'

services:

webapp:

image: web

build:

context: .

dockerfile: Dockerfile.web

expose:

- "8080"

ports:

- "8080:8080"

mongodb:

image: mongo:3.6.2

ports:

- "27017:27017"

COMMANDS:

docker –version 17.05.0-ce

docker info

docker ps

docker images

docker exec -it wacmss18\_webapp\_1 bash

docker build creates docker image

docker login log into the docker public registry on local machine

docker push publish tagged image to your repository

Start webapp with uploaded images (runs docker-compose.yml)

docker-compose up

Start webapp with local build

docker-compose -f docker-compose-local.yml up --build

Database Layer: MongoDB v3.6.2

MongoDB GUI with embedded shell -> robo3T

Webserver with Web-Application (backend) : Node.js

**Resources:**

/counter

HTTP GET delivers the counter

HTTP POST increments the counter by 1

**Authentication:** user/password

HTTP Basic Auth or JWT

Presentation Layer (frontend): Vue.js

Text text