Handover Document

Fontys Module Management System

Nils Nieuwenhuis, Loek Ehren, Sjoerd Brauer, Tobias Derksen

Fontys School of Technology
Informatics
Module Software Factory (SOFA)

Venlo, 8th January 2018

Contents

Chapter 1

Software Architecture

Chapter 2

Build

Chapter 3

Run

Chapter 4

Run with Docker

For each part of the project there is a docker file which can be used to run the software. The docker file automates the build process and encapsulates it into a container. These containers runs on every operating system and does not need any external dependencies besides the installed docker daemon. You can compose the separate containers to a full services which includes all parts of the project.

This method is the recommended way to deploy the project to a server.

4.1 Database

```
1 #!/bin/bash
2 cd database
3 docker build -t fmms-database .
```

Listing 4.1: Build Database Container

```
1 #!/bin/bash
2 docker run -d --name fmms-database -p 5432:5432 fmms-database
```

Listing 4.2: Run Database Container

4.2 Backend

```
1 #!/bin/bash
2 cd backend
3 docker build -t fmms-backend .
```

Listing 4.3: Build Backend Container

```
1 #!/bin/bash
2 docker run -d --name fmms-backend -p 8080:8080 fmms-backend
```

Listing 4.4: Run Backend Container

4.3 Frontend

```
1 #!/bin/bash
2 cd frontend
3 docker build -t fmms-frontend .
```

Listing 4.5: Build Frontend Container

```
1 #!/bin/bash
2 docker run -d --name fmms-frontend -p 4200:4200 fmms-frontend
```

Listing 4.6: Run Frontend Container

4.4 Compose

To run all parts of the software inside docker containers, Docker Compose can be used to run and supervise the docker containers. Therefore a docker compose file is needed which defines the structure of the application and the needed parameters. The following listing shows a docker compose file which contains all needed configuration to run the project on your local machine.
To use docker compose perform the following steps:

1. Install Docker and Docker-Compose
2. Build Database, Backend and Frontend as explained in sections 4.1, 4.2 and 4.3
3. Put the content of listing 4.7 into a file named „docker-compose.yml“
4. Run shell command „docker-compose up -d“ in the directory with the file created in the previous step

```
version: '2'

networks:
  fmms:
    driver: bridge

services:
  database:
    restart: always
    image: fmms-database
    ports:
      - 5432:5432
    networks:
      - fmms

  backend:
    restart: always
    image: fmms-backend
    ports:
      - 8080:8080
    environment:
      - HOST=0.0.0.0
      - PORT=8080
      - BASE=/fmms
      - DB=database:5432/modulemanagement
      - DB_USER=module
      - DB_PASSWD=fmms
      - AUTH_USER=fmms
      - AUTH_PASSWORD=fmms
    volumes:
      - maven:/root/.m2
    networks:
      - fmms

  frontend:
    restart: always
    image: fmms-frontend
    command: ["--no-live-reload", "--no-watch"]
    depends_on:
      - backend
    ports:
      - 4200:4200
    networks:
      - fmms

volumes:
  maven:
    driver: local
```

Listing 4.7: Docker Compose File