Web-Programmierung, 4-3131-TIT20B

Kurs B

Gruppe 1

Mitglieder: Oskar Droescher, Kai Arne Möbert, Mario Schenkewitz, Florian Schoon, Jos Trautermann

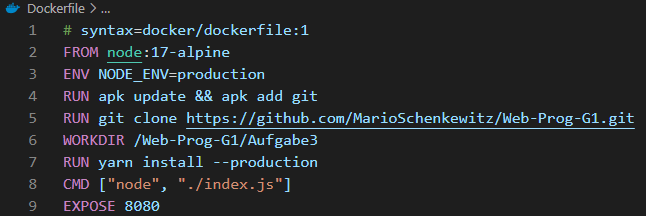
# Aufgabe 3 Dokumentation

Aufgabenteile:

1. Erstellen die eine Docker-Konfigurationsdatei und konfigurieren Sie das im Folgenden beschriebene Verhalten.
   1. Die Projektdateien (HTML-, CSS-Dateien, etc.) sollen aus ihrem jeweiligen GIT-Repository geladen werden.
   2. Installieren Sie ggf. einen node.js server (kann ggf. in einem base image inkludiert sein).
   3. Starten diesen node.js server inkl. entsprechendem Skript (z.B. express.js) das Ihre Webseite bereitstellen soll.
   4. Programmieren Sie das node.js Skript auf eine Weise das der Server auf dem Port 8080 lauscht.
   5. Konfigurieren Sie ihre Docker-Konfigurationsdatei so, dass der node.js-Server nach dem starten des Docker Containers, auf dem Host-System erreichbar ist bzw. per "http://localhost:8080/" erreicht werden kann.
   6. Dokumentieren sie mit welchen Parametern ihre docker Datei ausgeführt werden muss (z.B. docker run -t ...) damit die beschriebenen Anforderungen erfüllt werden.

Als Grundlage für die Webseite dienen die Dateien im Ordner Aufgabe3 der Projektdateien (siehe GitHub).

Dockerfile:



Beschreibung der Zeilen:

**# syntax=docker/dockerfile:1**   
Gibt an welche Syntax genutzt wird, wenn das Dockerfile analysiert wird.

In Zeile 2 wird mit „**FROM node:12-alpine**“ angegeben welches Image geladen werden soll, hierbei handelt es sich um das offizielle Node Image (<https://hub.docker.com/_/node>).

Die Umgebung wird nun auf „production“ gesetzt, was die Leistung verbessert (vgl. <https://docs.docker.com/language/nodejs/build-images/>).

Zeile 4 „**RUN apk update && apk add git**“ aktualisiert zunächst die verfügbaren Pakete und installiert dann git im Image, da wir in Zeile 5 git benutzen um unser Projekt Repository in das Image zu laden.

Das Verzeichnis wird zu /Web-Prog-G1/Aufgabe3 geändert, da sich hier die Webseite für diese Aufgabe befindet, im Anschluss darauf werden alle Abhängigkeiten des Projekts mit „**yarn install –production**“ installiert. Dies erstellt die node\_modules, sowie die package.json Dateien.

In Zeile 8 wird der Befehl „**node ./index.js“** ausgeführt um die Webseite zu starten (der Port ist in der Datei als 8080 angegeben), daraufhin wird in Zeile 9 Port 8080 nach außen freigegeben.

Um aus diesem Dockerfile ein Image zu erstellen muss das zuerst Dockerfile heruntergeladen werden. Danach kann über eine Kommandozeile zum Dockerfile navigiert werden und zum Beispiel mit dem Befehl   
**$ docker build -t gruppe\_b1\_aufgabe3 .**   
das Image erstellt werden.   
Das Image kann mit **$ docker run -p 8080:8080 gruppe\_b\_aufgabe3** oder über Docker Desktopgestartet werden. Achtung in Docker Desktop muss beim Start in den optionalen Einstellungen noch der localhost Port eingegeben werden.

**ACHTUNG**: Falls noch ein Image vom alten Stand des Repositories gemacht wurde, kann es sein, dass das Image aus dem Cache erstellt wird, hierbei wird das Repository nicht neu heruntergeladen.  
In diesem Fall muss der Befehl **$ docker build --no-cache -t gruppe\_b1\_aufgabe3 .** verwendet werden.