Oktay - **Deploy Angular Application in Docker Container**

Thursday, November 5, 2020 12:28 PM

1. Deploy Angular Application in Docker Container

DESCRIPTION

Deploy the Angular application in Docker. The Angular application should be built with the Angular CLI along with Docker Compose for development and production.

Problem Statement Scenario:
HTQual Technology Solutions hired you as a MEAN Stack Developer. The organization decided to implement DevOps to develop and deliver the products. Since HTQual is an Agile organization, they follow Scrum methodology to develop the projects incrementally. The Company decided to develop their website on Mean stack. Since you are the MEAN stack developer, you have to demonstrate that deploying the Angular application on Docker is always the best approach to develop a project and test it incrementally. You agreed upon the following:

- Setting up an image for code development
- Build the application in Docker and host it in Docker Hub
- List the advantages, disadvantages, and document the tasks involved

Your goal is to demonstrate the Angular application and run it in Docker container.

You must use the following tools:
- Docker – To package the application in a Docker container
- Node.js – To support the Angular application with the required node modules
- Angular CLI – To execute and bundle the dependencies together
- Linux (Ubuntu) – As a base operating system to start and execute the project

Following requirements should be met:
- **Document the step-by-step process from the initial installation to the final production**
- **Run the Angular application successfully in the Docker container**
- **Use Docker Compose to manage the Angular application running inside the Docker container**

1- Installing Angular CLI globally on Ubuntu 20.04 LTS
sudo apt install npm => to install npm package manager
sudo npm install -g @angular/cli@7.3.9 => This will install Angular globally in your system.

```
ares@kubernetes-master:~$ sudo npm install -g @angular/cli@7.3.9
npm WARN deprecated chokidar@2.0.4: Chokidar 2 will break on node v14+. Upgrade to chokidar 3 with 15x less dependencies
.
npm WARN deprecated fsevents@1.2.13: fsevents 1 will break on node v14+ and could be using insecure binaries. Upgrade to
 fsevents 2.
npm WARN deprecated urix@0.1.0: Please see https://github.com/lydell/urix#deprecated
npm WARN deprecated resolve-url@0.2.1: https://github.com/lydell/resolve-url#deprecated
/usr/local/bin/ng -> /usr/local/lib/node_modules/@angular/cli/bin/ng
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@^1.2.2 (node_modules/@angular/cli/node_modules/chokidar/node_mo
dules/fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"an
y"} (current: {"os":"linux","arch":"x64"})

+ @angular/cli@7.3.9
added 293 packages from 185 contributors in 22.305s
ares@kubernetes-master:~$
```

2- Create a new App
ng new MyApp => to create an app called MyApp. Enter "Y" to create a new app and then select "CSS".
Once done you will see "Successfully initialize git" message

```
ares@kubernetes-master:~$ ng new MyApp
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use?
❯ CSS
  SCSS   [ http://sass-lang.com/documentation/file.SASS_REFERENCE.html#syntax ]
  Sass   [ http://sass-lang.com/documentation/file.INDENTED_SYNTAX.html       ]
  Less   [ http://lesscss.org                                                 ]
  Stylus [ http://stylus-lang.com                                             ]
```

```
found 8 vulnerabilities (5 low, 3 high)
  run `npm audit fix` to fix them, or `npm audit` for details
    Successfully initialized git.
ares@kubernetes-master:~$
```

Incase if you receive vulnerabilities during installation, make sure you run the following command to fix the issues
sudo npm audit fix --force => to fix known vulnerabilities

```
ares@kubernetes-master:~/MyApp$ sudo npm audit fix --force
npm WARN using --force I sure hope you know what you are doing.
npm WARN deprecated request@2.88.2: request has been deprecated, see
```

3- Now go inside the directory created by Angular and create a Docker file with the written syntax below.
cd MyApp => to go inside the directory created earlier.
sudo touch Dockerfile => to create a Dockerfile
sudo nano Dockerfile => to open Dockerfile with nano text editor.

```
# base image
FROM node:14.15.0

# install chrome for protractor tests
RUN wget -q -O - https://dl-ssl.google.com/linux/linux_signing_key.pub | apt-key add -
RUN sh -c 'echo "deb [arch=amd64] http://dl.google.com/linux/chrome/deb/ stable main" >> /etc/apt/sources.list.d/google.list'
RUN apt-get update && apt-get install -yq google-chrome-stable

# set working directory
RUN mkdir /usr/src/app
WORKDIR /usr/src/app

# add `/usr/src/app/node_modules/.bin` to $PATH
ENV PATH /usr/src/app/node_modules/.bin:$PATH

# install and cache app dependencies
COPY package.json /usr/src/app/package.json
RUN npm install
RUN npm install -g @angular/cli@1.7.1

# add app
COPY . /usr/src/app

# start app
CMD ng serve --host 0.0.0.0
```

```
ares@kubernetes-master: ~/Projects/AngularCLI/myangularapp
 GNU nano 4.8                        Dockerfile
# base image
FROM node:14.15.0

# install chrome for protractor tests
RUN wget -q -O - https://dl-ssl.google.com/linux/linux_signing_key.pub | apt
RUN sh -c 'echo "deb [arch=amd64] http://dl.google.com/linux/chrome/deb/ sta
RUN apt-get update && apt-get install -yq google-chrome-stable

# set working directory
RUN mkdir /usr/src/app
WORKDIR /usr/src/app

# add `/usr/src/app/node_modules/.bin` to $PATH
ENV PATH /usr/src/app/node_modules/.bin:$PATH

# install and cache app dependencies
COPY package.json /usr/src/app/package.json
RUN npm install
RUN npm install -g @angular/cli@1.7.1

# add app
COPY . /usr/src/app

# start app
CMD ng serve --host 0.0.0.0
```

sudo touch .dockerignore => To create a file called .dockerignore and enter the following inside this file

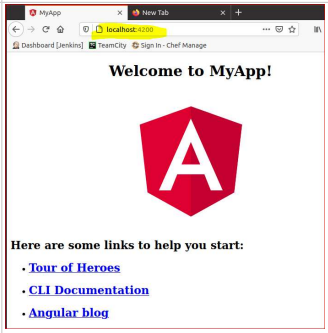
```
node_modules
.git
```

```
ares@kubernetes-master: ~/MyApp
 GNU nano 4.8                        .dockerignore
node_modules
.git
```

4- Run the application
sudo ng serve --open =>

```
ares@kubernetes-master:~/MyApp$ sudo ng serve --open
** Angular Live Development Server is listening on localhost:4200, open your browser on ht
tp://localhost:4200/ **
 93% after chunk asset optimization SourceMapDevToolPlugin es2015-polyfills.js generate So
 93% after chunk asset optimization SourceMapDevToolPlugin es2015-polyfills.js attach Sour

Date: 2020-11-04T03:15:02.678Z
Hash: fd7bb352ca8d7d6a6295
Time: 13500ms
chunk {es2015-polyfills} es2015-polyfills.js, es2015-polyfills.js.map (es2015-polyfills) 2
85 kB [initial] [rendered]
chunk {main} main.js, main.js.map (main) 11.5 kB [initial] [rendered]
chunk {polyfills} polyfills.js, polyfills.js.map (polyfills) 236 kB [initial] [rendered]
chunk {runtime} runtime.js, runtime.js.map (runtime) 6.08 kB [entry] [rendered]
chunk {styles} styles.js, styles.js.map (styles) 16.3 kB [initial] [rendered]
chunk {vendor} vendor.js, vendor.js.map (vendor) 3.77 MB [initial] [rendered]
i [wdm]: Compiled successfully.
```

5- Open your browser and type in the following
http://localhost:4200



6- Build and tag the docker image
#sudo docker build -t myangularapp . => to build a docker image called myangularapp. Building process of this image may take up to 5 mins.

```
ares@kubernetes-master:~/MyApp$ sudo docker build -t myangularapp .
Sending build context to Docker daemon  606.2kB
Step 1/12 : FROM node:9.6.1
 ---> 29831ba76d93
Step 2/12 : RUN wget -q -O - https://dl-ssl.google.com/linux/linux_signing_key.pub | apt-key add -
 ---> Running in 858c0f30d749
OK
Removing intermediate container 858c0f30d749
 ---> bc0db421b8e3
Step 3/12 : RUN sh -c 'echo "deb [arch=amd64] http://dl.google.com/linux/chrome/deb/ stable main" >>
st.d/google.list'
 ---> Running in 6858ddd74bbb
Removing intermediate container 6858ddd74bbb
 ---> d70b5a2c1665
```

To check all images in your Docker
sudo docker images => this will show you all images in your docker

```
ares@kubernetes-master:~/MyApp$ docker images
REPOSITORY                          TAG                 IMAGE ID            CREATED             SIZE
myangularapp                        latest              31f9f6e9e397        11 seconds ago      1.72GB
quay.io/coreos/flannel              v0.13.0             e708f4bb69e3        2 weeks ago         57.2MB
k8s.gcr.io/kube-proxy               v1.19.3             cdef7632a242        2 weeks ago         118MB
k8s.gcr.io/kube-controller-manager  v1.19.3             9b60aca1d818        2 weeks ago         111MB
k8s.gcr.io/kube-scheduler           v1.19.3             aaefbfa906bd        2 weeks ago         45.7MB
k8s.gcr.io/kube-apiserver           v1.19.3             a301be0cd44b        2 weeks ago         119MB
k8s.gcr.io/etcd                     3.4.13-0            0369cf4303ff        2 months ago        253MB
weaveworks/weave-npc                2.7.0               db66692318fc        3 months ago        41MB
weaveworks/weave-kube               2.7.0               a8ef3e215aac        3 months ago        113MB
k8s.gcr.io/pause                    3.2                 80d28bedfe5d        8 months ago        683kB
node                                9.6.1               29831ba76d93        2 years ago         676MB
ares@kubernetes-master:~/MyApp$
```

7- Now reload the container by executing the command below
docker run -it -v ${PWD}:/usr/src/app -v /usr/src/app/node_modules -p 4200:4200 - -rm myangularapp =>

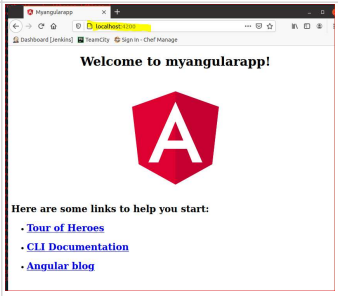
```
ares@kubernetes-master:~/Projects/AngularCLI/myangularapp$ docker run -it -v ${PWD}:/usr/src/app -v /usr/src/app/node_modules -p
4200:4200 --rm myangularapp
WARNING: This is a simple server for use in testing or debugging Angular applications
locally. It hasn't been reviewed for security issues.

Binding this server to an open connection can result in compromising your application or
computer. Using a different host than the one passed to the "--host" flag might result in
websocket connection issues. You might need to use "--disableHostCheck" if that's the
case.
** Angular Live Development Server is listening on 0.0.0.0:4200, open your browser on http://localhost:4200/ **

Date: 2020-11-05T16:57:44.414Z
Hash: 1e834b8942801644a08d
Time: 10471ms
chunk {es2015-polyfills} es2015-polyfills.js, es2015-polyfills.js.map (es2015-polyfills) 285 kB [initial] [rendered]
chunk {main} main.js, main.js.map (main) 11.5 kB [initial] [rendered]
chunk {polyfills} polyfills.js, polyfills.js.map (polyfills) 236 kB [initial] [rendered]
chunk {runtime} runtime.js, runtime.js.map (runtime) 6.08 kB [entry] [rendered]
chunk {styles} styles.js, styles.js.map (styles) 16.3 kB [initial] [rendered]
chunk {vendor} vendor.js, vendor.js.map (vendor) 3.77 MB [initial] [rendered]
i [wdm]: Compiled successfully.
```

8- Run the Angular application in Docker container in detach mode
Make sure image on Docker for NodeJs is greater than V10.***

docker run -d -v ${PWD}:/usr/src/app -v /usr/src/app/node_modules -p 4200:4200 - -rm myangularapp =>

```
ares@kubernetes-master:~$ docker run -d -v ${PWD}:/usr/src/app -v /usr/src/app/node_modules -p 4200:4200 --rm myangular
app
758589af10a53fdea9b5e562b3efe69c44ac56572c9a274e6fd3e271c5979429
ares@kubernetes-master:~$ docker ps -a
```

9- You can run the application by using your browser
http://localhost:4200

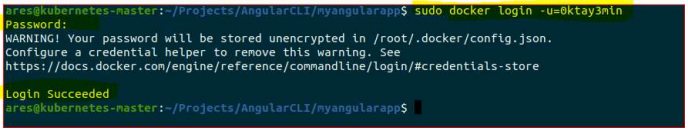
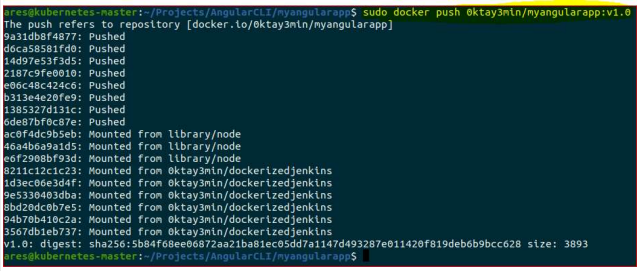
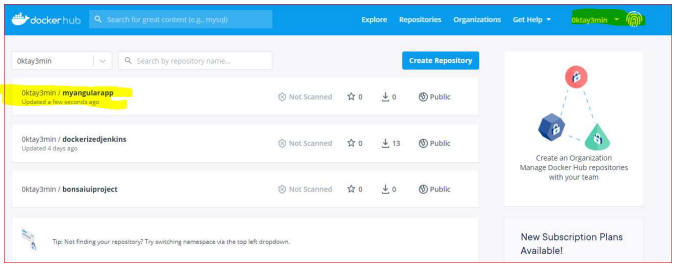


10- Docker images can be seen by entering the following command
sudo docker images => all images in Docker

```
ares@kubernetes-master:~/Projects/AngularCLI/myangularapp$ docker images
REPOSITORY                          TAG                 IMAGE ID            CREATED             SIZE
myangularapp                        latest              5cc6b0417b55        6 minutes ago       1.86GB
node                                14.15.0             b90fa0d7cbd1        8 days ago          943MB
quay.io/coreos/flannel              v0.13.0             e708f4bb69e3        3 weeks ago         57.2MB
k8s.gcr.io/kube-proxy               v1.19.3             cdef7632a242        3 weeks ago         118MB
k8s.gcr.io/kube-scheduler           v1.19.3             aaefbfa906bd        3 weeks ago         45.7MB
k8s.gcr.io/kube-controller-manager  v1.19.3             9b60aca1d818        3 weeks ago         111MB
k8s.gcr.io/kube-apiserver           v1.19.3             a301be0cd44b        3 weeks ago         119MB
k8s.gcr.io/etcd                     3.4.13-0            0369cf4303ff        2 months ago        253MB
weaveworks/weave-npc                2.7.0               db66692318fc        3 months ago        41MB
weaveworks/weave-kube               2.7.0               a8ef3e215aac        3 months ago        113MB
k8s.gcr.io/pause                    3.2                 80d28bedfe5d        8 months ago        683kB
ares@kubernetes-master:~/Projects/AngularCLI/myangularapp$
```

11- To tag a running Image
sudo docker tag 5cc6b0417b55 0ktay3min/myangularapp:v1.0 =>sudo

```
ares@kubernetes-master:~/Projects/AngularCLI/myangularapp$ sudo docker tag 5cc6b0417b55 0ktay3min/myangularapp:v1.0
ares@kubernetes-master:~/Projects/AngularCLI/myangularapp$
```

docker tag <Image Name> <DockerHubUSerName> / <ImageName>:<TagName>	
12- Now login to your DockerHub account by typing the following command # sudo docker login -u=0ktay3min => once enter is pressed, docker will ask you to enter the Dockerhub password	
13- Once Login authentication is completed, push the image to DockerHub account. #sudo docker push <DockerHubUserName>/<imageName>:<version> # sudo docker push 0ktay3min/myangularapp:v1.0 =>	 
Github Repository	https://github.com/0ktay3min/DeployingAngularAppInDockerContainer.git