Here they follow a few instructions on how to make a Docker image with a simple Web service

1. Make your Web service a stand alone `.jar`, with all dependencies included
2. Write a `Dockerfile` as it follows

```
FROM openjdk:8
COPY ./SOAP-WebService-1.0-SNAPSHOT.jar /usr/app/
WORKDIR /usr/app
EXPOSE 8080
ENTRYPOINT ["java", "-cp", "SOAP-WebService-1.0-SNAPSHOT.jar",
"it.sapienza.softeng.soapws.Server"]
```

3. After starting Docker engine in your specific OS, and in the same folder where the `Dockerfile` and the `.jar` are placed, build the image

   % docker build .

4. After a while (depending on the network bandwidth, computing performance, etc.) the process should be completed, you can check the new image with

   % docker images

   And you will see something like

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
<none>	<none>	f01d5e1a6e65	9 seconds ago	523MB
rmohr/activemq	latest	7c58d2d8d6af	2 years ago	569MB

5. Then you need to tag the newly created image and push on the repository.
   Here is presented how the professor did with its own repository on Docker Hub (the student can modify if willing to publish on another repository)

   % docker login
   Authenticating with existing credentials...
   Login Succeeded

   % docker image tag f01d5e1a6e65 mecellone/course-softengineering:soapws
   % docker image ls

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
soapws	latest	f01d5e1a6e65	2 hours ago	523MB
rmohr/activemq	latest	7c58d2d8d6af	2 years ago	569MB

   % docker image push mecellone/course-softengineering:soapws
   The push refers to repository [docker.io/mecellone/course-softengineering]
   5f70bf18a086: Pushed
   45b8f2c4d3b3: Pushed
   05b4da1d24c6: Pushed

30b0cc543ee7: Pushed
26863b4714ee: Pushed
5d5962699bd5: Pushed
a42439ce9650: Pushed
26270c5e25fa: Pushed
e2c6ff462357: Pushed
soapws: digest:
sha256:8d2d1e9d3cd932ba213cccd18784f3e28f9f037797ee516cdef68cb96ebd7aec
size: 2211