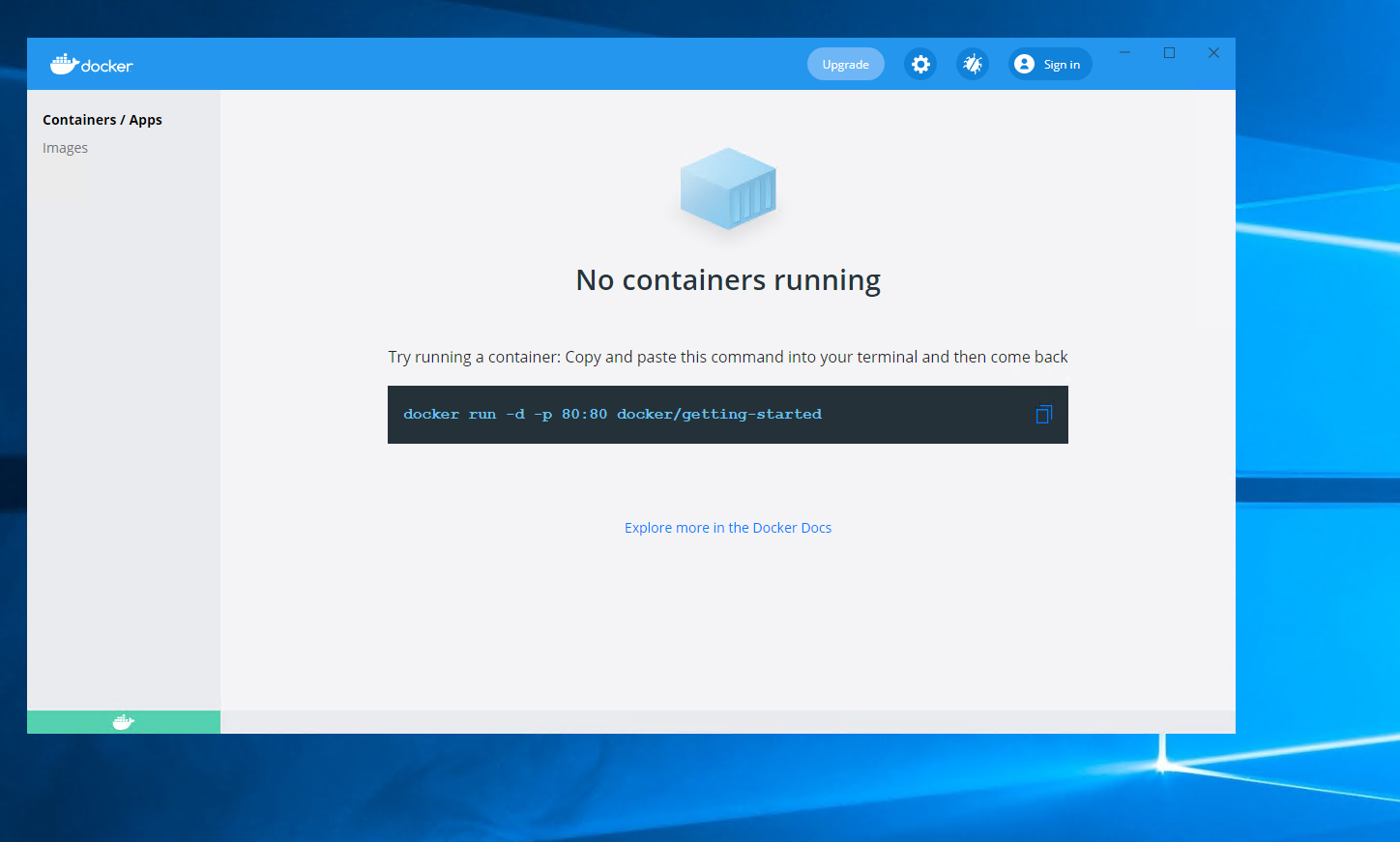
1) connect to Windows Server 2019 via RDP

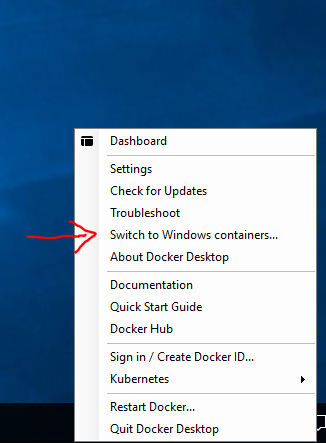
2) Download Docker Desktop for Windows and Install (Make sure that you checkmark 'Hyper-V' during the installation)

3) You will be asked to Restart the Server

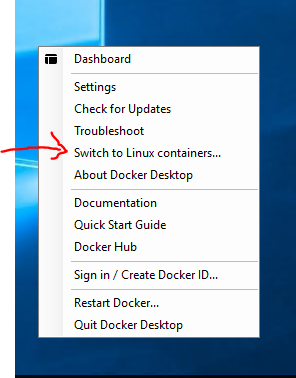
4) Docker Desktop App should be starting automatically after restart. If not, double-click on Docker Desktop. After Docker booted you should see:



5) Right-click on the Docker symbol in the task bar. You should see:

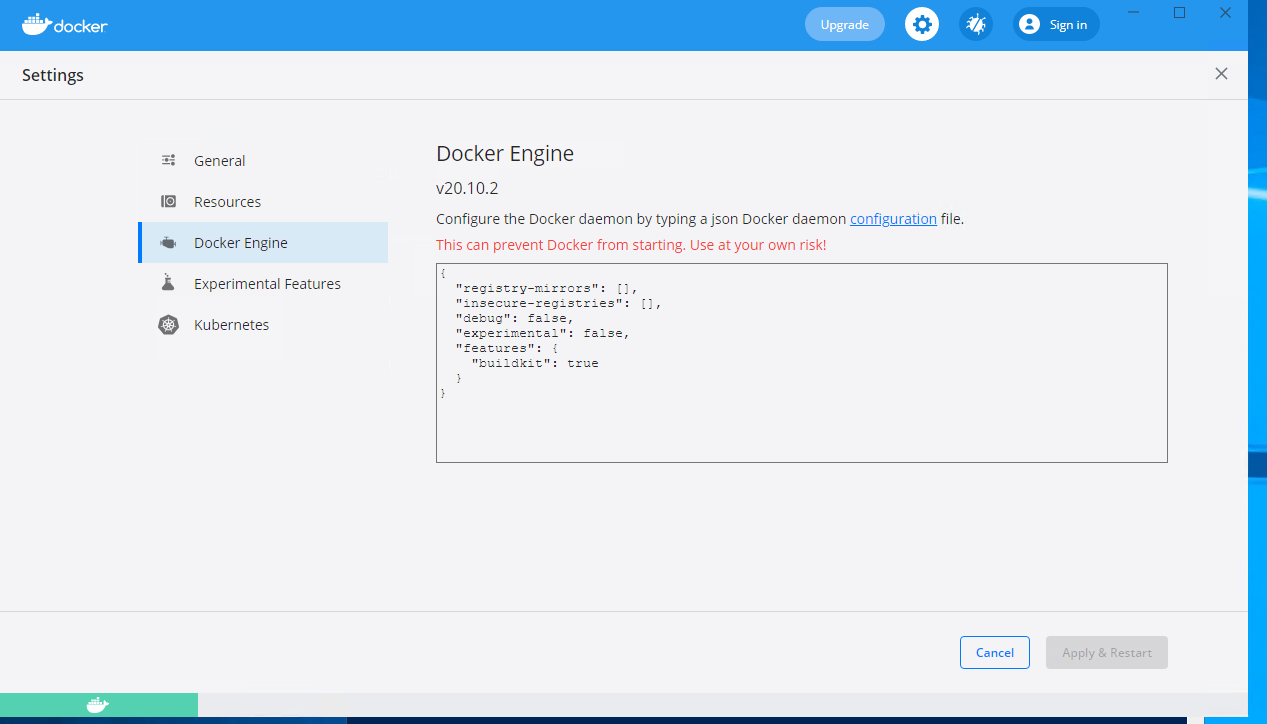


Click on „Switch to Windows containers“. Now you should see:



Click AGAIN on „Switch to Linux containers“ to return to initial state. Note that this step is necessary to execute Docker Linux commands. This will set up the Linux env.

6) Make sure you have these Docker Engine Settings:

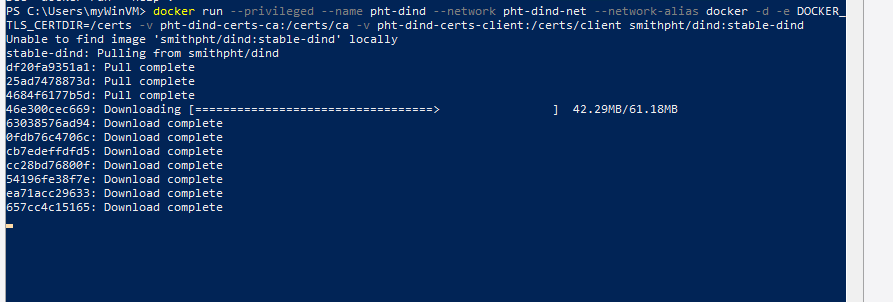


7) Open Powershell

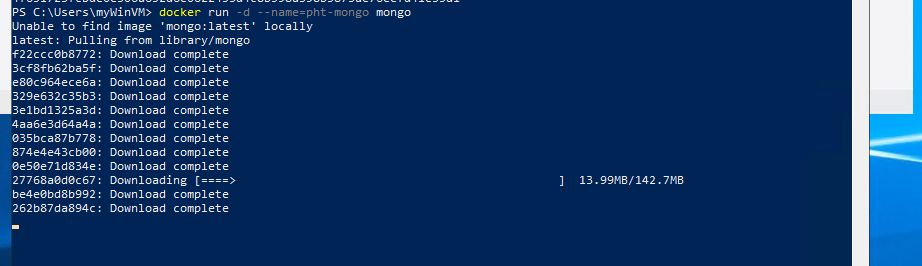
8) Type: docker network create pht-dind-net



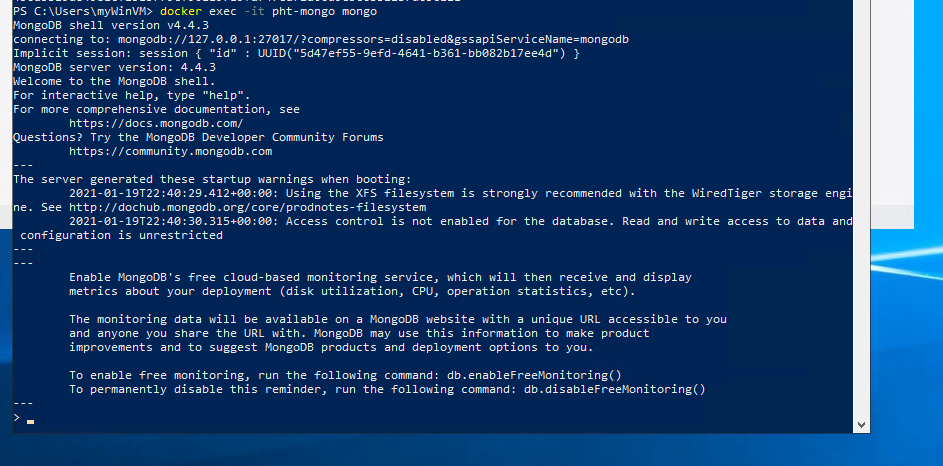
9) Type: docker run --privileged --name pht-dind --network pht-dind-net --network-alias docker -d -e DOCKER\_TLS\_CERTDIR=/certs -v pht-dind-certs-ca:/certs/ca -v pht-dind-certs-client:/certs/client smithpht/dind:stable-dind



10) Type: docker run -d --name=pht-mongo mongo

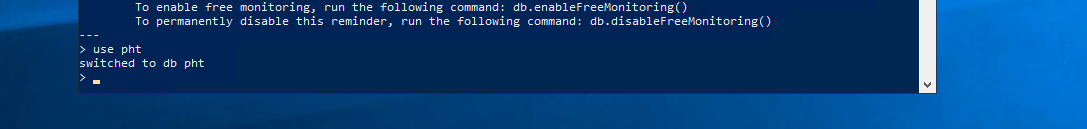


11) Type: docker exec -it pht-mongo mongo



Now, we create a first admin user for the station software.

12) Type: use pht



13) Type: db.createUser({

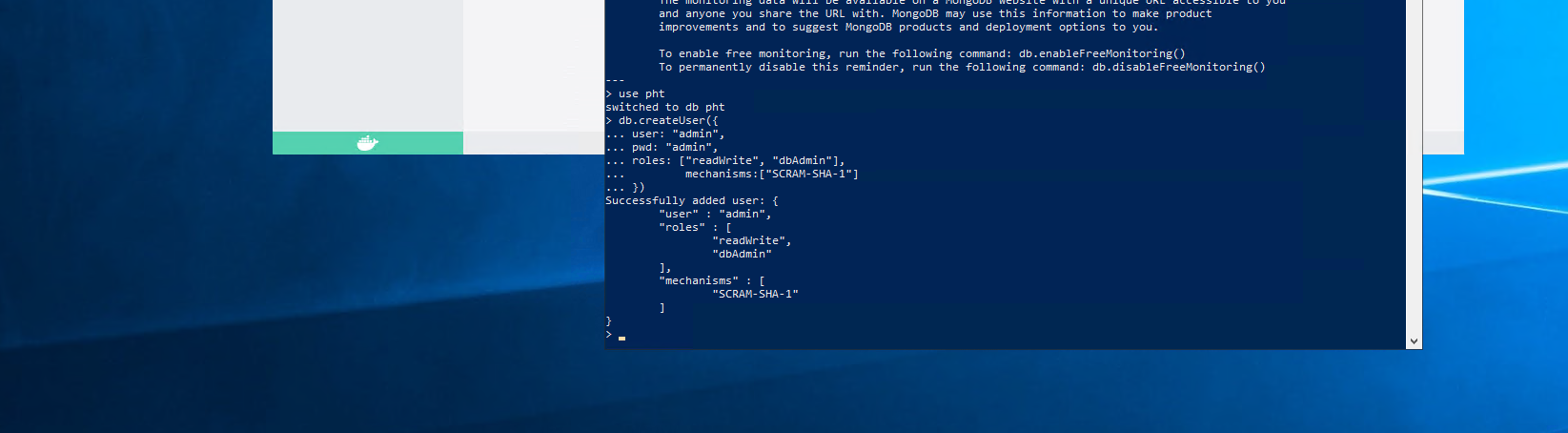
user: "admin",

pwd: "admin",

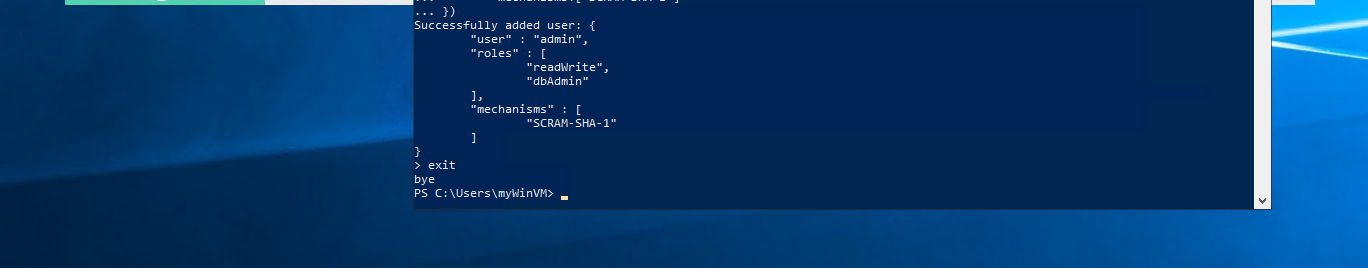
roles: ["readWrite", "dbAdmin"],

mechanisms:["SCRAM-SHA-1"]

})

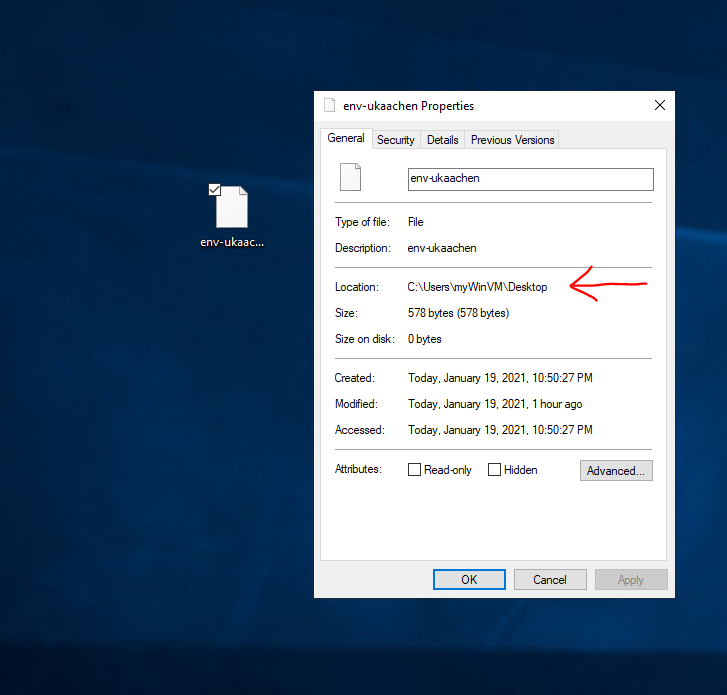


14) Type: exit



Now, we returned to the host machine.

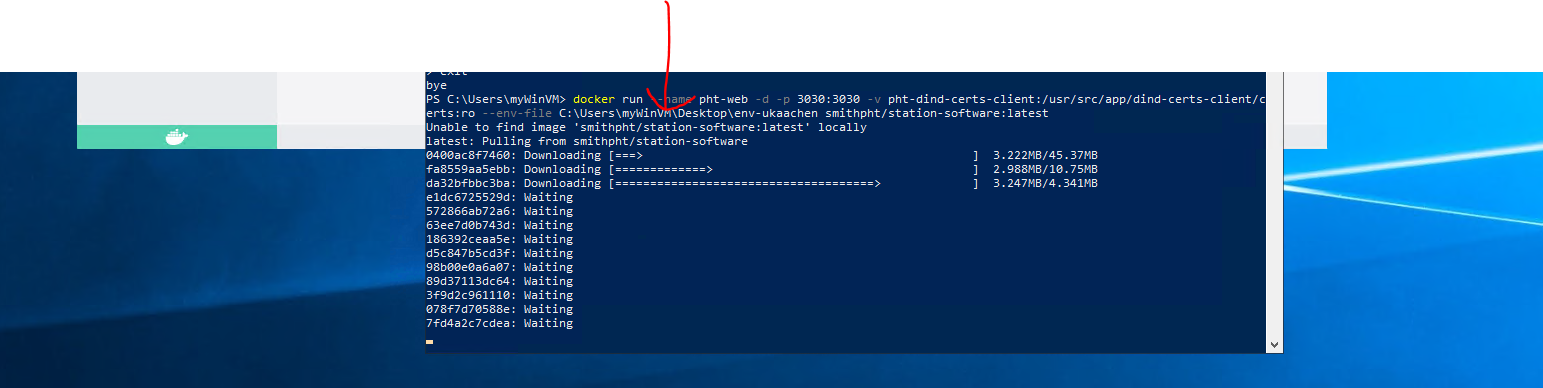
15) Now, we can start the Station Software itself. You need the „.env“-File, we provided. Make sure that you upload this to the server and remember the path to this file.



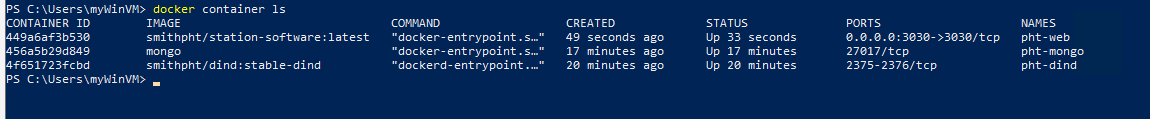
16) Type: docker run --name pht-web -d -p 3030:3030 -v pht-dind-certs-client:/usr/src/app/dind-certs-client/certs:ro --env-file <PATH\_TO\_ENV\_FILE> smithpht/station-software:latest

Replace <PATH\_TO\_ENV\_FILE> with the path to the env file:

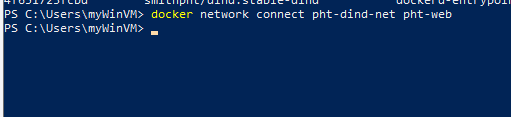
<PATH\_TO\_ENV\_FILE> = C:\Users\myWinVM\Desktop\env-ukaachen



17) After this step, you should have installed the Station Software successfully:

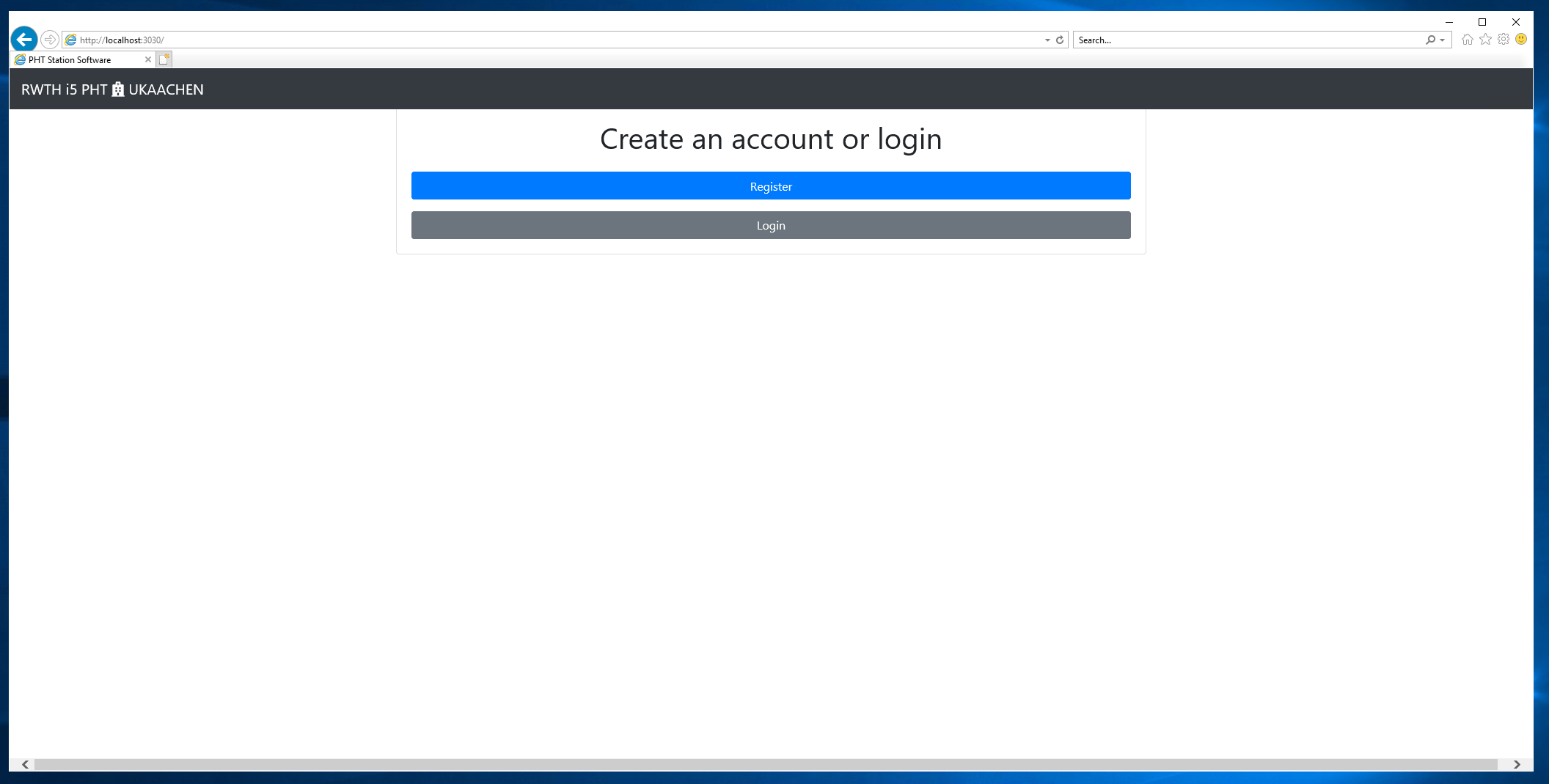


18) Type: docker network connect pht-dind-net pht-web



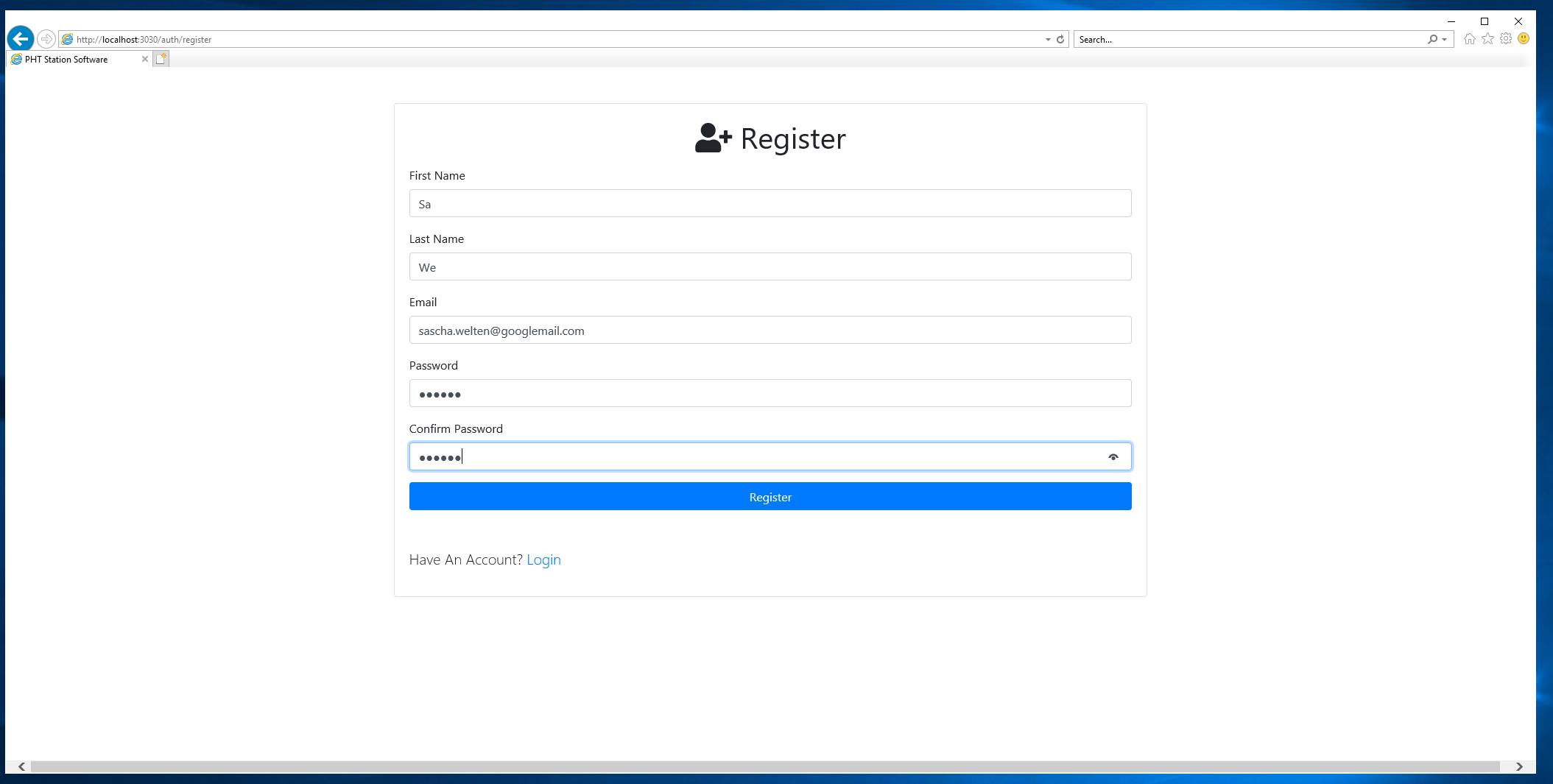
This connects the Station Software container to the Docker Engine and is the last step fort he deployment.

19) Open a browser, Type: <http://localhost:3030>

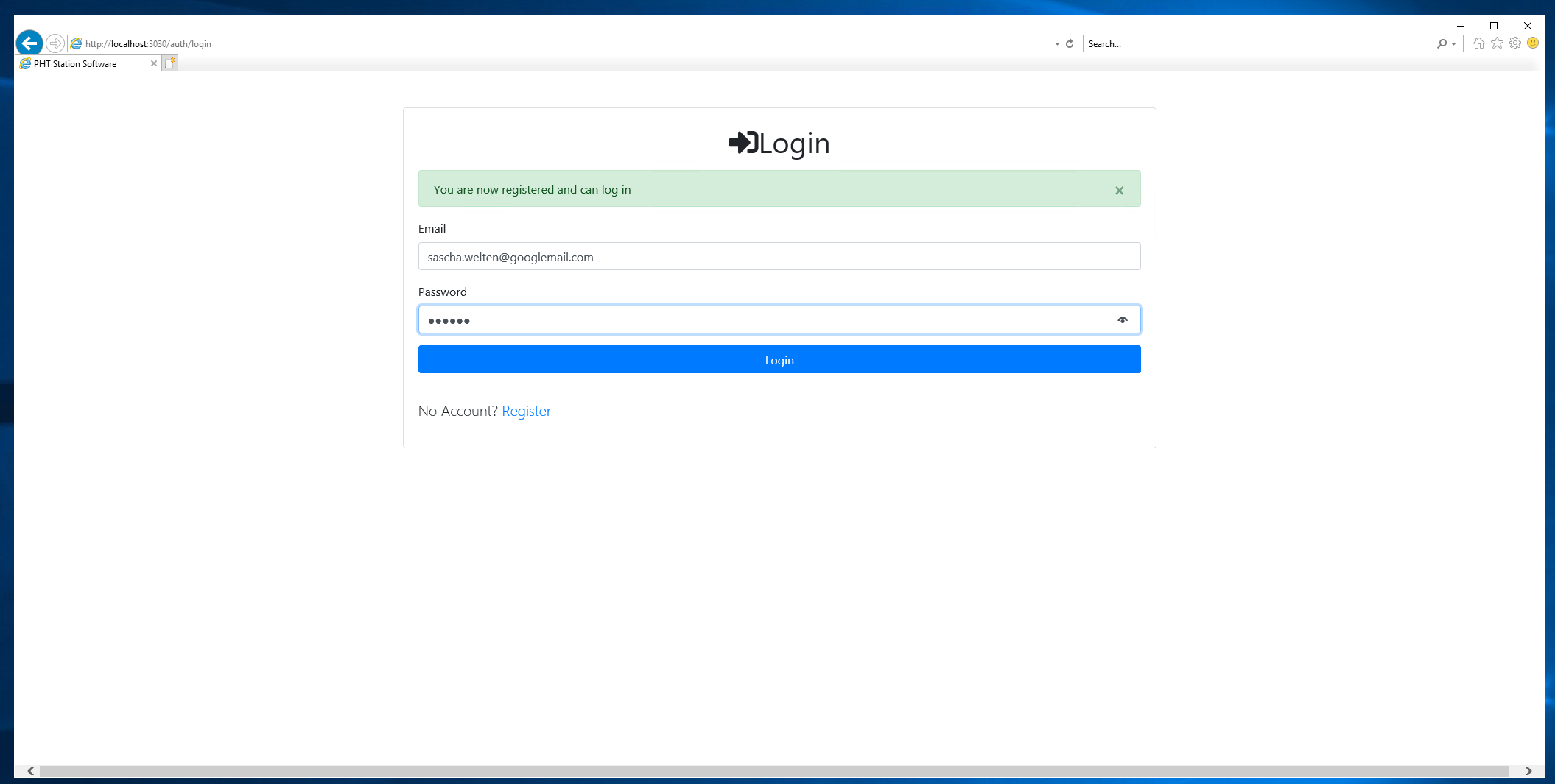


The Station Software should be prompted.

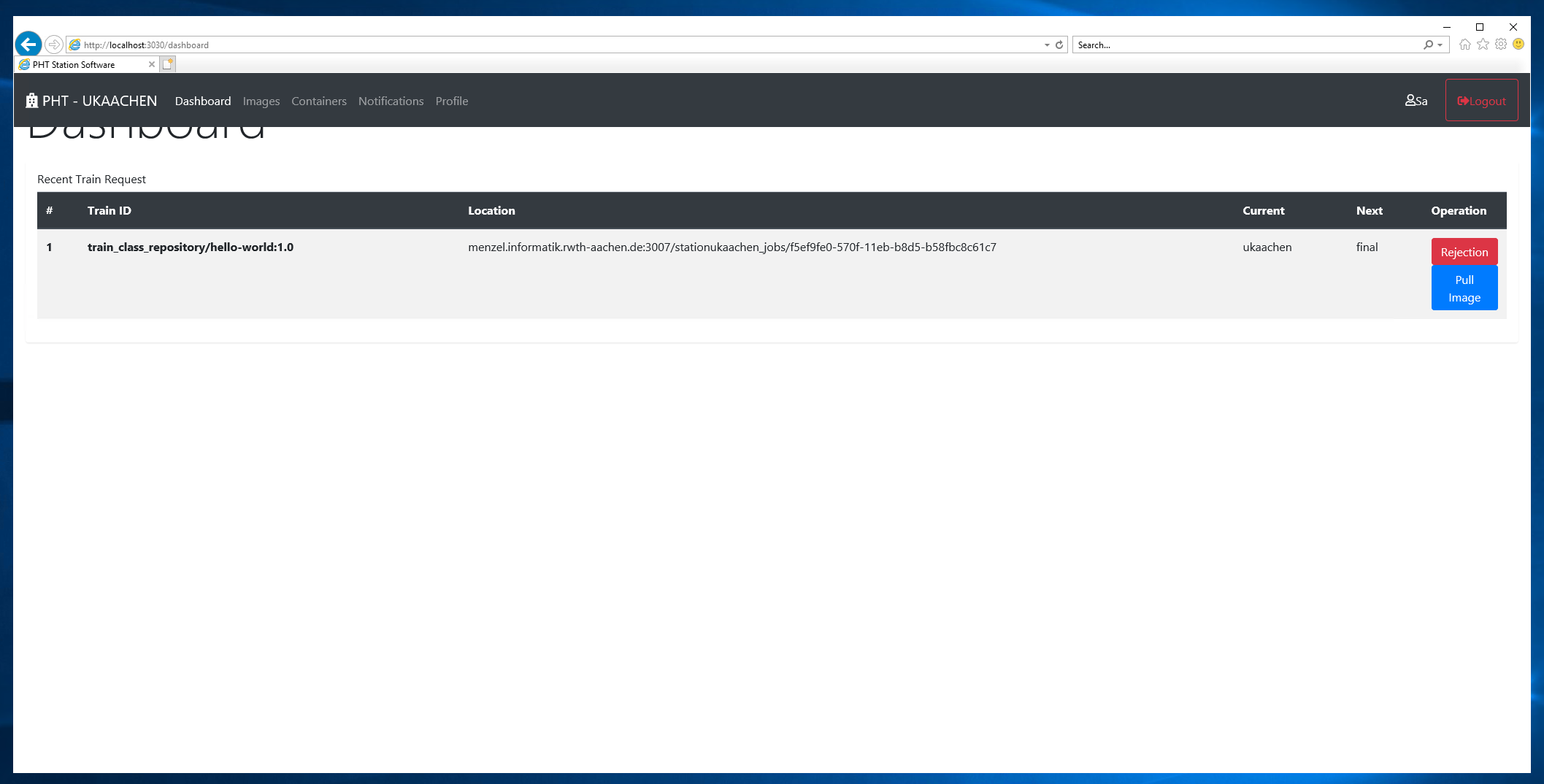
20) Now you need to register: Type in your desired credentials. Note that these credentials are stored locally in the MongoDB:



21) Login

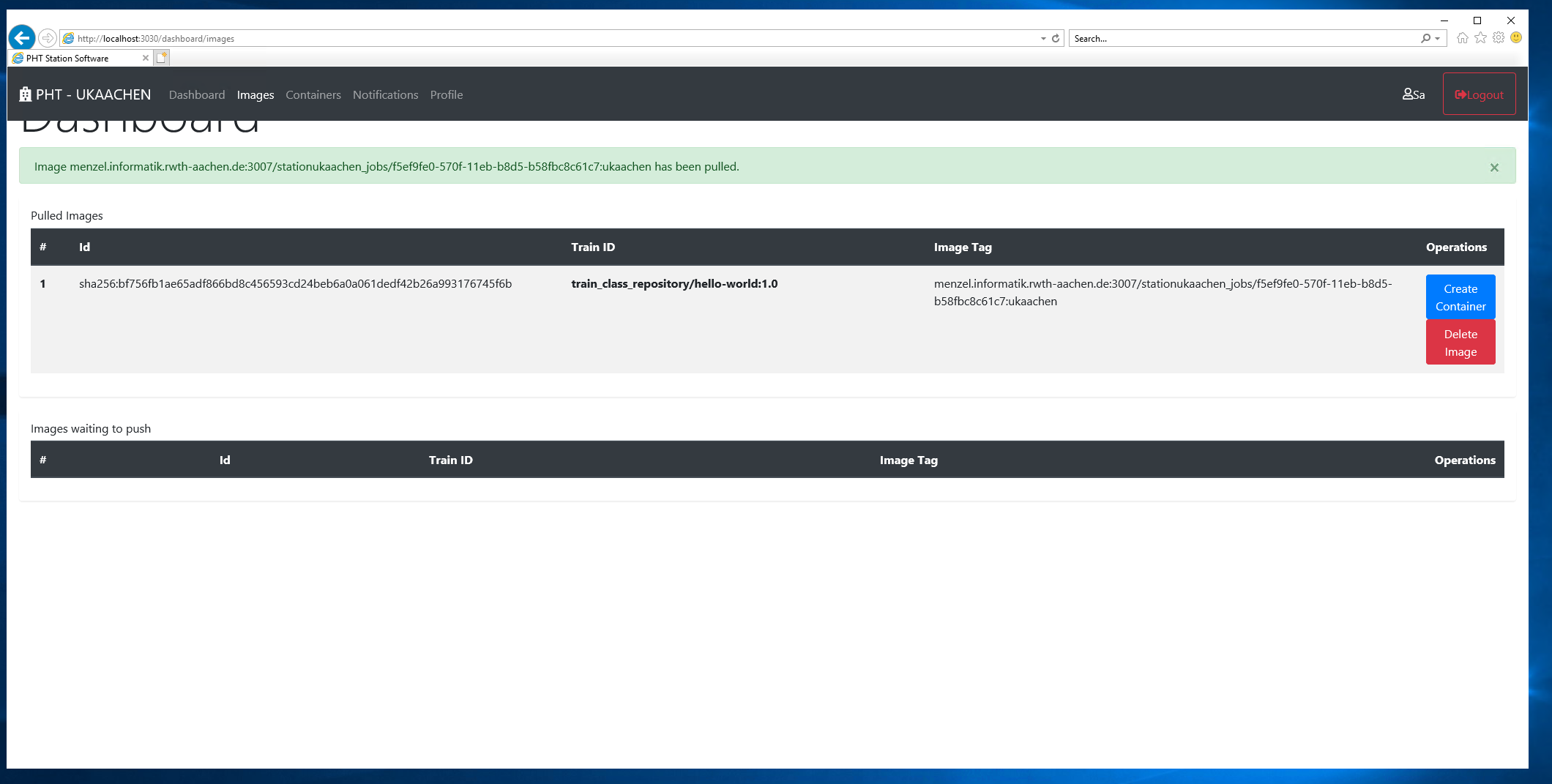


22) If the login was successful, you should see the Train Dashboard:



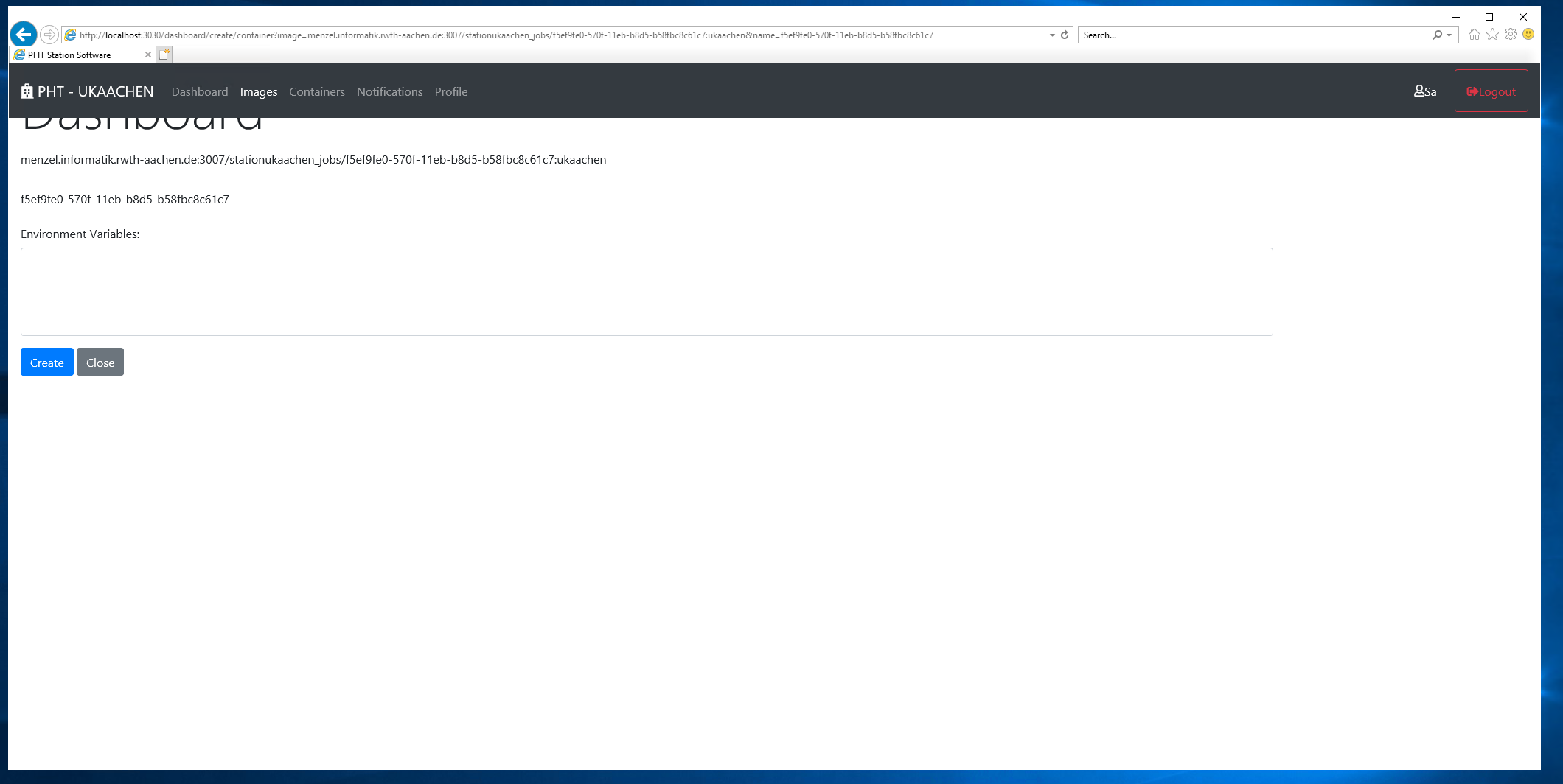
Currently, a Train train\_class\_repository/hello-world:1.0 is waiting.

23) Click on the blue button „Pull Image“:

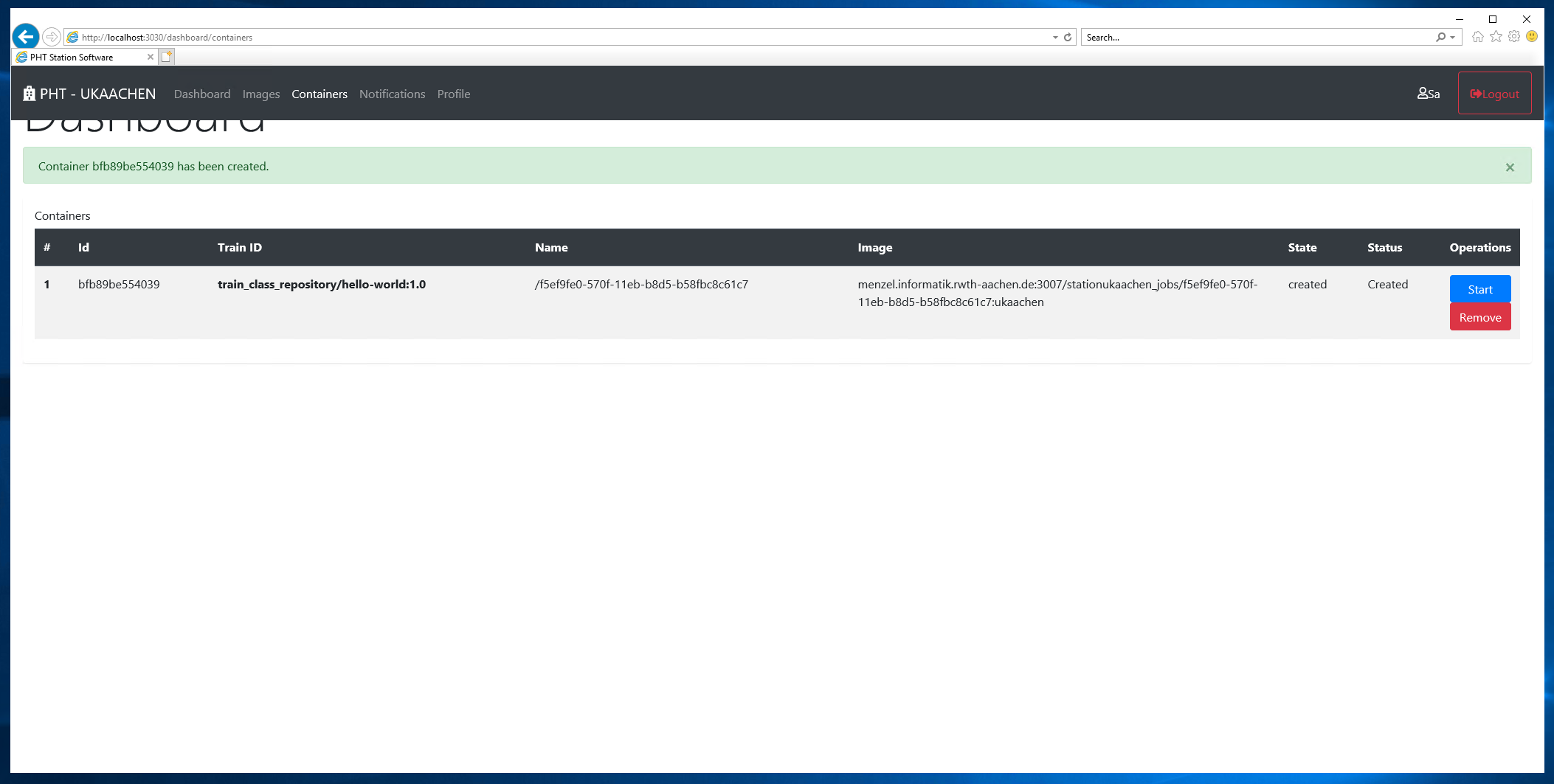


If you see a similar page, the connection to our central service is established.

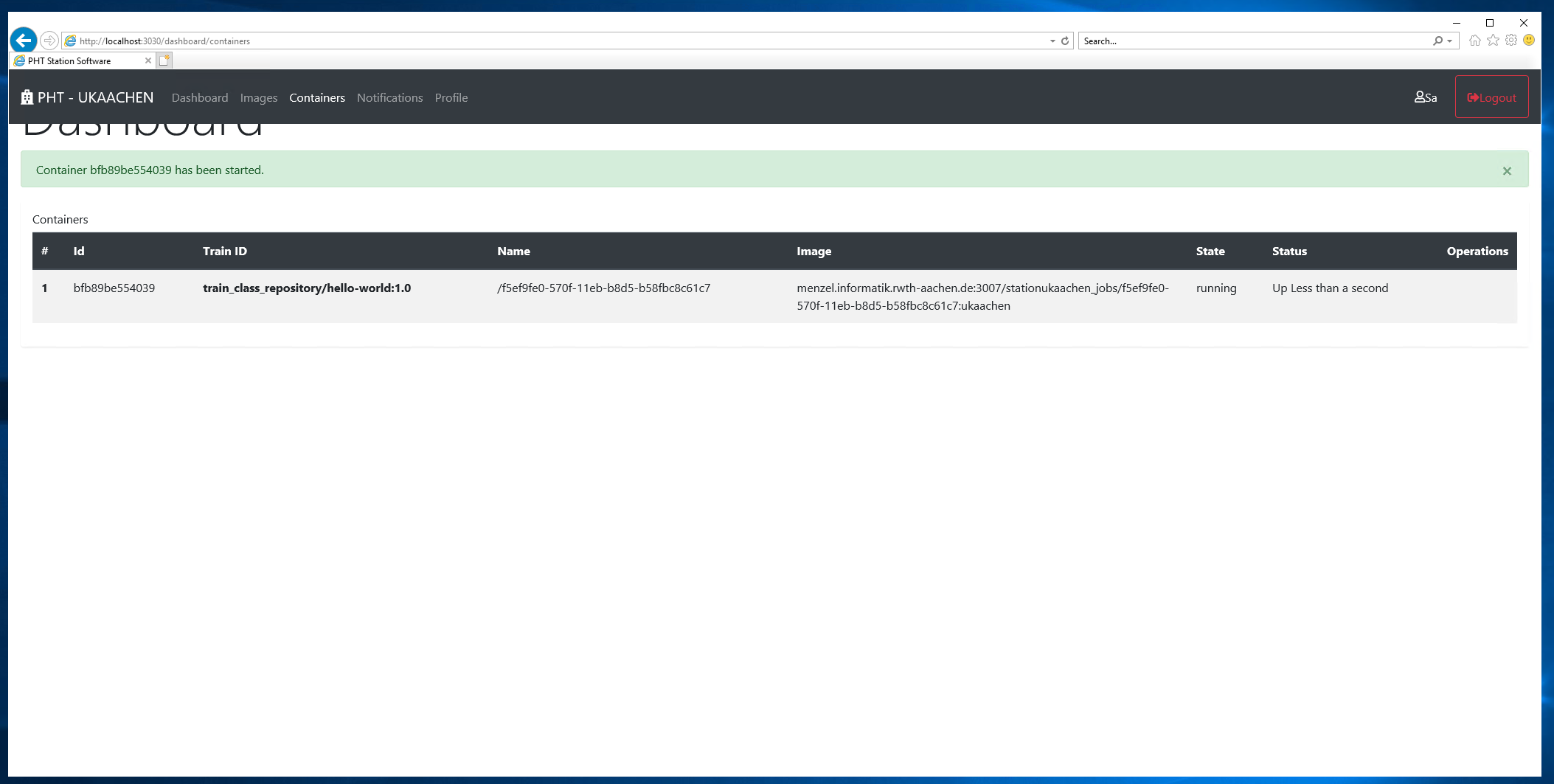
24) Click on the blue button „Create Container“:



25) Click on the blue button „Create“ since the hello world train does not need any env variables.

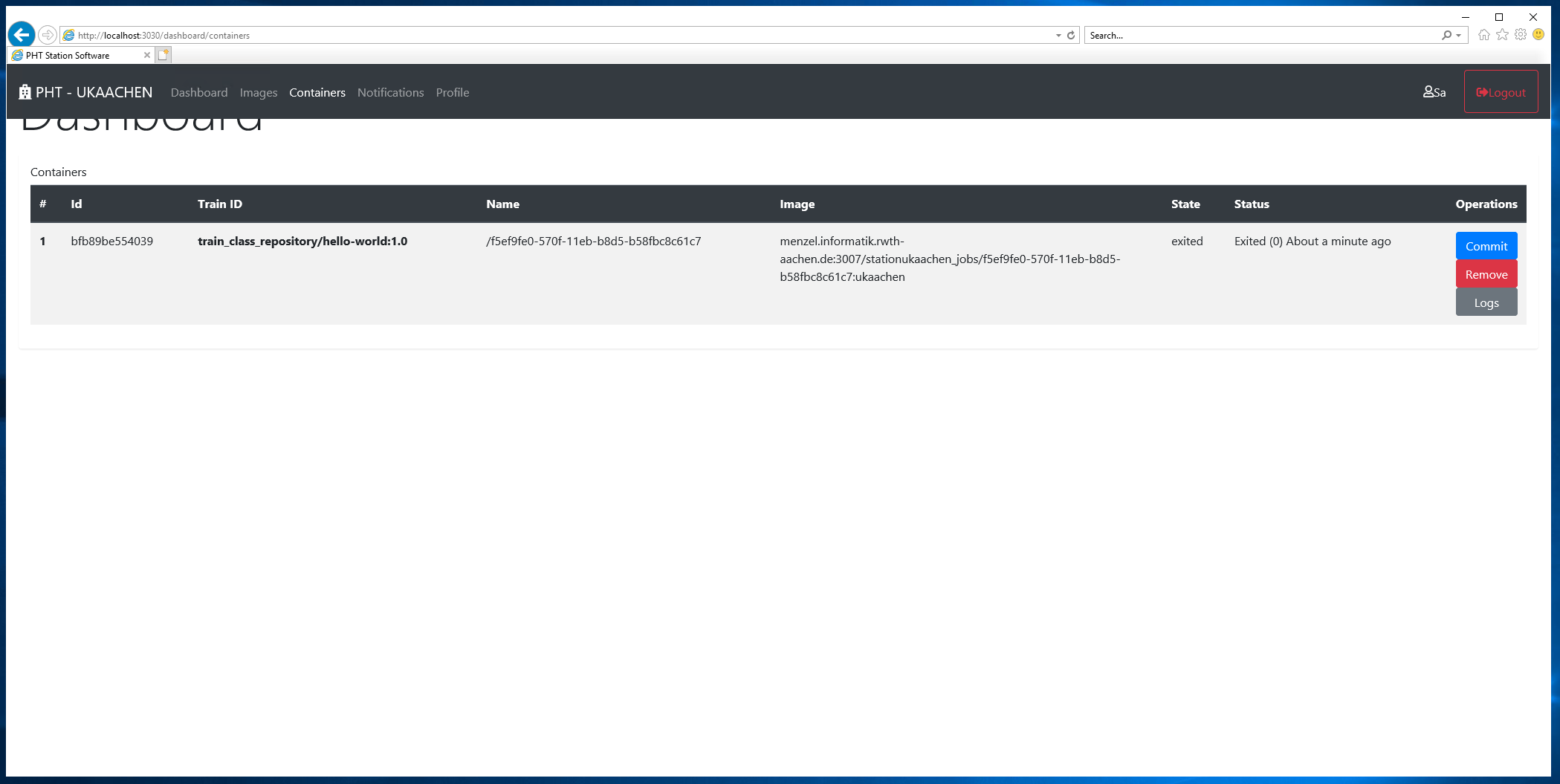


26) Click on the blue button „Start“:

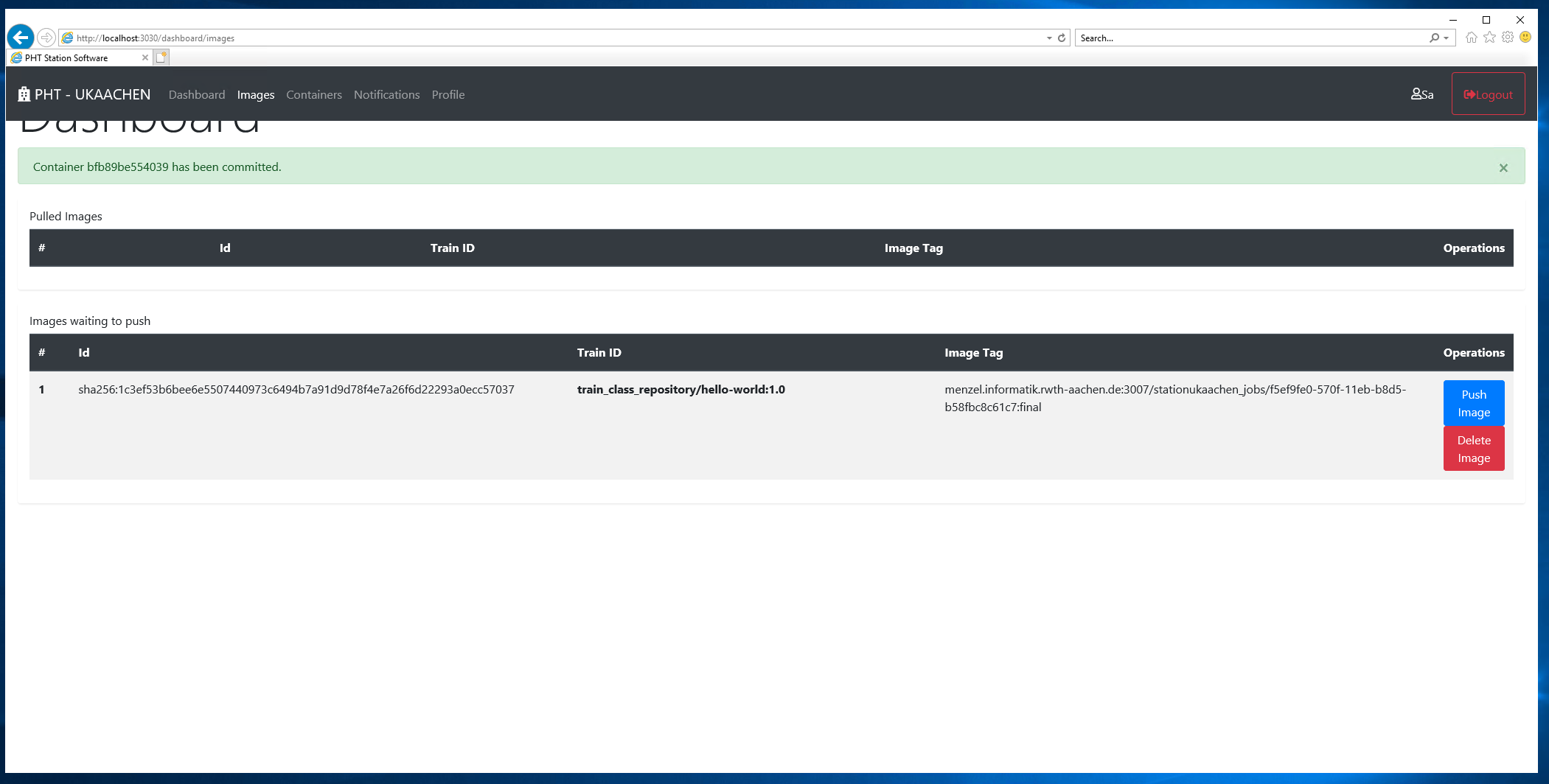


The container is currently running. Depending on the Train, the execution takes some time.

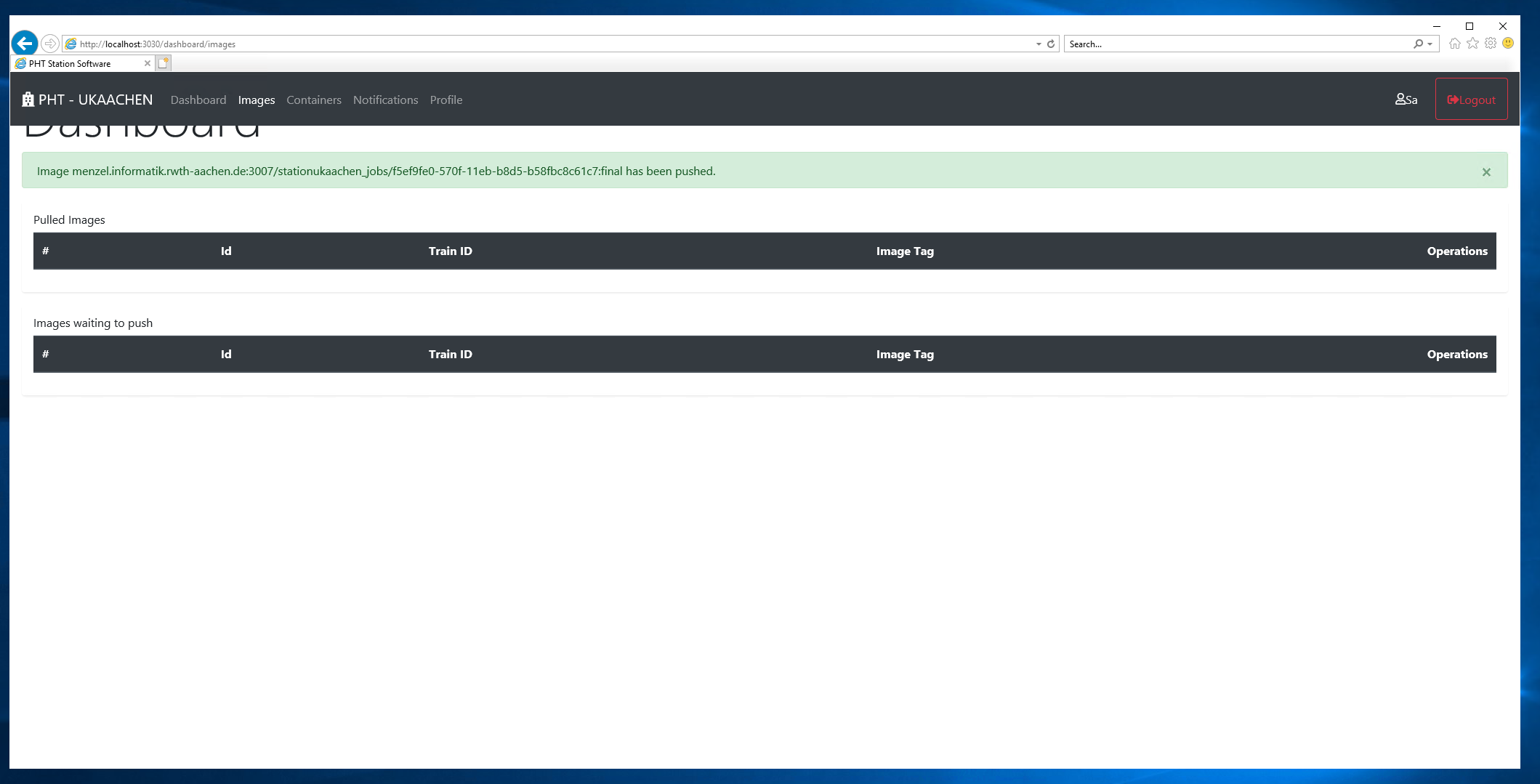
27) Refresh the page:



28) Click on the blue button „Commit“:



29) Click on the blue button „Push“:



After this step the deployment of the Station Software was successful. You are able to pull, execute, push a Train back to our central service component.

**Possible System Requirements:**

