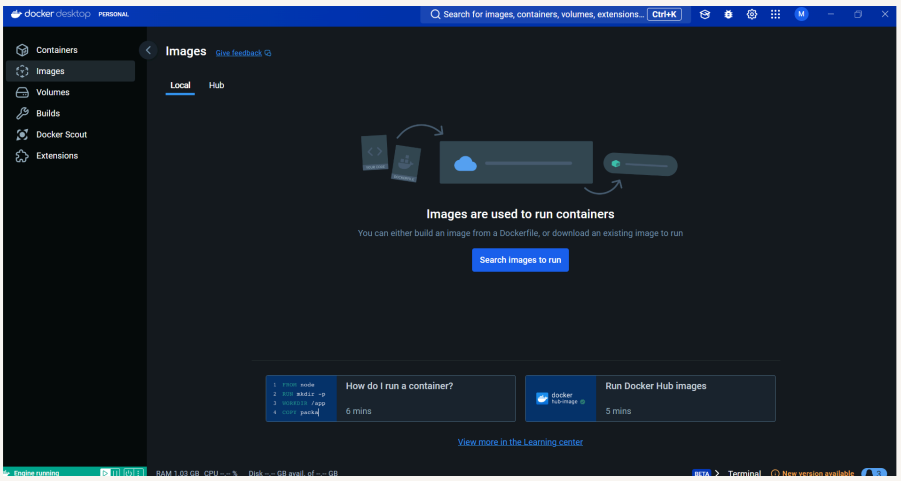
nextwork.org

Deploy an App with Docker

MA Marzena Pugo



MA
Marzena Pugo
NextWork Student

Introducing Today's Project!

What is Docker?

In this project, I used Docker to create containers based on containers images and set up my own container image.

One thing I didn't expect...

One thing I did not expect was seeing how quick is to deploy an application using Elastic Beanstalk.

This project took me...

This project took me 2.5 hours, including all the demo time.

MA
Marzena Pugo
NextWork Student

Understanding Containers and Docker

Containers

Containers are software packages that contain all the necessary components to run an application in any environment. They are useful because are used to deploy and run applications consistently across different environments,

A container image is a template / blueprint for creating containers. Containers spawned / created from the same image will behave in the same way which helps teams of developers have a unified experience when they're running the same application.

Docker

Docker is a containerization tool that allows users to build, share, and run applications. Docker containers host on a single physical server with a shared host OS. Docker Desktop is an application that allows users to build, share, and run container

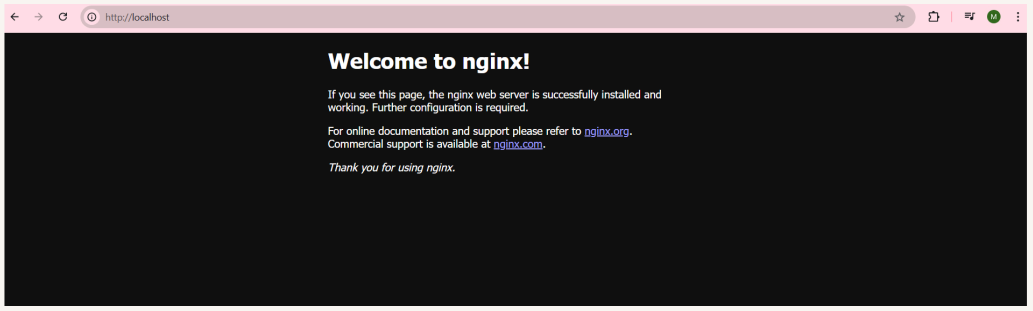
The Docker daemon is like an "engine" for Docker that recives commands we send trough clients e.g. client in the Docker Desktop or text commands sent in th terminal, and actually creates/ manages/ controls the containers.

MA
Marzena Pugo
NextWork Student

Running an Nginx Image

Nginx is a powerful web server/ software that helps with serving web content. Nginx is often refered to as a proxy server, which means it helps with distributing traffic to your application accross the insances running your appliction.

The command I ran to start a new container was docker run. I also set the flags -d -p 80:80 nginx, which means I'm running the container in the background (-d) and I'm matching port 80 in my host computer to the container's port 80 (-p 80:80).

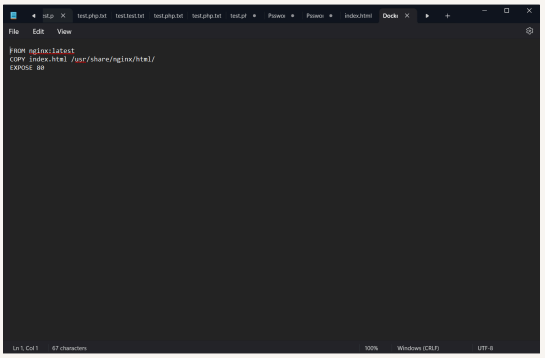


Creating a Custom Image

The Dockerfile is a set of instructions that tells Docker how to build your custom container image.

My Dockerfile tells Docker three things. First my custom container image uses the latest version of the Nginx container image as its base. Then I am modifying this base by replacing the default Nginx welcome page with own custom index.html.

The command I used to build a custom image with my Dockerfile was docker build. The '.' at the end of the command means that Docker can find the Dockerfile in the current directory i.e. the Compute folder on our desktop.



Running My Custom Image

There was an error when I ran my custom image because I tried to map my port 80 with the new container's port 80 but a running container was already using port 80. I resolved this by stopping the running container so that I can start a new one.

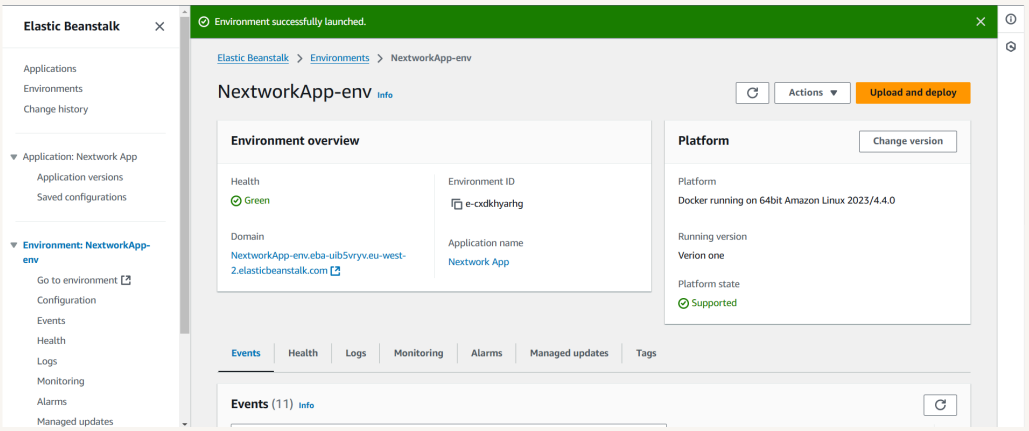
In this example, the container image is the template for creating a new container running an Nginx server that serves my custom index.html file. The container is the actual software that's running an Nginx web server with those cutomisations.

MA
Marzena Pugo
NextWork Student

Elastic Beanstalk

Elastic Beanstalk is an AWS service that helps deploying application to the cloud. It abstracts away a lot of the work with setting up cloud infrastructure when deploying applications.

Deploying my custom image with Elastic Beanstalk took me 10 minutes. This includes the time it took to launch the Elastic Beanstalk application.



The place to learn & showcase your skills

Check out nextwork.org for more projects