NextWork.org

Deploy Backend with Kubernetes

TA tahirgroot@gmail.com

TA
tahirgroot@gmail.com
NextWork Student
NextWork.org

Introducing Today's Project!

In this project, I will deploy the backend of an application with k8s this project will start from the scratch(i.e launching an ec2 instance) and go all the way to having a live backend deployed.

Tools and concepts

I used Kubernetes, Docker, Git, Github, kubectl, Amazon EKS Amazon ECR Amazon EC2, to deploy the backend of an application Key concepts include applying manifests files and bilding container images and pushing them to the container registry

Project reflection

N/a

Project Set Up

Kubernetes cluster

To set up today's project, I launched a K8 cluster. The cluster's role in this deployment is to be the control center for using k8's and orchestrating all the containers that will be running the application to deploy.

Backend code

I retrieved backend code by cloning my team member's repo in Github so we have a local copy of the work. Pulling code is essential to this deployment because I need to access it to build a container image of it. so that k8 will deploy.

Container image

Once I cloned the backend code, I built a container image because k8s needs my application to be package Without an image, it would be difficult for K8s to deploy application consistently across a cluster

I also pushed the container image to a container registry, which is the storage space for container image in the cloud ECR facilitates scaling for my deployment because k8s can easily use which ever image is tagged latest for each container it create

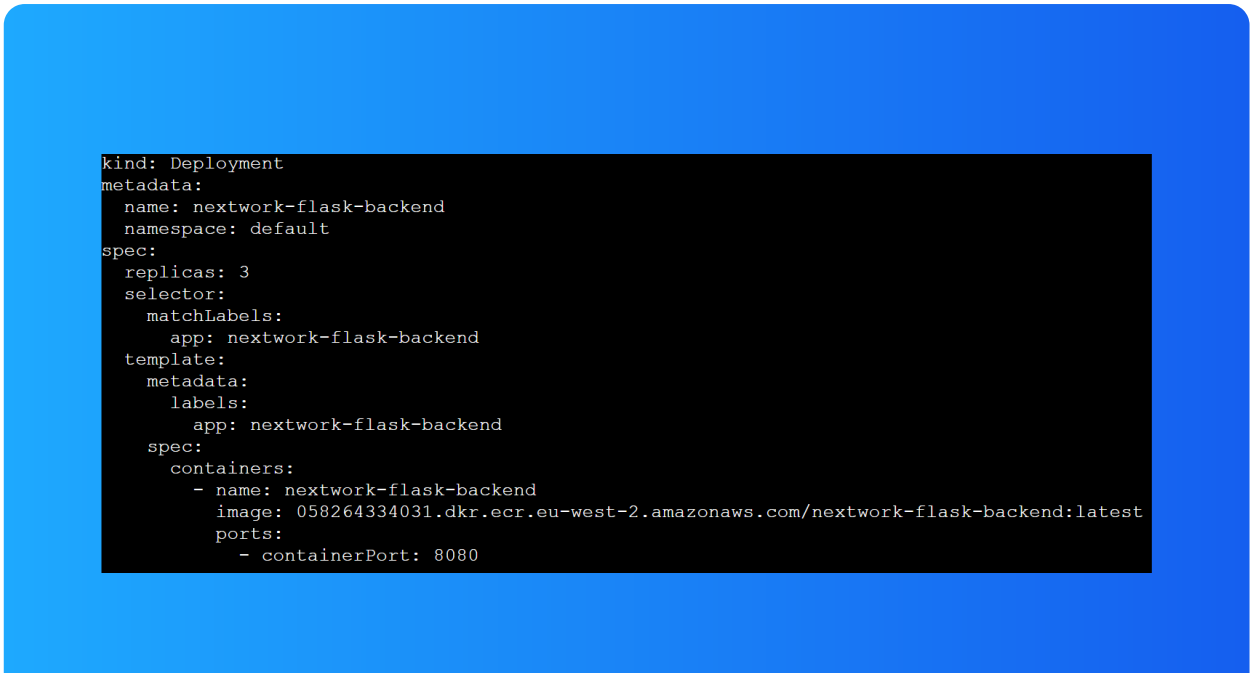
TA
tahirgroot@gmail.com
NextWork Student

Manifest files

K8 manifests are the instructions manuals on how i'd like k8 to deploy, expose and run my contarized applications. Manifests are helpful because they allow k8s to repeat a deployment using the same instructions.

A Deployment manifest manages the creation/setting up of a deployment resource that is respsonsible for creating containers and rolling out updates across EKS clsuter The container image URL in my Deployment manifest tells K8s where my image is

A Service resource exposes a deployed application for users or traffic My Service manifest sets up a service resource that exposes my backend application at port 8080 within the EKS cluster.



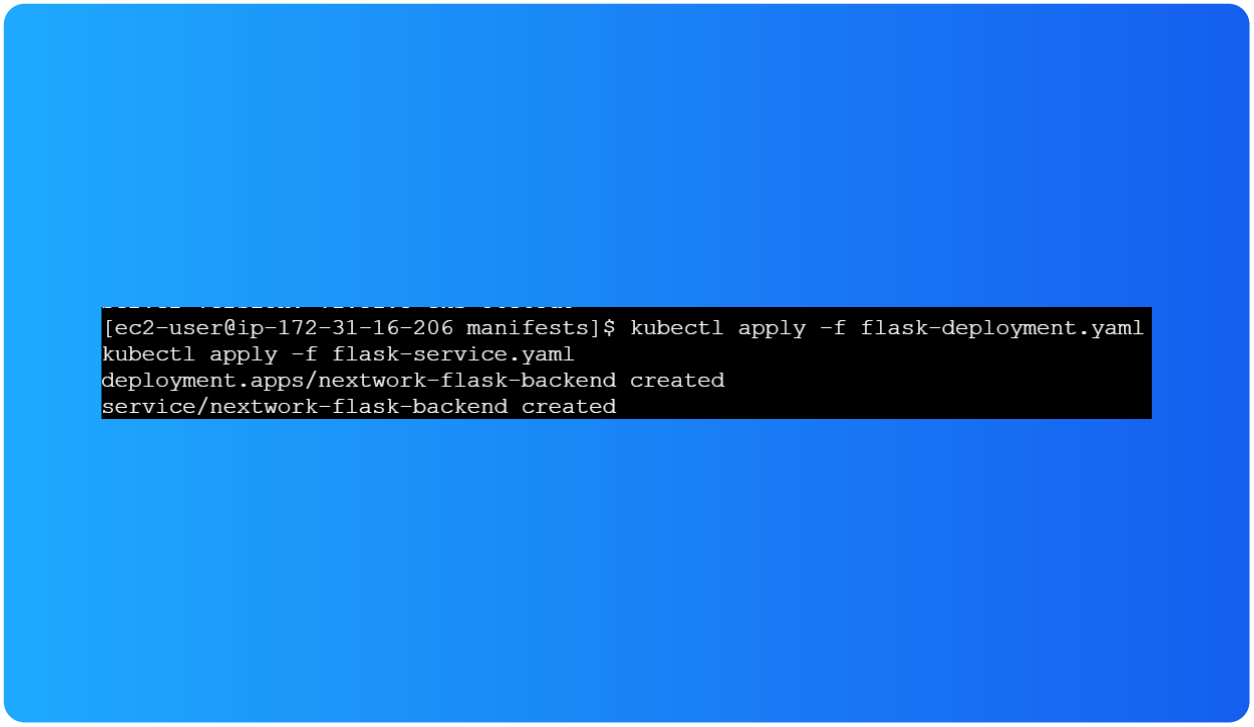
TA
tahirgroot@gmail.com
NextWork Student

Backend Deployment!

To deploy my backend application, I applied the manifest files that i created. Applying a manifest file i.e means giving k8s the templates for creating the deployment services/ resources to expose our application.

kubectl

kubectl is the command line tool for managing resources inside the cluster. I need this tool to deploy the application by creating the deployment resources I can't use eksctl for the job because that tool is for creating the cluster.



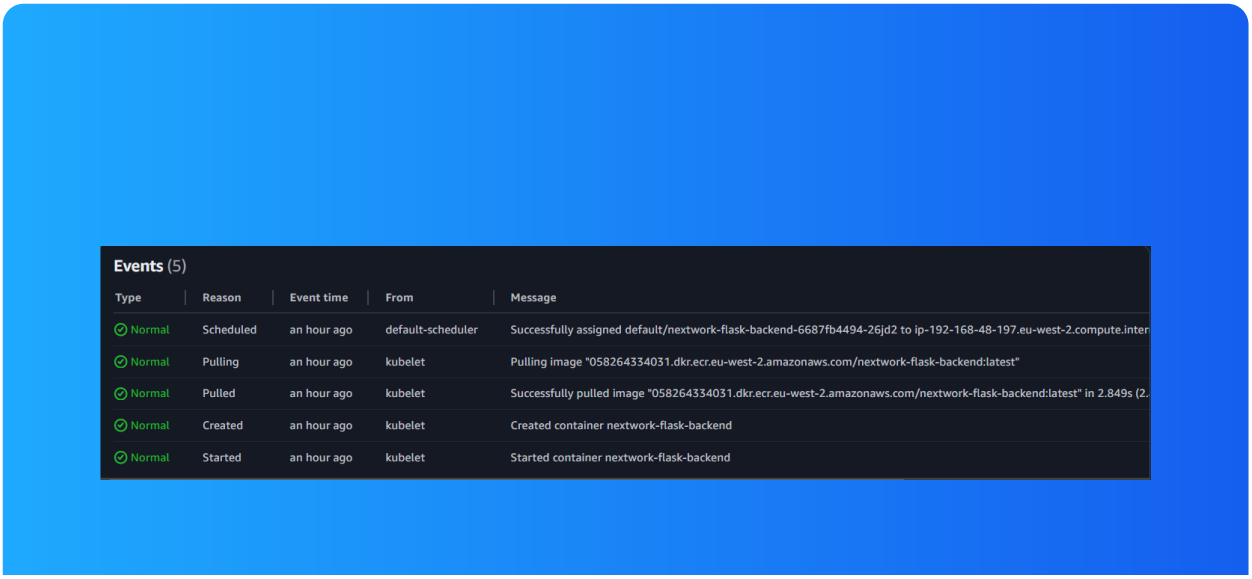
TA
tahirgroot@gmail.com
NextWork Student

Verifying Deployment

My extension for this project is to use the EKS console to verify my deployment. I had to set up IAM access policies because i dont have access to our clsuters nods by default. I setup access by running a create iamidentitymapping command

Once I gained access into my cluster's nodes, I discovered pods running inside each node. Pods are the smallest deployable unit and represent bundles of containers. Containers in a pod share networking and storage resources making the app efficently

The EKS console shows you the events for each pod, where I could see a container image getting pulled and used to create a new container running the backend application. This validated that the deployment was sucessful backend is running.



Everyone should be in a job they love.

Check out nextwork.org for more projects