Docker – Jenkins – K8s SkillCheck

This exercise is designed to check the knowledge and skills accumulated throughout the module. It will contain multiple processes and techniques from the lessons and is an open brief rather than a lab with steps.

The exercise has a set of requirements and starting brief that should be met, and this exercise should be done across the space of a day.

# Brief

You are hired as a DevOps engineer to deploy a web application for a customer called ‘QA Colours’, who have created an index.html and index.js and require it to be hosted online using containerisation and a CI CD server. They require the use of K8s as they believe the app will surge in demand and it requires scalability. Tech specifications are below:

* Nginx web server
* DockerFile containerisation
* GitHub Version Control
* Jenkins CI CD server
* Kubernetes Deployment

A template of the file structure and basic setup is hosted [here](https://github.com/Reece-elder/QAA-M4-SkillCheck) and should be used as a starting point for your skill check. You can either fork this repo and rename it, copy the files over to a repo of your choice or use this as inspiration for your own repo.

# Requirements

You must consider the following basic requirements when designing your deployment:

* Nginx web server should be proxy passing between the backend and the frontend load balancers
* Dockerfile for the backend which is running an express app that can be built and a Dockerfile for frontend which is our nginx and html, js to be built
* Proper Version control with GitHub using Branching and merge requests
* Jenkins CI CD Server using a pipeline to build and deploy the app (running a K8s manifest / script)
* Kubernetes deployment with 2x frontend pods and 2x backend pods running in a service allowing load balancing between the two
* Care and attention to not push anything confidential up to GitHub or Dockerhub

# Deliverable

You are expected to host your solution on a public GitHub repo under the name of QAA-M4-Skillcheck so we can easily find the appropriate repo. Your code must be merged to the main branch, and you should be using proper FBM (with the use of Dev and Feature branches) throughout.

There is a mark scheme associated with this Skill Check, you will be not getting back a grade, but there will be comments associated with different aspects of the skill check you have managed to do.

You should ensure your link to your repo is submitted to the relevant activity within bud by the deadline. You should make sure your repo is public so anyone who accesses the link via bud can see through the work you have done. You should expect feedback submitted via the same message board on Bud.