**Instructions**

The following files are contained in the application folder:

1. ChatServer: Creates a server and looks for connections. Asks for name of server when server is created. Responds with server name after connection is established.
2. ChatClient: Client connects to the server and waits for response.
3. Deployment.yml: It defines the configuration of deployment in the application. It defines the state of deployment and the replacas are to be created.
4. Service.yml: Defines configuration of service in Kubernetes.
5. Dockerfile: Gives the instructions to build docker image.

To execute the application the following instructions and commands should be followed using docker and minikube.

1. The first command to follow is **minikube start.** This command would create a local Kubernetes cluster.
2. To check the cluster created, run kubectl config get-clusters.

Docker side commands:

1. To create the image and run the dockerfile, we have to login to docker and build the image. The commands that should be run are docker login followed by **docker build -f Dockerfile . -t dadhija/final**. Here dadhija is my dockerhub username and final is my image file, latest is the version tag of the image.
2. To pull the image from dockerhub, the command is **docker pull dadhija/final:latest**.

Kubectl commands:

1. First we will apply the deployment and the service configuration to the kubectl cluster. This would involve commands **kubectl apply -f deployment.yml** and **kubectl apply -f service.yml**.
2. Then run the command **kubectl run week12 --image=dadhija/final** to make a pod.
3. Execute the command **kubectl get pods** to see that the image pods are running.

A computer screen shot of a program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

To check if everything is running properly:

* Kubectl get nodes
* Kubectl get deployment
* Kubectl get service
* Kubectl get pods (the pods should show running)

Running services:

1. Run the command **kubectl port-forward <deployment name running> 7080:32000**. 7080 is a port to forward to and 32000 is the server port. While the command is running, open another command line and run the java ChatClient localhost 7080 to connect the client and the server.

Client and server running:



