



**SOFE4630 Cloud Computing (Winter 2022 - Dr. M. El-darieby)**

**Lab 1: Project Milestone-- IaaS: Virtualization and  
Containerization**

**Final Submission**

**Date: 02/03/2022**

**Student Names:**

Alexander Pitcher (100693749)

Mitul Patel (100700131)

Clarissa Branje (100716458)

Amin Khakpour(100669547)

Activity Video 1 *HelloWorld Docker*:

[https://drive.google.com/file/d/1BhVnkhIRf5DoOe\\_i0NU-bfh2oRpNIPX-/view?usp=sharing](https://drive.google.com/file/d/1BhVnkhIRf5DoOe_i0NU-bfh2oRpNIPX-/view?usp=sharing)

Activity Video 2 *MultiContainer Docker*:

<https://drive.google.com/file/d/1cZZRIUFCR-UX0x2KcvVZJysIRnIOdXZ2/view>

Activity Video v3 *Deploying HTML File*:

[https://drive.google.com/file/d/1zi5bblhU5fqLiBNxcm3p3lr8\\_4oFo-l/view?usp=sharing](https://drive.google.com/file/d/1zi5bblhU5fqLiBNxcm3p3lr8_4oFo-l/view?usp=sharing)

Activity Video Final *GKE, Cluster, MyWebApp deployment*:

[https://drive.google.com/file/d/132OUEGLxkURml4TrB4UHfmDSh1\\_RiluT/view?usp=sharing](https://drive.google.com/file/d/132OUEGLxkURml4TrB4UHfmDSh1_RiluT/view?usp=sharing)

## Activity 1:

### 1. Answer the following questions

#### a. What are docker images, container, and registry?

- i. Docker images are files used to execute code within a container
- ii. A docker container is an executable package of software
- iii. A docker registry stores and delivers information held within a container

#### b. List the Docker commands used in the video with a brief description for each command and option.

- i. FROM: Choses the docker image to pull and what version of it to use
- ii. RUN: Creates a new directory for the application files
- iii. COPY: copies the application files from the host application
- iv. WORKDIR: Sets up directory for future commands
- v. CMD: Allows you to run a command you would in a terminal, like running the program

#### c. At the end of the video, there are two running containers, what commands can be used to stop and delete those two containers?

- i. To stop the two containers you can use `docker stop CONTAINER_ID`
- ii. To delete a container, you can use `docker rm CONTAINER_ID`

### Docker Command list

- *Docker images* => lists all the docker images
- *Docker build -t hello-world:1.0* => builds a docker image with the name hello-world and tag 1.0
- *Docker run hello-world:1.0* => runs the docker image on a container
- *Docker ps* => shows all running containers
- *Docker ps -a* => shows all running and non-running containers
- *Docker run -d hello-world:2.0* => runs the application in the background
- *Docker logs [id]* => shows the container output of based on the id
- *Docker stop [container id]* => stops the running container
- *Docker pause [id]* => pauses the container

### 2. Video Link: Activity 1

[https://drive.google.com/file/d/1BhVnkhIRf5DoOe\\_i0NU-bfh2oRpNIPX-/view?usp=sharing](https://drive.google.com/file/d/1BhVnkhIRf5DoOe_i0NU-bfh2oRpNIPX-/view?usp=sharing)

## Activity 2.

3.

### a. What's a multi-container Docker application?

■ A multi-container Docker application utilizes more than one container for an application to run. This could be adding a database or any other application extensions. With this being hard to keep track of and installing correctly, Docker ensures each container is performing only one thing correctly.

- An example would be MySQL would exist on one container
- Apache web server would exists on an another container

*Justification:*

*The main reason for deploying multiple containers for different services is due to scalability, deployment and modularity. This helps separate the concerns for an application to scale each component separately. This also provides more flexibility.*

### b. How are these containers communicated together?

- Each container communicate if they are part of the same network
- Docker creates a virtual network called a bridge
- Used to connect your container to other containers
- Ports can be exposed to the container and binded on runtime.
- In the network, each container is assigned an IP address.
- Containers communicate with the help of an IP address.

### c. What command can be used to stop the Docker application and delete its images?

■ `docker image rm -f (image id)` stop a running container and remove it with force

d. List the new docker commands used in the video with a brief description for each command and option

- i. ADD: This command allowed me to add the java file I am using to my docker server
- ii. EXPOSE: Allows you to expose a port to docker
- iii. CMD: Rather than allowing me to run the command in the terminal, I start up tomcat with it

## List of New Docker Command:

*//pulls mysql liberty from the docker.io library*

*Docker pull mysql*

*// Runs a Mysql container with the root password and db name along with the mysql take to use the library which was pulled earlier*

*Docker run --name app-db -d -e MYSQL\_ROOT\_PASSWORD=password -e MYSQL\_DATABASE=myDB mysql*

*//Runs the docker container with port 8080, with a -p option which means publish and binds port 8080 with the external port for 8080 and d flags allows for it to run in the background.*

*docker run --name app -d -p 8080:8080 my-web-app:1.0*

*//creates a network bridge which allows for containers to communicate with each other*

*Docker network create*

*//displays Network Id, name, driver and scope*

*Docker network ls*

*//Connects network to the app-db*

*docker network connect app-network app-db*

*//Runs the application container and publishes the port on 8080 and exposes the endpoint while connecting the web-app to the network.*

*Docker run --name app -d -p 8080:8080 --network my-web-app:1.0*

*//using docker compose, you can manage the build commands in a yml file called docker compose. The command bellow runs the docker-compose file*

*Docker-compose up - d*

## Link to Video 2 of webapp running on docker

<https://drive.google.com/file/d/1cZZRIUFCR-UX0x2KcvVZJysIRnIOdXZ2/view>

## Activity 3

## 5. Link to the video of the Google cloud running index.html for v3 Part 1:

[https://drive.google.com/file/d/1zi5bblhU5fqLiBNxcm3p3lr8\\_4oFo-I/view?usp=sharing](https://drive.google.com/file/d/1zi5bblhU5fqLiBNxcm3p3lr8_4oFo-I/view?usp=sharing)

## List of all used GCP commands and their description

## 6. Link to the video of deploying MyWebApp on Google Kubernetes

[https://drive.google.com/file/d/132OUEGLxkURml4TrB4UHfmDSh1\\_RiluT/view?usp=sharing](https://drive.google.com/file/d/132OUEGLxkURml4TrB4UHfmDSh1_RiluT/view?usp=sharing)

Commands	Explanation
<b>Part 1</b>	V3 index.html file
<code>docker run -p 8080:80 nginx:latest</code>	The following commands are explained in part 2 bellow
<code>docker cp index.html [container-id]:/usr/share/nginx/html/</code>	The following commands are explained in part 2 bellow
<code>docker commit [container-id] cad/web:version1</code>	The following commands are explained in part 2 bellow
<code>docker tag cad/web:version1 us.gcr.io/youtube-demo-255723/cad-site:version1</code>	The following commands are explained in part 2 bellow
<code>docker push us.gcr.io/youtube-demo-255723/cad-site:version1</code>	The following commands are explained in part 2 bellow
<b>Part 2</b>	Deploying mywebapp to google kubernetes using yml file
<code>docker build -t my-web-app:1.0</code>	Builds the docker image with the name my-web-app with version 1.0
<code>docker tag my-web-app:1.0 us.gcr.io/&lt;project_id&gt;/webapp:version1</code>	Tags the image with the name tag us.gcr.io/<project_id>/webapp:version 1. This tells google cloud about the deployment tag
<code>docker push my-web-app:1.0 us.gcr.io/&lt;project_id&gt;/webapp:version1</code>	Commands pushes the image app on to google container registry
<code>gcloud config set project &lt;project_id&gt;</code>	This sets the gcloud configuration to the project id and it is used to deploy container in GKE cluster

<i>gcloud config set compute/zone us-central1-a</i>	The following command is used to set the compute/zone on the GKE cluster
<i>gcloud container clusters create gk-cluster --num-nodes=1</i>	This command is used to create a GKE cluster with the number of nodes
<i>gcloud container clusters get-credentials gk-cluster</i>	This command is used to configure the kubectl to use the cluster I created
<i>kubectl apply -f webApp.yml</i>	This command applies the yml file on the application to build the containers
<i>kubectl create deployment web-server --image=us.gcr.io/&lt;project_id&gt;/cad-site:version1</i>	This command is used to deploy an application to the cluster
<i>kubectl expose deployment web-server --type LoadBalancer --port 80 --target-port 8080</i>	This command is used to expose the deployment cluster to a port 80 and a target port of 80
<i>kubectl get pods</i>	Lists the running pods to view the application
<i>kubectl get service</i>	Inspects the webapp to get the external ip address.

What is Kubernetes' pod, service, node, and deployment?

■ A pod is created and contains the application instance. Pod is an abstraction which represents a group of one or more containers. It has resources such as:

- Shared storage
- Networking with cluster Ip address
- Info on how different containers images versions and port run

■ A Node contains 1 or more pods. A node is a worker machine in kubernetes which may be a physical or virtual machine. Nodes can be managed by a control plane. A Node typically runs at least the following:

- Kubelet, which is a process that manages communication between the kubernetes control plane and Node.
- Docker or also a container runtime which is needed for pulling the container images from a registry.

■ Service: a logical abstraction for a cluster's deployed collection of pods (which all perform the same function). Because pods are ephemeral, a service assigns a name and a unique IP address to a set of pods that serve certain functions (web services, image processing, etc). (clusterIP).

■ A Kubernetes deployment is a process of deploying your containerized application which is typically run by a yml file to a cluster of nodes which contain their own pods.

What is meant by replicas?

- Replicas is essentially hosting web applications over several geographical locations and maintaining multiple data across many servers.
- This comes with its own challenges to maintain data consistency and data availability
- Kubernetes allows this by deploying the application on several nodes across different areas for replication.

What are the types of Kubernetes services? What is the purpose of each?

Kubernetes has several services such as the following;

- ClusterIP
- NodePort
- LoadBalancer
- ExternalName

ClusterIp:

- Responsible for assigning a cluster-internal IP address to ClusterIP service.
- This makes the services reachable within the cluster

NodePort:

- Responsible for exposing the service outside of the cluster through adding a cluster-wide port on top of ClusterIP

LoadBalancer

- Responsible for integrating nodeport with cloud-based load balancers

ExternalName

- It is a service which provides an external name map service to a DNS name.

The image below shows a Node overview.

The image below shows a Node overview.

## Node overview

