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

Project Milestone-- IaaS: Virtualization and Containerization

Group 5

Date: 02/03/2022
Student Names:
Alexander Pitcher (100693749)
Mitul Patel (100700131)
Clarissa Branje (100716458)
Amin Khakpour(100669547)

Individual Answers:

Clarissa Branje

1.
    a. What are docker images, container, and registry?
        - Docker image: A read-only template with instructions for creating a container for use on the Docker platform.
        - Container: Standard component that allows you to package your application and its dependencies in an easy to share way. They are portable, isolates, and lightweight. Great for all different types of applications (standalone, enterprise, database, languages, etc). Cheaper and has no requirement of a guest os like a VM.
        - Registry: A content delivery and storage system that holds named Docker images in various tagged versions.

    b. List the Docker commands used in the video with a brief description for each command and option.
        - FROM: Choses the docker image to pull and what version of it to use
        - COPY: copies from the host application
        - RUN: Creates a new directory
        - WORKDIR: Create a directory for future commands
        - CMD: Run a 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?
        - Stop: docker stop(container id)
        - Delete: docker rm(container id)
2. Video Link:

https://drive.google.com/file/d/1x3cUsgbHTZm2c0fHJuBvc8qjuLyTwXvc/view?usp=sharing

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.
    b. How are these containers communicated together?
        - Containers can communicate by networking with each other.
    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
        - ADD: Add the java file I am using to my docker server
        - EXPOSE: Expose a port to docker
        - CMD: Run with tomcat
4. Second video:
   https://drive.google.com/file/d/1G1rdB0pz2zFolwJ6K6MMx2bvAsWLM6Hs/view?usp=sharing
5. Third video:
   https://drive.google.com/file/d/1TSaKxru7Rj6tV79DhPrWxuPGVAVZj7oQ/view?usp=sharing
6. GCP shell commands
    a. docker run -d -p 8080:80 nginx:latest: Set the docker to run with port 8080
    b. docker cp index.html CONTAINER_ID:/usr/share/nginx/html/: Find and run the file (index.html)
7. Fourth video:
   Was not working with yml file so did not create video -> shows file attempt in fifth
8. Fifth Video:
   https://drive.google.com/file/d/1PyowvMPHKgxyTFvFr-2b9ld1_0W6D05S/view?usp=sharing
9.
    a. What is Kubernetes’ pod, service, node, and deployment?
        - Pod: Kubernetes' smallest and most basic deployable objects. In acluster, a Pod represents a single instance of a running process. One or more containers, such as Docker containers, are contained in pods. When many containers are running in a Pod, they are handled as a single entity that shares the Pod's resources.

    - 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).
    - Node: a Node is a worker machine that can be virtual or physical, depending on the cluster. The control plane is in charge of each Node. A Node can have numerous pods, and the Kubernetes control plane handles pod scheduling throughout the cluster's Nodes automatically.
    - Deployment: Used to tell Kubernetes how to create or modify an instance of a pod that contains a containerized application. Deployments allow you to increase the number of replica pods, allow controlled rollouts of updated code, and revert to previous deployment versions as needed.

b. What’s meant by replicas?

    - a process that runs many instances of a Pod while maintaining a consistent number of Pods. Its goal is to keep a certain number of Pod instances operating in a cluster at any given moment to prevent users from losing access to their applications if one of them dies or becomes unavailable.

c. What are the types of Kubernetes’ services? what is the purpose of each?

    - LoadBalancer: Send connections until the first server in the pool is full, then send new connections to the next available server. This algorithm is ideal when virtual machines are costly.
    - NodePort: An open port on each node of the cluster. Kubernetes transparently forwards incoming NodePort traffic to the service, even if the application is running on a different node.
    - ExternalName: A special case of a service that has no selector and uses the DNS name instead.
    - Headless Services: A service with a service IP, but instead of load balancing, it returns the IP of the associated pod. This allows you to interact directly with the pod without going through a proxy.