7.2P: Cloud-Native DevOps Project (Part 3 of 5)

Background

Our University Library is a cornerstone of academic resources, aims to enhance accessibility to educational materials through an advanced online platform. You has been asked to develop a cloud-native microservices architecture to support the library's diverse user base and streamline deployment processes. This project is divided into 5 parts as follows:

No.	Title	Task
1	Deploying Microservice with PostgreSQL on Render	Task 4.2P
2	Containerizing Microservices with Docker and Deploying to Local Kubernetes	Task 6.2P
3	Containerizing Microservices with Docker and Deploying to Azure Managed Kubernetes	Task 7.2P
4	Infrastructure as Code with Terraform	Task 8.2C
5	CI/CD with Github Actions using Terraform	Task 9.2D

By end of this project, you will gain a comprehensive understanding of essential DevOps practices and cloud-native application deployment techniques. More specifically, you will be able to do following:

1. Create Dockerfiles to containerize application and define the runtime environment.
2. Develop Kubernetes YAML files (deployment.yaml and service.yaml) to deploy and manage their microservice on Azure managed Kubernetes cluster.
3. Write Terraform scripts (main.tf, variables.tf, outputs.tf, provider.tf) to provision Azure infrastructure.
4. Deploy Azure Kubernetes Service (AKS) and integrate PostgreSQL for data storage.
5. Configure GitHub Actions workflows to automate the CI/CD pipeline.
6. Apply theoretical knowledge to real-world scenarios, enhancing their understanding of cloud computing and DevOps principles.

Tasks

In this task, you are continuing work from the previous project, focusing on Dockerizing the `book_catalog` and `inventory_management` microservices and deploying it to Azure Managed Kubernetes. By the end of this project, you will gain hands-on experience in setting up and managing containerized applications to **Azure Container Registry** and host them on **Azure Kubernetes Service**.

Steps

1. Downlaod code from the **Task Resources** and unzip.
2. Create a github repository named : `<your-name-sit722-part3>`.
3. Add given code to a github repository (local first then push to the remote repo).
4. Write Dockerfiles for both Microservices (`book_catalog` and `inventory_management`).
5. Write a Docker Compose file to orchestrate the microservices and PostgreSQL database (Use Azure PostgreSQL or Render PostgreSQL).
6. Upload docker images to the remote repository.
7. Create `deployment.yaml`.
8. Deploy Microservice to Azure Managed Kubernetes.

What will you submit?

You are asked to submit to OnTrack a single PDF document that contains (20 screenshots, URL for your Github Repo, 1 text answer) the following:

1. A screenshot of your **Resource Group** showing created resources *Container Registry* and *Kubernete Service*.
2. A screenshot of **both** dockerfiles.
3. A screenshot of `docker-compose.yml`.
4. A screenshot of the **console** with the output from the commands for each image:

   - `docker compose up ...`
   - `docker tag ...` **[Hint: You have to tag 2 different images.]**
   - `docker login ...`
   - `docker push  ...` **[Hint: You have to push 2 different images.]**

5. A screenshot of `az acr repository list --name <YOUR_REPOSITORY_URL> --output table`.

6. A screenshot of `deployment.yaml` file.
7. A screenshot of the **console** with the output from the commands:

   - `kubectl config current-context` (Output of this command will be **NAME_OF_AZURE_KUBERNETES_SERVICE** if it not selected as current-context then you have to change it)
   - `kubectl apply ...`
   - `kubectl get pods`
   - `kubectl get deployments`
   - `kubectl get services`

8. A screenshot of Deployed microservices (book_catalog) running via production URL (`<EXTERNAL_CLUSTER_IP>:<PORT>/docs`).
9. A screenshot of adding data of your favorite book. [Add **five** (5) books data but provide screenshot of one data].
10. A screenshot of getting all **five** (5) books data (`<EXTERNAL_CLUSTER_IP>:<PORT>/books/`).
11. A screenshot of updating data for any **one** (1) book.
12. A screenshot of the get using **id** of updated book.
13. A screenshot of deleting any **one** (1) book.
14. A screenshot of getting all books data. To verify deleted book data (`<EXTERNAL_CLUSTER_IP>:<PORT>/books/`).
15. A screenshot of Deployed microservices (inventory_management) running via production URL (`<EXTERNAL_CLUSTER_IP>:<PORT>/docs`).
16. A screenshot of adding data of inventory for any book. [Add **five** (5) data but provide screenshot of one data].
17. A screenshot of getting all **five** (5) inventories data (`<EXTERNAL_CLUSTER_IP>:<PORT>/inventories/`).
18. A screenshot of the **console** with the output from the commands:

    - `kubectl delete ...`

19. A screenshot of **Resource Group** from Azure Portal once all resources have been deleted.
20. A link to your public github repository.
21. Appendices - provide complete step-by-step instructions for how to deploy application on **Kubernetes**.

Note: Consider providing seperate screenshot for each question (where screenshot is required) and do not crop the same. Provide complete screen.

Make sure you take all screenshot correctly. Because you asked to delete all resources. If something is wrong then you have to perform all steps again.

Complete the Task Page Limit: No page limit for appendices, formatted reasonably, e.g., 2cm margins, 11 or 12 point font, appropriate headings/spacing, etc.