**Assignment 3**

Name:

Student Number:

**Submission Requirements:**

1. Create a Word File named with the following format: [***yourlastname\_Assignment#.docx***]
2. Include the questions you are answering in your submission file.
3. Include appropriate screenshots if applicable that are well formatted, cropped, labeled, and aligned.
4. Include your dockerfile for the application.
5. Include your .py file.

**Total 40 Marks**

**Note:** **Please read the rubric attached to the Lab Activity to comprehend the grading criteria**

**Scenario:**

**Note:** **Use personal accounts for these tasks**

**Note:** You are expected to implement your solution with considerations “Well Architected Framework” Principles (Operational Excellence, Security, Reliability, Performance Efficiency & Cost Optimization).

* Use **<yourlastname>** as a prefix for resource naming context.
* Regional deployment is your choice.
* Choose reasonable priced and sized resources.

**Note:** **DELETE ALL RESOURCES WHEN YOU HAVE COMPLETED YOUR TASKS!!!!!**

# **AWS:**

### **Microservice and Serverless**

**Tasks:**

* **Deploy a Python and Flask containerized web application using AWS ECS Fargate.**
  + Take Lab 4 code for your application and containerize it with Docker.
    - Modify the python app to say, **“Hello from *lastname* ECS Container.”**
    - Use GitHub to store your source code.
    - Use CodePipeline for CI/CD.
    - Use ECR to store your image.
    - Add container security scanning.
    - Use CodeBuild to build your container image.
    - Deploy custom VPC resources for your service.
    - Build auto-scaling and load balancing into your application deployment.
    - Deploy an ECS Cluster for your application.
    - Deploy ECS Service, Tasks and Task definition.
    - Ensure your application is running in private subnets.
* **Note:** Feel free to use IaC to deploy your infrastructure.

**Deliverables:**

**Note:** Accurate, high quality and well formatted screenshots required as proof/evidence. Please do not submit screen shots of your tasks process as the deliverables are only concerned with the results.

1. Dockerfile created and submitted **(1 Marks)**
2. Python file submitted (**1 Mark**)
3. Evidence of successful CodePipeline Execution (**4 Marks**)
   1. Should have Source, Approval, Build and Deploy Phases in your pipeline.
4. Evidence of buildspec.yaml file configuration (**3 Marks**)
5. Evidence of container scanning deployed. (**1 Mark**)
6. Evidence of HA & Secure Architecture deployed. (**5 Marks**)
   1. VPC, Subnets, ALB, Security & Target Groups
7. Evidence of ECS Fargate Cluster, Service and Task deployed and functional (**3 Marks**)
8. Evidence of Application running successfully via the load balancer (**2 Marks**)

# **GCP:**

### **Microservice and Serverless**

**Tasks:**

* **Take your app deployed on AWS ECS Fargate and deploy on GCP Cloud Run.**
  + You can use the same CI/CD pipeline to build your image in a public ECR for use on GCP or build and deploy with CloudBuild.
  + Deploy your application behind a public facing ALB.
  + Enable Auto-scaling and High Availability.
  + Ensure your application is running in private subnets.

**Deliverables:**

**Note:** Accurate, high quality and well formatted screenshots required as proof/evidence. Please do not submit screen shots of your tasks process as the deliverables are only concerned with the results.

1. Evidence of HA & secure Architecture deployed. (**5 Marks**)
   1. VPC, Subnets, ALB, Security & Targets etc.
2. Evidence of container services deployed and functional (**3 Marks**)
3. Evidence of Application running successfully via the load balancer (**2 Marks**)

# **Azure:**

### **Microservice and Serverless**

**Tasks:**

* **Take your app deployed on AWS ECS Fargate and deploy on Azure Container services.**
  + You can use the same public image in ECR or build and Deploy with Azure DevOps.
  + Deploy your application behind a public facing ALB or ALB like services.
  + Enable Auto-scaling and High Availability.
  + Ensure your application is running privately.

**Deliverables:**

**Note:** Accurate, high quality and well formatted screenshots required as proof/evidence. Please do not submit screen shots of your tasks process as the deliverables are only concerned with the results.

1. Evidence of HA & Secure Architecture deployed. (**5 Marks**)
   1. VNet, Subnets, ALB, Security & Targets
2. Evidence of container services deployed and functional (**3 Marks**)
3. Evidence of Application running successfully via the load balancer (**2 Marks**)

**Note:** **DELETE ALL RESOURCES WHEN YOU HAVE COMPLETED YOUR TASKS!!!!!**