SOFT8026 – Assignment 2

Due: 8pm, Friday 5th May, 2023
Worth: 50 marks

Overview

You must take your assignment 1 application and move it to a Kubernetes environment (e.g. using k3s). You must also implement testing and monitoring for the application. You must also discuss the use of a serverless function that could potentially be useful for your application.

Part 1 – Move to Kubernetes (20 marks)

Get your application running using Kubernetes. Lab 06 has a good example to work from – best approach is to create a Docker Hub repository for each service.

You must scale the pods (using deployments), but be careful not to overload your system. Exactly how many replicas you have isn't hugely important – what is more important is the ratio from one microservice to the next, e.g. do you have more web front ends or analysis microservices? You will need to explain why you scaled the way you did.

Include a discussion in your submission of approx. 500 words (450 to 550) on your scaling strategy. Why did you scale the way you did. Don't write in terms of absolute numbers of replicas, but rather the ratios / percentages. E.g. imagine you had 100 containers to distribute across your application, how would they be distributed from microservice to microservice?

Part 2 – Testing and Monitoring (24 marks)

You are to address the testing and monitoring of your application. You are to develop 1 functional test, 1 non-functional test, and 1 monitoring solution for your application.

1) Functional Test (8 marks)

Develop a test that tests whether your application is functioning correctly. You need only test 1 part of your application.

2) Non-functional Test (8 marks)

Develop a test that test how your application performs from 1 qualitative standpoint – e.g. responsiveness, security, load, failover, etc.

For each test developed:
Explain what type of test you have created (i.e. why it is functional or non-functional). Provide the following: - A paragraph on why you chose to implement the test - Screenshots or video of the test in action - Any supporting files, e.g. test scripts, along with instructions on how to run the test.

In the lectures, we looked at Gatling as a way to load test your application (non-functional). We also looked at Postman as a way of testing endpoints in your application (functional). You can investigate other tools if you wish.

2) Monitoring solution (8 marks) – develop using whatever tool(s) you choose a monitoring solution for your application. Explain what type of monitoring you are doing and why you have taken the approach you did. Provide the following: - A paragraph on why you chose to implement the monitor - Screenshots or video of the monitor in action - Any supporting files, e.g. config files, along with instructions on how to run the monitor.

You may not succeed in setting everything up, but document what you did get working with any relevant screenshots and some credit can be given for effort.

Part 3 – A serverless function (6 marks)

The concept of serverless functions has been covered in the lectures. Come up with a useful function for your application. Describe what it does and where it would fit into your architecture. You can also provide code / pseudocode to show what inputs it has, what processing takes place, and what outputs are generated – as well as where the input comes from and the output goes.

What to submit:

- The Word form containing all requested explanations (e.g. scaling strategy, why that test, why that monitor, the purpose of your serverless function, anything else you think is relevant to explain your efforts) and screenshots showing whatever you got up and running.
- A zip file containing your code and config files.

Notes:

1. You may have had issues getting the app from assignment 1 fully up and running. The functionality isn't hugely important in assignment 2. However, it would be good to have at least 3 or 4 microservices, so you can add any additional microservices of your choice so that you have enough when writing about your scaling strategy. Failing that, write about hypothetical microservices when discussing your scaling strategy.
2. A rubric will be found in Canvas under the assignment submission link. Consult it carefully as it will explain more about how the marks are allocated, allowing you to self-assess and therefore be strategic if you are finding certain parts difficult or are running out of time.