SOFT8026 – Assignment 2

Due: 23:59, Sunday 8th May, 2022
Worth: 50%

Overview

You must take your assignment 1 application and move it to a Kubernetes environment (e.g. using k3s) while making it more flexible to analyse multiple data sources. You must also implement testing and monitoring for the application. You must also create a serverless function that could potentially be useful for your application.

Part 1 – Move to Kubernetes (24%)

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.

In addition, you will need to access another dataset of your choice. This should be done in parallel to the other log file – i.e. don't just process the files one after the other in the same pod. Here you could choose to modify your existing data reading/extraction microservice to be more flexible (e.g. pass in the file as a parameter and the field(s) to look at) or have a microservice per file format. You can be a bit creative in coming up with an architecture – just explain your thinking in the Word form. If you want, you can change the statistics from assignment 1 to something else, e.g. something that applies to both datasets.

In the Word form, include screenshots of the application running with the web page showing statistics, etc.

Part 2 – Testing and Monitoring (18%)

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

1) Test (9%) – 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. Best marks will be provided where you can automate the tests (or at least explain how the tests could be automated).

In the lectures, we looked at Gatling as a way to load test your application. We also looked at Postman, but this would need an API in your application, which you may not have unless you are using REST (you might implement REST somewhere in your app for demo purposes). However, you may choose an alternative tool / approach.

2) Monitoring solution (9%) – 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. Best marks will be provided where you can visualize the monitoring in some way, e.g. a chart.

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 (8%)

Using Kubeless, create a function that would be useful for your application. It does not have to be called from within your application – it should run as per the quickstart guide from the command line. However, you must explain where in your application it would be placed and how it would be useful. Explain the inputs and outputs to/from your function.

See the following quick start guide: https://kubeless.io/docs/quick-start/

The hello-world function there is not much use. Come up with a useful function for your application. The function does not have to be embedded into your application – i.e. just run it from the command line as per the quick start guide. Include a screenshot of the function being run with its return value(s).

What to submit:

- The Word form containing all requested explanations (e.g. 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. For example, I am not too worried about grpc streaming, etc. in assignment 2. Just have 2 data sources feeding strings to an analysis service. Most marks in part 1 are for the set up and the explanations. If your web page wasn’t working, just create a functional web page that does something for you to test and monitor.
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.