SOFT8026 – Data-driven Microservices – Assignment 1

**Due:** Sunday 22nd March at 23:59

**Worth:** 50%

# *This is an initial draft that allows you to get started early if you wish. It may be modified in the coming days, but not substantially.*

# Overview

Thus far, we have covered the concept of microservices, bounded contexts and various architectural and integration concepts. We have also covered tools and technologies that can help us build and deploy microservices (see lab documents 1 to 4). This assignment requires that you analyse a requirement and devise an appropriate architecture and implementation.

# Problem Description

Your (fictional) employer is exploring Twitter analytics and is intrigued by the “microservices” way of architecting systems. They would like you to simulate, as part of a proof of concept, the streaming of tweets and their analysis in real time.

You should simulate a twitter stream by reading tweets from a data source (e.g. file or database) and serving them through a stream channel to an analytics client. The client should analyse each tweet as it arrives and calculate 3 different types of metric / result:

1. an aggregate metric, such as a total or average

2. a rolling 3-minute metric, e.g. sentiment within the last 3 minutes for a search term (which should be dynamic – passed in as a parameter)

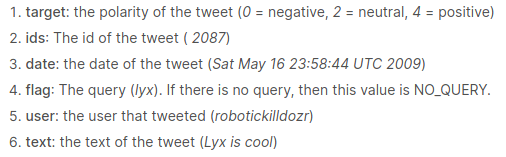
3. a single tweet which is the most or least of some criteria

There should be a single web page that displays those metrics / results whenever the page is loaded or refreshed. It is up to you how and where the web page gets the data.

It is recommended that on average 2 tweets per second should be streamed, though there should be some random variability.

# The Tweets

The file sentiment140.zip has 1.6 million tweets in CSV format. The file was downloaded from <https://www.kaggle.com/kazanova/sentiment140/data> and has the following comma separated fields:

If you decide to calculate the sentiment, then you should ignore the polarity value in the file and calculate your own using whatever python sentiment library you like.

# The Microservices

There are likely to be at least 3 microservices in your architecture, e.g. one that reads the tweets and streams them to a client connection, the client microservice that does the analytics, and a web server microservice that serves out that single web page.

# Technical Considerations

Use gRPC as the communication mechanism and use a stream channel to send the tweets to the client.

Flask is recommended for the web server and web page. A simple table is all that is required.

Because a sleep is placed between each tweet read, it should be possible to keep refreshing the web page (either automatically or manually) to see updated metrics / results.

Use Docker (with Dockerfiles) for each microservice and use Docker Compose to orchestrate your system. It should be possible to simply enter “docker-compose up” to bring your system up.

# What to Submit

* There will be a form to fill in (Word document) where you will explain your architecture and the metrics / results you chose to calculate. You will include sample output and complete a checklist to indicate which of the requirements you have completed. *This form will be uploaded in the next few days.*
* You will submit your project in zip format – exclude the dataset (at over 80MB), but indicate in the form where I should place the file in your directory structure.