[Fall-2021]

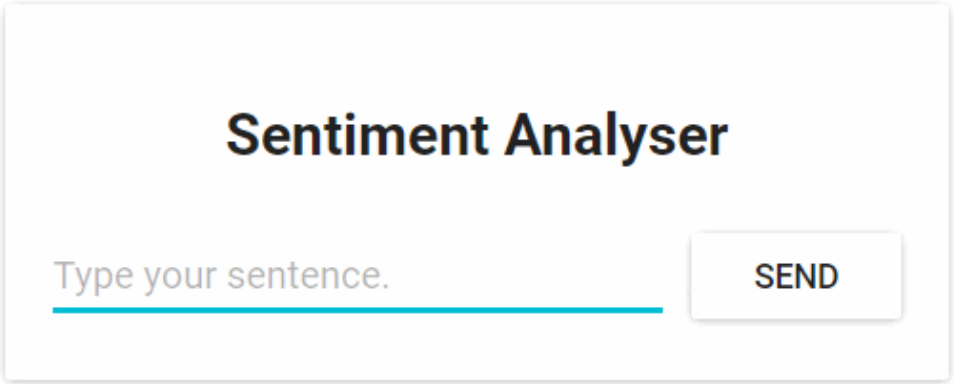
Extra-Credit Mini-Project-1

(10% bonus on Course Grade)

Deadline: October 14th 11:59PM EST (8:59PM PST)

General Description:

Orchestrate a microservice-based sentimental analysis app. This app takes a sentence as input and it uses text analysis to calculate the emotion of the sentence.



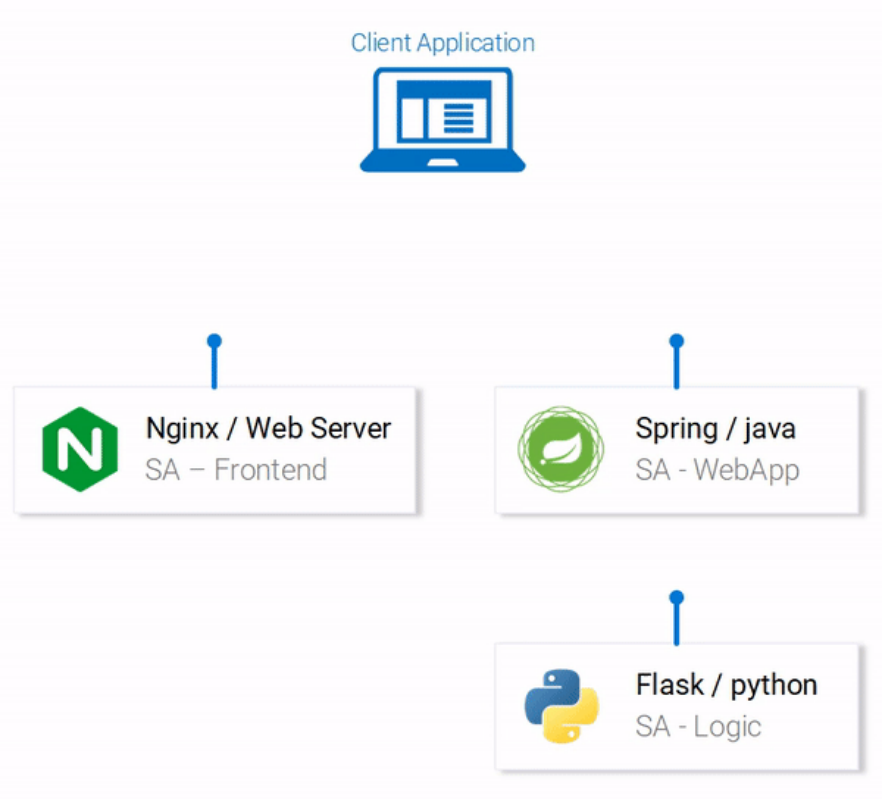
The application code is already developed and all the code can be cloned from the following GitHub repository: https://github.com/rinormaloku/k8s-mastery

This application consists of the following microservices:

- SA-Frontend: a Nginx web server that serves our ReactJS static files.
- SA-WebApp: a Java Web Application that handles requests from the frontend.
- SA-Logic: a python application that performs Sentiment Analysis.

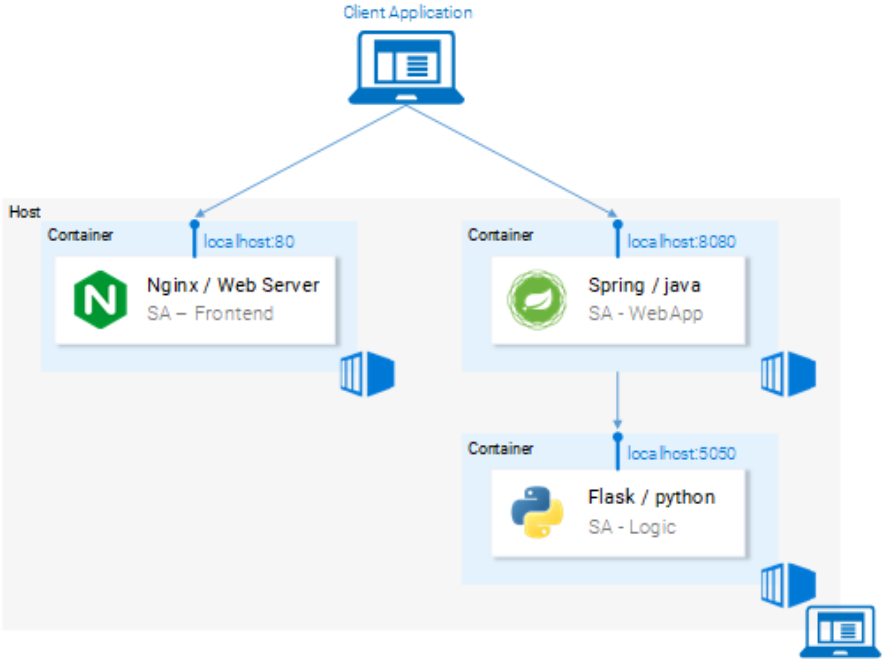
This interaction is best illustrated by showing how the data flows between them:

- A client application requests the index.html (which in turn requests bundled scripts of ReactJS application)
- The user interacting with the application triggers requests to the Spring WebApp.
- Spring WebApp forwards the requests for sentiment analysis to the Python app.
- Python Application calculates the sentiment and returns the result as a response.
- The Spring WebApp returns the response to the React app. (Which then represents the information to the user.)

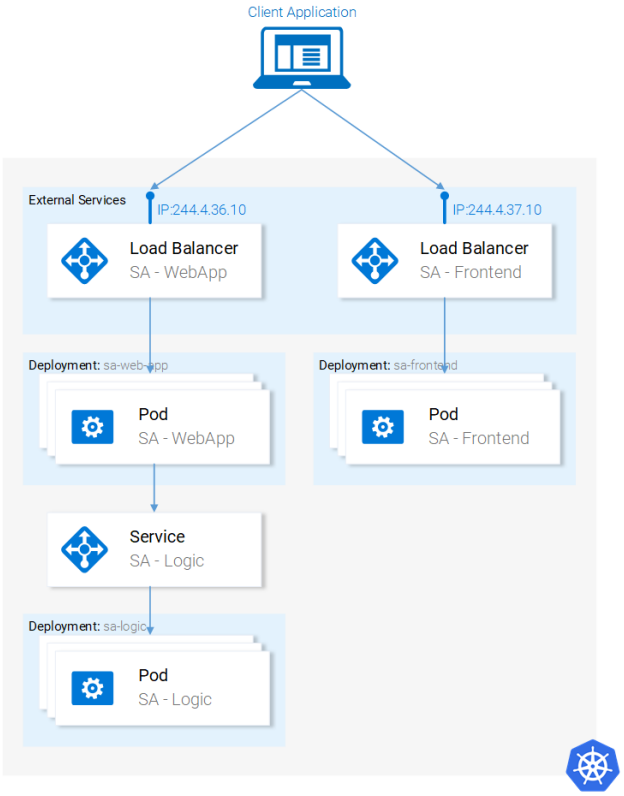


In order to complete this project, you need to complete the following steps:

Step-I: Build Container Image for Each Service and push them to Docker Hub.



Step-II: Orchestrate Sentiment Analyzer's containers and run them on GKE



Submission Guidelines:

- Post URL for your GitHub repository to Canvas. Make sure to keep your GitHub repository public.
- Create a folder in your GitHub repository and name it “Sentiment-Analysis”. Keep all related materials under this folder.
- Your repository should contain the following:
    1. (5 pts) Your ReadMe file should list all the steps to get the application to work on Google Kubernetes Engine.
    2. (5 pts) Screenshot for the execution of your application on Google Kubernetes Engine (i.e. screenshot showing the application running on the external IP through your Kubernetes Service)
    3. (5 pts) List of URLs for all of your Docker Hub images (Make sure they are publicly available).
    4. (5 pts) Video recording of code walkthrough and demo.

Don’t forget to disable billing after you complete the use of GCP

Common Penalties:

- Your GitHub repository is not public: 100% reduction (won’t be graded)
- Late submissions on Canvas or GitHub: 100% reduction (won't be graded)
- Docker Hub images are not public (25% reduction of maximum grade)