* How is your project architecture related to the theory taught in the lecture?
* Can you demonstrate, with some screen cap, how to increase capacity of your chat bot service?

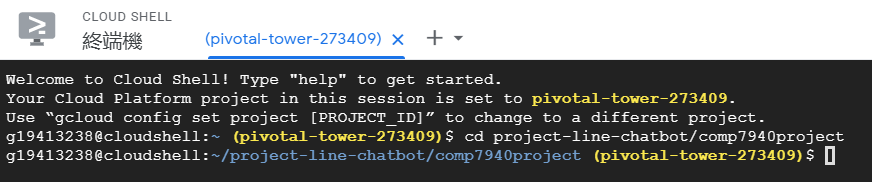
Here is how to increase the capacity of chatbots through Google App Engine.

1. Use the Cloud Shell to clone and browse to the "project" code.

Enter the following command in the Cloud Shell:

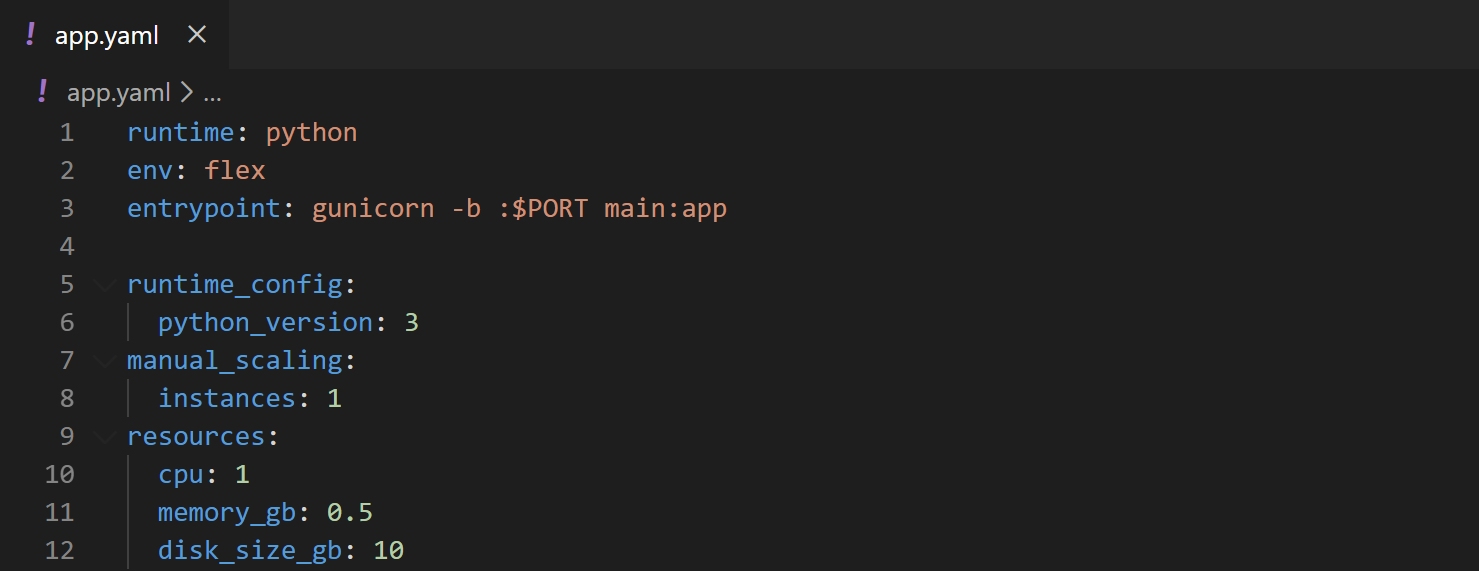
git clone <https://github.com/singsing215/project-line-chatbot.git>

cd project-line-chatbot/comp7940project



1. Create app.yaml file

Python applications in App Engine are configured using the app.yaml file. This file contains other general Settings such as CPU, memory, network and disk resources, expansion, and environment variables.

app.yaml code:

This file contains 1 CPU, 0.5GB of memory, and 10GB of disk resources.

1. Deploy to App Engine

Create an application:

gcloud app create

Deploy using the Cloud Shell:

gcloud app deploy app.yaml --project pivotal-tower-273409

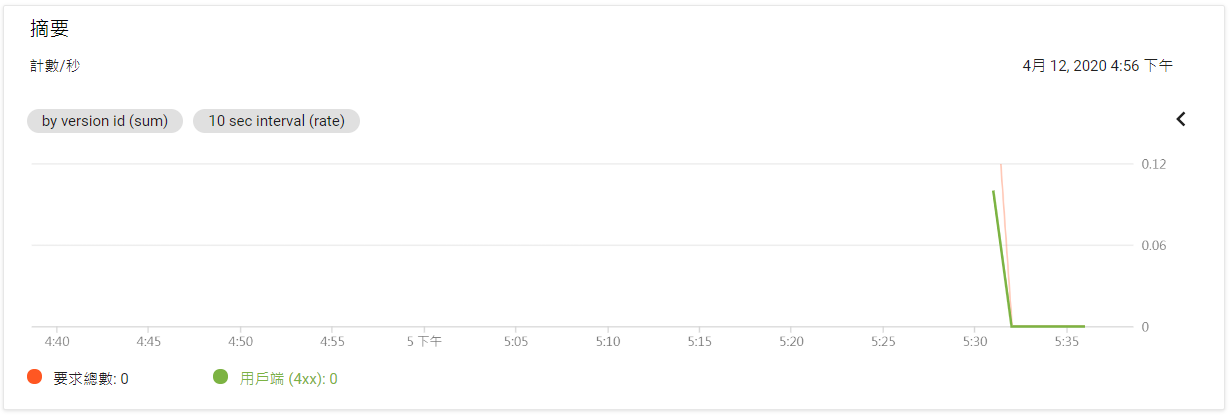
The application is deployed, and the default address is the subdomain on appspot.com, starting with my project ID: pivotal-tower-273409.appspot.com



1. View the application status

We can monitor the status of your application from the App Engine information home page:





All in all, we can deploy Google App Engine using the Cloud Shell to increase capacity of our chat bot service. If we want to adjust the CPU, memory, or disk of application, we can modify the parameters of app.yaml file.

* Can you identify if you bot is one of the examples of PaaS, IaaS, SaaS? Explain your answer.

Heroku is a PaaS platform, and LINE bot is implemented on the Heroku platform, so our bot is the examples of PaaS. Actually, our Heroku service also subscribed other service, including Redis, Tian API, Imgur API and YouTube API, which seems to make Heroku a SaaS. However, when it comes to the cloud delivery model used in our LINE chat bot, it should be PaaS.