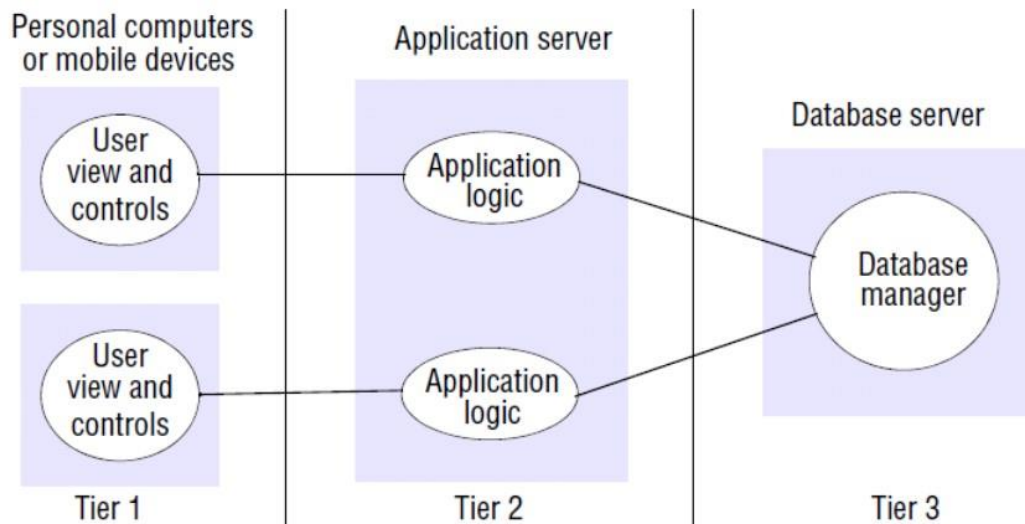# COMP7940 Cloud Computing
Group Project Milestone 4

15 April 2020

1. How is your project architecture related to the theory taught in the lecture?

The chat-bot can be viewed as three-tier architecture:

Firstly: Presentation logic: handling user interaction and updating the view of the application as presented to the user (LINE messenger: chat-bot);

Secondly: Application logic: concerned with the detailed application specific processing (Heroku server);

Thirdly: Data logic: concerned with the persistent storage of the application, such as a database management system (Heroku database, Heroku Redis, Redis database and Heroku PostgreSQL).

2. Can you demonstrate, with some screen cap, how to increase capacity of your chat bot service?

The LINE Bot is currently built on Heroku, which like other cloud providers is priced at "pay-as-you-go". Specifically, the resources of Heroku is measured in terms of dynos, which are Linux lightweight containers. Scaling up is achieved by purchasing dynos, if the free dynos associated with the account are not enough. For example, if there are more users subscribed to the LINE Bot account and talk to it, there will be more concurrent HTTP requests, and thereby generate greater traffic volume. In such a case, more web dynos should be purchased.

The LINE Bot currently has a few functions, including the query of a PostgreSQL database which stores location details of public hospitals in Hong Kong, and query of Redis database, which a web scraper will periodically update with the latest number of confirmed, discharged, hospitalized and death cases in Hong Kong.

There is also an external Redis database which stores predefined responses to address commonly encountered questions relating to COVID-19. So when users enter a set of our determined keywords, the LINE Bot will query the Redis database and provide an associated response.

The service for PostgreSQL is provided by Heroku PostgreSQL, which divides its services into various plans. Basically, fees will be charged if there are more connections (i.e. number of users of the LINE bot acquires) and more instances that the database will store. Heroku PostgreSQL does provide scale up options, one simply subscribes the different charging plans, based on anticipated usage. The following is a screen cap for different plans and the simplicity of the scaling by only clicking the right plan and install:

Hobby Dev	Free
Hobby Basic	$9/mo
Standard 0	$50/mo
Standard 2	$200/mo
Premium 0	$200/mo
Private 0	$300/mo
Premium 2	$350/mo
Shield 0	$350/mo
Standard 3	$400/mo
Private 2	$600/mo
Shield 2	$750/mo
Premium 3	$750/mo
Standard 4	$750/mo
Private 3	$1000/mo
Shield 3	$1200/mo
Premium 4	$1200/mo
Standard 5	$1400/mo
Private 4	$1500/mo
Shield 4	$1800/mo
Standard 6	$2000/mo

Postgres Extensions	
RAM	0 Bytes
Database Forks	
Direct SQL access	
Row Limit	10,000
Storage Capacity	
Database Followers	
Dataclips	
Continuous Protection	
Connection Limit	20
High Availability	
Rollback	
Rollback	0 Seconds
Encryption at Rest	
PostGIS	
PGBackups	
PGBackups Retained	2 Backups
Heroku Private Spaces	
Burstable Performance	
Heroku Postgres via PrivateLink	

Install Heroku Postgres

QUICK LINKS
Add-on Details
Region Availability
Plans & Pricing
Documentation

SHAREABLE DETAILS
Shareable Across Apps
Multiple Installs/App

ADD-ON CATEGORY
Data Stores

LANGUAGE SUPPORT
CLOJURE GO JAVA NODE
PHP PYTHON RUBY SCALA

In anticipating the growth in data storage given more and more countries are affected

by the pandemic and people may be interested in not only the data in Hong Kong, but also that in overseas, the use of other database service is an effective way of scaling-out. And this is done by way of installing the add-on Heroku Redis.

Similar to PostgreSQL, the service for Redis is provided by Heroku Reis, which also divides its services into various plans. Basically, fees will be charged if there are more connections (i.e. number of users of the LINE bot acquires) and the memory consumed. It can be equally easy to scale up by subscribing charged plans, based on anticipated usage. The following is a screen cap for different plans and the simplicity of the scaling by only clicking the right plan and install:

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

The LINE bot is purely implemented on Heroku platform. Although Heroku defines itself to be a PaaS platform, other services are able to be subscribed through Heroku, including Heroku Redis and PostgreSQL. Also, the LINE bot also relies on the the LINE Messaging API. So basically, our team's control over the LINE Bot is only the selection of add-on services and the provision of source code. Therefore, to a large extent, our LINE bot is inclined from PaaS towards SaaS.

Group project Group Members:

Lo Chi Leung (SID: 19451415) (GitHub id: HKBU-DavidLo)
Ye Wei Xiong (SID: 19451407) (GitHub id: handsome168-web)
Zhang Kai (SID: 19404549) (GitHub id: ZHANGSHAO1212)