Milestone 4

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

This project uses a distributed computing platform to realize the function of the chatbot. Start by breaking the system into smaller services, each member responsible for maintaining one service. Different services can be developed, tested, and not coupled with other functionality, which communicate with each other by calling each other.

The architecture of a system is a structure represented by individual components and their relationships. Web services are closely related to objects and components. Web services still based on encapsulation of behavior and accessed through interfaces. In terms of communication, Remote method invocation (RMI) is used. A web server is often a client of file servers.

The technology that comes with distributed systems is distributed caching. This project introduces the Redis cache system to share a cache and set up the web service. For the latest news retrieval function, the latest news data is cached through Redis. Redis can store a small amount of data and have high-speed read and write access. For a large number of data storage, distributed system support, data consistency can be guaranteed, and easy to add and delete cluster nodes. Finally, get the latest news interface through line.

We have used some important fundamental models like interaction models. For example, users can switch languages as they like.

Heroku is a PaaS that has established a pre-defined ready-to-use environment with already deployed and configured IT resources. We build our own chatbot service on Heroku, so conceptually, Heroku is a cloud service offered by its owner. And we are cloud consumer using this service. Our chatbot is the cloud service consumer. By accessing Heroku’s ready-made PaaS environment, we can be set free from the set-up process and maintenance burden of IT resources. Through this practical training, our concept of cloud service is clearer.

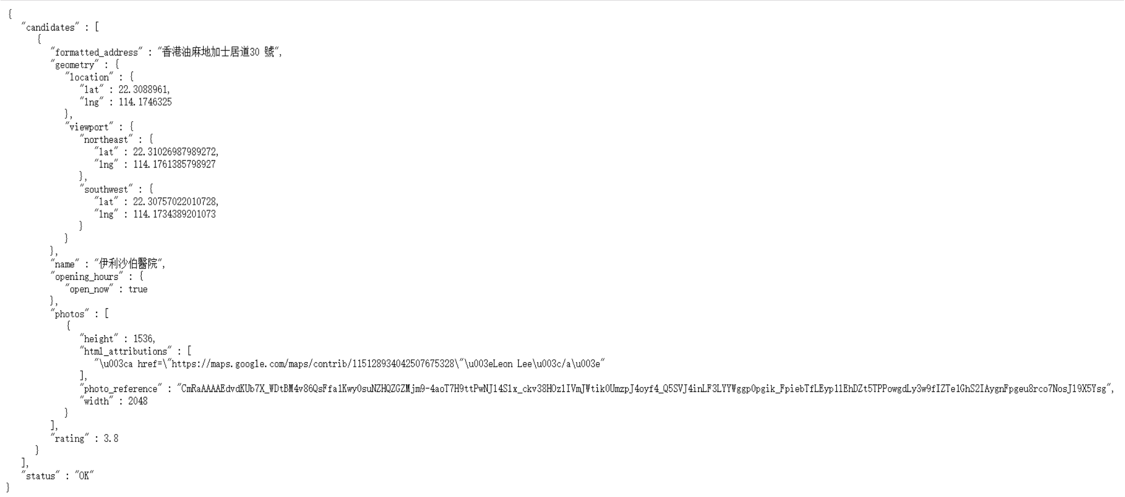
Meanwhile, through the knowledge of service management in the classroom, we have engaged in other services, like Baidu Translate API and Google Map API. By registering the developer, obtaining the key and opening the service, we can use the service. At the same time, you can monitor the memory usage and other conditions in the management data table or upgrade the service to enjoy higher-scale capabilities.

For the Google map API, it’s related to the theory: RESTful API taught in the lecture. REST is an acronym for Representational State Transfer. An API that satisfy REST specification is called a RESTful API. It may work with different platforms, different languages, different data format (XML, JSON). To use the Google map API, you first get an API key, then you can use this key by all Google map API, like Distance Matrix API, Directions API, Geocoding API, Geolocation API, Maps Elevation API, Maps Embed API, Maps JavaScript API, Maps Static API, Places API, Roads API, Street View Static API, Time Zone API, etc. You can use these API by URL, like:

"https://maps.googleapis.com/maps/api/place/findplacefromtext/json?input=Hospital&inputtype=textquery&fields=photos,formatted_address,name,rating,opening_hours,geometry&key=YOUR_KEY".

This is an URL will return the nearest hospital information based on the IP address. It will return a json include formatted_address, name, rating, opening_hours, geometry,etc.

The result:

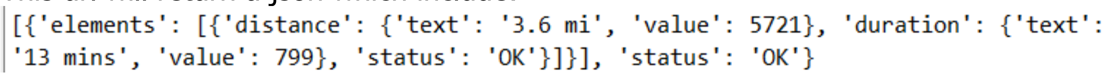


```
{
   "candidates" : [
      {
         "formatted_address" : "香港油麻地加士居道30 號",
         "geometry" : {
            "location" : {
               "lat" : 22.3088961,
               "lng" : 114.1746325
            },
            "viewport" : {
               "northeast" : {
                  "lat" : 22.31026987989272,
                  "lng" : 114.1761385798927
               },
               "southwest" : {
                  "lat" : 22.30757022010728,
                  "lng" : 114.1734389201073
               }
            }
         },
         "name" : "伊利沙伯醫院",
         "opening_hours" : {
            "open_now" : true
         },
         "photos" : [
            {
               "height" : 1536,
               "html_attributions" : [
                  "\u003ca href=\"https://maps.google.com/maps/contrib/115128934042507675328\"\u003eLeon Lee\u003c/a\u003e"
               ],
               "photo_reference" : "CmRaAAAAEdvdKUb7X_WDtBM4v86QsFfa1Kwy0suNZHQZGZMjm9-4aoT7H9ttPwNJ14S1x_ckv38HOz1IVmJWtik0UmzpJ4oyf4_Q5SVJ4inLF3LYYWggp0pgik_FpiebTfLEyp11EhDZt5TPPowgdLy3w9fIZTe1GhS2IAygnFpgeu8rco7NosJ19X5Ysg",
               "width" : 2048
            }
         ],
         "rating" : 3.8
      }
   ],
   "status" : "OK"
}
```

We can get it’s geometry by program, then we can use these data to do what we want.

In our project, we use Google map API to get duration and distance. If users send a localmessage, the chatbot will get the latitude and longitude from the localmessage and get duration and distance by url:

https://maps.googleapis.com/maps/api/distancematrix/json?units=imperial&origins="+latdata+","+lngdata+"&destinations="+hospital_location+"&key=" + key.
This url will return a json which include:

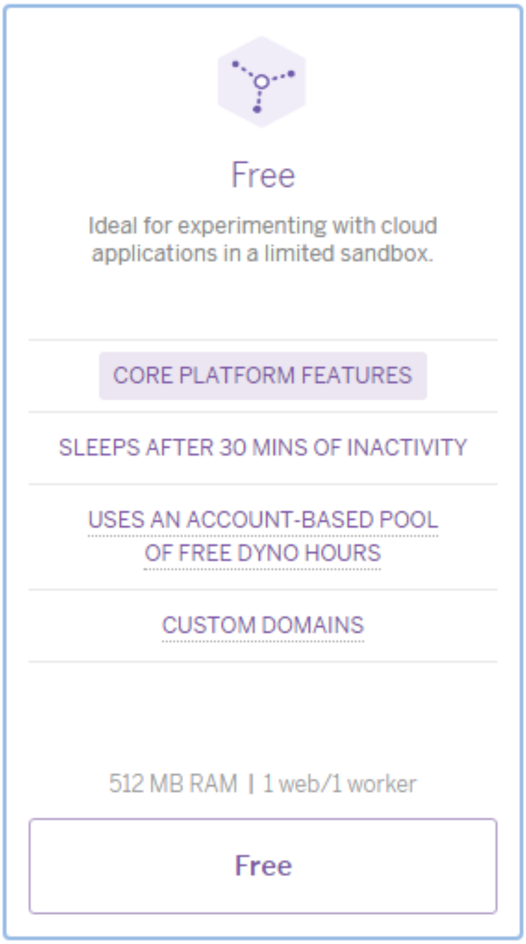


```
[{'elements': [{'distance': {'text': '3.6 mi', 'value': 5721}, 'duration': {'text':
'13 mins', 'value': 799}, 'status': 'OK'}]}], 'status': 'OK'}
```

So, we get the data we wanted. After that we just need to edit the message and send it to the user.

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

Now, we are using a free dyno which is ideal for experiment with limited features. And free dynos will sleep after a half hour of inactivity, which would cause a delay of a few seconds for the first request upon waking. The RAM is 512 MB and can only 1 web or 1 work project.



Free

Ideal for experimenting with cloud applications in a limited sandbox.

CORE PLATFORM FEATURES

SLEEPS AFTER 30 MINS OF INACTIVITY

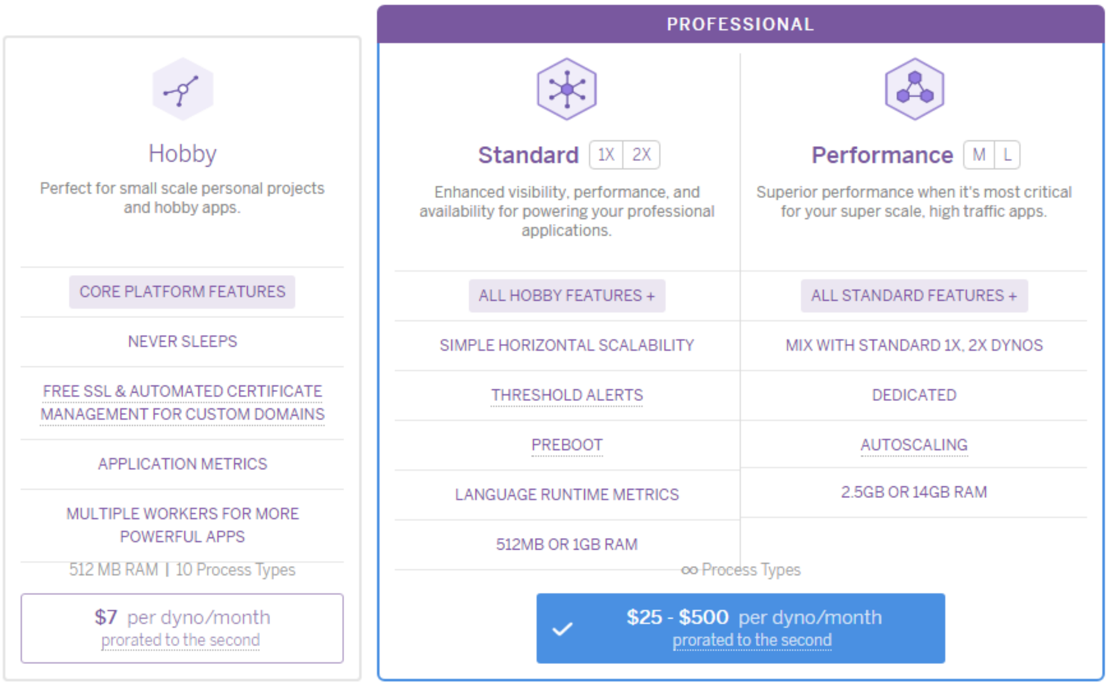
USES AN ACCOUNT-BASED POOL OF FREE DYNO HOURS

CUSTOM DOMAINS

512 MB RAM | 1 web/1 worker

Free

To avoid above situation, we can upgrade dyno to hobby, standard or professional dyno type. Thus, we can have scaled RAM and process type. Also, improve service sleep.



Hobby

Perfect for small scale personal projects and hobby apps.

CORE PLATFORM FEATURES

NEVER SLEEPS

FREE SSL & AUTOMATED CERTIFICATE MANAGEMENT FOR CUSTOM DOMAINS

APPLICATION METRICS

MULTIPLE WORKERS FOR MORE POWERFUL APPS

512 MB RAM | 10 Process Types

$7 per dyno/month
prorated to the second

PROFESSIONAL

Standard 1X 2X

Enhanced visibility, performance, and availability for powering your professional applications.

ALL HOBBY FEATURES +

SIMPLE HORIZONTAL SCALABILITY

THRESHOLD ALERTS

PREBOOT

LANGUAGE RUNTIME METRICS

512MB OR 1GB RAM

Performance M L

Superior performance when it's most critical for your super scale, high traffic apps.

ALL STANDARD FEATURES +

MIX WITH STANDARD 1X, 2X DYNOS

DEDICATED

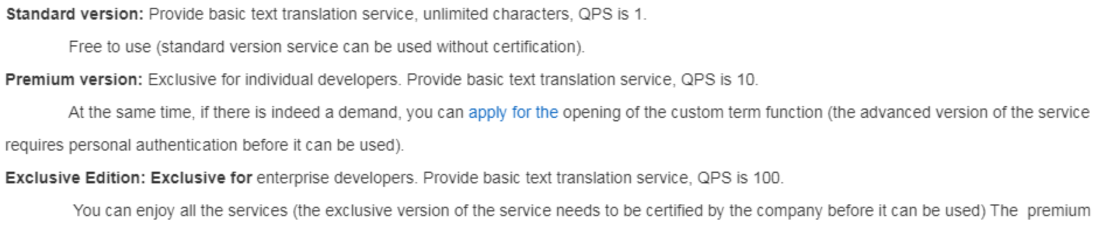
AUTOSCALING

2.5GB OR 14GB RAM

∞ Process Types

✓ $25 - $500 per dyno/month
prorated to the second

For Baidu Translate API, the currently version that we use is the Premium version. The QPS is 10. And we can upgrade it to Exclusive Edition, whose QPS is 100.



Standard version: Provide basic text translation service, unlimited characters, QPS is 1.

Free to use (standard version service can be used without certification).

Premium version: Exclusive for individual developers. Provide basic text translation service, QPS is 10.

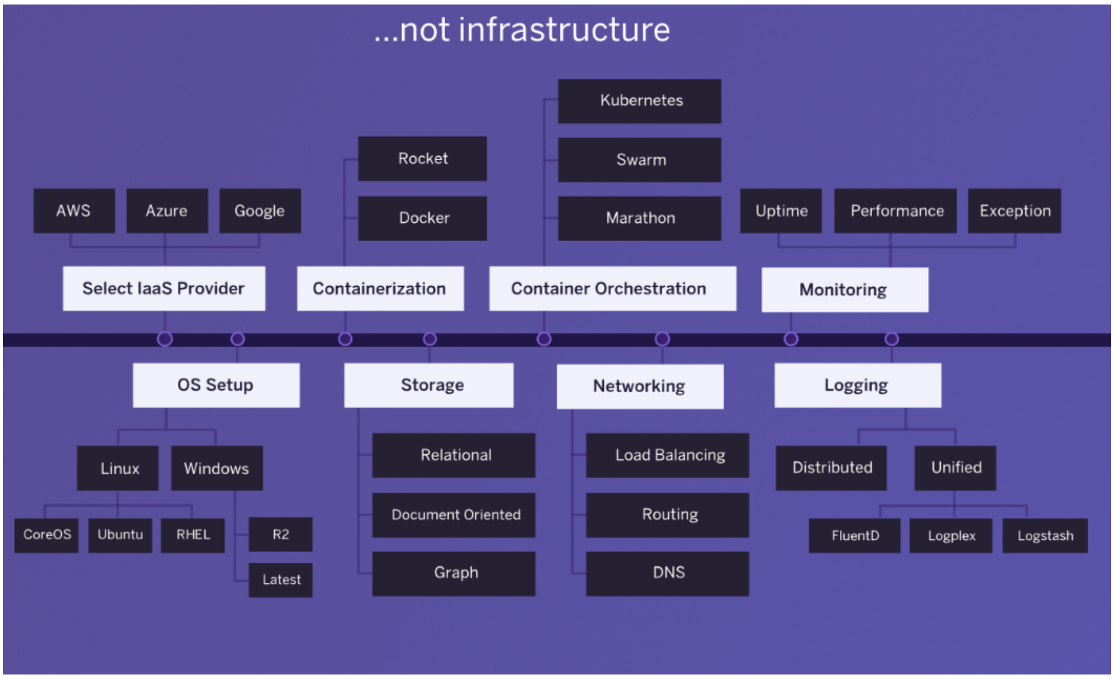
At the same time, if there is indeed a demand, you can apply for the opening of the custom term function (the advanced version of the service requires personal authentication before it can be used).

Exclusive Edition: Exclusive for enterprise developers. Provide basic text translation service, QPS is 100.

You can enjoy all the services (the exclusive version of the service needs to be certified by the company before it can be used) The premium

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

Our chatbot is an example of SaaS. Because applications on remote servers can be run over a network and the services consumed are accessed through web browsing. Our robot is deployed on Heroku cloud platform. This is a flow chart provided by Heroku's official website, which demonstrates the application building channel it advocates. Therefore, individual developers, entrepreneurial teams, and even businesses of all sizes can use Heroku in their own way to develop excellent applications.



...not infrastructure

So Heroku is a PaaS that has established a pre-defined ready-to-use environment with already deployed and configured IT resources. We build our own chatbot service on Heroku and offer this service to users. Thus, our chatbot is an example of SaaS.