

## Milestone 4

\* Submit:

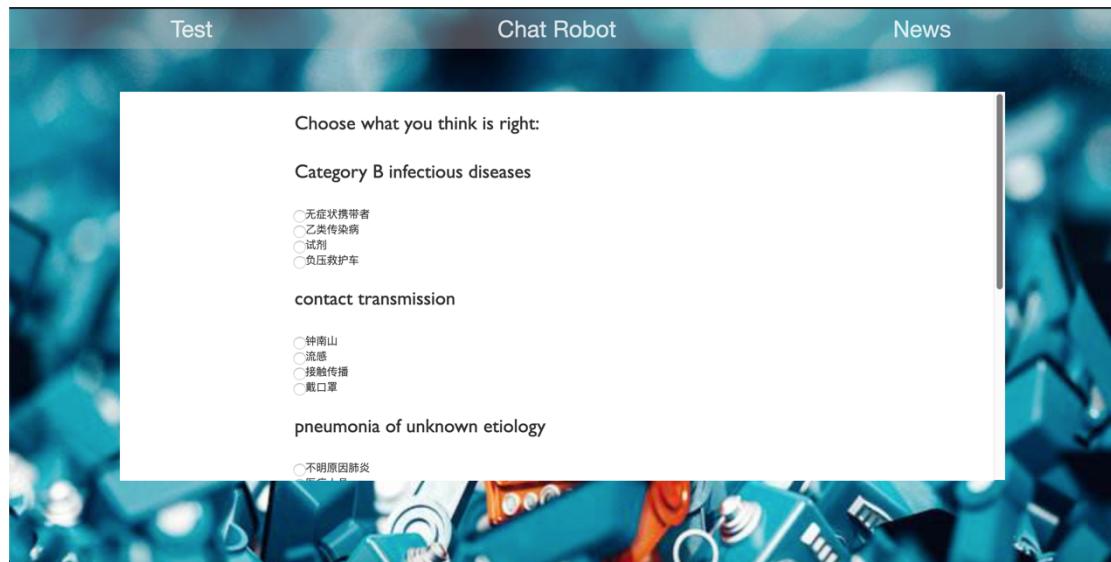
\* Update all your code on Github.

URL: <https://github.com/529352752/COMP7940-Project-Group5>

\* Deploy your code on cloud and run it.

There are three functions for our project.

The first of it is test user vocabularies relate to coronavirus-19



You may choose all answer which think correct.

### virus variation

- 紧平衡
- 诊断、治疗、追踪和筛查
- 自我隔离
- 病毒变异

If all of your answer are correct, it will display:

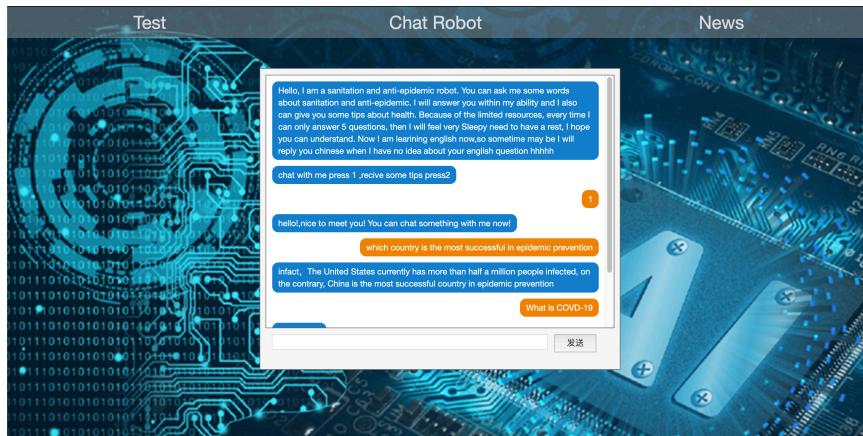
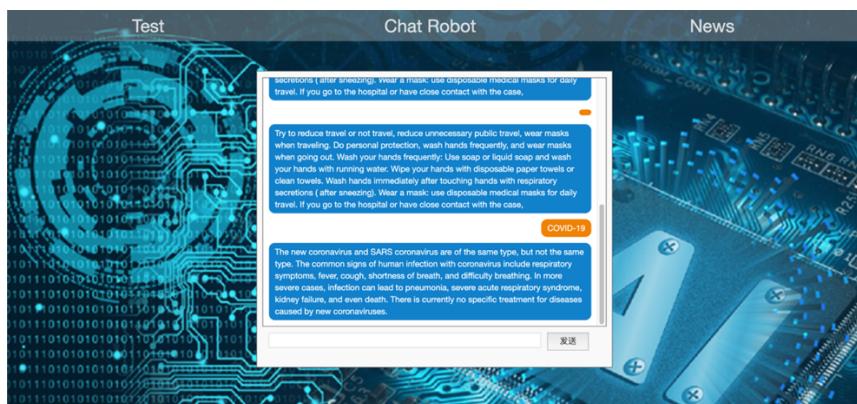
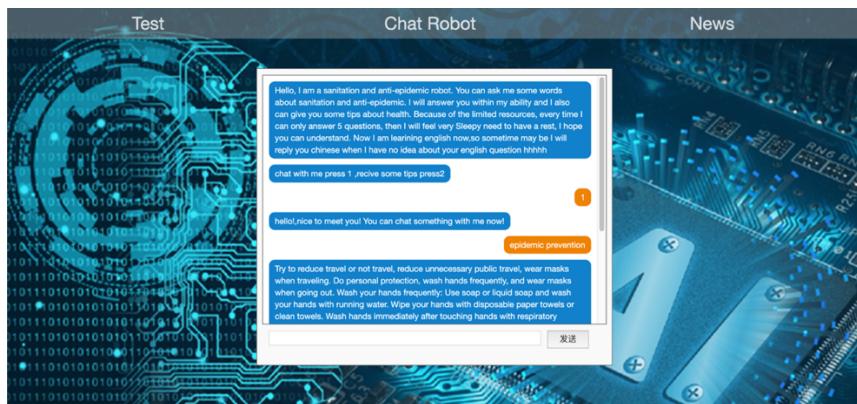
**Congrats! All correct!**

If some of your answer are not correct, it will remind which option were wrong and try again.

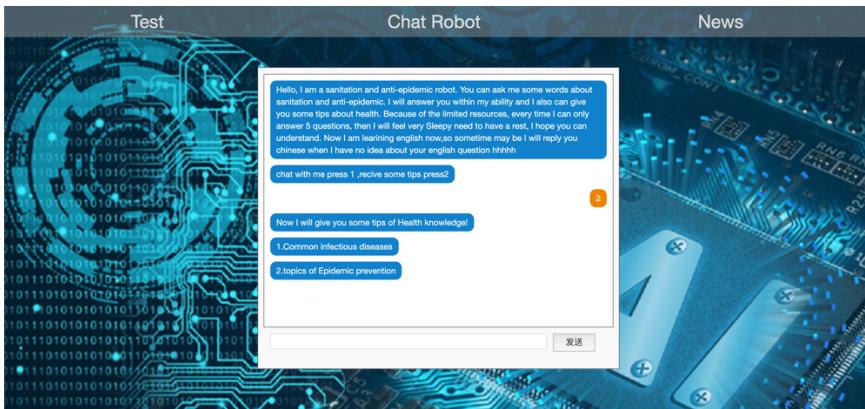
Your get wrong at **fibrosis, shortage of medical supplies, to close scenic spots, to monitor body temperature**  
Please learn them again!

Secondly, the main function was the chat bot, if you ask some question relative coronavirus-19, the chat will answer relative measures.

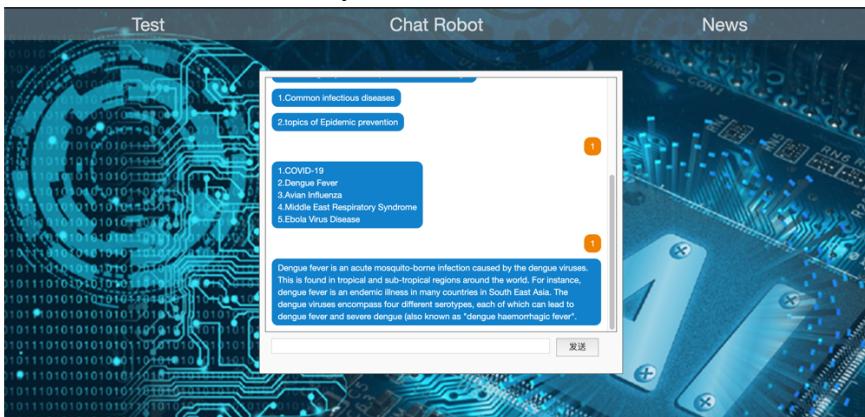
First of all, robot would ask you 'chat with me' or 'receive some tips'  
If you press 1, you could start chatting.



If you want some tips, you could press 2



select numbers which you want to know.



The third function is crawl all the news relative to coroavirus-19 from Ding Xiang Yuan App(Professional medical information app)



---

- \* Add a file called `milestone4.pdf` to your Github that answer the following questions:

- \* How is your project architecture related to the theory taught in the lecture?

We use the theories relative our lectures:

In Chapter02, our system is contemporary distributed system, which mean we do not have to construct a physical hardware, and do not have computer cluster. Only hire a service and push our code on it. we could use dynamic node in robot's service. We use cloud to compute the data and get the result.

In our vocabulary test function, we use AJAX in submit model, which make it load faster. AJAX means Asynchronous JavaScript and XML. Ajax can reduce the traffic travels between the client and the server, Usually, only data (in JSON or XML formats) will be sent, HTML and CSS codes are not transmitted. The server response time is faster so increases performance and speed. Data aren't processed in the server side, like being included in html.

```
    }else if(count > 0 && choose ==1){  
        $.ajax({  
            url: "https://bot7940.herokuapp.com/bot",  
            data: $("#talkwords").val(),  
            type: "POST",  
            success: function(response) {  
                console.log("OK: " + response);  
            }  
        });  
    }  
}
```

In Chapter03, lecturer introduce the RESTful idea, which is Create Data, Read Data, update data and Delete data. We use Get and Post methods to transmit data. Also, we use the Api to remote use data between our server side and client side.

In Chapter09, we use the cloud storage to storage our data for the chatbot, build the Failover system to make sure our server keeps stable.

- 
- \* Can you demonstrate, with some screen cap, how to increase capacity of your chat bot service?

```

1 from __future__ import unicode_literals
2
3 import os
4 import sys
5 import redis
6 import requests
7 from bs4 import BeautifulSoup
8 import re
9 from random import randint, sample
10 import random
11 import json
12 import time,datetime
13 import redis
14 from random import choice
15
16 from argparse import ArgumentParser
17
18 from flask import Flask, request, abort, json
19 from flask_cors import CORS, cross_origin
20 app = Flask(__name__)
21 cors = CORS(app)
22 app.config['CORS_HEADERS'] = 'Content-Type'

```

We use redis to improve our capacity, and we reduce the request of http while improve the connect speed.

```

    }
} else if(count ==2 && choose == 2){
    tep = secondchoose + "-" + $('#talkwords').val();
    console.log(tep);
    $.ajax({
        url: "https://bot7940.herokuapp.com/tips",
        data: tep,
        type: "POST",
        success: function(response) {
            console.log("OK: " + response);
            $("#words").append("<div class='btalk'><span id='bsay'>" + $('#talkwords').val() + "</span></div>");
            $("#words").append("<div class='atalk'><span id='asay'>" + response + "</span></div>");
            $('#talkwords').val("");
        },
        error: function(error) {
            console.log(error);
        }
    });
}

```

In this screenshot of background server, if our run status=200, it mean we are run this server successfully.

```

2020-04-15T12:26:27.300+00:00 heroku[web.1]: State changed from down to starting
2020-04-15T12:26:27.983252+00:00 app[web.1]: * Serving Flask app "back-end" (lazy loading)
2020-04-15T12:26:27.983271+00:00 app[web.1]: * Environment: production
2020-04-15T12:26:27.983338+00:00 app[web.1]:     WARNING: This is a development server. Do not use it in a production deployment.
2020-04-15T12:26:27.983394+00:00 app[web.1]:     Use a production WSGI server instead.
2020-04-15T12:26:27.983447+00:00 app[web.1]: * Debug mode: off
2020-04-15T12:26:27.984279+00:00 app[web.1]: * Running on http://0.0.0.0:19854/ (Press CTRL+C to quit)
2020-04-15T12:26:28.558832+00:00 heroku[web.1]: State changed from starting to up
2020-04-15T12:26:31.097749+00:00 heroku[router]: at=info method=GET path="/callback" host=bot7940.herokuapp.com request_id=03bb10f0-7b29-4ac3-ad09-bb396598f4e8 fwd="114.92.22.13" dyno=web.1 connect=1ms service=1672ms status=200 bytes=935 protocol=https
2020-04-15T12:26:31.095592+00:00 app[web.1]: 10.148.121.8 - - [15/Apr/2020 12:26:31] "[37mGET /callback HTTP/1.1[0m" 200 -
2020-04-15T12:26:42.688012+00:00 heroku[router]: at=info method=GET path="/news" host=bot7940.herokuapp.com request_id=6a24fc8e-bac5-4bbb-ac6b-425ab6776a43 fwd="114.92.22.13" dyno=web.1 connect=1ms service=5239ms status=200 bytes=10790 protocol=https
2020-04-15T12:26:42.679776+00:00 app[web.1]: 10.148.121.8 - - [15/Apr/2020 12:26:42] "[37mGET /news HTTP/1.1[0m" 200 -
2020-04-15T12:26:47.861488+00:00 app[web.1]: 10.148.121.8 - - [15/Apr/2020 12:26:47] "[37mGET /callback HTTP/1.1[0m" 200 -
2020-04-15T12:26:47.862977+00:00 heroku[router]: at=info method=GET path="/callback" host=bot7940.herokuapp.com request_id=c0486fea-a59b-4aa3-ae6c-e628ef21b781 fwd="114.92.22.13" dyno=web.1 connect=1ms service=503ms status=200 bytes=1185 protocol=https
2020-04-15T12:27:26.683426+00:00 app[web.1]: 10.63.251.11 - - [15/Apr/2020 12:27:26] "[37mPOST /bot HTTP/1.1[0m" 200 -
2020-04-15T12:27:26.685013+00:00 heroku[router]: at=info method=POST path="/bot" host=bot7940.herokuapp.com request_id=b59f5d5f-f643-46ce-bfc7-8eb57e4cdec0 fwd="114.92.22.13" dyno=web.1 connect=1ms service=2173ms status=200 bytes=218 protocol=https

```

In this segment of code, we combine two requests into http one, which make the data transmit with high efficiency

---

This code is for chat bot tips function, we use mset to reduce the code in server side.

```
PWD = "UrXY2jqaFoGABHS70AA1tq4AycoY0YX0"
PORT = "18573"
r = redis.Redis(host = HOST, password = PWD, port = PORT,decode_responses = True)
print('topics of Common infectious diseases and epidemic prevention')
print('topics of Epidemic prevention')
r.mset('1.COVID-19','Coronavirus Disease 2019 (COVID-19)" refers to the cluster of viral pneumonia cases occurring in Wuhan, Hubei Province, China, starting in December 2019. The disease is caused by a new strain of the coronavirus, SARS-CoV-2. It has spread rapidly across the world, leading to a global pandemic.')
print('Common infectious diseases')
r.mset('1.Why we wear Mask','Wearing mask is one of the measures to prevent respiratory tract infections. Surgical mask is a type of face mask that is used to cover the nose and mouth. It is made of three layers of fabric and has a pleated design that allows it to fit snugly around the nose and mouth. It is designed to filter out particles and droplets from the air that may contain viruses or bacteria.')
tep = ['1.COVID-19','2.Dengue Fever','3.Aavian Influenza','4.Middle East Respiratory Syndrome','5.Ebola Virus Disease']
cid = {1:'Why we wear Mask',2:'Why we use bleach',3:'Hand Hygiene',4:'Hand Hygiene',5:'how to drink Water healthily'}
print('tips of Health knowledge')
print('1.Common infectious diseases'+'\n'+2.topics of Epidemic prevention')

while True:
    k = int(input('What topic you want to know,press number 1 or 2+'\n'))
    if k == 1:
        print(cid[k])
    else:
        print(tep[k])
```

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

Our project are use PaaS(Platform as a Service), like Heroku. This model is a pre-defined ‘ready to use’ environment comprised of already deployed and configured IT resources. In this way, we no need to set-up and maintenance burden of IT resources on cloud consumer. There some advantage of PaaS, it makes our development and deployment more easily. And it has expansibility while reduce our coding works, Web service could be connected by many users in different device.