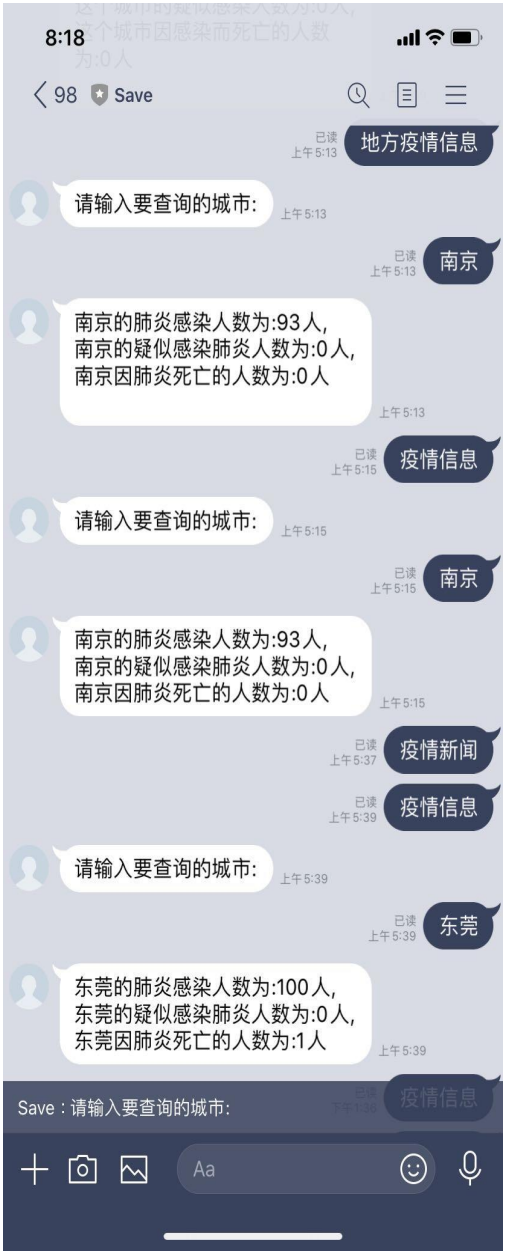
Milestone 4 project presentation and documentation

The method our project architecture related to theory taught in the lecture is as follow: We obtain the epidemic information of the domestic area on the web page through the web crawler and then change its status to json format for storage. When entering the “epidemic information”, it will further allow the user to enter the city name . After getting the city name, it get the specific information of the epidemic through string matching in the json file. Finally, these information will shown on the chat room automatically. If the users send a sticker to the chat bot , it will reply the box with two links. When users click the “国内疫情新闻” or “香港疫情新闻” the chat bot will turn to the page of websites respectively.



Demonstrate how to increase capacity of your chat bot service:

1. Improving server configuration, that is optimizing in terms of hardware

2. Multi-thread processing method:

We write the method of calling the function to separate each module, without the need to execute the program from top to bottom, which can reduce memory consumption, and the code looks more clear and concise, so that we can optimize later.

```python
def handle_Confirmed(event):
    input_text = event.message.text
    # 请求地址
    url = 'http://www.dzyong.top:3005/yiqing/area'
    # 发送get请求
    r = requests.get(url)
    t = r.json()

    if input_text == '疫情信息':
        line_bot_api.reply_message(
            event.reply_token,
            TextSendMessage(text='请输入要查询的城市:'))
    input_text_1 = event.message.text
    for item in t['data']:
        if item['cityName'] == input_text_1:
            confirm = item['confirmedCount']
            suspect = item['suspectedCount']
            dead = item['deadCount']
            line_bot_api.reply_message(
                event.reply_token,
                TextSendMessage(text=f'{input_text_1}的肺炎感染人数为:{confirm}人,\n{input_text_1}的疑似感染肺炎人数为:{suspect}人,\n{input_text_1}因肺炎死亡的人数为:{dead}人'))
```

```python
def handle_Confirmed(event):
    input_text = event.message.text
    # 请求地址
    url = 'http://www.dzyong.top:3005/yiqing/area'
    # 发送get请求
    r = requests.get(url)
    t = r.json()

    if input_text == '疫情信息':
        line_bot_api.reply_message(
            event.reply_token,
            TextSendMessage(text='请输入要查询的城市:'))
    input_text_1 = event.message.text
    for item in t['data']:
        if item['cityName'] == input_text_1:
            confirm = item['confirmedCount']
            suspect = item['suspectedCount']
            dead = item['deadCount']
            line_bot_api.reply_message(
                event.reply_token,
                TextSendMessage(text=f'{input_text_1}的肺炎感染人数为:{confirm}人,\n{input_text_1}的疑似感染肺炎人数为:{suspect}人,\n{input_text_1}因肺炎死亡的人数为:{dead}人'))


def handle_News(event):
    input_text = event.message.text
    if input_text == "疫情新闻":
        buttons_template = TemplateSendMessage(
        alt_text='Buttons Template',
        template=ButtonsTemplate(
            title='疫情新闻',
            text='了解实时疫情',
            thumbnail_image_url='https://timgsa.baidu.com/timg?image&quality=80&size=b9999_10000&sec=1586910091544&di=e1676a22ba60d21bbd925b0e9ed4767f&imgtype=0&src=http%3
            actions=[
```

Our bot is one of the examples of PaaS. Because we only need to develop our website program instead of buying a server or installing server software. It's advantage is that PaaS is primarily used by developers who build unique, customizable software or applications through the internet. In this case, by using PaaS, we don't need to start from scratch when creating applications, saving them a lot of time (and money) on writing extensive code. PaaS platforms are not only accessible by multiple users but also have the scalable characteristic which means we can choose from various tiers of resources to suit the size of our project. In addition, it built on virtualization technology and is easy to run without extensive system administration knowledge.

As for IaaS, users don't have to buy a server, and they can just buy a virtual machine at any place, but they still need to install the server software yourself. IaaS can provide you with the most flexibility and can also provide a common data center for data storage. However, it is the foundation of PaaS. That PaaS can reduce the need for system management.

When it comes to SaaS, users can just concentrate on operations and don't need to develop website programs. In this way, they just use the programs offered. Also, program upgrades, maintenance, and server additions are responsible by the companies. SaaS provides ready—to—use solutions that meet specific business needs (such as websites or e—mail). Most modern SaaS platforms are built on IaaS or PaaS platforms.