Milestone 3 - Information about the "another service"

Another Service 1 – Location Based Services

The core function of LINE Chatbot's Publish & Search Module is to publish and search for information about protective equipment such as face masks or cleaning substances based on geographic location information. Therefore, we need to use Location Based Services to obtain the user's location information.

LINE's dialog box has the function of sending location and can parse the Location Message into latitude, longitude and address information via google map API and return it to LINE Chatbot for processing. The following diagram 1 shows some scenarios where Location Based Services are used.
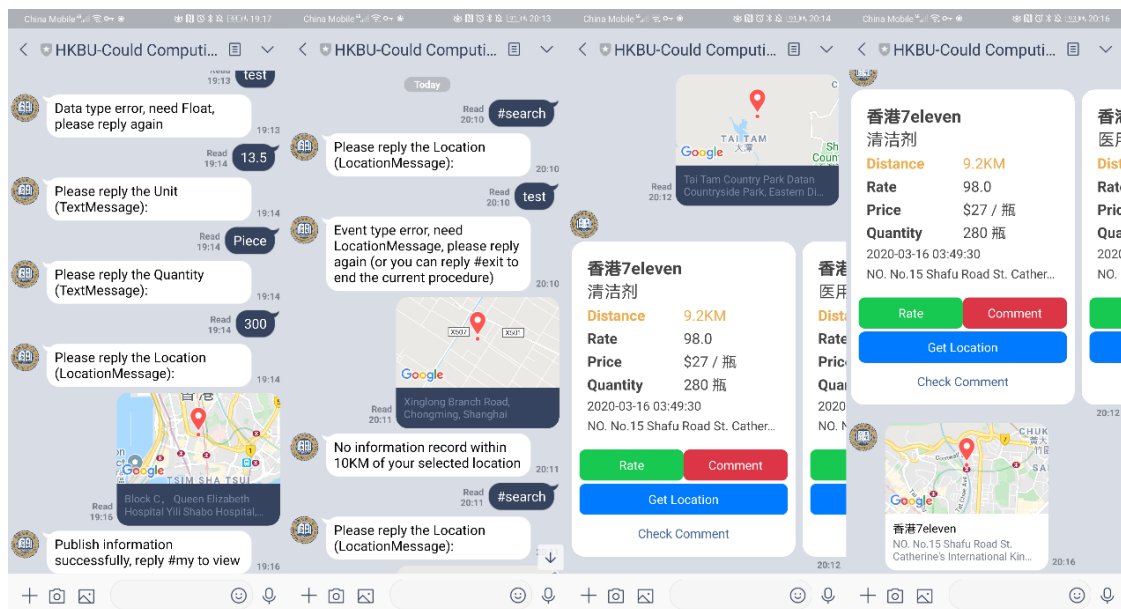

Figure 1

First, after the user responds to the "#publish" keyword, the user enters the process of publishing information. At the end of the process, the user needs to send his current location information. After the message, you can store the latitude and longitude data of the store together with other fields into the Redis database.

Then, at the same time, when other users reply with the "#search" keyword, the user sends the current location information. LINE Chatbot will calculate the distance between this location and other store locations stored in the Redis database, and finally returns all store information within the range of 10KM of current location in the form of a flex message. The information of these stores will be arranged in reversed order of published time, and the distance between each store and the current location will be highlighted. If there is no record that satisfied the requirements, it will return "No information record within 10KM of your selected location".

If you click the "Get Location" button, LINE Chatbot will reply a Location Message to indicate the location of the store. The user can click this Location Message and then click the "Shared location" button. Select "Directions from current location" to call Google map app to show directions to the store.

Sample Code:

Code of calculating the distance between two positions based on latitude and longitude:

```python
from geopy.distance import geodesic
LatLng1 = json.loads(self.__message)["latlng"]
LatLng2 = json.loads(dic["Location"])["latlng"]
distance = geodesic((LatLng1), (LatLng2)).km
```

Code of parsing the location message to the longitude, latitude and address:

```python
def handle_LocationMessage(event):
    dic = {
        "latlng": str(event.message.latitude) + "," + str(event.message.longitude),
        "address": event.message.address
    }
    text = json.dumps(dic)
    message = ProcessMessage(event.source.user_id,
                             text).public("LocationMessage")
```

Another Service 2 – News API Services

The main function of this section is to view news provided by the Hong Kong Government News Network (https://www.news.gov.hk/eng/index.html). It uses the news API provided by the government news network and obtains news data in XML format. The data file is parsed and converted into a carousel list on the LINE Chatbot.

When users use the Line Chatbot, they only need to send "@News" to the Line Chatbot, and the Line Chatbot will automatically obtain news data from the API of the Government News Network and parse the news data to generate a news list and push it to the news users. The details will show in figure 2.

User can view detailed information about this news by clicking the Read button below the news list. The details will show in figure 3 below.
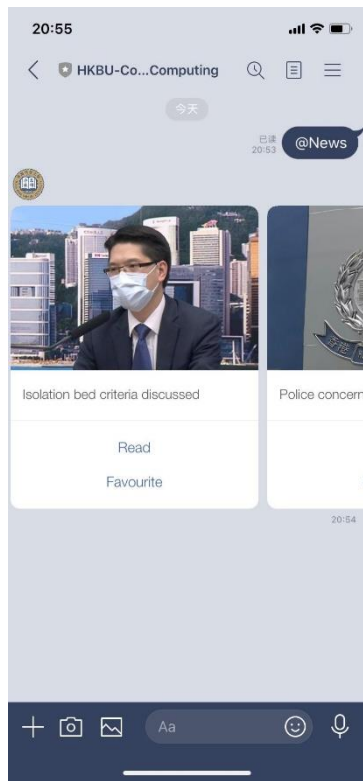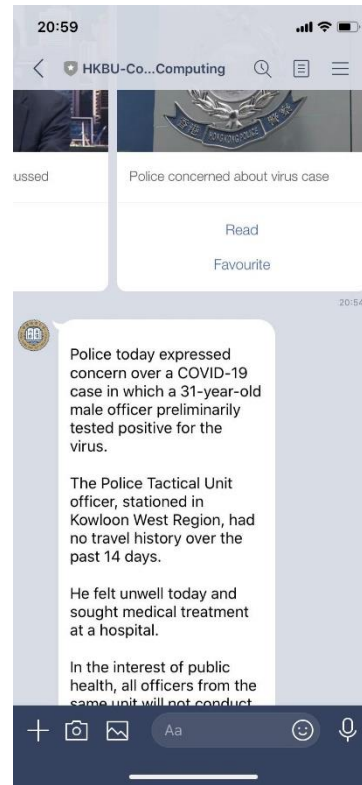

| Figure 2 | Figure 3 |

For some seniors, it is quite difficult to get updated news online. So, in this way, users who have difficulty of accessing the Internet can easily check the latest news of coronavirus issued by the government. So that they do not need to access websites to get the latest news. This Chatbot can simplify the operations when they are searching coronavirus news, and increase the efficiency. The most important thing is that this Chatbot provides news from reliable sources.

Sample Code:

Code of parsing XML and converted data into a carousel list on the LINE Chatbot:

```python
def __fetch_news(self):
    self.__redis.delete('temp')
    url = 'https://www.news.gov.hk/en/categories/covid19/html/articlelist.rss.xml'
    html = requests.get(url)
    xmlparse = xmltodict.parse(html.text)
    items = xmlparse['rss']['channel']['item']
    count = 0
    columns = []
    for item in items:
        count += 1
        if count == 4:
            break
        # title
        title = item['title']
        bs = BeautifulSoup(item['description'], 'lxml')
        # img
        image = bs.find('img')['src']
        # detail
        content = ''
        contents = bs.findAll('p')
        for paragraph in contents:
            content += paragraph.text
            content += '\n'
        detail = content
```

```python
        self.__redis.incr('index')
        index = self.__redis.get('index')
        self.__redis.hset('temp', index, json.dumps(News(title, detail, image).__dict__))
        columns.append(CarouselColumn(
            thumbnail_image_url=image,
            text=title,
            actions=[
                PostbackTemplateAction(
                    label='Read',
                    data='@Read=' + str(index)
                ),
                PostbackTemplateAction(
                    label='Favourite',
                    data='@Favourite=' + str(index)
                )
            ]
        ))
    message = TemplateSendMessage(
        alt_text='Carousel template',
        template=CarouselTemplate(
            columns=columns
        )
    )
    return message
```