Chatbot Operation Manual

Tingfeng Lin z5147976

Notes:

- Please use Python 3.6 or above to test my program.
- In deployment instructions, I use red bold underline italic style to highlight the operations. You can copy and paste these commands directly following my instructions to run my program.
- In this project, I am using Mlab (remote MangoDB system) as my database. If you are going to check my service logic and data, I would be glad to help you access to my remote database.
  - Website: https://mlab.com/
  - Account: tracy1111
  - Password: ltf9495!
- Because of the network environment, message response needs to wait 2-3 seconds. When you press *Enter*, please wait 2-3 seconds for response.
- If you have any problem during deployment, please send me an email z5147976@ad.unsw.edu.au

1. Project design principle

In this project, I am going to use Docker to pack doctor module, timeslot module and chatbot module into three separate services. Each service is isolated and communicate with others.

```
emondeMBP:chatbot-app lemontracy$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS                    NAMES
84aad90e6ef         time                "/bin/sh -c 'python3…"   About an hour ago   Up About an hour    0.0.0.0:9999->5000/tcp   hardcore_diffie
e2acfd4389c         chatbot             "/bin/sh -c 'python3…"   About an hour ago   Up About an hour    0.0.0.0:5002->5000/tcp   adoring_noether
2460c1bbf97         doctor              "/bin/sh -c 'python3…"   About an hour ago   Up About an hour    0.0.0.0:7777->5000/tcp   quizzical_goodall
```

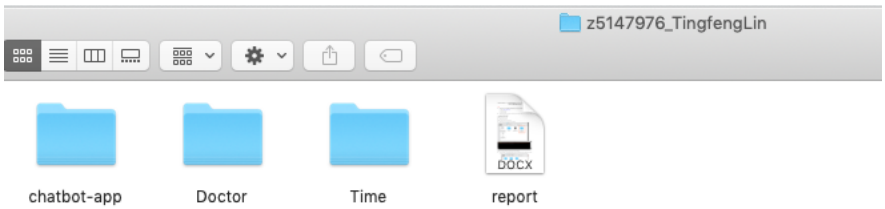
2. Deployment instructions

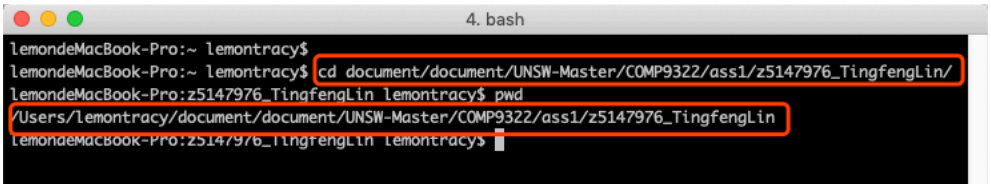
Step 1: Unzip the file in a folder

- Unzip the file in any folder.

Step 2: Using Terminal to access to the folder

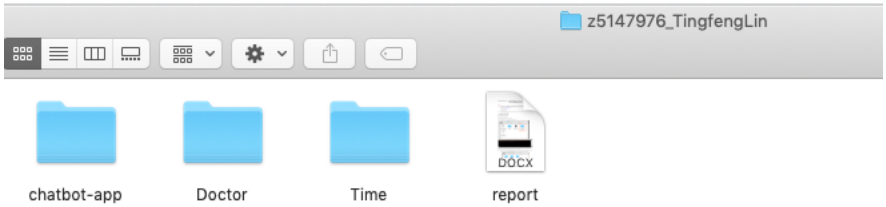
- Using Terminal to access the folder you just unzipped



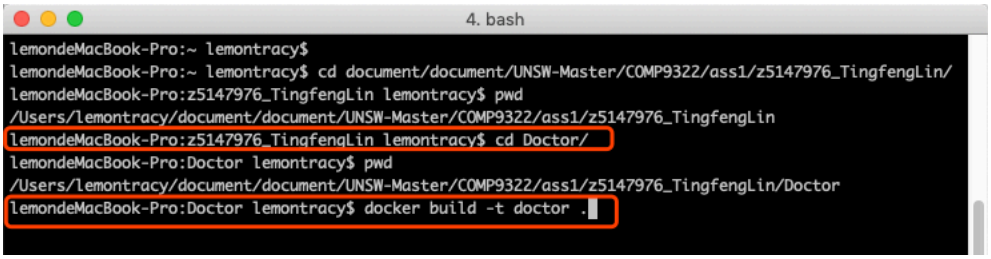


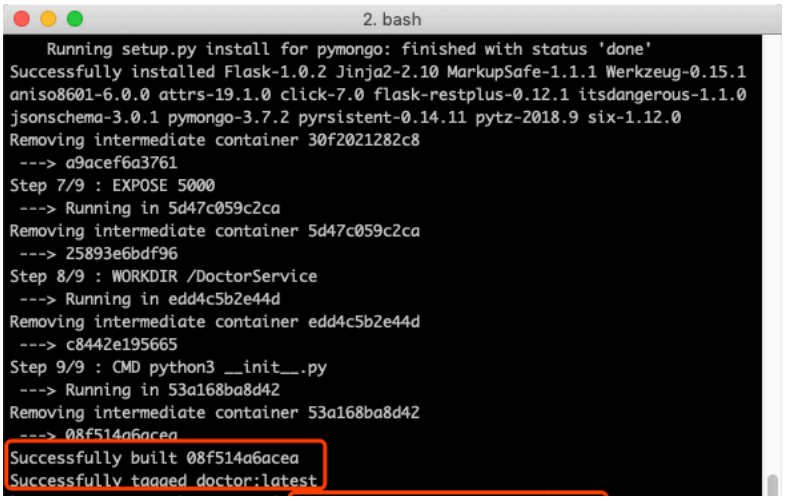
Step 3: Accessing the specific folder and execute the relevant commands

- In this folder, there are three sub-folders. One is Doctor which is the Doctor Service. Another is Time which is obviously the Time Service. The other is chatbot-app which is the interactive part (front end).

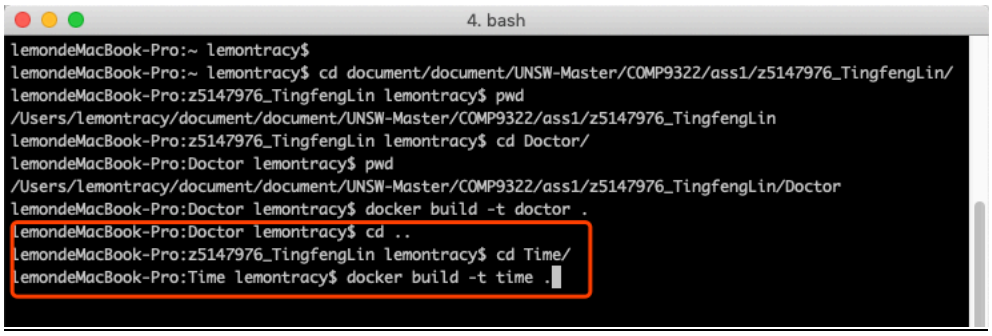


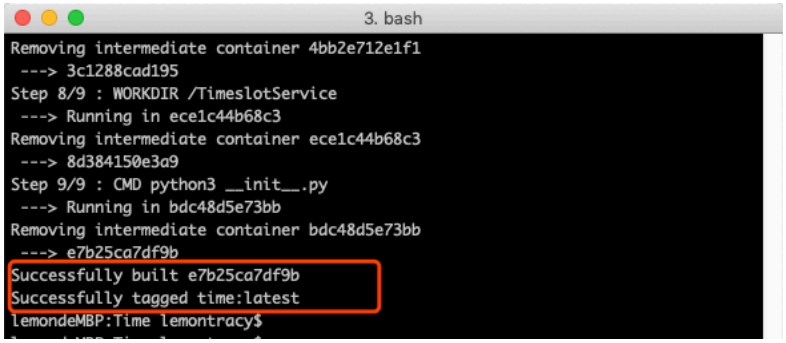
- For Doctor Service, ***cd Doctor*** folder and type command ***docker build -t doctor .*** to create an image *(Notes: There is a comma at the end of the command).* After building images, it will show 'Successfully built XXXX'. This means image is created perfectly.



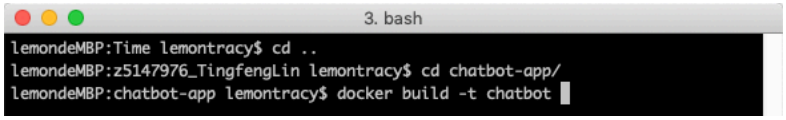


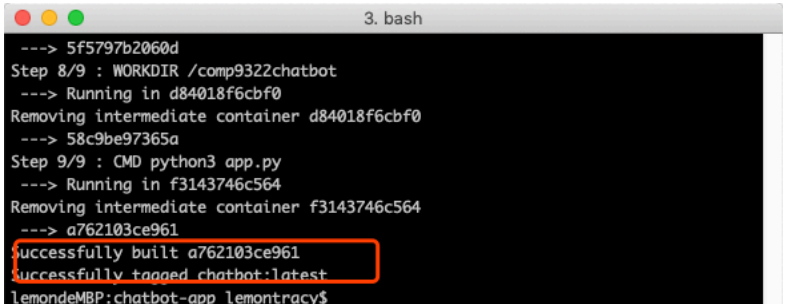
- For Time Service, ***cd ..*** then ***cd Time*** and type command ***docker build -t time .*** to create an image *(Notes: There is a comma at the end of the command).* After building images, it will show 'Successfully built XXXX'. This means image is created perfectly.





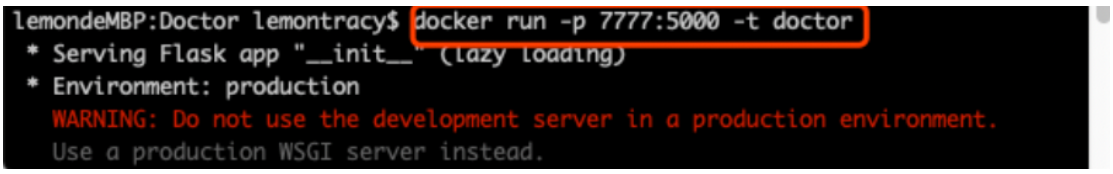
- For Chatbot Service, ***cd ..*** then ***cd chatbot-app*** and type command ***docker build -t chatbot .*** to create an image *(Notes: There is a comma at the end of the command).* After building images, it will show 'Successfully built XXXX'. This means image is created perfectly.



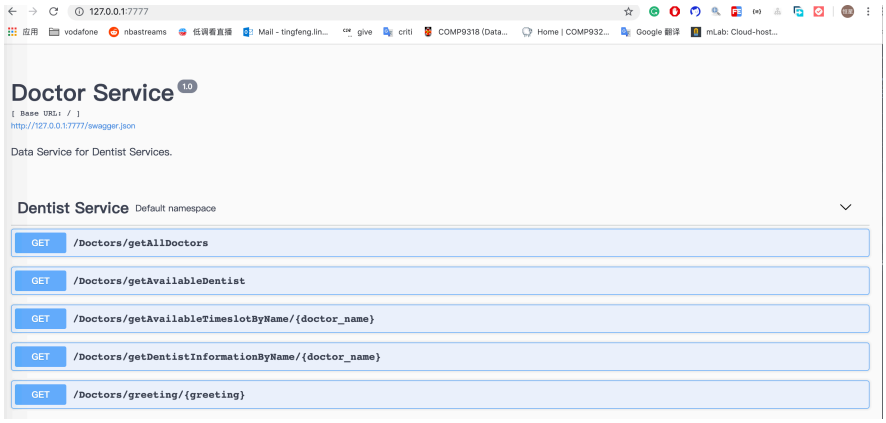


Step 4: Running Doctor image

- Next, type command ***docker run -p 7777:5000 -t doctor*** to run the image. It might take you 1-2 second to start this Doctor Service.

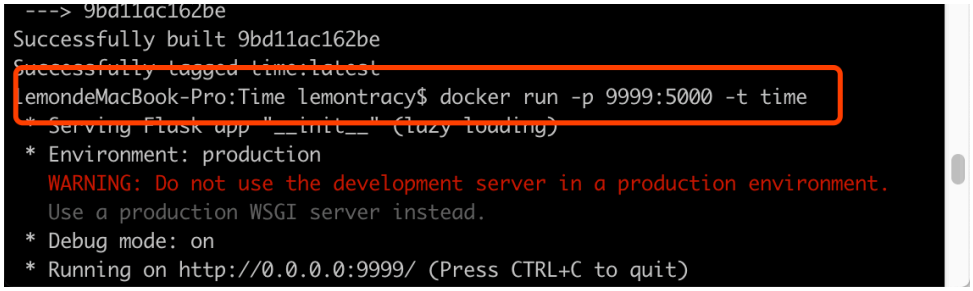


- After that, typing http://127.0.0.1:7777/in browser, then you will see the swagger interface as shown below

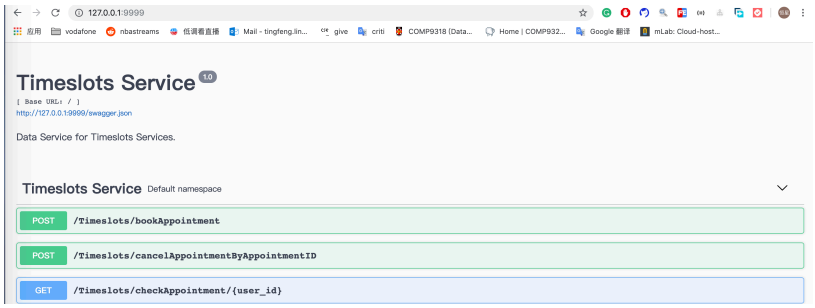


Step 5: Open a new Terminal and run Time images

- Open a new Terminal application then type command ***docker run -p 9999:5000 -t time*** to run the image. It might take you 1-2 second to start this Time Service.

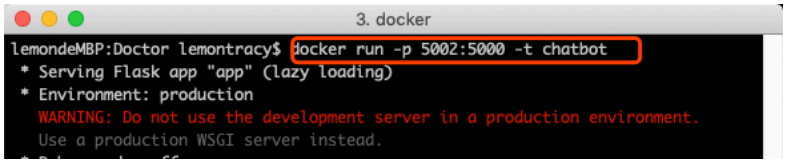


- After that, typing http://127.0.0.1:9999/ in browser, then you will see the swagger interface as shown below

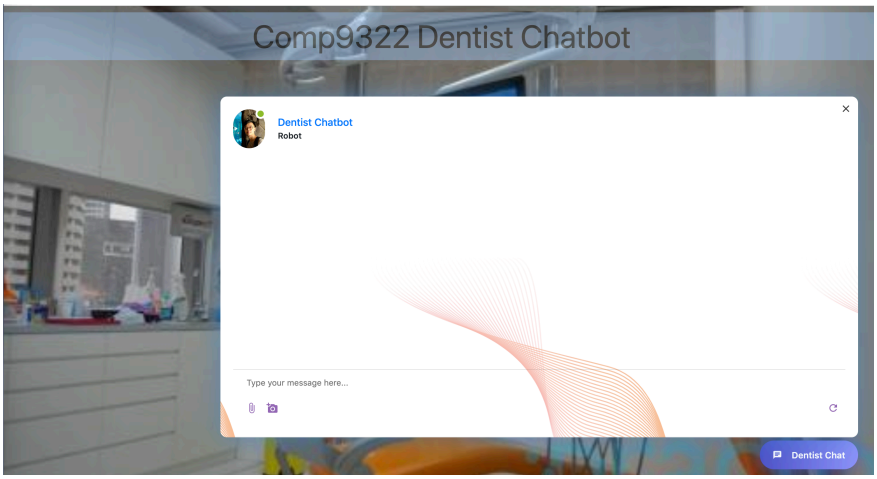


Step 6: Open a new Terminal and run Chatbot images

- Open a new Terminal application then type command ***docker run -p 5002:5000 -t chatbot*** to run the image. It might take you 1-2 second to start this Time Service.



- After that, typing http://127.0.0.1:5002/ in browser to access to the application.

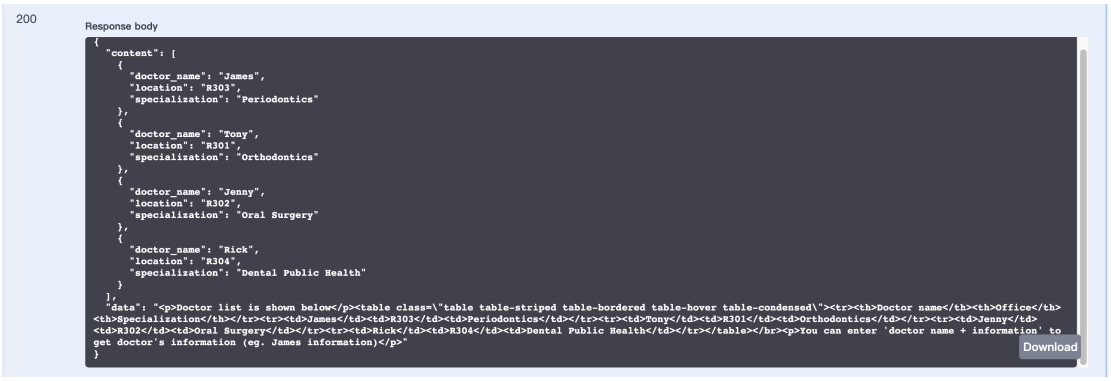


3. Restful API introduction

In Doctor Service, there are five APIs as shown below.

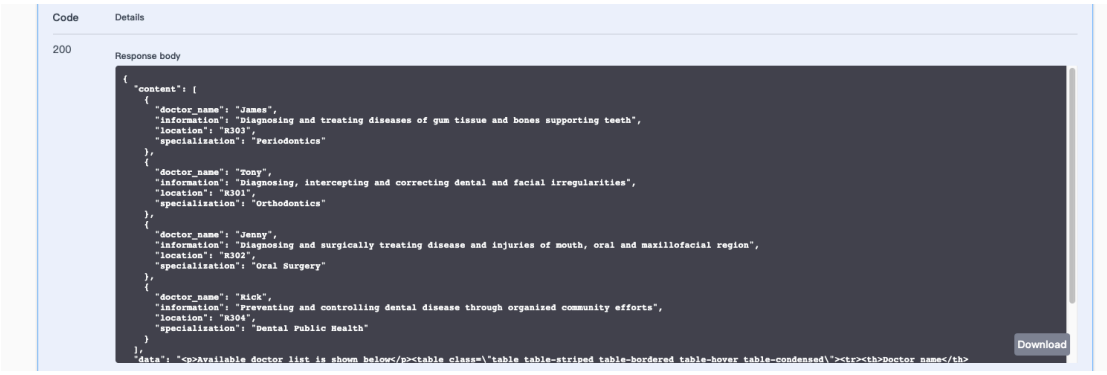
- **/Doctor/getAllDoctor**

  Executing this API by clicking button, it will show information of each doctor.



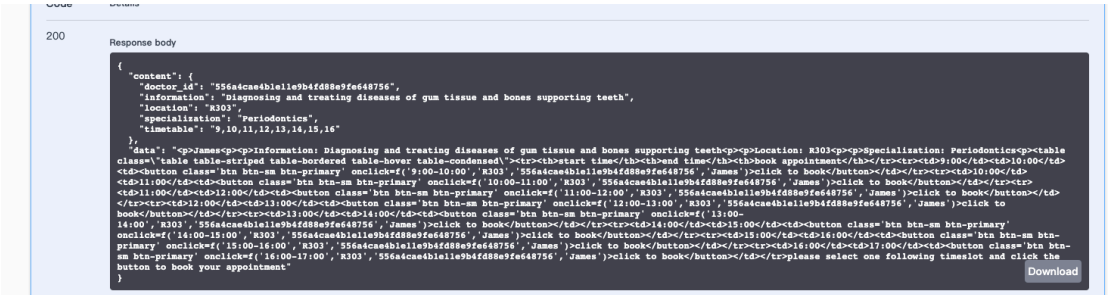
- **/Doctor/getAvailableDentist**

  Clicking the execution button, it will give you all dentists who still have a timeslot to be booked. If there has a doctor whose schedule is full. This doctor will not appear.



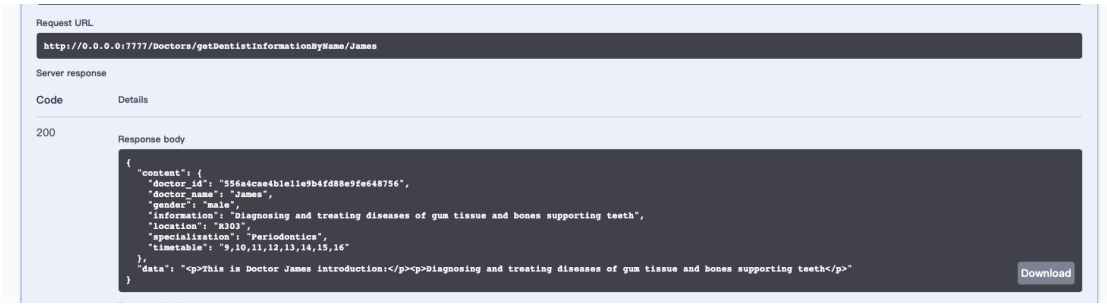
- **/Doctors/getAvailableTimeslotByName/{doctor_name}**

  In this API, it requires a user to type doctor's name. In this situation, you can firstly execute *getAllDoctor* API to get a doctor's name then type it into the input area. I am going to type James who is a doctor in the hospital.



- **/Doctors/getDentistInformationByName/{doctor_name}**

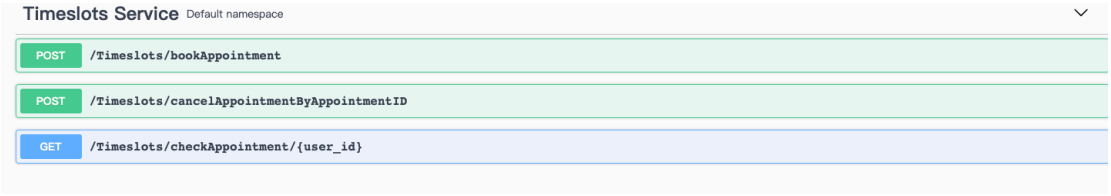
  This API also needs user to input a doctor's name. I type James into the input area.



- **/Doctors/greeting/{greeting}**

  I put a common greeting function in Doctor Service. Typing the common greeting in the input area then you will get feedback.

In Time Service, there are three APIs as shown below.

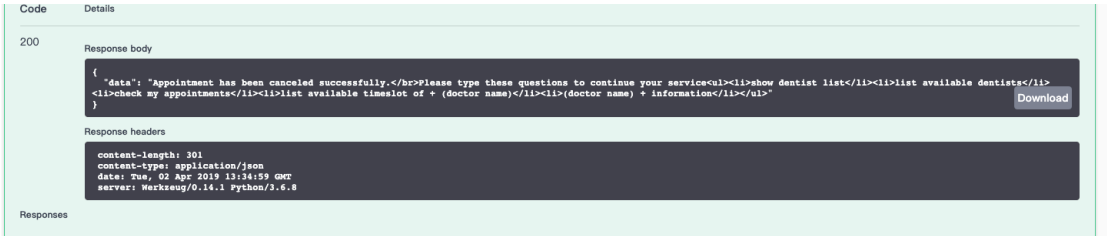


- **/Timeslots/bookAppointment**

  There are several values should be passed into this POST request.

- **/Timeslots/cancelAppointmentByAppointmentID**

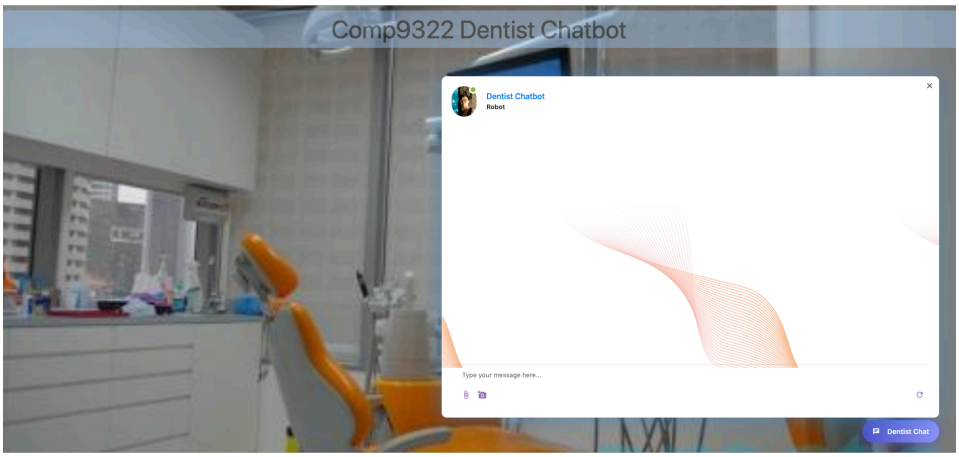
  In this API, the appointment ID should be passed into the API.



- **/Timeslots/checkAppointment/{user_id}**

  This is a GET API, a user has to enter their user id. In order to get the specific user id, you have to log in to my database to get the ID. There is no API can get the user ID.

4. Chatbot operation introduction

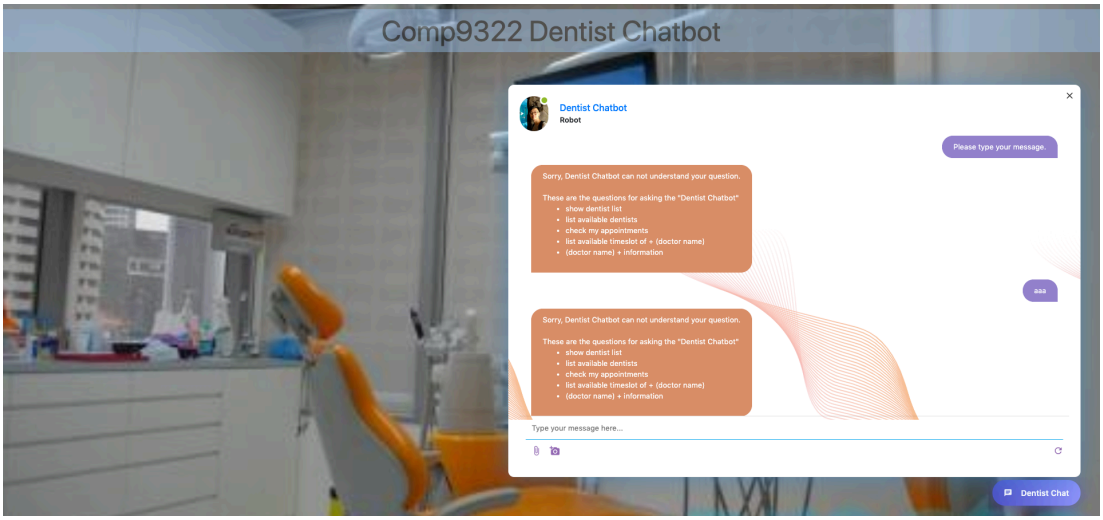


This is the interface of chatbot. The user needs to type question in the input area and press *Enter* to send the message. Due to the network environment, it might take you 2-3 second to send the message and get the response.

The chatbot can identify some specific or related sentences such as

- show dentist list
- list available dentists
- check my appointments
- list available timeslot of + (doctor name)
- (doctor name) + information

When you type these questions or other similar questions, chatbot might give you related response. However, when a user types a question that chatbot cannot identify or press enter without any question, it will response some hints and require the user to type a new question again.



5. Bonus part

In assignment requirement, it mention that display the timeslot graphically. I have completed this task. These are the screenshots.