

Virtual Clinic github repository <https://github.com/hp-gh/virtual-clinic>

Virtual Clinic was a project completed for COMP9323 by Adam Golding, Horace Pang, William Ling and Yash Diwan. The purpose of this project was to create a full web-app solution for doctors to use during the 2020 Pandemic to make telemed easier. The following documentation include instructions on how to deploy and use our solution.

Table of Contents

[Dependencies](#)

[AWS CLI](#)

- [Configure Your AWS CLI](#)
- [Create an S3 Bucket](#)

[Zoom API](#)

[Data Layer](#)

- [1\) Create Database and Insert Reference Data](#)
- [2\) .ini File](#)

[API Layer](#)

- [1\) Create a Virtual Environment](#)
- [2\) Activate the Virtual Environment](#)
- [3\) Install Requirements and Run RESTful API](#)

[Front End](#)

[Using Main Features](#)

- [Patient Portal](#)
- [Doctor Portal](#)

Dependencies

You will need to have the following installed on your local machine before executing the deployment instructions

Dependencies

[Python 3](#)

[Node.js](#)

[AWS CLI](#)

[PostgreSQL](#)

[Zoom](#)

AWS CLI

A dependency for running this project locally is having [AWS CLI installed on your system](#).

Configure Your AWS CLI

[Download your credentials CSV](#) and place that file in the highest level of this repository (Make sure your current working directory is the highest level of the repository as well). This file should end in either `accessKeys.csv` or `credentials.csv`.

Run the `aws_configure.sh` command by running the following command:

```
$ chmod 755 aws_configure.sh
$ ./aws_configure.sh <name_of_your_credentials_file>
```

If your machine cannot run shell scripts, you can manually configure your AWS CLI by running the `aws configure` command:

```
$ aws configure
AWS Access Key ID [*****]: <your AWS Access Key here>
AWS Secret Access Key [*****]: <your AWS Secret Access Key here>
Default region name [ap-southeast-2]: ap-southeast-2
Default output format [json]: json
```

If you are configuring your CLI manually it is important that you set the region to `ap-southeast-2`.

Create an S3 Bucket

Before running this project you must also create an AWS S3 Bucket.

Instructions to do so can be found [here](#). Ensure that the region of the bucket is set to `ap-southeast-2`.

Copy the name of your newly created bucket to the `s3.cfg` file located in:

```
/virtual-clinic/virtual_clinic/api/app
```

`s3.cfg`

```
S3_BUCKET = '<your S3 Bucket name here>'
```

Zoom API

In order for the Zoom functionality to work you must have the following:

Zoom Account - can be setup at <https://marketplace.zoom.us/>

Zoom JWT App - once logged into Zoom Marketplace:

1. Build a JWT app (Develop->Build App menu)
2. Access the credentials

```
API_KEY
API_SECRET
```

Edit the `zoom.ini` file and copy your credentials in the relevant fields. The `zoom.ini` file is located in

```
/virtual-clinic/virtual_clinic/api/app/zoom
```

zoom.ini

```
[zoom]
api_key=xxxxxxxxxxxxxxxx (copied from Zoom JWT app)
api_secret=xxxxxxxxxxxxxxxx (copied from Zoom JWT app)
user_email=yyyyyyyyyyyyyy@gmail.com (email address of your Zoom account)
```

Data Layer

The database is built using PostgreSQL. This can be deployed locally or as a cloud service.

1) Create Database and Insert Reference Data

1.1. Create database 'virtual_clinic'

1.2. Connect to the database 'virtual_clinic'. For example, in psql run:

```
\c virtual_clinic
```

1.3. This script contains the create tables, constraints and the role vc_user. The script can be executed in pgAdmin or psql command line

Note: Before executing this script please read the instructions in the header of this file if you have changed from the default database and/or user name

```
/virtual-clinic/database_scripts/create_tables_1.1.sql
```

1.4. This script contains the insert statements to load the reference data into the reference tables. The script can be run using pgAdmin or using the psql command line

```
/virtual-clinic/database_scripts/insert_reference_data_1.1.sql
```

2) .ini File

The .ini file contains a number of parameters for the database connection that are used by the Python data access object classes. The values in the .ini file have been defaulted to those used in the scripts. If you have used any different values they need to be changed here. The .ini file is located:

```
/virtual-clinic/virtual_clinic/api/app/dao/database.ini
```

This file has a number of sections:

```
[postgresql]
host=127.0.0.1 (name of the host database server)
database=virtual_clinic (name of the database)
user=vc_user (name of the database user/role that is used to connect to the database)
password=password123 (password of the database user/role)
```

The following section relates to the database connection pooling

```
[pool]
min=2 (the minimum number of connections in the pool)
max=4 (the maximum number of connections in the pool)
```

API Layer

To run the API layer and see the Swagger Docs for our Flask API, do the following:

1) Create a Virtual Environment

Create a virtual environment by running the following command while in the highest level of the git repo directory if you have Python 3.4 or newer:

```
$ python3 -m venv <nameofyourvenv>
```

If you have an older version of Python, you should then run the following to create a virtual environment, having installed the third-party tool virtualenv

```
$ virtualenv <nameofyourvenv>
```

2) Activate the Virtual Environment

If using Microsoft Windows, activate your venv by running the following:

```
$ <nameofyourvenv>\Scripts\activate
```

Otherwise, activate your virtual environment by running this:

```
$ source <nameofyourvenv>/bin/activate
```

3) Install Requirements and Run RESTful API

Then execute the following script to install all requirements for your virtual environment and run our flask app:

```
$ chmod 755 run_api.sh  
$ ./run_api.sh
```

If on Windows:

```
$ ./run_api.ps1
```

You can now see and explore the Swagger docs for the VirtualClinic API by going to <http://127.0.0.1:5000/> in your browser.

Front End

First, make sure the API layer is already running by following the above instructions.

Then change your directory into 'virtual_clinic':

```
$ cd virtual_clinic
```

Execute the following command to install all Node.js modules required:

```
$ npm install
```

Once all modules are installed, you can start the server with:

```
$ npm start
```

Open <http://localhost:3000> to view the app in the browser.

Using Main Features

Having deployed our solution, you can inspect and test the Flask API by visiting the Swagger Docs at <http://127.0.0.1:5000/>

You can explore our website by visiting <http://localhost:3000>.

Patient Portal

Booking an Appointment

On the home page of the patient portal, you can view all doctors that are available for a consultation. Choose the doctor you wish to book with and click 'Book with <Dr. Name>'.

This redirects you to the appointment booking page. Select a date and time before clicking submit to book your appointments.

Viewing Appointments

From the Home page, click on 'View Upcoming Appointments' on the top-right corner. This takes you to a page where you can upload medical files and view all upcoming appointments.

File Upload

To upload a file, click 'Choose File' under Upload Files and select the file you wish to show the practitioner before clicking 'Submit'. Once it's successfully uploaded, you will be able to see the file under Your Uploaded Documents. Simply click the file name to preview it.

File Deletion

To delete a file, simply click 'Delete' next to the file you have uploaded.

Appointment Cancellation

To cancel an appointment, simply click on 'Cancel Appointment' on the one you wish to cancel.

Starting the Appointment

To begin your appointment, click the link under 'Zoom URL' and it will take you to a Zoom meeting where the doctor will be meet you.

Doctor Portal

Viewing Consultations

From the Doctor home page, click on 'View Upcoming Consultations' in order to see all the upcoming consultations you have'

Viewing Patient Documents

For each upcoming consultation, you can click on the link under 'Uploaded Files' to view the documents uploaded by that patient. A modal will open up and clicking on the name of the file will allow you to view it.

Appointment Cancellation

Similarly to the patient, to cancel an appointment, simply click on 'Cancel Appointment' on the one you wish to cancel.

Hosting the Appointment

To begin your appointment, click the link under 'Zoom URL' and it will create a Zoom meeting where you are the host. The patient will be connected once they join