# Special Instructions (Deployment) of Ventilator Weaning Assistant System (VWAS)

Xu Shi, Xiaoyang Xia, Xingdan Wang, Kunming Zhu, Chen Hu, Chan Manfred
xshi307@gatech.edu, xxia44@gatech.edu, xwang3010@gatech.edu,
kzhu81@gatech.edu, chu93@gatech.edu, mchan83@gatech.edu

## 1 FRONT END

The frontend was developed using HTML with embedded JavaScript with several framework libraries and toolbox, including Bootstrap.js, d3.js, jQuery.js, vue.js, tf.js, and tensorflow.js.

*Note: It is not an APP-style so there is no need to deploy the frontend code. To run the frontend, simply start a local server and navigate to the index.html in your localhost.*

## 2 DEPLOYMENT ON GITHUB

To make the APP public accessible, we deploy it on GitHub using the Page feature. Simply we upload the frontend code to GitHub repo and turn on the page feature, the public URL https://github.gatech.edu/pages/xwang3010/VWAS/.

## 3 SOURCE CODE REPO

The source code, including frontend script (JS), Patient/Observation bundle generation (python), and Machine-Learning modeling (python), can be accessed via GitHub repo: https://github.gatech.edu/xwang3010/VWAS.

## 4 API

Once the frontend index.html initialized, script will asynchronized fetch the Patient and Observation data from FHIR Backend Database using URL route. After the query finished, the script will then do the processing and render the page.

## 5 FHIR BACKEND DATABASE (HAPI)

The backend data storage we are using is HAPI FHIR server (http://hapi.fhir.org/baseR4). We used the data uploading API provided by HAPI to upload the patient and observation resources to HAPI FHIR server. The public server used is UHN_HAPI Server (R4 FHIR). This is not a production server and all health information we upload to this server is synthetic data. No personal or confidential health data are used in our project. This server will be regularly purged and reloaded with fixed test data. We identify 20 synthetic patient data to be used for application demonstration. The 20 patient data are generated by Synthea (Section 6).

For each patient:

- We first created a patient resource JSON file from Synthea JSON output and uploaded it to the HAPI FHIR server using the provided API https://hapi.fhir.org/baseR4/Patient?_format=json. Patient ID in the server will be generated after patient resource upload.
- We then get observation reading of this patient from the observation.csv generated by Synthea and save it to "PatientID.csv". PatientID will be updated for each patient.
- Run updateFHIR.ipynb to convert observation data in csv file into observation bundle JSON file "observation.json".
- Finally, we upload the observation bundle data in "observation.json" to HAPI FHIR server endpoint: https://hapi.fhir.org/home?serverId=home_r4&pretty=true&_summary=&resource=

20 patient ID are listed below:

| List | Patient ID |
|---|---|
| 1 | 2049254 |
| 2 | 2049404 |
| 3 | 2051023 |

| | |
|---|---|
| 4 | 2051024 |
| 5 | 2051026 |
| 6 | 2051027 |
| 7 | 2051028 |
| 8 | 2051029 |
| 9 | 2051030 |
| 10 | 2051031 |
| 11 | 2051032 |
| 12 | 2051033 |
| 13 | 2051034 |
| 14 | 2051035 |
| 15 | 2051036 |
| 16 | 2051037 |
| 17 | 2051038 |
| 18 | 2051039 |
| 19 | 2051040 |
| 20 | 2051025 |

## 6 DATA GENERATION (SYNTHEA)

We use a modified version of Synthea, a synthetic patient data generator, to create our patient and observation datasets. The original program can be found at (https://github.com/synthetichealth/synthea), but our repo contains a modified clone of the full repo for ease of setup and execution. It is also available at (https://github.gatech.edu/mchan83/IHI-Synthea).

To generate patients, run the following:

```
bash ./run_synthea -p 1000 --exporter.csv.export true
```

The output will appear in ./output; with these options Synthea creates both CSV and json outputs. The former is processed to make the model training data, and the latter is used to create FHIR bundles.

To convert csv to class balanced ML model format, ensure global flag balance_dataset in convert_csv.py is set to True and run the following:

```
python ./convert_CSV.py
```

The output will appear in ./output/vent_data.csv

There are patients that are incorrectly generated due to bugs, which appear to be in how the underlying Synthea code handles probabilities. These patients are identified and excluded during the ML format conversion, and are listed in ./output/DO NOT USE THESE PATIENTS.txt; they remain in the json output and have to be filtered out before running the FHIR bundle code. To do so, run the following:

```
python ./removeDNU.py
```