**Local deployment walkthrough:**

This document is a walkthrough on how to deploy the website locally. It also covers how to deploy bulk-data-server and Synthea. It then walks through how to add your own data using Synthea to the bulk-data-server and then pulling that data to use it in the diabetes website.

**To deploy diabetes website locally**:

1. Download GitHub files locally - <https://github.com/K-Gatech/Diabetes-site>
2. Cd into the location and run *npm install* to install node\_modules.
3. With the existing files it should open up the website locally (run *npm start*)

This will open up the website with the existing data.json file that was pulled from smart health and manipulated to have the gender option “other” and corrected A1C data.

**To pull data from smart health**:

This will involve multiple steps that will be broken down in more detail below but generally they will include:

1. Download bulk-data-server locally and follow github instruction to deploy it <https://github.com/smart-on-fhir/bulk-data-server>
2. Download synthea and follow instructions on github - <https://github.com/synthetichealth/synthea>
3. Follow instructions to post synthea data into bulk-data-server
4. Download synthea data from SMART bulk-data-server
5. Place files in “data-read” respective locations and run the script
6. This will give you a json file called “data”
7. You will then place this “data” file into the src folder in the main diabetes website files

Bulk-data-server steps:

1. Follow instructions on github
   1. Run “npm i” and then “npm start”
2. When you open SMART bulk data server you scroll down to Database size and select 1000 patients
3. Then scroll down to “Try Sample App”
4. Then select “Start Export” and download the Observation.ndjson and the Patient.ndjson
5. Open the files and it should have data in them. This is to ensure that its actually working before we add data using synthea.
6. Follow the steps in the next section to add data using Synthea
7. After adding data using Synthea you will follow the above steps and then you will use the Observation.ndjson and the Patient.ndjson and place them in the “data-read” file

Adding correct data to smart bulk data server using Synthea:

1. Cd into the synthea folder and run “./run\_synthea -p 100 -m metabolic\*”. This will populate 100 patients at random and some will have A1C data.
   1. Note: you will need JAVA installed for this
2. Then open bulk-data-server repo while cd’d into it and run “npm run import -- -f 4 -d “/….synthea-master/output/fhir”. The … will depend on the location of your synthea fhir repository.
3. This will upload the data to the smart bulk data server.

Downloading the new Synthea data from smart bulk data:

1. Follow the steps mentioned in “Bulk-data-server steps”.
2. The downloaded observation and patient data will now include the new patients added.

Analyzing the data and pulling the correct json needed

1. Download the scripts in the “data-read” folder. This will be used to analyze the download data.
2. Add the path of the observation and patient files in the correct location
3. Run process-data-for-test script.
4. It will generate a data.json file
5. Place this file in the src folder of the main project folder

Running main Diabetes site locally

1. After placing data.json in the src folder of the main website you can load up the website using “npm start” and should display the new A1C patient data you pulled
2. Note: this data will be different than the data that’s on github since it was pulled and randomized through this process. It will also only contain male or female options.
3. The graphs will only display up to 10 patients since I had the cut off at 5 to accommodate 5 female, 5 male, and 5 other in my original dataset.
4. The A1C data is random so it won’t exhibit the same relationship discussed in the paper.