

Project Manual

Team Name- YMCA

Team Members- Dinesh Singh, Rajkamal, yuping Shao, John Stroud

Project Name- CDC Guidelines Decision Support System

GitHub Link- <https://github.gatech.edu/gt-hit-fall2016/CDC-Guidelines-Decision-Support>

What is our project:

This decision support tool can help practitioners making an informed decision for a patient based on zika symptoms or other factors like ; lab results, zika related symptoms or patient's travel history to the zika affected areas. This tool can save valuable time in identifying the potential diseases and ultimately will lead to timely treatment.

Our application makes api calls to Georgia tech FHIR server to get resources like patient details, lab results and symptoms contained in observation resource. The data is then fed to the decision algorithm based on bayesian network for zika infection probability calculations. if the probability of the patient infected with zika is let's say more 60% , then the practitioner is alerted to follow additional CDC Zika guidelines for the patient.

Application Directories and Files:

1- /Final Project/Final Application/Application/cs6440 : This directory contains web application and its modules.

index.html - Index page of the web application

get-data.js - module to get data from the decision engine and the fhir server

Fhir-client.js - fhir client application to connect to fhir server

2- /Final Project/Final Application/Application/guideline :This directory contains cdc guidelines and python based decision engine

**txt files-* Are CDC provided guidelines

**csv files-* Are our formatted guidelines which are used by the decision engine and are customizable.

zik_detection.py - flask app and diagnostic algorithm engine

What are different application Modules:

- **Web application-** get-data.js , index.html
- **Python decision engine-**
 - Library: Requires pandas, flask, flask-cors, python 2.7
- **Guidelines -**
 - See ~/Application/guideline/zika_sample_guideline.csv for full guideline format.

- Guideline can be customized by adding more symptom, lab-test, lab-result, affected area codes. Each record row could contain a single code or conjunction of codes. The only requirement is that each row MUST have Likelihood, Negative Likelihood, and Weight assigned to the row. No error handling present for incorrectly formatted guideline as of now.

How to run the application:

Requirements- Linux or windows OS, apache httpd server , python 2.7 with pandas (0.19.1), Flask (0.11.1) ,Flask-Cors (3.0.2) libraries.

Following instructions are for Centos 7.2 64 bit Operating system-

1- update the OS - # sudo yum -y update

1- Install apache httpd server : # sudo yum -y install httpd

2- Install python 2.7 : # sudo yum -y install python

3- Enable EPEL 7 repository for pip installation (pip will be used to install other python libraries like pandas and flask) -

rpm -ivh <https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm>

4- Install python libraries - #sudo pip install flask flask-cors pandas

5- Modify ~/Application/guideline/zik_detection.py - Go to the end of th file and change app.run(host="192.168.1.17") to your local/server IP app.run(host="YourLocalIP")

6- Go to /Application/guideline/ and run the Python decision engine by executing as a python script-

python zik_detection.py &

This will start the python decision engine in the background and start listening on port 5000 by default

You can test it by visiting the url <http://yourlocalIP:5000/>

You should get a "hi" in the browser

7- Modify the /etc/httpd/conf/httpd.conf - set the DocumentRoot directory for httpd to the application directory ::: DocumentRoot "/Final Project/Final Application/Application/cs6440" , **This is to direct httpd to look for index.html in the cs6440 directory.**

Allow the pages to be accessed by all :

-----sample config-----

DocumentRoot "/Final Project/Final Application/Application/cs6440"

<Directory "/Final Project/Final Application/Application/cs6440">

AllowOverride None

Options Indexes FollowSymLinks

Allow from all

</Directory>

-----sample config-----

Also change the httpd listening port under “**Listen 0.0.0.0:9090**” - our httpd is running on port 9090 - this is optional.

8- Modify the cs6440/get-data.js - change url: "<http://192.168.1.17:5000/detect>" to YourLocalIP where the Python Decision engine is running in step 6.

9- Start the httpd server - # sudo service httpd start

10- Navigate to your apache httpd server web application page at <http://yourlocalip:ApachePort/>

Now you can test your web application by entering patient's ID in “**Enter the patient id:** ” field and hit submit. The app will display the probability of the patient having Zika infection (current threshold is 60%)

To change probability threshold- go to ~/Application/guideline/ directory and modify file get-data.js --- if (prob > 60) , change accordingly.

For the ease of use, we have deployed the application at <http://cdc-zika.ddns.net:9090/>
The application deployed is accessible over the internet

Support- Please send an email to dinesh.singh@gatech.edu , yshao63@gatech.edu , rspv6@gatech.edu , John.Stroud@gatech.edu if you see any issues accessing the deployed webpage.