**FluTrackTweet Restful Service by Annie Wang**

The following restful web service will retrieve 10 tweets that relate to flu (along with some metadata about that tweet) with some information about the coming weather. Specifically, the weather information should be the average minimum and maximum temperature for the next five days at the location at which the tweet was sent.

<http://localhost:8080/getFluTrackTweetsWithWeather?s=FeverORFlu>

<http://localhost:8080/getFluTrackTweetsWithWeatherMThreads?s=FeverORFlu>

**Tools Used**

1. Spring boot, embedded tomcat and Maven are used to build/service this restful web service.
2. Eclipse 4.7.2.
3. Spring RestTemplate is used to consume the third parties web service API.
4. Three APIs from third parties to provide data feed.

<http://api.flutrack.org/?s=feverORflu> to retrieve 10 FluTrackTweets required. The number of tweets returned is configured via properties file.

<https://www.metaweather.com/api/location/search/?lattlong>= to retrieve the locations near the coordinates where the tweet is sent, then nearest location including woeid is identified by using distance from the coordinates.

<https://www.metaweather.com/api/location/woeid/> to retrieve MetaWeather info with ConsolidateWeather for 5 days forecast so that average min temp and max temp can be calculated.

**Java Classes**

App.java: Spring application bootstrap class.

FluTrackTweetController.java: the controller to serve restful web services.

POJOs : FluTrackTweet.java, Location.java ,MetaWeather.java , ConsolidateWeather.java.

**Performance Consideration**

1. For each of the 10 tweets returned, two separated calls: getlocaionsbyLattlong and getMetaWeatherbyWoied are needed to calculate average min and max temp. <https://www.metaweather.com/api/location/woeid/> API direct call from web browser takes more than 1 second. The total time to serve 10 tweets with Average min and max temp will cost more 10 seconds.
2. Another version of web service was implemented using ExecutorService for handling ten concurrent web queries. This improves web service performance significantly due to the fact that the time consuming part is IO bound.
3. Further improvement can be made to handle external website outages.

**Build, Deploy and Test**

Unzip flutracktweet.zip and import it into Eclipse.

Run POM.xml to build flutracktweet-1.0.jar.

Run java -jar target/flutracktweet-1.0.jar to boot the app.

Run any of Restful services below after tomcat is up.

<http://localhost:8080/getFluTrackTweetsWithWeather?s=FeverORFlu>

<http://localhost:8080/getFluTrackTweetsWithWeatherMThreads?s=FeverORFlu>