FluTracker Documentation

Sarang Joshi (pid: sarang87)

Vivek Akupatni (pid: vivekb88)

Jonathon Hellman (pid: jonatho7)

# **Application Overview**

# This is a web application for tracking Flu activity in the United States. This system is mostly useful for epidemiologists and the general public for entering their flu predictions based on domain knowledge. In order to make the predictions a user needs to sign up first. Once signed up the user can enter their predictions and finally, after CDC (Centers for Disease Control and Prevention) publishes actual data based on ground surveillance reports, the user predictions are evaluated in terms of accuracy. Each user is given points based on the accuracy of his estimates and points get accumulated over time to give the user a rank. In addition while making predictions users can also refer to previous predictions made by him as well as published data from various flu tracking systems like Google Flu trends, Health Map, and other followed users. Each user is given points based on the accuracy of his estimates and points get accumulated over time to give the user a rank. The website acts as a single place to download various data sets. The overall objective of the application is to increase flu awareness in the society and help data analysts and epidemiologists in better modeling and predicting flu activity. Also, this web application provides relevant information to casual users too.

The latest update to Flu Tracker (Project 4) adds a social layer to the site. Users will now be able to follow/unfollow other users so that they can view other users’ predictions and recent activities. Users also now have access to a profile page and can post comments on predictions. To encourage positive social norms on the site, such as avoiding the use of offensive comments, the site now includes multiple types of user accounts, which will be explained in further detail below.

# **Intended Audience**

# This application is intended for data analysts, epidemiologists, and the general public for making predictions, publishing them and connecting with other domain experts,

# **Actions performed by a registered user:**

# Create a new user account

* Login / logout of a user account
* Add / edit / delete predictions
* View predictions based on region or location
* Comment on predictions
* Follow / unfollow another user
* View / edit user profile
* View other users’ profile

# **Project 4 Requirements Addressed**

# **Multiple Types of User Accounts**

* Registered User – Can perform all actions listed in the “Actions performed by a registered user” section
* Moderator – Can perform all the actions a registered user can, and can also edit and delete user comments
* Admin – Can perform all the actions a moderator can, and can also delete a user account.

Details:

* Site includes a page for changing other users’ roles.
* Moderator and admin have a greater access level than a registered user and can view/edit a registered users’ dashboard page as if they were logged in as this user.

# **Profile Page for Users**

Fields: username (cannot be changed), first and last name, email address, password, and drop-down menu for email visibility. Form validation has been implemented.

**Activity Feeds**

Five unique user actions: change name, follow user, new prediction, modify prediction, delete prediction

**Add/remove friends OR Post comments on items**

Implemented both of these. Friends were implemented using the uni-directional (follow/unfollow) Twitter model, and comments could be added to predictions or activity feed activities.

**Crowdsourcing for manipulating data OR Crowdsourcing for user testing**

Used Amazon Mechanical Turk to gather feedback from about 10 crowd workers. The task for a worker was to select one of two curves which more closely resembled the actual data curve. The three curves were shown on a graph and were displayed in different colors. A worker needed to select the more similar curve for 6 iterations, and could enter comments about their selection. The task was expected to take 5 – 10 minutes per worker, and workers were paid $1.00 for completing the task. The worker qualifications were that a worker must reside in the U.S., have a 90% HIT completion rate, and must have completed at least 50 HITs.

Our implementation of the AMT task included hosting the flu tracker site on an Amazon EC2 instance, and using XAMPP in order to run our PHP server and MySQL database. When a worker completed a task, the data was saved directly to our MySQL database for us to view.

The results of the Amazon Mechanical Turk study were…(Explain in your summary document what you learned from this testing and how your design changed or didn't.)

#### **Running the project on your system:**

* Startup Apache and MySQL in the XAMPP control panel.
* Create a new database named “flu” using the phpMyAdmin interface provided in XAMPP.
* Import the flu.sql file into the flu database through phpMyAdmin. This will populate the database with the relevant records.
* Configure the SERVER\_PATH in the config.php file depending on the location of the application in your system. This file is located in xampp/htdocs/code/app. The SERVER\_PATH is used to reference all the public files. (For Mac users the application is in htdocs folder in xampp.).If set incorrectly all CSS and JavaScript would not be applied to the pages.
* Open a web browser and navigate to <http://localhost/flutracker> in order to explore the Flu Tracker web application.
* Login with username as vivekb88 and password as vivekb88 or you can sign up for a new account and use it to login to the application.