Software Engineering I Group II

Health Monitoring Analytics 2014

D. Yao, W. Fang, W. Zhang, Y. Sun, Y. Wu, Z. Zheng

https://sites.google.com/site/healthmonitoringanalytics2014 Nov.10, 2014

Individual Contributions Breakdown

“All team members contributed equally”

Table of Contents

1. Interaction Diagrams

1.1. Use Case 5: Data Collecting and Classify

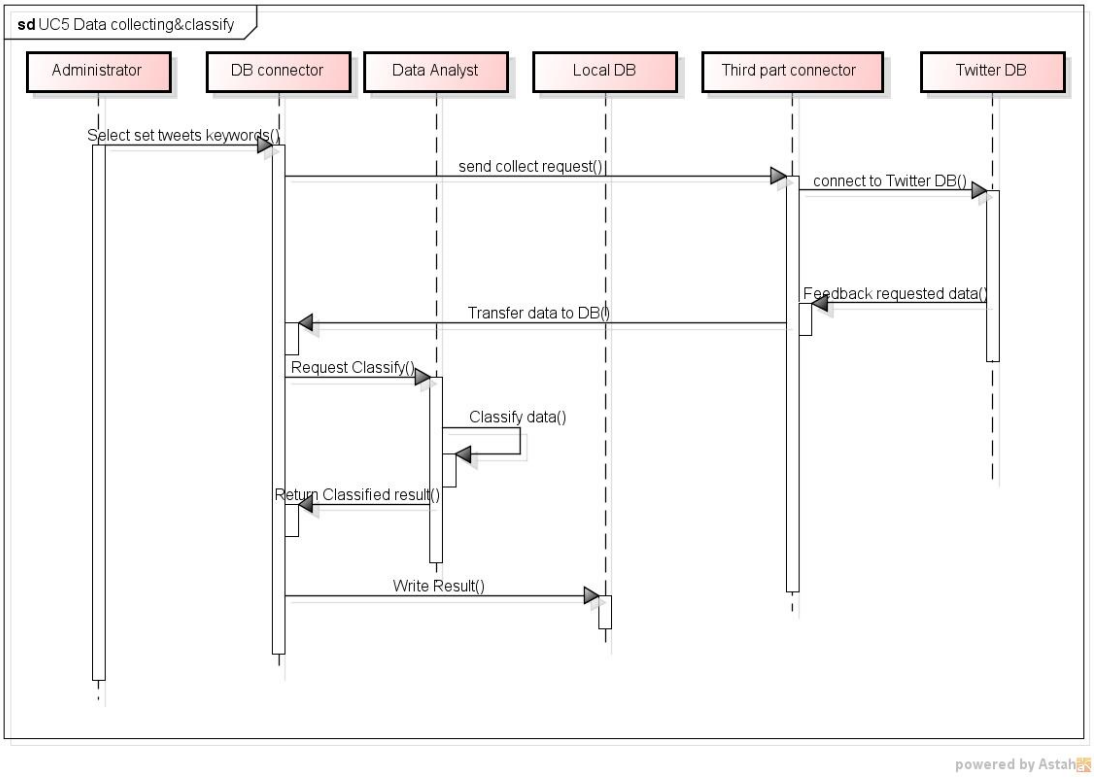

Figure 1-1 Use Case 5 Data: Collecting and Classify

The above interaction diagram is for the Use Case 5 Data Collecting & Classify. Firstly, the administrator connects the DB connector to select set tweets keywords, with using the function of Select set tweets keywords(). Secondly, the system sets up a connection with the DB connector to send collect request to the third party connector. Then the third part connector sends back the data to the DB connector. The DB connector will send data with classify request to the data analyst, and the data analyst will classify the data and send back the result to the DB connector. Finally the DB connector writes the result into the local database.

1.2. Use Case 17: Personal Suggestions

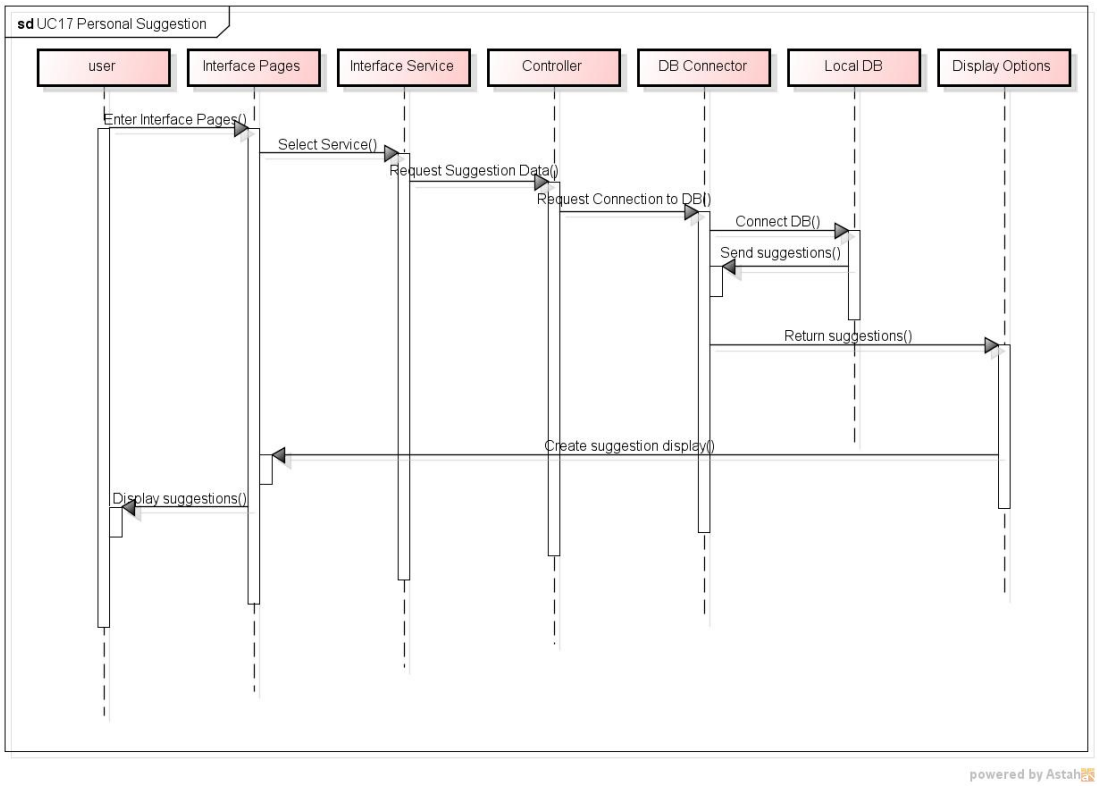

Figure 1-2 Use Case 17: Personal Suggestions

The function of UC17 is basically displaying the personal suggestions for the specific user. First, the login user clicks the 'get personal suggestions' button or enter related page on Interface Pages. After, the interface Pages selects related services existing in Interface Service. Then the Interface Service emits the request to Controller for getting Suggestion Data. Since the controller could not get touch with the database directly, it should send the request to the DB Connector for connection to the local DB. After Local DB returns the related information to the DB Connector. The specific data for user is sent to the Display Options for proper ways to display. Correlate display ways for suggestions are created by Display Options and should be posted to Interface Pages. At last the Interface Pages display the specific suggestions for user on the screen.

1.3. Use Case 20: Data Analysis

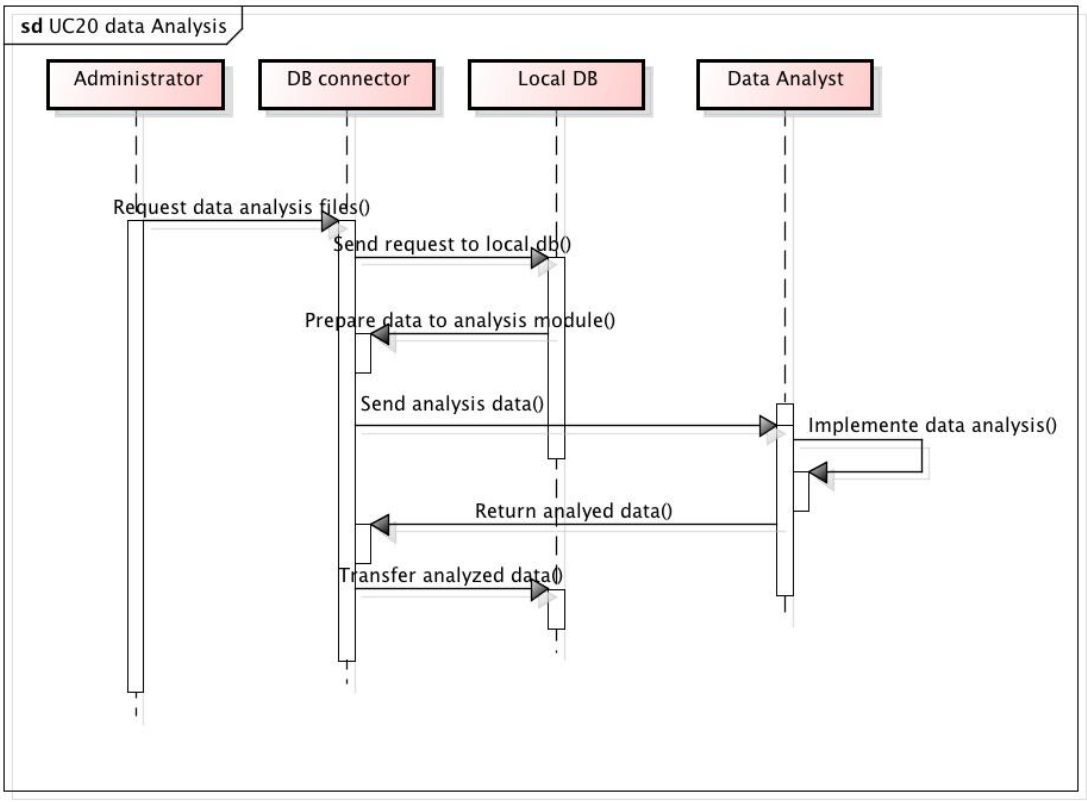

Figure 1-3 Use Case 20 Data analysis

The UC 20 is used to analyze data that collected from Twitter. First the Administrator calls the database connector for specific request. Next the database connector will send the request to the database and acquire specific data. Then, the database connector sends the acquired data to the data analyst, where the data will be analyzed and send back the result to the database connector. At the end, the database connector will write the result to the database.

2. Class Diagram and Interface Specification

2.1. Class Diagram

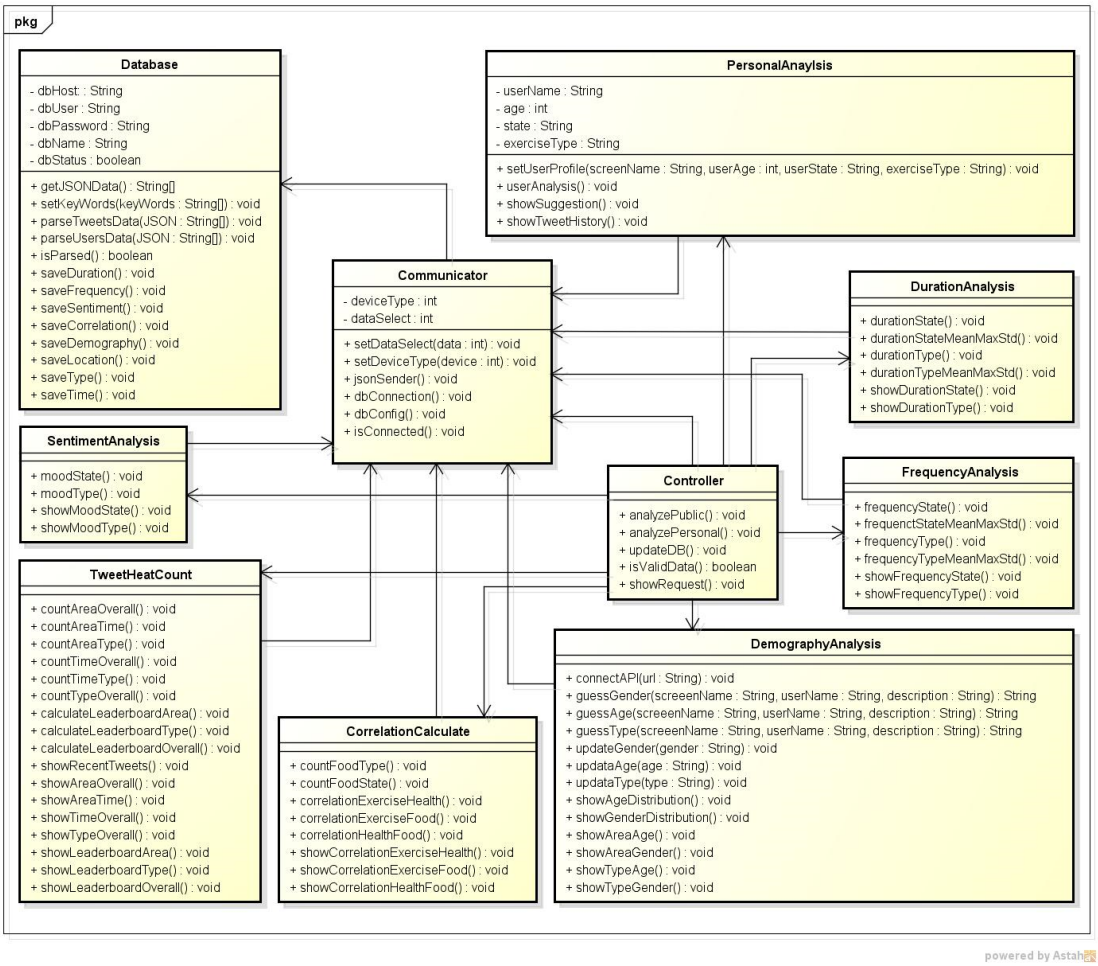

Figure 2-1. Class diagram

Above is the class diagram of the health monitoring system. It is constructed by total ten classes. Firstly, the methods within the database class would be called early because the system needed to retrieved tweets data in the first time. The database class mainly takes the responsibility of retrieving the data from twitter, parse the data in JSON format, manage the raw data and store the data after analysis.

Communicator class focuses on the communication between the showing function called by the user interface and the database. It could identify what device is requesting the data and what data is being requesting by the variables deviceType and dataSelect.

Controller deals with the issues occur in user's action or the system itself. In general saying, it is used to automatically run the analysis functions derived from all the classes related to analysis. Besides, it would handle the showing request sending by the users.

PersonalAnaylsis class is separated from other public analysis classes here. It is different from other analysis since user is asked to input several value like their screen name in twitter, their actual age and favorite exercise type, etc. The methods within this class would do the analysis about personal information and show the corresponding information such as the suggestion and user's health related tweet history.

Other classes are all about public analysis including the DurationAnalysis, FrequencyAnalysis, etc. They are some specific classes that would do the corresponding analysis about public health data and provide respectively functions to show the useful information we derived from the data after statistics and analysis. Following sections would give more detail about those classes.

2.2. Data Types and Operation Signatures

2.2.1. Database

The "Database" class contains the following five variables that stores the basic information of the database used for the connection.

dbHost String variable storing the host of the database.

dbUser String variable storing the username of the database.

dbPassword String variable storing the password corresponding to the username.

dbName String variable storing the database name.

dbStatus Boolean variable identifying the status of the database, open or closed.

The functions of this class listed below include several get and parse functions that are used to obtain data from Twitter, and several save functions that are used for storing analytical data into database.

getJSONData() This function is used for obtaining raw tweets data from Twitter Streaming API and returns an array of JSON value, each element represents one piece of tweet data.

setKeyWords(keyWords : String[]) This function is used for setting the keywords that are needed for filtering the tweets. All keywords are related to exercise and health.

parseTweetsData(json : String[]) This function is used for parsing the collected JSON data containing tweet information into readable tweet data and store them in the tweets table of the database.

parseUsersData(json : String[]) This function is used for parsing the collected JSON data containing user information into readable user data and storing them in the users table of our database.

isParse() This function is used for recording the parse result of each piece of JSON data. If it has been parsed, return true. Else, return false.

saveDuration() This function is used for saving the analytical data related to the exercise duration time into new feature tables of the database.

saveFrequency() This function is used for saving the analytical data related to the keyword frequencies into new feature tables of the database.

saveSentiment() This function is used for saving the analytical data related to the sentiment computing into new feature tables of the database.

saveCorrelation() This function is used for saving the analytical data related to the correlation computing into new feature tables of the database.

saveDemography() This function is used for saving the analytical data related to the demography information into new feature tables of the database.

saveLocation() This function is used for saving the analytical data related to different locations into new feature tables of the database.

saveType() This function is used for saving the analytical data related to the exercise types into new feature tables of the database.

saveTime() This function is used for saving the analytical data related to different time of the day into new feature tables of the database.

2.2.2. TweetHeat

The "TweetHeat" class contains several count functions that are used to count the number of tweets concerning different categories, several calculate functions to sort the count result to make the leaderboard, and also some show functions to display all the

responding charts onto the user interface.

countAreaOverall() This function is used for counting the number of tweets in different areas.

countAreaTime() This function is used for counting the number of occurrences of different areas in different time periods.

countAreaType() This function is used for counting the number of occurrences of different exercise types in different areas.

countTimeOverall() This function is used for counting the number of occurrences of different time periods.

countTimeType() This function is used for counting the number of occurrences of different time periods corresponding to different exercise types.

countTypeOverall() This function is used for counting the number of occurrences of different exercise types.

calculateLeaderboardArea() This function is used for sorting the count result of different areas and storing them in decreasing order.

calculateLeaderboardType() This function is used for sorting the count result of different exercise types and storing them in decreasing order.

calculateLeaderboardOverall() This function is used for sorting the count result of all tweets concerning exercise and health tweeted by each user and storing them in decreasing order.

showRecentTweets() This function is used for showing on map the most recently posted tweet and its user.

showAreaOverall() This function is used for displaying the analytical chart showing the number of tweets in different states.

showAreaTime() This function is used for displaying the analytical chart showing the number of tweets in different states in different time periods.

showTimeOverall() This function is used for displaying the analytical chart showing the number of tweets in different time periods.

showTypeOverall() This function is used for displaying the analytical chart showing the number of tweets concerning different exercise types.

showLeaderboardArea() This function is used for displaying the leader board ranked by the number of tweets concerning different areas.

showLeaderboardType() This function is used for displaying the leader board ranked by the number of tweets concerning different exercise types.

showLeaderboardOverall() This function is used for displaying the leader board ranked by the number of tweets concerning exercise and health.

2.2.3. FrequencyAnalysis

The "FrequencyAnalysis" class contains several functions that are used to calculate the frequency of exercise corresponding to different states and types and there are also some methods that could calculate the mean, max and standard deviation of such frequencies. Besides, there are some show functions to display all the responding charts onto the user interface.

frequencyState() This function is used for calculating the frequency of exercise in different states according to user's tweets.

frequencyStateMeanMaxStd() This function is used for counting the mean, maximum and standard deviation of those frequency calculated in fregquencyState().

frequencyType() This function is used for calculating the frequency of exercise in different exercise types according to user's tweets.

frequencyTypeMeanMaxStd() This function is used for counting the mean, maximum and standard deviation of those frequency calculated in fregquencyType().

showFrequencyState() This function is used for displaying the analytical chart that would show the frequency values of exercise corresponding to different states.

showFrequencyType() This function is used for displaying the analytical chart that would show the frequency values of exercise corresponding to different exercise types.

2.2.4. DurationAnalysis

The "DurationAnalysis" class contains several functions that are used to calculate the duration of exercise corresponding to different states and types and there are also some methods that could calculate the mean, max and standard deviation of such durations. Besides, there are some show functions to display all the responding charts onto the user interface.

durationState() This function is used for calculating the Duration of exercise in different states according to user's tweets.

durationStateMeanMaxStd() This function is used for counting the mean, maximum and standard deviation of those duration calculated in durationState().

durationType() This function is used for calculating the duration of exercise in different exercise types according to user's tweets.

durationTypeMeanMaxStd() This function is used for counting the mean, maximum and standard deviation of those duration calculated in durationType().

showDurationState() This function is used for displaying the analytical chart that would show the duration values of exercise corresponding to different states.

showDurationType() This function is used for displaying the analytical chart that would show the duration values of exercise corresponding to different exercise types.

2.2.5. SentimentAnalysis

The "SentimentAnalysis" class contains several functions that are used to calculate the sentiment value of the tweet text corresponding to different states and types. Besides, there are some show functions to display all the related charts on the user interface.

moodState() This function is used for calculating the mood value of tweets in different states according to user's tweet text.

moodType() This function is used for calculating the mood value of tweets for different exercise types according to user's tweet text.

showMoodState() This function is used for displaying the state map that would show the happiness degree corresponding to different states.

showMoodType() This function is used for displaying the analytical chart that would show the mood values corresponding to different exercise types.

2.2.6. CorrelationCalculate

The "CorrelationCalculate" class contains several functions that are used to count the number of the tweets that metioned food corresponding to different states and exercise types. There are also several functions that would calculate the linear regression value among the tweets count of health, exercise and food. Besides, there

are some show functions to display all the related charts on the user interface.

countFoodType() This function is used for counting the number of tweets related to different kinds of food according to multiple types of exercise.

countFoodState() This function is used for counting the number of tweets related to different kinds of food according to different states.

correlationExerciseHealth() This function is used for calculating the linear regression value between the counts of tweets related to exercise and health.

correlationExerciseFood () This function is used for calculating the linear regression value between the counts of tweets related to exercise and food.

correlationHealthFood () This function is used for calculating the linear regression value between the counts of tweets related to health and food.

showCorrelationExerciseHealth() This function is used for showing two line charts that would indicate the count number of tweets related to exercise and health separately.

showCorrelationExerciseFood() This function is used for showing two line charts that would indicate the count number of tweets related to exercise and food separately.

showCorrelationFoodHealth() This function is used for showing two line charts that would indicate the count number of tweets related to food and health separately.

2.2.7. PersonalAnalysis

The "PersonalAnalysis" class contains the following four variables that store the user's personal information.

username String variable storing the user's name.

age Integer variable storing the user's age.

state String variable storing the user's current location.

exerciseType String variable storing the user's favorite exercise type.

This class include functions that are used to give out the personal suggestions based on the information the users provide.

setUserProfile(screenName : String, userAge : int, userState : String, exerciseType :

String) This function is used for setting the basic information that is needed for the analysis.

userAnalysis() This function is used for making analysis on the proper healthy food and exercise duration time based on the information the users provide.

showSuggestion() This function is used for displaying the proper personal suggestions based on the analysis.

showTweetHistory() This function is used for displaying the past tweets the user posted and showing the time when the tweet has been created.

2.2.8. DemographyAnalysis

The "DemographyAnalysis" class has some guess functions that use third party's API to make guesses on the demographical information of the Twitter users based on their description, some update functions that update the guess results into the existing table, and some show functions that display the analytical results.

connectAPI(url : String) This function is used for connecting our database to the third party's API. The parameter "url" is the website of this API.

guessGender(screenName : String, username : String, description : String) This function is used for making a guess on the gender of a certain user based on his/her screen name, user name and the user description.

guessAge(screenName : String, username : String, description : String) This function is used for making a guess on the age of a certain user based on his/her screen name, user name and the user description.

guessType(screenName : String, username : String, description : String) This function is used for making a guess on the user type, organization or person, of a certain user based on his/her screen name, user name and the user description.

updateGender(gender : String) This function is used for updating the guess result of the user gender in the existing users table.

updateAge(gender : String) This function is used for updating the guess result of the user age in the existing users table.

updateType(gender : String) This function is used for updating the guess result of the user type in the existing users table.

showAgeDistribution() This function is used for displaying the age distribution among all the users who have posted exercise-related tweets.

showGenderDistribution() This function is used for displaying the gender distribution among all the users who have posted exercise-related tweets.

showAreaAge() This function is used for displaying the age distribution among users who have posted exercise-related tweets in different states.

showAreaGender() This function is used for displaying the gender distribution among users who have posted exercise-related tweets in different states.

showTypeAge() This function is used for displaying the distribution of different exercise types in different age groups.

showTypeGender() This function is used for displaying the distribution of different exercise types in different gender groups.

2.2.9. Controller

The "Controller" class works as an essential part in the system-to-be. All the executions of its functions are linked to some other classes. The user activates these functions and gets the analytical results corresponding to his/her choice.

analyzePublic() This function is used for making analysis based on public information we obtained from the Twitter.

analyzePersonal() This function is used for making analysis based on personal information we obtained form the user.

updateDB() This function is used for making updates on the existing database if some changes need to be made.

isValidData() This function is used for identifying whether a certain piece of data is valid. If valid, return true, else, return false.

showRequest() This function is used for displaying corresponding analytical charts or tables based on the request from the user.

2.2.10. Communicator

The "Communicator" class is linked to all the other classes that need to get data

from the database. It has two variables storing the type of the platform and which feature table we need to use.

deviceType Integer variable storing the type of the platform the JSON dada need to be sent to. We choose 0 representing the web platform, and 1 representing the IOS platform.

dataSelect Integer variable storing which feature tables we need to use for analysis. Different numbers represent different feature tables.

This class contains a jsonSender() function which is essential to send the database data to other platforms in JSON formation. Other functions are used for connecting the database.

setDataSelect(data : int) This function is used for setting which feature table we need to use for our analysis.

setDeviceType(device : int) This function is used for setting the type of platform we use to display the analytical results.

jsonSender() This function is used for packaging the data we need in certain tables and sending them to the platform we choose.

dbConnection() This function is used for making connection with the database.

dbConfig() This function is used for initializing the database.

isConnected() This function is used for testing the connection to the database. If connected, return true. Otherwise, return false.

2.3. Traceability Matrix

Domain Concepts	Software Classes									
	Controller	Communicator	Database	PersonalAnalysis	DemographyAnalysis	DurationAnalysis	FrequencyAnalysis	TweetHeatCount	CorrelationCalculate	SentimentAnalysis
Controller	X									
Profile Storage				X	X					
Interface Services				X	X	X	X	X	X	X
Interface Pages				X	X	X	X	X	X	X
Display Options				X	X	X	X	X	X	X
Treated Information			X							
DB Connector		X	X							
Data Analyst		X	X	X	X	X	X	X	X	X
Third Part Connector			X	X	X	X	X	X	X	X

Table 2-1 Traceability matrix

Apparently, there are many classes evolved from the same Domain Concept except the Controller, which is basically achieved by the single class that also named Controller.

Concept Profile Storage include user's personal data (like demography information). These information could derived from Classes PersonalAnalysis and DemographyAnalysis since both of the classes contain related methods to infer and output such message.

Interface Services, Interface Pages and Display Options construct the user interface part of the system, hence the last seven classes derived from them all.

Because the treated information is basically stored in the database of our system, there are not many classes involved with this concept. Only the class Database is needed to get such information.

Both of the two classes --- Communicator and Database --- take the responsibility of Concept DB Connector. Due to some specific reasons, we separate them by different functions. Database takes charges of the data management within the database. Communicator transfers the data between other classes and the database.

Data Analyst is separated into seven classes that related to specific types' analysis and two classes that enable them to do communication with database.

At last, because it is hard to isolate the third part connector in codes, it is involved in many of the classes that would show the information in the user interface and the class that retrieves data from twitter.

3. System Architecture and System Design

3.1. Architectural Style

"Architectural styles are reusable packages of design decisions and constraints that are applied to an architecture to induce chosen desirable qualities". Our system scrawls and stores data from Twitter database, analyzes the data for different business goals (features), finally presents the clear results to users. The users should be able to access our system to find what they want, but the reverse is not allowed. Based on these system characteristics, our system would be better to take the advantage of client-server architectural style instead of peer-to-peer. The client-server architecture is shown as the Figure 3-1.

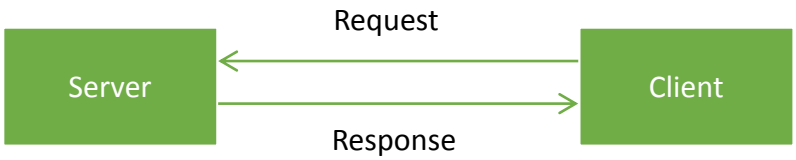

Figure 3-1. Client-Server sketch

Besides, because of the three steps of our system talked above, our system is suit for three-tier client-server architecture – data tier, logic tier and presentation tier. The sketch for three-tier architecture is shown as the Figure 3-2.

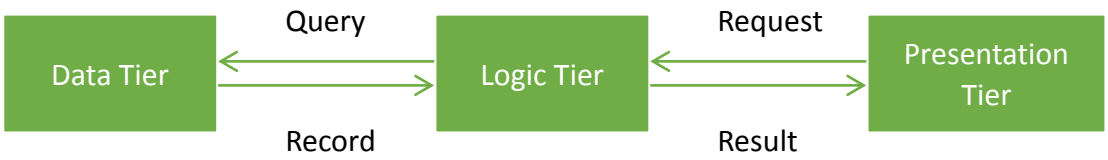

Figure 3-2. Three-Tier architecture sketch

3.2. Identifying Subsystems

As shown in the Figure 3-3, there are six main subsystems in our system – collection, store, analytics, transmission, presentation and plot. There are also dependencies among them shown as the arrows in the Figure 3-3. For example, the

package presentation points to the package plot that means the presentation uses some elements in the plot.

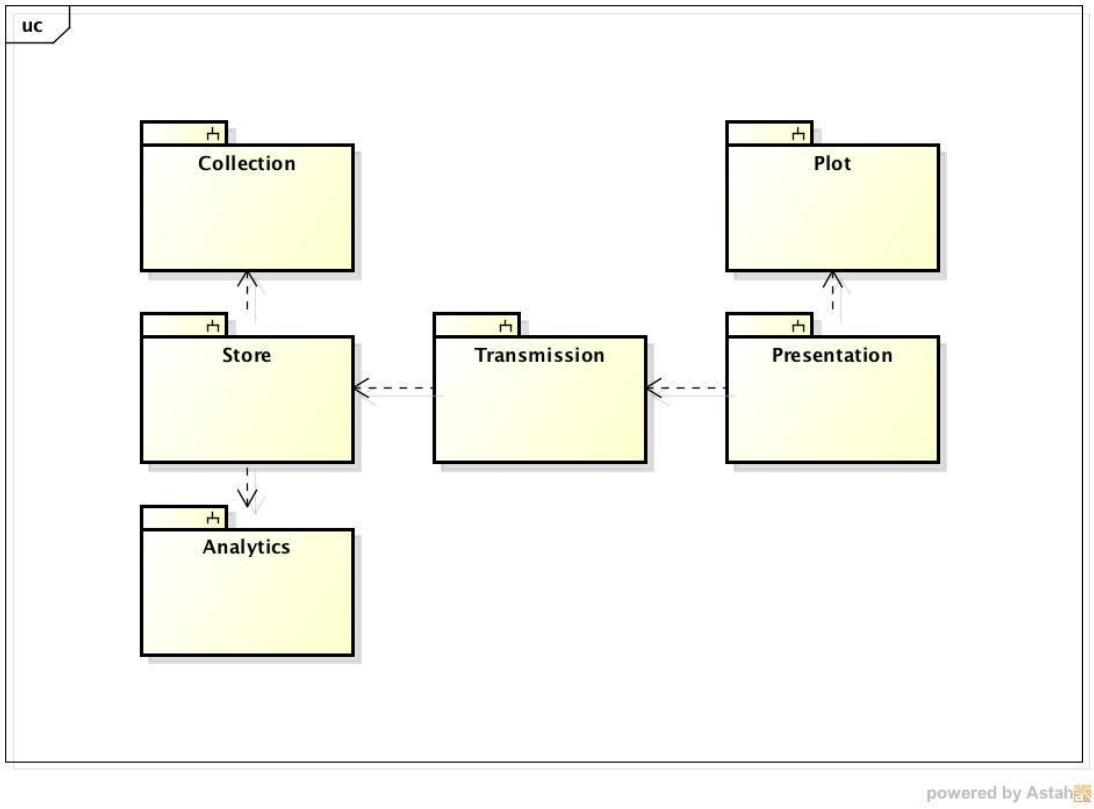

Figure 3-3. Subsystems sketch

The collection subsystem is responsible for scrawling data from Twitter database to our database. The data will be in a JSON format and stored directly in the json_cache table in our database. The store subsystem will parse (extract) the properties such as screen name, date, tweet from JSON data in the json_cache table and insert the properties into other tables. The analytics subsystem will compute results from the data stored in the tables mentioned above in order to fulfill the different business goals, and insert the results in feature tables. The presentation subsystem simply structures the user interface and listens to the events triggered by users. The subsystem will call the functions in the plot subsystem to draw plots like bar chart, pie chart, line chart and maps. Finally, the transmission subsystem is the bridge between the front-end (client) and the rear-end (server). We still use JSON format for transmission. The presentation will not get all the feature data at the initialization stage since data is large and loading is slow. Instead, the presentation will tell the transmission to send which feature table's records by sending an offset. Then the transmission will get the feature table's records from store and send back to the presentation in JSON format.

3.3. Mapping Subsystems to Hardware

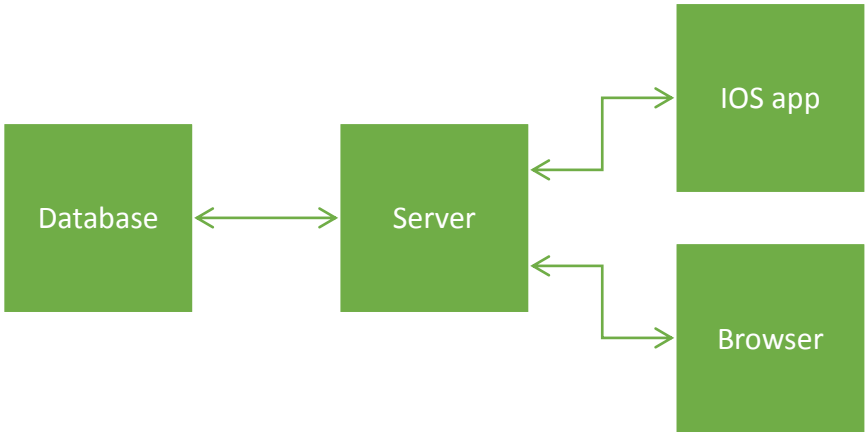

Figure 3-4. Hardware map

In the Figure 3-4, database and server could be distributed into two computers or into one computer. IOS app could be operated on iPhone or iPad, while browser could be operated on any platform as long as it supports the browser, e.g., computer, smart phone.

What is the relationship between the subsystems and the hardware? The collection, store, analytics and transmission subsystems will be deployed on the server. The computer for database only stores the data, and will be accessed by the server. If it is a web application, the presentation and plot subsystems will be also deployed on the computer for server. The user interface structure and the data for plot will be downloaded by the browser. If it is an IOS application, the presentation and plot subsystems will be deployed on the IOS device. It means only the data for plot will be transmit from the server to the IOS device.

3.4. Persistent Data Storage

Our system uses the rational database MySQL to store the data. With the database, the application will maintain the data for the next running. The tables storing Twitter JSON data and the data after being parsed are shown as the Figure 3-5.

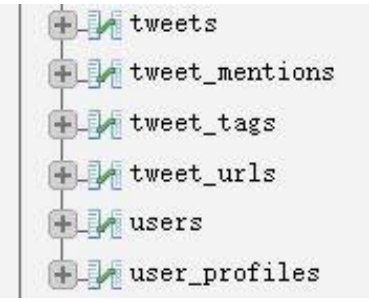

Figure 3-5. Twitter tables

The feature tables are shown as the Figure 3-6.

Figure 3-6. Feature tables

3.5. Network Protocol

In our system, the communication between our server and the Twitter server is based on the GET method in HTTP protocol. Thus, our server should be linked to the Internet. The communication between our server and the browser is also based on the GET method in HTTP protocol because of the jQuery. But the communication between our server and the IOS device is based on the POST method in HTTP protocol. POST is much securer than GET since the data will be appended to the address when using GET. For these two communications, the server, the browser and the IOS device are in the same local area network (LAN) because currently our server is not accessible by the Internet – no domain name. The network is shown in the Figure 3-7.

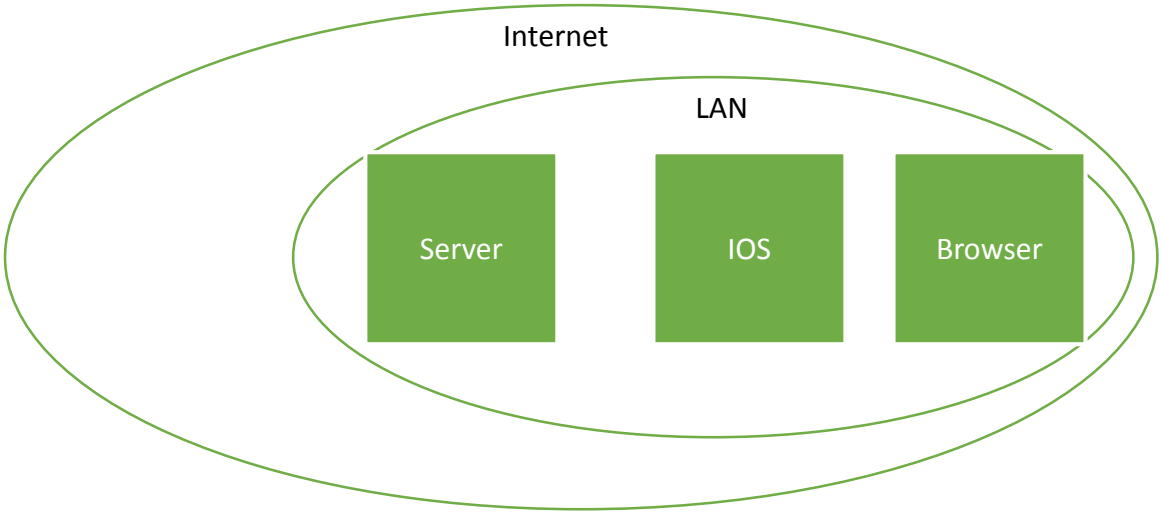

Figure 3-7. Network

3.6. Global Control Flow & Hardware Requirements

Our system is an event-driven system. Users will interact with the user interface elements, and trigger the events and get the responses. Our system uses a timer to reload the data from feature tables to make our system real-time.

Now our database has 700+ thousand tweets, and it needs 5 GB to store the data. At least, it needs a computer to be the database, the server and the browser and an IOS device to run the IOS app.

4. Algorithm and Data Structure

4.1. Algorithms analysis

4.1.1. Improve data reliability using weight index

When we set key words and collect data, there is unavoidable noise in the result, some tweet text may reflect our searching intention, some may have no relationship with the health topic. For example, if we use run as key words, the result tweet text may contain runny or rune. Thus when we do data analyze, we need to eliminate irrelevant tweet text and keep useful information. The methods we use is putting weight index to each kind of sport. And the methods will implement in emotion analyze and heat map analyze feature.

1. Weight index math model description

The basic weight index equation shows below:

$$\textit{key}\text{ word weight index} = \frac{\textit{each}\text{ field veritable tweet text}}{\text{each field total tweet text}} \qquad 4\text{-}1$$

If weight index is greater than 80%, we define the keyword as reliable, otherwise we define the keyword as unreliable.

However, because the database is too large to test them all, so we use systematic sample method to get limited tweet text for each key words. The above equation will change to

$$\textit{key}\text{ word weight index} = \frac{\textit{each}\text{ field }\textit{sample}\text{ veritable tweet text}}{\text{each field }\textit{sample}\text{ total tweet text}} \qquad 4\text{-}2$$

2. Implementation

a) We implements this method in emotion analyze. First we set a set of key words which can represent a kind of emotion, then we use equation 4-2 to analyze each key word's reliability, if it is unreliable, we eliminate or replace it with another key word.

b) We implements this method in heat map. For each kind of exercise, we use its key word weight index multiple area's total tweet text, and use the result as total number of people in a area who actually doing exercise.

4.1.2. Personal suggestion based on vector space model (VSM)

Personal suggestion needs to find relationship between the user's information and our existing data and draw a conclusion. To fulfill this function, we use the VSM methods. The steps describe below:

a） With the help of GATE, both tweet texts(document) in our database and login user's information(query) are able to represented as vectors[1]

$$\vec{d}_i = (\omega_{i,1}, \omega_{i,2}, ..., \omega_{i,3}) \qquad 4\text{-}3$$

Each dimension represents a weight for term j in tweet text i, if the term showed up in the tweet text, then its value is none-zero.

b) Then we calculate the similarity of a document vector to a query vector using the cosine of the angle between them:

$$sim(d,q) = \cos\theta = \frac{d_i \cdot q}{|d_i||q|} = \frac{\sum_j \omega_{i,j} \times \omega_{q,j}}{\sqrt{\sum_j \omega^2_{i,j}} \sqrt{\sum_j \omega^2_{q,j}}} \qquad 4\text{-}4$$

particular, sim(d,q)=1 when d=q;

sim(d,q)=0 when d and q share no terms

From 4-4, we can define the similarity to the tweet text and user's information, then predict what kind of sports they will like based on other user's tweet text.

4.1.3. Sports correlation analyze based on linear regression

To analyze the relationship between two sports, we use linear regression methods to draw a line to show their correlation degree.

1. Linear regression math model description.

To implement linear regression analysis, we use generalized least squares methods (GLS)[2]. The roughly description shows below:

According to wiki [3], The GLS is applied when the variances of the observations are unequal, or when there is a certain degree of correlation between the observations. In our project, we assume the linear is:

$$\varphi(x) = a + bx \tag{4-5}$$

According to the GLS theory, the solution equation will be:

$$\begin{bmatrix} \sum_{i=1}^{m} 1 & \sum_{i=1}^{m} x_i \\ \sum_{i=1}^{m} x_i & \sum_{i=1}^{m} x^2{}_i \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{m} y_i \\ \sum_{i=1}^{m} x_i y_i \end{bmatrix} \tag{4-6}$$

Within 4-6, m is the total number of point we use. From the solution equation, we can get a and b, then we put them back into 4-5 and get the linear equation.

To improve the accuracy, we introduce the Pearson product-moment correlation coefficient. According to the wiki [3], Pearson product-moment correlation coefficient is a measure of the linear correlation (dependence) between two variables X and Y. The mathematical equation is

$$r_{xy} = \frac{\sum_{i=1}^{m}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{m}(x_i - \bar{x})^2 \cdot \sum_{i=1}^{m}(y_i - \bar{y})^2}} \tag{4-7}$$

The r_{xy} range from -1 to 1, we define $0.8 \le |r_{xy}| \le 1$ as highly correlated, $|r_{xy}| \le 0.3$ as no correlated.

2. Implementation

a) First we divide one day in to 24 hours and choose two sports that we want to compare, then we count how many related tweet text for each sport is showed in each hour. An example is showed as Figure 4-2.

b) Using equation 4-7 to analyze the data correlation degree, if they are highly correlated we use the generalized least squares methods to draw the linear equation in the coordinate and use the slope to judge if they are positive correlated or negative correlated.

	swimming	running
1h	500	1000
2h	250	600
3h	1500	2800
…	…	…

Figure 4-1. Example.

4.1.4. Sentiment analysis based on probability density function of a normal distribution

This method is derived from the project website of Visualizing Twitter Sentiment [4].

For sentiment analysis, the ANEW [5] dictionary provides measures of valence, arousal, and dominance for 1,034 English words. Each word is rated on a nine-point scale ranging from 1 to 9.

Ratings for a common word are combined into a mean rating and a standard deviation of the ratings for each dimension.

For example, for the word house, ANEW reports:

house,

$$v = (\mu: 7.26, \sigma: 1.72),\ a = (\mu: 4.56, \sigma: 2.41),\ d = (\mu: 6.08, \sigma: 2.12),\ f_q = 591 \quad 4\text{-}8$$

This shows that house has a mean valence v of 7.26 and a standard deviation of 1.72, a mean arousal a of 4.56 and a standard deviation of 2.41, a mean dominance d of 6.08 and a standard deviation of 2.12, and a frequency f_q of 591 ratings.

However if multiple words documented in ANEW dictionary for instance:

Congrats to @HCP_Nevada on their health care headliner win!

ANEW's measure of the n = 2 words' means and standard deviations of valence and arousal are:

health,

$$v = (\mu: 6.81,\ \sigma: 1.88),\ a = (\mu: 5.13,\ \sigma: 2.35),\ f_q = 105 \qquad 4\text{-}9$$

win,

$$v = (\mu: 8.38, \sigma: 0.92),\ a = (\mu: 7.72, \sigma: 2.16],\ f_q = 55 \qquad 4\text{-}10$$

To combine the means for health and win, we assume that the individual ratings reported for each word form a normal distribution. Intuitively, if a word has a higher standard deviation, for example, a higher $\sigma_{v,i}$ for valence, the valence ratings for the word were spread across a wider range of values. If $\sigma_{v,i}$ were lower, ratings for the word clustered closer to the mean. Based on this, we use the probability density function[6] of a normal distribution to estimate the probability of the word's rating falling exactly at the mean.

Notice that if we'd simply used an arithmetic mean to compute the overall mean valence M_v, we would have reported $M_v = (6.81 + 8.38)/2 = 7.56$. However, the standard deviation of valence for health ($\sigma_{v,1}$ = 1.88) is higher than the standard deviation for win ($\sigma_{v,2}$ = 0.92). Because of this, we weight win's mean valence $\mu_{v,2} = 8.38$ higher than health's mean valence $\mu_{v,1} = 6.81$. How we allocated the weight is explained below:

The normal distribution is parametrized in terms of the mean and the variance, denoted by μ and σ^2 respectively, giving the family of densities

$$f(x; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \qquad 4-11$$

Hence, the probability of the word health's rating falling exactly at the mean could be calculated by the formula above.

When $x = \mu_{v,1} = 6.81$, $\sigma_{v,1} = 1.88$, we derive $f(x) = \frac{1}{1.88*\sqrt{2\pi}} = 0.212$.

Now we allocate the word 'health' with weight $W_1 = 0.212$.

We could use the same steps to derive the weight of the word win $W_2 = 0.434$

Then we can calculate the overall mean:

$$M_v = (\mu_{v,1} * W_1 + \mu_{v,2} * W_2)/(W_1 + W_2) = 7.86 \qquad 4-12$$

Hence, the result is an overall mean M_v = 7.86 that falls closer to win's mean valence. A similar result can be seen for overall mean arousal M_a.

The probabilities are applied as weights when we sum the means. Using this formula, we compute an overall mean valence and arousal of: M_v = 7.86, M_a = 6.48

For other situations that multiple words documented in ANEW dictionary, we could use the similar way to combine them:

$$M_v = \frac{\mu_1 * W_1 + \mu_2 * W_2 + \cdots + \mu_n * W_n}{W_1 + W_2 + \cdots + W_n} \qquad 4-13$$

In Fig 4-2 it shows the result by using the above mathematical method. Mood value has been calculated for corresponding distinct exercise types.

type	mood_value
running	6.08273
cycling	5.76242
swimming	6.07984
basketball	5.96129
volleyball	6.52989
tennis	6.15888
football	5.56301

Fig 4-2. Exercise types and their mood value

4.2. Data Structures

4.2.1. Set Filter keywords

Before embarking on crawling raw data from twitter database keywords libraries has been set by carefully analysis and comparison. First seven popular exercising types involving 'running', 'cycling', 'swimming', 'basketball', 'volleyball', 'tennis' and 'football' have been decided for keywords of exercise.

However only by using the above exercising keywords solely will seriously impact the accuracy of the searching results, which includes a large amount of unrelated information. For example, through searching the tennis, the following shows the 10 results:

WTA @WTA Nov 8

15 MILLION! @MariaSharapova becomes 1st #tennis player to hit 15 MILLION LIKES on Facebook--> http://wtatenn.is/vOYoQc

SI Tennis @SI_Tennis 15m15 minutes ago

Petra Kvitova wins a thriller to deliver Czech Republic its 3rd Fed Cup title in four years. http://www.si.com/tennis/2014/11/09/petra-kvitova-wins-epic-clinch-fed-cup-title-czechs …

Women Love Sports @Women_Sports 2h2 hours ago

The mark of great sportsmen is not how good they are at their best, but how good they are their worst. -Martina Navratilova (tennis)

Sportupdate_ID @SportUpdate_ID 3h3 hours ago

#SportUpdate - Andy Murray begins ATP World Tour Finals with a straight sets defeat to Kei Nishikori of Japan ... http://dailym.ai/1AMM0OY

SI Tennis @SI_Tennis 3h3 hours ago

Kei Nishikori gets his first ever win over Andy Murray, beats him 64 64. Ugly match all around, but Murray now in a tough hole in Group B.

Ben Rothenberg @BenRothenberg 4h4 hours ago Czech Republic

Kerber won the set this weekend when she trailed a double break 0-3*, but lost all four sets when she had 4-2 leads. Tennis, man. #fedcup

SI Tennis @SI_Tennis 4h4 hours ago

Tough weekend for Kerber. Led by a break in both sets vs. Safarova and lost both, served 3x for 1st set vs. Kvitova, and led 41 in the 3rd.

SI Tennis @SI_Tennis 4h4 hours ago

Petra Kvitova seals it. Back from a break down in 1st & 1-4 in 3rd to beat Kerber 76(5), 46 64. 3rd title in 4 years for the Czechs.

rennae stubbs @rennaestubbs 4h4 hours ago

@TennisChannel thank god for tennis channel plus so I can watch the finals of fed cup! Great match between @AngeliqueKerber v @Petra_Kvitova

Anne Keothavong @annekeothavong 5h5 hours ago

Entertaining tennis and lots of drama in this #FedCup final. Fantastic fans creating a great atmosphere in the arena too

Because the main goal of our project is to research on the exercising time duration, frequency, consuming calories, it is easy to find that none of above tweets is related to our topic. Moreover though calculate the rate of relevance of 100 tweets about 10 tweets are useful. Thus the accuracy of using one exercising keyword is absolute low.

Combination keywords

To improve the searching accuracy combination keywords have been set within the library.

The following are searching keywords.

Exercising Part

'running min', 'running mins', 'running minutes', 'running hour', 'running hours',

'cycling min', 'cycling mins', 'cycling minutes', 'cycling hour', 'cycling hours’,

'swimming min', 'swimming mins', 'swimming minutes', 'swimming hour', 'swimming hours’,

'basketball min', 'basketball mins', 'basketball minutes', 'basketball hour', 'basketball hours’,

'volleyball min', 'volleyball mins', 'volleyball minutes', 'volleyball hour', 'volleyball hours',

'tennis min', 'tennis mins', 'tennis minutes', 'tennis hour', 'tennis hours’,

'football min', 'football mins', 'football minutes', 'football hour', 'football hours',

'exercise min', 'exercise mins', 'exercise minutes', 'exercise hour', 'exercise hours’,

'exercises min','exercises mins', 'exercises minutes', 'exercises hour', 'exercises hours’,

'exercising min','exercising mins', 'exercising minutes', 'exercising hour', 'exercising hours’,

Lifestyle Part

'keepfit', 'fitness', 'keep in shape', 'bodybuilding', 'keep healthy', 'loose weight', 'loosing weight',

'health apple', 'healthy apple', 'fitness apple', 'exercising apple', 'loose weight apple’,

'health banana','healthy banana', 'fitness banana', 'exercising banana', 'loose weight banana',

'health lemon','healthy lemon', 'fitness lemon', 'exercising lemon', 'loose weight lemon’,

'health orange','healthy orange', 'fitness orange', 'exercising orange', 'loose weight orange’,

'health pear','healthy pear', 'fitness pear', 'exercising pear', 'loose weight pear’,

'health milk','healthy milk', 'fitness milk', 'exercising milk', 'loose weight milk’,

'health meat','healthy meat', 'fitness meat', 'exercising meat', 'loose weight meat’,

'health vegetable','healthy vegetable', 'fitness vegetable', 'exercising vegetale', 'loose weight vegetale',

'healthcare', 'health life’

We can see clearly that besides exercising type keywords, time limitations are used to improve accuracy of results. Again we calculate the relevance level of those keywords by testing 50 tweets.

Exercising Type	Searching Keywords	Number of related tweets
running	running min	20/ 50
	running mins	17 / 50
	running minutes	11/50
	running hour	7/50
	running hours	13/50
swimming	swimming min	12/50
	swimming mins	16/50
	swimming minutes	13/50
	swimming hour	18/50
	swimming hours	17/50

cycling	cycling min	20/50
	cycling mins	29/50
	cycling minutes	19/50
	cycling hour	22/50
	cycling hours	25/50
basketball	basketball min	8/50
	basketball mins	11/50
	basketball minutes	15/50
	basketball hour	12/50
	basketball hours	21/50
volleyball	volleyball min	23/50
	volleyball mins	23/50
	volleyball minutes	12/50
	volleyball hour	15/50
	volleyball hours	18/50
football	football min	18/50
	football mins	16/50
	football minutes	17/50
	football hour	20/50
	football hours	16/50
tennis	tennis min	17/50
	tennis mins	21/50
	tennis minutes	14/50
	tennis hour	19/50
	tennis hours	17/50

Table 4-1. Number of related tweets

From the above statistics searching results we find that by using combination keywords the rate of related results is obviously improved to about 48% compared to 10% when using sole keywords.

Besides the exercising combination keywords we also add 9 array food keywords to research on the relationship between food and exercising.

4.2.2 Raw Data

After ensuring the above keywords we use streaming API provided by Twitter to obtain tweets. Since the unprocessed raw data includes many unrelated information for the project. Thus it should extract main valuable information including the following ones.

tweet_id	Every tweet has its exclusive tweet id to identify the tweet.
tweet_text	Tweet_text includes concrete tweets information published by users.
created_at	Created_at documents the time when the tweets were published.
geo_lat	Geo_lat documents the latitude position where users published tweets.
geo_long	Geo_long documents the longitude position where users published tweets.
user_id	The exclusive labels that identify tweet users.
screen_name	The name of tweet users that displays on Twitter website.
profile_image_url	The web linkage of image of user profile.
location	The location where Tweet users belong to.
description	A introduction depiction of Tweet users.
followers_count	The follower number of a user.
friends_count	The friends number of a user.
statuses_count	The number of published Tweets.
time_zone	The time zone a Tweet user belongs to.
Gender	Gender of Tweet users
Age	Age of Tweet users

Table 4-2. Extracted useful Twitter information

Location information Analysis

The location information provided by tweet users is relatively unexpected. Here exists two main problems have to been processed.

(a) The first one is part of users only fill out partial location information, however an entire one involves state position and city position such as " San Francisco, CA". When considering different levels of belonging location, results of partial users will be ignored because of missing location information. For example if a user only fills in the location "San Francisco", then it can not be identified that belongs to CA.

To deal with the problem we add an extra location matching table to ensure the state locations when only city locations are given.

State	City
CA	Los Angeles
	San Francisco
	San Diego
	San Jose
	Long Beach
	Oakland
N.Y	New York
Illinois	Chicago
Tex.	Houston
	San Antonio
	Dallas
	Austin
Pa.	Philadelphia
Ariz.	Phoenix
Ga.	Atlanta
N.C.	Charlotte
Mass.	Boston
Wash.	Seattle
	DC

Table 4-3. Belonging states of popular cities

By index to the above table when only city locations are given, their corresponding state locations can be obtained. Specifically when count statistics in states, if we could not index

state locations provided by users, then will index to the table by using provided city locations, once matching state locations are found, we can get their state locations. However to reduce the system response time only top 20 popular cities are given in the table.

(b) Another problem about location information is identical location information representing by different methods. For example when considering the state "CA", some users filled CA, some C.A, while some California. Thus when we use CA as state keyword to filter tweets another part using different state expressions will be missed.

To resolve the problem we should construct a state information table which involving all possible representation of a state.

state		
Alabama	AL	A.L
Alaska	AK	A.K
Arizona	AZ	A.Z
California	CA	C.A
Colorado	CO	C.O
Delaware	DE	D.E
Florida	FL	F.L
Georgia	GA	G.A
Hawaii	HI	H.I
Illinois	IL	I.L
Indiana	IN	I.N
Minnesota	MN	M.N
New Jersey	NJ	N.J
New York	NY	N.Y
North Carolina	NC	N.C
Texas	TX	T.X
Washington	WA	W.A
Wisconsin	WI	W.I

Table 4-4. Different expressions of cities

To test the numbers of related tweets by using different location expressions, we take five popular state involving “CA, FL, NJ, NY, GA” to test. The following table shows the analysis results.

California	2642
CA	5618
C.A	9
Florida	2086
FL	2028
F.L	8
New Jersey	728
NJ	1036
N.J	21
New York	3942
NY	4296
N.Y	63
Georgia	10
GA	1326
G.A	8

Table 4-5. Number of related tweets in 5 popular cities by different expressions

It is obvious to find from the table that when only Abbr. of states are used to identify locations, about a half information will be missed. Thus by filter tweets by using multiple location expressions will almost double the numbers of tweets.

Gender and Age Information Analysis

Gender and Age information are important ones that could not originally be extracted by using Twitter API. To obtain those information that will be critical index to analyze exercising when considering distinct gender and age people, third part API is used. However its accuracy becomes primary problems to use this API. To test the accuracy we send analysis information involving screen names, names and user descriptions of 30 famous people whose gender and age are known to the API.

"user_1" => array("screen_name" => "TomCruise", "name" => "Tom Cruise",

```
    "description" => "Actor. Producer. Running in movies since 1981.",
    "actural_gender" => "male", "actural_age" => "52"),

"user_2" => array("screen_name" => "ladygaga", "name" => "Lady Gaga",
    "description" => "The lady herself is not just a chameleon in person, but a
                      chameleon in music.",
    "actural_gender" => "female", "actural_age" => "28"),

"user_3" => array("screen_name" => "DwightHoward", "name" => "Dwight
            Howard",
    "description" => "No matter how far you fall you are never out of the fight.",
    "actural_gender" => "male", "actural_age" => "28"),

"user_4" => array("screen_name" => "kobebryant", "name" => "Kobe Bryant",
    "description" => "Dream Epic",
    "actural_gender" => "male", "actural_age" => "36"),

"user_5" => array("screen_name" => "JalenRose", "name" => "Jalen Rose",
    "description" => "Drum Major for Justice, Peace & Righteousness(MLK).
                      JRLA Founder. ABC/ESPN/Grantland Analyst. Phillipians
                      4:13.",
    "actural_gender" => "male", "actural_age" => "41"),

"user_6" => array("screen_name" => "SarahKSilverman", "name" => "Sarah
            Silverman",
    "description" => "We're all just molecules, Cutie.",
    "actural_gender" => "female", "actural_age" => "43"),

"user_7" => array("screen_name" => "PeteCarroll", "name" => "Pete Carroll",
    "description" => "Seattle Seahawks head coach. Always Compete. Win
                      Forever.",
    "actural_gender" => "male", "actural_age" => "63"),

"user_8" => array("screen_name" => "KevinSpacey", "name" => "Kevin
            Spacey",
    "description" => "Former shoe salesman now making a go at film and
                      theater. Wish me luck...",
    "actural_gender" => "male", "actural_age" => "55"),

"user_9" => array("screen_name" => "AliciaKeys", "name" => "Alicia Keys",
    "description" => "Passionate about my work, in love with my family and
                      dedicated to spreading light. It's contagious!;-)",
    "actural_gender" => "female", "actural_age" => "33"),
```

```
"user_10" => array("screen_name" => "Pink", "name" => "P!nk",
    "description" => "it's all happening",
    "actural_gender" => "female", "actural_age" => "35"),

"user_11" => array("screen_name" => "brookeburke", "name" => "Brooke
            Burke-Charvet",
    "description" => "Mommy first, wife, host, actress, fitness guru, CEO of
            @ModernMom, Author of The Naked Mom, co-
            creator/designer @CAELUM Lifestyle, Foodie,
            @operationhmchef",
    "actural_gender" => "female", "actural_age" => "43"),

"user_12" => array("screen_name" => "mindykaling", "name" => "Mindy
            Kaling",
    "description" => "You can sit with us. Instagram: mindykaling",
    "actural_gender" => "female", "actural_age" => "35"),

"user_13" => array("screen_name" => "", "name" => "Nathan Fillion",
    "description" => "It costs nothing to say something kind. Even less to shut up
            altogether.",
    "actural_gender" => "male", "actural_age" => "43"),

"user_14" => array("screen_name" => "GordonRamsay", "name" => "Gordon
            Ramsay",
    "description" => "Somewhere always near food.",
    "actural_gender" => "male", "actural_age" => "48"),

"user_15" => array("screen_name" => "Ali_Sweeney", "name" => "Alison
            Sweeney",
    "description" => "",
    "actural_gender" => "female", "actural_age" => "38"),

"user_16" => array("screen_name" => "ElizabethBanks", "name" => "Elizabeth
            Banks",
    "description" => "Amateur Goofball; proud native, Pittsfield, MA",
    "actural_gender" => "female", "actural_age" => "40"),

"user_17" => array("screen_name" => "ninadobrev", "name" => "Nina Dobrev",
    "description" => "Where ever you go... there you are. Going day by day... so
            let's see where it takes me! Namaste.",
    "actural_gender" => "female", "actural_age" => "25"),

"user_18" => array("screen_name" => "AudrinaPatridge", "name" => "Audrina
            Patridge",
```

```
    "description" => "~~Host of 1stLook!!! Airing after SNL on NBC~~
                      Instagram-AudrinaPatridge",
    "actural_gender" => "female", "actural_age" => "29"),

"user_19" => array("screen_name" => "nerdist", "name" => "Chris Hardwick",
    "description" => "Stand-upper, Zombie Therapist, Talking Snake and
                      POINTS giver",
    "actural_gender" => "male", "actural_age" => "42"),

"user_20" => array("screen_name" => "elizadushku", "name" => "Eliza
                 Dushku",
    "description" => "Official Eliza Dushku. Be forewarned: I'm accused of
                      speaking my own language here... Enjoy",
    "actural_gender" => "female", "actural_age" => "33"),

"user_21" => array("screen_name" => "ColinHanks", "name" => "Colin Hanks",
    "description" => "music geek/fan of sports/ husband/father/brother/son/
                      person of interest to few/possibly that guy from that one
                      thing you think is way underrated",
    "actural_gender" => "male", "actural_age" => "36"),

"user_22" => array("screen_name" => "paulfeig", "name" => "Paul Feig",
    "description" => "Paul is a guy who wears suits and tries not to screw things
                      up. He also created Freaks & Geeks, directed Bridesmaids
                      and The Heat and is currently making Spy.",
    "actural_gender" => "male", "actural_age" => "52"),

"user_23" => array("screen_name" => "ShannonElizab", "name" => "Shannon
                 Elizabeth",
    "description" => "Co-founder of @ShansenJewelry, actress, director, writer,
                      producer, entrepreneur, vegan, animal activist &
                      philanthropist",
    "actural_gender" => "female", "actural_age" => "41"),

"user_24" => array("screen_name" => "katyperry", "name" => "KATY
                 PERRY",

    "description" => "CURRENTLY✨BEAMING✨ON THE PRISMATIC

                      WORLD TOUR 2014!",
    "actural_gender" => "female", "actural_age" => "30"),

"user_25" => array("screen_name" => "selenagomez", "name" => "Selena
                 Gomez",
    "description" => "Get ‘The Heart Wants What It Wants’ and pre-order my
```

```
                    new collection ‘For You’ - http://smarturl.it/sga1
                    Philippians 4:13",
    "actural_gender" => "female", "actural_age" => "22"),

"user_26" => array("screen_name" => "BradPaisley", "name" => "Brad Paisley",
    "description" => "In 1972, a crack commando unit was sent to prison by a
                    military court for a crime they didn't commit. I was also
                    born.",
    "actural_gender" => "male", "actural_age" => "42"),

"user_27" => array("screen_name" => "OzzyOsbourne", "name" => "Ozzy
                Osbourne",
    "description" => "The Prince of Darkness",
    "actural_gender" => "male", "actural_age" => "65"),

"user_28" => array("screen_name" => "elissakh", "name" => "Elissa",
    "description" => "Lebanese & International singer, 3 times World Music
                    Award! I m in halethob with my new album #halethob",
    "actural_gender" => "female", "actural_age" => "42"),

"user_29" => array("screen_name" => "ashleytisdale", "name" =>
                "AshleyTisdaleFrench",
    "description" => "My official twitter page!! Hoping my tweets inspire
                    you :)",
    "actural_gender" => "female", "actural_age" => "29"),

"user_30" => array("screen_name" => "ashleytisdale", "name" => "Cher",
    "description" => "Stand & B Counted or Sit & B Nothing. Don't Litter,Chew
                    Gum,Walk Past Homeless PPL w/out Smile.DOESNT
                    MATTER in 5 yrs IT DOESNT MATTER THERE'S
                    ONLY LOVE&FEAR",
    "actural_gender" => "female", "actural_age" => “68")
```

The feedback results of the third part API are that 23 out of 30 gender prediction is correct, whose accuracy is 76.67% and 18 out of 30 age prediction is correct whose accuracy is 60%.

The test results show the accuracy of analysis gender and age is relatively high used to analyze common users’ genders and ages.

4.3 Data Organization

In computing, a hash table (hash map) is a data structure used to implement an associative array, a structure that can map keys to values. A hash table uses a hash function to compute an index into an array of buckets or slots, from which the correct value can be found.[7]

Ideally, the hash function will assign each key to a unique bucket, but this situation is rarely achievable in practice (usually some keys will hash to the same bucket). Instead, most hash table designs assume that hash collisions—different keys that are assigned by the hash function to the same bucket—will occur and must be accommodated in some way.

In a well-dimensioned hash table, the average cost (number of instructions) for each lookup is independent of the number of elements stored in the table. Many hash table designs also allow arbitrary insertions and deletions of key-value pairs, at constant average cost per operation.

In many situations, hash tables turn out to be more efficient than search trees or any other table lookup structure. For this reason, they are widely used in many kinds of computer software, particularly for associative arrays, database indexing, caches, and sets.

For the above reasons we choose to take hash table to organize extracted Twitter data. For example the following table shows how we construct Tweets table.

tweet_id	tweet_text	created_at	geo_lat	geo_long	user_id	last_updated

Table 4-6. Hash table organization of tweets information

The tweets table shows that we organize tweets information into different hash and it is easy to reference each information by searching hash.

5. User Interface Design and Implementation

In this document, we will tell you, the user, to the IOS mobile app and explain how to use it. Our goal is to help users monitor the overall health related information of the American people from different aspects and get health related suggestions. The data we use here is a week data, from Oct. 13 to 19, crawled from the Twitter database.

When you first navigate to the page, you will encounter the following screen:

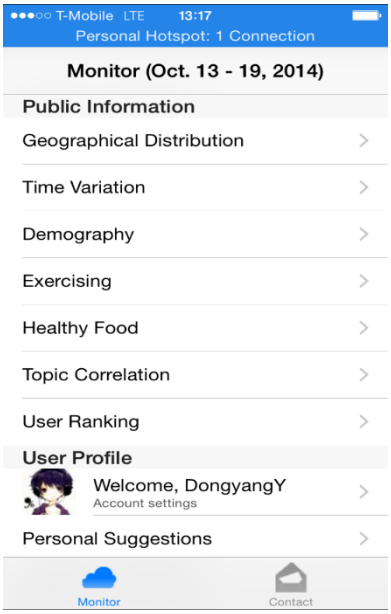

Figure 5-1. Monitor

In Figure 5-1, this interface contains two list views, “Public Information” and “User Profile”. Each item in the “Public Information” list shows the specific analysis of the public who are tweeter users. Each item in “User Profile” list shows analysis concerning user’s own tweet data. If you click the Geographical Distribution, this app will navigate you to the next screen:

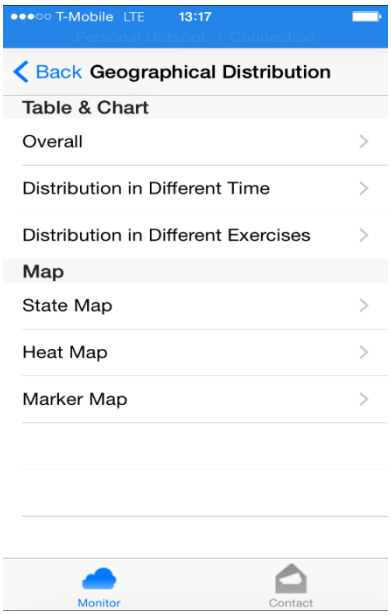

Figure 5-2. Geographical Distribution

In Figure 5-2, this interface contains a sub menu in which each item contains analytical information concerning the exercise intensity, distribution of the public. In the geographical distribution, you can see the amount of tweets in different locations such as a state map as below:

Figure 5-3. State Map

In Figure 5-3, this interface shows a US map in which each state contains information of the amount of tweets concerning exercise and health. The more the number of tweets being tweeted, the darker the color of the certain state will be. Thus California has the most amount of tweets. You may also see a heat map as below:

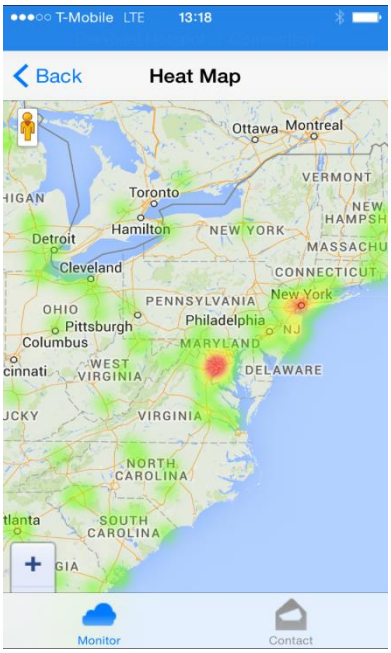

Figure 5-4. Heat Map

In Figure 5-4, when the color is closer to red, the amount of tweets is larger. You may also see a maker map as below:

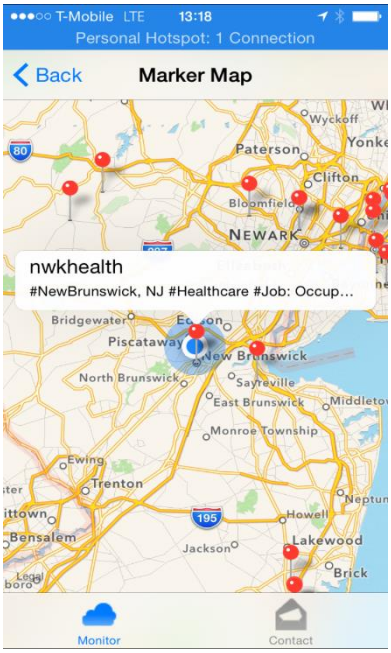

Figure 5-5. Marker Map

In Figure 5-5, this interface shows a Google map where marks will be showed on it if users post their tweet with location info. The interface will pop out certain user's name and his tweets if a single mark is clicked. You can see the recent health tweets around you, such as this figure above. Besides, we are willing to support you to see that

information in different time period in a day and in different exercise types.

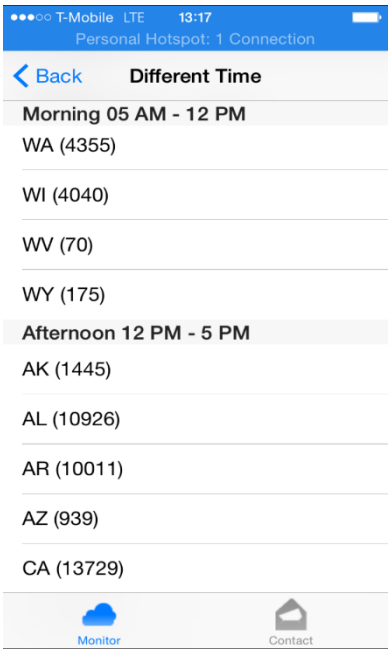

Figure 5-6. State Tweets by Different Time

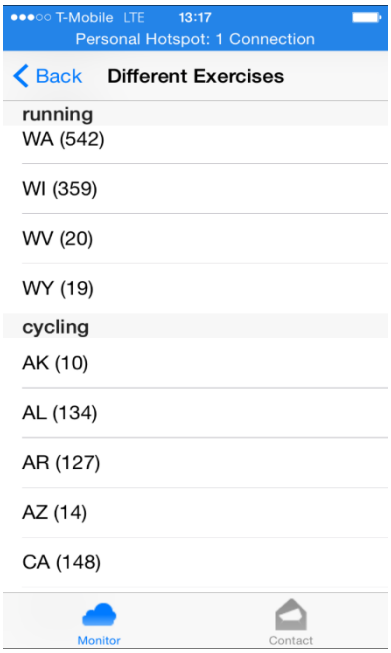

Figure 5-7. State Tweets by Different Exercise

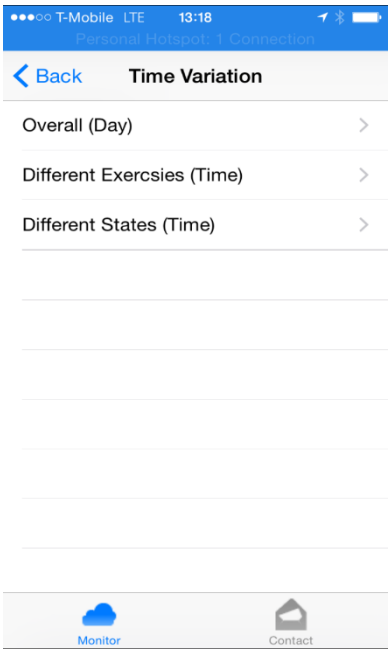

Figure 5-8. Originally: Time Variation

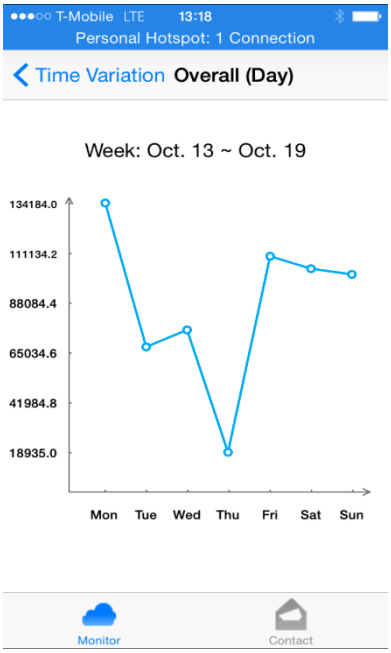

Figure 5-9. Trend of the week

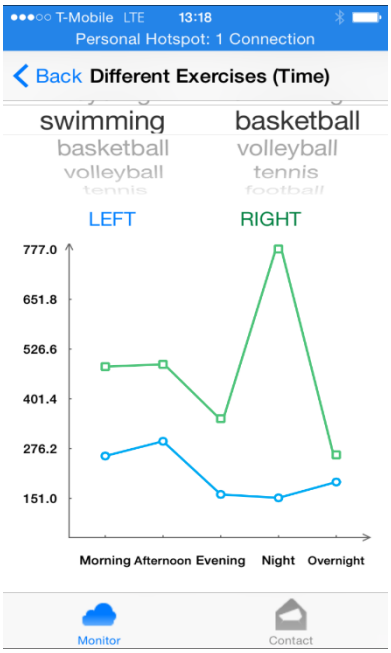

Figure 5-10. Trend by Different Exercise

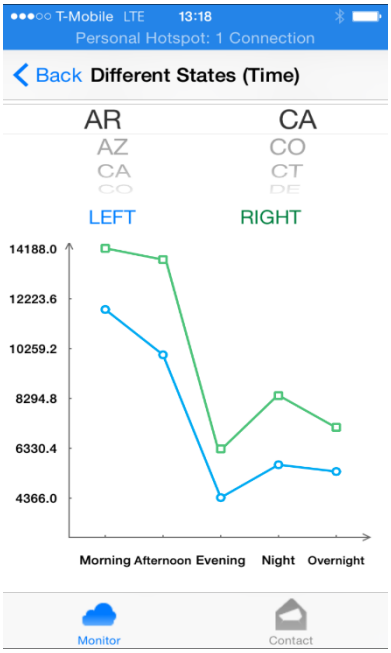

Figure 5-11. Trend by Different States

In the time variation, you may see the trend based on different days in Figure 5-9, and based on different time periods in different exercises in Figure 5-10, you can compare them like this. We also have states classification in Figure 5-11.

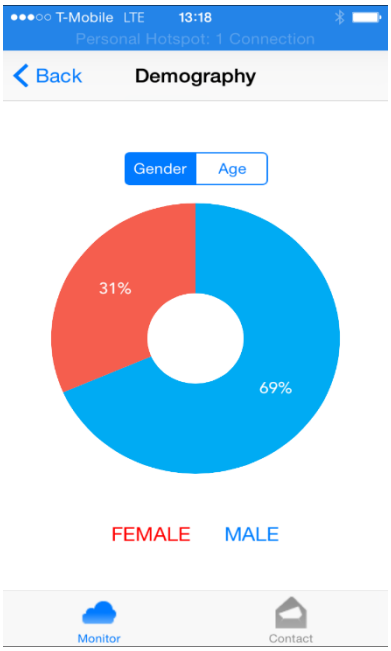

Figure 5-12. Gender Distribution

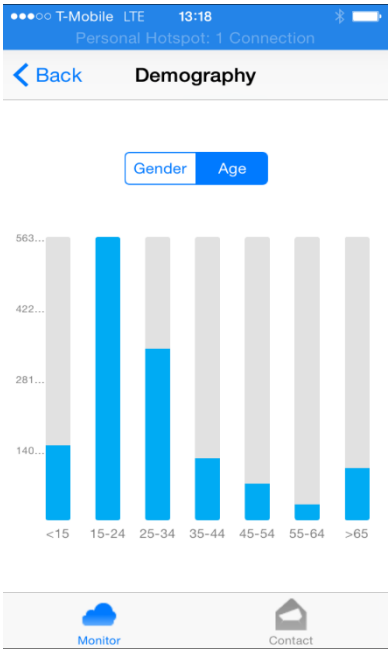

Figure 5-13. Age Distribution

In the demography, we have overall gender distribution in Figure 5-12, and age distribution in Figure 5-13.

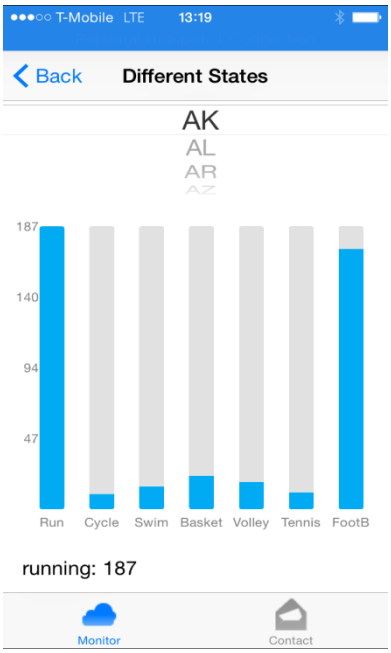

Figure 5-14. Exercise Classification by Different State

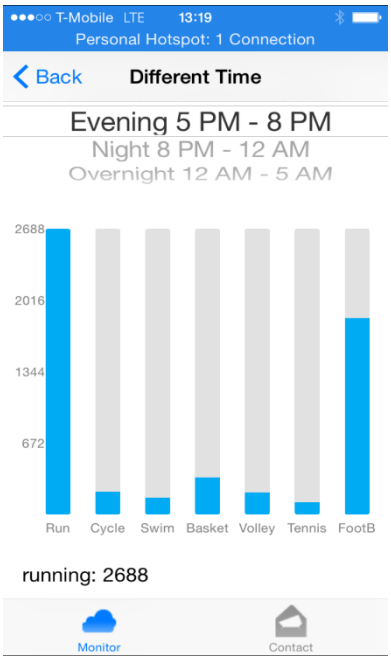

Figure 5-15. Exercise Classification by Different Time

In the exercising classification, you still can see the amount of tweets in different states in Figure 5-14 and time in Figure 5-15, you can touch the bar to see the detail.

Figure 5-16. Exercise time by Different State

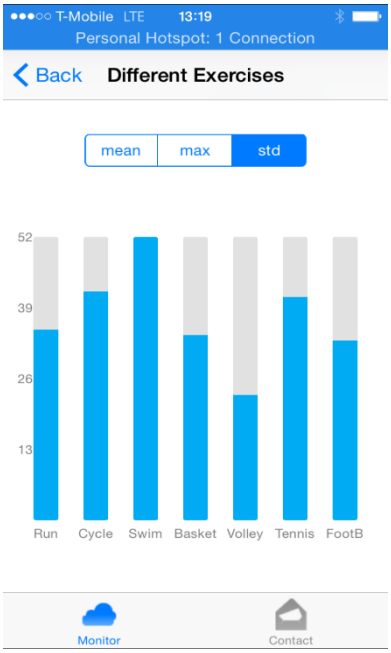

Figure 5-17. Exercise time by Different Types

Also, you can see the average, maximum, and individual difference of exercising time duration in Figure 5-16. This interface displays a bar chart of the “mean”, “max” and “standard deviation” of exercise duration time varied by different exercise types in Figure 5-17.

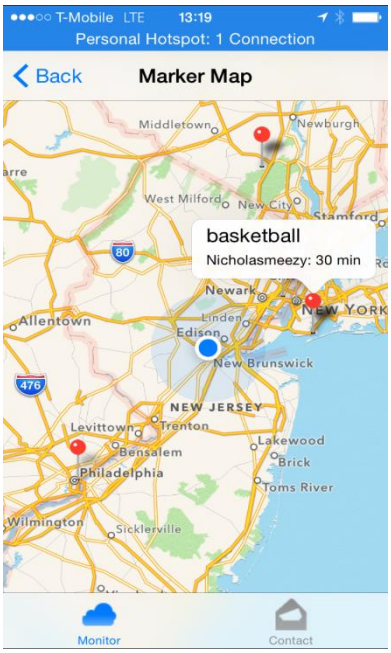

Figure 5-18. Exercise Marker Map

In Figure 5-18, we also have a marker map here. This interface shows markers on the map for users who have tweeted a tweet concerning exercise and mentioned his exercise duration time. Once the marker is clicked, the interface will pop out the duration time and the exercise type on the top of the marker. For example, a person called Nicholasmeezy has just played basketball for 30 minutes.

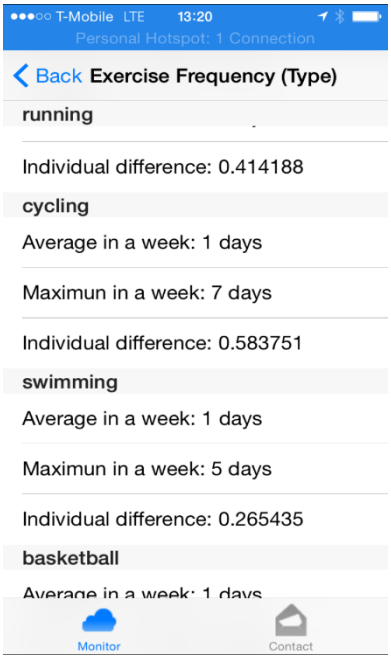

Figure 5-19. Exercise Frequency Estimation

We also have exercising frequency estimation in Figure 5-19.

In the sentiment state map, you can see the mood when people are exercising in Figure 5-20. For example, Alaska's mood value is the average.

Figure 5-20. Sentiment State Map

In Figure 5-21, this interface includes the analysis of healthy food, sorted by either different states or different exercise. For example, in Arizona people like apple most in Figure 5-22 and in the running type people like apple and pear most in Figure 5-23.

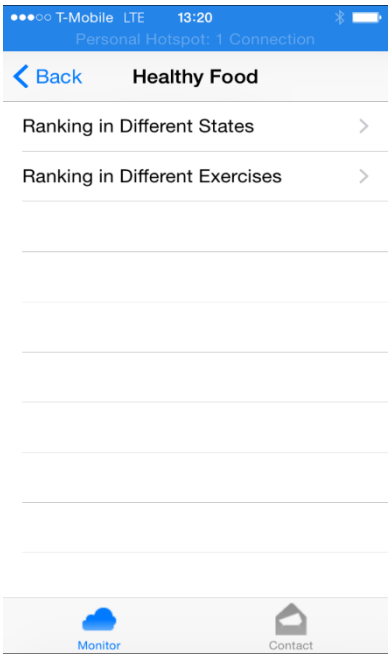

Figure 5-21. Healthy Food

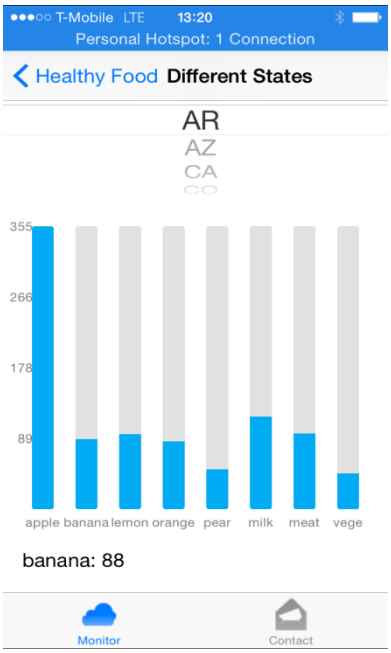

Figure 5-22. Food Analysis by Different State

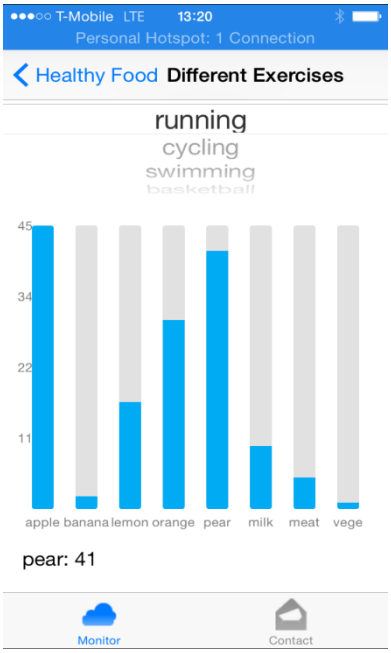

Figure 5-23. Food Analysis by Different Exercise Types

In Figure 5-24, this interface shows topic correlation between exercise and health, exercise and fruit, health and fruit. For instance, form the figure below you may see the amount of tweets about health is positive correlated with the amount of tweets about exercise.

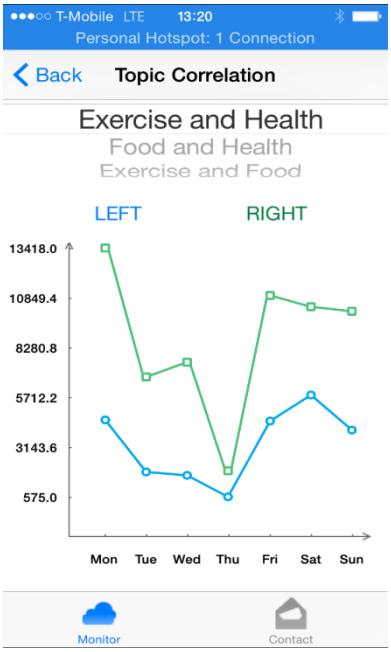

Figure 5-24. Topic Correlation

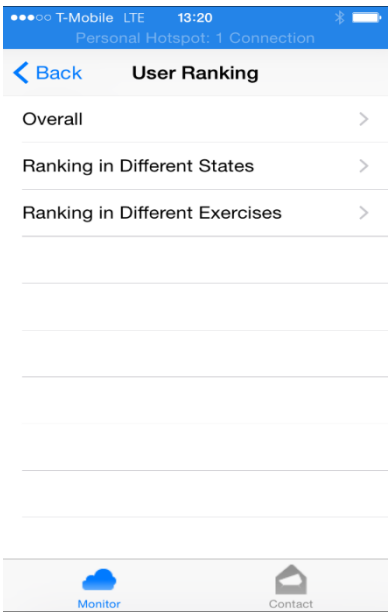

Figure 5-25. User Ranking

In Figure 5-26, this interface displays the ranking list of the users who exercise the most often in different states.

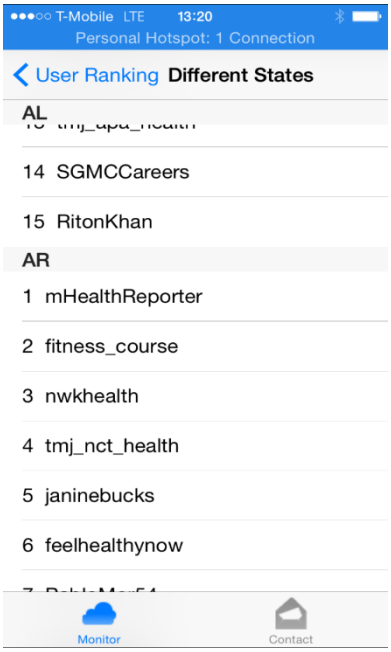

Figure 5-26. User Ranking by Different States

In Figure 5-27, this interface displays the ranking list of the users who exercise the most often in different exercise types.

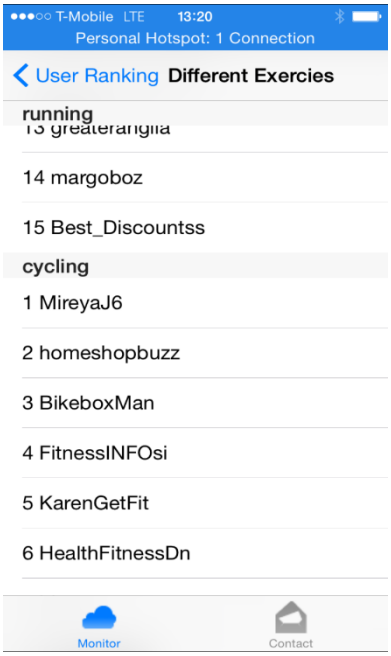

Figure 5-27. User Ranking by Different Exercise

In Figure 5-28, in this interface the users can see an analysis of their personal exercise history. Personal exercise duration, average level in area and the difference can be shown.

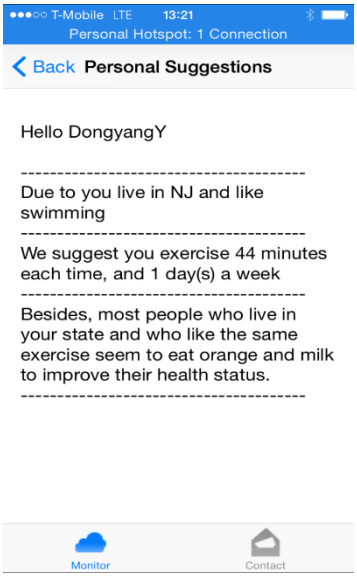

Figure 5-28. Personal Suggestions

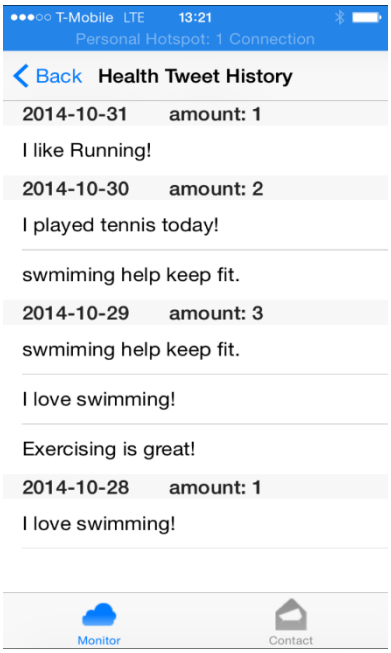

Figure 5-29. Health Tweet History

In Figure 5-29, this interface shows the history of user tweets related about Health.

6. Design of Tests

6.1. Overall Description

The design of tests aims at testing the basic function of our system. The test will be separated into two parts: the function unit test and the integrating system test. Because some function units use the same coding methodology, therefore we group them as a class and choose one unit to test. The test table is shown in table 6-1.

Function unit group	**Unit within group**	**Test unit**
Twitter data acquisition	Twitter retrieve	Twitter retrieve
Data Base setup	Database	Database
Exercise duration	Duration in different states Duration in different exercise	Duration in different states
Ranking	Leader board in different exercise Leader board in different states	Leader board in different area
Demography	Exercise demography distribution	Exercise demography distribution
Google map display	Heat map State map Marker map	Heat map

Table 6-1. The Test Table

6.2. Functional Unit Tests

The test table show from table 6-2 to table 6-7.

6.2.1. Test Unit: Twitter Retrieve

States	States Description	Input Requirement	Out Put Expected	Comment
Valid	Get Related JOSN File Successfully	Key words related to health and exercise topic	Twitter data related to health and exercise topic	This test is to make sure that whether the certain keywords can get useful tweets, and can return JSON files
Invalid	Get Unrelated JOSN File Successfully	Key words have no relationship with health and exercise topic	Twitter data have no relationship with health and exercise topic	

Table 6-2. Twitter Retrieve Test

6.2.2. Test Unit: Data Base Setup

States	States Description	Input Requirement	Out Put Expected	Comment
Valid	Data Base Set Up With Specific Data	JOSN File Extract From Twitter	Tweet Text, User Profile, Geo Information	This test is to make sure that our database will contain the exact data extract from twitter
Invalid	Data Base Set Up With Other Data	Any data	Any Data	

Table 6-3. Database Setup Test

6.2.3. Test Unit: Exercise Duration in Different States

States	States Description	Input Requirement	Out Put Expected	Comment
Valid	Table will contain number of max, min, average exercise time in different states	User's exercise time	Max, min, average exercise time in different states	This test is to make sure that

Invalid	Table will contain no information about max, min, average exercise time in different states	Any data but no User's exercise time information	No exercise time information	exercise duration can be calculated correctly

Table 6-4. Exercise Duration Test

6.2.4. Test Unit: Leader Board in Different Area

States	**States Description**	**Input Requirement**	**Out Put Expected**	**Comment**
Valid	Table will contain top 15 user who send most tweet text in specific area	More than fifteen user send Tweet text.	User profile and their total send tweet text number	This test is to make sure that number of Tweet text can be calculated correctly
Valid	If user number are less than 15 in an area, table will contain all their information but the rest will be set name to none and it's information will all set to zero	Less than fifteen user send Tweet text.	Valid user with their total send tweet text number, fake user is named none and information is zero	
Invalid	Table will contain no user information	Any data but not Tweet text	No tweet text count information	

Table 6-5. Leader Board Test

6.2.5. Test Unit: Exercise Demography Distribution

States	States Description	Input Requirement	Out Put Expected	Comment
Valid	Table will contain percentage of male and female who exercise in certain area.	specific amount of male and female users information	The percentage of male and female users.	This test is to make sure that the percentage of male and female user will be calculated correctly.
Valid	Table will contain no information about male	Only female users information	Female users are 100%	
Valid	Table will contain no information about female	Only male users information	Male users are 100%	

Table 6-6. Demography Distribution Test

Note that the gender information is not contain in Twitter database, so we can only guess the gender based on their tweet texts.

To test the gender prediction accuracy, we choose some celebrity who own a Twitter account and we have already known their gender. Then we use the guess method to see the prediction result. If the accuracy is greater than 70%, we treat this method as feasible.

6.2.6. Test Unit: Heat Map Display

States	States Description	Input Requirement	Out Put Expected	Comment
Valid	The color of the map will range from green to red according to amount of tweet text in different area	Amount of tweet texts in different area	Area with more tweet texts will become redder.	This test is to make sure that heat map will show correct color which reflect the amount of tweet texts in a certain area.
Invalid	The color of the map will remain green if no tweet texts input	No tweet texts input into any area.	Heat map will be green	

Table 6-7. Map Display Test

6.3. Integrating tests

To test the integration feature of our system, it is essential for us to focus on the jsonsender.php. Because this file takes all responsibilities to enable the communication between the database and the front-end. Once we make sure it works successfully, the integration test is finished.

States	**States Description**	**Input Requirement**	**Out Put Expected**	**Comment**
Success	The jsonsender can communicate with the website and the database	Database table with user's information	Data can be request.	This test is to show who is asking for the data and what data is being asking
Failure	The jsonsender could not communicate with the website and the database	Database table with user's information	Data could not be request	

Table 6-8. Integrating Test

7. Project Management and Plan of Work

Conference date and location:

Our team holds conferences twice a week on Tuesday and Thursday at the study room 1 in the Library of Science and Medicine.

7.1. Merging the Contributions from Individual Team Members

This report was compiled by all the team members based on their own works.

There were of course several issues encountered when combing our works. For instance, since to the large amount of data we retrieved from Twitter, it is hard to transfer it and implementing the analysis on all of our own computers. Also, the computer took the responsibilities to retrieved tweets unexpectedly went down. It seriously impacted our combining. But eventually we disassembled that computer and successfully took out the important data.

Other troublesome issue is that several operating files could not be run successfully in both Mac OS and Windows platforms due to some limits of authority problem and different rules of multiple browsers. Finally we tackled this problem by changing the method that used to enable the communication between the device/website and the server.

7.2. Project Coordination and Progress Report

7.2.1. Schedule before full report 2

Data Collecting (Server)	**14-09-12**	**14-11-11**
Investigate Twitter API	14-09-12	14-09-15
Investigate the database codes from the former groups	14-09-14	14-09-15
Re-establish the database from the former group 1 and refine the keywords	14-09-16	14-09-16
Discontinuously download the data by Twitter streaming API	14-09-17	14-09-22
Implement Twitter rest API	14-09-25	14-09-26
Investigate Facebook API, Google+ API and text	14-09-29	14-10-10

analytics APIs for demography information		
Figure out gender, age and type by text analytics API	14-10-11	14-10-16
Download a full week data	14-10-13	14-10-20
Data Analyzing (Database)	**14-09-16**	**14-11-11**
Investigate the features in reports from former groups	14-09-16	14-09-22
Tweet heat > geographical distribution	14-09-23	14-10-09
Tweet heat > variation tendency	14-09-23	14-10-09
Tweet heat > exercising classification	14-09-23	14-10-09
Tweet heat > user ranking	14-09-23	14-10-09
Exercising duration	14-09-23	14-10-16
Personal Diagnosis	14-10-10	14-10-16
Personal diagnosis supplement	14-10-16	14-10-30
Topic correlation	14-10-23	14-10-30
Exercising frequency	14-10-21	14-10-30
Tweet mood	14-10-16	14-10-30
Demography	14-10-16	14-10-30
Data Displaying (IOS & Web)	**14-09-16**	**14-11-11**
Set up the communication between front-end and rear-end	14-10-07	14-10-09
Structure the IOS app	14-09-16	14-10-06
Implement the IOS app	14-10-10	14-10-16
Investigate map graphical API	14-09-16	14-10-09
Implement marker, heat and state maps	14-10-10	14-10-16
Implement all features from former groups	14-10-09	14-10-09
Display improves compared to former groups	14-10-16	14-10-16
Refine the IOS app	14-10-16	14-10-30

Table 7-1. Past works

Above describe what have been done till now. In recent work, most of the use cases proposed before have been implemented. However, there are still some exceptions as listed below:

UC3: Advertiser information adding. This use case is thrown away due to its useless.

UC8: Advertisement updating. This use case is thrown away due to the same reason as the former.

UC15: Facilities searching. We still work on this use case.

UC16: Score system. This use case is suspended because of the limited time and

the doubt about its essence.

What is currently being tackled would be discussed in the next section.

7.3. Plan of Work

7.3.1. After full report 2

Data Collecting (Server)	**14-11-11**	**14-12-01**
Adjust the keywords and improve the method for retrieving tweets data	14-11-11	14-11-15
Download another full week data by the new method and keywords	14-11-15	14-11-22
Change access from private network to public network	14-11-22	14-12-01
Data Analyzing (Database)	**14-11-11**	**14-12-01**
Word frequency	14-11-11	14-11-18
Refine the method for recording the exercise intensity and frequency	14-11-11	14-11-18
Refine the method for calculating the Mood Value	14-11-11	14-11-25
Refine the method for inferring the demography information of user	14-11-11	14-11-28
Improve the accuracy of other features	14-11-18	14-12-01
Data Displaying (Android & Web)	**14-11-11**	**14-12-05**
Structure the website	14-11-11	14-11-15
Implement the website	14-11-15	14-11-20
Refine the website	14-11-21	14-12-05
Implement the displaying on Android	14-11-11	14-12-05

Table 7-2. Future plans

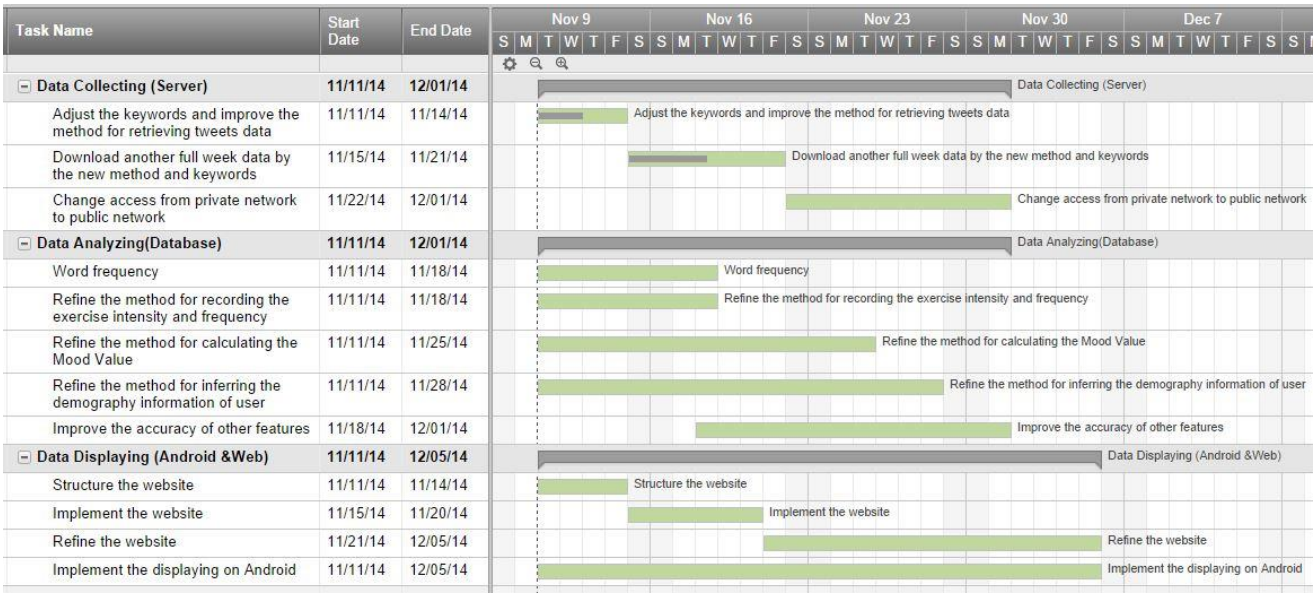

Figure 7-1. Future plans

7.4. Breakdown of Responsibilities

7.4.1. Project basic structure work

Each project basic structure work is distributed to one or more team members. However, we do not suggest each team member take part into the whole basic structure work at the very beginning. The basic structure works are mostly fixed and low-layered, and they need little changed when develop the high-layered works – features development. This arrangement contributes to improve the basic structure setting up efficiency, and diminish the reproduce and conflict when several team members do the same work. Although each team member will only do some parts of the structure work, they need to know the whole picture. Thus each team member would report his work to everyone at the conference. Besides, as pushing the work forward, the team members in different basic structure parts need to communicate with each other for the integration. So actually, each team member realizes the whole picture of our work in the end.

Assignments	D. Yao	Z. Zheng	W. Zhang	Y. Wu	W. Fang	Y. Sun
Data Collecting	√				√	
Data Storing & Rearrangement			√	√	√	√
Website Display		√	√	√		√

IOS Display	√	√				
Android Display					√	
Management & Integration	√					

Table 7-3 Basic structure

In the Table 7-3, the target of data collecting is to get the public information, e.g. the users' non-private information and tweets from Twitter, and users' private information from a demography speculation API. The team members in the part also need to schedule the date to download data, and export and import the database structure and data to other team members. Data storing and rearrangement is the design of the tables in the database. The team members in website and IOS display need to design the UI and display clear charts and maps to the viewer. The work for management is to collect creative ideas and breakdown works to each team member. The work for integration is to combine the work from each team member. The uniform rules of work are needed before the work distribution. Otherwise a lot of renaming works will appear later. The integration work also include writing communication files for JSON sending and receiving between the front-end work and the rear-end work.

7.4.2. Product ownership

Features	D. Yao	Z. Zheng	W. Zhang	Y. Wu	W. Fang	Y. Sun
Tweet heat in geographical distribution			√			
Tweet heat in variation tendency		√				
Tweet heat in exercising classification						√
Tweet heat in demography					√	
User ranking				√		
Exercising duration			√			

Exercising frequency	√					
Word frequency				√		
Correlation topics						√
Tweet sentiment		√				
Personal diagnosis	√					

Table 7-4 Features distribution

Classes	D. Yao	Z. Zheng	W. Zhang	Y. Wu	W. Fang	Y. Sun
Controller	√	√				
Communicator	√	√				
Database				√	√	√
PersonalAnalysis	√					
Demography Analysis					√	
DurationAnalysis			√			
FrequencyAnalysis			√	√		
TweetHeatCount			√	√	√	√
CorrelationCalculate						√

SentimentAnalysis		√	√			

Table 7-5 Classes distribution

In the Table 7-4 and 7-5, each team member is assigned with several feature works. That is, everyone in the team takes part into the data analyzing part – feature development. When a team member thinks out a new feature, he will report at the conference, then the work will be discussed and distributed.

Tests	D. Yao	Z. Zheng	W. Zhang	Y. Wu	W. Fang	Y. Sun
Coordinate the Integration	√					
Demography_test					√	
Lb_type_test				√		
Etime_test						√
Mood_type_test			√			
Integration Testing		√				

Table 7-6 Tests distribution

As what table 7-6 shows, it is how we arrange our integration and implement the units test and integration test.

7.4.3. Report Writing

Assignments	D. Yao	Z. Zheng	W. Zhang	Y. Wu	W. Fang	Y. Sun
Interaction Diagrams		√	√	√	√	√

Class Diagram and Interface Specification		√			√	
System Architecture and System Design	√					
Algorithm and Data Structures			√			
User Interface Design and Implementation						√
Design of Tests				√		
Project Management and Plan of Work		√				

Table 7-7 Report writing

In the Table 7-7, the writing works for each part of the full report 2 are distributed as shown. Although each part of the full report 2 is wrote by different team member, at the conference, each team member takes part into revising the report for the consistence.

8. References

[1] *The VSM model*:
http://www.csee.umbc.edu/~ian/irF02/lectures/07Models-VSM.pdf
[2] *The least square method and accuracy analysis*:
http://wenku.baidu.com/link?url=gwzOkBt_AQSSeWbnytidM9qEQT007jsxqC9Uqpt7B5qSUPBZicnFyQGs2LRFwPrr8zvaR0PmA9nR0uOVKUvpj8s60PDma8mYILzbK617wyq
[3] *Pearson product-moment correlation coefficient*:
http://en.wikipedia.org/wiki/Pearson_product-moment_correlation_coefficient
[4] *Healey & Ramaswamy. Visualizing Twitter Sentiment*:
http://www.csc.ncsu.edu/faculty/healey/tweet_viz/
[5] Margaret M. Bradley and Peter J. Lang. *Affective Norms for English Words (ANEW).*
[6] *Probability density function*:
http://en.wikipedia.org/wiki/Probability_density_function
[7] Hash table
http://en.wikipedia.org/wiki/Hash_table