`

Software Engineering

Health Monitoring Analytics

The Report-1 of Group-1

Junjie Feng, Kai Kang, Ruotian Zhang
Tianzhe Wang, Weidi Zhang, Zhijie Zhang

Table of contents

1 Customer Requirement Statement

1.1 Problem Statement

With the development of modern society, people now have more and more ways to entertain themselves. There are also more and more kinds of delicious food for people to choose, you can get all kinds of sweets or protein that you can ever imagine of. It seems that people are having a happier life with so many options to enjoy themselves.

On the other hand, people need to work really hard to feed themselves and to be able to afford all the new kinds of ways of entertainment. Work pressure is so huge these days that people seems to forget the most important thing for having a happy life, which a healthy body.

No doubt can be made on the point that the word is suffering from health crisis nowadays. So many people have problems with obesity, diabetes and these problems are actually threatening their lives. Exercising is a very effective solution for those problems.

Encourage people to get involved in more kinds of sports and do more exercise is the first goal we are trying to achieve in this project. And even let’s say that the health problem has been recognized by the public, but most of them just stop at being aware but lack of knowledge involving what sort of exercises shall be suitable for them to achieve personal healthy goal, where can they have access to find partners who share similar exercise habits including type, time and location, where would be satisfied sports buildings to take their preferred exercises, what are popular exercise that be taken by many people around them? Our second goal is to teach people what is best for themselves and what is the most efficient way for them to exercise, and how can they be involved with others when exercising.

So what we need to develop is a suite of applications that can encourage already enrolled users to exercise more and more, as well as encourage potential customers to getting onboard.

As a standing of an ordinary person, they do concern about first an immediate effect and then a longer effect. What we talk about immediate effect, we can use coupon of sports shops and gyms to encourage and attract customers. Customers are encouraged to post their information on twitter to indicate that they are exercising and we can use certain methods to verify this information. Once the information is verified and uploaded to our database, we can analysis and compare to the data the user did before and neighbor users’ exercise data to give users certain kinds of quantity of scores to quantify their work on exercise. Coupons can be given out using this method. Thus, users are encouraged to post more and more information on websites such as twitter and make us easier to get users’ information. Users are also encouraged to share this application to their friends and family members.

To attract more people to use our product, we look forward to work with companies to create a win-win situation for both the company and the exercising people. We plan to invite companies involved in the field of health and fitness to give coupons to people who uses our product and exercise hard.

We plan to gather user's data from twitter and tell them what other people are doing to keep themselves fit. So that users will not feel they are along when doing exercise, this will help users to keep exercising. The data we are going to ask from user includes the text of tweets, the user id, screen name, profile image, friends of user, and the location of tweets. And we plan to use streaming API to keep a continuous stream of public information from Twitter.

1.2 Glossary of terms

Application Programming Interface (API)

In computer programming, an application programming interface (API) is a set of routines, protocols, and tools for building software applications. An API expresses a software component in terms of its operations, inputs, outputs, and underlying types. An API defines functionalities that are independent of their respective implementations, which allows definitions and implementations to vary without compromising the interface. A good API makes it easier to develop a program by providing all the building blocks. A programmer then puts the blocks together. (https://en.wikipedia.org/wiki/Application_programming_interface)

Health Care Analysis (HCA)

Health care analytics is a product category used in the marketing of business software and consulting services. It makes extensive use of data, statistical and qualitative analysis, explanatory and predictive modeling. (https://en.wikipedia.org/wiki/Health_care_analytics)

Figure 1-1(HCA)

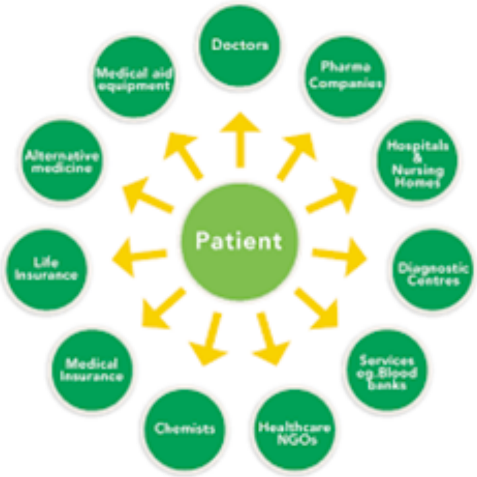


Figure 1-2

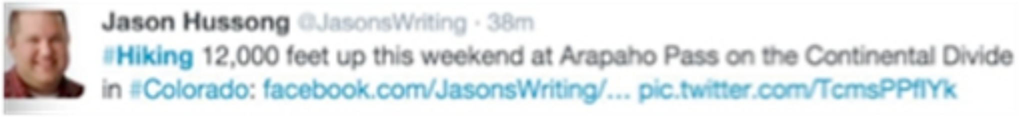
Twitter

Twitter is an online social networking service that enables users to send and read short 140-character messages called "tweets". Tweets are publicly visible by default, but senders can restrict message delivery to just their followers. Users can tweet via the Twitter website, compatible external applications, or by Short Message Service available in certain countries. (http://en.wikipedia.org/wiki/Twitter) Figure 1-3 is the logo of Twitter, Inc.

Figure 1-3

Hash-tags

A hash-tag is a word or a phrase prefixed with the number sign ("#"). It is a form of metadata tag. Words or phrases in messages on micro blogging and social networking services such as Facebook, Google+, Instagram, Twitter, or VK may be tagged by entering "#" before them, either as they appear in a sentence or appended to it. The term "hash-tag" can also refer to the hash symbol itself when used within the context of reciting a hash-tag. (http://en.wikipedia.org/wiki/Hashtag) Figure 1-4 is the Tweets with Hash-tags.



Jason Hussong @JasonsWriting · 38m
#Hiking 12,000 feet up this weekend at Arapaho Pass on the Continental Divide in #Colorado: facebook.com/JasonsWriting/... pic.twitter.com/TcmsPPflYk

Figure 1-4

Tag Cloud

A tag cloud (word cloud or weighted list in visual design) is a visual representation for text data, typically used to depict keyword metadata (tags) on websites, or to visualize free form text. The Tag Cloud could be used to show the most frequent hash-tags using the size or color of the words as a weighting factor. (https://en.wikipedia.org/wiki/Tag_cloud)

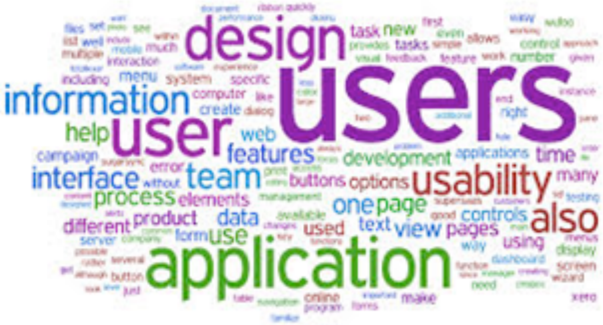


Figure 1-5

Server and Database

A server is a running instance of an application (Software) capable of accepting requests from the client and giving responses accordingly. A database is an organized collection of data. The data are typically organized to model aspects of reality in a way that supports processes requiring information. For example, modeling the availability of rooms in hotels in a way that supports finding a hotel with vacancies.(https://en.wikipedia.org/wiki/Database_server).

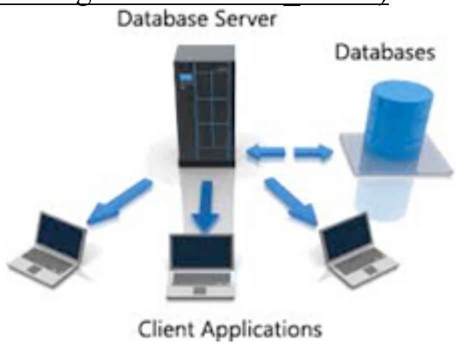


Figure 1-6

Social Networking Service

A social networking service is a platform to build social networks or social relations among people who share interests, activities, backgrounds or real-life connections (https://en.wikipedia.org/wiki/Social_networking_service).

Google Map

Google Maps is a desktop and mobile web mapping service application and technology provided by Google, offering satellite imagery, street maps, and Street View perspectives. Also supported are maps embedded on third-party websites via the Google Maps API, and a locator for urban businesses and other organizations in numerous countries around the world. (http://en.wikipedia.org/wiki/Google_Maps)

Figure 1-7

Dynamic Heat Map

A heat map is a graphical representation of data where the individual values contained in a matrix are represented as colors. Dynamic Heat Map is based on real-time data, showing the trends in the community every day in an updating frequency. (https://en.wikipedia.org/wiki/Heat_map)

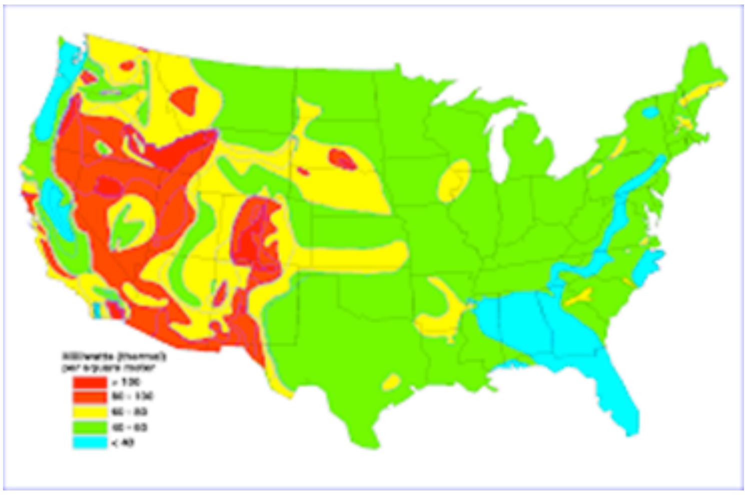


Figure 1-8

2 System Requirements

2.1 Enumerated Functional Requirements

Identifier	PW	Requirement
REQ1	5	The system shall attain raw data from Twitter and save it in a local/online database. These Tweets should be separated according to distinct classifications of exercise.
REQ2	5	The system shall attain acquire information of Twitter users such as user ID, location, follower numbers, Tweets numbers.
REQ3	5	The system shall process raw data from the local/online database and obtain the required classification information.
REQ4	5	The system shall allow users to register and provide personalized services.
REQ5	5	The system shall show leader board of popular exercises and cities.
REQ6	5	The system shall show heat maps indication active level of certain exercise in specific areas.
REQ7	5	The system shall allow users to record their daily diet information and to track them over a long period of time.
REQ8	5	The system shall provide proper diet suggestions to individual users.
REQ9	5	The system shall provide food coupon to proper users.
REQ10	4	The system shall have a reward mechanism to encourage users to exercise more.
REQ11	4	The system shall have a reminder mechanism to remind users to keep to the plans.
REQ12	4	The system shall allow users to setup and modify personal profiles.
REQ13	4	The system shall notify users by email of updating information of database.
REQ14	4	The system shall allow users to search for related information of specific type of exercise.
REQ15	4	The system shall allow users to search for other users using key words such as cities and types of exercises.
REQ16	3	The system shall recommend potential partners to users when they search for related information.
REQ17	3	The system shall recommend exercise locations to users when they search for related information.
REQ18	3	The system shall recommend exercise activities to users when they search for related information.
REQ19	3	The system shall compare the popularity of different exercises

		in aspects of distinct time interval and cities and display the results.
REQ20	3	The system shall allow users to share information of their exercise activities to social networking websites such as Facebook.
REQ21	3	The system shall allow users to invite friends to the system by sending emails.

Table 2.1 Functional Requirements

2.2 Enumerated Nonfunctional Requirements

Identifier	PW	Requirement
REQ22	5	The system should be easily accessible to any user through the website.
REQ23	5	The system should be able to update the information related to tweeter users.
REQ24	5	The system should have backup of important data incase of emergency.
REQ25	4	The system should have quick respond to large amount of users.
REQ26	5	All the data should have consistency retrieved from twitter only.
REQ27	5	The system should be easily maintained and can be fixed in a short of time when breaks down.
REQ28	4	The user interface of the website should have beautiful look and easy to navigate.
REQ29	3	The visualization of data should be direct, simple yet nice-looking.
REQ30	3	The system should not share any of users' information to any third party.
REQ31	3	The system should support high level of testing to find bugs.

Table 2.2 Nonfunctional Requirements

List, prioritize, and describe the FURPS+ requirements (also check Concepts: Requirements). The non-functional requirements numbering should continue the functional requirements list.

2.3 On-Screen Appearance Requirements

Requirements

Registered user clicks on “Sign In” button to log in our application.

People who want to register can click on the “Sign Up” button. They will be leaded to signup window.

This window shows recent tweets which about exercise.

It offers both Android and IOS app to users in order to get a better user experience.

User can share our website with other SNS.

It allows the user to search for cities, communities, neighbor-hoods, relevant hash- tags, or other users.

Users can find their partner by searching location, exercise program and time.

Add leader board with more classifications that tweet about health in different states and city.

This window shows data analysis about the amount of people in a given area exercise and the timeline of people exercise, etc.

This window shows the frequency of exercise by different states and cities. Meanwhile, any new tweet can be found on this map.

On this page, you can find a lot of information about your entered community, like Health score, Exercise rank, Total number of twitter exercisers, Exercisers percentage, Exercise frequency, Exerciser distribution, Popular sports, devices

On this page, you can view device introduction, positive experience, negative experience, device overall grade, and comments. Clicking on the “share” button, user will be linked to a “share your comment” page. Clicking on the name in the comments module, user will be linked to this reviewer’s twitter home page.

This page is where the user manages their personal account such as changing their login or password or location, etc.

Website appearance

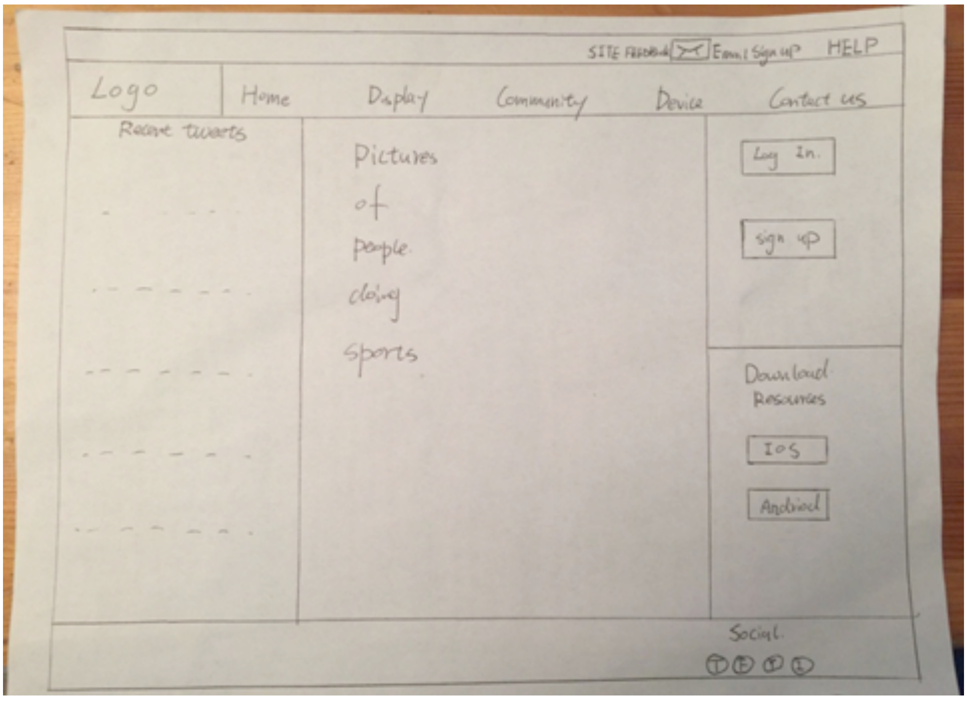


Figure 2-1

The figure 2-1 shows the homepage of our website. The users can easily see the menu: Home, Display, Community, Device and Contact Us. When clicking the Home menu, the users can see the recent tweets about health and exercise. Also, members can login to manage their personal profile. The Android and IOS apps are provided to free download. When clicking the Display menu, the users can browse several analysis charts or graphs about health and exercise. Details are shown in figure 2-2. When clicking the Community menu, members who has logged in can use the forum to contact with other members by asking or answering questions. When clicking the

Device menu, the users can see the leader board of device recommended by members. When clicking the Contact us menu, the users can contact us by making any suggestion or improvement.

3 Functional Requirement Specification

3.1 Stakeholder

For the reason that our system mainly focus on providing solutions to problems involved in sports, exercise and health monitor, these people and organizations are included in the stake holders:

System Users

Defined as people who would be our customers and will actually use our product. For exercise topic, they can use our system to view exercise heat map, leader board, score system, to find friends and gym location, to find suitable sports for individual. For health related topic they can find useful information about diet and sleep on twitter via our system, can acquire suggestions on health issues through our system, can get prize by uploading their outcome of exercise on our application

Systems architects and developers

Defined as people who designed the system who take responsibility on building the system that fulfills the user's requirements, testing and maintaining the system, and provide technical support to other stakeholders.

Academic Researchers

Defined as third party, can use our system to gain information about people's health and exercise situation for academic research purpose. They can get all kinds of dates they would want from our product, without the need to do anymore investigation.

Gym and sports equipment advertisers

Defined as our profit supporters, can pay to put advertisement in our system. Advertisement is restricted to healthy and exercise fields. They can also promote themselves by giving gifts or coupons to the lead runner of our product.

3.2 Actors and Goals

User (Initiating type)

Goals: to interact with the system, acquire exercise and health information they need, make friends and find useful exercise places via the system.

Administrator (Initiating type)

Goals: has the top priority to collect data, access, manage, and maintain the database, provide service to the user.

Advertiser: (Initiating type)

Goals: analyze data, put and manage advertisement on the system to attract customers to buy their commodities.

Google server and database (Participating type)

Goals: Get access to heat map using Google API.

Twitter server and database (Participating type)

Goals: Download the data that needs to be analyzed from Twitter, it should be dynamic.

Demography text analytics server and database (Participating type)

Goals: Should be able to analyze the text we fetched from twitter.

Our server and database (Participating type)

Goals: Easy for us to use, stable.

3.3 Use case

3.3.1 Causal Description

The summary use cases are as follow:

UC1 Data Collecting and Management

Allow the administrator to retrieve Twitter's data and save them in the system's database. The administrator can look up and check the twitter raw data and the derived data at anytime.

Derived from REQ 1,2,3.

UC2 User Registration

Allow users to create their own account on the system to use the personalized service.

Derived from REQ 4.

UC3 Data Classifying

Allow the administrator to classify the twitter raw data by some specific conditions.

Derived from REQ 2,3.

UC4 Leaderboard

Allow users to see the leaderboard of popular exercises and cities.

Derived from REQ 5.

UC5 Heatmap

Allow users to see the exercise heat in different area, time, exercise type and even their trends on the heatmap.

Derived from REQ 6.

UC6 Diet Tracking

Allow users to record and track their daily diet information.

Derived from REQ 7.

UC7 Diet Suggestions

Allow users to get the diet suggestions from the system according to their diet information.

Derived from REQ 8.

UC8 Food Coupon

Allow users to download food coupon provided by the system.

Derived from REQ 9.

UC9 Reward Score

Allow users to see their exercise score which they attain by continue to exercise and use our system.

Derived from REQ 10.

UC10 Reminder

Allow users to receive reminders from system to remind themselves to keep on their exercise plan.

Derives from REQ 11.

UC11 Profile editing

Allow users to change their own profile at anytime.

Derives from REQ 12.

UC12 Notification

Allow users to receive notification via email from the system of major changing of the database and important updates.

Derives from REQ 13.

UC13 Searching for information

Allow users to search information they interested in by using related keywords such as type of the exercise, cities and exercises locations.

Derived from REQ 14, 15, 17, 18.

UC14 Friends Recommendation

Allow users to get potential partners recommendation from system.

Derived from REQ 16.

UC15 Popularity Information

Allow users to see the comparison of the popularity of different exercises in aspects of distinct time interval and cities.

Derived from REQ 19.

UC16 Sharing to Social Networks

Allow users to share their exercise experiences to their social network website such as Facebook and Twitter.

Derived from REQ 20.

UC17 Inviting friends

Allow users to invite their friends to our system and can get reward score if their friends register our system.

Derived from REQ 21.

UC18 Third Party API

Allow the administrator to get verification of accessing and retrieving data from Twitter for further data analysis.

Derives from REQ 1,2.

UC19 Data Deleting

Allow the administrator to delete data or adjust database if necessary.

Derives from REQ 1,2.

UC20 Display

Allow user to access all the functions using graphic user interface.

Derives from REQ 7,12,15.

3.3.2 Use case diagram

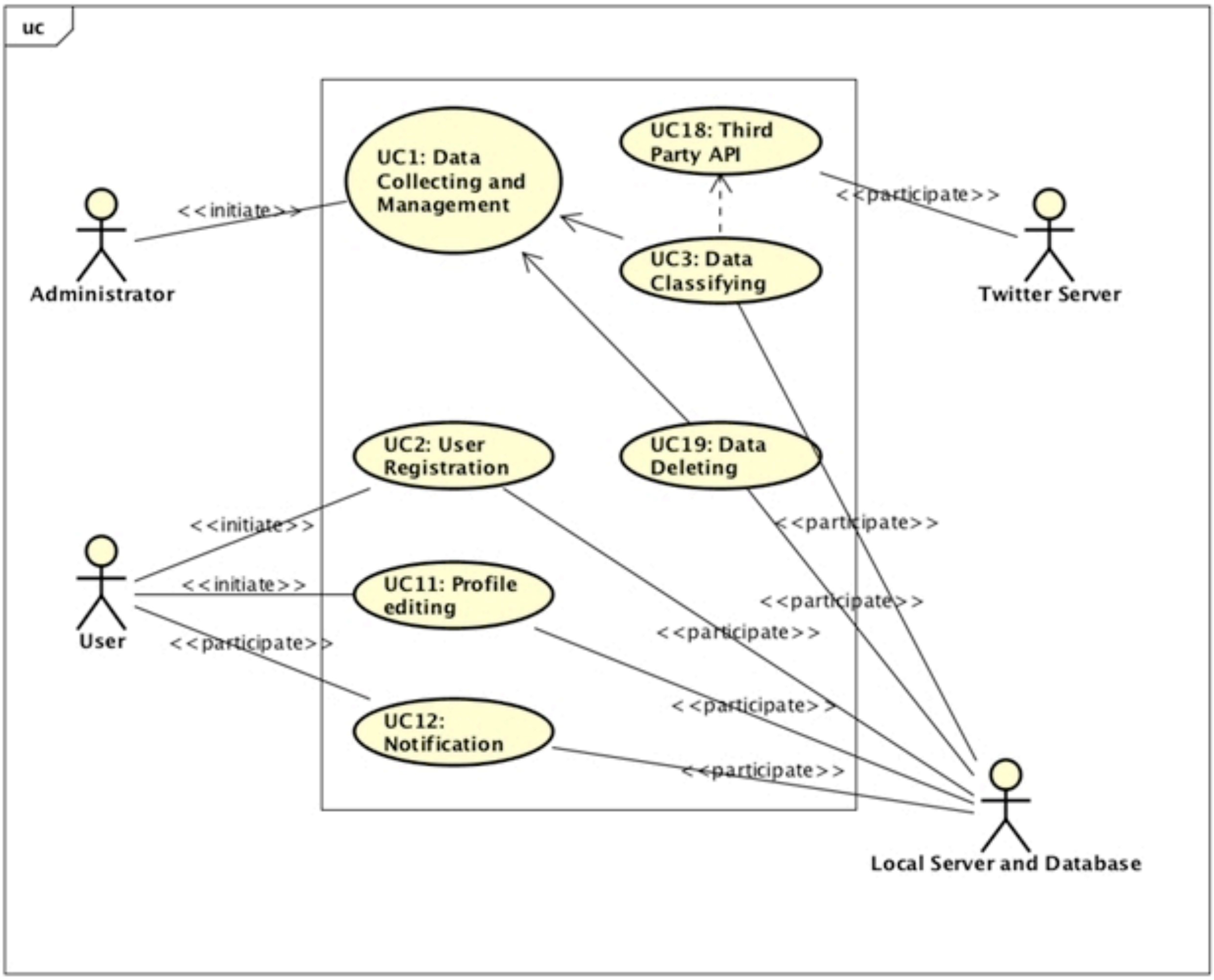


Figure 3-1 Use case diagram of server and database management subsystem

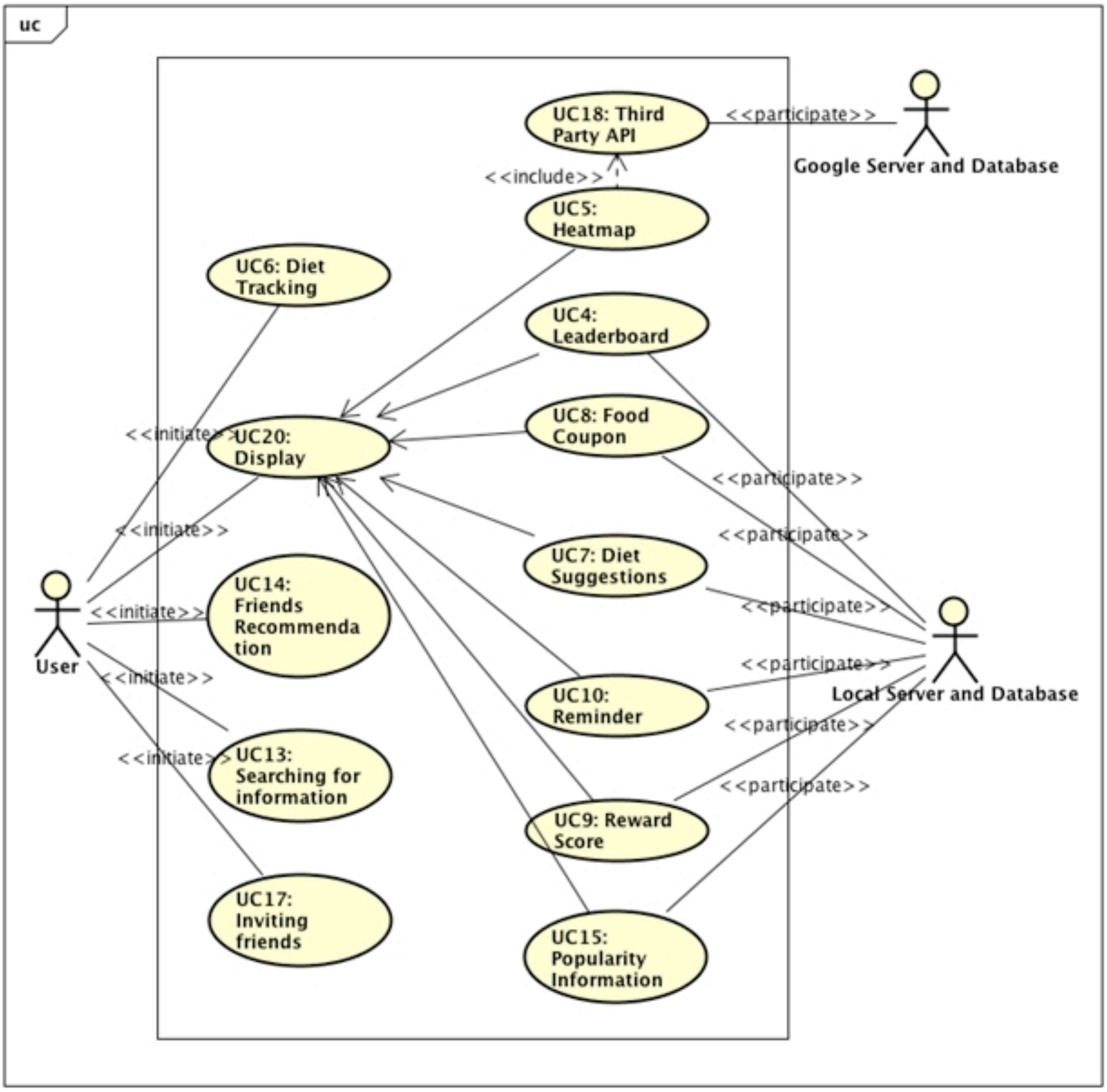


Figure 3-2 Use case diagram of displaying subsystem

3.3.3 Traceability Matrix

REQ	PW	UC-1	UC-2	UC-3	UC-4	UC-5	UC-6	UC-7	UC-8	UC-9	UC-10	UC-11	UC-12	UC-13	UC-14	UC-15	UC-16	UC-17	UC-18	UC-19	UC-20
1	5	✓																	✓	✓	
2	5	✓		✓															✓	✓	
3	5	✓		✓																	
4	5		✓																		
5	5				✓																

6	5					✓															
7	5						✓														✓
8	5							✓													
9	5								✓												
10	4									✓											
11	4										✓										
12	4											✓									✓
13	4												✓								
14	4													✓							
15	4													✓							✓
16	3														✓						
17	3													✓							
18	3													✓							
19	3															✓					
20	3																✓				
21	3																	✓			
MAX PW		**5**	**5**	**5**	**5**	**5**	**5**	**5**	**5**	**4**	**4**	**4**	**4**	**4**	**3**	**3**	**3**	**3**	**5**	**5**	**5**
TOTAL PW		**15**	**5**	**10**	**5**	**5**	**5**	**5**	**5**	**5**	**5**	**5**	**5**	**14**	**3**	**3**	**3**	**3**	**10**	**10**	**13**

Table 3-1 Traceability matrix

3.3.4 Fully-Dressed Description

Use Case UC3	Data Classifying
Related Requirements	REQ 2,3
Initiating Actor	Developer
Actor's Goal	Enable administrator to classify the twitter raw data by some special conditions.
Participating Actors	System's Server & Database
Pre-conditions	Developer has been authorized to claw information from twitter. Database is all set and ready to store new data

Post-conditions	Useful information from related Tweets is stored in system's database.
Flow of events for main success scenario: The developers set up the database and determine the keywords for searching tweets. Download the tweets information filtered by the keywords. The developers store the data in the system's database.	

Table 3-2 Fully-dressed description UC3

Use Case UC9	Reward score
Related Requirements	REQ 10
Initiating Actor	Developer
Actor's Goal	Allow users to see their exercise score which they attain by continuing to exercise and use our system.
Participating Actors	System's Server & Database
Pre-conditions	Related information is stored in the system's database
Post-conditions	The reward as coupons gives to the users by the exercise score rank in their region
Flow of events for main success scenario: The user navigated to the main interface of the system The user logs into the system using their username and password The database gives out the statistical result of the users' score rank in the local region	

Table 3-3 Fully-dressed description UC9

Use Case UC13&20	Search for information and display
Related Requirements	REQ 7,12,14,15,17,18
Initiating Actor	Developer
Actor's Goal	Allow users to search information they interested in by using related keywords and show the result in graphics.
Participating Actors	System's Server & Database

Pre-conditions	Related information is stored in the system’s database
Post-conditions	System displays the chosen diagrams on the screen
Flow of events for main success scenario: The users navigated to the main interface of the system The users choose certain distribution and type to see the statistical graphs The system displays the diagrams as requested	

Table 3-4 Fully-dressed description UC13&20

3.4 System Sequence Diagrams

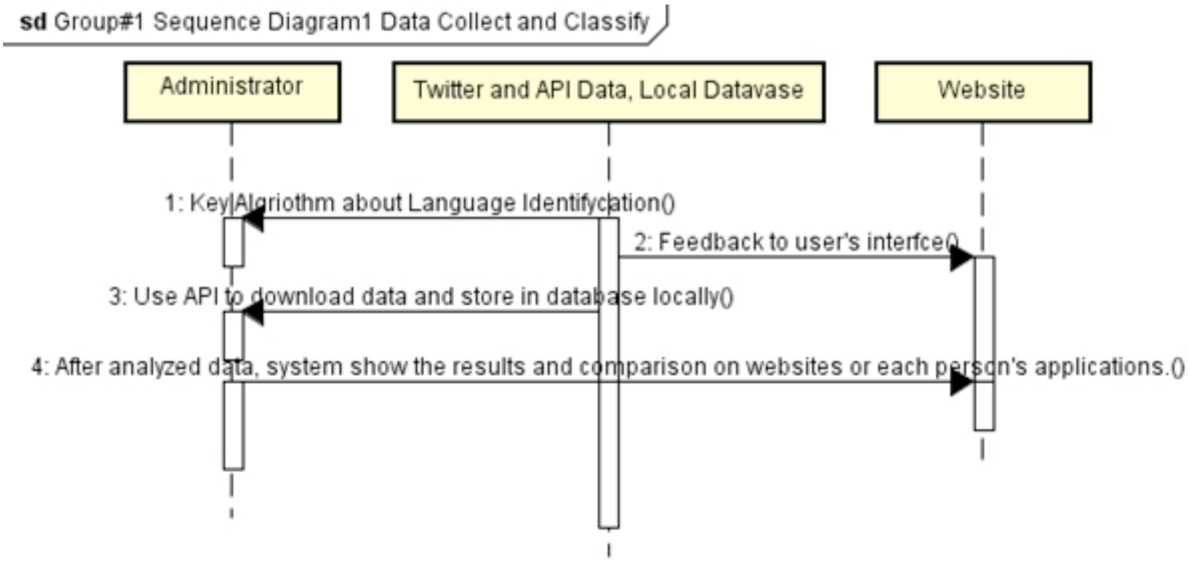

sd Group#1 Sequence Diagram1 Data Collect and Classify
Administrator
Twitter and API Data, Local Datavase
Website
1: Key Algriothm about Language Identifycation()
2: Feedback to user's interfce()
3: Use API to download data and store in database locally()
4: After analyzed data, system show the results and comparison on websites or each person's applications.()

Figure 3-3 Sequence Diagrams 1

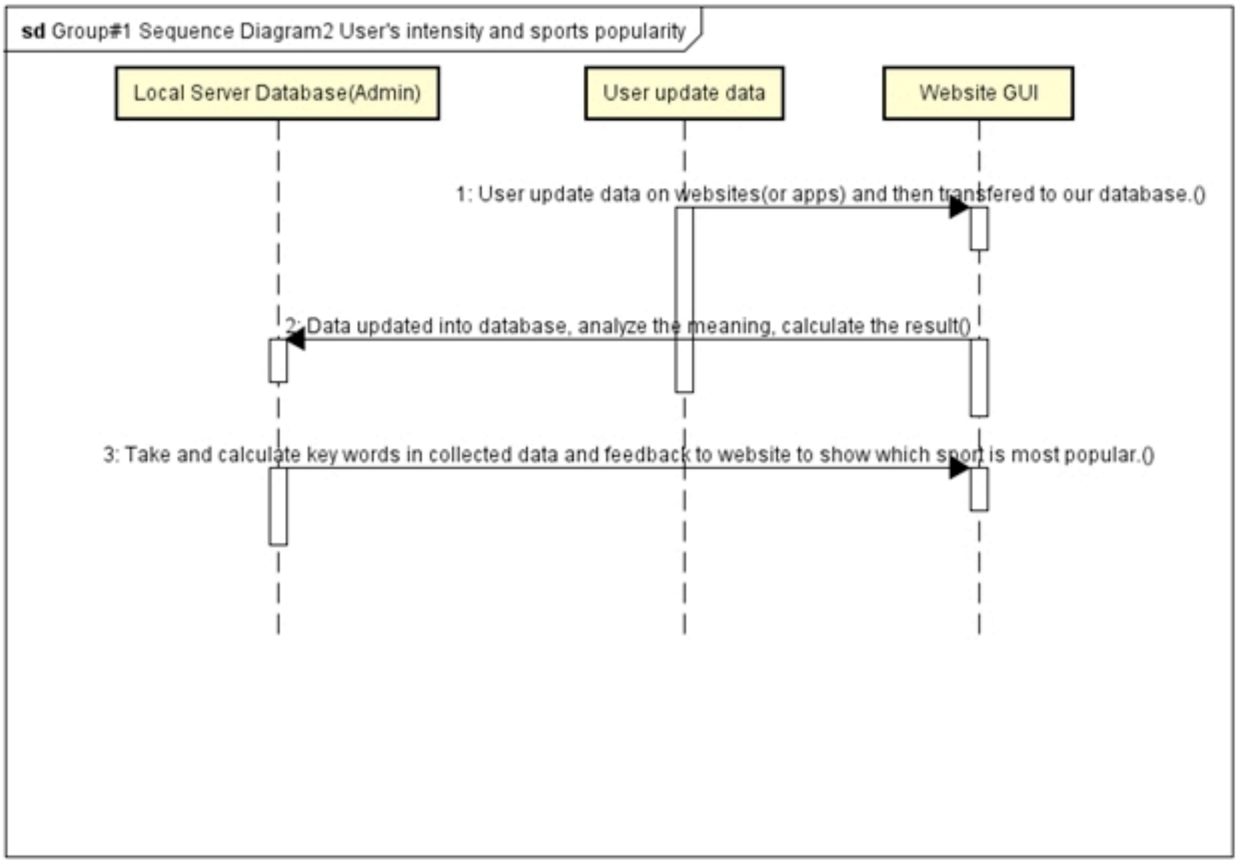

sd Group#1 Sequence Diagram2 User's intensity and sports popularity
Local Server Database(Admin)
User update data
Website GUI
1: User update data on websites(or apps) and then transfered to our database.()
2: Data updated into database, analyze the meaning, calculate the result()
3: Take and calculate key words in collected data and feedback to website to show which sport is most popular.()

Figure 3-4 Sequence Diagrams 2

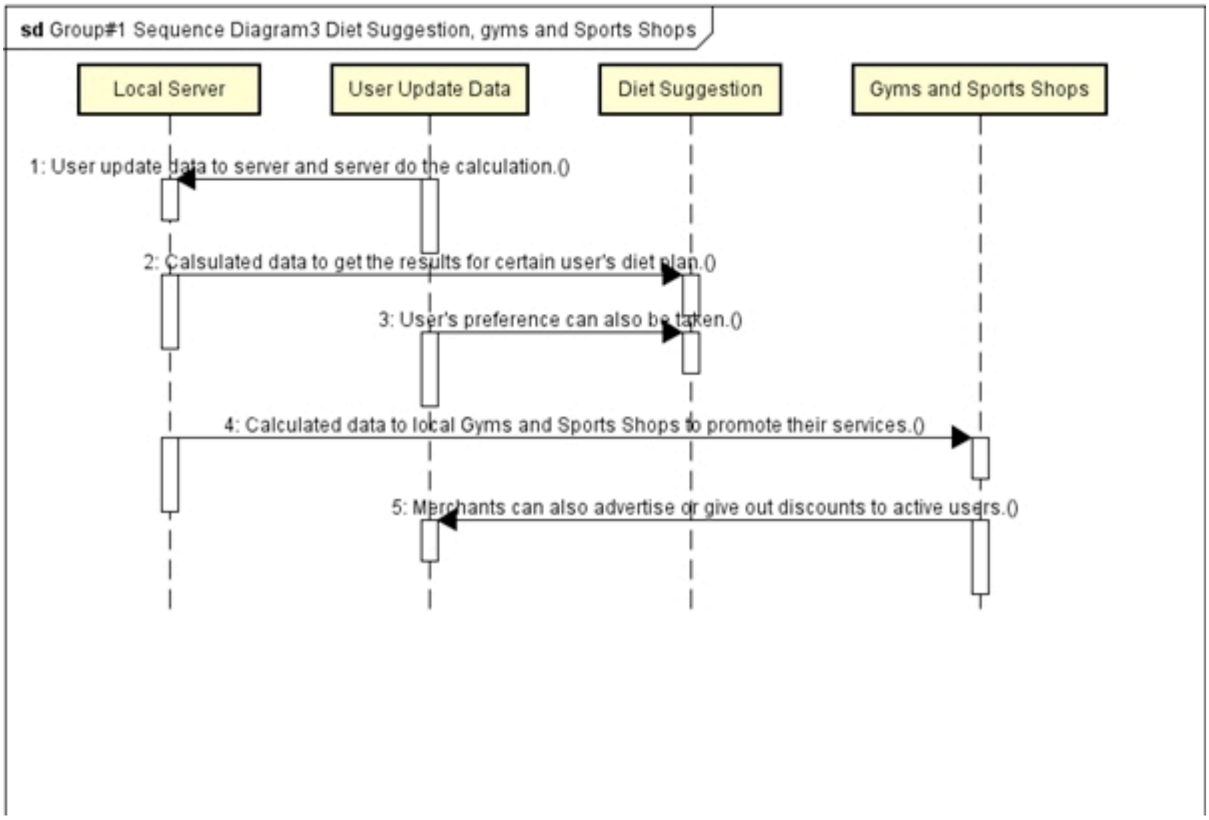

sd Group#1 Sequence Diagram3 Diet Suggestion, gyms and Sports Shops
Local Server
User Update Data
Diet Suggestion
Gyms and Sports Shops
1: User update data to server and server do the calculation.()
2: Calsulated data to get the results for certain user's diet plan.()
3: User's preference can also be taken.()
4: Calculated data to local Gyms and Sports Shops to promote their services.()
5: Merchants can also advertise or give out discounts to active users.()

Figure 3-5 Sequence Diagrams 3

4 User Interface Specification

4.1 Preliminary Design

We will give the proposed key user interface in this section, also it will contain the Preliminary Design and data analysis.

Users will log into the website's 'personal info' page to see their long term exercise and diet influence their body. The health condition may include weight, height, and body fat rate and sleep condition.

In the social section, Users would find someone who has similar interests and schedules to work out together. They want to find someone if they search for specific keywords. For example, if they type in "Jogging", a list of people who love jogging will show up on the screen.

Users also can learn others' experience in the 'experience' page if they are just begin their sports or who have specific exercise goal, hey would like to read experience and suggestions from those who are expertise on sports and those who have reached their exercise goals.

On the mobile phone app, Users can easily look for a gymnasium and to see their location and reviews from other Users. Users also can see the parks and playground nearby to find a better place to do exercise.

Users log in the website's 'personal schedule' section, after they enter their height, weight, hours for exercise per week, a specific, detailed and pertinent exercise plan will be provided their schedule.

In the 'bulletin' in the website Users may share their reviews of different sports equipment, such as, Yoga mats, sports shoes and sports legging, etc. So they can communicate about the sports equipment and make a proper purchasing decision.

Users can simply share the information through tweeter and facebook on the app on the phone, since Users may want to share their exercise activities and plans with their friends online so that they can get more suggestion and encouragement.

Information above is the main features for our website and apps on the phone, more features would be added in as stated in the system requirements.

4.2 Users effort estimation

We need to estimate the effort the customer puts into the system to obtain the required results.

Typical usage scenario 1: user wants to log in our website

NAVIGATION:

a. Click "Login" input field

b. Type User name

c. Type password
d. Click “Login” button

Typical usage scenario 2: user wants to sign up our system.
NAVIGATION:
a. Click “Sign up” Button
b. Type User Name, Email, Password,
c. Click “Submit” key

Typical usage scenario 3: log in user wants to change his password.
NAVIGATION:
a. Click “Change Password” button
b. Type User Name, Password, New Password, Confirm Password
c. Click “Submit” button

Typical usage scenario 4: user wants to see the activity partners in certain area

NAVIGATION: total 4mouse clicks and 3keystrokes
a. Click “search for partner” link
b. Type in the sports you want to play
c. Choose the area you want to find
d. Click “Search” button to see the overview information

Typical usage scenario 5: user wants to share the sports experience to his Twitter or Facebook friends and share through email.

NAVIGATION: total 7 mouse clicks
a. Click “Twitter Share” button on the upper right corner
b. Click “Tweet” button on the pop-up website
c. Click “Facebook Share” button on the upper right corner
d. Click “Share Link” button on the pop-up website
e. Click”Email Share” button on the upper right corner
f. Click “Send” button on the pop-up website.

5 Domain Analysis

5.1 Domain Model

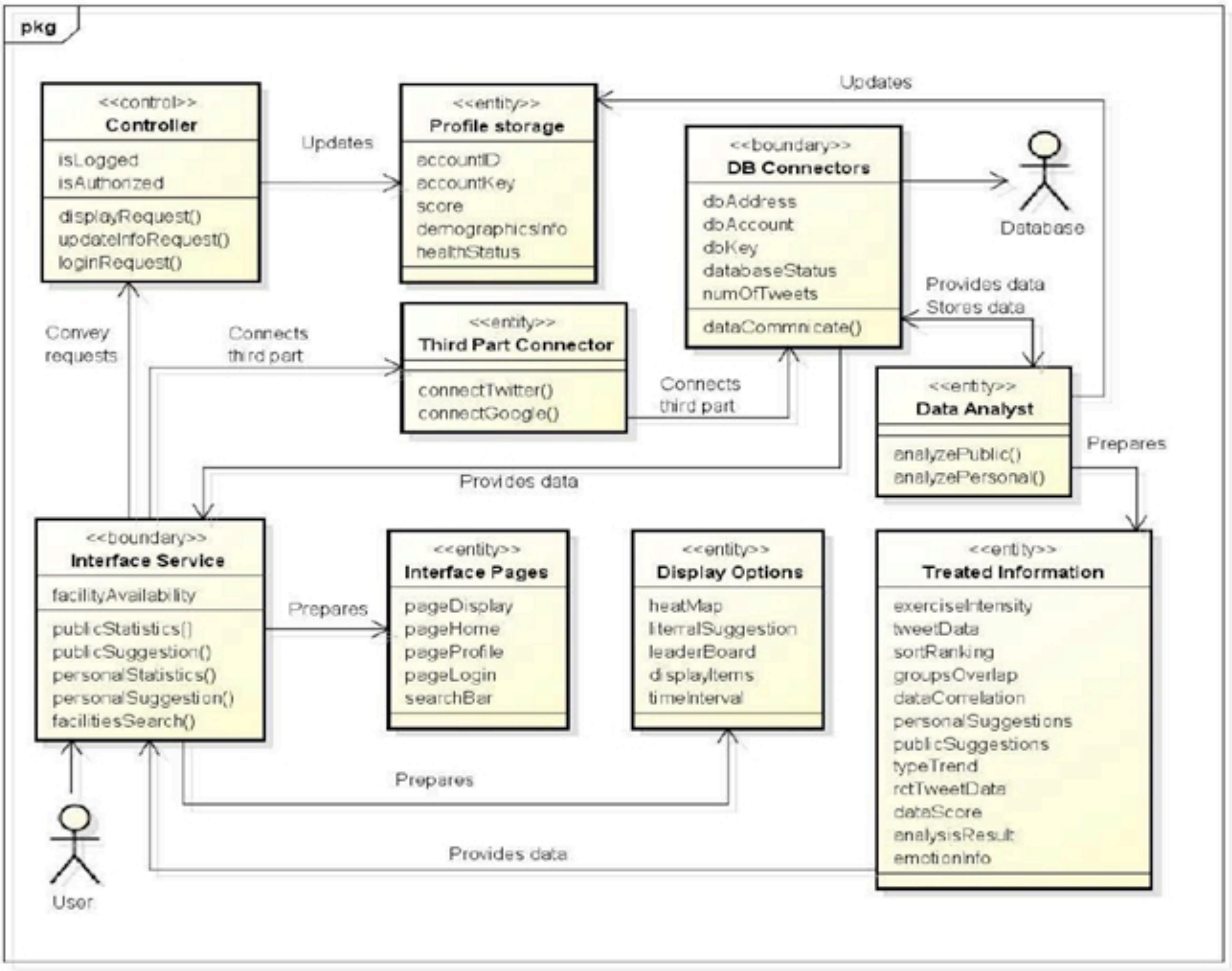


Figure 5-1 Domain model diagram

5.1.1 Concept definitions

We derive the domain model concepts from detailed user cases. Table 5-1 lists the responsibilities and the assigned concepts.

Responsibility	Type	Concept
R1: Prompt the user to make movement for available services.	D	Controller
R2: Handle requests from users.	D	Controller
R3: Deal with login issue (check users key, approve or deny).	D	Controller
R4: Container for the collection of valid keys and account	K	Profile

profile associated with users.		Storage
R5: Show pages for user to create account, login and logout.	D	Interface Service
R6: Show search engine and diversity display buttons for user to choose.	D	Interface Service
R7: Display related information in literal, numerical, graphical and map forms.	D	Interface Service
R8: Static websites and mobile phone interface that shows the user the current context, what services could be used, and outcomes of the previous request.	K	Interface Pages
R9: Specific parameters and options for information display, including search options, types of graph and display items.	K	Display Options
R10: Related information after all analysis, inferring and statistics for display.	K	Treated Information
R11: Manage interactions with the database.	D	DB Connector
R12: Classify, do statistics, analyze and infer related information for suggestion, statistics display and recommendation.	D	Data Analyst
R13: Retrieve related data from Twitter Database.	D	Third Part Connector
R14: Retrieve related services from Google Database.	D	Third Part Connector
R15: Relate to third party API, so that users can invite their friends.	D	Third Part Connector
R16: Use an algorithm to calculate the energy user take in and display.	D	Data Analyst
R17: Use an algorithm to suggest a healthy diet for user.	D	Data Analyst

Table 5-1 Concept definitions

5.1.2 Association definitions

Some concepts defined above need to work together in order to achieve specific functions. The concepts work together are called concept pair. The definition and description of concept pair of the system are listed in Table 5-2.

Concept pair	Association description	Association name
Controller ↔ Profile Storage	Controller updates User Profile when the users change his/her information.	Updates
Controller ↔Interface Service	Controller passes requests to Interface Service and receives pages prepared for displaying.	Convey requests

Interface Service ↔ Interface Pages	Interface Service prepares the Interface Pages.	Prepares
Interface Service ↔ Display Options	Interface Service prepares the Display Options.	Prepares
Interface Service ↔DB Connector	Database Connection passes the retrieved data to Interface Service to render them for display and show.	Provides data
Interface Service ↔ Treated Information	Interface Service extracts information from Treated Information for display.	Provides data
Interface Service↔ Third Part Connector	Third Part Connector enable Interface Service to connect the Third Part to ask for service (Like the Google map and graphs).	Connects Third Part
DB Connector ↔Third Part Connector	Third Part Connector enables DB Connector to connect the Third Part to retrieve related data.	Connects Third Part
Data Analyst ↔ Treated Information	Data Analyst prepares the Treated Information.	Prepares
Data Analyst ↔Profile Storage	Data Analyst changes the information (like score and health status) in Profile Storage.	Updates
DB Connector ↔Data Analyst	Database Connection passes the retrieved data to Data Analyst for analysis, inferring and statistics. Data Analyst stores useful data back into Database after analysis.	Provides data, Stores data

Table 5-2 Association definitions

5.1.3 Attribute definitions

Attributes of domain concepts are derived in Table 5-3.

Concept	Attribute	Attribute Description
Controller	displayRequest	Send user’s requests to retrieve display service
	updateInfoRequest	Send user’s requests to retrieve profile change service
	loginRequest	Use the data the user input to request a login operation
	isLogged	Identity parameter to determine whether the user is login
	isAuthorized	Identity parameter to determine whether the user is authorized
Profile Storage	accountID	Identity number used to determine the user
	accountKey	Specific key to determine the user’s credentials

	score	User's current score in the score system
	demo graphicsInfo	User's personal information like age, gender, location, etc.
	healthStatus	User's exercise-related status given by the analyst
Interface Service	publicStatistics	Show the information about public health related statistic
	publicSuggestion	Show the exercise suggestions for the whole public
	personalStatistics	Show the information about personal health related statistic
	personalSuggestion	Show the health-related suggestions for user
	facilitiesSearch	Search for facilities information with given conditions
	facilityAvailability	Identity parameter to determine whether facility is available
Interface Pages	pageDisplay	Pages for related information display.
	pageHome	Home page of the website or application.
	pageProfile	User's profile information page.
	pageLogin	Section on the website or in the application for user to login
	searchBar	In site search bar for user to search for related display service
Display Options	heatMap	Heat map for showing intensity or amount of relation data
	graph	Graph type information display
	literalSuggestion	Literal suggestion given out by the analyst showed in word
	leaderBoard	Leaderboard used to show the ranking of different data
	displayItems	Specific items for user to choose for display, such items are set as exercise amount, intensity,cites, etc.
	timeInterval	Time interval for user to choose for information display
Treated Information	exerciseIntensity	People's exercise regularity and intensity calculated by analyst
	tweetData	Data of amount and location of health-related tweets
	sortRanking	Amount, intensity or other attributes of tweets ranking among different set like cities, states, users, exercise types, etc.
	groupsOverlap	Data of the overlap between people who

		concern about wellness and who exercises and also between people who exercises and who talking about diet
	dataCorrelation	Correlation between different type of health related tweets
	personalSuggestions	Personal suggestion include exercise intensity and regularity suggestion, personal ranking, etc.
	publicSuggestions	Public suggestion based on the average intensity and regularity of exercise among the whole country
	typeTrend	Trend of amount, intensity, regularity in different exercise type
	rctTweetData	Contents, location, user information of related tweets collected recently
	dataScore	User's data in score system, including exercise score, ranking in the system, award based on the score, etc.
	analysisResult	Analysis based on User's historical exercise data, like average exercise intensity and regularity, etc.
	emotionInfo	Data about distribution, amount and type of emotion extracted from tweets.
DB Connector	dbAddress	Address of relation database
	dbAccount	Account to manage the database
	dbKey	Key to manage the database
	databaseStatus	Identity parameter to determine whether the database is open
	numOfTweets	total number of parsed tweets
	dataCommunicate	Retrieve and store data from third part server, due with data communication inside system.
Data Analyst	analyzePublic	Analysis for public information, including statistics data, suggestion inferring, trends calculating, correlation analysis, etc.
	analyzePersonal	Analysis for personal information, including historical record, personal health status deduction, specific suggestions, score calculating, etc.
Third Part Connector	connectTwitter	Connect Twitter by API Auth. to retrieve tweets
	connectGoogle	Connect Google by API Auth. to retrieve map, chart and place finding services

Table 5-3 Attribute definitions

Operation	Data analyzing

Operation	Data Classifying
Pre-conditions	Developer has been authorized to claw information from Twitter API. Database is all set and ready to store new data database="open" For initialization of the database, numOfTweets=0
Post-conditions	Useful information from related Tweets is stored in system's database. database="closed" All JASON data are analyzed in the database by many kinds of algorithms For finishing the input, numOfTweets=total number of Tweets

Operation	Reward score
Pre-conditions	Related information is stored in the system's database System has the authorization to get the tweets from database System's database can be changed Calculate the degree of different customers depending on their posted data
Post-conditions	The result can write back into the local and system's database The diagram will show on the user's termination The reward such as coupons gives to the users by the exercise score rank in their region

5.2 System Operation Contracts

Table 5-4 Data Classifying

Table 5-5 Reward score

Pre-conditions	Related information is stored in the system’s database Read the user’s input and put it into “Keyword” Use assigned algorithms to analyze related data System’s database can be changed
Post-conditions	New tables have been built in the database Statical data are stored in the related table System displays the chosen diagrams on the screen

Table 5-6 Data analyzing

Operation	Search for information
Pre-conditions	Related information is stored in the system’s database Read the user’s input and put it into “Keyword” Use assigned algorithms to analyze related data System’s database can be changed
Post-conditions	New tables have been built in the database Statical data are stored in the related table System displays the chosen diagrams on the screen

Table 5-7 Searches for information

Operation	Displaying
Pre-conditions	Related information is stored in the system’s database
Post-conditions	System displays the chosen diagrams on the screen

Table 5-8 Displaying

5.3 Mathematical Model.

5.3.1 Data process

We use several data processing methods to analyze the tweeter in order to give a ratio about the sports self-participation or audience. The ratio can indicate the way people enjoy sports in certain area.

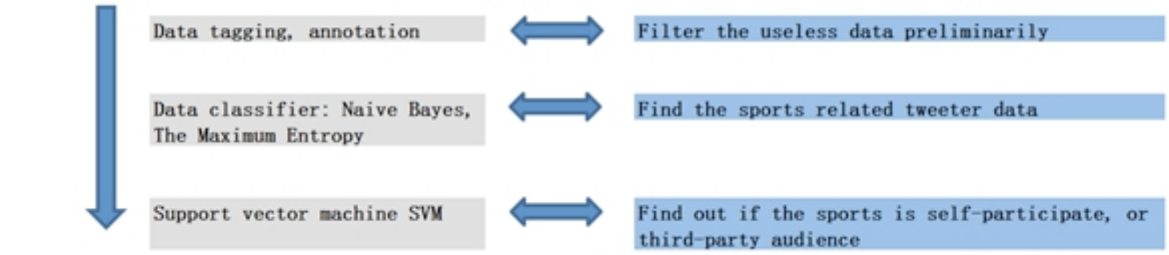

Data tagging, annotation
Filter the useless data preliminarily
Data classifier: Naive Bayes, The Maximum Entropy
Find the sports related tweeter data
Support vector machine SVM
Find out if the sports is self-participate, or third-party audience

Figure 5-2

5.3.1 Implement the NLTK method to tag the tweeter data

As we download mass raw data from twitter with certain key words, the result always can be blurry or incorrect, we need several methods to identify and make sure the valued data go to the correct position while the incorrect one will be disposed. So we use NLTK with python, to do these works.

By convention in NLTK, a tagged token is represented using a tuple consisting of the token and the tag.

For example, we want to know if the user really do the sports, then we can two twitter

a.

```
>>>text = nltk.word_tokenize(" I swim 20 miles this morning")
>>>nltk.pos_tag(text)
[('I', 'P'), (‘swim, 'v'), (' 20 ', 'NUM'), (' miles ', DET),
(this, 'RB'), (morning, 'JJ')]
```

b.

```
>>>text = nltk.word_tokenize("it’s really a swimming pool")
>>> nltk.pos_tag(text)
[('It’s', 'EX'), (‘really, 'ADV'), (' a ', 'IN'), (' good ', 'ADJ'),
(swimming, 'ADJ'), (pool, 'N')]
```

We can differentiate the meaning of swim by the tag ‘V’ and ‘ADJ’, so we can know that the first sentence is the one we want to take while second one will be discarded.

Thus sports can be represented by several key words.

Here is the meaning of each tag:

Tag	Meaning	Examples
ADJ	adjective	new, good, high, special, big, local
ADV	adverb	really, already, still, early, now
CNJ	conjunction	and, or, but, if, while, although
DET	determiner	the, a, some, most, every, no

EX	existential	there, there’s
FW	foreign word	dolce, ersatz, esprit, quo, maitre
MOD	modal verb	will, can, would, may, must, should
N	noun	year, home, costs, time, education
NP	proper	Alison, Africa, April, Washington
NUM	number	twenty-four, fourth, 1991, 14:24
PRO	pronoun	he, their, her, its, my, I, us
P	preposition	on, of, at, with, by, into, under
TO	the	word to to
UH	interjection	ah, bang, ha, whee, hmpf, oops
V	verb	is, has, get, do, make, see, run
VD	past	said, took, told, made, asked
VG	present participle	making, going, playing, working
VN	past participle	given, taken, begun, sung
WH	whdeterminer	who, which, when, what, where, how

Table 5-9

There are two tagging method

1. Unigram taggers

Unigram taggers are based on a simple statistical algorithm: for each token, assign the tag that is most likely for that particular token.

We train a Unigram Tagger by specifying tagged sentence data as a parameter when we initialize the tagger. The training process involves inspecting the tag of each word and storing the most likely tag for any word in a dictionary that is stored inside the tagger.

2. General N-Gram Tagging

When we perform a language processing task based on unigrams, we are using one segment of context. In the case of tagging, we consider only the current token, which is in isolation of any larger context. Given such a model, the best we can do is tag each word with its priority that mostly similar to tag. This means we would tag a word such as wind with the same tag, regardless of whether it appears in the context.

An n-gram tagger is a generalization of a unigram tagger whose context is the current word together with the part-of-speech tags of the n-1 preceding tokens, as shown in next picture. The tag to be chosen is circled and the context is shaded in grey. In the example of an n-gram tagger shown in Figure 5-5, we have n=3; that is, we consider that the tags of the two preceding words in addition to the current word.

An n-gram tagger picks the tag that is most likely in the given context

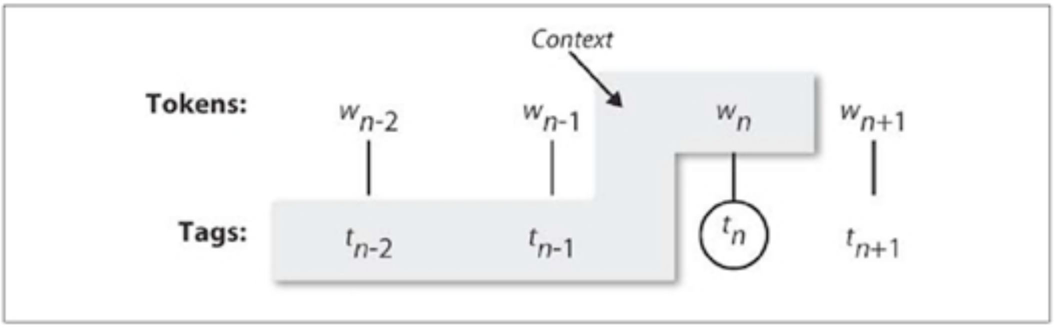


Figure 5-3

5.3.2 naive Bayes Classifiers

After tag, we need to classify every twitter text. There are several ways we can use, first we consider Naive Bayes Classifiers

Abstractly, naive Bayes is a conditional probability model: given a problem instance to be classified, represented by a vector representing some n features (independent variables), it assigns to this instance probabilities

$$p(C_k|x_1,\dots,x_n)\ \mathbf{x}=(x_1,\dots,x_n)$$

for each of K possible outcomes or classes.

The problem with the above formulation is that if the number of features n is large or if a feature can take on a large number of values, then basing such a model on probability tables are infeasible. We therefore reformulate the model to make it more tractable. Using Bayes' theorem, the conditional probability can be decomposed as

$$p(C_k|\mathbf{x})=\frac{p(C_k)\ p(\mathbf{x}|C_k)}{p(\mathbf{x})}.$$

In plain English, using Bayesian probability terminology, the above equation can be written as

$$\text{posterior}=\frac{\text{prior}\times\text{likelihood}}{\text{evidence}}.$$

In practice, there is interest only in the numerator of that fraction, because the denominator does not depend on $C$and the values of the features $F_i$are given, so that the denominator is effectively constant. The numerator is equivalent to the joint probability model

$$p(C_k,x_1,\dots,x_n)$$

which can be rewritten as follows, using the chain rule for repeated applications of the definition of conditional probability:

$$\begin{aligned}p(C_k, x_1, \dots, x_n) &= p(C_k)\ p(x_1, \dots, x_n|C_k)\\ &= p(C_k)\ p(x_1|C_k)\ p(x_2, \dots, x_n|C_k, x_1)\\ &= p(C_k)\ p(x_1|C_k)\ p(x_2|C_k, x_1)\ p(x_3, \dots, x_n|C_k, x_1, x_2)\\ &= p(C_k)\ p(x_1|C_k)\ p(x_2|C_k, x_1)\ \dots p(x_n|C_k, x_1, x_2, x_3, \dots, x_{n-1})\end{aligned}$$

Now the "naive" conditional independence assumptions come into play: assume that each feature $F_i$is conditionally independent of every other feature $F_j$for $j \neq i$, given the category C. This means that

$$p(x_i|C_k, x_j) = p(x_i|C_k),$$

$$p(x_i|C_k, x_j, x_q) = p(x_i|C_k),$$

$$p(x_i|C_k, x_j, x_q, x_l) = p(x_i|C_k),$$

and so on, for $i \neq j, q, l$. Thus, the joint model can be expressed as

$$\begin{aligned}p(C_k|x_1, \dots, x_n) &\propto p(C_k, x_1, \dots, x_n)\\ &\propto p(C_k)\ p(x_1|C_k)\ p(x_2|C_k)\ p(x_3|C_k)\ \cdots\\ &\propto p(C_k) \prod_{i=1}^{n} p(x_i|C_k)\,.\end{aligned}$$

This means that under the above independence assumptions, the conditional distribution over the class variable $C$is:

$$p(C_k|x_1, \dots, x_n) = \frac{1}{Z} p(C_k) \prod_{i=1}^{n} p(x_i|C_k)$$

where the evidence $Z = p(\mathbf{x})$is a scaling factor dependent only on $x_1, \dots, x_n$, that is, a constant if the values of the feature variables are known.

Constructing a classifier from the probability model

The discussion so far has derived the independent feature model, that is, the naive Bayes probability model. The naive Bayes classifier combines this model with a decision rule. One common rule is to pick the hypothesis that is most probable; this is known as the maximum a posteriori or MAP decision rule. The corresponding classifier, a Bayes classifier, is the function that assigns a class label $\hat{y} = C_k$for some k as follows:

$$\hat{y} = \underset{k \in \{1, \dots, K\}}{\operatorname{argmax}}\ p(C_k) \prod_{i=1}^{n} p(x_i|C_k).$$

5.3.3 The Maximum Entropy Model

The Maximum Entropy classifier uses a model that is very similar to the model employed by the naive Bayes classifier. But rather than using probabilities to set the model's parameters, it uses search techniques to find a set of parameters that will maximize the performance of the classifier. In particular, it looks for the set of parameters that maximizes the total likelihood of the training corpus, which is defined as:

$$P(\mathit{features}) = \Sigma_{x \in \text{corpus}} P(\mathit{label}(x)|\mathit{features}(x))$$

Where P(label|features), the probability that an input whose features are features will have class label label, is defined as:

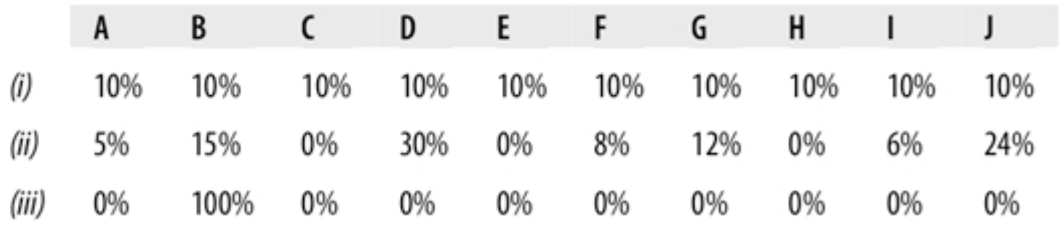
$$P(\mathit{label}|\mathit{features}) = P(\mathit{label}, \mathit{features})/\Sigma_{\mathit{label}} P(\mathit{label}, \mathit{features})$$

The Maximum Entropy classifier model is a generalization of the model used by the naive Bayes classifier. Like the naive Bayes model, the Maximum Entropy classifier calculates the likelihood of each label for a given input value by multiplying together the parameters that are applicable for the input value and label. The naive Bayes classifier model defines a parameter for each label, specifying its prior probability, and a parameter for each (feature, label) pair, specifying the contribution of individual features toward a label's likelihood.

Given the joint-features for a Maximum Entropy model, the score assigned to a label for a given input is simply the product of the parameters associated with the joint-features that apply to that input and label:

$$P(\mathit{input}, \mathit{label}) = \Pi_{\mathit{joint\text{-}features}(\mathit{input},\mathit{label})} w[\mathit{joint\text{-}feature}]$$

Suppose we are assigned the task of picking the correct word sense for a given word, from a list of 10 possible senses (labeled A–J). At first, we are not told anything more about the word or the senses. There are many probability distributions that we could choose for the 10 senses, such as:



	A	B	C	D	E	F	G	H	I	J
(i)	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%
(ii)	5%	15%	0%	30%	0%	8%	12%	0%	6%	24%
(iii)	0%	100%	0%	0%	0%	0%	0%	0%	0%	0%

Figure 5-4

Although any of these distributions might be correct, we are likely to choose distribution (i), because without any more information, there is no reason to believe that any word sense is more likely than any other.

5.3.4 SVM (Support Vector Machine)

Support Vector Machines (SVMs) are a popular machine learning method for classification, regression, and other learning tasks. LIBSVM is a library for Support Vector Machines (SVMs).

LIBSVM has gained wide popularity in machine learning and many other areas.

LIBSVM supports various SVM formulations for classification, regression, and distribution estimation.

We need to analyze if we are the one who really participate in the sports, or just watching. We use SVM to analyze.

C-Support Vector Classi_cation

Given training vectors xi 2 Rn; i = 1; : : : ; l, in two classes, and an indicator vectory 2 Rl such that yi 2 f1;�1g, C-SVC (Boser et al., 1992; Cortes and Vapnik, 1995) solves the following primal optimization problem.

$$\begin{aligned} \min_{\boldsymbol{w},b,\boldsymbol{\xi}} \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{l}\xi_i \\ \text{subject to} \quad & y_i(\boldsymbol{w}^T\phi(\boldsymbol{x}_i)+b) \geq 1-\xi_i, \\ & \xi_i \geq 0, i = 1,\ldots,l, \end{aligned}$$

where _(xi) maps xi into a higher-dimensional space and C > 0 is the regularization parameter. Due to the possible high dimensionality of the vector variable w, usually we solve the following dual problem.

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \frac{1}{2}\boldsymbol{\alpha}^T Q\boldsymbol{\alpha} - \boldsymbol{e}^T\boldsymbol{\alpha} \\ \text{subject to} \quad & \boldsymbol{y}^T\boldsymbol{\alpha} = 0, \\ & 0 \leq \alpha_i \leq C, \qquad i = 1,\ldots,l, \end{aligned}$$

where e = [1; : : : ; 1]T is the vector of all ones, Q is an l by l positive matrix, Qij _ yiyjK(xi; xj), and K(xi; xj) _ _(xi)T_(xj) is the kernel function. After problem (2) is solved, using the primal-dual relationship.

$$\boldsymbol{w} = \sum_{i=1}^{l} y_i\alpha_i\phi(\boldsymbol{x}_i)$$

And the decision function is

$$\operatorname{sgn}\left(\boldsymbol{w}^T\phi(\boldsymbol{x})+b\right) = \operatorname{sgn}\left(\sum_{i=1}^{l} y_i\alpha_i K(\boldsymbol{x}_i,\boldsymbol{x})+b\right).$$

We store yi_i 8i, b, label names,4 support vectors, and other information such as kernel parameters in the model for prediction.

Implementation

We use this function to calculate the weight

$$\phi_i(x) = \frac{\mathrm{tf}_i \log(\mathrm{idf}_i)}{\kappa}$$

Tf is the terminology means the number of occurrence in the x. idf is the ratio that the number with the entire file with the file containing the value you are testing.

Use this function; we get every feature weight for tweeter we want to predict. Use makes them the format which libsvm can use.

```
1 1:-0.555556 2:0.5 3:-0.694915 4:-0.75
3 1:-0.166667 2:-0.333333 3:0.38983 4:0.916667
2 1:-0.333333 2:-0.75 3:0.0169491 4:-4.03573e-08
1 1:-0.833333 3:-0.864407 4:-0.916667
1 1:-0.611111 2:0.0833333 3:-0.864407 4:-0.916667
3 1:0.611111 2:0.333333 3:0.728813 4:1
3 1:0.222222 3:0.38983 4:0.583333
2 1:0.222222 2:-0.333333 3:0.220339 4:0.166667
2 1:-0.222222 2:-0.333333 3:0.186441 4:-4.03573e-08
```

Format: label feature1:value1 index2:value2...

We use python or c++ project which libsvm provide to predict the result, which categories the tweeter can go into.

For example (Python):

Creating the model

```
>>mod = svm_model(prob, param)
>>target = cross_validation (prob, param, n)
```

save the model and load

```
>>mod.save('modelfile')
>>mod2 = svm_model('modelfile')
```

Test

```
r = mod.predict ([1, 1, 1])
d = mod.predict_values([1, 1, 1])
prd, prb = m.predict_probability([1, 1, 1])
```

d is the result we need to know: 1 for class0, -1 for class1.

6 Plan of Work

6.1 Project Management

Meeting date and location:

Our team holds group meeting twice a week on Monday and Wednesday at Study Room 1 or 3, Library of Science and Medicine.

6.1.1 Project basic structure work

We have one or more team members to work on each project basic structure. Since every basic structure works are mostly fixed and low-layered, little change need to be implemented when we develop the high-layered feature upon them. Therefore we suggest everyone to work on the basic structure first and integrate with other team member later. By doing so, we are able to speed up the developing process and diminish the overlap and conflict of different team members working on the relevant part. It is true that each team member may mainly working on a specific part of the project. Yet everyone needs to have a good understanding of the whole project. That's why during our group meeting, we will report out own work to the whole group. To push the whole group moving forward, we have specific deadline for each part of the work as presented in the following sections.

Assignments	Junjie Feng	Kai Kang	Ruotian Zhang	Tianzhe Wang	Weidi Zhang	Zhijie Zhang
Data Collecting		√		√		√
Data Management	√		√		√	
Website Display			√	√		√
Integration and Management	√	√			√	

Table 6-1 Basic structure

The assignments of basic structure are shown in Table 6-1. Data collecting is to get the public information and tweets from Twitter. Also using demography speculation API to get users' private information. The team members who are responsible for this part need to download data to our local database. And then they need to export and import the database structure and the data for other team member to use them. Data management means designing the tables in our database. The website display parts are responsible for design and implement the UI and display the results. The last part is for combining everyone's work and integrates it to the whole project. Integration also involves communication between front-end and rear-end which is using JSON files.

6.1.2 Product ownership (Features)

Assignments	Junjie Feng	Kai Kang	Ruotian Zhang	Tianzhe Wang	Weidi Zhang	Zhijie Zhang
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 6-2 Feature distribution

As shown in the Table 6-2, every team member is assigned with several feature works. Therefore every one will work on the data analyzing of the project. If any team member comes up a new feature, the whole group will discuss the new feature and decide whether that feature should be added into our project.

6.1.3 Breakdown of responsibilities (Report)

Assignments	Junjie Feng	Kai Kang	Ruotian Zhang	Tianzhe Wang	Weidi Zhang	Zhijie Zhang
Problem Statement & Reference		√				

Glossary of Terms	√					
Functional Requirements			√			
Non-Functional Requirements					√	
On-Screen Appearance Requirements				√		√
Stakeholders				√		
Actors and Goals			√			
Casual Description		√				
Use Case Diagram						√
Traceability Matrix	√					
Fully- Dressed Description			√			
System Sequence Diagrams						√
User Interface Specification					√	
Domain Model				√		
System Operation Contracts	√					
Mathematical Model			√			
Plan of Work		√				
Layout						√

Table 6-3 Report writing

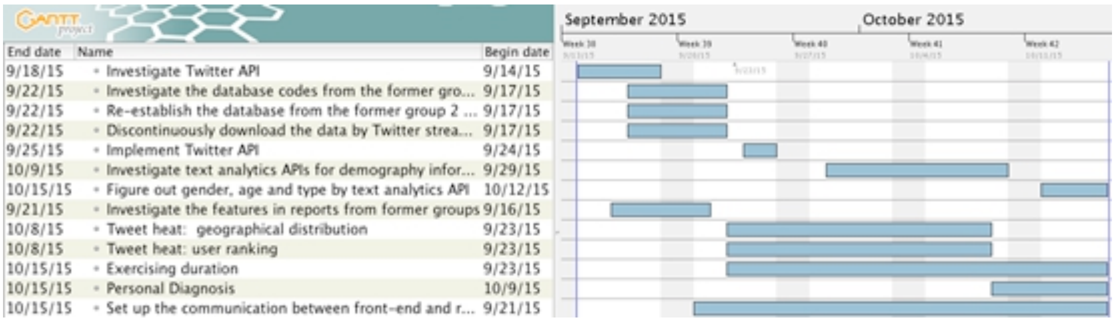
As shown in Table 6-3, the assignments of the report 1 are distributed to different team members. Although each team member only needs to write several parts of the report, the whole group revises the report 1 together.

6.2 Project Schedule

6.2.1 Before Report 1

Data Collecting (Server)	09/14/2015	10/16/2015
Investigate Twitter API	09/14/2015	09/18/2015
Investigate the database codes from the former groups	09/17/2015	09/23/2015
Re-establish the database from the former group 2 and refine the keywords	09/17/2015	09/23/2015
Discontinuously download the data by Twitter streaming API	09/17/2015	09/23/2015
Implement Twitter API	09/24/2015	09/26/2015
Investigate text analytics APIs for demography information	09/29/2015	10/10/2015
Figure out gender, age and type by text analytics API	10/11/2015	10/16/2015
Data Analyzing (Database)	09/16/2015	10/16/2015
Investigate the features in reports from former groups	09/16/2015	09/22/2015
Tweet heat: geographical distribution	09/23/2015	10/09/2015
Tweet heat: user ranking	09/23/2015	10/09/2015
Exercising duration	09/23/2015	10/16/2015
Personal Diagnosis	10/10/2015	10/16/2015
Data Displaying (Website)	09/20/2015	10/16/2015
Set up the communication between front-end and rear-end	09/20/2015	10/16/2015

Table 6-4 Previous Work



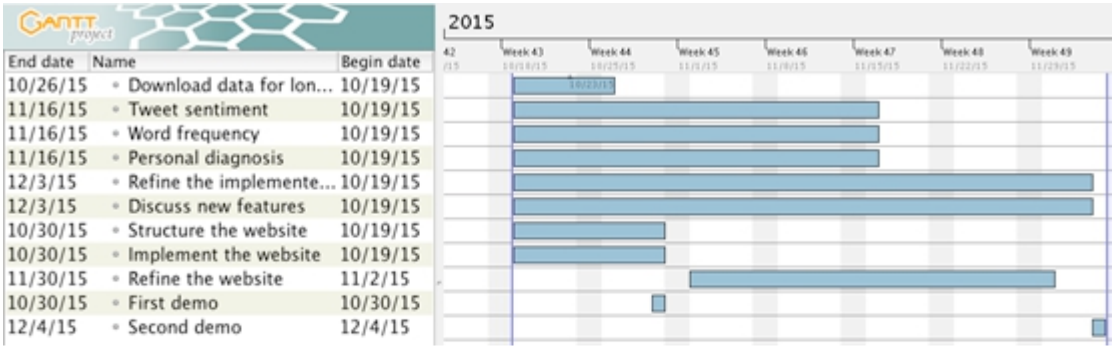
End date	Name	Begin date
9/18/15	Investigate Twitter API	9/14/15
9/22/15	Investigate the database codes from the former gro...	9/17/15
9/22/15	Re-establish the database from the former group 2 ...	9/17/15
9/22/15	Discontinuously download the data by Twitter strea...	9/17/15
9/25/15	Implement Twitter API	9/24/15
10/9/15	Investigate text analytics APIs for demography infor...	9/29/15
10/15/15	Figure out gender, age and type by text analytics API	10/12/15
9/21/15	Investigate the features in reports from former groups	9/16/15
10/8/15	Tweet heat: geographical distribution	9/23/15
10/8/15	Tweet heat: user ranking	9/23/15
10/15/15	Exercising duration	9/23/15
10/15/15	Personal Diagnosis	10/9/15
10/15/15	Set up the communication between front-end and r...	9/21/15

Figure 6-1 Previous Work

6.2.2 After Report 1

Data Collecting (Server)	10/17/2015	10/27/2015
Download data for long period	10/17/2015	10/27/2015
Data Analyzing (Database)	10/17/2015	12/05/2015
Tweet sentiment	10/17/2015	11/17/2015
Word frequency	10/17/2015	11/17/2015
Personal diagnosis	10/17/2015	11/17/2015
Refine the implemented features	10/17/2015	12/04/2015
Discuss new features	10/17/2015	12/04/2015
Data Displaying (Website)	10/17/2015	12/05/2015
Structure the website	10/17/2015	10/31/2015
Implement the website	10/17/2015	10/31/2015
Refine the website	11/01/2015	12/01/2015
First demo	10/31/2015	10/31/2015
Second demo	12/05/2015	12/05/2015

Table 6-5 Future work



End date	Name	Begin date
10/26/15	Download data for lon...	10/19/15
11/16/15	Tweet sentiment	10/19/15
11/16/15	Word frequency	10/19/15
11/16/15	Personal diagnosis	10/19/15
12/3/15	Refine the implemente...	10/19/15
12/3/15	Discuss new features	10/19/15
10/30/15	Structure the website	10/19/15
10/30/15	Implement the website	10/19/15
11/30/15	Refine the website	11/2/15
10/30/15	First demo	10/30/15
12/4/15	Second demo	12/4/15

Figure 6-2 Future work

7 References

[1] Ivan Marsic. Software Engineering. September 10, 2012.

[2] Nathan Fraenkel, Varun Gupta, Jason Lucibello, Boyang Zhang. Language Disambiguation for Disease Analysis. 2013-2014

[3] Setup Mysql: https://www.cnblogs.com/onlycxue/p/3291889.html.

[4] Setup PHP & Apache & Mysql: http://www.jb51.net/article/53787.htm. http://www.cnblogs.com/good_hans/archive/2010/04/01/1702059.html.

[5] Steven Bird, Ewan Klein, Edward Loper. Nature language processing with python. June 2009.

[6] Matthew A. Russell. Mining Social web. January 2011.

[7] Setup Apache: http://jingyan.baidu.com/article/c85b7a642df6f7003bac95d9.html

[8] Twitter API: https://dev.twitter.com/overview/documentation

[9] Rest API: https://github.com/abraham/twitteroauth

[10] Demographic API: http://textalytics.com/core/userdemographics-info

[11] Google API: https://developers.google.com/apis-explorer/#p/

[12] Setup PHP Mysql environment: http://www.cnblogs.com/good_hans/archive/2010/04/01/1702059.html

[13] A maximum entropy approach to natural language processing

[14] Phirehouse: https://github.com/fennb/phirehose

[15] S. Kumar, F. Morstatter, H. Liu. Twitter Data Analytics. Aug. 19, 2013.

[16] JSON: http://en.wikipedia.org/wiki/JSON