Foody : Food Community and Eating Pattern Tracker

V. M. Jayawardana
Department of Computer Science and Engineering, University of Moratuwa, Sri Lanka.
vindula.13@cse.mrt.ac.lk

Abstract— **Foody is a community concept based product which was developed with the main motive of catering for the social need of having one common platform for sharing the food related experiences of the users while tracking and monitoring the impact that those experiences will make on the health of each user individually. The system is designed and developed as a mobile application which includes all the essential social network features and additional nutrition related features which serve for tracking and monitoring the impact of the food on the user's health. Apart from that, the product contains a web portal to get the involvement of the food outlets with the community by adding their dishes and advertisements making the concept more attractive and financially worthy. Mobile application has been powered by the cross mobile application development technology using Phonegap while both the portals were developed accordance with the MVC architecture. The product has been able to achieve the expected outcomes within a short period of time after the trial launch proving the necessity and the novelty of the idea.**

Keywords—phonegap; cordova; MVC; cross mobile application

I. Introduction

Food is a major concern of the human as it serves as an essential of the human survival. Even though food for survival was the need in the past, with the development of technology and social network involvements, food is not anymore playing a sole act on human life as a matter of survival. Food has become a fashion, an industry and a hot topic for the social networking. Even though food has such an impact on the human life, the effect of those impacts on the health of the human is quite unaddressed particularly. On the contrary, a network of people who are passionate of food, who are willing to share their experiences and ideas on food is a platform which has a huge marketable value. From a marketing perspective, such a platform is an ideal commodity for the food outlets to pitch their products and attract more customers. The major advantage of such advertising is, all the advertising is reached only by the people who are intended to reach as everyone in the community is a food lover, resulting the maximum impact from the advertising. However this particular opportunity was an untouched opportunity in the current context. Foody comes to live as a technological solution for the above mentioned social need and a hole filler for the marketing opportunity which has not been addressed before.

Foody is a platform for food related activities and personal eating pattern tracking. The comprehensive functionalities which are inbuilt within Foody are designed to provide the user with rich experience in social networking and healthier eating. Foody is not just another social networking service which provides the same social networking related functionalities. Instead of being such, Foody integrates the power of social networking with the need of healthy eating while being a powerful advertising media. The lack of a common platform which addresses the above factors led the path for development of the Foody.

As a product which comes to alive as a solution for a social need with an added marketing value, Foody is important in several aspects. Basically it serves as a social network which is devoted for food related activities while in parallel it helps the user to keep track of his eating pattern with respect to different nutrition factors. This allows the actual user of the system to enjoy the delicious meals and share his/her experience with others while making sure he is not just focusing on eating but on healthy eating. Apart from that, the product provides the feasibility for effective marketing for the food outlets as the community gathered around the product is the market segment which is basically the key customer base of them.

This paper is structured as follows. Section I provides a brief introduction on background, motivation, importance and the purpose of the product and the concept. Section II discusses the literature relevant to similar systems which are existing in the same domain. Section III describes the system models under two sub categories; system requirements and system design. System implementation details are provided in section IV. A summarized system testing and analysis is included in section V follows by a conclusion and the future work of the product in section VI.

II. Literature Review

Modern day people are more concern about the healthier lifestyles and social interactions with each other. Due to these kind of interests of people, solutions and products based on the health and social networking aspects are becoming popular. As Foody is a product which operates in such a rigorous domain, a review on the other products which serve the same motive is a necessity. In terms of the existing systems which are operating in the same domain and catering for the same motive, "YouFood [2]", "Calorie Counter [3]", "MyFitnessPal [4]", "MyNetDiary [5]" and "MyPlate [6]" could be emphasized.

"YouFood" is a food community based currently available system which was developed by TwoGrand, Inc. It provides the facilities such as food related photo sharing, recipes sharing and commenting. The product is available for both android and iOS. However it doesn't address user's personal space by tracking his/her personal eating patterns. It only addresses the social networking requirements associated with food related activities. In contrast, "Foody" is not just an application which only caters for the social networking use cases but also facilitates the user to track his/her personal eating pattern while enjoying the food. "Instagram [7]" is also facilitating such post sharing but the difference with "Foody" is that, "Instagram" does not provide personal eating pattern tracking and most importantly they are not specifically developed as a community of food lovers. "Calorie Counter" is another free online resource for those who want to live a healthy lifestyle. Members use Calorie Count as an online diet program, a nutrition coach, and a workout partner. However it does not provide the feasibility of user to engage in social networking activities like food post sharing as it does not maintain a community for that purpose. However Foody on the other hand serves both requirements by being a social network devoted for personal eating pattern tracking. Features such as providing diet programs, being a nutrition coach that are provided by "Calorie Counter" are also inbuilt to Foody in its user's personal space. Hence any user can have the same experience they are having with apps such as "Calorie Counter" with added up features like possible health condition predictor and location based food consumption monitor. This will ultimately enhance the user experience of the users. The same scenario is applied to the other free calorie counting apps such as "MyFitnessPal", "MyNetDiary" and "MyPlate" which offers exactly the same features as "Calorie Counter". No solution provider has addressed both social and personal aspects which gives a conceptual and as well as a competitive advantage to Foody over the others.

Considering the similar systems which are operating in the same domain, as stated above "Foody" can easily unique itself from the rest of the applications. It should be noted that as the Foody comes with supportive web portal where food outlets can also join the community, integration of all the parties who need to be present for a rich food experience is perfectly completed. This particular integration cannot be seen in any of the currently existing systems. Considering all the factors as a whole, although there are some same domain applications present in the current context, Foody can be clearly differentiated among them due to the rich and diverse feature integration of it.

III. SYSTEM MODELS

A. System Requirements

In terms of the functional requirements, the major separation can be done with respect to the type of portal being used to perform those functions; web or mobile. Most of the functional requirements are associated with the mobile application. Those includes registration, adding posts, view posts, follow users, view outlets/dishes/users, rating, enter eating records, view eating pattern and few other low level use cases which are performed by the user as the actor. Apart from that, there are few system level use cases such as generating todays special, generate health tips and calculate nutrition rates which are performed by the system itself.

In terms of the functional requirements associated with the web portal, outlet registration, adding an outlet dish and posting advertisements can be considered as the major use cases. Apart from that, there are several low level use cases which act as supportive functionalities for the major use cases.

With respect to the non-functional requirements, major attention is paid to the usability and the performance of the system. As the system is intended to go to the user's hand as a mobile application, usability and performance are two crucial factors to consider. Hence several measure were taken to ensure those two non-functional requirements are satisfied. Apart from that availability of the system, security and reliability were also given emphasis in order to safeguard the user's privacy.

Figure 1, is the high level use case diagram of the system. As illustrated, there are three major actors, namely mobile user, web user (outlet) and the system (admin or system itself). As the diagram illustrates, majority of the use cases are associated with the mobile application hence usability and the performance become the crucial factors with the increased complexity.

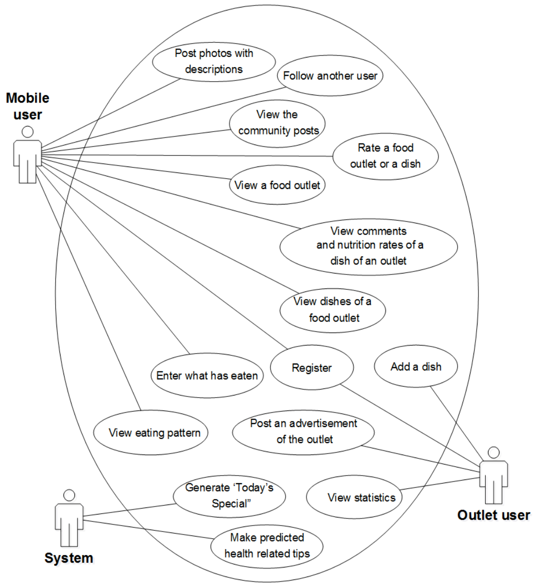

Mobile user
Post photos with descriptions
Follow another user
View the community posts
Rate a food outlet or a dish
View a food outlet
View comments and nutrition rates of a dish of an outlet
View dishes of a food outlet
Enter what has eaten
Register
Add a dish
View eating pattern
Post an advertisement of the outlet
Generate 'Today's Special"
View statistics
Outlet user
System
Make predicted health related tips

Fig. 1. Use case diagram

B. System Design

A common entity based MVC architecture was used in both mobile and web portal designing. The common code base includes the major entity classes in the problem domain and supportive classes such as database class etc. An overview of the domain level entities is shown in Figure 2. Here, inheritance can be seen in two places; user as outlet user and mobile user and dish as outlet dish post and mobile user dish post. Apart from that aggregation and composite can be seen as illustrated in Figure 2. These particular designs are done in order to simplify the system design and make the design more realistic and accurate. In terms of the associations, one to one and many to one, associations are can be seen.

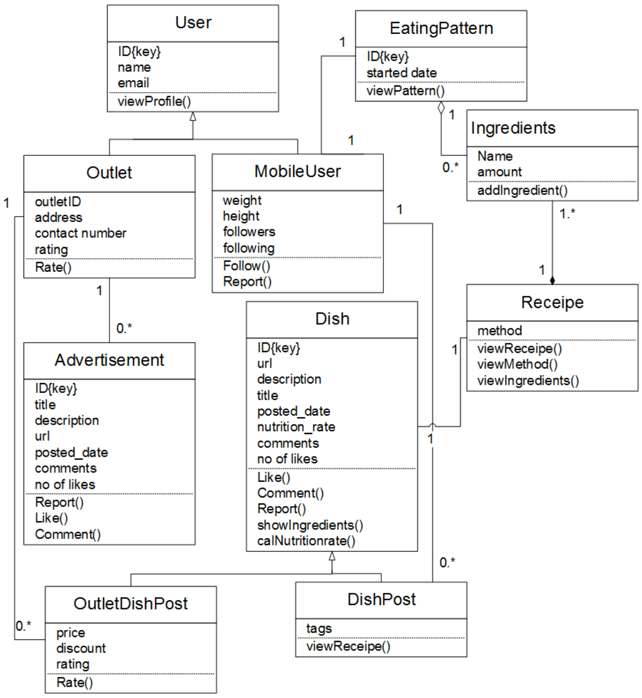


Fig. 2.Class Diagram

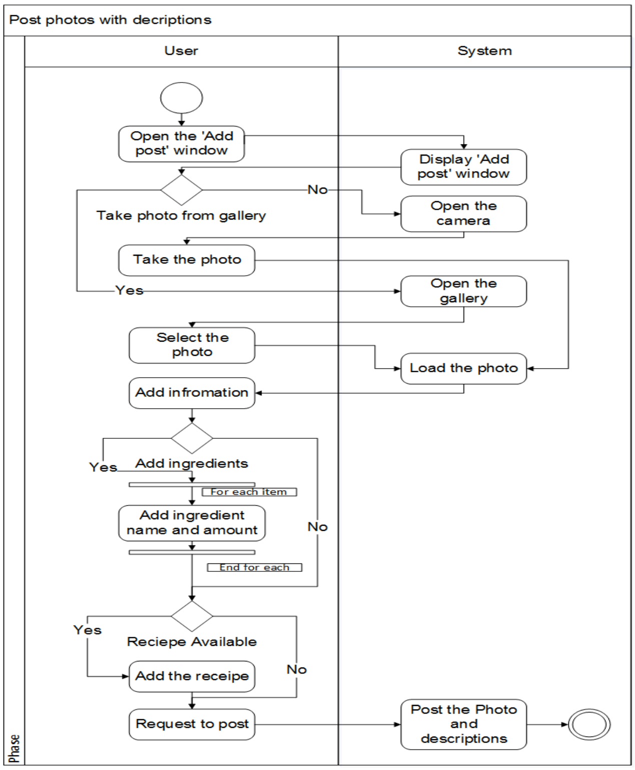


Fig. 3. Activity diagram – Post photos with descriptions

Figure 3, elaborates the activity flow in 'publishing a post'. First user has to open up the add post window. Then he will have to select and load a photo from either from the gallery or take directly from the camera. After loading the photo to share, user has to add the basic information of the post such as title. After that, user needs to add the ingredients of the dish with their amounts. Note that this option can be skipped where necessary. After that user can add the method /recipe of the dish. Again this option also can be skipped at the user's desire. After completing the inputs, user can post the photo with added information to the Foody community.

C. *Database Design*

As seen in the diagram Figure 4, ER diagram of the system is developed with an eye towards the extended ER features. The diagram includes extended ER features such as generalization to make the ER design more realistic and simple. A role base mechanism can be seen for the implementation of the 'following a user' feature. It can be seen that login and nutrition entity sets are defined to be weak entity sets with defined discriminator for each one of them. Disjoint design constrain has been used within the two generalization cases to ensure the consistency of the schema. Apart from that total and partial participations are used in where necessary to ensure the non-redundancy of the data.

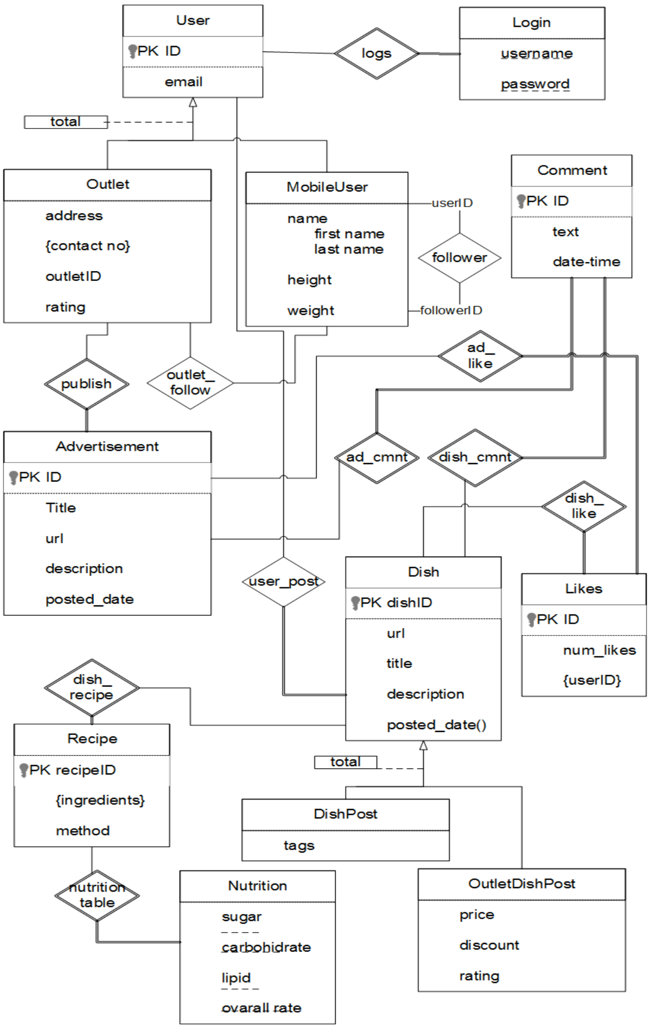


Fig. 4. ER Diagram

IV. System Implementation

A. Implementation Procedure

In terms of the development of the system, two approaches were used for the mobile and web portal development. However both the system were built on top of the MVC structure and a common code base.

Mobile application was developed using the cross platform hybrid application development tool- PhoneGap [8]. Due to that reason, the mobile application is ready to be deployed to all the major mobile platforms including android, iOS and windows. In the development, few verified Cordova [9] plugins were used as a brought in components for the ease of development and to reduce the complexity.

Cordova-plugin-camera 2.1.1 "Camera" plugin was used for accessing the device camera and device media gallery. Cordova-plugin-file-transfer 1.5.1 "File Transfer" and Cordova-plugin-file 4.1.1 "File" were used for uploading the files (images) to a remote server though the internet connection. Cordova-plugin-dialogs 1.2.1 "Notification" plugin was used to generate local notifications. To give the application a native look Cordova-plugin com.telerik.plugins.nativepagetransitions 0.6.5 "Native Page Transitions" was used. Apart from that mobile application interfaces were powered by HTML, CSS and JavaScript.

In terms of the web portal development, again MVC based architecture was used as the foundation. The web portal was powered by PHP, HTML, CSS and JavaScript. In addition to that, few design patterns were used in both the mobile and web developments such as singleton and adapter design patterns.

Both the portals were developed using the PhpStorm IDE. MySQL was used as the underlying database management system and Bit bucket was used for the version controlling.

In terms of the development methodology, a plan driven approach was used due to the pre-planned, well defined functionalities. However it should be noted that the development process was entirely based on the test driven development approach providing certain agile features to the development process as well.

B. Materials

In terms of the existing data which was used in the system, NDB API based data should be emphasized [1]. All the nutrition value calculations were done based on the nutrition values provided by the given API. The source is freely available for use after signing up for a free API key.

NDB API data format includes both JSON and XML. However JSON format was used in the current system. Out of the few different APIs they provide, Nutrition report API and Search API were used in the system.

In the Nutrition report API, not all the nutrition values were requested, as to reduce the transmission delay of the data. Only the nutrition values corresponding to lipids (nutrient id=204), calories (nutrient id=208), carbohydrates (nutrient id=205) and sugar (nutrient id=269) were used in the system. In the Search API a maximum of 25 items are requested.

C. The Algorithm

```
Start
    Open the add post window;
    If (takes photo from gallery) {
           Open the device gallery;
           Takes the photo;
           Verify the server URL;
           Transfer photo to the server ;}
    If (takes photo from camera) {
           Open the device camera;
           Takes the photo;
           Verify the server URL;
           Transfer photo to the server ;}
    Add title;
    While (ingredients to be add) {
            Search the ingredient;
            Add the ingredient;
            Set the amount;
            Set the type;
            Add to ingredient basket ;}
    Add the method of preparation;
    Save the post;
    If (title! =null && photo! =null) {
           Show confirmation;
           If (confirmation) {
                 Update the database;
                 Post the photo to public ;}
    } Else {Alert ("Post is not completed") ;}
    If (post cancel) {
            Delete the uploaded photo ;}
End
```

Fig.5. Pseudocode

From the code base, one of the important code segment is corresponding to the add post use case. This particular functionality is one of the most complex functionality as it involves dealing with the device's physical camera and gallery. Figure 5, illustrates the pseudo code involved with the regarded functionality.

As explained in the pseudo code itself, initially user has to open the relevant window. Then he has to take the photo from the gallery or from the camera. Both the scenarios involve checking the validity of the server and uploading the image in background to the server to reduce the uploading delay overhead. After adding the photo, user has to add the title and then ingredients of the meal with their amounts and type. This process needs to be done repeatedly for all the ingredients of the meal. After completing it, user has to enter the method of preparation of the dish. It should be noted that adding the ingredients and method of preparation is optional and hence can be skipped. After that, post can be posted. However in a case of cancelation of the post, the uploaded photo need to be removed from the server to avoid unwanted memory consumption in the server side.

D. Main Interfaces

The main interface of the mobile system is listed below. Figure 6, shows the sign up and home page which facilitates new user registration and news feed. Figure 7, shows the user profile and pattern tracking overview. Figure 8, shows the home page view of the web portal.

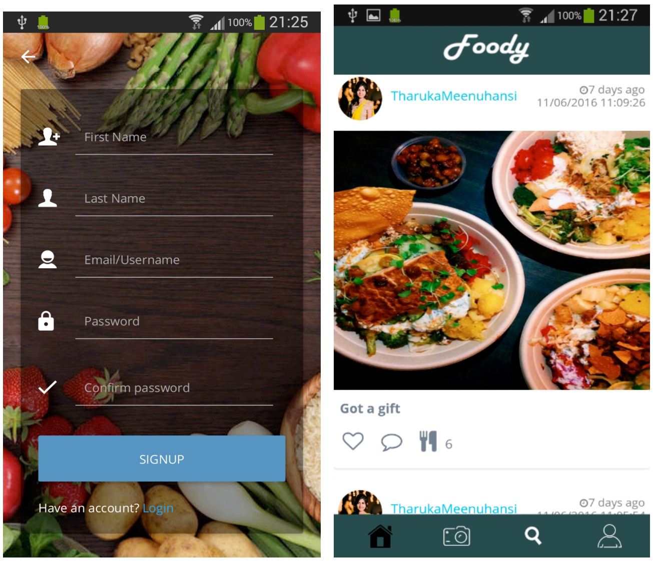


Fig. 6. Sign up and Home page

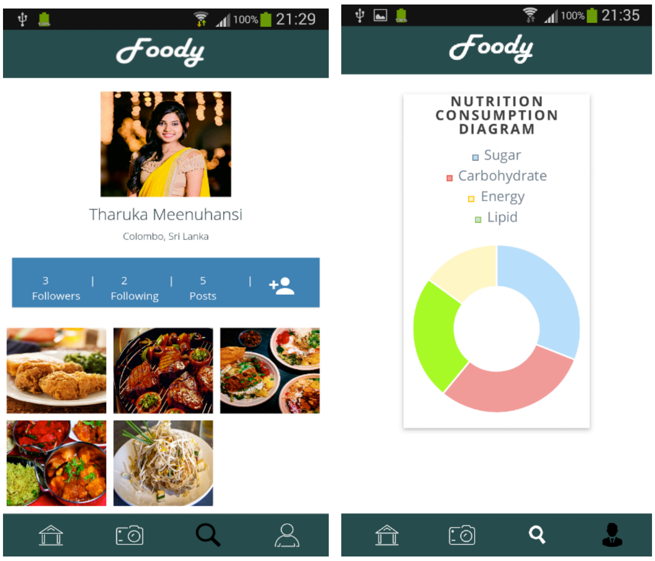


Fig. 7.User profile and View eating pattern

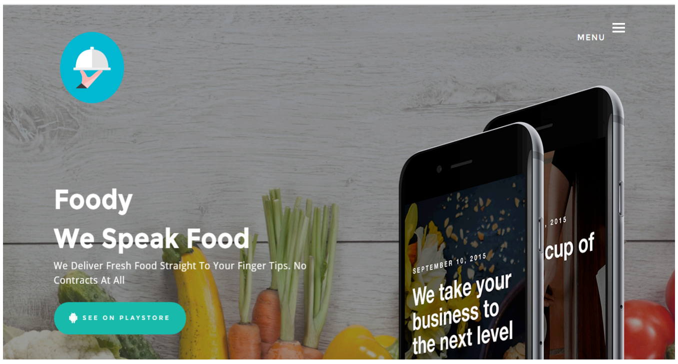

MENU
Foody
We Speak Food
We Deliver Fresh Food Straight To Your Finger Tips. No Contracts At All
SEE ON PLAYSTORE
SEPTEMBER 10, 2015
We take your business to the next level
2015
cup of

Fig. 8.Add dish

V. System Testing and Analysis

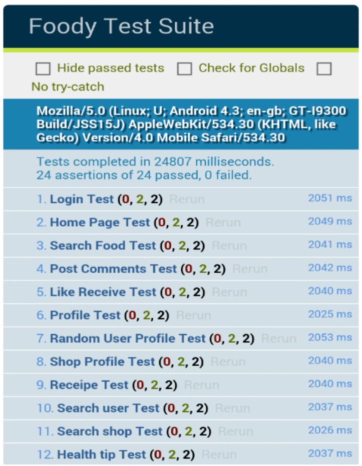
A. *Testing approach*

In terms of the testing, Foody is undergoing series of testing before it is actually deployed into the market. The main objective behind such kind of rigorous testing is to ensure relevant and expected quality of the product has been achieved. Having such motive on hand, the mission of the evaluation effort in the product can be briefed as follows: “Providing information to the project about the quality of the features of the solution and as well as ensuring the satisfactions of the stakeholders on their expectations.”

Testing plan of the system consists of several types of testing techniques. In order to ensure the system is not holding any data corruptions or incorrect functioning target behaviors due to the impact of the underlying database and data, a data and database integrity testing was conducted. To ensure systems’ functionality wise correct operation including navigation, data entry, processing and retrieval of data, functional testing was carried out using Qunit framework [10]. Apart from that to ensure that the UI provides the user with the appropriate access and navigation through the functions and objects within the UI, a comprehensive UI testing was carried out using the Selenium web driver [12]. A performance testing suit was carried out to ensure the performance requirements were achieved which are specified in the project requirements specification such as response times, transaction rates. For this purpose Qunit testing framework and JMeter [11] were used. To provide necessary application level security and system level security, Security and Access Control Testing was carried out. As the mobile application is expected to run on different mobile devices, a configuration testing was carried out.

B. *Unit Testing, Results and analysis of testing*

As “Foody” was developed based on a test driven development methodology, each unit, function and component is tested before it is actually put it in to the working system. Hence at the end of development process a complete unit test suite was developed using the Qunit framework. Figure 9, demonstrates the unit test suite execution window.



Foody Test Suite

□ Hide passed tests □ Check for Globals □ No try-catch

Mozilla/5.0 (Linux; U; Android 4.3; en-gb; GT-I9300 Build/JSS15J) AppleWebKit/534.30 (KHTML, like Gecko) Version/4.0 Mobile Safari/534.30

Tests completed in 24807 milliseconds.
24 assertions of 24 passed, 0 failed.

```
1. Login Test (0, 2, 2) Rerun                   2051 ms
2. Home Page Test (0, 2, 2) Rerun               2049 ms
3. Search Food Test (0, 2, 2) Rerun             2041 ms
4. Post Comments Test (0, 2, 2) Rerun           2042 ms
5. Like Receive Test (0, 2, 2) Rerun            2040 ms
6. Profile Test (0, 2, 2) Rerun                 2025 ms
7. Random User Profile Test (0, 2, 2) Rerun     2053 ms
8. Shop Profile Test (0, 2, 2) Rerun            2040 ms
9. Receipe Test (0, 2, 2) Rerun                 2040 ms
10. Search user Test (0, 2, 2) Rerun            2037 ms
11. Search shop Test (0, 2, 2) Rerun            2026 ms
12. Health tip Test (0, 2, 2) Rerun             2037 ms
```

Fig. 9. Unit test suit

C. Aspects related to performance, security, failures

In terms of the performance testing, using the JMeter and Qunit framework, throughput and the response time for each use case was tested to ensure the correct operation of the system. Throughput and deviation were used as an assertions on performance of the system.

In terms of the security testing, system was tested for the following criteria,

- Different types of actors with different access levels can access only those functions or data for which their user type is provided permissions.
- Actors with access to the system and applications are permitted to access them.

Apart from that to ensure the target-of-test can successfully failover and recover from variety of hardware, software or network malfunctions without loss of data or data integrity, series of simulations were carried out. These simulation series includes different types of possible failure conditions and it logs the exercising of the recovery processes (manual and automated) to restore the database, applications, and system to a desired, known, state.

VI. Conclusion

In brief, Foody is a platform which integrates the power of social networking with the need of healthy eating while being a powerful advertising media. The product under the stated requirements and constrains, has been developed completely and ready to be deployed to the market. However as future enhancements of the product following can be stated.

Currently the system does not have an individual user based notification system. As a system of this caliber should consists of such facility, within the near future it is expected to add that to the system. Apart from that, other popular social media like Facebook and Instagram is not integrated with the current system. As such integration could provide the user with the facility to switch between different social networks at minimum effort, such integration is expected to be added to the system. In the long run, the current system is expected to develop to a level which the system treats each user individually according to his/her own desires.

As a conclusion, it can be stated that Foody is already up and running and ready to Speak Food. Research suggests that the market condition is very much stable and ready to accept the new product as soon as it is officially launched. With an open market as such, the product Foody is expected to yield the expected outcomes within a short period of time and hence leading to a well-known product brand within the food loving generation of people.

References

[1] Agricultural Research Service, "National Nutrient Database for Standard Reference Release 28," United States Department of Agriculture, September 2015. [Online]. Available: https://ndb.nal.usda.gov/ndb/. [Accessed 12 March 2016].

[2] I. TwoGrand, "You Food," TwoGrand, Inc., [Online]. Available: http://www.twogrand.com/. [Accessed 16 June 2016].

[3] "Calorie Count," [Online]. Available: https://www.caloriecount.com/. [Accessed 13 October 2016].

[4] I. MyFitnessPal, "myfitnesspal," MyFitnessPal, Inc., [Online]. Available: https://www.myfitnesspal.com/. [Accessed 13 October 2016].

[5] M. Inc., "MyNetDiary," MyNetDiary Inc., [Online]. Available: http://www.mynetdiary.com/. [Accessed 13 October 2016].

[6] "ChooseMyPlate.gov," USDA Center for Nutrition Policy and Promotion, [Online]. Available: https://www.choosemyplate.gov/. [Accessed 13 October 2016].

[7] "Instagram," Instagram, [Online]. Available: https://www.instagram.com/. [Accessed 13 October 2016].

[8] A. S. Inc., "Adobe PhoneGap," Adobe Systems Inc., [Online]. Available: http://phonegap.com/. [Accessed 13 October 2016].

[9] A. S. Foundation, "Apache Cordova," Apache Software Foundation, [Online]. Available: https://cordova.apache.org/. [Accessed 13 October 2016].

[10] T. j. Foundation, "QUnit," The jQuery Foundation, [Online]. Available: https://qunitjs.com/. [Accessed 13 October 2016].

[11] A. S. Foundation, "Apache JMeter," Apache Software Foundation, [Online]. Available: http://jmeter.apache.org/. [Accessed 13 October 2016].

[12] Selenium, "SeleniumHQ," [Online]. Available: http://docs.seleniumhq.org/ [Accessed 13 October 2016].