****

**FINAL YEAR PROJECT REPORT**

**ONLINE SHOPPING SYSTEM FOR ORGANIC PRODUCTS**

**CHOK KAI XIAN (1402971)**

**LEE KONG CHIAN FACULTY OF ENGINEERING AND SCIENCE**

**UNIVERSITI TUNKU ABDUL RAHMAN**

**JAN 2018**

**TABLE OF CONTENTS**

**TABLE OF CONTENTS 1**

**LIST OF FIGURES 3**

**CHAPTER 1 INTRODUCTION 4**

* 1. Background of The Problem 4
  2. Problem Statement 4
  3. Project Objectives 5
  4. Proposed Solution 5
  5. Proposed Approach 7

1.6 Project Scope7

**CHAPTER 2 LITERATURE REVIEW 9**

2.1 Introduction 9

2.2 PHP MVC Framework for Web Development 9

2.2.1 Benchmarking 11

2.2.1.1 System Load Average 11

2.2.1.2 Amount of Functions Call 11

2.2.1.3 Request per Second 12

2.2.1.4 Response Time 13

2.2.1.5 Memory Usage 13

2.2.1.6 Number of Files 14

2.2.2 Specifications 15

2.2.3 Results and Discussion 15

2.3 Software Development Methodologies 16

2.4 Online Shopping System 18

2.4.1 Web-based and Android-based Online Shopping System 18

2.4.2 E-commerce System (MIK Online Shop) 20

**CHAPTER 3 METHODOLOGY AND WORK PLAN 24**

3.1 Approach and Methodologies 24

3.2 Work Plan 24

3.2.1 WBS 24

3.2.2 Gantt Chart 26

**CHAPTER 4 PROJECT INITIAL SPECIFICATIONS 28**

4.1 Non-functional Requirements 28

4.1.1 Security Requirements 28

4.1.2 Availability 28

4.1.3 Compatibility 28

4.1.4 Usability 28

4.2 Functional Requirements 28

4.2.1 User Account/Profile 28

4.2.2 Customers 28

4.2.3 Administrators 29

4.3 Use Case Diagram 30

**REFERENCES 31**

**APPENDICES** **33**

Appendix A: Use Case Description 33

Appendix B: Questionnaire 40

**LIST OF FIGURES**

Figure 1.1 Laravel Framework Architecture 6

Figure 2.1 MVC Architecture 10

Figure 2.2 Average system load time 11

Figure 2.3 Amount of function called 12

Figure 2.4 Request per second 12

Figure 2.5 Response time 13

Figure 2.6 Memory Usage 14

Figure 2.7 Number of files needed 14

Figure 2.8 Modern (agile) web development process 17

Figure 2.9 System Homepage 19

Figure 2.10 User shopping cart 19

Figure 2.11 MIK Online Shop Homepage 21

Figure 2.12 Add Category Page for admin 21

Figure 2.13 Product Details Page 22

Figure 2.14 PayPal purchase form 22

Figure 3.1 WBS Page 1 25

Figure 3.2 WBS Page 2 25

Figure 3.3 Gantt Chart Page 1 26

Figure 3.4 Gantt Chart Page 2 27

Figure 4.1 Use Case Diagram 30

* 1. **INTRODUCTION**
  2. **Background of The Problem**

Alertness of the devastation of organic materials has elevated the problem of environmental conservation. “Green consumerism”, an ecological consumption will then be built. (Moisander, 2007). There are more consumers have considered for greener products as they are more people interested to buy them (Nimse et al., 2007). There are many benefits of online shopping such as it will be more convenient to shop online and there is no crowd of people (Jusoh and Ling, 2012). A solution to the problem is to create a web-based online shopping system for organic products to save customers’ time.

* 1. **Problem Statement**

**1.2.1 Uncertainty of genuineness of the organic products**

Currently there are many online shopping system in the market like Lazada, Lelong.my, 11street and so on in the market in Malaysia. However, most of them are selling products not by their own but from other retailers which sometimes couldn’t guarantee the product quality. The genuineness of the organic products is very important as there are people who are only willing to consume organic products but not other inorganic food. An online shopping system for organic products is necessary to assure to the customers that the products that are selling are 100% organic.

* + 1. **Inconvenient to shop in organic shops**

We want everyone to be able to buy organic products without having to leave their houses. Today there are lacking of online shopping system of organic products in Malaysia. Online shopping is good to most of the customers because it is convenient and easy to access (Wolfinbarger and Gilly, 2001) as consumers can do online shopping any time even at their home comfortably and save their time as well as effort. An online organic shop with delivery services will be a good service to make the customers to be convenient on buying organic food without leaving their home.

* + 1. **Hard to categorize the items and make comparison**

It’s good to have all the products in the shop to be categorized so that it will allow the customers to compare the products in terms of price, quality or other perspectives easily. However it will be nearly impossible for the organic shops to categorize their items in a detailed way as one item could be in 2 or more categories. In order to settle this, there will be a feature to search for different organic items in different categories and the details of the products will be shown in a way for customer to compare them easily.

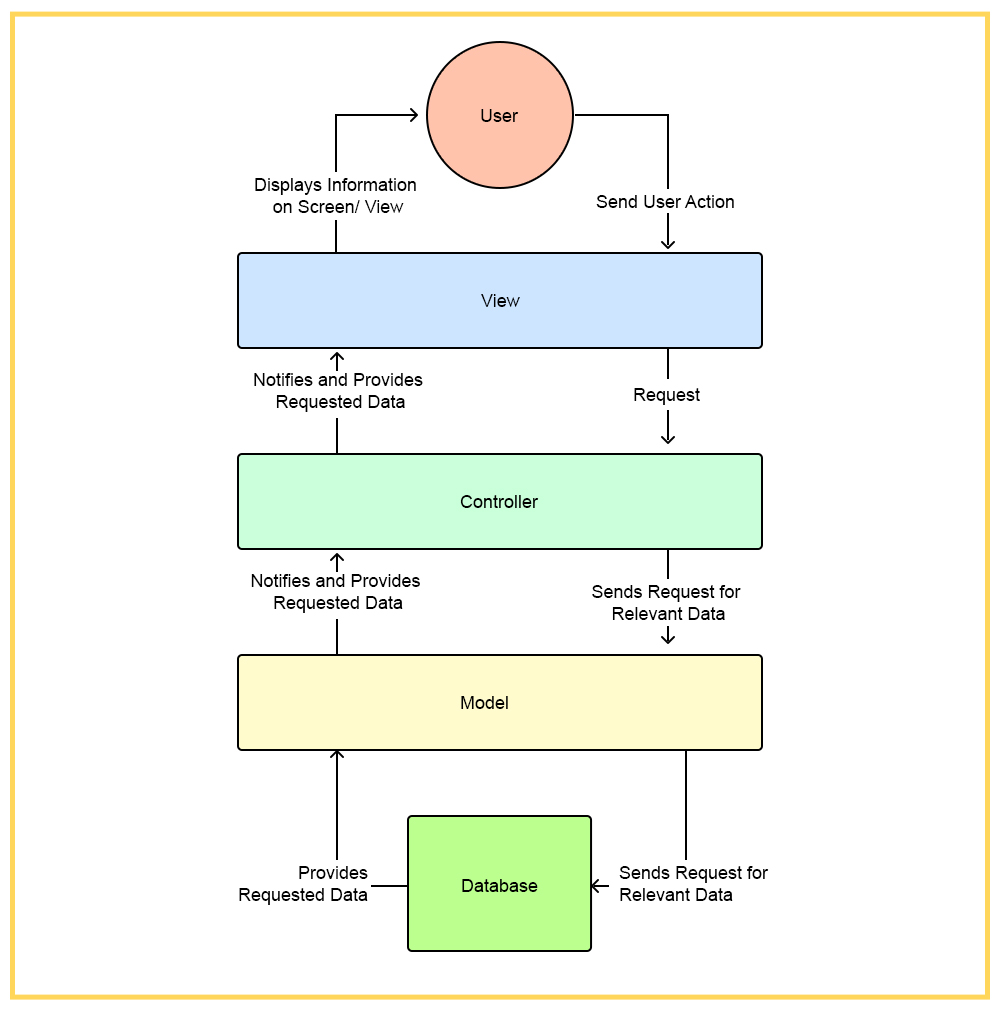
* + 1. **Freshness of the organic products is unsure**

All of the customers wish to receive the organic products from online shop is as fresh as they bought from those local organic shops. Unfortunately there will always be the time that the products that you received will be rotten or expired. Some of the online shops offered to return funds or exchange for products if the products are unsatisfied but it will be too troublesome if this happen for many times. A way to solve it is to allow the customers to make payment on the items after they have received them from the courier service and examined on the products they have ordered.

* 1. **Project Objectives**
* Examine the existing feature used in the online shopping system that fits with the needs of the organic products consumers.
* Develop the web application that consists of the feature and design required by the organic products consumers with the implementation using suitable approach and methodology.
  1. **Proposed Solutions**

There a few potential solutions that might be worked against the issues that have stated above and also achieve the project objectives.

Firstly, develop a web-based system for online shopping using PHP Laravel Framework. Laravel is a PHP framework that has beautiful syntax. Using Laravel is fun as it is easy to use. It helps to reduce to workload of developers by easing usual tasks that are used in most of the web development. The aim of Laravel is to allow developers develop the functionality of the system while having fun in coding. It also includes all the advantages in other frameworks. The diagram below shows the architecture of Laravel framework.



**Figure 1.1 Laravel Framework Architecture**

Secondly, MySQL will be used as the system database as MySQL a well-known database management system. The data is collected in structure. It allows user to store, modify, update, and delete data from the database. PHPMyAdmin is used to handle the administration of MySQL via the internet.

* 1. **Proposed Approach**

The purposed approach for this project is the Agile and Extreme Programming (XP) methodology. Recently, the use of agile methodology is getting more and more for all scale of projects because the web application grow rapidly (Goto, Tsuchida and Nishino, 2014). Agile consists of a group of software development methodologies. By implementing agile methodology, the whole project is broken down into small functions. Each of them is developed from design to testing stage iteratively. Thus, the functions can be done in a quick time to let the clients to review on them at the early stage (Goto, Tsuchida and Nishino, 2014). Agile is popular in web application development because it allows small team formation, encourage communicating face-to-face as well as develop prototype in a quick manner (Kumar, Sharma and Gupta, 2016).

Extreme Programming (XP) is suitable for small development team which contains not more than 15 members according to Kumar, Sharma and Gupta (2016). By applying this approach, documentation can be reduced because it encourages face-to-face communication rather than writing on documents. This helps the team to focus more on coding and automated testing and make the development process faster. The disadvantage of XP is when the team is getting larger where effective communication couldn’t be applied. There are also other shortcomings caused by the characteristics of XP which is rapid iterations, fast introduction, and also lack of documentation. However, the team can be adjusted in order to solve the issues.

* 1. **Project Scopes**

The web-based shopping system will include at least these features:

* Provide information about various type of organic products in different categories.
* Allow customers to login to the system and make payment online.
* Allow customers to compare similar products.
* Helpdesk service is provided if the customers has any problems.
* Allow admins to login as well as access and manage confidential client information.
* Notify customers with notification email for related activities.
* Allow customers to save products to their wish list.
* Provide cash on delivery option.

**CHAPTER 2 LITERATURE REVIEW**

**2.1 Introduction**

This chapter reviews the literature of the proposed online shopping system, various types of methodology and PHP frameworks. The first part makes comparison of different MVC design framework and evaluate their functionality which best fit the system. Afterwards, different kind of methodologies are discussed in term of their strengths and weaknesses in web development. Lastly, some existing online shopping systems are investigated and the issues in the systems is evaluated in order to understand and improve our system.

**2.2 PHP MVC Framework for Web Development**

PHP is an open source and flexible scripting language which is intuitive, easy to setup, compiled fast, cross platform. PHP has become the famous web development language and is being widely-used. While developing web applications, some developers choose to assemble all the issues such as data representation, business logic, and data access layer which consequently create troubles. Model View Controller (MVC) design pattern is thus implemented in development which can effectively separate code with multi layers with different kind of activities from each other (Olanrewaju, Islam and Ali, 2015). MVC design pattern is made up of three parts which is model, view and controller. Model is on the lowest level which take care of data access and maintain it. View is responsible for data representation by displaying data to user. Controller handles the interaction between the View and the Model.

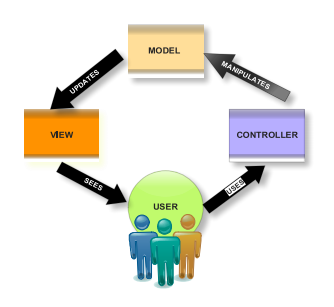


Figure 2.1 MVC Architecture

(Olanrewaju, Islam and Ali, 2015)

There are few languages which implement MVC design pattern such as CakePHP, Symfony, CodeIgniter and Laravel. Performance of each of them is evaluated at below.

CakePHP

The main goal of CakePHP is to allow developers to create web-based system rapidly and also in a flexible way. It provides various tools to help the programmers to kick start the development process, which could help to reduce the workload by avoid reinventing new tools to use.

Symfony

Symfony framework by French company Sensio Labs offers a lot of reusable components, libraries, and packages. With it, a web developer may deal with tasks like object configuration, authentication, templates, etc. Packed with lots of features, it is not too overwhelming to get familiar with, too. In terms of structure, it is a flexible full stack and uses Twig as a template engine.

CodeIgniter

It is a framework with a small size, which only has 2MB. It is a toolkit that is easy and elegant as well for programmers to create a complete web app.

Laravel

Laravel is a PHP framework that make the development process is regulated, automatically handling several non-business logic relationship so that developers can be aim his attention at implementation of business logic. It is said that web design based on Laravel framework which consists automated processing for part of the design, allows efficiently developing as it has scalability and robust scalability (Yu, 2015).

**2.2.1 Benchmarking**

Several factors are being investigated such as system load average, amount of function calls, request per second, response time, memory usage, and number of files needed on each framework. Web design which consist of “hello world” was conducted on apache (ab –c 200 –n 50000) to evaluate the performance of each PHP framework (Olanrewaju, Islam and Ali, 2015).

**2.2.1.1 System Load Average**

Figure 2.2 shows the system load time within one minute. Based on the graph, Laravel has the lowest times (0.98) which is the best compared to others (Olanrewaju, Islam and Ali, 2015).

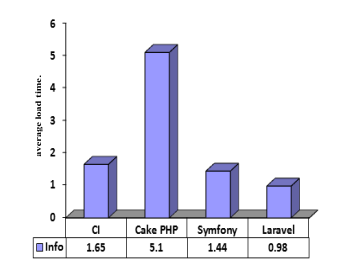
****

Figure 2.2 Average system load time

(Olanrewaju, Islam and Ali, 2015).

**2.2.1.2 Amount of functions calls**

Number of functions called for 1 “hello world” page is checked in this test. Laravel is most effective as it has the least number of function calls with 238 calls in Figure 2.3 (Olanrewaju, Islam and Ali, 2015).

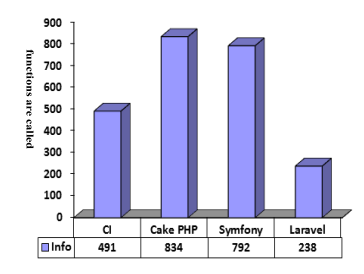


Figure 2.3 Amount of function called

(Olanrewaju, Islam and Ali, 2015)

**2.1.1.3 Request per second**

Figure 2.4 shows that Laravel outperforms the others as it was able to manage 3000 request per second. Hence it is the most effective one as bigger output means better performance (Olanrewaju, Islam and Ali, 2015).

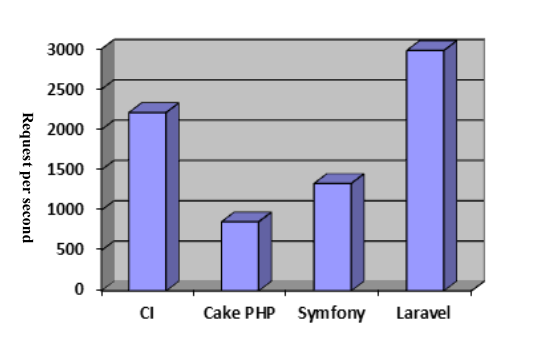


Figure 2.4 Request per second

(Olanrewaju, Islam and Ali, 2015)

**2.2.1.4 Response Time**

The response time is measured by how many millisecond of page request to response from framework. The figure below indicates that Laravel is the fastest as it has the least response time which is 4.46 milliseconds (Olanrewaju, Islam and Ali, 2015).

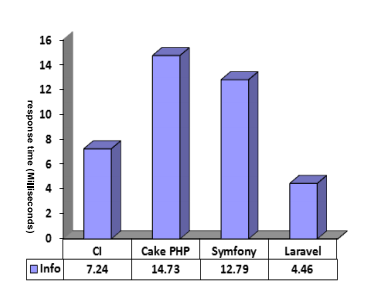
****

Figure 2.5 Response time

(Olanrewaju, Islam and Ali, 2015)

**2.2.1.5 Memory Usage**

This test checks how much memory is utilized in one word “hello world” page. As Figure 2.6 shows that the memory usage in Laravel is lowest (518.36KB), it means that Laravel is more memory usage efficient than others (Olanrewaju, Islam and Ali, 2015).

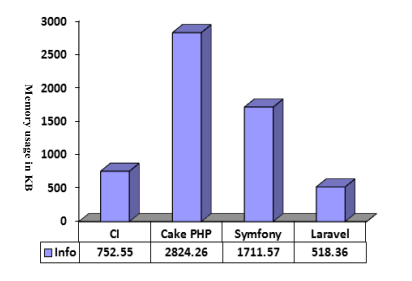
****

Figure 2.6 Memory Usage

(Olanrewaju, Islam and Ali, 2015)

**2.2.1.6 Number of Files**

This benchmarking checks the total amount of files needed to be loaded when running the one “hello world” page. As shown in Figure 2.7, Symfony comes up with the least files of 15, which is better when compared to others (Olanrewaju, Islam and Ali, 2015).

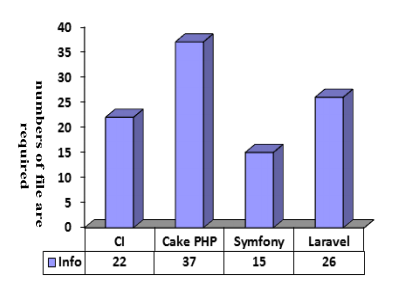
****

Figure 2.7 Number of files needed

(Olanrewaju, Islam and Ali, 2015)

**2.2.2 Specifications**

A research is conducted among these four famous PHP frameworks based on their different core criteria. The table below indicates the difference between the frameworks in terms of facilities. By analyzing the table carefully, we can know that the Laravel framework has more advantages over other frameworks (Olanrewaju, Islam and Ali, 2015).

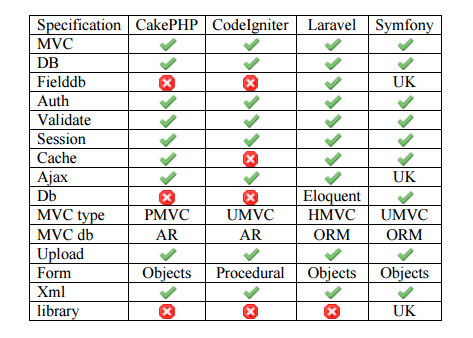


Table 2.1 Comparison among four PHP frameworks

(Olanrewaju, Islam and Ali, 2015)

**2.2.3 Results and Discussion**

The results acquired from analyzing the four frameworks based on the criteria such as system load average, number of function calls, request per second, response time, memory usage and number of files needed as well specification of each framework indicates that the Laravel PHP framework outperforms other MVC frameworks. 3000 request per second was recorded for Laravel. The outputs obtained for all other criteria shows that Laravel is flexible in developing web application as well as contain more functions for developers that makes it usable to most of the web developers. Easy migration, eloquent ORM, enriched library, template system and good community support that helps to build web application easily. Thus, it is recommended that Laravel is the best choice which is suitable for building web application (Olanrewaju, Islam and Ali, 2015).

**2.3 Software Development Methodologies**

Below discusses some of these traditional methodologies, as well as explain their strengths and weaknesses in web development.

Waterfall Model

The waterfall methodology involves a set of sequential process only allow small degree of repetition between each stage. The major issue with adapting the waterfall methodology web development is the structure rigidity and lack of iteration between each stage (Howcroft and Carroll, 2000) because the developer cannot go back to the previous phase it is a cascading process. Therefore, the modification will be very complex to execute if there are issues occurred in previous stage (Shaffi and Al-Obaidy, 2013). The web is an environment that move in a fast pace and new technologies are keep being invented thus, methodology that is used for the website development must have enough flexibility to cope with any modifications (Howcroft and Carroll, 2000).

Prototyping Model

It is designed to address the issues with waterfall model. The idea of this model is that using the throwaway prototype requirements of the development understands on the basis of currently known requirements rather than fixing the requirements before proceeding to coding or design phase. Its development includes designing, coding and testing that are not done formally or completely. Thus using this methodology, end users can feel the actual system, an interaction with the prototype will makes user to understand about the requirements more. It is effective when the user is more focus on the user interfaces than the development processes of the system (Howcroft and Carroll, 2000).

Rapid Application Development

The essence of RAD is its flexibility to adapt to changing customer vision throughout the development cycle. Its starts by defining a loose set of requirements, so developers get an idea of what the product needs to achieve. The aim of it is to enable the system to be developed or produced much quicker. (Shaffi and Al-Obaidy, 2013). Powell, Jones and Cutts (1998) discuss the failure of adapting IS development methodologies in web development process and pointed that the RAD methodology is the worst as it involves constructing the wrong site several times until the right site appear.

Agile and Extreme Programming (XP) Methodology

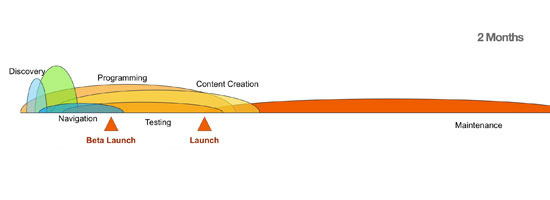


Figure 2.8 Modern (agile) web development process.

(Six Revisions, 2011)

Recently, the use of agile methodology is getting more and more for all scale of projects because the web application grow rapidly (Goto, Tsuchida and Nishino, 2014). Agile consists of a group of software development methodologies. By implementing agile methodology, the whole project is broken down into small functions. Each of them is developed from design to testing stage iteratively. Thus, the functions can be done in a quick time to let the clients to review on them at the early stage (Goto, Tsuchida and Nishino, 2014). Agile is popular in web application development because it allows small team formation, encourage communicating face-to-face as well as develop prototype in a quick manner (Kumar, Sharma and Gupta, 2016).

Extreme Programming (XP) is suitable for small development team which contains not more than 15 members according to Kumar, Sharma and Gupta (2016). By applying this approach, documentation can be reduced because it encourages face-to-face communication rather than writing on documents. This helps the team to focus more on coding and automated testing and make the development process faster. The disadvantage of XP is when the team is getting larger where effective communication couldn’t be applied. There are also other shortcomings caused by the characteristics of XP which is rapid iterations, fast introduction, and also lack of documentation. However, the team can be adjusted in order to solve the issues.

**2.4 Online Shopping System**

This section talks about several online shopping systems which are proposed by others. Their objectives of developing the systems has a similarity which is to allow their users to buy their products virtually from the internet without needing to travel to the real store. The proposed system in this project will have the same goal with those projects.

**2.4.1 Web-based and android-based online shopping system**

This project is conducted to develop a web-based online shopping system for an existing store as well as in android platform. The main goal of the system is to allow customers to buy items from the store via the Internet. The system is accessible anywhere as the application is available on mobile. It also provides delivery after the order has been made to allow customers to pick up their products at their door which is more convenient for them rather than travelling to the departmental store. The system consists of 3 types of users which are admin, moderator and customer. Each of them has different privileges when using the application. Below are some samples of the system user interfaces.

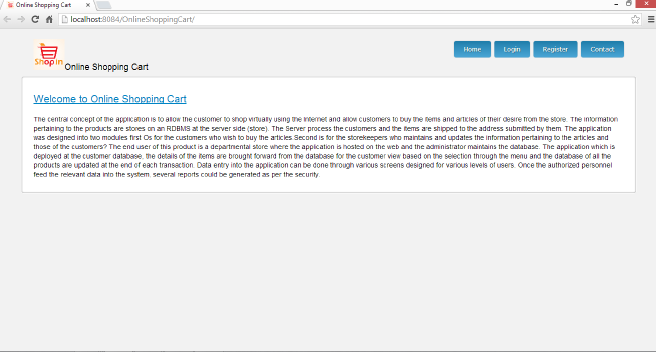


Figure 2.9 System Homepage

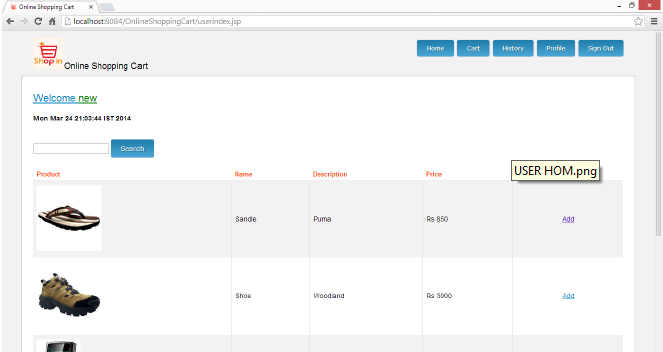


Figure 2.10 User shopping cart

In my opinion, I think the system is good on separating few types of users and gives them different rights on managing the system. This could avoid mixing up the rights of each type of user. However, there are several problems that I think the system could have done better. Firstly, the system isn’t implementing any PHP framework while developing which causes the development process to be far slower than those which are using PHP framework like Laravel as those frameworks implemented a set of useful libraries for basic web operations. Moreover, the user interfaces of the system isn’t attractive enough to grab the attention of the customers. It can be improved by creating a more colorful or neat user interfaces as well as with more pictures and less words. Lastly, the system didn’t save the shipping addresses of the customers in their profiles but expecting them to key in the addresses whenever they make an order. This is very troublesome to the users and might not able to retain the interest of customer to keep using this application.

**2.4.2 E-commerce System (MIK Online Shop)**

The purpose of this system is to solve the problem that customer can’t examine the products they wish to buy like when they go to departmental stores as the details of the products couldn’t be checked via the Internet. Thus, this system emphasizes its goal on providing as much accurate information of the products as possible to the customer by supporting a higher resolution image or video of the selling products which leads to higher reliability of product quality when purchasing. Figures below shows some of the screen layouts of this system.

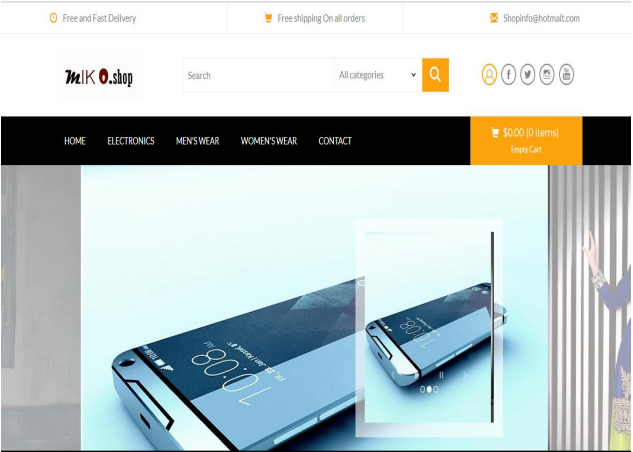


Figure 2.11 MIK Online Shop Homepage

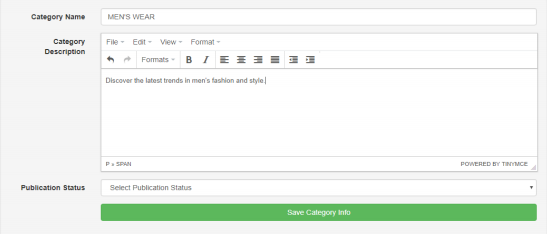


Figure 2.12 Add Category Page for admin

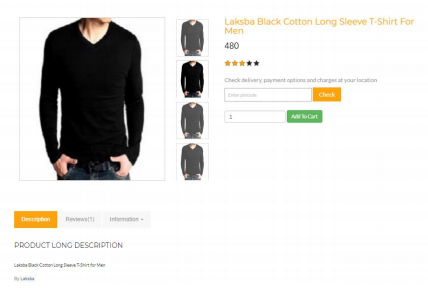


Figure 2.13 Product Details Page

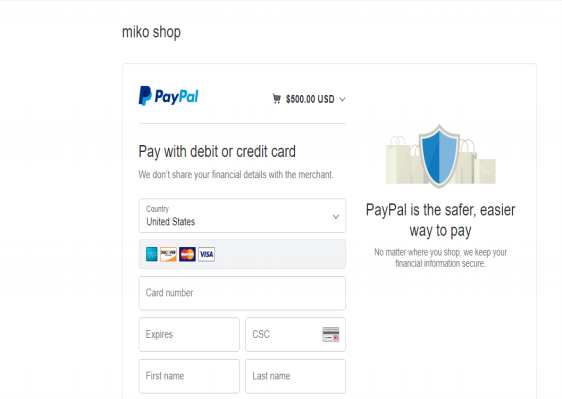


Figure 2.14 PayPal purchase form

Based on my thought, I think the system is better than the system proposed by Chittil, Chittil and Rishkese (2014). Firstly, it has a much better design of user interfaces which looks more attractive and neat compared to Chittil’s proposed system as the contrast of the color is sharper. It also allows customers to key in the shipping address and save it into database so it can be retrieved whenever user makes any purchases. Admins of the websites are allowed to add more categories in order to group the products in a more detailed way so that customers can search for the desired products more easily. When adding products to the database, admin also required to enter much information of the products and thus, customers can understand more about the products’ details. Furthermore, payment with PayPal is also allowed in this system which is convenient to those who are a PayPal user.

However, there are some shortcomings in this system. Waterfall model is implemented while developing the system. Waterfall model is proved that it is not suitable for website development as high flexibility of the development methodology that can cope with changes is necessary because web environment is moving fast and new technologies are being invented rapidly. Besides, customers aren’t allow to give rating to the products, which is quite a disadvantage as rating of the products can give confidence and as a reference for other customers who are interested to buy the same product. On the other hand, comparison among similar products isn’t provided as well. This is a good feature for customers to make a better decision before making an order.

**CHAPTER 3 METHODOLOGY AND WORK PLAN**

**3.1 Approach and Methodology**

Agile methodology divides the projects into parts and develop each of them in a short iteration (Srivastava, 2017). As the primary goal of the project is to develop a web application, this lightweight development methodology is chosen. Nada (2015) states that agile is a good approach for web development if users wanted to get the review of the small parts in early stage. There are many people using waterfall model or spiral model lastly, but those require many resources such as skill and staff. Therefore, it is not suitable for projects that only have a developer (Tarawneh, 2012). There are a lot of agile manifesto like XP, SCRUM, Feature Driven Development, Lean and more. The reason that XP is chosen for this project is because of its programming practices definition as well as it is more detailed than others, where the others focus on the planning phase and project management (En.wikiversity.org, 2017).

**3.2 Work Plan**

**3.2.1 Work Breakdown Structure (WBS)**

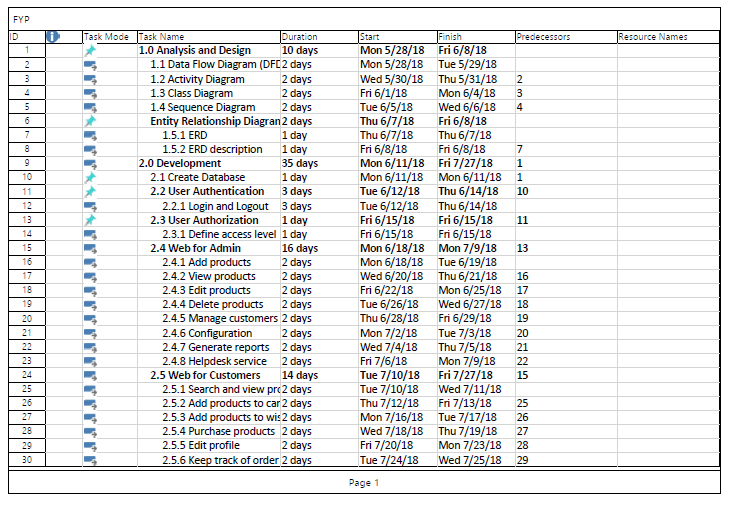
****

Figure 3.1 WBS Page 1

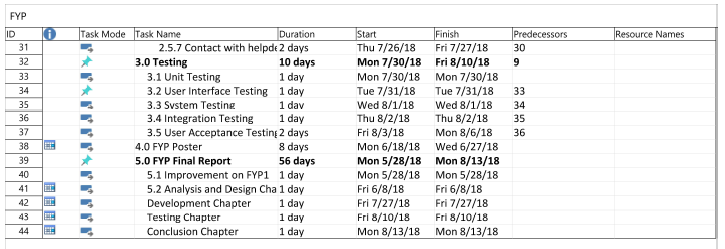
****

Figure 3.2 WBS Page 2

**3.2.2 Gantt Chart**

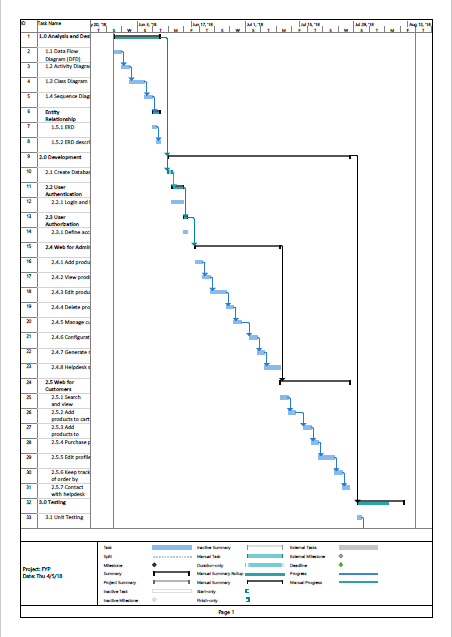
****

Figure 3.3 Gantt Chart Page 1

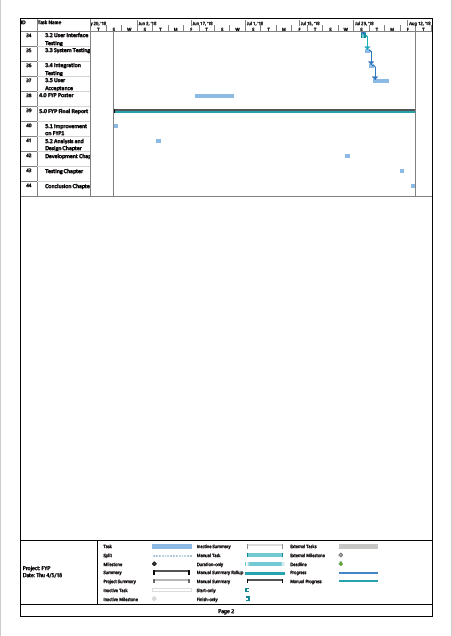
****

Figure 3.4 Gantt Chart Page 2

**CHAPTER 4 PROJECT INITIAL SPECIFICATION**

**4.1 Non-Functional Requirements (NFRs)**

**4.1.1 Security**

* Administrators and customers will have different access level to the web application. Customers can only view and purchase products as well as edit personal information. Administrators can manage products and users.
* All login passwords must contains at least 8 characters including letters and numbers.

**4.1.2 Availability**

* System must be accessible for customers for 24 hours.

**4.1.3 Compatibility**

* The web application must be compatible for all web browsers.

**4.1.4 Usability**

* The user interface elements must be easy to understand and easy to use.
* Error messages should be shown to users including solutions for the errors.
* Confirmation for the actions that cannot be reverted should be asked.

**4.2 Functional Requirements**

**4.2.1 User Account/Profile**

* Users shall be allowed to register and login.
* Users shall be allowed to edit their profile and change the passwords.

**4.2.2 Customers**

* Customers shall be allowed to search products by categories or keywords.
* Customers shall be allowed to view products details including images.
* Customers shall be allowed to add or remove products to cart.
* Customers shall be allowed to add or remove products to wish list.
* Customers shall be allowed to purchase products inside the cart.
* Customers shall be allowed to compare similar products.
* Customers shall be allowed to keep track their orders.

**4.2.3 Administrators**

* Administrators shall be allowed to configure the system such as types of categories, method to purchase an order and more.
* Administrators shall be allowed to add new products.
* Administrators shall be allowed to edit existing products.
* Administrators shall be allowed to delete existing products.
* Administrators shall be allowed to generate reports of the data.

**4.3 Use Case Diagram**



Figure 4.1 Use Case Diagram

**Reference**

1. Jusoh, Z.M. and Ling, G.H., 2012. Factors influencing consumers’ attitude towards e-commerce purchases through online shopping. *International Journal of Humanities and Social Science*, *2*(4), pp.223-230.
2. Wolfinbarger, M. and Gilly, M.C., 2001. Shopping online for freedom, control, and fun. *California Management Review*, 43(2), pp. 34‐55.
3. Moisander, J., 2007. Motivational complexity of green consumerism. *International Journal of Consumer Studies*, 31(4), pp.404-409.
4. Nimse, P., Vijayan, A., Kumar, A. and Varadarajan, C., 2007. A review of green product databases. *American Institute of Chemical Engineers*, 26(2), pp. 131‐7.
5. Olanrewaju, R.F., Islam, T. and Ali, N., 2015. An empirical study of the evolution of php mvc framework. In *Advanced Computer and Communication Engineering Technology* (pp. 399-410). Springer, Cham.
6. Yu, H.R., 2015. Design and implementation of web based on Laravel framework. *Atl. Press, no. Iccset*, *2014*, pp.301-304.
7. Wojtkowski, W.G., Wojtkowski, W., Wrycza, S. and Zupancic, J. eds., 2012. *Systems development methods for the next century*. Springer Science & Business Media.
8. Powell, T.A., Jones, D.L. and Cutts, D.C., 1998. *Web site engineering: beyond Web page design*. Prentice-Hall, Inc.
9. Howcroft, D. and Carroll, J., 2000. A proposed methodology for Web development. *ECIS 2000 Proceedings*, p.73.
10. Shaffi, A.S. and Al-Obaidy, M., 2013. Analysis and comparative study of traditional and web information systems development methodology (WISDM) towards Web development applications. *International Journal of Emerging Technology and Advanced Engineering*, *3*(11), pp.277-282.
11. Chittil, S., Chittil, N. and Rishkese, M., 2014. Online Shopping System.
12. Six Revisions, 2011. *Agile Web Development That Works.* Available at: https://www.webpagefx.com/blog/web-design/agile/ (Accessed: 1 April 2017).
13. En.wikiversity.org (2017) *Agile software development - Wikiversity*. Available at: https://en.wikiversity.org/wiki/Agile\_software\_development (Accessed: 1 April 2017).
14. Tarawneh, M. Al (2012) ‘*Agile Development Methods for Developing Web Application in Small Software Firms*’. Available at: http://www.academia.edu/4218708/Agile\_Development\_Methods\_for\_Developing\_Web\_Application\_in\_Small\_Software\_Firms (Accessed: 18 July 2017).
15. Goto, T., Tsuchida, K. and Nishino, T. (2014) ‘EPISODE: An Extreme Programming Method for Innovative Software Based on Systems Design’, in *2014 IIAI 3rd International Conference on Advanced Applied Informatics*. IEEE, pp. 780–784.
16. Kumar, S. R., Sharma, R. and Gupta, K. (2016) ‘Strategies for web application development methodologies’, in *2016 International Conference on Computing, Communication and Automation (ICCCA)*. IEEE, pp. 160–165.
17. Srivastava, B. (2017) *What is Agile methodology?Disadvantage of waterfall model.* Available at: https://www.linkedin.com/pulse/what-agile-methodologydisadvantage-waterfall-model-bikesh-srivastava (Accessed: 18 July 2017).

**APPENDICES**

**Appendix A: Use Case Description**

**Register**

|  |  |  |
| --- | --- | --- |
| Use Case Name: Register | ID:001 | Importance Level: Medium |
| Primary Actor: Administrators and Customers | Use Case Type: Overview, Essential | |
| Stakeholders and Interests:  Administrators – Administrators should be able to register.  Customers – Customers should be able to register. | | |
| Brief Description: New administrators and customers should be able to register as new user by providing specific personal information. | | |
| Trigger:  When the user is new to the system. | | |
| Normal Flow of Events:   1. Administrators or customers click the register button. 2. Administrators or customers fill in information needed. 3. Administrators or customers confirm the registration. | | |
| Alternate/Exceptional Flows: | | |

**Login/Logout**

|  |  |  |
| --- | --- | --- |
| Use Case Name: Login/Logout | ID:001 | Importance Level: Medium |
| Primary Actor: Administrators and Customers | Use Case Type: Overview, Essential | |
| Stakeholders and Interests:  Administrators – Administrators should be able to login or logout.  Customers – Customers should be able to login or logout. | | |
| Brief Description: Administrators and customers should be able to login to their account as well as logout. | | |
| Trigger:  When existing users navigate to the system. | | |
| Normal Flow of Events:   1. Administrators or customers enter their id and password. 2. Administrators or customers click the login button. 3. Administrators or customers click the logout button to logout. | | |
| Alternate/Exceptional Flows: | | |

**Search and view products**

|  |  |  |
| --- | --- | --- |
| Use Case Name: Search and view products | ID:001 | Importance Level: High |
| Primary Actor: Customers | Use Case Type: Overview, Essential | |
| Stakeholders and Interests:  Customers – Customers should be able to search the products and view the details of the products. | | |
| Brief Description: Customers should be able to search the products by keyword or categories and view the details of them such as price, description, size and more. | | |
| Trigger:  When customers want to find the products they interested. | | |
| Normal Flow of Events:   1. Customers enter the keyword in the search field. 2. Customers click the search button. 3. System generates results based on the keyword. 4. Customers click on the products to view the details of it. | | |
| Alternate/Exceptional Flows:   1. Customers click the categories tab and choose the desired category. 2. The system show the products that fall into that category. 3. Customers click on the products and view the details of it. | | |

**Add/Remove products to/from cart**

|  |  |  |
| --- | --- | --- |
| Use Case Name: Add or remove products to or from cart | ID:001 | Importance Level: Medium |
| Primary Actor: Customers | Use Case Type: Overview, Essential | |
| Stakeholders and Interests:  Customers – Customers should be able to add or remove products to or from cart. | | |
| Brief Description: Customers should be able to add products to cart or remove products from it before purchasing. | | |
| Trigger:  When customers want to buy the desire products. | | |
| Normal Flow of Events:   1. Customers click the add cart button to add the products to cart. | | |
| Alternate/Exceptional Flows:   1. Customers click the ‘x’ button in the cart to remove it from cart. | | |

**Add/Remove products to/from wish list**

|  |  |  |
| --- | --- | --- |
| Use Case Name: Add or remove products to or from wish list | ID:001 | Importance Level: Medium |
| Primary Actor: Customers | Use Case Type: Overview, Essential | |
| Stakeholders and Interests:  Customers – Customers should be able to add or remove the products to or from the wish list. | | |
| Brief Description: Customers should be able to add the products to wish list or remove unwanted products from it. | | |
| Trigger:  When customers interested on the products but not going to purchase. | | |
| Normal Flow of Events:   1. Customers click the “Star” button to add the products to wish list. | | |
| Alternate/Exceptional Flows:   1. Customers click the ‘x’ button in the wish list to remove unwanted products. | | |

**Purchase orders**

|  |  |  |
| --- | --- | --- |
| Use Case Name: Purchase orders | ID:001 | Importance Level: High |
| Primary Actor: Customers | Use Case Type: Overview, Essential | |
| Stakeholders and Interests:  Customers – Customers should be able to purchase the orders. | | |
| Brief Description: Customers should be able to purchase the products inside the cart. | | |
| Trigger:  When customers confirm to buy all the products in the cart. | | |
| Normal Flow of Events:   1. Customers click the checkout button in the cart. 2. Customers choose the existing shipping address or enter new shipping address. 3. Customers select payment method. 4. Customers confirm the purchase. 5. System verifies the purchase and return back the results. | | |
| Alternate/Exceptional Flows: | | |

**Keep track of orders**

|  |  |  |
| --- | --- | --- |
| Use Case Name: Keep track of orders | ID:001 | Importance Level: High |
| Primary Actor: Customers | Use Case Type: Overview, Essential | |
| Stakeholders and Interests:  Customers – Customers should be able to keep track of the order status. | | |
| Brief Description: Customers should be able to keep track of the order status and being notified when the status is updated. | | |
| Trigger:  After customers purchased an order. | | |
| Normal Flow of Events:   1. Customers click the order history tab to navigate to all the orders he or she have made. 2. The status of the orders will be shown. 3. The processing order will place on the top to let customers easier to see. 4. When the status of order is updated, customers will receive an email about the order status. | | |
| Alternate/Exceptional Flows: | | |

**Manage products**

|  |  |  |
| --- | --- | --- |
| Use Case Name: Manage products | ID:001 | Importance Level: High |
| Primary Actor: Administrators | Use Case Type: Overview, Essential | |
| Stakeholders and Interests:  Administrators – Administrators should be able to manage the products. | | |
| Brief Description: Administrators should be able to add, view, edit and delete products. | | |
| Trigger:  When addition or modification of products is needed. | | |
| Normal Flow of Events:  A1   1. Administrators click the add product tab. 2. Administrators fill in required products information. 3. Administrators click the add product button for confirmation.   B1   1. Administrators click the product name to view the products.   C1   1. Administrators click the edit button. 2. Administrators change the information of the products. 3. Administrators click the update product button.   D1   1. Administrators click the delete button. 2. Administrators click the confirm button to confirm the deletion. | | |
| Alternate/Exceptional Flows:  D1   1. Administrators click the delete button. 2. Administrators click the no button to cancel the deletion. | | |

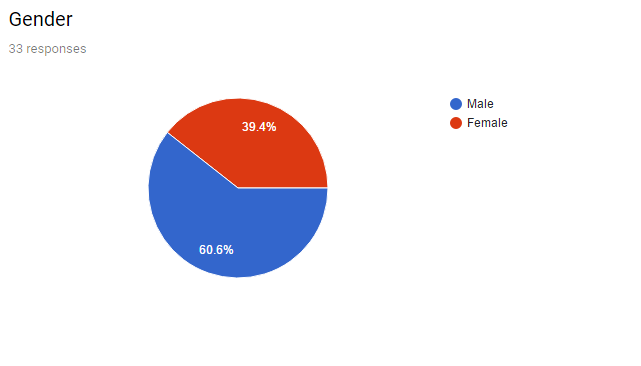
**Manage customers**

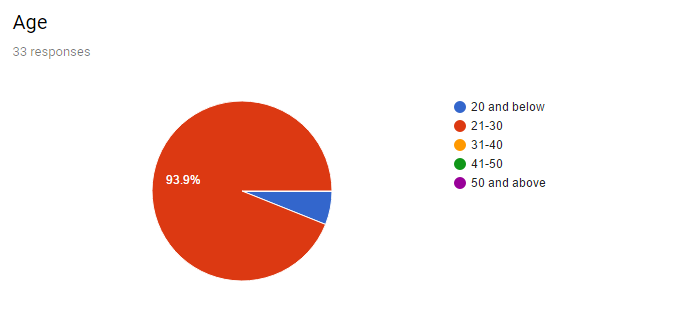
|  |  |  |
| --- | --- | --- |
| Use Case Name: Manage customers | ID:001 | Importance Level: Medium |
| Primary Actor: Administrators | Use Case Type: Overview, Essential | |
| Stakeholders and Interests:  Administrators – Administrators should be able to manage customers. | | |
| Brief Description: Administrators should be able to manage the customers’ accounts. | | |
| Trigger:  When changes for customers’ accounts are essential. | | |
| Normal Flow of Events:   1. Administrators search customers by keyword. 2. Administrators manage the information of the customers. | | |
| Alternate/Exceptional Flows: | | |

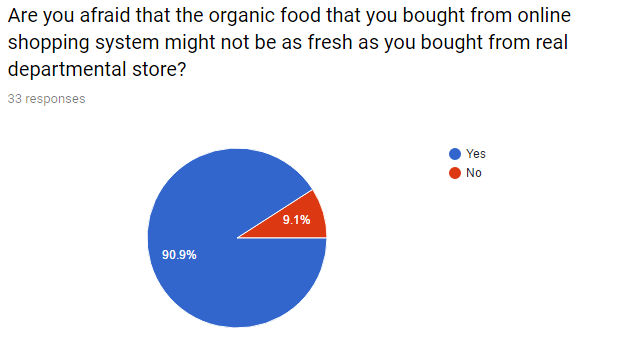
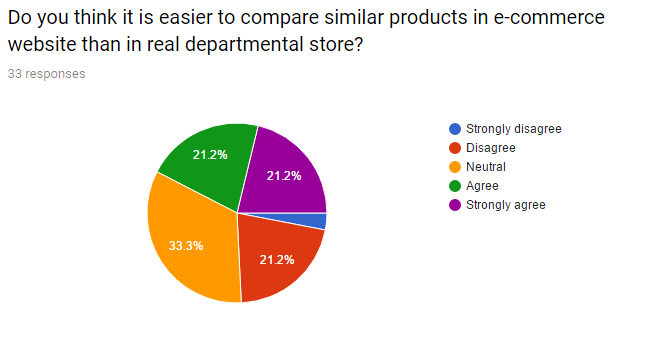
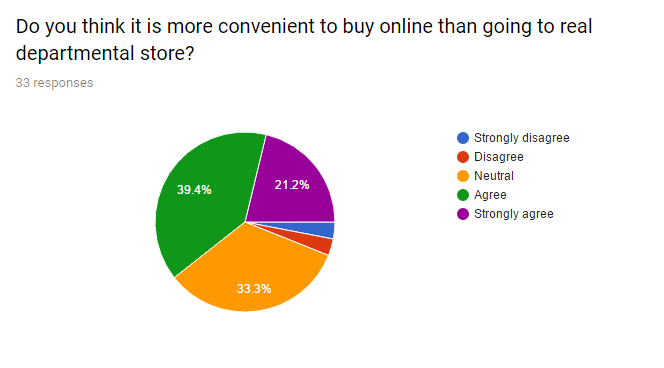
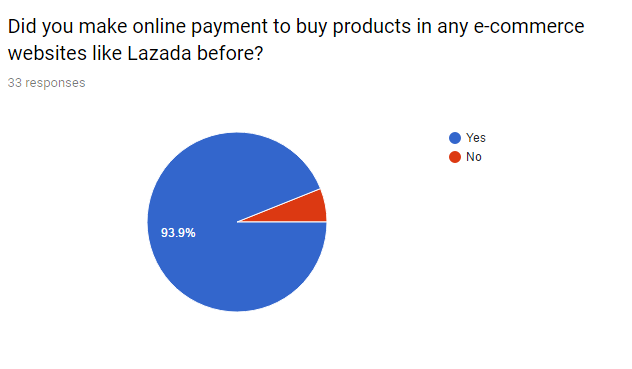
**Generate reports**

|  |  |  |
| --- | --- | --- |
| Use Case Name: Generate reports | ID:001 | Importance Level: Medium |
| Primary Actor: Administrators | Use Case Type: Overview, Essential | |
| Stakeholders and Interests:  Administrators – Administrators should be able to generate reports. | | |
| Brief Description: Administrators should be able to generate reports based on different kind of data. | | |
| Trigger:  When administrators want to conduct data analysis on different kinds of products. | | |
| Normal Flow of Events:   1. Administrators click the report tab. 2. Administrators select the report type and key in required information. 3. Administrators click generate report to view the report. | | |
| Alternate/Exceptional Flows: | | |

**Appendix B: Questionnaire**

****

****

****

**Summary of Questionnaire**

There are 33 responses in total for this survey. 60.6% are males and the rest are females. Most of them are between 21 to 30 years old. Most of them has purchase order in e-commerce websites before. Around 60% of the respondents agree that it is more convenient compared to real departmental store, while 33% stay neutral, the remaining disagree with that. There are 42% of the responses agree that it is easier to compare similar products in e-commerce websites while 33% remain neutral, and 21% disagree. Over 90% of the respondents afraid that the organic products from e-commerce websites are not genuine and not fresh as those in real departmental store.