ONE-CLICK PICK (ONLINE SHOPPING SYSTEM)

NUR AKLINDA BINTI MANSOR

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ONE-CLICK PICK (ONLINE SHOPPING SYSTEM)

NUR AKLINDA BINTI MANSOR

This report is submitted in partial fulfillment of the requirements for the Bachelor of [Computer Science (Database Management] with Honours.

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2020/2021

DECLARATION

I hereby declare that this project report entitled

[ONE-CLICK PICK (ONLINE SHOPPING SYSTEM]

is written by me and is my own effort and that no part has been plagiarized without citations.

STUDENT : NUR AKLINDA BINTI MANSOR Date : 27/6/2021

I hereby declare that I have read this project report and found this project report is sufficient in term of the scope and quality for the award of Bachelor of [Computer Science (Database Management] with Honours.

SUPERVISOR : Ts. Dr. Yahaya bin Abd Rahim Date : 27/6/2021

DEDICATION

To my beloved, I would like to dedicate this Final Year Project because always giving me the spirit to keep learning and keep trust me to not give up since I am young until today. I also would like to dedicate this Final Year Project to my supervisor Ts. Dr. Yahaya bin Abd Rahim, for his idea, help, keep motivated me, give efforts tells any mistakes that made by me and endless support to complete this project and also to all my beloved friend Nur Tajalli Kamalputri binti Kamarulzaman, Nor Haliza binti Abd Hamid and Aina Najeehah binti Anuar that helped me during the development of this project.

ACKNOWLEDGEMENTS

This system and report were made successfully with the help of the lecturer, friend, and family. Please allow me to express my greatest gratitude to the following significant advisor and contributors. I would like to thank Ts. Dr. Yahaya bin Abd Rahim for giving assistant to complete this project successfully. He has been my greatest supporter and advisor during the development of this project. He keeps motivating me to finish this project and, he gives a lot of idea on how my project is going to be developed. It would not have been possible without his professional guidance, advice, expert, encouragement, and support throughout this PSM. He has given me the spirit of adventure regarding research and convinced me to do better as always being supportive for me by giving valuable guidance as well as extended encouragement.

I would also like to thank my beloved parents who have been giving me support and motivation throughout my project. Thank you for all the endless moral support, financial support, and idea to complete this project. Finally, I also want to thank my friend Nur Tajalli Kamalputri binti Kamarulzaman, Nor Haliza binti Abd Hamid and Aina Najeehah binti Anuar for their help and time to teach me a lot to finish this project. This project could not be done without help from them.

ABSTRACT

In the covid-19 season we are going through right now, there are a lot of limited things for example like outdoor activities. One of the limited things is business or buying and selling activities. Business activities become limited during this covid deck season because the government has set a time limit for a business to be conducted. In addition, the government also stipulates the types of businesses that can be opened during this covid season, for example, such as only grocery stores are allowed to operate during this season. So, for those who run a business other than opening a grocery store for example a food business, then they must temporarily close their business premises. This causes traders to have problems in terms of income or more clearly in terms of finance. As for the customers, it is quite popular for them to go out to buy goods with the time limit for someone to leave the house. Not only that, the government also set a limit on the number of families leaving the house. So, this kind of thing is quite difficult for Malaysians to accept. Therefore, One-Click Pick is a suitable system to solve problems experienced by both parties, namely the customer and the dealer. In this system has been equipped with a function where customers can buy the goods they want and payment can be made either by COD (Cash on Delivery) or online payment that is Online Banking. On behalf of traders, this system has provided a function where traders can trade online. That is, the merchant can include a picture of the sales product and information on that product. So, when the customer is attracted to the product then, the dealer can arrange the delivery. Therefore, this system has also helped in reducing the number of Malaysians who go out to buy equipment.

ABSTRAK

Pada musim covid-19 yang sedang kita lalui sekarang, terdapat banyak perkara yang terbatas contohnya seperti aktivit-aktiviti luar. Antara salah satu perkara yang terbatas adalah aktiviti perniagaan atau jual beli. Aktiviti perniagaan menjadi terhad pada musim covid ini dek kerana kerajaan telah menetapkan had masa bagi sesebuah perniagaan dijalankan. Selain itu, kerajaan juga menetapkan jenis perniagaan yang boleh dibuka sepanjang musim covid ini contohnya seperti hanya kedai runcit sahaja yang dibenarkan beroperasi sepanjang musim ini. Jadi bagi mereka yang menjalankan perniagaan selain membuka kedai runcit contohnya perniagaan makanan, maka mereka terpaksa menutup sementara premis perniagaan mereka. Hal ini menyebabkan peniaga-peniaga mengalami masalah dari segi pendapatan atau lebih jelasnya dari segi kewangan. Bagi sisi pelanggan pula, agak suka bagi mereka keluar membeli barang dengan adanya had masa bagi seseorang keluar rumah. Bukan itu sahaja, kerajaan juga menetapkan had bilangan bagi satu keluarga keluar rumah. Jadi hal seperti ini agak sukar diterima oleh rakyat Malaysia. Maka dengan itu, One-Click Pick adalah sistem yang sesuai bagi menyelesaikan masalah yang dialami oleh kedua-dua belah pihak iaitu pihak pelanggan dan peniaga. Di dala sistem ini telah dilengkapi dengan fungsi dimana pelanggan boleh membeli barang yang mereka inginkan dan pembayarang boleh dibuat sama ada dengan cara COD (Cash on Delivery) atau pembayaran secara atas talian iaitu Online Banking. Bagi pihak peniaga pula, sistem ini telah menyediakan fungi dimana peniaga boleh berniaga secara atas talian. Maksudnya, peniaga boleh memasukkan gambar produk jualan dan maklumat produk tersebut. Jadi apabila pelanggan tertarik dengan produk tersebut maka, peniaga boleh menguruskan penghantaran. Oleh yang demikian, sistem ini juga sekaligus telah menmbantu dalam pengurangan jumlah rakyat Malaysia yang keluar rumah untuk membeli barangan kelengkapan.

TABLE OF CONTENTS

PAGE

DECLARATION ..... **II**

DEDICATION ..... **III**

ACKNOWLEDGEMENTS ..... **IV**

ABSTRACT ..... **V**

ABSTRAK ..... **VI**

TABLE OF CONTENTS ..... **VII**

LIST OF TABLES ..... **XI**

LIST OF FIGURES ..... **XIII**

LIST OF ABBREVIATIONS ..... **XVI**

LIST OF ATTACHMENTS ..... **XVII**

CHAPTER 1: INTRODUCTION **1**

1.1 About Chapters and First Subtitle 1

1.1.1 Problem Statement 2

1.1.2 Objective 3

1.1.3 Scope of the project 3

1.1.4 Expected Outcome/Proposed solution 5

1.1.5 Conclusion 5

CHAPTER 2: LITERATURE REVIEW AND PROJECT METHODOLOGY **7**

2.1 Introduction 7

2.2 Database Development Methodology 8

2.3 Project Schedule and Milestone 10

2.4 Conclusion 12

CHAPTER 3: ANALYSIS 13

3.1 Introduction 13

3.2 Problem Analysis 13

3.3 The Proposed Improvements/Solutions 17

3.4 Requirement Analysis of the To-Be-System 17

3.4.1 Functional Requirement (Process Model) 17

3.4.1.3 Data Flow Diagram Level 1 (Seller) 18

3.4.1.5 Data Flow Diagram Level 2 (Admin) 19

3.4.2 Non-Functional Requirement 22

3.4.3 Others Requirement 23

3.5 Conclusion 27

CHAPTER 4: DESIGN 28

4.1 Introduction 28

4.2 Introductory Preview to this Chapter 28

4.2.1 Database Design 28

4.2.2 Conceptual Design 28

4.2.3 Logical Design 32

4.2.4 Physical Design 38

4.3 Graphical User Interface (GUI) Design 47

4.4 Conclusion 59

CHAPTER 5: IMPLEMENTATION 60

5.1 Introduction 60

5.2 Software Development Environment Setup 60

5.2.1 Software Environment Setup XAMPP Server 61

5.2.2 Database Environment Setup 65

5.3 Database Implementation 67

5.3.1 Data Definition Language (DDL) 67

5.4 Conclusion 80

CHAPTER 6: TESTING 81

6.1 Introduction 81

6.2 Test Plan 82

6.2.1 Test Organization 83

6.2.2 Test Environment 84

6.2.3 Test Schedule 87

6.3 Testing Strategies 88

6.3.1 Classes of Test 89

6.4 Test Design 90

6.4.1 Test Description 90

6.4.2 Test Data 95

6.5 Test Result and Analysis 103

6.6 Conclusion 107

CHAPTER 7: CONCLUSION 108

7.1 Introduction 108

7.2 Observation on Weaknesses and Strengths 108

7.3 Proposition for Improvement 109

7.4 Project Contribution 109

7.5 Conclusion 110

REFERENCES 111

APPENDICES 112

USER MANUAL FOR ADMIN 112

USER MANUAL FOR SELLER 113

USER MANUAL FOR CUSTOMER 116

LIST OF TABLES

PAGE

Table 1.1.1 List of each user scope in One-Click Pick 4
Table 2.1: Milestone for PSM 1 10
Table 3.1: Hardware Requirement for PSM 1 27
Table 5.1: Trigger in One-Click Pick (Online Shopping System) 77
Table 5.2: List of Procedures in One-Click Pick (Online Shopping System) 78
Table 6.1: List of Tester and Their Responsibilities 83
Table 6.2: Environment Setup Specification 84
Table 6.3: Application Environment 85
Table 6.4: System Software 85
Table 6.5: System Hardware Tools 86
Table 6.6: Test Schedule 87
Table 6.7: White Box Testing Vs Black Box Testing 88
Table 6.8: Registration Module 90
Table 6.9: Login Module 91
Table 6.10: Change Password Module 91
Table 6.11: Update Profile Module 92
Table 6.12: Seller Module (Add Product) 92
Table 6.13: Seller Module (Send Message) 93
Table 6.14: Seller Module (Edit Product) 93
Table 6.15: Customer Module (Insert Banking Detail) 94
Table 6.16: Customer Module (Add Product in Cart) 94
Table 6.17: Test Data of Registration 95
Table 6.18: Test Data of Login 96
Table 6.19: Test Data of Change Password 97

Table 6.20: Test Data of Update Profile **97**
Table 6.21: Test Data of Seller (Add Product) **98**
Table 6.22: Test Data of Seller (Send Message) **100**
Table 6.23: Test Data of Seller (Edit Product) **100**
Table 6.24: Test Data of Customer (Insert Banking Detail) **102**
Table 6.25: Test Data of Customer (Add Product in Cart) **103**
Table 6.26: Registration Module Test Result **103**
Table 6.27: Login Module Test Result **104**
Table 6.28: Change Password Module Test Result **104**
Table 6.29: Update Profile Module Test Result **105**
Table 6.30: Seller (Add Product) Module Test Result **105**
Table 6.31: Seller (Send Message) Module Test Result **105**
Table 6.32: Seller (Edit Product) Module Test Result **106**
Table 6.33: Customer (Insert Banking Detail) Module Test Result **106**
Table 6.34: Customer (Add to Cart) Module Test Result **106**
Table 7.1: Strength and Weakness **108**

LIST OF FIGURES

PAGE

Figure 2.1: Database Life Cycle (DBLC) 8
Figure 3.1: Flowchart of Customer 14
Figure 3.2: Flowchart of Seller 15
Figure 3.3: Flowchart of Admin 16
Figure 3.4: One-Click Pick Context Diagram 17
Figure 3.5: Admin Data Flow Diagram level 1 18
Figure 3.6: Seller Data Flow Diagram level 1 18
Figure 3.7: Customer Data Flow Diagram level 1 19
Figure 3.8: Admin Data Flow Diagram level 2(Manage Customer) 19
Figure 3.9: Admin Data Flow Diagram level 2(Manage seller) 20
Figure 3.10: Admin Data Flow Diagram level 2(Manage Category) 20
Figure 3.11: Seller Data Flow Diagram level 2(Add new product) 20
Figure 3.12: Seller Data Flow Diagram level 2(Manage Order) 21
Figure 3.13: Customer Data Flow Diagram level 2(View/Select product) 21
Figure 3.14: Customer Data Flow Diagram level 2(Manage Payment) 22
Figure 4.1: One-Click Pick Entity Relationship Diagram (ERD) 29
Figure 4.2: Data Dictionary for tbl_color, tbl_customer, tbl_end_category 32
Figure 4.3: Data Dictionary for tbl_mid_category, tbl_order, tbl_payment 33
Figure 4.4: Data Dictionary for tbl_product 34
Figure 4.5: Data Dictionary for tbl_product_color, tbl_product_photo, tbl_product_size 35
Figure 4.6: Data Dictionary for tbl_rating, tbl_seller 36
Figure 4.7: Data Dictionary for tbl_size, tbl_category, tbl_user 37
Figure 4.8: One-Click Pick Home page 47

Figure 4.9: One-Click Pick Customer Registration Page **47**
Figure 4.10: One-Click Pick Customer Login Page **48**
Figure 4.11: One-Click Pick Customer Dashboard Page **48**
Figure 4.12: One-Click Pick Customer Update Profile Page **49**
Figure 4.13: One-Click Pick Customer Order History Page **49**
Figure 4.14: One-Click Pick Customer Update Password Page **50**
Figure 4.15: One-Click Pick Customer Front Page **50**
Figure 4.16: One-Click Pick Customer Front Page **51**
Figure 4.17: One-Click Pick Customer Cart Page **51**
Figure 4.18: One-Click Pick Customer Payment Page **52**
Figure 4.19: One-Click Pick Seller Login Page **52**
Figure 5.1: XAMPP Server Web Sites Download Folder **61**
Figure 5.2: User Account Control (UAC) **61**
Figure 5.3: XAMPP Setup Wizard **62**
Figure 5.4: Components Selection Interface **62**
Figure 5.5: Choose the Installation Directory **63**
Figure 5.6: Installation Process Started **63**
Figure 5.7: Complete Installation **64**
Figure 5.8: Starting XAMPP Modules **64**
Figure 5.9: phpMyAdmin First Page **65**
Figure 5.10: Creating A Database as 'oneclickpick' **66**
Figure 5.11: List Databases That Created **67**
Figure 5.12: Create Database **68**
Figure 5.13: Create Table Tbl_color **68**
Figure 5.14: Create Table Tbl_customer **69**
Figure 5.15: Create Table Tbl_end_category **69**
Figure 5.16: Create Table Tbl_mid_category **70**
Figure 5.17: Create Table Tbl_order **70**
Figure 5.18: Create Table Tbl_payment **71**
Figure 5.19: Create Table Tbl_product **72**
Figure 5.20: Create Table Tbl_product_color **73**
Figure 5.21: Create Table Tbl_product_photo **73**
Figure 5.22: Create Table Tbl_product_size **74**
Figure 5.23: Create Table Tbl_rating **74**

Figure 5.24: Create Table Tbl_seller..... 75
Figure 5.25: Create Table Tbl_size..... 75
Figure 5.26: Create Table Tbl_top_category..... 76
Figure 5.27: Create Table Tbl_user 76
Figure 5.28: Create Trigger Customer Email 77
Figure 5.29: Create Stored Procedure Display Total Seller..... 78
Figure 5.30: Create Stored Procedure Display Total Customer..... 78
Figure 5.31: Create Stored Procedure Display Details of Customer..... 79
Figure 5.32: Create Stored Procedure Display Details of Seller..... 79

LIST OF ABBREVIATIONS

FYP	-	Final Year Project
PSM	-	Projek Sarjana Muda
PHP	-	Hypertext Preprocessor
CSS	-	Cascading Style Sheets
HTML	-	Hypertext Markup Language
SQL	-	Structured Query Language

LIST OF ATTACHMENTS

Chapter 1: INTRODUCTION

1.1 About Chapters and First Subtitle

Nowadays, one of the biggest transformations using technologies is the online shopping system. It is also has almost the same size of the market as a general physical market possesses. Online shopping system is a process in which user are being provided with the option of purchasing goods and services directly from the seller all-in real-time environment. People in large number are doing online shopping today and it is not only because it is convenient as one can shop from home, but also because there are ample number of varieties available, with a high competition of prices and it is also easy to navigate for searching regarding any item.

A web-based program intended for online retailers is the One-Click Pick (Online Shopping System). This application's main purpose is to make it interactive and simple to use. It will simplify searching, displaying and choosing a product. It provides a sophisticated consumer search engine to search for items that are unique to their needs. The search engine offers a simple and efficient way to search for products in which a user can interactively search for a product and optimize the products available based on the feedback of the user. The full specification of each product can then be displayed by the consumer. They may also view reviews of the product and write their own reviews as well. The application also includes a drag-and-drop feature, so that by dragging the item into the shopping cart, a user may add a product to the shopping cart. The main focus lies on creating a user-friendly search engine to explain the desired results and its drag-and-drop action effectively.

Many users have trouble going out shopping during this time of covid-19. This is because the government has set a cap for households that want to go out and purchase products in the Covid-19. So, indirectly, this makes it impossible for consumers to purchase their necessities. Therefore, users do not have to think about

the presence of such an application, since this application provides users with shopping facilities, such as in the supermarket.

1.1.1 Problem Statement

The problem statement is the description of the problem that currently exist that needs to be addressed:

i. **Hard-to-buy basic necessities during the COVID-19 pandemic.**

As can be seen, throughout the covid-19 season everyone had trouble going out to buy goods due to the time limits, distances and number of house representatives that had been enforced by superiors. This has caused problems as not everyone has free time within the set time limit. In addition, the set distance also causes inconvenience to many people where their homes and places to buy goods exceed the set distance limit.

ii. **Difficult to compare cost and quality of necessities from variety of shop.**

In order to get high quality goods at affordable prices, consumers need to be wise to make comparisons. However, to compare the quality and price of an item is quite difficult when it needs to be done physically for example, consumers need to visit many places just to get a reasonable price and quality.

iii. Difficult to shop as a result of time and work constraints.

For those who work, it is quite difficult for them to buy daily necessities due to time constraints. In addition, most convenience stores are closed on weekends. So, this will cause a bit of a problem to those who work on weekdays from Monday to Friday.

1.1.2 Objective

i. To develop an easy-to-use web-based interface where user can easily buy their things without going out.
ii. To provide an easy and convenient way to search and compare for products that specific to their needs or budgets.

iii. To create a system that can save user's time.

1.1.3 Scope of the project

(a) Module to be developed

i. Admin Module
Admin can manage information of the seller and user. Admin have the authority to view all the request of registration and can approve or reject the request from the seller or user. Admin also can add new features in the system.

ii. Seller Module
Seller module is to insert new product of their shop and also before that seller need to register to login as seller. Seller also can view all the list of feedback that have been written by user for their product. Seller can update the data of products such as the price or functionality.

iii. Customer Module
In this module, customer can register to login if they want to buy something. After login, user can make any purchase and also payment using this system. Customer also can update their profile and view their profile.

(b) Target User

Table 1.1.1 List of each user scope in One-Click Pick

SYSTEM SCOPE	EXPLANATION
Admin	**i.** Admin can activate or deactivate the user/seller status account. **ii.** Admin can edit website features. **iii.** Admin can add product category. **iv.** Admin can update profile.
Seller	**i.** Register/Login as seller. **ii.** Seller can add new product by inserting photo, description and many more. **iii.** Seller can manage user orders. **iv.** Seller can view total product for their shop. **v.** Seller can view total completed/pending orders and completed/pending shipping.

	vi. Seller can edit their profile.
Customer	**i.** Register/Login as customer. **ii.** Customer can add product into cart. **iii.** Customer can choose payment type. **iv.** Customer can edit profile. **v.** Customer can view their order history.

1.1.4 Expected Outcome/Proposed solution

According to the system that will be developed, it can facilitate all the shopping process. This is because, as we can see sometimes it is not easy for user to go shopping by their own as the place is crowded and as the covid-19 is still cannot be conducted it will be dangerous for user to go to a crowded place. This system also to ease user find the best product that they want because user can make a comparison in terms of price and quality. This system also can reduce time, example it will ease the shopping process as example user just need to click at the item that they want to buy or just drop the item that they want to cancel.

1.1.5 Conclusion

Overall, this chapter is critical in the development of a system that is more efficient, stable, and user-friendly. Furthermore, the system can be constructed

quickly because the developer will be familiar with the system's issue description, objective, and expected outcome. This chapter aids developers in understanding what should be built based on real-world data. To summarise, the new system that will be constructed will be able to improve the current system's functionality and efficacy.

This project's literature review and project methodology will be discussed in Chapter 2

CHAPTER 2: LITERATURE REVIEW AND PROJECT METHODOLOGY

2.1 Introduction

The process of gathering real-world requirements, assessing requirements, creating the data and functions of the system, and then executing the operations in the system is known as Database System Development. To ensure that the system was developed efficiently, the Database Life Cycle (DBLC) technique is used as database development methodology. In Section 2.2 Database Methodology, the phases of DBLC are covered. After each stage is completed, the life cycle advances to the next.

2.2 Database Development Methodology

Figure 2.1: Database Life Cycle (DBLC)

Figure 2.1 shows Database Life Cycle (DBLC) which contains six phases. These phases started with database initial study and followed database design, implementation and loading, testing and evaluation, operation, and maintenance and evolution.

i. Database Initial Study

The mission objectives of the database project are defined at this phase. The mission statement identifies the overall goals of the database application, whereas the objective identifies a specific activity that the database can help with. The database project's scope and boundaries are then specified. Knowing these things can help you define the data structures you'll need, as well as the type and number of entities you'll need. This phase also gathered information on how the planned system is supposed to work now and in the future. In this phase, the company's situation is assessed, and problems and constraints are identified.

ii. Database Design

The database model that will support safety inventory activities and objectives is designed in the second phase. The conceptual design, logical design, and physical design are all created during this phase, as well as the DBMS software choices. Conceptual database design, logical database design, and physical database design are the three phases of database design. The goal of conceptual database design is to create a conceptual model of the database that includes the identification of key items, relationships, and attributes. Data modelling aids in the comprehension of data and the transmission of information requirements. The logical database design process involves converting the conceptual representation into the database's logical structure, which includes defining the relationships. Physical database design, which is specialised to individual DBMS, determines how the logical structure will be physically implemented.

iii. Implemetation and Loading

The physical manifestation of the database and application design takes place in this phase. To implement a database in a DBMS, DDL and DML were used. The data must be loaded into the database tables after the database has been built. The application is written in a programming language.

iv. Testing and Evaluation

Testing and evaluation are the process of running the database system with intent of finding errors. The database is tested to ensure that it maintains the integrity and security of the data. This phase demonstrates that database and application program appear to be working according to requirements.

v. Operation

The database is considered operational once it has passed the evaluation step. The database, its management, its users, and its application program create a comprehensive information system at that time. During this phase, all the necessary information flows, including all the queries.

vi. Maintenance and Evolution

The database should be updated on a regular basis. Preventive and corrective maintenance (backup and recovery), adaptive maintenance, and access authorization assignment for old and new users are all required on a regular basis. During this phase, the database administrator must be ready to undertake routine database maintenance tasks.

2.3 Project Schedule and Milestone

There are specific schedule and milestone that must be followed during the implementation of PSM 1.

Table 2.1: Milestone for PSM 1

WEEK	ACTIVITY
Week1 **(15/3 – 21/3)** Meeting 1	Proposal PSM: Discussion with Supervisor
	Proposal assessment & verification
Week 2 **(22/3 – 28/3)**	Proposal Correction/Improvement
	Proposal submission to Committee via email
	Proposal Approval
	List of Supervisor/Title
Week 3 **(29/3 – 4/4)** Meeting 2	Proposal Presentation & Submission via PSM ULearn
	Chapter 1 (System Development Begins)

Week 4 **(5/4 – 11/4)**	Chapter 1
Week 5 **(12/4 – 18/4)**	Chapter 2
Week 6 **(19/4 – 25/4)** Meeting 3	Chapter 2 Project Progress Student Status
Week 7 **(26/4 – 2/5)**	Chapter 3
Week 8 **(3/5 – 9/5)**	Chapter 3
Week 9 **(10/5 – 16/5)**	**MID SEMESTER BREAK**
Week 10 **(17/5 – 23/5)** Meeting 4	Chapter 4 Project Progress Student Status
Week 11 **(24/5 – 30/5)** Demonstration	Project Demo Determination of student status (Continue/Withdraw)
Week 12 **(31/5 – 6/6)**	Project Demo PSM1 Report
Week 13 **(7/6 – 13/6)** Meeting 5	Project Demo PSM1 Report Schedule the Presentation
Week 14 **(14/6 – 20/6)**	Project Demo

Week 15 **(21/6 – 27/6)** Final Presentation	**FINAL PRESENTATION** Submission of the PSM1 Report onto the PSM ULearn.
Week 16 **(28/6 – 4/7)**	**REVISION WEEK** Correction on the draft report based on the Supervisor and Evaluator's comments during the final presentation session. Submit PSM1 Logbooks to PSM ULearn. Submit an EoS Survey form.
	Submission of overall marks to PSM/PD committee
Week 17 & Week 18 **(5/7 – 18/7)**	**FINAL EXAMINATION WEEKS**

2.4 Conclusion

Finally, the technique for database construction for the project is discussed in this chapter. The technique to developing the system is based on the Database Life Cycle (DBLC). Furthermore, the DBLC never ends because database monitoring, change, and maintenance are all part of the life cycle and continue long after a database is implemented. As a result, the DBLC covers the database's whole lifecycle.

The project analysis will be explored in detail in the next chapter (Chapter 3). There will be a problem analysis, potential improvements or solutions, non-functional requirements, functional requirements, and other requirements.

CHAPTER 3: ANALYSIS

3.1 Introduction

The analysis stage involves critical evaluation of the database development planning carried out beforehand. In this stage, developer assesses the plan of database development against the elements like cost, time-period, development platform, programming languages, and forecasted development results, to analyze the effectiveness associated with the planned database system. This phase involves analyzing the current system and the system to be developed.

3.2 Problem Analysis

To round up this analysis, I have made additional notes about how shopping and selling activities were a little difficult throughout the covid-19 season. Both the seller and the customer are affected by this issue. Consumers are unable to go out shopping freely as a result of the Covid-19 case, which is becoming increasingly disturbing in our country. As a result, the government has been forced to issue a directive establishing a house exit limit. This directive is difficult to accept, yet people must follow the instructions given.

Furthermore, the problem that sellers encounter is that their company operations are disrupted, and merchants are forced to temporarily cease their operations. This is a major source of concern for merchants, as it has disturbed their income to some extent.

CUSTOMER

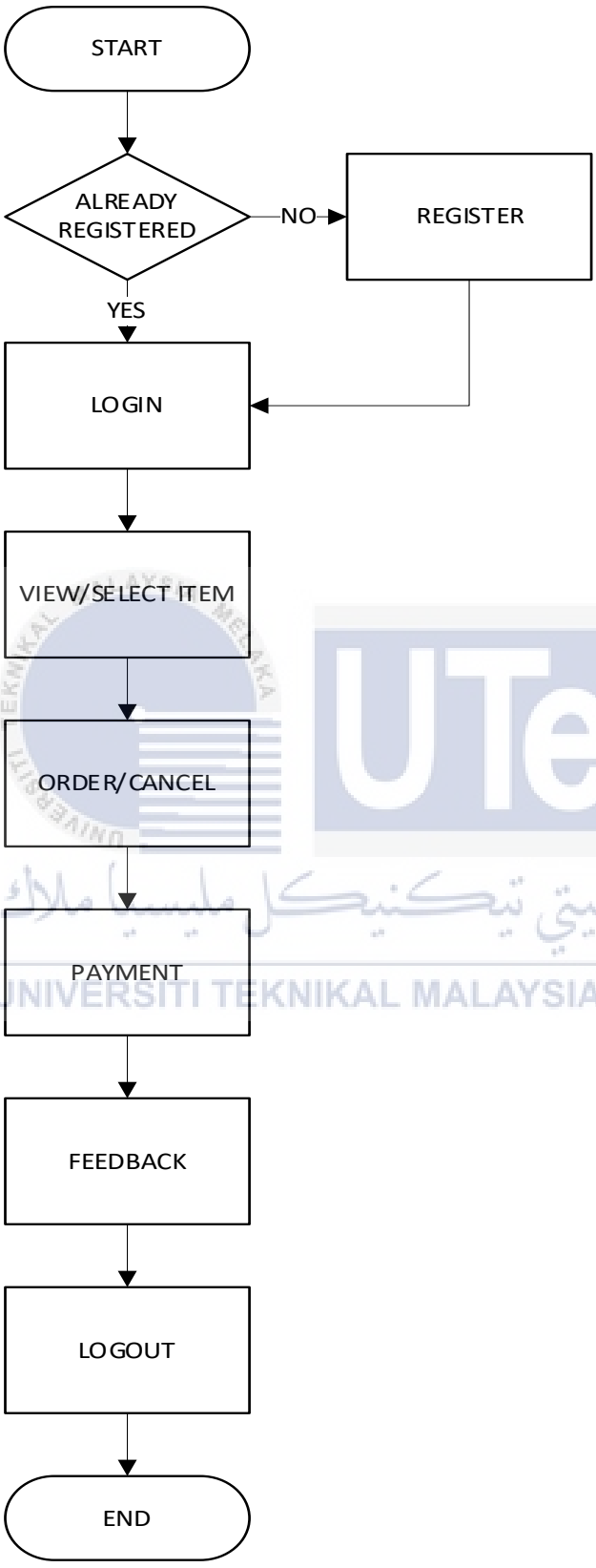

Figure 3.1: Flowchart of Customer

SELLER

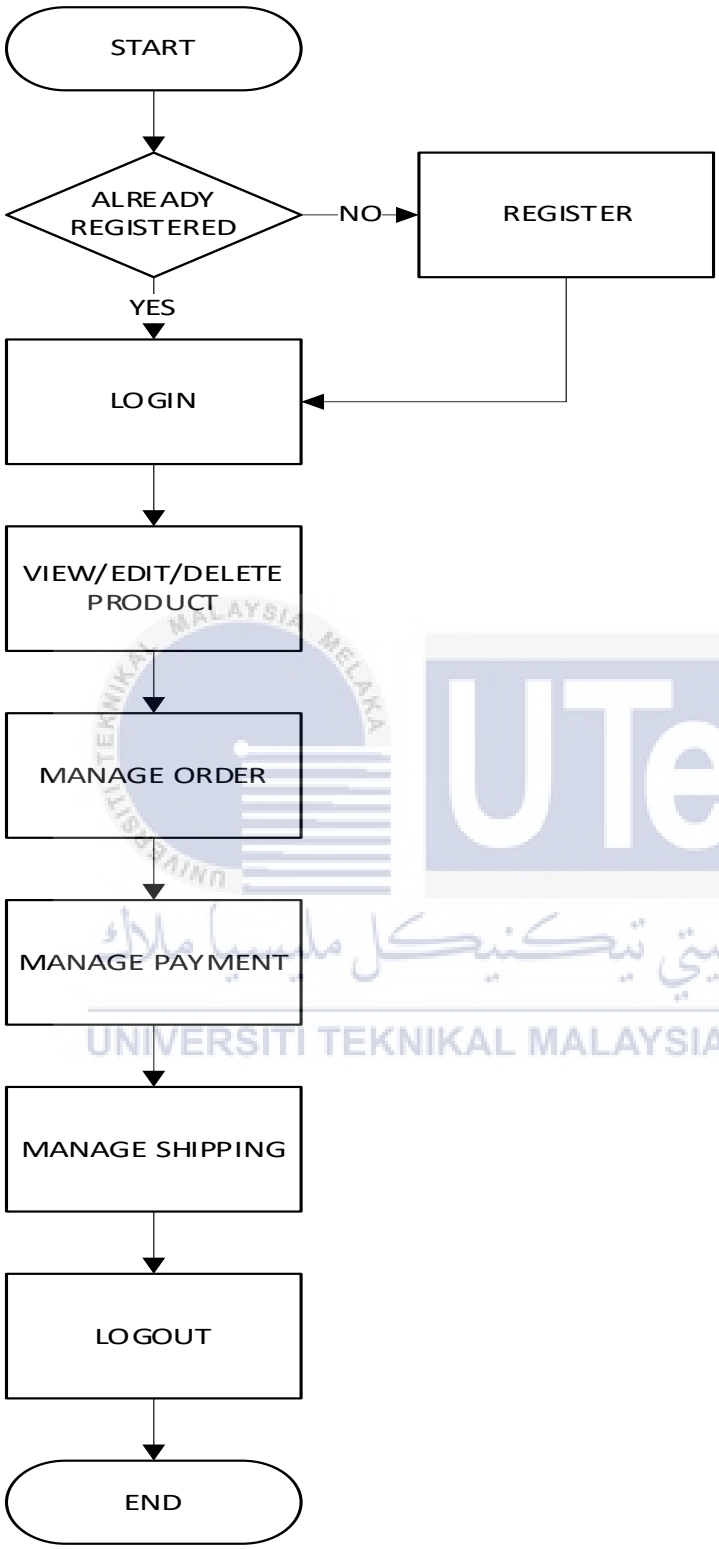

Figure 3.2: Flowchart of Seller

ADMIN

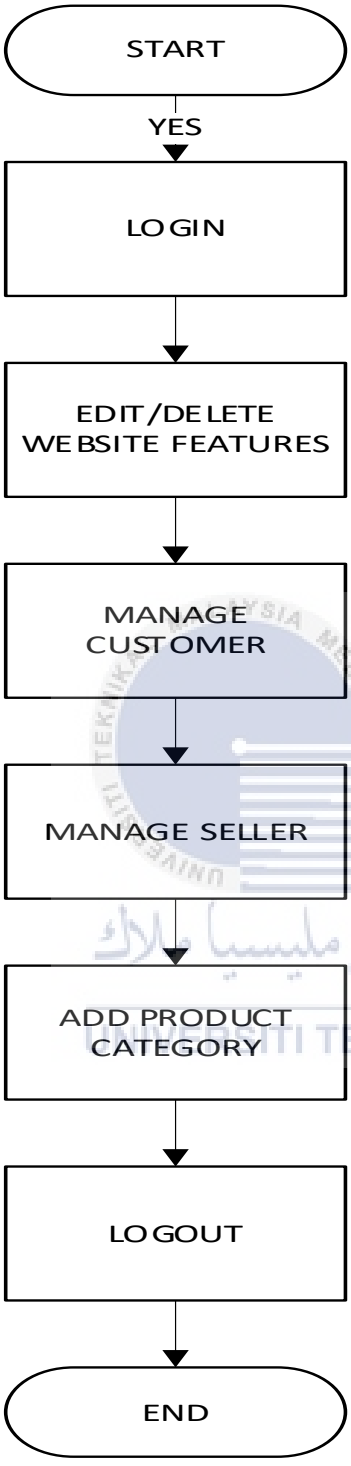

Figure 3.3: Flowchart of Admin

3.3 The Proposed Improvements/Solutions

New system proposed to help user easily buying their necessities without going out and also helps seller a lot on selling their product. Besides, this system also help user to survey the best product with reasonable price.

3.4 Requirement Analysis of the To-Be-System

3.4.1 Functional Requirement (Process Model)

3.4.1.1 Data Flow Diagram Level 0

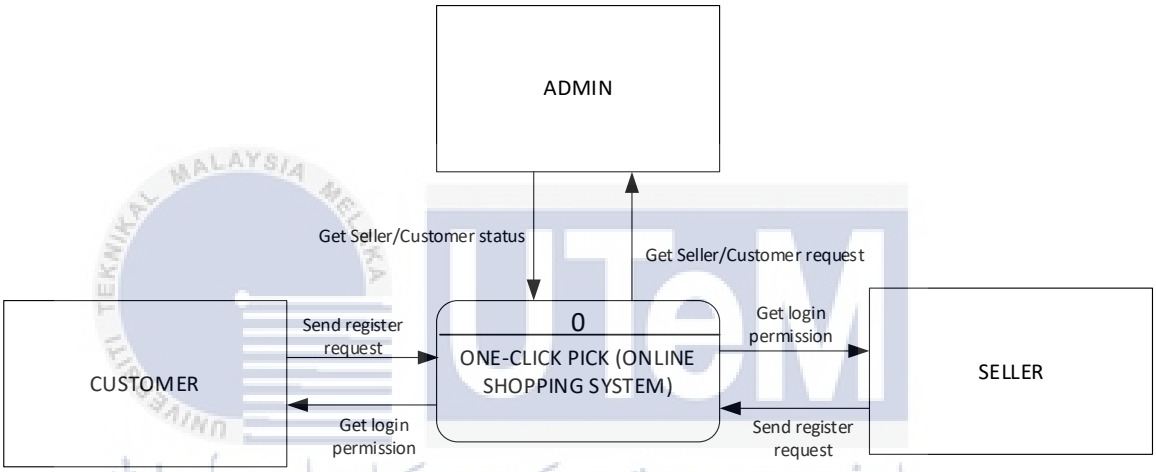

Figure 3.4: One-Click Pick Context Diagram

The figure shows how the system interact with external entities. Seller and Customer can request the login permission from admin after register their account. Admin can update the seller and customer login permission.

3.4.1.2 Data Flow Diagram Level 1 (Admin)

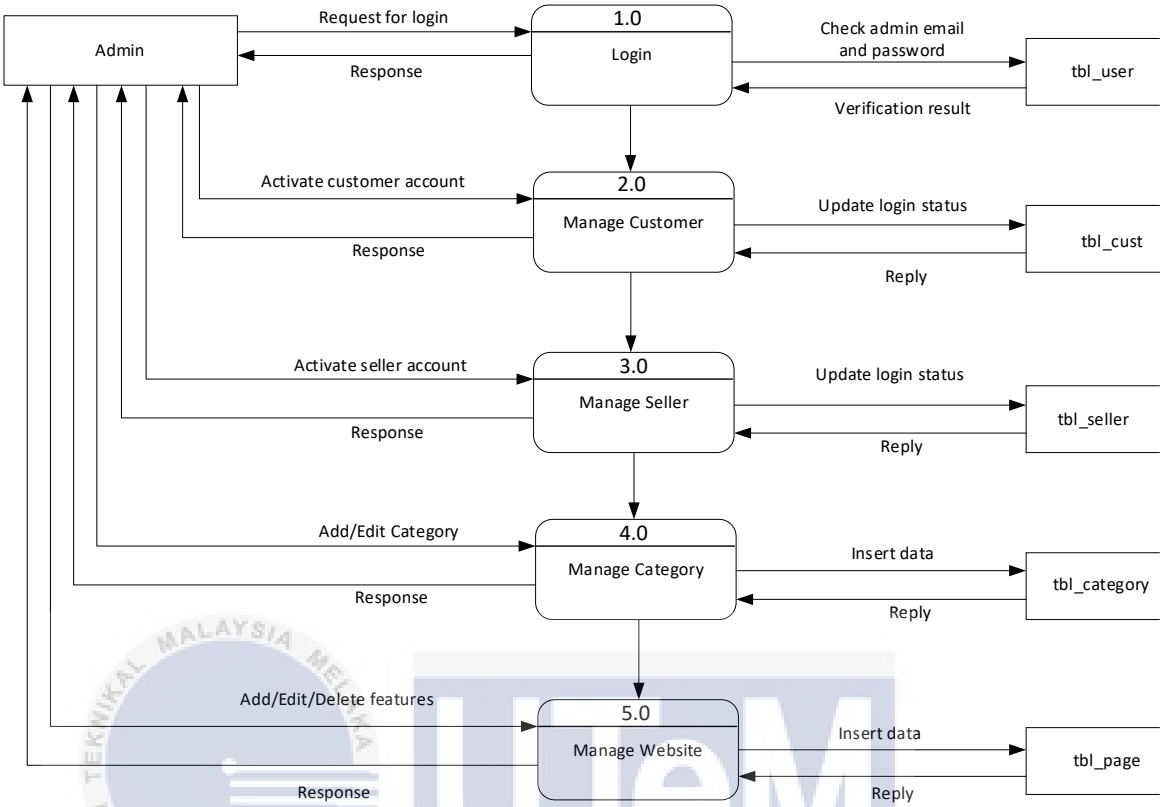

Figure 3.5: Admin Data Flow Diagram level 1

3.4.1.3 Data Flow Diagram Level 1 (Seller)

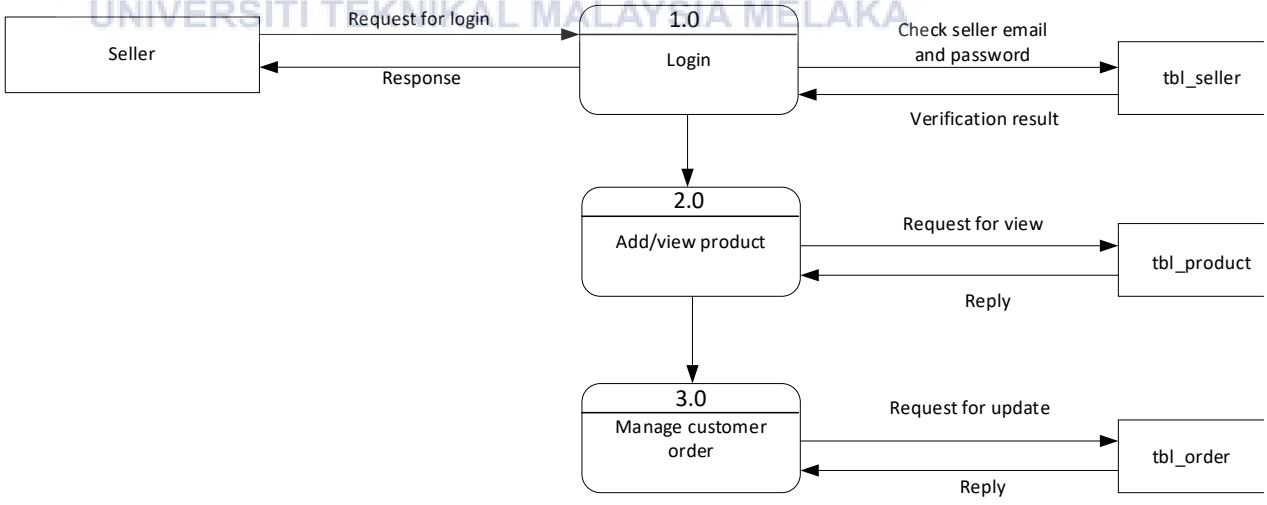

Figure 3.6: Seller Data Flow Diagram level 1

3.4.1.4 Data Flow Diagram Level 1 (Customer)

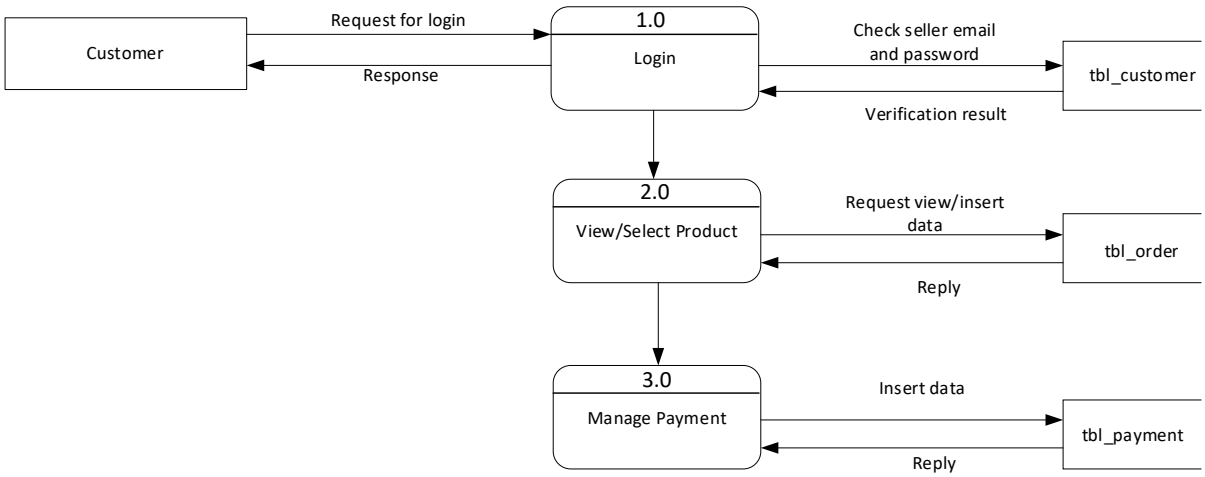

Figure 3.7: Customer Data Flow Diagram level 1

3.4.1.5 Data Flow Diagram Level 2 (Admin)

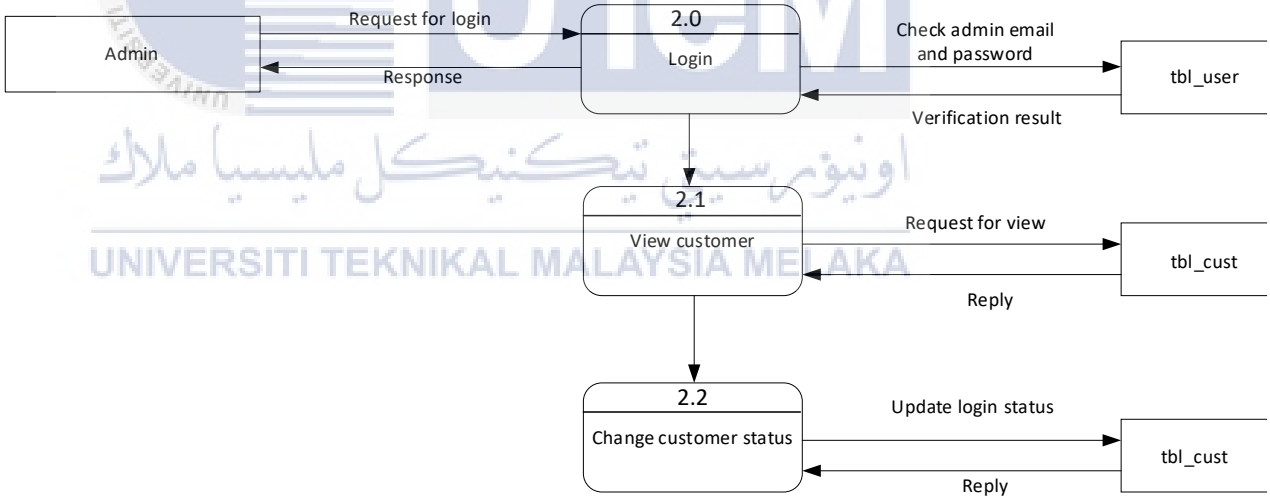

Figure 3.8: Admin Data Flow Diagram level 2(Manage Customer)

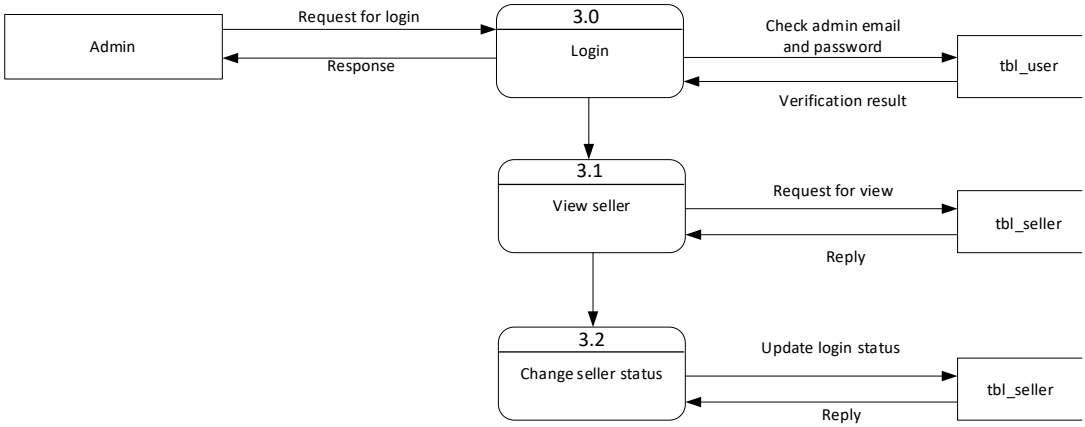

Figure 3.9: Admin Data Flow Diagram level 2(Manage seller)

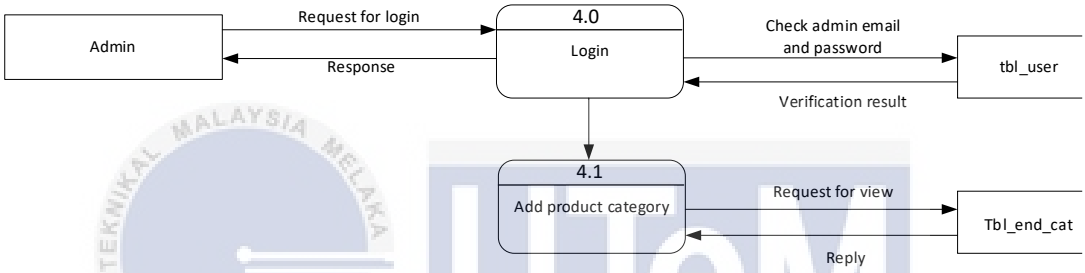

Figure 3.10: Admin Data Flow Diagram level 2(Manage Category)

3.4.1.6 Data Flow Diagram Level 2 (Seller)

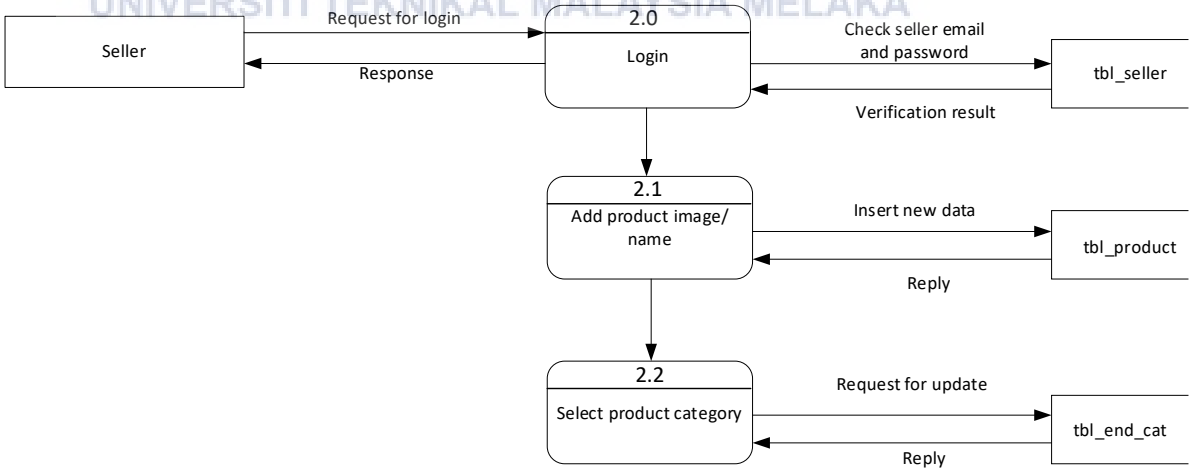

Figure 3.11: Seller Data Flow Diagram level 2(Add new product)

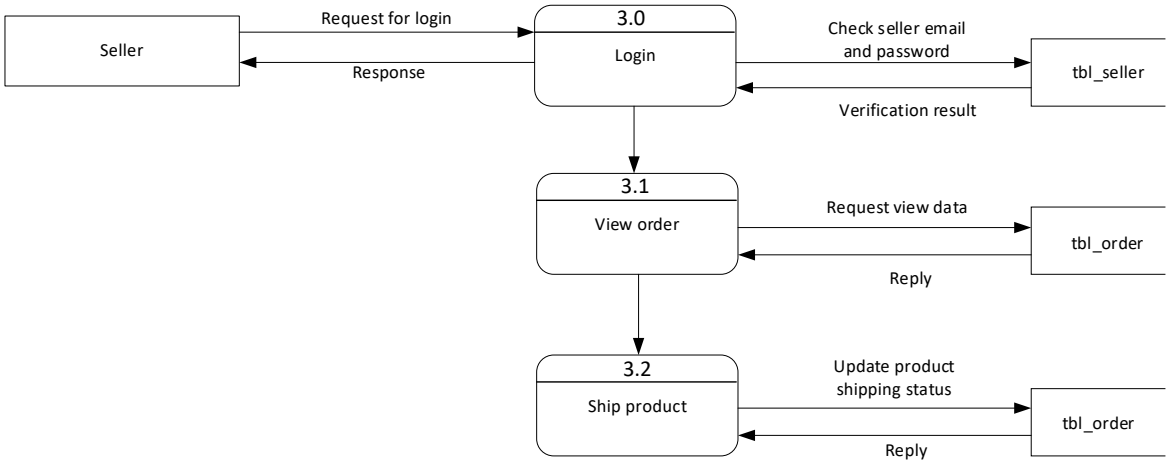

Figure 3.12: Seller Data Flow Diagram level 2(Manage Order)

3.4.1.7 Data Flow Diagram Level 2 (Customer)

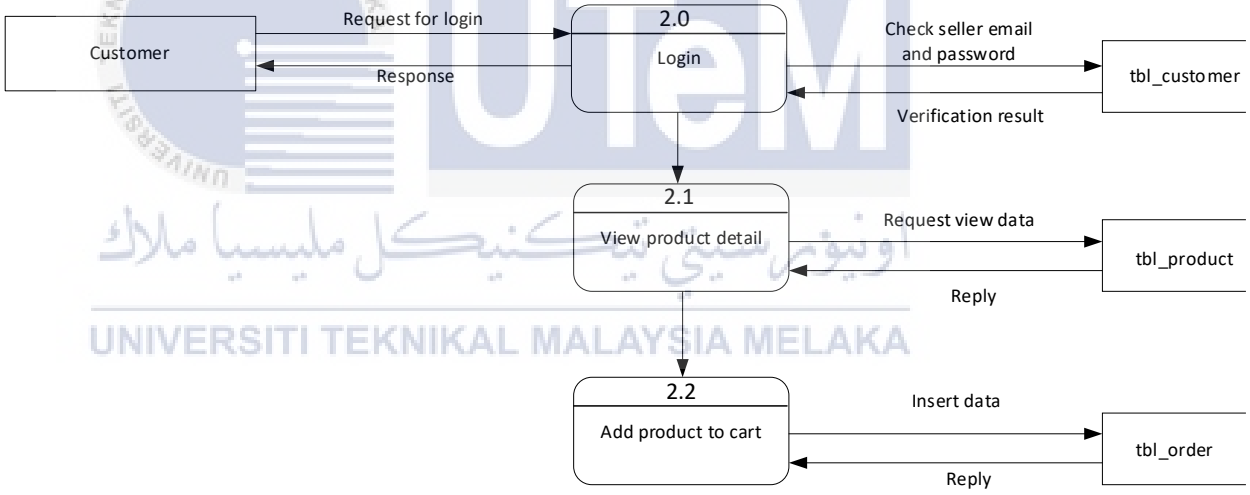

Figure 3.13: Customer Data Flow Diagram level 2(View/Select product)

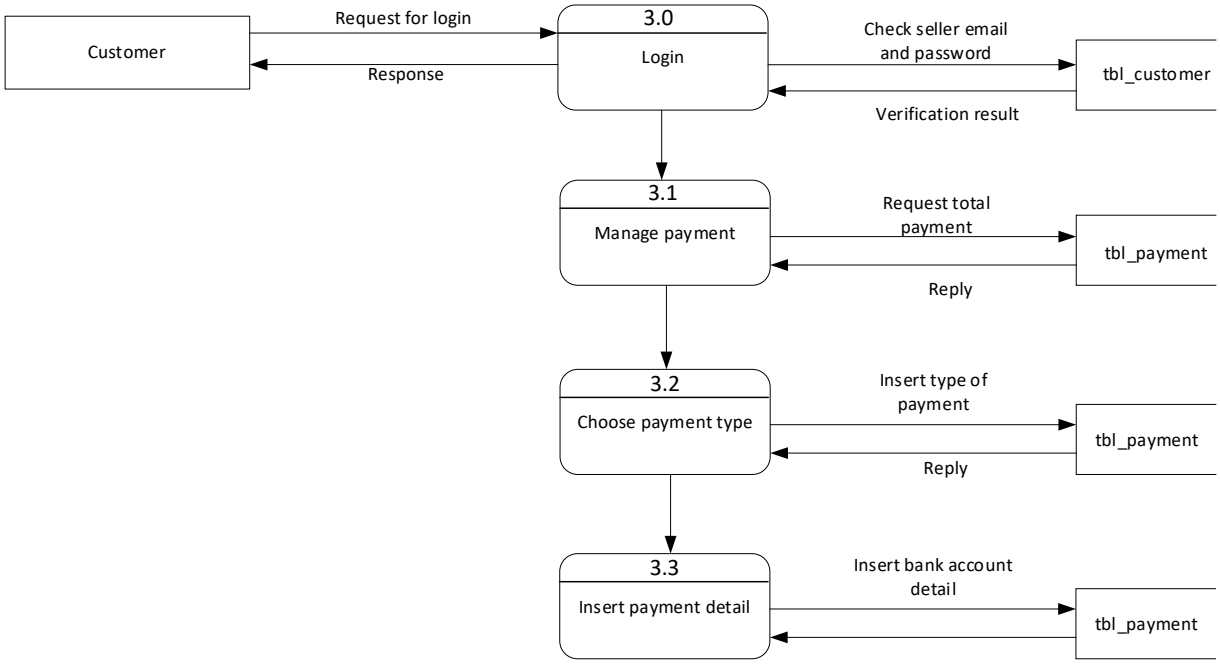

Figure 3.14: Customer Data Flow Diagram level 2(Manage Payment)

3.4.2 Non-Functional Requirement

(a) Authorisation

- Security requirements to ensure users can access only certain functions within the application such as user can only view their interfaces.

(b) Access Security

- Only admin can update the login permission status either accept or decline.

- Password shall never be viewable at the point of entry or at any other time.

3.4.3 Others Requirement

3.4.3.1 Software Requirement

(a) Documentation

- Microsoft Office 365

Office 365 is a version of Office that's available Office 365. This product contains Access, Excel, OneNote, Outlook, PowerPoint, Publisher, Skype for Business and Word. Microsoft Word is a word processor software that used in this project for documentation purpose.

(b) Programming Language

- PHP

PHP is a server-side scripting language. that is used to develop Static websites or Dynamic websites or Web applications. PHP stands for Hypertext Pre-processor, that earlier stood for Personal Home Pages. PHP is open source and free. PHP has in built support for working hand in hand with MySQL; this doesn't mean you can't use PHP with other database management systems. You can still use PHP with Postgres, Oracle, MSSQL Server, ODBC and etc. PHP is cross platform this means you can deploy your application on a number of different operating systems such as windows, Linux, Mac OS etc.

- HTML

Hypertext Markup Language is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets and scripting languages such as PHP.

- CSS

Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and PHP.

(c) DBMS

- MySQL

MySQL is an Oracle-backed open-source relational database management system (RDBMS) based on Structured Query Language (SQL). MySQL runs on virtually all platforms, including Linux, UNIX and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web applications and online publishing. MySQL is an important component of an open-source enterprise stack called LAMP. LAMP is a web development platform that uses Linux as the operating system, Apache as the web server, MySQL as the relational database management system and PHP as the object-oriented scripting language. (Sometimes Perl or Python is used instead of PHP.)

- phpMyAdmin

phpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web. phpMyAdmin supports a wide range of operations on MySQL and MariaDB. Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc) can be performed via the user interface, while you still have the ability to directly execute any SQL statement.

(d) Operating System

- Microsoft Windows 10

Microsoft Windows is a group of several graphical operating system families, all of which are developed, marketed and sold my Microsoft. The version that used is Windows 10.

(e) Server

- XAMPP

Xampp is a free and open-source cross-platform web server software for installing and using the Apache Web server. It has provided component including MYSQL database, PHP, HTTPS (SSL), Common gateway interface (CGI), Server side includes (SSI).

3.4.3.2 Hardware Requirement

The hardware that being used in this project is laptop. The hardware requirement in developing this system is listed below:

- Laptop

Table 3.1: Hardware Requirement for PSM 1

COMPONENT	SPECIFICATION
Microprocessor	(1.1 GHz base frequency, up to 2.6 GHz burst frequency, 4 MB cache, 2 cores)
Hard Drive	500 GB 5400 rpm
Memory	4 GB DDR4-2400 SDRAM (1 x 4 GB)
Keyboard	Full-size island-style keyboard with numeric keypad
Display	15.6" diagonal HD SVA BrightView WLED-backlit (1366 x 768)

Table 3.1 shows the hardware requirements for developing this project. The processor used is Intel Core i7 and the memory of the hardware is 4GB DDR4.

3.5 Conclusion

Problems of the present existing system has been analyzed during analysis phase. This chapter discusses the project requirement which incorporates the software requirements, hardware requirements and other requirements. Last, the deliverable of this chapter which are the flow chart of the present system, flow chart of the propose system, context diagram and data flow diagrams can help to realize an honest analysis. The structure and meaning of the info within the organizations are often understood easily by reviewing this analysis. All this analysis is going to be used as a suggestion to style the database.

Chapter 4 will explain about the design phase for this system.

CHAPTER 4: DESIGN

4.1 Introduction

This chapter summarizes design phase cares with the physical construction of the system. Included are the planning or configuration of the network like hardware, operating system, programming, design of user interfaces, design of system interfaces and security issues. it's important that the proposed design be tested for performance, and to ensure that it meets the wants outlined during the analysis phase. In other words, the main objective of this phase is to rework the previously defined requirements into a complete and detailed set of specifications which can be used during subsequent phase.

4.2 Introductory Preview to this Chapter

4.2.1 Database Design

Database Design is a collection of processes that facilitate the designing, development, implementation and maintenance of enterprise data management systems. Properly designed database is easy to maintain, improves data consistency and are cost effective in terms of disk storage space. This stage consists of three parts which are conceptual design, logical design and physical design. Some methodologies integrate the logical design stage into the other two stages. This section is not intended to be a definitive discussion of database design. Rather, it aims to introduce you to the topic.

4.2.2 Conceptual Design

Conceptual design phase is to create a conceptual model based upon the previously identified requirements, but closer to the ultimate physical model.

4.2.2.1 Entity Relationship Diagram (ERD)

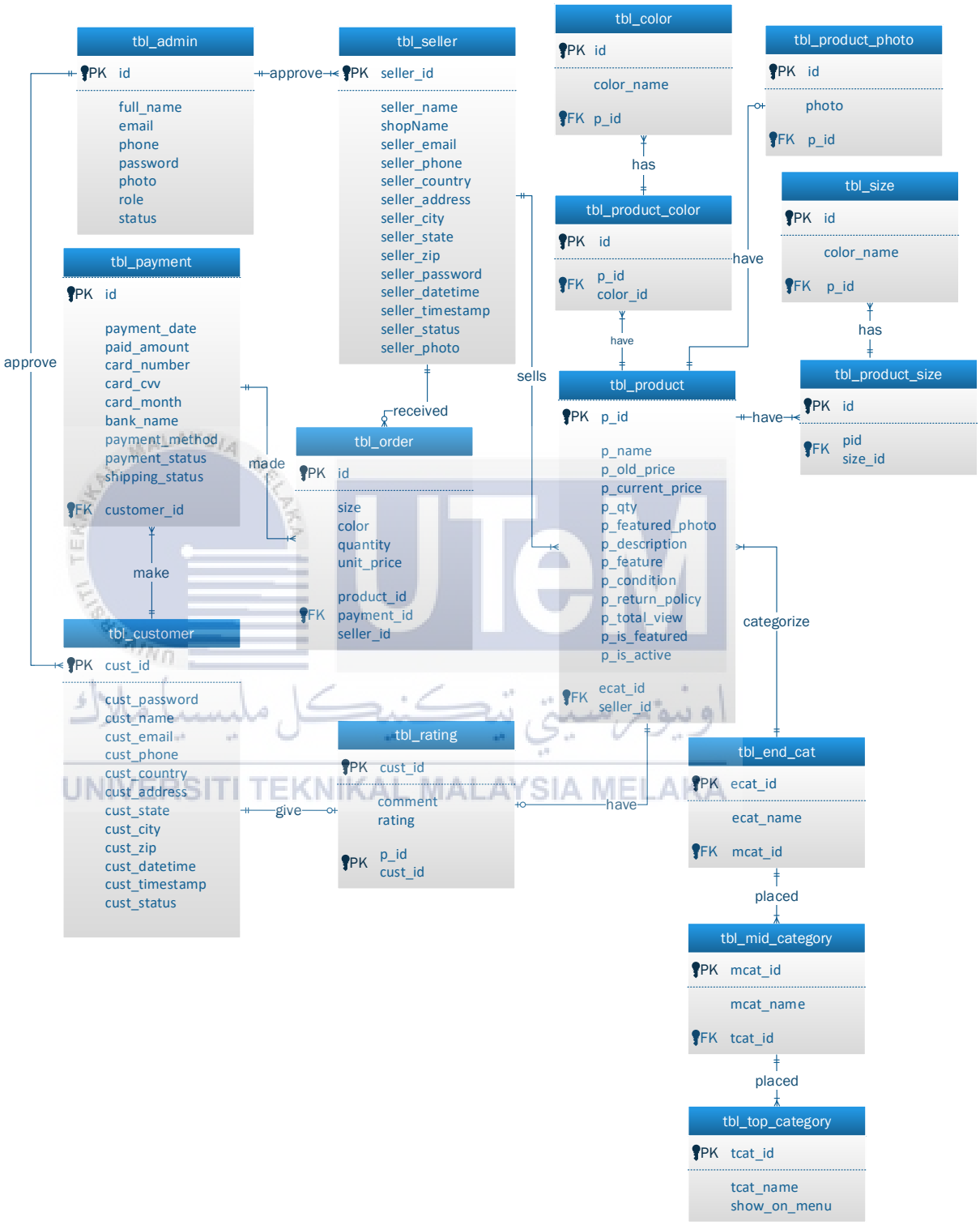

Figure 4.1: One-Click Pick Entity Relationship Diagram (ERD)

4.2.2.2 Business Rules

i. One admin manages one or many sellers' registration.

A seller can only be approved by one admin.

ii. One admin manages one or many customers' registration.

A customer can only be approved by one admin.

iii. One admin can deactivate one or many sellers' account if the account is more than 10 years.

One or many sellers' account will be deactivated by one admin if the account is more than 10 years.

iv. One admin can deactivate one or many sellers' account if the account is more than 10 years.

One or many sellers' account will be deactivated by one admin if the account is more than 10 years.

v. One seller can sell one or many products.

One or many products belongs to one seller.

vi. One seller can receive zero or many orders.

Zero or many orders can be accepted by one seller.

vii. One product can have one or many product colors.

One or many product colors are available for one product.

viii. Each product has one or zero photo.

One or zero photo is provided for each product.

ix. Each product has one or many product sizes.

One or many product sizes are available for each product.

x. One or many products are categorized in one category.

One category can contain one or many products.

xi. Each product can receive one or zero rating.

One or zero rating can be given for one product.

xii. Each customer can write one or zero rating.

One or zero rating are given by one customer.

xiii. Each customer can make one or many payments.

One or many payments are belonging to each customer.

4.2.3 Logical Design

4.2.3.1 Data Dictionary

tbl_color

Column	Type	Null	Default	Comments
color_id *(Primary)*	int(11)	No		
color_name	varchar(255)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	color_id	20	A	No	

tbl_customer

Column	Type	Null	Default	Comments
cust_id *(Primary)*	int(11)	No		
cust_name	varchar(100)	No		
cust_email	varchar(100)	No		
cust_phone	varchar(50)	No		
cust_country	int(11)	No		
cust_address	text	No		
cust_city	varchar(100)	No		
cust_state	varchar(100)	No		
cust_zip	varchar(30)	No		
cust_password	varchar(100)	No		
cust_datetime	varchar(100)	No		
cust_timestamp	varchar(100)	No		
cust_status	int(1)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	cust_id	2	A	No	

tbl_end_category

Column	Type	Null	Default	Comments
ecat_id *(Primary)*	int(11)	No		
ecat_name	varchar(255)	No		
mcat_id	int(11)	No		

Figure 4.2: Data Dictionary for tbl_color, tbl_customer, tbl_end_category

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	ecat_id	0	A	No	
mcat_id	BTREE	No	No	mcat_id	0	A	No	

tbl_mid_category

Column	Type	Null	Default	Comments
mcat_id *(Primary)*	int(11)	No		
mcat_name	varchar(255)	No		
tcat_id	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	mcat_id	0	A	No	
tcat_id	BTREE	No	No	tcat_id	0	A	No	

tbl_order

Column	Type	Null	Default	Comments
order_id *(Primary)*	int(11)	No		
product_id	int(11)	No		
product_name	varchar(255)	No		
size	varchar(100)	No		
color	varchar(100)	No		
quantity	varchar(50)	No		
unit_price	varchar(50)	No		
payment_id	int(11)	No		
seller_id	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	order_id	0	A	No	
payment_id	BTREE	Yes	No	payment_id	0	A	No	
seller_id	BTREE	No	No	seller_id	0	A	No	
product_id	BTREE	No	No	product_id	0	A	No	

tbl_payment

Column	Type	Null	Default	Comments
payment_id *(Primary)*	int(11)	No		

Figure 4.3: Data Dictionary for tbl_mid_category, tbl_order, tbl_payment

customer_id	int(11)	No		
customer_name	varchar(255)	No		
customer_email	varchar(255)	No		
payment_date	varchar(50)	No		
paid_amount	int(11)	No		
card_number	varchar(50)	No		
card_cvv	varchar(10)	No		
card_month	varchar(10)	No		
card_year	varchar(10)	No		
bank_name	text	No		
payment_method	varchar(20)	No		
payment_status	varchar(25)	No		
shipping_status	varchar(20)	No		
seller_id	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	payment_id	16	A	No	
customer_id	BTREE	No	No	customer_id		A	No	
seller_id	BTREE	No	No	seller_id		A	No	

tbl_product

Column	Type	Null	Default	Comments
p_id *(Primary)*	int(11)	No		
p_name	varchar(255)	No		
p_old_price	varchar(10)	No		
p_current_price	varchar(10)	No		
p_qty	int(10)	No		
p_featured_photo	varchar(255)	No		
p_description	text	No		
p_short_description	text	No		
p_feature	text	No		
p_condition	text	No		
p_return_policy	text	No		
p_total_view	int(11)	No		
p_is_featured	int(1)	No		
p_is_active	int(1)	No		
ecat_id	int(11)	No		
seller_id	int(11)	No		

Indexes

Figure 4.4: Data Dictionary for tbl_product

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	p_id	4	A	No	
ecat_id	BTREE	Yes	No	ecat_id	4	A	No	
				seller_id	4	A	No	

tbl_product_color

Column	Type	Null	Default	Comments
id *(Primary)*	int(11)	No		
color_id	int(11)	No		
p_id	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	46	A	No	
color_id	BTREE	Yes	No	color_id	46	A	No	
				p_id	46	A	No	

tbl_product_photo

Column	Type	Null	Default	Comments
pp_id *(Primary)*	int(11)	No		
photo	varchar(255)	No		
p_id	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	pp_id	51	A	No	
p_id	BTREE	No	No	p_id	51	A	No	

tbl_product_size

Column	Type	Null	Default	Comments
id *(Primary)*	int(11)	No		
size_id	int(11)	No		
p_id	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	102	A	No	
p_id	BTREE	No	No	p_id	102	A	No	

Figure 4.5: Data Dictionary for tbl_product_color, tbl_product_photo, tbl_product_size

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
size_id	BTREE	No	No	size_id	51	A	No	

tbl_rating

Column	Type	Null	Default	Comments
rt_id *(Primary)*	int(11)	No		
p_id	int(11)	No		
cust_id	int(11)	No		
comment	text	Yes	*NULL*	
rating	float	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	rt_id	8	A	No	
cust_id	BTREE	No	No	cust_id	2	A	No	
p_id	BTREE	No	No	p_id	8	A	No	

tbl_seller

Column	Type	Null	Default	Comments
seller_id *(Primary)*	int(11)	No		
seller_name	varchar(100)	No		
shopName	varchar(255)	Yes	*NULL*	
seller_email	varchar(100)	No		
seller_phone	varchar(50)	No		
seller_country	int(11)	No		
seller_address	text	No		
seller_city	varchar(100)	No		
seller_state	varchar(100)	No		
seller_zip	varchar(30)	No		
seller_password	varchar(100)	No		
seller_datetime	varchar(100)	No		
seller_timestamp	varchar(100)	No		
seller_status	int(1)	No		
seller_photo	varchar(255)	Yes	*NULL*	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	seller_id	6	A	No	

Figure 4.6: Data Dictionary for tbl_rating, tbl_seller

tbl_size

Column	Type	Null	Default	Comments
size_id *(Primary)*	int(11)	No		
size_name	varchar(255)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	size_id	28	A	No	

tbl_top_category

Column	Type	Null	Default	Comments
tcat_id *(Primary)*	int(11)	No		
tcat_name	varchar(255)	No		
show_on_menu	int(1)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	tcat_id	3	A	No	

tbl_user

Column	Type	Null	Default	Comments
id *(Primary)*	int(10)	No		
full_name	varchar(100)	No		
email	varchar(255)	No		
phone	varchar(100)	No		
password	varchar(255)	No		
photo	varchar(255)	No		
role	varchar(30)	No		
status	varchar(10)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	2	A	No	

Figure 4.7: Data Dictionary for tbl_size, tbl_category, tbl_user

4.2.4 Physical Design

4.2.4.1 Data Definition Language (DDL)

1) Tbl_color

```
CREATE TABLE `tbl_color` (
`color_id` int(11) NOT NULL,
`color_name` varchar(255) NOT NULL,
PRIMARY KEY (color_id));
```

2) Tbl_customer

```
CREATE TABLE `tbl_customer` (
 `cust_id` int(11) NOT NULL,
 `cust_name` varchar(100) NOT NULL,
 `cust_email` varchar(100) NOT NULL,
 `cust_phone` varchar(50) NOT NULL,
 `cust_country` int(11) NOT NULL,
 `cust_address` text NOT NULL,
 `cust_city` varchar(100) NOT NULL,
 `cust_state` varchar(100) NOT NULL,
 `cust_zip` varchar(30) NOT NULL,
 `cust_password` varchar(100) NOT NULL,
 `cust_datetime` varchar(100) NOT NULL,
 `cust_timestamp` varchar(100) NOT NULL,
 `cust_status` int(1) NOT NULL,
 PRIMARY KEY (cust_id) );
```

3) Tbl-end_category

```
CREATE TABLE `tbl_end_category` (
  `ecat_id` int(11) NOT NULL,
  `ecat_name` varchar(255) NOT NULL,
  `mcat_id` int(11) NOT NULL,
   PRIMARY KEY (ecat_id),
   FOREIGN KEY (mcat_id) REFERENCES
tbl_mid_category(mcat_id));
```

4) Tbl_mid_category

```
CREATE TABLE `tbl_mid_category` (
  `mcat_id` int(11) NOT NULL,
  `mcat_name` varchar(255) NOT NULL,
  `tcat_id` int(11) NOT NULL,
   PRIMARY KEY (mcat_id),
   FOREIGN KEY (tcat_id) REFERENCES tbl_top_category
(tcat_id));
```


5) Tbl_order

```
CREATE TABLE `tbl_order` (
  `order_id` int(11) NOT NULL,
  `product_id` int(11) NOT NULL,
  `product_name` varchar(255) NOT NULL,
  `size` varchar(100) NOT NULL,
  `color` varchar(100) NOT NULL,
  `quantity` varchar(50) NOT NULL,
  `unit_price` varchar(50) NOT NULL,
  `payment_id` int(11) NOT NULL,
  `seller_id` int(11) NOT NULL,
   PRIMARY KEY (order_id),
```

```
  FOREIGN KEY (product_id) REFERENCES tbl_product
(product_id),
  FOREIGN KEY (payment_id) REFERENCES tbl_payment
(payment_id),
  FOREIGN KEY (seller_id) REFERENCES tbl_seller
(seller_id));
```

6) Tbl_payment

```
CREATE TABLE `tbl_payment` (
  `payment_id` int(11) NOT NULL,
  `customer_id` int(11) NOT NULL,
  `customer_name` varchar(255) NOT NULL,
  `customer_email` varchar(255) NOT NULL,
  `payment_date` varchar(50) NOT NULL,
  `paid_amount` int(11) NOT NULL,
  `card_number` varchar(50) NOT NULL,
  `card_cvv` varchar(10) NOT NULL,
  `card_month` varchar(10) NOT NULL,
  `card_year` varchar(10) NOT NULL,
  `bank_name` text NOT NULL,
  `payment_method` varchar(20) NOT NULL,
  `payment_status` varchar(25) NOT NULL,
  `shipping_status` varchar(20) NOT NULL,
  `seller_id` int(11) NOT NULL,
  PRIMARY KEY (payment_id),
  FOREIGN KEY (customer_id) REFERENCES tbl_customer
(customer_id));
```

7) Tbl_product

```
CREATE TABLE `tbl_product` (
  `p_id` int(11) NOT NULL,
  `p_name` varchar(255) NOT NULL,
  `p_old_price` varchar(10) NOT NULL,
  `p_current_price` varchar(10) NOT NULL,
  `p_qty` int(10) NOT NULL,
  `p_featured_photo` varchar(255) NOT NULL,
  `p_description` text NOT NULL,
  `p_short_description` text NOT NULL,
  `p_feature` text NOT NULL,
  `p_condition` text NOT NULL,
  `p_return_policy` text NOT NULL,
  `p_total_view` int(11) NOT NULL,
  `p_is_featured` int(1) NOT NULL,
  `p_is_active` int(1) NOT NULL,
  `ecat_id` int(11) NOT NULL,
  `seller_id` int(11) NOT NULL,
  PRIMARY KEY (p_id),
  FOREIGN KEY (ecat_id) REFERENCES tbl_end_category
(ecat_id),
FOREIGN KEY (seller_id) REFERENCES tbl_seller
(seller_id));
```

8) Tbl_product_color

```
CREATE TABLE `tbl_product_color` (
  `id` int(11) NOT NULL,
  `color_id` int(11) NOT NULL,
  `p_id` int(11) NOT NULL,
   PRIMARY KEY (id),
   FOREIGN KEY (color_id) REFERENCES tbl_color (color_id),
FOREIGN KEY (p_id) REFERENCES tbl_product (color_id));
```

9) Tbl_product_photo

```
CREATE TABLE `tbl_product_photo` (
`pp_id` int(11) NOT NULL,
`photo` varchar(255) NOT NULL,
`p_id` int(11) NOT NULL,
   PRIMARY KEY (pp_id),
   FOREIGN KEY (p_id) REFERENCES tbl_product (p_id));
```


10) Tbl_product_size

```
CREATE TABLE `tbl_product_size` (
  `id` int(11) NOT NULL,
  `size_id` int(11) NOT NULL,
  `p_id` int(11) NOT NULL,
   PRIMARY KEY (id),
   FOREIGN KEY (size_id) REFERENCES tbl_size (size_id),
FOREIGN KEY (p_id) REFERENCES tbl_product (p_id));
```

11) Tbl_rating

```
CREATE TABLE `tbl_rating` (
  `rt_id` int(11) NOT NULL,
  `p_id` int(11) NOT NULL,
  `cust_id` int(11) NOT NULL,
  `comment` text DEFAULT NULL,
  `rating` float NOT NULL,
   PRIMARY KEY (rt_id),
   FOREIGN KEY (p_id) REFERENCES tbl_product (p_id),
FOREIGN KEY (cust_id) REFERENCES tbl_customer (cust_id));
```

12) Tbl_seller

```
CREATE TABLE `tbl_seller` (
  `seller_id` int(11) NOT NULL,
  `seller_name` varchar(100) NOT NULL,
  `shopName` varchar(255) DEFAULT NULL,
  `seller_email` varchar(100) NOT NULL,
  `seller_phone` varchar(50) NOT NULL,
  `seller_country` int(11) NOT NULL,
  `seller_address` text NOT NULL,
  `seller_city` varchar(100) NOT NULL,
  `seller_state` varchar(100) NOT NULL,
  `seller_zip` varchar(30) NOT NULL,
  `seller_password` varchar(100) NOT NULL,
  `seller_datetime` varchar(100) NOT NULL,
  `seller_timestamp` varchar(100) NOT NULL,
  `seller_status` int(1) NOT NULL,
  `seller_photo` varchar(255) DEFAULT NULL,
   PRIMARY KEY (seller_id));
```

13) Tbl_size

```
CREATE TABLE `tbl_size` (
  `size_id` int(11) NOT NULL,
  `size_name` varchar(255) NOT NULL,
PRIMARY KEY (size_id));
```

14) Tbl_top_category

```
CREATE TABLE `tbl_top_category` (
  `tcat_id` int(11) NOT NULL,
  `tcat_name` varchar(255) NOT NULL,
  `show_on_menu` int(1) NOT NULL,
PRIMARY KEY (tcat_id));
```

15) Tbl_user

```
CREATE TABLE `tbl_user` (
  `id` int(10) NOT NULL,
  `full_name` varchar(100) NOT NULL,
  `email` varchar(255) NOT NULL,
  `phone` varchar(100) NOT NULL,
  `password` varchar(255) NOT NULL,
  `photo` varchar(255) NOT NULL,
  `role` varchar(30) NOT NULL,
  `status` varchar(10) NOT NULL,
PRIMARY KEY (id));
```

4.2.4.2 The Usage of Stored Procedures, Triggers

Stored procedures are a piece of the code in written in PL/SQL to do some specific tasks. Stored procedures can be invoked explicitly by the user. It's like a java program it can take some input as a parameter then can do some processing and can return values.

On the other hand, trigger is a stored procedure that runs automatically when various events happen such update, insert, delete. Triggers are more like an event handler they run at the specific event. Trigger can not take input and they can't return values.

1) Procedure GetSeller

```
BEGIN
  SELECT COUNT(seller_id)
  FROM tbl_seller;
END
```

2) Procedure GetCustomer

```
BEGIN
        SELECT * FROM tbl_customer;
END
```

3) Procedure GetCustomerDetail

```
BEGIN
SELECT *

        FROM tbl_customer t1

        JOIN tbl_country t2

        ON t1.cust_country = t2.country_id;
  END
```

4) Procedure GetSellerDetail

```
BEGIN
SELECT *
        FROM tbl_seller t1
        JOIN tbl_country t2
        ON t1.seller_country = t2.country_id;
END
```

5) Trigger

```
CREATE TRIGGER after_cust_insert
AFTER INSERT
ON tbl_customer FOR EACH ROW
BEGIN
   IF NEW.cust_email IS NULL THEN
      INSERT INTO reminders(cust_id, message)
      VALUES(new.id,CONCAT('Hi  ',  NEW.cust_name,  ',
please update your email.'));
   END IF;
```

4.3 Graphical User Interface (GUI) Design

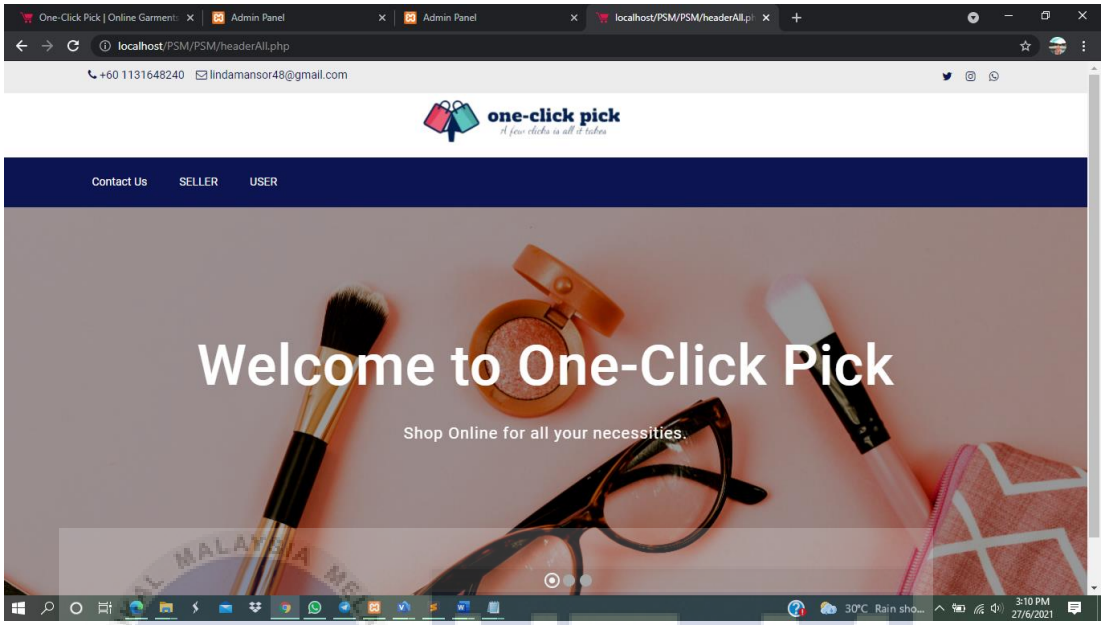

Figure 4.8: One-Click Pick Home page

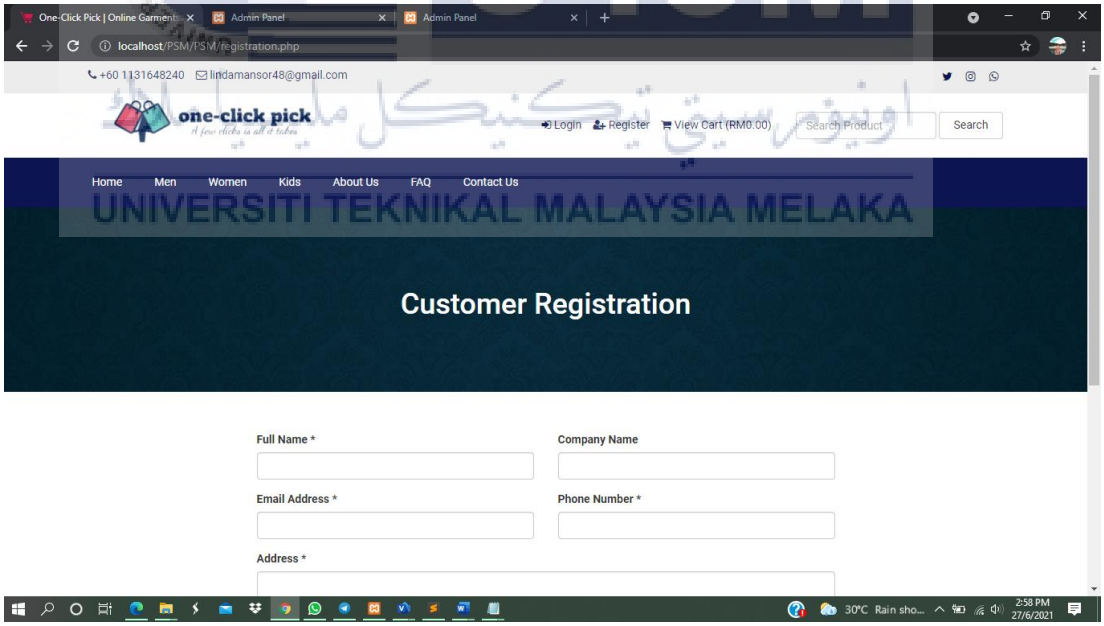

Figure 4.9: One-Click Pick Customer Registration Page

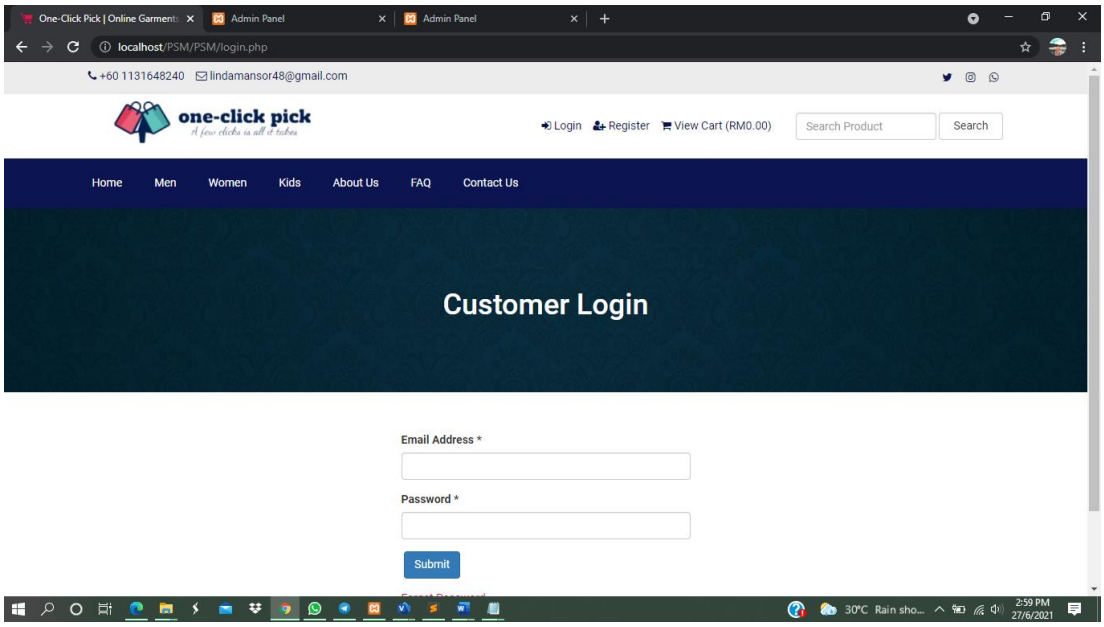

Figure 4.10: One-Click Pick Customer Login Page

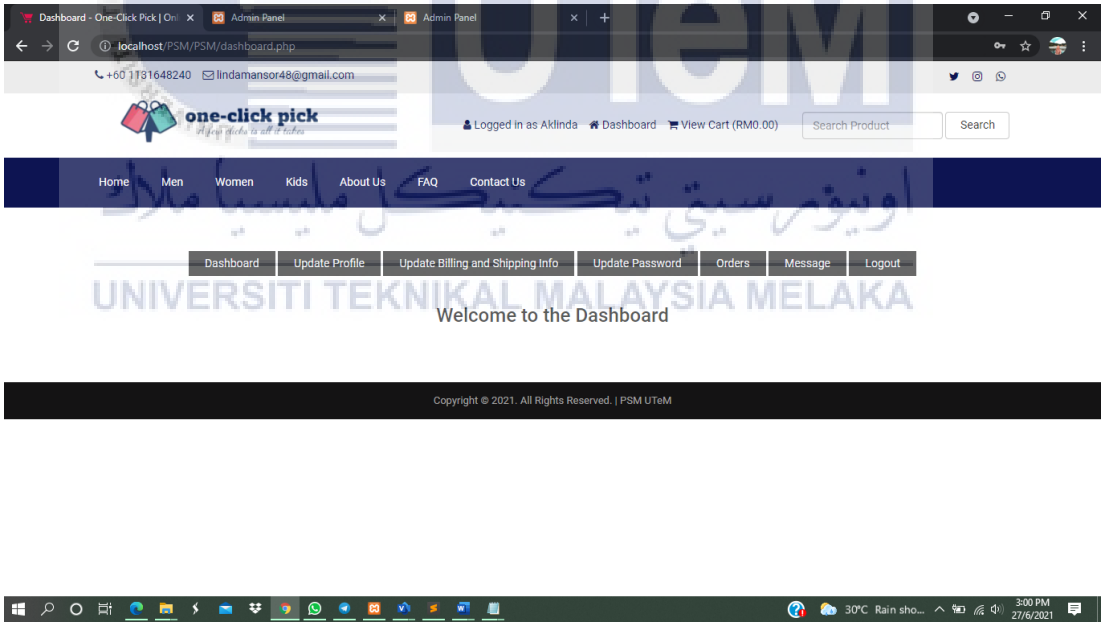

Figure 4.11: One-Click Pick Customer Dashboard Page

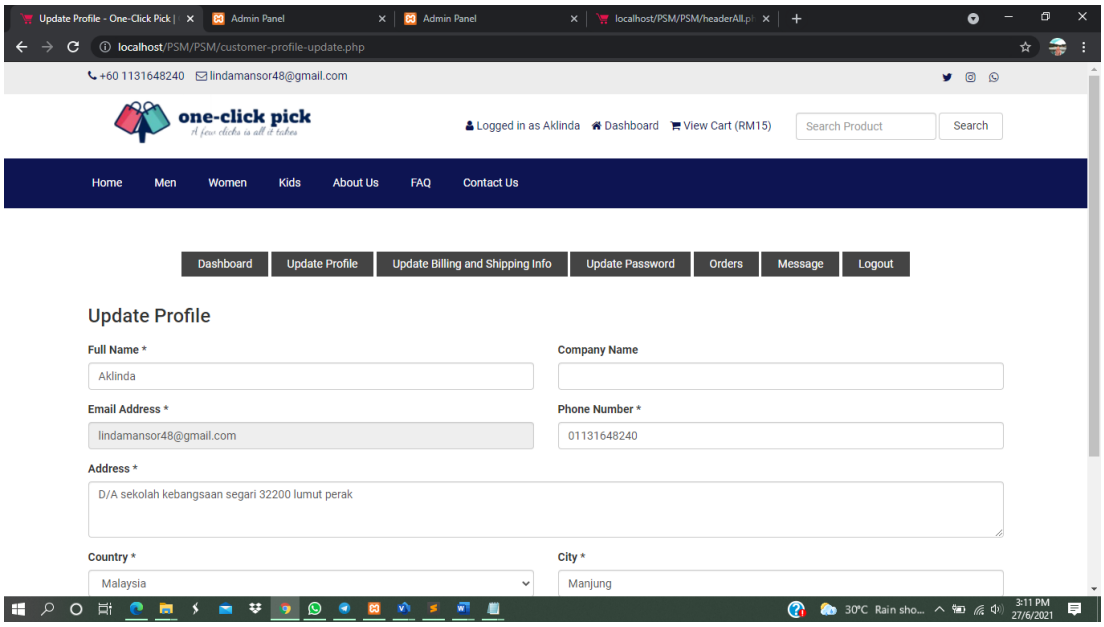

Figure 4.12: One-Click Pick Customer Update Profile Page

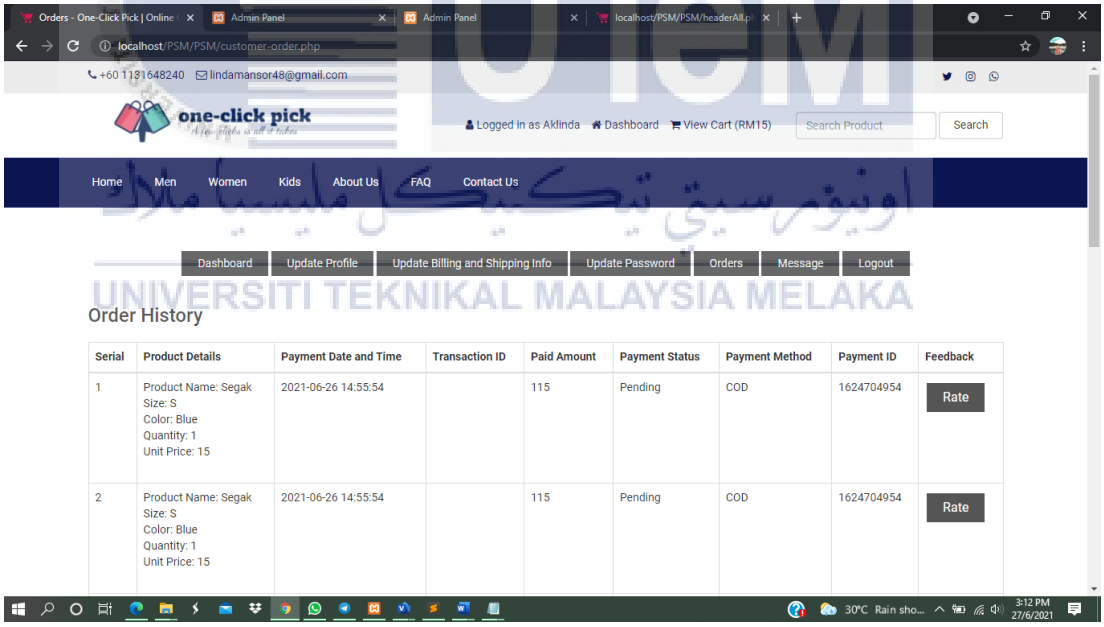

Figure 4.13: One-Click Pick Customer Order History Page

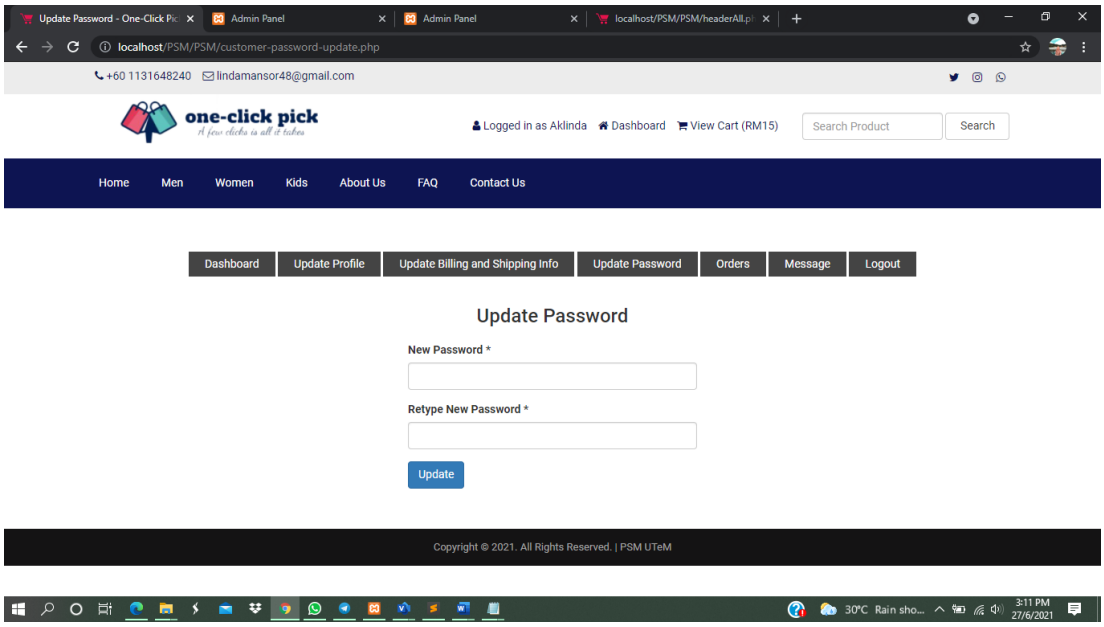

Figure 4.14: One-Click Pick Customer Update Password Page

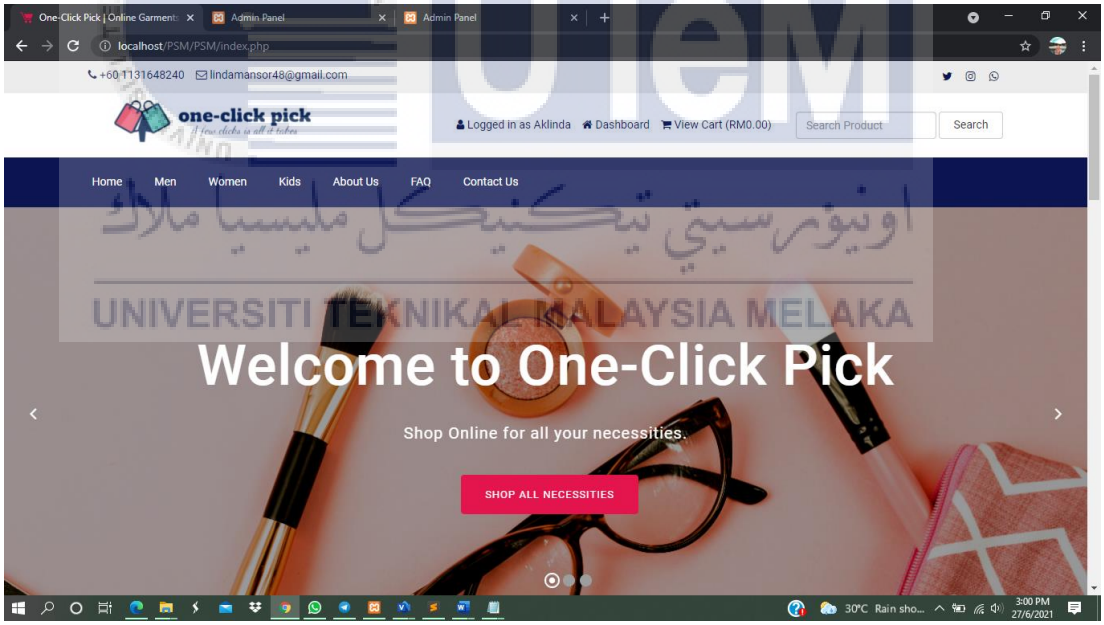

Figure 4.15: One-Click Pick Customer Front Page

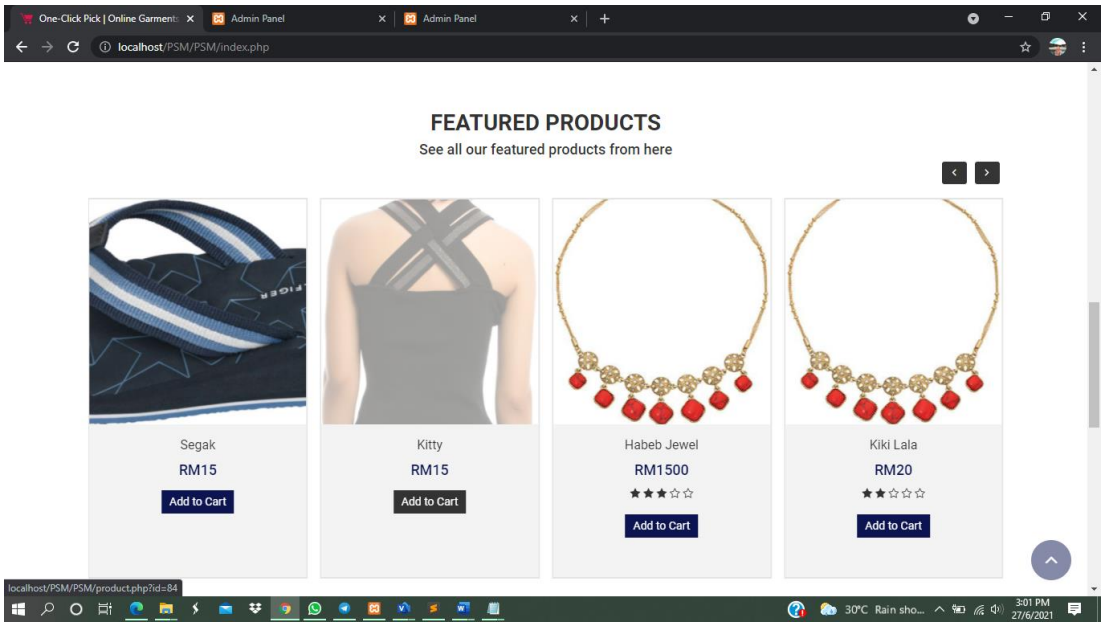

Figure 4.16: One-Click Pick Customer Front Page

Figure 4.17: One-Click Pick Customer Cart Page

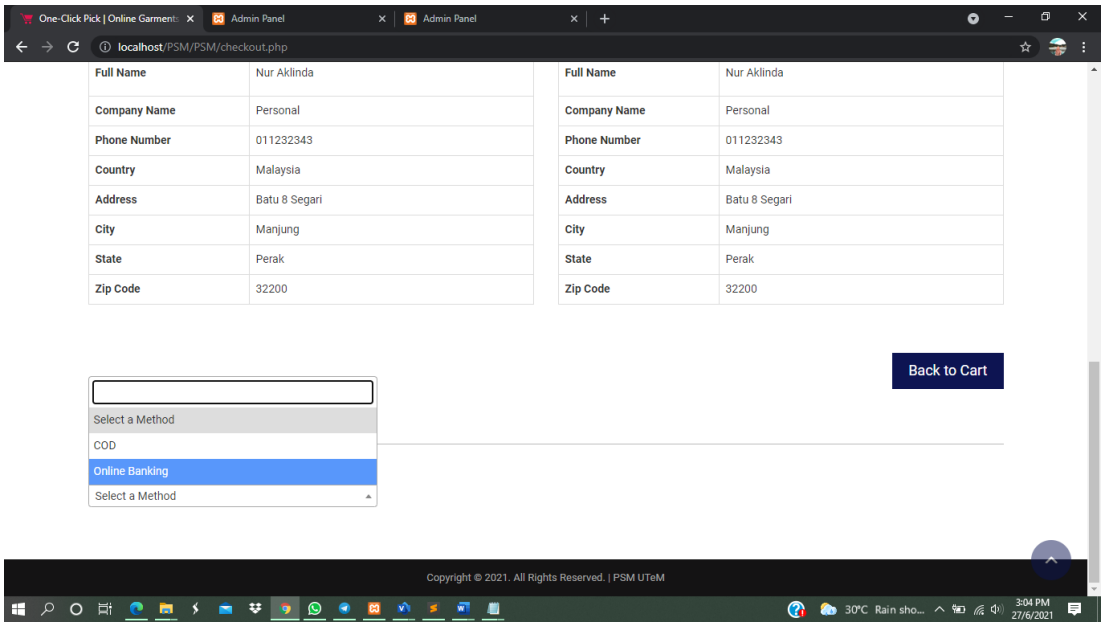

Figure 4.18: One-Click Pick Customer Payment Page

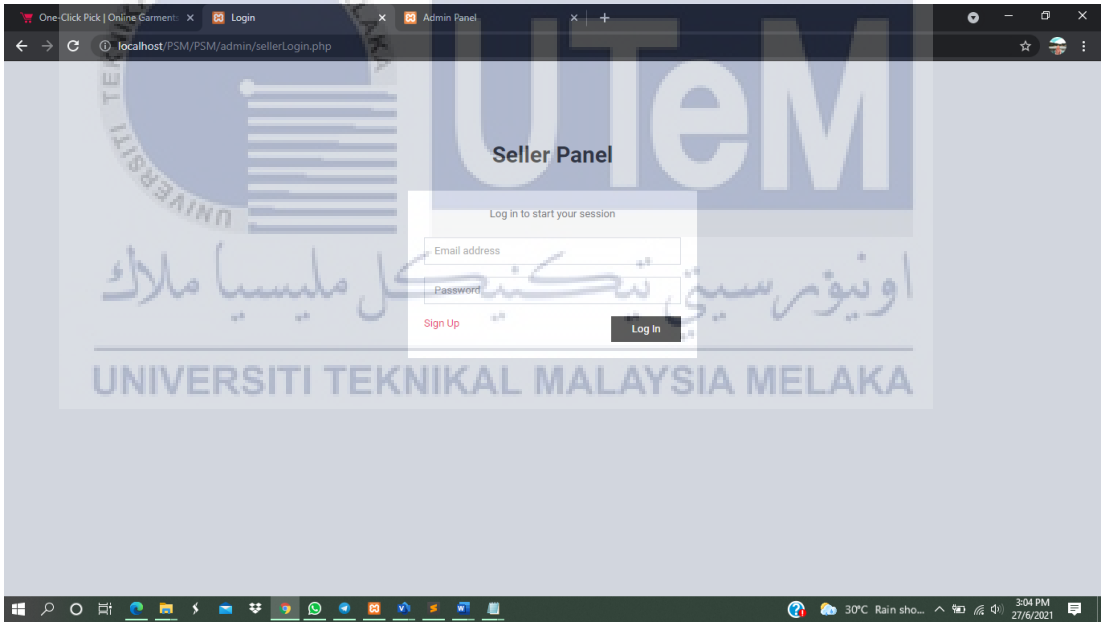

Figure 4.19: One-Click Pick Seller Login Page

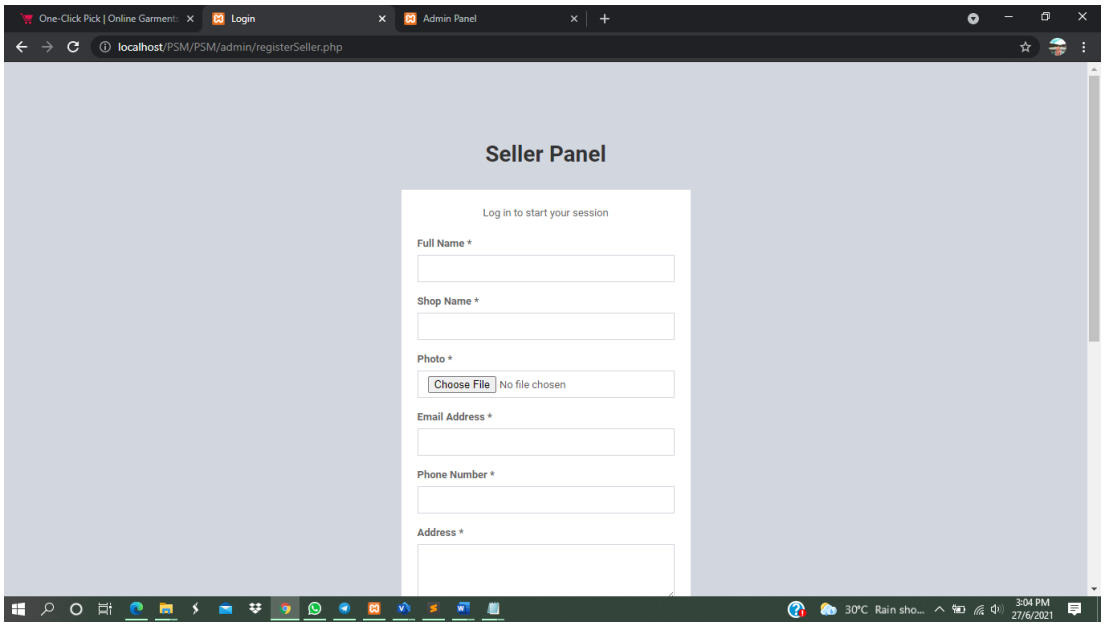

Figure 4.20: One-Click Pick Seller Registration Page

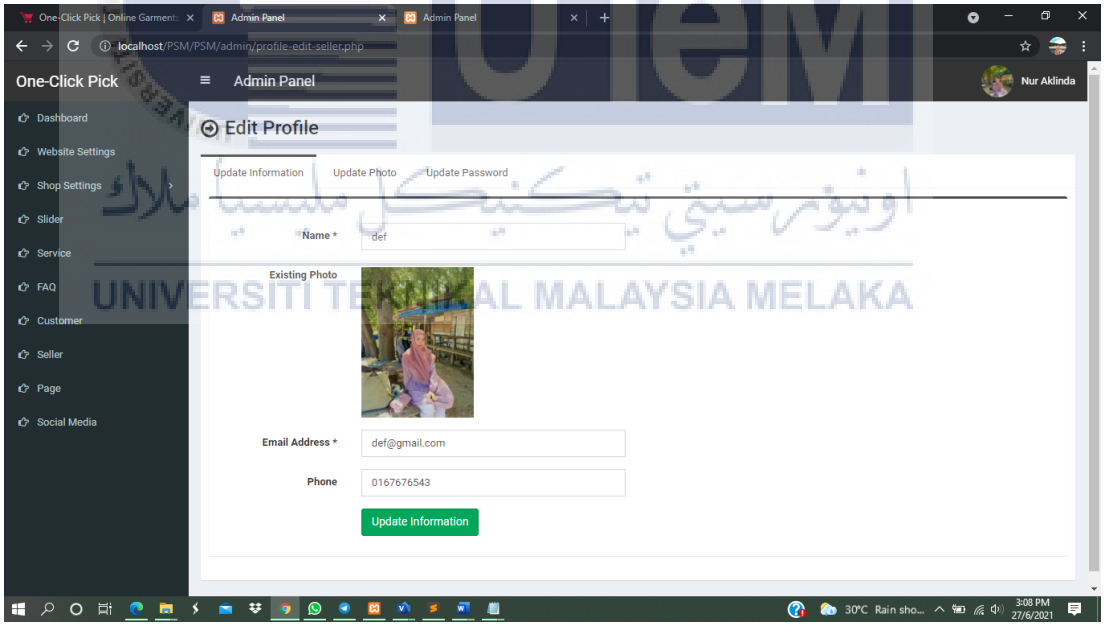

Figure 4.21: One-Click Pick Seller Edit Profile Page

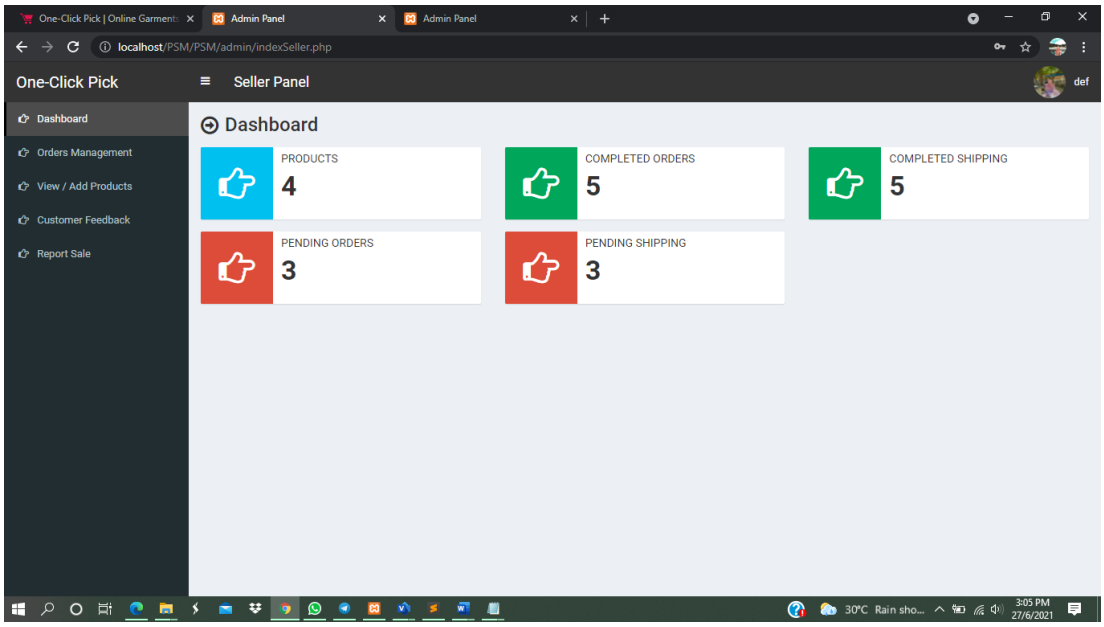

Figure 4.22: One-Click Pick Seller Dashboard Page

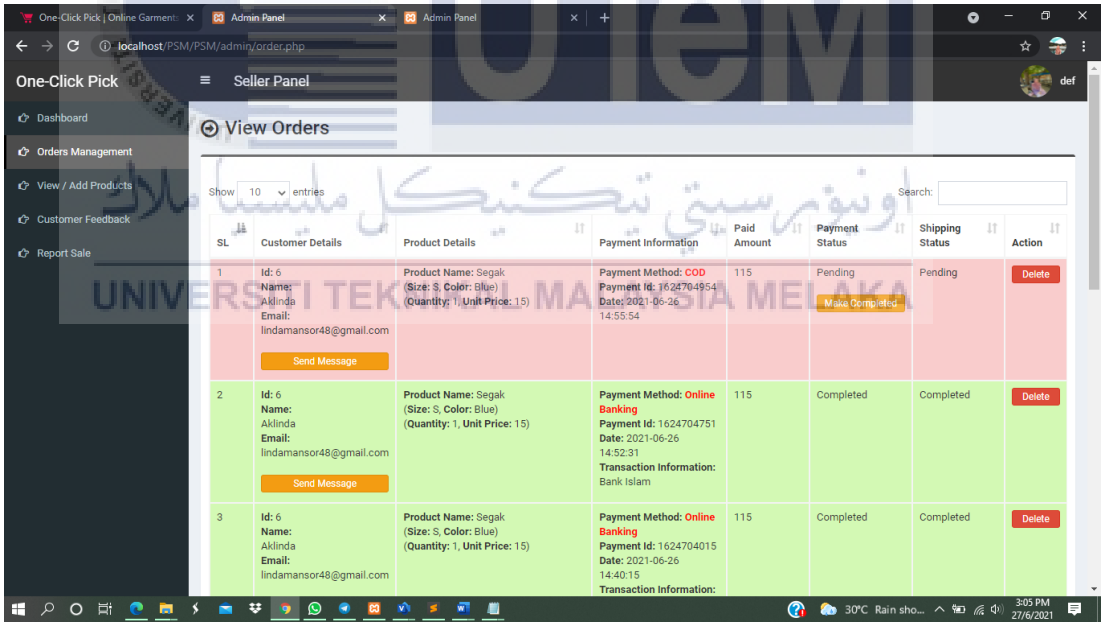

Figure 4.23: One-Click Pick Seller Order Management Page

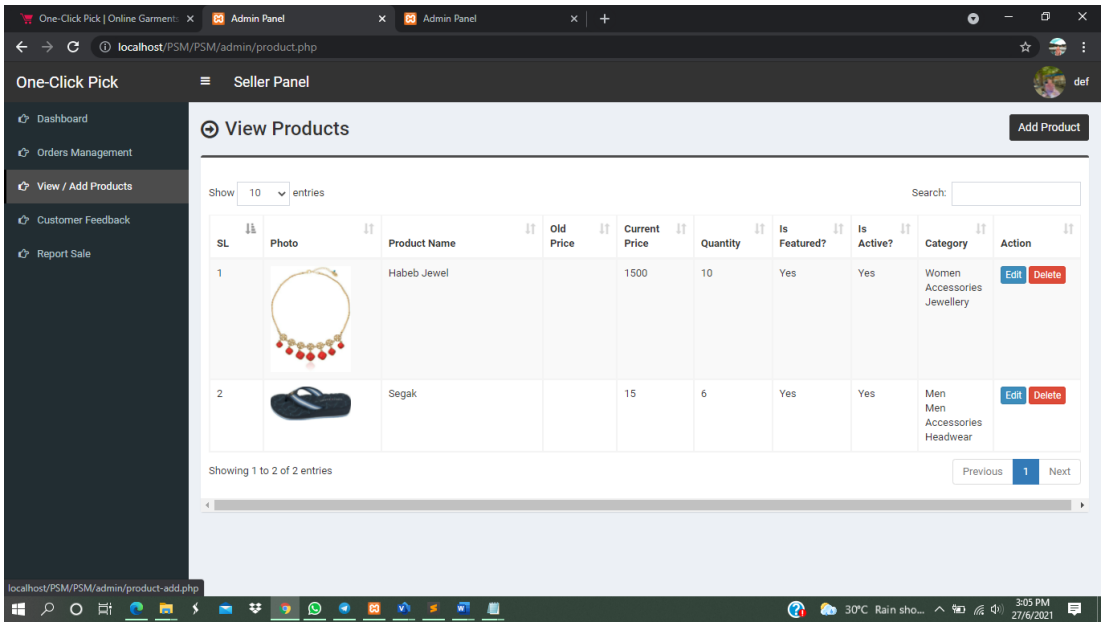

Figure 4.24: One-Click Pick Seller View Product Page

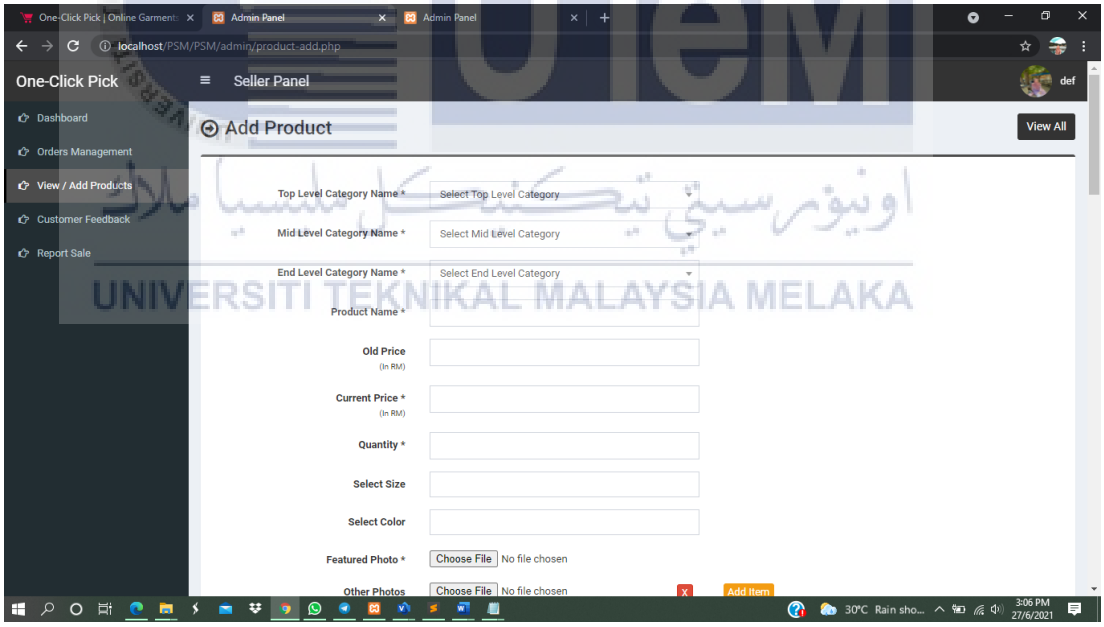

Figure 4.25: One-Click Pick Add Product Page

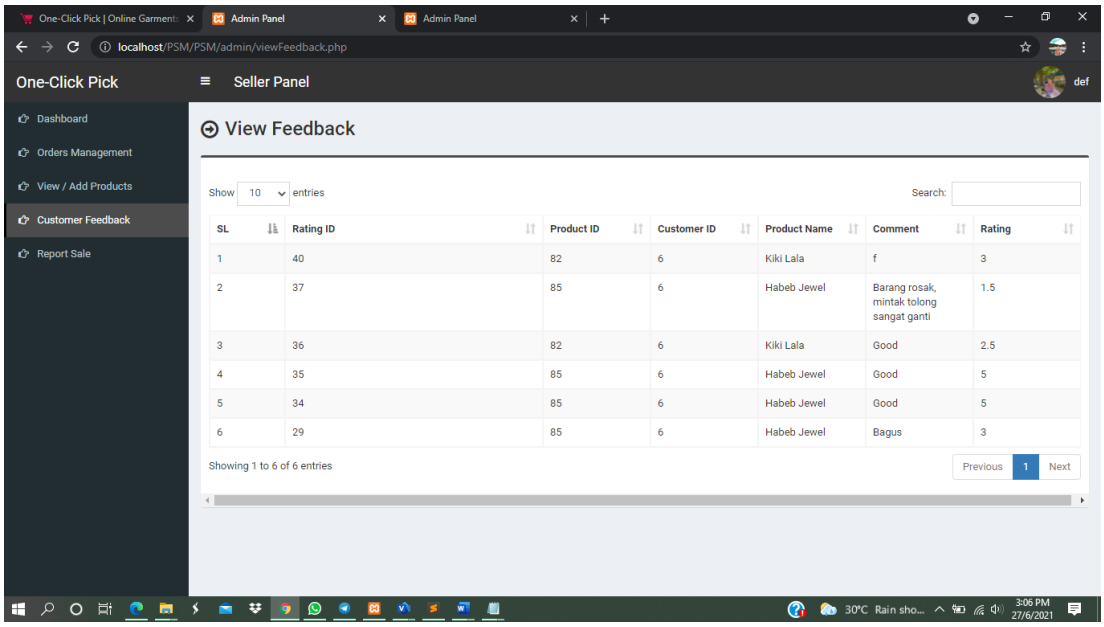

Figure 4.26: One-Click Pick Customer Feedback Page

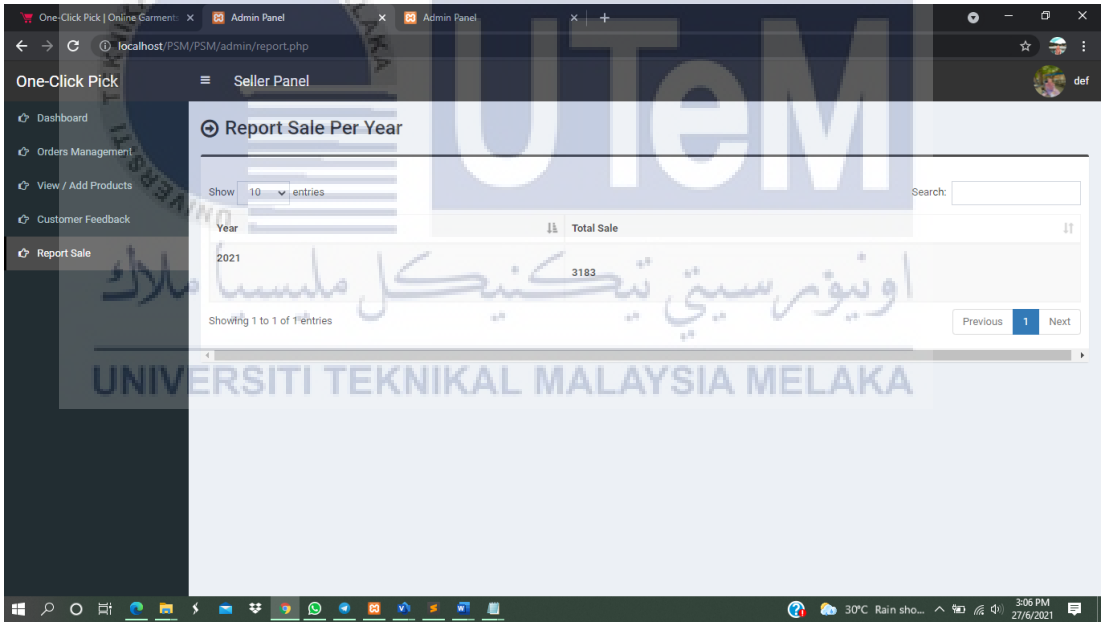

Figure 4.27: One-Click Pick Seller Report Sale Per Year Page

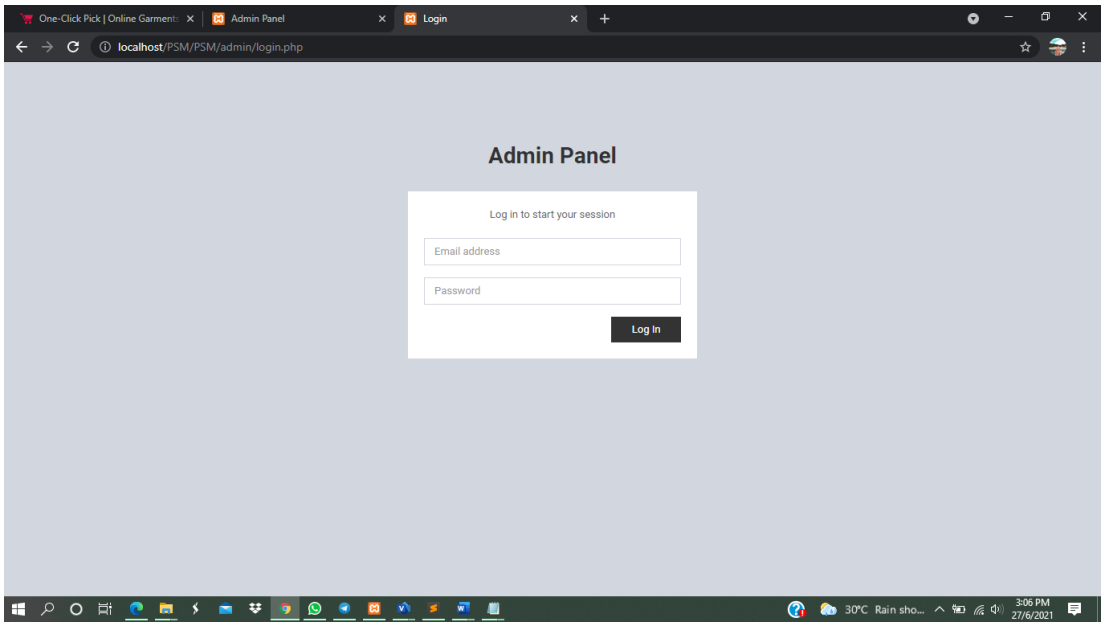

Figure 4.28: One-Click Pick Admin Login Page

Figure 4.29: One-Click Pick Admin Dashboard Page

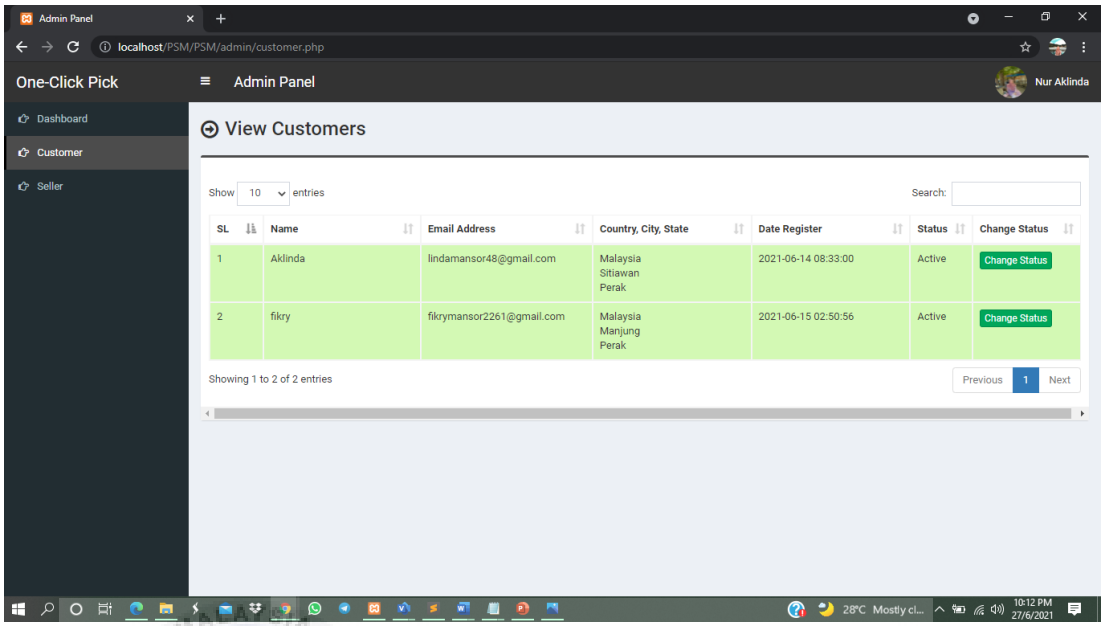

Figure 4.30: One-Click Pick View Customer Page

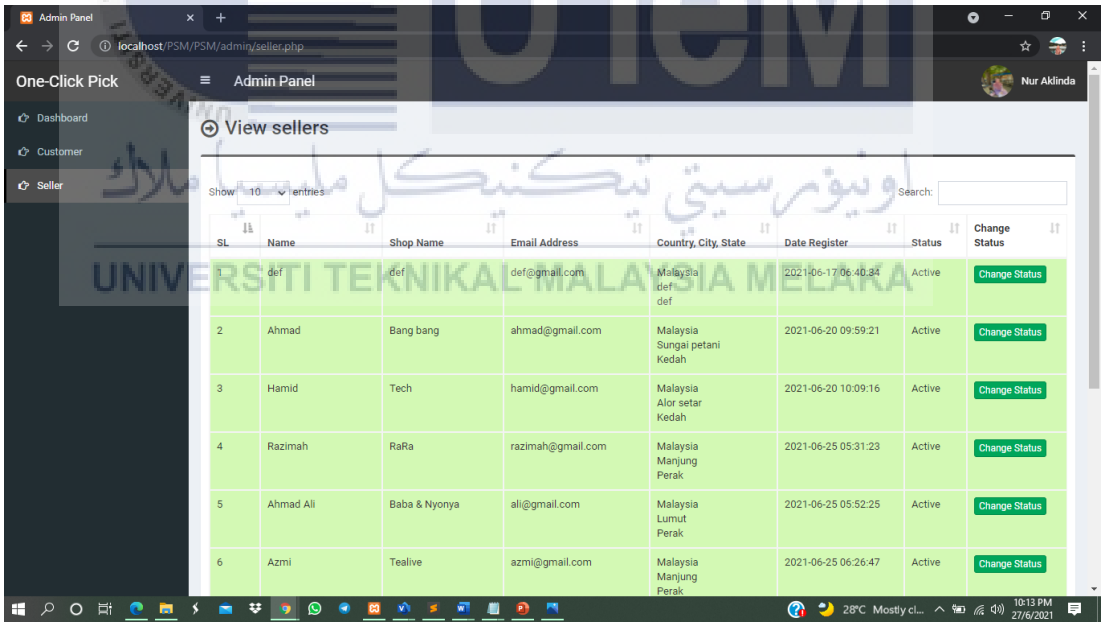

Figure 4.31: One-Click Pick View Seller Page

4.4 Conclusion

In conclusion, designing the database is important because this can provide a solution to the problems specified in the requirement document in analysis phase. The design document for this system act as a plan or blue print for the solution and will be used later for implementation, testing and maintenance. Based on the logical, physical and data dictionary that are provided in this document, overall, the database structure are created with the specific modules that will be developed in MySQL database using phpMyAdmin. The relationship between each entity that depends on each other will be shown by an overall view of the conceptual design.

CHAPTER 5: IMPLEMENTATION

5.1 Introduction

This chapter discusses about the implementation of software development environment setup and system database implementation. The software developing environment will be explained the steps of installation of teh software, executing and configuring the software and database.

For the database implementation, it will be explained about DBMS that was the schedule which is MySQL Database using phpMyAdmin with XAMPP Server, the Data Definition Language (DDL), Data Manipulation Language (DML) and lastly about the main processes such as stored procedures and trigger that implemented on this system using PLSQL programming language.

5.2 Software Development Environment Setup

The software that is used for the development process of the One-Click Pick (Online Shopping System) is the Sublime Text 3 as a platform to write a PHP code. Then for database management, this system used MySQL Database to store data and the database object using phpMyAdmin to handle the administration of MySQL. Lastly, for the server platform, this system used XAMPP Server to be a server that will handle the database and the system together.

5.2.1 Software Environment Setup XAMPP Server

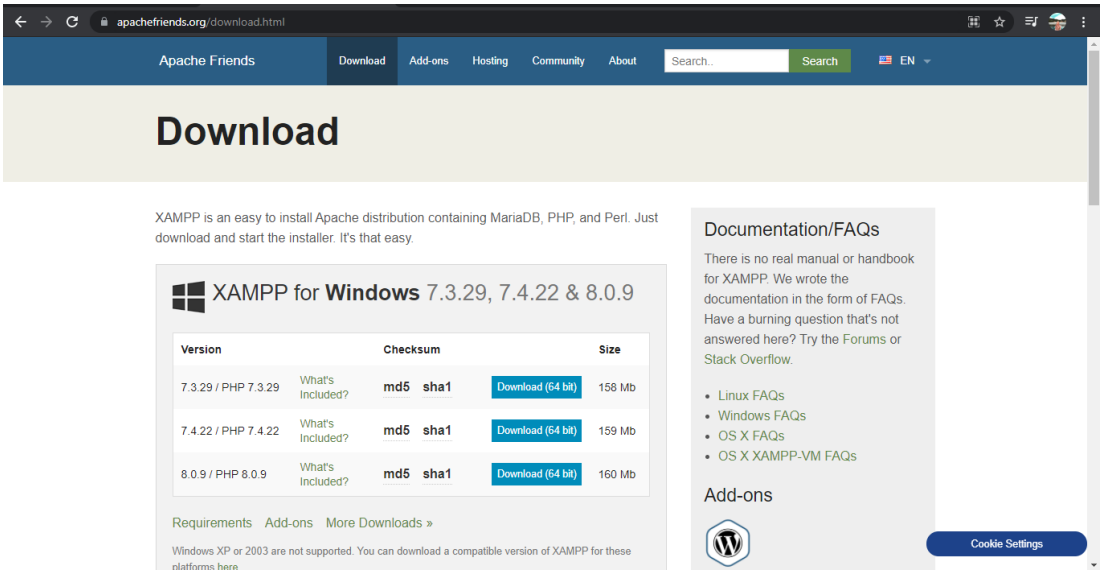

Figure 5.1: XAMPP Server Web Sites Download Folder

Step 1: Go to https://www.apachefriends.org/download.html and start to download the XAMPP Server installer. Figure 5.1 show the web site to download the XAMPP Server installer. Make sure before downloading, choose the right version of the XAMPP Server installer that will be used on the computer configuration either 64 bit (x64) or 32 bit (x86), and also choose the right operating system for the computer.

Figure 5.2: User Account Control (UAC)

Step 2: Click 'OK' to continue the installation if windows pop up warning about User Account Control (UAC)

Figure 5.3: XAMPP Setup Wizard

Step 3: Next, Welcome to the XAMPP Setup Wizard will appear. Click ‘Next’ to continue the installation.

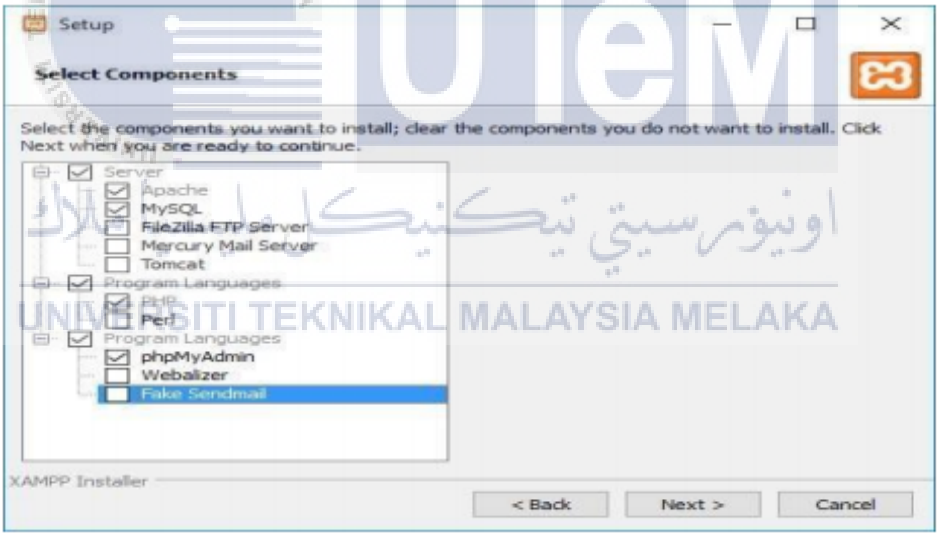

Figure 5.4: Components Selection Interface

Step 4: The component selection screen will appear next. This screen asked to make the selection which components of the software would like to install and which ones do not want. Click ‘Next’ to continue.

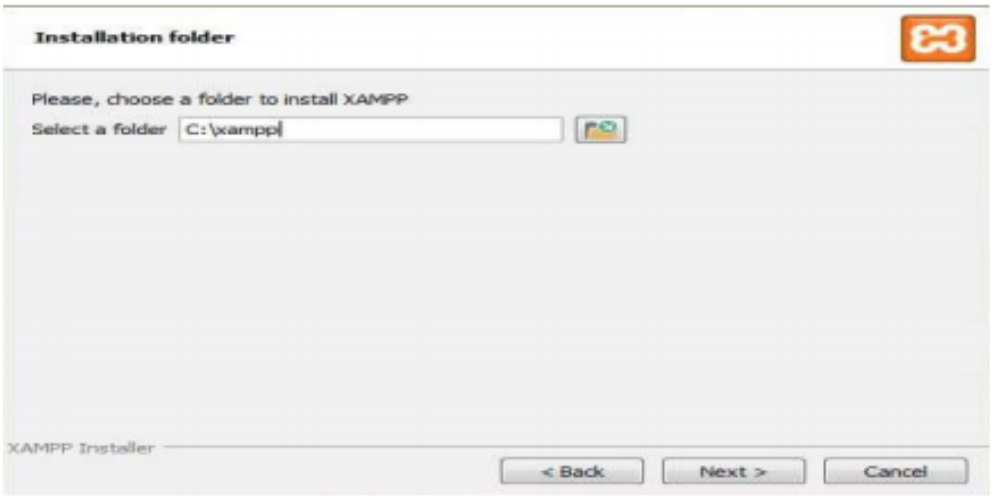

Figure 5.5: Choose the Installation Directory

Step 5: In this next step, users need to choose the directory for the XAMPP software packet to be installed. If the user chooses for the standard setup, then the folder name XAMPP will be created under C:\. To continue the installation, click 'Next'.

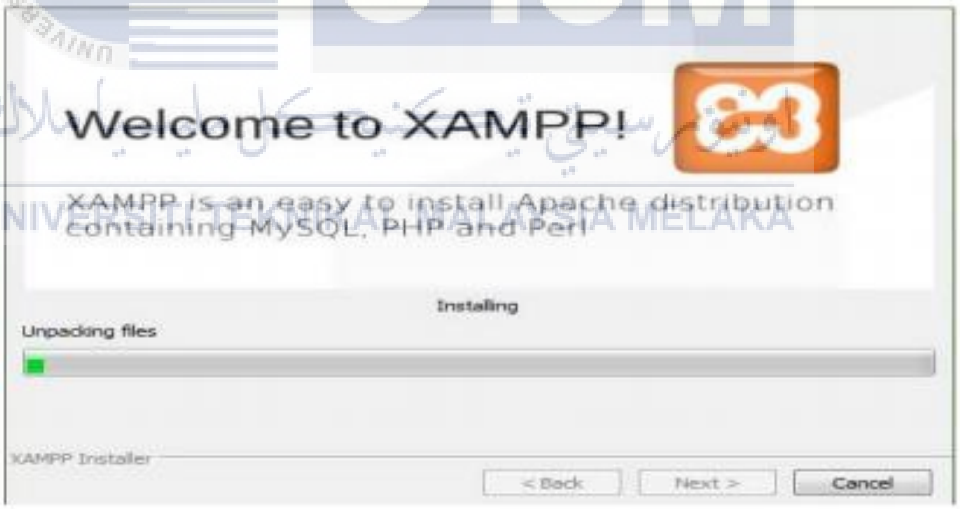

Figure 5.6: Installation Process Started

Step 6: XAMPP will be started installing the files to the location selected in the previous step.

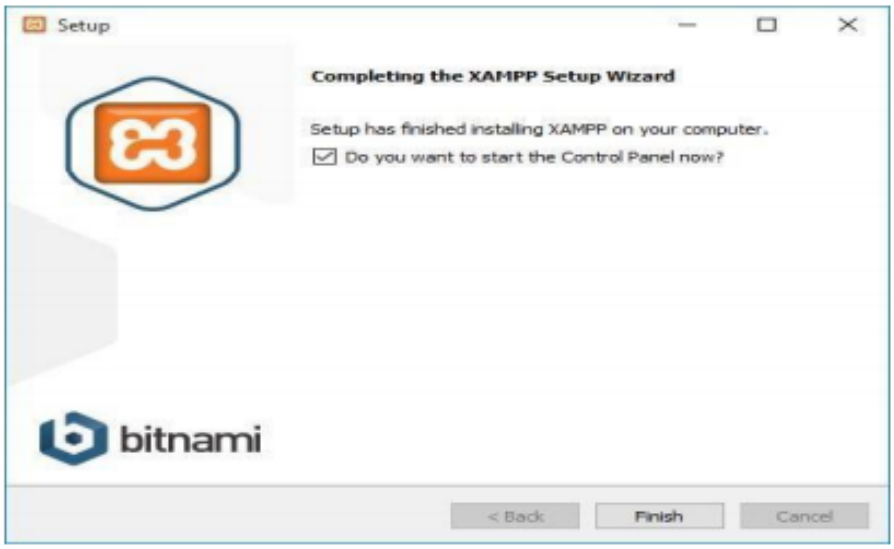

Figure 5.7: Complete Installation

Step 7: After all the components are unpacked and installed, users can click 'Finish'. Click to tick the corresponding and open the XAMPP Control Panel.

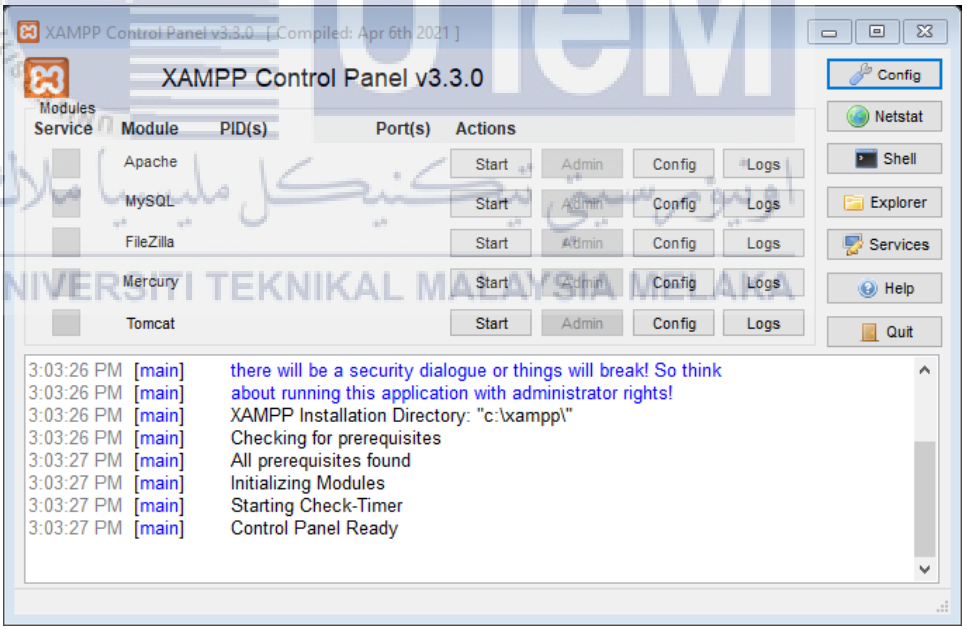

Figure 5.8: Starting XAMPP Modules

Step 8: The Control Panel will automatically open, but if users unchecked the option in the previous window, users can search the XAMPP folder and open the XAMPP Control Panel manually. After that, modules can be started or stopped on the XAMPP Control Panel on the buttons under 'Actions'.

5.2.2 Database Environment Setup

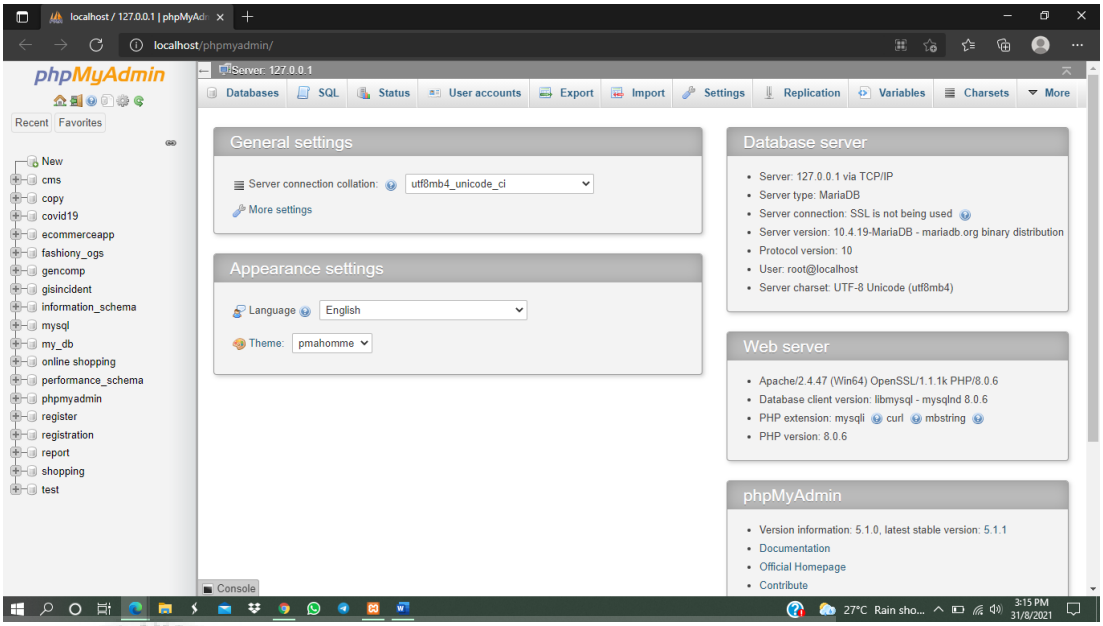

Figure 5.9: phpMyAdmin First Page

Step1: Users can access phpMyAdmin by clicking on the button 'Admin' on the same line with MySQL in XAMPP. The phpMyAdmin first screen will appear as in Figure 5.9. On the left is a list of databases that already exist. Users need to create a new database, click Databases in the top navbar, or click the SQL.

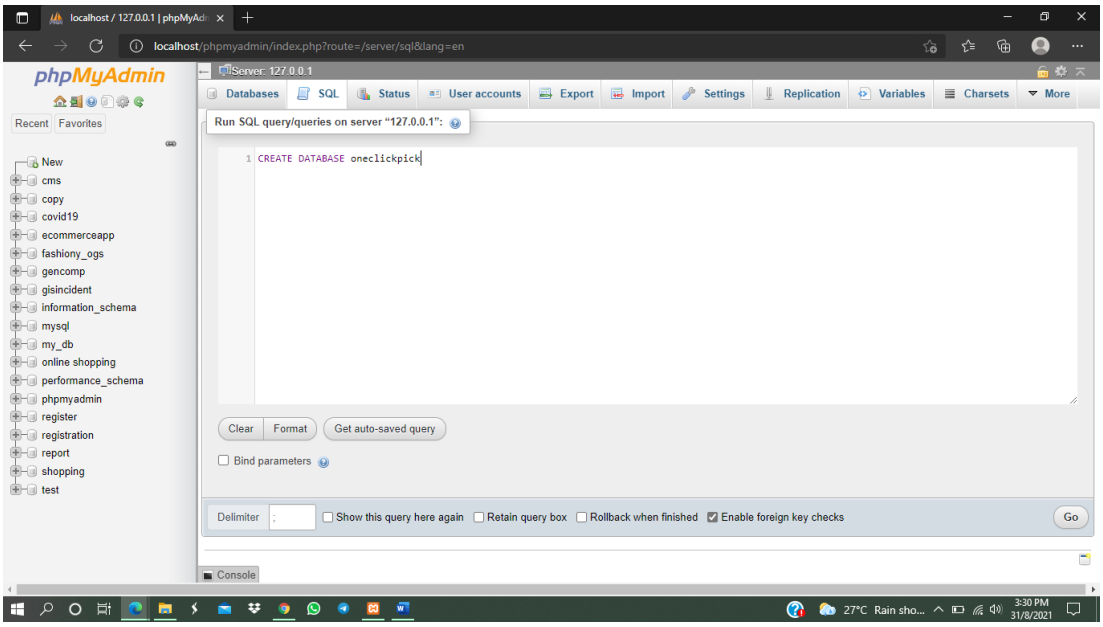

Figure 5.10: Creating A Database as 'oneclickpick'

Step 2: Using SQL language, type the command to create a new database such as 'oneclickpick' for One-Click Pick (Online Shopping System).

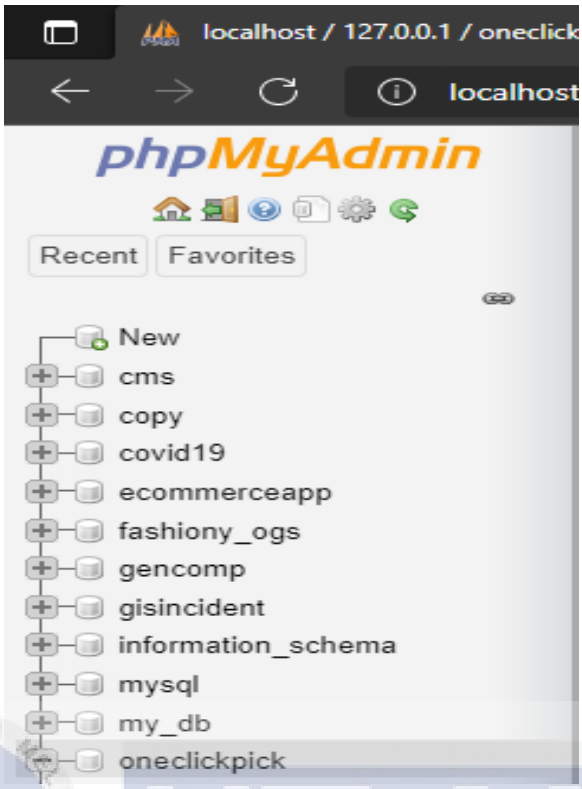

Figure 5.11: List Databases That Created

Step 3: After the database is successfully created, the message successfully will pop up. Users can view the databases that are created and can start using the database such as create a new table.

5.3 Database Implementation

The database implementation phase explained the Data Definition Language (DDL), trigger and store procedures to shows the functionality in this system.

5.3.1 Data Definition Language (DDL)

In the implementation phase of the database, Data Definition Language (DDL) is primarily used by database administrators to create the database and table. Below are shown the DDL that is used in One-Click Pick (Online Shopping System).

5.3.1.1 Create Database

The first thing users need to create the database by using the ‘Create Database’ command. After that user can create any new table in the database. The database name for this system is ‘oneclickpick’. Figure 5.12 shows the command for creating a database.

```
CREATE DATABASE oneclickpick
```

Figure 5.12: Create Database

5.3.1.2 Create Table

After the database is successfully created, the user needs to create the table in the database by using the ‘Create Table’ command and need to add the attribute name, data type, field length, constraint, primary key and foreign key. Figure 5.13 to Figure 5.27 shows all the tables that are created into the oneclickpick that required for develop One-Click Pick (Online Shopping System).

1) **Tbl_color**

```
CREATE TABLE `tbl_color` (
`color_id` int(11) NOT NULL,
`color_name` varchar(255) NOT NULL,
PRIMARY KEY (color_id));
```

Figure 5.13: Create Table Tbl_color

2) Tbl_customer

```
CREATE TABLE `tbl_customer` (
  `cust_id` int(11) NOT NULL,
  `cust_name` varchar(100) NOT NULL,
  `cust_email` varchar(100) NOT NULL,
  `cust_phone` varchar(50) NOT NULL,
  `cust_country` int(11) NOT NULL,
  `cust_address` text NOT NULL,
  `cust_city` varchar(100) NOT NULL,
  `cust_state` varchar(100) NOT NULL,
  `cust_zip` varchar(30) NOT NULL,
  `cust_password` varchar(100) NOT NULL,
  `cust_datetime` varchar(100) NOT NULL,
  `cust_timestamp` varchar(100) NOT NULL,
  `cust_status` int(1) NOT NULL,
  PRIMARY KEY (cust_id) );
```

Figure 5.14: Create Table Tbl_customer

3) Tbl_end_category

```
CREATE TABLE `tbl_end_category` (
  `ecat_id` int(11) NOT NULL,
  `ecat_name` varchar(255) NOT NULL,
  `mcat_id` int(11) NOT NULL,
  PRIMARY KEY (ecat_id),
  FOREIGN KEY (mcat_id) REFERENCES
tbl_mid_category(mcat_id));
```

Figure 5.15: Create Table Tbl_end_category

4) Tbl_mid_category

```
CREATE TABLE `tbl_mid_category` (
  `mcat_id` int(11) NOT NULL,
  `mcat_name` varchar(255) NOT NULL,
  `tcat_id` int(11) NOT NULL,
   PRIMARY KEY (mcat_id),
   FOREIGN KEY (tcat_id) REFERENCES tbl_top_category
(tcat_id));
```

Figure 5.16: Create Table Tbl_mid_category

5) Tbl_order

```
CREATE TABLE `tbl_order` (
  `order_id` int(11) NOT NULL,
  `product_id` int(11) NOT NULL,
  `product_name` varchar(255) NOT NULL,
  `size` varchar(100) NOT NULL,
  `color` varchar(100) NOT NULL,
  `quantity` varchar(50) NOT NULL,
  `unit_price` varchar(50) NOT NULL,
  `payment_id` int(11) NOT NULL,
  `seller_id` int(11) NOT NULL,
   PRIMARY KEY (order_id),
   FOREIGN KEY (product_id) REFERENCES tbl_product
(product_id),
   FOREIGN KEY (payment_id) REFERENCES tbl_payment
(payment_id),
   FOREIGN KEY (seller_id) REFERENCES tbl_seller
(seller_id));
```

Figure 5.17: Create Table Tbl_order

6) Tbl_payment

```
CREATE TABLE `tbl_payment` (
  `payment_id` int(11) NOT NULL,
  `customer_id` int(11) NOT NULL,
  `customer_name` varchar(255) NOT NULL,
  `customer_email` varchar(255) NOT NULL,
  `payment_date` varchar(50) NOT NULL,
  `paid_amount` int(11) NOT NULL,
  `card_number` varchar(50) NOT NULL,
  `card_cvv` varchar(10) NOT NULL,
  `card_month` varchar(10) NOT NULL,
  `card_year` varchar(10) NOT NULL,
  `bank_name` text NOT NULL,
  `payment_method` varchar(20) NOT NULL,
  `payment_status` varchar(25) NOT NULL,
  `shipping_status` varchar(20) NOT NULL,
  `seller_id` int(11) NOT NULL,
  PRIMARY KEY (payment_id),
  FOREIGN KEY (customer_id) REFERENCES tbl_customer
(customer_id));
```


Figure 5.18: Create Table Tbl_payment

7) Tbl_product

```
CREATE TABLE `tbl_product` (
  `p_id` int(11) NOT NULL,
  `p_name` varchar(255) NOT NULL,
  `p_old_price` varchar(10) NOT NULL,
  `p_current_price` varchar(10) NOT NULL,
  `p_qty` int(10) NOT NULL,
  `p_featured_photo` varchar(255) NOT NULL,
  `p_description` text NOT NULL,
  `p_short_description` text NOT NULL,
  `p_feature` text NOT NULL,
  `p_condition` text NOT NULL,
  `p_return_policy` text NOT NULL,
  `p_total_view` int(11) NOT NULL,
  `p_is_featured` int(1) NOT NULL,
  `p_is_active` int(1) NOT NULL,
  `ecat_id` int(11) NOT NULL,
  `seller_id` int(11) NOT NULL,
  PRIMARY KEY (p_id),
  FOREIGN KEY (ecat_id) REFERENCES tbl_end_category
(ecat_id),
FOREIGN KEY (seller_id) REFERENCES tbl_seller
(seller_id));
```

Figure 5.19: Create Table Tbl_product

8) Tbl_product_color

```
CREATE TABLE `tbl_product_color` (
  `id` int(11) NOT NULL,
  `color_id` int(11) NOT NULL,
  `p_id` int(11) NOT NULL,
   PRIMARY KEY (id),
   FOREIGN KEY (color_id) REFERENCES tbl_color (color_id),
FOREIGN KEY (p_id) REFERENCES tbl_product (color_id));
```

Figure 5.20: Create Table Tbl_product_color

9) Tbl_product_photo

```
CREATE TABLE `tbl_product_photo` (
`pp_id` int(11) NOT NULL,
`photo` varchar(255) NOT NULL,
`p_id` int(11) NOT NULL,
  PRIMARY KEY (pp_id),
  FOREIGN KEY (p_id) REFERENCES tbl_product (p_id));
```


Figure 5.21: Create Table Tbl_product_photo

10) Tbl_product_size

```
CREATE TABLE `tbl_product_size` (
  `id` int(11) NOT NULL,
  `size_id` int(11) NOT NULL,
  `p_id` int(11) NOT NULL,
   PRIMARY KEY (id),
   FOREIGN KEY (size_id) REFERENCES tbl_size (size_id),
FOREIGN KEY (p_id) REFERENCES tbl_product (p_id));
```

Figure 5.22: Create Table Tbl_product_size

11) Tbl_rating

```
CREATE TABLE `tbl_rating` (
  `rt_id` int(11) NOT NULL,
  `p_id` int(11) NOT NULL,
  `cust_id` int(11) NOT NULL,
  `comment` text DEFAULT NULL,
  `rating` float NOT NULL,
   PRIMARY KEY (rt_id),
   FOREIGN KEY (p_id) REFERENCES tbl_product (p_id),
FOREIGN KEY (cust_id) REFERENCES tbl_customer
(cust_id));
```


Figure 5.23: Create Table Tbl_rating

12) Tbl_seller

```
CREATE TABLE `tbl_seller` (
  `seller_id` int(11) NOT NULL,
  `seller_name` varchar(100) NOT NULL,
  `shopName` varchar(255) DEFAULT NULL,
  `seller_email` varchar(100) NOT NULL,
  `seller_phone` varchar(50) NOT NULL,
  `seller_country` int(11) NOT NULL,
  `seller_address` text NOT NULL,
  `seller_city` varchar(100) NOT NULL,
  `seller_state` varchar(100) NOT NULL,
  `seller_zip` varchar(30) NOT NULL,
  `seller_password` varchar(100) NOT NULL,
  `seller_datetime` varchar(100) NOT NULL,
  `seller_timestamp` varchar(100) NOT NULL,
  `seller_status` int(1) NOT NULL,
  `seller_photo` varchar(255) DEFAULT NULL,
  PRIMARY KEY (seller_id));
```

Figure 5.24: Create Table Tbl_seller

13) Tbl_size

```
CREATE TABLE `tbl_size` (
  `size_id` int(11) NOT NULL,
  `size_name` varchar(255) NOT NULL,
PRIMARY KEY (size_id));
```

Figure 5.25: Create Table Tbl_size

14) Tbl_top_category

```
CREATE TABLE `tbl_top_category` (
  `tcat_id` int(11) NOT NULL,
  `tcat_name` varchar(255) NOT NULL,
  `show_on_menu` int(1) NOT NULL,
PRIMARY KEY (tcat_id));
```

Figure 5.26: Create Table Tbl_top_category

15) Tbl_user

```
CREATE TABLE `tbl_user` (
  `id` int(10) NOT NULL,
  `full_name` varchar(100) NOT NULL,
  `email` varchar(255) NOT NULL,
  `phone` varchar(100) NOT NULL,
  `password` varchar(255) NOT NULL,
  `photo` varchar(255) NOT NULL,
  `role` varchar(30) NOT NULL,
  `status` varchar(10) NOT NULL,
PRIMARY KEY (id));
```


Figure 5.27: Create Table Tbl_user

5.3.1.3 Create Trigger

The trigger is a condition that automatically checks the data manipulation event had happened on the selected table. The trigger is performing before or after operations occurred. This system contains 1 trigger that are created to control the data. Table 5.1 shows trigger implemented in One-Click Pick (Online Shopping System).

Table 5.1: Trigger in One-Click Pick (Online Shopping System)

Type of triggers	Implemented in table	Use of trigger
After insert	Customer table	Ask customer to insert email address if the email address field is empty.


```
CREATE TRIGGER after_cust_insert
AFTER INSERT
ON tbl_customer FOR EACH ROW
BEGIN
   IF NEW.cust_email IS NULL THEN
      INSERT INTO reminders(cust_id, message)
      VALUES(new.id,CONCAT('Hi ', NEW.cust_name, ', please update
your email.'));
   END IF;
```

Figure 5.28: Create Trigger Customer Email

5.3.1.4 Create Stored Procedure

The stored procedures will accept or not accept input parameters and can be used by the users using different parameters. Table 5.2 shows list of the procedures that are created in this system.

Table 5.2: List of Procedures in One-Click Pick (Online Shopping System)

Type of procedure	Implemented in table	Use of procedure
Stored procedure for select statement.	Admin	For viewing details data of each customer and seller.

1) Procedure GetSeller

```
BEGIN
    SELECT COUNT(seller_id)
    FROM tbl_seller;
END
```

Figure 5.29: Create Stored Procedure Display Total Seller

2) Procedure GetCustomer

```
BEGIN
        SELECT * FROM tbl_customer;
END
```

Figure 5.30: Create Stored Procedure Display Total Customer

3) Procedure GetCustomerDetail

```
BEGIN
SELECT *
        FROM tbl_customer t1
        JOIN tbl_country t2
        ON t1.cust_country = t2.country_id;
  END
```

Figure 5.31: Create Stored Procedure Display Details of Customer

4) Procedure GetSellerDetail

```
BEGIN
SELECT *
        FROM tbl_seller t1
        JOIN tbl_country t2
        ON t1.seller_country = t2.country_id;
END
```

Figure 5.32: Create Stored Procedure Display Details of Seller

5.4 Conclusion

In conclusion, the implementation phase is the final phase before goes to the testing phase. One-Click Pick (Online Shopping System) is developed using the XAMPP Server and MySQL using phpMyAdmin to control the database. In this project, it contains 15 tables, 1 trigger and 4 stored procedures that data storage implemented database.

Chapter 6 will explain about the testing phase of this project.

CHAPTER 6: TESTING

6.1 Introduction

Software testing may be prepared of assess the usefulness of a program application with an expectation to find whether the created software met the required necessities or not and to recognize the error or bugs to guarantee that the software is defect-free in arrange to create the quality software.

This chapter explained about the testing phase of One-Click Pick (Online Shopping System). This phase is for make sure the system developed have all the requirements. The testing that has been done on One-Click Pick (Online Shopping System) to check and confirm all the functionality meets the objective of proposed system.

In this venture, the testing methodologies to be embraced are black-box and white-box testing. Black-box testing is performed utilizing computer program interfacing to guarantee that they work as anticipated whereas white-box testing looks interior the program (coding) and employments that information as portion of the testing prepare. White-box testing requires inside information of the framework and programming abilities.

6.2 Test Plan

Test plan reflects the complete extensive testing schedule and approach that the developer utilizes within the testing stage. A test plan consists of test organization, test environment and test schedule. Test plan also help the testing steps and check the effectiveness of One-Click Pick (Online Shopping System).

Test organization is to explain the involvement of person within the testing procedure. Test environment is to explain the environment of testing to be carried out and to characterize hardware, configuration, arrangements and preparing the testing. Test schedule is to characterize how numerous cycles and duration of the test to be conducted.

6.2.1 Test Organization

Test organization is a group of people who is responsible to handle the test procedure of this project. Test procedures will be great if people in different background involve because few points of view may be delivered due to their own knowledge in information technology.

Software Developer is the person that responsible in developing One-Click Pick (Online Shopping System). Then the client is targeted person who will be the end user of this system such as staff and visitor. Table 6.1 shows the list of testers that involve for testing this system and their responsibilities.

Table 6.1: List of Tester and Their Responsibilities

Tester ID	Roles	Responsibilities
Tester_1	Software Developer	1) Developing, executing, reviewing the integration and component of current system. 2) Elaborate ideas of improvement the systems. 3) Writing the system code 4) Maintaining the system 5) Preparing the user's manual
Tester_2	Client	1) Act as the end user to get some feedback, which is seller and customer. 2) Testing the module of system developed. 3) Give feedback or opinion for improvements of the system.

6.2.2 Test Environment

A testing environment may be a setup of software and hardware for the testing to execute the One-Click Pick (Online Shopping System). The test environment setup must imitate the generation environment in arrange to reveal any configuration related issues. It will figure out whether this system can be adaptable to run on different platform of hardware and software.

6.2.2.1 Environment Setup

The environment setup depicts the environment of testing which can be carried out reliably all through the testing, the testing modules include the software and hardware utilized. Table 6.2 shows the environment setup specification in testing phase.

Table 6.2: Environment Setup Specification

Environment Configuration	Specification
Operating System	Windows 10
Processor	Intel(R)Core(TM) i3-10110U CPU @ 2.10GHz 2.59GHz
Random Access Memory (RAM)	12GB
Database	MySQL
Server	Apache
Programming Language	1) Hypertext Preprocessor (PHP) 2) Hypertext Markup Language (HTML)

6.2.2.2 Software Application

Software application consist all the application or module inside One-Click Pick (Online Shopping System). Table 6.3 shows all the application in One-Click Pick (Online Shopping System).

Table 6.3: Application Environment

System Application	1) System Login / Logout 2) Add, Update, Delete, View and Search in each module for admin, seller, and customer. 3) Generate Excel report for seller.

6.2.2.3 System Software

System software consists all the tools that have been used in One-Click Pick (Online Shopping System). Table 6.4 shows all the software that have been used in developing this system.

Table 6.4: System Software

System Software	1) Windows 10 Home Single Language 2) XAMPP Server v3.3.0 (Apache, MySQL) 3) Google Chrome 4) Sublime Text

6.2.2.4 System Hardware

System hardware consist all the hardware that have been used in this system. Table 6.5 shows the hardware used for developed One-Click Pick (Online Shopping System).

Table 6.5: System Hardware Tools

System Hardware	1) Laptop 2) Mouse 3) Keyboard

6.2.3 Test Schedule

Test plan is the technique of testing all the data collected within the length of time which has been set. The test schedule is to assign about the schedule for testing of One-Click Pick (Online Shopping System) such as when and by whom the test will be conduct. The schedule is function as guideline for developer to perform the testing on time as in project timeline. Table 6.6 shows the test schedule for admin, seller, and customer.

Table 6.6: Test Schedule

Module / Component	Activity	Duration (Day(s))
Registration	Error handling test and integration test	1
Login	Error handling test and integration test	1
Change Password	Error handling test and integration test	1
Update Profile	Error handling test and integration test	2
Seller	Error handling test and integration test	2
Customer	Error handling test and integration test	2

6.3 Testing Strategies

Test strategy is characterized as a set of directing guideline that illuminates the test plan and controls how testing should be done.

White Box Testing also called as glass box or structural testing. White Box Testing is checking at the structure of the code inside the system and use that knowledge as process of testing. Should know about the internal structure of system to performed this testing smoothly and can make sure the system run according to specification.

Black Box Testing is more to interact with the system. It means that to perform this testing techniques it not requires any knowledge about the internal part of the system but need to know how it should perform. Table 6.8 shows explanation about White Box Testing Vs Black Box Testing.

Table 6.7: White Box Testing Vs Black Box Testing

Approach	Explanation
White Box Testing	• The test that has been performed to check and evaluate the internal structure of the system through the developer using internal structure (code). • Known as “Structural Testing”. • Developer and tester (have programming knowledge) will involve in this testing.
Black Box Testing	• The test that has been performed to test the system through the functional and non-functional (based on its behavior) without knowing about internal structure of the system. • Known as “Functional Testing”. • Tester involves in this testing.

6.3.1 Classes of Test

Two types of tests were selected to use in this phase which are error handling test and integration test.

1) Error Handling Test

   Error handling testing is a type of testing that is performed to check whether the system is capable of or able to handle the errors. This testing performed with the help of both developers and the testers and not only focuses on error but also focuses on the exception handling. This test will validate only connect and accurate data from the client. Error message will pop up on the screen to inform user.

2) Integration Test

   Integration testing is to test the interfaces between the modules. This test is to ensure this system captures data into the database correctly based on input from user. If the process is successful, means that the system is well integrated with the database.

6.4 Test Design

There are two parts of test design which are test description and test data. Test design will clarify about the test that has been performed for each module. Test description is the test case identification, test cases and result for each testing module. As for the test data, it incorporates the genuine life information that will be chosen. Test design will explain about the test has been performed for each module.

6.4.1 Test Description

Test description is a test case that documented set of the data input and operating condition needs to run a test item. Table 6.8 to Table 6.16 shows the test cases and expected result for each module.

Table 6.8: Registration Module

Test Case ID	Description	Testing Type	Expected Result
OneClickPick_01-1	Some field is blank	Error Handling Test	"Full Name cannot be empty" will pop up
OneClickPick_01-2	Valid input for each field	Unit Testing/Integration	"Your registration is completed. Your account will be activated soon." Will pop up

Table 6.9: Login Module

Test Case ID	Description	Testing Type	Expected Result
OneClickPick_02-1	Invalid username and password	Error Handling Test	"Email Address does not match" Or "Password do not match." Will pop up
OneClickPick_02-2	Valid input for each field	Unit Testing/Integration	User can login to the system successfully.

Table 6.10: Change Password Module

Test Case ID	Description	Testing Type	Expected Result
OneClickPick_03-1	New password field is blank	Error Handling Test	"Password cannot be empty" will pop up
OneClickPick_03-2	Valid current new password and confirm new password with match requested format	Unit Testing/Integration	"Password is updated successfully." will pop up

Table 6.11: Update Profile Module

Test Case ID	Description	Testing Type	Expected Result
OneClickPick_04-1	Some mandatory fields are blank	Error Handling Test	"Please include an '@' in the email address. 'linda1234' is missing an '@'." will pop up. AND "Name cannot be empty" will be pop up
OneClickPick_04-2	Valid input for each field	Unit Testing/Integration	"Admin Information is updated successfully." will be pop up.

Table 6.12: Seller Module (Add Product)

Test Case ID	Description	Testing Type	Expected Result
OneClickPick_06-1	Some mandatory field is blank	Error Handling Test	"Product name cannot be empty Price cannot be empty Quantity cannot be empty" will pop up.
OneClickPick_06-2	Valid input for each field	Unit Testing/Integration	"Product is added successfully." will pop up

Table 6.13: Seller Module (Send Message)

Test Case ID	Description	Testing Type	Expected Result
OneClickPick_07-1	Send message with blank field	Error Handling Test	"Subject cannot be empty Message cannot be empty" will pop up.
OneClickPick_07-2	Valid input for message field	Unit Testing/Integration	"Your message to customer is sent successfully" will pop up

Table 6.14: Seller Module (Edit Product)

Test Case ID	Description	Testing Type	Expected Result
OneClickPick_08-1	Some mandatory fields are blank	Error Handling Test	"Product name cannot be empty Current Price cannot be empty" will pop up.
OneClickPick_08-2	Valid input for each field	Unit Testing/Integration	"Product is updated successfully." will pop up

Table 6.15: Customer Module (Insert Banking Detail)

Test Case ID	Description	Testing Type	Expected Result
OneClickPick_09-1	Some mandatory fields are blank	Error Handling Test	"Card CVV cannot be empty" will pop up.
OneClickPick_09-2	Valid input for each field	Unit Testing/Integration	"Congratulation! Payment is successful." will be pop up.

Table 6.16: Customer Module (Add Product in Cart)

Test Case ID	Description	Testing Type	Expected Result
OneClickPick_10-1	Click more than once on Add to Cart button at the same product	Error Handling Test	"This product is already added to the shopping cart." will pop up.
OneClickPick_10-2	Valid input	Unit Testing/Integration	Product will be added to the shopping cart.

6.4.2 Test Data

Test data are used to get expected output based on test design of this system. This test data is performed by validating, testing, and verifying the software behavior using the real data from user. Table 6.17 to Table 6.25 shows the test data.

Table 6.17: Test Data of Registration

Test Data ID	TD_01-1	TD_01-2
Test Case ID	OneClickPick_01-1	OneClickPick_01-2
User registration form	User does not fill up the fields correctly. Full Name: Aklinda binti Mansor Shop Name: BerryStoberry Photo: user-1.jpg (upload photo) Email Address: linda1234 Phone Number: 0113 Address:	User does fill up the fields correctly. Full Name: Aklinda binti Mansor Shop Name: BerryStoberry Photo: user-1.jpg (upload photo) Email Address: linda1234@gmail.com Phone Number: 01131653432

	Country: Malaysia City: Manjung State: Perak Zip Code: 32200 Password: linda0923 Retype Password: linda0923	Address: No 19A Batu 19 Jalan Kemudi Raya 32200 Lumut Perak Country: Malaysia City: Manjung State: Perak Zip Code: 32200 Password: linda0923 Retype Password: linda0923
Test Data Result	New user registration failed due to incomplete form.	New user's information is successfully submitted.

Table 6.18: Test Data of Login

Test Data ID	TD_02-1	TD_02-2
Test Case ID	OneClickPick_02-1	OneClickPick_02-2
Username	linda1234	linda1234@gmail.com
Password	linda0923	linda0923
Test Data Result	Login failed because of incorrect username or password.	Successfully login to system.

Table 6.19: Test Data of Change Password

Test Data ID	TD_03-1	TD_03-2
Test Case ID	OneClickPick_03-1	OneClickPick_03-2
Current Password	linda0923	linda0923
New Password	linda2309	linda2309
Confirm New Password		linda2309
Test Data Result	Change password is failed due to blank confirm new password field.	Change Password Success.

Table 6.20: Test Data of Update Profile

Test Data ID	TD_04-1	TD_04-2
Test Case ID	OneClickPick_04-1	OneClickPick_04-2
Update Profile Form	User does not fill up the fields correctly. Name: Email Address: linda1234 Phone: 01131653432	User does fill up the fields correctly. Name: Linda Mansor Email Address: linda1234@gmail.com Phone: 01131653432

Test Data Result	New user information failed recorded due to incomplete form.	New user's information is successfully updated.

Table 6.21: Test Data of Seller (Add Product)

Test Data ID	TD_06-1	TD_06-2
Test Case ID	OneClickPick_06-1	OneClickPick_06-2
New Product Form	Seller does not fill up the fields correctly. Top Level Category Name *: Women Mid Level Category Name *: Beauty Products End Level Category Name *: Skincare Product Name *: Price *: Quantity *: Select Size: Select Color:	Seller does fill up the fields correctly. Top Level Category Name *: Women Mid Level Category Name *: Beauty Products End Level Category Name *: Skincare Product Name *: Nafura Price *: 39.90 Quantity *: 300 Select Size:

	Featured Photo *: product.jpg (photo file) Other Photos: Description: Short Description: Features: Conditions: Return Policy: Is featured? Yes Is Active? Yes	Select Color: Featured Photo *: product.jpg (photo file) Other Photos: Description: Short Description: Features: Conditions: Return Policy: Is featured? Yes Is Active? Yes
Test Data Result	New product information failed to record due to incomplete mandatory (*) form.	New product information is successfully submitted.

Table 6.22: Test Data of Seller (Send Message)

Test Data ID	TD_07-1	TD_07-2
Test Case ID	OneClickPick_07-1	OneClickPick_07-2
Send Message	Seller does not write anything in the message form. Subject: Message:	Seller fill the message field with some words. Subject: Hello Linda Message: Your item will be post soon!
Test Data Result	Failed to send message due to blank message field.	Message successfully submitted.

Table 6.23: Test Data of Seller (Edit Product)

Test Data ID	TD_08-1	TD_08-2
Test Case ID	OneClickPick_08-1	OneClickPick_08-2
Update Product Form	Seller does not fill up the fields correctly. Top Level Category Name *: Women Mid Level Category Name *: Beauty Products End Level Category Name *: Skincare	Seller does not fill up the fields correctly. Top Level Category Name *: Women Mid Level Category Name *: Beauty Products End Level Category Name *: Skincare

	Product Name *: Old Price:43.30 Current Price *: Quantity *: 300 Select Size: Select Color: Featured Photo *: product.jpg (photo file) Other Photos: Description: Short Description: Features: Conditions: Return Policy: Is featured? No Is Active? No	Product Name *: LovelieeBeauty Old Price: 43.30 Current Price *: 39.90 Quantity *: 300 Select Size: Select Color: Featured Photo *: product.jpg (photo file) Other Photos: Description: Short Description: Features: Conditions: Return Policy: Is featured? No Is Active? No

Test Data Result	New product information failed to update due to blank mandatory (*) form.	New product information is successfully updated.

Table 6.24: Test Data of Customer (Insert Banking Detail)

Test Data ID	TD_09-1	TD_09-2
Test Case ID	OneClickPick_09-1	OneClickPick_09-2
Online Banking Form	Customer does not fill up the fields correctly. Bank Name: Bank Islam Card Number: 4332 2334 2343 2333 Card CVV: Card Month: 04 Card Year: 2023	Customer not fill up the fields correctly. Bank Name: Bank Islam Card Number: 4332 2334 2343 2333 Card CVV: 321 Card Month: 04 Card Year: 2023
Test Data Result	Banking information failed to record due to incomplete form.	Banking information is successfully submitted.

Table 6.25: Test Data of Customer (Add Product in Cart)

Test Data ID	TD_10-1	TD_10-2
Test Case ID	OneClickPick_10-1	OneClickPick_10-2
Add to Cart button	Customer click on the button at the same product more than once.	Customer click on the button at the product once.
Test Data Result	Product failed to add in the cart.	Product added successfully.

6.5 Test Result and Analysis

For the test result, it clarifies the result of testing that followed using test data from user. This part will explain each module test case results which consists of the success or failure. All test case will be tested and the result will be shown on the table below. Table 6.26 to Table 6.35 shows the test result and analysis.

Table 6.26: Registration Module Test Result

Module/Component			**Result**		
TEST CASE ID	TEST DATA ID	Testing Date	Description	Success	Fail
OneClickPick_01-1	TD_01-1	17/8/2021	Some field is blank	√	
OneClickPick_01-2	TD_01-2	17/8/2021	All fields were filled.	√	

Table 6.27: Login Module Test Result

Module/Component			**Result**		
TEST CASE ID	TEST DATA ID	Testing Date	Description	Success	Fail
OneClickPick_02-1	TD_02-1	18/8/2021	User key in incorrect username and password.	√	
OneClickPick_02-2	TD_02-2	18/8/2021	User key in correct username and password.	√	

Table 6.28: Change Password Module Test Result

Module/Component			**Result**		
TEST CASE ID	TEST DATA ID	Testing Date	Description	Success	Fail
OneClickPick_03-1	TD_03-1	19/8/2021	User does not fill confirm password field.	√	
OneClickPick_03-2	TD_03-2	19/8/2021	All fields were filled.	√	

Table 6.29: Update Profile Module Test Result

Module/Component			**Result**		
TEST CASE ID	TEST DATA ID	Testing Date	Description	Success	Fail
OneClickPick_04-1	TD_04-1	20/8/2021	Some field is blank	√	
OneClickPick_04-2	TD_04-2	20/8/2021	All fields were filled.	√	

Table 6.30: Seller (Add Product) Module Test Result

Module/Component			**Result**		
TEST CASE ID	TEST DATA ID	Testing Date	Description	Success	Fail
OneClickPick_06-1	TD_06-1	22/8/2021	Some field is blank	√	
OneClickPick_06-2	TD_06-2	22/8/2021	All fields were filled	√	

Table 6.31: Seller (Send Message) Module Test Result

Module/Component			**Result**		
TEST CASE ID	TEST DATA ID	Testing Date	Description	Success	Fail
OneClickPick_07-1	TD_07-1	22/8/2021	Message field is blank	√	
OneClickPick_07-2	TD_07-2	22/8/2021	Message fields were filled	√	

Table 6.32: Seller (Edit Product) Module Test Result

Module/Component			**Result**		
TEST CASE ID	TEST DATA ID	Testing Date	Description	Success	Fail
OneClickPick_08-1	TD_08-1	23/8/2021	Some field is blank	√	
OneClickPick_08-2	TD_08-2	23/8/2021	All fields were filled	√	

Table 6.33: Customer (Insert Banking Detail) Module Test Result

Module/Component			**Result**		
TEST CASE ID	TEST DATA ID	Testing Date	Description	Success	Fail
OneClickPick_09-1	TD_09-1	24/8/2021	Some field is blank	√	
OneClickPick_09-2	TD_09-2	24/8/2021	All fields were filled	√	

Table 6.34: Customer (Add to Cart) Module Test Result

Module/Component			**Result**		
TEST CASE ID	TEST DATA ID	Testing Date	Description	Success	Fail
OneClickPick_10-1	TD_10-1	25/8/2021	Click button at the same product more than once.	√	
OneClickPick_10-2	TD_10-2	25/8/2021	Click button once.	√	

6.6 Conclusion

To conclude, testing stage clarified around the procedure that has been utilize to approve and confirm One-Click Pick (Online Shopping System) to create beyond any doubt the all work meets the necessity and module that have been created. Software testing is the hard part in every software development. All testing action must be arranged well and conducted through due period and the fetched of settling the bugs and errors. Test arrange comprise the test cases that are utilized to look at changes anticipate of the system.

Chapter 7 will explain about the conclusion of the project. A conclusion will be explained by pointing out the strengths and the weakness.

CHAPTER 7: CONCLUSION

7.1 Introduction

To conclude this project, this chapter will explain the weakness and strengths of this system. In this chapter also explain about the proposition for improvement on One-Click Pick (Online Shopping System) which are any suggestion that can make this system better for the user. All the strength and weakness of this system is based on a module that developed in this system. Lastly, this chapter also explains about project contribution such as to university, faculty, company, or individual that relate to this project.

7.2 Observation on Weaknesses and Strengths

Every system that has been developed always have their strength and weakness. Table a list of the strength and weakness of One-Click Pick (Online Shopping System).

Table 7.1: Strength and Weakness

Strength	1) Easy for customer to buy their necessities. 2) Give opportunities to everyone who are interested in doing business. 3) Helps customer to shop wisely by easily comparing prices.
Weakness	1) Does not have procedure to download/save report. 2) Does not implemented password encryption for security purpose.

7.3 Proposition for Improvement

After analyzing the strength and weaknesses that state above, there are several ideas for improvements that can help in future use. One-Click Pick (Online Shopping System) will be better with implement a download report procedure. This procedure is being implemented for seller, so that it will be easier for seller to save or view the report. Furthermore, this is also a precautionary measure when there is a possibility of problems with the seller's account and resulting in the loss of sales records. So, by using this procedure seller can download the report and save it as backup data.

Generally, this system is not secure on user account because this system does not implement password encryption. For preventing user account from any hacker or something bad happen, this system should implement password encryption to secure the user account from unauthorized person login into the system.

7.4 Project Contribution

Project contribution for this project is divide into the university, company, and individual. For the first is the university, this project is belonging to the university in University Technical Malaysia Melaka. This also can be the source of ideas, finding and information to other university students in developing management system, especially for online shopping management systems.

Furthermore, for company contribution, especially for the business company which can manage its system and process to further expand the business by using online method. The data of sales or any information will be stored in the database and can ease management to track and manage the income earned for each month.

Finally, for individual contribution is during project documentation to get any knowledge or information and ideas to develop and implement another system or make any improvement of their system.

7.5 Conclusion

The conclusion of this project that can be concluded is this system has been developed to facilitate customer and seller who want to do buying and selling activities by using this One-Click Pick (Online Shopping System). The system has been developed also meets objective and can solve the problem statement that has been studied in the first chapter in this project and still need improvement for the better in future use. The proposition of improvement that has been analyzed need to be implemented in this project to make sure this system more efficient, reliable, and more secure.

REFERENCES

Alba, J. L. (1997). Interactive home shopping: Consumer, retailer, and manufacturer incentives to participate in electronic marketplaces. Journal of Marketing, 61, 38–53.

Alba, J. L.–5. (n.d.).

Darden, W. R. (1990). An action strategy approach to examining shopping behavior,. Journal of Business Research, 21,289-308.

Forsythe, S. &. (2003). Consumer patronage and risk perceptions in Internet shopping. Journal of Business Research, 56,867-875.

Hoffman, D. L. (1996). Commercial scenarios for the web: Opportunities and challenges. Journal of Computer Mediated Communications, 1(3).

Hoffman, D. L. (1996). Marketing in the hypermedia computer-mediated environments: Conceptual foundations. Journal of Marketing, 60(3),50-68.

Junghyun Kim, R. L. (2006, 6 23). Wiley Online Library. Retrieved from Interactive E-Commerce: Promoting Consumer Efficiency or Impulsivity?: https://onlinelibrary.wiley.com/doi/10.1111/j.1083-6101.2004.tb00234.x

Uwe Matzat, C. S. (2004, 11 1). Oxford Academic. Retrieved from Journal of Computer-Mediated Communication: https://doi.org/10.1111/j.1083-6101.2012.01594.x

APPENDICES

USER MANUAL FOR ADMIN

1) To Login to the system
    a) Admin needs to fill in email address in the username field and password.
    b) Then, click on button login.

2) To Edit Profile
    a) Admin needs to click on the profile photo at the top right of the page.
    b) After that, click on the Edit Profile button and it will show new page which is Edit Profile page.
    c) In this page, there are three tabs which is the first one is Update Information tab, followed by the second tab Update Photo tab and the last one is Update Password tab.
    d) If admin wants to update information, admin can insert new information either at the Name field, Email Address field or Phone No field.
    e) After updating the information, just click on Update Information button.
    f) Then, if admin wants to update profile photo just click on the Update Photo tab.
    g) By clicking the Choose File button, admin can choose any photo as Profile Photo.
    h) After selecting photo, click at the Update Photo button and the profile photo will be successfully updated.
    i) Next, to update new password just click at the Update Password tab.
    j) Fill in the New Password and Retype Password field.
    k) Lastly, click at the Update Password button and the new password is successfully updated.

3) To View Total Seller and Customer
    a) At the sidebar, there are three section which is Dashboard, Customer and Seller.
    b) To view the total Seller and Customer just click at the Dashboard.
    c) It will show the total sellers and customers in this system and it also display the total of new seller and new customer who are requesting for account activation.

4) To View Customer/Seller List and Activate Customer Account
    a) Admin need to click on Customer/Seller section at the sidebar.
    b) Then, it will display all customer/seller list including customers/sellers who have just registered and waiting for account activation. For customers/sellers whose account has been activated, the customer/seller information will be displayed in green color while the other will be displayed in red color which mean the customer/seller account is not active yet.
    c) To activate customers/sellers account, admin needs to click on the Change Status button and customer/seller account have been successfully activate.

5) To Logout
    a) Admin needs to click on the profile photo at the top right of the page.
    b) After that, click on the Log out button and admin will be logout successfully.

USER MANUAL FOR SELLER

1) To Register
    a) To register, seller need to click on Sign Up at the login page.
    b) Then, it will display registration form that seller needs to fill in.
    c) After completing the registration form, seller needs to click at the Register button.

d) It will pop up a message which inform seller that the account will be activate soon.

2) To Login to the system

a) After seller's account is activated, seller can login to the system.
b) First thing is seller needs to enter Email Address and Password.
c) Then, click on Login button and seller is successfully login to the system.

3) To Edit Profile

a) Seller needs to click on the profile photo at the top right of the page.
b) After that, click on the Edit Profile button and it will show new page which is Edit Profile page.
c) In this page, there are three tabs which is the first one is Update Information tab, followed by the second tab Update Photo tab and the last one is Update Password tab.
d) If seller wants to update information, seller can insert new information either at the Name field, Email Address field or Phone No field.
e) After updating the information, just click on Update Information button.
f) Then, if seller wants to update profile photo just click on the Update Photo tab.
g) By clicking the Choose File button, seller can choose any photo as Profile Photo.
h) After selecting photo, click at the Update Photo button and the profile photo will be successfully updated.
i) Next, to update new password just click at the Update Password tab.
j) Fill in the New Password and Retype Password field.
k) Lastly, click at the Update Password button and the new password is successfully updated.

4) To View Total Product, Completed Payment/Shipping and Pending Payment/Shipping
    a) At the sidebar, there are three section which is Dashboard, Customer and Seller.
    b) To view the Total Product, Completed Payment/Shipping and Pending Payment/Shipping just click at the Dashboard.
    c) It will show the Total Product, Completed Payment/Shipping and Pending Payment/Shipping.

5) To View Orders/Send Message
    a) Click at the Orders Management at the sidebar and it will display the order list.
    b) From the list, seller can view Payment status and Shipping status for each customer. Once customer has completed the payment, seller can click on the Make Completed button and it is same with Shipping status. Once seller has completed the shipping, then just click on the Make Completed button.
    c) At this page, seller also can send message to customer by clicking at the Send Message button.
    d) Then, fill in the Subject and Message form and click on the Send Message button. The message will be sent to the customer.

6) To Add/View/Edit Products
    a) Click at the View/Add Products at the sidebar and it will display the product list.
    b) To Edit product, just click at the Edit button, and click on Add Product button if seller wants to add new products.
    c) As Edit button has been clicked, it will display products information form and seller can update the information.
    d) As Add Product button has been clicked, it will display blank product information form and seller needs to fill in all the requested information.

7) To View Customer Feedback
    a) Click on Customer Feedback at the sidebar.
    b) Feedback list from customer will be displayed.

8) To View/Download Report
    a) Click on Report Sale at the sidebar.
    b) Report Sale Per Year will be displayed.
    c) Seller can click on Export button to download the report sale.

USER MANUAL FOR CUSTOMER

1) To Register
    a) To register, click on Register at the top of the page.
    b) It will display Customer Registration form and customer need to fill in all the requested information.
    c) After completing the form, click on Register button and wait until the account is activate.

2) To Login to the system
    a) After customer's account is activate, customer can login to the system.
    b) First thing is customer needs to enter Email Address and Password.
    c) Then, click on Login button and customer is successfully login to the system.

3) To View/Buy Product
    a) Click on Home at the navbar.
    b) It will display all the products sold in this system.
    c) If customer wants to buy the product, just click on Add To Cart button.
    d) After that, to make a payment just click on View Cart at the navbar.
    e) It will display detail product and price, so just click on Proceed To Checkout button to continue the payment process.
    f) Then, customer can choose type of payment whether using Online Banking or Cash on Delivery (COD).

g) Before continue with the payment, customer is required to fill in the shipping and billing form if it has not yet been filled out. Then, customer can continue with the payment process.
h) If customer choose to use Online Banking, customer need to fill in the Bank Information form and the payment will be completed after that.
i) If customer choose to use Cash on Delivery (COD), just click on Confirm button.

4) To Update Profile
   a) Click on Update Profile button in Dashboard section.
   b) Then, fill in the information that want to be update.
   c) Lastly, click on Update button and it will be updated successfully.

5) To Update Billing and Shipping Info
   a) Click on Update Billing and Shipping Info button in Dashboard section.
   b) Then, fill in the information that wants to be updated.
   c) Lastly, click on Update button and it will be updated successfully.

6) To Update Password
   a) Click on Update Password button in Dashboard section.
   b) Then, enter new password and retype the new password in the field provided.
   c) Lastly, click on Update button to confirm the new password.

7) To View/Rate Orders
   a) Click on Orders button in Dashboard section.
   b) It will display order history and Rate button.
   c) If customer wants to rate the product, just click on Rate button.
   d) Fill in the feedback and rating form then click Send Message button.

8) To View Message from Seller
   a) Click on Message button in Dashboard section.
   b) Message from seller will be displayed in this page.