

**Perpustakaan SKTM**

**WEB PAGE DESIGN FOR ELECTRONIC COMMERCE  
(Online Shopping Administration Subsystem)**

**SESSION 2002/2003**

**LEE FONG WAI  
WET000068**

**SUPERVISOR: MR. TEH YING WAH**

**MODERATOR: MR. PHANG KEAT KEONG**

## ABSTRACT

This report covers all the specification development of the Online Shopping Administration Subsystem. The first part covers the introduction of the project including the project overview, problem definitions, project objectives and scopes, project schedule and expected outcomes. Also includes a brief explanation about the interaction of this subsystem with the Online Shopping Submission and Processing Subsystem, which is developed by my teammate. The second part is the literature review. This part covers the introduction of electronic commerce and online shopping, features of the online shopping softwares available in the market, data mining technology, programming languages, web development tools, database technology and also security issues. The third part is the project development methodology. It covers the description of reasons, justification and strength in the chosen methodology. The fourth and fifth parts cover the requirement analysis and system design. Through the project objectives and analysis conducted on the available technologies, the requirements of the system (functional and non-functional requirements, hardware and software requirements) were identified and the overall design of the system was developed. The sixth part covers the system implementation that involved the transformation of modules and algorithm into implementable commands by using the specified programming languages. The seventh part covers the system testing to prove that whether the system can perform based on the system requirements and specifications. The final part covers the problems encountered during system development process, the solutions taken, system evaluation by end users, the strength and limitation of the system, future enhancement of the system, knowledge and experience gained and conclusion towards the project.

## **ACKNOWLEDGEMENT**

I would like to thank a few people for their guidance and help throughout the whole semester in designing and developing this Online Shopping Administration Subsystem.

First and foremost, I would like to express my greatest thanks and appreciation to Mr. Teh Ying Wah, my supervisor of this project for being so friendly, patient and kind throughout the whole preparation period of this project. He has given me a lot of encouragement and freedom in designing this subsystem. Also I would like to thank my moderator, Mr. Phang Keat Keong, for his invaluable advice and comment.

Last but not least, I would like to express my appreciation as well to my teammate, Miss Lee Wai Tien, for willing to share her ideas and opinions with me.

# TABLE OF CONTENTS

<b>ABSTRACT</b>	<b>ii</b>
<b>ACKNOWLEDGEMENT</b>	<b>iii</b>
<b>LIST OF TABLES</b>	<b>ix</b>
<b>LIST OF FIGURES</b>	<b>x</b>

## **CHAPTER 1: INTRODUCTION**

1.1 Project Overview	1
1.2 Problem Definition	2
1.3 Project Objectives	2
1.4 Project Scope	3
1.5 Project Schedule	5
1.6 Expected Outcome	6

## **CHAPTER 2: LITERATURE REVIEW**

2.1 Electronic Commerce	7
2.2 Online Shopping	7
2.2.1 What Is Online Shopping?	7
2.2.2 Growth Of Online Shopping	8
2.2.3 Advantages And Disadvantages Of Online Shopping	8
2.2.3.1 Advantages	8
2.2.3.2 Disadvantages	10
2.2.4 Existing Online Shopping System	11
2.2.4.1 GeniusCart Pro	11

2.2.4.2 Line9 Tek Pro V6.2	11
2.2.4.3 SupraCart 4.0	12
2.2.4.4 uShop Online Shopping System	12
2.2.4.5 uStorekeeper Online Shopping System	13
2.2.4.6 Conclusion	13
2.3 Data Mining Technology	13
2.3.1 What Is Data Mining?	13
2.3.2 Why Use Data Mining?	14
2.3.3 Classification	15
2.3.3.1 Classification By Decision Tree Induction	16
2.4 Programming Languages	18
2.4.1 General Purpose Programming Languages	18
2.4.1.1 Java	18
2.4.1.2 Visual Basic	18
2.4.2 Markup Language	20
2.4.2.1 HTML	20
2.4.3 Client-side Scripting Languages	20
2.4.3.1 JavaScript	20
2.4.3.2 VBScript	21
2.4.4 Server-side Scripting Languages	22
2.4.4.1 Active Server Page (ASP)	22
2.5 Web Development Tools	23
2.5.1 Microsoft FrontPage 2000	23
2.5.2 Microsoft Visual Interdev 6.0	24
2.5.3 Macromedia Dreamweaver MX	24

2.6 Web Server	26
2.6.1 Apache 2.0.39	26
2.6.2 Internet Information Server (IIS)	28
2.6.3 Jigsaw 2.2.1	29
2.6.4 Personal Web Server (PWS) 4.0	30
2.7 Database Technology	33
2.7.1 Database Access Standard Interface – ODBC and OLE DB	33
2.7.1.1 ODBC (Open Database Connectivity)	33
2.7.1.2 OLE DB	33
2.7.2 Database Access Libraries – DAO, RDO, ADO	35
2.7.2.1 Data Access Objects (DAO)	35
2.7.2.2 Remote Data Objects (RDO)	35
2.7.2.3 ActiveX Data Objects (ADO)	36
2.7.3 Database Management System (DBMS)	38
2.7.3.1 Microsoft Access 2000	38
2.7.3.2 Microsoft SQL Server 7.0	39
2.8 Security Considerations	40
2.8.1 Network Security	40
2.8.1.1 What Is Network Security?	40
2.8.1.2 Confidentiality, Integrity and Availability	41
2.8.1.3 Identification and Authentication (I&A)	41
2.8.1.4 Accountability	42
2.8.1.5 Access Controls	42
2.8.1.6 Audit	43
2.8.1.7 Non-repudiation	43

2.8.1.8 Data Encryption	44
2.9 Open Source Code	45
2.9.1 WEKA 3 – Data Mining With Open Source Machine Learning Software In Java	45

## **CHAPTER 3: METHODOLOGY**

3.1 Overview	47
3.2 Waterfall Model With Prototyping	47
3.3 Why Choose Prototyping?	50
3.4 Information Gathering Technique	51
3.4.1 Research	51
3.4.2 Internet Surfing	51
3.4.3 Discussion With Supervisor	51

## **CHAPTER 4: SYSTEM ANALYSIS**

4.1 Requirement Analysis And Specification	52
4.1.1 Functional Requirements	52
4.1.2 Non-functional Requirements	58
4.2 Hardware And Software Requirements	60
4.2.1 Hardware Requirements	60
4.2.2 Software Requirements	61
4.2.2.1 Programming Languages	61
4.2.2.2 Web Development Tools	61
4.2.2.3 Development Platform And Web Server	61
4.2.2.4 Database Access Standard Interface and Database Access	61

Libraries	
4.2.2.5 Database Management System (DBMS)	62
4.2.2.6 Others	62
<b>CHAPTER 5: SYSTEM DESIGN</b>	
5.1 Overview	63
5.2 System Architecture	63
5.2.1 Two-tier Architecture Vs Three-tier Architecture	63
5.2.2 Advantages Of Three-tier Architecture	66
5.3 Database Design	68
5.4 Process Design	74
5.4.1 Data Flow Diagram (DFD)	74
5.4.1.1 Context Diagram	75
5.4.1.2 Functional Decomposition Diagram	75
5.4.1.3 Event Diagram	76
5.4.1.4 System Diagram	77
5.4.1.5 Primitive Diagram	77
5.4.2 Framework	99
5.5 User Interface Design	100
<b>CHAPTER 6: SYSTEM IMPLEMENTATION</b>	103
6.1 Overview	103
6.2 Coding Style	104
6.2.1 User Interface Layer	105
6.2.2 Data Layer	105

6.2.3 Transaction Layer	106
6.3 Program Optimization	106
6.4 Programming Documentation	106
6.4.1 Internal Documentation	106
6.4.2 External Documentation	107
 <b>CHAPTER 7: SYSTEM TESTING</b>	 108
7.1 Overview	108
7.2 The Testing Process	108
7.2.1 Unit Testing	109
7.2.2 Module Testing	109
7.2.3 Subsystem Testing	110
7.2.4 System Testing	110
7.3 Test Data	111
7.4 Online Shopping Administration Subsystem Testing	112
 <b>CHAPTER 8: DISCUSSION</b>	 121
8.1 Problems Encountered And Solution	121
8.1.1 Defining The System Scope Due To The Wide Area Of Study	121
8.1.2 Program Coding	121
8.1.3 Retrieving Data And Inserting Data Into A Recordset	122
8.1.4 Understanding The Error Messages	122
8.1.5 Understanding And Applying Data Mining	122
8.1.6 Lack Of True Customer Information	123
8.2 Evaluation By End Users	123

8.3 System Strength	124
8.3.1 Easy Management And Maintenance	124
8.3.2 User Friendly Graphical User Interface	124
8.3.3 Informative Messages	124
8.3.4 Authorization And Authentication	124
8.3.5 User Input Validation	125
8.3.6 Improved Functionality	125
8.4 System Constraints	125
8.4.1 Lack Of Functional Modules	126
8.4.2 Database Not Encrypted	126
8.4.3 Lack Of Accuracy	126
8.5 Future Enhancements	126
8.5.1 Provide More Functionality	126
8.5.2 Improve Accuracy Of The New Generated Customer Information	127
8.6 Knowledge And Experience Gained	127
8.7 Conclusion	128

## **APPENDIX A – User Manual**

## **APPENDIX B – User Evaluation Form**

## **APPENDIX C – REFERENCE**

## LIST OF TABLES

Table 2.1 Comparison between the four popular web servers	32
Table 5.1 The 3-tier Architecture	65
Table 5.2 The attributes of the Online Shopping System	68
Table 5.3 Table for storing customer personal information. Table's name is Customer.	69
Table 5.4 Table for storing prepaid card information. Table's name is Prepaid.	70
Table 5.5 Table for storing product information. Table's name is Product.	71
Table 5.6 Table for storing balance information. Table's name is Balance.	71
Table 5.7 Table for storing order information. Table's name is Orderdetail.	71
Table 5.8 Table for storing supplier information. Table's name is Supplier	72
Table 5.9 Table for keeping user's login name and password where only authorized user can access the system. Table's name is User	72
Table 5.10 Table for storing customer's shopping cart details. Table's name is Cart.	73
Table 5.11 Table for storing personalized web page details. Table's name is Association_rules.	73
Table 5.12 Symbol of Data Flow Diagram	74

## LIST OF FIGURES

Figure 1.1 Online Shopping System	3
Figure 1.2 Project Schedule	5
Figure 2.1 A decision tree for the concept buys-computer, indicating whether or not a customer at ALLElectronic is likely to purchase a computer. Each internal (nonleaf) node represents a test on an attribute. Each leaf node represents a class (either buys_computer = yes or buys_computer = no)	17
Figure 3.1 System Development Model	49
Figure 5.1 The 3-tier Architecture	66
Figure 5.2 Context diagram of Online Shopping Administration Subsystem	78
Figure 5.3 Functional decomposition diagram of Online Shopping System	79
Figure 5.4 External event diagram of Process User's Login Event	80
Figure 5.5 External event diagram of Process Customer's Information Event	81
Figure 5.6 External event diagram of Process Supplier's Information Event	82
Figure 5.7 External event diagram of Process Customer's Order Event	83
Figure 5.8 External event diagram of Process Product's Information And Amount of Product Left Event	84
Figure 5.9 External event diagram of Update The Amount Of Product Event	85
Figure 5.10 External event diagram of Generate New Information Event	86
Figure 5.11 External event diagram of Process Prepaid Card's Information Event	87
Figure 5.12 External event diagram of Manage User Account Event	88
Figure 5.13 System diagram of Online Shopping Administration Subsystem	89
Figure 5.14 Primitive diagram of Process User's Login Event	90

Figure 5.15 Primitive diagram of Process Customer' Information Event	91
Figure 5.16 Primitive diagram of Process Supplier's Information Event	92
Figure 5.17 Primitive diagram of Process Customer's Order Event	93
Figure 5.18 Primitive diagram of Process Product's Information And Amount Of Product Left Event	94
Figure 5.19 Primitive diagram of Process Update The Amount Of Product Event	95
Figure 5.20 Primitive diagram of Process Generate New Information Event	96
Figure 5.21 Primitive diagram of Process Prepaid Card's Information Event	97
Figure 5.22 Primitive diagram of Manage User Account Event	98
Figure 5.23 Framework of Online Shopping Administration Subsystem	99
Figure 5.24 Login page of Online Shopping Administration Subsystem.	101
Figure 5.25 Supplier record module.	102

# **CHAPTER 1:**

## **INTRODUCTION**

### **1.1 Project Overview**

Nowadays, it has become a trend for overseas companies especially for those large and successful companies to have an online shopping system. Organizations have seen the bright future of the on-line sale market. They take this as an opportunity to expand and globalize their company. A lot of money was spent to own a leased line and Web servers in order to host their commercial Web site and provide the on-line sale service.

This Online Shopping Administration Subsystem is a web-based tool, which is used by administrator to archive the customer order, the information of customers, suppliers and products and calculate the sale of products. This subsystem is part of the system that we have called as “Online Shopping System”. It is set to improve services and upgrade the efficiency of the existing online shopping system.

In this subsystem, all the information of customers and transaction data will be study and analyze in order to identify the best customers. Then e-retailer can use this valuable information to identify market segments, plan, manage campaigns, develop strategies to attract more customers and interact with them in a personal way. Finally, it can help to decrease costs, increase sales and build long-lasting customer relationships.

## 1.2 Problem Definition

Below is the problem that has been recognized in most existing Online Shopping Administration Subsystem:

### ➤ **Inefficient customer data management**

No tool is provided to examine large pools of current and potential customer data in order to identify customers' on-line buying patterns, and understand their online behavior. It is such a waste since we can use this valuable information to design websites that better target, attract, engage, and retain current and future customers.

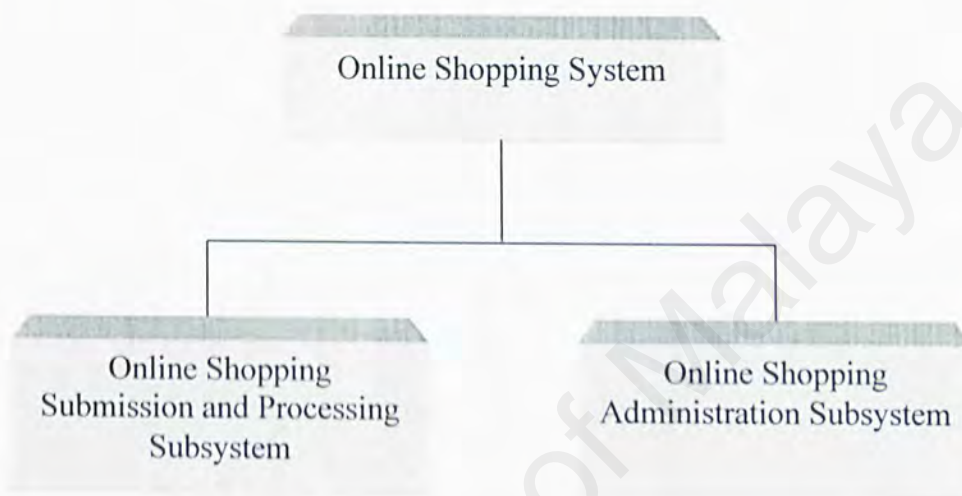
## 1.3 Project Objectives

The objectives of Online Shopping Administration Subsystem are specified below:

- Provides a system that can used to manage the customer personal information and customer order.
- Provides a system that can used to manage the supplier information and product information.
- Provides inventory control.
- Identify customers' online buying patterns and understand their online behavior.
- Use customers' information to identify market segments, plan, manage campaigns, develop strategies to attract more customers and interact with them in a personal way.

## 1.4 Project Scope

There are two students involved in developing the Online Shopping System. This system is divided into two subsystems, which are Online Shopping Submission And Processing Subsystem and Online Shopping Administration Subsystem. This project will mainly focus on Online Shopping Administration Subsystem.



**Figure 1.1 Online Shopping System**

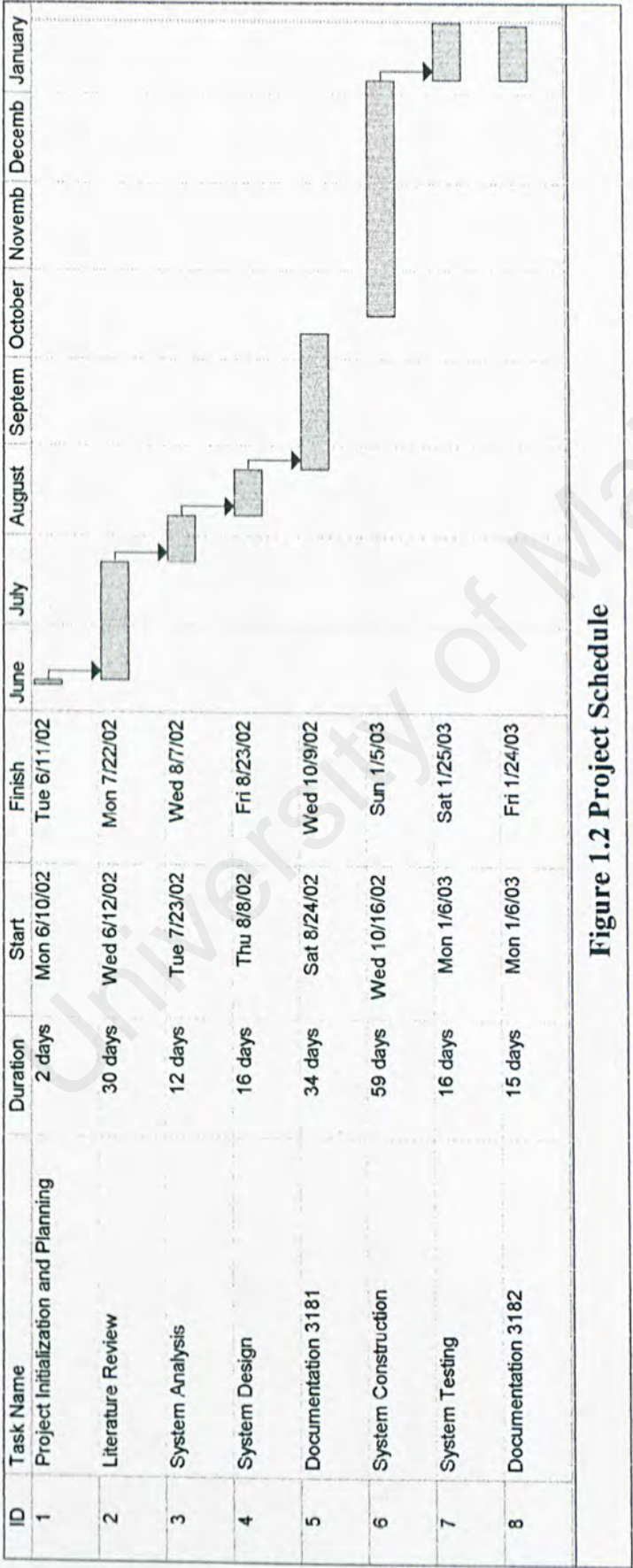
The scope of this subsystem covers the areas as specified below:

- Product ordering information such as item, amount, price and product description.
- Customer information such as name, address, age, occupation, income, etc.
- Inventory control including amount of stock, cost and profit.
- Product supplier information such as company's name, the type of products it provides, etc.
- Security of the system

Basically, the core function of this Online Shopping Administration Subsystem is searching and retrieving information from the database, adding, deleting and updating the database.

University of Malaya

1.5 Project Schedule



## 1.6 Expected Outcome

Below are the expected outcomes of the project:

- The Online Shopping Administration Subsystem will includes most of the business function that needed for retailing. These include inventory control, database management, etc. Due to the lack of time, the function of the subsystem will be limited to the basic business function.
- This subsystem can be integrated with Online Shopping Submission and Processing Subsystem to form the Online Shopping System.
- The Online Shopping System can interact in a more personal way with users and identify users' online buying patterns and online shopping behavior.
- The Online Shopping System can improve services and upgrade the efficiency of the existing system.

## **CHAPTER 2:**

### **LITERATURE REVIEW**

#### **2.1 Electronic Commerce**

Electronic commerce, or e-commerce, is the use of computers and electronic communications networks in doing business. It is a business tool that can be used to enhance business efficiency and effectiveness. E-commerce covers many aspects of communication, marketing, sales, publishing and customer service such as customer research, advertising, promotions, information publishing and dissemination, pre-sales enquiries, purchasing, transaction, funds transfer, public relation, delivery, production, after-sales service, customer support, etc.

Electronic commerce provides a new way of doing business to retailer, which will supersede many traditional commercial relationships. By using Internet-related technology, Electronic commerce brings benefits and cost-effective opportunities not just to large corporations and government departments, but also small to medium business in operating their business locally and internationally.

#### **2.2 Online Shopping**

##### **2.2.1 What is Online Shopping?**

Online shopping has grown greatly in just a few year times to become a multi-billion dollar industry. It is predicted to continue to grow greatly in the future. Online shopping basically is the ability or method to purchase goods and services from our home or office through computer. The procedures of purchasing

are just same as the procedures that we would do in a regular store. To begin the online shopping, we must connect to any online shopping website, browse their products and put all the selected items into the virtual “shopping cart”. To finish the online shopping, we must proceed to the virtual “check out”.

### **2.2.2 Growth Of Online Shopping**

There are over 70,000 new websites put on the Internet every hour. Internet traffic is doubling every three months. Projected annual Internet commerce revenue has grown from \$35 billion in 1998 to \$1.4 – \$3.2 trillion in 2003. *(Taken from CISCO Systems WEB Site)*

### **2.2.3 Advantages And Disadvantages Of Online Shopping**

#### **2.2.3.1 Advantages**

##### **➤ Variety**

Through online shopping, we are given an innumerable amount of products. There are many different kinds of products such as books, clothes, mobile phone, software, etc. Our choices for each kind of product are greatly increased.

##### **➤ Easy to find**

We can track down the list of all top vendors of a particular product by using the search engines on Internet, which will shorten the time for gathering information compare to the conventional ways of phone directories, advertisements and word of mouth.

➤ **Best Prices**

We are given the chance to compare the price of product since it is very easy for us to access so many vendors. There are many sites such as DealPilot (<http://www.dealpilot.com>) provides service of finding best prices for the product we want by scanning several online stores.

➤ **Time efficient**

Buying product in shopping mall will cause us to spend time in queuing up to pay money. Through online shopping, we need not to do so. Besides that, online shopping also saves our time in delivery. For example, many airlines now provide electronic tickets. After we purchase the ticket online, our job is finished. We can get the ticket from airport check in and no need to wait at the travel agency anymore.

➤ **Convenient**

Online shopping is open 24 hours per day, 7 days per week. Time is no more a problem. We can shop whenever we want. If there are any interruptions, we can stop our shopping and continue our shopping later when we want to.

➤ **Hard to find and specialized items**

Through online shopping, we can access those products that we are rarely come into contact with. For example, we can find hand-made items from many craftsmen that are only available at certain country fairs and community bazaars.

➤ **Low pressure**

In real life shopping, we always get bothered by a salesperson. But on the Internet, it would not happen. No one is going to corner us online (At least not yet).

### 2.2.3.2 Disadvantages

➤ **Can't touch**

We can't actually see, touch, taste, and/or smell the item we are interested in, therefore online shopping will not be our preferred method. However we could do our initial research online.

➤ **Delayed gratification**

We need to spend some time for the delivery of the purchased item. So if the item is needed immediately, then it is better to be purchased in a conventional shop. When buying online, there are some purchased items can be delivered in the same time or even faster than buying from conventional shop. For example, the software packages can be obtained in downloadable form and it is most convenient for us to buy online if the software will be installed on the same computer on which we are shopping. But if the purchased software was too big, let's say over 30 megabytes in size, but we are using a 56K modem, then surely it is a very long wait. Also when buying online, it is hard for us to obtain a replacement if something happen in the file (corrupted or infected by virus).

### ➤ The dangers

Shopping online brings some risks. Issues like whether it is safe to submit your credit card information, whether there is any hidden costs, and so on may arise.

## 2.2.4 Existing Online Shopping System

There are many online shopping systems available in the market and Internet. Below are some examples of the applications:

### 2.2.4.1 GeniusCart Pro™

GeniusCart Pro™ is manufactured by GeniusWeb.com. It includes online store for processing customer order. For the purpose of administration, it provides basic functions for managing customer order, product, customer information, and includes a content management tool, etc.

### 2.2.4.2 Line9 Tek9 Pro V6.2

It's Shopping Cart Software, automated Warehouse, and 1-Click Inventory. It's Intranet, Extranet, Reporting, Real-time Shipping and Credit Card Processing. For the administration, Line9 Tek9 Pro V6.2 is a web-based tool to configure and manage your entire web. It includes functions for generating report of orders according to manufacturer, customer or day, managing user account, inventory, shipping information, a content management tool (need not upload/download static html via ftp), a permission manager to control user access and an EDI tool (for import existing inventory or database of products into Tek9 Pro database). Also included a newsletter editor for edit or delete existing emails and a single-screen

interface for composing and sending out your newsletter. Text-only and HTML emails are both supported. Send your newsletter to literally thousands of people at once! Emails sent out can also provide a link at the bottom (if you enable this feature) allowing recipients to automatically unsubscribe themselves. [U1]

#### **2.2.4.3 SupraCart 4.0**

SupraCart 4.0 includes an online store for processing customer order, a site management tools for manage online store catalog content and over 170 function managers. Also it includes the abilities to set administrator access levels for varying degrees of access control. SupraCart 4.0 is an open source software, which allows user to create additional modules and functionality to further enhance one's site!

#### **2.2.4.4 uShop<sup>TM</sup> Online Shopping System**

uShop<sup>TM</sup> is an Java-based online shopping system that manufactured by Microburst Technologies, Inc. It consists a series of Java Applets and JavaScripts that plugs easily onto any web page. uShop<sup>TM</sup> includes an online store for processing customer online shopping and a control panel for the purpose of administration and content management. It includes the abilities for managing customer personal data, supplier data, product data, processing customer order, a built-in search engine, and built-in interfaces to LinkPoint, Authorize.Net and Planet Payment credit card processing systems. Besides that uShop<sup>TM</sup> provides a programmer's guide to assist with the development of own custom Java Applets and JavaScripts that interface with uShop<sup>TM</sup>.

#### **2.2.4.5 uStorekeeper<sup>TM</sup> Online Shopping System**

uStorekeeper<sup>TM</sup> is an online shopping system that manufactured by Microburst Technologies, Inc. It is developed by using Perl Script. It includes an online store for processing customer online shopping and a control panel for the purpose of administration and content management. It includes the abilities for managing customer personal data, supplier data, and product data, processing customer order, a built-in search engine, flat-file database and built-in interfaces to LinkPoint and Authorize.Net credit card processing system.

#### **2.2.4.6 Conclusion**

For the purpose of administration, the existing online shopping systems just provide functions like managing customer personal data, supplier data, product data, processing customer order, a built-in search engine, and also content management tool. But no tool is provided to examine large pools of current and potential customer data to identify customers' on-line buying patterns, and understand their online behavior.

### **2.3 Data Mining Technology**

#### **2.3.1 What Is Data Mining?**

Data mining is the task of discovering interesting patterns from large amount of data where the data can be stored in database, data warehouse, or other information repositories. It is a young interdisciplinary field, drawing from areas such as database systems, data warehousing, statistics, machine learning, data visualization, information retrieval, and high performance computing. Other contributing areas include neural networks, pattern recognition, spatial data analysis,

image database, signal processing, and many application fields, such as business, economics, and bioinformatics. [1] There are some other names/terms used to refer to data mining such as knowledge discovery (mining) in databases (KDD), knowledge extraction, data/pattern analysis, data archeology, data dredging, information harvesting, business intelligence, etc.

### 2.3.2 Why Use Data Mining?

Dr. Penzias, a Nobel Prize winner interviewed in ComputerWorld in January 1999, comments: *“Data mining will become much more important, and companies will throw away nothing about their customers because it will be so valuable. If you’re not doing this, you’re out of business.”* [2]

Data mining can be used in database analysis and decision support, which cover three main areas as below:

#### 1) Market analysis and management

Data mining can be used for target marketing, where clusters of “model” customers who share the same characteristics (such as interest, income level, spending habits, etc) can be found. For cross-market analysis, we can find the association/co-relations between product sales to make any prediction based on the association information. Data mining can help us to archive customer profiling, which provide information about what types of customers buy what products (clustering or classification). Data mining also helps us to identify customer requirements such as the best product for different customers and use prediction to find out the factors to attract new and more customers.

## **2) Risk analysis and management**

Data mining can be used in finance planning, asset evaluation and resource planning. It also used for the purpose of competition to keep track competitors and market directions, group clients into classes and a class-based pricing procedure. So that pricing strategy can be set in a highly competitive market.

## **3) Fraud detection and management**

Data mining techniques can be used in frauds detection. For example, inappropriate medical treatment was detected where Australian Health Insurance Commission identifies that in many cases blanket screening tests were requested (save Australian \$1m/yr). While for retail, analysts estimate that 38% of retail shrink is due to dishonest employees.

Besides that, data mining is used in other applications such as text mining (news group, email, documents), Web analysis and intelligent query answering [U2]

### **2.3.3 Classification**

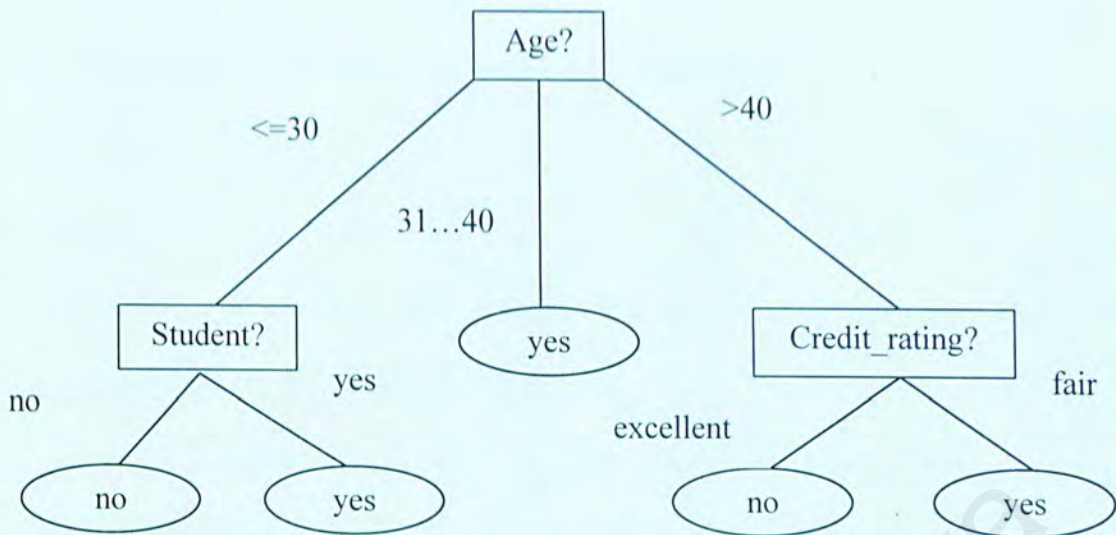
Databases are rich with hidden information that can be used for making intelligent business decisions. Classification is a form of data analysis that can be used to extract models describing important data classes or to predict future data trends. Classification is used to predict categorical labels. It classifies data (constructs a model) based on the training data set and the values (class labels) in a classifying attribute and use it in classifying new data. For example, a classification model may be built to categorize bank loan applications as either safe or risky. Researchers in machine learning, expert systems, statistics and neurobiology have

proposed many classification methods to handle small or large data size. The typical applications for classification method are credit approval, target marketing, medical diagnosis, treatment effectiveness analysis, etc. [1]

Basically, classification is a two-step process that consists of model construction and model usage. Model construction is used to describe a set of predetermined classes, where each sample/tuple is assumed to belong to a predefined class, as determined by the class label attribute. Usually the set of tuples used for model construction is training set. The model is represented as classification rules, decision trees, or mathematical formula. Model usage is used for classifying future or unknown objects. It is used to estimate accuracy of the model. The known label of test sample is compared with the classified result from the model. Accuracy rate is the percentage of test set samples that are correctly classified by the model. Test set is independent of training set, otherwise over-fitting will occur. [U3]

#### **2.3.3.1 Classification By Decision Tree Induction**

There are some different methods for classification. Classification by decision tree is one of the famous methods. A decision tree is a flow-chart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and leaf nodes represent classes or class distributions [U3]. An example of typical decision tree is shown in Figure 2.1.



**Figure 2.1** A decision tree for the concept buys-computer, indicating whether or not a customer at AllElectronic is likely to purchase a computer. Each internal (nonleaf) node represents a test on an attribute. Each leaf node represents a class (either buys\_computer = yes or buys\_computer = no). [1]

An unknown sample can be classified by testing the attribute values of the sample against the decision tree. Then a path is traced from the root to leaf node that holds the class prediction for that sample. Decision tree can easily be converted to classification rule.

Generating classification rules from a decision tree.

IF age = "<=30" AND student = "no"	THEN buys_computer = "no"
IF age = "<=30" AND student = "yes"	THEN buys_computer = "yes"
IF age = "31...40"	THEN buys_computer = "yes"
IF age = ">40" AND credit_rating = "excellent"	THEN buys_computer = "no"
IF age = ">40" AND credit_rating = "fair"	THEN buys_computer = "yes"

## 2.4 Programming Languages

### 2.4.1 General Purpose Programming Languages

#### 2.4.1.1 Java

##### Strengths

- A powerful object-oriented programming language, which can be run on the client side or on the server side, or both.
- Has a better bounds-checking capability and memory management.
- Machine portability
- Platform-independent. A single application can run on a computer running Win95/98/NT, UNIX, MVS or a Mac OS.

##### Limitations

- Less common compare to C++. Therefore the pre-built Java components are fewer.
- Not so easy to use. Considerable skill and training are needed.

#### 2.4.1.2 Visual Basic

##### Strengths

- It can be used to create databases, front-end applications, and scalable server-side components for most popular database formats, including Microsoft SQL Server and other enterprise-level databases.
- ActiveX technologies allow us to use the functionality provided by other applications, such as Microsoft Word word processor, Microsoft Excel spreadsheet, and other Windows applications. Also applications and objects created using the Professional or Enterprise editions of Visual Basic can be automate.

- Internet capabilities, which provide access to documents and applications across the Internet or intranet from within your application, or to create Internet server applications.
- The finished application is a true executable file (.exe) that uses a Visual Basic Virtual Machine that we can freely distribute.

### Limitations

- VB is not object-oriented. Some of the Microsoft Controls are only in existence due to this lack of object-orientation. For example, there is a listbox control and a filelistbox control. On object-oriented languages, the filelistbox control would be a specialization of the listbox control that can be further be specialized by the developer instead.
- Platform dependence. Visual Basic only runs on Microsoft operating system. There is a tendency to try to lock developers into support tools, such as MS SQL Server and MS Visual SourceSafe in order to make it more cumbersome for the integration of other vendor products. Besides that, ADO is the data access standard that VB follows, which is another Microsoft self-imposed standard that currently being followed only by the Microsoft set of tools.
- VB runs only on Microsoft's MTS application server and conforms only to the COM object model.
- There are shortcomings in reusability capability that make VB best suited to developing only small to medium-sized applications. The VB model does not support any Java access.

## **2.4.2 Markup Language**

### **2.4.2.1 HTML**

#### Strengths

- Used for displaying text, images, and multimedia.
- Gives instructions to Web browsers in order to control how documents are viewed and how they relate to each other.
- Can include hyperlinks.
- HTML supports add-on function. User can add their VBScript and also JavaScript inside HTML to make it become a dynamic HTML.

#### Limitations

- The content, layout, and format of pages are not separated. We can't separate the definition of content from layout and format.
- Without style definitions.

## **2.4.3 Client-side Scripting Languages**

### **2.4.3.1 JavaScript**

#### Strengths

- An easy to use object scripting language designed for creating live online applications, which link objects and resources on both clients and servers together.
- JavaScript scripts run on the browser that supports JavaScript, regardless of the operating system.

- It includes abilities for checking form contents, communicating with the user based on their actions, and modifying the web page dynamically without reloading the web page or without using Java, plug-ins or ActiveX controls.

#### Limitations

- JavaScript is keeping track of all the different versions. If we start using JavaScript functionally beyond the basic, we will have to worry about versions and which browser supports which version. [3]

#### 2.4.3.2 VBScript

##### Strengths

- Easy to use and learn.
- Language scalability. Once we learn VBScript, it is easier for us to learn Visual Basic for Applications and vice versa.
- It can work with the ActiveX, OLE and Java.
- The de facto language for ASP.

##### Limitations

- Limited to platforms running the Internet Explorer. Netscape browser and others do not support VBScript.
- Cannot perform file I/O. We cannot use API (Application programming Interface) calls to direct access to the underlying operating system.

## 2.4.4 Server-side Scripting Language

### 2.4.4.1 Active Server Page (ASP)

#### Strengths

- Active Server Pages (ASP) provides an open, compile-free application environment. It allows combination of HTML, scripts, and reusable ActiveX server components to create and run dynamic, interactive and high performance web pages from the server side.
- ASP enables ActiveX scripts and ActiveX server components run on the server. ActiveX components that are developed in any language, such as C++, Visual Basic, Java, COBOL, and others are supported. Powerful web-based applications with dynamic content can be created easily by combining scripts and components.
- ASP is dynamically generated from information kept in a database. Web site is updated when the database is updated.
- The communication between browser and server is limited to minimize network traffic.
- Provides quicker loading time. Because in the end, we are only actually downloading a plain HTML page.
- Can be used to run programs in languages that aren't supported by user's browser.
- Data provided to clients does not reside on the client's machine.
- More secure because the coding can never be viewed from the browser.

#### Limitations

- Uses proprietary technologies (technologies that manufactured and sold by a particular firm, under patent). Only Internet Information Server (IIS) and

Personal Web Server (PWS) support ASP. But ChiliASP can be install to run ASP on other web servers.

## **2.5 Web Development Tools**

### **2.5.1 Microsoft FrontPage 2000**

Microsoft FrontPage 2000 is part of the Microsoft Office 2000 family. It is one of the simplest and least complex HTML editors available in the market today. It runs on the Windows 95/98/NT/2000 Operating System. Since it is integrated into the Office 2000 package, it interacts easily with other components of Office 2000, such as Word, Excel, Access and PowerPoint.

#### **Strengths**

- Provides a WYSIWYG (what you see is what you get) environment for editing pages that do not require any programming knowledge and skill.
- Include themes that provide consistent look and feel, where a single theme can be applied to entire pages in a web site to create a consistent appearance for the site. Different themes also can be applied to each page in the site.
- There is a very easy and intuitive interface to let us paint a page.
- It supports many modern technologies, such as HTML 4.0, Cascading Style Sheets 1, insertion of Java language applets, ActiveX controls, Jscript and VBScript, themes, and Dynamic HTML effects.

#### **Limitations**

- Use proprietary technology where third-party database access products are not supported.

### 2.5.2 Microsoft Visual Interdev 6.0

Visual Interdev is an integrated development environment (IDE) for creating dynamic web sites. It is part of the Microsoft Visual Studio family, which consists of VB, Visual C++, Visual FoxPro, and Visual J++.

#### Strengths

- It combines many tools into a single package to simplify development jobs.
- Support project management; integrated databases access, syntax highlighting of different types of web files such as ASP or HTML, and automatic synchronization of local files with web server files.
- Pages that use Microsoft's ActiveX technologies and other similar computer technologies can be assembled.
- Visual Interdev also provides a robust development environment with a Scripting Object Model, design time control (DCTS), and an extensible toolbox for the purpose of rapid design, testing and debugging of web pages.

#### Limitations

- Not so easy to learn the use of Visual Interdev because there is a long list of technologies for the developer to absorb.
- Use, proprietary browser, which is Internet Explorer. However we also can build applications using other browsers by changing the default setting but there are some loss of VI features and functionality.

### 2.5.3 Macromedia Dreamweaver MX

The world's best way to create professional websites is now the easiest way to build powerful Internet applications. For the first time, you can work in a single environment to quickly create, build, and manage websites and Internet applications.

Get visual layout tools. Rapid web application development. And extensive code-editing support. All in one complete, integrated solution—Dreamweaver MX. [U4]

### Strengths

- Dreamweaver UltraDev's web application development features and Macromedia HomeSite's code editing support are combined to provide a single integrated environment to create, build, and manage websites and Internet application.
- An integrated development environment is provided to develop HTML, XHTML, XML, JSP, PHP, ASP, ASP.NET, and Macromedia ColdFusion websites.
- Provides complete control over code and design.
- Runs on Windows and Mac and supports cross-platform development environment for J2EE and .NET. Also an open integration with other tools like Macromedia Flash MX and Fireworks MX is provided.
- Makes the development of common Internet applications easier and faster by using libraries of code to create database insertion and update forms, recordset navigation pages, and user authentication pages.
- Enhanced coding features such as code hints, color coding, code validation, tag editors and tag choosers are included.

### Limitations

- There are some limitations when working in layout view, which is designed to make it easier to design with tables. It is not possible to nest a table inside of an existing table cell when in Layout View. It is not possible to insert a layer on your page when working in Layout View. It is not possible to convert from tables to layers when working in Layout View.

Layout View is designed to afford the same flexibility as laying out a page using Layers. Therefore, it doesn't necessarily make sense to convert your pages to layers when working in Layout View—particularly when considering the limited cross-browser compatibility of layers.

**Solution:** Switch to Standard View and make the desired changes. Once complete, you can switch back into Layout View. [U5]

## 2.6 Web Server

Web server is used to allow you to serve content over the Internet using the Hyper Text Markup Language (HTML). The Web server accepts requests from browsers like Netscape and Internet Explorer and then returns the appropriate HTML documents. A number of server-side technologies can be used to increase the power of the server beyond its ability to deliver standard HTML pages; these include CGI scripts, server-side includes, SSL security, and Active Server Pages (ASPs). [U6]

### 2.6.1 Apache 2.0.39

Apache remains the king of Web servers despite intense efforts by Microsoft and Netscape to gain dominance in the market. In fact, the latest Netcraft surveys indicate that the freeware Apache is widening its lead over the rest of the field. Apache users have come to rely on the server's rock-solid reliability, outstanding performance, and rich set of features.

#### Strengths

- Price (freeware)
- Rock-solid reliability
- Extensibility

- Quick tech support via Usenet newsgroup
- Streamlined interface
- Provides cross-platform support. Apache runs on Windows (95/98/NT), OS/2, and all the major variants of Unix.
- The server is fully compliant with HTTP/1.1 and supports API and ISAPI (NT).
- Apache distributes a core set of modules that handle everything such as user authentication, cookies, typo correction in URLs, etc. Also include many other tried and true custom modules.
- Unquestionable features of security, performance, and robustness. Many of the most accessed sites in the world run Apache or Apache derivatives. Public distribution of the source code results in patches for the software being distributed quickly, and allowing public scrutiny helps ensure that security holes in the software are promptly caught and reported. As a result, Apache's large user base has allowed its developers to create a package that is extremely stable and secure and one that is also able to compete more effectively with commercial packages in terms of both raw speed and integrated features.

#### Limitations

- Lack of visuals, wizards, and/or browser-based administration tools. Setup and maintenance of the server are accomplished via command-line scripting tools. Unlike most popular commercial servers, Apache offers neither browser-based maintenance capabilities nor any GUI configuration/administration tools. This is an advantage for some developers, but for others it can translate into higher deployment and maintenance costs,

especially if the site's administrators are unfamiliar with the fundamentals of the server.

- No Mac version available
- NT version is in its infancy (still lacks a number of the UNIX versions' performance enhancements)
- More extensive technical support requires the purchase of a third-party support contract. [U7]

### **2.6.2 Internet Information Server (IIS)**

Microsoft's IIS version 5, which comes exclusively as part of the Windows 2000 Server operating system, contains many new features along with performance and reliability enhancements.

#### **Strengths**

- IIS v5.0 can be used by those familiar and comfortable with Windows operating systems as a first-time web server, and by hosting providers and large corporate installations as a high-end server.
- Provide Indexing.
- Performance and security enhancements. Microsoft has consolidated security tips in its documentation under Administration | Server Admin | Security | IIS Security Checklist. Tips include restricting guest accounts and setting appropriate file permissions. However, setting up client-side certificates is still far too complex and poorly documented.

- Provides wizards to help simplify some common tasks. For example, the Permissions Wizard (to synchronize and align Web and NTFS security settings), the Web Server Certificate Wizard (to obtain and install server certificates), and the CTL Wizard (to create and modify certificate trust lists).
- Well-integrated server administration tools.
- Easy to configure to enable multiple machines to share the load and deliver more reliable Web services.
- Web-based Distributed Authoring and Versioning (WebDAV) support makes for easier collaborative publishing. WebDAV is an emerging standard designed to simplify the construction of intranets and enable multiple users to publish documents to a common Web server.

#### Limitations

- No Unix version.
  - Documentation lacking on newest features.
  - Only runs on Server edition of Windows 2000.
  - Require at least a computer with a 200 MHz Pentium with 128 MB of RAM.
- [U8]

### 2.6.3 Jigsaw 2.2.1

Jigsaw, as developed by the World Wide Web Consortium (W3C), is designed to be a technology demonstration rather than a full-fledged release. It's purposely intended as a project to showcase new technologies, but in the case of Jigsaw, this Web server also ends up being more robust than the average Web

server. Most importantly, though, Jigsaw serves as a useful blueprint to the future of the HTTP protocol and object-oriented Web servers.

#### Strengths

- Freely available under open-source license
- Jigsaw is written totally in Java and works within an object-oriented framework. Since it's written entirely in Java, it can run under any operating system supporting the Java Development Kit (JDK); W3C reports that Jigsaw works under Windows 95/98/NT and Solaris 2.x, while other users have reported success with AIX, OS/2, BeOS and MacOS. Java also enables Jigsaw to be extensible and portable across server platforms.
- Applet-based remote administration
- Excellent documentation

#### Limitations

- Lacks many tools needed for commercial Web server, including advanced authentication and security tools, Microsoft FrontPage Extensions support, and browser-based administration. [U9]

### 2.6.4 Personal Web Server (PWS) 4.0

Microsoft's Personal Web Server (PWS) is a scaled-down version of the commercial Information Internet Server (IIS) included with the Server edition of Microsoft Windows NT. Designed for Windows 95 and Windows 98, PWS is a great entry-level Web server that makes it easy to publish personal home pages, serve small Web sites, and share documents via a local intranet.

#### Strengths

- Freely downloadable.

- Easy to setup and use.
- Support Active Server Page (ASP).
- Excellent for serving small Web sites and personal home pages. PWS serve as a platform for testing out Web sites (check the validity of links, scripts, and applications, etc) on your Windows 95/Windows 98 computers before hosting them on the Internet.
- Includes many of the same features found in IIS.

#### Limitations

- Without UNIX version.
- Lacks some of the more advanced features included with IIS (such as the Index Server, Certificate Server, and Microsoft Site Server Express tools).
- Only runs on Windows 95/98/Windows NT Workstation.
- Slower than IIS. [U10]

**Table 2.1 Comparison between the four popular web servers**

Web Server	PWS	IIS	Apache	Jigsaw
Version	4.0	5.1	2.0.39	2.2.1
Company	Microsoft Corporation	Microsoft Corporation	Apache Group	W3C
Released	12/4/97	4/11/02	5/8/02	4/19/02
Brief Description	Supports up to 10 concurrent connections, no longer supports FTP.	The best Web server available for Windows NT gets even better with this version	The most popular web server currently in use on the internet.	Java-based server that provides a blueprint for future web development.
Features	A great entry-level web server for publishing your own web pages.	Updated version of IIS. Index server, Message Queue Server (MSMQ) and Transaction Server (MTS)	High performance, extremely reliable web server.	Administration tools; completely object-oriented design; written entirely in Java.
Platform	Windows 95/98, and Windows NT	Windows 2000 Server, Advanced Server	Unix, Windows 95/98, and Windows NT	Windows NT, 95/98, Unix (Solaris 2.x)
Status	Freeware - packaged with Microsoft IIS as part of NT 4.0 Option Pack	Included free with all Windows 2000 Server versions	Freeware	Freely available under open-source license.

## **2.7 Database Technology**

### **2.7.1 Database Access Standard Interface – ODBC and OLEDB**

#### **2.7.1.1 ODBC (Open Database Connectivity)**

ODBC is designed for maximum interoperability—that is, the ability of a single application to access different database management systems (DBMSs) with the same source code. Database applications call functions in the ODBC interface, which are implemented in database-specific modules called drivers. The use of drivers isolates applications from database-specific calls in the same way that printer drivers isolate word processing programs from printer-specific commands. Because drivers are loaded at run time, a user only has to add a new driver to access a new DBMS; it is not necessary to recompile or relink the application. [U11]

##### **Strengths**

- Provides an ability of DBMS independent for processing relational data.

##### **Limitations**

- Cannot support the processing of non-relational data.

#### **2.7.1.2 OLE DB**

OLE DB, a set of interfaces for data access, is Microsoft's component database architecture that provides universal data integration over an enterprise's network — from mainframe to desktop — regardless of the data type. Microsoft's Open Database Connectivity (ODBC) industry-standard data access interface continues to provide a unified way to access relational data as part of the OLE DB specification. Over time, OLE DB is expected to lead new database products that are assembled from best-in-class components rather than from the monolithic products available today.

## Strengths

- OLE DB provides a flexible and efficient database architecture that offers applications, compilers, and other database components efficient access to Microsoft and third-party data stores.
  - OLE DB is the fundamental Component Object Model (COM) building block for storing and retrieving records and unifies Microsoft's strategy for database connectivity. It will be used throughout Microsoft's line of applications and data stores.
  - OLE DB defines interfaces for accessing and manipulating all types of data. These interfaces will be used not just by data-consuming applications but also by database providers. By splitting databases apart, the resulting components can be used in an efficient manner. For example, components called service providers can be invoked to expose more sophisticated data manipulation and navigation interfaces on behalf of simple data providers.
- [U12]

- Enable consistent access to SQL, non-SQL and also unstructured data source across the enterprise and the Internet.
- It is faster and easier to use compare to ODBC.

## Limitations

- To enable the access into OLE DB database by using OLE DB interface, we must connect through an OLE DB data provider that supports OLE DB version 2.0 or higher.

## **2.7.2 Database Access Libraries – DAO, RDO, ADO**

There are different ways to open and manipulate databases. These three objects can accomplish exactly the same task.

### **2.7.2.1 Data Access Objects (DAO)**

Data Access Objects, or DAO, was Microsoft's first object oriented solution to manipulate database using the Jet Database Engine. It enables VB applications to communicate with a database (the JET Engine) through ODBC.

#### **Strengths**

- DAO's programming interface is robust and easy to use.
- It provides a framework that enables directly accessing and manipulating database objects.
- Provides database control from any application that supports Visual Basic for Applications, such as Microsoft Excel, Microsoft Access, and Microsoft Visual Basic. DAO let user to create and manage local or remote databases in many different formats.

#### **Limitations**

- Many DAO commands that have been kept for backward compatibility can make the syntax quite unpleasant to look at.

### **2.7.2.2 Remote Data Objects (RDO)**

RDO is an interface used to enable VB applications to communicate with a relational database (various RDBMS) through ODBC. Because of the considerations to optimize speed and control, developer ignored JET when creating transaction centric application to a RDBMS. Therefore, RDO was created to overcome this problem.

## Strengths

- It is an object interface that enables optimal speed, control and ease of programming by directly calling ODBC.
- RDO provides access to server side cursor in order to minimize network traffic.

## Limitations

- Unable to create database objects such as tables, views, and stored procedures by using RDO.
- Requires Enterprise Edition of Visual Basic to use RDO.

### 2.7.2.3 ActiveX data Objects (ADO)

ADO allows VB application or other web tools (browsers) to interface with various data sources. ADO programming model represents the best of the existing Microsoft data access programming models. If you are familiar with Data Access Objects (DAO) or Remote Data Objects (RDO), you will recognize the interfaces and will be able to work with them very quickly. There are considerable improvements in the model, and tasks that were awkward in previous models have either been fixed or eliminated from the ADO model.

## Strengths

- The ADO objects provide you with the fastest, easiest and most productive means for accessing all kinds of data sources. The ADO model strives to expose everything that the underlying data provider can do, while still adding value by giving you shortcuts for common operations.
- ADO is Microsoft's strategic, high-level interface to all kinds of data. ADO provides consistent, high-performance access to data, whether you're creating

a front-end database client or middle-tier business object using an application, tool, language, or even an Internet browser. ADO is the single data interface you need to know for 1- to n-tier client/server and Web-based data-driven solution development.

- ADO is designed as an easy-to-use application level interface to Microsoft's newest and most powerful data access paradigm, OLE DB. OLE DB provides high-performance access to any data source, including relational and non-relational databases, email and file systems, text and graphics, custom business objects, and more. ADO is implemented with a small footprint, minimal network traffic in key Internet scenarios, and a minimal number of layers between the front-end and data source-all to provide a lightweight, high-performance interface. ADO is easy to use because it is called using a familiar metaphor - the OLE Automation interface, available from just about any tool and language on the market today.
- ADO was designed to combine the best features of, and eventually replace RDO and DAO, it uses similar conventions with simplified semantics to make it easy to learn for today's developers. [U13]

#### Limitations

- Although ADO is one of the newest technologies for database access, but in almost every case RDO rivals its performance.
- Microsoft has stated that ADO is the technique for Internet use only.

## 2.7.3 Database Management System (DBMS)

### 2.7.3.1 Microsoft Access 2000

Microsoft Access 2000 is a powerful relational database application that enables a desktop user to create and manipulate database systems easily and efficiently. Microsoft Access most suitable for the use of individuals and workgroups to manage megabytes of data.

Since its first introduction in 1992 as Microsoft Access, the product has become a leader in the desktop database category both for its power and ease of use. Now, Access 2000 provides stronger integration with other Office applications as well as with enterprise-level databases.

#### Strengths

- Microsoft Access lets users find and manage their data easily. It includes ease-of-use wizards such as the Database Wizard (for getting up and running quickly), and the Simple Query Wizard (for easily finding information from the data).
- More consistent interface with other office applications.
- Adds Data Access Pages, which are HTML pages that let user interact with data over the web from any location.
- Allows and maintains live links to the database.
- Its support of OLE DB, can act as a front-end to high-end database engines.
- Scalability. The Microsoft Access Upsizing Wizard utility takes a Jet database and creates an equivalent database on SQL Server with the same table structure, data, and many other attributes of the original database. It will create same table structures, indexes, validation rules, defaults, auto numbers, and relationships. Also takes advantage of the latest SQL Server

functionality wherever possible. But there are no modifications made to reports, queries, macros or security.

#### Limitations

- It is a desktop relational database management system (RDBMS) intended for individuals and workgroups only.
- The JET and MSDE engines in Access 2000 can only support 2GB per database.

#### 2.7.3.2 Microsoft SQL Server 7.0

Microsoft SQL Server 7.0 is a Microsoft's database product. SQL Server is a relational database management system (RDBMS) intended for a wide range of corporate customers and independent software vendors (ISVs).

#### Strengths

- Scalability. Able to scale from the laptop to the enterprise using the same code base, by offering full code compatibility
- Support auto-configuration and self-tuning
- Includes an integrated online analytical processing (OLAP) server
- Provides integration with Data Transformation Services
- Provides Data Warehousing Framework as the first comprehensive approach to solve the metadata problem
- Provide multi server management for large numbers of servers
- Wide array of replication options of any database
- Universal Data Access, Microsoft's strategy for enabling high-performance access to a variety of information sources

## Limitations

- Platform dependent. SQL Server runs only on Windows platforms.
- When the database detects corruption or a hardware problem that cannot be repaired during runtime, it immediately shuts down and make the engine offline. This can significantly affect the reliability of the system.
- Not all features available on all platforms. Because of the limitation inherent in the desktop OS, some features of the SQL Server 7.0 Enterprise Manager are not available on Windows 95/98 implementations.
- SQL Server 7.0's MMC-based Enterprise Manager cannot connect to SQL Server 6.x database because of SQL-DMO incompatibilities. Thus, users must run two different sets of database management tools until conversion is complete.

## 2.8 Security Considerations

### 2.8.1 Network Security

#### 2.8.1.1 What Is Network Security?

Defined in its simplest form, security is "freedom from danger or risk of loss; safety." Thus network security can be any effort made to protect a network from danger or risk of loss; or in other words, to make the network safe from intruders, errors, and other threats. While this definition may be an over-simplification, it establishes two underlying assumptions about network security: (1) that the network contains data and resources necessary for the successful operation of the

organization; and (2) that the network's data and resources are of sufficient value to be worth protecting [U14].

#### 2.8.1.2 Confidentiality, Integrity and Availability

There are three main areas in computer security, which are confidentiality, integrity and availability of electronic data.

##### ➤ **Confidentiality**

System only allows those who are authorized to access certain electronic data.

##### ➤ **Integrity**

System only allows those who are allowed to modify electronic data to do so.

##### ➤ **Availability**

System allows those authorized to access the electronic data when it is needed to do so. Also need to have adequate backup and disaster-recovery plans.

#### 2.8.1.3 Identification and Authentication (I&A)

Identification and authentication are used to ensure that only authorized persons or computers are allowed for data access or data modification on a network.

- **Identification** refers to the process that occurs during initial login whereby a person provides some type of security token, typically a unique username or

user ID, to identify that user on the system. In effect, identification is akin to telling the system, "This is who I am."

- **Authentication** is a verification process that requires the user to provide another token, typically a password known only to the user, to affirm that the identity is being assumed by its rightful owner. The user is essentially telling the system, "Here is some private information to prove that I am who I claim to be." [U14]

There are a few form of authentication, such as smart cards, biometrics (fingerprint, retinal scanners, etc) and password. Password is the most commonly used today. Therefore it is very important to protect the authentication secrets (passwords) themselves to prevent any unauthorized access by any user.

#### 2.8.1.4 Accountability

Based on the user identity provided during the login, the system can determine which of the network's resources the user is authorized to access and at what permission level. A key goal of I&A is to enable individual accountability: the ability of the system to reliably determine who you are when you log in and to track your actions while you are logged in. When the I&A mechanisms are sufficiently strong, you can be reasonably assured that any actions taken under a particular user account were in fact performed by the authorized user of that account. [U14]

#### 2.8.1.5 Access Controls

In a network, access controls are applied to the data based on the users' system identities. These controls rely on identification and authentication, since users must be identified and have their identities authenticated before the system can enforce any access controls. There are two types of access control.

- **Discretionary access controls** restrict access to files or directories based upon the identity of the user. They are discretionary in that the owner of a file (or any user with sufficient rights) is free to assign or revoke the access rights of other users to that file.
- **Mandatory access controls** restrict access by means of special attributes that are set by the security administrator and enforced by the operating system. These controls cannot be bypassed or changed at the discretion of non-privileged users. Mandatory access controls are typically based on *labeling*, where the administrator labels each item of information with a classification name such as Unclassified, Confidential, Secret, and so on. Each user is assigned a clearance level from the same set of classifications. The operating system then controls access to information based upon these classifications. [U14]

#### 2.8.1.6 Audit

It is very important for a secure system to be able to audit authorized access by keeping a record of who did what and when, and who was prevented from doing what and when. Usually, auditing is done by maintaining audit trails, or logs of significant events and also user actions that take place within the system. Auditing information must be protected to prevent intruders from being able to erase all traces of their actions. [U14]

#### 2.8.1.7 Non-repudiation

Occasionally, it might be necessary to prove that a particular user made a particular transaction long after it happened. Since the ability to modify data often includes the ability for users to remove their own access rights, it is possible for an unscrupulous user to change a piece of data, remove his or her rights to that data,

and then deny any involvement in the change. A security system must therefore provide something beyond mere access control. With non-repudiation, irrefutable evidence of the transaction is created and stored. The entity which made the access or modification cannot falsely deny (or repudiate) its actions later. (Non-repudiation is common in secure e-mail systems, where it is used to prove that a particular user did in fact send a message purporting to come from him, and that a user did get a message he might deny having received.) [U14]

#### **2.8.1.8 Data Encryption**

Data Encryption is the most common and most viable method to protect sensitive data. Encryption is the transformation of data into a form unreadable by anyone without a secret decryption key. Its purpose is to ensure privacy by keeping the information hidden from anyone for whom it is not intended, even those who can see the encrypted data [U15].

The SSL (Secure Sockets Layer) Handshake Protocol [Hic95] was developed by Netscape Communications Corporation to provide security and privacy over the Internet. The protocol supports server and client authentication. The SSL protocol is application independent, allowing protocols like HTTP (HyperText Transfer Protocol), FTP (File Transfer Protocol), and Telnet to be layered on top of it transparently. Still, SSL is optimized for HTTP; for FTP, IPsec might be preferable. The SSL protocol is able to negotiate encryption keys as well as authenticate the server before data is exchanged by the higher-level application. The SSL protocol maintains the security and integrity of the transmission channel by using encryption, authentication and message authentication codes.

The SSL Handshake Protocol consists of two phases: server authentication and an optional client authentication. In the first phase, the server, in response to a client's request, sends its certificate and its cipher preferences. The client then generates a master key, which it encrypts with the server's public key, and transmits the encrypted master key to the server. The server recovers the master key and authenticates itself to the client by returning a message authenticated with the master key. Subsequent data is encrypted and authenticated with keys derived from this master key. In the optional second phase, the server sends a challenge to the client. The client authenticates itself to the server by returning the client's digital signature on the challenge, as well as its public-key certificate.

A variety of cryptographic algorithms are supported by SSL. During the "handshaking" process, the RSA public-key cryptosystem is used. After the exchange of keys, a number of ciphers are used. These include RC2, RC4, IDEA, DES, and triple-DES. The MD5 message-digest algorithm is also used. The public-key certificates follow the X.509 syntax. [U16]

## **2.9 Open Source Code**

### **2.9.1 WEKA 3 – Data Mining With Open Source Machine Learning Software In Java**

Weka is a collection of machine learning algorithms for solving real-world data mining problems. It is written in Java and runs on almost any platform. The algorithms can either be applied directly to a dataset or called from your own Java code. Weka is also well-suited for developing new machine learning schemes. Weka is open source software issued under the GNU General Public License.

Implemented schemes for classification include: decision tree inducers, rule learners, naive Bayes, decision tables, locally weighted regression, support vector machines, instance-based learners, logistic regression, voted perceptrons and multi-layer perceptron.

Implemented schemes for numeric prediction include: linear regression, model tree generators, locally weighted regression, instance-based learners, decision tables and multi-layer perceptron.

Implemented "meta-schemes" include: bagging, stacking, boosting, regression via classification, classification via regression and cost sensitive classification.

Also included are clustering methods (Cobweb and an EM algorithm), and an association rule learner (Apriori). Apart from actual learning schemes, Weka also contains a large variety of tools that can be used for pre-processing datasets. The Java-based Weka system (Weka 3) replaces Weka 2 (which was based on C, Tcl/Tk, Prolog, and the kitchen sink). [U17]

## **CHAPTER 3:**

### **METHODOLOGY**

#### **3.1 Overview**

A methodology is a set of procedures, techniques, tools and documentation aids that used by system developers to develop and implement a system.

#### **3.2 Waterfall Model With Prototyping**

The Online Shopping Administration Subsystem follows a system development model, which is waterfall model with prototyping.

Basically, the overall development is broken into distinct phases same like the waterfall model, with the exception that prototypes are used during the development of the modules of the subsystem. The incremental aspect of the model comes in when additional functions are added iteratively to each of the module prototypes.

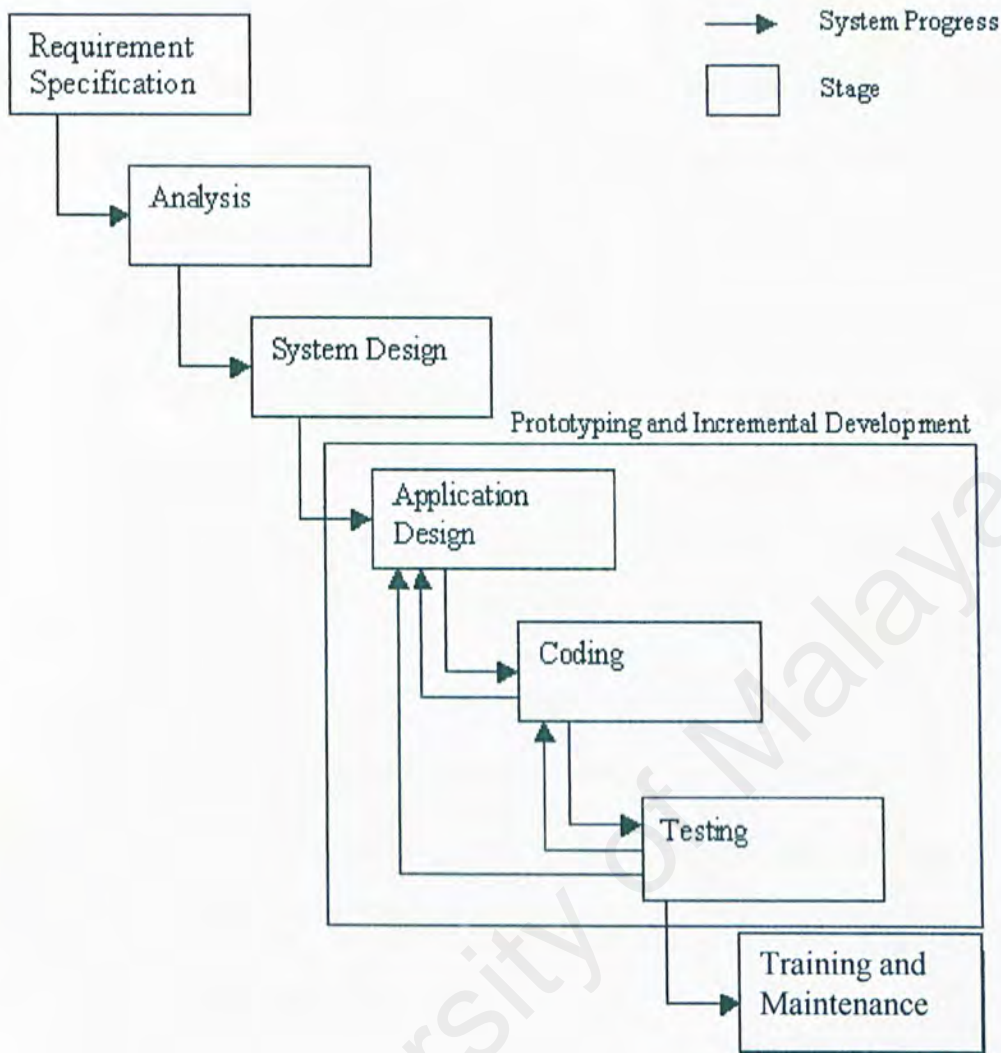
A prototype model is a partially developed product that helps users and developers to examine some aspects of the proposed system to determine if it is suitable and fulfills the requirements. Prototyping enables developers to seek initial reactions from users and management of the prototype, users' suggestions to change or clean up the prototyped system, possible innovations for it and revision plan about the parts of system that need to be done first. So that the developer can improve prototype and also modify the system plans with minimum risks, disruptions, uncertainties and expenses.

Prototyping is an information-gathering technique for gathering quickly the specific information about user's information requirements. The prototype system is

actually an operational portion of the eventual system that will be built. It is not a complete system, only some essential functions will be included in the model.

The initial stages of the development model are conducted on a general basis, covering most of the Online Shopping Administration Subsystem as a whole. Upon reaching the prototyping stage, individual modules are developed separately and prototyping is also conducted separately. During this time, the future integration needs of the separate modules also have to be catered for, and are handled during this time for future needs.

Once each of the individual modules has been sufficiently developed to a reasonable stage of completeness, they are brought together for the final integration into the complete Online Shopping Administration Subsystem, unifying the separate parts and forming a coherent system. Final testing is then conducted on the Online Shopping Administration Subsystem as a whole. Following the conclusion of this project, the Online Shopping Administration Subsystem may be further added with more modules to increase its functionality.



**Figure 3.1 System Development Model**

The phases that this system development model must pass through from start to finish are:

➤ **Requirement Specification**

It is the first phase of this system development model, which the needs and requirements of the Online Shopping Administration Subsystem are identified, prioritized and arranged. A detailed plan will be developed for implementing and conducting the remaining phases of the model.

➤ **System Analysis**

The current Online Shopping Administration Subsystems are studied and various fact-finding methods are used to identify the new requirements. All features of this Online Shopping Administration Subsystem are described independently.

➤ **System Design**

The logical specifications of this Online Shopping Administration Subsystem are transformed into technology-specific details.

➤ **System Implementation (Application Design, Coding And Testing)**

A quick design is formulated. The quick-design focuses on the top-level architecture and data designs issues rather than on detail procedural design. Then, the quick-design lead to the construction of a prototype. The prototype is tested and evaluated to refine the requirements. It is a process of iteration and occurs until all requirements are formalized or until the prototype has evolved into a product system.

➤ **System Maintenance**

The Online Shopping Administration Subsystem will be systematically repaired and improved.

### 3.3 Why Choose Prototyping?

There are a few reasons on why the prototype model is chosen:

➤ **Able to change the system early in its development**

Prototyping is flexible and adaptable, which enables prototype to change many times based on the feedback that calls for a change. Feedback will tell you what and where changes are wanted. When changing a prototype, the developer do not need to worry about wasting time to developed a full-blown

system only to find that it needs modifications. Besides that, it is much more easier to trace and detect system problems in a prototype with limited features and limited interfaces than they are in a complex system.

➤ **Throwing undesirable functions and systems**

There is a possibility of throwing a suggested system when it does not meet user's expectation. Prototyping can prevent from wasting time and money on a project that is unworkable.

➤ **Designing a system for user's needs and expectations**

Prototyping involves interaction with the users throughout the systems development cycle. Therefore the prototype can be used as an interactive tool that shapes the final system to accurately reflect user's requirements.

### **3.4 Information Gathering Technique**

Some techniques are used to collect the information needed to determine this subsystem's requirements.

#### **3.4.1 Research**

Research on books and journals has been done in order to gather relevant information needed in this subsystem. Besides that, Document Room in FSKTM also serves as a good place to provide valuable information.

#### **3.4.2 Internet Surfing**

It is a very efficient way to collect all the up-to-date and useful information, which is needed in developing the system.

#### **3.4.3 Discussion With Supervisor**

The discussions with supervisor have helped me to understand the needs of the system and collect more information.

## **CHAPTER 4:**

### **SYSTEM ANALYSIS**

#### **4.1 Requirement Analysis And Specification**

##### **4.1.1 Functional Requirements**

A functional requirement describes an interaction between the system and its environment. [4] Functional requirements are system services that are expected by the user. It consists set of functions that are required to be included in the system in order to achieve the objective of Online Shopping Administration Subsystem. The functional requirements of this Online Shopping Administration Subsystem will be explained in the 8 modules below:

##### **1) Customer Record Module**

This module keeps the entire customer's personal information, interested products and buying record.

##### **➤ Query**

It allows users to search for the customer's record from the database.

The searching can be performed through inputting or selecting the customer's name or their ID number or their location or combination of name, ID number and location. The system then matches user inputs with the records in the database. If there is a match, the information of the customer will be displayed.

## 2) Supplier Record Module

This module keeps the entire supplier's information and order record.

### ➤ Query

It allows users to search for the supplier's record from the database. The searching can be performed through inputting or selecting the supplier's name or their ID number or company's name or combination of name, ID number and company's name. The system then matches user inputs with the records in the database. If there is a match, the information of the supplier will be displayed.

### ➤ Add

It allows users to add new supplier record into the database. User will first fill in the supplier's data. Then the system will search the database whether the supplier's record exist. If the same supplier ID exists, the error message to deny user add operation will be prompted. If there is no same supplier ID found, user is allowed to proceed the add operation.

### ➤ Edit

It allows users to change or update the supplier's record if there are any changes. The user first needs to verify whether the record exists or not in the database, and then make the amendments.

### ➤ Delete

It allows users to delete the supplier's record from the database, which are not needed anymore. The user will first search for the supplier's record that needs to be deleted. Then only the user click

delete and the warning message will be prompted to ensure the deletion of the record.

### 3) Customer Order Module

This module displays the entire customer order file.

#### ➤ Query

It allows users to search for the customer order file from the database.

The searching can be performed through selecting the date of ordering or the order ID or the delivery status. Then the selected order file will be displayed.

#### ➤ Delete

It allows the users to delete the order file. The user will first search for the order record that needs to be deleted. Then only the user click delete and the warning message will be prompted to ensure the deletion of the record.

### 4) Inventory Control Module

This module keeps all the product's profile, the supplier of the products, the current amount of stock

#### ➤ Add

It allows the user to add in a new product's profile to the database. User will first fill in the product's details. Then the system will search the database whether the product's record exist. If the same product ID exists, the error message to deny user add operation will be prompted. If there is no same product ID found, user is allowed to proceed the add operation.

➤ **Delete**

It allows the users to delete a product's profile that is not needed anymore. A warning message will be prompted to ensure the deletion of the record.

➤ **Query**

It allows users to search the profile of any product from the database. The searching can be performed through inputting or selecting the product ID, product's name or category or combination of name, ID number and category. Then the selected order file will be displayed.

➤ **Edit**

It allows the users to update or change the details of the products in the database only if the record want to edit exists in the database.

## 5) **Product Delivery And Stock Update Module**

This module keeps track the type and amount of products in the store by updating the amount of products in the database every time the products are sent to the customer.

➤ **Update**

User can update the delivery status of the customer order. If the delivery status is set to "True" then the amount of products in the database will be update by deduct the amount of product in database with the quantity of ordered product. If the quantity of ordered product is more than the stock then update operation is denied. A warning message will be prompted. Otherwise the update operation is granted and amount of current stock will be displayed.

## 6) New Information Generator Module

Data mining technique is used in this module to classify customer into different class label.

### ➤ **Generate**

Generate button is provided where users can click on it to generate new customer information. Customer will be classified to different class label such as “buy computer”, “buy cosmetic” or “buy fashion” by using classification rule. A warning message will be prompted to the user to remind the user that the generating process is time consuming because it involves many database select and insert operation.

### ➤ **Clear**

Clear button is provided where users can click on it to clear all customer class labels.

### ➤ **Query**

It allows users to search the customer profile from the database based on the customer class label. The searching can be performed through selecting the customer class label. Then the customer profile that matches the selected class label will be displayed.

## 7) User Management Module (Security Module)

This module is used to manage user accounts and provide system access control. The concepts of discretionary access control and mandatory access control are implemented in this module. Users are categorized into different user levels that are admin1, admin2, general1, general2 or management where admin1 is the highest user level. Only user with user level admin1 is

allowed to access this module and perform add, query, edit or delete operation.

➤ **Login**

It allows the authorized users to login to the system if the user has type in the correct login name and password. If the user has typed in the wrong login name and password, the system will prompt a warning message to warn the user.

➤ **Add**

It allows user with user level admin1 to add new user account into database.

➤ **Query**

It allows user to search the user account details from the database based on the username or user level. The searching can be performed through selecting the username or user level. Then the selected record will be displayed.

➤ **Delete**

It allows user with user level admin1 to delete user account that is no needed anymore.

➤ **Edit**

It allows user with user level admin1 to modify existing user account details.

## 8) Prepaid Card Module

This module is used to manage prepaid card records. Only user with user level admin1 is allowed to access this module and perform add, edit, delete and query operations.

➤ **Add**

It allows user to add new prepaid card details

➤ **Edit**

It allows user to edit existing record in the database.

➤ **Query**

It allows user to search the required prepaid card record

➤ **Delete**

It allows user to delete the unwanted prepaid card record from the database.

#### **4.1.2 Non-Functional Requirements**

A non-functional requirement or constraint describes a restriction on the system that limits our choices for constructing a solution to the problem. [4] Non-functional requirement narrow down the selection of language, platform or implementation of techniques or tools. Some of the non-functional requirements are:

##### **1) Reliability**

Reliability is the extent to which a system can be expected to perform its intended function with required precision and accuracy. Thus, the system should be reliable in performing its daily functions and operations. The system will not produce dangerous or costly failures when it is used in a reasonable manner. All the possible failures and errors, which could occur, are taken into consideration. Besides that, to ensure no lost of data during system failure, there is a backup facility to retrieve back all the data in the database.

## **2) Robustness**

Robustness refers to the quality of the system that is able to handle or at least avoid disaster when faced with unexpected circumstances such as when given improper data. This system support robustness by developing program logic to process anticipated errors in the input of the data.

## **3) Efficiency**

Efficiency means that the processes or procedures can be called or accessed in an unlimited number of times and will produce similar outcomes or output. There is no cause of delay in processing the user's request or in the midst of retrieving information.

## **4) Modularity**

Modularity means breaking down a large or complex system into a smaller or more manageable program modules. Modularity is applied from the beginning as this will lead to easy modification hence enhance maintainability.

## **5) User-friendly**

The graphical user interface (GUI) should be easy-to-use, clear, concise and not cluttered. It is designed in a way that follows the logical flow of the thinking. It displays confirmation messages to assure the user of what has been done and also prompts warning or error messages to help users to correct the mistakes and proceed with the operations. All these are able to guide user better in operating the system.

## **6) Response time**

The system is able to retrieve the information at a reasonable range of time.

### **7) Short Learning Curve**

This system should be kept simple to ensure the short learning curve for all the users of the system so that it is always possible that new users can continue to be added and each of those users should be able to learn easily the fundamentals of the applications.

### **8) Scalability**

Scalability means the ability of the system to scale hardware and software to more users and more demanding workloads. It is the ability to increase or decrease size or capability in cost-effective increments with minimal impact on the unit cost of business and the procurement of additional services. [U18]

### **9) Unity of Environment**

To ensure a single working environment but not a mixing of distinctly separate unit is provided by this subsystem, the interface design and method for function invocation must be standardized.

## **4.2 Hardware And Software Requirements**

### **4.2.1 Hardware Requirements**

The hardware requirements for developing this subsystem are listed as below:

- Personal computer with Celeron 466Mhz processor or above
- 64MB of RAM
- 20GB of hard disk space
- Other standard desktop PC compliance

### **4.2.2 Software Requirements**

After analyzing all the available tools and technologies in Chapter 2, the appropriate tools and technologies have been chosen for the development of this Online Shopping Administration Subsystem.

#### **4.2.2.1 Programming languages**

ASP, VBScript and JavaScript are chosen to build this subsystem. ASP is chosen because of the shortcomings of HTML, which unable to make the web pages truly dynamic. VBScript is de facto language for ASP and will be used as the server-side scripting. JavaScript will be used as a client-side scripting so that this subsystem will be compatible on both Internet explorer and Netscape Navigator web browser.

#### **4.2.2.2 Web Development Tools**

Macromedia Dreamweaver MX is chosen simply because it provides an ideal environment for developers to visually construct sophisticated HTML and ASP pages. Also it is easy to use.

#### **4.2.2.3 Development Platform and Web Server**

Windows 98 SE is chosen to be the development platform and the Personal Web Server 4.0 (PWS 4.0), which is released together with Windows 98, will be the system's web server. PWS enables the ASP scripts to process the user instructions from the database.

#### **4.2.2.4 Database Access Standard Interface and Database Access Libraries**

The ODBC driver will be used to access the database because Online Shopping System is using relational database. Also ADO is considered the best choice because it provides most of the critical features that are needed in this subsystem.

#### **4.2.2.5 Database Management System (DBMS)**

The Microsoft Access 2000 will be used as the subsystem's database to store all the information and data. Microsoft Access 2000 is chosen because of its Upsizing Wizard that is able to take advantage of the Microsoft SQL Server 7.0

functionality wherever possible. This tool allows developers who design client-server applications on their desktop in Access to generate a SQL Server database from their prototype. Besides that, this subsystem is still on its development stage where it only involves a small number of data and simultaneous users. If this system were to expand, Microsoft Access 2000 is also able to cater for it because it could take the advantage of Microsoft SQL Server 7.0. Moreover, Microsoft Access 2000 is also easy to install and use.

#### **4.2.2.6 Others**

Besides that, some other softwares are used throughout the project. They are listed as below:

- Microsoft Word 2000 (for documentation)
- Microsoft Project (for drawing gantt cart)
- Microsoft Powerpoint (for VIVA presentation)
- Microsoft Internet Explorer 5.0 (as web browser)
- Microsoft Visio (for drawing DFD)
- Adobe Photoshop (for interface design)

# **CHAPTER 5:**

## **SYSTEM DESIGN**

### **5.1 Overview**

System design is the creative process of transforming the problem into a solution. It is used to design and implement improvements in the functioning of business that can be accomplished through the use of computerized information system. We use the requirements specifications to define the problem. Then we declare something to be a solution if it satisfies all the requirements in the specifications. In many cases, the number of possible solutions is limitless.

### **5.2 System Architecture**

There are 2 types of system architecture that the system can adapt, the two-tier architecture or the three-tier architecture. The system architecture that this Online Shopping Administration Subsystem is going to apply is the three-tier architecture.

#### **5.2.1 Two-Tier Architecture Vs Three-Tier Architecture**

A two-tier architecture is where a client talks directly to a server, with no intervening server. It is typically used in small environments (less than 50 users). [U19] Two-tier applications perform many of the functions of standalone systems. They present a user interface, gather and process user input, perform the requested processing, and report the status of the request. This sequence of commands can be repeated as many times as necessary. Because servers provide only access to the data, the client uses its local resources to perform most of the processing. The client application must contain information about where the data resides and how it is

organized in the database. Once the data has been retrieved, the client is responsible for formatting and displaying it to the user. [U20]

Today's client/server application is the multi-tier application, also known as n-tier architecture. A three-tier architecture introduces a server (or an "agent") between the client and the server. The role of the agent is many folds. It can provide translation services (as in adapting a legacy application on a mainframe to a client/server environment), metering services (as in acting as a transaction monitor to limit the number of simultaneous requests to a given server), or intelligent agent services (as in mapping a request to a number of different servers, collating the results, and returning a single response to the client. [U19] Most systems will perform the following three main tasks, which correspond to three tiers of the n-tier model (shown in Table 5.1).

**Table 5.1 The 3-tier Architecture [U20]**

Task	Description
<b>User interface and navigation</b>	Tier 1, where this layer comprises the entire user experience. Not only does this layer provide a graphical interface so that users can interact with the application, input data, and view the results of requests, it also manages the manipulation and formatting of data once the client receives it. In Web applications, the browser performs the tasks of this layer.
<b>Business logic</b>	Tier 2, between the interface and data services layers, is the domain of the distributed application developer. Business logic, which captures the rules that govern application processing, connects the user at one end with the data at the other. The functions that the rules govern closely mimic everyday business tasks, and can be a single task or a series of tasks.
<b>Data services</b>	Tier 3 where data services are provided by a structured (SQL, Oracle database) or unstructured (Microsoft Exchange, Message Queuing) data store, which manages and provides access to the application data. A single application may enlist the services of one or more data stores.

The three-tier architecture isolates each major piece of functionality, so that the presentation is independent of the processing rules and business logic, which in turn is separate from the data. This model requires much more analysis and design up front, but greatly reduces maintenance costs and increases functional flexibility in

the long run. The following diagram depicts the Microsoft technologies that service the various tiers in the new system design. [U20]

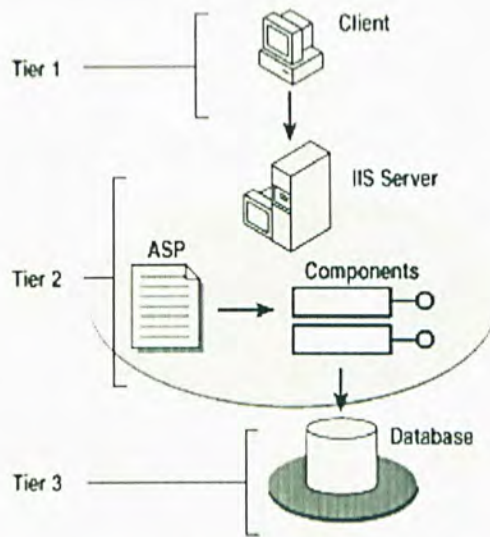


Figure 5.1 The 3-tier Architecture [U20]

### 5.2.2 Advantages Of 3-Tier Architecture

As previously mentioned 3-tier architecture solves a number of problems that are inherent to 2-tier architectures. Naturally it also causes new problems, but these are outweighed by the advantages.

- Clear separation of user-interface-control and data presentation from application-logic. Through this separation more clients are able to have access to a wide variety of server applications. The two main advantages for client-applications are clear: quicker development through the reuse of pre-built business-logic components and a shorter test phase, because the server-components have already been tested.
- Re-definition of the storage strategy won't influence the clients. RDBMS' offer a certain independence from storage details for the clients. However,

cases like changing table attributes make it necessary to adapt the client's application. In the future, even radical changes, like let's say switching from an RDBMS to an OODBS, won't influence the client. In well-designed systems, the client still accesses data over a stable and well-designed interface, which encapsulates all the storage details.

- Business-objects and data storage should be brought as close together as possible, ideally they should be together physically on the same server. This way - especially with complex accesses - network load is eliminated. The client only receives the results of a calculation - through the business-object, of course.
- In contrast to the 2-tier model, where only data is accessible to the public, business-objects can place applications-logic or "services" on the net. As an example, an inventory number has a "test-digit", and the calculation of that digit can be made available on the server.
- As a rule servers are "trusted" systems. Their authorization is simpler than that of thousands of "untrusted" client-PCs. Data protection and security is simpler to obtain. Therefore it makes sense to run critical business processes, which work with security sensitive data, on the server.
- Dynamic load balancing: if bottlenecks in terms of performance occur, the server process can be moved to other servers at runtime.
- Change management: of course it's easy and faster to exchange a component on the server than to furnish numerous PCs with new program versions. To come back to our VAT example: it is quite easy to run the new version of a tax-object in such a way that the clients automatically work with the version from the exact date that it has to be run. It is, however, compulsory that

interfaces remain stable and that old client versions are still compatible. In addition such components require a high standard of quality control. This is because low quality components can, at worst, endanger the functions of a whole set of client applications. At best, they will still irritate the systems operator. [U21]

### 5.3 Database Design

Data storage is considered as the heart of an information system. Therefore it is important to design the database properly to ensure the data are handled efficiently and systematically. Some of the objectives in designing the data storage are:

- Data availability – data are available when needed
- Data integrity – data are accurate and consistent
- Efficient data storage
- Efficient updating and retrieval
- Purposeful data retrieval – data obtained must be useful for managing, planning, controlling or decision making

The database used in this Online Shopping System is a relational database model. The Microsoft Access 2000 is selected to develop the database for this system. The attributes of the database are shown in Table 5.2.

**Table 5.2 The attributes of the Online Shopping System**

Database name	osm.mdb
Data source name	osm
Type of database	Microsoft Access 2000 relational database
Number of tables (for the use of Online Shopping Administration Subsystem).	9

The following are the 9 tables in this Online Shopping Administration Subsystem database:

**Table 5.3 Table for storing customer personal information. Table's name is Customer.**

Column Name	Data Type	Field Size /Format	Description
User_id	Text	15	Customer's unique ID
Password	Text	15	Customer's password
First_name	Text	50	Customer's first name
Last_name	Text	50	Customer's last name
Age	Number	15	Customer's age
Gender	Text	6	Customer's gender
Income	Currency	-	Customer's income
Email	Text	50	Customer's email address
Add	Text	50	Customer's address
Add2	Text	50	Customer's address
City	Text	20	City name
State	Text	20	State name
Zipcode	Number	5	City zip code
Tel	Text	15	Customer's housephone number
Handphone	Text	15	Customer's handphone number
Industry	Text	50	Industry that customer involves in
Bought_Fashion	Yes/No	-	Bought any fashion before
Bought_Cosmetic	Yes/No	-	Bought any cosmetic before
Bought_Bag	Yes/No	-	Bought any bag before

Bought_Watch	Yes/No	-	Bought any watch before
Bought_Shoes	Yes/No	-	Bought any shoes before
Bought_Jewelry	Yes/No	-	Bought any jewelry before
Label_Buy_Fashion	Yes/No	-	Whether customer will buy fashion or not
Label_Buy_Cosmetic	Yes/No	-	Whether customer will buy cosmetic or not
Label_Buy_Bag	Yes/No	-	Whether customer will buy bag or not
Label_Buy_Watch	Yes/No	-	Whether customer will buy watch or not
Label_Buy_Shoes	Yes/No	-	Whether customer will buy shoes or not
Label_Buy_Jewelry	Yes/No	-	Whether customer will buy jewelry or not

**Table 5.4 Table for storing prepaid card information. Table's name is Prepaid.**

Column Name	Data Type	Field Size/Format	Description
No	AutoNumber	Long Integer	Numbering for prepaid card
Card_no	Text	12	Prepaid card pin no
Date	Date/Time	Short Date	Expiry date
Card_value	Currency	5	Credits in the card
Used	Yes/No	-	Indicate whether the card is used or no

**Table 5.5 Table for storing product information.**

**Table's name is Product.**

Column Name	Data Type	Field Size/Format	Description
Category	Text	15	Product's category
Product_id	Text	15	Product's unique ID
Product_name	Text	50	Product's name
Description	Text	100	Product's description
Price	Currency	-	Price of the product
Imageurl	Text	50	Indicate where the image for the product is stored
Stock	Number	Long integer	Quantity of product in the stock
Color	Text	50	Product's color
Size	Text	50	Product's size
Supplier_id	Text	15	Supplier ID for that product

**Table 5.6 Table for storing balance information.**

**Table's name is Balance.**

Column Name	Data Type	Field Size/Format	Description
User_id	Text	15	Customer's unique ID
Credit	Currency	-	Balance of the credit

**Table 5.7 Table for storing order information.**

**Table's name is Orderdetail.**

Column Name	Data Type	Field Size/Format	Description
Order_id	Text	5	Order's unique ID
Order_date	Date/Time	Short date	The date user make order
Order_time	Date/Time	Medium time	The time user make order
User_id	Text	15	Customer's unique ID
ShipAdd	Text	50	Address to receive the

			ordered product
ShipAdd2	Text	50	Address to receive the ordered product
ShipCity	Text	20	City name
ShipState	Text	30	State name
ShipZipCode	Number	Long Integer	Postcode
ShipPhone	Text	20	Customer's telephone number
Delivery	Yes/No	-	Indicate whether the ordered has been sent or not

**Table 5.8 Table for storing supplier information.**

**Table's name is Supplier**

Column Name	Data Type	Field Size/Format	Description
Supplier_id	Text	15	Supplier's unique ID
First_name	Text	50	Supplier's first name
Last_name	Text	50	Supplier's last name
Company	Text	100	Supplier's company name
Add	Text	50	Supplier's company address
Add2	Text	50	Supplier's company address
City	Text	20	City name
State	Text	20	State name
Zipcode	Number	20	City zip code
Phone	Text	20	Supplier's phone number
Fax	Text	20	Supplier's fax number
Email	Text	50	Supplier's email address
URL	Text	50	Supplier's company URL
Category	Text	50	Supplier's category
Remarks	Text	50	Remarks

**Table 5.9 Table for keeping user's login name and password where only the authorized user can access the system. Table's name is User**

Column Name	Data Type	Field Size/Format	Description
Name	Text	30	User's name
Code	Text	5	User's unique ID
Username	Text	15	User's username
Password	Text	15	User's password
User_level	Text	15	User access level

**Table 5.10 Table for storing customer's shopping cart details. Table's name is Cart**

Column Name	Data Type	Field Size/Format	Description
Order_id	Text	15	Order's ID
User_id	Text	15	Customer's ID
Product_id	Text	15	Product's ID
Description	Text	50	Product's Description
Quantity	Text	10	Product's quantity
Price	Currency	-	Product's price
Color	Text	50	Product's color
Size	Text	50	Product's size
Total	Currency	-	Total amount

**Table 5.11 Table for storing personalized web page details. Table's name is Association\_rules.**

Column Name	Data Type	Field Size/Format	Description
ID	Number	Byte	Fix value (for SQL purpose). Number "1" was used
Times	Number	Byte	The times user visits the site to generate

			personalized web page.
Min_support	Number	Byte	Minimum support used in association rule
Min_confidence	Number	Byte	Minimum confidence used in association rule.

## 5.4 Process Design


### 5.4.1 Data Flow Diagram (DFD)




A data flow diagram (DFD) is graphic representation of a system that uses a small number of symbol shapes to illustrate how data flows through interconnected processes [5]. DFD is perhaps the most natural way to document processes. It has 4 chief advantages over narrative explanation of the way data moves through the system. The advantages are:

- Freedom from committing to the technical implementation of the system.
- Further understanding of the interrelatedness of systems and subsystems.
- Communicating current system knowledge to users through data flow diagrams.
- Analysis of a proposed system to determine if the necessary data and processes have been defined.

There are four basic symbols used to chart the data movement on DFD. They are shown in Table 5.12.

**Table 5.12 Symbol of Data Flow Diagram**

Symbols	Meaning	Description
	Entity	Used to depict an external entity that can send data to or receive data from the system

	Flow of Data	Movement of data from one point to another, with the head pointing towards the data destination
	Process	Show occurrence of a transformation of data
	Data Store	Used to represent a data store where the data is held within the system

#### 5.4.1.1 Context Diagram

The context diagram of this Online Shopping Administration Subsystem is shown in Figure 5.2, which represents the overview of this subsystem at the highest possible level. As shown in the context diagram, this subsystem is mainly focus on the processing of customers' order, customers' information, suppliers' information, products' information and generating new information from existing customers' information and buying record. The entities are the users (administrators), suppliers, customers and warehouse.

#### 5.4.1.2 Functional Decomposition Diagram

A decomposition diagram shows the top-down functional decomposition or structure of a system. [6] The functional decomposition diagram for this subsystem is shown in Figure 5.3. The numbers in the diagram represent the meanings as following:

- 1) The root process of the entire Online Shopping System
- 2) The system is decomposed into two subsystems, which are Online Shopping Processing And Submission Subsystem and Online Shopping Administration Subsystem.

- 3) The Online Shopping Administration Subsystem is decomposed into 8 sub modules that are user management module (security module), customer record module, customer order module, supplier record module, product inventory module, product delivery and stock update module, prepaid card module and new information generator module.
- 4) All the modules are factored into specific transactions as following:

User Management Module	→ Process user's login and manage user account.
Customer Record Module	→ Process customer's information
Supplier Record Module	→ Process supplier's information
Customer Order Module	→ Process customer's order
Inventory Control Module	→ Process product's information and amount of products left.
Product Delivery And Stock Update Module	→ Update the amount of product
New Information Generator Module	→ Generate new information
Prepaid Card Module	→ Process prepaid card's information

#### 5.4.1.3 Event Diagram

The functional decomposition diagram is decomposed into 8 sub-events that are labeled as EV1 in Figure 5.4, EV2 in Figure 5.5, EV3 in Figure 5.6, EV4 in Figure 5.7, EV5 in Figure 5.8, EV6 in Figure 5.9, EV7 in Figure 5.10, EV8 in Figure 5.11 and EV9 in Figure 5.12.

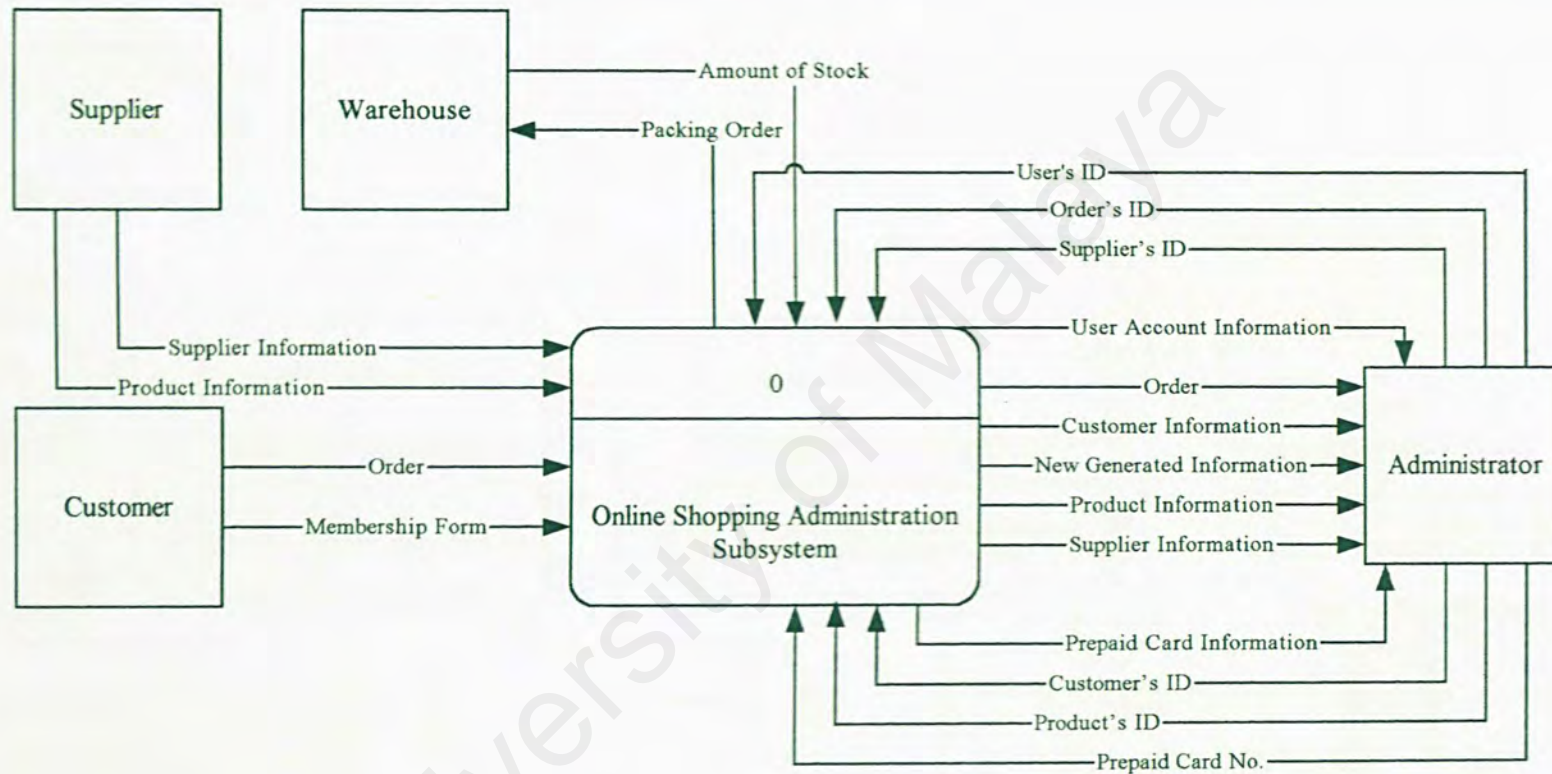
#### **5.4.1.4 System Diagram**

The event diagrams are very useful to validate the accuracy of each event that the subsystem must response to it. However these events cannot occur alone. Therefore a system diagram is constructed as shown in Figure 5.13 by combining all the events of the sub modules and demonstrates how event processes truly communicate with each other by sharing data stores.

#### **5.4.1.5 Primitive Diagram**

To reveal more details, all event processes in Online Shopping Administration Subsystem are exploded into a primitive data flow diagram as following:

- Figure 5.14 is the primitive diagram of Process User's Login Event.
- Figure 5.15 is the primitive diagram of Process Customer's Information Event.
- Figure 5.16 is the primitive diagram of Process Supplier's Information Event.
- Figure 5.17 is the primitive diagram of Process Customer's Order Event.
- Figure 5.18 is the primitive diagram of Process Product's Information And Amount Of Products Left Event.
- Figure 5.19 is the primitive diagram of Update The Amount Of Product Event.
- Figure 5.20 is the primitive diagram of Generate New Information Event.
- Figure 5.21 is the primitive diagram of Process Prepaid Card's Information Event.
- Figure 5.22 is the primitive diagram of Process User Account Event.



**Figure 5.2 Context diagram of Online Shopping Administration Subsystem**

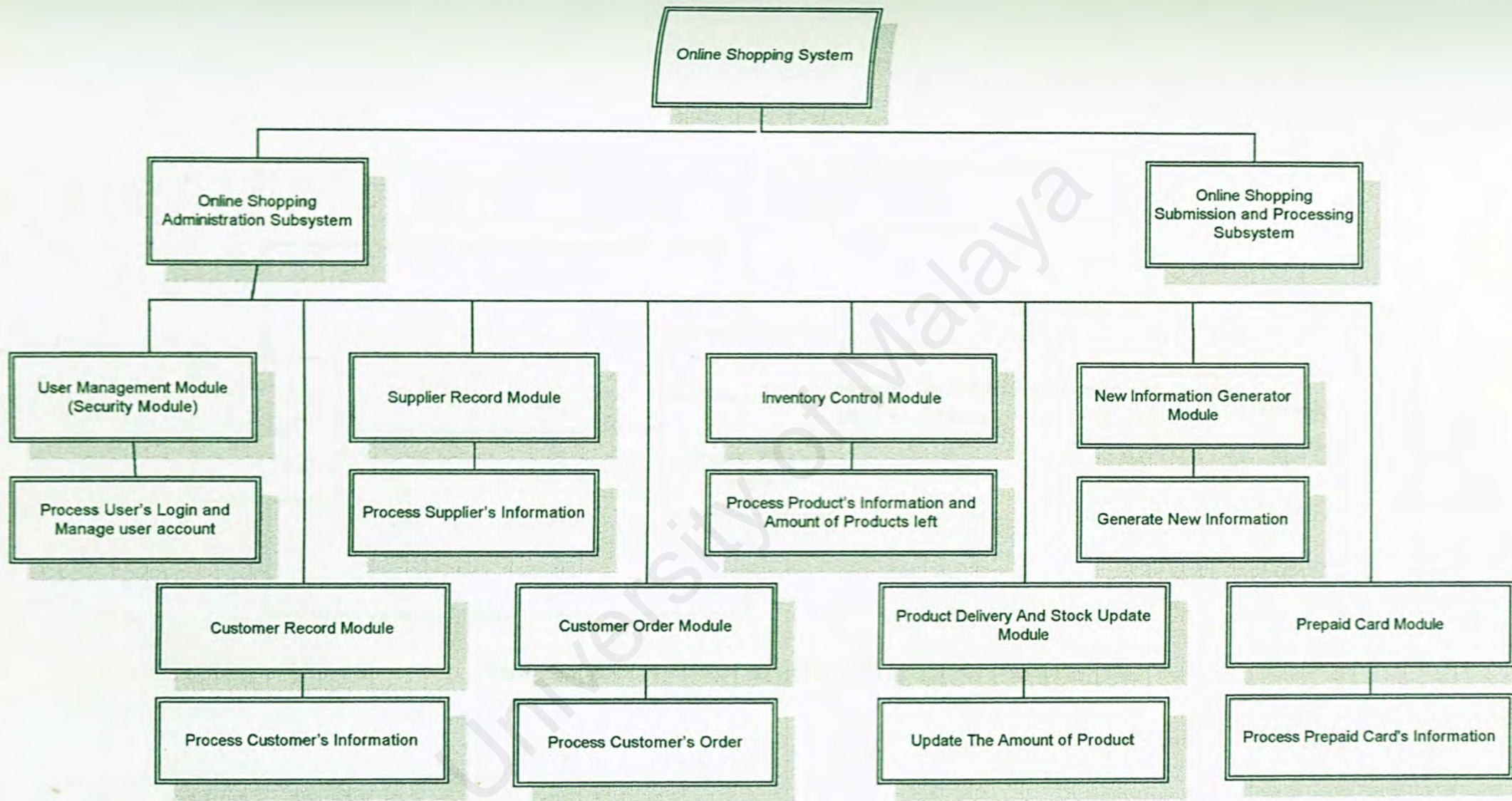
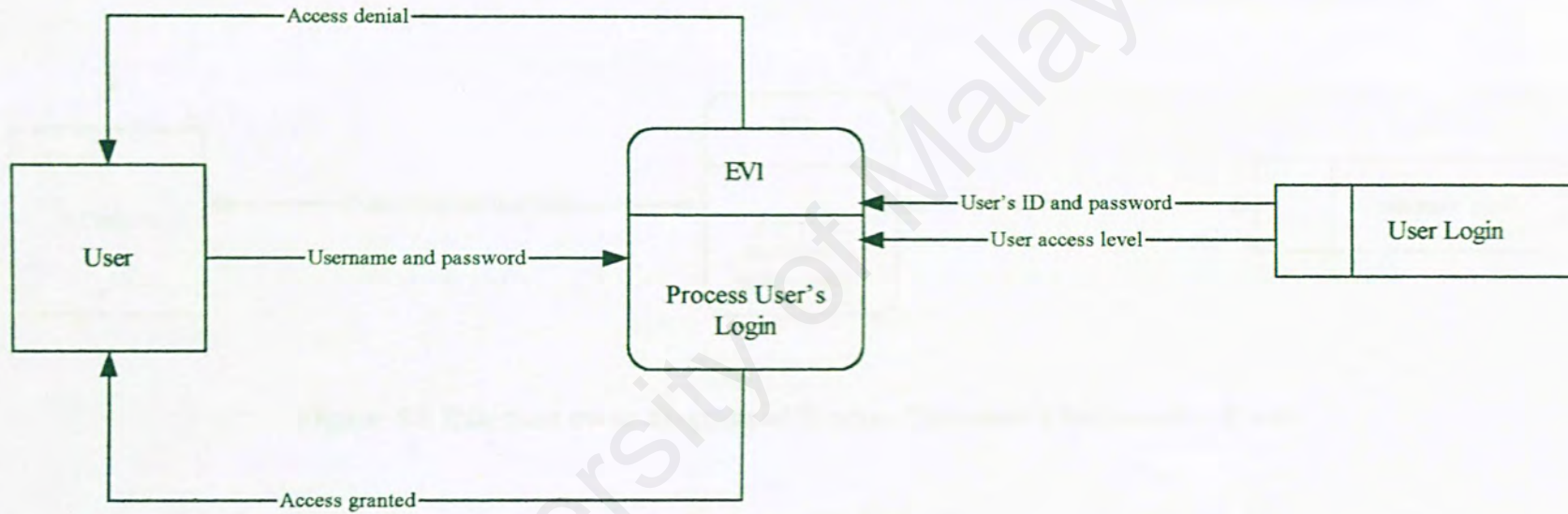
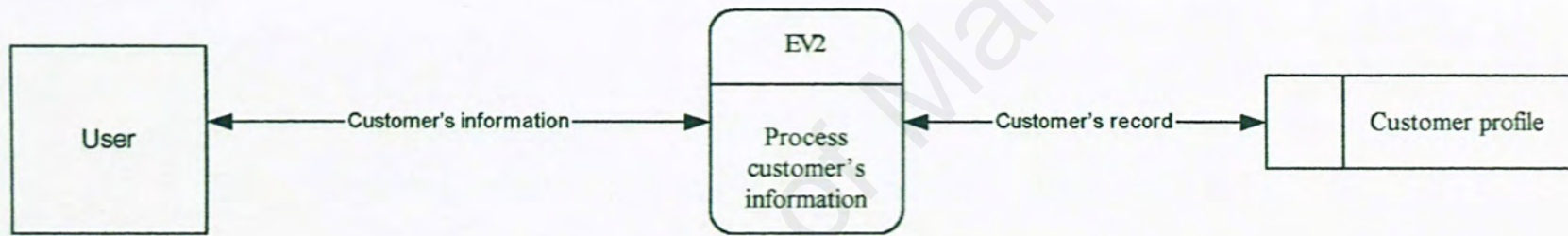


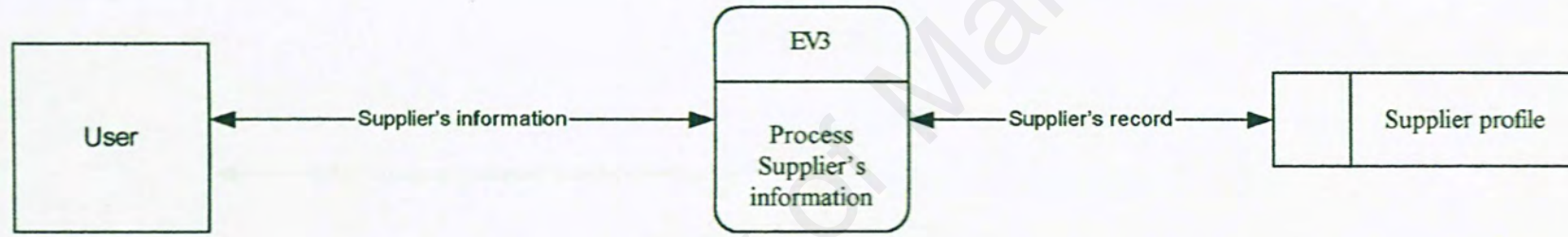
Figure 5.3 Functional decomposition diagram of Online Shopping System



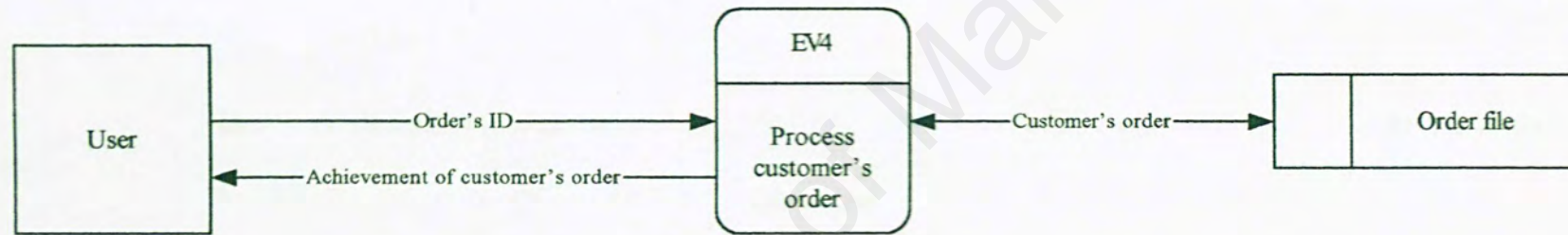
**Figure 5.4 External event diagram of Process User's Login Event**



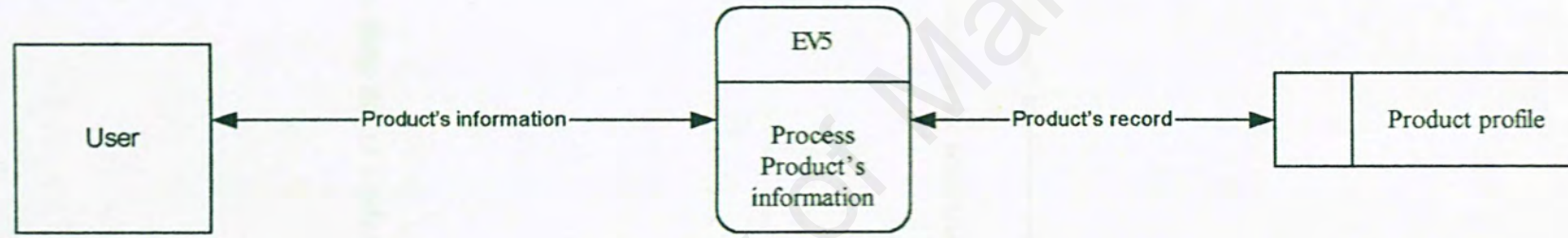
**Figure 5.5 External event diagram of Process Customer's Information Event**



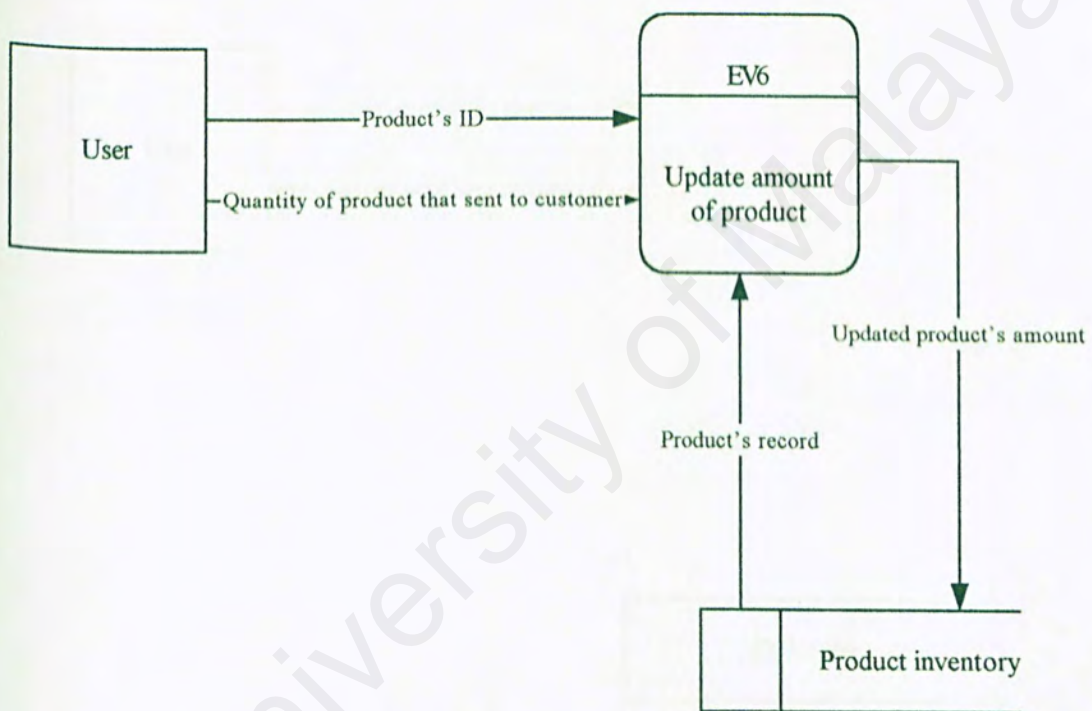
**Figure 5.6 External event diagram of Process Supplier's Information Event**



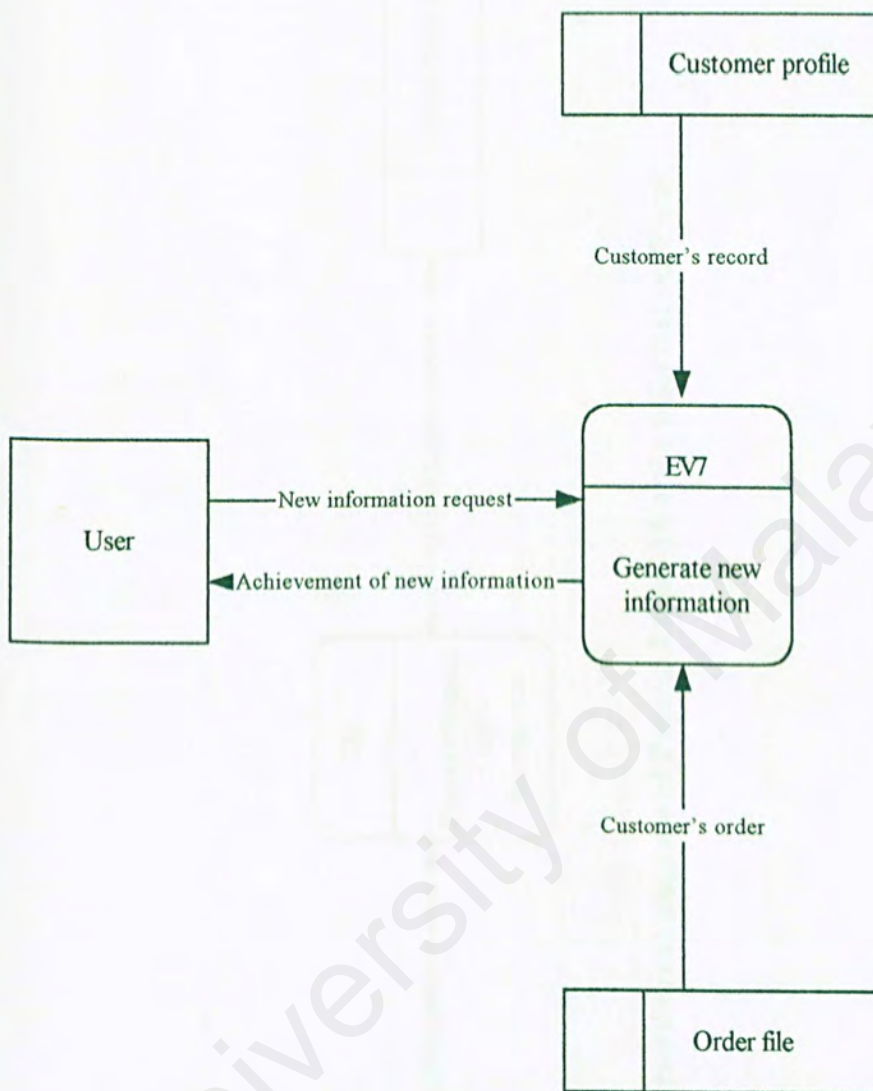
**Figure 5.7 External event diagram of Process Customer's Order Event**



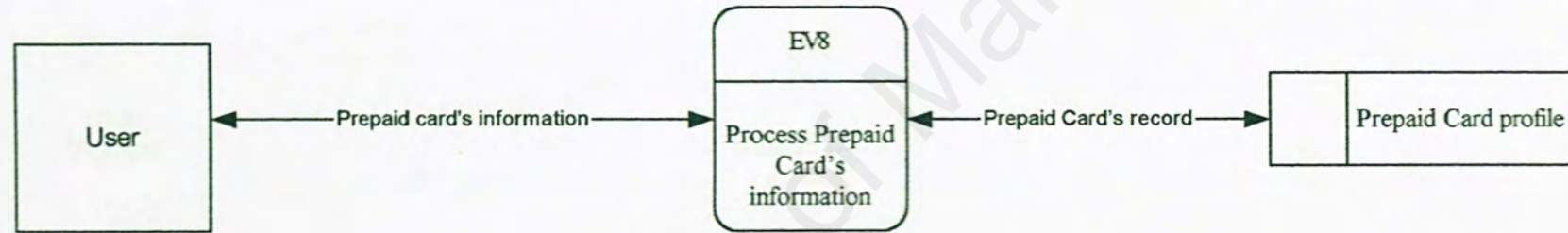
**Figure 5.8 External event diagram of Process Product's Information And Amount Of Product Left Event**



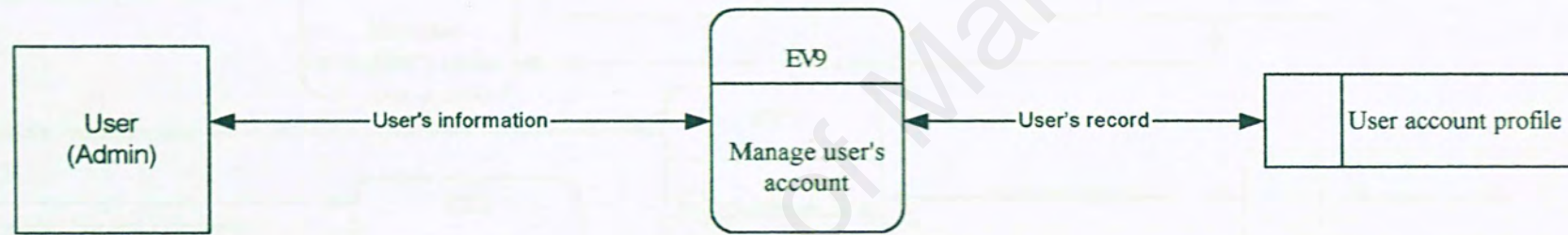
**Figure 5.9 External event diagram of Update The Amount Of Product Event**



**Figure 5.10 External event diagram of Generate New Information Event**



**Figure 5.11 External event diagram of Process Prepaid Card's Information Event**



**Figure 5.12 External event diagram of Manage User Account Event**

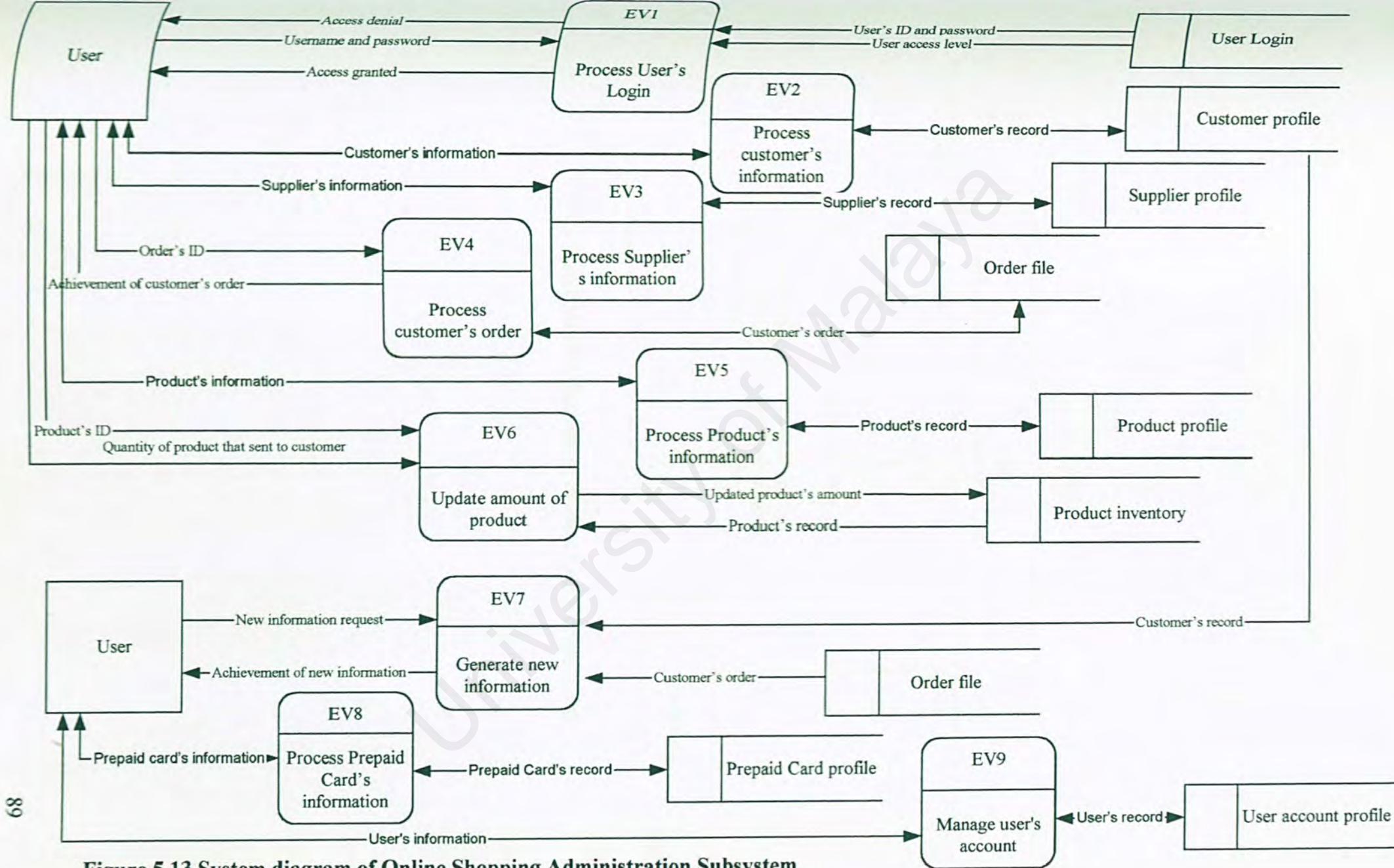
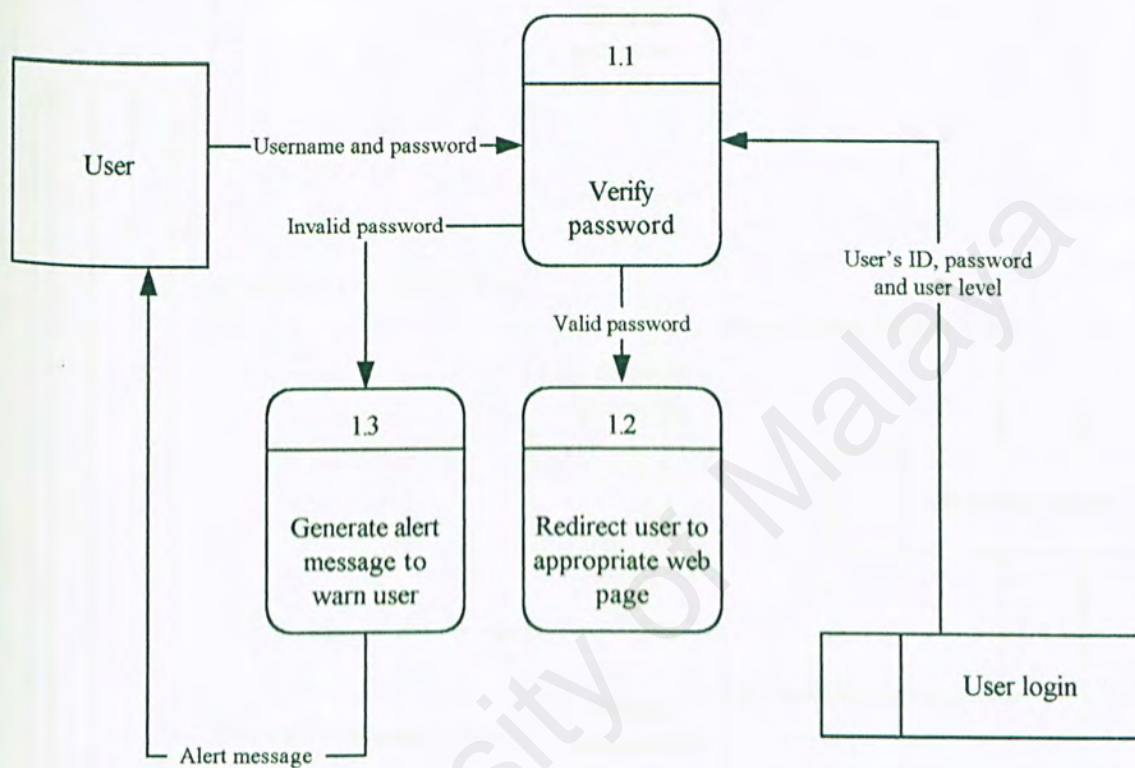
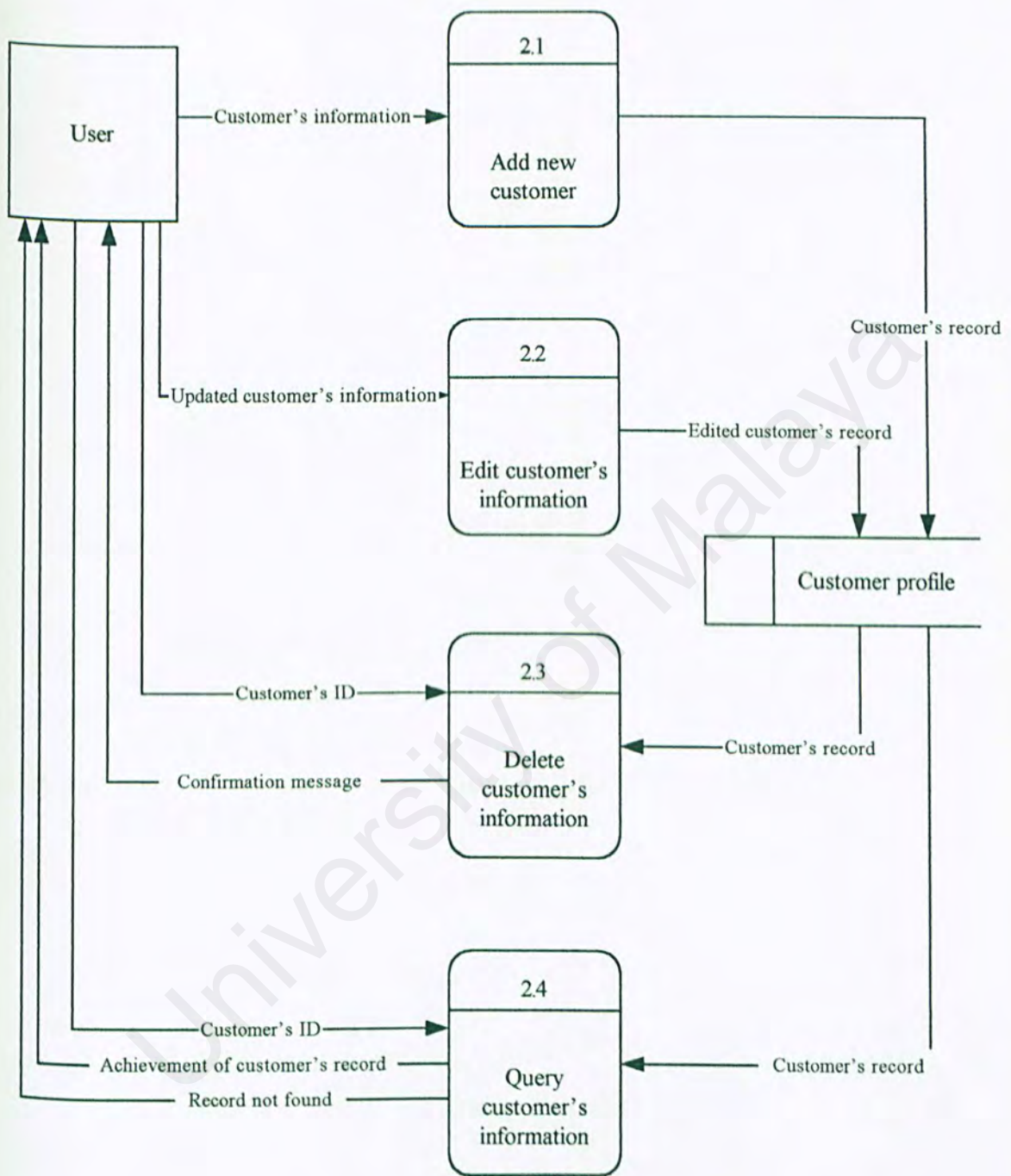


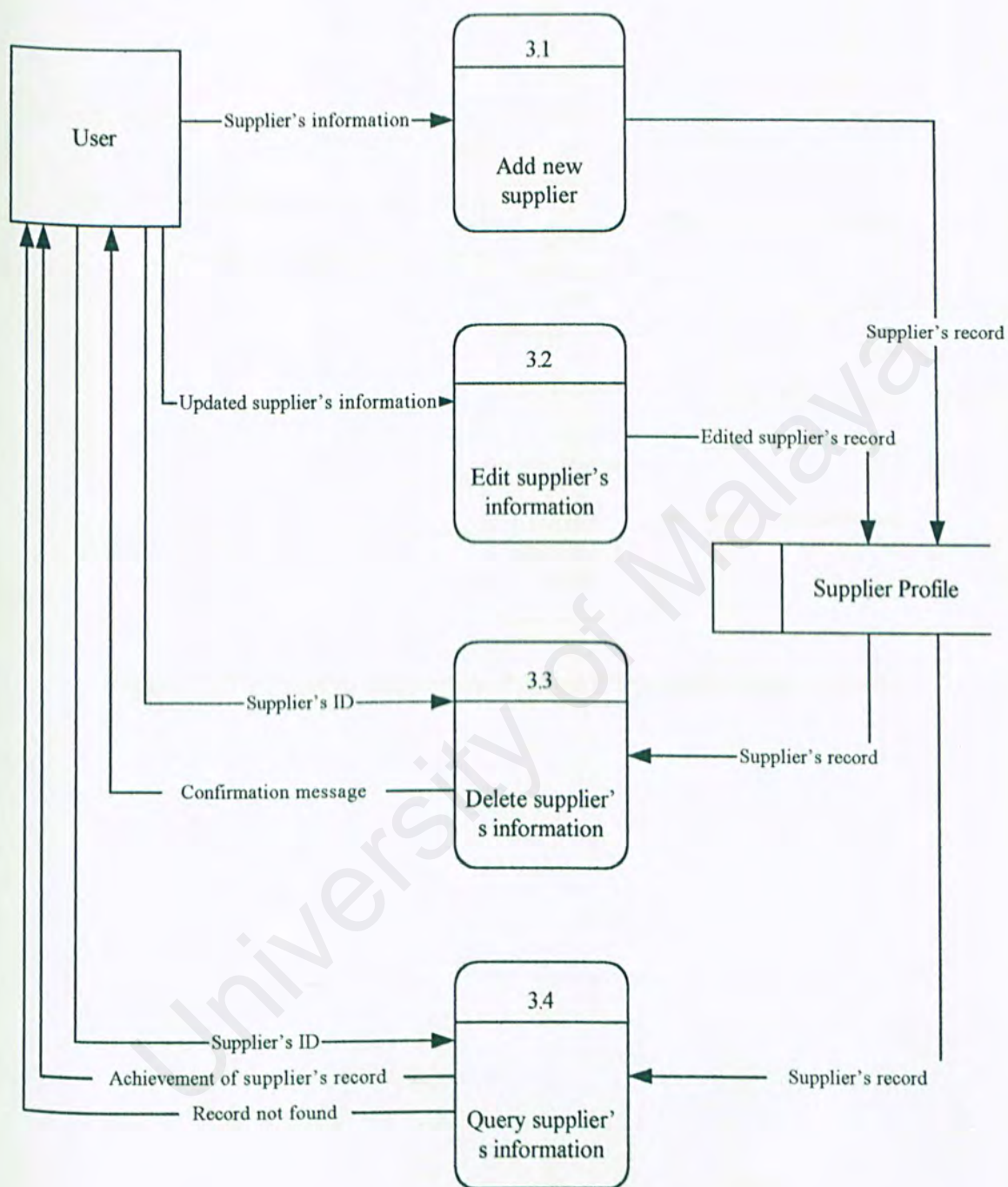
Figure 5.13 System diagram of Online Shopping Administration Subsystem



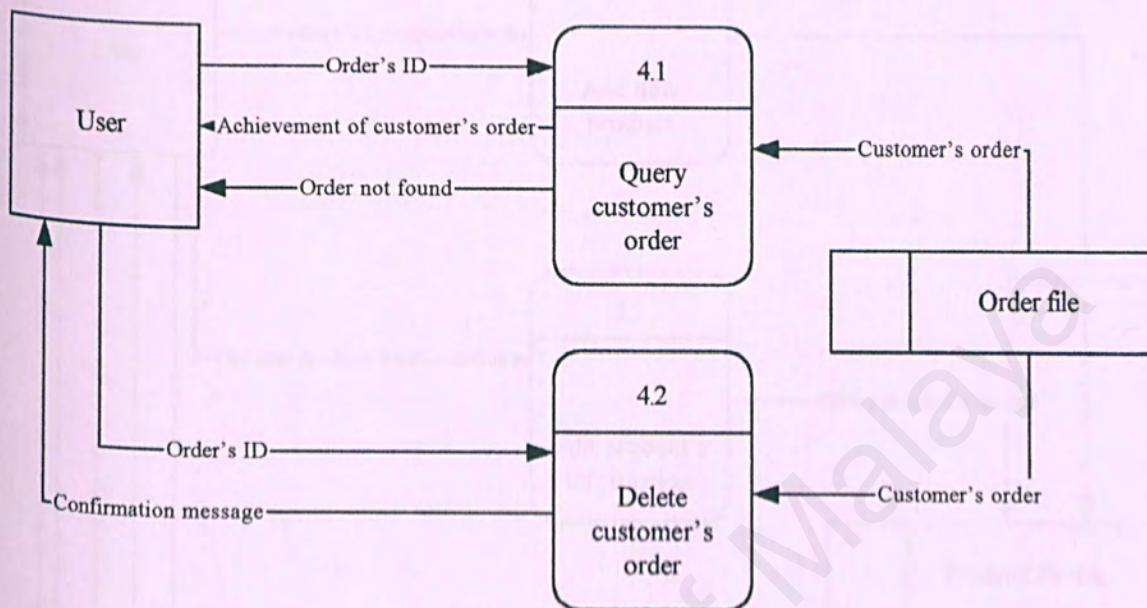
**Figure 5.14 Primitive diagram of Process User's Login Event**



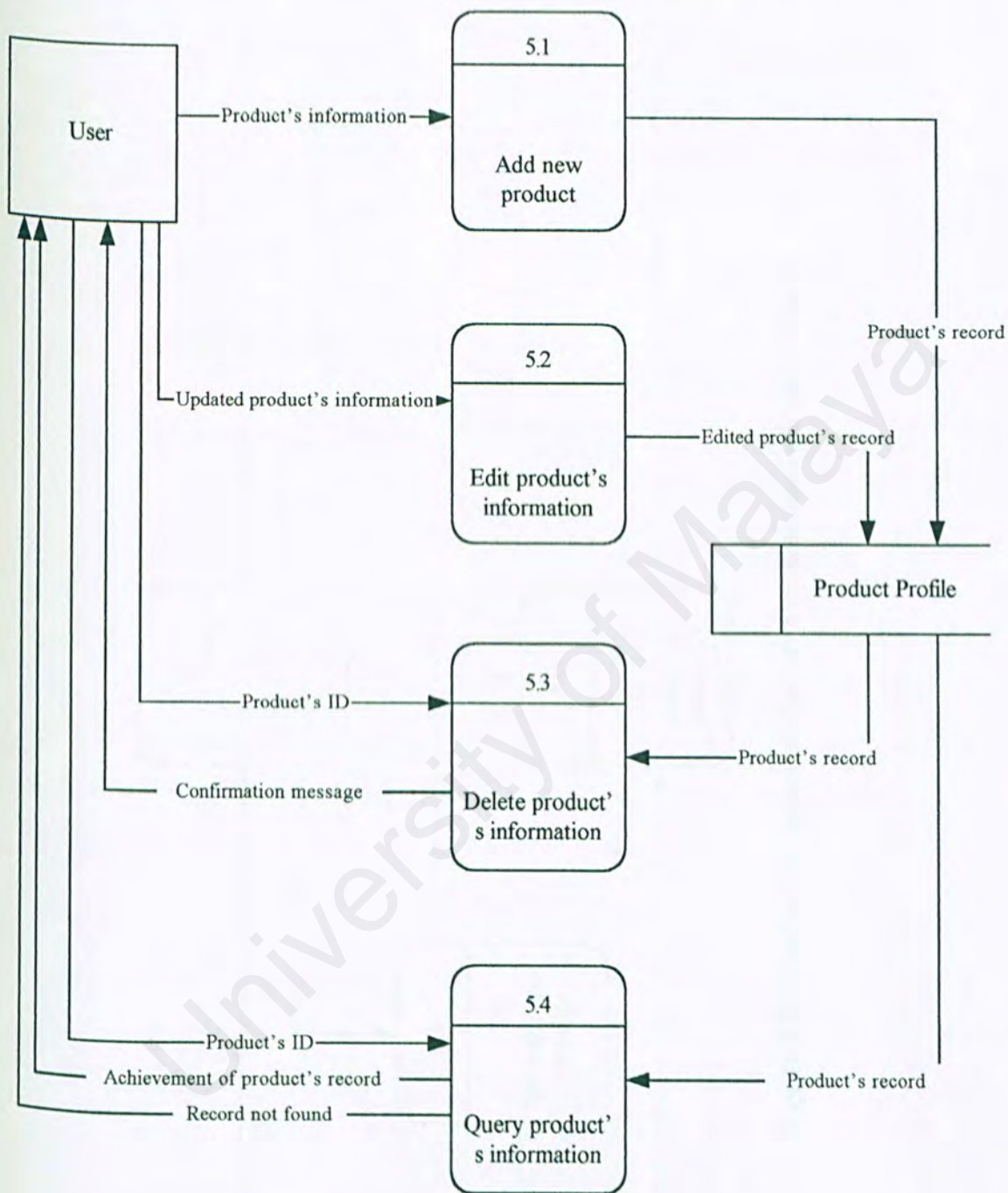
**Figure 5.15 Primitive diagram of Process Customer's Information Event**



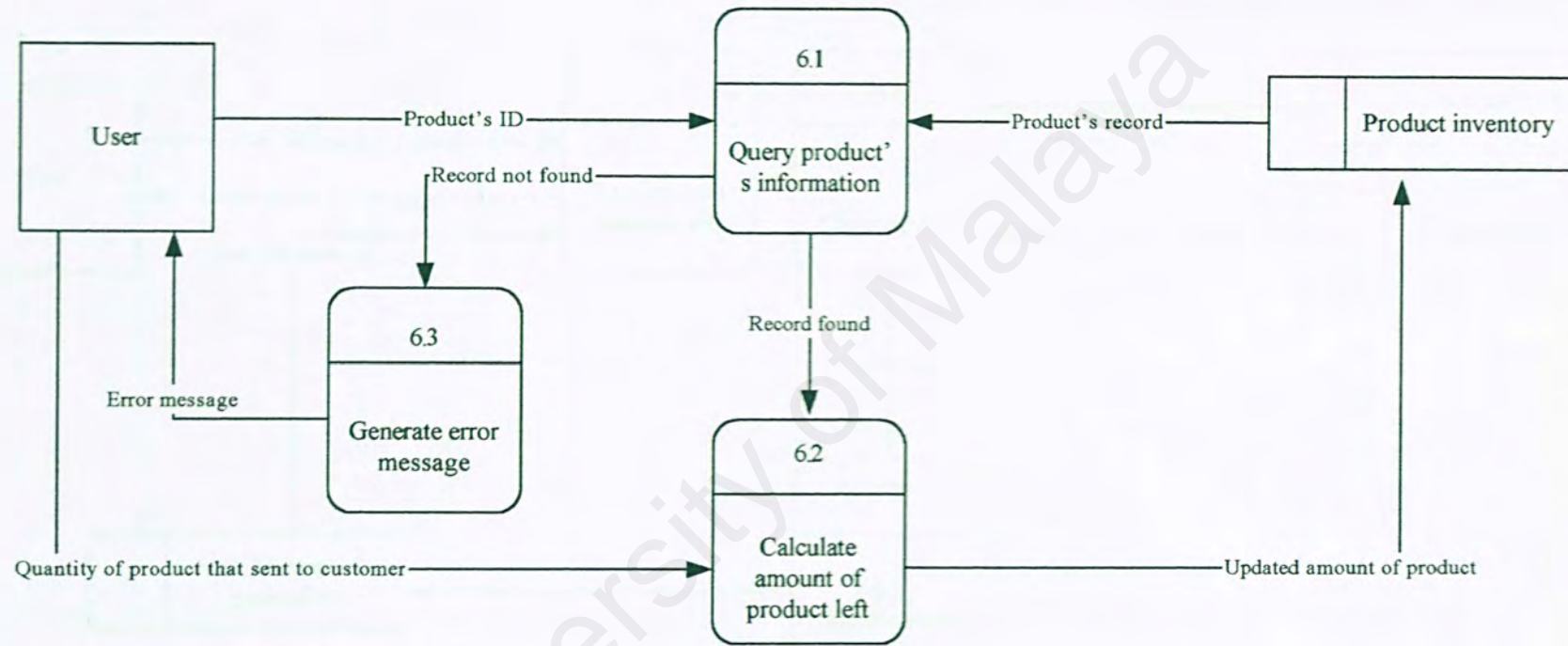
**Figure 5.16 Primitive diagram of Process Supplier's Information Event**



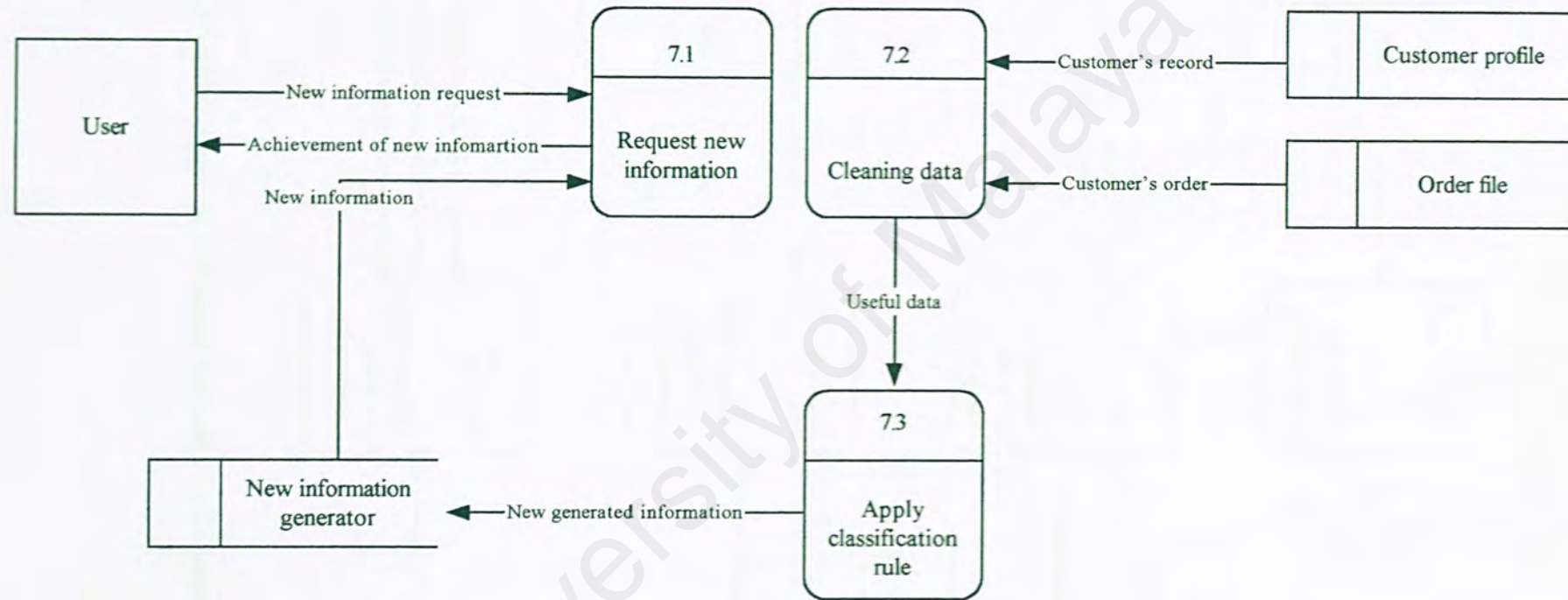
**Figure 5.17 Primitive diagram of Process Customer's Order Event**



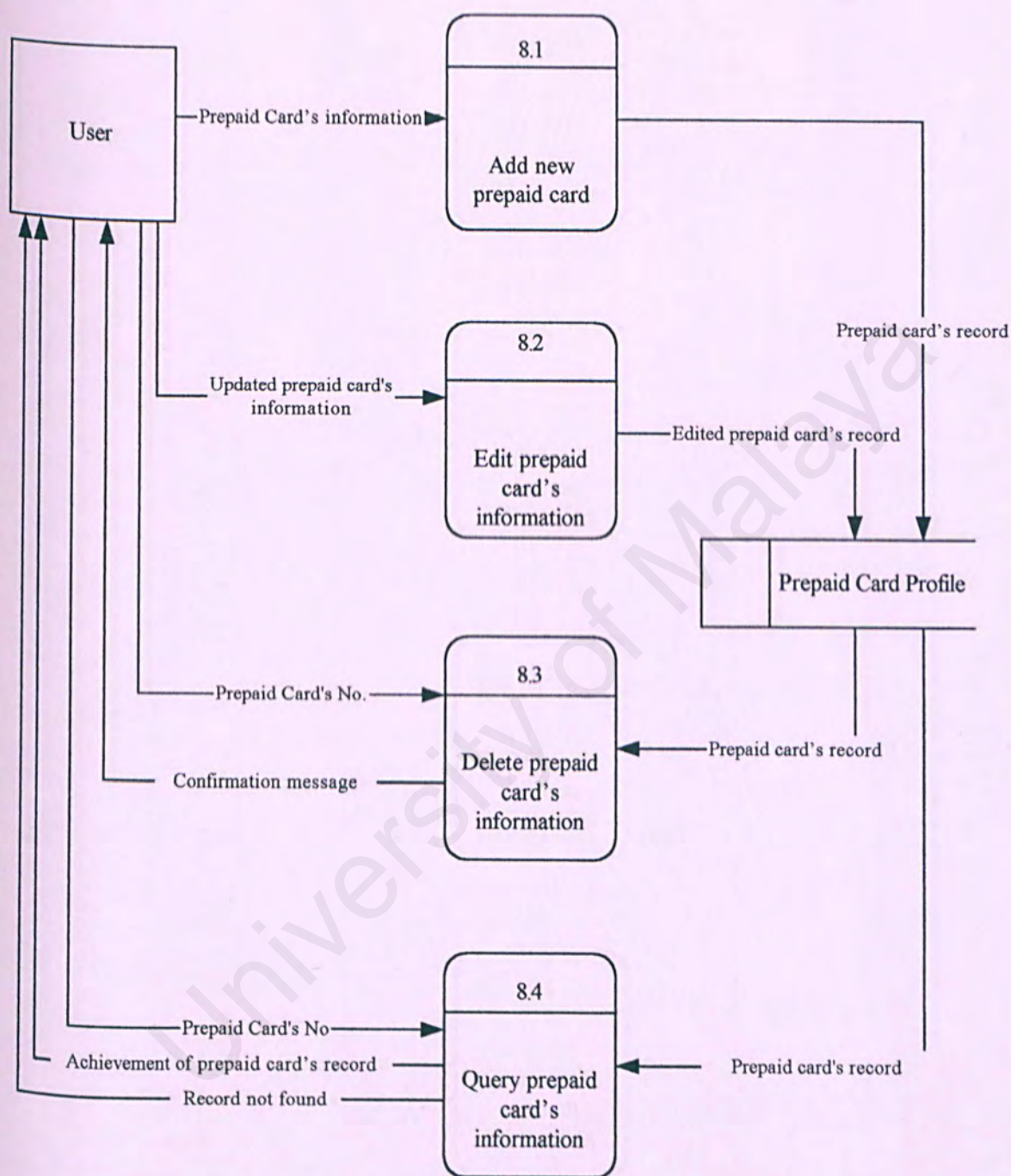
**Figure 5.18 Primitive diagram of Process Product's Information And Amount Of Product Left Event**



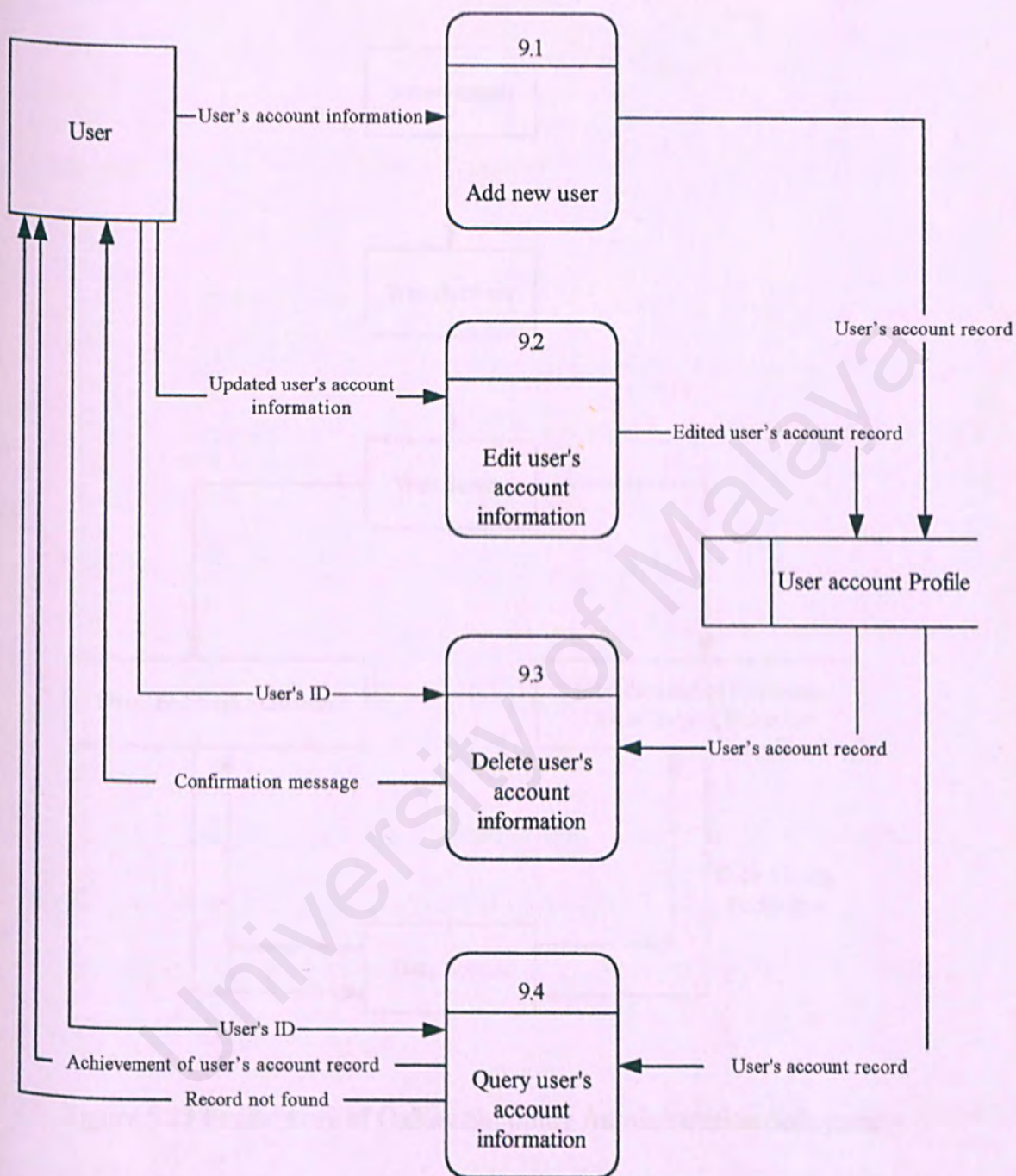
**Figure 5.19 Primitive diagram of Update The Amount Of Product Event**



**Figure 5.20 Primitive diagram of Generate New Information Event**



**Figure 5.21 Primitive diagram of Process Prepaid Card's Information Event**



**Figure 5.22 Primitive diagram of Manage User Account Event**

5.4.2 Framework

Below is the framework of the Online Shopping Administration Subsystem.

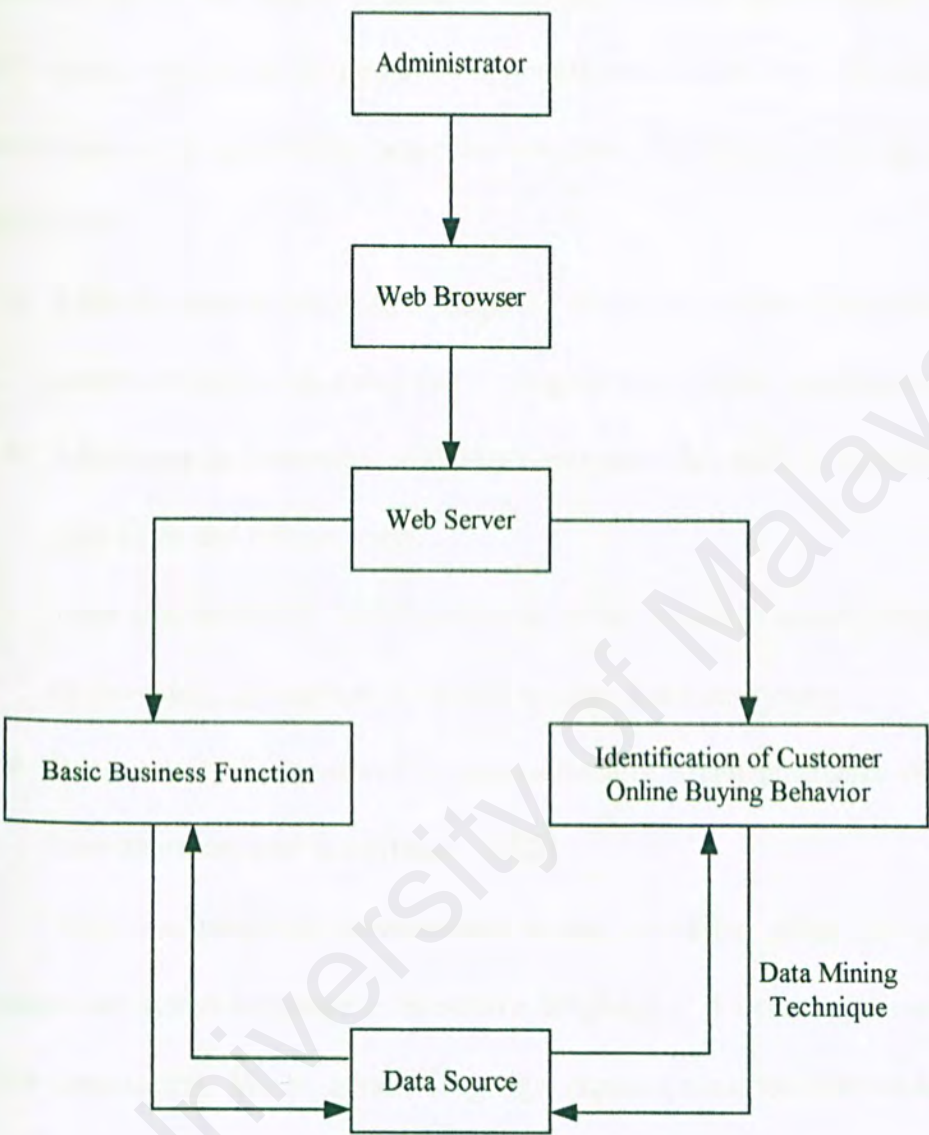


Figure 5.23 Framework of Online Shopping Administration Subsystem

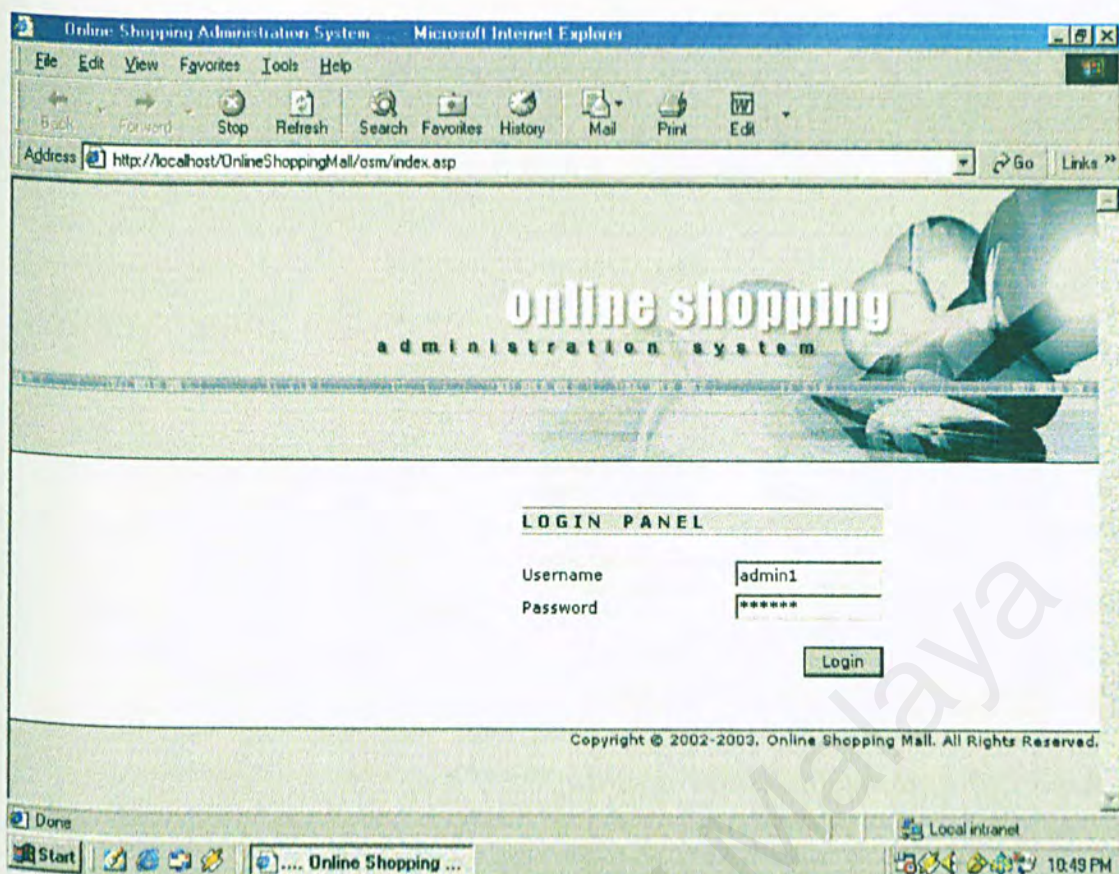
## 5.5 User Interface Design

For most users, the interface is the system. Interface is the representation of the system no matter the design is good or bad. An interface is intended to allow the user to access the system's internal components in an easier way. The easier it is for a user to access the system the better the interface. The objectives in designing user interface are:

- **Effectiveness** as achieved through the design of interfaces that allow users to access the system in a way that is congruent with their individual needs.
- **Efficiency** as demonstrated through interfaces that both increase the speed of data entry and reduce errors.
- **User consideration** as demonstrated in the design of suitable interfaces and by providing appropriate feedback to users from the system.
- **Productivity** as measured by ergonomically sound principles of design for user interfaces and workspaces. [U22]

There are two main components in user interface, which are presentation language and action language. Presentation language is the computer-to-human part of the transaction. While action language characterizes the human-to-computer portion. There are several different types of user interface, but the user interface for this Online Shopping Administration Subsystem is the Graphical User Interface (GUI), which allows direct manipulation of the graphical representation on the screen, which can be done with a keyboard input, a mouse or a joystick.

The user interfaces of the Online Shopping Administration Subsystem are shown in the following pages.



**Figure 5.24 Login page of Online Shopping Administration Subsystem.**

The first page of this Online Shopping Administration Subsystem is the login page as shown in Figure 5.24. This page only lets authorized user to enter the system, where user must key-in valid username and password.

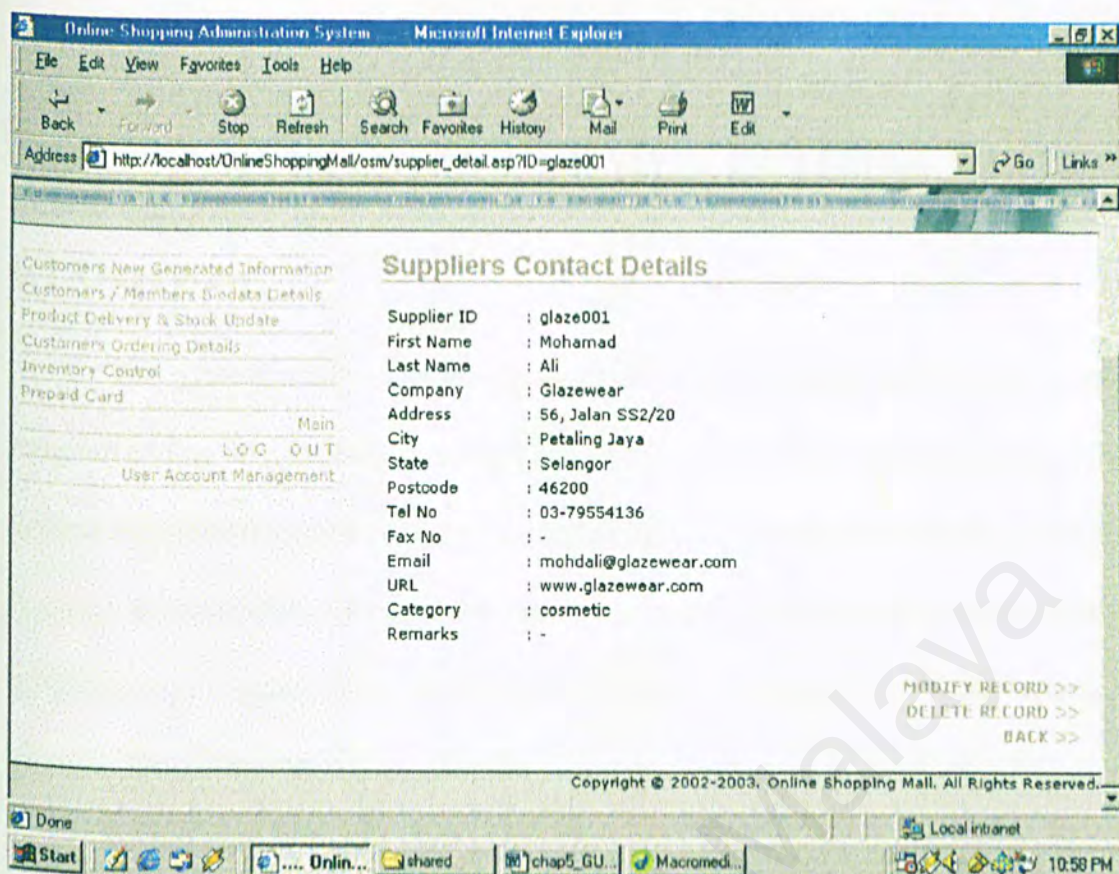


Figure 5.25 Supplier record module.

After user entered the system successfully, this page will be displayed. 8 modules are represented in 8 buttons, where user can click on it to select the intended module. For this supplier record module, the information of the selected supplier will be displayed and the function Modify and Delete button are provided.

# CHAPTER 6:

## SYSTEM IMPLEMENTATION

### 6.1 Overview

This system implementation involves the system development environment, program coding and database development. In this system implementation phase, the system requirements and design are converted into program code. This phase always involves some modifications to the previous design due to the limitations of the programming language used. There are a few tasks involved in the implementation phase of this subsystem.

- **Install new software**

All the required software packages (e.g. Macromedia Dreamweaver MX, PWS) are installed.

- **Build Online Shopping System Database**

Database was built according to the database schema specified during system design phase.

- **Code and test the programs**

Technical design statement, plan for programming and test data developed during system design phase are the primary input for this activity. Each module in this Online Shopping Administration Subsystem was developed separately and later integrated into a fully functional system once every module has been tested successfully. Finally the fully complete Online Shopping System is developed (after the system integration phase).

## 6.2 Coding Style

The following coding styles are used throughout the development of the system:

- **Maintainability** – codes are well organized. Task-oriented codes are centrally located. For example, a separate include file that contain procedures that can be used to open and close database connection was built to set up the connection to Online Shopping System database (let's say the file's name is connection.asp). Whenever changes are made upon the database connection such as change of database's name, only that single file (connection.asp) is required to make changes.
- **Reusability** – task-oriented codes are easily developed for reuse, especially for tasks that cross application boundaries. For example, the coding to validate the user input is written in function form. Whenever validation is required, that particular function is called instead of written the same coding again and again.
- **Testability** – modules can be tested easily. Modularization breaks up the code coverage task into smaller and manageable units. For example, each module consists of separate different asp file. Each separate file can be run, test and debug easily.
- **Speed** – module code can safely be optimized without affecting the calling procedures.

Online Shopping Administration Subsystem was developed using an architecture approach, which consists of 3 different layers: user interface layer, data layer and transaction layer.

### 6.2.1 User Interface Layer

The User Interface layer is the only portion of the application that is responsive to user interaction. The User Interface layer is there when all the data is presented to the user and when all inputs or modifications of data are made via window objects. The User Interface layer includes all event-handlers or event. They are subroutines that are called upon in response to some user interaction (click, mouse over) or during the change in status of a window object (form resize, lost focus). It also includes those procedures that either fill controls with data or retrieve data from controls. In short, User Interface layer perform the following:

- Displaying all applications data or information via window objects
- Responding to the changing states of window objects
- Initiating all user requests

### 6.2.2 Data Layer

The Data layer supplies all data to the User Interface layer for display in window objects and all the data to the Transaction layer. The Data layer includes any routine that will perform operations on an application data. Its responsibilities should include operations such as:

- Validating the data in the structure
- Formatting of data for display in the User Interface layer
- Formatting of data for the use in the Transaction layer
- Sorting the data in arrays or structures

### 6.2.3 Transaction Layer

The Transaction layer uses the Data layer as its application data repository and transfers or manipulates all data to and from an external data source. The Transaction layer includes operations such as the following:

- Initiation / building of all queries
- Handling violation of external database
- Controlling all transactional logic

## 6.3 Program Optimization

Program optimization is a process of improving the efficiency of the system. The speed at which information appears on the screen often gives the user an impression on how well the program will perform. There are 2 ways in optimizing the program:

### 1) Increase the execution speed of the program

- Avoid using variant data types because the variant data requires additional internal program standards to identify the information being stored.

### 2) Decrease the program size

- Reviewing codes for unused variants, constants and “dead-code” and remove it from the program codes.
- Assigning the string variables to zero-length string if it is no longer needed.

## 6.4 Programming Documentation

### 6.4.1 Internal Documentation

Internal comments are used in providing guidance during the maintenance phase of the system. It is a way to let programmer communicate with the readers.

Comments are used to indicate the purpose for the function of the module and to describe processing function.

**Header Comment Block** is placed at the beginning of each component in a set of components. The following information is included:

- 1) The name of the component
- 2) The name of the creator
- 3) The date the component is written and revised
- 4) The purpose of the component.

**Program Comments** is placed within the code to provide additional information to let the readers understand how the intention is implemented in the code.

**Meaningful Variables Names and Statement Labels** are used where the names for the variables and statements can reflect their use or meaning.

**Formatting to Enhance Understanding.** Indentation and spacing of statements can reflect the basic control structure

#### **6.4.2 External Documentation**

External documentation is intended to be read by the one who may never look at the code. External documentation is the full blown of the thesis report

## **CHAPTER 7:**

### **SYSTEM TESTING**

#### **7.1 Overview**

There are a few modules in Online Shopping Administration Subsystem. Each module is coded and tested separately. After that, these modules have to be integrated and tested in a whole system.

The tester is not the system designer himself. The tester has to ensure each module is running smoothly and each function is performed perfectly. Therefore asking the tester to try out the system will test the usability of the user interface, whether the interface is self-explanatory or not, or whether the tester know what should be the steps taken to run the system. If the tester felt messy and complicated while testing out the system, the user interface should be revised and improved. Advice that is asked from the tester is to improve the usability of the interface. Sometimes the misuse of the wordings or language in the system might mislead the users.

#### **7.2 The Testing Process**

Except for small programs, systems should not be tested as a single, monolithic unit. Large systems are built out of sub-systems, which are built out of modules, which are composed of procedures and functions. The testing process should therefore proceed in stages where testing is carried out incrementally in conjunction with system implementation. The most widely used testing process consists of five stages:

### 7.2.1 Unit Testing

This is a small unit testing where testing are done on individual components of the system to ensure that they operate correctly. Each function is tested independently, without other system components. The levels, which are tested, are basically at the field level/form level. In this system, Online Shopping Administration Subsystem is tested by form or document submission and input validation.

#### ➤ **Form / Document Submission**

The feedback message will show whether the document is submitted successfully. Another way to verify is by checking the database and see whether the data are already inserted in the database or not.

#### ➤ **Input Validation**

Input validation is to test the inserted data whether they are valid and according to the specified criteria. Each condition of the criteria is entered into the form/field and tested. If the input is incorrect, a pop-up message will appear to warn the user.

### 7.2.2 Module Testing

A module is a collection of dependent components such as an object class, an abstract data type or some looser collection of procedures and functions. The main objective of doing module testing is to test the interfacing and integration between the tested units that form the module and to test the integration between the modules that form the system.

The modules are tested with some dummy data. If an error occurs, the related error to that unit is checked and then modified. Then, these units are integrated until no

error occurs on the integration module. Sometimes the error is due to the integration coding of the units. If this is the case, the integration code has to be checked in all related units. Another possibility of the error occurring in module integration is misspelling the name of the database files.

### **7.2.3 Subsystem Testing**

This phase involves testing collections of modules, which have been integrated into sub-systems. Sub-systems may be independently designed and implemented. The most common problems that arise in large software are sub-systems interface mismatches. The sub-system test should therefore concentrate on the detection of interface errors by rigorously exercising these interfaces.

### **7.2.4 System Testing**

The sub-systems are integrated to make up the entire system. The testing process is concerned with finding errors, which result from anticipated interactions between sub-systems and system components. It is also concerned with validating that the system fulfills the functional and non-functional requirements. System testing can be categorized into a few types:

#### **1) Recovery testing**

Recovery testing is a system test that forces the software to fail in a variety of ways and verifies that recovery is properly performed. If recovery is automatic (performed by the system itself), reinitialization, check pointing mechanisms, data recovery and restart are evaluated for correctness. If recovery requires human intervention, the mean-time-to-repair (MTTR) is evaluated to determine whether it is within acceptable limits.

## 2) Security testing

Security attempts to verify that protection mechanisms built into a system will protect it from improper penetration. During security testing, the tester plays the role(s) of the individual who desires to penetrate the system. The tester will type in the file address (for example, [http://localhost/osm/prepaid\\_card\\_form.asp](http://localhost/osm/prepaid_card_form.asp)) in order to gain unauthorized access to the part of the system that they are not allowed to access.

## 3) Stress testing

This is to determine the program fulfills the requirements defined for it. It is equally important to ensure that the program works, as it should even under extreme conditions. In order to perform stress testing, execute the system in a manner that demands resources in abnormal, quantity, frequency or volume.

## 4) Performance testing

For real-time and embedded systems, software that provides required function but does not conform to performance requirements is unacceptable. Performance testing is designed to test the run-time performance of software within the context of an integrated system. Performance testing occurs throughout all steps in the testing process.

## 7.3 Test Data

Sample data is needed to prepare and keyed in for execution. Sample output is then compared with the actual results to find errors. There are 3 basic types of test data used :

### 1) Normal test data

Normal test data will provide a simple test to determine whether the program is error-prone or error-free.

### 2) Extreme test data

Extreme test data will test how well the system in handling huge amounts of data without affecting the overall system's accuracy and efficiency.

### 3) Stress test data

Stress test data is performed where errors are keyed into the system intentionally. The purpose is to determine how the system will react and handle incorrect data.

## 7.4 Online Shopping Administration Subsystem Testing

The following are the final test cases for each module. The results were correct for all of them.

No.	Test Condition	Expected Result	Pass/Fail	Remarks
1	User enters valid username and password and clicks "Login" button.	Access is granted where main page is displayed.	Pass	-
2	User enters invalid username and password and clicks "Login" button.	Access is denied where alert message is displayed.	Pass	-
3	User clicks "Login" button without enter username or password or both.	Alert message will be prompted. User must enter username and	Pass	-

		password before logins the system.		
4	User with user level admin1 entered the system	All the modules can be selected from the main page. Add, search, read, update and delete operation is granted.	Pass	-
5	User with user level admin2 entered the system	All the modules can be selected from the main page excluding the user management module. Add, search, read, update and delete operation is granted.	Pass	-
6	User with user level general1, general2 and management entered the system.	All the modules excluding the prepaid card module and user management module can be selected from the main page. Only search and read operation are granted.	Pass	-
7	User clicks on every buttons	User will be redirect to	Pass	-

	that represent each module.	appropriate web page.		
8	User clicks on the “Add” button provided in each module.	The form to add new record is displayed.	Pass	-
9	User clicks on “Add to database” button without filled up the mandatory fields.	Alert message will be prompted. User must filled up the mandatory field before submit the form	Pass	-
10	User fills up information in wrong format into the provided field and clicks on “Add to database” button. E.g. enter alphabet into number field, enter date value without following the requirement (MM/DD/YYYY).	Alert message will be prompted. User must filled up information in correct format before submit the form.	Pass	-
11	User fills up the form correctly without miss up any mandatory field and clicks on the “Add to database” button.	A new page is displayed where user can review the entered information.	Pass	-
12	At the review page, user	Alert message will be	Pass	-

	clicks on the “Add to database” button.	prompted to confirm the proceeding of the add operation.		
13	User clicks “OK” on the alert message.	Add operation is conducted where new record is added into the database.	Pass	-
14	User clicks “Cancel” button on the alert message	The add operation is aborted.	Pass	-
15	User clicks “Back” button in each modules.	User will be redirect to the previous page.	Pass	-
16	User clicks “Search” button without entered any keywords or made any selection.	Alert message will be prompted. User must entered the search keywords or at least made one selection to begin the searching.	Pass	-
17	User enters at least one keyword or makes one selection and clicks on the “Search” button.	The search result will be displayed in a new page. All the records that found will be displayed in row. If no record was found,	Pass	-

		message “no records found” will be displayed.		
18	At the search result page, user clicks on the provided hyperlink to view the details of the record.	The details of the requested record are displayed in a new page.	Pass	-
19	At the view details page, user clicks “Modify record” button.	A new page with all the details of the selected record is displayed and the details are presented in editable form.	Pass	-
20	User clicks on “Save changes” button by deleting the data in the mandatory fields.	Alert message will be prompted. User must fill up the mandatory field before submit the form	Pass	-
21	User modifies the information in wrong format into the provided field and clicks on “Save changes” button.	Alert message will be prompted. User must filled up information in correct format before submit the form.	Pass	-

22	User modifies the information correctly without miss up any mandatory field and clicks on the "Save changes" button.	A new page is displayed where user can review the entered information.	Pass	-
23	At the review page, user clicks on the "Add to database" button.	Alert message will be prompted to confirm the proceeding of the update operation.	Pass	-
24	User clicks "OK" on the alert message.	Update operation is conducted where the record is updated into the database.	Pass	-
25	User clicks "Cancel" button on the alert message	The update operation is aborted.	Pass	-
26	At the view details page, user clicks on the "Delete record" button.	A review page to confirm with the delete operation is displayed.	Pass	-
27	At the review page, user clicks "Delete record" button.	Alert message will be prompted to confirm with the delete operation.	Pass	-
28	User clicks "OK" on the alert message.	Delete operation is conducted where the	Pass	-

		record is deleted from the database.		
29	User clicks "Cancel" button on the alert message	The delete operation is aborted.	Pass	-
30	In the product delivery and stock update module, user clicks "update delivery" button.	<p>If the amount of the ordered product is less than the amount in stock, update operation is allowed. Current stock is the amount of product in database deducts the amount of the ordered product. The amount of product in database will be updated. However if the amount of the ordered product is more than the amount in stock, update operation is denied and alert message will be displayed.</p>	Pass	-

31	In the new information generator module, user clicks on the “Generate” button to create customer class label.	Customer will be classified to different class labels by using classification rule.	Pass	The generate process involve many database select and insert operation. Therefore the process is time consuming.
32	In the new information generator module, user clicks on the “Clear” button to clear customer class label.	All customer class labels will be cleared.	Pass	-
33	User clicks on “logout” button.	User will redirect to the login page. User must reenter username and password to login the system.	Pass	-
34	User logins the system but leave the system idle after 10 minutes. After that user clicks on any button.	System will be times out after 10 minutes. User will be redirect to login page and must reenter username and	Pass	-

		password to login the system.		
35	User types in the file address (for example, <a href="http://localhost/osm/main.asp">http://localhost/osm/main.asp</a> ) to skip the login page.	User will be redirect to the login page. User cannot gain access to the system by skipping the login page.	Pass	-
36	Authorized user with user level other than AdminI types in the file address (for example, <a href="http://localhost/osm/user_mgt_form.asp">http://localhost/osm/user_mgt_form.asp</a> ) to gain access to the part of the system that they are not allowed to access.	User will be redirect to the main page. User cannot gain access to the part of the system that they are not allowed to access.	Pass	-

## **CHAPTER 8: DISCUSSION**

### **8.1 Problems Encountered And Solution**

Various problems were encountered throughout the development of Online Shopping Administration Subsystem:

#### **8.1.1 Defining The System Scope Due To The Wide Area Of Study**

The online shopping application is not new in the market. There are many more that is available from the Internet. At the beginning, I faced difficulty to determine the features and functions that I should include in the system because of limited information gathered. In order to really understand the online shopping administration's workflow, much time was spent in gathering the information of user's needs and asking people for their opinions. Besides that, research and reading on the books also helped me in determining the features that I should include in my system.

#### **8.1.2 Program Coding**

Basically the problem encountered in web pages coding involves the HTML coding, VB scripting, Java scripting and ASP programming especially the study of ASP which is very crucial to the development of the system. Most of the coding and scripting problems were encountered in the early stage of the system development. This is due to the ambiguity and lack of understanding in the early stage. To understand such a large amount of information in a short time was not an easy task.

The only solution to overcome is to adopt a divide-and-conquer approach by first concentrating and understanding the basic concepts of the programming. As the development of the system goes on, the understanding gradually builds up and most of the problems encountered in the earlier stage were overcome easily.

### **8.1.3 Retrieving Data And Inserting Data Into A Recordset**

There were several issues that need to be examined before retrieving data from the recordset. At the early stage of the project, it was difficult to understand how the database actually works. But after referring the reference books, I came to understand the requirements that need to be done before retrieving the data from the database.

### **8.1.4 Understanding The Error Messages**

The error messages derived from the Personal Web Server (PWS) usually consist of error code and error message that are sometimes vague and unclear of the cause. Understanding the error messages was a constant struggle because I need to continue debugging, correcting, undoing and testing it until the program is correct. By referring to reference books and asking friends, some errors were solved easily with a clear understanding of why the error occurred but some were solved without knowing why the errors occurred or why the solution works. Through the trial and error process, I can conclude that more experience and practice in ASP is the way to understand what is the cause of the error message.

### **8.1.5 Understanding And Applying Data Mining**

Due to lack of knowledge in data mining, a lot of time was used to understand the concept of data mining and how to apply the concept into the system. But after referring the reference books, I finally able to understand and implement data mining technique in the system.

### 8.1.6 Lack Of True Customer Information

I faced difficulty to gain customer information needed in my system database. Therefore I forced to create the information I needed in the database in order to test the data mining technique that I have applied in the system.

## 8.2 Evaluation By End Users

Upon the completion of system testing, user evaluation was conducted to assess the overall outlook of the system, effectiveness of the system towards user's request and to determine whether the system meets user's needs and expectations.

The user evaluation form can be obtained from appendix E at the end of the report.

Following is the result of the user evaluation forms:

#### ➤ User Interface Design

Most of the user found that the system user interface design is attractive and consistent. The color combination is appropriate. Text style in the web pages is easy to read.

#### ➤ The way to present data

Most of the user satisfied with the way of the system to present the requested data. They can easily search, add, modify and delete data. Not much of time is needed to learn how to use the system.

#### ➤ Effectiveness of the system

Most of the user agreed that the system is effective and meets user requirements and expectations.

## **8.3 System Strength**

### **8.3.1 Easy Management And Maintenance**

The system allows the users to manage and maintain the system easily through user-friendly graphical user interface. User can add, search, delete and update the data in the database just by a 'click' on the button. Users need not write and run any SQL statement in order to add, change or query the data in the database.

### **8.3.2 User Friendly Graphical User Interface**

The GUI was designed to have an uncluttered look to ease the user to manage the system. Buttons was clearly and appropriately labeled. It has a very user-friendly and consistent environment that is similar to other windows application. This will shorten the learning curve and reduce the training cost which involves money and time.

### **8.3.3 Informative Messages**

The system will prompt error messages to the user whenever the user attempts to perform an illegal action. It also provides message after the user has completed a certain task. These messages are very helpful and important to the users because it provides up-to-date information to users about what is going on and what has been done to the system.

### **8.3.4 Authorization And Authentication**

Every user has a login name and password where only the authorized users are allowed to access the system. Both discretionary access control and multilevel access control are used in the system where only user with user level "admin1" is allow to access whole system whereas other users with other user level are not

allowed to access user management module and prepaid card module. For user level other than “admin1” and “admin2” are restricted to read operation only (add, update and delete operation are not permitted). The login page will be displayed to the user first if the user attempts to enter the system before logging in.

### **8.3.5 User Input Validation**

All the user inputs will be validate with client-side script, which is JavaScript before process by server in order to avoid unnecessary server-side processing and to speed up system response to user request. Besides that validation is also conducted with server-side script, which is ASP. Double validation must be used in the system because some browsers might have disable client-side script or cannot support JavaScript. In case of the occurrence of this kind of situation, user input validation still can work with server-side script.

### **8.3.6 Improved Functionality**

The New Information Generator Module in the system is the new suggested function to add on to the system. This module applied the data mining technique where user information such as favorite, industry involved, buying record is used to identify customer class label (e.g. buy computer, buy cosmetic and buy fashion). Then Administrator can use these new information to interact with customer in a more personal way.

## **8.4 System Constraints**

Due to time constraints, Online Shopping Administration Subsystem does not include all the functional modules that a complete administration system should have. Some of the modules have been simplified to show the rough skeleton of what the basic administration system should do.

#### **8.4.1 Lack Of Functional Modules**

This system does not really reflect a real Online Shopping Administration System. It can be considered as a miniature of a complete administration system. Currently the available functional modules in the system are limited. This is due to the lack of sufficient information collected during data collection.

#### **8.4.2 Database Not Encrypted**

The database is not secured enough because all the information are stored in a plain text form rather than in an encrypted format.

#### **8.4.3 Lack Of Accuracy**

Since I faced difficulty in gathering the real customer information needed in the system database, I forced to create false data to test the New Information Generator Module that I have developed. Therefore the accuracy of the new generated information is forced to sacrifice. However with all the tested data, I am able to prove that this module is functioning well and have achieved the goal of the module.

### **8.5 Future Enhancements**

#### **8.5.1 Provide More Functionality**

As mentioned above, this Online Shopping Administration Subsystem does not reflect the real administration system. Therefore, there should be additional modules to be included in the system to enhance the capability of the system. For example, the modules, which can be included, are printing the customer's order, having sales report, etc.

### **8.5.2 Improve Accuracy Of The New Generated Customer Information**

To improve the accuracy of the new generated customer information, real and large customer information must be obtained. Besides that calculation must be conducted to determine the root attribute to be use in building the decision tree and finally convert the decision tree to the classification rules.

## **8.6 Knowledge And Experience Gained**

Knowledge gained throughout the development of the system is really valuable. Knowledge was gained on Internet development environment, the Internet technologies, programming coding and concepts and as well as web-server configuration. Besides that good software engineering techniques are also important in developing a system. Here, theories and knowledge gained during the course of Information Technology like *System Analysis and Design* and *Software Engineering* were literally put into practice. There was also an exposure to the software tools such as Personal Web Server (PWS), Microsoft Access, Macromedia Dreamweaver MX and some popular programming languages (e.g. ASP, JavaScript and VBScript)

Besides that, the knowledge gained from this project is the awareness of user's needs. It was found that users need a user-friendly environment, a readable homepage and clear instructions and guidance. And the programmer needs to find and organize the information acquired and analyze the behavior.

There are improvements in skills of finding information, classifying fields, solving problems and independently plan and accomplish the project on schedule without much supervision.

No doubt that experience has been gained and new knowledge has been acquired. More importantly, the process of doing it has been an exposure to me on

how to really plan and work on a project. One important thing I realized is that it is really useful and important to have an up-to-date knowledge and information with the fast and ever changing technology of information technology. This project is really beneficial to me in helping me to prepare myself for the working environment in future.

## **8.7 Conclusion**

Building a web-based application package is a challenging task. A lot of research, time and effort have been involved in making this project successful and in fulfilling the tasks requirements. A comprehensive knowledge in building a web-based application is also necessary and important especially knowledge about the client and server. It is because they are the main communication feature to a web-based application.

In conclusion, Online Shopping Administration Subsystem was completed on time and integrated with Online Shopping Submission And Processing Subsystem successfully. It has actually reached its objectives in developing a more useful Online Shopping System that can interact in a more personal way with the customers. The system is easy to learn and use where users can master it within a short learning time.

## REFERENCE

### Book Resources

- [1] Jiawei Han and Micheline Kamber (2001), *Data Mining: Concepts and Techniques*, Morgan Kauffman Publisher.
- [2] R. Groth (2000), *Data Mining Building Competitive Advantage*, Prentice Hall.
- [3] Deitel, Deitel & Nieto (2000), *Internet & World Wide Web – How To Program*, Upper Saddle River, NJ: Prentice Hall.
- [4] Shali Lawrence Pfleeger, *Software Engineering Theory And Practice*, 2<sup>nd</sup> ed, Prentice Hall, 2001
- [5] McLeod Raymond Jr (1998), *Management Information System*, Prentice Hall International, Inc.
- [6] Jeffrey L. Whitten, Lonnie D. Bentley and Kevin C. Dittman (2000), *System Analysis & Design Methods*, 5<sup>th</sup> ed, New York, NY: McGraw-Hill Irwin, Inc.

### Web Resources

- [U1] Line9.com  
<http://line9.com/tek9.asp?pg=products&specific=jnmofnp0>
- [U2] Data Mining: Concepts And Techniques  
<http://www.cs.sfu.ca/~han/bk/1intro.ppt>
- [U3] Data Mining: Concepts And Techniques  
<http://www.cs.sfu.ca/~han/bk/7class.ppt>

- [U4] Macromedia Dreamweaver MX  
<http://www.macromedia.com/software/dreamweaver/>
- [U5] Macromedia Dreamweaver MX support center  
[http://www.macromedia.com/support/dreamweaver/ts/documents/lay\\_view  
\\_limits.htm](http://www.macromedia.com/support/dreamweaver/ts/documents/lay_view_limits.htm)
- [U6] ServerWatch (Web server)  
<http://dev-serverwatch.internet.com/webservers.html>
- [U7] ServerWatch (Apache)  
<http://dev-serverwatch.internet.com/reviews/web-apache.html>
- [U8] ServerWatch (IIS)  
<http://dev-serverwatch.internet.com/reviews/web-msiisv5.html>
- [U9] ServerWatch (jigsaw)  
<http://dev-serverwatch.internet.com/reviews/web-jigsaw.html>
- [U10] ServerWatch.(Personal Web Server)  
<http://dev-serverwatch.internet.com/reviews/web-mspws.html>
- [U11] Microsoft (ODBC)  
<http://www.microsoft.com/data/odbc/prodinfo.htm>
- [U12] Microsoft (OLEDB)  
<http://www.microsoft.com/data/oledb/prodinfo.htm>
- [U13] Microsoft (ADO)  
<http://www.microsoft.com/data/ado/prodinfo.htm>
- [U14] Network security  
<http://developer.novell.com/research/appnotes/1997/november/01/apv.htm>
- [U15] Data Encryption Page  
<http://www.anujseth.com/crypto/basic.html>

[U16] SSL

[http://www.its.otko.kz/information/cryptography/rsa\\_faq/faq/5-1-2.html](http://www.its.otko.kz/information/cryptography/rsa_faq/faq/5-1-2.html)

[U17] Weka

<http://www.cs.waikato.ac.nz/~ml/weka/>

[U18] e-focus Glossary

<http://www.efocus.com.au/Glossary.htm>

[U19] Client-Server Management

[http://home.global.co.za/~ajvr/client\\_server.htm](http://home.global.co.za/~ajvr/client_server.htm)

[U20] Microsoft Press

<http://www.microsoft.com/mspress/books/sampchap/3250.asp>

[U21] 3- and n-Tier-Architecture

<http://www.corba.ch/e/3tier.html>

[U22] Chap 8: Designing The User Interface

<http://digital.lib.kmutt>