# FACULTY OF COMPUTER SCIENCE & INFORMATION TECHNOLOGY UNIVERSITY OF MALAYA

**Electronic Accounting & Stock Management System (eASM)**

By

**Song Siow Ching**
**WEK030200**

Under supervision of

**Pn. Fazidah Othman**

And moderated by

**Dr. Mazliza Othman**
**Mr. Por Lip Yee**

A Project Report Submitted to the
**Faculty of Computer Science and Information Technology,**
**University of Malaya**
Dissertation submitted in partial fulfillment of the requirement for the
Bachelor of Computer Science
Session 2005/2006
Submission Date: 5 April 2006

# Abstract

In these early years of 21st century, information technology is playing an important role in our lives. With the explosion of knowledge and information nowadays, Information Technology must be adopted to better manage of these aspects especially to increase the quality of store data and information. And this is why Electronic Accounting and Stock Management System (eASM) comes in to bring a better experience in accounting and stock management.

The eASM is developed to assist the medium size company. The main objective of this system is to develop an online based web application system to store accounting information and manage item inventory control of the company. eASM is a system assists the users on better management of the accounts and stock in order to have better overview of the shared data and also provide the users with the consistent, efficient service that ensures satisfaction. The purpose of eASM system not only takes into account the user friendly aspects in the development methodology but also the usability aspect. It uses user friendly graphic user interface to provide usability functions to store, modified, and delete records.

ASP .NET is chosen to develop eASM. This is to take advantage of the ASP .NET can support VB .NET and c# that is more powerful. It's also control or event based programming that similar to VB6. Separation between logic and presentation make the system easier to maintain.

# Acknowledgement

The development of eASM is done through the advice, assistance, and contribution of many individuals. First of all, I would like to express my utmost gratitude to my final year project supervisor, Pn. Fazidah Othman who provides me with ultimate support, a lot valuable guidance, and advice throughout the whole development stage of eASM.

To my family members for being supportive, understanding and caring on the period I develop my system. Especially to my lovely parents and brother, I really garner my respect and appreciation to you all!

To my fellow course-mates who ready to share their knowledge while doing the final year project. I would like to take this opportunity to say my highest appreciation to them.

I would like to convey a special thanks to my sister, Song Siow Lin because willing to teach me the basic knowledge of accounting system and providing me very useful information.

Of course, I would like to extend my gratitude to my industrial training colleagues who enlarge my view on web application and give me confidence to build a system myself. I thank them all.

My highest appreciation refers to my entire supporter, so that I can finish my final year project smoothly. Thank you very much!

# Tables of Contents

## Chapter 3 Methodology



## Chapter 4 System Analysis

# List of Figures

# List of Tables

# Chapter 1

# Introduction

# Chapter 1 Introductions

## 1.1 Overview

Electronic Accounting and Stock Management (eASM) System is the system assists the users on better management of the accounts and stock in order to have better overview of the shared data and also provide the users with the consistent, efficient service that ensures satisfaction. It is similar to the accounting system that exists in the market, called UBS Accounting Software that many company is using it nowadays.

This eASM system is purposely developing for any company use to store their accounts data and information, like payment, invoices, purchase order, sales order and etc. Other than that it also can use to manage and maintain their inventory or stock, branches, customer, supplier, forwarder and etc. The enhancement that implement in eASM System is the cryptographic technique that will be use to encrypt the login password for the user to prevent unauthorized users to enter into the system. It also can prevent the hackers or password crackers hacks into system using brute force technique or others technique.

This eASM system will be design in three main tiers plus one common utilities tier. The three main tiers are GUI (Graphic User Interface) presentation layer, business logic component layer and the data access component layer. The function of these three layers are GUI presentation layer is use for design the project template that to be show to the end users, business logic component layer is use to communicate between GUI layer and the data access layer, and the data access layer is use to connect the whole project with

the database. Besides that this project also include common utilities layer to add some function that will frequently use in this project so that can reduce the line of coding with repeating the same code.

eASM will be develop using Microsoft Visual Studio .Net and the database that will be use is Microsoft SQL Server. Other than that the reports and the payment receipts will be generated by using Report Manager Designer.

## 1.2 Objective

The major objective of this system is to develop an efficient e-Accounting and Stock Management System for midsize enterprise to reinstate existing manual system or stand alone system in order to save their time and money. It can save time and money because eASM system is paperless and it is an online system and run in the platform. Users just need to type in appropriate data in the correct field and save it. After finishing enter the particular entire record, user can just print out all the reports or information of that record without doing by manually or hands-on.

### 1.2.1 To develop a web base application

By developing web base application, all the data and information about account and stock can be broadcast through the internet. It will give the company to maintain their accounts and stock efficiency and easier if they have branches or not. They can access their company database anywhere and anytime as long as their server is not down.

### 1.2.2 To develop a database

The manipulation of the database is to store the data input such as invoices, customer payment, delivery order and etc by the authenticate users. So that users can review their information online and amend those data. In addition, the data that stored is updatable.

### 1.2.3 To generate invoice, purchase order (PO), delivery order (DO), and payment receipt and etc

Users may generate the invoice, PO, DO or receipt after they complete input the mandatory data from time to time. It will be show out in .PDF form so that can save it or print it out when necessary.

### 1.2.4 To generate reports

After user had entered the entire data for particular record, user may print out the record information or data by just clicking the "Print" button. System will automatically generate a report and display it in .PDF form in the web browser.

## 1.3 Expected Outcome

eASM System is an online accounting system which targeted users will be company clerks or accountants. First the user has to login the page by key in user id and password. After authenticate user is a valid user, user may choose the appropriate module to input data into database or modified the existing data. Each user has different access control. For example, administrator can login to all of the modules in this system, but some users may login to sales, purchase and stock modules only but cant login to the maintenance or administrator module. This is to ensure the consistency of the data.

After choose the module, it will show the search screen of that module with four search parameters. Besides that, user also can see all the records for that module. User can filter the records by using filter parameters. It is to easier the user to find the records faster and efficient. After that user can choose either one of the record and preview all the data of the entire module. For this side, it have the ability to let the end users modified the existing data or records. Other than that, user also may create a new record to store the others information.

For each record, there will be a main page and details page on it. On the main page user must key in the mandatory field before go into the details page. Details page is the most important because all the important information will be keep on this page. For example, customer payment module, the main page will keep the information of the customer and the details page just will keep the data on the purpose of payment, how much that customer paid and which invoice that refer to.

When user completes the procedure, they can preview the receipt, order form, purchase form or others by just click the "Print" button and the entire record will show out in PDF form. So user can save it in a copy as a reference or print it out immediately. Besides that, user also able to trace their stock inventory summary (on-hand) with using eASM System. Hence eASM system is very efficient and user friendly.

## 1.4 Scope

eASM project is an online accounting and stock management system. It will provide completed function for storing company accounts data and also stock inventory management. Each module will have sub-modules.

This system can be divided into six modules as below:

**Module 1: Security Setting**

1.1 Users

1.2 User Group

1.3 User Authentication

1.4 User Access Control

This module is to create a new user and assign it into a user group. Each user has to authenticate before they start using the system. This is to ensure the system is secure and the data inside is protected. Besides that each user also has a different access right. The most powerful user is the administrator.

**Module 2: Maintenance**

2.1 Customer

2.2 Supplier

2.3 Branch

2.4 Forwarder

2.5 Currency

2.6 Country

2.7 Shipment Term & Mode

2.8 Payment Mode

2.9 Packing Type

Maintenance module is used to manage the company, people, and others miscellanies. It's used to customize the customers, suppliers and branches with the standard and appropriate information or data. On the other hand, admin also can maintain the currency exchange rate, item measurement or shipment mode

**Module 3: Purchase**

3.1 Purchase Order (PO)

3.2 Purchase Credit Note (CN)

3.3 Purchase Debit Note (DN)

Purchase module is use to manage the business between company and supplier. Supplier will supply the goods to company to produce the new item or product. The purpose of this module is to store all the entire data and process information.

**Module 4: Sales**

4.1 Sales Order (SO)

4.2 Delivery Order (DO)

4.3 Sales Invoice

4.4 Packing list

4.5 Sales Credit Note (CN)

4.6 Sales Debit Note (DN)

4.7 Customer Payment

A sale is the process company sells the product to the customer. In the selling process, it may have an order form, invoice, CN, DN, and the payment receipt. This module is used to store the entire data, information and generate the form, and receipt.

**Module 5: Stock Management**

5.1 Stock Receiving

5.2 Stock Transfer

5.3 Stock Category/Group

5.4 Stock Information

5.5 Unit Of Measurement (UOM)

Stock management is used to manage the whole inventory in the company. Its can be use to trace the stock quantity faster and easily. This module just can use by the power users only.

## 1.5 Problem Statement

eASM System is useful online accounting system that aid to the small, medium or even huge company. However this system is constraints by internet connection. If the company has many branches, each branch must have an internet connection to connect with the server sites application. If the connection is down, then the employee in the branch can't do anything besides wait only. But this is not a major problem nowadays because our country has many companies that provide the broadband connection services.

The second concern is the server. Because eASM System is an online system, so it must store the whole system in the server side. But the problem is when the server is down or broken, the whole system cannot run and reach by any other client side. And it also has risk that the data in the database are lost! It is very dangerous and may take a long time to recover back. So to overcome this issue, company should backup the whole database from time to time to make sure backup data is the latest information.

Third, the problem of the license of the database that use for this system. eASM is using Microsoft SQL Server 2000 database. It is not an open source. For those company that want to use this system should buy the license of SQL Server. Even though it is licensing software, but the price is cheaper than Oracle database. It just takes 25% the cost of Oracle. eASM System use this database because it is easy to use and easy to maintain.

Other than that, the weakness of using eASM is each of the pc must have Adobe Reader software to preview the receipts, forms, quotation and etc. Therefore each of the computers must install with this software before using it.

Nevertheless, the system will be a project that comes from me and my group member. So the satisfaction of completion a project will still there. But there is not a major concern in this eASM System.

## 1.6 Project Schedule

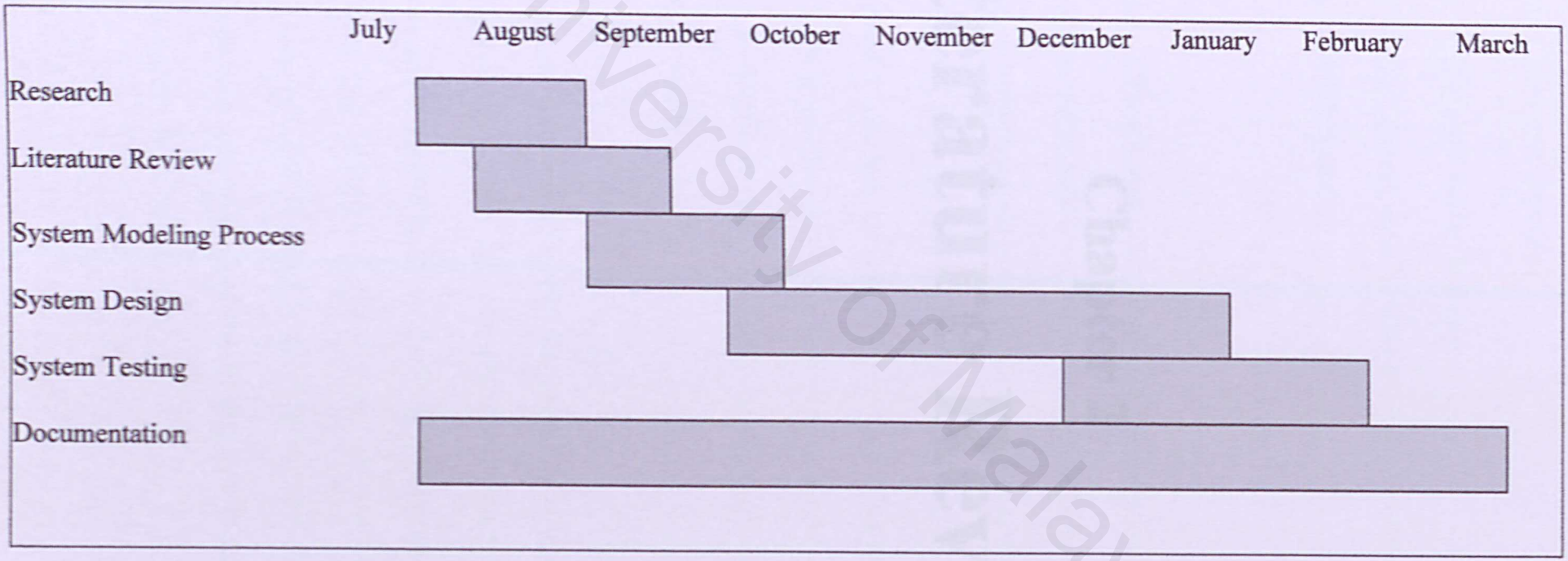



**Figure 1.1 Project schedule time line**

# Chapter 2

# Literature Review

# Chapter 2 Literature Review

## 2.1 Survey on Current System

### 2.1.1 Case Study 1 – UBS Accounting Software

UBS Accounting Software is stand alone business application software for small and medium enterprises (SMEs) market.

**Weakness**

- **Stand alone** – UBS software is a stand alone system that can't make it online. As mention in the above weakness, stand alone system can't communicate with other branches. So the accounts and stock for the company are non-unification.

- **Not user friendly** – the layout of the UBS Accounting Software can consider as complex. It is hard to a fresh employee because the newer would not know how to use it. So the users have to go for training class to learn how to use it to store the data and information.

- **Hard to manage** – for those companies that using this software to manage their accounts and stock, by the time they want summarize their accounts or want to manage their stock at the month end, they have to backup the data in UBS for each branch and send the copy to the head office. After the head office gets the backup data, they must restore it to their computer and then just can do summarize their accounts. So it is hard for the employee to trace the whole accounts and data for all of the branches.

- **Expensive for network version** – UBS Software is available for networking version. That means company also may damn their data in a server through UBS Software Networking version. But the price is higher than the stand alone version. And another of the disadvantages of UBS is one UBS log only can use by one use. Hence if the company has ten branches in Malaysia, so they have to buy ten logs for ten branches.

**Strength**

- **Anytime, anywhere access** – eASM is an online system. All the data are store in the server side. Everyone can serf to that side and they will able to manage their company accounts anytime and anywhere. Even after office hours, users also may do their work at home as long as they have computer with internet connection.

- **User friendly** – the interface of eASM System is very tidy and understandable. Even the newer also may use this system. It also provides a user manual or tutorial to teach the fresh users use eASM System.

- **Operational efficiency** – because eASM is an online system and it just use only one database, so the account and stock management for the whole company can be see by the all users especially the head office accountant. Therefore they can easily trace the profit and lost of the company.

- **Cheaper** – eASM only need one license for one company even though they have ten branches or twenty branches. Other than that because this system is a web base application, therefore client no need worry to spend extra money buy the networking version or stand alone version system.

### 2.1.2 Case Study 2 – CYMA IV Accounting

CYMA IV Accounts Payable system provides advanced tools to manage vendor relationships, process invoices and manage payments.

**Weakness**

- **No way to track stock** - CYMA Accounting does not have an inventory or stock management module. This module is the most important part in the accounting system to manage the company stock for a retail business.

**Strength**

- **Inventory module** – this module can overcome the limitation of the CYMA IV Accounting software. So the end users may calculate how much inventory is kept on hand and when need to order new stock.

### 2.1.3 Case Study 3 – E-Accounting System

This is a system that developed by one of the senior in FSKTM session 2004/2005. The objective of this system is to develop an efficient online accounting system for small and mid-size enterprises and to reinstate their existing manual system in order to save their time and money.

**Weakness**

- **Lack of security** – this system is embedded with the user id and password. But it never uses any security method like cryptographic technique to ensure the safety and security of the system. It is very dangerous if the hacker hack into their system and stole company proprietary data.

- **ASP vs ASP .NET** – E-Accounting System is using ASP to develop their modules. It just support the VB script that less powerful. It uses the top-down programming like most of the scripting languages. And there is no separation between logic and presentation layer. It also may contain multiple forms per page.

**Strength**

- **Using encryption technique** – the data in the system is very confidential and private. Implementing cryptographic technique to this project can ensure the system is secure and protected because it can prevent the malicious hackers, Denial of Service (DoS), organized criminals or spies.

- **ASP .NET** – eASM is using ASP .NET to develop the system. ASP .NET can support VB .NET and c# that is more powerful. It's also control or event based programming that similar to VB6. Separation between logic and presentation make the system easier to maintain. And it is use only one form per page.

## 2.2 Tools

1. Microsoft Visual Studio .Net – ASP .Net
2. Microsoft SQL Server 2000 Standard Edition
3. Report Manager Designer
4. Acrobat Reader
5. Adobe Photoshop 6.0
6. Microsoft Internet Information Service (IIS)

### 2.2.1 Reason why I choose Microsoft Visual Studio .Net – ASP .Net:

**Advantages of ASP.Net:**

**Powerful database-driven functionality**

Like ASP (Microsoft's language preceding ASP.Net), ASP.Net allows programmers to develop web applications that interface with a database. The advantage of ASP.Net is that it is object-oriented and has many programming tools that allow for faster development and more functionality.

**Faster web applications**

Two aspects of ASP.Net make it fast -- compiled code and caching. In the past, the code was interpreted into "machine language" *when* your website visitor viewed your page. Now, with ASP.Net the code is compiled into "machine language" *before* your visitor ever comes to your site.

ASP.Net allows programmers to set up pages or areas of pages that are commonly reused to be cached for a set period of time to improve the performance of web applications.

**Memory leak and crash protection**

ASP.Net automatically recovers from memory leaks and errors to make sure that your website is always available to your visitors.

**Multiple language support**

Programmers can actually write their code in more than 25 .Net languages (including VB.Net, C#, and JScript.Net). This allows programmers to develop your site in the language they know best and it means that you can more easily find programmers to support the work on your site.

| ASP | ASP .Net |
|---|---|
| VB Script – Less Powerful | VB.Net or C# - More Powerful |
| Top-down programming - like most scripting languages. | Control/Event based programming – similar to VB6 |
| No separation between logic and presentation | Separation between logic and presentation |
| Multiple Forms per page | Only one form per page |

**Table 2.1 Comparison between ASP and ASP .Net**

### 2.2.2 Reason why I choose Microsoft SQL Server 2000 Standard Edition:

- In term of independent analysis of price/performance ratio, Microsoft SQL Server 2000's licensing is 25% the cost of Oracle.
- Microsoft SQL Server 2000 is easier to maintain compare to Oracle 10g and MySQL.

### 2.2.3 Reason why I choose Report Manager Designer:

It supports development environment accepting ActiveX controls (Visual Basic, Visual FoxPro, any Visual Studio.Net language, etc).

A true net and web report server with no license fees and multiprocessor support.

It works in Windows and Linux, You can distribute the report designer, so you modify the reports without modifying your application, and the result can be exported to Adobe PDF format.

### 2.2.4 Reasons why I choose Acrobat Reader:

- To generate report in PDF form.

### 2.2.5 Reasons why I choose Adobe Photoshop 6.0

- To design the icons, buttons and header of the system.

### 2.2.6 Reason why I choose IIS:

IIS is a powerful Web server that provides a highly reliable, manageable, and scalable Web application infrastructure for all versions of Windows Server 2003.

IIS helps organizations increase Web site and application availability while lowering systems administration costs.

# Chapter 3

# Methodology

# Chapter 3 Methodology

## 3.1 Introduction

Methodology can be defined as System Development Life Cycle (SDLC) model. It consists of several steps to create eASM system. System Development Life Cycle, or SDLC, is used by a systems analyst to develop an information system, including ni..u on9 arequirements, validation, training, and user ownership through investigation, analysis, design, implementation and maintenance. SDLC is also known as information systems development or application development. An SDLC should result in a high quality system that meets customer expectations, within time and cost estimates, works effectively and efficiently in the current and planned system infrastructure, and is cheap to maintain and cost-effective to enhance. SDLC is a systems approach to problem solving and is made up of several phases, each comprised of multiple steps:

- The software concept - identifies and defines a need for the new system
- A requirements analysis - analyzes the information needs of the end users
- The architectural design - creates a blueprint for the design with the necessary specifications for the hardware, software, people and data resources
- Coding and debugging - creates and programs the final system
- System testing - evaluates the system's actual functionality in relation to expected or intended functionality.

**The six official phases are:**

- Preliminary Investigation
- Systems Analysis

- Systems Design
- Systems Development
- Systems Implementation
- Systems Maintenance

## 3.2 Development Methodology

There are various type of methodology used in the developing system. The common methodologies used are waterfall model, prototyping model and etc. the methodology chosen to develop eASM is "Waterfall Model".

**Waterfall Model**

In Royce's original waterfall model, the following phases are followed perfectly in order:

- Requirements specification
- System Design
- Construction (implementation or coding)
- Integration
- Testing and debugging (verification)
- Delivery
- Maintenance

To follow the waterfall model, one proceeds from one phase to the next in a purely sequential manner. For example, one first completes "requirements specification" — they set in stone the requirements of the system. (Example requirements for eASM may

be "eASM allows admin have full control for the whole system; eASM enables normal to search for records", although real requirements specifications will be much more complex and detailed.) When and only when the requirements are fully completed, one proceeds to design. The software in question is designed and a "blueprint" is drawn for implementers (coders) to follow — this design should be a plan for implementing the requirements given. When and only when the design is fully completed, an implementation of that design is made by coders. Towards the later stages of this implementation phase, disparate system components produced by different teams are integrated. (For example, admin may have been working on the "Setting" module of eASM and normal user may have been working on the "Sales" or "Purchase" modules of eASM. These modules must be integrated together to produce the whole system.) After the implementation and integration phases are complete, the software product is tested and debugged; any faults introduced in earlier phases are removed here. Then the system is released, and later maintained to introduce new functionality and remove bugs.

Thus the waterfall model maintains that one should move to a phase only when it's proceeding phase is completed and perfected. Phases of development in the waterfall model are thus discrete, and there is no jumping back and forth or overlap between them.

However, there are various modified waterfall models (including Royce's final model) that may include slight or major variations upon this process.

# Chapter 4

# System Analysis

# Chapter 4 System Analysis

Requirement analysis is done during analyzing system needs. Requirement analysis includes analyzing and determining functional requirements and non-functional requirements.

## 4.1 Functional Requirements

Functional requirements are functions or characteristics that determined by the developer with agreement from users and stated by them to be developed into the system. The system is considered incomplete if any of the necessary function is not include. It is because the basic software requirement is correct, complete and consistent. Data flow diagram has been use to show the requirement for the flow of data. The diagram shows the flow of the data into the system, the way they are transformed and integrated in the system.

### 4.1.1 Security Module

This module checks for the end user's level when login to the system and bring the end users to the control that they had access. User shall be able to key in a valid user name and password to be able to use the eASM System. New user is only can created by the administrator and the user may change their password from time to time. Administrator shall create, delete or modify user's information.

### 4.1.2 Search Function

Each of the modules might have the search screen. End users may be filtering the records by records number, generated date, branch, status and others. Other than that user also can sort the records by ascending or descending with clicking the tab at the title bar.

### 4.1.3 Add New Record

This function is implement or all of the modules inside eASM System. It allows users to generate or create the new branch, user, item, record, and so on. Each record must have unique record id since they are many records in every modules. Users no need to key in the record id by themselves because the system will auto generates the new record id when the users create a new record. The record id will begin with the four alphabets prefix and ten digit numbers subsequently.

### 4.1.4 Edit Record

This function is only allowing users to make changes if the status of the record is 'Outstanding'. When the record status has been change to 'Complete', then users may not able to modify the record anymore. To update the particular record, users just have to click on the entire side that they would like to modify, do the changes and lastly save the changes. If the record details are involving stock or others inventory quantity, systems will rollback the particular item quantity from the database and save the latest amount.

### 4.1.5 Delete Record

This function is to allow users to delete the record that they had been wrongly created or unnecessary record. Same as the edit record function, after user delete the record, system also will rollback item quantity automatically. The record will be permanently deleted and it cannot retrieve back anymore. The purpose of this is to keep the database integrity and maintainability in space.

### 4.1.6 Print Preview

This function is to generate the reports and receipts in real time. User shall be preview the report immediately after they complete fill in the record information in the .PDF form. User may save it or just print it out as a reference.

## 4.2 Functional Description

### 4.2.1 Home

*4.2.1.1 Welcome*

- Welcome page will be show to user after they login to let user know that they had success login to the system.

### 4.2.2 Security Setting

*4.2.2.1 User Group*

- User group is a function that user can use to manage a collections of user
- User can easy control the user like user access right, privilege & etc from user group functions

*4.2.2.2 Users*

- Users is a function that user can use to manage the system users.
- User can control the system user access right on all the functionalities from Users function.

*4.2.2.3 User Authentication*

- The system supports various levels of security and authorization in logons.
- The systems stores user preferences, which are accessed during logon and help the system, anticipate the workflow needs of the user.

*4.2.2.4 User Access control*

- User access control is help to control user on the level of security and authorization in the system
- Access privilege to every functionalities in the system is controllable by user or by user group
- User cannot access to the functions if he/she does not have permission.

### 4.2.3 Maintenance

*4.2.3.1 Customer*

Allow to search by customer code, customer name and customer address.

Allow to print customer listing with the customer details.

- Allow to print customer label.
- Forwarder name – If using FEDEX, no address needed, only contact number.
- Allow to select more than 1 item for customer inner code maintenance.
- All documents to customers included item number, customer inner code, and alternate part number.

*4.2.3.2 Supplier*

- Allow to search by supplier code, supplier name, supplier address, brand and product line.
- Allow to print supplier listing with the supplier details.
- Allow to print supplier label.
- Allow to add new items in brand & product line.
- Allow multiple selections for brand & product line.

*4.2.3.3 Item*

- Allow to search by item code, item brand, item group and product line.
- Allow to print Item listing with the item details.
- Each item will have a reorder point.
- Alternate Item No. to be included in all documents to supplier (RFQ, PO, etc).

*4.2.3.4 Currency*

- Currency table to be maintain by the users as and when is needed.
- This table will be used during the quotes consolidation part whereby currency conversion is required to do the quotes comparison.

**4.2.4 Purchase**

*4.2.4.1 Purchase Order*

- Purchase Order is a function for user generate a purchase order to supplier
- Main functions of Purchase Order include: Search, Save and Print
- Purchase Order is aggregated into the following attributes:
- For non-inventory items PO,
  - User select supplier info and details of the order (free text).
- For inventory items,
  - User can generate PO based on sales order and supplier's quotation.
  - User can select items from once or more sales order from different customer.

- System automatically retrieves the price from supplier quotation. User can amend the price

- Once user confirm the entry
  - The system checks for missing information and validates the data. If all information is present and valid, the system acknowledges the entry with purchase order no and stores it into purchase order data store with status "New".
  - The system automatically update sales order for the purchased qty
- User can select either fax or through electronic email to supplier
- User prints out the order which is without sales order or order qty more than sales order qty for approval purpose
- For editing purchase order,
  - Revision of purchase order is allowed up to 5 times.
  - User able to view the received qty and rejected qty
  - User cancels the purchase order by changing the status to "Cancelled".
  - No amendment allowed if status is "Canceled" or revision more than 5 times
  - Amendment of purchase order will be stated as amended copy and highlight the amendment item when send to supplier
  - User can cancel purchase order line items even though there is no revision is allowed.
  - If all order items have been received or cancelled, the system automatically changes the status to "Closed"

- If cancelled line items is based on sales order, the system automatically updates the sales order for the cancelled qty

- Document Printout
  - For inventory item (project), user can choose to show or hide the item list in the print out document that issued to supplier. Only the price for the project will be displayed and price for individual items will not show in the printout.
  - Revision number will show in the printout
  - Amended item will put with an indicator at the side (or bold)

*4.2.4.2 Credit Note*

- Credit Note is a function for user to create goods purchase that are undercharged by supplier
- Main functions of Credit Note include: Search, Save and Print
- Credit Note is aggregated into the following attributes:
- User search for the item by item no, description, item group, delivery order no and invoice no
- The system automatically lists the items that match the search criteria
- The attributes of search result are:
  - Item No
  - Description
  - Item Group
  - Purchase Order

- PO Price

- User can select the items from the search result, key in the quantity, price & remarks, then click "Add" button to add the items to details of Credit Notes.
- Once user confirms the entry
  - The system checks for missing information and validates the data. If all information is present and valid, the system acknowledges the entry with Credit Note no and stores it into Credit Note data store.
- Additional validation/control
  - At least one line item required for a Credit Note
  - After month end closing, user cannot do any amendment on Credit Note transaction that generated before that.

*4.2.4.6 Debit Note*

- Debit Note is a function for user to create goods purchase that are overcharged by suppliers
- Main functions of Debit Note include: Search, Save and Print
- Debit Note is aggregated into the following attributes:
- User search for the item by item no, description, item group, delivery order no and invoice no
- The system automatically lists the items that match the search criteria
- The attributes of search result are:
  - Item No

- Description
- Item Group
- Purchase Order No
- PO Price

- User can select the items from the search result, key in the quantity, price & remarks, then click "Add" button to add the items to details of Debit Notes.
- Once user confirms the entry
  - The system checks for missing information and validates the data. If all information is present and valid, the system acknowledges the entry with Debit Note no and stores it into Debit Note data store.
- Additional validation/control
  - The overcharged price cannot greater than PO Price
  - At least one line item required for a Debit Note
  - After month end closing, user cannot do any amendment on Delete Note transaction that generated before that.

## 4.3 Non-Functional Requirement

### 4.3.1 User Friendly

The system should have a user friendly interface because the users may not be computer literate personnel. The system should use the graphical user interface (GUI) approach to provide better understanding on how to use the system. It also serves better communication among system and users. The interfaces should use easy understanding titles and icon or buttons that may help the users to use the system with confidence. The system should provide a simple menu so that the user can obtain an overview of the system. The system should display a friendly error message if errors occur instead of display the professional error page.

### 4.3.2 Reliability

The reliability is the extents to which a system can be expected to perform its intended function with required precision and accuracy. Thus, the system should be reliable on performing each functions and operations. For example, whenever a button is clicked, the system should be able to perform some function or generate same message to inform the user what is happening. This is important to make the users confident with the system they are using.

### 4.3.3 Scalability

The scalability is to promise the capability of the system to migrate a client or server to machines of greater or less power, depending upon requirements with little or no change to underlying components. Database scalability issues can be resolved using distributed

database architecture whereas web application scaling can be addressed by increasing bandwidth or additional web servers.

### 4.3.4 Usability

The system should be developed in such as way that it is easy to use. It will enhance and support rather than limit or restrict the office processes. Human interfaces need to be intuitive and consistent with the eASM System and within itself.

### 4.3.5 Security

The system should be equipped with sufficient security. Each access by the user should be authenticate and validate by the system. The system should not show any potential of leakage of information. The password should be encrypted.

### 4.3.6 Data Backup

The system should be able to restore to its normal operation from any potential disaster. There should be second backup for data to ensure continues of daily operation of the company.

### 4.3.7 Manageability

The modules within the system should be easy to manage. This will make the maintenance, protection of data integrity and enhancement work simpler and not times consuming.

### 4.3.8 Flexibility

The system should have the capability to take advantages of new technologies and resources. The system should be able to implement in the changing environment. Such as when the companies add one more branch, the system should be able to update their data to the latest information with the new branch option in each module.

# Chapter 5

# System Design

# Chapter 5 System Design

## 5.1 Introduction

Systems design is the process or art of defining the hardware and software architecture, components, modules, interfaces, and data for a computer system to satisfy specified requirements. One could see it as the application of systems theory to computing. Some overlap with the discipline of systems analysis appears inevitable.

Prior to the standardization of hardware and software in the 1990s which resulted in the ability to build modular systems, systems design had a more crucial and respected role in the data processing industry. The increasing importance of software running on generic platforms has enhanced the discipline of software engineering at systems design's expense.

A design specification describes the features of the system, the components or elements of the system and their appearance to the user. In this chapter, emphasis will stress on how the system meet the requirement identified during system analysis.

The system is created to incorporate system features that are easy to understand and to prevent improper procedures that will cause the system failure. The functions are built in a manner where it is meaningful to user from various levels. This is a measure to deter user from making error.

## 5.2 Architectural Design

A large system can be decomposed into sub-systems that provide some related set of services. Thus, architectural design is the initial design process of identifying these sub-systems and establishing a framework for sub-system control and communications. The process model for architectural design requires system structuring and modular decomposition.

### 5.2.1 System Structuring

In system structuring, the system is structured into a number of principal sub-systems. A sub-system is a system in its own right, and its operation does not depend on the services provided by other sub-system. Each sub-system is composed of modules and has defined interfaces that are used for communication with other sub-systems.

### 5.2.2 Modular Decomposition

In modular decomposition, each identified sub-system is decomposed into modules. A module is a system component that provides one or more services to other modules. It makes use of services provided by other modules. Modules are usually composed of a number of other simpler system components.

A modular design reduces complexity, facilitates (a critical aspect of system maintainability) and also results in easier implementation of the system because parallel development can be carried out. To achieve a modular design, the following characteristics have to follow:

### 5.2.3 High Cohesion and Low Coupling

A system can consist of many modules. Two modules are highly coupled when there is a great ideal of dependency between them. On the other hand, loosely coupled modules have some dependence, but the interconnection among them is weak.

The goal of system is to keep the degree of coupling as low as possible. If the coupling is loose, then only a few other components will be affected by the change and might be candidates for modification or replacement. But if coupling is high, then large parts of the system may be perturbed by the change.

## 5.3 Database Development

eASm is a relational database model. A relational database is perceived o be a collection of tables on which data are stored. Each table is matrix consisting of series of row and column intersections. Tables are related back to each other by sharing a common entity characteristic.

The relationships between two tables are defined by matching the values in one table to values in another table. In order to create a relationship one or both of the table requires that the values used in the match to be unique. Normally an index (usually the primary key) in the controlling or parent table is used, and a field (called the foreign key) in the child table is matched.

In addition to defining a relationship, the referential integrity rules in the relationship dialog box must be checked. When the referential integrity is enforced, any orphan records in the child table will be ignored.

### 5.3.1 Data Dictionary

**Table Name: tbl_user**

| Column Name | Data Type | Field Length | Field Description |
|---|---|---|---|
| user_login_id | Char | 15 | User unique login ID |
| user_name | Char | 100 | User full name |
| user_pswd | Char | 15 | User account password |
| user_group | Char | 50 | User group name |
| user_role | Char | 100 | User role play |
| user_email | Char | 20 | User email address |
| user_all_records | Bit | 1 | User permission to view all the records<br>1 = True, 0 = False |
| user_change_pswd | Bit | 1 | User permission to change the password<br>1 = True, 0 = False |
| user_fix_pswd | Bit | 1 | User cannot change password<br>1 = True, 0 = False |
| user_never_expired | Bit | 1 | User account will never expired<br>1 = True, 0 = False |
| user_acc_disable | Bit | 1 | User account had been deleted<br>1 = True, 0 = False |

**Table 5.1 User Table**

**Table Name: tbl_customer**

| Column Name | Data Type | Field Length | Field Description |
|---|---|---|---|
| cust_customer_code | Char | 10 | Customer unique auto number assigned to each record in the table |
| cust_name | Char | 100 | Customer full name |
| cust_invoice_address | Char | 500 | Customer invoice mailing address |
| cust_delivery_address | Char | 500 | Customer delivery address |
| cust_phone | Char | 15 | Customer phone number |
| cust_fax | Char | 15 | Customer fax number |
| cust_email | Char | 20 | Customer email address |
| cust_contact_person | Char | 100 | Customer side contact person name |
| cust_contact_person_email | Char | 20 | Customer contact person email address |
| cust_fwd_id | Char | 15 | Forwarder ID number |
| cust_fwd_accno | Char | 15 | Forwarder account number |
| cust_purchaser_incharge | Char | 100 | Purchaser in charge name |
| cust_sales_person_incharge | Char | 100 | Sales person in charge name |
| cust_credit_terms | Char | 50 | Customer credit terms |
| cust_payment_mode | Char | 50 | Customer payment mode e.g. Cash, Credit Card |
| cust_currency_code | Char | 5 | Currency code e.g. RM = Ringgit Malaysia |
| cust_shipment_mode | Char | 50 | Shipment mode |
| cust_shipment_terms | Char | 50 | Shipment terms |
| cust_remarks | Char | 200 | Remarks |

**Table 5.2 Customer Table**

**Table Name: tbl_supplier**

| Column Name | Data Type | Field Length | Field Description |
|---|---|---|---|
| supp_supplier_code | Char | 10 | Supplier unique auto number assigned to each record in the table |
| supp_supplier_name | Char | 100 | Supplier full name |
| supp_address | Char | 500 | Supplier mailing address |
| supp_phone | Char | 15 | Supplier phone number |
| supp_fax | Char | 15 | Supplier fax number |
| supp_email | Char | 20 | Supplier email address |
| supp_contact_person | Char | 100 | Supplier side contact person name |
| supp_contact_person_email | Char | 20 | Supplier contact person email address |
| supp_credit_terms | Char | 50 | Supplier credit term e.g. Cash, Credit Card |
| supp_credit_limit | Numeric | 20 | Supplier credit limitation |
| supp_currency_code | Char | 5 | Currency code e.g. RM = Ringgit Malaysia |
| supp_payment_mode | Char | 50 | Supplier payment mode e.g. Cash, Credit Card |
| supp_shipment_mode | Char | 50 | Shipment mode |
| supp_shipment_terms | Char | 50 | Shipment terms |
| supp_remarks | Char | 200 | Remarks |

**Table 5.3 Supplier Table**

**Table Name: tbl_currency**

| Column Name | Data Type | Field Length | Field Description |
|---|---|---|---|
| curr_currency_code | Char | 5 | Currency code<br>e.g. RM, USD |
| curr_currency_name | Char | 20 | Currency name<br>e.g. RM = Ringgit Malaysia,<br>USD = US Dollar |

**Table 5.4 Currency Table**

**Table Name: tbl_item_transfer**

| Column Name | Data Type | Field Length | Field Description |
|---|---|---|---|
| itra_no | varchar | 15 | Item transfer unique auto number assigned to each record in the table |
| itra_gen_date | datetime | 8 | Item transfer auto generated date |
| itra_gen_by | varchar | 80 | Name of person who generate the record |
| itra_branch | varchar | 80 | Branch code for the company branch |
| itra_trans_type | varchar | 50 | Transfer In/Out |
| itra_ref_no | varchar | 100 | Reference number for transfer in intem |
| itra_from | varchar | 80 | Branch name |
| itra_to | varchar | 80 | Branch name |
| itra_approved_by | varchar | 80 | Approved person name |
| itra_reason | varchar | 200 | Reason for transfer item |
| itra_remarks | varchar | 200 | Remarks |

**Table 5.5 Item Transfer Table**

**Table Name: tbl_transfer_details**

| Column Name | Data Type | Field Length | Field Description |
|---|---|---|---|
| itde_no | Char | 10 | Item transfer unique auto number assigned to each record in the table |
| itde_item_code | Date | 15 | Item code number that retrieve from table tbl_item |
| itde_desc | Char | 10 | Description of the particular tem |
| itde_uom | Char | 10 | Item unit of measurement |
| itde_on_hand | Date | 15 | Item unit on hand |
| itde_qty | Char | 5 | Item quantity that wish to transfer |
| itde_unit_price | Numeric | 20 | Goods target price |
| itde_remarks | Int | 10 | Remarks |

**Table 5.6 Item Transfer Details Table**

## 5.4 Design Specification

### 5.4.1 Structure Chart

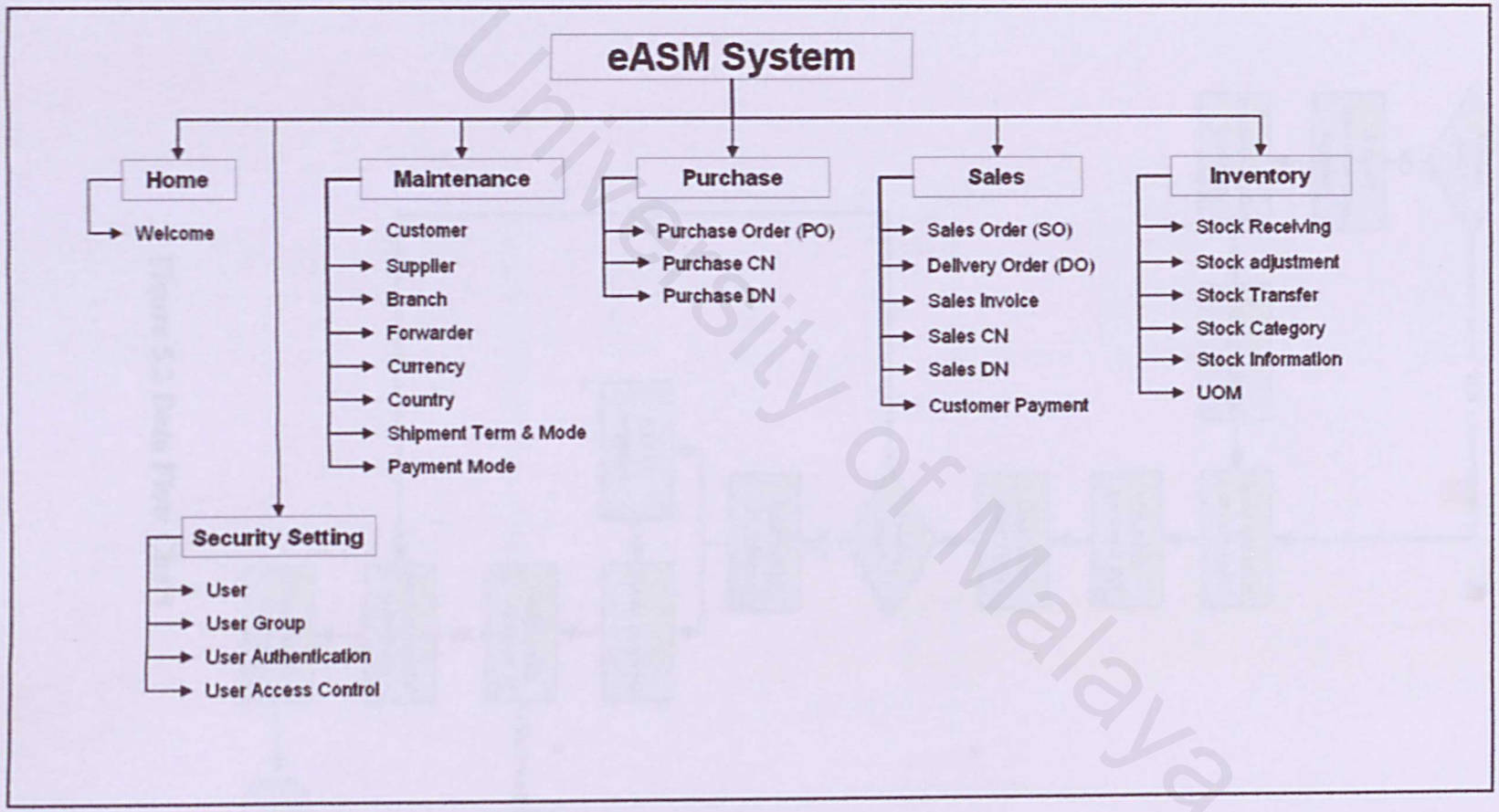



Figure 5.1 Structure Chart

### 5.4.2 Data Flow Chart

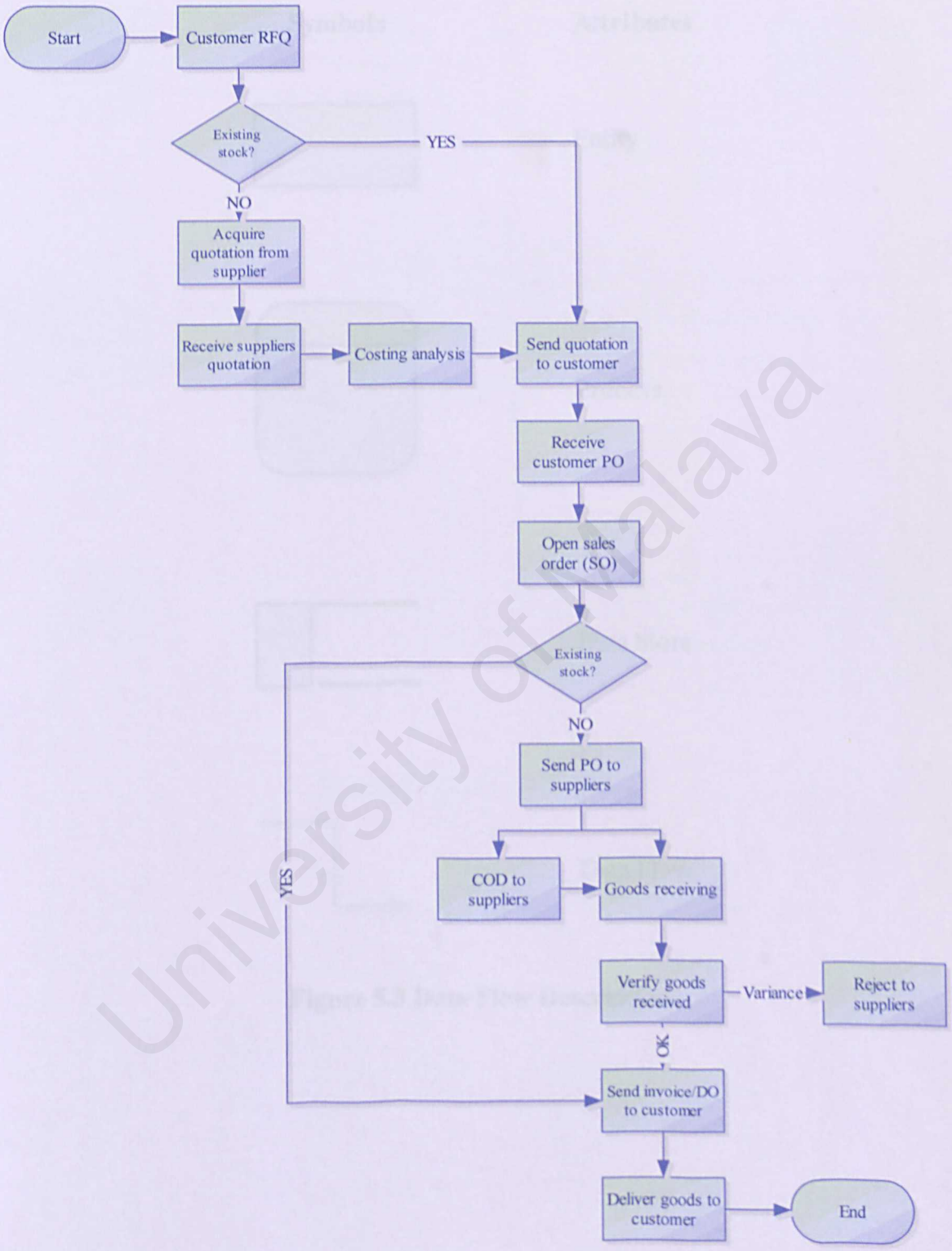



**Figure 5.2 Data Flow Chart**

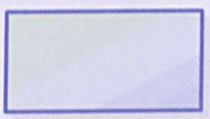
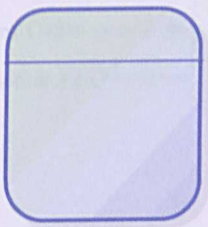
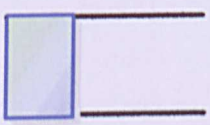
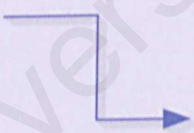
| Symbols | Attributes |
| --- | --- |
|  | Entity |
|  | Process |
|  | Data Store |
|  | Data Flow |

**Figure 5.3 Data Flow Description**

### 5.4.3 Data Flow Diagram (DFD)

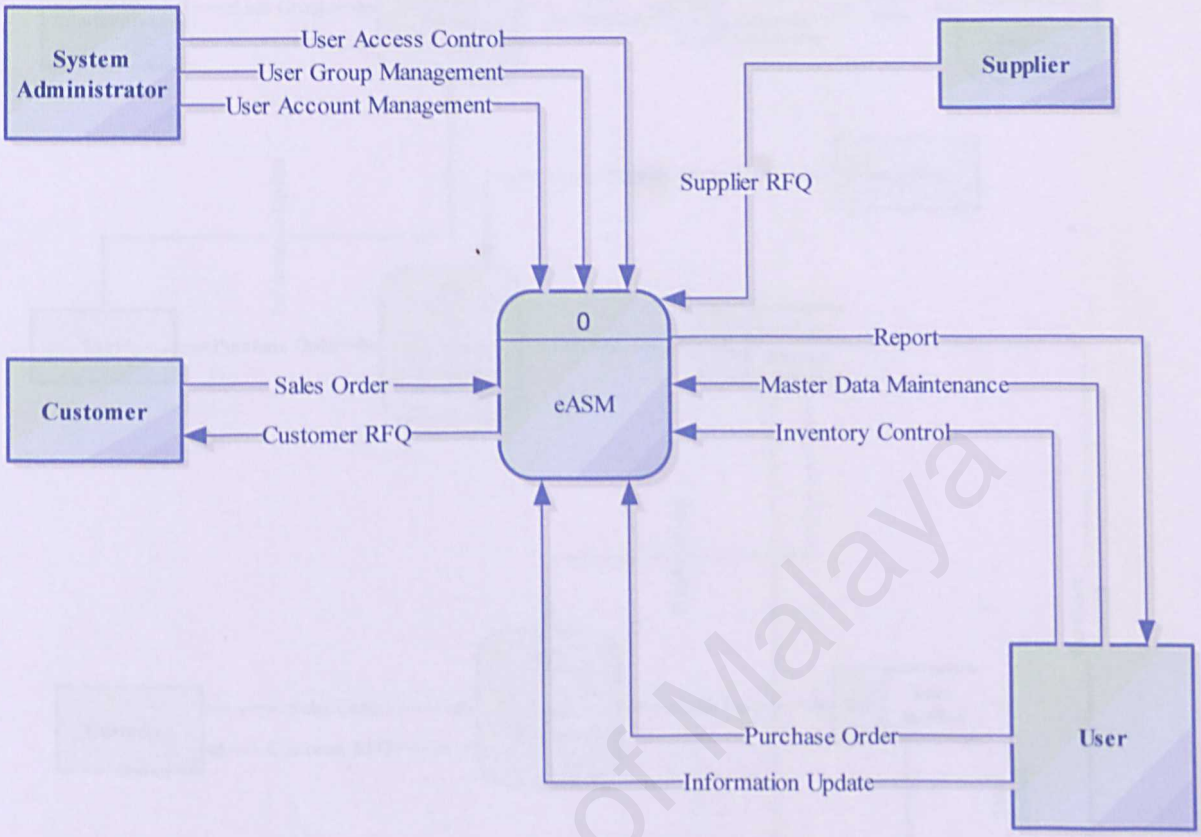



**Figure 5.4 Context Diagram**

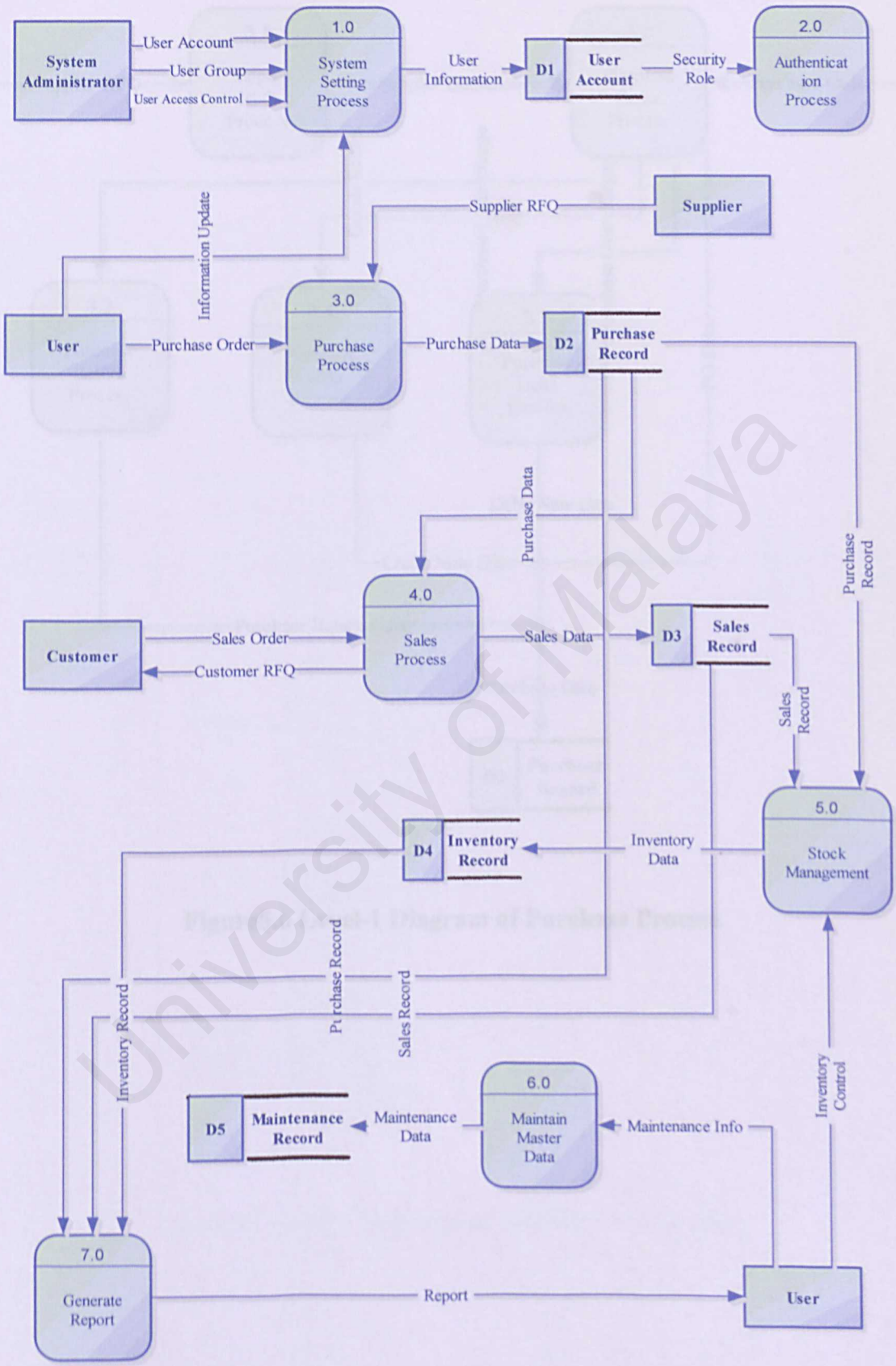



**Figure 5.5 Level-0 Diagram**

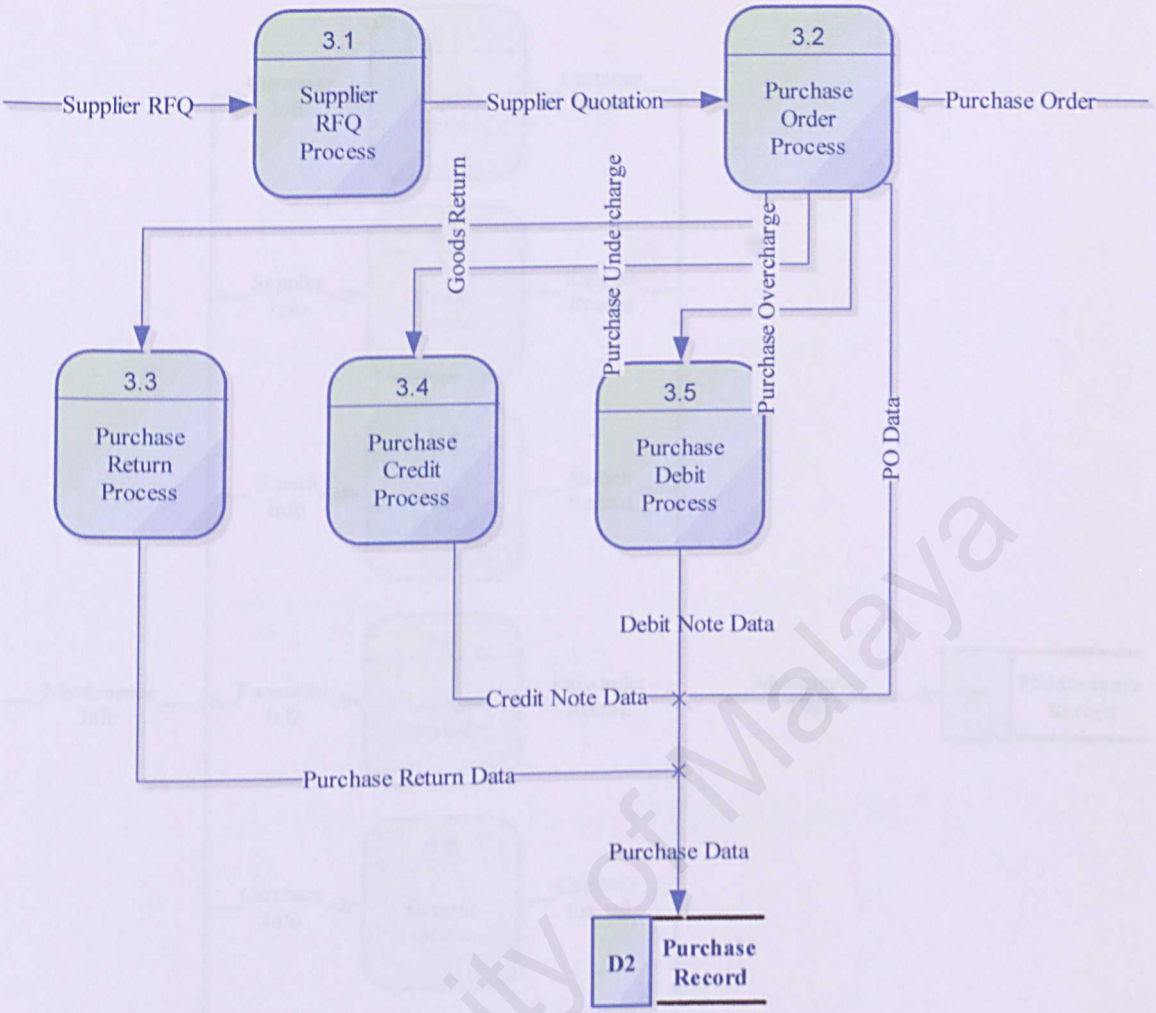



**Figure5.6 Level-1 Diagram of Purchase Process**

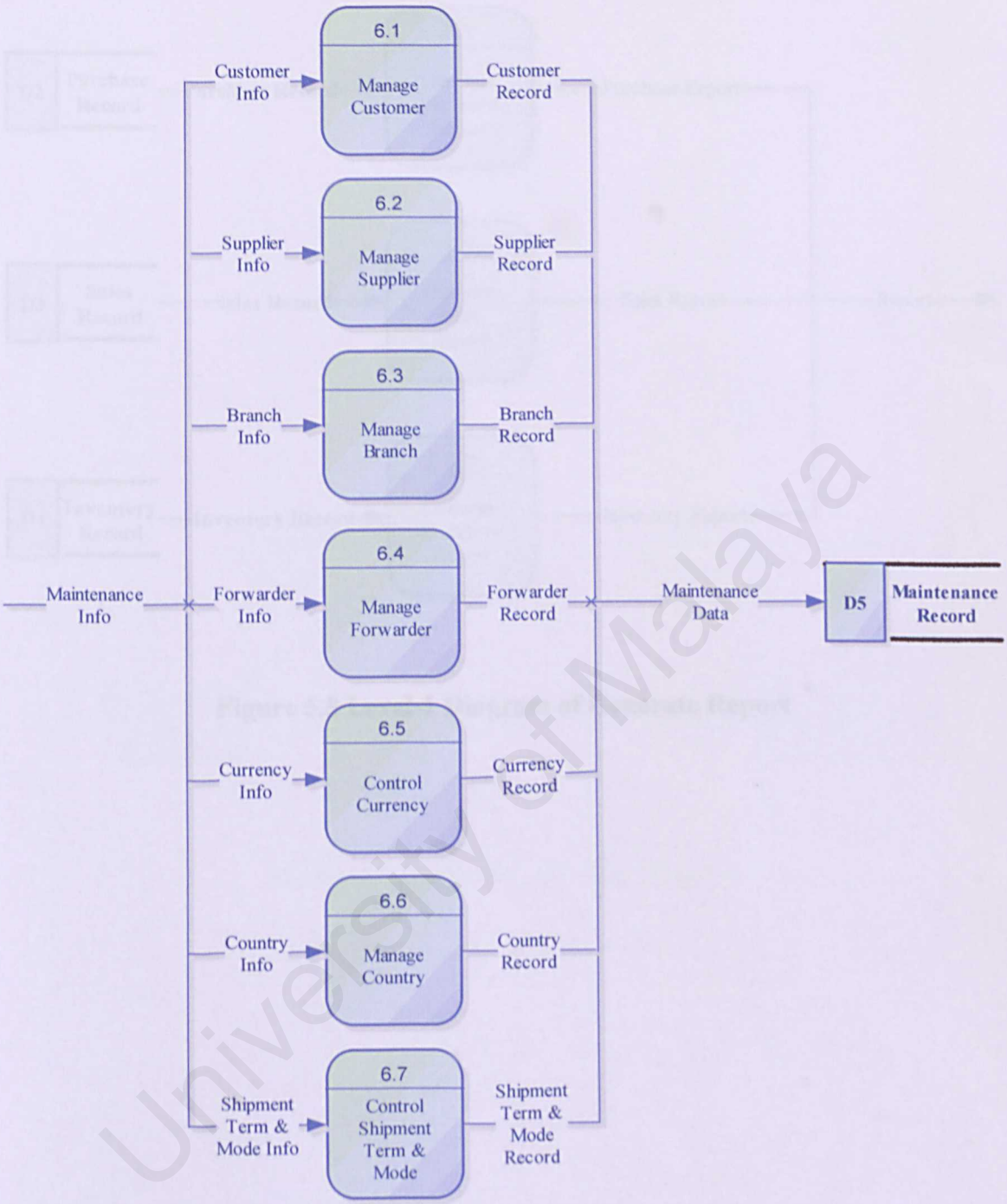



**Figure5.7 Level-1 Diagram of Maintain Master Data**

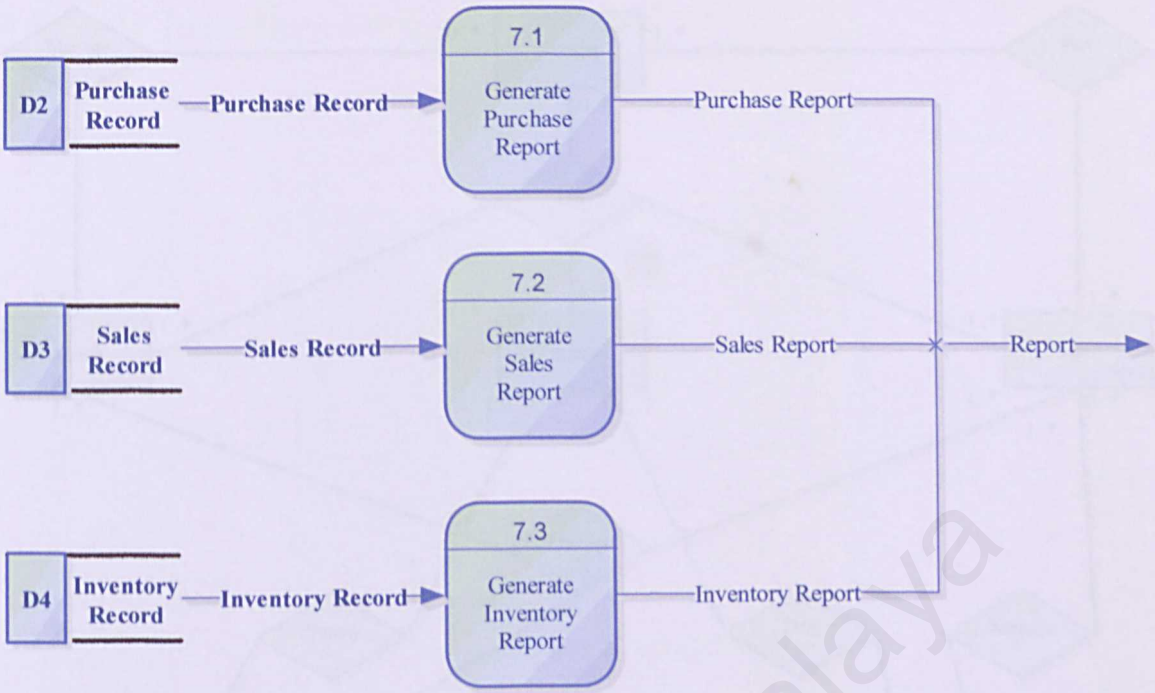



**Figure 5.8 Level-1 Diagram of Generate Report**

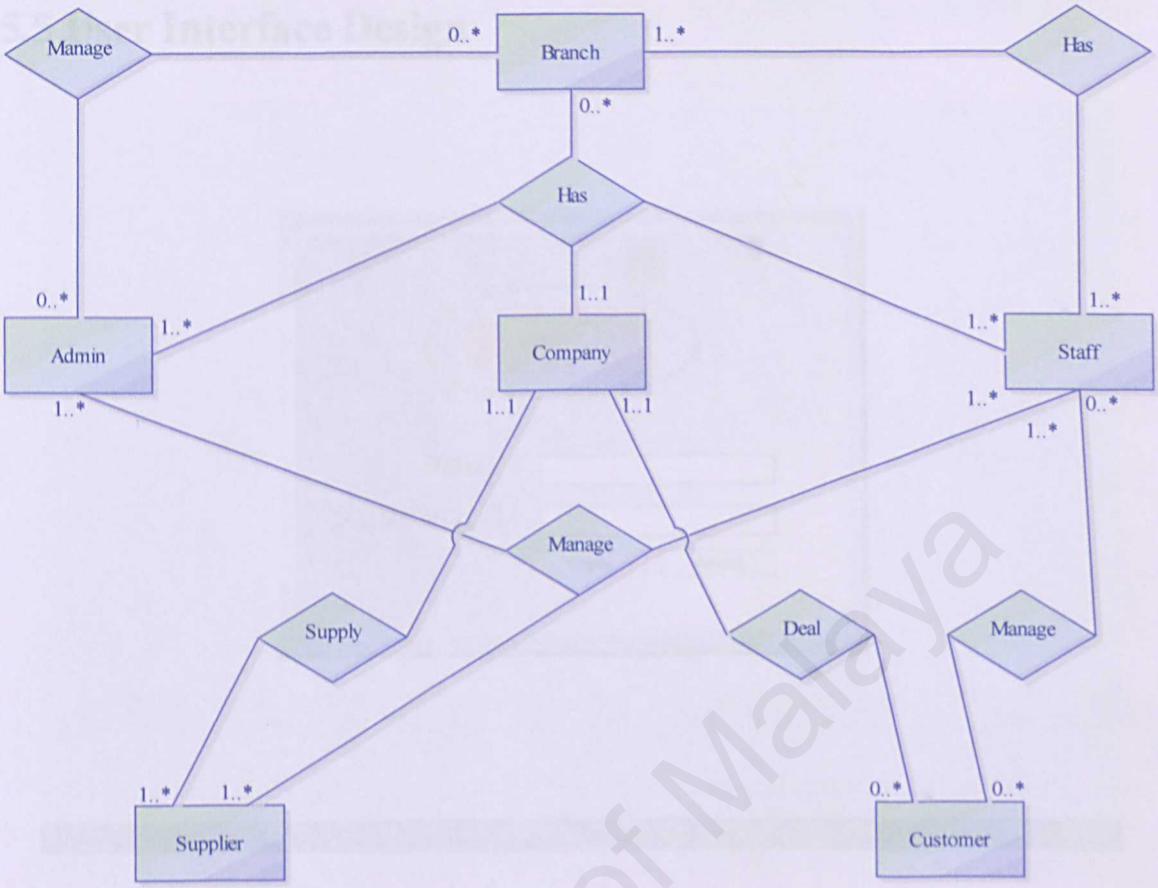



**Figure 5.9 Entity Relationship Diagram**

## 5.5 User Interface Design


**Figure 5.10 Login Page** – Admin and user login interface.

Only authenticate users may login to the system.


**Figure5.11 Home Page** – After user login to the system, welcome page will be shown. User can choose the module from the toolbar in the window (in red colour bracket).

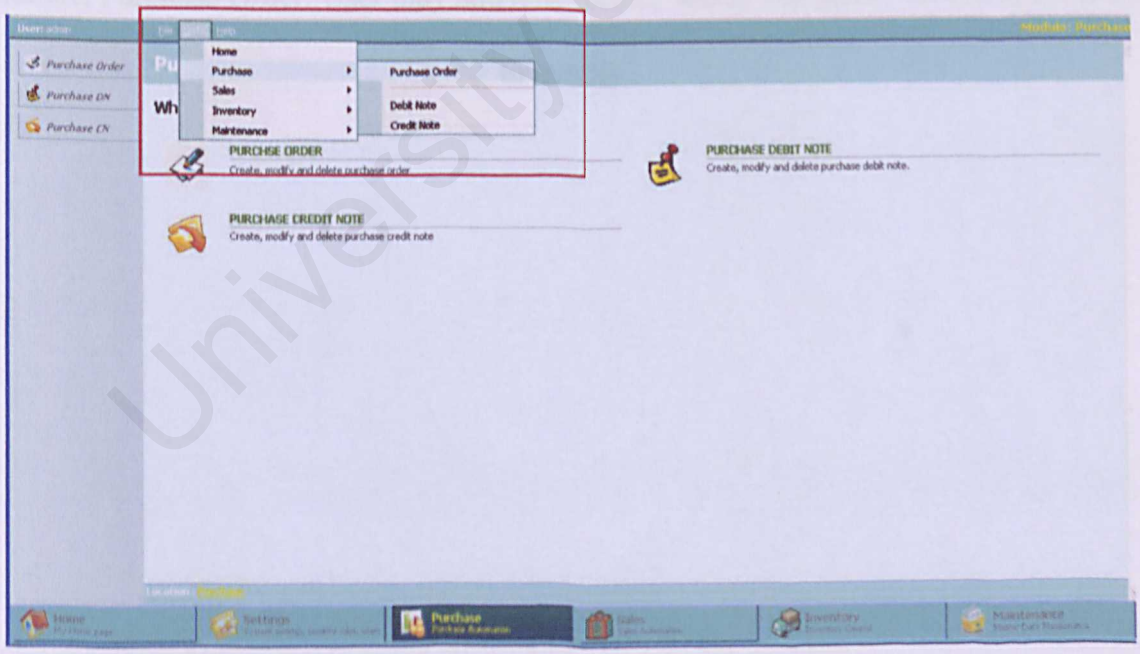



**Figure 5.12 Purchase Home Page** – User also may choose the module or sub-module from the menu (in red colour bracket) at the top of the windows.

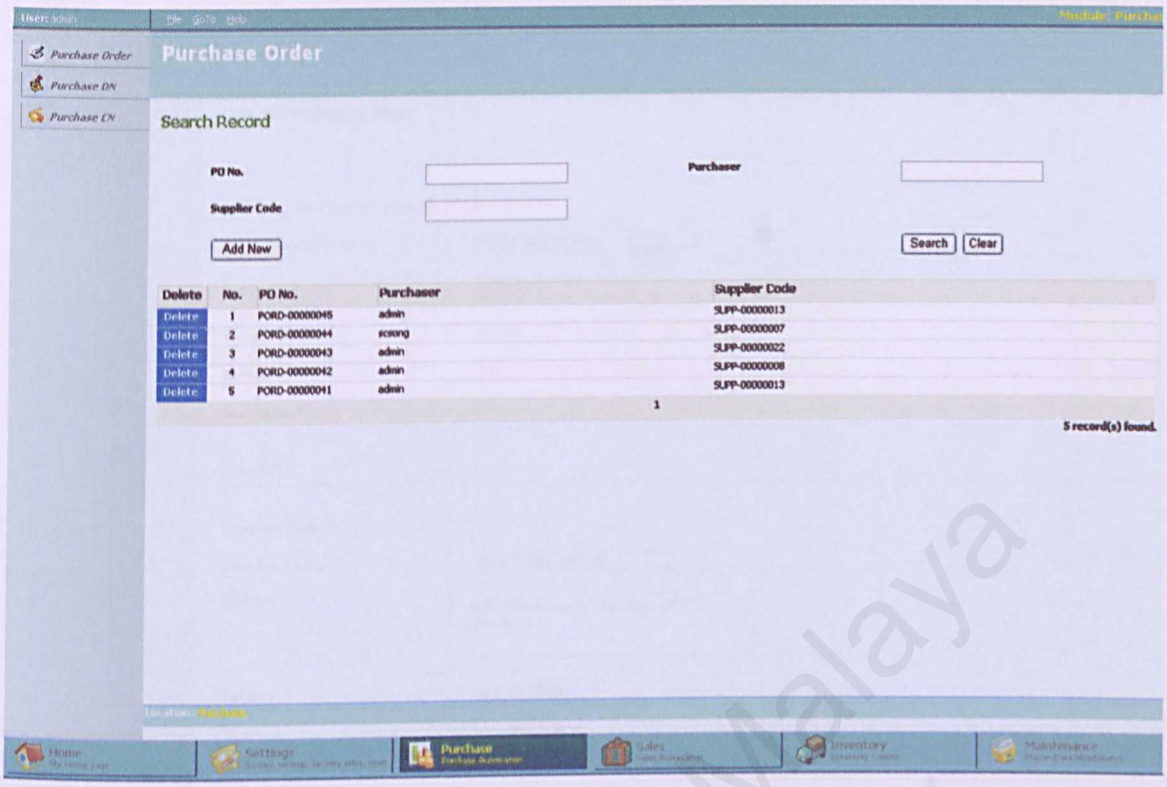



**Figure 5.13 Purchase Order Home Page** – Home page for one of the Purchase sub-module, Purchase Order. User may add new record, search and select the existing record to modify.

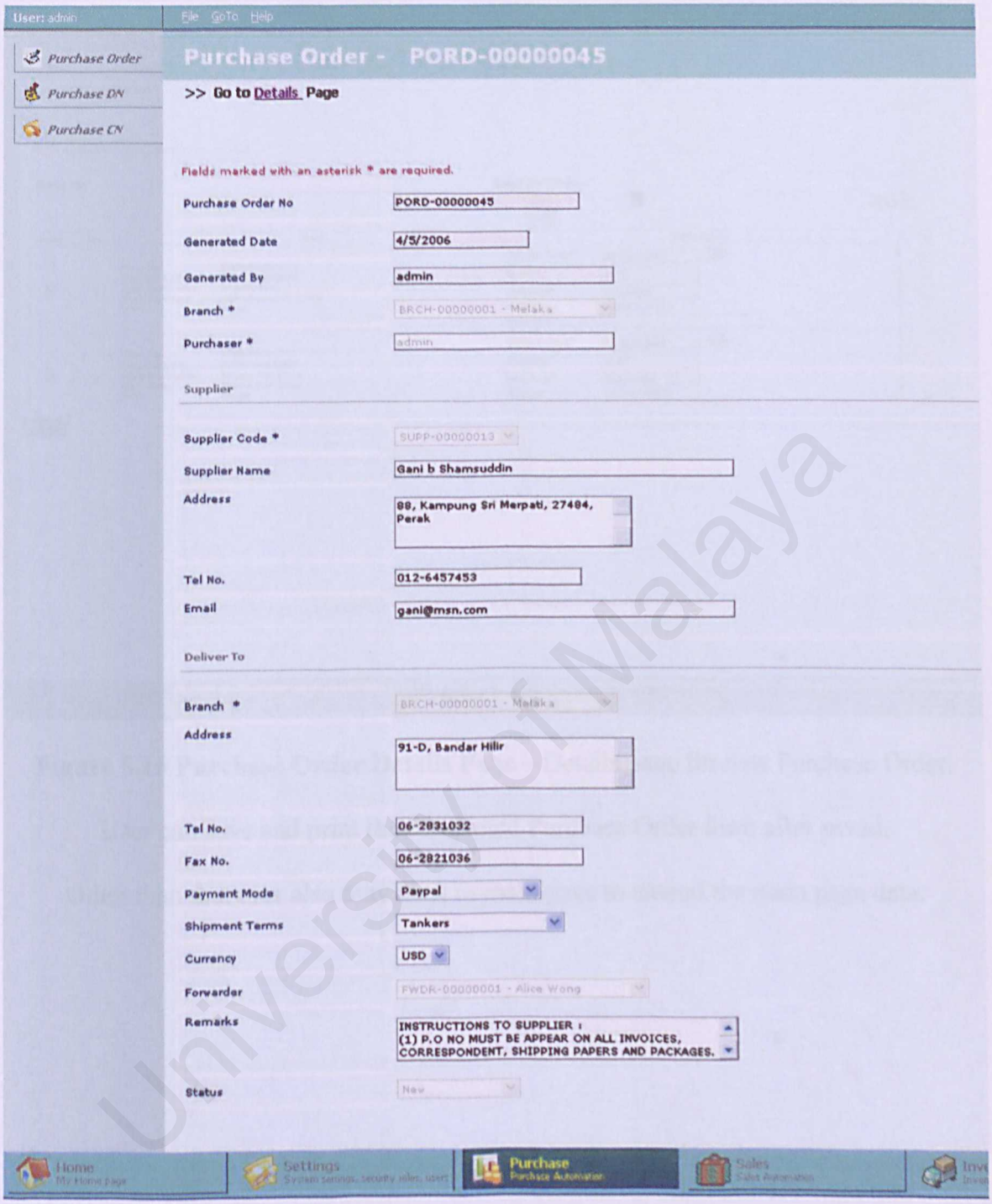



**Figure 5.14 Purchase Order Main Page** – Main page for the new Purchase Order.

User has to fill in the entire mandatory field before save or go to details page.

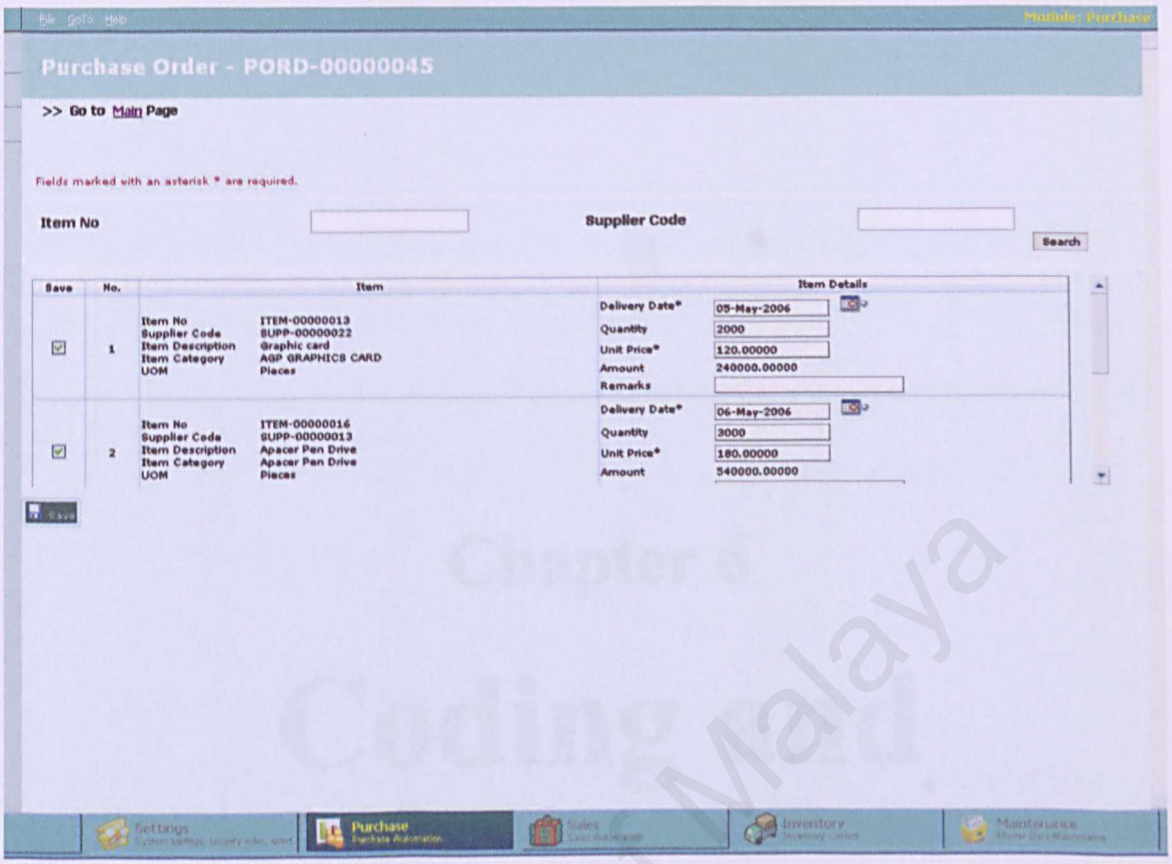



**Figure 5.15 Purchase Order Details Page** – Details page for new Purchase Order.

User can save and print the completed Purchase Order form after saved.

Other than that user also may back to main page to amend the main page data.

# Chapter 6

# Coding and Implementation

# Chapter 6 Coding and Implementation

## 6.1 Coding Approach

Coding design is an important approach to ensure an efficient final implementation. Section 6.1.1 is the description of the coding approach that can be used. eASM was developed using Bottom-up approach. Top-down approach is not recommended because all decisions depend on the starting goal of the project, and some cannot be made depending on how specific that description is.

### 6.1.1 Top-down and Bottom-up design

Top-down and bottom-up are strategies of information processing, mostly involving software, and by extension other humanistic and scientific System Sciences.

In the top-down model an overview of the system is formulated, without going into detail for any part of it. Each part of the system is then refined by designing it in more detail. Each new part may then be refined again, defining it in yet more detail until the entire specification is detailed enough to validate the model. The top-down model is often designed with the assistance of "dark boxes" that make it easier to bring to fulfillment but insufficient and irrelevant in understanding the elementary mechanisms.

By contrast in bottom-up design individual parts of the system are specified in detail. The parts are then linked together to form larger components, which are in turn linked until a complete system is formed. Strategies based on his bottom-up information flow

seem potentially necessary and sufficient because they are based on the knowledge of all variables that may affect the elements of the system.

Bottom-up design had been chosen to develop eASM system. This is because the lower level functions have to developed and proceed before moving to the higher level modules. Settings and Maintenance module are developed in the lower level. When the lower layers are ready, then the upper layer module like Inventory, Purchase and Sales modules just can proceed.

### 6.1.2 Coding Style

Coding style is an important attribute of source code and determines the efficiency of a system. An easy to read the source code makes the system easier to maintain and enhance. The elements of coding style include internal documentation and approach to statement construction.

Most of the coding inside eASM system is dealing with the SQL database to insert, retrieve, and update information using form. So the communications with SQL database are needed. eAsM application layer will communicate with business tier, and the business tier will call the database tier to communicate with the SQL database.

Internal documentation provides a clear guide during the maintenance phase of the system. Comments provide the developers a means of communicating with other readers of the source code. Statement of purpose and descriptive comment that is embedded with the body of the source code is needed to describe processing functions.

## 6.2 Implementation

System implementation is a process of developing a system based on a requirement that get in the analysis phase. It converts the system design into program code.

### 6.2.1 Web Page Development

Coding in the system divided into 3 + 1 project. The 3 main projects are the 3-tier architecture which is application tier, business tier and database tier. And the + 1 project is the common utilities project. The first project is eASM application (presentation layer) project. This project include with HTML code, JavaScript, ASP code, code behind (VB code), XML (Extensible Markup Language) and CSS (Cascading Style Sheet) document.

The second project is the business tier which named by "BizTier". Business tier is the middleman between application layer and database tier. Its provide services to the application layer with get all the database information/data that request by the application layer which is web browser as well. Other than that BizTier also send the entire data that will insert or update to the database to the database tier.

The third project is the database tier called "DbTier". DbTier is directly connecting with the SQL database. The functions that operate in the DbTier include retrieve database information/data, insert new raw data, update data, and delete data. The data retrieve from the database will be store as SQL data reader or dataset. Then the entire data will be use at the application layer and display as a text to the end user.

The + 1 common utilities project is to store the common used function instead of write the code each time developer need to use it. For example, convert to null function will be use in the most of the module. This function is to convert the value to null if the value is nothing.

### 6.2.2 Web Server Used

The web server used for this system is Microsoft Internet Information Server (IIS). IIS is tightly integrated with the Windows NT server operating system. IIS is chosen the web browser for this project because it supports an application environment called Active Server Page dot net (ASP .net). Under this environment, HTML code, scripting language and Active X components can be combine to create powerful web application. Besides, IIS is easy to manage, fast, secure, and can be up and down in a minutes because it has tight relationship with Windows NT server.

The web page serves as an interface between the user and the system. When user sent a request to the system, the system will response and complete the operation in the server before sent back the information to the client browser. All the response and request between the end user and server is implemented using server scripting architecture. The server scripting is run at server and put as the HTML format before it is transmitted to the client server.

# Chapter 7

# Testing and Evaluation

# Chapter 7 Testing and Evaluation

## 7.1 Introduction

After eASM has been successfully implemented, it will be tested for bugs and errors to identify its level of functionality. Testing is very important to ensure that the implemented system is executed correctly and conforms to the requirement specified during system analysis and design phase. The eASM system must be executes by running through internet browser.

There are two techniques applied in testing eASM system which is white box testing and black box testing. For a complete system examination, both white box and black box tests are required.

## 7.2 White Box Testing

Also known as glass box, structural, clear box and open box testing. System testing technique whereby explicit knowledge of the internal workings of the item being tested are used to select the test data. Unlike black box testing, white box testing uses specific knowledge of programming code to examine outputs. The test is accurate only if the tester knows what the system is supposed to do. We can then see if the system diverges from its intended goal. White box testing does not account for errors caused by omission, and all visible code must also be readable.

The purpose of white box testing is to:

- Initiate a strategic initiative to build quality throughout the life cycle of a system.

- Provide a complementary function to black box testing.
- Perform complete coverage at the component level.
- Improve quality by optimizing performance.

### 7.2.1 Testing Analysis - Identification

- The identification of the test items is done primarily based on the specifications of the system. These specifications would be related to:
- Functions (exhaustive list) of the system
- Response criteria (benchmarking and stress testing)
- Volume constraints (number of users, hits, stress testing)
- Stability criteria (24 hour testing with fast operations)
- Database responses (flushing, cleaning, updating rates etc.)
- Compatibility (environments, browsers, etc.)
- User Interface / Friendliness Criteria
- Modularity (Ability to easily interface with other tools)
- Security

### 7.2.2 Types of White Box testing

- Static and Dynamic Analysis – Static analysis techniques do not necessitate the execution of the software, dynamic analysis is what is generally considered as ``testing``, i.e. it involves running the system.
- Statement Coverage– Testing performed where every statements is executed at least once.

- Branch Coverage – Running a series of tests to ensure that all branches are tested at least once.
- Path Coverage – Testing all paths.
- All-definition-use-path Coverage – All paths between the definition of a variable and the use of that definition are now identified and tested.

## 7.3 Black Box Testing

Black box test also known as functional testing. Black box testing is testing that occurs from the viewpoint of an end user. Black box tests find bugs such as incorrect functions, interface problems, and database errors.

Black box tests are also the only form of test the customer is likely to understand; mention "basis path testing", and you'll have him frowning. Therefore, black box testing is absolutely mandatory for acceptance testing. The customer must be able to understand these tests, so that he/she will know for sure whether or not you've met the contract requirements.

The advantages of this type of testing include:

- The test is unbiased because the designer and the tester are independent of each other.
- The tester does not need knowledge of any specific programming languages.
- The test is done from the point of view of the user, not the designer.
- Test cases can be designed as soon as the specifications are complete.

## 7.4 Unit Testing

A unit test is a procedure used to validate that a particular module of source code is working properly. The idea about unit tests is to write test cases for all functions and methods so that whenever a change causes a regression, it can be quickly identified and fixed. Ideally, each test case is separate from the others; constructs such as mock objects can assist in separating unit tests. This type of testing is mostly done by the developers and not by end-users.

The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. Unit testing provides a strict, written contract that the piece of code must satisfy. As a result, it affords several benefits.

## 7.5 Integrated Testing

Integration testing is the phase of system testing in which individual system modules are combined and tested as a group. It follows unit testing and precedes system testing. Integration testing takes as its input modules that have been checked out by unit testing, groups them in larger aggregates, applies tests defined in an Integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

The purpose of Integration testing is to verify functional, performance and reliability requirements placed on major design items. These "design items", are exercised through their interfaces using Black box testing, success and error cases being simulated via appropriate parameter and data inputs. Simulated usage of shared data areas and inter-

process communication is tested; individual subsystems are exercised through their input interface. All test cases are constructed to test that all components within assemblages interact correctly.

## 7.6 System Testing

System testing is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of Black box testing, and as such, should require no knowledge of the inner design of the code or logic.

Alpha testing and Beta testing are sub-categories of System testing.

As a rule, System testing takes, as its input, all of the "integrated" software components that have successfully passed Integration testing and also the software system itself integrated with any applicable hardware systems. The purpose of Integration testing is to detect any inconsistencies between the software units that are integrated together called assemblages or between any of the assemblages and hardware. System testing is more of a limiting type of testing, where it seeks to detect both defects within the "inter-assemblages" and also the system as a whole.

### 7.6.1 Types of system tests

The following are different types of testing that should be considered during System testing:

- Functional testing
- User interface testing

- Usability testing
- Compatibility testing
- Security testing
- Performance testing
- Reliability testing
- Recovery testing
- Installation testing
- Maintenance testing
- Accessibility testing

## 7.6 Test Protocol (Test Plan)

**Test Protocol**

**Project: eASM**
**Module: Settings - User Account**

**Release Type: New**
**Date: 3-March-2006**

Error
Enhancement
Solve by programmer
OK Bugs free

| Test Case | Test Environment | Expected Result | | Tested Result | Remarks |
|---|---|---|---|---|---|
| Search Page | Query | Fields - User Name, Role, Login ID, Email | | Sort the record by User Name, Ascending | |
| | Delete | Delete record | | Delete button not function | |
| | Clear | Clear search data | OK | | |
| | Paging links | Show page number | OK | | |
| | Add New User | Click "Add New" button to go to create new user screen | OK | | |
| | | | | | |
| Add New User & Update User Info | User Name | Mandatory text field, TEXT(80) | | No duplicated Login ID is allowed | |
| | User Group | Mandatory Dropdown selection | OK | | Select from User Group database |
| | Role | Text field, TEXT(80) | | Not readonly text box. User may key in themselves | |
| | Login ID | Mandatory text field, TEXT(80) | OK | | |
| | Password | Mandatory password text field, TEXT(15) | OK | | |
| | Email | Text field, TEXT(80) | OK | Check the validity of the email address by '@' and '.' | |
| | Authorized access all users records | Check box | OK | | |
| | Change password | Check box | OK | | Disable when user tick the 'Never Expired' and 'Fixed Password' check box |
| | Never expired | Check box | OK | | |
| | Fixed password | Check box | OK | | |
| | Account disabled | Check box | OK | | |
| | Required Fields | | | No Asterisk mark for required fields | |

**Table 7.1 User Account module Test Protocol example**

# Test Protocol

**Project: eASM**
**Module: Inventory - Item Transfer**

**Release Type: New**
**Date: 5-March-2006**

Error
Enhancement
Solve by programmer
OK Bugs free

| Test Case | Test Environment | Expected Result | | Tested Result | Remarks |
|---|---|---|---|---|---|
| Search Page | Query | Fields - Item Transfer No., Generated Date, Branch, Transfer Type | | Sort the record by Item Transfer No., Descending | |
| | Delete | Delete record | | Delete button not function | |
| | Clear | Clear search data | OK | | |
| | Paging links | Show page number | OK | | |
| | Add New User | Click "Add New" button to go to create new user screen | OK | | |
| | | | | | |
| Main | Item Transer No | ReadOnly text field, TEXT(15) | OK | | Auto generate Item Transfer No after user save the record |
| | Generated Date | ReadOnly text field, TEXT(15) | OK | | Auto generate today date in date format 'DD-MMM-YYYY' for new record |
| | Generated By | ReadOnly text field, TEXT(80) | | Auto generate User ID from the session("username") for new record and not key in manually by user | |
| | Branch | Mandatory Dropdown selection | OK | | |
| | Transfer Type | Dropdown selection | OK | | |
| | Ref No | Dropdown selection | OK | | Show when choose 'Transfer In'. Data get from the transfer out item record |
| | TransferFrom Location | Dropdown selection | OK | | Show branch name when choose 'Transer Out' |
| | Transfer To Location | Dropdown selection | OK | | |
| | Approved By | Mandatory text field, TEXT(80) | | No maximum text length control | |
| | Reason for Transfer | Mandatory text area, TEXT(200) | | No maximum text length control | |
| | Remarks | Text area, TEXT(200) | | No Asterisk mark for required fields | |
| | Required Fields | | OK | | Check the validation of each |
| | Details | Button, Navigate to details page | | | |
| | | | OK | Type * to get all item | To use for filter item. |
| Details | Item No | Text field, TEXT | OK | Type * to get all item | |
| | Item Description | Text field, TEXT | | Sort the record by item no descending | |
| | Data Grid | Data grid | | Prompt the message when user did not select any data to save | |
| | Save | Check box | | No validation for numeric | |
| | Quantity | Numeric data control, TEXT(20) | | No validation for numeric | |
| | Unit Price | Numeric data control, TEXT(20) | OK | | |
| | Total Amount | Text field, TEXT | OK | | |
| | Remarks | Text field, TEXT(40) | OK | | |
| | Save button | Check box | OK | | |
| | Main | Button, Navigate to main page | | No Asterisk mark for required fields | |
| | Required Fields | | | | |

**Table 7.2 Item Transfer module Test Protocol example**

# Chapter 8

# System Evaluation

# Chapter 8 System Evaluation

## 8.1 Introduction

System evaluation requires balancing of many factors. Below are the factors that should be considered:

- Costs of hardware and software - will vary despite identical functionality
- Speed and capacity of hardware
- Quality and costs of system
- Position in the marketplace
- Reports from other users about quality of system
- References are a useful way of obtaining this information
- Appropriate customer references should be supplied by each vendor

### 8.1.1 The Evaluation Procedure

The following criteria were kept in mind while considering the usefulness of eASM.

- Simplification & Ease of Use
    - Does the eASM make it easier for end user to insert, update, and delete information to the system?
        - Answer: Yes, user may use the system easily just only use the muse click and keyboard and without the deep knowledge of an accounting.
    - Are the functions and applications easy for end user to learn and use?
        - Answer: Yes, there will be an user manual for the user to refer and it contain the steps how to use the whole system.
- Flexibility

- Is the eASM ability to use in different environment such as Windows 98, Windows NT, Linux or UNIX?
  - Answer: eASM can be browse through the internet as long as user have web browser. But it is more on the Windows OS.
- Can end user easily retrieve their record from anywhere, anyplace?
  - Answer: Yes, of course!
- Does the eASM support a range of databases?
  - Answer: Yes, because eASM is using Microsoft SQL server database so it can support for a big range of database.

- Features & Functionality
  - What additional features does the eASM provide?
    - Answer: eASM can generate the payment, invoice and etc automatically.
  - What are the special functions that eASM support and can attract user to use it?
    - Answer: eASM is user friendly compare to the accounting system that available in the market nowadays. The important thing is eASM is a web base application and can support wide range of users.

## 8.2 Future Enhancement

### 8.2.1 More Functionality

eASM can be enhance in the future with some additional functions. Function that possible to add in the eASM is the monthly report function. System can auto calculate and summarize the entire raw data of the particular month. User can get the monthly inventory, purchase and sales report just by click on the mouse and the system will auto generate the report and display it as a .PDF file to the user. In the report the figure of the item can be summarize and display as a pie chart, bar chart or the line chart. It will be easier for user to read the data instead of display the numbers.

Other than that, eASM also can add on with the time attendance and payroll module. With these two new modules, the system can trace the employee attendance and count the salary for each employee easily. Therefore it can save the manpower and cut down the time consuming.

### 8.2.2 More Secure

eASM system can be more secure if the encryption functions implement on each module. To not make the system become slower when the data is encrypt or decrypt, the cryptographic technique can just implement to the record ID only. Others data are not necessary to encrypt. If the record ID was encrypted, power user may not be able simply change the record information or data from the SQL database. Therefore, the entire system is safe and secure and also precise.

## 8.3 Conclusion

eASM system has been completed successfully with some strengths as well as limitation as mentioned. It is target for the medium size companies that wish to manage their accounting and stock management system online. But it is also can be used offline as well. It is an online system which is designed to runs in Windows platform by using ASP .net programming language, XML, JavaScript, and Report Manager.

eASM is a low cost system development which is easy to use and learn. It can easily modify and has a wide potential of usage. Other than that, eASM also can maintain easily by other developer as long as they have the knowledge to use the tools because all of the source code are written in tidily. Enhancement with the encryption of the user account password can prevent the account gain by the unauthorized user or hacker get into the system illegally.

The development tools such as ASP .net and Microsoft SQL server have potential to grow further in the future. ASP.NET is not just a simple upgrade or the latest version of ASP. ASP.NET combines unprecedented developer productivity with performance, reliability, and deployment. ASP.NET redesigns the whole process. It's still easy to grasp for new comers but it provides many new ways of managing projects. ASP .net also is an easy programming model and it is a flexible language options.

Finally, eASM has achieved the system objective define during the analysis stage and fulfill all the functional and non-functional requirements.

# Appendix A

# User Manual

# Appendix A. User Manual

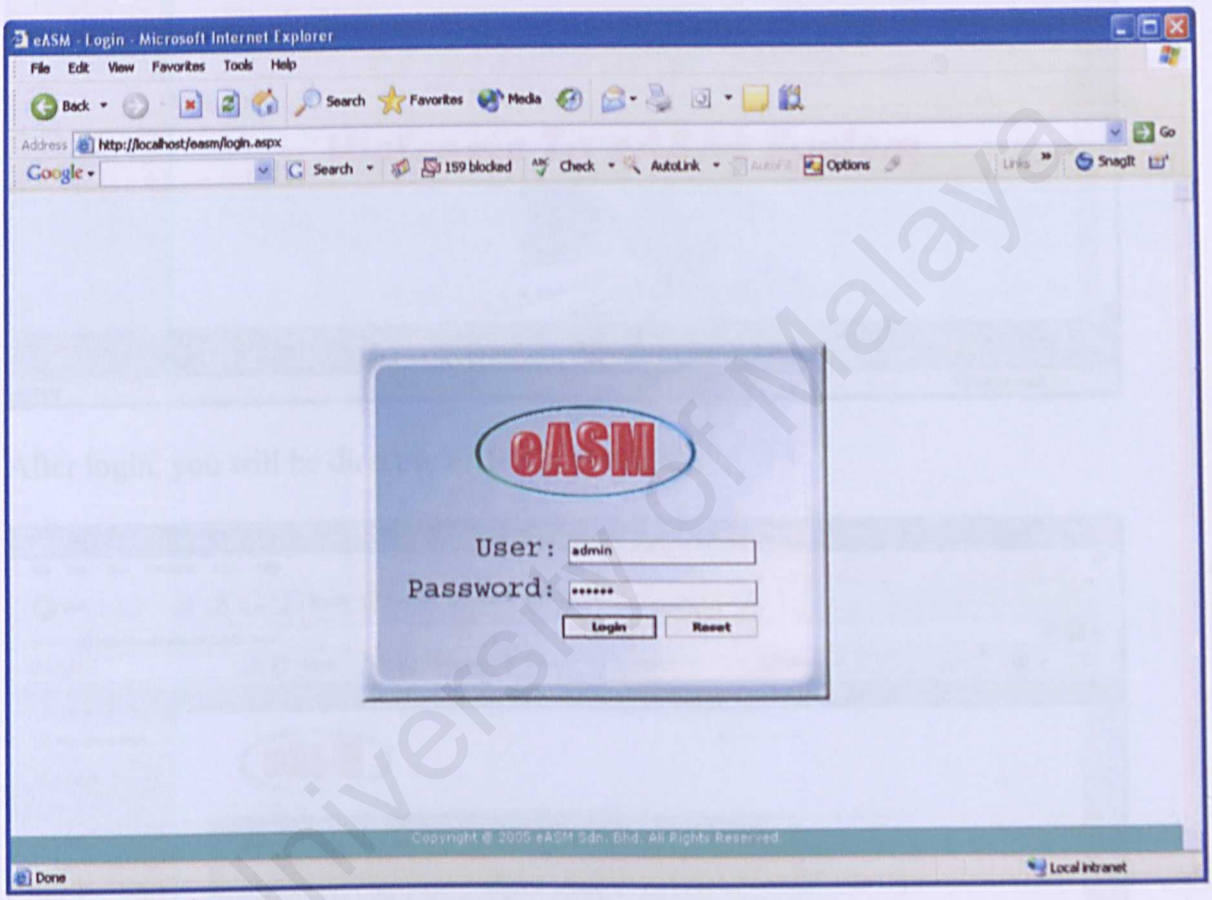
## Introduction

E Accounting & Stock Management (eASM) system is an online accounting system that makes accounting more effectively and systematically.

This user manual gives guidance to system user in the tasks accomplishment.

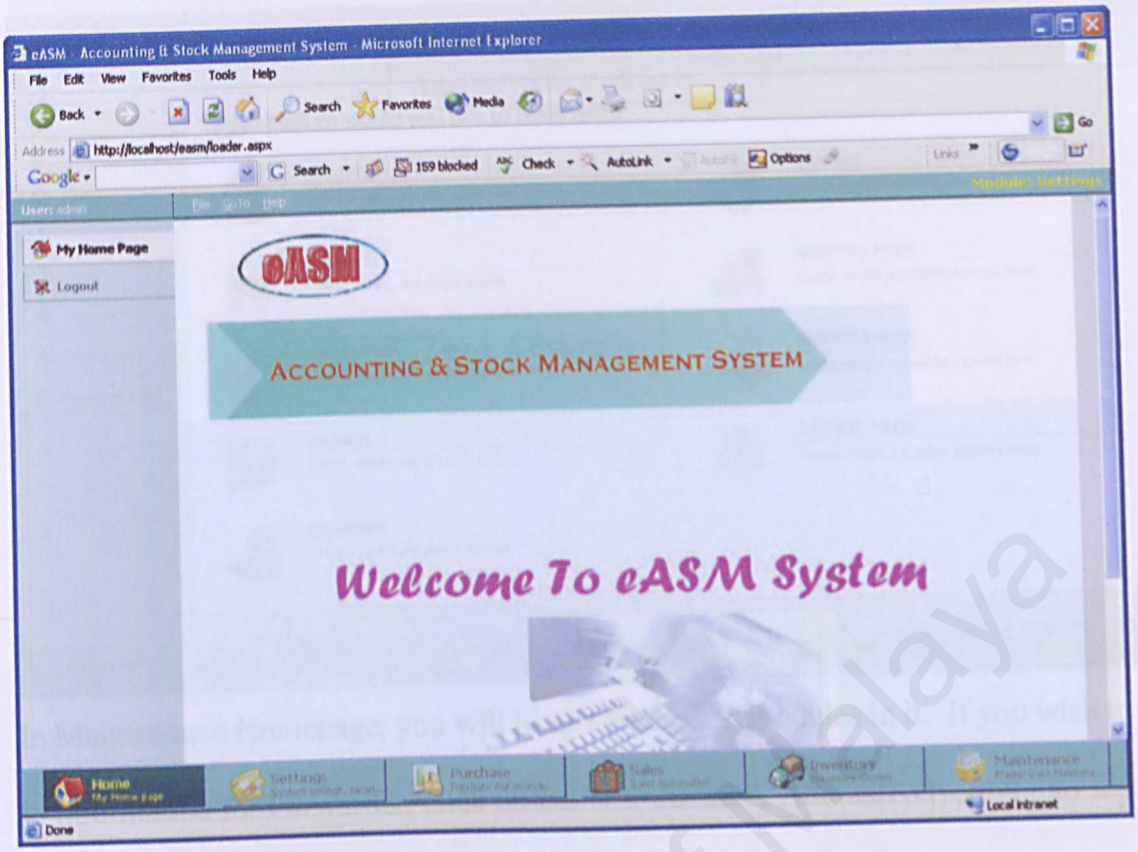
## Getting Started

### User Login

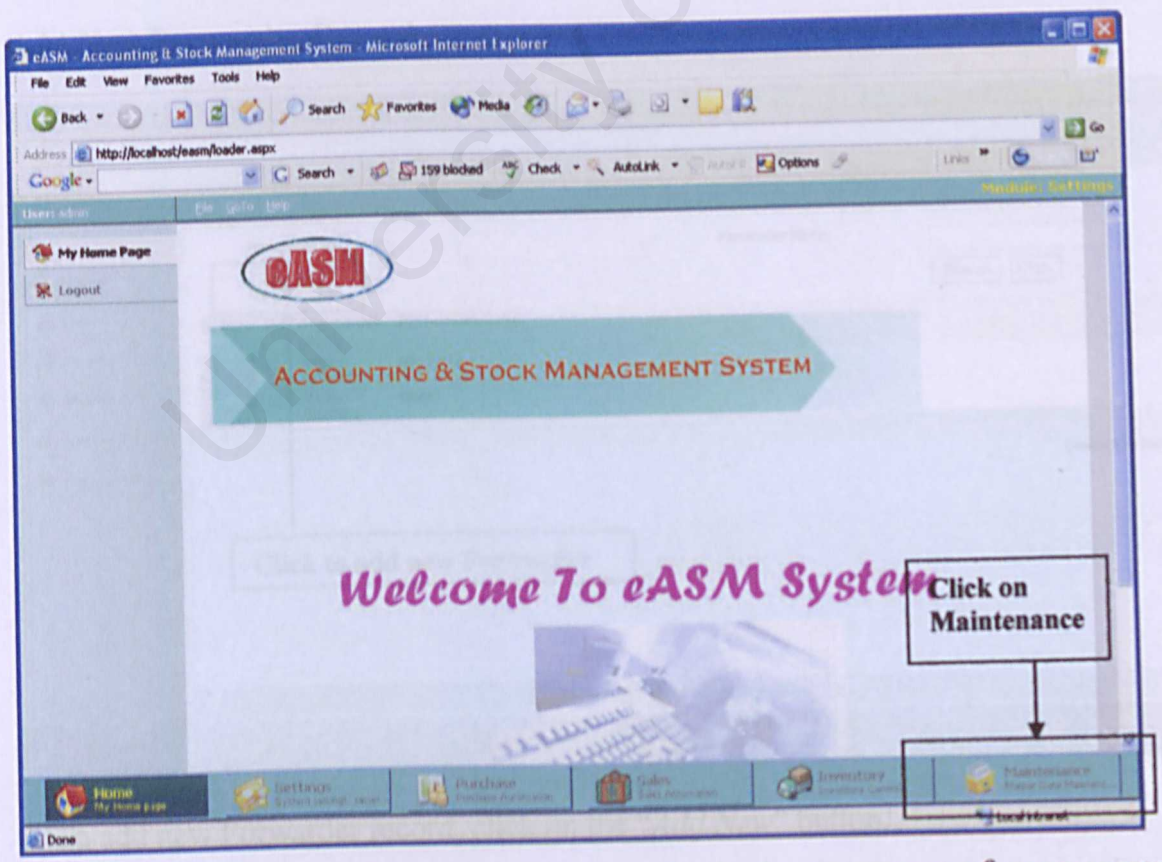



Begin using the system by login to the page at "http://<domain_name_or_ip> /eASM /login.aspx" Key in the username and password given by the system administrator. If you do not have an account, you should apply for an account from the system administrator.

**eASM Homepage**

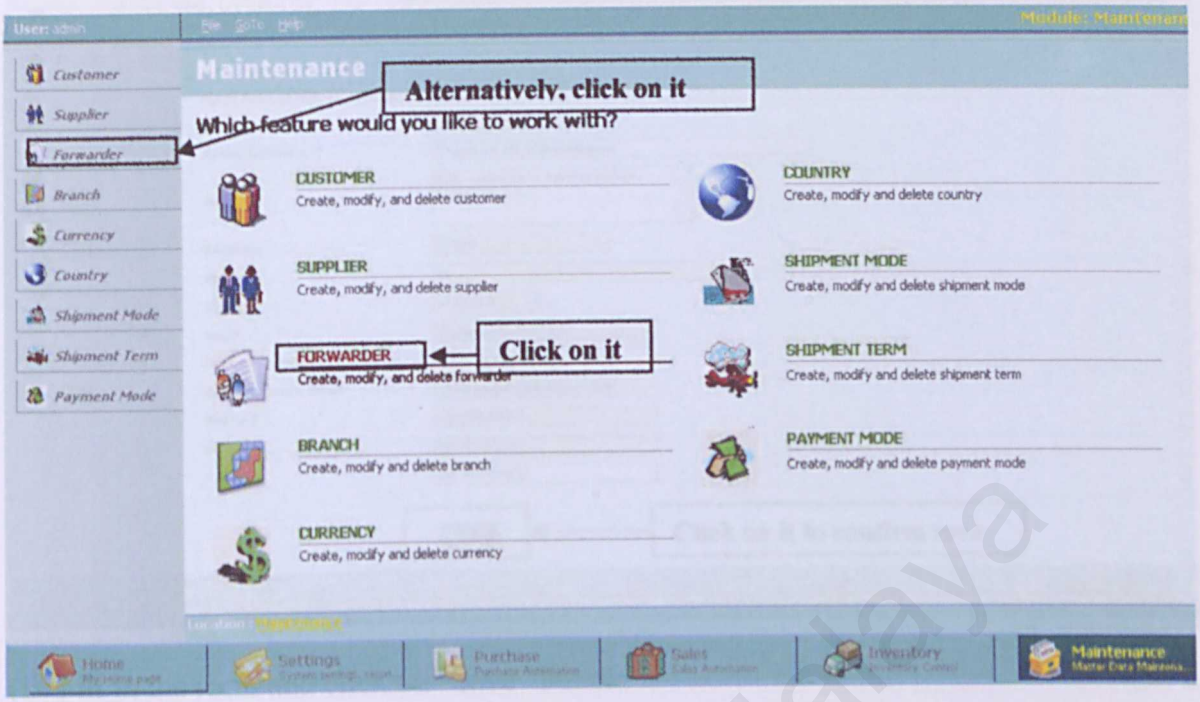



After login, you will be direct to eASM's homepage.



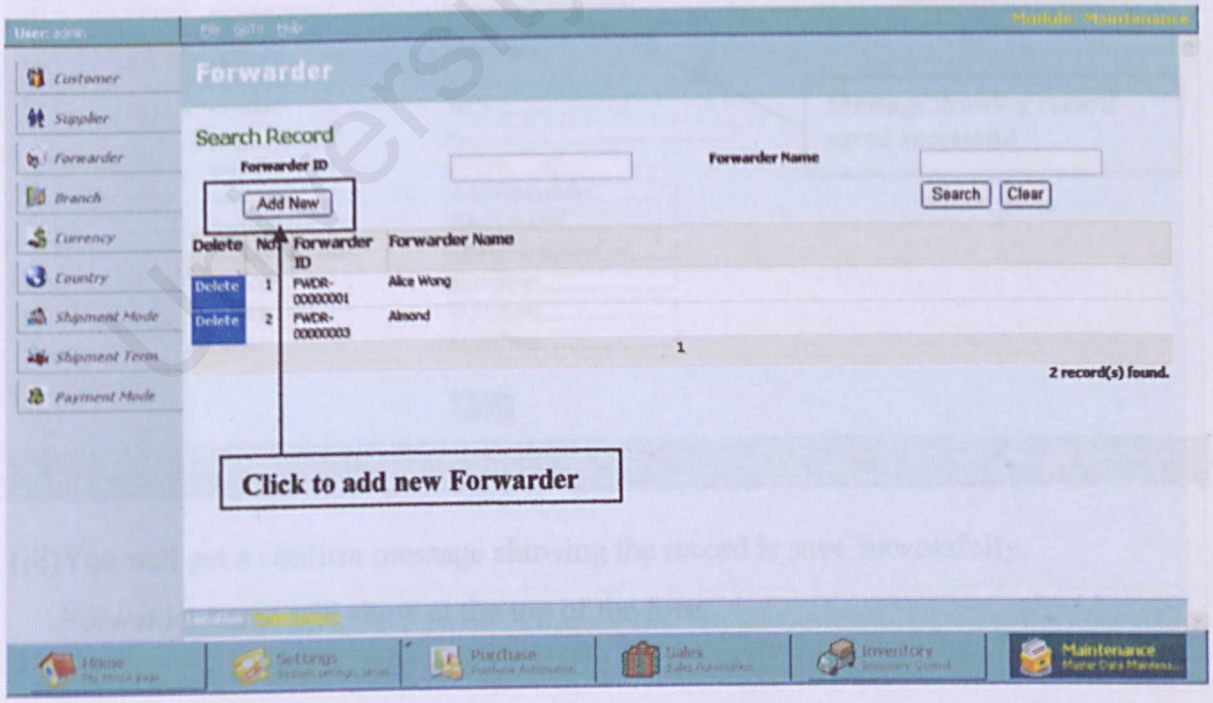

To start, first click on "*Maintenance*" to set the basic component of your accounting system.

## Maintenance Homepage

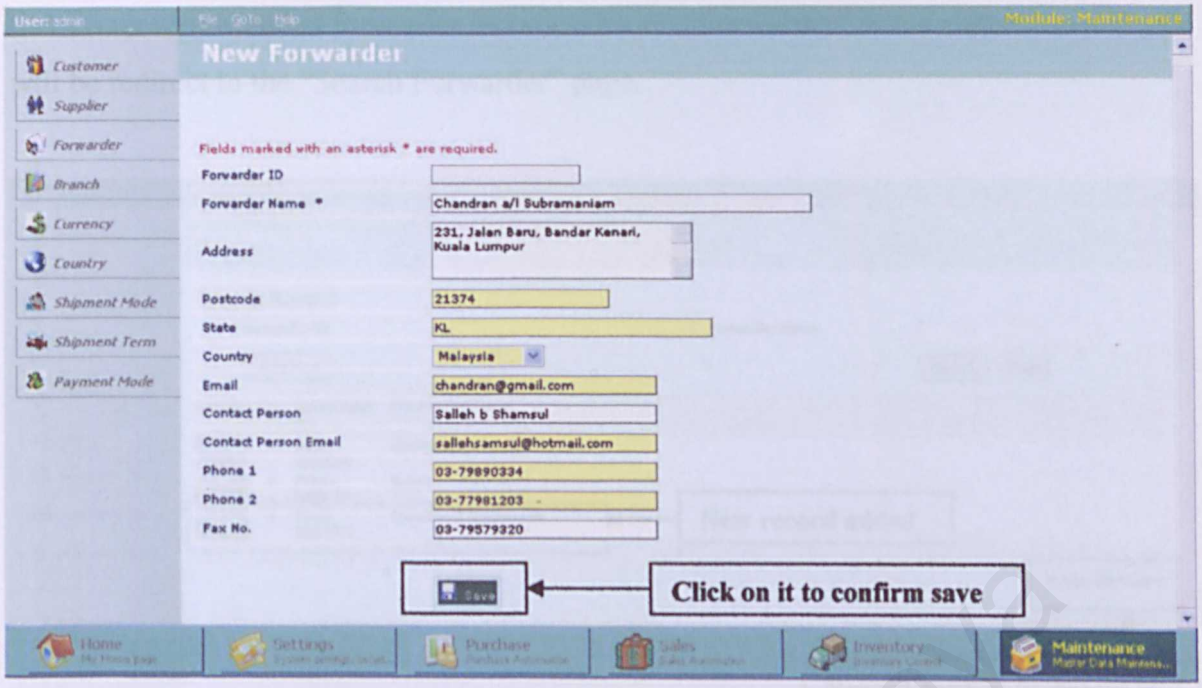



In Maintenance Homepage, you will have various sub-modules in it. If you wish to key in information for Forwarder, click on the "*Forwarder*". Alternatively, you may also click on the side menu on Forwarder to go into Forwarder Module.

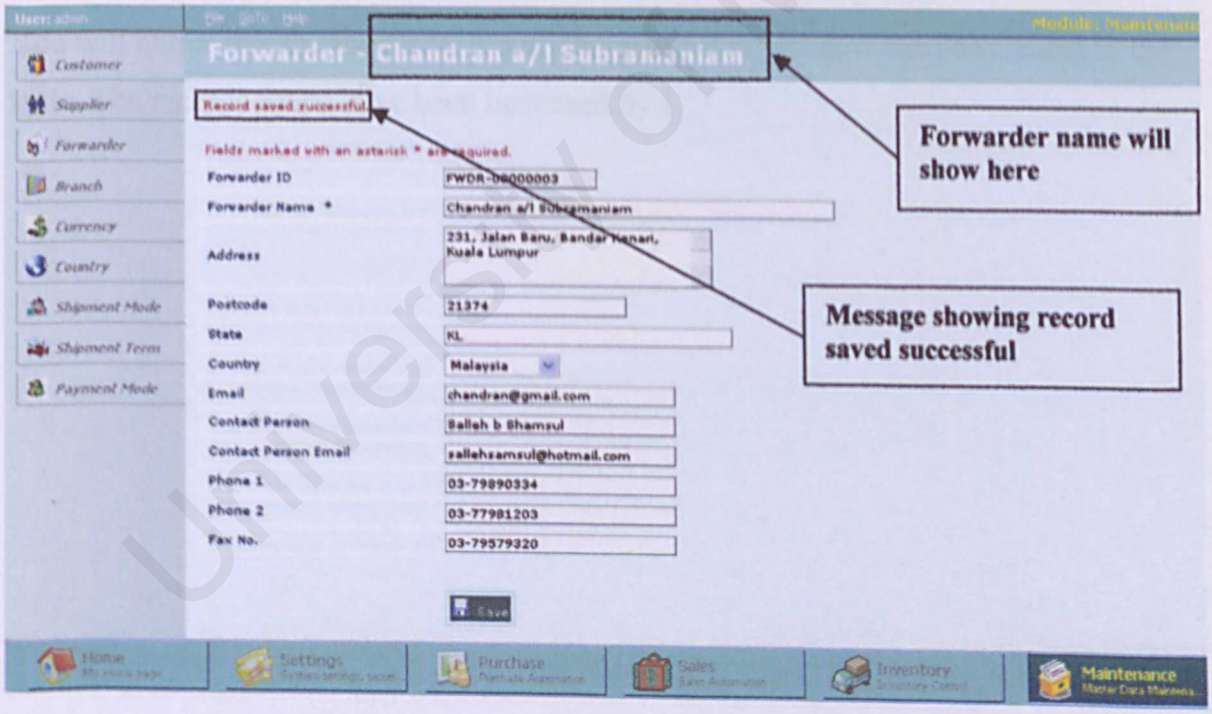
## Add New Forwarder Record




(i)To add new Forwarder record, click on the "*Add New*" button.



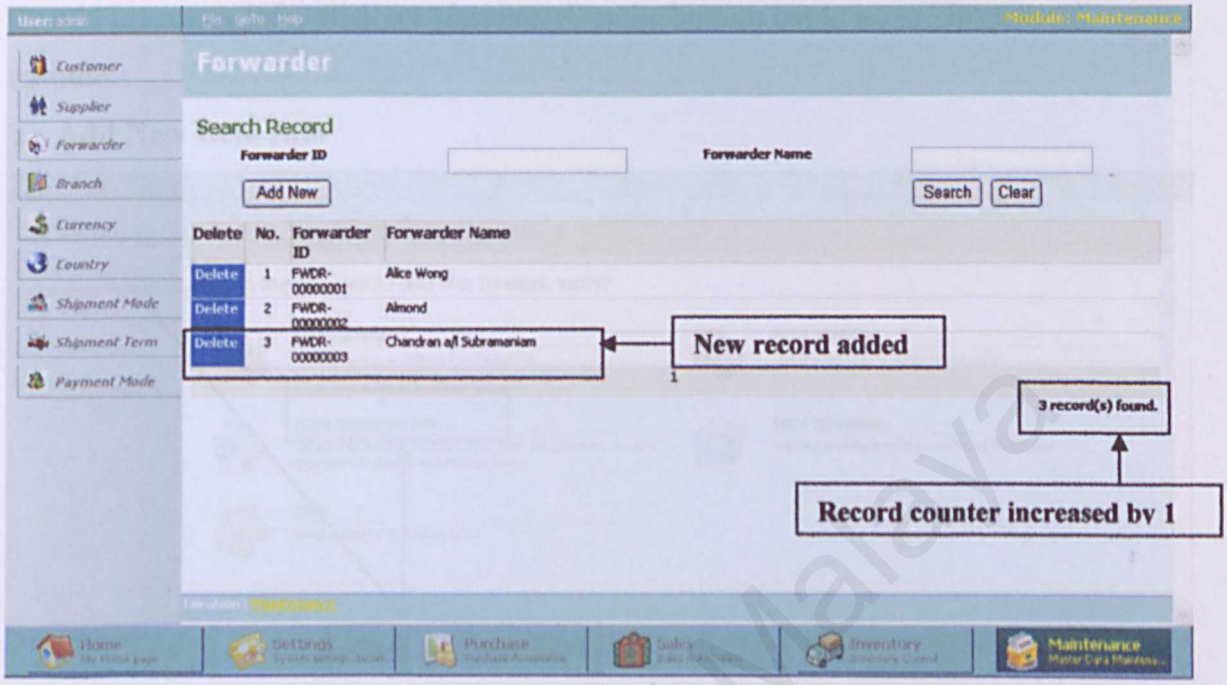

(ii) Fill in the form and click on "Save" button.




(iii)You will get a confirm message showing the record is save successfully.

Forwarder name will show at the top of the form.

(iv) To view new added forwarder record, click on "Forwarder" at the side menu and you will be redirect to the "Search Forwarder" page.

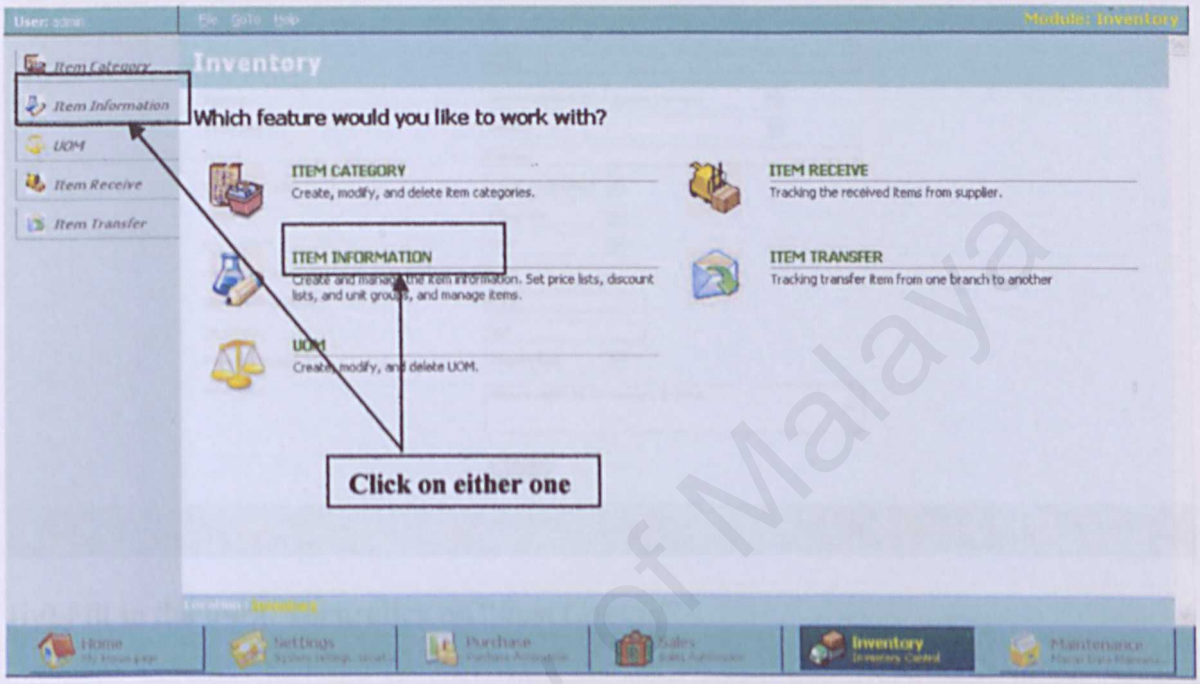



You will notice that the forwarder record that you fill in just now has been added to the table. The records found have been increased by 1.

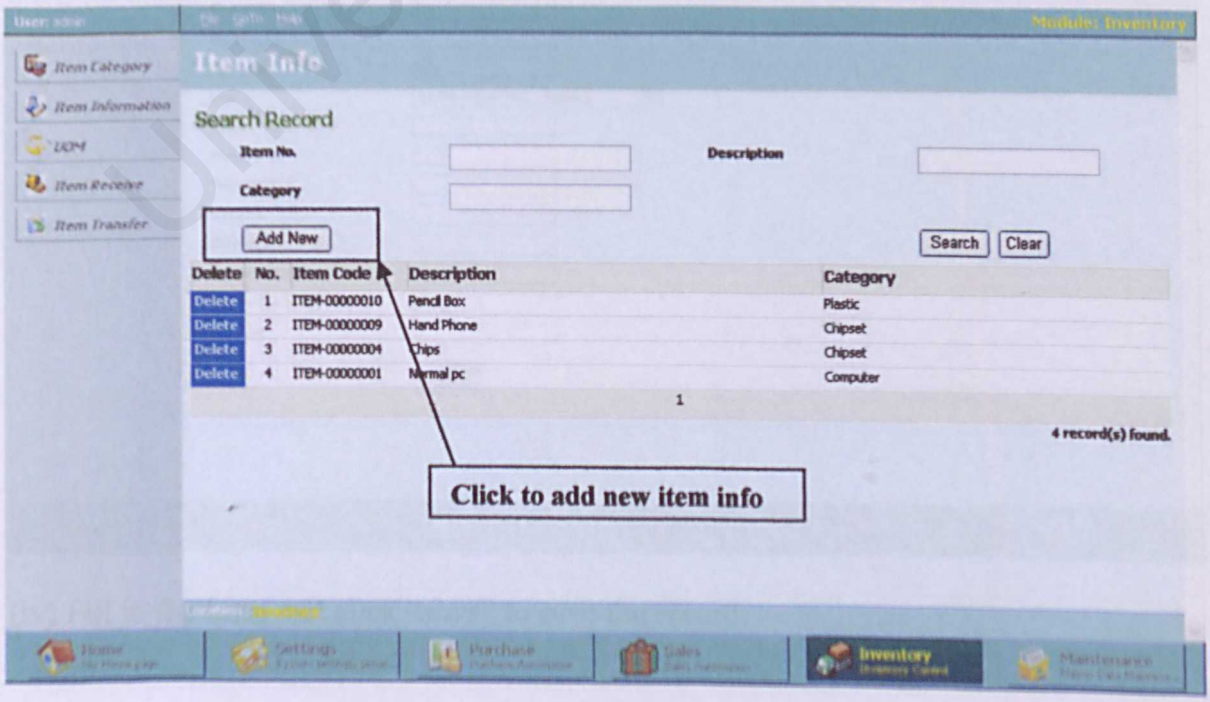
## Inventory


To add Inventory info, click on "*Inventory*" at the bottom bar to go to "Inventory" module.

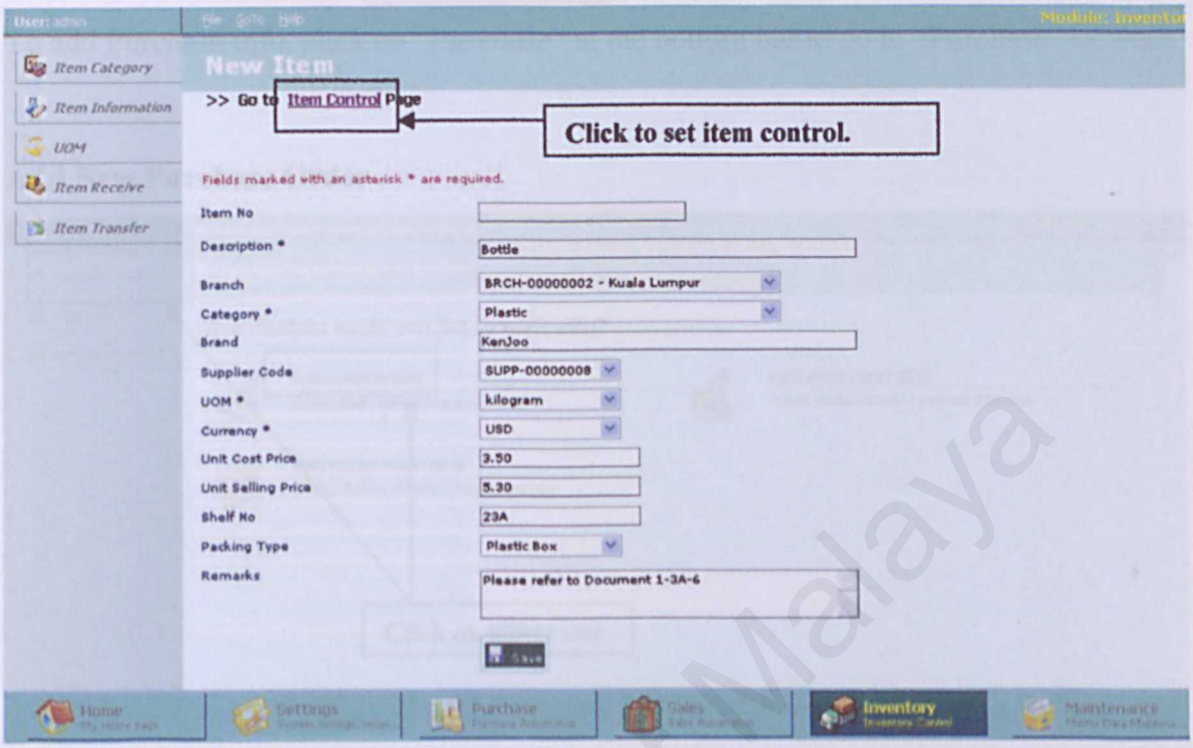
### To Add New Item Info



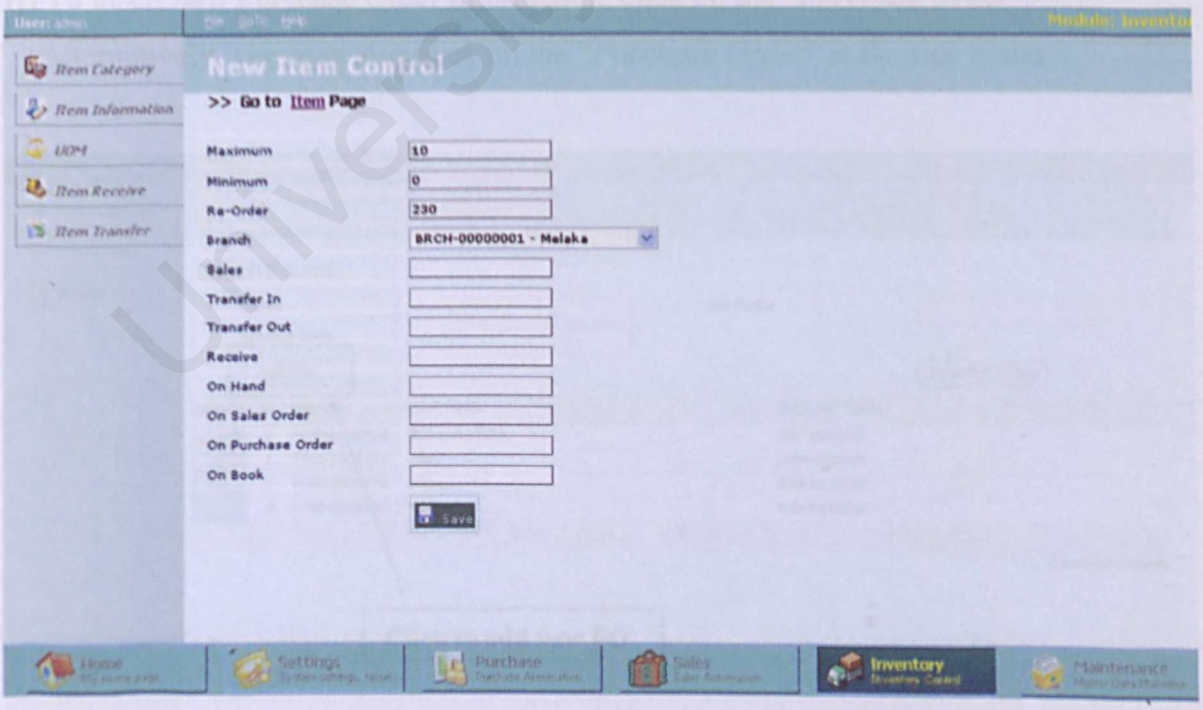

(i) To insert new item information, click on the "*Item Information*". Alternatively, you may also click on the "*Item Information*" at the side menu.



(ii) To add new item information, click on the "*Add New*" button.




(iv) Fill in the form. Then, click on "*Item Control*"

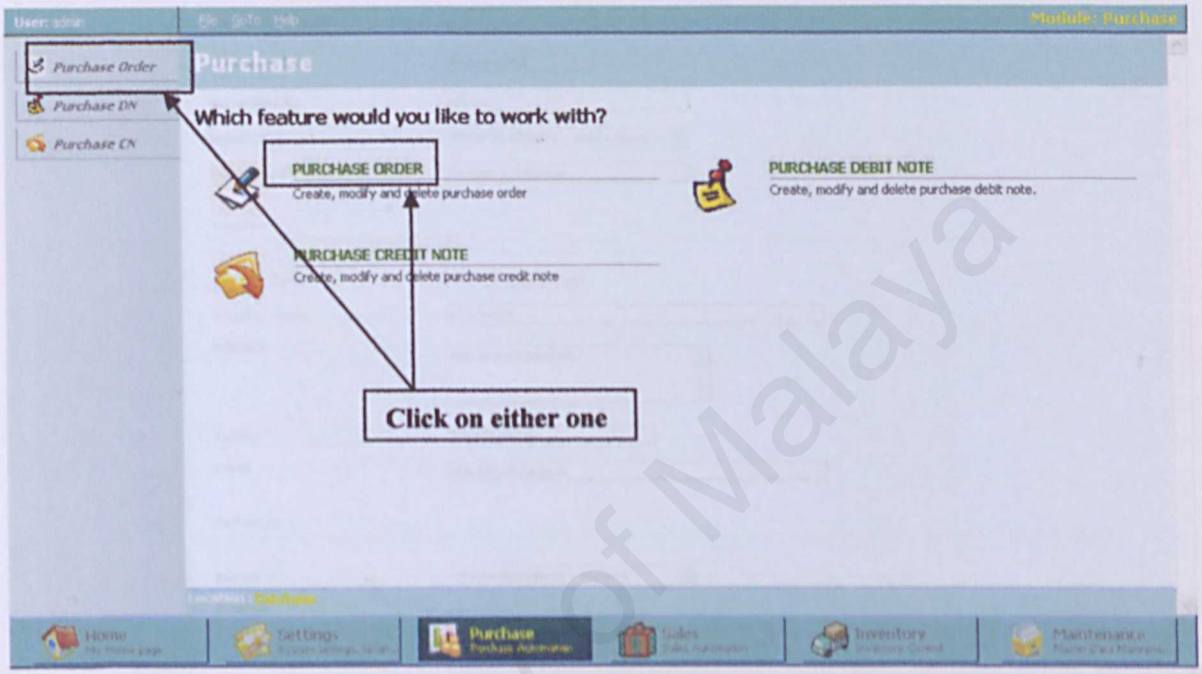



(iv) Fill in the form and click "*save*" to save the record.

## Purchase

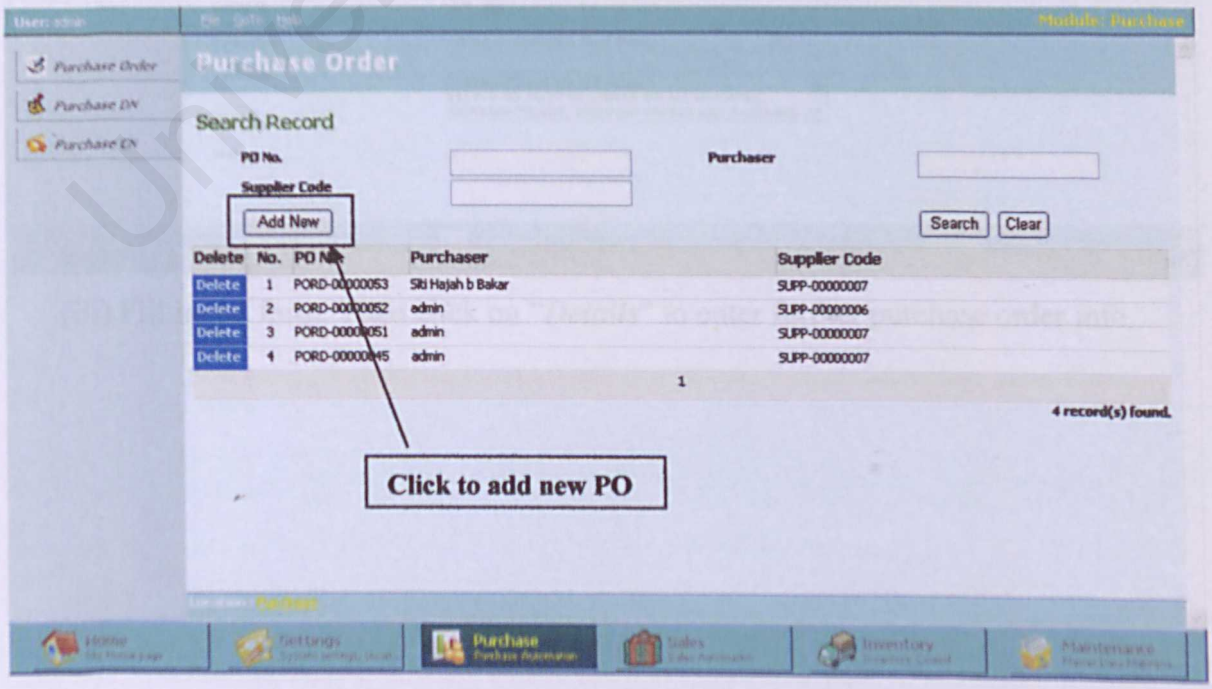

To add Purchase info, click on "*Purchase*" at the bottom bar to go to "Purchase" module.

## Add New Purchase Order

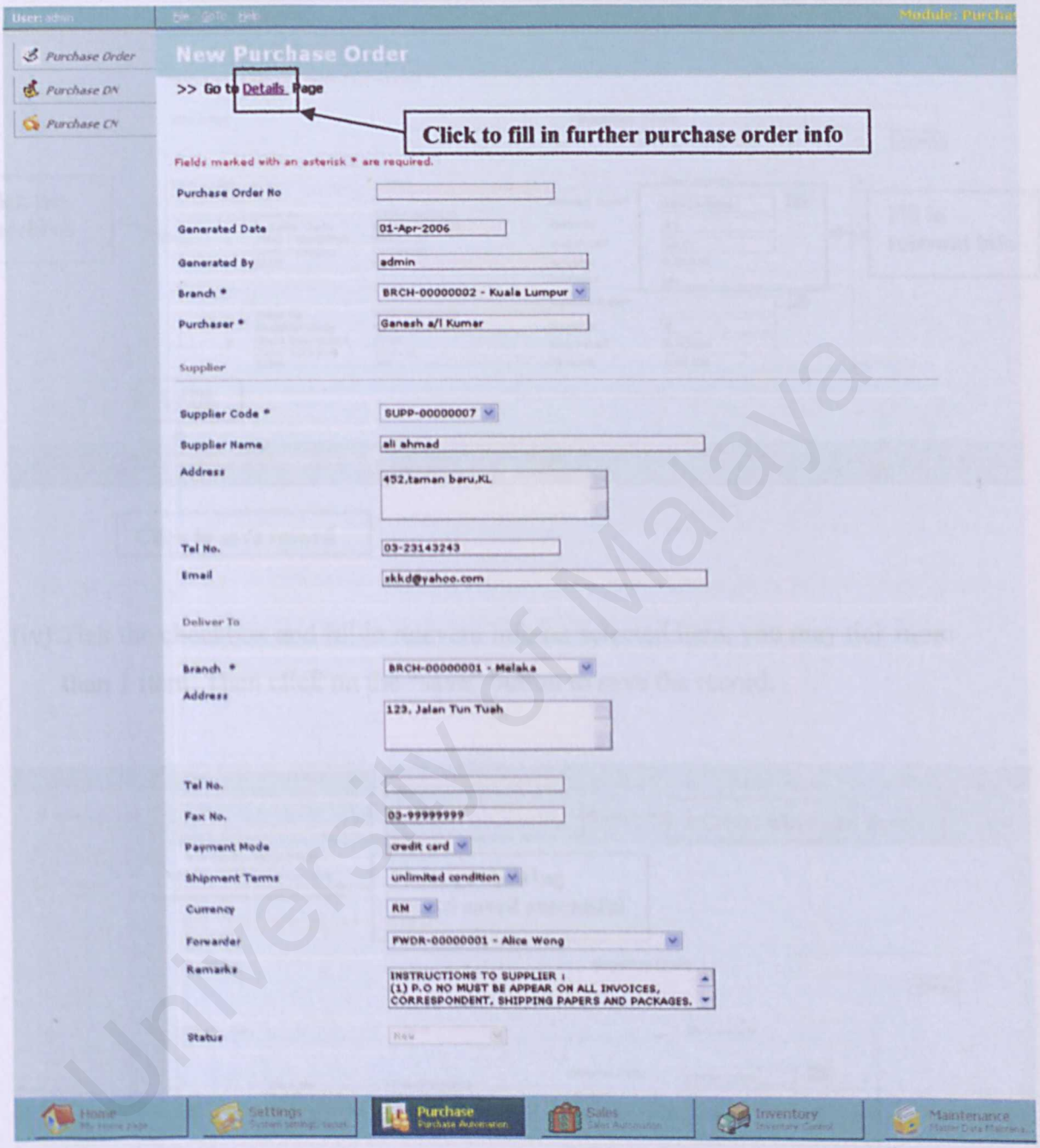



(i) To insert new Purchase Order information, click on the "*Purchase Order*".

Alternatively, you may also click on the "*Purchase Order*" at the side menu.

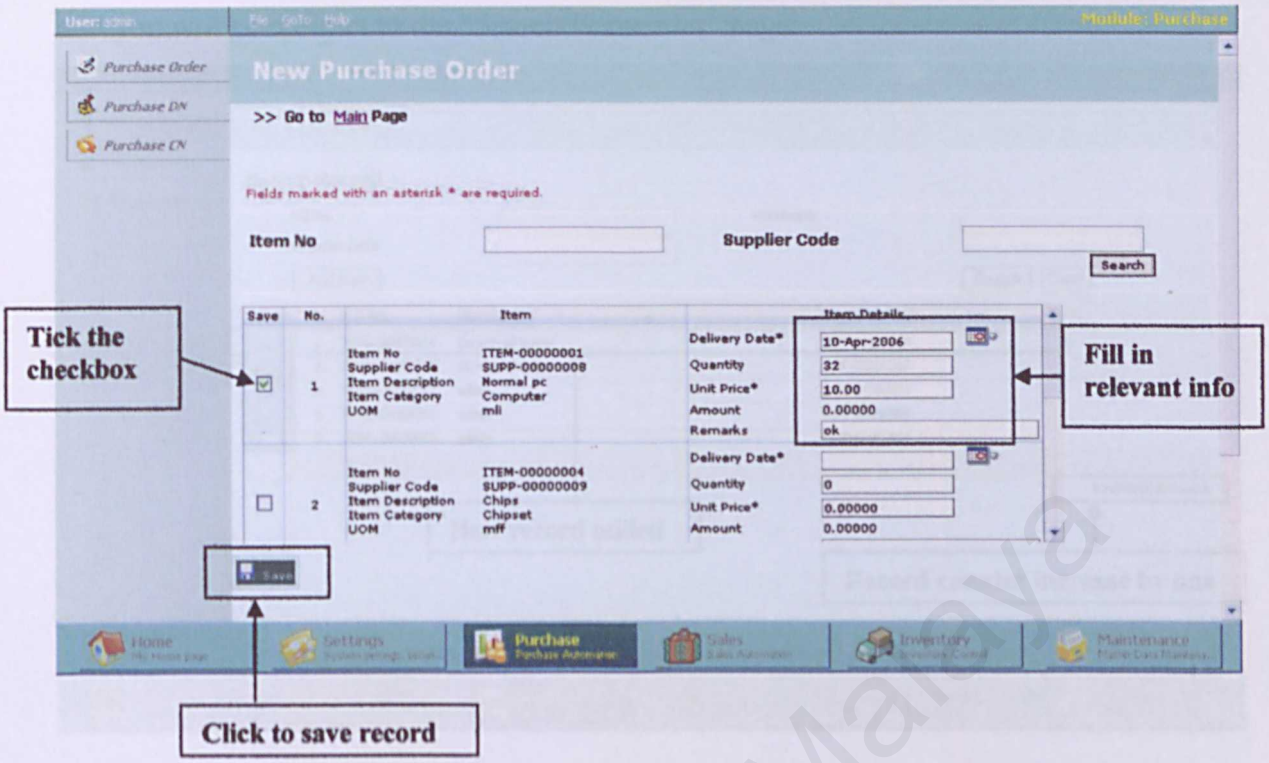


(ii) To add new Purchase Order record, click on the "*Add New*" button.

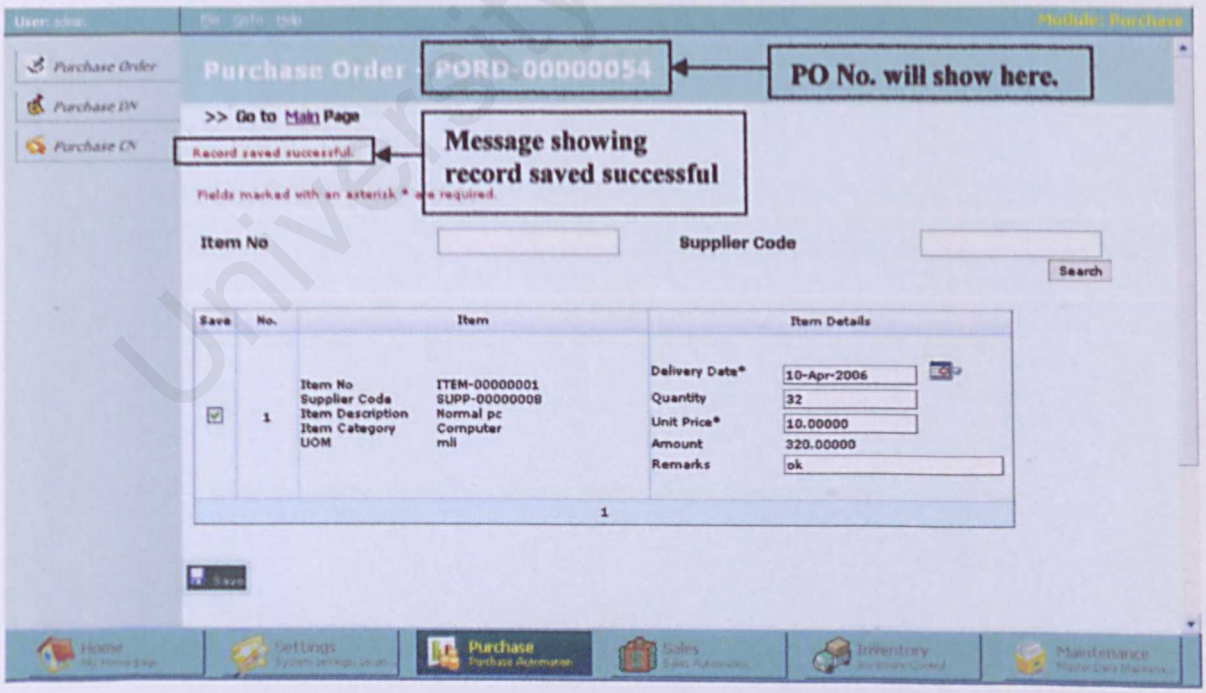



(iii) Fill in the form. Then click on "*Details*" to enter further purchase order info.

**Add New Purchase Order Details**

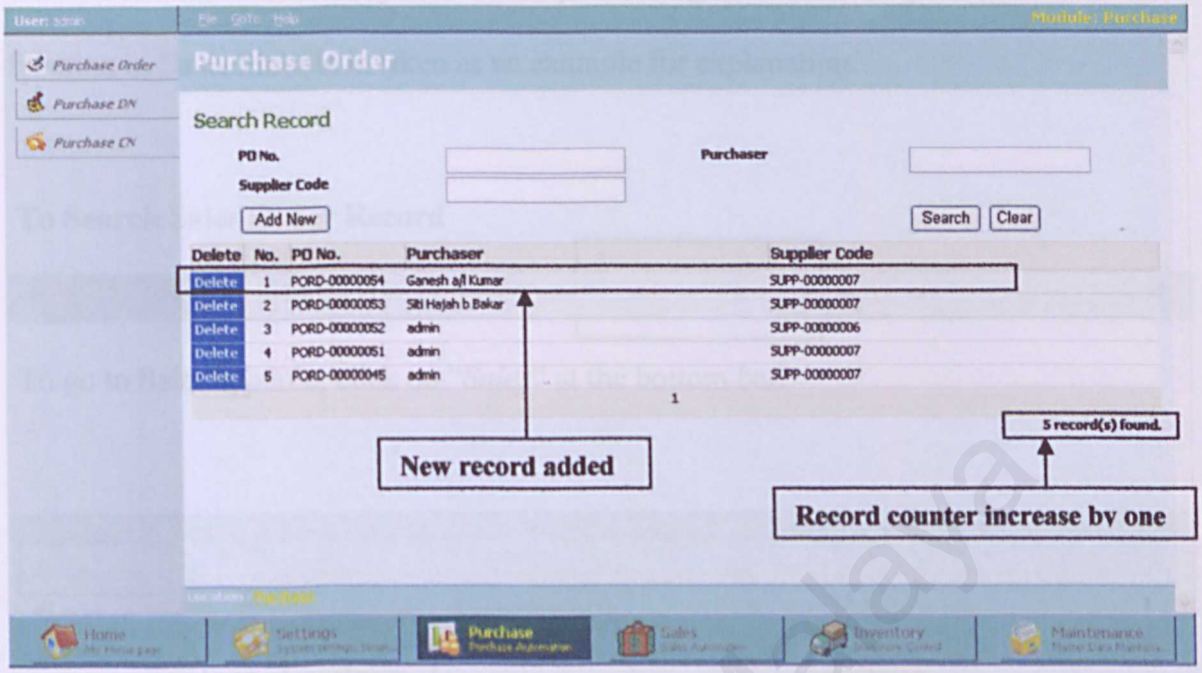



(iv) Tick the checkbox and fill in relevant info on selected item, you may tick more than 1 item. Then click on the "save" button to save the record.




(v) You will get a confirm message showing the record is save successfully. Purchase Order no. will show at the top of the form.

To view new added Purchase order record, click on "*Purchase Order*" at the side menu and you will be redirect to the "Search Forwarder" page.

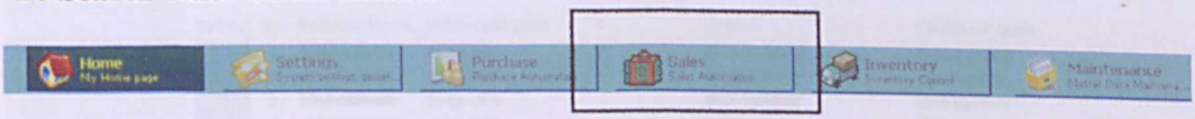



You will notice that the Purchase Order record that you fill in just now has been added to the table. The records found have been increased by 1.

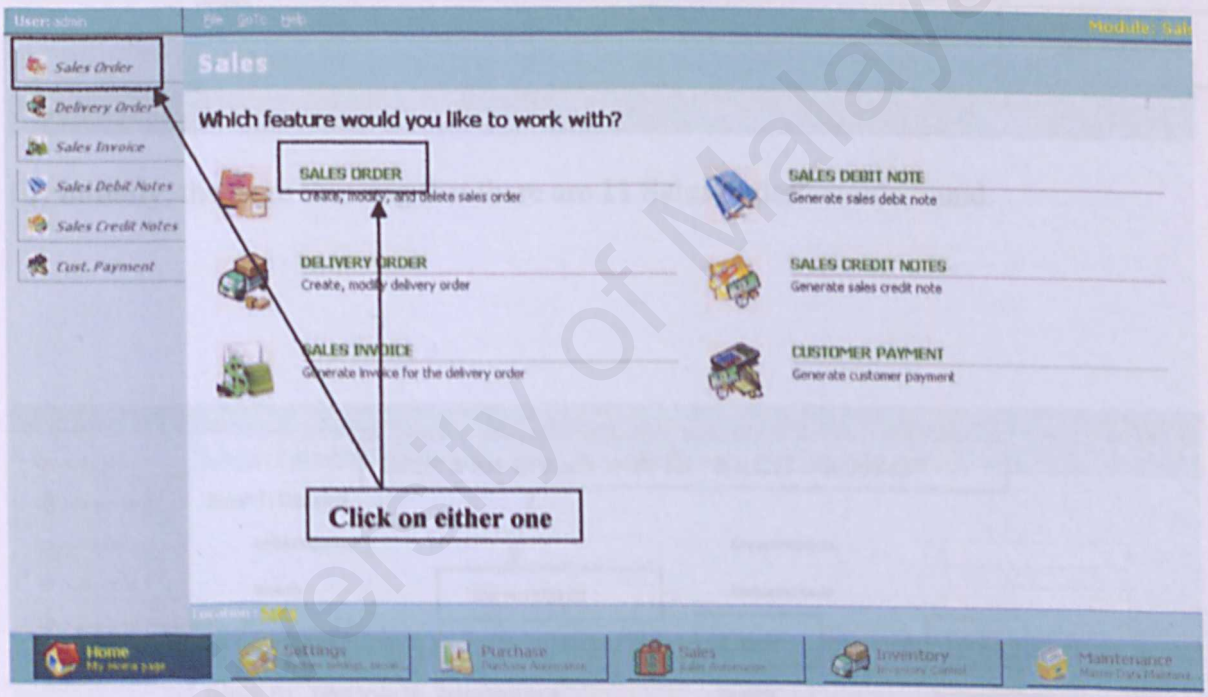
## (A) SEARCH FUNCTION

The search functions in every module are performing in a similar way. So the search function in Sales Module is taken as an example for explanation.

### To Search Sales Order Record

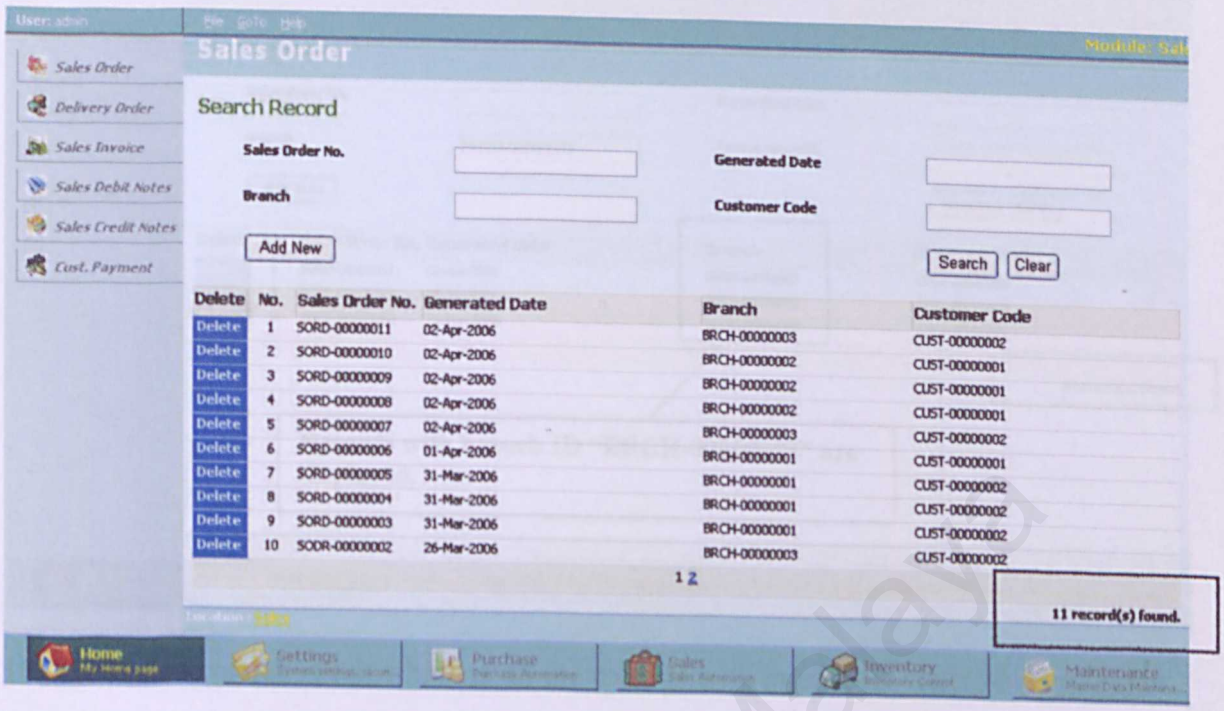



To go to Sales Module, click on "*Sales*" at the bottom bar.

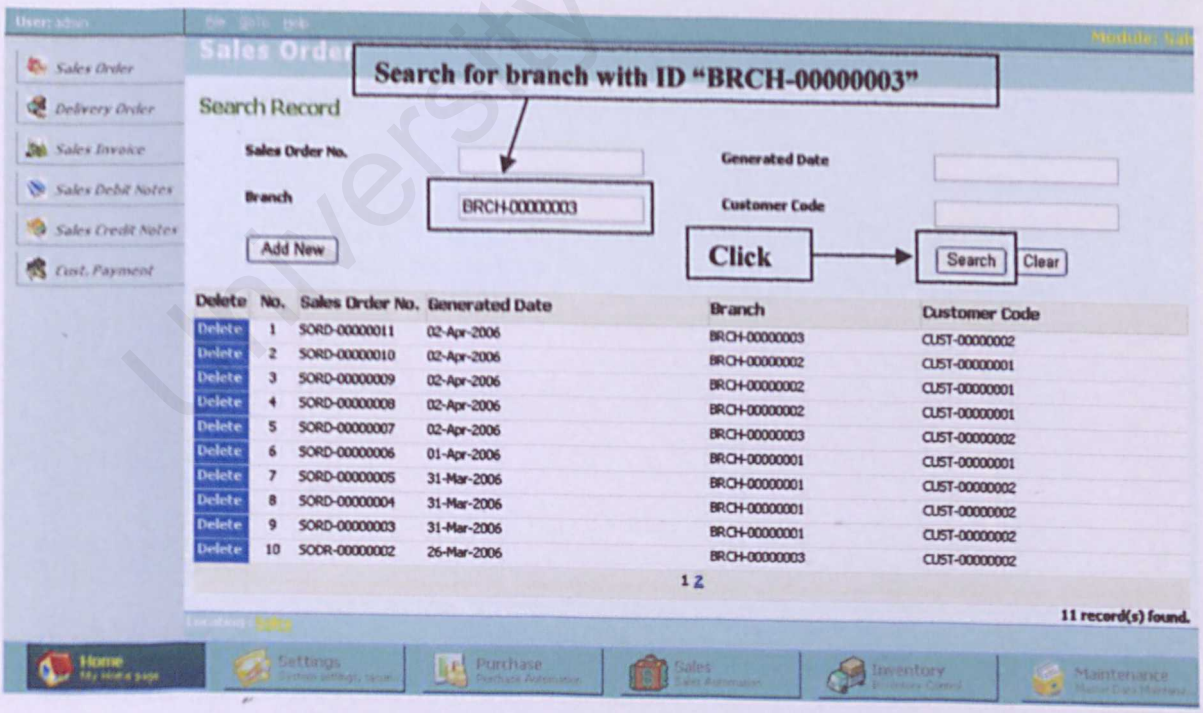



(i) The Sales Order Module is selected as an example to explain for the search function.

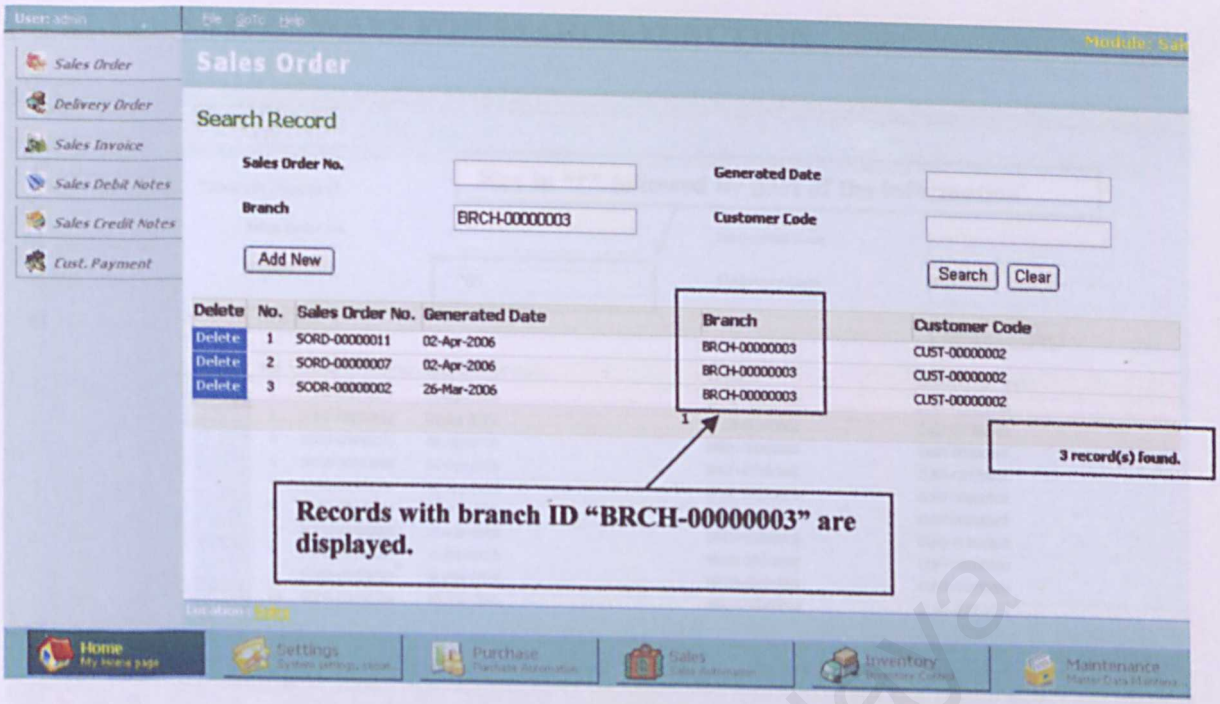
After click on the sales order, you will redirect to the following page:




(ii) Initially, the page showing that there are 11 Sales Order records found.

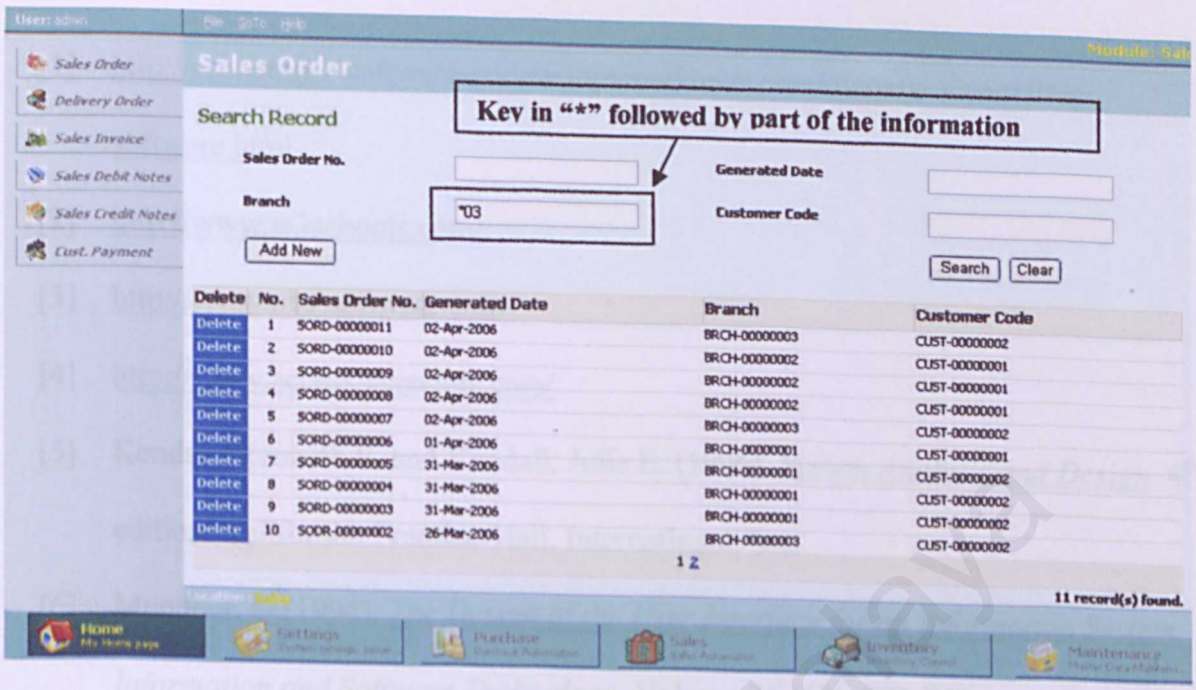



(iii) If the users wish to search for Branch with Branch ID "BRCH-00000003", then user can key in "BRCH-00000003" in text box at branch and then click search

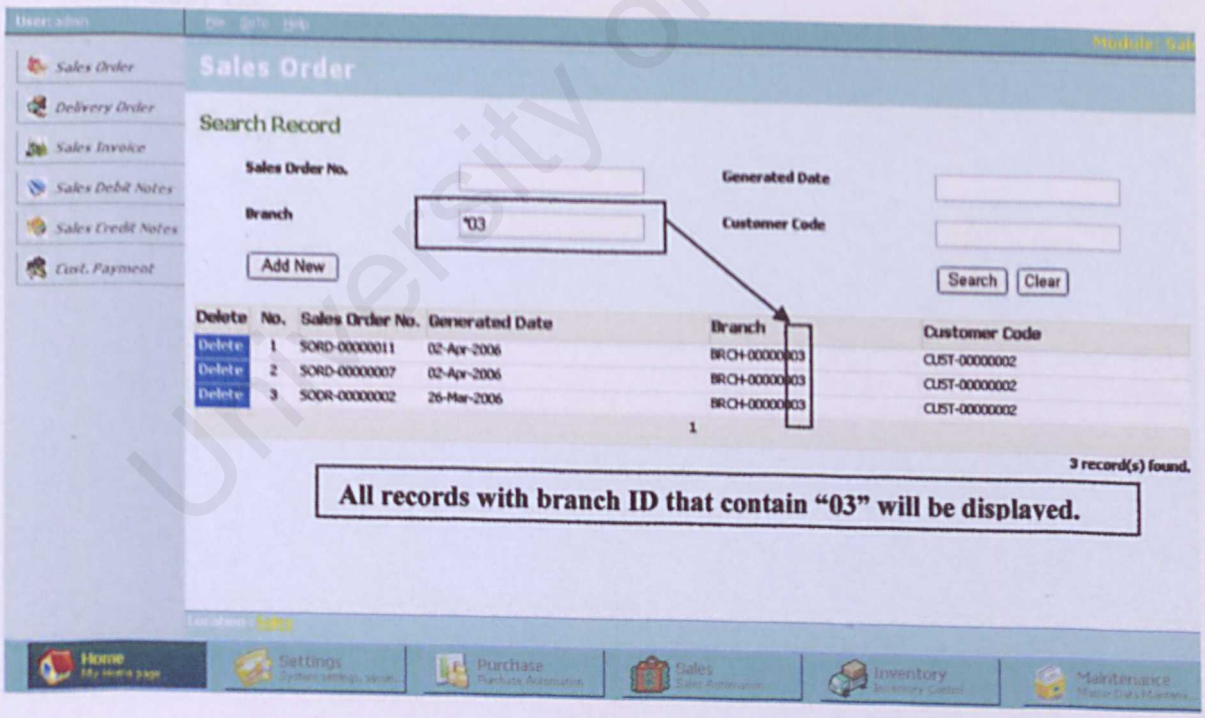



(iv) After that, search result which is all the records with branch ID "BRCH-00000003" will be displayed. Search result shows that there are 3 records found.

## (B) ALTERNATIVE WAYS FOR SEARCH FUNCTION




(i) If users unable to remember the entire information for certain column, users might key in "*" followed by part of the information.




For this case, users might key in "*03" at the branch column to look for any record with branch that contain "03" on it. So the same search result will be displayed.

**NOTE: The same searching method can be apply to all the search pages in eASM.**

# References


[1] http://accounting-software-review.toptenreviews.com/cymaiv-accounting-software.html

[2] http://www.w3schools.com/

[3] http://www.ubs-software.com

[4] http://www.access-accounts.com/

[5] Kendall, Kenneth E. and Kendall, Julie E. (1996). *System Analysis and Design*. 4th edition. California: Prentice-Hall, International, Inc.

[6] Mundher, G.(1994). *The Design of the User Interface for an Information System. Information and Software Technology.* Volume 36 (12): 773-742

[7] Sommerwille, I. (1995). *Software Engineering*. 5th edition. Reading: Addison-Wesley Ltd.