UNIVERSITY OF MALAYA

Inventory Management System

For

FCSIT

Faculty of Computer Science & Information Technology
University of Malaya

ONG POH LING (WEK98136)

Under the supervision of
ENCIK MUSTAFFA KAMAL

Moderator
Puan Salimah Mokhtar

A Final Year Project report

Submitted to the

Faculty Computer Science and Information Technology
University of Malaya

In Partial Fulfillment of the Requirements For
The Degree of
Bachelor of Computer Science
Session 2000/2001

Submission Date (22 January 2000)

ABSTRACT

The inventory management system is a client server system with the objective to computerize and enhance the inventory management system for Faculty Computer Science and Information Technology (FCSIT). The server side contains all related basic inventory management rules, documents, and database. Meanwhile the client consists two sections, which are User and Administrator. The User section allows user to browse through the faculty information and search available items. The Administrator section is for the system administrator to perform all administrative tasks namely maintaining the inventory database.

The inventory management system for faculty was developed using Microsoft Visual Basic 6.0 to create the application programs, Microsoft Access 97 as the prototype testing, Microsoft SQL Server 7.0 as the database server and Windows NT Server 4.0 as the network operating system.

The inventory management system for the faculty is developed with the purpose to enhance the inventory management system for the faculty. Most of the current inventory management applications are developed to suit the requirement of manufacturer and not for the university. Therefore, the proposed system is to customize an inventory management application to suit the inventory control needs of the faculty.

ACKNOWLEDGEMENT

First of all, I would like to thank Encik Mustaffa Kamal, my supervisor for giving me to join this project and provide me the fully equipment. Secondly, I would like to thank him for his constructive advice, generous guidance, encouragement, support and supervision along the project. His hardworking and kindness in helping me throughout the project is deeply appreciated.

Thanks to Puan Salimah Mokhtar, as the project moderator who contributed suggestions and ideas to further enhance value of this project. The appreciation is also dedicated to all the FCSIT's officers especially Encik Bahar, Encik Silen and others who have provided important information for the project. Thanks to Mr Lee Tung Chai and Ms Leong Yin May who are work in Bahagian Perkhidmatan Komputer at University Malaya. They was provide me a lot of usefulness information for the project. I want to express my gratitude to Puan Minah, who is database inventory management system in Jabatan Harta Benda at University Malaya. She was gave me a lot of suggestion and idea to improve the functionality of the current system.

Thanks to all my course mate for their valuable idea. Thanks to my senior, Mr. Lee Chee Weng, Mr. Chim Sai Mun, Mr. Yap Say Leong and Mr. Lee Boon Ping for their comments and advice throughout the project.

Thank you very much!

Table of Contents

List of Figures

List of Tables

CHARTER 1: INTRODUCTION

1.1 Project Background

As we walk into the Faculty of Computer Science and Information Technology (FCSIT), we will notice a reasonable amount of items ranging from furniture to advance computer equipments. In order to keep track of all these item in an organized manner there must be an inventory management control that can use to monitor these times either manually or electronically.

Inventory management has come a long way from the day when the stockroom expeditor knew which parts were obsolete or had been replaced by some other part, when a full accounting of the inventory was taken only once a year, and when inventory levels were maintained at the discretion of the users. Today the well-organized inventory manager brings together the requirements of people in university to provide an inventory management system that serves individual needs and effectively contributes to them.

Inventory as defined by the Oxford Advanced Learners dictionary means a detailed list of goods, furniture and jobs. An inventory system is a system that keeps track the list of registered goods or furniture. As a model for a computer inventory system, the Faculty's inventory system is chosen. The purpose of the inventory system is to keep track and record all the equipments that is owned by the faculty. The system does not include equipments from others faculty.

1.2 Project Aims, Objective and Significance

1.2.1 Aims

This project is aimed to provide a computerized inventory management in the faculty. Beside that it also provides a user-friendly system that meet user requirement, reduce time spent on applying for inventory, and increase work performance and efficiency in faculty.

It is also important to provide adequate safeguards to protect the system from unauthorized users. The access levels of the administrator, the approver, staffs and lecturers are different. This is to protect unauthorized users from editing the contents of the database and system.

1.2.2 Objectives

The main objectives of the system are:

i) To Enhance current inventory management application.

ii) To Ensure sufficiency of inventory.

iii) To increase quality and accuracy in data keeping.

iv) To develop interactive interface for Inventory database.

v) To provide timely and meaningful reports to aid management decisions.

vi) To optimize cost of ordering supplies and stock holding costs.

vii) To improve management control on purchases, receipts and requisitions by providing up-to-date information

v) The current system is not flexible due to the fact that the current application can only searches for items using item code and item tag (users have to refer back to the code and tag in a book)

vi) The current system does not provide much information/reports about the inventory because the current system will only generate a summary of items in the inventory

vii) The current system is not user friendly

1.4 Project Scope

The implementation of a complete Inventory Management System (IMS) will require a significant amount of man-hours, so we have to define the project scope so that we can complete the system in a limitation of time. For this project, our scope is only limited to building the system for the FCSIT. It only covers some of the basic module that fulfills the need of the faculty. This system will include the academic staff and also the non-academic staff in FCSIT.

1.4.1 Functional

The following are some of the functionality of Inventory Management System:

i) The administrator can query the information and generate reports easily.

ii) Super administrator can add, change or delete the data in the inventory.

iii) It can be divided into web-based system or stand-alone system.

iv) The system is able to support for the growth and future enhancement.

v) The system is stable, reliable and user-friendly.

vi) The complexity of the system is kept as low as possible while the quality is always assured.

1.4.2 Modules

The project is a client server database system. It will provide complete information of applicants, transaction involved, quantity, type of inventory and status of particular inventory.

The project can be divided into 6 modules as below:

i) Module 1, Inventory information module

The system is designed to allow creating, editing and deleting inventory information. The inventory information can be searched according to inventory number, inventory category, inventory location and inventory's preferred vendor.

ii) Module 2, Vendor's information module

This module is designed to allow creating, editing and deleting vendor information. The vendor information can be searched according to vendor number and vendor name.

iii) Module 3, Inventory Part information module

The system is designed to allow creating, editing and deleting inventory parts information.

iv) Module 4, Inventory Directory information module

The system is designed to allow creating, editing and deleting inventory parts information. It includes item code, item name and group.

v) Module 5, Inventory Department information module

The system is designed to allow creating, editing and deleting inventory parts information. It contain department code, faculty and department name

vi) Module 6, Personal information module

The system is designed to allow authorized users to access Inventory Management System (IMS) by using a user ID and password. Only valid users are allowed to access into the IMS, while unauthorized users will not be entertained.

1. 5 Project Schedules

The timeline of the project will be show on Figure 1.2. In overall, the project will be start at 5th June 2000 until 10th February 2001.

No	Task Name	Start	End								2000	
		Date	Date	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb
1.	Concept of Definition	05/06/00	09/06/00	■								
2.	Requirement Specification & Analysis	10/06/00	01/08/00	■	■							
3.	System Design	02/08/00	21/08/00			■						
4.	Incremental Prototyping	22/08/00	22/12/01			■	■	■	■	■		
5.	Integration & Testing	23/12/01	10/02/01							■	■	■

Figure 1.1 Project Timeline

1.6 Project Expectation

Basically, for any project, certain expectations of the outcome are projected before the work started. A few factors have to be considered in making these expectations. One of the important factors is the amount of time available to complete the project and also the technologies and resources available. The following are some of the expectations of the project:

i) System can perform some basic function and meet some criteria such as stable, consistency, user friendly and also reliability.

ii) The systems will be able to fulfill the requirement of faculty and can perform the required functions efficiently and effectively

iii) The proposed system is quite a complete solution. However, it needs to be enhanced so that more functionality can be added.

iv) The final implementation should allow for future enhancement as well as additional module to add functionality to the Inventory Management System that suit to the need of FCSIT.

Chapter 2: LITERATURE REVIEW

Before a project is being developed, it needs a lot of studies and analysis. We have chosen Faculty of Computer Science and Information Technology (FCSIT), University of Malaya, as the test bed for the project. It is because many process in our faculty office are not automated and the staffs face a lot of problem everyday. After gathering some feedback, complaint, suggestion and information, it is decided to develop a system that could overcome the shortcomings and maximize the functionality of the administrative processing in the faculty inventory control.

2.1 Current Inventory Management System in FCSIT

The current Inventory system in FCSIT is using dBASE IV to manipulation the data in the system. The dBASE IV program just can operates in MS DOS environment. It cannot operate in the windows environment.

What is dBASE IV?

The dBASE IV program is sophisticated records management and application development software. The dBASE IV have all of the features of dBASE III Plus, the version IV added a number of new commands and functions. With dBASE IV, Ashton-Tate added a new Control Centre for report, label, query, and applications design. DBASE IV lets users create their own functions, called user-defined functions, which may be added to the standard set of functions supplied with the dBASE IV program.[1]

Below is the **process flow** of the current inventory management system in FCSIT.

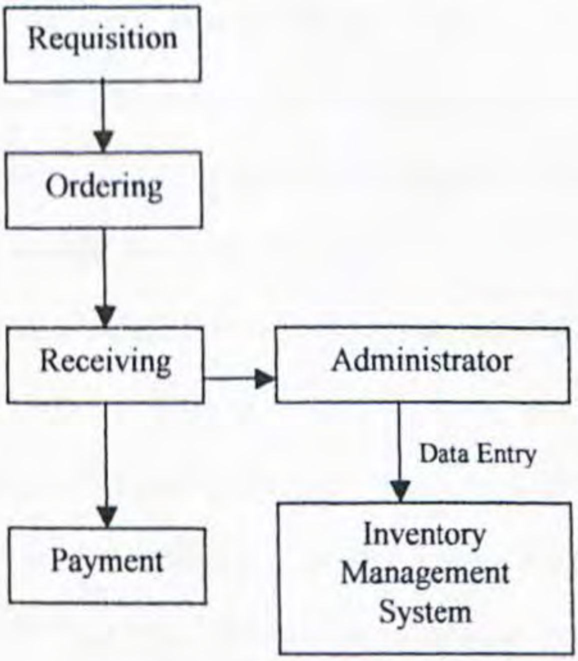


Figure 2.1 Current Inventory Management Systems

When the faculty requires certain items, the administrator (person-in-charge) will request PO form from the office for ordering purpose. After that, the administrator will order the items from the supplier. When the administrator receives the items. The administrator must go to the system to record received items. Since the current system is a stand-alone computer, which does not have any network connection, the administrator has to go to the specific computer to key in the data. The total cost of the items will be paid by the treasurer (Pejabat Bendahari of University Malaya).

Interview

Interview activities have been carried out in order to understand the loopholes and weakness of the current system. A few interviews had been carried out with the office workers. First of all, I interview Encik Bahar to get the detail about the whole process of the inventory management. From the interview, I have found out how the inventory record is stored inside the database and how the system functioning. We discovered that Encik Bahar faced a lot of problem in handling the current Inventory Management System. He gave me some opinion and suggestion to improve the current system to minimize the burden faced by the administrator. He also had provided us some information about the workflow of inventory management process.

Besides, I interview Ms Leong Yin May, who is developed the current inventory system in FCSIT. She was provided us a lot of information about the current system. I also interview Mr Lee Tung Chai, who is work in Bahagian Perkhidmatan Komputer (BPK) at University Malaya. He had development inventory system in window environment. Mr Lee Tung Chai had provided us with a lot of essential information that can be used to design our new inventory management system.

I also interview Puan Minah, who is administrator inventory management system work in Jabatan Harta Benda at University Malaya. She was provide me a lot of useful information like record structure of inventory item, item codes for system inventory and a list code of the faculty. She also explanted the menu options on the inventory system in Jabatan Harta Benda.

Encik Silen gave us a lot of suggestion and idea to improve the functionality of the current system. Beside that, my supervisor Encik Mustaffa Kamal and my moderator

Puan Salimah Mokhtar had provided me with some valuable advice and updated information that is very useful in designing the system.

Through the interviews, I conclude that the current Inventory Management System is not good enough to support the current situation. The user can only searching at the local PC for inventory information because it is stand-alone system. That mean the user must go to the system to searching the inventory information. The current inventory system is not flexible by the user to use. The administrator (person-in-charge) to store all the record and to print out the inventory report at the local database and the administrator faced a lot of problem in the current situation.

2.2 Finding

Reviews have been done on three main areas. They are

- Inventory management system
- Related programming concept
- Development tools

2.2.1 Inventory Management System

2.2.1.1 Definition of Inventory

Inventory is defined as a stock of goods. More generally, inventory could be regarded as a resource that has economics value. An inventory is made up of one or more items where each item is a unique supply item, raw material, purchase or manufactured part, assembly, or final product [2].

2.2.1.2 Types of Inventory Management System

It is difficult to classify the various inventory management systems in an orderly fashion. Common types of inventory control systems are the perpetual, two-bin, periodic, optional replenishment, and material requirements planning systems. The perpetual, two-bin, periodic, and optional replenishment systems usually apply to end items, while the material requirements planning system applies to materials and component used to produce an end item [3].

a. Perpetual Inventory System

A perpetual inventory system keeps records of the amount in storage, and it replenishes when the stock drops to a certain level. This system is based on the concepts of economic order quantity and reorder point. Under this system the reorder point and order quantity are fixed, the review period and demand rate are variable, and the lead-time can be fixed or variable.

With the perpetual system, each time a unit is issued from stock the withdrawal is logged and the stock position is compared with the reorder point. If the stock position is at or lowers than the reorder point, an order is prepared for a fixed number of units. If the stock position is higher than the reorder point, no action is taken. Thus, with the perpetual system there is constant or perpetual accountability on all items.

The advantages of a perpetual system are as follows:

- An efficient, meaningful order size

- Safety stock needed only for the lead time period
- Relative insensitivity to forecast and parameter changes
- Less attention for slow-moving items

A perpetual system can have the following weaknesses:

- If manager do not take the time to study inventory levels of individual items, order quantities tend to be established by clerks.
- Reorder points, order quantities and safety stock may not be restudied or changed for years.
- Delays in posting transactions can render the system useless for control.
- Numerous independent orders can result in high transportation and freight costs
- Large combined orders, which can frequently result in supplier discounts based on dollar value, must be foregone.

b. Two-Bin Inventory System

The distinguishing features of the two-bin system are the absence of a perpetual inventory record. The two-bin system is a fixed order size system, and it has several advantages. The most important advantage is the reduction in paperwork. Records are not maintained for each transaction. The reorder point is determined by visual observation. When the stock in one bin is depleted, an order is initiated, and demands are then filled from the second bin.

The system can even be used with only one bin. An order can be triggered when the inventory level reaches a physical marks such as a painted line or a given volume level. The reorder point quantity can also be placed in a bag or container, so that when the stock is drawn down to the sealed quantity an order is placed.

The two-bin system is best suited for items of low value, fairly consistent usage and short lead-time, such as office supplies, nuts, bolts and so forth.

c. Periodic Inventory System

In a periodic inventory system the number of items in the storage is reviewed at a fixed time interval. A count must be taken of the goods on hand at the start of each period. In the perpetual system an actual count is not required since the inventory records contain receipts, issues, and balances on hand. With the periodic system the quantity to be ordered is not fixed, and the decision maker changes the quantity ordered to reflect changes in the demand rate. Under this system, the review period is fixed; the order quantity, the demand rate, and the reorder point are variable; and the lead-time can be fixed of variable. Figure 2 describes the behavior of the periodic inventory system for a single item. A maximum inventory level E is established for each item. The order quantity is the maximum inventory level minus the inventory position on the review date.

In the perpetual system, replenishment order is initiated as soon as the inventory level drops to the reorder point. In the periodic system, the inventory position is checked only at specified time intervals. The perpetual system treats them discretely and dependently.

Frequently it is worthwhile to treat items in a dependent manner and order them in joint groups. The advantages of joint orders are as follows:

- A reduction in ordering cost may be possible because items are processed under a single order
- Suppliers may offer discounts for purchases exceeding a given dollar volume. The lumping of several items into a single order can be making the discount attainable.
- Shipping costs may be significantly decreased if an order is of a convenient size, such as a boxcar. The simultaneous ordering of several items can result in convenient sizes.

d. Optional Replenishment Inventory System

The optional replenishment inventory system, which is also referred to as a min-max system, is a hybrid of the perpetual and periodic systems. Stock levels are reviewed at regular intervals, but orders are not placed until the inventory position had fallen to a predetermined reorder point. The maximum inventory level is at established for each item. If the inventory position is above the reorder point on the review date, no order is placed. If the inventory positions is at or lower the reorder point on the review date, an order is placed. The order quantity is the maximum inventory levels minus the inventory level at the review period.

e. Material Requirements Planning Inventory System

The material requirement planning (MRP) inventory system is used extensively with planned production. For items that are materials or components used by end items, stock levels are derived from the requirements dictated by the end item. The material requirement planning system is a derived order quantity system.

This system functions by working backward from the scheduled completion dates of end products of major assemblies to determine the dates and quantities of the various component parts and materials that are to be ordered. The system works well when a specific demand for an end product is known in advance the demand for an item is tied in a predictable fashion to the demand for other items.

2.2.1.3 Aggregate Inventory Measurement

Aggregate inventory measurement relates to the overall level of inventory and the techniques for its measurement [3]. Four common ways to measure aggregate inventory are as follows:

- Aggregate inventory value
- Ratio of aggregate inventory value to annual sales
- Days of supply
- Inventory turnover

An organization may use one or more of the above for aggregate inventory measurement. Aggregate inventory value is simply the total value of inventory at cost. Many organizations set dollar limits or budgets on the amount that can be invested in each general class of materials. They are usually applied to broad classes and not to individual items. The dollar limits indicate the upper investment limit which aggregate inventory value should not exceed. Aggregate inventory value is very simple and easy to use, but it neglects the dynamic nature of inventory value and its other financial interactions.

The inventory to sale ratio is the aggregate inventory at cost divided by annual sales. This ratio recognizes the dynamic relation between inventory and sales, but it can vary substantially due to cost and selling price changes. If profit margins change, the ratio can become distorted for comparison purposes.

The days of supply is the total value of inventory at cost divided by the sales per day at cost. The time supply of inventory is dynamic in nature, but it can become confounded if the cost of sales is not maintained and controlled.

Inventory turnover refers to the cycle of using and replacing materials. It is the ratio of the average inventory at cost to the annual sales at cost. It indicates the number of "turnovers" of the investment in inventory for a given time period (usually a year). If an organization sells $600,000 worth of products a year and has an average inventory valued at $300,000, it has two turnovers per year. If the organization could generate the same sales with an average inventory worth $150,000, it would have four turnovers per year.

Aggregate inventory measurement techniques usually reduce inventory items to a common financial denominator of dollars. The techniques measure result in absolute terms or ratios. The desirable range of performance is established historically by industry data, or by management judgment. While measurement in financial terms is desirable, inventory should also be viewed through other dimensions.

2.2.1.4 Review of Current Inventory Management Application

In order to understand how the inventory management application in the real world function, several current inventory management application have been studied.

They are many applications have been studied, but only three applications that are more suitable were discussed. They are:

- Big Business
- FF Inventory PRO 98
- Win sell Express

Study of these applications focused on what functionality that are provided and the design of graphic user interface. Below are the functions that are provided for each application:

2.2.1.4.1 Big Business [4]

- View customer and vendor information, including address and financial data
- Customer Card, Quote, Sales Order, Invoice, and Receive Money
- Create customer orders
- Record sales
- Generate invoices without line item details
- Assess finance charges against customers with past due invoices

- Enter payments that you receive and apply them to invoices
- Create purchase orders
- Record purchases from vendors
- Enter and apply payments
- View and change your bill payment schedule
- View pricing, purchasing, and other item information
- Add or remove items from inventory
- View and change information for existing categories, create new categories, and review and change budgets

2.2.1.4.2 FF Inventory PRO 98 [5]

- Vendors, purchase orders, receivers, invoices, payments, vendors and products. Limited only by hard-disk space!
- Includes the award-winning Billing Manager PRO for invoicing.
- Supports user-defined code/ sort fields.
- Provides automatic and 9 batch price-changing levels.
- Automatically calculate sale prices from list or cost.
- Displays real-time stock levels when entering PO's or invoices.
- Supports automatic or manual product receivers.
- Features stock reorder and overstock level watches.
- Provides for non-stock and non-depleting stock codes.
- Separate product descriptions for PO's and invoices.
- Create line item and service (memo) PO's and invoices.

- POWER reporting with speed-sort, export and sub-total options.
- Provides for partial payments and deposits on account.
- Prints user-defined, multi-page PO's and invoices using any text, fonts and colors!
- Print packing slips, file and remittance copies.
- Bills late charges using flat fee or interest percentages.
- Batch print PO's, invoices and customer statements.
- Features password protection and audit trails.
- Partial searches on vendors, customers, products, invoices, etc.
- Track point-of-sale and cash-based transactions.
- Create and print customized form letters including customer data.
- Backup data files to any device supported by Windows.

2.2.1.4.3 Win Sell Express [6]

- Editable "Added Items" grid. Can edit quantity, tax and unit price right in the grid. Just highlight the field and type the new value.
- Ability to delete line items.
- Lint item comments can be added to each line item on the invoice grid for reference.
- Full customer order capabilities.
- Edit and re-post orders.
- Easily turn orders into invoices.

- Re-price function allows you to recalculate prices based on current margin and new costs or recalculate margins based on current price and new costs.
- Gives purchase history for each item on the vendor receipt.
- Allows ad hoc price changes.
- Ability to define multiple currencies with exchange rates.
- Shows "Amount Due" in foreign currency.
- Give change in local currency.
- Invoice void function. Allows posted invoices to be voided. Protected by security levels.
- Purge history feature. Allows historical data to be archived and the database to be purged of old transactions.

2.2.2 Related Development Concepts

Related development concepts include

- Information System Planning concepts
- System architecture concepts
- Database implementation concepts

2.2.2.1 Information System Planning Concepts

In this section a new methodology for information system planning introduced by reference [7] will be review.

The methodology is composed of the following stages [7]:

- Definition of the organizational processes and objectives
- Definition of the computer-related goals set for each organizational objective
- Definition of the alternative information system processes required for each of the computer-related goals
- Identification of alternative information system processes and selection of preferred system
- Preparation of a request for proposal
- Selection of the preferred alternative

2.2.2.2 System Architectures Concepts

a. Client/ Server Databases

Client server database splits the database processing between two systems: the client PC that runs the database application and the database server which runs all or part of the actual DBMS [8].

For example, a database server provides data in response to an SQL request issued by the client application. Local processing by the client might calculate the invoice amount and format the response to the workstation screen [9].

i) Advantages of Client/ Server Databases

The primary advantages of a client/ server system arise from splitting the processing between the client system and the database server. Sine the bulk of the database processing is done on the back-end, the speed of the DBMS is not tied to the speed of the workstation. As a result, the workstation need only be run the front-end software, effectively extending the life many older of smaller PCs which do not have the horsepower needed to run on complex DBMS [8].

Another benefit of separating the client from the server is workstation independence; users are not limited to one type of system or platform. Besides that, client/ server is the preservation of data integrity.

This division of work also reduces the load on the network connecting the workstations. Instead of sending the entire database file back and forth on the wire, the network traffic is reduced to queries to and responses from the database server [8].

ii) Disadvantages of Client/Server Databases

The major disadvantage of client/ Server is the increased cost of administrative and support personnel who maintain the database server [8].

There is also an increase in hardware costs. The database server should run on its own dedicated machine to ensure performance and data integrity. This usually means purchasing a high-powered system with a large amount of RAM and hard disk space [8].

There is also the issue of complexity. With so many parts comprising the entire client/ server system, it is harder to track down problem when the system crashes.

b. Parallel Architectures for Databases [10]

To achieve a reduction of the communication and data conversion costs, the DBMS must be split into a client DBMS, running with the application on the same workstation, and a server DBMS, running on the server machine. A data caching mechanism may thus be used to reduce client-server communication and allow a local navigation in the client machine [10]. This method is known as parallel architectures for database. On a cache miss, the client sends a page or an object demand to the server. The server is thus reduced to a page server, consuming few CPU resources.

With this architecture, a multiple cache coherency protocol is needed, either to make sure that an update in a page is automatically propagated to all clients who cache it or to invalidate pages that are no longer up-to-date.

i) Advantages of Parallel Architectures for Databases

- Reduced communication thanks to client cache
- Use of the local client's processors (less costly than server's processors) to achieve inter-query parallelism
- Offload the server of some DBMS work.
- Simple server

ii) Disadvantages of Parallel Architectures for Databases

- Swapping if the client's memory is not large enough
- Cost of the multiple copy coherency protocol
- High communication volume if there is little data locality
- Complex to optimize

2.2.2.3 Database Implementation Concepts

Database implementation concepts include

- Relational database

- Database Management System
- Relational Database Management System
- Structured Query Language
- Stored Procedure

a. Relational Database

The basic data components of a relational database model are entities and their attributes. These basic data components fit into a logical construct, known as a table. Thus, a relational database is composed of tables. The process of organizing data into tables in a consistent and complete format is referred to as normalizing the database [11]. Each independent table can be related to each other by creating relationships among them. A primary key of a table is a field of this table selected to uniquely identify all other attribute values in any given row, and cannot contain null entries. A foreign key is an attribute, or a combination of attributes in one table, whose values must either match the primary key in another table or be null.

b. Database Management System

A database management system (DBMS) is an environment specially created for the purpose of working with databases. It refers to an electronic system, or a computer program designed to work with databases [11]. Examples of DBMS include Microsoft Access and FoxPro.

c. Relational Database Management System

In a relational database management system (RDBMS), such as SQL Server, a database is not necessarily tied to a file. It is more of a logical concept that is based on a collection of related objects [12].

c. Structured Query Language

Structured Query Language (SOL) is designed specially for database queries. This language has three main parts, namely Data Definition Language, Data Manipulation language, and Model Language [13]. Data Definition Language (DML) is used to manipulate data stored in the database, such as inserting new records, deleting records, and retrieving records. SQL can be used within source codes from programming languages, such as COBOL. This is called embedded SQL. Therefore, it is a Model Language.

d. Stored Procedure

A stored procedure is a series of precompiled SQL statements and control-of-flow language statements. Stored procedures can enhance standard SQL by allowing you to use parameters, make decisions, declare variables, and return information [14].

2.2.3 Development Tools

A review on development tools was carried out in order to identify the most suitable tools to develop this system.

This review is divided into three parts. The first part focuses on the tools to develop the processing part and the interface part of the system. The second part focuses on the tools to develop the database of the system. The third part discusses the operating system for the development.

2.2.3.1 Program Development Tools

The target of review included Microsoft Visual Basic 6.0, Microsoft Visual C++ 6.0 and Delphi.

a. Microsoft Visual Basic 6.0.

The first review is Microsoft Visual Basic 6.0. Its Standard module enables production of stand-alone applications. It combines RAD and object-oriented programming.

The new tools are built on ActiveX Data Objects (ADO) 2.0, Microsoft's standard high-level interface for data access. This has a simple object model, a common user interface for local and remote data access, support for hierarchical record sets (tables with data derived from multiple source tables, say) and accessible data binding [15].

ADO components are created during development by dragging the required data elements, such as tables; views and fields, from the new Data Environment Designer into screen forms to create data bound objects. The Data Environment Designer is used to define your project's data sources, and the tables, views and stored procedures that are available from each source. These can then be dragged into the appropriate positions on your forms.

Microsoft Visual Basic 6.0 was also an interpreted language system, so users could test and debug application on the fly from within the development environment.

i. Advantages of Microsoft Visual Basic 6.0.

- Component-based development aids and database access model
- Server-side transaction support
- Integration with Web clients.

ii. Disadvantages of Microsoft Visual Basic 6.0.

- Performance remains second-rate compared with C++ and other fully compiled languages.

b. Microsoft Visual C++ 6.0

Visual C++ is Microsoft's version of the C++ programming language. Based on the C programming language, C++ is an improved version of C that takes the C language to the next level of evolution of programming languages, those that provide object oriented programming.

The Enterprise Edition of Microsoft Visual C++ 6.0 provides many tools and components for building and validating enterprise-level distributed Component Object Model (COM) applications. Visual C++ also provide ADO and OLE DB [16].

i. Advantages of Microsoft Visual C++ 6.0

- Good performance
-

ii. Disadvantages of Microsoft Visual C++ 6.0

- Not really a RAD tool, so it is difficult to be used for window programming compare to Visual Basic and Delphi.

c. Delphi

Delphi is Borland's best-selling rapid application development (RAD) product for writhing Windows applications. With Delphi, Windows programs can be written more quickly and more easily than was ever possible before. User can create Win32 console applications of Win32 graphical user interface (GUI) programs. When creating Win32 GUI applications with Delphi, user have all the power of a true compiled programming language (Object Pascal) wrapped up in a RAD environment. What this means is that user can create the user interface to a program (the user interface means the menus, dialog boxes, main Window, and so on) using drag-and-drop techniques for true rapid application development [17].

Borland build Delphi around its powerful Object Pascal language. Though Object Pascal's roots are in standard Pascal, it includes true object orientation with support for inheritance, encapsulation, and polymorphism. The Pascal derivative in Delphi is sophisticated and powerful, yet easier to comprehend and manage than C++. Like Microsoft Visual Basic, Delphi uses dot notation to reference object's properties and methods. Unlike Microsoft, however, the Delphi language relies consistently on objects and allows you to create your own classes by sub classing from other objects and inheriting their properties and methods. This powerful feature, the cornerstone of OOP, gives user significantly more control in creating well-structured applications. It's also an excellent vehicle for code reuse [15].

i. Advantages of Delphi

- Distributed data set processing
- Full control of Microsoft Transaction Server interactions
- Speed comparable to most C++ compilers.

ii. Disadvantages of Delphi

- Visual Studio's integrated Multilanguage packaging

2.2.3.2 Database Implementation Tools

Review was done to identify the most appropriate database management system to store and manage acquired data. Selection criteria were based on usability and effectiveness in the context of cross platform deploying, storage space required, and the portability of the records.

The target of review included Microsoft Access 97, Microsoft SQL Server 7 and Oracle.

a. Microsoft Access 2000

Microsoft Access is a relational database management system that is used to create the relational databases [11]. By using the ODBC driver for Access, data can be retrieved from the database in a client/ server based system.

b. Microsoft SQL Server 7.0

Microsoft SQL Server is a client/ server based relational database management system [8]. Microsoft SQL Server is an ideal database engine for web sites because it can be queried and updated via popular web browsers such as Internet Explorer. Microsoft SQL Server 7.0 runs in Windows NT and not Windows 95 or Windows 98. Microsoft SQL Server is a significant tool in many regards. From data warehousing to applications that require not only a large amount of information, but also many different simultaneous users, SQL server is a key component in answering data management requirement. It is a powerful and comprehensive database. [18]

Microsoft SQL Server is a perfect example of an n-tier system. The user can manipulate the data directly from the client side. Most of the time, the data is validated first before it is updated into the database in server side. It is tightly integrated with the Microsoft BackOffice family product to enable organization to improve decision-making and streamline the business process. It is the best database for Windows NT Server.

Microsoft SQL Server maintains referential integrity and security and ensures that operation can be recovered in the event of numerous types of failure. SQL server can control the access for the type of information that can be retrieved by the user.

SQL Server supports Internet database integration. It allows the user to automate the publishing of database information in HTML documents. It allows us to build active web sites and let us conduct processes on the Internet. When combining with Internet Information Server and the SQL server Internet Connector, it gives user the complete Internet database publishing capabilities.

It provides the function for transparent distributed transactions. This means that it provides automatic distributed update capability across two or more SQL Server transparent to the desktop application, making it a simple to use. It guarantees the integrity of transaction of updating spanning multiple servers.

i. Advantages of Microsoft SQL Server 7.0

The most significant advantage of the Microsoft SQL Server is the wide variety of third-party client support available. Every type of front-end software is available, from those designed strictly for application development to add-in SQL Server access modules for standard PC-based DBMSs such as Paradox and dBASE. There are even access modules that let users query the database from the leading spreadsheet program [8].

Microsoft SQL Server 7.0 running on Windows NT is an advantage. Window NT database server is network independent and can be used with any of the popular NOSs.

ii. Disadvantages of Microsoft SQL Server 7.0

Windows NT Advanced Server is not as popular as other NOSs. Adding a Windows NT server to another NOS can be done, but it does increase the level of complexity in the overall system [8].

c. Oracle 8I

Oracle Corporation's reputation as a database company is firmly established in its full-featured, high-performance RDBMS server. With the database as the cornerstone of its product line, Oracle has evolved into more than just a database company, complementing its RDBMS server with a rich offering of well-integrated products that are designed specifically for distributed processing and client/ server applications. As Oracle's database server has evolved to support large-scale enterprise systems for transaction processing and decision support, so too have its other products, to the extent that Oracle can provide a complete solution for client/ server application development and deployment [19].

Oracle is the world's leading vendor of database software. Oracle's ability to have all data and documents stored in a small number of high-performance databases benefits customers by centralizing all their data, making information management and access easier, more reliable, and less expensive. [20]

The ground-breaking capabilities of Oracle8*i*'s Internet File System (IFS) provides a single, easy to use data management interface for all data types, thus minimizing customers' reliance on a proprietary operating system. Oracle is an open solution and it supports all kind of platform.

Oracle's advanced security features allow for enforced granular privileges, advanced auditing, enhanced access control, secure distributed processing and replication, and the ability to use additional external authentication mechanisms.

Oracle uses a Java-based utility that provides everything needed to get a pre-tuned and pre-configured Oracle8*i* database up and running. Oracle Enterprise Manager provides a single integrated management console for central administration of multiple servers. It also contains some advance functionality for tuning and diagnosing the database, and managing complex change in the database environment.

i. Advantages of Oracle 8I

Oracle 7 has the second-largest number of third party, front-end support product. Only Microsoft SQL Server has slightly more support [8].

Oracle's database servers are popular due to their scalability across a wide range of platforms [8].

Oracle 7 support for almost all of the major procedural languages. It is relatively simple matter for an experienced programmer to learn PL/ SQL and start creating Oracle 7 applications right away.

ii. Disadvantages of Oracle 8I

Total cost of ownership for Oracle is high.

2.2.3.3 Operating System

a. Microsoft Windows NT

Microsoft Windows NT is one of the powerful operating systems for business computing. It combines the ease-of-use of Windows 95 with the power and reliability of Window NT. NT is also a powerful OS that reliable, secure, multithreaded, symmetric processing, support client/server system. [21]

There is an extensive security support in NT. NT can control the access control of user in accessing certain file or application. This can use for implemented the access control for the Attendance Management System. Besides, NT supports a wide range of networks protocol and Remote Access Protocol. This makes it easy for us to develop the distributed application.

Windows NT Server is a complete platform available for building and hosting web-based application. It is the best platform to publish and share information securely over corporate Intranet and Internet. It is so reliable that when an application have problem it doesn't crash the whole program.

Windows NT allows Object Linking and Embedding (OLE). It can combine the information from several applications into one compound document using the special OLE capabilities of window-based application.

Window NT also enables the capabilities of integrating applications on a single computer or even across multiple computers.

b. UNIX

UNIX is an increasingly popular operating system. Traditionally used on minicomputers and workstations in the academic community, UNIX is now available on personal computers and the business community has started to choose UNIX for its openness.

UNIX, like other operating systems, is a layer between the hardware and the applications that run on the computer. It has functions that manage the hardware and functions that manage the executing of applications

UNIX includes the traditional operating system components. In addition, a standard UNIX system includes a set of libraries and a set of applications. It includes the file system and process control and a set of libraries.

One of the greatest strength of UNIX is the consistent way in which it treats files. It is very easy for the users to work with files because users don't need to learn special commands for every new task.

Besides UNIX is not known only for its longevity and versatility as an operating system, but also for the variety and number of utility programs that called tool.

c. Windows 95

Windows 95 is an operating system for a single machine. Windows 95 was designed especially for home users and for people in small offices. It works great on notebook (portable) computers, it's easy to connect with networks, and the Plug and Play features make it a lot easier to add a new piece of hardware.

2.2.3.4 ODBC

ODBC allows a single uniform language to access different databases, instead of using the propriety language of each data source by designing a standard set of API's [Fleet,

1997]. By using ODBC, you can deal with a level of abstraction above the particular database. It is based on the X/Open Call-Level Interface and uses SQL.
When one accesses a database through ODBC, the database must be registered as an ODBC data source. By registering the database as a data source, the application only needs to know its data source name. The location of the database makes no difference, nor even what the type of database it is.

There are three types of data sources you can create: System, User, and File. A system data source is one that is available to any user of the system. A user data source is one that is only available to that user. Finally, a file data source is a description of the database. It can be used to hook up to the database without having to register the database itself with the system.

There have been a few different versions of the ODBC API that have been released. The current version is ODBC 4.0. This refers to the revision level of the API. There have been several changes to the API resulting in versions: 1.0, 2.0, 2.5, 3.0 and now 4.0.

Typically when you install ODBC on your machine you are installing the ODBC Manager.

Win32::ODBC was written to conform to the ODBC API 2.0 specification. Since ODBC 3.0 is backward compatible it should not cause any compatibility problems.

ODBC 3.0 is completely backward compatible. When using the ODBC 3.0 driver manager:

- An ODBC 2.x application works with an ODBC 3.0 driver or an ODBC 2.x driver.
- An ODBC 3.0 application works with an ODBC 3.0 driver.
- An ODBC 3.0 application works with an ODBC 2.x driver as long as the application uses only ODBC 2.x features.

Where necessary, the ODBC 3.0 driver manager performs appropriate function and/ or argument mapping to ensure backward compatibility.

And the ODBC drivers available in ODBC version 4.0 are,

- Microsoft Access
- Microsoft dBase
- Microsoft Excel
- Microsoft FoxPro
- Microsoft ODBC for Oracle
- Microsoft Paradox
- Microsoft Text
- SQL server

2.3 Summary

2.3.1 Inventory Management System

Basically, inventory management system can be divided to four main types; they are the perpetual, two-bin, periodic, optional replenishment, and material requirements planning systems. Below is the summary table for each type.

	Perpetual	**Two-bin**	**Periodic**	**Optional Replenishment**	**Material Requirement Planning**
Order Quantity	Fixed	Fixed	Variable	Variable	Variable
Reorder point	Fixed	Fixed	Variable	Fixed	Variable
Review period	Variable	Variable	Fixed	Fixed	Fixed/Variable
Demand rate	Fixed/Variable	Fixed/Variable	Fixed/Variable	Fixed/Variable	Fixed
Lead time	Fixed/Variable	Fixed/Variable	Fixed/Variable	Fixed/Variable	Fixed/Variable
Safety stock	Medium	Medium	Large	Very large	Small/none

Table 2.1 Type of Inventory Management System

Aggregate inventory measurement relates to the overall level of inventory and the techniques for its measurement. Four common ways to measure aggregate inventory are as follows:

- Aggregate inventory value
- Ratio of aggregate inventory value to annual sales
- Days of supply
- Inventory turnover

After surveying the three real world applications, it can be concluded that although there are a lot of features for each application, but all these features can be grouped into six main categories.

Table below show the main features for each category.

Customer	Vendor	Sales	Inventory	Purchase Order	Report
• View customer • Add, update and delete customer record • Search record	• View vendor • Add, update and delete vendor record • Search record	• View sales • Add, update and delete invoices • Search record	• View inventory • Add, update and delete inventory record • Prices update • Receive order • Search record	• View purchase order • Add, update and delete purchase orders • Vendor's purchase orders, and Bill • Search record	• Generate various type of reports

Table 2.2 Main functions of current inventory management application

2.3.2 Related Programming Concepts

The information system planning methodology is composed of the following stages:

- Definition of the organizational processes and objectives
- Definition of the computer-related goals set for each organizational objective
- Definition of the alternative information system processes required for each of the computer-related goals
- Identification of alternative information system processes and selection of preferred system
- Preparation of a request for proposal
- Selection of the preferred alternative

A client-server computational model implies a relationship between two processes in which one makes request to the others. To achieve a reduction of the communication and data conversion costs parallel architectures for database can be applied.

- Database implementation concepts include
- Relational database
- Database Management System
- Relational Database Management System
- Structured Query Language
- Stored Procedure

2.3.3 Development Tools

Microsoft Visual Basic 6.0, Microsoft Visual C++ and Delphi are providing standard module enables production of stand-alone applications.

Microsoft Access is a relational database management system that is used to create the relational databases. By using the ODBC driver for Access, data can be retrieved from the database in a client/ server based system. Microsoft SQL Server is a client/ server based relational database management system. Microsoft SQL Server 7 runs in Windows NT and not Windows 95 or Windows 98. Finally, Oracle has evolved into more than just a database company, complementing its RDBMS server with a rich offering of well-integrated products that are designed specifically for distributed processing and client/ server applications.

Windows NT Server 4.0 is a network operating system. New management tools in Windows NT Server 4.0 include setting up of web sites. UNIX is available on personal computers and the business community has started to choose UNIX for its openness. While, Windows 95 is an operating system for a single machine.

Chapter 3 SYSTEM ANALYSIS

3.1 Analysis Current Inventory System in FCSIT

The current inventory system use by the faculty was provided by the BPK (Bahagian Perkhidmatan Komputer) in the Faculty of Science. This inventory system was developing in the year of 1996 using Dbase 3 and it is running in Dos mode. Until now, it is still use in every faculty and also in the "Jabatan Harta Benda" for inventory management.

The inventory records in this system are stored in a number of data files (with the extension .dbf). These data files are InvDept file, InvDir file, InvParts file, Gunasama file, Inventry file and InvDel file.

The InvDept file is used to store the Department Code, the faculty name an the department name. The InvDir file is used to store the item code, item name and group name for the purpose of categorization of the item in the inventory. The Inventry file is used to store the detail of the item. The InvParts file is used to store the details of the parts/components of an item. The Gunasama file is used to store the detail of the items that will be share among the department. The InvDel file is used to store the detail of the item, which has deleted from the inventory.

In this system, it using the Item Tag identifies all items. This tag number is an auto-generate number. It will be assigned to a particular item when the item is first recorded in the database. The tag number is consisted of two parts. The first part is the department code, which is normally used to identify which department the item is belongs to. The second part is the item number, which is use to identify the items within the department.

These items are categorized by using Item Code. The "Jabatan Harta Benda" provided it. The items code is consist of 12 alphanumeric characters, which is separated into 3 parts. The first part is the major category of the item; the second part will be the sub- category of the item and is same for the third part.

There are six module provided by the system for managing the inventory. These modules are ADD, DELETE, FIND, DELETE, PRINT, GUNASAMA and UTILITIES.

3.1.1 Module ADD

This module provides four options for adding items into the inventory file. These options are 'add items', 'add item components', ' add similar items' and 'add tagged items'.

Option 'add items' is for entering new items into the inventory file. There are two categories of items, which are items without components and items with components. Examples of Items without components are furniture and general office equipment. Example of items with components is microcomputer with components like mouse and monitor. To differentiate between the two categories, the field "PARTS" is used. For items without components, PARTS = 0. For items with components, PARTS = n where n is the number of parts.

Option 'add item Components' is for entry of components that are part of an existing item. If for some reason any component was left out when the main item was being entered, the component can be entered using this option. However, it does not include components that are purchased separately, e.g. spare parts, which are purchase on a separate purchase order.

Option "add similar items" is for entries of similar items e.g. bulk acquisition of a certain item like forty microcomputers (of same kind). It reduces the number of keystrokes for repetition.

Option 'add tagged items' is for add items that already have item tag number which was given by another department. This does not include items that are on loan.

3.1.2 Module EDIT

In this module, there are two option provided by the system, which are 'edit items' and 'edit item components'. Option 'edit items' is for editing the information of particular items, which is accessed by the item tag number. Option 'edit item components' is for editing the information of a particular component of a particular item. The item tag number and subtag number access the item component.

3.1.3 Module FIND

This module provides two options for searching/ enquiry an item. The first option is 'search by item Tag" which allows for search/enquiry for a particular item and its components by keying the items tag number. Scrolling from one item to another is allowed. The second option "search by item code" allows item to be searched by the item code. All items with similar item code will be search and displayed.

3.1.4 Module DELETE

The two option provided by this module is 'delete item' and 'delete item components'. Option 'delete item' allows items to be deleted from the inventory file. Once the item is deleted, its item tag number cannot be reused. The second option 'delete item components' allows for deletion of item component of an item.

3.1.5 Module PRINT

This module provides three options for printing a listing of the inventory items. The first option, 'print by item tag' will generate a listing of all inventory items ordered by item tag. The second option, 'print by room' will generate a listing of all inventory items ordered by location/room number.

The last option 'print by selected items" contains four sub-option. The first sub-option is 'by item tag', is for listing of selected items specified by item tag. This requires user to specify the item tag to start printing from and how many items to print starting from the item tag specified. Other sub-options, which are 'by items code', 'by item name', 'by room', allow printing of a list of selected items specified by item code, item name and location/room number respectively.

3.1.6 Module GUNASAMA

This module is for entering items to be shared among the department into the Gunasama file. The item must already be in the Inventory File. Otherwise, it will be rejected. Options provided by this module are 'Add', 'Edit', 'Find', 'Delete' and 'Print' ('by item tag', 'by item code' and 'by item name'). All of these options are same as the options mention in the previous modules.

3.1.7 Module UTILITIES

Some of the important options provided by this module are 'Backup Data files', 'Restore Data Files', 'Vet Item Codes' and 'Vet Item Tags'. Option 'Backup Data files' enabled user to backup data files into diskettes. The backup data files will be needed to restore the data files in the system if any of them gets corrupted. The user is advised to use this utility after every session ie. Before exiting from the system. Option 'Restore Data Files' allows user to restore the latest backup of copy of the data file into the system. Option 'Vet Item Codes' allows for vets the inventory data file for erroneous item codes. Lastly, option 'Vet Item Tags', allows for vets the inventory data file for duplicate item tags.

3.1.8 System Strength

There are several system strengths that is available in this system that make it continue using for inventory management in the faculty.

- **Easy to install**

  The system is very small which can be put in a single diskette. To install, it just need to copy the program file into the directory of the computer which have the Dbase program installed, or the program file itself already compiled with the dbase system file to convert into setup file, for installing into the computer with no Dbase program installed.

- **Use little resources**

  Since the system is a Dos program, it takes very little system resources to run. Therefore, it can be run the old generation like 386 computer.

- **Easy to backup and restore of inventory Record**

  The inventory file for this is so small that it is able to store in a single piece of diskette. The system has provided the option for backing up the inventory file into the diskette. If the inventory data in the computer has corrupted, it can be restore from the diskette into the computer.

3.1.9 System Weakness

Below are the weakness that resides in the current system.

- **Not user friendly**

  There is no mouse support for the current system. The user has to type in many keystrokes to perform their job. Beside, the interface is very simple interface with limited detail of information can be stored in the computer. For e.g. it didn't store the detail information for the vendor. Beside, the search capability of the current system is very limited. For example, it cannot differentiate between among hardware, software and office equipment. It also does not provide information like how many a particular item available in the faculty.

- **Not Security enabled**

  It does not include a security module that allows only authorized people to access the system. Therefore, the record in the system is not safe and may be edited by any people without able to trace who did it.

- **Cannot support multi-user environment**

  The current system is a standalone application. Only one person can use the system in a time. If more than one user have to use the current system, another copy of the current system have to be install to another computer. This has allows the duplication of data for common data. Any update to common data will have to be doing separately to every computer that has install the system.

- **No online support**

  The current system is not accessible from the internet/intranet. Therefore, users have to go to the computer, which have installed the current system instead of accessing from any computer using normally web browser. This is certainly very inconvenient for the users.

- **Limitation in types of data that can be storable in the system**

  There is a limitation of types of data that can be stored in the current system. For example, it will not able to store data types like image file.

3.2 System Analysis

During this stage, we have to work on some analysis process like gathering and interpreting facts and diagnosing problems. The result of this process will be used to recommend improvement to the system. [22] Through system analysis, we may add, delete and modify system components toward the goal of improving the overall system. The information gathered during this phase has provided alternative strategies to develop this system.

It is the process of developing alternative solutions and choosing among those solutions. Regards to the reviews, it is also analysis into three main areas. They are

- Inventory Management System
- Related programming concepts
- Development tools

3.2.1 Inventory Management System

From the finding, it can be concluded that most of the current inventory management systems only provided functionality on managing the inventory without providing any analysis ability. Based on the books that have been studied, in order to manage the inventories effectively, a lot of analyses have to be done.

3.2.1.1 Type of inventory management system

Common types of inventory management systems are the perpetual, two-bin, periodic, optional replenishment, and material requirements planning systems.

With the perpetual system, each time a unit is issued from stock the withdrawal is logged and the stock position is compared with the reorder point. It is not wrong to say that perpetual system is an online system. Perpetual system is not suitable for inventory management system for FCSIT.

Two-bin system is best suited for items of low value, fairly consistent usage, and short lead-time, such as office supplies, nuts, bolts, and so forth. This obviously means that two-bin system is not suitable for inventory management system for FCSIT.

In a periodic inventory system the number of items in the storage is reviewed at a fixed time interval. Periodic inventory system can be considered a suitable type for inventory management for FCSIT.

In optional replenishment, the order quantity is the maximum inventory level minus the inventory level at the review period if the inventory position is at or lower the reorder point. This mean that the stock level of an item will be added to the maximum level if the item position is at or lower the reorder point which is not true for inventory management for FCSIT. In real world, FCSIT will order an item if only the item is needed.

The material requirement planning (MRP) inventory system is used extensively with planned production. This means that this system is more suitable for the manufacturer.

3.2.1.2 Aggregate Inventory Measurement

Aggregate inventory measurement relates to the overall level of inventory and the techniques for its measurement. Four common ways to measure aggregate inventory are as follows:

- Aggregate inventory value
- Ratio of aggregate inventory value to annual sales
- Days of supply
- Inventory turnover

Aggregate inventory value has it own weaknesses, but it is quite suitable for inventory management for FCSIT because they are usually applied to broad classes and not to individual items.

The inventory to sale ratio can vary substantially due to cost and selling price changes. This make inventory to sale ratio is no suitable because FCSIT don't have selling item so that indirect it is not selling price at all.

The days of supply is the total value of inventory at cost divided by the sales per day at cost. It is not wise to implement this technique since are a lot of items for a FCSIT.

The inventory turnover is not suitable for inventory management system for FCSIT. It is quite suitable for inventory management system for supermarket and real world supermarket.

3.2.1.3 Review of Current Inventory Management Application

After analysis the three real worlds, it concluded all features could be grouped into six main categories, but for FCSIT just only grouped into four main categories. There are only vendor, inventory, personal, report because FCSIT have not selling item. Table below show the main features for each category.

Vendor	**Inventory, Parts, Department, Directory**	**Personal**	**Report**
• View vendor • Add, update and delete vendor record • Search record	• View inventory, parts, department, and directory. • Add, update and delete inventory, parts, department, directory record • Search record	• View Personal • Add, update and delete Personal record • Search record	• Generate various type of reports

Table 3.1: Main functions of current inventory management for FCSIT

3.2.2 Related Development Concepts

3.2.2.1 Information System Planning Concepts

The methodology for information system planning introduced by reference [BERG93] is very suitable to be used as a guide in building the inventory management system for FCSIT because it provide a systematic steps in planning an information system.

3.2.2.2 System Architectures Concepts

A client-server computational model implies a relationship between two processes in which one makes request to the others. It provides a lot of advantages and it is suitable to be used in this case in order to increase the performance of the system. Sine the bulk of the database processing is done on the back-end, the speed of the DBMS is not tied to the speed of the workstation. As a result, the workstation need only be run the front-end software, effectively extending the life many older or smaller PCs that do not have the horsepower needed to run on complex DBMS. This can really save some cost.

Although parallel architectures for database can achieve a reduction of the communication and data conversion costs, but this concepts is complex to optimize and the cost of the multiple copy coherency protocol is high. However, some of the concept can be modified and adopted in the inventory management system for FCSIT.

3.2.2.3 Database Implementation Concepts

Relational database can be considered suitable to be implemented in the inventory management system for FCSIT because relational database has complete flexibility in describing the relationships between the various data items.

A stored procedure is suitable to be used to communicate with the server. In a client/ server architecture, as the number of users increases, so does the number of SQL request, and the network can quickly become the performance bottleneck. Stored procedure offer performance improvement because a single call to a stored procedure can invoke multiple SQL statement.

3.2.3 Development Tools

A review on development tools was carried out in order to identify the most suitable tools to develop this system.

This review is divided into three parts. The first part focuses on the tools to develop the processing part and the interface part of the system. The second part focuses on the tolls to develop the database of the system. The third part discusses the operating system for the development.

3.2.3.1 Program Development Tools

Although Microsoft Visual Basic 6.0, Microsoft Visual C++ 6.0 and Delphi are suitable to build an application program.

In this case Microsoft Visual Basic 6.0 will be chosen because all the needs of the application program that going to be built can be fulfilled by Microsoft Visual Basic 6.0. It also provides a convenient method for building user interfaces compared to Microsoft Visual C++.

Microsoft Visual Basic 6.0 is the easier of the two to learn. We simply cannot beat Basic for its simplicity and readability. Microsoft Visual Basic 6.0 is also more forgiving of poor coding style, making it friendlier to new programmers. One example of this is Microsoft Visual Basic 6.0 default Variant data type. A Variant can handle numeric, string, or date/ time values. While the Variant type makes it easy to code without considering the data types of your variables, it can also lead to sloppy coding and wasteful use of system memory (Microsoft recommends that a Variant be used only in certain special cases). Delphi, on the other hand, is anything but forgiving in this respect. Microsoft Visual Basic 6.0 is also an interpreted language system, so users could test and debug application on the fly from within the development environment.

3.2.3.2 Database Implementation Tools

SQL Server is more ease-to-use system compared to the oracle database. It provides more user-friendly graphical tools for installation, configuration and administration. SQL Server can run only on windows but for oracle the operating system will essentially become irrelevant. Oracle supports all kind of platform. [20]

With integrated management of text, images, audio and video, Oracle8*i*'s inter-media enables customers to take advantage of the multimedia nature of the Web. For SQL, it advocates a strategy of storing non-traditional data in flat files in separate servers and linking them together using OLE-DB.

Microsoft SQL Server 7.0 tightly integrated with other Microsoft Products

- Seamless integration with Window NT provides security, a web application environment and Microsoft Transaction Server support
- Integration with Microsoft Exchange Server provides reliable and scalable Internet and Intranet collaboration and messaging - supporting SQL server initiated trigger and store procedure-based messaging and replication of Exchange public folders.

3.2.3.3 Operating System

Microsoft SQL Server and Internet Information Server 4.0 run on Windows NT Server 4.0 and Window will be used as a server. Windows 95 is an operating system for a single machine. It can be used for initial development of this project and as a client.

There is a lot of supporting software for Window NT especially the free downloadable option packs.

UNIX is hard to install compared to Window NT 4.0. Sometimes it take two weeks to configure a UNIX machine but only one day to set up a NT box. [21]

NT 4.0 supports the Microsoft Back Office Product but UNIX doesn't. To use a UNIX OS, a lot of command needs to be entered. Compared to NT, it provide user-friendly interface that ease the job of the user

3.3 Requirement Analysis

Requirement analysis is done during analyzing system needs. Requirement analysis activities include analyzing and determining functional requirement and non-functional requirement of inventory management system for FCSIT.

3.3.1 Functional Requirements

Functional requirements are functions or features, which are expected by the user, and stated by them to be incorporated into the system [23]. The system is considered incomplete if any of the necessary functions is not included. This section states the functional requirements for server database and Inventory application.

3.3.1.1 Server Database

1. All unneeded archive records should be deleted automatically at certain time.
2. The referential actions supported by the foreign key integrity constraints should include update cascading. The update cascade action specifies that when rows containing referenced key values are updated. Therefore, the update cascades. For example, if a row in a parent table is update, and this row's primary key value is referenced by one or more foreign key values in a child table, the rows in the child table that reference the primary key value are also updated from the child table.

3. The referential actions supported by the foreign key integrity constraints should include delete cascading. The delete cascade action specifies that when rows containing referenced key values deleted, all rows in child tables with dependent foreign key values are also deleted. Therefore, the delete cascades. For example, if a row in a parent table is deleted, and this row's primary key value is referenced by one or more foreign key values in a child table, the rows in the child table that reference the primary key value are also deleted from the child table that reference the primary key value are also updated from the child table.

3.3.1.2 Inventory Application

The Inventory application is divided into four modules. They are inventory, inventory parts, inventory department, inventory directory information module, vendor's information module, personal information module, and personal information module.

3.3.1.2.1 Inventory information Module

The purpose of inventory, inventory parts, inventory department, inventory directory information module is to manage the inventory in FCSIT. The functionality of this module could be divided into the administrator and user functions.

For administrator, this module allows to search, creating, editing and deleting inventory information. Meanwhile, for the users, this module allows individual to search and view the list of available inventory.

i. **Search**

The inventory information can be searched according to inventory code, inventory category, inventory location and inventory's preferred vendor

ii. **Add New Item**

This function allows the administrator to insert new records. Since the inventory code is unique for each item, the administrator should provide all the other information like description, brand name, category, total unit, and supplier code. If a new product is purchased, the administrator should give a specific inventory code for the product.

iii. **Edit Item Record**

This function allows only the administrator to update records in inventory master file table. To update a particular record, the administrator will just have to click on the field that he/she would like to change, do changes and then save the changes.

iv. **Delete Item Record**

This delete function allows the administrator to delete records in inventory master file table. In deleting process, the database will perform a relational check to ensure that the inventory code is not in use in other module. If in use, deletion is not allowed.

v. **List Inventory**

Beside, viewing the entire available inventory, this function also allows the administrator to print inventory information instantaneously. There are wide varieties of listing to choose. Among them are a list of inventory grouped by inventory code, category and supplier code.

3.3.1.2.2 Vendors Information Module

The vendor plays an important role in any merchandising form. Vendor's information module is where the system displays the product information from the database. This module is only found in administrator section. This module allows the administrator to search, creating, editing, deleting and saving the vendor information.

i. **Search**

This function allows the administrator to search the vendors by criteria that he/she can choose. He/she scale down the search by selecting the criteria:

- Vendor ID
- Vendor Name
- Date Record Created
- Last Logon Date
- Last Update Date
- Vendor shipping address
- Vendor mailing address

ii. **Add New Vendor**

The Vendor ID is unique for each record. The administrator should provide all the other information like vendor name, contact person, vendor address, contact number and etc.

iii. Edit Vendor Record

This function allows only the administrator to make changes in the vendor database table content. To edit, the user just click on the record that he/she would like to change and the save the changes. All the fields can be updated except the Vendor ID.

iv. Delete Vendor Record

Before any record can be deleted, the database will perform a relation check to ensure that the Vendor ID is not in use in other module. If in use, deletion is not allowed.

3.3.1.2.3 Personal Information Module

This module will display the current logon person most updated information from the database record and the person maybe an administrator. He/ she may make changes on all the fields except his/ her login ID and both the last logon and last update date. It is require authentication and authorization in order to protect the system and inventory information stored from unauthorized users. Only staffs of inventory department like administrator have access the information to inventory application. Authorized users are allows to changes their password.

3.3.2 Non-Functional Requirement

Following are some of the non-functional requirement of the system:

3.3.2.1 Reliability

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 in performing its daily functions and operations. For example, whenever a button is clicked, the system should be able to perform some functionality or generate some message to inform the user what is happening.

3.3.2.2 Scalability

The scalability is to promise the capability of the system to migrate as a client or server to machines of greater or lesser 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 by additional web servers.

3.3.2.3 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 Inventory Management System and within itself.

3.3.2.4 Security

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

3.3.2.5 Data and Services Backup

The system should be able to restore to its normal operation from any potential disaster. There should be a second backup for data and services to ensure the continuous of operation.

3.3.2.6 Inter-Operability

The system has the capability to work with different types of applications to share data and process

3.3.2.7 On-Time

The system should be developed within the given time frame. In this period of time, all the requirement and also testing should be completed.

3.3.2.8 Manageability

The modules within the system should be easy to manage. This will make the maintenance and enhancement works simpler and not times consuming.

3.3.2.9 Flexibility

The system should have the capability to take advantage of new technologies and resources. The system should be able to implemented in the changing environment.

3.4 Development Requirements

3.4.1 Hardware Requirements

Development of this system, requires a machine with

i. at least 64 MB RAM

ii. at least a Pentium 200 MHz processor

iii. CD-ROM

iv. Other standard peripherals that include mouse, keyboard and monitor.

3.4.2 Software Requirements

Development of the system requires

i. Windows NT Server 4.0 as the network operating system

ii. Microsoft SQL Server 7.0 as the database server

iii. Microsoft Access 2000 as the prototype testing

iv. Microsoft Outlook 2000 as the mail application

v. Microsoft Visual Basic 6.0 to create Inventory application

vi. Crystal Report 4.6 to generate report in Inventory application

vi. ForeHelp 5.0 to built help in Inventory application

3.5 Run Time Requirements

3.5.1 Hardware Requirements for Server

The server should have

i. at least 128 MB RAM

ii. at least a Pentium II 266Mhz processor

iii. is connected to the local area network

iv. other standard computer peripherals that include mouse, keyboard and monitor

3.5.2 Software Requirements for Server

The server needs the following

i. Windows NT Server 4.0 as the network operating system

ii. Microsoft SQL Server 7.0 which is a database server that stores the system's data and information

3.5.3 Hardware Requirements for Client

The client machine is recommended to have

v. at least 32 MB RAM

vi. at least a Pentium 166Mhz processor

vii. is connected to the local area network

3.5.4 Software Requirements for Client

The client needs the following

i. Windows 95 or Windows 98 Server operating system

ii. Microsoft Outlook 2000

Chapter 4: METHODOLOGY AND SYSTEM DESIGN

4.1 Methodology

Process model is very important during the development of software. It can form a common understanding, of the activities, resources and constraints involved in software development. When a process model is created, it can help us to find the inconsistencies, redundancies and omissions in the process. There are many types of process models like Waterfall model, V model, Prototyping model, Transformational model, spiral model and others.

The process model for this IMS is a mixture of the Waterfall, Prototyping and Incremental development models (Figure 4.1). It is actually using Waterfall model concept but the prototyping and incremental concept is added during the modules development.

The Waterfall model presents a very high-level view of what happening during development and it suggests to us the sequence of events, which will encounter. It helps us to plan what we need to do [23]. The overall stage of the process model will be discussed one by one as below:

Concept Definition

The concept, purpose and functionality of the **Inventory Management System** are identified and defined.

Requirement Specification

This stage will involve the collection of information from a feature of the system or a description of something that the system is capable to do in order to fulfill the system purpose. In this case, I get most of the requirements from some of the non-academic staff in the FCSIT and also some ideas from my supervisor. After collecting the information, it has to be rewritten so that it can be transformed into a good system design.

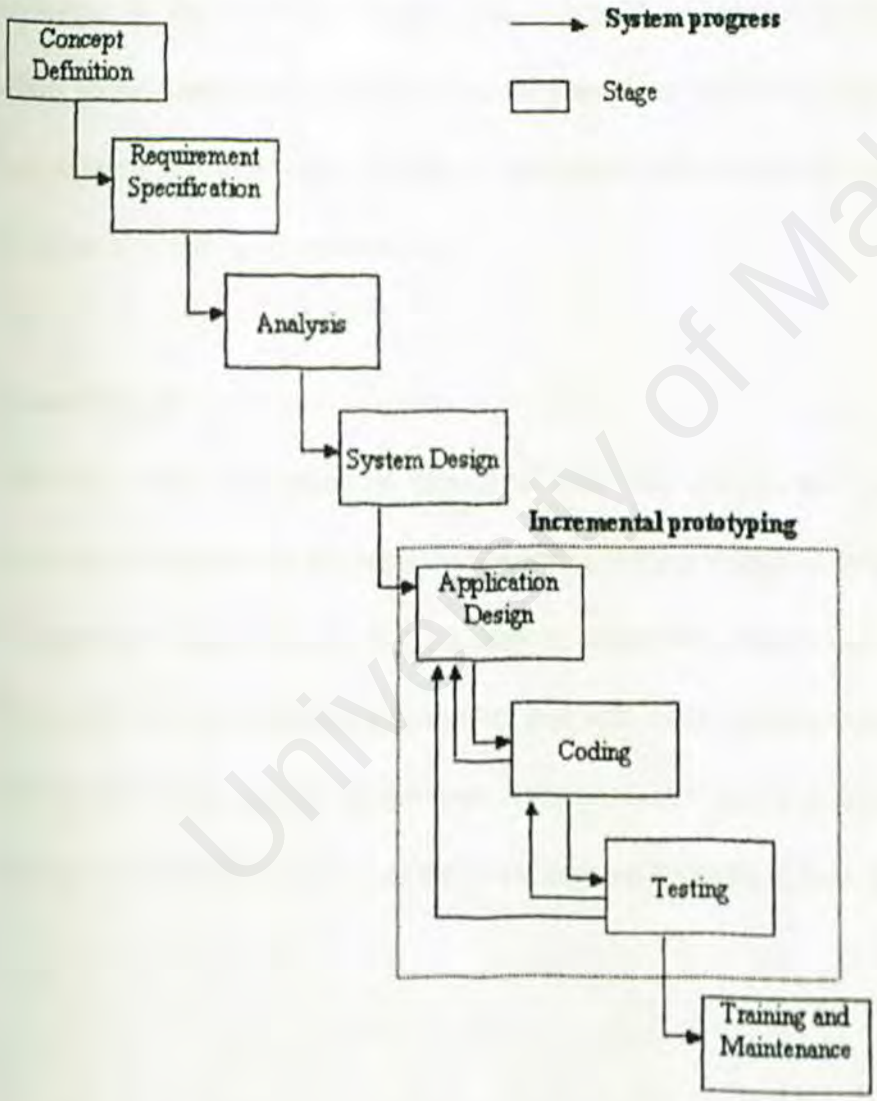


Figure 4.1 Inventory Management System Development Model

System Analysis

During this stage, system analysis for IMS was attempts to understand how the existing system helps to solve the problem identified in systems investigation and answer the question, "What must the system do to solve the problem?" The process involves understanding the broader aspects of the system that would be required to solve the problem and limitations of the existing system, as identified in system investigation. The overall emphasis of analysis was to gather data on the existing system and the requirements for the IMS system, and to consider alternative solution to the problem within these constraints and the feasibility of these solutions. The primary deliverable from system analysis was a listing of systems requirements and priorities. After finish this stage it comes to system design.

System Design

Under this stage, describes in details of how this system will meet the requirements identified during system analysis. In general, a system design is formulated to include the incorporates system features that are easy to understand, deters user error or carelessness, prevent failures or improper procedures that will cause system failure and functions in a manner that seems natural to the user. Its also an alternative solution will be set. All the application or modules design at the next stage will strictly follow this solution.

Application or Modules Design

For this project, the incremental prototyping method will be used during this stage and the following stage. Under this stage, the prototype will be created based on each module that is specified in the project. Then, it is followed by the coding stage.

Coding

Under this stage, a variety of tools like Microsoft Visual Basic 6.0, Crystal Report 4.5, Microsoft SQL 7.0 and Microsoft Access 97 will be used to write the program base on the prototype. If there is any problem, it will refer back to the application and modules design.

Testing

After coding the program components, testing is needed to examine the code to spot faults and eliminate it. It is necessary to ensure that the application program written in isolation work properly when they are integrated into the total system. Any problem faced on this stage will refer to the coding stage or the application or modules design.

There are two techniques applied in testing this system: white box testing and black box testing. White box testing is a test case design method that uses the control structure of the procedural design to derive test cases. Whereas block box testing focus on the functional requirements for a program. Block box testing is not an alternative to white box techniques; rather it is a complementary approach that is likely to uncover a different class of errors.

Training and Maintenance

This will be the last stage. The purpose of this stage is to help user to understand and feel comfortable of the new Inventory Management system. This stage will give guidance to the user about the usage of the new system and how they can use it effectively.

4.1.1 Waterfall Model

People and projects following an engineered approach to software development generally pass through a series of phases, or stage. Royce [26] was the first to coin the phrase "the waterfall model" to characterize the series of software engineering stages. Figure 4.2 shows the origina~~l mo~~del that Royce presented.

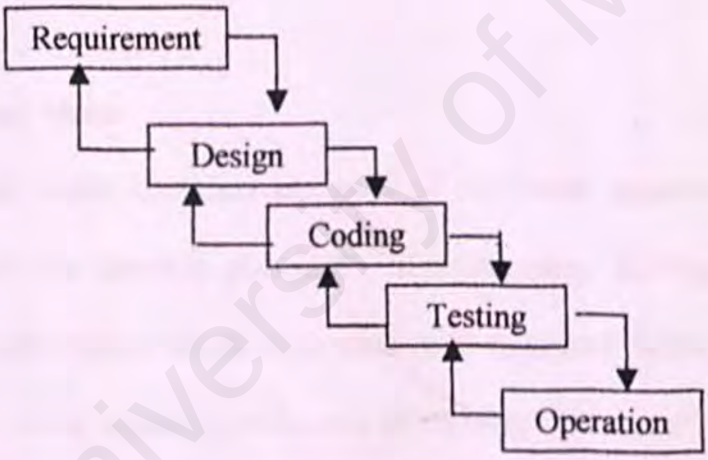


Figure 4.2: Standard Waterfall Life Cycle Model

The stages are depicted as cascading from one to another. One development stage will be completed before going down to the next stage. The Waterfall Model presents a very high level view of what is going on during the development and if suggests to developers the sequence of events they expect to counter.

The waterfall model can be very useful in helping developers lay out what they need to do. Its simplicity makes it easy to explain to customers who are not familiar with software development. It makes explicit which intermediate products are necessary in order to begin the next stage of development [23].

Linear cycle phases of waterfall model cycle are chosen to encourage top-down problem solving. Designers must first define the problem to be solved and then use an ordered set of steps to reach a solution. The linear cycle gives the project direction, and provides guidance on what should be done as the project proceeds. It is integrated with the management process through reports on project status and keeping track of resource needs [27].

4.1.2 Incremental Model

The incremental model combines elements of the linear sequential model (applied repetitively) with the iterative philosophy of prototyping. As Figure 4.3 shows, the incremental model applies linear sequences in a staggered fashion as calendar time progresses. Each linear sequence produces a deliverable "increment" of the software [32]. For example, word-processing software developed using the incremental paradigm might deliver basic file management, editing, and document production functions in the first increment; more sophisticated editing and document production capabilities in the second increment; spelling and grammar checking in the third increment; and advanced page layout capability in the fourth increment. It should be noted that the process flow for any increment could incorporate the prototyping paradigm.

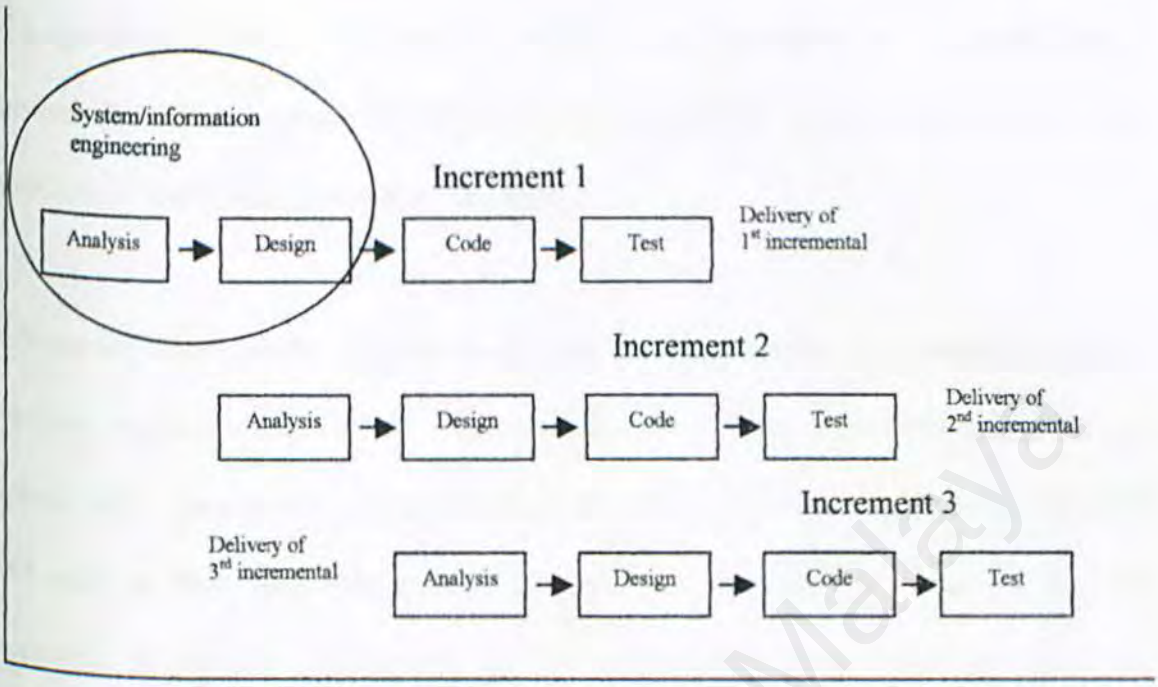

System/information engineering
Increment 1
Analysis
Design
Code
Test
Delivery of 1st incremental
Increment 2
Analysis
Design
Code
Test
Delivery of 2nd incremental
Increment 3
Delivery of 3rd incremental
Analysis
Design
Code
Test

Figure 4.3 The Incremental Model

When an incremental model is used, the first increment is often a core product. That is, basic requirements are addressed, but many supplementary features (some known, others unknown) remain undelivered. The core product is used by the customer (or undergoes detailed review). As a result of use and/or evaluation, a plan is developed for the next increment. The plan addresses the modification of the core product to better meet the needs of the customer and the delivery of additional features and functionality. This process is repeated following the delivery of each increment, until the complete product is produced.

The incremental process model, like prototyping and other evolutionary approaches, is iterative in nature. But unlike prototyping, the incremental model focuses on the delivery of a operational product with each increment. Early increments are "stripped down" versions of the final product, but they do provide capability that serves the user and also provide a platform for evaluation by the user.

Incremental development is particularly useful when staffing is unavailable for a complete implementation by the business deadline that has been established for the project. Early increments can be implemented with fewer people. If the core product is well received, then additional staff (if required) can be added to implement the next increment. In addition, increments can be planned to manage technical risks. For example, a major system might require the availability of new hardware that is under development and whose delivery date is uncertain. It might be possible to plan early increments in a way that avoids the use of this hardware, thereby enabling partial functionality to be delivered to end users without inordinate delay.

4.1.3 Prototype Model

The development strategy used in IMS is the software prototyping methodology. This methodology was chose because there was a high degree of uncertainty in several areas in the system requirements. The emphasis is on trying out ideas and providing assumptions about the requirements, not on system completeness.

Prototyping is a process that enables the developer to create a model of the software that must be built. This prototyping model consists of six steps as shown in figure 4.4.

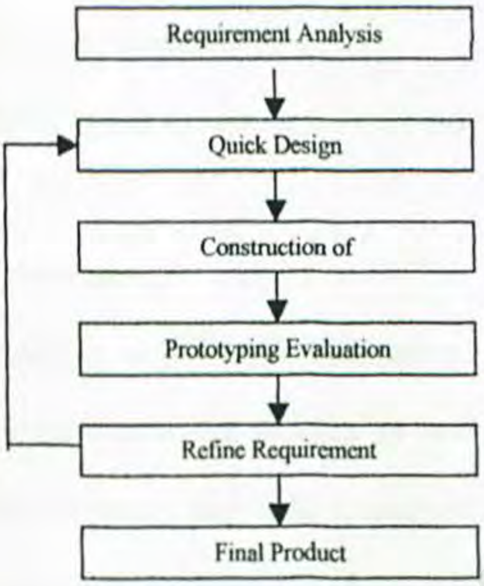


Figure 4.4 The Prototyping Model

Like all approaches in software development, prototyping begins with requirements gathering. After identifying the known requirements, a quick design is the formulate. The quick design focuses on the top-level architecture and data design issues, rather than on detail procedural design. Later the quick design leads to the construction of a prototype. The prototype is tested and evaluated to refine the requirements and maybe additional requirements. A process of iteration occurs until all requirements are formalized or until the prototype has evolved into a production system [25].

The prototype model is used because

i) Change can be made early in development.

ii) System developed can meet users' needs more easily.

iii) It provides a common baseline and frame. Developers and users can communicate better.

iv) The developer can understand the system much better.

Advantages of Prototyping

i) Changing the early in its development

Successful prototyping depends on early and frequent user feedback to help modify the system and make it more responsive to actual needs. As with any systems effort, early changes are less expensive than changes made late in the project's development.

ii) Scrapping undesirable systems

A second advantage of using prototyping is the possibility of scrapping a system that is just mot what users and analysis had hoped. Once again, the issue of time and money spent arises. A prototype represents much less of an investment than a completely developed system.

iii) Designing a system for users' needs and expectations

A third advantage of prototyping is that the system being developed should be a better fit with users' needs and expectations. Many studies of failed information systems indict the long interval between requirements determination and the presentation of the finished systems while sequestered away from users during this critical period.

Disadvantage of Prototyping

i) Managing the project

Although several iterations of the prototype may be necessary, extending the prototype indefinitely also creates problems. It is important that the systems analysis team devises and then carries out a plan regarding how feedback on the prototype will be collected, analyzed, and interpreted. Set up specific time periods during which you and management decision makers will use feedback to evaluate how well the prototype is performing.

ii) Adopting an incomplete system as complete

A second major disadvantage of prototyping is that if a system is needed badly and welcomed readily, the prototype may be accepted in its unfinished state, and pressed into service without necessary refinement. While superficially this may seem an appealing way to short circuit a lot of development effort, it works to the business and team's disadvantage.

4.2 Program Hierarchy Chart

IMS is constructed by a collection of independent but interacting modules. The following figures show those modules in hierarchy charts.

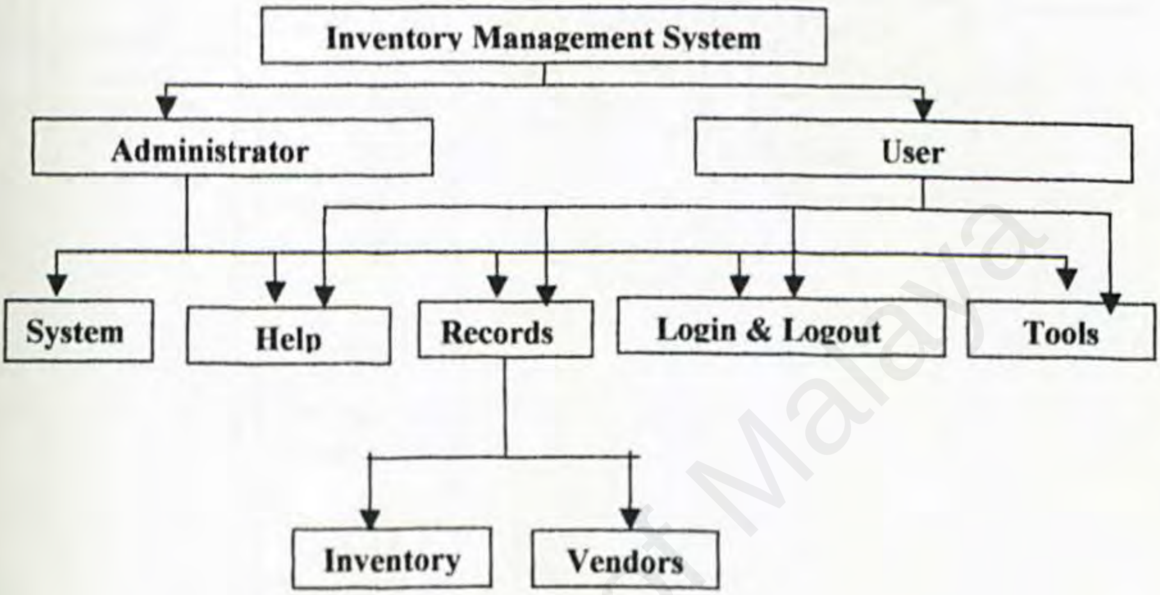


Figure 4.5 IMS hierarchy chart

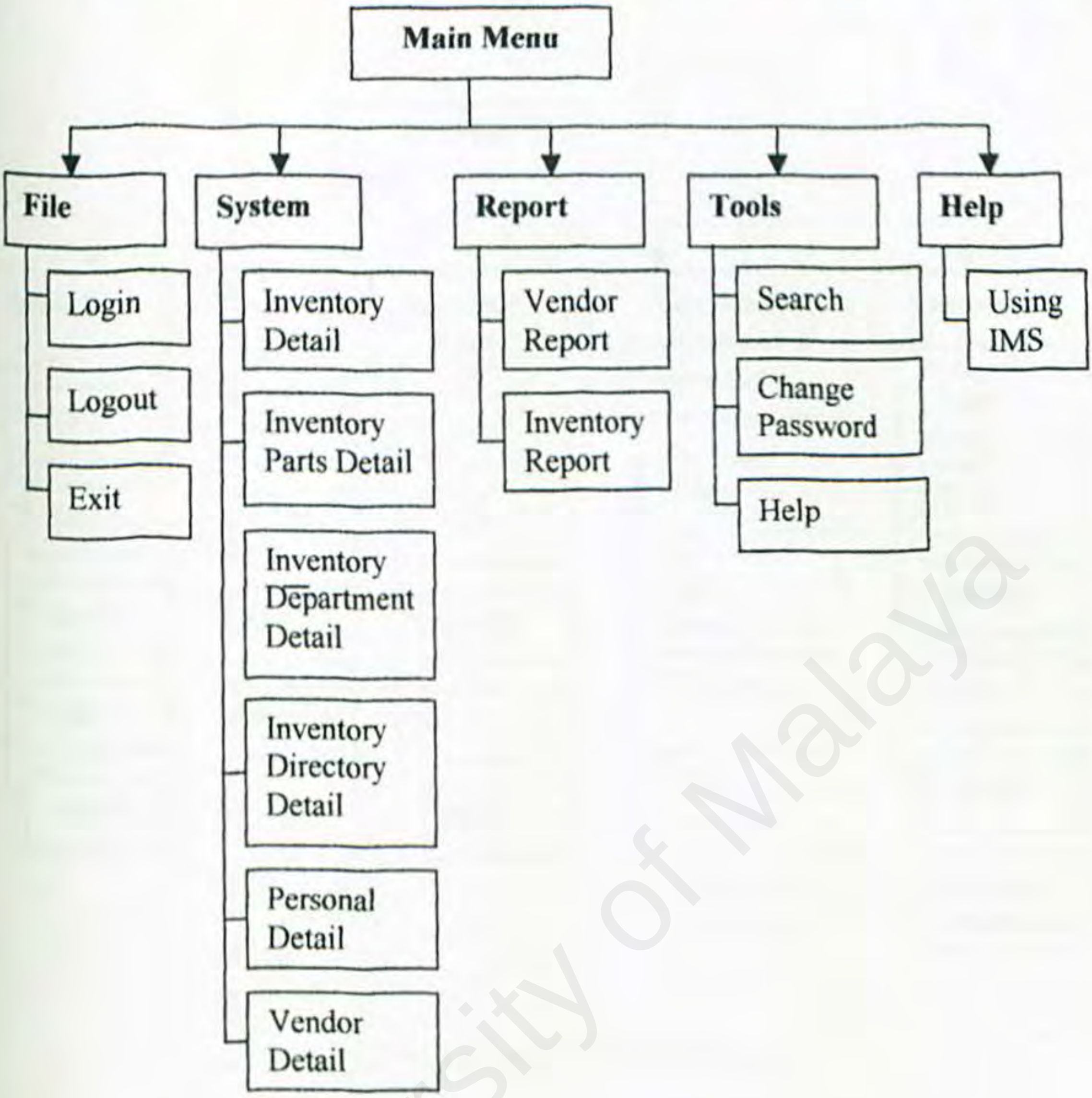


Figure 4.6 Main menu module hierarchy chart

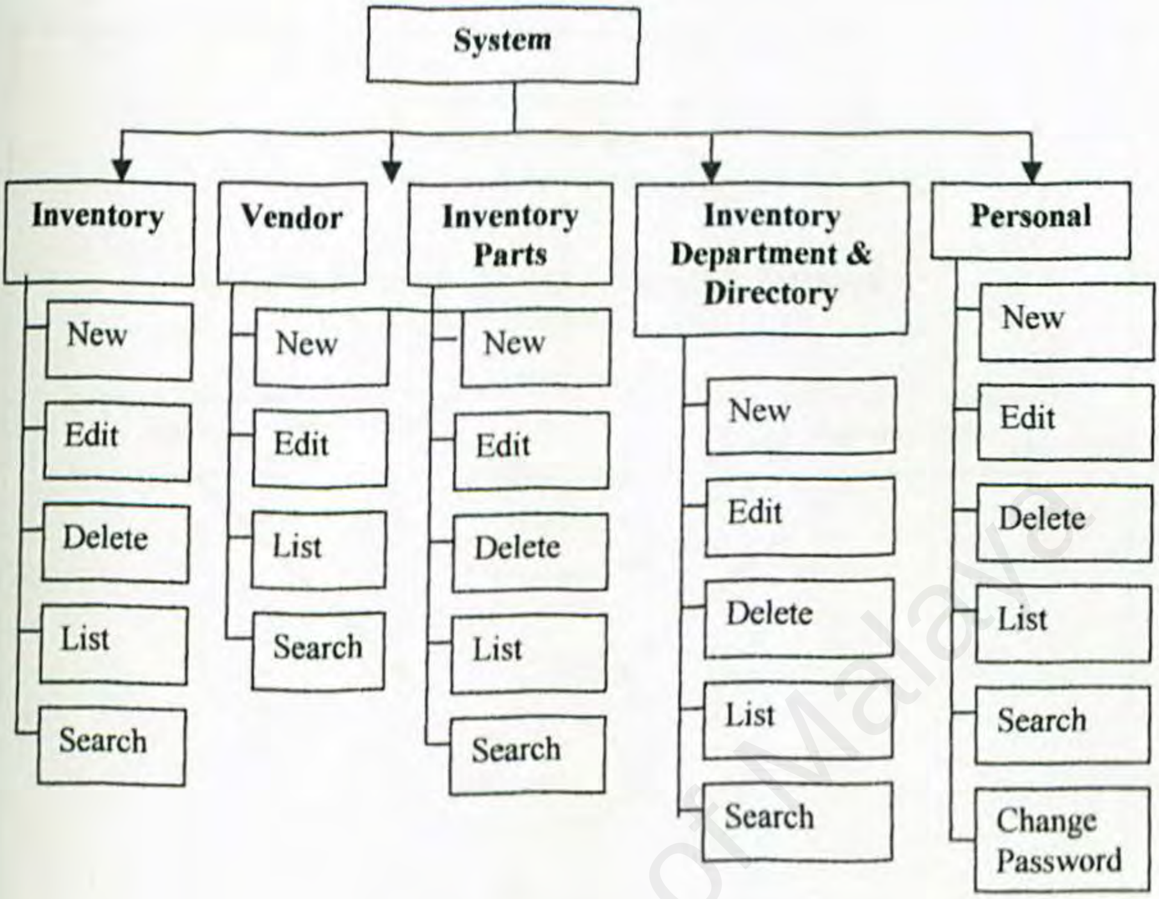


Figure 4.7 System module hierarchy chart

4.3 Data Flow Diagram

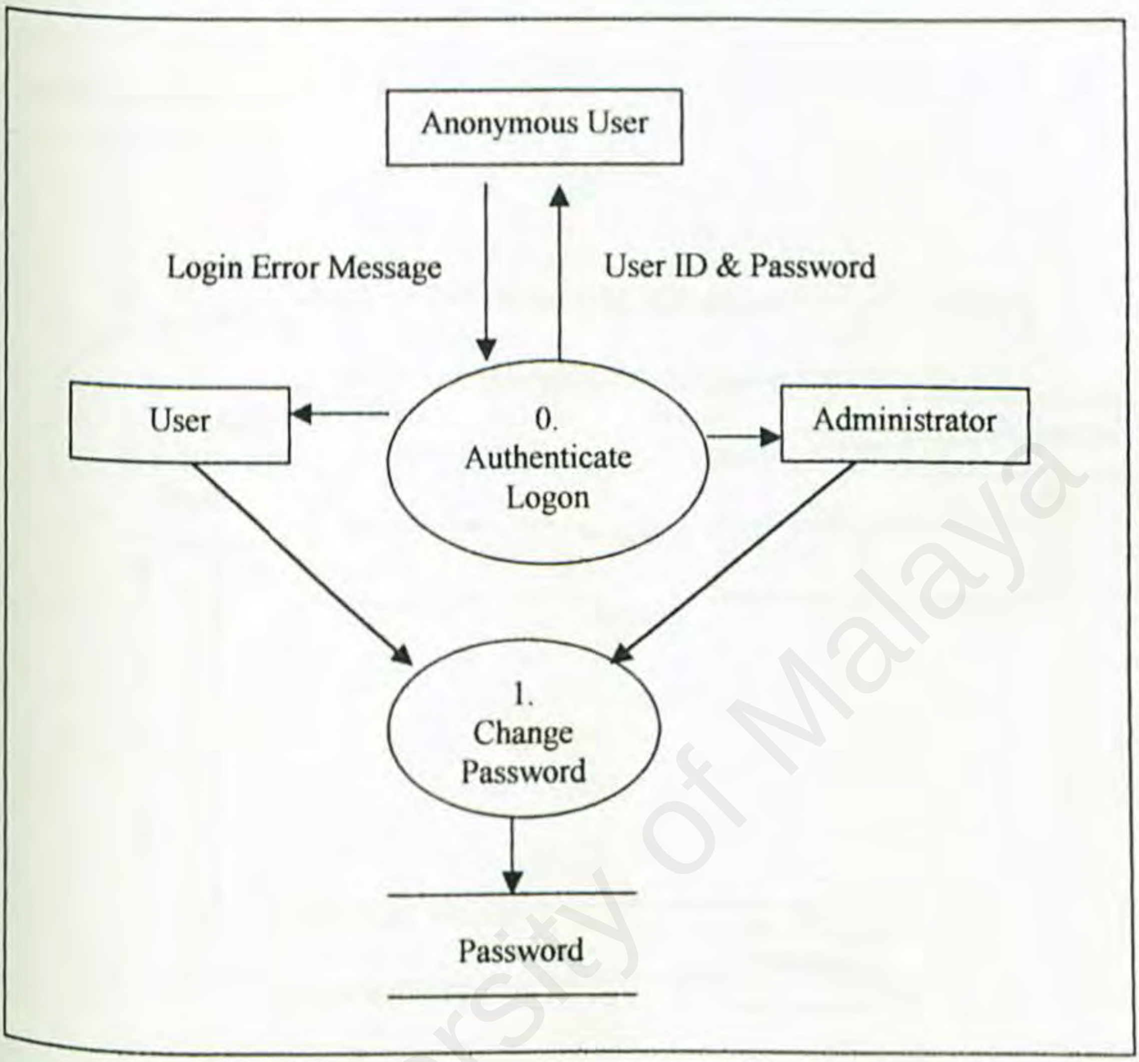


Figure 4.8 Context Data Flow Diagram for Logon

From the data flow diagram, we know that the process of authenticate logon will need the user login ID and password. The authenticate logon process will verify the user login ID and password and detect the user type whether the user is super admin or other user. For the administrator, the data will flow through the process of password changes.

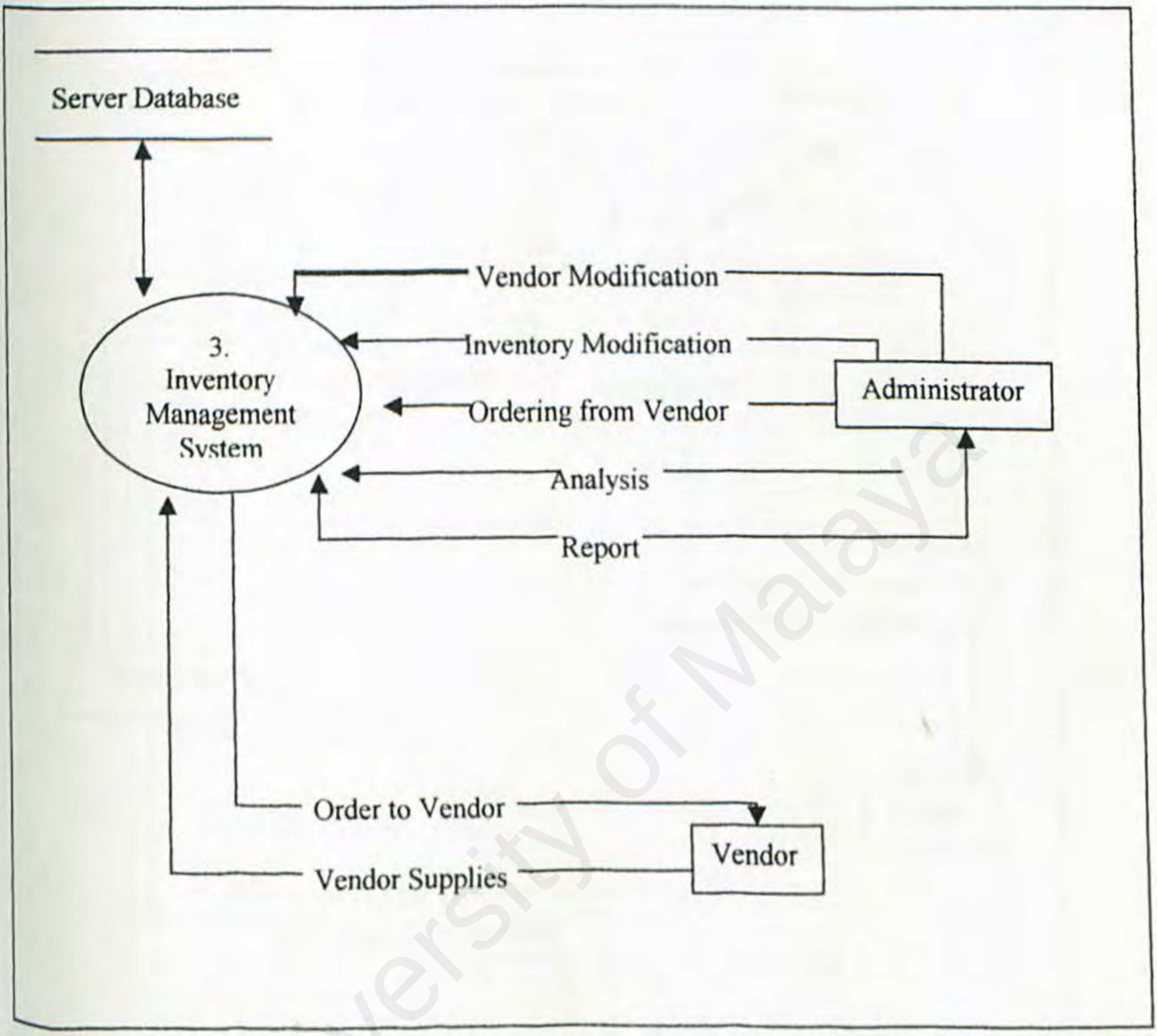


Figure 4.9 Context data flow diagram for inventory management system

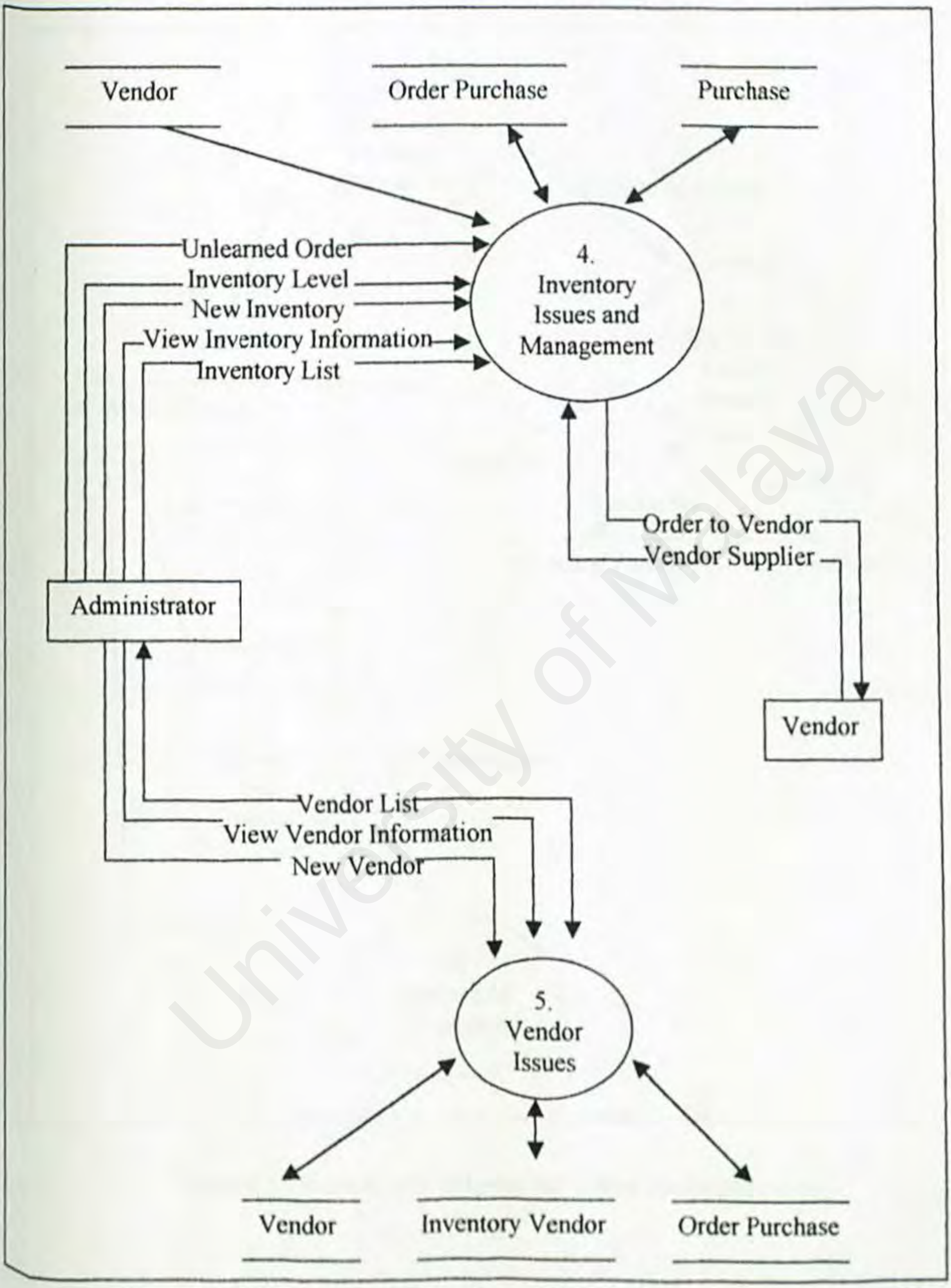


Figure 4.10 First-level data flow diagram for inventory management

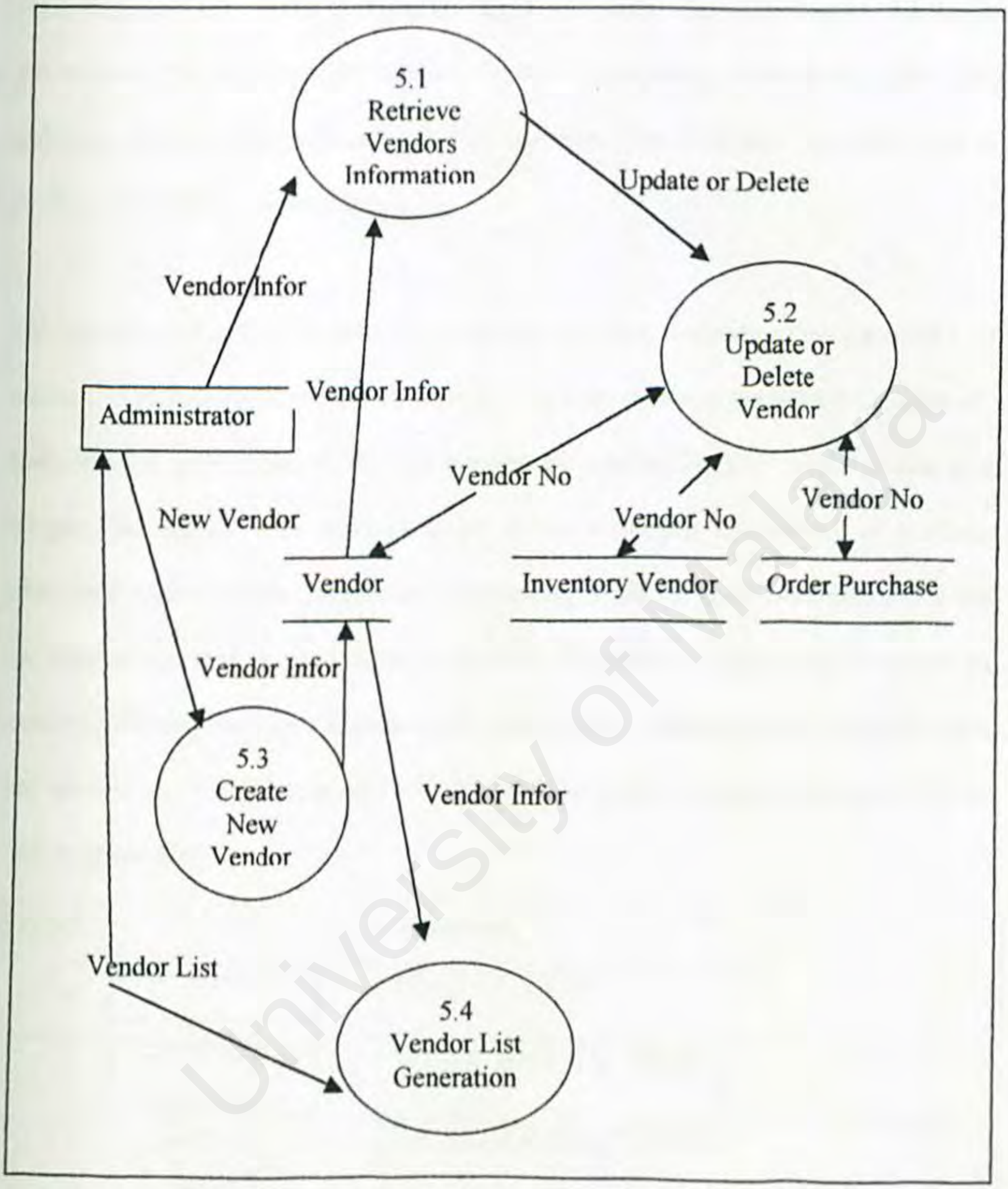


Figure 4.11 Second-level diagram for vendor issues processing

4.4 Enhanced-Entity-Relationship model

Entity-Relation (ER) model is a popular high-level conceptual data model. This model and its variations are frequently used for the conceptual design of database application, and many database design tools employ its concepts. The ER model describes data as entities, relationships, and attributes.

The Enhanced-ER (EER) model includes all the modeling concepts of the ER model. In addition,, it includes the concepts of subclass and super class and the related concepts of Specialize and generalization. Another concept included in the EER model is that of a category. Associated with these concepts is the important mechanism of attribute inheritance. Unfortunately, no standard terminology exists for these concepts, so we use the terminology that is used most commonly. Alternative terminology is given in footnotes. We also describe a diagrammatic technique for displaying these concepts when they arise in an EER schema. We call the resulting schema diagrams enhanced-ER or EER diagrams [24].

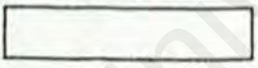
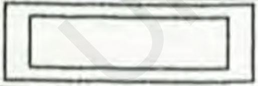
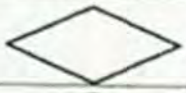
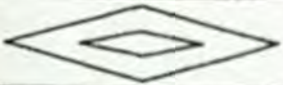
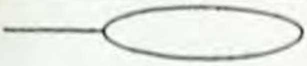
Symbol	Meaning
	ENTITY TYPE
	WEAK ENTITY TYPE
	RELATIONSHIP TYPE
	IDENTIFYING RELATIONSHIP TYPE
	ATTRIBUTE

Figure 4.12 Summary of ER diagram notation

Below show the EER diagrams for Inventory Management System.

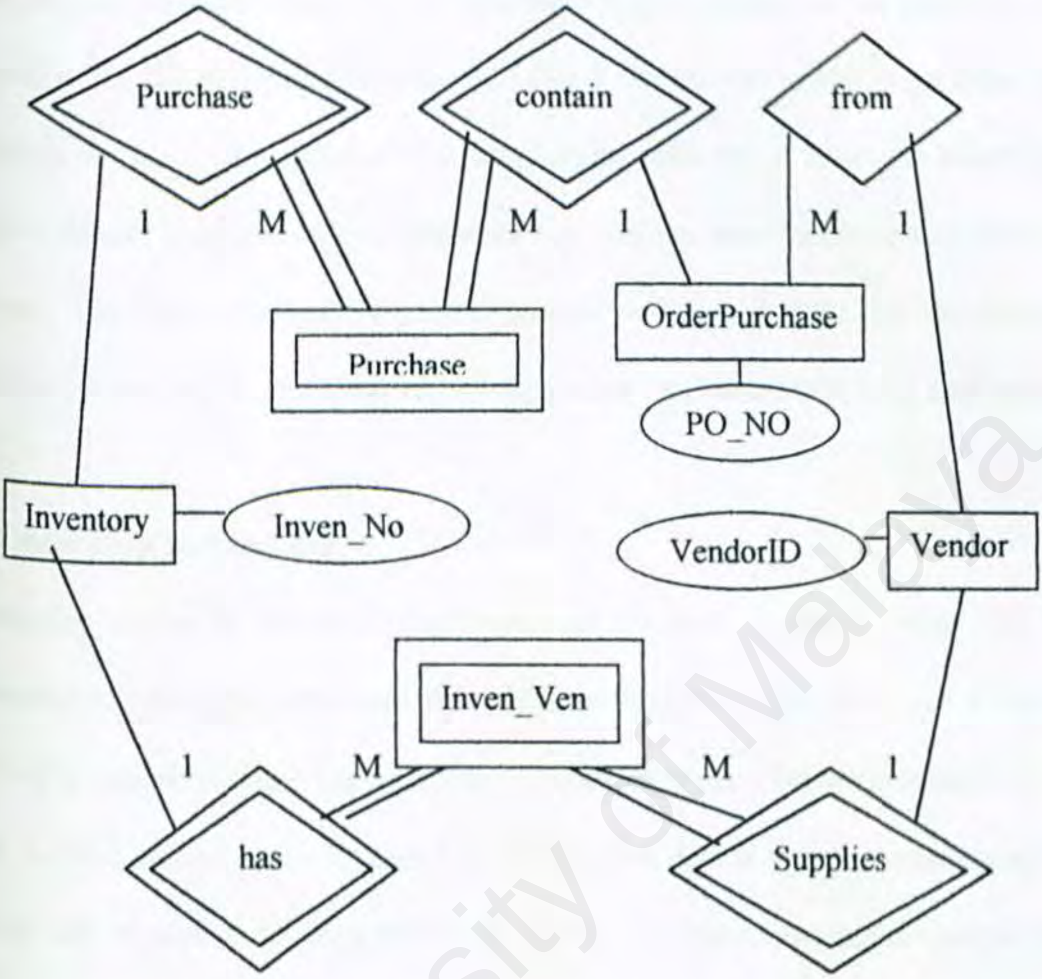


Figure 4.13 Enhanced entity relationship diagram for inventory management system

4.5 Database Design

Databases and database technology are having a major impact on the growing use of computers. It is fair to say that database will play a critical role in almost all areas where computers are used, including business, engineering, and etc. A Database Management System (DBMS) is a collection of programs that enables users to create and maintain a database. The DBMS is hence a general-purpose software system that facilitates the processes of defining, constructing, and manipulating databases for various applications.

4.5.1Controlling Redundancy

Redundancy occurs in database when much of the data is stored twice [28]. This redundancy in storing the same data multiple times leads to several problems. First, there is a need to perform a single logical update in multiple times. This leads to duplication of effort. Second, storage space is wasted when the same data is stored repeatedly, and this problem may be serious for large databases. Third, files that represent the same data may become inconsistent. This may happen because an update is applied to some of the files but not to others. For consistency, we should have a database design that stores each logical data item in only one place in the database. This does not permit any inconsistency, and it saves storage space.

4.5.2 Normalization

The Normalization process, as first processed by Codd in 1972, takes a relation schema through a series of test to "certify" whether or not it belongs to a certain normal form. [29]. Normalization of data can be looked on as a process during which unsatisfactory relation schemas are decomposed by breaking up their attributes into smaller relation schemas that process desirable properties. [28]. One objective of the original normalization process is to ensure that the update anomalies do not occur. The original normalization process also minimizes the storage space that using in database and prevents redundancy.

4.5.3 Database Security

Security has always been a major concern when implementing information technology. However, the level of security that should be included in an information system involves some judgment about the dangers associated with the system and the resource implications of various means of avoiding or minimizing those dangers. Current thinking in information system security is that the issues center on confidentiality, integrity, and availability [30].

The information system is confidentiality when the information is only disclosed to those users who are authorized to have access to it. In the integrity information system, the information is modified only by those users who have the right to do so. When authorized users can access the information resources when needed, the system is availability.

Database security is concerned with the ability of the system to enforce a security policy governing the disclosure, modification or destruction of information. Humans typically use a database as a technical tool for storing, processing and communicating information. The database relays the messages by persistently memorizing the corresponding data following the three phase procedure: 'accept message = = > store / process data = = >

assemble, duplicate and communicate data on demand [31]. The quality of mediation is dependably assured by special protocols enforcing completion of transactions and integrity constraints on stored data. Mediation is shared among many users and is required to be efficient in time and space. We can classify the requirements for database security into four basic categories [31].

1. *The authorization policies*, which govern the disclosure and modification of information in the database (access control). We can identify three major types of authorization policies that include (as see below) the mandatory policies, the discretionary policies, and the personal knowledge approach.
2. *The data consistency policies*, which govern the consistency and correctness of information in the database. Three basic issues involved in this area are the integrity constraints, the database recovery policies, and the concurrency policies.
3. *The availability policies*, which govern the availability of the information in the database.
4. *The identification, authentication and audit policies*, which govern the identification and authentication procedures to be followed and the auditing policies for the database.

4.5.3.1 A Secure Database Development Methodology

The problem of developing a secure database system consists of three-issue [31].

1. The definition of the semantics of the secure database to be developed, that is to characterize the needed security properties in terms of the database semantics.

2. The implementation of those semantics on a database system that is on a DBMS and on the data it handles.
3. Assuring that the implemented system provides the needed security properties.

A development methodology has the purpose of specifying how each one of these three secure database development issues can be achieved. This is usually accomplished by guiding the various steps of the development, by providing modeling and analysis tools, and by organizing these three issues into a global framework allowing them to achieve consistency of the whole development process and of the target system.

The five possible steps of such a methodology for secure database development are outlined below [31].

1. Preliminary Analysis

   The main objective of the preliminary analysis is to determine the scope and feasibility of the target system. This phase corresponds to the feasibility study done, for example, for information system development, where the cost effectiveness of the future system and of the project itself is estimated, in order to decide about the necessary applications and to set priorities among them. In this preliminary analysis phase the following issues are studied:

(a) The threats of major concern for the organization to which the database system is directed.

(b) The characteristics of the environment of the target database system

(c) The applicability of the software and hardware products that are available in the security market.

(d) The cost/benefit ratio of the target system in terms of security.

2. Requirements Analysis

In the requirement analysis phase, the database designer determines in details that the database users are and their needs are. Normally, this includes identifying what information is of interest for each user role, the characteristic of the information, and how the information relates to other pieces of information. Automated tools already available could help the database designer gather, store, and crosscheck such user requirement information.

Conceptual Design

In the conceptual design phase, a formal specification of the database security policies is given through a security conceptual model. This model shows the system subjects, objects, operations and permitted modes of access by subjects to objects, according to the database security policies, etc. The conceptual model should also ıbstract the physical details of the target database system and be independent of the lata model supported by the DBMS used. The purpose is to have a formal pecification of the secure database system semantics, which can be tested for ırrectness, completeness and ambiguity.

4. Logical Design

In the logical design phase, the conceptual model is mapped into the data model supported by the specific DBMS (e.g. relational). The necessary data schemas for the specific DBMS used are thus designed, keeping the security requirements into account. For example, the number and contents of the external schemas are stated, based on the view that each applications is allowed to have on the data, Issues such as schema normalization are also considered in this step of the methodology.

5. Physical Design

Based on the logical representation of data obtained as a result of the logical design step, in the physical design phase the developer:

(a) Decides on the access request processing strategies. This includes, the physics structured for security data, the methods of accessing this structure in order to make the authorization decision, the result of such a decision (Which can simply be a 'Yes/No' answer or a more sophisticated system action), and so on.

(b) Keeps into account performance and overloading parameters for the future secure database system. These are quantitative parameters (as for example response times, storage costs, update costs, data organization costs, etc.) as well as qualitative parameters (as for example, flexibility, adaptability, convertibility to new environments, recovery/restart capability, etc.)

(c) Selects the appropriate set of security mechanisms capable to implement the given requirements and policies, which have been based on the degree of trust required by the system and on the expected performance.

4.5.4 Data Dictionary

The tables in Inventory database are as follow:

1. Inventory Table

This table stores all the inventory records. Listed below in Table 4.1 are all attributes for this table.

Field Names	Data Type	Size	Description
KOD	Text	50	Item Code
TAG	Text	50	Tag
PARTS	Number (Double)	10	Parts
NAMA	Text	50	Name
JENAMA	Text	50	Brand
PO_NUM	Text	50	Purchase Order Number
PO_DATE	Date/Time	50	Purchase Order Date
DEPT	Text	50	Department
BILIK	Text	50	Room Number
PEMBEKAL	Text	50	Vendor
NO_SIRI	Text	50	Serial Number
HARGA	Number (Double)	10	Price
TARAF	Text	50	Status
BIAYAI	Text	50	Finance
CATITAN	Text	50	Records
ENTRY_DTE	Date/Time	8	Entry Date
GUNASAMA	Text	50	Sharing

Table 4.1 Inventory Table

2. Inventory Parts Table

This table stores all the parts or inventory records. Listed below in Table 4.2 are all attributes for this table.

Field Names	Data Type	Size	Description
KOD	Text	50	Item Code
TAG	Text	50	Tag
SUBTAG	Text	50	Sub Tag
NAMA	Text	50	Name
JENAMA	Text	50	Brand
DEPT	Text	50	Department
BILIK	Text	50	Room Number
NO_SIRI	Text	50	Serial Number
TARAF	Text	50	Status
CATITAN	Text	50	Records
ENTRY_DTE	Date/Time	8	Entry Date

Table 4.2 Inventory Parts Table

3. Inventory Department Table

This table stores all the department of the inventory records. Listed below in Table 4.3 are all attributes for this table.

Field Names	Data Type	Size	Description
DEPTCODE	Text	50	Department Code
FACULTY	Text	50	Faculty
DEPTNAME	Text	50	Department Name

Table 4.3 Inventory Department Table

4. Sharing Table

This table stores all sharing of the inventory records. Listed below in Table 4.4 are all attributes for this table.

Field Names	Data Type	Size	Description
TAG	Text	50	Tag
PERSON1	Text	50	Person 1
PERSON2	Text	50	Person 2
MAINTAIN	Text	50	Maintain
CHARGE	Text	50	Charge
REMARKS	Text	50	Remarks

Table 4.4 Sharing Table

5. Personal Table

This table stores all the personal records. Listed below in Table 4.5 are all attributes for this table.

Field Names	Data Type	Size	Description
USER_NO	Text	50	User Number
PASSWORD	Text	50	Password
ACTUAL_NAME	Text	50	User Actual Name
STATUS	Number(Single)	10	User Status (1->Administor, 2 ->User)
EMAIL	Text	50	User Email Address

Table 4.5 Personal Table

6. Last Number Table

This table stores the last number of the inventory records. Listed below in Table 4.6 are all attributes for this table.

Field Names	Data Type	Size	Description
Running	Number (Int)	10	Last Number in Inventory Table

Table 4.6 Last Number Table

7. Inventory Directory Table

This table stores all the directory inventory records. Listed below in Table 4.7 are all attributes for this table.

Field Names	Data Type	Size	Description
ITEMCODE	Text	50	Item code
ITEMNAME	Text	50	Item name
GROUP	Text	50	Group

Table 4.7 Inventory Directory Table

8. Inventory Delete Table

This table stores all the deleting the inventory records. Listed below in Table 4.8 are all attributes for this table.

Field Names	Data Type	Size	Description
KOD	Text	50	Item code
TAG	Text	50	Tag
SUBTAG	Text	50	Sub Tag
PARTS	Number (Double)	50	Parts
NAMA	Text	50	Name
JENAMA	Text	50	Brand
PO_NUM	Text	50	Purchase Order Number
PO_DATE	Date/Time	50	Purchase Order Date
DEPT	Text	50	Department
BILIK	Text	50	Room Number
PEMBEKAL	Text	50	Vendor
NO_SIRI	Text	50	Serial Number
HARGA	Number (Double)	10	Price
TARAF	Text	50	Status
BIAYAI	Text	50	Finance
CATITAN	Text	50	Records
ENTRY_DTE	Date/Time	8	Entry Date
GUNASAMA	Text	50	Sharing
REASON	Text	50	Reason
DTE_DELETED	Date/Time	8	Date Deleted

Table 4.7 Inventory Delete Table

9. Vendor Table

This table stores all the vendor records. Listed below in Table 4.9 are all attributes for this table.

Field Names	Data Type	Size	Description
KOD	Text	50	Vendor Number
PEMBEKAL	Text	50	Vendor
ALAMAT	Memo	255	Address
TELEFON_NO	Text	50	Telephone
EXT	Text	50	Extension
FAX	Text	50	Fax Number
EMAIL	Text	50	E-mail Address
CONTACT_PERSON	Text	50	Contact Person Name
CP_TELEFON_NO	Text	50	Contact Person Telephone
REMARKS	Text	50	Remarks

Table 4.9 Vendor Table

4.6 User Interface Design

All Visual Basic forms in this application will always appear at the middle of the user's screen. This provides consistency in screen display. A user or an administrator can navigate through the application, or exit the application by using the command buttons or menu that is provided.

A menu interface provides user with an onscreen list of available selection. In responding to the menu user is limited to the option displayed. The user need not know the system nut does need to know what the task should be accomplished.

In the main page the user are required to choose whether to logon as an administrator or a user in the file module. If the user is a super administrator, he/ she will allow to use the system module, report module and the tool module. However, if the user is a lecturer, offices or other administrator, they just allow to use the tool module include calendar, calculator and search function.

This Graphic user interface for Main Page of Inventory Management System include file module, system module, report module and tool module. It also has a toolbar for easy to use the function inside the modules. Every button of the toolbar also has each tool tip.

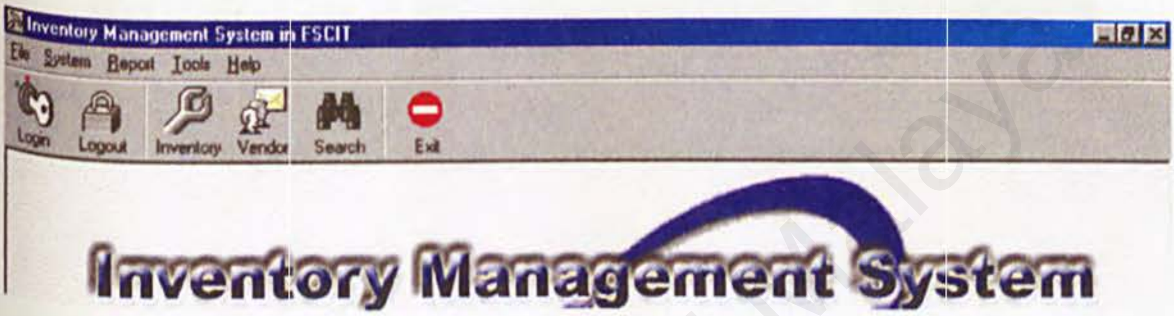

Inventory Management System in FSCIT
File System Report Tools Help
Login
Logout
Inventory
Vendor
Search
Exit
Inventory Management System

Figure 4.14 Main page for Inventory Management System

This graphic user interface show File module include login, logout and exit function.

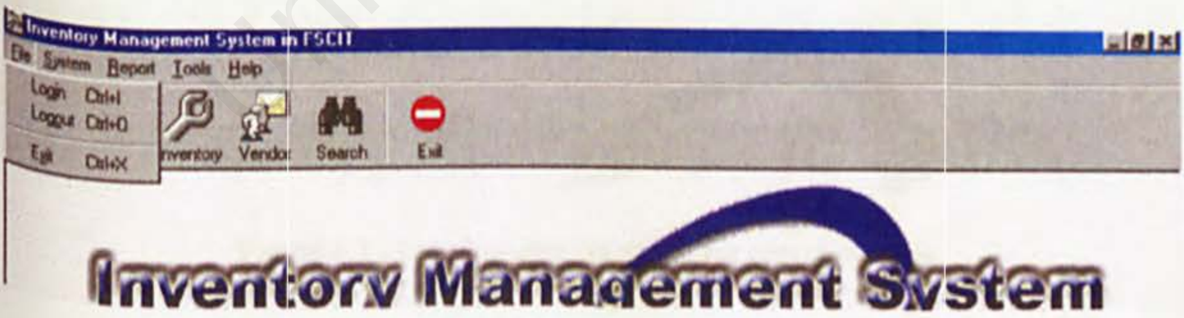

Inventory Management System in FSCIT
File System Report Tools Help
Login Ctrl+I
Logout Ctrl+O
Exit Ctrl+X
Inventory
Vendor
Search
Exit
Inventory Management System

Figure 4.15 Graphic user interface for file module

This graphic user interface show System module includes supplier detail, inventory detail and order & Purchase detail.

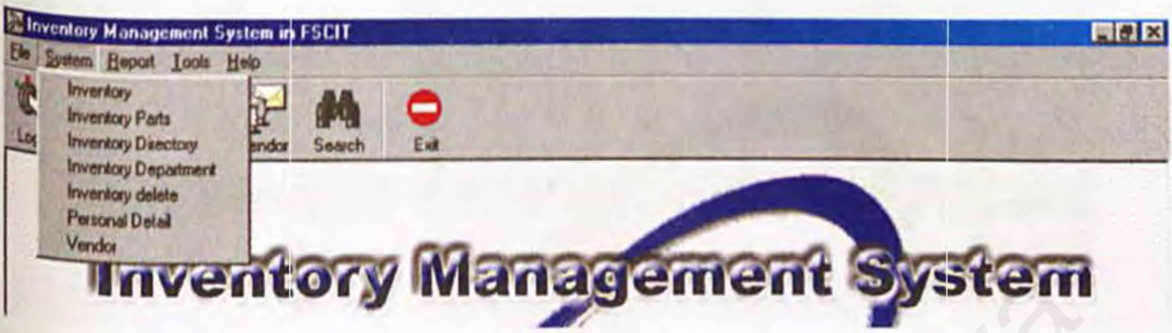


Figure 4.16 Graphic user interface for system module

This graphic user interface shows Report module includes vendor report and Inventory report. For Inventory report has report base by selected room number, room number, tag and code of the inventory. For vendor report has report base by name and code of the vendor.

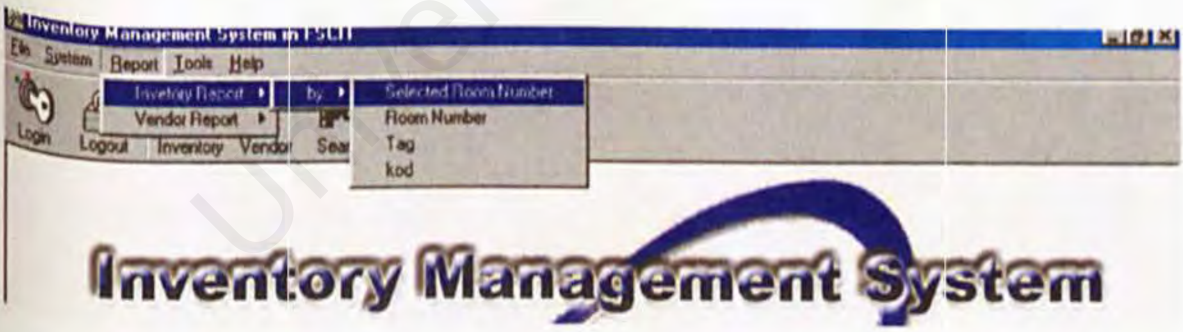


Figure 4.17 Graphic user interface for report module

This graphic user interface shows Tools module includes change password and search the item.

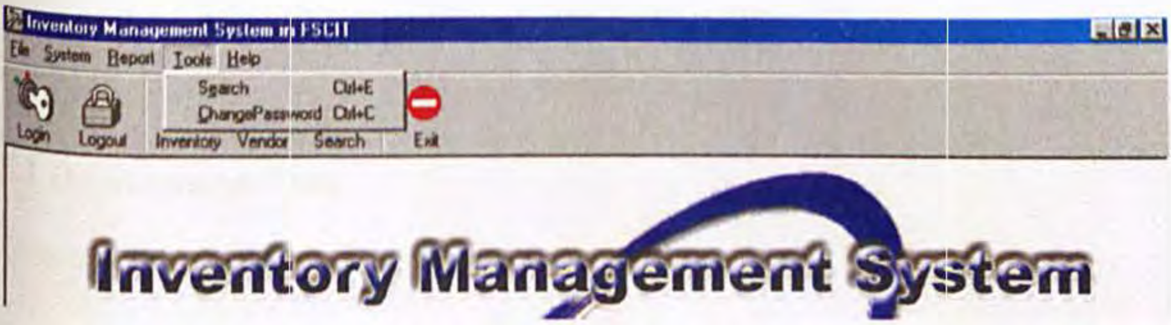


Figure 4.18 Graphic user interface for tools module

This graphic user interface shows Help module include Using Inventory Management System (IMS).

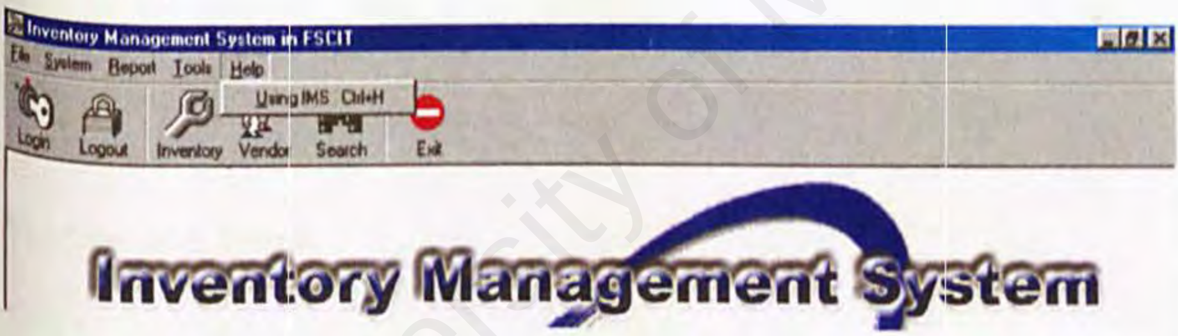


Figure 4.19 Graphic user interface for help module

Chapter 5: System Implementation

Implementation is the last step in developing a new computer. System implementation is a process that converts the system designs into program codes.

5.1 Development Tool

Tools used for development include

i. Operating System – Windows NT 4.0, Windows 98

ii. Program coding – Microsoft Visual Basic 6.0

iii. Report generation – Crystal Report 4.6

iv. Database development – Microsoft Access 2000, Microsoft SQL Server 7.0

v. Graphic creation – Paint, Adobe Photoshop 5.0

vi. Help builder – ForeHelp 5.0

5.1.1 Operating System

Table 5.1 states the operating systems used and their purposes.

Operating System	Purpose
Windows 95 or Windows 98	Initial development when using Microsoft Visual Basic 6.0
Windows NT 4.0	Needed when implementing database with Microsoft SQL Server 7.0, testing the local area network

Table 5.1 Operating systems used and their purpose

5.1.2 Program Coding Tool

Table 5.2 state program-coding tools used and their purposes.

Tool	Purpose
Standard Module in Microsoft Visual Basic 6.0	Create Inventory application

Table 5.2 Program coding tools used and their purposes

5.1.3 Database Development Tool

Table 5.3 states database development tools used and their purposes.

Tool	Purpose
Microsoft Access 97	Build database structure as a prototype
Microsoft SQL Server 7.0	Database used for inventory management system for inventory's FSKTM
Ms Query in Microsoft SQL Server 7.0	Execute SQL Statement
SQL Service Manager in Microsoft SQL Server 7.0	Set the SQL Server running
SQL Enterprise Manager in Microsoft SQL Server 7.0	View and Edit tables created in the database, and create jobs

Table 5.3 Database development tools used and their purposes

5.1.4 Report Creation Tool

Table 5.4 states tools used to create report for Inventory application

Tool	Purpose
Crystal Report 4.6	Create various report for Inventory application

Table 5.4 Report creation tool used and its purposes

5.1.5 Graphic Creation Tool

Table 5.5 states tools used to create graphic for inventory management system of Inventory.

Tool	Purpose
Paint	Create graphic and icon
Adobe Photoshop 5.0	Create graphic and icon

Table 5.5 Graphic creation tools used and their purposes

5.1.6 Help Builder

Table 5.6 states tools used to built help for Inventory application.

Tool	Purpose
ForeHelp 5.0	Built Help for Inventory application

Table 5.6 Graphic creation tools used and their purposes

5.2 Platform Development

The platform development will include setting up the Windows NT and SQL server.

5.2.1 Setting Windows NT 4.0

Before the system is being developed, it needs to run under Windows NT server. During the installation of Windows NT 4.0, the hard disk is formatted using NT File system format to ensure a more stable and secured NT transaction across the platform. Several step are involved during the installation.

- First, install Windows NT 4.0.
- Then, install NT Service Pack 3.

- After that, install the NT Option Pack 4 that include the Internet Information Server 4.0 and Microsoft Transaction Server.
- Finally, install NT Service Pack 4 followed by NT Service Pack 6.

5.2.2 Setting SQL Server

The SQL server is installed in another Compaq server. It is separated from the IIS and Exchange Server. After the Microsoft SQL server has been installed successfully, a database name INVENTORY is created. Then, we create the table according to the database design. This database will become the database storage for the system. The tables are created for keeping the data used in all modules of the Inventory Management System.

Besides creating table, we insert SQL server cursor in order to perform daily operation from the server. This operation includes sending email to a particular person when the person forget to login after 12:00 p.m. The cursor is very powerful because the SQL command can be generated from the server. Further more, we configure the SQL server so that it can perform the database backup daily.

We allocated the hard disk space for the database to maximize the performance of the SQL server and to ensure there is enough of space to store the record. The file growth of the database is set to 10% of the original database size.

In order to map the database to the web server, a system Data Source Name (DSN) was created. The system DSN named **INVENTORY** was registered through the ODBC in the web server machine. The ODBC could map to the storage even it is in the different machines. Likewise, the web server could open the database storage by just calling the system DSN.

5.3 Modules Implementation

The Inventory Management System is divided into 2 main modules, which are the user module and the administrator module. Each module is developed using Visual Basic. The following sections explain the details about the implementation of each module throughout the project.

5.3.1 User Module

There is using the stand-alone system for the Inventory Management System. The stand-alone system is developed using the visual basic. This stand-alone PC is connected to the ODBC data source (32-bit) in Control Panel. The system DSN named **INVENTORY** was registered through the ODBC in this PC in order to map the database through ODBC 32-bit.

5.3.1.1 Stand-alone System

This user module consists of several sub-modules, which are tool module; authenticate logon module and report module. The module is developed using the Visual Basic 6.0. The entire module is implemented individually before there are integrated together.

5.3.1.2 Authenticate Logon

This module is implemented to verify the login ID and password entered by the user.

5.3.1.3 Tools Inventory

This module is implemented to search the inventory record. It is using the List View control from the Visual Basic to display the inventory list. Beside that, tools also include change password function to let user and administrator to change its own password. In addition, it also includes help function for user and administrator. It is a user manual to teach user or administrator to use the inventory management system.

5.3.1.4 Report Inventory

This module is implemented to view and print the inventory record. It is using the Crystal Report to generate the report of the inventory.

5.3.2 Administrator Modules

The administrator module also using the stand-alone system the stand-alone system is developed using the visual basic too. This stand-alone PC is connected to the ODBC data source (32-bit) in Control Panel. The system DSN named **INVENTORY** was registered through the ODBC in this PC in order to map the database server through ODBC 32-bit.

The stand-alone system is built to let the administrator to configure the inventory, tool for search, authenticate logon and generate report. Visual Basic can able to call the crystal report to view.

Another disadvantage of the stand-alone system is that, it needs to be installed into the client machine and it needs another extra license to run the system. Further more, it requires another Crystal report license to perform the report printing. This will increase the cost needed to perform daily operation.

5.3.3 Integrate Modules

User module and the administrator module are integrated to the inventory Management System. The module is developed together in order to reduce any duplicate function as well as prevent any confused between the two systems.

5.4 Coding Approach

Inventory management system for Inventory was developed modularly, mainly by using the bottom-up approach. This approach develops functions and procedures before proceeding to the higher-level modules.

Firstly, all table were stored in server database were developed. This was followed by the development of Inventory application.

5.4.1 Coding for Inventory Application

i. Inventory application was developed with the Standard.exe module of Microsoft Visual Basic 6.0. Each unit was a Visual Basic Form, and has an extension of FRM or Visual Basic module, and has an extension of bas.

ii. All forms and modules were stored in a Visual Basic Project.

iii. This project was later compiled to obtain an EXE file.

iv. Package and Deployment Wizard was later used to create setup files so that this application can be installed in client.

5.4.2 Creating Graphics for Inventory Management Systems

Then, Paint and Adobe Photoshop 6.0 were used to create graphics and icon for the Inventory application. For example: main graphics in mdiMain.frm using Adobe Photoshop 6.0 to modify from some picture and integrated its.

Inventory Management System
Faculty of Computer Science
& Information Technology
University Of Malaya

5.4.3 Builder Help for Inventory Management Systems

Then, ForeHelp 5.0 was used to build help for the Inventory application.

Chapter 6: System Testing

6.1 Introduction

System testing ensures that the system is doing and producing the perceived output. It is also a critical element of software quality assurance and represents the ultimate review of specifications, design and coding [33].

Testing is carried out to meet the following objectives:

- To execute a program with the intent of finding an error.
- To demonstrate that software functions appear to be working according to specification and that performance requirements appear to have been meet.

Testing is not the first place where fault finding occurs; but testing is focused on finding faults, and there are two ways to make the testing efforts more efficient and effective.

6.2 Defect Testing

There are two approaches to defect testing in MIS:

1. Functional or black box testing where the tests are derived from the program specification.
2. Structural or white-box testing where the tests are derived from knowledge of the program's structure and implementation.

6.2.1 White-Box Testing

White box testing involves the structure of the code directly [34]. This testing was carried out at the early stages of the testing process to ensure that the internal operations of the system perform according to specification.

White-box testing ensured that

i. All independent paths within a module have been exercised at least once.

ii. All logical decisions are executed on both their true and false sides.

iii. All loops are executed at their boundaries, and within their operational bounds

iv. All internal data structures are exercised to ensure their validity.

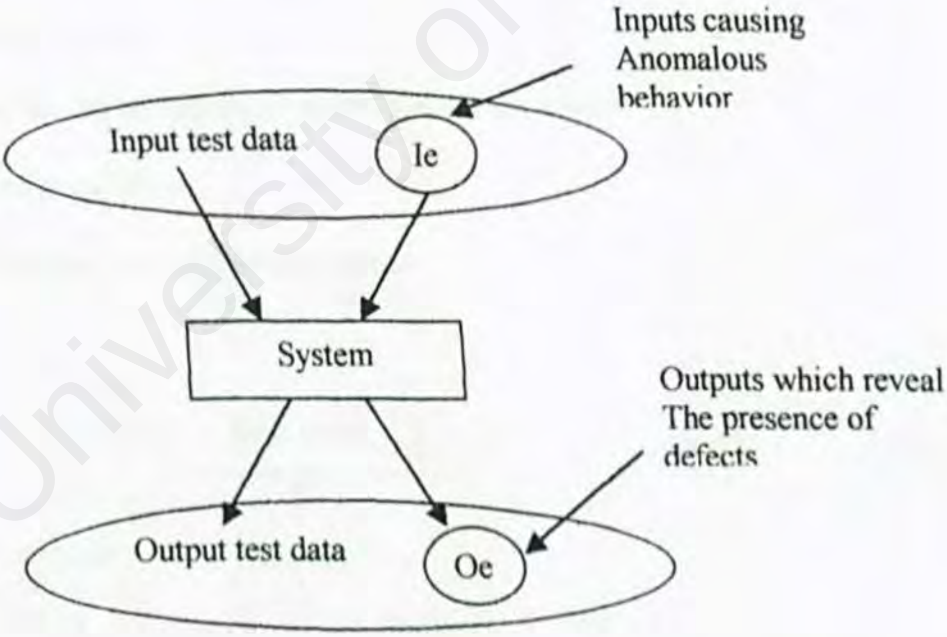


Figure 6.1: Black-box testing

6.1.2 Black Box Testing

Black box testing assumes that the logic structure of the code is unknown. It is a "black box". This is the point at which the function of a module is tested [34]. This testing was used during the later stages of the testing process.

Black box testing is not an alternative to white box testing. It is a complementary approach that can uncover a different class of errors from those uncovered by using white box testing.

Black box testing has found

i. Missing, or incorrect functions

ii. Interface errors

iii. Errors in data structures, or external database access

iv. Performance errors

v. Initialization and termination errors

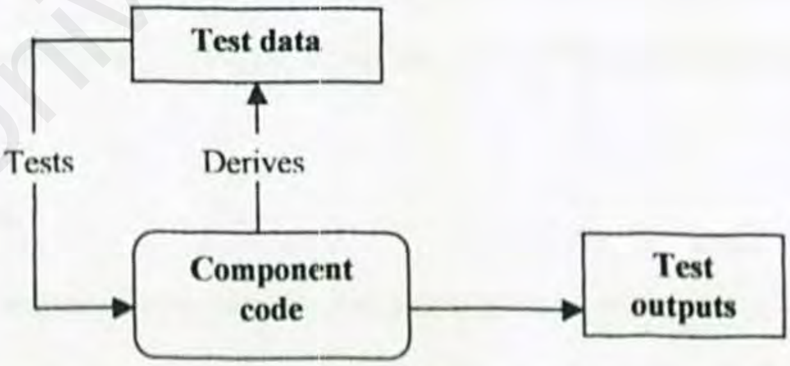


Figure 6.2 White-box testing

6.3 Testing Strategies

Testing conducted include unit testing, module testing, sub-system testing, and system testing. Refer to Figure 6.1 for the testing process.

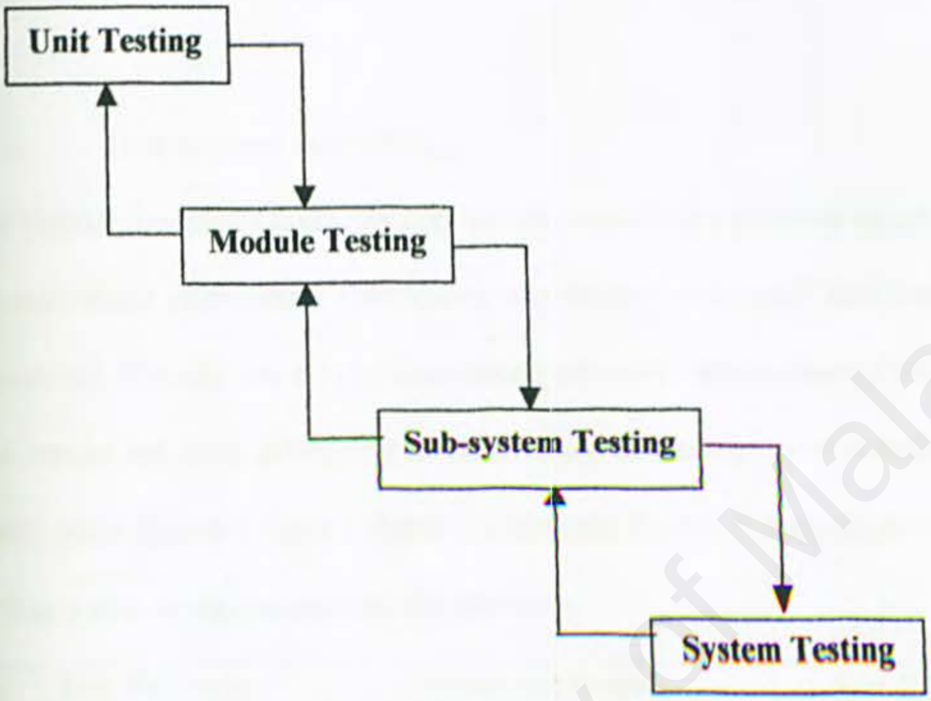


Figure 6.3 Testing process

Bottom-up testing strategy was used where testing starts with the fundamental components, and later works up the hierarchy of modules until the final module is tested [34]. Before performing any of the testing, test data have to be loaded into the database.

6.3.1 Unit Testing

Unit testing tests individual components, to ensure that they operate correctly. These components include functions and subroutines. Each component is tested independently, without other system components [34].

In the development of inventory management system for FSKTM, unit testing was done after the development of each of the component and not at the end of development of the whole system. There were too many unit test cases involved. Therefore, only three will be shown as example.

i. Unit test case example 1

The Vendor module in Inventory application provides the ability to insert, edit and deleted vendor information. Unit testing was carried out on each function once it was completed. If it was tested to be functioning correctly, development of the next function was carried out. Else, debugging was carried out to identify the error before having it tested again. Below Table 6.1 shows the test case for unit testing on the function of adding a new vendor record into the database.

No	Test Procedure	Expected Output	Test Result Analyzing
1	Select Vendor -> Add button	The New Vendor form is displayed	The New Vendor form is displayed at the center of the screen.
2	Fill in the New Vendor form. Click the Update button	The record is inserted permanently.	Record is inserted successfully. Objective of inserting a new vendor record is achieved.

Table 6.1 Test case for Adding New Vendor Record

6.3.2 Module Testing

A module is a collection of dependent components. A module encapsulates these related components. Module testing enables each module to be tested independently without other system modules [34].

i. Module test case example 1

After the New Records module for Inventory application was completed, module testing was carried out to ensure all the tables could be new as expected. Below Table 6.2 shows the test case for New Records module.

No	Test Procedure	Expected Output	Test Result Analyzing
1	Select Vendor -> Add button.	The New Vendor form is displayed.	The New Vendor form is displayed at the center of the screen.
2	Fill in the New Vendor form. Click the Update button.	The record is inserted permanently.	Record is inserted successfully.
3	Positioning the cursor at the position of the record that just added. Click the Edit button. Edit certain column of the record. Click the Update button.	The record is edited permanently.	Record is edited successfully.
4	Delete the record just added by positioning the cursor at the record position. Click the Delete button.	The record is deleted permanently.	Record is deleted successfully. Objective of integrating all the functions of Vendor module is achieved.

Table 6.2 Test case for Vendor Module

6.3.3 Sub-System Testing

Sub-system testing involves testing a collection of modules that are integrated into sub-systems. The most common problem is sub-system interface mismatches. Therefore, this test should concentrate on the detection of interface errors [34].

i. Sub-system test case example 1

For Inventory application, all modules were tested together by using sample data to ensure all the modules can be integrated properly. Below Table 6.3 shows partial of the test case for sub-system of Inventory application. It shows the some of integration testing between inventory modules and inventory directory modules.

No	Test Procedure	Expected Output	Test Result Analyzing
1	Select Inventory -> Inventory menu	The entire text box in Inventory form is displayed showing the list of all the inventory records and a data grid is displayed showing the list of all the inventory parts records.	The entire text box in Inventory form is displayed showing the list of all the inventory records and a data grid is displayed showing the list of all the inventory parts records.
2	Clicked the combo box of Inventory code.	Inventory code, name and group is displayed in the list.	Inventory code, name and group is displayed in the list.
3	Choose the list of the combo box with the double click at the record position.	Inventory code and name is displayed in the inventory text box.	Inventory code and name is displayed in the inventory text box.
4	Add the record inventory just click the Update button and add the record inventory parts just added by positioning the cursor at the record position. Click the entry key.	The record inventory and inventory parts is adding permanently.	The record inventory and inventory parts is adding successfully.
5	Click the Refresh button.	The New record inventory is adding permanently.	The New record inventory is adding successfully. Objective of integrating the Inventory module and Inventory Directory module is achieved.

Table 6.3 Partial Test case for Sub-system of Inventory application

6.3.4 System Testing

The sub-systems are integrated to obtain the entire system. System testing is concerned with finding errors, which result from unanticipated interactions between sub-systems and system components. It is also concerned with ensuring that the system meets its functional and non-functional requirements [34]. Once the entire system is validated, it must also be combined with other system elements such as hardware, end-user and databases. System testing verifies that elements are functioning properly, and the overall system performance and objectives are achieved, which in this case, inventory management system for Inventory must achieve reliability, robustness, accuracy, flexibility and modularity.

System testing was carried out Usability Test. This test is an attempt to find human-factor, or usability, problems. The following is a list illustrating the kinds of considerations [35] that have been tested and the test result analyzing with the help from the five non-computer base testers and five computer base tester.

No	Consideration	Test Result Analyzing
1	Has each user interface been tailored to the intelligence, education background and environmental pressures of the end user?	100% of the non-computer base testers do not facing any problem in using the system. Could be concluded that the system is not only tailored to the intelligence end user.
2	Are the outputs of the program meaningful, no abusive, devoid of "computer gibberish" and son on?	100% of the testers agreed that all the outputs of the program are meaningful and proper labeled.
3	Does the total set of user interfaces exhibit considerable "conceptual integrity, an underlying consistency and	88% of the testers think the user interface of the system are consistent while 12% of the testers think the

	uniformity of syntax, conventions, semantics, format, styles, and abbreviations?	interface less attractive. Could be concluded that the testers are satisfied with the consistency of the user interface except for the interface more professional but less attractive.
4	Is the program easy to use? For instance, does entering a command into a time-sharing system require repeated shifts between upper- and lower-case characters?	100% of the testers agreed that the system is easy to use. Could be concluded that the system is easy to use.

Table 6.4 Usability testing

Chapter 7 System Evaluation and Conclusion

System evaluation is a process of evaluating the system developed, by identifying the system strength, system limitations, and future enhancements. It also highlights the knowledge gained, and problems encountered during the development of the system and solutions taken to overcome these problems.

7.1 System Strength

7.1.1 Update and delete cascading

These inventory management system enables delete and update cascading. When user updating and deleting certain record, integrity constraints will be enforced automatically by table triggers that free user from the need to understand the architecture of the server database.

7.1.2 Limited access to authorized users

In Inventory application, a user only has access to modules that are required to perform his or her duties. For instance, only the manager of super administrator will has access to all modules in Inventory application, while the user will not has access to system module and change the information of inventory.

7.1.3 Graphical user interface (GUI)

Inventory application is developed on the principal that it is easy to use. By using Microsoft Visual Basic 6.0, Inventory application is able to provide a fully interactive and user-friendly feature for the user.

7.1.4 Authorization and authentication

Every user has a user ID and password. Inventory application only allows authorized users to access the system. Furthermore, it has the capability of preventing unauthorized users from accessing the applications. An invalid login message will be prompted if a user tries to use the system without logging in or logging in wrongly. It ensures users login before entering the system.

7.1.5 Informative message

Inventory application provides error messages when a user attempts to perform illegal actions. Besides, Inventory application also provides messages after a certain task has been completed. These messages allow the users to understand what is going on, and keep users informed of what has been done.

7.1.6 Consistency

The screen design maintains its consistency throughout the system. Thus, users face less difficulty in seeking for a particular option that they require.

7.1.7 System transparency

System transparency refers to the condition where the users do not need to know where the database resides, the system structure, the database management system, (DBMS) or anything related to the system built.

7.1.8 Scalability

Hardware and applications could be easily added to the existing system without influence the existing applications. This is because the system is not hardware dependency.

7.1.9 Help

The help is useful to provide guidelines for the user to use the system effectively. The help will be displayed as soon as the user clicks on the Using IMS menu in the help module. This reduces the time for searching it through the user manual in the book.

7.2 System Limitations

The system has some limitations due to the time constraints and other factors

7.2.1 Database not encrypted

Inventory information stored in server database is not secured enough because it is stored in a plain text format rather than in encrypted format.

7.2.2 Graphical user interface (GUI)

Inventory application is developed on the principal that it is standard form to use. However, by using Microsoft Visual Basic 6.0, Inventory application is not able to provide a fully attractive interface for the user.

7.3 Future Enhancements

Due to time constraint, only the main modules of the inventory management system for Inventory are built, there are still rooms for future enhancement on the system. Below show some of the features that could be added into the current system.

7.3.1 Provide additional reports

In real world, various reports are needed for the inventory management system. Current Inventory applications only provide some of the main needed reports. There should be additional function that enable user to customize the fields in certain report.

7.3.2 Provide ability to make the purchase order to vendor

Current Inventory application only have vendor information, it will be easier for administrator if the Inventory application will be able to generate purchase order.

7.3.3 More administrative task

Administrator task could be further enhanced to include more features to ease maintenance process. Among the features may be included are report generation, multiple user grouping according to access right, and database backup.

7.3.4 Support More Email Services

To benefit all the users and administrators, an email service can be integrated in to the system to response and confirm order via an automatic email reply.

7.4 Problem Encountered

During the entire development of the Inventory Management System, many and various problem was encountered. Some of them could be overcome through certain solution while some of them were not. The problem that encountered can be divided into different type. The following are some of the problem that arises during the development process.

7.4.1 Set Up and Configuration

The set up of the server are critical for the operation of the application developed. However, the setup process take a long time because lack of experience. Besides, the repeated failure of the server required re-installation of the server as a remedy and this consumed time and effort.

We are also encountered problem during the exportation of database from Access to SQL server and SQL server to SQL server. After the exportation, some of the data structure of the database is changed and we need to design the database again.

The installation of the Windows NT Server 4.0 is troublesome. This is because we need to restart the server each time we upgrade the Service Pack and configured the network setting.

7.4.2 Requirement Changes from the User

It is very difficult to develop and implement the system when the requirement changes very frequently. Sometimes it is easy to change the requirement, however, the coding need to be changed a lot in order to follow the new requirement.

7.5 Knowledge Gained

Knowledge gained throughout the development of inventory management system for FCSIT is undoubtedly valuable.

Knowledge was gained on the client-server development environment, the database system, programming and concepts, as well a LAN configuring.

There was an exposure to software tools that include Microsoft Visual Basic 6.0, Microsoft SQL Server 7.0, Microsoft Access 2000 and Crystal Report 4.6 that proves to be beneficial. Programming in Visual Basic proves to be a valuable experience. Visual Basic technology turned out to be a suitable technology to develop client-server system.

It is undeniable that programming skills and techniques are important in developing the system. However, good software engineering techniques must also be applied in software development. Here, theories and knowledge gained throughout the course of computer science studies like system analysis, design and software engineering were literally put into practice.

There is also better understanding and grasp of inventory management concepts and inventory management cycles.

There is also improvement in skills of finding information and classifying facts, besides enriching experience in problem solving, and acquiring the ability to work independently. Better documentation and report writing skills were also obtained. There is also improvement in time management skills.

7.6Conclusion

Inventory Management System for Inventory has been completed successfully, with some strengths as well as limitations as mentioned. Inventory management system has achieved and fulfilled the objectives and requirements, as stated in the proposal and system analysis.

However, there are still rooms for improvement in the inventory management system for FCSIT. Enhancements include the introduction of additional features in the Inventory application for future versions.

Finally, this project has achieved its objective in giving the undergraduates an opportunity to undergo different challenges in different phases of the system development, that include research, system planning, system analysis, system design, system implementation, and system testing.

BIBLIOGRAPHY

[1] Russell A. Stultz, *Illustrated dBASE IV*, Version 1.1, 1991

[2] Spencer B.Smith, *Computer Based Production And Inventory Control*, 1989

[3] Richard J. Tersine, *Principles of Inventory and Materials Management*, North Holland, 1982

[4] www.bigsoftware.com

[5] www.mrtec.com

[6] www.winsellpos.com

[7] B. Bergsten, M. Couprie and P. Valduriez, *Overview of Parrallel Architectures for Database*, Computer Journal, Vol 36, 1993

[8] Joe Salemi, *Guide to Client/Server Databases*, Ziff-Davis Press Second edition, 1995

[9] Patrick Smith, *Client/Server Computing*, Sams Second Edition, 1994

[10] Abraham Mehrez, Geoffrey S. Howard, Yaakov Laugassi and Perezt Ahoval, *Information System Planning and Selection: A Multiattribute Theoretic Approach*, Computer Journal, Vol 36 1993

[11] P. Rob, C, Coronel, *Database System Designs, Implementation, and Management*, 1997

[12] L. Mortensen, R. Sawtell, and M. Lee, *SQL Server 7 Administration Study Guide*, Sybex, 1998

[13] D. Zantinge, and P. Adriaans, *Managing Client/Server*, Addison Wesley, 1996

[14] Stephen Wynkoop, *Special Edition Using SQL Server*, QUE, 1996

[15] www.zdnet.co.uk/pcmag/labs/2000/04/Visual

[16] Online book of Visual Studio (MSDN)

[17] Kent Reisdorph, *Teach Yourself Borland Delphi 4 in 21 Days*, Sams, 1996

[18] Wynkoop, Stephen, *Special Edition Using Microsoft SQL Server 7.0*, Que, 1999.

[19] Michael Richards, *Oracle Unleashed*, Sams, 1996

[20] *Introduction: Oracle 8i vs SQL Server 7.0*
www.oracle.com/html/8i_vs_sql.html

[21] Cowart, Robert, *Windows NT Server 4*, Sybex, 1997

[22] Fertuck, Len, *System Analysis & Design*, Wm. C.Brown Communications, Inc, 1995.

[23] Pfleeger, Shari Lawrence, *Software Engineering Theory and Practice*, Prentice Hall, 1998.

[24] Ramez Elmasri & Shamkant B. Navathe, *Fundamentals of Database Systems*, World Student Series Edition, 1994

[25] Kenneth E.kendall & Julie E.kendall, 1999, *System Analysis & Design 4th Edition*, Prentice Hall International, Inc.

[26] Royce, W.W., *Managing the Development of Large Software Systems: Concepts and Techniques*, Proc. WESCON, August 1970.

[27] Igor Hawryszkiewycz, 1998. *Introduction to Systems Analysis and Design, 4th Edition*, Prentice Hall Australia Pty. Ltd.

[28] Elmasni, R-1994, *Fundamental of Database System 2nd Edition*, CA, Penjamin-Cumminmgs Company IMC.

[29] Worpoys, M-1989, *Relational Database: A Theoretical Primer. Information and Software Technology*.Vol 31.No 3, Page 115-112.

[30] Nilakanta S-1989, *Controling User Authorization in Relational Database Management System.Information and Software Technology.* Vol 31.No 6, Page 290-294.

[31] Pangalos, G-1994, *Tutorial of Secure Database System.Information and Software Technology*.Vol 36.No 2, Page 717-723.

[32] McDermid, J., and P.Rook, *Software Development Process Models*, in Software Engineer's Reference Book, CRC Press, 1993, pp 15/26-15/28

[33] Pressman, R.S., "Software Engineering: A Practitioner's Approach", McGraw-Hill, Inc., 1997

[34] Roger S. Pressman, Ph.D., "Software Engineering: A Practitioner's Approach", *fitth edition* McGraw-Hill, Inc., 2001

[35] Abraham Mehrez, Geoffrey S. Howard, Yaakov Laugassi and Perezt Ahoval, "Information System Planning and Selection: A Multiattribute Theroetic Approach", Computer Journal, Vol 36, 1993.

Table of Contents

Introduction

The inventory management system for FCSIT is a system that helps user to manage inventory easily.

This system has a major application, namely **Inventory Application**. This is an application program will be installed in client side.

About This Manual

This manual is organized into 4 main sections, namely

i. **How to Use the Inventory Management System for FCSIT**

This section gives an overview of what should be done from the very first step.

ii. **Setting Up database**

This section contains instruction to be followed to create a database at server.

iii. **System installation**

This section contains instruction to be followed in installing the inventory management system for FCSIT.

iv. **Using Inventory Application**

This section explains how a user uses the Inventory application. Tasks involved include handling of inventory information, vendor information, reports, user information and change of user password.

Part 1 How to Use the Inventory Management System

The administrator should

1. Setting up database
2. Install Inventory application
3. Configure the DSN for server database

Part 2 Setting Up database

Inventory management system for FCSIT is a server/client system. A server database should be created at server before the application program could be executed.

2.1 Create Database

Before the system could be used, a database named INVENTORY should be created at NT server using Microsoft SQL Server 7.0. Follow steps below to create the INVENTORY database.

1. Click on Start – Programs – Microsoft SQL Server 7.0 – Enterprise Manager.
2. Select the SQL Server Group and right click on the database
3. Click on the new database
4. Type in INVENTORY as the name of the database and click on OK button.

2.2 Create Tables

After you have created the INVENTORY database, you have to create tables below in order to use the system.

1. **Inventory Table**

This table stores all the inventory records.

Field Names	Data Type	Size	Description
KOD	Text	50	Item Code
TAG	Text	50	Tag
PARTS	Number (Double)	50	Parts
NAMA	Text	50	Name
JENAMA	Text	50	Brand
PO_NUM	Text	50	Purchase Order Number
PO_DATE	Date/Time	50	Purchase Order Date
DEPT	Text	50	Department

BILIK	Text	50	Room Number
PEMBEKAL	Text	50	Vendor
NO_SIRI	Text	50	Serial Number
HARGA	Number (Double)	10	Price
TARAF	Text	50	Status
BIAYAI	Text	50	Finance
CATITAN	Text	50	Records
ENTRY_DTE	Date/Time	8	Entry Date
GUNASAMA	Text	50	Sharing

2. Inventory Parts Table

This table stores all the parts or inventory records.

Field Names	Data Type	Size	Description
KOD	Text	50	Item Code
TAG	Text	50	Tag
SUBTAG	Text	50	Sub Tag
NAMA	Text	50	Name
JENAMA	Text	50	Brand
DEPT	Text	50	Department
BILIK	Text	50	Room Number
NO_SIRI	Text	50	Serial Number
TARAF	Text	50	Status
CATITAN	Text	50	Records
ENTRY_DTE	Date/Time	8	Entry Date

3. Inventory Department Table

This table stores all the department of the inventory records.

Field Names	Data Type	Size	Description
DEPTCODE	Text	50	Department Code
FACULTY	Text	50	Faculty
DEPTNAME	Text	50	Department Name

4. Sharing Table

This table stores all sharing of the inventory records.

Field Names	Data Type	Size	Description
TAG	Text	50	Tag
PERSON1	Text	50	Person 1
PERSON2	Text	50	Person 2
MAINTAIN	Text	50	Maintain
CHARGE	Text	50	Charge
REMARKS	Text	50	Remarks

5. Personal Table

This table stores all the personal records.

Field Names	Data Type	Size	Description
USER_NO	Text	50	User Number
PASSWORD	Text	50	Password
ACTUAL_NAME	Text	50	User Actual Name
STATUS	Number (Single)	10	User Status (1->Administrator, 2->User)
EMAIL	Text	50	User Email Address

6. Inventory Directory Table

This table stores all the directory inventory records.

Field Names	Data Type	Size	Description
ITEMCODE	Text	50	Item code
ITEMNAME	Text	50	Item name
GROUP	Text	50	Group

7. Inventory Delete Table

This table stores all the deleting the inventory records.

Field Names	Data Type	Size	Description
KOD	Text	50	Item code
TAG	Text	50	Tag
SUBTAG	Text	50	Sub Tag
PARTS	Number (Double)	10	Parts
NAMA	Text	50	Name
JENAMA	Text	50	Brand
PO_NUM	Text	50	Purchase Order Number
PO_DATE	Date/Time	8	Purchase Order Date
DEPT	Text	50	Department
BILIK	Text	50	Room Number
PEMBEKAL	Text	50	Vendor
NO_SIRI	Text	50	Serial Number
HARGA	Number (Double)	10	Price
TARAF	Text	50	Status
BIAYAI	Text	50	Finance
CATITAN	Text	50	Records
ENTRY_DTE	Date/Time	8	Entry Date
GUNASAMA	Text	50	Sharing
REASON	Text	50	Reason
DTE_DELETED	Date/Time	8	Date Deleted

8. Vendor Table

This table stores all the vendor records.

Field Names	Data Type	Size	Description
KOD	Text	50	Vendor Number
PEMBEKAL	Text	50	Vendor
ALAMAT	Memo	255	Address
TELEFON_NO	Text	50	Telephone
EXT	Text	50	Extension
FAX	Text	50	Fax Number
EMAIL	Text	50	E-mail Address
CONTACT_PERSON	Text	50	Contact Person Name
CP_TELEFON_NO	Text	50	Contact Person Telephone
REMARKS	Text	50	Remarks

9. Last Number Table

This table stores running records only.

Fields Name	Data Type	Size	Description
Running	Number (Int)	10	Keep the last number of the inventory table

Part 3 System Installation

There is a application to install and set, inventory application. This application should be installed in the client side. Please check the hardware and software requirements in section 3.1 and section 3.2 before installing the system.

3.1 Hardware and Software Requirements for Server

i. At least 32 MB RAM

ii. At least Pentium 166 MHz processor

iii. At least 100 MB free hard disk space

iv. And other standard peripherals, that include keyboard, mouse, and monitor

v. Running on Window NT Server 4.0

vi. Microsoft SQL Server 7.0

3.2 Hardware and Software Requirements for Clients

i. At least 32 MB RAM

ii. At least Pentium 166 MHz processor

iii. At least 20 MB free hard disk space

iv. And other standard peripherals, that include keyboard, mouse, and monitor

v. Crystal Report 4.6 and above

3.3 Installing Inventory Application

1. Run install\setup.exe in the CD-ROM in folder Inventory
2. Follow the instructions given.

3.4 Configuration of DSN for Server Database

Before Inventory application can be functioned properly, DSN for server database should be configured. Note that the SQL Service must be set running before the configuration. Do this by going to the SQL Service Manager, and double click on Start/Continue.

1. Click on Start – Setting – Control Panel.
2. Click on the icon of ODBC Data Sources (32 bit) from the screen.
3. Select the tad of System DSN.
4. Click on the Add button. A driver selection list will be displayed.
5. Select SQL Server from the list
6. Enter the DSN as INVENTORY
7. Select the server where the server database located and click on the next button.
8. Select the option of "With SQL Server authentication using login ID and password entered by the user".
9. Filled up the login ID as "sa" click on the next button.
10. Change the default database to INVENTORY database and click on the next button.
11. Without change any of the default and click on the finish button.
12. Click on the "Test Data Source" button to test whether it is successful to connect to the server database.
13. If the connection is successful click on the OK button, else reconfigure from the step 1.

3.5 Configuration of DSN for Local Database

Before Inventory application can be functioned properly, DSN for local database should be configured.

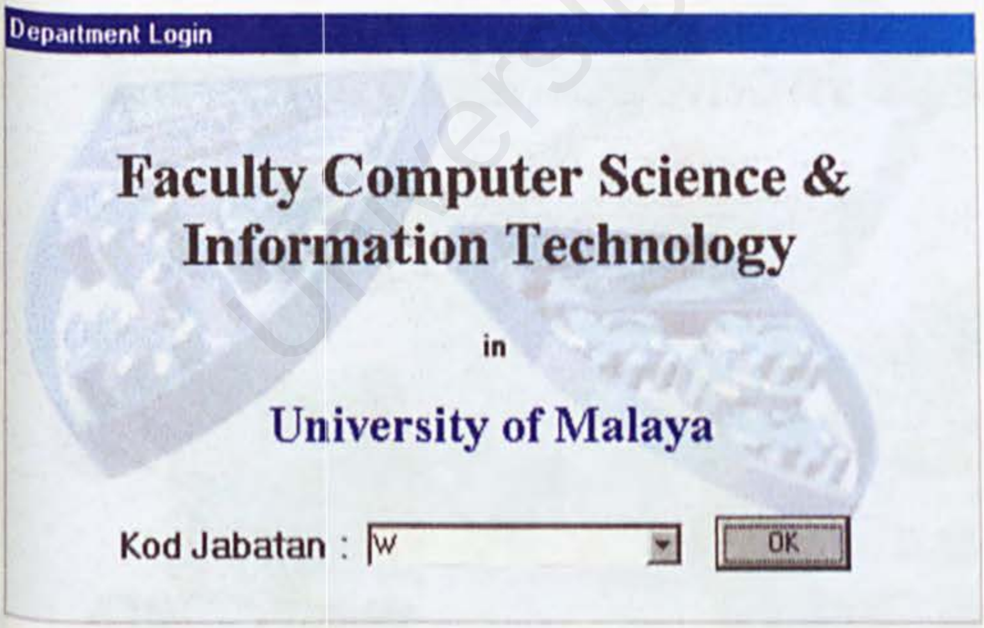
1. Click on Start – setting – Control Panel.
2. Click on the icon of ODBC Data Sources (32 bit) from the screen.
3. Select the tad of User DSN.
4. Click on the Add button. A driver selection list will be displayed.
5. Select SQL Server from the list
6. Enter the DSN as INVENTORY
7. Click on the select button to select the INVENTORY database. (It will be located at the path that you have selected to install the inventory application) and then click on the OK button.

Part 4 Using Inventory Application

Note that the SQL Service must be set running before the administrator and users can access the database. Do this by going to the SQL Service Manager, and double click on Start/Continue.

4.1 Begin

1. Click on Start – Inventory – Inventory to start this application
2. Login department form to certain faculty of the University Malaya is displayed.
3. The Login department form set with Faculty Computer Science and Information Technology with the code department which is "W".
4. Click OK to login to Inventory of the FCSIT or other code department – "W" to exit this application. (Due to this application just only for Inventory in the FCSIT)



Department Login

Faculty Computer Science & Information Technology

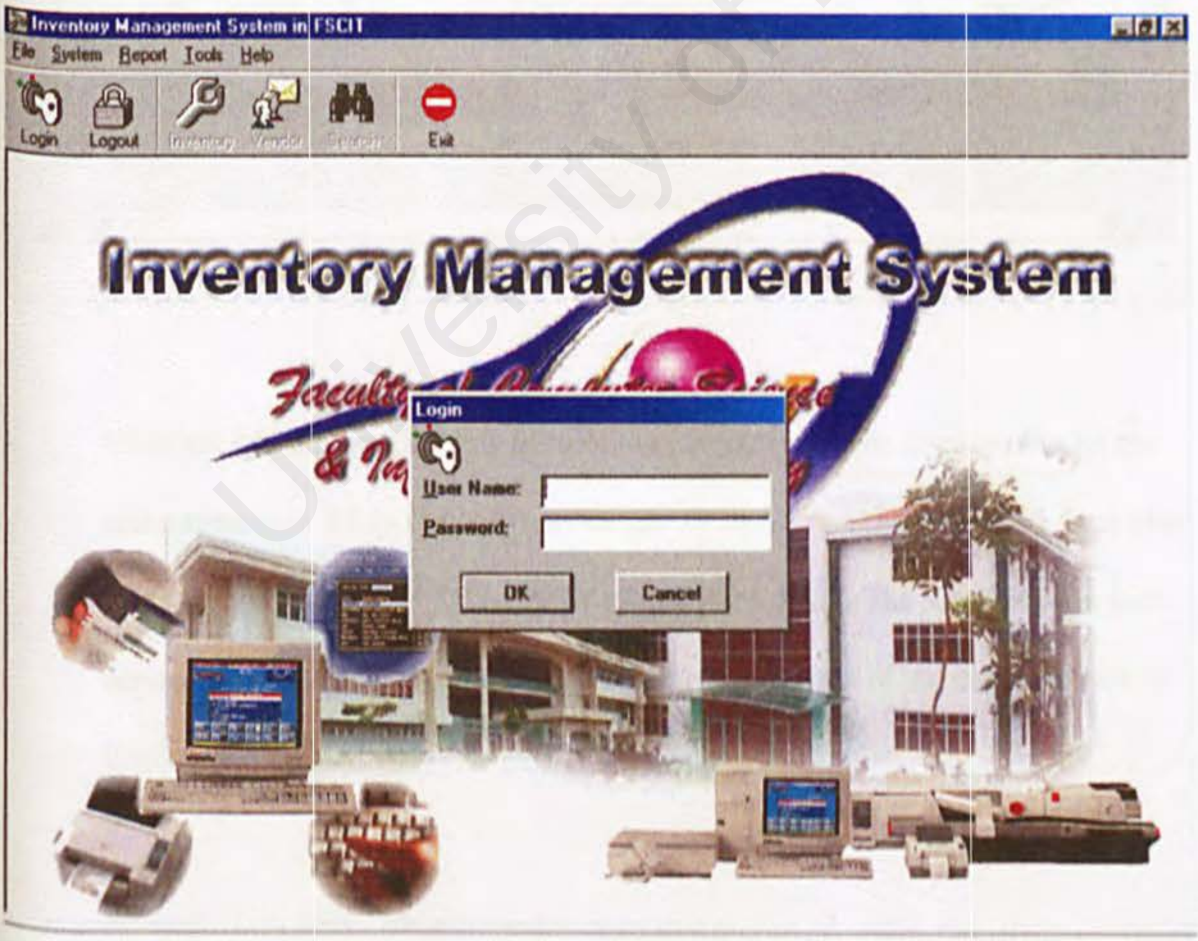
in

University of Malaya

Kod Jabatan : W

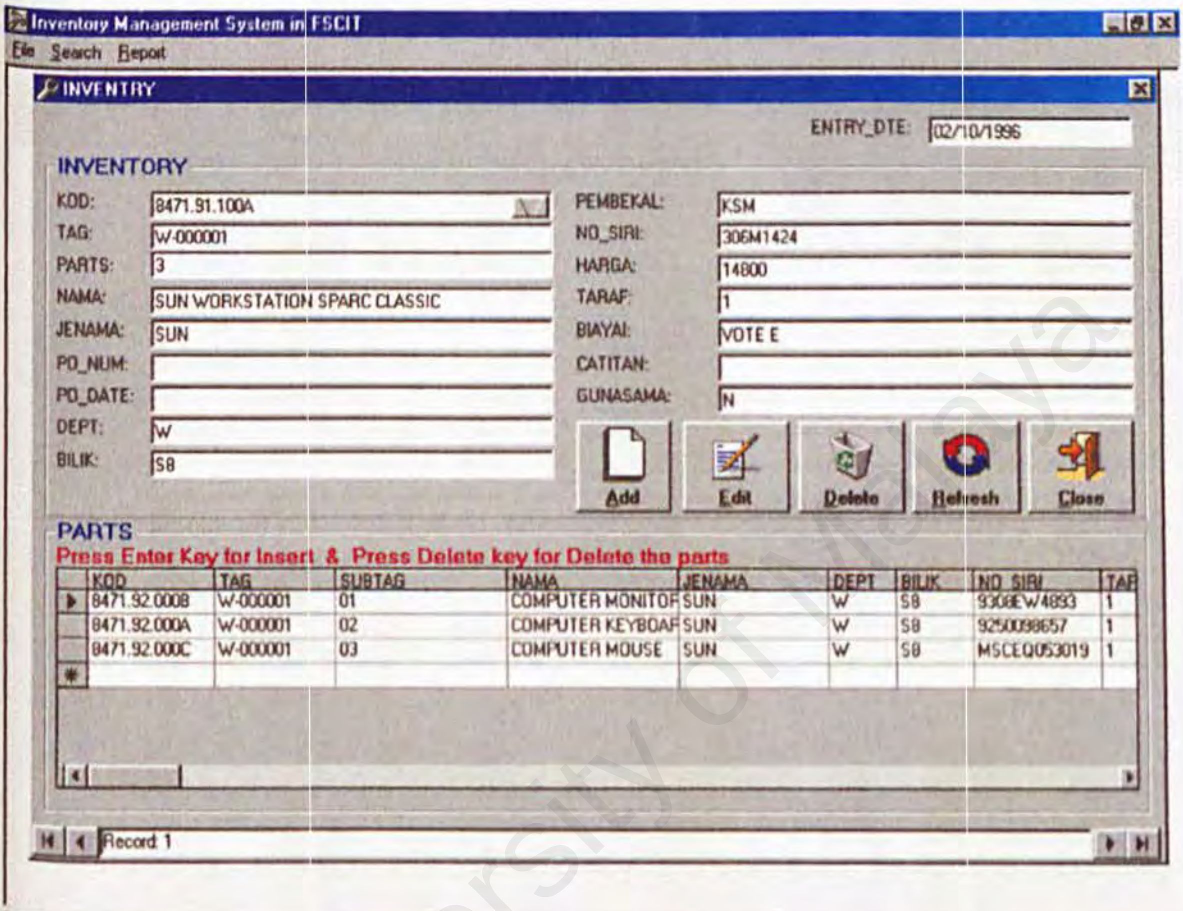
4.2 Main Menu

1. After login department successfully, a main menu with the login screen is displayed.
2. Login to continue or Click on Exit to exit this application.
3. If the Login status is Super administrator, all the modules in the main menu will be used. If the Login status is User, just have only report; tools and help module will be used for User. Choose the module that needed by clicking using the mouse or using keyboard by pressing together the Alt button and the alphabet that have been underlined.
4. Certain function can be accessed through the toolbar.


Inventory Management System in FSCIT
File System Report Tools Help
Login
Logout
Inventory
Vendor
Search
Exit
Inventory Management System
Faculty of Computer Science & Information Technology
Login
User Name:
Password:
OK
Cancel

4.3 How to Create New Inventory

1. From the main menu, select System module to choose the Inventory. The Inventory form will be displayed.



ENTRY_DTE: 02/10/1996

INVENTORY

KOD:	8471.91.100A	PEMBEKAL:	KSM
TAG:	W-000001	NO_SIRI:	306M1424
PARTS:	3	HARGA:	14800
NAMA:	SUN WORKSTATION SPARC CLASSIC	TARAF:	1
JENAMA:	SUN	BIAYAI:	VOTE E
PO_NUM:		CATITAN:	
PO_DATE:		GUNASAMA:	N
DEPT:	W		
BILIK:	S8		

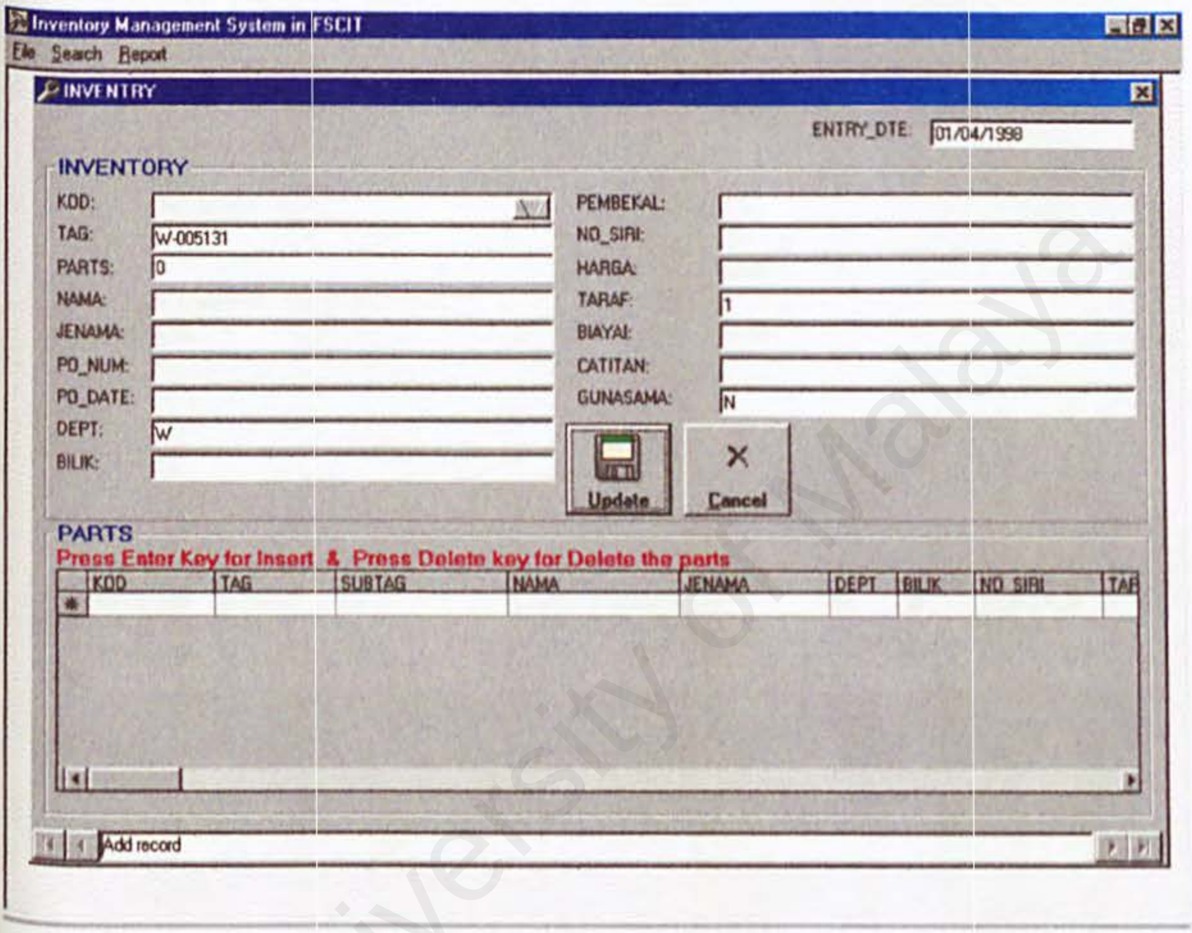
PARTS

Press Enter Key for Insert & Press Delete key for Delete the parts

KOD	TAG	SUBTAG	NAMA	JENAMA	DEPT	BILIK	NO_SIRI	TAF
8471.92.000B	W-000001	01	COMPUTER MONITOF	SUN	W	S8	9308EW4893	1
8471.92.000A	W-000001	02	COMPUTER KEYBOAF	SUN	W	S8	9250098657	1
8471.92.000C	W-000001	03	COMPUTER MOUSE	SUN	W	S8	MSCEQ053019	1

2. Click the add button, the New form of the Inventory will be displayed to let the administrator to fill in the inventory detail. At the same time, inventory form also provide administrator to fill in the inventory parts detail. The administrator just need to press the enter key on the keyboard, the new row of the data grid will be displayed.

3. After fill in the inventory detail, click the update button to update in the database or cancel button to cancel the add function. At the same time, after fill in the inventory parts detail just click other space of the data grid the inventory parts will automatic update to the inventory parts database



ENTRY_DTE: 01/04/1998

INVENTORY

KOD:		PEMBEKAL:	
TAG:	W-005131	NO_SIRI:	
PARTS:	0	HARGA:	
NAMA:		TARAF:	1
JENAMA:		BIAYA:	
PO_NUM:		CATITAN:	
PO_DATE:		GUNASAMA:	N
DEPT:	W		
BILIK:			

PARTS

Press Enter Key for Insert & Press Delete key for Delete the parts

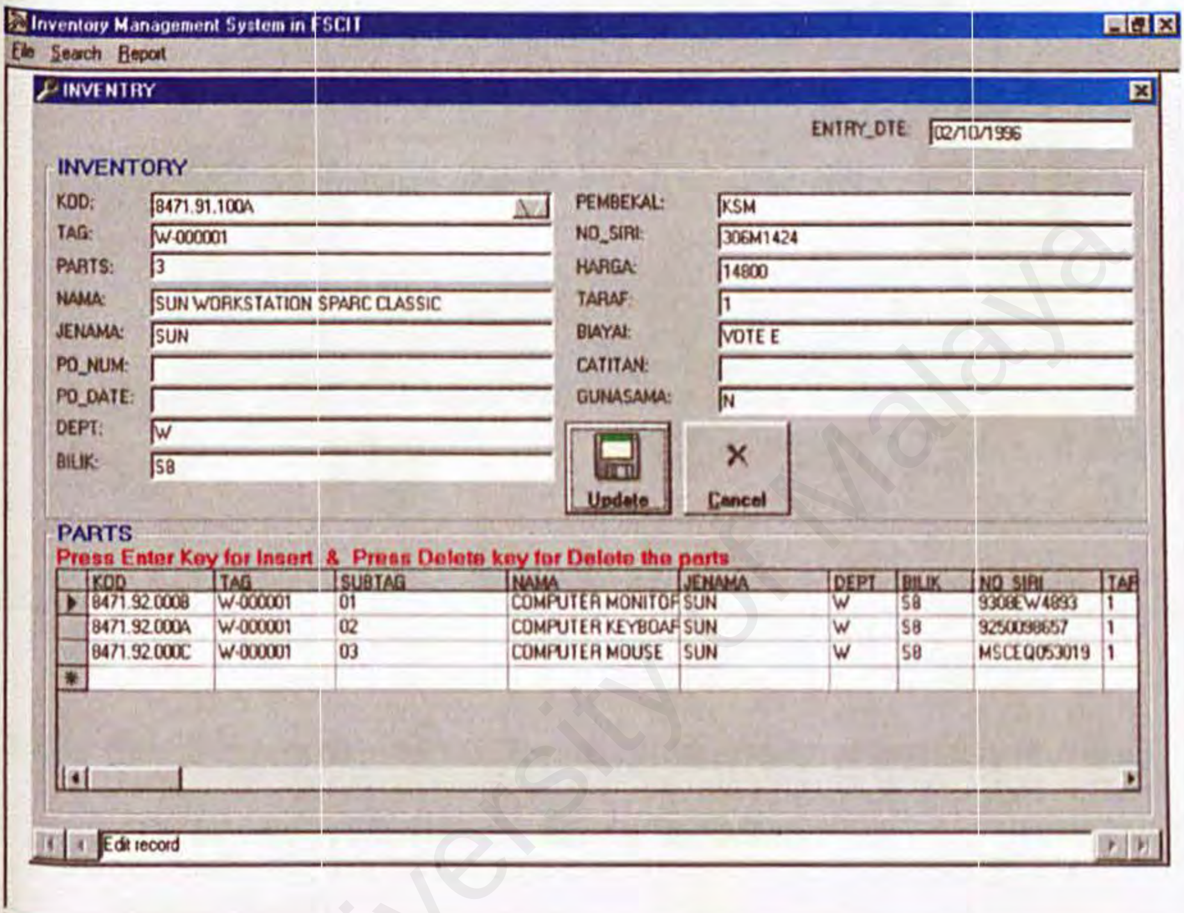
	KOD	TAG	SUBTAG	NAMA	JENAMA	DEPT	BILIK	NO SIRI	TAR
*									

Add record

4.4 How to Edit or Delete Inventory

1. From the Inventory form, click the edit button, you can modify the inventory information and click on the update button to confirmation the modification or click on the Cancel button to undo the modification. For edit the inventory parts, you just need to change its information and click to the other space of the data grid. It is auto edit its information.

2. If you click on the Delete button, the record will immediately deleted from the database. For delete the inventory parts, you just need to highlights all to row in the data grid and press the delete key on the keyboard it will automatic delete the information from the inventory parts database.



ENTRY_DTE: 02/10/1996

INVENTORY

KOD:	8471.91.100A	PEMBEKAL:	KSM
TAG:	W-000001	NO_SIRI:	306M1424
PARTS:	3	HARGA:	14800
NAMA:	SUN WORKSTATION SPARC CLASSIC	TARAF:	1
JENAMA:	SUN	BIAYAI:	VOTE E
PO_NUM:		CATITAN:	
PO_DATE:		GUNASAMA:	N
DEPT:	W		
BILIK:	58		

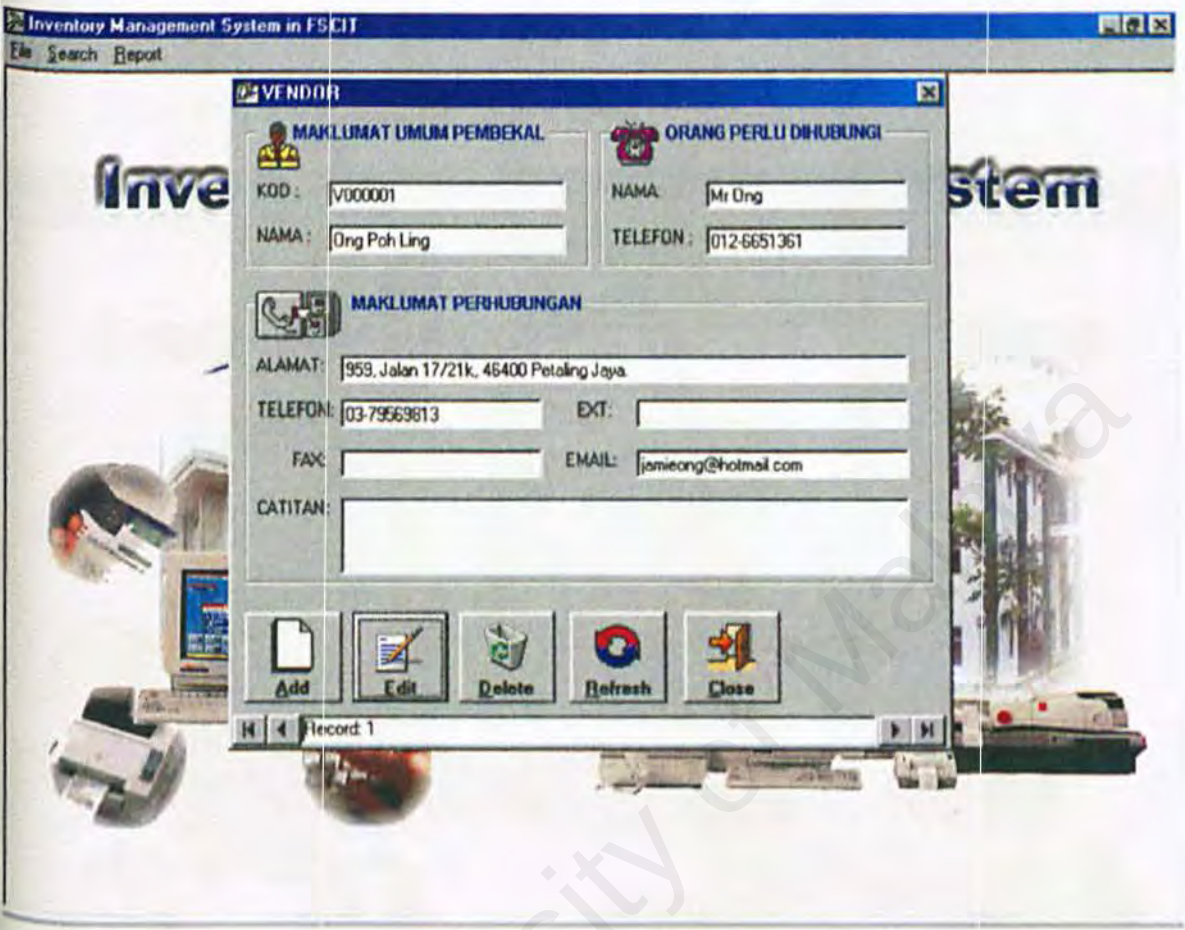
PARTS

Press Enter Key for Insert & Press Delete key for Delete the parts

	KOD	TAG	SUBTAG	NAMA	JENAMA	DEPT	BILIK	NO SIRI	TAF
▶	8471.92.000B	W-000001	01	COMPUTER MONITOF	SUN	W	58	9308EW4893	1
	8471.92.000A	W-000001	02	COMPUTER KEYBOAF	SUN	W	58	9250098657	1
	8471.92.000C	W-000001	03	COMPUTER MOUSE	SUN	W	58	MSCEQ053019	1
*									

4.5 How to Create New Vendor

1. Form the main menu, Select the System module and choose the vendor.



VENDOR

MAKLUMAT UMUM PEMBEKAL

KOD : V000001

NAMA : Ong Poh Ling

ORANG PERLU DIHUBUNGI

NAMA: Mr Ong

TELEFON : 012-6651361

MAKLUMAT PERHUBUNGAN

ALAMAT: 959, Jalan 17/21k, 46400 Petaling Jaya

TELEFON: 03-79569813

EXT:

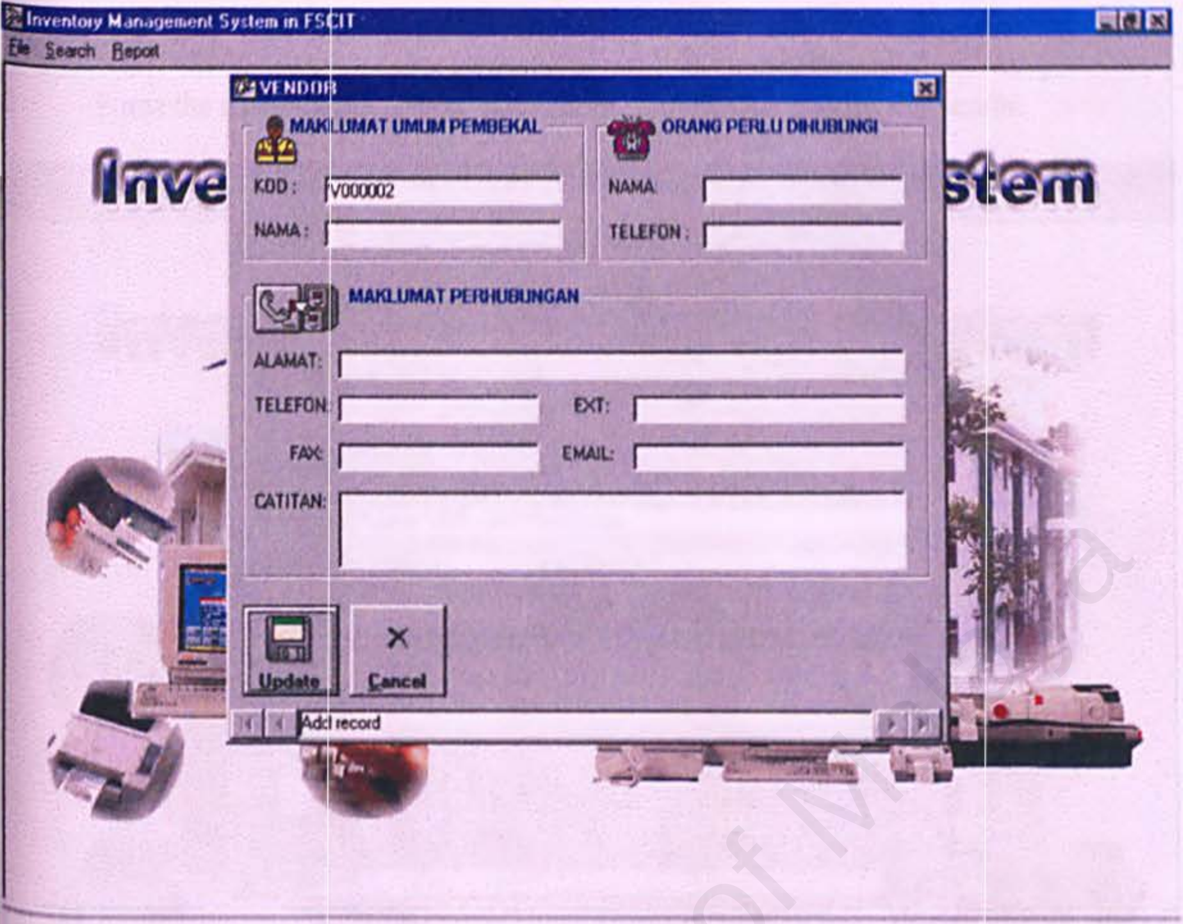
FAX:

EMAIL: jamieong@hotmail com

CATITAN:

Record: 1

2. Click the add button on the vendor form and enter the vendor's code in the first field. This is the only required field on the vendor information screen.
3. Tab through the fields and enter in the information that is appropriate for each field.
4. Click on the other tabs on this screen to enter notes.
5. When all of the information is entered, click the update button to save the vendor.

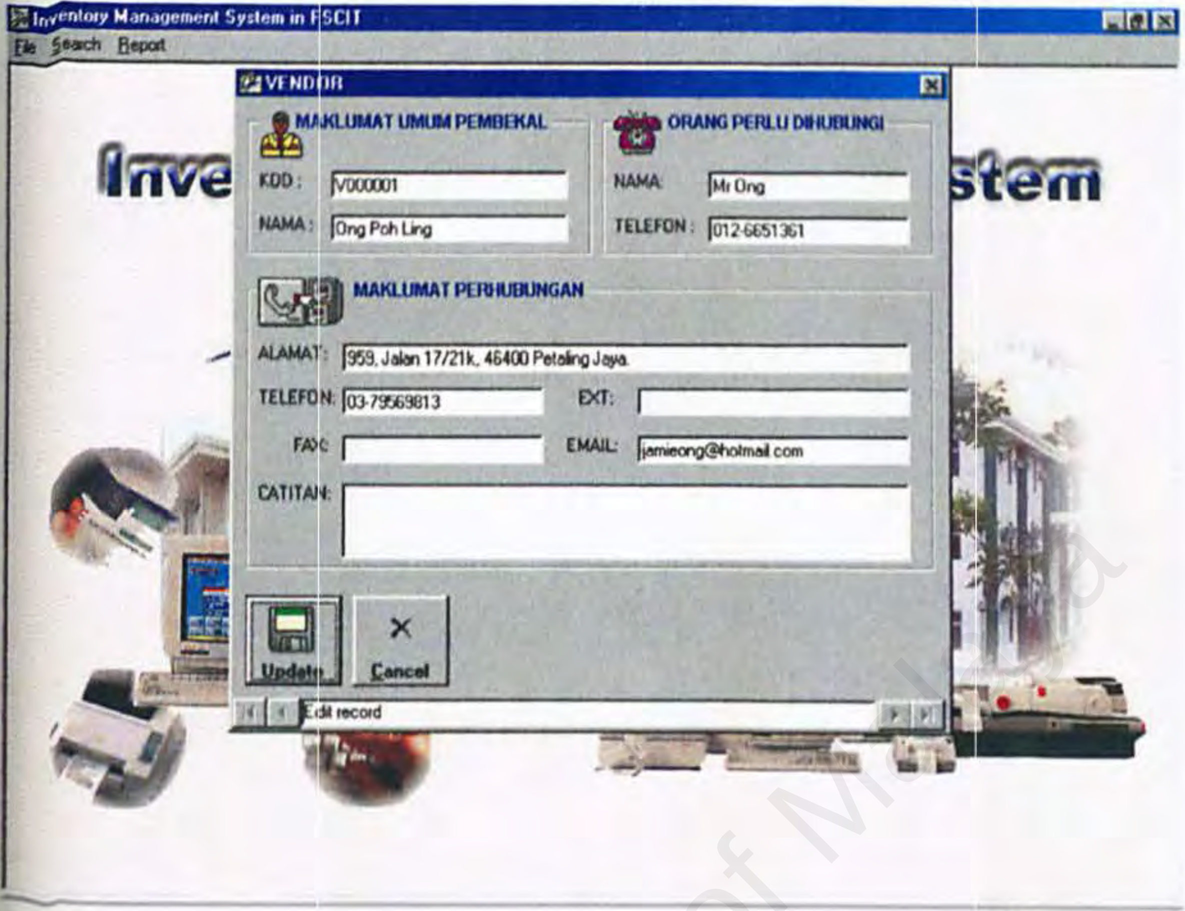


KOD: V000002

NAMA:

4.6 How to Edit or Delete Vendor

1. From the Vendor form, find the records that need to be edited or deleted and click on the Edit or Delete button.
2. To find the records, you can either click the first, previous, next or last button.
3. If you click the Edit button, you can modify the vendor information and click on the Update button to confirmation the modification or click on the Cancel button to undo the modification.
4. If you click on the Delete button, the record will immediately deleted from the database.



MAKLUMAT UMUM PEMBEKAL

KOD: V000001

NAMA: Ong Poh Ling

ORANG PERLU DIHUBUNGI

NAMA: Mr Ong

TELEFON: 012-6651361

MAKLUMAT PERHUBUNGAN

ALAMAT: 959, Jalan 17/21k, 46400 Petaling Jaya.

TELEFON: 03-79569813

EXT:

FAX:

EMAIL: jamieong@hotmail.com

CATITAN:

4.7 How to Create New Parts of Inventory, department of Inventory, directory of the Inventory, personal and delete of Inventory.

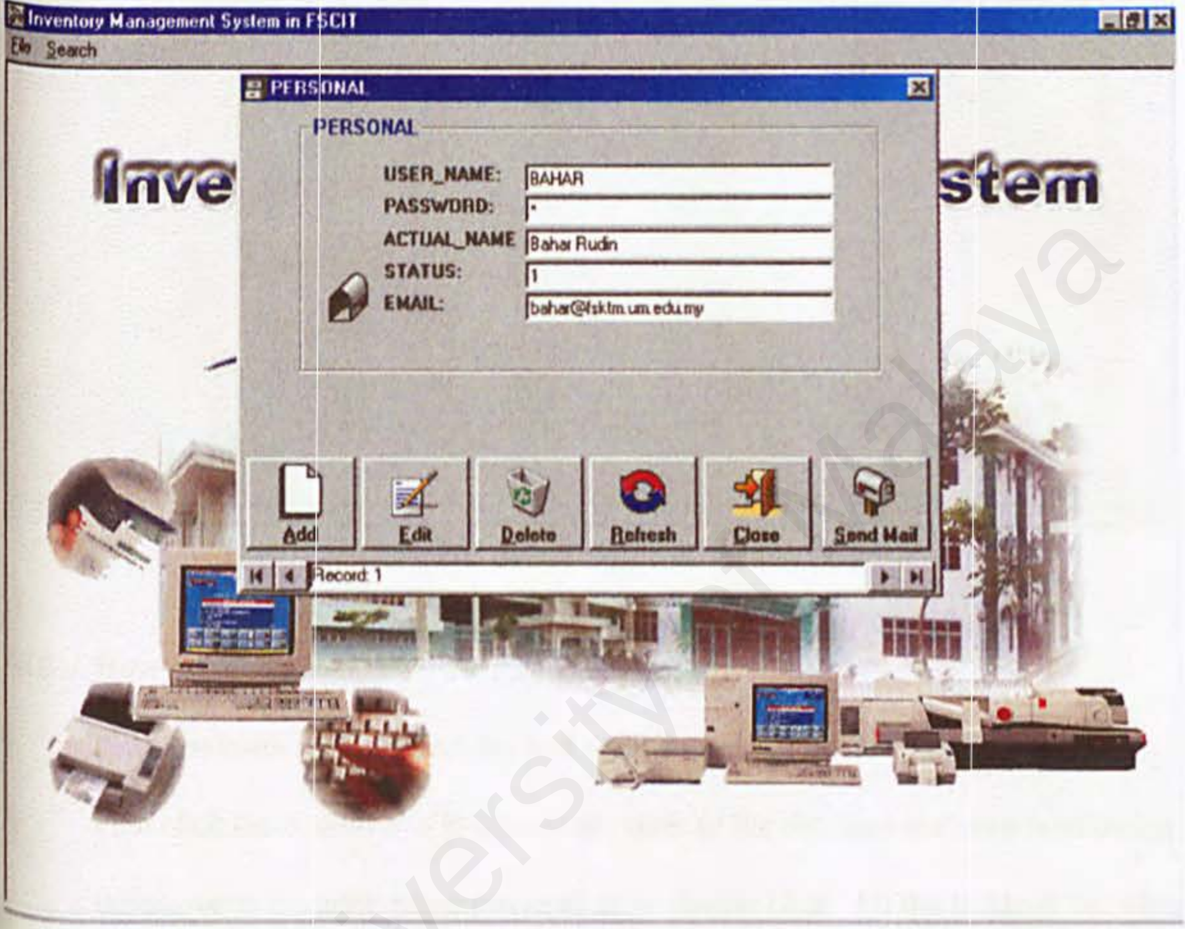
(Refer to the 4.5)

4.8 How to Edit or Delete Parts of Inventory, department of Inventory, directory of the Inventory, personal and delete of Inventory.

(Refer to the 4.6)

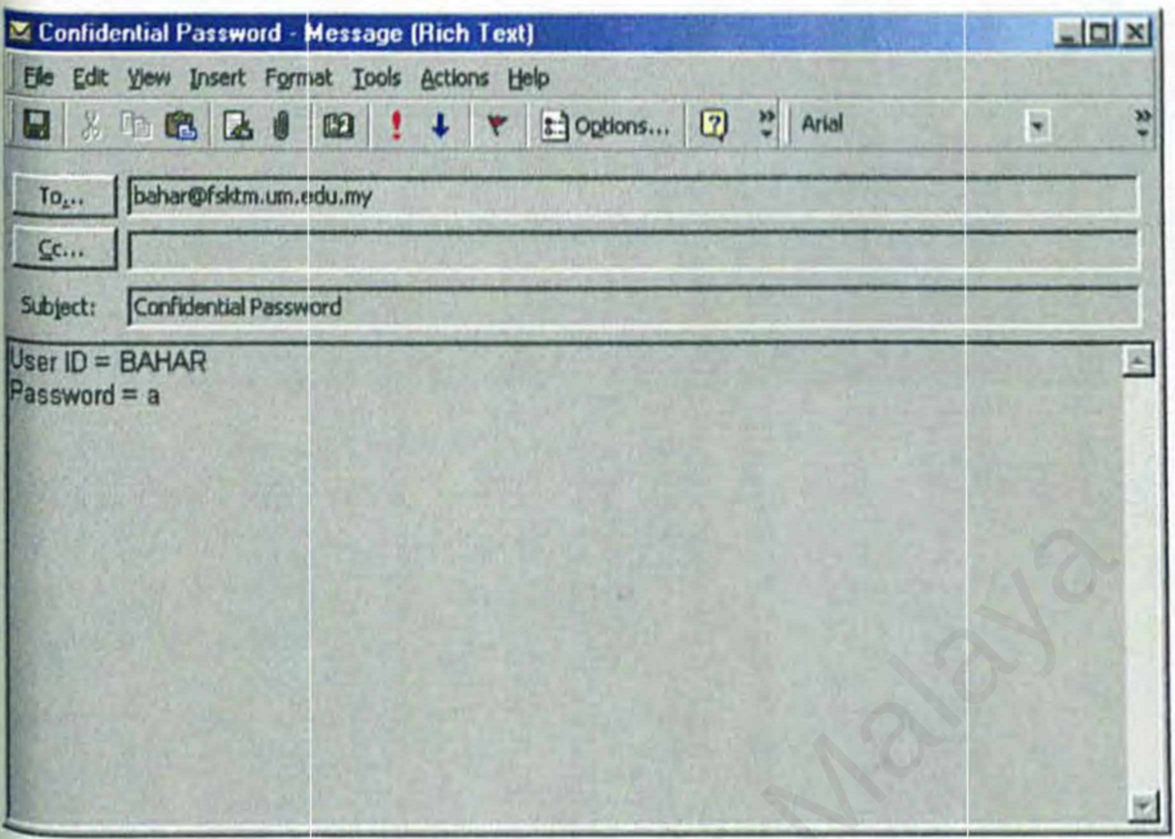
4.9 How to use Send Mail in Personal form

1. From the personal form, just click the Send Mail button and automatic call out outlook application (Office 2000). In the inventory application, it needs to install Office 2000 with the outlook application.



USER_NAME: BAHAR
PASSWORD: *
ACTUAL_NAME Bahar Rudin
STATUS: 1
EMAIL: bahar@fsktm.um.edu.my

2. Purpose of send mail button is let the super administrator to send mail to the users who is wanted to confirm the user ID and password or who is forgot the password.


Confidential Password - Message (Rich Text)
File Edit View Insert Format Tools Actions Help
Options...
Arial
To...
Cc...
Subject:

bahar@fsktm.um.edu.my

Confidential Password

User ID = BAHAR
Password = a

4.10 How to Use the Search

1. From the main menu, Select the tool and choose the search function.
2. First click the combo box to choose the table of the database and then positioning the cursor at the table name you need to be double click. All the fields of the table will be displayed in the field combo box.
3. Positioning the cursor at the field you need and double click.
4. Fill in the containing text you want to find or search, click the Find Now button for finding the information to display in the list view or click the Close button to exit the searching function.
5. Beside that, each of the form in the System module also has each searching function to help the super administrator to find the information related.

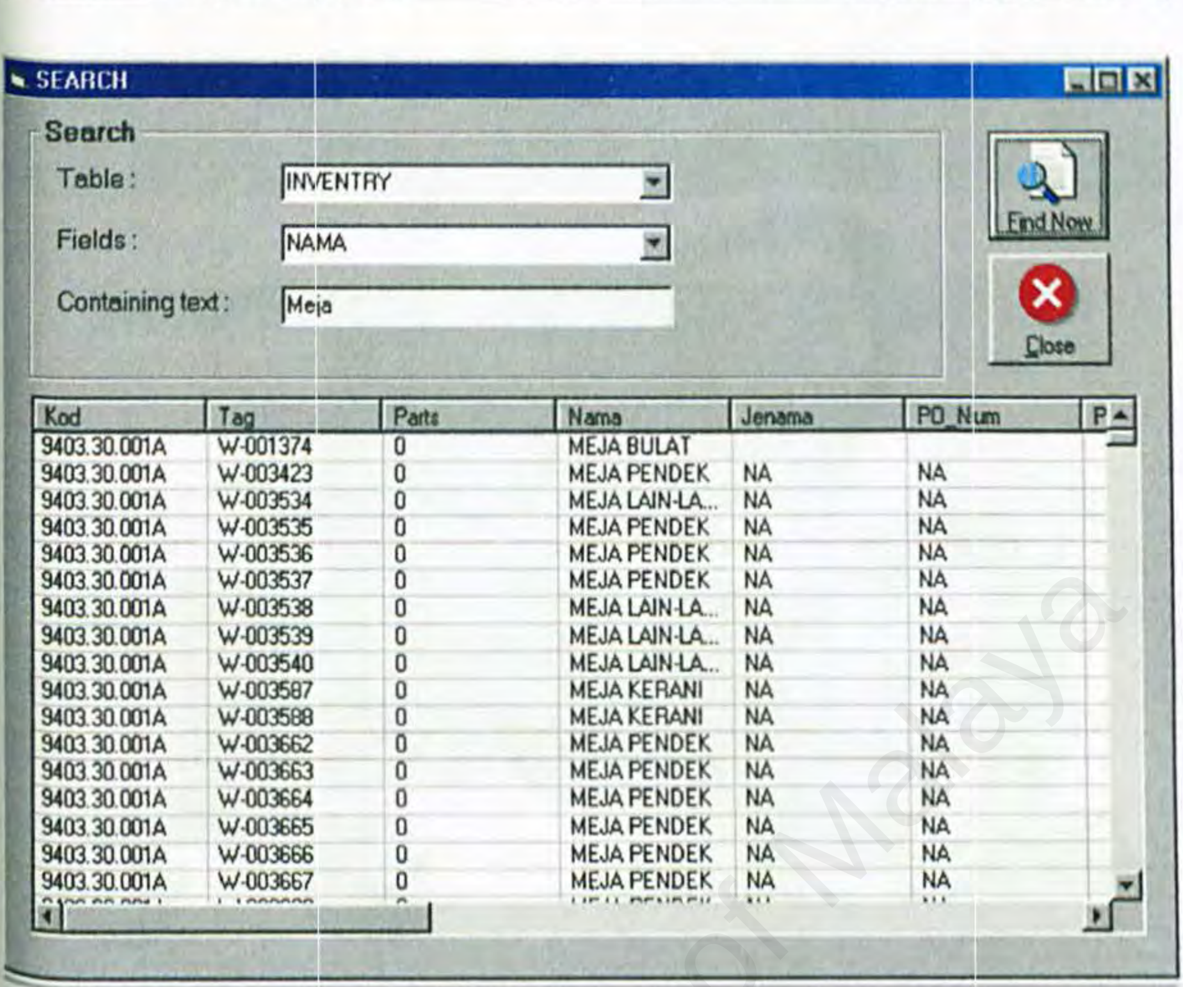


Table : INVENTRY

Fields : NAMA

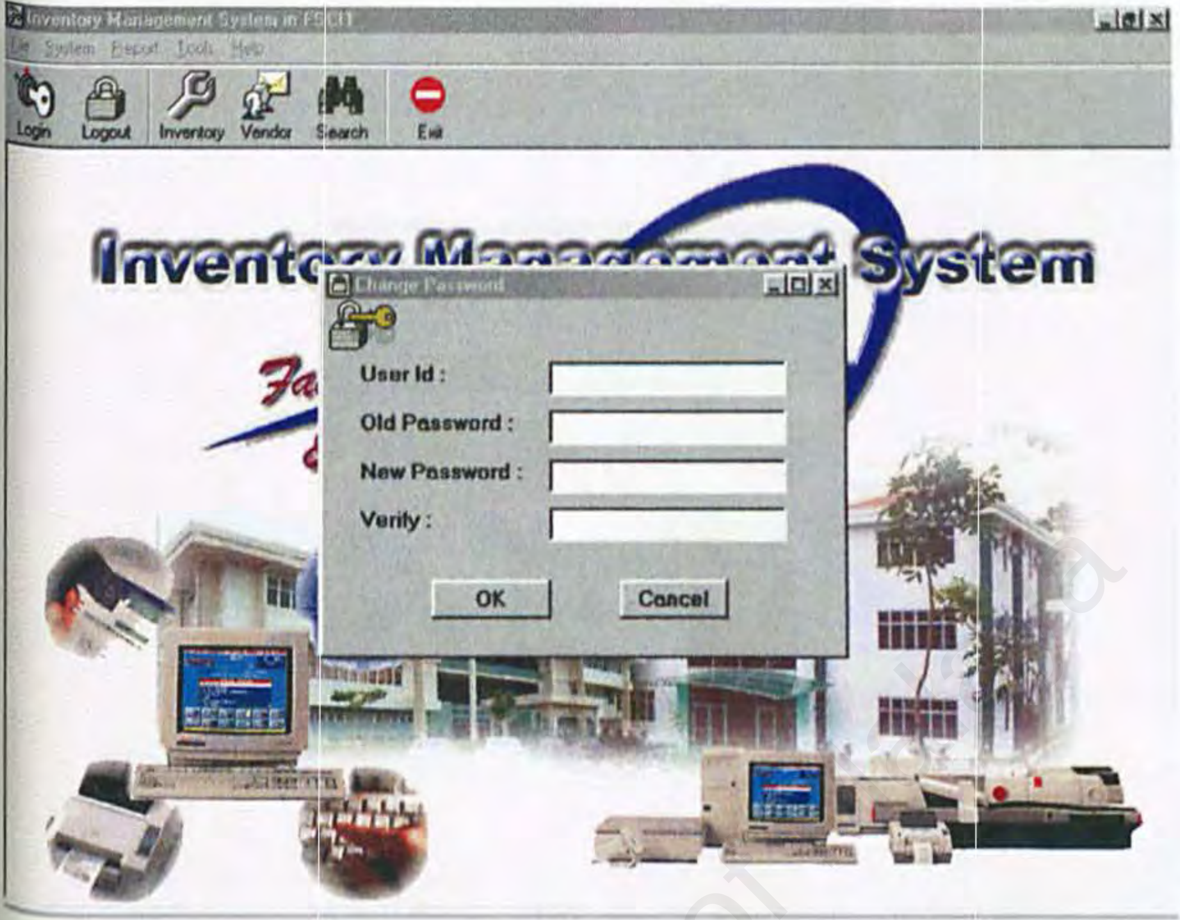
Containing text : Meja

Kod	Tag	Parts	Nama	Jenama	PO_Num	P
9403.30.001A	W-001374	0	MEJA BULAT			
9403.30.001A	W-003423	0	MEJA PENDEK	NA	NA	
9403.30.001A	W-003534	0	MEJA LAIN-LA...	NA	NA	
9403.30.001A	W-003535	0	MEJA PENDEK	NA	NA	
9403.30.001A	W-003536	0	MEJA PENDEK	NA	NA	
9403.30.001A	W-003537	0	MEJA PENDEK	NA	NA	
9403.30.001A	W-003538	0	MEJA LAIN-LA...	NA	NA	
9403.30.001A	W-003539	0	MEJA LAIN-LA...	NA	NA	
9403.30.001A	W-003540	0	MEJA LAIN-LA...	NA	NA	
9403.30.001A	W-003587	0	MEJA KERANI	NA	NA	
9403.30.001A	W-003588	0	MEJA KERANI	NA	NA	
9403.30.001A	W-003662	0	MEJA PENDEK	NA	NA	
9403.30.001A	W-003663	0	MEJA PENDEK	NA	NA	
9403.30.001A	W-003664	0	MEJA PENDEK	NA	NA	
9403.30.001A	W-003665	0	MEJA PENDEK	NA	NA	
9403.30.001A	W-003666	0	MEJA PENDEK	NA	NA	
9403.30.001A	W-003667	0	MEJA PENDEK	NA	NA	

4.11 How to Change password

1. From the main menu, Select the tool and choose the change password function.

2. Fill in the entire text; click the OK button for change the password.



Inventory Management System in FSKTM
File System Report Tools Help
Login
Logout
Inventory
Vendor
Search
Exit
Inventory Management System
Change Password

User Id :

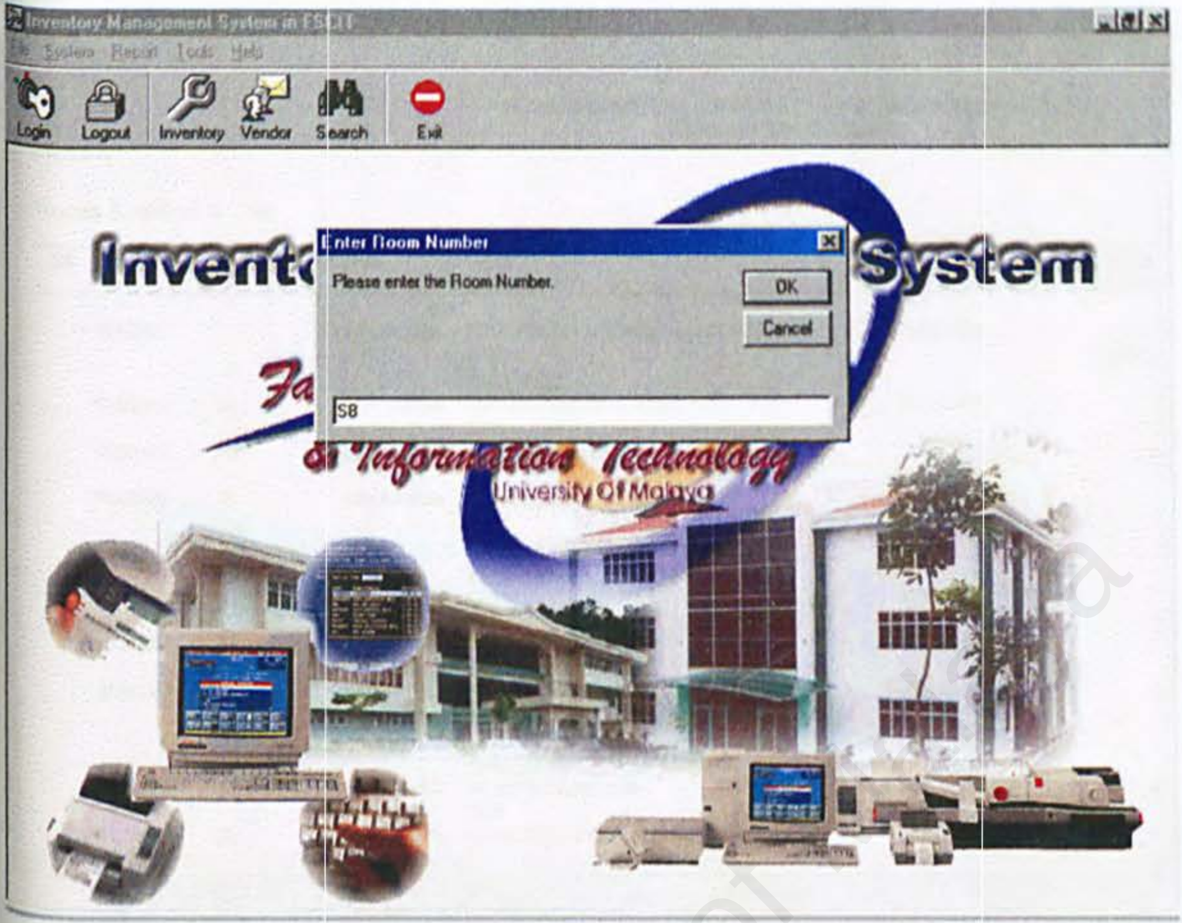
Old Password :

New Password :

Verify :

4.12 How to use the Report Module

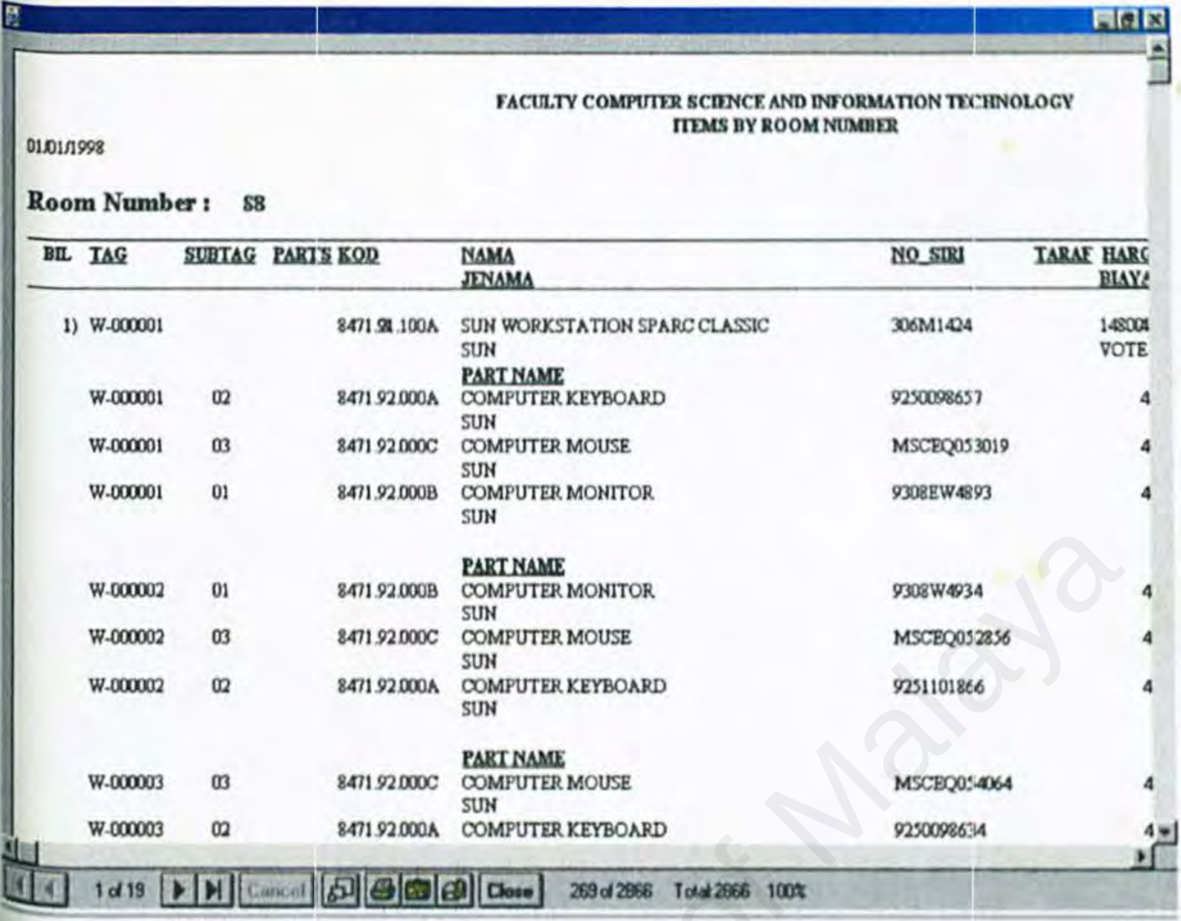
1. From the main menu, select Report module to choose the report from vendor or inventory. Vendor report has base by name and code of the vendor and Inventory report base by selected room number, room number, tag and code. Just click the report you want the report will be displayed.



Please enter the Room Number.

S8

2. When you selected inventory report by the selected room number, the input box will displayed. Just fill in the room number you want, click ok button to display the report of the report base by the room number you wanted or click cancel button will cancel the report.



FACULTY COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
ITEMS BY ROOM NUMBER

01/01/1998

Room Number : S8

BIL	TAG	SUBTAG	PARTS KOD	NAMA JENAMA	NO_SIRI	TARAF	HARC BIAY
1)	W-000001		8471.91.100A	SUN WORKSTATION SPARC CLASSIC SUN	306M1424		14800 VOTE
				PART NAME			
	W-000001	02	8471.92.000A	COMPUTER KEYBOARD SUN	9250098657		4
	W-000001	03	8471.92.000C	COMPUTER MOUSE SUN	MSCEQ053019		4
	W-000001	01	8471.92.000B	COMPUTER MONITOR SUN	9308EW4893		4
				PART NAME			
	W-000002	01	8471.92.000B	COMPUTER MONITOR SUN	9308W4934		4
	W-000002	03	8471.92.000C	COMPUTER MOUSE SUN	MSCEQ052856		4
	W-000002	02	8471.92.000A	COMPUTER KEYBOARD SUN	9251101866		4
				PART NAME			
	W-000003	03	8471.92.000C	COMPUTER MOUSE SUN	MSCEQ054064		4
	W-000003	02	8471.92.000A	COMPUTER KEYBOARD	9250098634		4

3. Click the printer button for printing report. You must adding printer in your PC before print the report.