Software e-Inventory Management System

Name	: Chew Kok Seng
Matrix No.	: WEK000070
Supervisor	: En. Mohd. Nizam Hj. Ayub
Moderator	: Dr. Rosli Salleh
Session	: 2002/2003

A Graduation Project Submitted to the
Faculty of Computer Science & Information Technology
University of Malaya
In partial fulfillment of the requirement for the
Bachelor of Computer Science (Hon)

ABSTRACT

Inventory has been part and parcel of business since its existence. And through generations, many inventory management systems have been developed in order to cope up with the ever-growing business world. With the invention of computer, the business world went into a breakthrough to more organized, efficient and systematic way of handling and integrating all areas of business, thus enhancing the ability of Enterprise Resource Planning. Improvement did not stop, yet it went another peak when web technology and networking grew popular throughout the world. With the emergence of E-commerce, E-business and globalization, the dream of business without boundary has become a reality.

Based on the concepts of E-commerce and Enterprise Resource Planning (ERP) strategy, this project is to build a web-based e-Inventory Management System (Software). The purpose of this project is to build an enhanced enterprise's inventory management system for all the Information Technology and Communication (ICT) Companies in Malaysia. Any user on any platform at any time will be able to retrieve or upload software inventory information through this system. This system will provide the control & management of inventory, ensuring the sufficiency of inventory, accuracy of data keeping and more profitable relationship with supplier.

In order to produce a more efficient and better quality system, this project will be develop through several stages, by implementing the Waterfall Model methodology. The first two stages, which are requirement analysis and system design will be completed in the first semester. While for system coding, system testing, and system operation is completed in the second semester. This methodology approach was chosen because a web-based management system needs a well organized and structured planning of system design and implementation.

Finally, this Inventory system will be integrated with other sub-systems in the enterprise to provide a fully integrate inventory utilities. The other module (besides e-Inventory (Software)) that involved is: e-Inventory Management System (Hardware).

ACKNOWLEDGEMENT

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

1.1 Project General View

The purpose of this project **e-Inventory Management System (Software)** is to develop a self-maintained, user friendly web-based and convenient inventory system, which it's functional, is to manage and control a company's software inventory. It is an approach to ERP system.

Cambridge International Dictionary of English defines inventory as a detailed list of all the items in a place. An inventory is also the amount of goods a shop has, or the value of them.

Through a peer discussion, we found that traditional manual inventory system is a very time consuming job especially manage and keep track with tones of records. But with web based e-Inventory, it could be munching easier and convenient. After e-Inventory is developed, it can be integrated with all the other modules under the same inventory system that we hope to achieve as we are working in a join project of inventory team. Throughout this project, we have decided to use an IT company or software house as a model company to utilize this inventory system.

e-Inventory consists of two different main purposes as listed below:

- Software inventory system(manage and keep track)

- Hardware inventory system(manage and keep track)

1.2 Project Background

The amalgamation of computing and communication technologies and traditional business concerns is impacting all industries and is the latest phase in the ongoing evolution of business. Thus, the number of traditional businesses changing phase into E-Business in Malaysia are increasing hastily. Malaysia local Information Technology and Communication (ICT) companies are the fastest growing market segment in E-Business industry nowadays. They are also the key-enabling factor to the profitable and EDI (Electronic Data Interchange) services in the telecommunication business.

In order to raise the fertility and competitive advantages of their E-Business technology, people start to recognize the value of extending their existing Enterprise Resource Planning (ERP) technology for strategic business arrangement and analysis, while incorporating data from many other key data sources throughout the enterprise.

The E-Business approach of customer-centrism requires placing the entire corporation into an integrated transaction environment. This strategy implies having one common platform instead of many software applications in independent silos. Today's ERP systems, when fully installed as integrated suites, can be thought of as central repositories of internal corporate information derived from five major processes: finance, logistics, manufacturing, human resources, and sales and marketing.

1.3 Project Definition

This project is developed under the concept of Enterprise Resource Planning (ERP). Thus it is important for us to have a glance on the definition of ERP.

ERP (Enterprise resource planning) is an industry term for the broad set of activities supported by multi-module application software that helps a manufacturer or other business manage the important parts of its business, including product planning, parts purchasing, maintaining inventories, interacting with suppliers, providing customer service, and tracking orders. ERP can also include application modules for the finance and human resources aspects of a business. Typically, an ERP system uses or is integrated with a relational database system. The deployment of an ERP system can involve considerable business process analysis, employee retraining, and new work procedures.

ERP could also be described as the spinal cord of the digital nervous system, facilitating the exchange of data through the unification of key processes. It is an advanced core business IT system that integrates data across major functions and activities as well as across organizational and geographical boundaries, providing a platform for increased business performance and competitive advantage.

1.4   Project Motivation

In the past decades, organizations have spent an estimated $40 billion on enterprise resource planning (ERP), according to Gardner Research. By viewing information as a competitive tool, organizations can more proactively respond to ever-changing competitive landscapes while integrating all sources of information about customers, suppliers and partners in a consistent and easy-to-use format.

Today, organization that once viewed enterprise resource planning (ERP) solutions as a necessary implementation to exploit their investments for competitive advantage. They recognize the value of extending their existing ERP technology for tactical business planning and analysis, while incorporating data from many other key data sources throughout the enterprise.

While the majority of these ERP implementations have helped to improve transaction processing streamline operations and standardize corporate IT environments, they have not fully addressed the information needs of these organizations. Thus, we should try to unlock the wealth of information collected and stored within their ERP environments.

Because of the foreseen potential of ERP system, it motivate us to try to develop this project as there is merely few IT firm in Malaysia who has tried it.

As for **e-Inventory,** the need of it is significant because whether the hardware and software you buy are stored in a basement or a climate-controlled warehouse, all e-commerce entrepreneurs will need to keep track of inventory, ensuring that products get to customers on time. The key to success is incorporating software solutions,

developing strong relationships with delivery services, and providing good customer

support throughout the process. With e-Inventory, it is hope that all this functions and

solutions can be achieved successfully.

1.5 Project 0bjective

In this project, our major objective is to build an integrated web based inventory system, which is based on ERP concepts and internet technologies. Although there are other ERP systems available in the market today, but there is only a few of these systems, which originates from Malaysia's own web-technology.

This e-Inventory Management System project is committed to delivering the leading Enterprise and E-Business solution purposed for Malaysia Information Technology and Communication (ICT) companies. This is achieved by expanding the line of business application capabilities for its existing customer base to meet the growing demand for an integrated enterprise resource planning (ERP) solution. The entire solution is based on proven and popular Microsoft technologies and platforms to provide an enterprise application framework.

1.5.1 Objective of e-Inventory Management System

As I am developing the e-Inventory (software) Project, the objectives I aim to achieve for this specific module are as following:

- To enable the control and management of software inventory to be done easily via web technology.

- To ensure the sufficiency of inventory and data integrity.

- To improve the quality and accuracy in data keeping.

- To provide online help desk. (Documentation and follow up)

1.6 Project Scope

e-Inventory, by definition, is an electronic based inventory management system online. Its purpose is to allow managing and controlling of inventory being done via web-based technology. As we are looking at the point of system efficiency and effectiveness, we expected to provide customers with *"real-time visibility" strategy*, where inventory levels-supplying for the internal control and management needs no sooner or later than the records are needed for updating or reordering.

The e-Inventory can be distributed into back-end and front-end sections, where the inventory department's staff and managerial board (administrators) use the back-end section; while the front-end section is to be used by others authorize module in e-Inventory. However, there will be one extra sub-module, which is the privilege for e-Inventory registered supplier. Generally in e-Inventory, there will be a design of several sub-modules, which will cover the functions needed under inventory management such as the following:

1) Login Authentication

2) User account management (Add, Edit, Delete)

3) Inventory Management/Maintenance (Add, Edit, Delete)

4) Information Retrieving and Sorting

5) Software Problems Support and Troubleshooting

6) Generating Summary Report

1.7 Project Limitation

The limitations of this project are shown as below:

- This project will not support bar coding for the stock code/ product code, as we do not have bar code reader support.

- SQL server, web server and Windows 2000 server will perform into a same computer.

- User does not have the availability to customize the layout of web page.

- Delivery functions are not included.

- Currency conversion is not included in this project.

- This project only supports single communication language, which is English.

- The annual physical stock counting for inventory is not included in this project because the model company only exists virtually. However, there will be a function for generating inventory summary report of stock information.

1.8 Project Requirement

Below are minimum list of facilities that will be needed or essential for the project

development:

Hardware:

Computers with at least 64Mb Ram: one running as server and another as workstation.

Processor: Pentium® III 500MHz or equivalent

VGA: 640 X 480(256 color)

Software:

- Windows 2000 server
- Internet Information Server 4.0
- Ms SQL server 7.0
- Internet Explorer 6.0

1.9 Project Schedule

Project Scheduling is used to plan and control a project efficiently and can determine:

1. The minimal expected completion time for a project.

2. The delay of project activities.

3. The earliest and latest time each activity can be started and completed.

4. The amount of slack for each activity.

5. Whether or not a current project is on schedule or is being completed within budget.

In this project, I have chosen to use Gantt chart to determine a clear timeline between the starting date and the finishing date. An important advantage of this chart over other time-charting technique is its simplicity. By using Gantt chart, a schedule of earliest possible start and finish times for the activities is given that will meet the earliest possible project completion date. There are six major phases in this project: Literature Review, System Analysis, System Design, System Coding, System Testing, and Documentation.

The schedule of this project is as shown in the Gantt chart below:

The Project Report Layout

...to give an overview of the major phases in the
development of the project. The purpose of this report is to document all the essential
information gathered and used to develop this system.

Chapter 1: Introduction

This chapter serves as an introduction to the entire project. It made an overview of the
project background, project objectives, project motivation, problem scope, project
limitation, project requirements and the project schedule.

Chapter 2: Literature Review

This chapter covers all the literature survey done on this project, including reviews on
the features, capabilities, system architecture, and system designing tools and so on.

Chapter 3: Methodology

This chapter fairly discusses the development methodology, the functional and non-
functional requirements, and also the tools and technology for this project.

Chapter 4: System Design

This chapter describes the design considerations including processing design, the user
interface design and also the database design of this project.

Chapter 5: System Implementation

This chapter focuses on how the system being implementing successfully, and also

ID	❶	Task Name	Jun '02	Jul '02	Aug '02	Sep '02	Oct '02	Nov '02	Dec '02	Jan '03
1	⊞	Literature Review								
2	⊞	System Analysis								
3	⊞	System Design								
4	⊞	System Coding								
5	⊞	System Testing								
6	⊞	Documentation								

Legend:
Task | Project Summary
Split | External Tasks
Progress | External Milestone ◆
Milestone ◆ | Deadline
Summary

Project: Schedule
Date: Mon 24/6/02

Page 1

Figure 1.1 Gantt chart for Project Scheduling

11

1.10 Project Report Layout

The following layout is to give an overview of the major phases involved during the development of the project. The purpose of this report is to document all the essential information gathered and used to develop this system.

Chapter 1: Introduction

This chapter serves as an introduction to the entire project. It made an overview on the project background, project objectives, project motivation, project scope, project limitation, project requirements and the project schedule.

Chapter 2: Literature Review

This chapter covers all the literature survey done on this project, including reviews on the features, capabilities, system architecture, and system designing tools and so on.

Chapter 3: Methodology

This chapter fairly discusses the development methodology, the functional and non-functional requirements, and also the tools and technology for this project.

Chapter 4: System Design

This chapter describes the design considerations including processing design, the user interface design and also the database design of this project.

Chapter 5: System Implementation

This chapter focuses on how the system being implementing successfully, and also

shows example of program coding.

Chapter 6: System Testing

This chapter is about the method of system testing.

Chapter 7: System Evaluation & Conclusion

This chapter discusses about the system strengths, limitations, future enhancements, and also the conclusion of the whole project.

1.11 Summary

This chapter focuses primarily on the introduction of this project.

The first part is the brief introduction about the project background and its definition. Besides, the project motivation was also stated. Then the project objectives for both the e-Inventory were discussed and determined.

Secondly, the scope of this project was shown in section 1.6, where all the system features and functionality components were stated there. Then, section 1.7 shows the project limitations of those functions which most likely unable to achieve in this project; while section 1.8 show the project development requirements (hardware and software).

Finally, there is a project schedule shown by the Gantt chart *(figure 1.1)* in section 1.9 to tell the project's timeline of development; and the project report layout was shown in section 1.10.

Chapter 2: Literature Review

What is Literature Review?

A review of the literature is an essential part of academic research project. The review is a careful examination of a body of literature pointing toward the answer to research topic.

A literature or a body of literature is a collection of published research relevant to a research question. All good research and writing are guided by a review of the relevant literature. In this case, the research topic is electronic inventory.

Literature review will be the mechanism by which research is viewed as a cumulative process. That makes it an integral component of the scientific process. [1]

By another world, a literature review summarizes, interprets, and evaluates existing literature (or published material) in order to establish current knowledge of a subject. The purpose for doing so relates to ongoing research to develop that knowledge: the literature review may resolve a controversy, establish the need for additional research, and/ or define a topic of inquiry.

Purpose of Literature Review

The purpose of the literature review remains the same regardless of the research methodology that use. Is an essential test of the research question against that which is already known about that subject. Though the literature review it will discover whether someone else already has answered certain research question. [28]

How to write a literature review?

If a researcher find that his/her research question has not been answered satisfactorily by someone else, then search out the answers to these questions: [28]

- What is known about the proposed system?
- What is chronology of the development of knowledge about this system?
- Are they any gaps in knowledge of this system? Which opening for research has been identified by other researchers? How does a researcher intend to bridge the gaps?
- Is there a consensus on relevant issues? Or is there significant debate on issues? What are the various positions?
- What is the most fruitful direction that researcher can see for his/her research as a result of the literature review? What directions are indicated by the work of other researchers?

Technicalities of a literature review

Literature review will have two components:

- The search through printed resources and electronic resources.
- The writing of the review

2.1 The Internet

2.1.1 Definition

Internet is a global network of networks enabling computers of all kinds to directly and transparently communicate and share services throughout much of the world. Because the Internet is an enormously valuable, enabling capability for so many people and organizations, it also constitutes a shared global resource of information, knowledge, and means of collaboration, and cooperation among countless diverse communities. On October 24, 1995, the FNC unanimously passed a resolution defining the term Internet, This definition was developed in consultation with members of the Internet and intellectual property rights communities.

RESOLUTION: The Federal Networking Council (FNC) agrees that the following language reflects our definition of the term "Internet". [1]

"Internet" refers to the global information system that:-

- *is logically linked together by a globally unique address space based on the Internet Protocol (IP) or its subsequent extensions/follow-on;*

- *is able to support communications using the Transmission Control Protocol/Internet Protocol (TCP/IP) suite or its subsequent extensions / follow-on, and/or other IP-compatible protocols; and*

- *Provides, uses or makes accessible, either publicly or privately, high level services layered on the communications and related infrastructure described herein.*

2.1.2 Growth of the Internet

The Internet and World Wide Web surely will be listed among the most important and profound creation of humankind. The Internet has changed much in the two decades since it came into existence. It was conceived in the era of time-sharing, but has survived into the era of personal computers, client-server and peer-to-peer computing, and the network computer. It was designed before LANs existed, but has accommodated that new network technology, as well as the more recent ATM and frame switched services. It was envisioned as supporting a range of functions from file sharing and remote login to resource sharing and collaboration, and has spawned electronic mail and more recently the World Wide Web. But most important, it started as the creation of a small band of dedicated researchers, and has grown to be a commercial success with billions of dollars of annual investment. [1]

The Internet has not finished changing, it continues to change and evolve at the speed of the computer industry to remain updated and relevant. It is now changing to provide new services such as real time transport, in order to support, for example, audio and video streams. The availability of networking (i.e., the Internet) along with powerful affordable computing and communications in portable form (i.e., laptop computers, two-way pagers, PDAs, cellular phones) provides the ability of having nomadic computer and communication services.

In less than 30 years, the Internet has become one of the most amazing technological and social accomplishments of the century. Millions of people are using a complex, interconnected network of computers. The computers are located in almost every country of the world. The opening of the Internet to business activity helped increase the Internet's growth dramatically. The following shows the amazing growth of

the Internet during the past decade. [2]

Growth in the number of Internet hosts 1991-2000
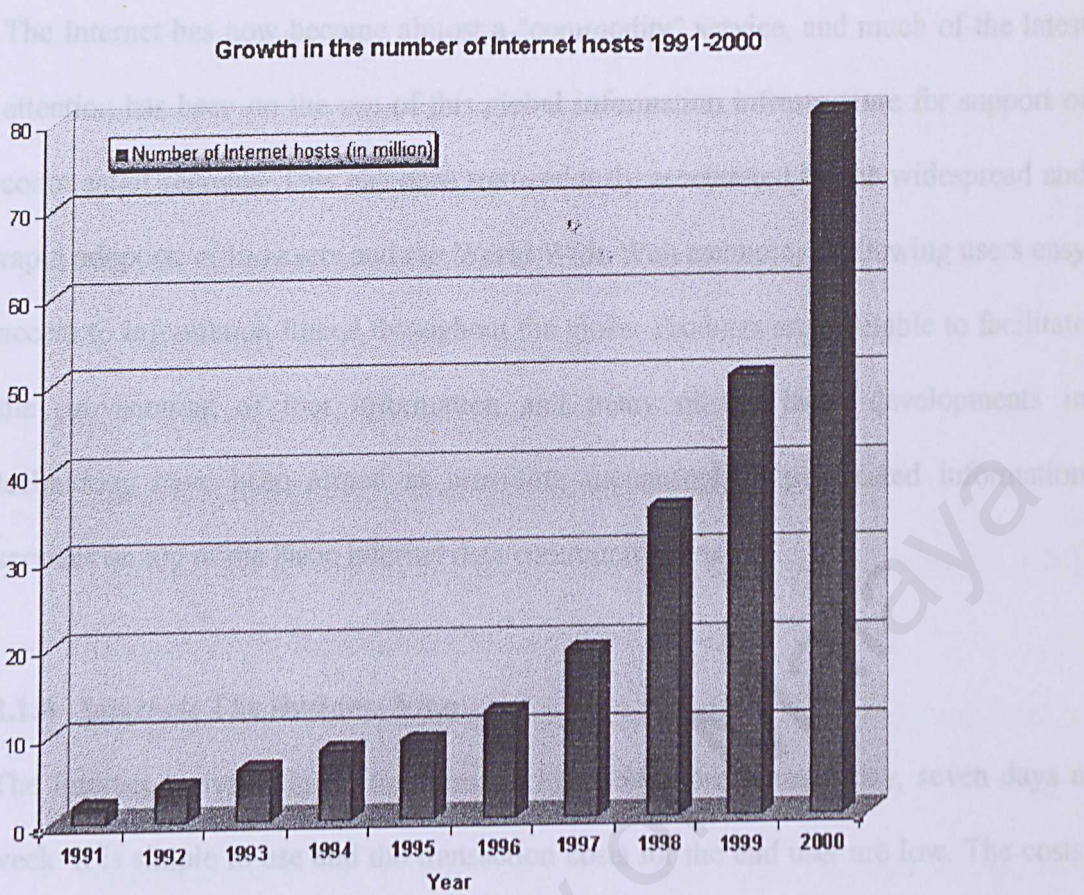
Figure 2.1 Growth of the Internet [2]

2.1.3 Commercialization of the Internet

Commercialization of the Internet involved not only the development of competitive, private network services, but also the development of commercial products implementing the Internet technology. In the early 1980s, dozens of vendors were incorporating TCP/IP into their products because they saw buyers for that approach to networking. Unfortunately they lacked both real information about how the technology was supposed to work and how the customers planned on using this approach to networking. But through years of research efforts, studies and implementations, we are able to find a better way to fully utilize the facilities of

Internet in commercial purpose.

The Internet has now become almost a "commodity" service, and much of the latest attention has been on the use of this global information infrastructure for support of commercial services. This has been tremendously accelerated by the widespread and rapid adoption of browsers and the World Wide Web technology, allowing users easy access to information linked throughout the globe. Products are available to facilitate the provisioning of that information and many of the latest developments in technology have been aimed at providing increasingly sophisticated information services on top of the basic Internet data communications

2.1.4 Internet: The Business View

The Internet is available all over the world, twenty-four hours a day, seven days a week. It is simple to use and the transaction costs for the end user are low. The costs are also extremely low for the vendors on the Internet, compared to traditional distribution channels. The Internet allows two-way communications and is built around open standards. The two-way communication allows for direct feedback of customers and open standards mean interoperability between companies, web sites and services. Once they have been digitized, it is easy to integrate processes, services and products.

2.1.4 (a) Why Should Businesses GO Online

- Expanding Market - business is opened to market of several million people from all walks of life globally.

- Visibility - allows a company to present its products and services at very low cost.

20

- Responsiveness - increase responsiveness to customers and partners.

- Introducing New Services - New services can be provided to customers and partners by publish a trial version on the web. Besides, a service for the employees can also be provided in self-service form

- Rapid Information Access - accessing of information over the Internet is much faster transmission and transfer via fax or postal courier services.

2.1.4 (b) The Fear for Business Go Online

- Change - Management and staff may resist change.

- Customer Acceptance - Many companies are afraid that their customers won't accept the new channel.

- Competition - As barriers to entry fall, competition is changing from local to global.

- Interdependence: Previously guarded processes now happen outside the company, leading to a shift in the balance of power.

- Intellectual Property - Once information has been published on the Internet, it becomes easy for others to copy and use it.

- Loyalty - The Internet is less personal, so people are not bound to a certain supplier or merchant.

- Pricing - The New Economy makes it easier to compare prices. Prices will drop, so quality and add-on services become more important. Issues of different pricing strategies and multiple currencies need to be addressed.

- Security - Most companies are very concerned about security on the Internet.

After analyzing the existing cases of business going online, the Internet's potential for business purpose is no doubt great and undeniable. Therefore, upon weighing the pros and cons of it, it is obviously worth while a try to let business go online, for expanding business and strengthening future prospects.

2.2 E-Business

2.2.1 Definition

Electronic business encompasses three stages, which are e-commerce, e-business, and e-partnering. The early stages of a company's e-business activity are almost always focusing on reaching the customer, the later stages on streamlining value-chain activities to deliver more value to the customer. [3]

E-Business is the complex fusion of business processes, enterprise applications and organizational structure necessary to create a high performance business model. It refers to the use of digital technologies to transform both the internal processes as well as an organization's interactions with external parties. In other words, it combine the resources of traditional information systems with the vast reach of the Web and connect critical business systems directly to critical business constituencies - customers, employees, partners and suppliers using Intranets, Extranets and the World Wide Web.

E-Business, the Internet and the globalization all depend on each other. The more global players exist, the more E-Business they want to do. The more E-Business is online, the more people will be attracted to get direct Internet access. And the more people are online, the more global players will arise.

2.2.2 Three Main Area of E-Business

1. It can be within the organization using the so-called Intranet. The Intranet uses Internet standards for electronic communication. People on the Intranet are able to see organization-specific web sites. These web sites are separated from the rest of

the world by firewalls and other security measures. People from outside of the organization are not able to see these private pieces of information.

2. The second area is the business-to-business (B2B) deals that are done over the Extranet. The Extranet consists of two Intranets connected via the Internet, whereby two organizations are allowed to see confidential data of the other. Normally only small parts of information are made available to the partner, just enough to enable the business. Business-to-business networks have existed long before the Internet. Many organizations have had private networks to talk to their partners and customers. But maintaining them was very expensive. Through the usage of the Internet the costs have been cut dramatically. In order to keep the business transactions private virtual private networks (VPNs) are used in most cases.

3. Thirdly there is the business-to-consumer (B2C) area. Traditionally this is what most people know as e-commerce, selling products on the web. Also to keep in mind is the consumer-to-consumer area, also known as peer-to-peer commerce, such as auction sites or digital exchanges like Napster.

The concept of electronic business had been invented before the Internet became popular. In the 1970's E-Business was already popular for financial networks, for example, which used propriety hardware and software solutions. Electronic Data Interchange (EDI) was also available long before the Internet was used for it. But without the Internet E-Business would not have been possible on such a large scale. The private networks, which were used in the 1970's and 1980's, cost too much for smaller enterprises and were not accessible for public use.

E-business improves business performance by using electronic information technologies and open standards to connect suppliers and customer at all steps along the value chain. It significantly improves business performance by strengthening the linkages in the value chain between businesses and the ultimate consumer. E-business focuses on effectiveness through improved customer service, and reduced costs.

2.3 E-Commerce

2.3.1 Definition

In general. Electronic Commerce is the exchange of goods or services - buying and selling through the Internet. It is modern business methodologies that address the needs of organizations, merchants, and consumers to cuts cost while improving the quality of goods and services and increasing the speed of service delivery. [4]

E-Commerce can also be defined as the use of electronic data transmission to implement or enhance any business process. Frequently people use this term to refer to commerce on the Internet or Web because they are most extensive data transmission networks. [2]

2.3.2 Areas of e-commerce application

E-commerce includes transactions within a global information economy. In almost any business function today, e-commerce is at work for some companies. Here are some of the areas in which businesses are applying e-commerce:

- **Corporate Purchasing**
  - E-commerce allows supply chain management to link inventory, billing and shipping between suppliers and customers to ensure a smooth coordination of supply and demand.

- **Marketing and Promotions**
  E-commerce allows an online channel to run various marketing and promotion activities. Generally, marketing uses e-commerce to:
  - Attract new customers through marketing

-     o   Provide customer service and support for existing customers

-     o   Develop new markets and distribution channels for existing products

-     o   Coordinate sales and marketing efforts

- **Brand Management and Awareness**

  The Web enables organizations to get the word out faster and to larger audiences

  than ever before.

- **E-Mail Advertising**

  In business-to-business e-commerce, e-mail advertising can be used to increase

  brand awareness and to introduce new products and services to business partners.

- **Inventory Management Between Branches**

  Ordering, storing, and tracking inventory over a large geographical area presents a

  major challenge that can be answered by electronic commerce. By using either

  decentralized or distributed inventory databases, organizations and their business

  partners can use e-commerce to access those databases around the world, at any

  facility at any time. Inventory management also helps ensure that inventory will

  arrive when needed. This means that suppliers can also improve their own internal

  efficiency by knowing inventory requirements in advance.

- **Supply Chain Management**

  Every member of the supply chain needed accurate and current information about

  orders, shipping and the response of customers. E-commerce helps solve this

  problem by automating the information system so that it can be used by the

  members of the supply chain.

- **Customer Relationship Management**

  Electronic commerce increased speed and accuracy of information sharing between organizations and their customers; improved relationships with customers; and allow faster response to customer orders, requests, and problems, which ultimately helps increase customer satisfaction.

- **Help Desks**

  Online help desks allow customers to check the status of orders, have questions answered and access an information database that can help them solve problems.

2.3.3 E-commerce Issues

For companies that wish to join the realm of e-commerce, one must first make consideration and studies on the following:

- o The Electronic Commerce Standards
- o The Electronic Commerce Architecture
- o The Electronic Commerce Security Issues

- Electronic Commerce Standards

  There are a number of standards associated with e-commerce. Such as: Bank Internet Payment System (BDPS), Joint Electronic Payments Initiative (JEPI), Open Profile Standard (OPS), Open Buying On the Internet (OBI), and Open Trading Protocol (OTP) etc. All these have been developed targeted specifically to support commerce on the Web. For example, OBI was created to support Business-To-Business (B2B) commerce where it manages the exchange of

pricing, tax, and authorization data. It also leverages electronic data interchange technology, SSL, and digital signatures for security. [4]

- Electronic Commerce Architecture

  System architecture is important in developing e-commerce applications. Because e-commerce is not a specific product but actually a business solution, therefore it must include several components and features for the maximum effectiveness. The characteristic of the system architecture should include the following feature:

  - o Flexibility and scalability - The e-commerce design and infrastructure must be able to adjust and expand. The e-commerce software must be able to interpret different data formats and transform these formats into a universal format.
  - o Security Authentication, encryption and restricted access must be anticipated carefully in the applications.
  - o Web support - Software modules in an e-commerce system must support Web capabilities.
  - o Data storage - Data must be regularly saved in archive storage for appropriate transaction management.
  - o User Support - E-commerce design and operations personnel must continuously support users.

- Electronic Commerce Security Issues

  Electronic commerce security is extremely important. Any commerce security policy has to address secrecy, integrity, necessity, and intellectual property rights. Threats to E-commerce can occur anywhere in the commerce chain,

beginning with a client computer and ending with the commerce and back office servers. Thus, effective security policies along with adequate detection and enforcement are the only ways to safeguard electronic communications and commerce transactions. [2]

Secure Socket Layer (SSL) encryption and Secure Electronic Transaction (SET) encrypted authentication are among the most common examples of e-commerce security measurement recently. SSL provides a security handshake in which the client and server computers exchange their information through authentication and data encryption technology. SSL resides on top of the TCP/IP layer of the Internet protocol suite, thus it can secure many different types of communications between computers. SET is a method for providing secure credit card transaction on the Internet; the SET protocol is designed to allow consumers, merchant, and banking software companies to independently develop software for their respective clients and to have them inter-operate successfully.

Besides all these protocols and software, there is also service such as CERT - the Computer Response Team, which was formed to address security outbreaks by linking key scientific experts. When large security outbreaks occur, the member scientists can converge and discuss method to locate and eliminate the electronic attacker

2.4 ERP: Enterprise Resource Planning

2.4.1 Definition

Enterprise Resource Planning, also known as ERP, is a structured approach to optimizing a company's internal value chain. It is an enabling technology, a set of integrated software modules that make up the core engine of internal transaction processing. [3]

The software connects the components of the enterprise through a logical transmission, across major functions and activities as well as across organizational and geographical boundaries, providing a platform for increased business performance and competitive advantage.

What ERP really does is organize, codify, and standardize an enterprise's business processes and data. The software transforms transactional data into useful information and collates the data so that it can be analyzed. In this way, all of the collected transactional data becomes information that companies can use to support business decisions. For example: when data such as a sale becomes available at one point in the business, it courses its way through the software, which automatically calculates the effects of the transaction on other areas, such as manufacturing, inventory, procurement, invoicing, and booking the actual sale to the financial ledger.

2.4.2 ERP background

The background of ERP is in the traditional inventory control that dictated the style of software packages in the 1960s. In the 1970's, production companies implemented materials requirements planning (MRP) systems to rationalize and streamline

production.

In the 1980's these information systems matured into manufacturing resource-planning (MRPII) systems, which embraced other functions such as processing, manufacturing, and distribution. The integration it embodied naturally made it attractive to other functions, and so it gradually spread its tentacles to finance, human re-sources, and project management. During the 1990s, the goal of ERP has been to integrate core business processes. An order shipped should automatically generate an invoice and decrement the finished goods inventory system. Waste and redundancy would be eliminated. The ERP system would provide an end-to-end window for mission-critical operations, enabling workgroups and departments to view the ramifications of their activities for the enterprise as a whole. [5]

2.4.3   Between ERP and E-Business

ERP, as what lies behind the web page (a back end unit) is necessary to fulfill the promises made on the web page, which are the promises made by e-business, the front end unit (web site). ERP is focused on internal process efficiency and effectiveness. E-business is focused on external, cross-enterprise process efficiency and effectiveness and on product promotion. While ERP technology supports current business strategy, e-business opens the doors to new strategic opportunities.

Except for a very few companies that do nothing but aggregate information and match buyers and sellers, all companies need some sort of internal transaction engine to match internal information flow with the actual flow of goods or services, as well as money. Every company has to perform their business information tasks, and all of these are facilitated by ERP. The Web based technology (e-business) comprises

the 'electronic' portions of the electronic enterprise, but the ERP comprises the 'enterprise' portion. Also, each company needs to decide on its own strategy, in terms of both ERP implementation and e-business. That strategy will be driven by customer demands, competitive pressure, and the current state of enterprise (its technical capabilities). [3]

2.4.4 Complementary Technologies of ERP and E-business

ERP is the internal technological hub of a single enterprise. Web based technology extends each enterprise's internal information infrastructure into external environment, representing the company brand and projecting it to the marketplace. To marry ERP and web based technology successfully and push each to achieve its maximum benefit, the providers of each must understand the benefits that each provides to the other. As illustrated in figure 2.4.4, web based technology provides connections via the Internet to a host of external parties. ERP software helps organizations effectively and efficiently manages all their internal information resources to meet overall goals. Stand-alone e-business software is available for each of theses relationships, usually from providers who focus on building the best applications to handle any given process.

2.4.5 ERP Functions

An integrated ERP system is the hub of an enterprise and is used to support existing business strategies. This strategy implies having one common platform instead of many software applications in independent silos. ERP provides a company the flexibility required to improve customer responsiveness (the demand side) and to better manage production needs and inventory (the supply side). With ERP a company can create a new information foundation by replacing many legacy systems of varying vintages that house data in different ways. ERP provides a consistency of information across a global enterprise and integrates the following: [5] & [3]

- **Back-office operations:** First to be covered by ERP system is Resource Planning which provides function such as forecasting and planning, purchasing and material management, warehouse and distribution management, product distribution, and accounting and finance. All these disparate functions are being integrated into one package.

- **Supply Chain Management (SCM):** This function enables companies to optimize the entire logistics, production, and distribution process, from acquiring raw materials from suppliers to scheduling production and shipping products to customers on time. This reduce cycle time and inventory levels, also improves a company's cash position.

- **Sales force automation (SFA):** SFA adds functionality to aspects of the sales process such as contact management, sales forecasting, and order management.

- **Customer relationship management (CRM):** CRM connects the various

experiences a customer has with a company-marketing, product selection, purchasing, receiving, complaints and post-sale support-into a managed relationship, where a company is able to identify which customers it should focus its limited resources on and how to do so effectively.

- **Knowledge Management:** includes creating a data warehouse, a central repository for the enterprise's data; performing business analysis on this data; providing decision support for enterprise leadership; and creating future customer-based strategies.

In a fully integrated ERP system, all the activities stated above are accomplished by utilizing five tightly integrated modules: finance, manufacturing, logistics, sales and marketing, and human resources. These modules comprising ERP operate specific business processes within functional areas. These enable companies to plan and manage resources on a real-time basis. When changes are entered into one module of the system, other related data elements and modules are automatically updated. The underlying integration across various modules provides operational transparency, which allows managers to follow what's happening across the business, even on a global basis.

2.4.6 ERP Architecture

ERP applications are most commonly deployed in a distributed and often widely dispersed manner. While the servers may be centralized, the clients are usually spread to multiple locations throughout the enterprise. Generally speaking, there are three functional areas of responsibility that are distributed among the servers and the clients. First, there is the database component - the central repository for all of the data that is transferred to and from the clients. Then, of course, there are the

clients; here raw data gets entered, re-quests for information are submitted, and the data satisfying these requests is presented. Lastly, we have the application component that acts as the intermediary between the client and the database. Where these components physically reside and how the processes get distributed will vary somewhat from one implementation to the next. The two most commonly-implemented architectures are outlined below. [6]

- Two-tier Implementations

In typical two-tier architecture, the server handles both application and database duties. The clients are responsible for presenting the data and passing user input back to the server. While there may be multiple servers and the clients may be distributed across several types of local and wide area links, this distribution of processing responsibilities remains the same. Figure 2.2 below provides a simple illustration of a two-tier implementation.
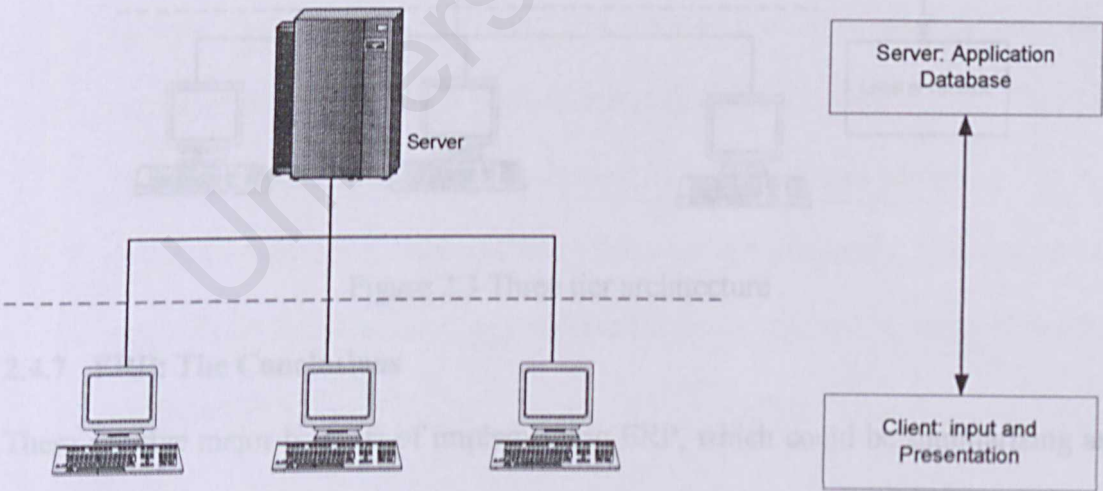
Figure 2.2 Two Tier architecture

- Three-tier Client/Server Implementations

In three-tier architectures, the database and application functions are separated. This is very typical of large production ERP deployments. In this scenario, satisfying client requests requires two or more network connections. Initially, the client establishes communications with the application server. The application server then creates a second connection to the database server. Figure 2.3 below illustrates this type of implementation.
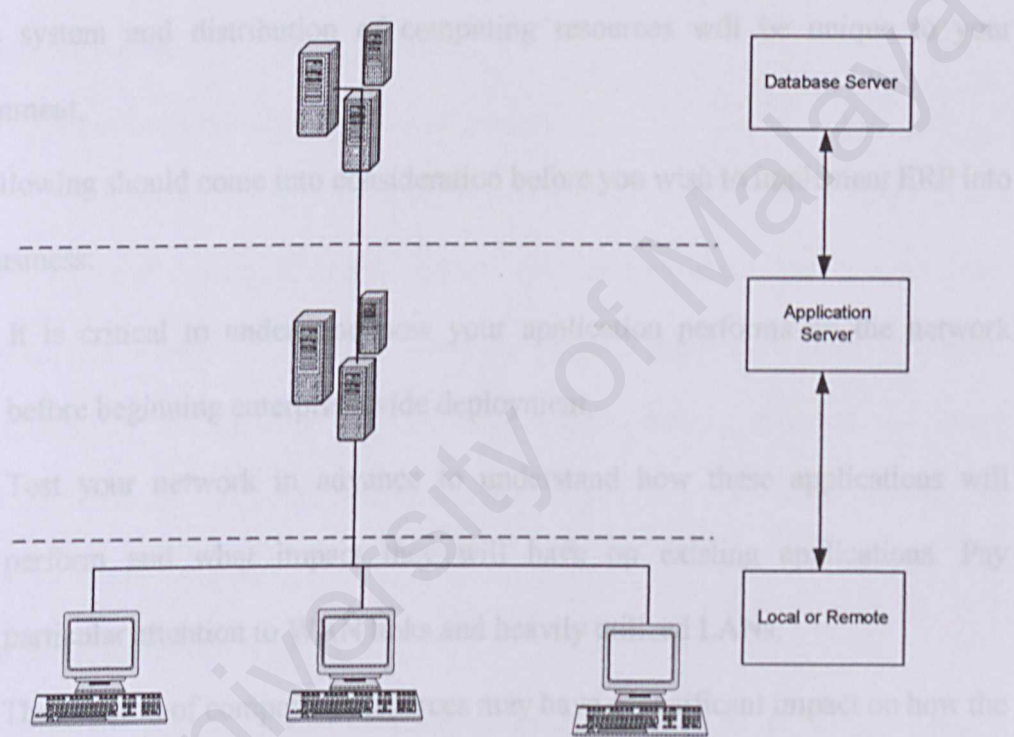
Figure 2.3 Three tier architecture

2.4.7 ERP: The Conclusions

There are five major benefits of implementing ERP, which could be summarizing as the following:

- o on-line/real time information throughout all the functional areas of an organization

o "best-practices" included in the applications

o the efficiency they force an organization to undertake

o the analysis and reporting that can be used for long term planning

Besides the benefits, we should also understand that an ERP applications and transactions are probably unlike anything else that you have running on your network. It is not safe to assume that the network will handle them well. Every ERP deployment is different. The application modules are customized, and the architecture of the system and distribution of computing resources will be unique to your environment.

The following should come into consideration before you wish to implement ERP into your business:

o It is critical to understand how your application performs on the network before beginning enterprise-wide deployment.

o Test your network in advance to understand how these applications will perform and what impact they will have on existing applications. Pay particular attention to WAN links and heavily utilized LANs.

o The location of computing resources may have a significant impact on how the application performs, particularly in three-tier environments. Understand the traffic flows between the client and the server as well as the server-to-server communication flows.

2.5 Inventory Management

2.5.1 Definition

Inventory is defined as a detailed list of all the items in a place. [29]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. [7]

Inventory Management involves the control of the assets that are produced to be sold in the normal course of the firm's operations. [8]

2.5.2 Types of Inventory Management System

It is hard to classify the various inventory management systems in an orderly fashion. Common types of inventory control system are the permanent, two-bin, periodic, optional replenishment, and material requirements planning system. The permanent, 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. [9]

a) Permanent Inventory System

A permanent inventory system keeps records of the amount in storage, and it replenish 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 permanent system, each time a unit is issued from stock the withdrawal is jogged 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 permanent system there is constant or

permanent accountability on all items.

Advantages of Permanent System:

- 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

Disadvantages of Permanent System:

- 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 combination orders, which can frequently results in supplier
  discounts based on dollar value, must be foregone.

b) Two-Bin Inventory System

The distinguishing features of two-bin system are the absence of a permanent

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 trigged when the inventory level reaches a physical marks such as a painted line or a given volume level. The reorder point quality can also be placed in a bag or container, so that when the stock is drawn to the scaled 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, bolt 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 permanent 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 makes 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 or variable.

In the permanent 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 permanent system treats them discretely and dependently.

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 permanent and periodic systems. Stocks 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 than 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 requirements planning (MRP) inventory system is used extensively with planned production. For items that are materials or components used by end items, stocks levels are derived from the requirements dictated by the end item. The material requirement planning system is a derived order quantity system. This system function 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 is 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.

f) Just-In-Time Inventory Control System

Just-In-Time inventory control system has the aim to operate with lowest average level of inventory possible; within the EOQ (Economic Order Quantity) model the basics are to reduce (1) ordering costs and (2) safety stocks. The relative new JIT

approach to inventory control is growing in popularity as an attempt to obtain additional cost savings by reducing the level of inventory a firm needs to have in hand. Instead of depending solely one its own inventories, the firm relies on its vendors to furnish supplies "just-in-time" to satisfy the firm's production requirement. The MRP tends to be static, because once numbers are entered into the computer; no one has responsibility to change them. JIT works exactly the opposite way. As soon as improvements are made, new improvements are sought. There are many benefits to a successful implementation of JIT such as reduced work-in-process inventory (therefore less space and cost), higher productivity, short lead times, and visible & predictable amounts of inventory.

2.5.3 The Major Functions of Inventory

The usual/basic function of inventory control is making decision for the following:

- How often the inventory status should be determined?
- When a replenishment order should be placed?
- How large the replenishment order should be?

There are five major functional classifications of Inventory: [10]

(i) **Decoupling stock** - Inventory can act as a buffer between processes to avoid delay and inefficiencies. Large amount of inventory tend to exist between firms, or between divisions within a firm. An effective supply chain management must take into account to reduce inventory levels.

(ii) **Cycle inventory** - an attempt to order or produce in batches instead of one units at a time. The reasons for batch replenishments include, economies of scale (because of large setup cost), quantity discounts in purchase price or freight cost (suppliers offer discounts for large orders)

(iii) **Anticipation inventory -** consists of stock accumulated in advance of expected peaks in sales. Storage according to supply and demand (different season different rates). Thus during the period of high anticipated requirements, extra demand can be serviced from stock rather than from working overtime in the plant.

(iv) **Safety stock -** is the amount of inventory kept on hand, on the average, to allow for the uncertainty of demand and the uncertainty of supply in short run. Safeties stocks are not needed when the future rate of demand and length of time it takes to get complete delivery of an order are known with certainty.

2.5.4 Important Factors and Modeling Strategies for Inventory System

Through empirical studies and deductive mathematical modeling, a number of factors have been identified that are important for inventory management. Below is a table (figure 2.5.4), showing us the Inventory Planning Decision Variables based on the source from *A. Saipe, Partner, Thome Stevenson & Kellogg, "Managing Distribution Inventories," Executive Development Program, York University, Toronto, Canada, September 12, 1982*

Service Requirements	Supply Situation
• Customer expectations • Competitive practices • Customer promise time requirement • Order completeness required • Ability to influence and control customer • Special requirements for large customers	• Lead times • Reliability • Flexibility • Ability to expedite • Minimum orders • Discounts (volume, freight) • Availability
Demand Patterns	**Customer Ordering Characteristics**
• Variability • Seasonality • Extent of deals and promotions • Ability to forecast • Any dependent demand? • Substitution?	• Order timing • Order size • Advanced information for large orders • Extent of open or standing orders • Delay in order processing
Cost Factors	**Nature of the Product**
• Stock out (pipeline versus customer) • Carrying costs • Expediting	• Consumable • Perishable • Recoverable / repairable

• Write-offs	
• Space	
• Spoilage, etc	

Table 2.1 Inventory Planning Decision Variables

Now that we have identified the costs and other key variables for inventory planning

decision, we comment on three types of modeling strategies. Three types of strategies

involve some modeling:

(i) Detailed modeling and analytic selection of the values of a limited number of

decision variables.

(ii) Broader scope modeling, with fewer attempts at optimization.

(iii) Minimization of inventories with very little in the way of associated

mathematical models.

(I) **Detailed modeling and analytic selection of the values of a limited number of**

decision variables

The strategy here is to develop a mathematical model that permits the selection of

the values of a limited set of variables so that some reasonable measure of

effectiveness can be optimized. A classical sample is the economic order quantity

(EOQ) that, under certain assumptions, minimizes the total of ordering and

inventory carrying costs per unit time. In general, a mathematical model may permit

a deductive (closed-form) solution, an interactive solution (such as in the simplex

method of linear programming), or a solution by some form of trial-and-error

procedure (such as in the use of a simulation model).

(II) Broader Scope Modeling with Less Optimization

Here the strategy is to attempt to develop a more realistic model of the particular situation. However, the added realism often prevents any clearly defined optimization; in fact, there may not even be a mathematically stated objective function. One strives for a feasible solution that will provide reasonable performance. This is the philosophy underlying Material Requirement Planning (MRP).

(III) Minimization of Inventories with Little Modeling

Here the strategy is to attempt to minimize inventories without the help of mathematical models. The Just-In-Time (JIT) and Optimized Production Technology (OPT) philosophies fall into this category. They strive for elimination of waste (including inventories) and for continuous improvement. These powerful philosophies have been supplemented with the mathematical models that help managers refine and explain their operation. However, the philosophies are not grounded in models, but in an approach to doing business.

2.6 Existing System Review

Throughout the literature process, many web sites have been visited. Because of the proposed system is a web based system, thus website is chosen for peer review but not others resources, such as book, journal and proceeding paper. There are software available on the web, which contains integrated solutions designed to organize goods, track inventory, and manage accounts, or we can choose an inventory-management firm to handle the business. There are also many systems on the web, which offer standalone inventory management software.

Let us make a review on these existing systems, what services they have for inventory management & control. Below are the selected web sites together with their respective URL address.

2.6.1 Alcie Computer Network Corp. (www.alcie.com) — ERP software[11]
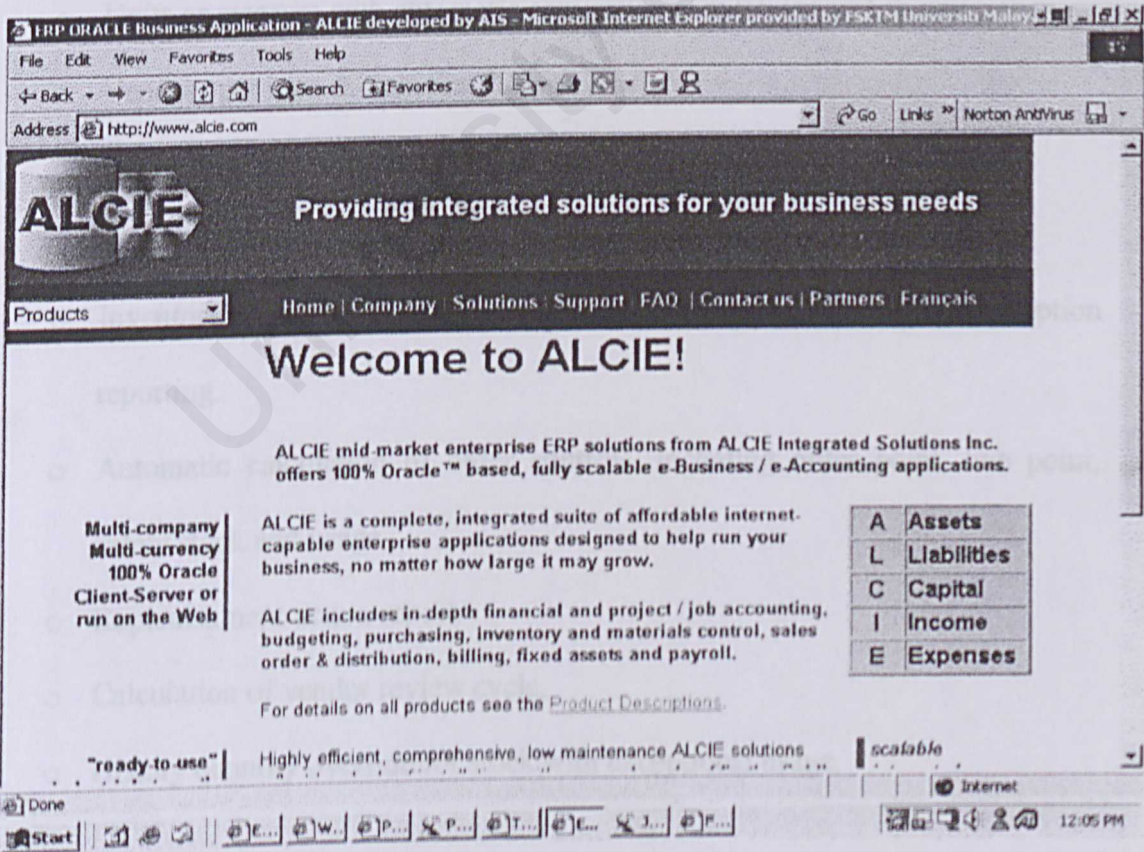
The Aide's Inventory Control component provides multi-company, multi-currency enterprise-wide inventory management application. It provides full inventory control including receipts, issues, transfers, physical cycle counts and ABC analysis. Automatically integrates with other installed modules such as Purchase Order & Requisitions, Sales Order Processing, Bill of Materials, Bulk Distribution, Project Accounting, Job Shop Manufacturing, General Ledger, Accounts Payable, and Accounts Receivables. The following shows some of it features:

- Supports stocked, non-stocked and phantom items.
- Virtually unlimited warehouses and stocking locations.
- Real-time inventory receipts, issues, and adjustment transactions
- Stock transfers with in-transit tracking and reporting
- Hazardous Material processing.
- Units or measure with automatic conversion at receiving and shipping time.
- Vendors per item & Substitute items.
- Accessories or options per item & Item comments.
- general Inventory stock status query and reporting
- Inventory valuation. Usage exception, ABC analysis, & Lead time exception reporting.
- Automatic calculation of order controls, including order point, line point, safety stock and usage
- Replenishment action report.
- Calculation of vendor review cycle.
- History quantity override for stock with exceptional usage.

2.6.2   Epicor Software Corporation (www.epicor.com) — ERP software[12]
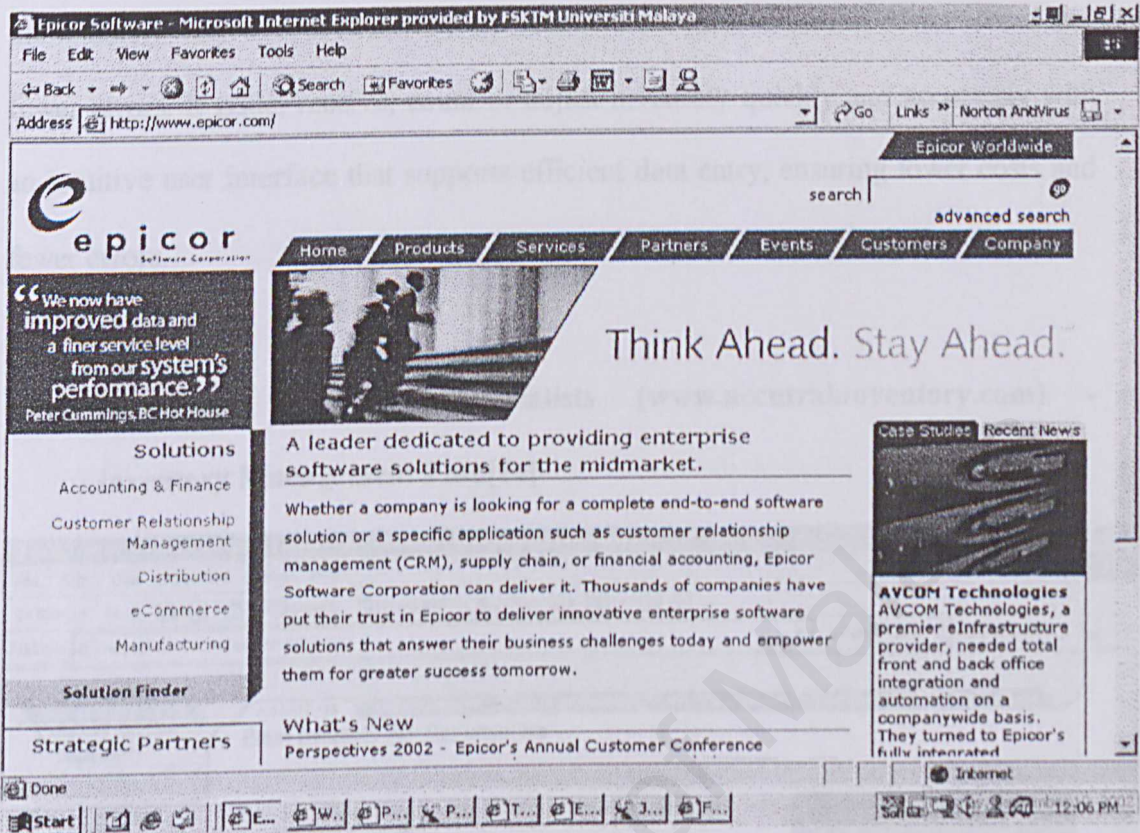
The inventory management (under the module eDistributiori) by Epicor allows matching of inventory levels, product profitability, and customer service. It provides all these function stated below:

- o   Inventory tracking to improve accuracy throughout all locations

- o   Integrated with EDI and Bar-coding technology

- o   Warehouse management including multiple locations

- o   Powerful pricing and promotions options

- o   Instant access to all costing information

- o   Extensive reporting capabilities

- o   Flexible controls to meet your specific business process needs

eDistributtons's Inventory application lies at the heart of the distribution suite and lets you manage your inventory with the control and level of flexibility most appropriate

for your business. It simplifies routine tasks with automatic processing, which can immediately improve productivity in your warehouses, ensuring lower costs and fewer errors. It issue, receive, count or adjust inventory quickly and accurately with an intuitive user interface that supports efficient data entry, ensuring lower costs and fewer errors.

2.6.3 AccuTrak Inventory Specialists (www.accutrakinventory.com) - inventory Management Finn[14]
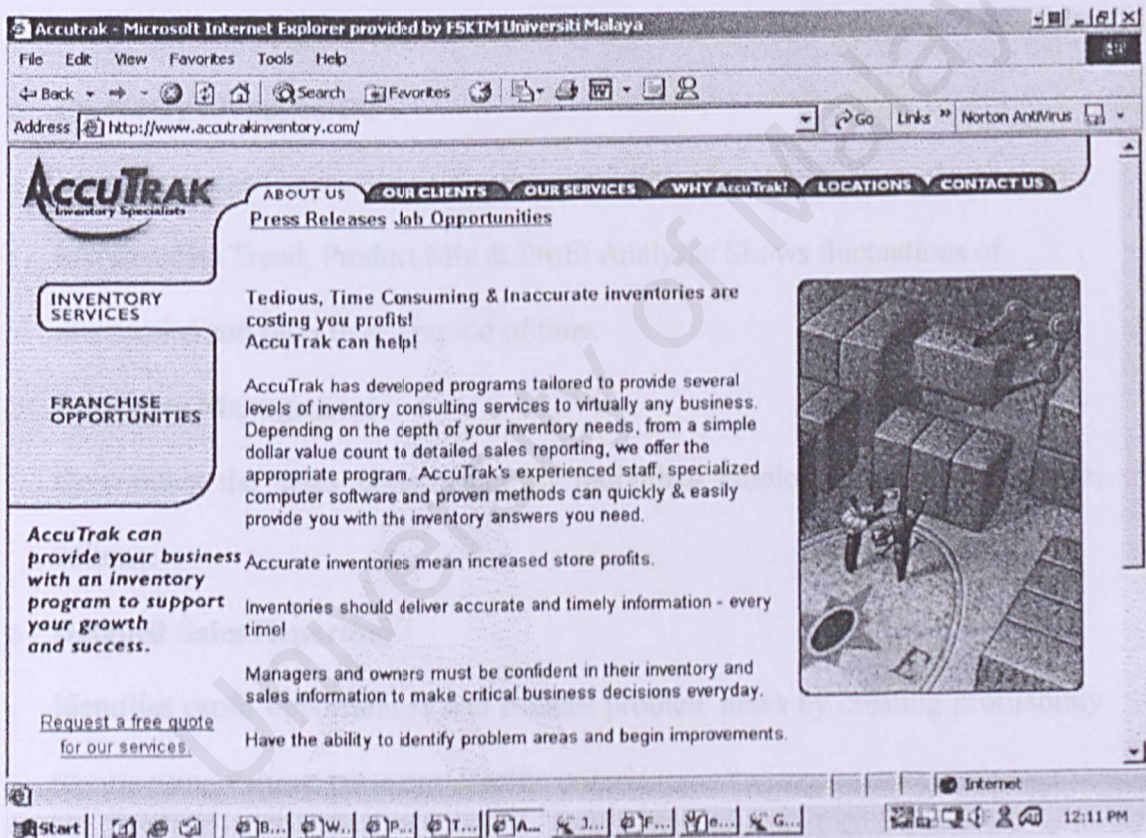
AccuTrak, a company leading the inventory consulting services industry, offering prospective business owners an opportunity to build a successful franchise and generate increased personal income. The entire retail and wholesale market segments are potential AccuTrak customers.

AccuTrak has developed programs tailored to provide several levels of inventory

consulting services to virtually any retail sales company. Depending on the depth of your clients inventory analysis needs, AccuTrak offers an appropriate program. Specialized AccuTrak computer hardware and software can aid you in generating valuable reports to assist your clients in successful inventory control. AccuTrak Services are as following:

o **Inventory Reporting**

Simple Valuation: Calculates the total dollar value of   inventory - especially useful for tax purposes or in a transfer of ownership. Quantity On-Hand Report: Assists owners by providing relative stock levels for all inventories.

o **Inventory Management**

Loss Shrinkage Control: Identifies the possibility of customer or employee theft Merchandise Trend, Product Mix & Profit Analysis: Shows fluctuations of inventories and sales over a period of time.

o **Employee Management**

Determines the sales performance of individual employees by tracking sales volume.

o **Detailed Sales Reporting**

Identifies profit opportunities and isolates problem areas by creating profitability reports using various customer specific criteria.

2.6.4 Millennium Software, LLC web site -- Atrex inventory control/POS software

Atrex is an inventory control/point-of-sale (POS) package for the business that desires complete control over stock levels and inventory tracking. Designed for the Microsoft Windows environment, Atrex is comprehensive, intuitive, and easy to use.

Features of Atrex include:

o Point-of-sale invoicing, order entry, and quotations

o Recurring invoicing & Purchase orders

o Vendor and Customer RMAs

o Multi-job service orders

o E-Commerce Links

o Support for temporary "Show" locations

o Simple receivables, payables, and bank account tracking

o 15 character alphanumeric stock codes

o Handles serialized, non-serialized, container, and assembly items

o Multiple pricing levels (5), unlimited pricing categories, and automatic price adjustments

o dBASE data format to allow for external reporting

o Import capabilities for stock codes, customers, and vendors

o Store notes on individual customers, stock codes, or transactions

o Barcode reader support

o Menu level security

o Data backup capabilities built in

o All transactions are real time to allow up-to-the minute reporting

- o Network and multi-user ready

- o Supports add on tax rates

- o Integrates with ICVerify and Authorizer for credit card authorization

- o Year 2000 compliant

2.6.5 Comparison of e-Inventory with existing system

In this section, I will do comparison of my system, which is the e-Inventory with the existing traditional inventory control system, and also with the computerized inventory control system (table 2.2). There is a fact that most of the local ICT companies still practicing traditional/manually inventory system, while a few of them are using the inventory software in running their business. Thus, this comparison is being done to show the strength of e-Inventory to improve the usage of information technology in local companies.

Traditional Inventory System	e-Inventory
• Time consuming and inefficient (record handling by manually)	• Allow computerized data and record keeping (fast & efficient)
• Unable to construct ' real-time' visibility strategy	• Provide 'real- time' visibility for inventory level supplying
• No integration between inventory control, monitoring and retailing system	• Integrated with other relevant modules. (e-Help Desk)
• Time consuming to prepare order item lists	• Order item lists can be prepared fast and easy

Table 2.2 Comparison of e-Inventory with existing system

2.7 Microsoft .NET

Microsoft .NET is a platform for building, running, and experiencing the next generation of distributed applications. It spans clients, servers, and services, and consists of:

- A programming model that enables developers to build Extensible Markup Language (XML) web services and applications.

- A set of XML Web services, such as Microsoft .NET My Services (formerly code-named internally as "Hailstorm"), that helps developers deliver a simple and integrated user experience.

- A set of servers, including Windows® 2000, SQL Server™, and BizTalk™ Server, that integrates, runs, operates, and manages XML Web services and applications.

- Client software, such as Windows" XP and Windows® CE, that helps developers deliver a deep and compelling user experience across a family of devices.

- Tools, such as Visual Studio* .NET, to develop XML web services as well as Windows-based and Web applications for, rich and compelling user experience.

2.7.1 .NET framework

The .NET Framework is a product that forms the development foundation for the Microsoft .NET platform. The .NET Framework and the device-focused .NET Compact Framework provide a managed, secure execution environment for XML

web services and applications, with comprehensive support for XML. The key

technologies in the .NET Framework are the Common Language Runtime, the class

libraries, and ASP.NET.

The Common Language Runtime

The Common Language Runtime includes technologies to make applications more

reliable (by, for example, eliminating memory leaks). It also provides a multi-

language execution environment, enabling full integration of components and XML

web services regardless of the programming language they were created in. Currently,

you can build .NET applications in more than 20 languages, including C++,

Microsoft® Visual Basic .NET, JScript®, and Microsoft's newest language — C#. A

large number of third-party languages are also available for building .NET

Framework applications, including COBOL, Eiffel, Perl, Python, Smalltalk, and

others.

The Class Libraries

The unified classes provide a consistent method of accessing the platform's

functionality, eliminating the need to learn and master different API architectures to

write your applications.

2.7.2 ASP.NET

ASP.NET builds on the .NET Framework's programming classes, providing a "web

application model" in the form of a set of controls and infrastructure that make it

simple to build web applications and XML web services. Developers are exposed to a

set of ASP.NET controls that encapsulate common HTML user interface widgets such as text boxes, drop down menus, and so on. These controls actually run on the web server, however, and simply project their user interface as HTML to a browser.

2.7.3 ADO.NET

ADO.NET is the loosely-coupled, XML-based evolution of today's ADO data access model. Built on and for loosely-coupled applications, ADO.NET employs a disconnected data cache that enables developers to quickly create high-performance, reliable XML web services and as well as today's n-tier applications.

Enterprise services

Built on the Windows 2000 Application Services, including the COM+ services and Microsoft Message Queuing Services, the .NET Framework Enterprise Services provide the scalability and reliability features needed for the most demanding enterprise applications such as transaction processing. In addition, deep integration between the Windows 2000 Application Services and the .NET Framework enables users to easily transform today's COM-based applications into XML web services through an administrative interface without writing any code.

Windows Forms

Windows Forms provide the smart client graphical user interface functionality in the .NET Framework. It provides a superset of the features found in today's Visual Basic component library and Windows Foundation Classes, as well as high-performance and simple access to the underlying Win32* APIs.

2.7.4 Benefits of the .NET Framework
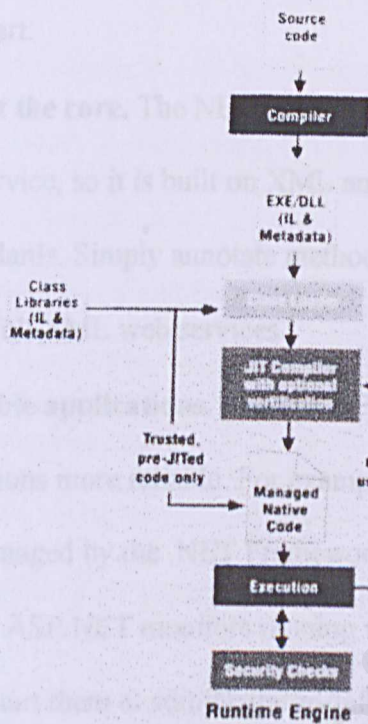
Figure 2.4 .NET framework

What are the benefits of the .NET Framework? Put simply, faster time to market, easier deployment and administration, and improved performance. Here are some of the key benefits:

- **Use any programming language.** The .NET Framework enables developers to use any programming language, and for applications written in any programming languages to integrate deeply with each other, enabling current development skills to go forward without retraining.
- **Write less code.** The .NET Framework uses a highly modular, plumbing-free design that enables developers to focus on writing business logic. Developers

don't need to write IDL or Registry code, and ASP.NET, for example, includes dozens of controls that encapsulate common programmer tasks such as a shopping cart.

- **XML/SOAP at the core.** The NET Framework was built for delivering software as a service, so it is built on XML and the SOAP family of integration standards. Simply annotate method calls and the .NET Framework turns them into full XML web services.

- **Run more reliable applications.** The .NET Framework includes technologies to make applications more reliable. For example, memory, threads, and processes are managed by the .NET Framework to ensure that memory leaks don't occur. And ASP.NET monitors running web applications and can automatically restart them at administrator-defined intervals.

- **Improve performance.** The .NET Framework improves the performance of typical web applications. ASP.NET includes advanced compilation and caching features that improve performance by a factor of two to three over existing Active Server Pages applications.

What are XML web services?

XML web services are the fundamental building blocks in the move to distributed computing on the Internet. Open standards and the focus on communication and collaboration among people and applications have created an environment where XML web services are becoming the platform for application integration. Applications are constructed using multiple XML web services from various sources that work together regardless of where they reside or how they were implemented.

There are probably as many definitions of XML web service as there are companies building them, but almost all definitions have these things in common:

- XML web services expose useful functionality to web users through a standard web protocol. In most cases, the protocol used is SOAP.

- XML web services provide a way to describe their interfaces in enough detail to allow a user to build a client application to talk to them. This description is usually provided in an XML document called a Web Services Description Language (WSDL) document.

- XML web services are registered so that potential users can find them easily. This is done with Universal Discovery Description and Integration (UDDI).
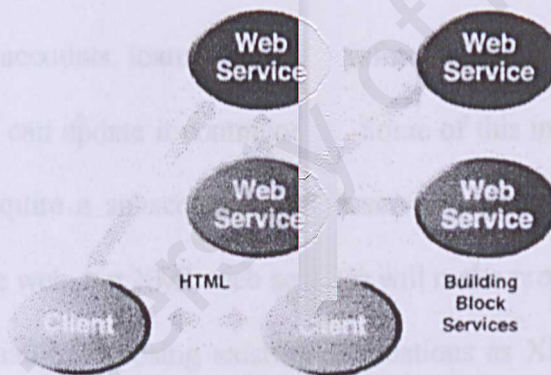
Figure 2.5 XML services

One of the primary advantages of the XML web services architecture is that it allows programmes written in different languages on different platforms to communicate with each other in a standards-based way. A significant advantage that XML web services have over previous efforts (CORBA.DCE etc) is that they work with standard web protocols—XML, HTTP and TCP/IP. A significant number of companies already

have a web infrastructure, and people with knowledge and experience in maintaining it, so again, the cost of entry for XML web services is significantly less than previous technologies.

We've defined an XML web service as a software service exposed on the web through SOAP, described with a WSDL file and registered in UDDI. The next logical question is. "What can I do with XML web services?" The first XML web services tended to be information sources that you could easily incorporate into applications — stock quotes, weather forecasts, sports scores etc. It's easy to imagine a whole class of applications that could be built to analyze and aggregate the information you care about and present it to you in a variety of ways; for example, you might have a Microsoft® Excel spreadsheet that summarizes your whole financial picture — stocks, 401K, bank accounts, loans, etc. If this information is available through XML web services. Excel can update it continuously. Some of this information will be free and some might require a subscription to the service. Most of this information is available now on the web, but XML web services will make programmatic access to it easier and more reliable. Exposing existing applications as XML web services will allow users to build new, more powerful applications that use XML web services as building blocks. For example, a user might develop a purchasing application to automatically obtain price information from a variety of vendors, allow the user to select a vendor, submit the order and then track the shipment until it is received. The vendor application, in addition to exposing its services on the web, might in turn use XML web services to check the customer's credit, charge the customer's account and set up the shipment with a shipping company.

In the future, some of the most interesting XML web services will support applications that use the web to do things that can't be done today. For example, one of the services that the Microsoft .NET. My services project will support is a calendar service. If your dentist and mechanic exposed their calendars through this XML web service, you could schedule appointments with them on line or they could schedule appointments for cleaning and routine maintenance directly in your calendar if you like. With a little imagination, you can envision hundreds of applications that can be built once you have the ability to programme the web.[26]

2.8 Technology Consideration

This Literature Review also included the consideration of technology that is currently available. The purpose is to choose a suitable development environment and development tools to make this project a success. The fields I have covered for technology consideration can be separate into the following:

1. Development Approach
2. Web Servers
3. Server Platforms / Operating Systems
4. Client/Server Architecture
5. Databases
6. Programming Environment and Languages
7. Development Tools
8. Others

2.8.1 Development Approach

a) Waterfall Model

One of the first models to be proposed is the waterfall model where the stages are depicted as cascading from one another. One development stage will be completed before the next begins. Thus, when all the requirements are elicited form the customer, analyzed for completeness and consistency, and documented in a requirements document, then the development team can go on to system design activities. [16]

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 intermediated products are necessary in order to begin the next stage of development. The advantages and disadvantages of the waterfall model are as follow: [16] & [17]

Advantages:

1. Enforced discipline through document. No phase is complete until the documents are done.

2. Present a very high-level view of what goes on during development and it suggests to developers the sequence of events they should expect to encounter.

3. Testing is inherent in every phase. Continuously as well as at end of phases.

4. It is very useful in helping developers lay out what they need to do.

5. Its simplicity makes it easy to explain to customers who are not familiar with software development.

Disadvantages:

1. Requirements must be well defined first

2. User must wait until the end in order to see the developed product

3. Each individual stage must be completed before mowing on

4. Product failure signals process failure

5. Increases in cost-to-fix or change software occur throughout the life cycle process

b) Prototyping Model

A prototype is a partially developed product that enables customer and developers to examine some aspects of proposed system and decide if it is suitable or appropriate for the finished product. In other words, prototyping means building a small version of a system, usually with limited functionality that can be used to help the user or customer identify the key requirements of a system and demonstrate of a design or

approach. [16]

Prototyping is often used to design a good user interface: the part of the system with which the user interacts. Since the prototyping model allows all or part of a system to be constructed quickly to understand or clarify issues, it has the same objectives as engineering prototype, where requirements or design require repeated investigation to ensure that the developer, user and the customer have a common understanding both of what is needed and what is proposed. One or more of the loops for prototyping requirements, design or the system may be eliminated, depending on the goals of the prototyping. However, the overall goal remains the same that is reducing risk and uncertainty in development. The advantages and disadvantages of the prototyping model are as follow: [19]

Advantages:

1. Allow errors to be detected early

2. More opportunity for changes

3. User orientation (to develop system that meet user needs to a greater extend)

4. Fast development time

5. Reduces development cost

Disadvantages:

1. Object system may be less efficient

2. System planning is not always complete

3. Visible use of computer resources 4 Requires cooperation between user and

   Information System

4. The development process of advanced versions does not always exploit the

investment made in the earlier stages.

2.8.2 Web Servers

a) Apache Web Server

Apache Web Server is among one of the popular web server on the market today. Some of its strengths are its reliability, performance and rich set of features. In large part, its popularity is due to the fact that its code is freely distributed. Among its features are [20]

- **Freeware**

  The software can be downloaded free of charge.

- **Multiple platforms support**

  Apache can support a variety of platforms such as Windows 9x, Windows NT, Linux, O/S 2, Novell Netware, Macintosh and others.

- **Logging**

  Apache can write to multiple log files and these log files can be automatically cycled or archived. The server also can generate referrer log entries and non-hit entries (such as comment). It can track individual users in log file.

- **Protocol Support**

  Apache can support the Windows Common Gateway Interface (CGI), HTTP/1.1 protocol. User can access the server state variables through CGI or scripting languages.

Despite all its strength, Apache also has some weaknesses. These include:

- Setup and maintenance of the server is done via command-line scripting tools. Apache does not offer browser-based maintenance facility.

- It does not have any Graphical User Interface (GUI) configuration or administration tools.

- The security is low and it does not support secure communication using Secure Socket Layer (SSL).

b) Internet Information Server (IIS)

Microsoft Internet Information Server (IIS) has the reputation for being one of the best web servers on the market. Although it is only available for Windows 20007 Windows NT, IIS has transformed their platform into a viable solution for deliver web-based applications. IIS is considered by experts to be just as powerful as and much easier to set up and maintain than many of its UNIX-based competitors.

The three major goals of Internet Information Server are that it is focused on the server side of the Internet world, to have the capability of supporting in intranet as well as the Internet and to form the basis that will enable future product development. Internet Information Server provides three basic services to users, which are World Wide Web server, File Transfer Protocol server and Gopher server. Some of its features are: [21]

- **Free download**

  It is available for free download.

- **Superior administration control**

  Site administration for IIS is performed using the Microsoft Management Console (MMC). Via this interface the users can manage access and security restrictions at the site, directory and file level. If the users are using virtual sites, then they can specify the estimated daily traffic for each site (which controls how much memory

IIS allocates for each web site) and limit the amount of server bandwidth a particular site can use. Most settings can also be configured remotely using Microsoft Internet Explorer. Active Server Pages (ASP) improvements in the latest release of IIS include additional support for transaction processing and memory isolation.

o **Excellent collection of server tools**

It offers a variety of tools including a Transaction Server (for building distributed applications, Certificate Server (managing digital certificates), Site Analyst (site management and usage), and Internet Connection Services for Microsoft Remote Access Service (creation of Virtual Private Networks), Mail Server and Network News Transfer Protocol (NNTP) News Server.

o **Indexing tool also handles Microsoft Office documents**

It also includes an indexing tool called Index Server. The Index Server is used for indexing HTMP pages and can also handle Microsoft Office documents.

o **Integrated search engine**

Internet Information Server provides the integrated search engine capabilities, which the users can create custom, search forms with Active Server pages, ActiveX Data Objects and SQL queries.

The weaknesses of Internet Information Server (IIS) are as follows:

o Lack of support for UNIX platform.

o Only runs on Server edition of Windows NT/Windows 2000

o Complicated to configure

o SMTP does not support POP mailboxes

o Mediocre documentation

2.8.3   Server Platform / Operating Systems

a) Windows 98

One of the products in Microsoft's evolution of the Windows operating system for personal computer is Windows 98. An important of the user interface of Windows 98 is the Web technology and it was released with its tightly integrated browser. Microsoft Internet Explorer is an integral part of the operating system in the Windows 98. Users can view and access desktop objects that reside on the World Wide Web as well as local files and application using the Active Desktop of Windows 98. In fact, Windows 98 desktop is a Web page with HTML links and features that exploit Microsoft's ActiveX controls.

It also provides a 32-bit file allocation table (FAT32) that allowing a single-partition disk drive larger than 2Gbytes. The other features in Windows 98 includes supports for Universal Serial Bus (USB), which make it easy to plug in new devices support for Digital Versatile Disc (DVD), support for a new industry standard form of power management called Advanced Configuration and Power Interface (ACPI). Windows 98 enables the news and other content to be set up and pushed to the user from specified Web sites. When using it as Web server, Personal Web Server (PWS) need to be installed in order for the Windows system to serve Web pages through Internet. [30]

b) Windows NT

Windows NT is the operating system for personal computer created for users and business requiring advanced capability. It is actually comprised of two products: Microsoft NT Workstation and Microsoft NT Server. The workstation is a little safer than Windows 98 and Windows 95. It is designed for users especially business users, who need faster performance. The server is designed for business machines that need to provide services for LAN-attached computers. Together with an Internet server such as Microsoft's Internet Information Server (IIS), it is required for a Windows system that plans to serve Web pages. The latest version, the Windows NT Server version 5.0 is now available, called Windows 2000.

Whereas, system requirements for Windows NT 4.0 Option Pack are the system need to be install both Windows NT Service Pack 3 and Service Pack 4 and Microsoft Internet Explorer 4.01. In order to run Internet Information Server 4.0 (IIS) and any other features on the Windows NT 4.0 Option Pack.

In term of networking, Microsoft Windows NT Server 4.0 works with Microsoft LAN Manager, Microsoft Windows for Workgroups operating system, AppleTalk, DECPATH WORKS, IBMLAN Server, IBMSNA networks, the Internet, NFSnetworks, Novell NetWare, Remote Access Services by way of ISDN, X.25, and standard phone lines and TCL/IP networks. [30]

c) Windows 2000

The latest version of Microsoft's evolving Windows operating system is Windows 2000. Before that, it is called Windows NT 5.0. Microsoft emphasizes that Windows 2000 is evolutionary and is built on NT technology. Most users of Windows 98 and Windows NT will in time move to Windows 2000. It is designed especially for small

business and professional users as well as to more technical and larger business market for which the NT was designed.

It was reported in earlier reviews that Windows 2000 is more stable that Windows 98/NT systems. It is less likely to clash. A significant new feature is Microsoft's Active Directory that enables virtual private networks to be set up by a company, data locally on the network to be encrypted and to give users access to share files in a consistent way any network computer.

The other features of the Windows 2000 are it has a fully customizable administrative console that can be based on tasks rather than files, applications, or users. Also Dynamic Domain Name Server (DNS), which replicates changes in the network using the Active Directory Services, the Dynamic Host Configuration Protocol (DHCP), and the Windows Internet Naming Service (WINS) whenever a client is reconfigured. Besides that, it also has the ability to create, extend, or mirror a disk volume without having to shut down the system and to back up data to a variety of magnetic and optical storage media. In addition, it also has close integration with and supports for Microsoft's Message Queue Server, Transaction Server, and Internet Information Server (IIS). [30]

d) Red Hat Linux

What is now known as Linux began life as a small student project at the University of Helsinki in Finland. The Linux operating system is a small, UNIX-like operating system, which is a rather remarkable operating system that works very well, is very stable, and rivals offerings from Microsoft, Apple, and Sun. Linux offers the following features:

   o Multitasking - Linux natively runs more than one task (program) at a time in

a UNIX-like fashion. It also   supports preemptive multitasking,   where

priorities can be set for different processes.

o Multi-user - Many users can be networked to a single Linux server. Linux is

   the least expensive, and perhaps best-featured, multi-user operating system on

   the planet today.

o User-friendly - The X Window System interface and a slew of great tools

   make Linux easy to use.

Today, Linux claims half a million users, and that number keeps growing every day.

One reason is cost—Linux itself is free of charge and only those who package a Linux

distribution charge for it (and many don't charge at all). These distributions, such as

Red Hat Linux, vary as to their installation processes and accompanying tools. [22]

However, Linux also have several weaknesses: [23]

o Installation problem - Even though Linux does a reasonably good job of

   detecting all the hardware components on a computer, it does require some

   enlightened input from the installer. Disk partitioning and mounting of file

   systems are relatively advanced concepts, especially for users who are used to

   the simple drive letters of Windows.

o Lack of high-end features - Linux developers have not so far access to high-

   end and expensive hardware. It not yet scale well beyond 4 Central

   Processing Unit (CPUs), has no support for high-availability clustering,

   Cache-Coherent Non-Uniform Memory Access (CCNUMA) architectures

   and cannot host multiple independent operating system on the same machine.

2.8.4 Client/Server Architecture

A client- server computing can be defined as the logical extension of modular programming where modular programming has as its fundamental assumption that separation of large piece of software into its constituent parts(refers to modules)creates the possibility for easier development and better maintenance. Client server computing takes this a step furthers by recognizing that modules need not all are executed within the same memory space.

As on other words, a client means that a single user workstation, which provides presentation services and the appropriate computing, connectivity, and database services and interfaces relevant to any communication needs, whereas a server would means a computer, usually high-powered workstation, a minicomputer, or a mainframe, that houses information for manipulation by networked clients.

Currently, the most common type of server is the database server, which usually controls the relational database. This kind of server enables many clients to share access to same database and also enables the use of high-performance computer system to manage the database. The client-server computing is considered as distributed computing where users, applications, and resources are distributed in response to business requirement and linked by a single Local Area Network (LAN) or Wide Area Network (WAN) or by Internet of networks. That is the Two-tier architecture and the Three-tier architecture.

a) 2-Tier Client/Server Architecture

The two-tiered architecture contains two computers -- a client, and a server- with areas of logic combined on the client. The three components of an application- presentation, processing, and data are divided between the two tiers: client application

code and database server. A robust client application development language and a versatile mechanism for transmitting client requests to other server are essential for a two-tier implementation. Presentation is handled exclusively by the client, processing is split between client, and server, and data is stored on and accessed through the server.

Advantages of Two- Tier Client-Server

- Application development speed. In most cases, a two-tier system can be developed in a small fraction of the time it would take to code a comparable but less- flexible legacy system.

- Most tools for two-tier are very robust and lend themselves well to iterative prototyping and rapid application development (RAD) techniques.

- Two-tier architectures work well in relatively homogeneous environments with fairly static business rules.

Disadvantages of Two-Tier Client-Server

- The two-tier architecture faces several potential version control and application redistribution problems, a change in business rules would require a change to the client logic in each application in a corporation's portfolio affected by the change.

- System security in the two-tier environment can be complicated because a user may require a separate password for each SQL server accessed.

- Client tools and the SQL middleware used in two-tier environments are also highly proprietary and the PC tools market is extremely volatile. The volatility of the client/server tool market raises questions about the long-term viability

of any proprietary tool and organization may commit to and complicates implementation of two-tier systems.

b)   3-Tier Client/Server Architecture

The components of three-tiered architecture are divided into three layers: a presentation layer, a functionality layer, and the data layer. Each of these layers must be logically separate. The three-tier architecture attempts to overcome some of the limitations of the two-tier scheme by separating presentation, processing, and data into separate distinct entities. The middle-tier performs calculations or makes requests as a client to additional server. Middle-tier functionality servers may be multithreaded and can be accessed by multiple clients, even those from separate applications. Although three-tier systems can be implemented using a variety of technologies, the calling mechanism from client to server in such a system is most typically the remote procedure call, or RFC (remote procedure call).

Advantages of 3-Tier Client-Server

- Unlike in most two-tier implementations, the three-tier presentation client is not required to understand SQL. This added flexibility allows a firm to access legacy data and simplifies the introduction of new data base technologies.

- Having separate software entities allows for the parallel development of individual tiers by application specialists.

- Provides for more flexible resource allocation. Middle-tier functionality servers are highly portable and can be dynamically allocated and shifted as the needs of the organization change. Network traffic may be reduced.

- Reusable logic reduces subsequent development efforts, minimizes

the maintenance workload, and decreases migration costs when switching client applications.

Disadvantages of 3-Tier Client-Server

4   Three-Tier brings with it an increased need for network traffic management, server load balancing, and fault tolerance.

5   Current tools are relatively immature and require more complex 3GLs for middle-tier server generation.

2.8.5   Databases

a)   Microsoft SQL Server Version 7.0

Microsoft SQL Server Version 7.0 is the most robust database for the Windows family. SQL Server is a client/server relational database management system (RDBMS) that is highly integrated with the Windows NT operating system. By using SQL Server, modern application can be developed that separate the client application and the database service. SQL Server Transact-SQL supports the ANSI-92 standard and provides extensions to the SQL language. [24]

Microsoft SQL Server™ supports a set of features that result in the following benefits:

   o   **Ease of installation, deployment and use**

       SQL Server includes a set of administrative and development tools that improve your ability to install, deploy, manage and use SQL Server across several sites.

   o   **Scalability**

       The same database engine can be used across platforms ranging from laptop

computers running Microsoft Windows 95/98 to large, multiprocessor server running Microsoft Windows NT, Enterprise Editor.

- o **Data warehousing**

  SQL Server includes tools foe extracting and analyzing summary data for Online Analytical Processing (OLAP). SQL Server also includes tools for visually designing database and analyzing data using English-based questions.

- o **System integration with other server software**

  SQL Server integrates with e-mail, the Internet and Windows. SQL Server includes OLAP Services, Data Transformation Services and English Query, and works with over 45 ISVs that form Data Warehousing Alliance. By using Windows 2000, SQL is the fastest database for SAP, based on the SAP Retail benchmark.

b)   Microsoft SQL 2000

Microsoft SQL 2000 is an enterprise-level database and is the latest version of Microsoft SQL 7 It is enhanced with Extensible Markup Language (XML) support that enables data return in XML and also enables XML be used to insert, update, and delete values in the database. SQL Server 2000 supports enhancements to distributed partitioned views that allow users to partition tables horizontally across multiple servers. Its full-text search includes change tracking and image filtering.

In the security point of view, SQL Server 2000 uses Kerberos to support mutual authentication between the client and the server, as well as the ability to pass the security credentials of a client between computers. Moreover, SQL Server 2000 introduces a new, more easily understood model for specifying backup and restores options. In term of scalability, it Scales up to 32 CPUs and 64 GB RAM.

c)   Microsoft Access

Microsoft Access 2000 is a Windows-based database management system, which runs under the Windows 95/98/2000/NT operating system. Access offers an easy-to-use database for managing and sharing data. It also adds increased integration with the Web for easier sharing of data across a variety of platforms and user levels. It enables sharing of database among the co-workers over the Internet, searching and retrieving the information quickly, and taking advantage of automated, pre-packaged solutions to quickly create databases.

Also, Stat/Transfer can be used to convert data between Microsoft Access and your favorite spreadsheet, database or statistical package. Besides that, data in Microsoft Access can be migrated to the Microsoft SQL Server.

Benefits of Microsoft Access:

- o An easy-to-use tool for easily finding information that provides consistency and integration with the other applications in the office suite.

- o Access 2000 allows easily sharing information via the corporate Intranet and the ability to easily host a database within the browser.

- o User may create solutions that combine the easy-to-use of the Access interface (client) with the scalability and reliability of SQL server.

d)   Oracle 9i Database

Oracle 9i Database is the latest database product from Oracle Corporation. It provides transparent application scalability by sharing cluster-wide caches for coordinated data access; it also includes business intelligence capabilities, and provides programmatic access, centralized management, and multi-channel delivery of Internet services. A key feature of the database is the facility to recover from disaster situations.

Besides that, Oracle 9i database is designed with integrated manageability that creates a complete business view of all components powering e-business processing. As any other database software in the market, Oracle 9i provides multiple layers of security to prevent any unauthorized access to the database. Oracle 9i is compatible with UNIX and Windows NT.

2.8.6 Programming Environment and Languages

a)   Active Platform [30]

Microsoft has a very 'active' perspective on how application architectures should be created. For that matter, the word 'active' or more specifically, the term Active X, has become a standard part of packing just about every product concept released from Microsoft within the last years.

Active Platform is the name given to Microsoft component-based application development model for the web. Active Platform is divided into 3 major sections:

- Active Desktop - an Active X enabled client architecture designed to house component, and language-independent scripting environments.

- Active Server - primarily intended to provide an execution environment for running business logic distributed across enterprises application servers.

- Active X Technologies - the object-based underpinnings of all the above.

         - Commonly referred to as the Component Object

                  Model (COM).

Active X

In the simplest terms, Active X is an architecture that lets a program (the active X control) interacts with other program over a network (such as the Internet). It is quite

a different animal than Java, which is an entire new programming language plus a specification for a virtual machine. The Active X architecture uses Microsoft's Component Object Model (COM) and distributed COM (DCOM) standards, whereby:

- COM allows different application to talk to each other locally;
- DCOM provides communication over a network.

It is use to activate web pages, and also add application functionality. Furthermore, it is faster than a speeding Java applet.

To write an Active X control, the developer can use a number of popular programming languages, including Microsoft Visual C++ 5.0, Microsoft Visual Basic 6.0, etc. Active X controls are self-contained pieces of a program or stand-alone components. Developers can reuse them in other programs even programs written in another language. For example, you can take a control written in Visual Basic and insert it into a program in Visual C++.

Active X evolved from OLE; in fact, Active X objects are basically OLE objects with additional functions that enable them to work on the World Wide Web. Because OLE has been around for a while, developers have written many OLE objects that can now be used as ActiveX controls.

b)   CGI (Common Gateway Interface) [30]

The Common Gateway Interface (CGI) is a standard way and format that browsers use to send data to the server, as well as the format the server uses to hand the data off to a script.

The Common Gateway Interface was introduced as a standard protocol for extending the functionality of web server with additional applications. Most CGI applications

are simple executables that are launched every time they are requested.

The CGI provides a consistent way for data to be passed from the user's request to the application program and back to the user. This means that the person who writes the application program can make sure it gets used no mater which Operating System the server uses. It's simply a basic way for information to be passed from the web server about your request to the application program and back again.

c)   **Visual Basic.NET** [26]

Microsoft Visual Basic® .NET, included in Microsoft Visual Studio .NET Professional, Enterprise Developer and Enterprise Architect editions, is the latest version of Visual Basic built specifically for existing Visual Basic developers who want to get the most out of the software development experience.

In addition to more power, productivity, and application stability, Visual Basic .NET provides key enhancements that solve the most pressing challenges that Visual Basic developers face today. From the new integrated development environment (IDE) to a modern, streamlined Visual Basic language, Visual Basic .NET delivers the top requested features built for today's Visual Basic developer:

Seamless Deployment

Visual Basic .NET solves the most pressing issues around Windows-based application deployment and makes "DLL Hell" and component versioning issues a thing of the past. New XCOPY deployment enables developers to install a Windows-based application simply by copying files to a directory. With Visual Basic .NET and new auto-download deployment, Windows-based applications can be installed and executed simply by pointing a Web browser to a URL.

81

More Robust Code

Visual Basic .NET delivers the feature most requested by existing Visual Basic

developers—fewer bugs in the code they write. Features in the new Visual Studio

.NET IDE, such as the real-time background compiler and the task list, keep Visual

Basic developers up-to-date on any coding errors as they occur, enabling quick and

effective error resolution. Enhancements to the Visual Basic language, such as strict

type checking and structured exception handling, enable developers to write code that

is more robust, maintainable, and less prone to run-time errors.

Powerful Windows-based Applications

Visual Basic .NET is the most productive tool for constructing powerful Microsoft

Windows®-based applications. The new Windows Forms Designer enables

developers to get their desktop applications to market in less time. New features

include control anchoring and docking to eliminate the need for complex resize code,

the in-place menu editor to deliver WYSIWYG menu creation, and the tab order

editor to provide rapid application development (RAD) organization of controls.

Powerful, Flexible Data Access

Visual Basic .NET provides developers with both the ActiveX Data Objects (ADO)

data access programming model that they know and love, plus the new XML-based

Microsoft ADO.NET. With ADO.NET, developers gain access to more powerful

components, such as the Dataset control and a new, strongly typed programming

model that provides Microsoft IntelliSense® statement completion within data access

code.

Simplified Component Creation

Visual Basic .NET brings RAD to component development. Developers can use non-visual toolbox and server explorer components to easily incorporate resources such as message queues, event logs, and performance counters into their applications without writing a single line of code.

Enhanced Control Creation

Visual Basic .NET provides unprecedented flexibility in building customized user controls. Developers can easily extend preexisting user controls and Windows Forms controls as well as design their own controls that generate custom user interfaces.

Enhanced Control Creation

Visual Basic .NET provides unprecedented flexibility in building customized user controls. Developers can easily extend preexisting user controls and Windows Forms controls as well as design their own controls that generate custom user interfaces.

Integrated Reporting with Crystal Reports

Upgrading to Visual Studio .NET Professional Edition (or later) provides Visual Basic developers with the power of Crystal Reports directly within the IDE. Crystal Reports delivers the most productive, integrated, and RAD experience for creating highly graphical and interactive relational data reports. These reports can be generated for the entire array of Visual Basic .NET application types, including Windows, Web and mobile applications.

Easy Web-based Application Development

Visual Basic .NET delivers "Visual Basic for the Web." Using new Web Forms, you

can easily build true thin-client Web-based applications that intelligently render on any browser and on any platform. Web Forms deliver the RAD programming experience of Microsoft Visual Basic® 6.0 forms with the full power of Visual Basic .NET rather than limited scripting capacity. The new HTML designer delivers IntelliSense statement completion for HTML tags and the separation of user interface (UI) and code enable more efficient team-based development.

Existing Investments Carry Forward

Visual Basic .NET enables developers to leverage their existing investments in code and skills. Windows Forms provides a robust container for Microsoft ActiveX® controls. Component Object Model (COM) Interoperability provides bi-directional communication between existing Visual Basic applications and those written with Visual Basic .NET. The upgrade wizard enables developers to seamlessly migrate up to 95 percent of existing code to Visual Basic .NET.

d)   Java [31]

The JAVA programming language is a high-level language that can be characterized as Object-oriented, architecture-neutral, portable, interpreted, distributed, high performance, secure, dynamic, and robust.

A Java program is compiled into byte codes, and a Java Virtual Machine (JVM) will interpret these byte codes. Java technology is both a programming language and a platform. The Java platform differs forms most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.

The Java platform has 2 components:

- o The Java Virtual Machine (JVM)

- o The Java Application Programming Interface (API)

The following figure (figure 2.7.5c) depicts a program that's running on the Java platform. As the figure shows, the Java API and the virtual machine insulate the program from the hardware.
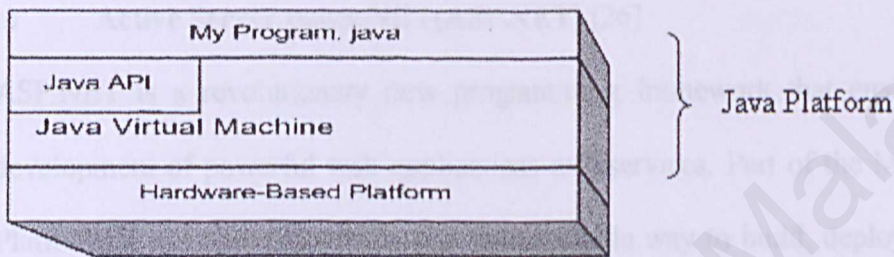
Figure 2.6 Java Platforms

JVM base for the Java platform and is ported onto various hardware-based platforms. The Java API is a large collection of ready-made software components that provide many useful capabilities, such as Graphical User Interface (GUI) widgets...etc. The API supports all these kinds of program within packages of software components. Every full implementation of the Java platform gives the following features:

- o The essentials: Object, strings, threads, numbers, input and output etc.

- o Applets: The set of conventions used by applets.

- o Networking: URLs, TCP (Transmission Control Protocol), UDP (User Datagram Protocol) sockets, and IP (Internet Protocol) addresses.

- o Internationalization: Help for writing programs that can be localized for users worldwide.

- o Security: Both low level and high level, including electronic signatures,

public and private key management, access control and certificates.

- o Object serialization: Allows light weight persistence and communication via Remote Method Invocation (RMI).

- o Java Database Connectivity (JDBC™):   Provide uniform access to a wide range of relational database.

The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more.

e)   Active Server pages.NET(ASP.NET) [26]

ASP.NET is a revolutionary new programming framework that enables the rapid development of powerful web applications and services. Part of the Microsoft .NET Platform, it provides the easiest and most scalable way to build, deploy and run web applications that can target any browser or device.
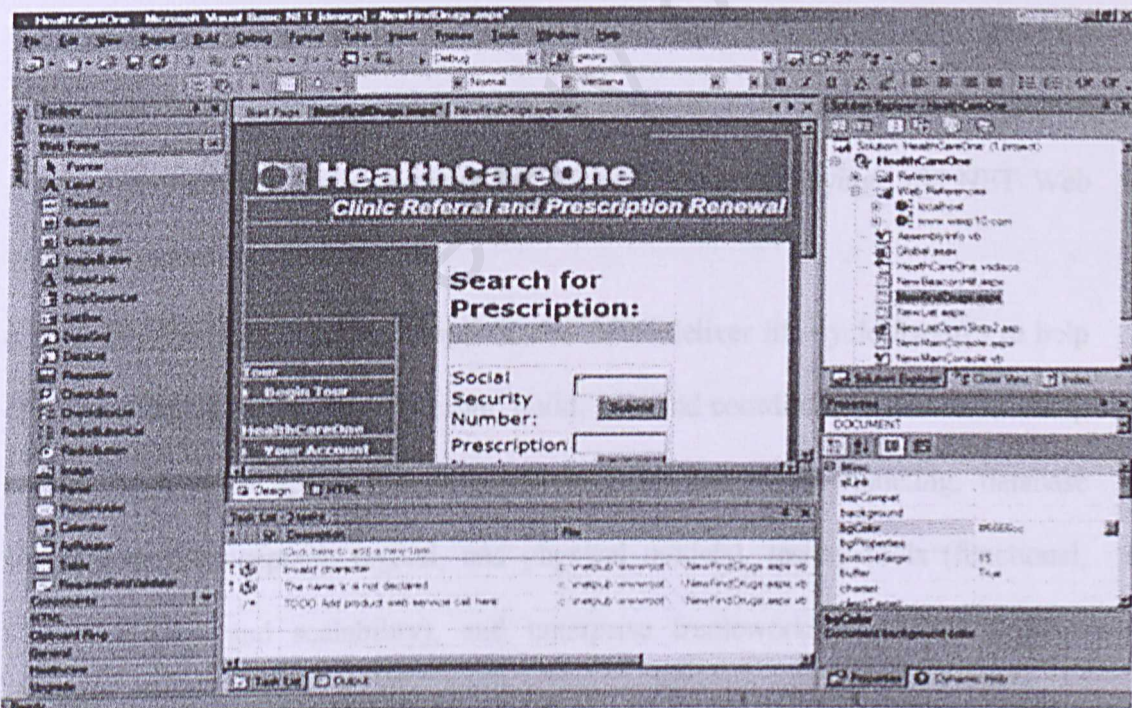
- **Easy Programming Model**. ASP.NET makes building real world Web applications dramatically easier. ASP.NET server controls enable an HTML-like

style of declarative programming that let you build great pages with far less code than with classic ASP. Displaying data, validating user input, and uploading files are all amazingly easy. Best of all, ASP.NET pages work in all browsers -- including Netscape, Opera, AOL, and Internet Explorer.

- **Flexible Language Options**. ASP.NET lets you leverage your current programming language skills. Unlike classic ASP, which supports only interpreted VBScript and JScript, ASP.NET now supports more than 25 .NET languages (including built-in support for VB.NET, C#, and JScript.NET -- no tool required), giving you unprecedented flexibility in your choice of language.

- **Great Tool Support**. You can harness the full power of ASP.NET using any text editor -- even Notepad! But Visual Studio .NET adds the productivity of Visual Basic-style development to the Web. Now you can visually design ASP.NET Web Forms using familiar drag-drop-double-click techniques, and enjoy full-fledged code support including statement completion and color-coding. VS.NET also provides integrated support for debugging and deploying ASP.NET Web applications.

- The Enterprise versions of Visual Studio .NET deliver life-cycle features to help organizations plan, analyze, design, build, test, and coordinate teams that develop ASP.NET Web applications. These include UML class modeling, database modeling (conceptual, logical, and physical models), testing tools (functional, performance and scalability), and enterprise frameworks and templates, all available within the integrated Visual Studio .NET environment.

- **Rich Class Framework**. Application features that used to be hard to implement, or required a 3rd-party component, can now be added in just a few lines of code

using the .NET Framework. The .NET Framework offers over 4500 classes that encapsulate rich functionality like XML, data access, file upload, regular expressions, image generation, performance monitoring and logging, transactions, message queuing, SMTP mail, and much more.

f) Java Server Pages (JSP)

Java Server Pages allows web developers and designers to rapidly develop and easily maintain, information-rich, dynamic web pages that leverage existing business systems. As part of the Java family, Java Server Pages enables rapid development of web-based applications that are platform independent. Java Server Pages technology separates the user interface from content generation enabling designers to change the overall page layout without altering the underlying dynamic content. Java Server Pages technology uses XML-like tags and scriptlets written in the Java programming language to encapsulate the logic that generates the content for the page. Additionally, the application logic can reside in server-based resources that the page accesses with these tags and scriptlets. Any and all formatting (HTML or XML) tags are passed directly back to the response page. This makes Java Server Pages technology makes it faster and easier than ever to build web-based applications. Together, JSP technology and servlets provide an attractive alternative to other types of dynamic web scripting/programming that offers platform independence, enhanced performance, and separation of logic from display, ease of administration, extensibility into the enterprise, and most importantly, ease of use. [31]

g)   PERL(V5)

Perl is a high-level programming language and derived from the C programming language Perl's process, file, and text manipulation facilities make it particularly well-suited for tasks involving quick prototyping, system utilities, software tools, system management tasks, database access, graphical programming, networking, and world wide web programming. Perl is flexible and extensible to use on virtually any task, from one-line file-processing tasks to large, elaborate systems. For many people, Perl serves as a great replacement for shell scripting. For others, it serves as a convenient, high-level replacement for most of what they'd program in low-level languages like C or C++. But Perl has some drawbacks. Firstly, it has somewhat of a reputation for being unreadable and this can leads to difficulties in maintaining the problem. Secondly, as Perl does have any formal support, it sometimes difficult to get any support when needed. Lastly, there would be a penalty in the execution time of Perl Language as it is an interpreted language.

h)   Java Script

Java Script is a lightweight scripting language developed by Netscape Communications Corporation in such a way that it allows users to create dynamic Web pages. However, Java Script is not Java. It is merely a scripting language that is parsed and executed by the parser. Java Script is a simple scripting language that is very similar to C programming.

Java Script is an object-based scripting language that is designed for developing Internet applications. Java Script is used as a means to tell an application what to do, unlike languages that are used to create applications; it cannot do anything without the

application.

User can develop server applications or client applications with Java Script. The term server" is referring to the computer where the Web page resides. The term "client" is referring to the browser application that loads and displays Web page. Java Script is an extension to HTML that lets users create more sophisticated Web pages than they ever could with HTML alone.

Using Java Script, even less-experienced developers will be able to direct response from variety of events, objects and actions. It provides anyone who can compose HTML with the ability to change image and play different sounds in response to specified events, such as a users' mouse click or screen exit and entry.

i)   VBScript

VBScript, the newest member of the Visual Basic family of programming languages, brings active scripting to wide variety of environments, including Web client scripting in Microsoft Internet Explorer version 3.0 and Web server scripting in Microsoft Internet Information Server version 3.0.

For people who are already known Visual Basic or Visual Basic for Applications, VBScript will be very familiar. Even if they don't know Visual Basic, once they learn VBScript, they are on their way to programming with the whole family of Visual Basic language.

VBScript talks to host applications using ActiveX Scripting. With ActiveX Scripting, browsers and other host applications don't require special integration code for each scripting component. ActiveX Scripting enables a host to compile scripts, obtain and call entry points and manage the namespace available to the developer.

With ActiveX Scripting, language vendors can create standard language run times for

scripting. Microsoft is working with various Internet groups to define the ActiveX

Scripting standard so that scripting engines can be interchangeable.

2.8.7   Development Tools

a)   Microsoft Web Matrix

Features Overview for Microsoft Web Matrix:

Visual Design Tools for XML Web Services and Application

- Visually Specify and Communicate Application Architecture

  Use Microsoft Visio®–based Unified Modeling Language (UML) modeling to

  specify and communicate application architecture and functionality.

- Gain Unparalleled Support for Database Modeling

  Capture business requirements easily with the Visio-based conceptual

  modeling capabilities. Full roundtrip engineering with logical and physical

  data models enable communication from the business analyst to the database

  designer.

- Orchestrate Business Processes

  Visually link business processes together using a full-featured version of

  Microsoft BizTalk™ Server designed for developers.

Solid Architectural Guidance

- Easily Share Best Practices

  Senior developers and architects can customize enterprise template projects to

  provide application starting points and guidance for less experienced

  developers.

- Define and Communicate Development Policies

  Template Description Language can be used with enterprise template projects to specify development policies and guidelines. Developers can immediately view any policy violations and discrepancies in their code in the Task List Window.

Open Tools Platform for Developing XML Services

- Use the Industry-Leading Development Environment

  Leverage the powerful capabilities in Web Matrix for rapidly building XML Web services and applications that span devices and platforms.

- Get Support for the Complete Life Cycle

  The Web Matrix Integration technologies bring a broad array of partners to Visual Studio .NET developers and enables third-party languages, components, and tools to integrate tightly with Visual Studio .NET through all stages of the development life cycle

b)   ColdFusion

ColdFusion is a popular set of product for building Web sites and serving pages to users, which enable a content database to be built by using input templates and combine these with application programs to create a Web site in which pages are developed dynamically as they are served. It consists of ColdFusion Studio, which is used to build a site, and ColdFusion Server, which serves the pages to users. ColdFusion Studio is described as "a complete integrated development environment (IDE)" and ColdFusion Server as "a deployment platform".

The most valuable feature for ColdFusion is the ability to build Web sites as "piece parts" that can be stored in a database and then reassembled for Web pages, e-mail, newsletters and other users. ColdFusion provides a visual interface for building Web page directly or for building the "piece parts". ColdFusion is also a popular tools for building e-commerce sites

ColdFusion has its own page markup language called ColdFusion Markup Language iCFML) CFML encompasses the Web's Hypertext Markup Language HTML and Extensible Markup Language (XML). A just-in-time (JIT) compiler turns the CFML into the pages that get served. ColdFusion can be coordinated with distributed applications that use CORE A or Microsoft's DCOM to interact with other network applications.

2.8.8 Others Tools

a)   Adobe Photoshop 6.0

Photoshop offers a lot of features for image rendering. Photoshop uses palettes for quick access to functions such as navigation, colors, layers, and actions (macros). User can draw objects and text that remain editable and that print at the resolution of the output device, producing crisp edges no matter how much the user enlarges the image. The new custom Shapes palette makes it easy to generate standard geometric shapes, or to draw preset shapes (hearts stars) in the document. The smoothness of text edges can be controlled, and a neat warping tool lets user instantly shape your text into arcs, bulges, and fish eyes.

b)   Microsoft Paint

Microsoft Paint provides another alternative in image editing. Paint's interface is

straightforward that its usage does not require any help with most common tasks. Besides that, it also has a set of advanced tools to perform to ease image-editing process. Paint supports both bitmap (raster) images and vector objects. But unlike the rest of the packages, Paint integrates its vector tools seamlessly and elegantly so that it is easy to switch back and forth between vector and bitmap tools.

But, Paint fails in one critical area: print proofing. It does not include a standard CMYK preview.

2.9 Summary

All these researches were done mainly to gain information for this project. The information gathered includes information regarding the Internet, E-Business, E-commerce, Enterprise Resource Planning (ERP), concepts and strategies of Inventory Management, reviews on the existing ERP and Inventory Management application, development methodologies, and development tools.

Research on Internet, E-business, E-commerce and ERP is to study the strengths and values from their business aspect. Meanwhile the study on Inventory Management's concepts and strategies is to help the developer to have a better understanding on the requirements of this project. Assessing current existing systems/applications allows identification of the weaknesses that are to be overcome in this project.

As for the development methodologies, this review focuses on two development models, the Waterfall Model and Prototyping Model, each having its own distinct features. The development model that has been chosen for this project will be mentioned in the next chapter.

Various development tools for the project were analyzed, using the information gathered from the Internet and reference books. Information on four different server platforms were gathered, they are Windows 98/NT/2000 and red hat Linux. The strengths and weaknesses of these operating systems are summarized into a comparison table and will be shown in the following chapter.

Besides, the Web Servers such as Apache Web Server and Internet Information Server (IIS) are also studied. Other information such as system architecture, databases, programming environment and languages are also analyzed, including Client-Server Architecture, SQL server, Active Platform, Visual Basic.NET, Java,

Active Server Pages.NET (ASP.NET), Java Server Pages (JSP), Perl and etc.

Finally, there is research on suitable development software, where in this project; I have made consideration between Microsoft Visual Studio.NET, ColdFusion and. In Chapter 3, we shall mention about all the chosen tools.

Chapter 3: Methodology & System Analysis

After the process of survey, summarization, analysis and synthesis in chapter 2: Literature Review, to produce the project system, a methodology is selected based on the congruence between the method and the system, which also recognize as development strategy. The development strategy is a set of steps or plans to develop a system. In developing a system, it is very essential to have this strategy, as it will help the developer to organize the project well. Every system development needs the user to determine the requirements. These requirements describe the system. [16]

3.1 Justification of Methodology

The steps taken throughout the project must be monitored to give a good indication of progress. The use of a development model allows the project members to plan work and monitor the completion of tasks from the original requirements, through the specification, design, implementation, testing, results, and evaluation and conclusion parts of the project. It was decided that the development model that was chosen should have clear steps to be taken, and closure at the end of each stage so progression could be easily monitored. A model with these characteristics was desired as this is the first large project undertaken as an individual. It was also required to be able to see exactly what word had been completed and what work was pending at any instant.

After studying the two development models in previous chapter, I have found that Prototyping model was not favorable, as it required much time and resources. In addition, the prototyping model requires high skilled and professional programmer in

developing the prototype. Therefore, I have decided to use the *Waterfall Model* as my development methodology for this project.

The Waterfall Model was chosen because of the following reasons:

- **Very structured**

  The system is design using a logical flow.

- **Predictable**

  It allows estimation of the completion of each stage so that the system can be developed within the time frame given.

- **Involves user participation**

  Require information gathering form the user in order to develop a system that meet user needs to a greater extend.

- **Good visibility**

  All the requirements can be identified and well defined.

- **More efficiency**

  The time and resources can be well determined in order to enable developers to manage the project more efficiency.
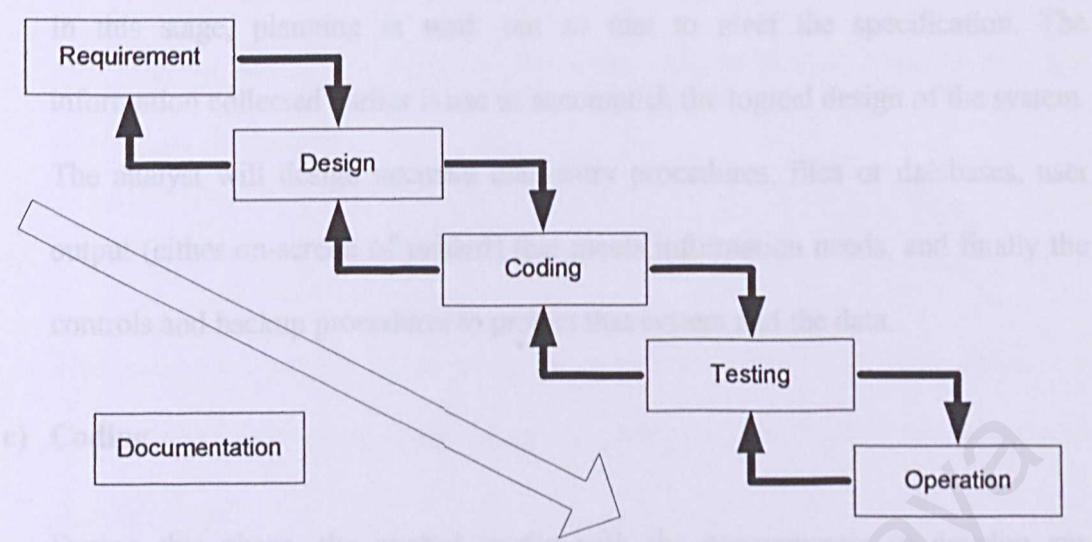
The Waterfall Model is shown as below:

Figure 3.1: Waterfall Life Cycle Model

a) **Requirements**

Software requirement definition and analysis defines functional capabilities,

performance, design constrains and system interfaces. It provides the software

designer with representation of information and function that can be translated to

data, architectural and procedural design. The requirement analysis task is a

process of discovery, refinement, modeling and specification. The software scope,

initially established by the system engineer and refined during software project

planning, is refined in detail. Models of the required information and control flow,

operational behavior and data content are created. Alternative solutions are

analyzed and allocated to various software elements.

b) Design

In this stage, planning is work out so that to meet the specification. The information collected earlier is use to accomplish the logical design of the system.

The analyst will design accurate data-entry procedures, files or databases, user

output (either on-screen of printed) that meets information needs, and finally the

controls and backup procedures to protect that system and the data.

c) Coding

During this phase, the analyst works with the programmers to develop any

original software that is needed. Programmers have a key role in this phase

because they design, code and remove syntactical errors form the program. To

ensure quality, a programmer may conduct either a design or a code walk-through

to explain complex portions of the program to a team of other programmers.

d) Testing

Before the system can be used, it must be tested. It is much less costly to catch

problems before the system is signed over to users. A series of tests to pinpoint

problems is run first with sample data and eventually with actual data from the

current system.

e) Operation

In this last phase of the system development, the analyst helps implement the

system that involves training the users to handle the system. Some training is done

by the vendors, but oversight of trainings is the responsibility of the system

analyst. Additionally, the analyst needs to plan for a smooth conversion files form old systems to new one. This process includes converting files from old formats to new ones or building a database, installing equipment and bringing the new system into production.

f) Documentation

This integrated process takes place every phase. Activities of each phases are documented in the report form so that to provide a clear view of the progress of the each stage.

3.2   Requirement Analysis

As stated above, this project is going to be developed using the Waterfall Model approach. Thus, the strategy or plan to develop this project will follow closely the steps that are involved in the Waterfall Model. The steps taken for this project is described below:

3.2.1   Identifying requirements

In developing a system it is very important to identify the system requirement. In order to identify them, a lot of information is needed. As for this project, information was gathered through research and analysis.

a) Research

Research for this project was done on the Internet and Reference Books. The results of this research were described in Chapter Two.

b) Review & Analysis

Review and analysis was conduct in order to identify problems, requirements, opportunities and objectives of the project.   .

3.2.2 Requirements for this project

After gong through all the information gathered and analyzing it, the requirements for this project were outlined. Basically, they are divided into **functional requirements** and **non-functional requirements**.

a) Functional Requirements

A functional requirement described a communication between the system and its environment. Since the requirement described a system's behavior, hence the functional requirement also describes how the system should behave given the certain stimulus.

For e-Inventory, there will be a design of several sub-modules, which will cover the functions needed under inventory management such as the following:

(i)   **Login Authentication**

To take security measurement, there will be a user authentication for the entire registered user of e-Inventory. This is to make sure that only the authorized user can access the confidential area of the system, which is the back-end of the company.

(ii)   **User Account Management**

Only the authorized Administrators can use this function. They can view user information, add a new user, edit user information, delete existing user account and search for user.

(iii)   **Inventory Management/Maintenance**

This module is sub-divided into three main categories as follows:-

- **Manage Software Inventory Record (Add/Edit/Delete)**

  - The *Add* option is used for entering new objects into the inventory record. In this page you can find two special fields (Parts and Software), which are optional to the users. We need these two fields because some of the hardware might come with parts and original licensed software. And this is related to each other.

  - Meanwhile the *Edit* option enables us to make modifications to the existing inventory record like making changes to the software records.

  - For the *Delete* function, we need it to erase object that has been write of or expired.

- **Manage Supplier Record (Add/Edit/Delete)**

  - The *Add* option is used for entering new software supplier record. In this page you can only add new item into the inventory record.

  - While the *Edit* option enables us to make modifications to the existing supplier inventory record. For examples, changes of address and telephone number.

  - As for the *Delete* function, we need it to erase the supplier records that no longer supply any software to the company.

- **Manage e-Help Desk (View and Reply)**

  - *e-Help Desk* is a simple web portal system that gives the user of staff to get support from system administrator via web technology.

- This module can give opportunity to users to post their entire problem to system administrator and IT support team. Besides, it also enable user to post their purchasing request to inventory staff. And this function also helps the inventory staff on documentation and follow-ups.

- The *view* function enables the inventory staff to view all the problems face by the users and purchasing request by the users.

- The *reply* function enables the certified professional experts to solve the problems face by the users using virtual computing software. Besides it also let purchasing department to answer request from the user.

(iv) **Search(View/Sort)**

The user can look for the inventory or supplier information they wanted by using this search function. They can also sort out all the information they wish to review by using this function.

(v) **Generating Summary Report**

The user may also generate their inventory status report of selected store form time to time. In a report, one can see inventory data stated above for the selected store.

b) Non-Functional Requirements

A non-functional requirement or constraint describes a restriction on the system that limits ones choice for constructing a solution to the problem [16]. The non-functional requirement for this project is describe as below:

(i) **Browsability**

"Browsability means logically classifying the data and information"

As the size of information system grows, so does the need to logically organize the data. This implies grouping conceptual sets of data with similar conceptual sets of data. Browsability becomes apparent when it is coupled with hypertext and logical groupings of information. A browsable information system can provide advantages such as: entire system can be see at a glance, no need of vocabulary knowledge, Similar items are grouped together, easy to navigate, and stimulates thinking

(ii) **Readability**

"Readability means good page layout"

Readability 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 as generate some message to inform the user what is happening.

(iii)   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 charge to underlying components. Database scalability issues can be resolved using distributed database architecture where as web application sealing can be addressed by increasing bandwidth or by additional web servers.

(iv)   Usability

The system should be developed so that it is easy to use it. It will enhance and support rather than limit the office processor. Human interfaces need to be intuitive and consistent with the inventory management system and within it self.

(v)   Security

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

(vi)   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 of data and services to ensure the continuous of operation.

(vii)   **Manageability**

The modules within the system should be easy to manage. This will make the maintenance and enhancement works simpler and less times consuming.

(viii)   **Flexibility**

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

3.3   Technology Consideration

Analysis has been done in order to select the most appropriate programming

technologies and languages that suit the requirements of this system. Below are the

choices that have been decided and chosen:

3.3.1   Selected Development Tools

After doing comparison between all currently available web technologies, I have

found that **Active Server Pages.NET (ASP.NET)** technology is most appropriate for

building the e-Inventory System. **Microsoft Visual Studio.NET** has been chosen as

the development tool, as it is the best development tool for supporting Active Server

Pages (ASP) compared to other web-development software such as ColdFusion and

Macromedia Drumbeat 2000.

3.3.1 (a)   Reason for Choosing ASP.NET

Easy Programming model

ASP.NET makes building real world web applications dramatically easier. ASP.NET

server controls enable an HTML-like style of declarative programming that let you

build great pages with far less code than with classic ASP. Displaying data,

validating user input, and uploading files are all amazingly easy. Best of all,

ASP.NET pages work in all browsers — including Netscape, Opera, AOL, and

Internet Explorer.

Flexible language options

Unlike classic ASP, which supports only interpreted VBScript and JScript, ASP.NET now supports more than 25 .NET languages (including built-in support for VB.NET, C#, and JScript.NET — no tool required), giving you unprecedented flexibility in your choice of language.

Great tool support

The Enterprise versions of Visual Studio .NET deliver life-cycle features to help organizations plan, analyze, design, build, test, and coordinate teams that develop ASP.NET web applications. These include UML class modeling, database modeling (conceptual, logical, and physical models), testing tools (functional, performance and scalability), and enterprise frameworks and templates, all available within the integrated Visual Studio .NET environment.

Rich class framework

Application features that used to be hard to implement, or required a 3rd-party component, can now be added in just a few lines of code using the .NET Framework. The .NET Framework offers over 4,500 classes that encapsulate rich functionality like XML, data access, file upload, regular expressions, image generation, performance monitoring and logging, transactions, message queuing, SMTP mail, and much more, improved performance and scalability.

Compiled execution

ASP.NET is much faster than classic ASP, while preserving the "just hit save" update model of ASP. However, no explicit compile step is required! ASP.NET will automatically detect any changes, dynamically compile the files if needed, and store the compiled results to reuse for subsequent requests. Dynamic compilation ensures that your application is always up to date, and compiled execution makes it fast.

Rich output caching

ASP.NET output caching can dramatically improve the performance and scalability of your application. When output caching is enabled on a page, ASP.NET executes the page just once, and saves the result in memory in addition to sending it to the user. When another user requests the same page, ASP.NET serves the cached result from memory without re-executing the page.

Web-Farm session state

ASP.NET session state lets you share session data user-specific state values across all machines in your Web Farm.

Enhanced reliability

ASP.NET ensures that your application is always available to your users.

Memory leak, deadlock and crash protection

ASP.NET automatically detects and recovers from errors like deadlocks and memory leaks to ensure your application is always available to your users.

Easy deployment

ASP.NET takes the pain out of deploying server applications.

Dynamic update of running application

ASP.NET now lets you update compiled components without restarting the web

server. With ASP.NET, you simply copy the component over the existing DLL —

ASP.NET will automatically detect the change and start using the new code.

Easy migration path

You don't have to migrate your existing applications to start using ASP.NET.

ASP.NET runs on IIS side-by-side with classic ASP on Windows 2000 and Windows

XP platforms.

XML web services

XML web services allow applications to communicate and share data over the

Internet, regardless of operating system or programming language.

Mobile web device support

ASP.NET Mobile Controls let you easily target cell phones, PDAs — over 80 mobile

web devices — using ASP.NET. You write your application just once, and the mobile

controls automatically generate WAP/WML, HTML, or iMode as required by the

requesting device.

3.3.1 (b)   Reason for Choosing Microsoft Visual Studio.NET

Through the use of Visual Studio .NET and the .NET Framework, Microsoft provides developers with a full set of development tools in which to quickly and easily create state-of-the-art applications and XML web services.

Faster development

- Through the use of the Common Language Runtime, a part of the .NET Framework, developers can create XML web services using any modern programming language, greatly increasing the pool of available developer resources as well as allowing the developers on hand the freedom to use the programming language most suited to solve the problem at hand.

- XML web services are reusable. Their modular nature allows developers to quickly stitch together solutions. When creating a commerce service for a book company, the developer can use preexisting XML web services for user authentication, determining taxes and shipping, allowing a more efficient use of resources on developing key business functions, in this case the catalog.

- Using Visual Studio .NET, XML web services written in different programming languages can be debugged. This greatly increases the amount of coding resources available.

- COM components, a staple of Windows development, can be readily incorporated in applications and XML web services on the .NET platform.
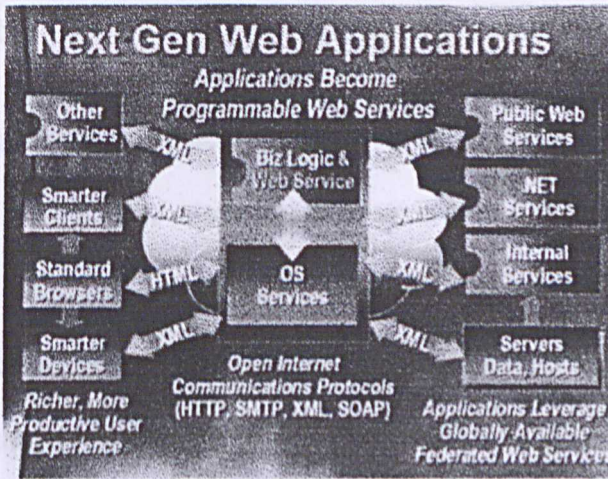
Figure 3.2 Web Application connect with XML

Less code

- Microsoft .NET reduces the amount of code developers need to write. They will be able to reuse existing XML web services, thus saving valuable time.

- Visual Studio .NET largely automates the transformation of an application into an XML web service, decreasing the time and effort necessary to quickly create and deploy solutions.

- XML by its nature separates data from how it is displayed. Uncoupling the display characteristics from the .NET experience makes it easy to add new interface technologies, like speech and handwriting recognition, without needing to rewrite the application.

- .NET My Services, and other .NET services, provides core capabilities needed by many applications — user authentication, notification capabilities, contacts list, and the like — without requiring additional coding.

Greater reliability

- Harnessing abundance of processing power and bandwidth currently available, the .NET platform takes advantage of the power of distributed computing. Processing occurs wherever it makes the most sense or wherever the XML web service resides, spreading out the CPU load and reducing network traffic.

- The Common Language Runtime provides for a managed execution environment eliminating memory leaks, access violations, and versioning problems.

- The .NET Framework enforces type safety, explicit code sharing, and application isolation, guaranteeing that no XML web service can affect or illegally call another.

Integration

- Microsoft .NET's reliance on XML — an open standard managed by the World Wide web Consortium (W3C) — removes barriers to data sharing and software integration. The Simple Object Access Protocol (SOAP), an XML-based messaging technology standardized by the W3C, specifies all the necessary rules for locating XML web services, integrating them into applications, and communicating between them. A Common Language Runtime (CLR) provides a unified management environment, enabling XML web services written in any programming language to work together. .NET also has database access capabilities, allowing developers to bring ODBC-compliant data stores into their application architecture.

- Microsoft .NET allows IT departments to reduce internal costs and expand the capabilities they can deliver to their customers by enabling them to tap their legacy applications and data stores, and other vendors' XML web services for expertise and outsourced services. Microsoft .NET is designed to enable aggregation of value and services from multiple sites and companies into coherent experiences for users.

3.3.2 Selected Platform

After all the consideration, I have finally decided to choose Windows 2000 as my system platform, as it is the most suitable operating system that are able to support all the development tools I choose to bring this project into success.

Advantages of Windows 2000 Server:

- An enhancement of Windows NT 4.0.
- Built-in application such as Microsoft Internet Information Server 4.0 (IIS) and Internet Explorer 5.0 browser
- It is suitable for enterprise or organizational level.
- Showing high performance, reliable, secure and easy-to-manage characteristics for information sharing and running applications.

3.3.3 Selected Web Server

I have decided to use Microsoft Internet Information Server (IIS) as the system web server, because it can be fully supported by Windows 2000 and provide powerful security, administration and development functionality. It is also one of the best web servers on the market and it is a high-end enterprise-level server. IIS is considered by

experts to be just as powerful as and much easier to set up and maintain than many of

its UNIX-based competitors.

Advantages of IIS:

- Offers a superb platform for building sophisticated Internet applications.

- Easy to install and uninstall

- Accessible since all kinds of browsers can work with it.

- Allows for hosting multiple sites.

- Provides capabilities for secure transactions with the SSL (Secure Sockets Layer) support and for authentication.

- Windows-based Web authoring and development tools are supported.

- Integration with existing industry-standard database and other ODBC-compliant databases.

3.3.4 Selected Database Server

The **Microsoft SQL Server** 7.0 has been chosen as the database server of the e-Inventory. The Microsoft SQL 7.0 provides enterprise-level database. It is proven to be stable and also allow a larger storage of databases compare to Microsoft Access 2000. Through tight integration with IIS, SQL Server can be queried and updated via Web browsers. Also, the SQL server is able to handle more concurrent users as compared to Microsoft Access.

3.4 Hardware Requirement

Basically, the hardware requirement for this e-Inventory system is as follow:

- Pentium 166MHz of faster processor(200 MHz recommended)

- Memory - 64Mb RAM or above

- 4.0GB hard disk or above

- Network card NIC 10/100

118

3.5 Summary

By following the Waterfall Model methodology approach, the requirement analysis and election of development tools has been conducted. Method used for identifying these requirements has been mentioned above. The requirements for the e-Inventory such as functional and non-functional requirement were also determined.

The process of choosing suitable tools is also carried out smoothly according to the steps -this methodology. Below is a list of all the development tools and language chosen for this project, together with its hardware requirement:

- Server Platform   -   Microsoft Windows 2000 Server
- Web Server   -   Microsoft Internet Information Server (IIS)
- DBMS   -   Microsoft SQL Server 2000
- Programming Language   -   ASP.NET, Visual Basic.NET
- Development Tool   -   Microsoft Visual Studio.NET
- Processor   -   Pentium 166 MHz and above (or equivalent)
- Memory   -   64 MB RAM or above
- Hard disk   -   4GB with minimum of 650MB of free space

Chapter 4: System Design

4.1   Introduction

System design is an important stage of system development where the requirements for the system are translated into the system characteristics to meet the user requirement and satisfaction. It is a creative process of transforming problems into a solution. The system design includes a complete description of the functions and interactions involved. Although the system design phase of a software project describes only appearance and functionality but it is a key factor determining the success and accomplish of a software project.

To design a system is to determine a set of components and inter-component interfaces that satisfy a specified set of requirements. There are many ways to create good designs. The choice is based on designer preferences. However, every system design involves some kind of decomposition. It is a process that starts with a high level description of the system's key features and functions will fit together. [25]

System design is an iterative process. The process is iterative because, in actually, the designer move back and forth among activities involving understanding the requirements, proposing possible solutions, testing aspects of a solution for feasibility, presenting possibilities to the customers, and documenting the design for the programmers. Thus, the waterfall model methodology is suitable for the proposed system's system design.

The stages in the design process are architectural design, database design, user interface design, and form design, processes, entities and data stores involved along with the relevant description.

In order to meet the requirements mentioned in previous chapter, the guidelines for

my system design are listed below:

1. Specify logical elements

   Detailed design specification that describe the features of an information system;

   input, output, database and procedures.

2. Meet user requirement

   Meet user needs stated in terms of:

   - performing appropriate procedures correctly

   - presenting proper forms of information

   - providing accurate results

   - using appropriate method of interaction

   - providing overall reliability

3. Easy to use

   - favorable human engineering

   - Ergonomic design that is physically comfortable and contributes to user

     effectiveness and efficient.

4.1.1 CONCEPTUAL AND TECHNICAL DESIGNS

To transform requirements into a working system, designers must satisfy both

customers and the system builders on their development team. The customers

understand what the system is to do. At the same time, the system builders must

understand how the system is to work. For this reason, design is really a two-part

iterative process.

First, developer produces conceptual design that tells the customer exactly what the system will do. Once the customers approves the conceptual design, developer translate the conceptual design into munch more detailed document, the technical design, that allows system builders to understand the actual hardware and software needed to solve the customer's problem.

The design documents describe the same system, but in different ways because of the difference audiences for the documents. Thus, the conceptual design concentrates on the system's function, and the technical design describes the form the system will take. [20]

4.2   System Architecture

Following is a diagram of the 3-tier client/server architecture that will be implemented in this system (Figure 4.1).
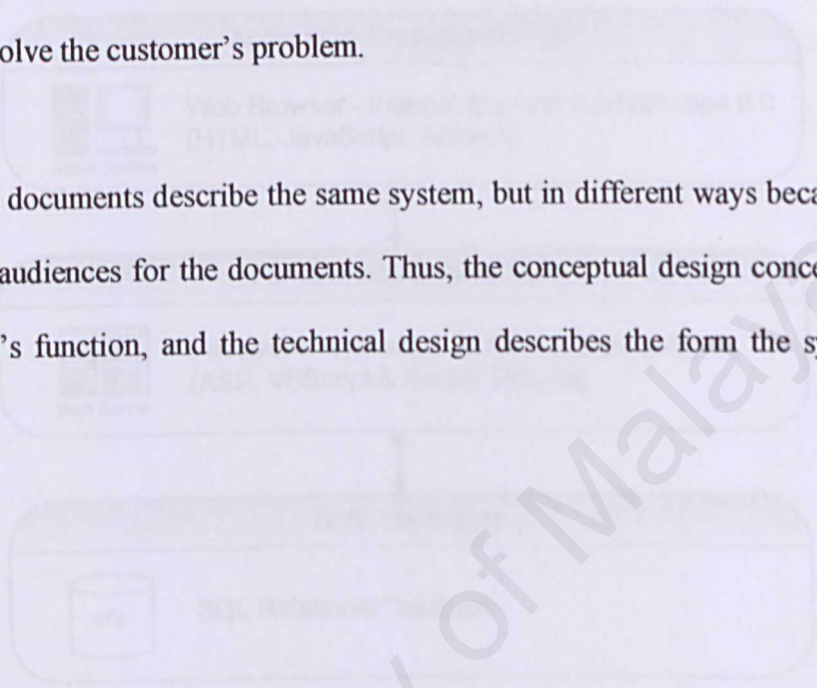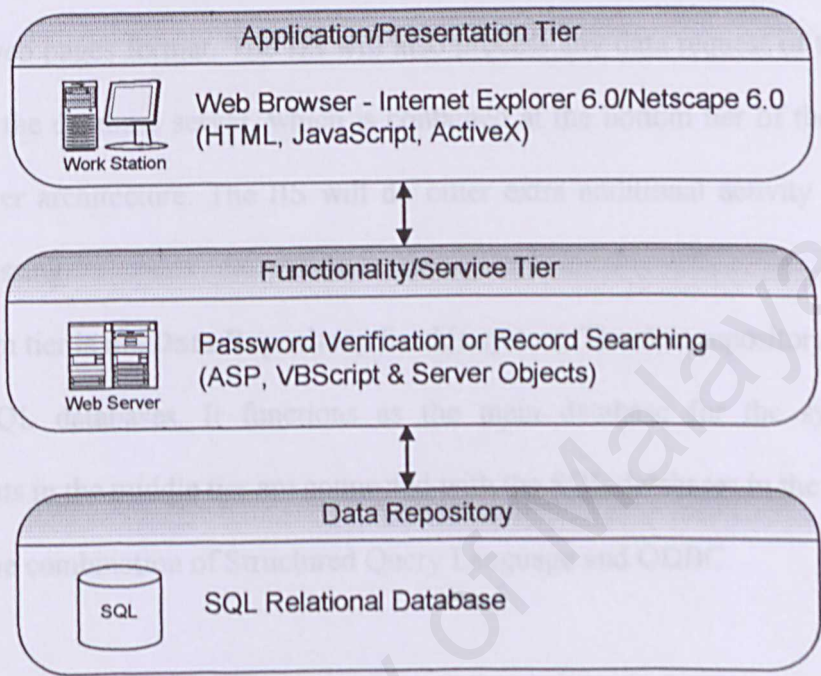
Figure 4.1   3-Tiers Client/Server Architecture

The **Application/Presentation Tier** consists of all the necessary application. In this layer, the main application component that appear to the user is the Web browser, which will provide the user interface. HTML, JavaScript and Active X Controls are used to activate the application layer. All of them provide the most flexible and dynamic interface for the users. The application always resides within the web server, which is the Internet Information Server (IIS).

The middle tier is known as the **Functionality/Service Tier**. The communications between this tier and the application layer depends on the Hypertext Transfer Protocol (HTTP) for the web pages transfer. Then the functionality tier consists of the

components that are created to support this system as password verification, searching

for records, and others configuration. All these components require Active Server

pages (ASP), VBScript and Active X objects to perform the functions in the web

servers. The IIS in this tier will process the requests from the client and produces the

result in web pages format. The IIS will also process any data request of the user by

linking to the database server, which is contained at the bottom tier of the three-tier

client/server architecture. The IIS will do other extra additional activity during the

data processing.

The bottom tier is the **Data Repository** for this system. The data repository is built up

by the SQL databases. It functions as the main database for the system. The

components in the middle tier are connected with the SQL databases in the bottom tier

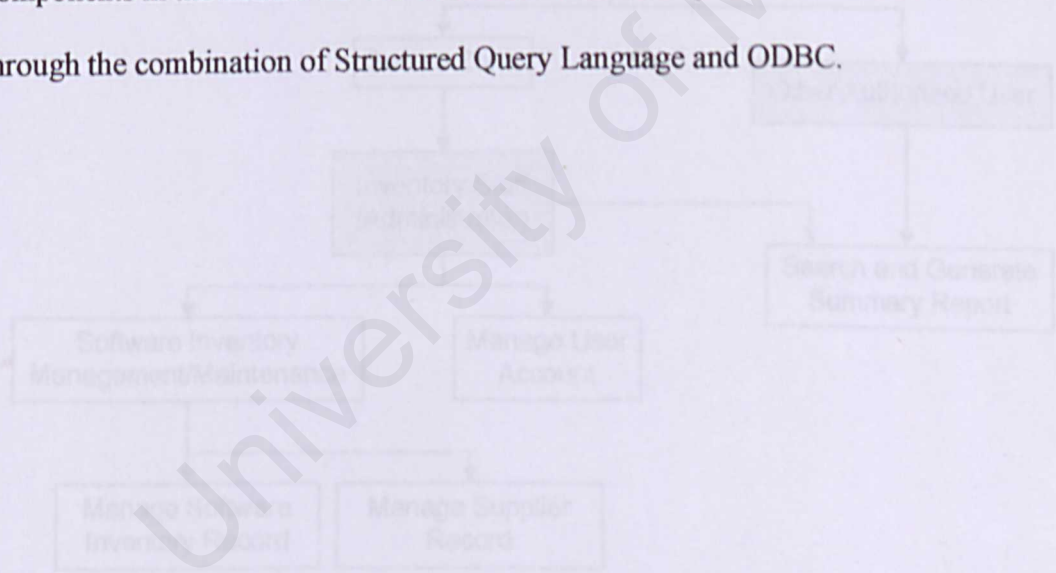through the combination of Structured Query Language and ODBC.

4.3   System Structure

We will start the system design phase by determining the System Structure for the proposed system. Structure chart is used to depict high-level abstraction of a specified system. It describes the iteration between independent modules. Major functions from the initial component part of the chart can be broken into detailed sub-component.

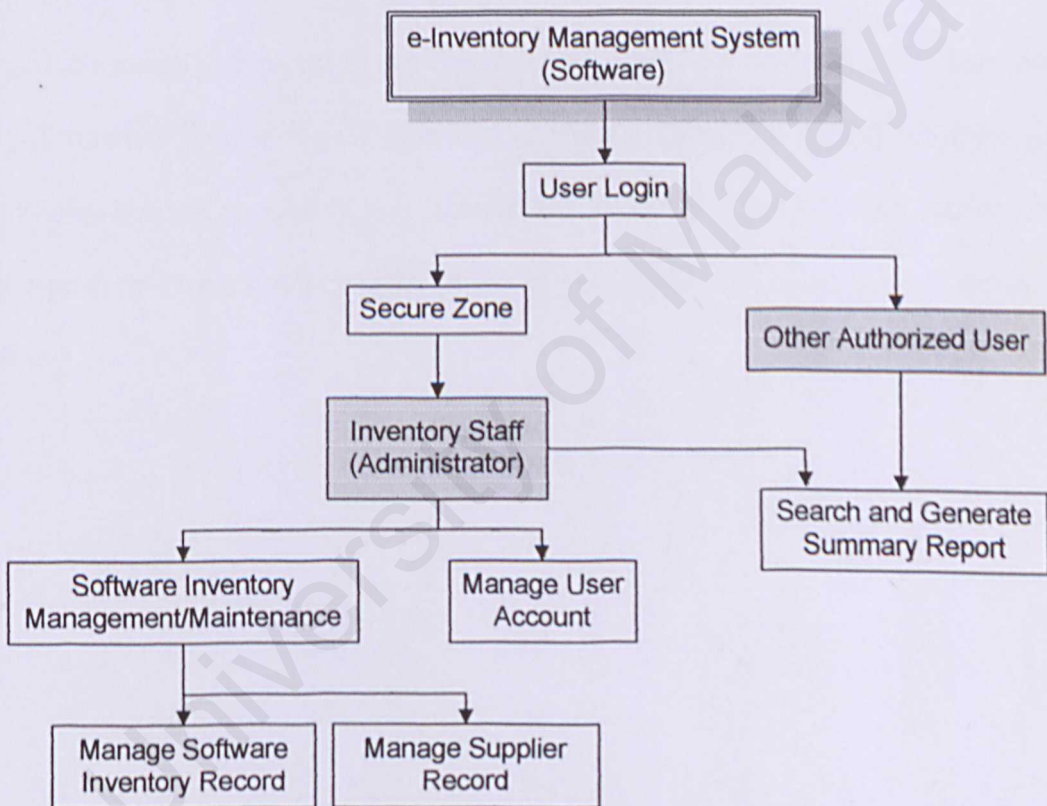The following Figure 4.2 shows the complete structure of the proposed e-Inventory Management System (Software).

Figure 4.2   System Structure Chart

The proposed system consists of two types of users that can use this system: the Inventory Staff (Administrators) and other authorized users. It is important to design the Inventory Staff to be in a *secure zone* to ensure that the data of inventory can only be modified by the people in inventory department. Under the secure zone, the

inventory staff may have sub-modules such as managing the user accounts, managing inventory or supplier record. Other authorized users can only retrieve information (inventory/supplier information), and to generate summary report.

4.4 Data Flow Design

Data Flow Diagram (DFD) of the system is based on computational concept such as functions. These data flow diagrams are used to show data is processed at different stages in the system and to show how the system reacts to internal and external events. DFD uses a number of fixed symbols to represent systems. DFD depicts the broadest possible overview of system inputs, processes, and outputs, which corresponded to data movement through the system.

A process model is depicted in the flowchart and data- flow model in DFD, functional transformation process their inputs and produce outputs. As data flows from one numbered process to another it is transformed. In order to simplify and clarify what the data flow diagram is portraying, there are supplemental conventions as shown in Table 4.1
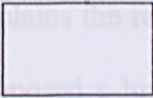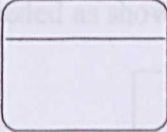
Symbol	Name	Description
	Entity	External sources or destinations of data. It interacts with system but is outside its boundary.
	Process	It represents the transformation or processing of information within a system
	Data Store	It is used for showing the data storage or referred by a process
→	Data Flow	It is used to show the movement of data from an origin to a destination with the head of arrow pointing towards the destination.

Table 4.1   Symbols using Gane and Sardon Method

Context Level Diagram (Figure 4.3) is the first level of Data Flow Diagram. It clearly explains the relationship and information/data flows among all the modules with the proposed e-Inventory Management System (Software). All the entities involve will have the access towards the process of e-Inventory and retrieve information they needed as shown in the diagram.
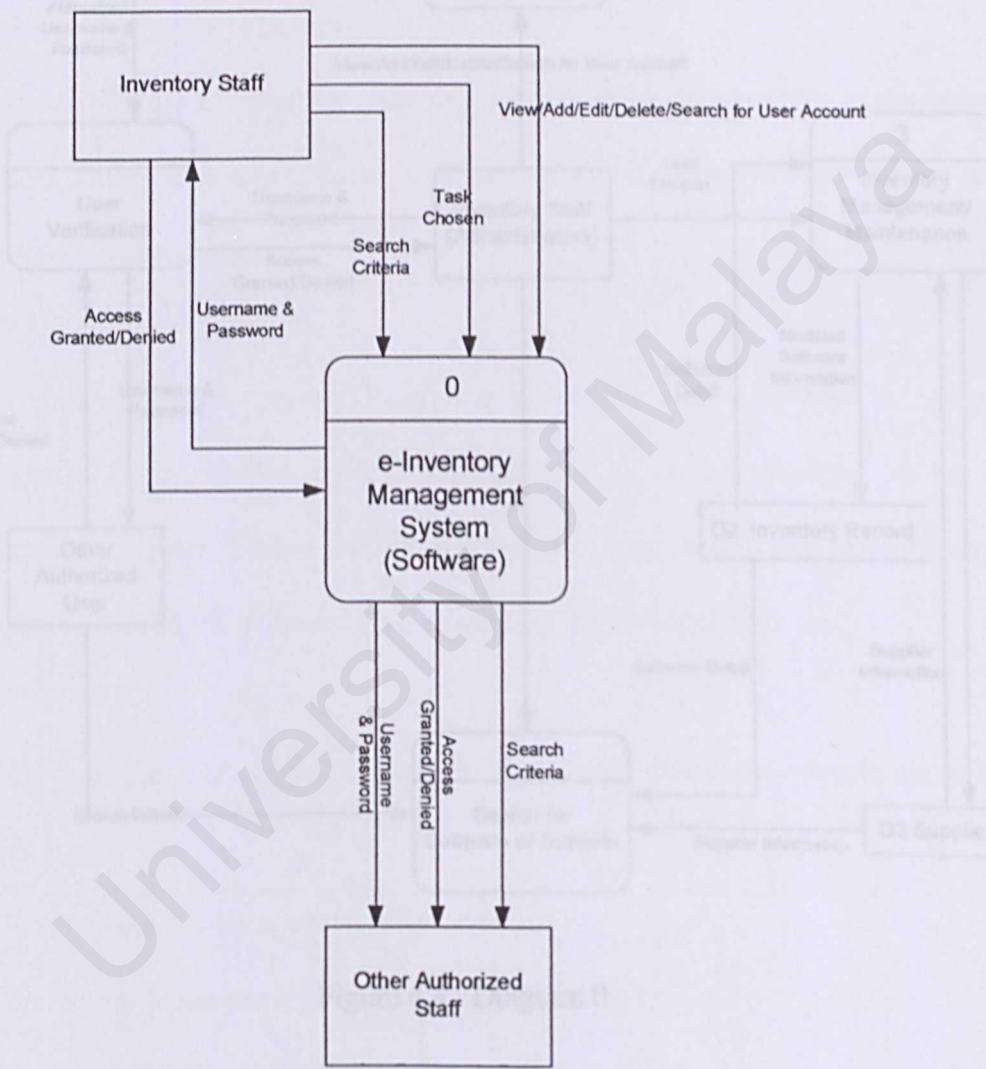

Figure 4.3 Context Diagram

Next is a Diagram 0 for this system (Figure 4.4). From this diagram, we can see how

the users from each module login to the inventory control system and carry out their

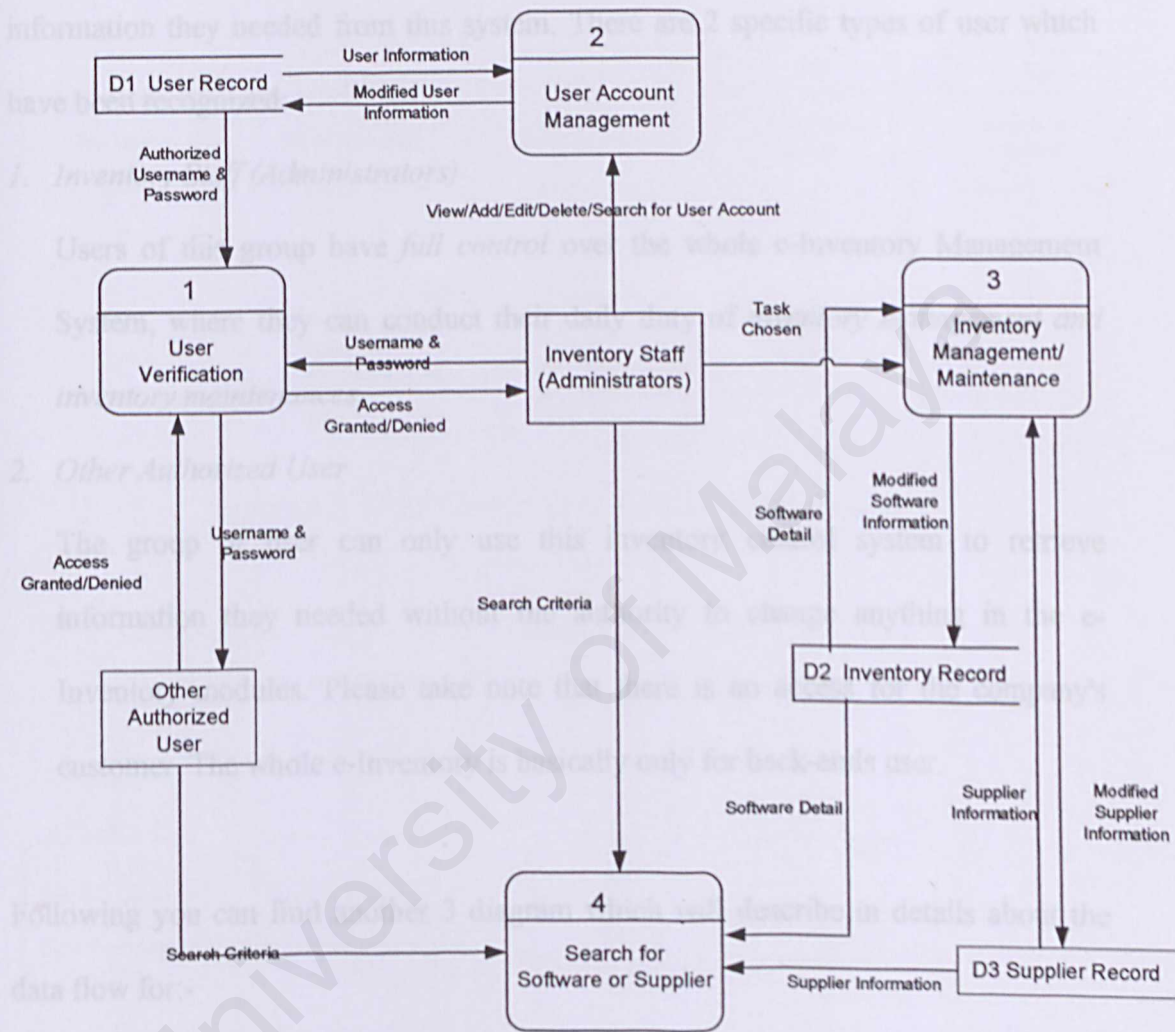own task in the different level of inventory functions

Figure 4.4   Diagram 0

The first phase of the data flow is **user verification.** This is necessary because e-Inventory acts like a back-end module for a company, so it only allow specific users to use this system, whereas other ordinary users can just retrieve some related information they needed from this system. There are 2 specific types of user which have been recognized:

1. *Inventory Staff (Administrators)*

   Users of this group have *full control* over the whole e-Inventory Management System, where they can conduct their daily duty of *inventory management and inventory maintenances.*

2. *Other Authorized User*

   The group of user can only use this inventory control system to retrieve information they needed without the authority to change anything in the e-Inventory modules. Please take note that there is no access for the company's customer. The whole e-Inventory is basically only for back-ends user.

Following you can find another 3 diagram which will describe in details about the data flow for:-

1. User Account Management (Figure 4.5)

2. Inventory Management/Maintenance (Figure 4.6)

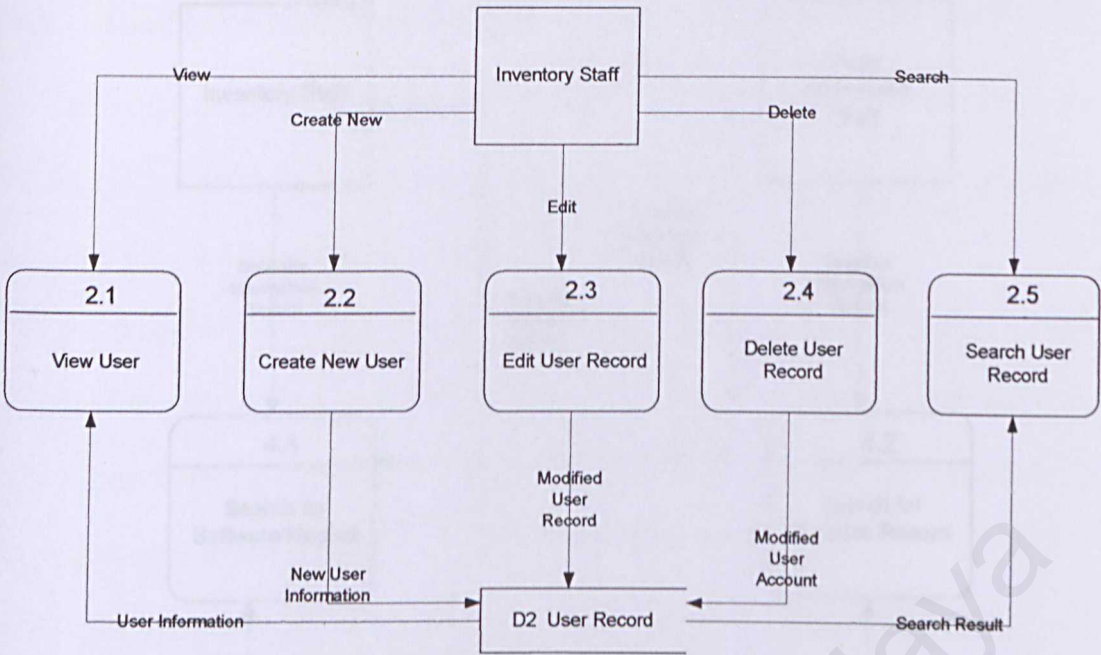3. Search and Generate Summary Report (Figure 4.7)

Figure 4.5   Data Flow of User Account Management
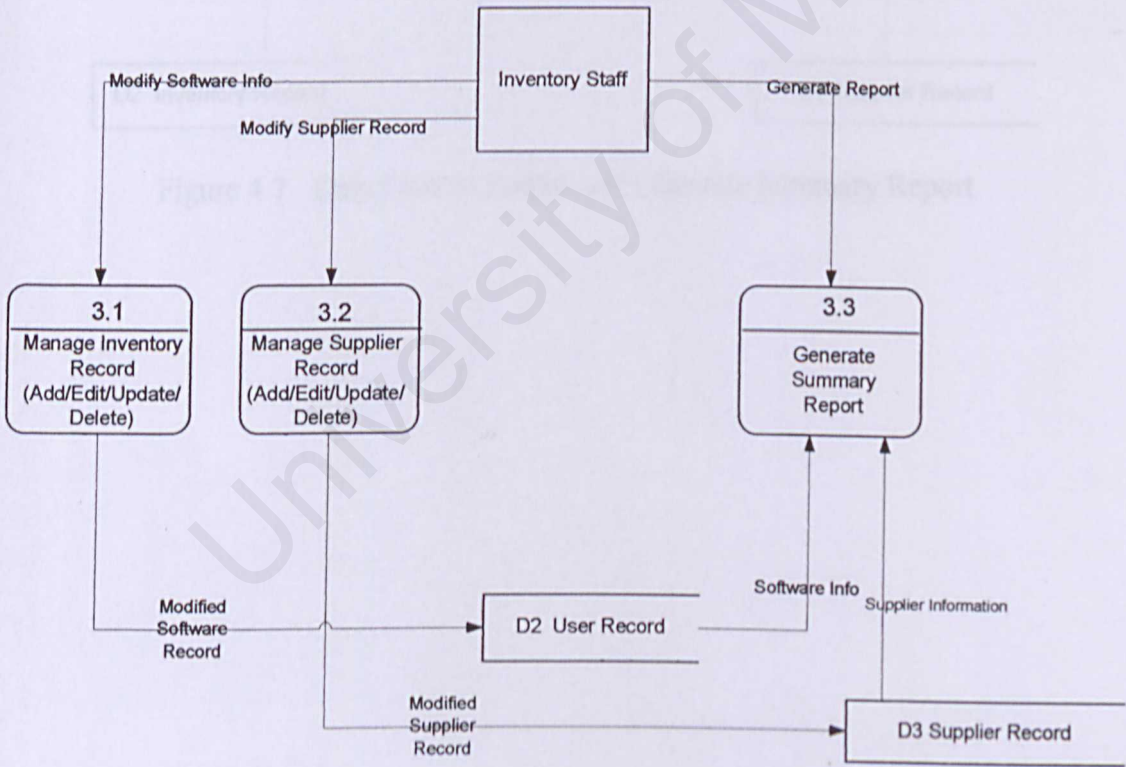
Figure 4.6   Data Flow of Inventory Management/Maintenance
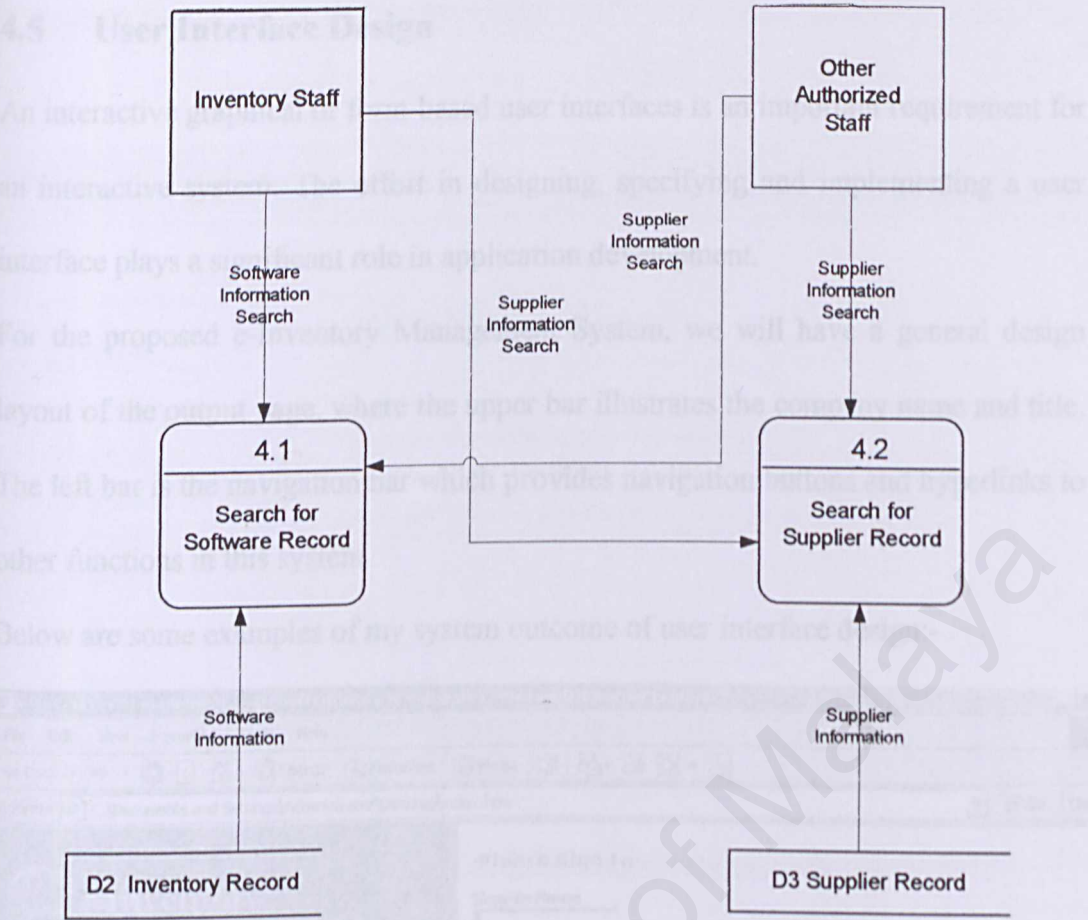
Figure 4.7   Data Flow of Search and Generate Summary Report

133

4.5   User Interface Design

An interactive graphical or form-based user interfaces is an important requirement for

an interactive system. The effort in designing, specifying and implementing a user

interface plays a significant role in application development.

For the proposed e-Inventory Management System, we will have a general design

layout of the output page, where the upper bar illustrates the company name and title.

The left bar is the navigation bar which provides navigation buttons and hyperlinks to

other functions in this system

Below are some examples of my system outcome of user interface design:-
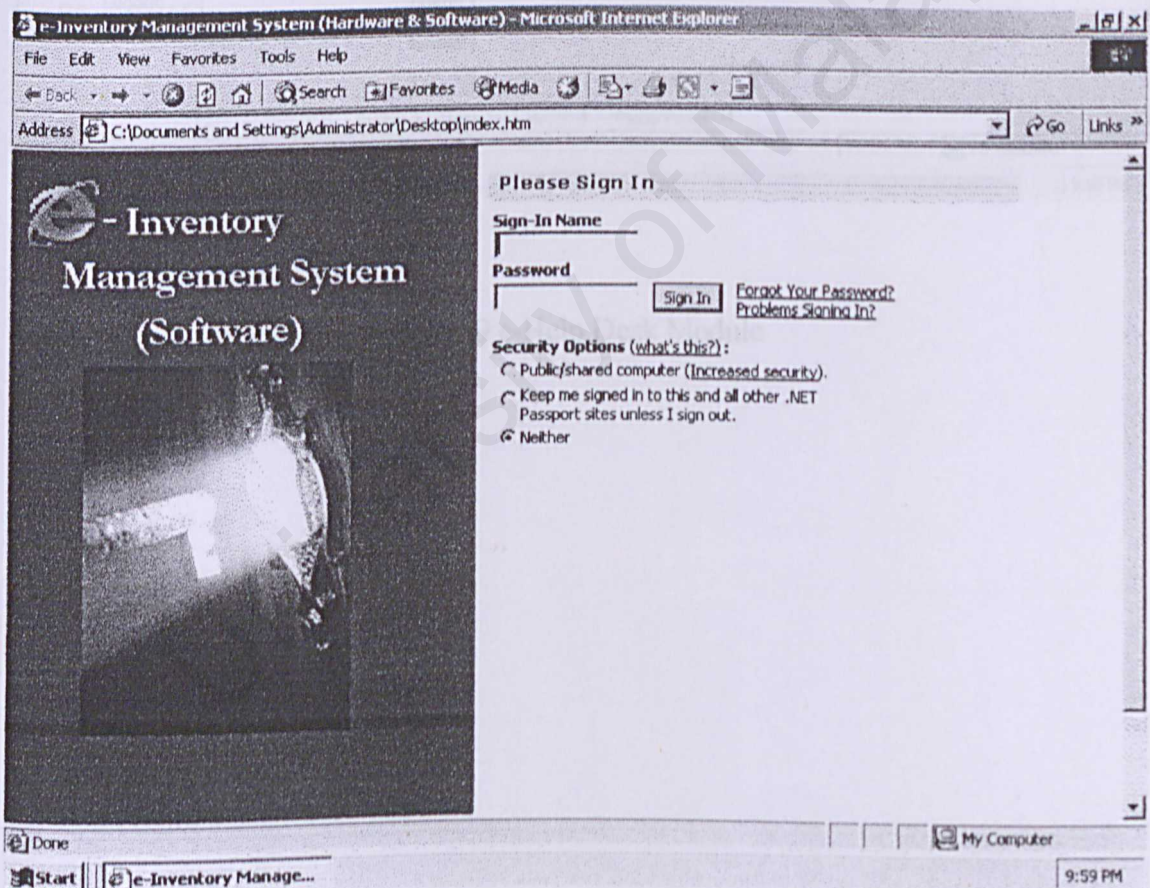
Figure 4.8 Login Module
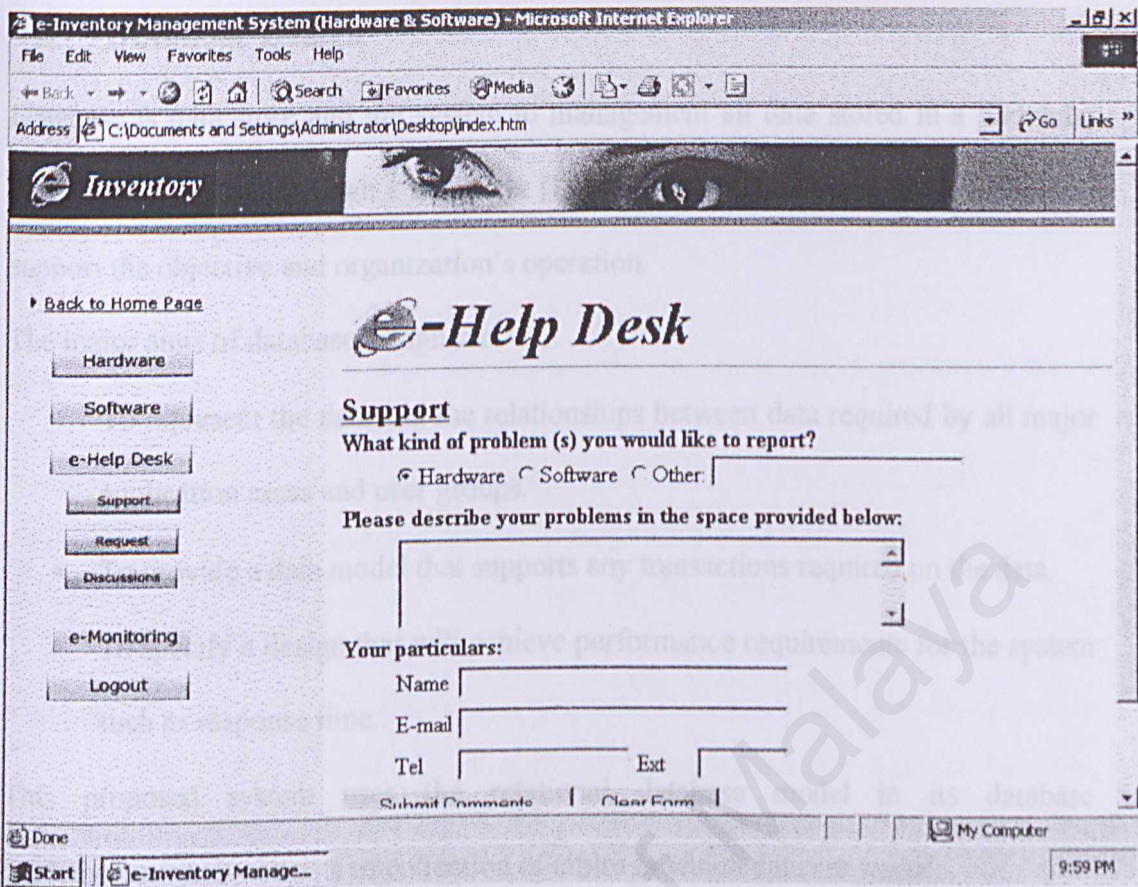
Figure 4.9 e-Help Desk Module

4.6   Database Design

Database is data store and the system to management all data stored in a particular format and accessed through a computer [18]. The design of the database model will support the objective and organization's operation.

The major aims of database design are:

- To represent the data and the relationships between data required by all major application areas and user groups.

- To provide a data model that supports any transactions required on the data.

- To specify a design that will achieve performance requirements for the system such as response time.

This proposed system uses the relational database model in its database implementation. It consists of collection of tables in which data are stored.

4.6.1 Data Requirement and Types

The following shows some of the Microsoft SQL Server data types which are most

commonly to be used in the proposed system.

Data Type	Description
Int	To store integers from -2,147,483,648 to 2,147,483,648
Float	To store floating number data from -1.79E + 308 to 1.79E + 308
Smalldatetime	To store dates range from Jan 1, 1900 to Jun 6, 2079
Char	Stores character strings, such as STATE abbreviations; for example, the length of the column will always be two characters long.
Varchar	Variable-length Unicode character data of n characters. n must be a value from 1 through 4000. Storage size, in bytes, is two times the number of characters entered. The data entered can be 0 characters in length.

Table 4.2   Microsoft SQL Server Data Types

Below you can find three major parts of data requirements for the proposed system

and these data are outlined in their tables respectively.

Attribute	Data Type	Length	Description
UserID	Varchar	8	User Identity Code
LoginID	Varchar	10	Login Identity Code
password	Varchar	10	Password
usergroup	Varchar	50	What type of user
Name	Varchar	2	User's Name
Position	Varchar	40	User's Position
Deptcode	Varchar	15	User's department code
OfficePhone	Varchar	4	User's Office phone
HandPhone	Varchar	5	User's Hand phone
Email	Varchar	12	User's email
LastLogin	Datetime	1	Phone number

Table 4.3   User Record Table (D1)

Attribute	Data Type	Length	Description
Code	Varchar	12	Item code
Tag	Varchar	13	Item tag
Parts	Float	8	No. of parts
Name	Varchar	40	Item Name
Brand	Varchar	15	Item Brand
PONo	Varchar	8	PO number
PODate	smalldatetime	4	PO Date
Dept	Varchar	4	Department Code
Room	Varchar	5	Room Code
Supplier	Varchar	20	Supplier
SerialNo	Varchar	12	Serial Number
Price	money	8	Price
Status	Varchar	1	Item status
Sponsor	Varchar	1	Sponsor
Remark	Varchar	12	Remark
Entry Dte	smalldatetime	20	Entry Date

Table 4.4   Inventory Record Table (D2)

Field Name	Data Type	Size	Description
Name	Varchar	11	Supplier Name
Street	Varchar	2	Street
City	Varchar	40	City
PostCode	Varchar	15	Post code
State	Varchar	4	State
Country	Varchar	5	country
ContactPerson	Varchar	12	Contact Person
Telephone	Varchar	1	Phone number
Fax	Varchar	12	Fax number
Email	Varchar	20	Email of contact person
Website	Varchar	100	Supplier website
Remark	Varchar	20	Remark

Table 4.5   Supplier Record Table (D3)

4.7   Expected Outcome

A web-based e-Inventory Management System that will provide an efficient and easy to use system that can carter all parties, including the company, supplier and customers.

4.8   Summary

This chapter has given the overview of the e-Inventory Management System (Software).

It discusses the system design in details.

From this chapter, we can conclude that the job sequence of developing a new system is

extremely important. In order to achieve a good outcome of the project, we must follow

the development path before going into the coding phase (which will be discussed in the

coming chapters). Thus the first step is to decide on the system architecture, as has been

discussed in Section 4.2.

After that, in Section 4.3 and Section 4.4, I have started the structure and Data Flow

design of the system. When designing a new system, data flow must be defined before

other components. It is used to show the flow of the information/data in the whole

system. While in section 4.5, several outputs of user interface design was shown. (*Figure*

4.8, 4.9)

Data Dictionary is then created after the data flow diagram to identify all the necessary

attributes of the database in Section 4.6 of this chapter.

Chapter 5: System Implementation

5.1 Introduction

The requirement analysis, methodology and system design phases do not have a clear boundary in a software project. Each phase tends to overlap one another. System implementation is a process that converts the system requirements and design into program codes.

5.2 Development Environment

Development environment has certain impact on the development process of a good software system. The suitability of the hardware and software chosen is very important because it will not only help to expedite the system development but determine the success of the project. The hardware and software tools used to develop the entire system are as below:

5.2.1 Hardware requirements

The hardware used to develop the system is listed:

- Pentium III 500MHz or equivalent
- Memory-64Mb RAM and above
- 5GB hard disk

5.2.2 Software Tools Requirements

Table 5.2 illustrates the software tools used to develop the entire system:

Software	Purpose	Description
Microsoft Windows XP	System Requirement	Operating System [Final Stage]
Microsoft Internet Information Service	System Requirement	Web Server host [Final Stage]
Microsoft .NET Framework	System Requirement	.NET development environment [Final Stage]
Microsoft ASP.NET Web Matrix	System Development	Coding the web pages
Microsoft Internet Explorer 6.0	System Development	Web Browser
Microsoft SQL Server 2000	System Development	Database Design and Database Server
Macromedia Dreamweaver MX	Interface Design	Interface design and creation
Microsoft FrontPage	Interface design	Interface modification
Microsoft Visio 2002	System Design	Diagram Creation [Earlier stage]

Table 5.2: Summary of software tools used

5.3 System Development

System development is the process of creating programs that are needed to satisfy requirement of the system.

There are two approaches in the developing process, which are top-down and bottom-up. The top-down approach is a way of building the high-level modules as backbone that are used to call the lower-level modules that contains of functions and events, whereas the bottom-up approach is the reserve.

Software e-Inventory Management System was developed using both approaches. The reason of using top-down approach is to involve separating and determining the high-level modules that are to be refined into functions and procedures. When the outline has been done, we used the bottom-up approach to finish the system development process. The reasons and advantages of using bottom-up are testing can begin on some of the modules while others are still being developed. Some critical functions can be donned first to test their efficiency.

Besides, system development consists the used of methodology chosen, web-based development tools, web-pages coding and database connection. The details are illustrated as below:

5.3.1 Methodology

This project is developed using the waterfall methodology. The development of this project will consist of five stages, which are requirement, design, coding, testing and operation. The system is design using logical flow and it allows the estimation of the milestones. Each stage must be completed before proceed to the next stage to ensure that the system is built according to the requirement and specifications.

5.3.2 Web-Based Development Tools

Microsoft ASP.NET is used as the main development tool for this project. This tool enables easy performance of the many complex programming and database tasks required in the creation of a web site as well the incorporation of HTML, formatting and Layouts, graphics and other multimedia components.

All the buttons, graphics and animations in this project are created using Macromedia Dreamweaver MX, Microsoft FrontPage and Microsoft Visual Basic.NET. The Data Flow Diagram are created using Microsoft Visio 2002.

5.3.4 Database Connection

The database for this project is created using Microsoft SQL Server 2000. The processes of creating database, creating and modifying tables and their relationship is made easy.

5.4 Program Documentation

Program documentation is a set of written description that explains to a reader what the programs do and how they do it. [16]

5.4.1 Naming Conversions

Naming convention is an abbreviation of the control name or the object name. This system uses a naming convention to ensure uniformity of the control and object names. The purpose of this naming convention is to increase the readability of the codes. For example:

Control	Control Name	Example
Command Button	cmd_	cmd_Open cmd_Close
Text Box	txt_	txt_stkCode

5.4.2 Internal Documentation

Internal documentation is a description material written directly within the code [16]. It means that internal documentation refers to comments within the codes. This is needed to enhance readability of the code by someone other than the programmer. It will also help the programmer to recall the function or meaning of certain codes.

For HTML, the comment tag is illustrated below:

<-This is a comment->

Any text inside this pair of symbol will be ignoring, and the comment text will be in grey color.

For Visual Basic.NET, the comment tag is illustrated below:

'This is a comment

Any text after this symbol 'will be ignored during execution time. All the comments are in green color.

Codes are also formatted to enhance understanding. Spacing or line brake in between different section of the codes will enhance readability too.

Module	Sub-Modules	Description	Functionality
e-Inventory Administrator Level	View Inventory Record	To view all the inventory record	An easy way for manager of inventory to search and view all related information for inventory record
	Add Inventory item	To add a new inventory item to the record	Allow administrator to add new inventory record
	Edit Item	To edit the existing item	Allow administrator to edit the list of the inventory record
	Delete Item	To delete the unnecessary item	Allow administrator to delete the inventory item
	View all vendors	A list of vendors display for administrator	Allow administrator to view the vendor information
	Vendor Maintenance	An Editable data table for maintenance	Allow Administrator to edit and delete the vendor information
	User record management	An editable table of user information	Only the authenticated administrator can update and delete the user information
	Register a new user	A registration form for adding new user	Allow Administrator to add new user to the system

148

5.5 Description of the System Modules and Functionalities

Basically e-Inventory can categorized into 3 modules, and inside each module there are sub-modules, which will be responsible for all the functionalities. The table below shows the description on each module and also its functionalities. (*The 5.5a: Description of System Modules & Functionality*)

Module	Sub-Modules	Description	Functionality
e-Inventory Administrator Level	View Inventory Record	To view all the inventory record	An easy way for manager of inventory to search and view all related information for inventory record
	Add Inventory item	To add a new inventory item to the record	Allow administrator to add new inventory record
	Edit Item	To edit the existing item	Allow administrator to edit the list of the inventory record
	Delete Item	To delete the unnecessary item	Allow administrator to delete the inventory item
	View all vendors	A list of vendors display for administrator	Allow administrator to view the vendor information
	Vendor Maintenance	An Editable data table for maintenance	Allow Administrator to edit and delete the vendor information
	User record management	An editable table of user information	Only the authenticated administrator can update and delete the user information
	Register a new user	A registration form for adding new user	Allow Administrator to add new user to the system

Module	Sub-Modules	Description	Functionality
	Change Password	A list of field that validate the user password hint	Allow administrator to change password
	Quick Search	A drop down box that perform as search function	Allow administrator to search by inventory, user and vendor just with a single click

Module	Sub-Modules	Description	Functionality
e-Inventory User Level	View Inventory Record	To view all the inventory record	An easy way for user of inventory to search and view all related information for inventory record
	Add Inventory item	To add a new inventory item to the record	Allow user to add new inventory record
	Edit Item	To edit the existing item	Allow user to edit the list of the inventory record
	Delete Item	To delete the unnecessary item	Allow user to delete the inventory item
	View all vendors	A list of vendors display for administrator	Allow user to view the vendor information
	Vendor Maintenance	An Editable data table for maintenance	Allow user to edit and delete the vendor information
	Change Password	A list of field that validate the user password hint	Allow user to change password
	Quick Search	A drop down box that perform as search function	Allow user to search by inventory, user and vendor just with a single click

5.6 Summary

This Chapter shows the details on the system implementation, which include: development environment, system development methodology, system development tools, program documentation, and description of system modules and functionalities.

After a brief introduction for system approach at section 5.1, the development environment and system implementation requirement was describe in section 5.2 where the entire hardware and software requirement were listed out.

Then I have discuss about the system development approach at section 5.3, where we are using top down approach for determining the high-level modules dependencies, then we used the bottom-up approach to finish up the modules functionalities. Besides, I have also described about the programming method I am using for developing this system.

In section 5.4, the program documentation was explained and also some examples were shown.

Lastly, in section 5.5, all the system functionalities for each module were summarize in the tables.

Chapter 6: System Testing

6.1 Introduction

Software testing is one of the main phases in the Waterfall Life Cycle model. In this phase, the process of testing and debugging are done to detect and bugs of a system. These processes are usually done incrementally with system development.

This phase is also often referred to as Verification and Validation (V & V). Verification refers to the set of activities that ensure the software correctly implements a specific function. Validation refers to different set of activities that ensure the software has been built is traceable to user requirements. A successful tests us one is which no errors are found.

The objectives of software testing are [32]:

- To reveal different classes or errors and do so with minimum amount of time and effort
- To demonstrate that software functions appear to be working according to specification
- To demonstrate that behavioral and performance requirements appear to have been meet

In addition, data collected as testing is conducted provide a good indication of software reliability and some indication of software quality as whole. However, testing cannot show the absence of errors and defects, it can show only that software errors and defects are present [32].

6.2 Testing Planning

Above all, a test plan for e-Inventory Management System has been draw as a guideline in designing and organizing tests, so that testing is carried out appropriately and thoroughly.

The test plan includes the following steps:
1. Establishing test objectives
2. Designing test cases
3. Writing and Testing test cases
4. Executing tests
5. Evaluating test results

This project was tested with following generic characteristic:
- Testing begins at module level and works "outward" toward the integration of the entire system.
- Different testing techniques are appropriate at different points in time
- Testing and debugging are different activities, but debugging must be accommodated in any testing strategy

The testing process that has been carried out for this project is explained as following sections:

6.3 Unit Testing

Unit testing focuses verification effort on the smallest unit of design which is the software component or module. All the important control paths in this project are tested to uncover errors within the boundary of the module. The relative complexity of test and uncovered errors is limited by the constrained scope established for unit testing. The unit test is usually white-box oriented and the step can be conducted in parallel for multiple components.

The test that occurs as part of unit tests is illustrated schematically in Figure 6.1. The module interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure us examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm's execution. Boundary conditions are tested to ensure that the module operate properly at boundaries established to limit or restrict processing. All independent paths (basis path) through the control structure are exercised to ensure that all statements in a module have been executed at least once. Finally, all error-handling paths are tested.

Step	Test Procedure	Expected Output	Test Result Analyzing
1	Add a new record information with a unique stock Code.	The record is inserted permanently	Record is increased successfully.
2	Insert the specific	The information for that specific record was selected and showed to the user	Record is shown successfully and correctly.
3	Press Delete button to delete the selected stock	The record is deleted permanently from table	The record with that unique stock code is

The following areas were tested during unit testing for this project:

a) Boundary value analysis

Ensure that the module operates properly at boundaries established to limited or restrict processing.

b) Error handling paths

Ensure that the specific module executes the recovering process while an error occurs. For example, the updating process should be able to continue to function again after encountering duplicate record in the database.

c) All possible independent program paths are executed

Ensure that the control structures are implemented correctly.

6.3.1 Unit Testing Example

a. **Unit Test Case**

Each table in server database has at least associated with two trigger programs, which are Record Adding, and Record Deleting. Unit testing was carried out on each trigger program once it was completed. Table below shows the test case for unit testing on the Stock Record Adding and Deleting trigger program: (Table 6.3.1a)

Step	Test Procedure	Expected Output	Test Result Analyzing
1	Add a new Record Information with a unique stock Code.	The record is inserted permanently.	Record is increased successfully.
2	Insert the specific	The information for that specific record was selected and showed to the user	Record is shown successfully and correctly.
3	Press Delete button to delete the selected stock	The record is deleted permanently from table:	The record with that unique stock code is

	record	Stock record and also table: serial number	deleted permanently from table: Stock record and also table: serial number.
			Objective of delete cascading is achieved.

Table 6.3.1.a: Test case for Record Deleting Trigger Program

6.4 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.

6.4.1 Module Testing Example

After all of the trigger programs for certain table were developed, module testing was carried out to ensure the module functioning as expected. Table below shows the test case for module testing on the trigger programs for table: InventoryRecord. (Table 6.4.1)

Step	Test Procedure	Expected Output	Test Result Analyzing
1	Click Login Button to login into e-Inventory as a staff of the company	Link to the correct web page	The correct web page was shown successfully.
2	Click 'Create New Record' to insert new inventory record.	Link to the inventory record insertion form.	The correct web-form was shown successfully.
3	Finished insert all stock information and click 'Add' button to	The records are inserted permanently.	Records are inserted successfully.

	create record.		
4	Click the 'home' button to go back to staff	Link to the correct web page.	The correct web page was shown successfully.
5	Click 'Display record' to view the record inserted	The record from table: InventoryRecord was selected and showed to the user.	The correct web page together with new record was shown successfully.
			Objective of integration all trigger programs for table: InventoryRecord is achieved

Table 6.4.1 Module Testing Example

6.5 Integration Testing

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with the interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. This testing will ensure that the section in this project, which are call module, to be arranged correctly.

The approach used in this phase is an incremental strategy, the bottom-up integration and regression testing. The incremental integration is the antithesis of the high bang approach. This web application is constructed and tested in small increments where errors are easier to isolate and correct. All the interfaces are tested completely and a systematic test approach is applied.

For e-Inventory project, the integration was using bottom-up approach. Bottom-up integration testing begins construction and testing with modules at the lowest levels of the system and then moving upward to the modules at the higher levels of the system. Regression testing is the re-execution of some subset of tests that already been conducted to ensure that changes have not propagated unintended side effects. It is the activity that helps to ensure that changes (due to testing or for other reason) do not introduce unintended behavior or additional errors.

6.6 Validation Testing

At the culmination of integration testing, this project is completely assembled as a package, interfaces and linking errors have been uncovered and corrected. A final series of software tests that is the validation testing has been carried out during this phase.

Software validation is achieved through a series of black box tests that demonstrate conformity with requirements. For this project, a test plan has outline the classes of tests to be conducted, and also a test procedure is used to defines specific test cases that will be used to in an attempt to uncover errors in conformity with requirements. Both the plan ad procedure is designed to ensure that:

- All functional requirements are satisfied
- All behavior performer requirements are attained
- Documentation is correct
- Other requirements are met(e.g. error recovery, maintainability, compatibility)

Alpha test and Beta test are also being carried out to uncover errors that only the end-user seems able to find. Alpha test is conducted at the developer's site by an end-user in a controlled environment. Beta test is conducted at one or more customer sites by the en-user of the software and it is a "live" application of the software.

6.7 System Testing

System testing is a series tests designed to fully exercise the software system to uncover its limitations and measure its capabilities. The objective is to test an integrated system and verify that it meets specified requirements. Although each test in this project has a different, all work to verify that system elements have been properly integrated and perform allocated functions.

There are several types of system testing that are worthwhile for a software system. For this project, three types of system testing are used:

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

6.7.2 Security Testing
These system tests will attempts to verify that protection mechanism built into the system will protect it from improper penetration.

6.7.3 Performance Testing
The purpose of this testing is to test the run-time performance of software within the context of an integrated system. It requires both hardware and software instrumentation. Resource utilization is measured in an exacting fashion.

Types of performance testing carried out for e-Inventory (Software Management System) are:

- Volume tests
  The fields and records are checked to see if they can accommodate all expected data.

- **Security Measurement tests**
  This test ensures that the application fulfills the security requirements.

- **Human factor tests**
  Display of the web page and messages are examined to determine user friendliness.

6.8 Summary

This chapter is all about testing, for we have the unit testing, modules testing, integration testing, validation testing and system testing.

The first thing we to do for testing is to build a testing planning to ensure testing are carried out appropriately and thoroughly. The testing steps for e-Inventory (Software Management System).

The testing done for e-Inventory has been summarizing within this chapter. The testing dine were: unit testing (*refer section 6.3*), modules testing (*refer section 6.4*), integration testing (*refer section 6.5*), validation testing (*refer section 6.6*), and finally the system testing which included recovery testing, security testing and performance testing (*refer section 6.7*).

Through all the testing phases, it is easier to ensure the system qualities and strength. Debugging and fixing of the program coding can be done but there is still some system's limitation that is not able to fix due to lack of time.

As conclusion, the testing phase is a very important phase in the System Development Life Cycle (SDLC), and it must be done in repeat and carefully to assure of good quality system output.

Chapter 7: System Evaluation and Conclusion

7.1 Introduction

All of the objectives stated in e-Inventory have been achieved successfully. The e-Inventory Software Management System is developed to achieve the following objectives:

- ✓ To enable the control and management of inventory to be done easily via web technology.
- ✓ To ensure the sufficiency of inventory.
- ✓ To improve the quality and accuracy in data keeping.
- ✓ To manage better transactions activities with vendors.

During the period of coding and implementation of this system, various problems were encountered. These problems were solved through research and studies in fields such as the Internet, ERP websites and ASP.Net forums. Besides that, a lot of system analysis has been done on technological and programming concepts to grasp the concept of Internet programming. The system's strengths, limitations and future enhancements were identified.

7.2 Problems Encountered and Solutions

There are several problems encountered throughout the development of this project.
These include:

7.2.1 Difficulty in choosing a suitable development technology, programming language and tools

There are many software tools available to develop an e-Business solution system.
Choosing suitable technology and tools was a critical process as all tools possesses
their own strengths and weaknesses. In addition, the availability of the required tools
for development was also a major consideration.

In order to solve the problem, seeking advises and views from project supervisor,
course mates and even seniors engaging in similar project were carried out.
Furthermore, a lot of research and studies were done before any decision was made.

7.2.2 Lack of knowledge in ASP.net Web Matrix

Previous knowledge in traditional, two-tier non web-based system does not seem to
solve the lack in web-based system. Since there was no prior knowledge of
programming in ASP.net Web Matrix and VBScript, there was an uncertainty on how
to organize the codes in a web page. These programming languages and concepts
were never taught before and to implement such as application requires a fair grasp of
the languages. These programming approaches seem to be totally different from
traditional programming languages.

Although it really cause a lot of time to learn the new technology, but choosing to
program in ASP.net Web Matrix and VBScript proved to be a wise move. Most of the
problems faced were manageable through browsing the Internet for related materials
such as online ASP.net forum and tutorial. Besides, discussion with friends especially
course mates using the same technology was a great help. A more efficient method
was through trail and error during the coding phase.

7.2.3 Readability problem in ASP.net

Almost all web-base programming languages do not support for variety types of variables. It decreases the readability for the programmer. Problems become more tedious when there are bugs detected in the program. Since the readability of the program is decreased, then the code maintainability also decreased. It is very difficult to debug the errors especially semantics errors.

In order to minimize the problem in this area, a series of testing mist be taken if the components are to be used in implementation of this project. Minimization in using the component is need for a stable system.

7.3 System Strengths

During the development of this project, several system strengths were identified and described as follows:

7.3.1 Provide an easy to use tool

The commands and the layout of this project are simple and well organized, therefore it is easy to use, simple to learn and understandable. Normal users with some computer knowledge will find the web page is easy to handle.

7.3.2 User friendliness and easy to use interface

Some useful Graphical user Interface (GUI) such as command buttons; drop-down list boxes and record sorting button are provided which attract the users to navigate through the system and give faster access. This user-friendly interface can shorten the learning curve and reduce training costs, which include money and time. The menu-driven is built to facilitate the individual needs of the users.

7.3.4 Provide database access

All the data are organized and stored in the form of database using Microsoft SQL Server 2000. It is real-time database information and any changes made to the records can be updated instantly to the e-Inventory database.

7.3.5 Security Features

Security issues are taken into consideration for the each and every module so as to prevent any unauthorized users from manipulating the data stored in the database. The maintenance module is protected with password and specific login ID. Only the authorized person is allowed to update and add data into the database.

7.3.6 System Transparency

System transparency refers to the condition where the users do not need to know the underlying structure of the system, where the database resides, and the database management system and anything related to the system design. For example, information retrieval and downloading of records are similar to a system accessing the local database. This is done to ensure that user are not confused when retrieval information.

7.4 System Limitations

Owing to the time constraint and the constraint of the programming language itself, there were some limitations in this e-Inventory system. These include:

7.4.1 Browser Limitation

e-Inventory system can only run in Internet Explorer 6.0 and above. e-Inventory system requires a browser that can understand VB.net script, the default supporting language for ASP.net. User uses browsers that do not support these features will not be able to use the functions available in this system.

7.4.2 Lack of software information

The second limitation of this system is the lack of related useful information for reference in the menu.

7.4.3 Lack of Firewall

The third limitation of this system is the lack of use of firewall. Firewall normally is used to create security checkpoints at the boundaries of a private network. By providing the routing function between the private network and the Internet, firewall inspect all communications passing between the two networks and either pass of drop the communications depending on how they match the programmed policy rules. However, Firewall is not implemented together with this system due to some circumstances.

7.5 Future Enhancement

Further development and many new ideas have come about while the system was being implemented. Owing to time constraint and other factors, not all of the ideas could be incorporated into the system. It is hoped that the following aspects could be considered in future.

7.5.1 Extent the ability of browser

As stated, e-Inventory requires Internet Explorer 6.0 or above for execution. In future, this system can be turned to fulfill other browser requirements such as Netscape Navigator for execution. This is because Netscape has a sizeable share in the browser market besides Internet Explorer and Netscape has a lot of users in the world.

7.5.2 Implementation of Firewall

Firewall must be implemented if the system will support real-time online transaction in the future. This is very important because the connection between the system and the local banks must be kept as secure as possible.

Firewall can be considered a border security because it will inspect and then approve or reject each connection attempt made between the system and external networks. It also allows the administrator to centralize all external security services in a machine that are optimized for and dedicated to task.

7.5.3 Other Language Support

Future enhancement for this system will include other language support. This will enable information to be displayed in different languages such as Mandarin, Japanese, French and others. This will broaden the usage of the system as the system will be used by the users form different countries with different national languages.

7.6 Summary

Overall, this project has achieved and fulfilled the objective and requirements as an e-Inventory application that provides the opportunity of managing company's Inventory over the Internet over the Internet. This completion of this project will also encourage Malaysia small and medium sized companies (SMBs) to accept electronic business as an integral part of their daily lives.

A lot of research and studies was done during literature review in order to gain the information needed for the development of this project. The information gathered includes knowledge on E-Business and Enterprise Resource Planning (ERP), inventory management concept and system example, development methodologies, and development tools, all these information were analyzed to determine the suitability of the project scope, system functionalities, development methodology and development tools to be implemented and used in this project.

Throughout the development of this system, a lot of knowledge was gained such as knowledge in setting up Internet technologies, Microsoft SQL Server 2000. Programming in ASP.net, HTML, and VB.net proved to be a valuable experience. The core of the ASP.net is the implementation of object-oriented technology. As such, the object-oriented programming still has also improved.

Above all, this project gives me an opportunity to build a full application from scratch. It is a great challenge to develop an integrated system in a group. From this project, I have gained knowledge and experience during the progress of it. The great attitude and cooperation from my partner has brought this system to a success.

References

[1]	Internet Society (ISOC) Al 1 About the Internet A Brief History of the Internet http://www.isoc.org/internet-history/brief.html#Introduction
[2]	Gary P.Schneider & James T.Perry. Electronic Commerce. Course Technology, 2000
[3]	Grant Norris, James R.Hurley, Kenneth M.Hartley, John R.Dunleavy, John D.Balls. E-Business and ERP: Transforming the Enterprise. John Wiley & sons, 2000
[4]	Jerry Chin. E-Commerce Success: Building a Global Business Architecture. Computer Technology Research Corp, 1999
[5]	Introduction on E-Business and ERP http://www. export-edge, com/ebgoing. htm#overw
[6]	ERP White paper. Ward Cobleigh, Director of Business DevelopmentGanymede Software Inc, 1999
[7]	Spencer B. Smith. Computer Based Production and Inventory Control, 1989
[8]	Arthur J.Kweon, J.William Petty, David F.Scott Jr., John D.Martin. Foundations of Finance: The Logic Practice of Financial Management. Prentice Hall 1998
[9]	Richard J. Tersine. Principles of Inventory and Materials Management. North Holland, 1982
[10]	Edward A.Silver, David F.Pyke, Rein Peterson. Inventory Management & Production Planning and scheduling. John Wiley & sons 1998
[11]	www.epicor.com http://www.epicor.eom/solutions/e/ebackoffice/distribution/
[12]	www.alcie.com http://www.alcie.com/newstyle/lnventorySpec.html
[13]	Deitel & Deitel, T.R. Nieto. Visual Basic.NET: How To Program. Prentice Hall, 2002
[14]	www.accutrakinventory.com http://www.accutrakinventory.com/setservices.html

[15]	Atrex Inventory Software http://www.1000years.com/hardware.shtml http://www. 1000years.com/atrex.shtml
[16]	L.S. Pfleeger. Software Engineering Theory and Practice. Prentice Hall, 2001
[17]	E. Visser. Waterfall Model:Characteristic. http://ftp.cs.uu.nl/~visser/teaching/../, 1999
[18]	Kendall & Kendall. System Analysis and Design 4th Edition. Prentice Hall, 1999
[19]	O. Drori. Planning and Design of Information System using OODPM. http://shum.huji.ac.il/~-offerd/paprs/oodpmtxt.htm, 1996
[20]	Detailed Information About Apache http://webcompare.internet.com/cgi-bin/detail.pl? 03_Server=Apache
[21]	Detailed Information About Internet Information Server 4.0 http://webcompare.internet.com/cgi-bin/detail. pi? 03_Server=Internet Information Server
[22]	Red Hat Linux Documentation CD, www.redhat.com
[23]	S.Milo, 2000b Linux
[24]	SQL Server 7.0 Database Implementation Training, Microsoft Press, 1999
[25]	ICLISE. ICLISE Knowledge Portal. 13 June 2001. 5 July 2001. http://myplant.communityzero.com.iclice
[26]	PC.COM (April issue 2002) http://www.pcdc.com.my
[27]	Internet and world wide web-How to program DEITEL™ (Deitel Deitel Nieto)
[28]	THE UNION INSTITUTE RESEARCH ENGINE. What is the Literature Review? N.d. 15 June 2001 http://tui.edu/Research/Resouces/ResearchHelp/LitReviewLgPg.html
[29]	CAMBRIGDE INTERNATIONAL DICTIONARY of ENGLISH Cambridge University Press
[30]	Microsoft Press http://www.mirosoft.com

| [31] | Sun Microsystems. Notes for Servlets JSP and ASP, N.d 20 July 2001. http://www.cpsc.ucalgary.ca/~heerema/presentation/jsp.htm |
| [32] | Pressman, R.S., Software Engineering: A Practitioner's Approach. New York, Mcgraw Hill, 1992 |

Table of Content

User Manual

Table of Content

List of Figures

Chapter 1: Introduction

1.1 Welcome to Software e-Inventory Management System

e-Inventory is an integrated system used to manage a company daily software transaction and also company user record. The e-Inventory contain of four major modules: Software management, Vendor management, User management and Quick search function. The following chapters will show a complete user manual for e-Inventory. The e-Inventory provides an effective software controlling and managing method to arrange all the inventory record in a systematic order. It also provides the user an easy way to evaluate the supplier and to communicate with their supplier.

1.2 Hardware and Software Requirement

e-Inventory is specially designed for small and medium businesses (SMBs) use it, it can store plentiful records of inventory details. The requirements of this application are:

a) **Hardware requirement**

    i.   Pentium III 800MHz or higher processor

    ii.   128MB of RAM for Windows 2000

    iii.   at least 1GB of available disk space

    iv.   VGA 800x600 or higher-resolution screen supported by Microsoft Windows

    v.   A keyboard

    vi.   A mouse or other suitable pointing device

b) **Software requirement**

    i.   Microsoft Windows 2000/XP Professional

    ii.   Microsoft SQL Server 2000

    iii.   Internet Information Services 4.0 (IIS 4.0)

    iv.   Microsoft dotNet framework

1.3 Installation Guide

1.3.1   Microsoft Windows 2000/XP Professional

Microsoft Windows 2000/XP Installation guide can be found in the Microsoft Online Support Site or in the software CD.

NOTE:- You must set-up an Active Directory Services in Windows 2000 in order to use

Active Directory Authentication Services during web site login.

- You must also install the Internet Information Server while installing Windows

2000.

1.3.2   Microsoft SQL Server 2000 Installation

The installation guide for the SQL Server 2000 is not covered in this manual. Please refer to the installation guide that comes together with the Microsoft SQL Server 2000 Software.

NOTE: - You must select Windows Authentication in the Authentication Mode dialog box during the installation or Microsoft SQL Server 2000.

Chapter 2: System Login

2.1 Security Login

e-Inventory is provided with a security login screen to determine the user's ID. The fields that are required for the login are 'User ID' and 'User Password'. Only the user with authorize ID can have the access into the 'back-end' modules such as Software Management, Vendor Management and User Management. The administrator shall fully control the user account management. (*Figure 2.1*)
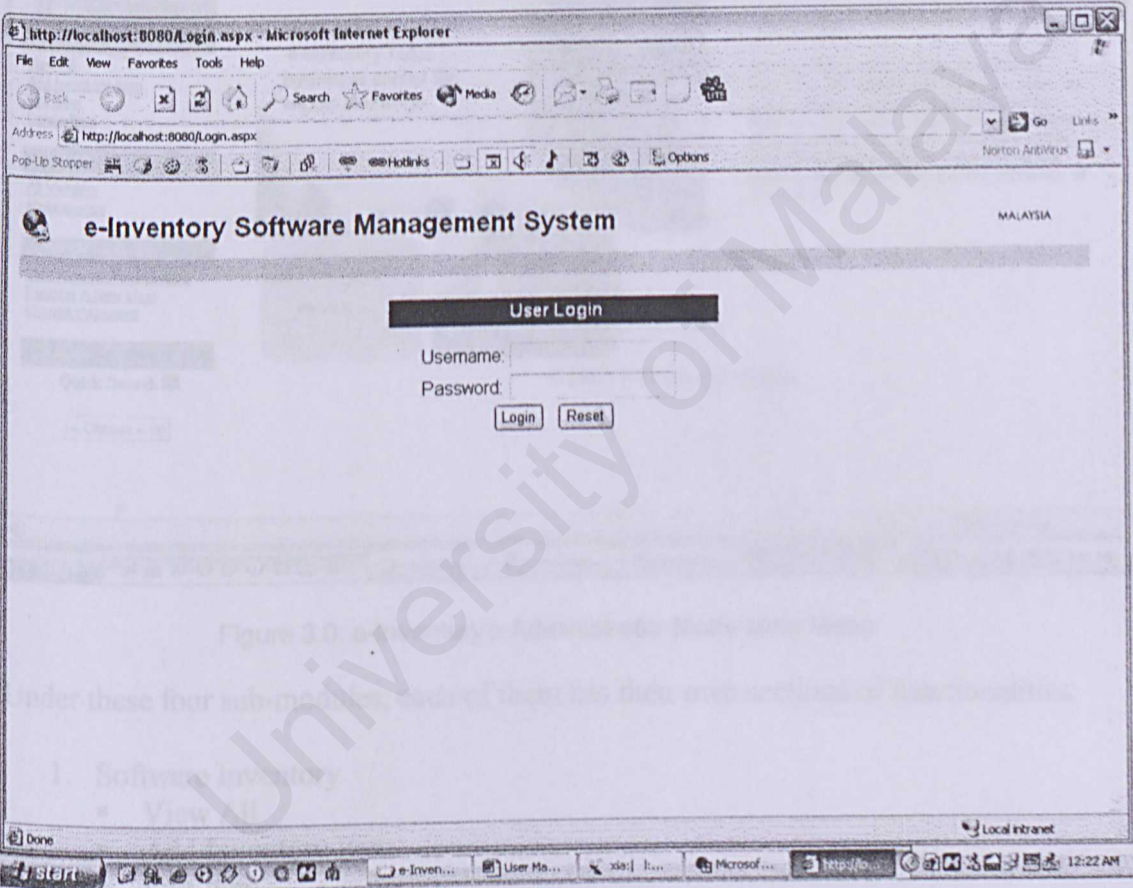
Figure 2.1: e-Inventory's Login Page

There are two access mode for the e-Inventory, which is Administrator mode and normal user mode.

Chapter 3: Administrator Mode of e-Inventory

This module is design especially for Monitoring of e-Inventory by the administrator of the company. There are four sub modules under the administrator module of e-Inventory. Their functionality and operating method are discussed below: (*Figure 3.0*)
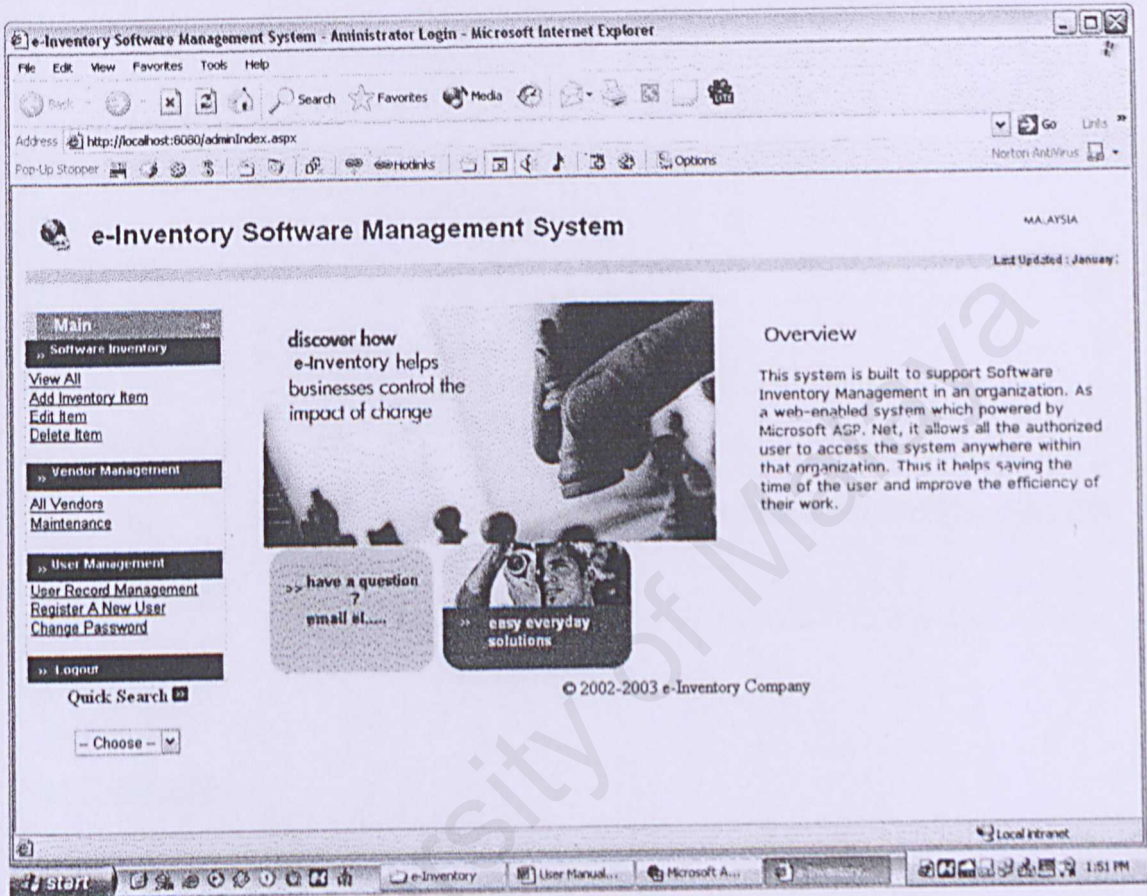
Figure 3.0: e-Inventory's Administrator Mode Main Menu

Under these four sub-modules, each of them has their own sections of functionalities:

1. Software Inventory
   - View All
   - Add Inventory Item
   - Edit Item
   - Delete Item

2. Vendor Management
   - All Vendors
   - Maintenance

3. User Management
   - User Record Management
   - Register A New user

3.1 So&...   ▪ Change Password

4. Quick Search
   ▪ By software
   ▪ By Vendors
   ▪ By User

Figure 3.1(a): View Inventory Record

3.1 Software Inventory

The administrator and user of the company can view the full listing of the software inventory record. The titles at each related field in the record table were use to sort the inventory record ascending or descending according to its fields. Following shows an example of software inventory record viewing. (*Figure 3.1*)
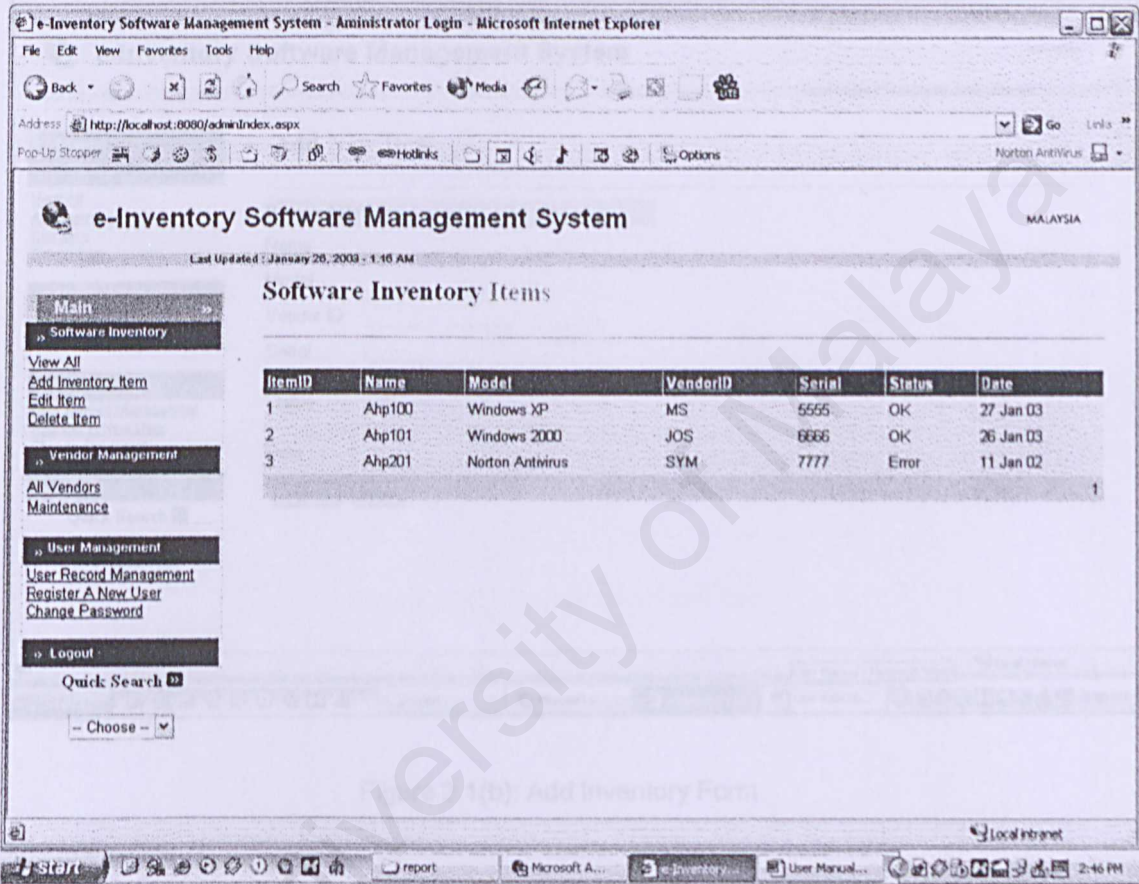
Figure 3.1(a): View Inventory Record

In Software Inventory module, it also let administrator and user to Add Inventory Item, Edit the previous Inventory Item and also Delete the record. Below show some of the add, edit and delete sub module. (Figure 3.1(b), (c) and (d))
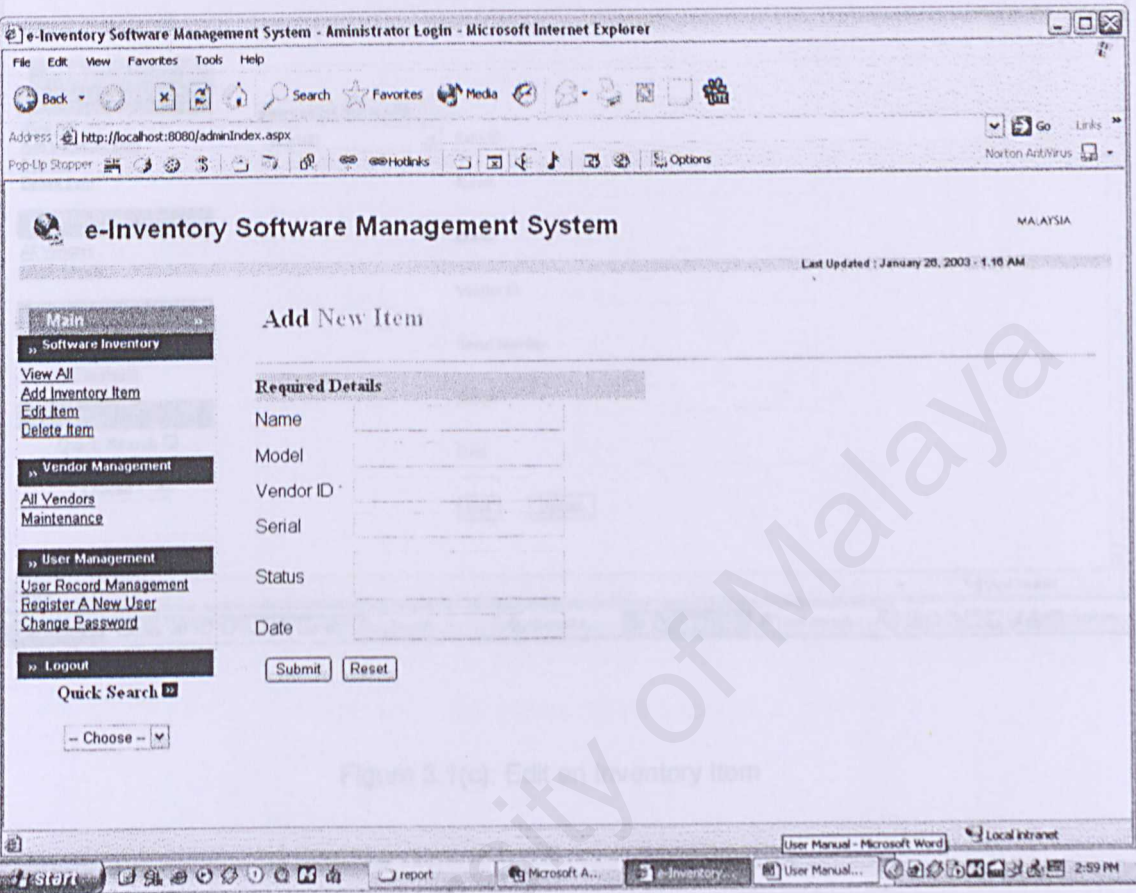
Figure 3.1(b): Add Inventory Form
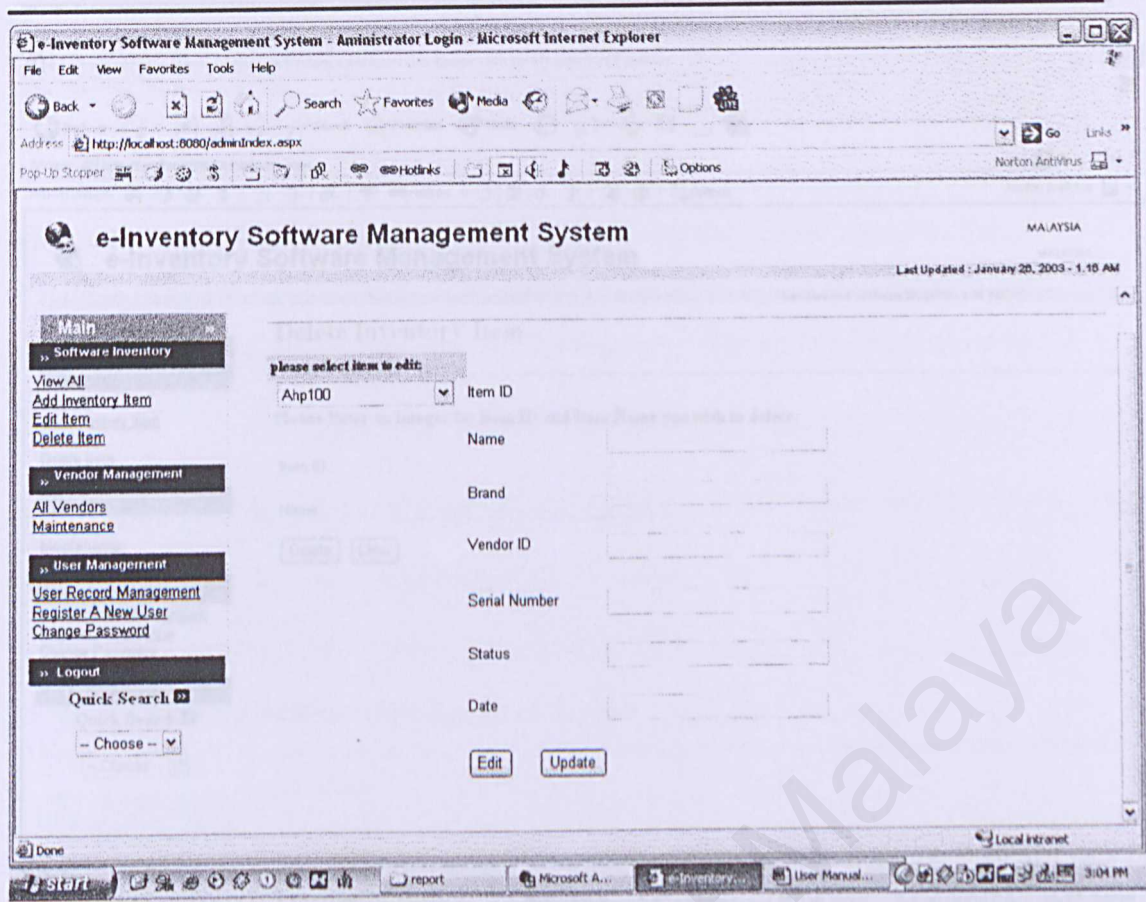
Figure 3.1(c): Edit an Inventory Item
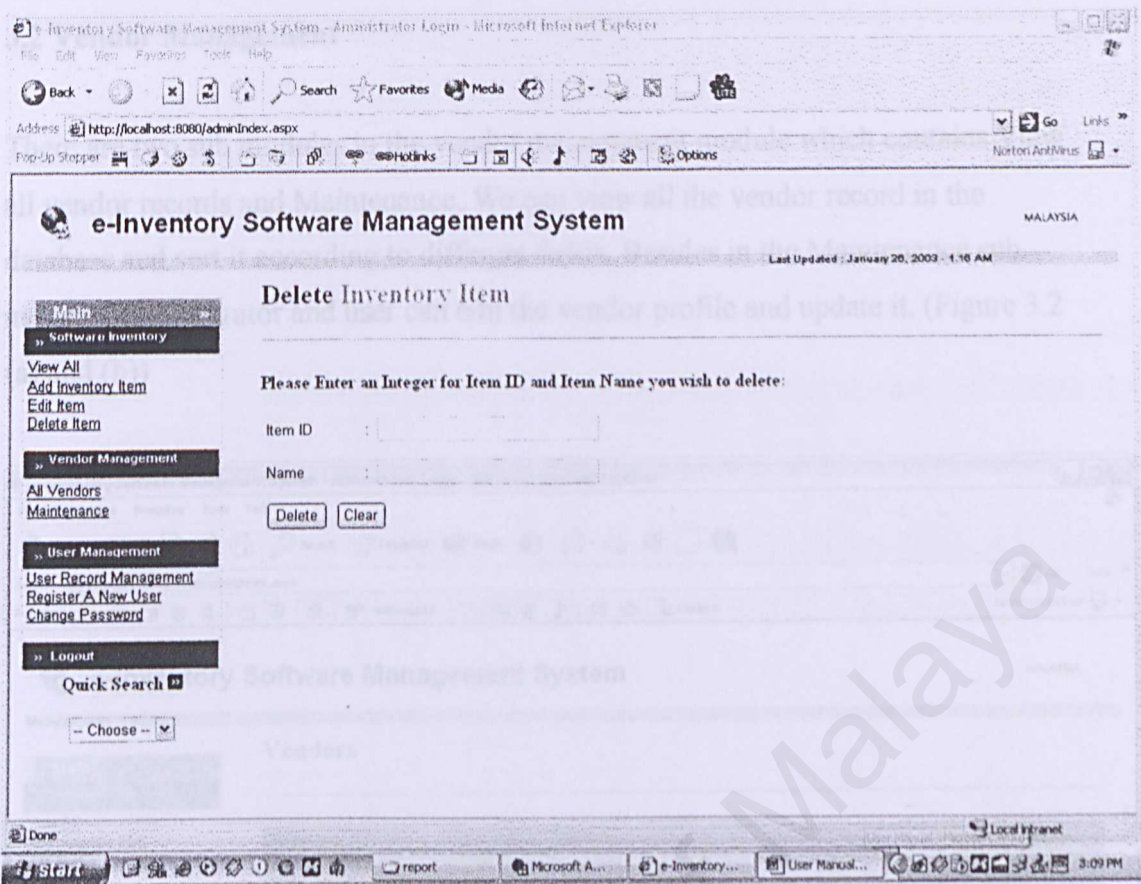
Figure 3.1 (d): Delete an Inventory Item

3.2 Vendor Management

There are two sub modules in the vendor management module which contains View all vendor records and Maintenance, We can view all the vendor record in the database and sort it according to different fields. Besides in the Maintenance sub module, administrator and user can edit the vendor profile and update it. (Figure 3.2 (a) and (b))
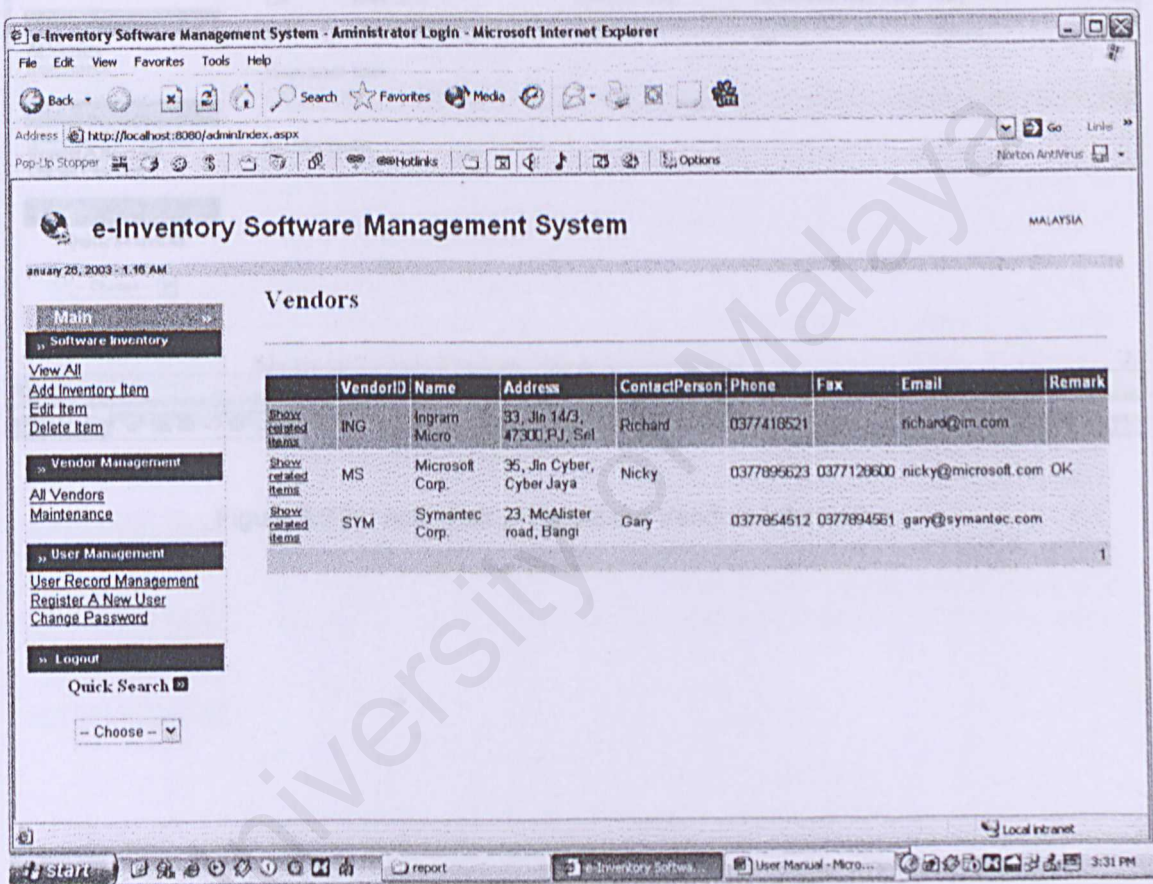
Figure 3.2(a): View all the Vendors Information
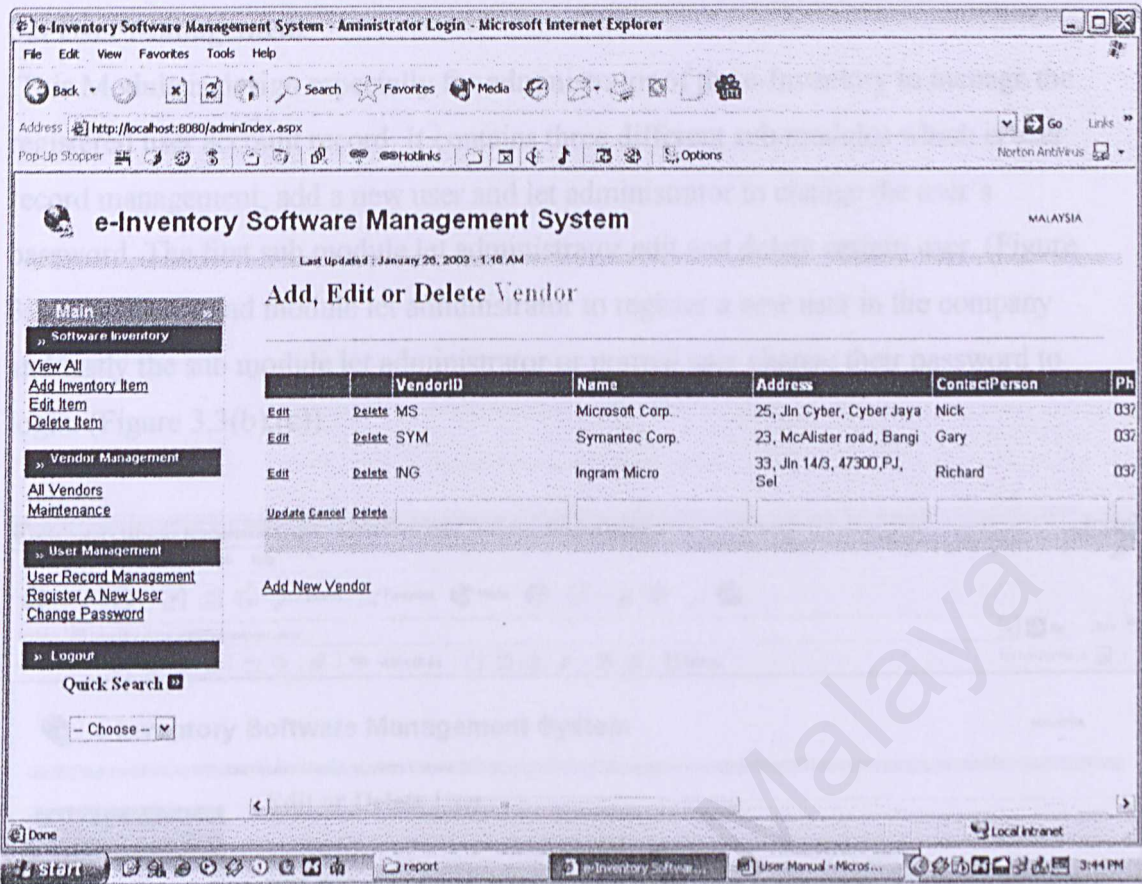
Figure 3.2(b): Add, Edit or Delete the Vendors Information

3.3 User Management

This Module is design especially for administrator of the e-Inventory to manage the registered user account record. It contains three different sub modules which is user record management, add a new user and let administrator to change the user's password. The first sub module let administrator edit and delete certain user. (Figure 3.3(a)). The second module let administrator to register a new user in the company and lastly the sub module let administrator or normal user change their password to login. (Figure 3.3(b),(c))
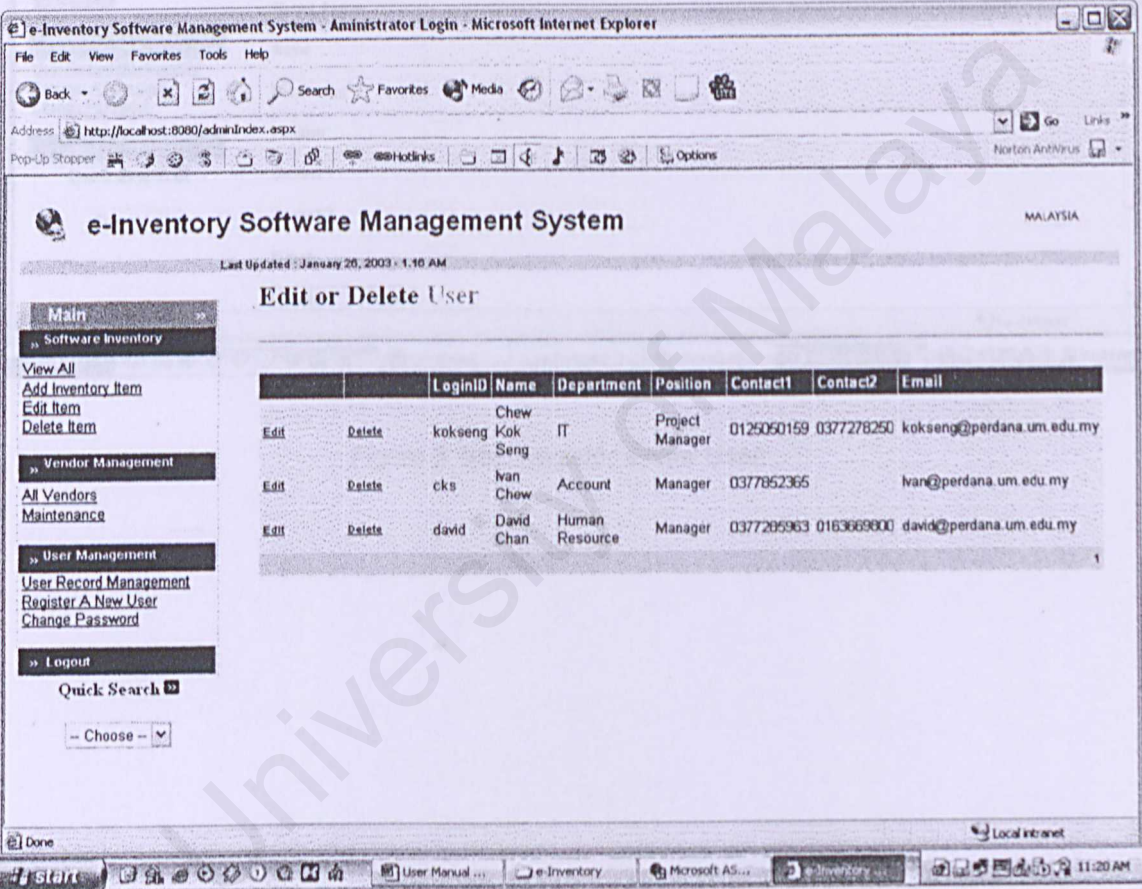
Figure 3.3(a): Edit and Add User
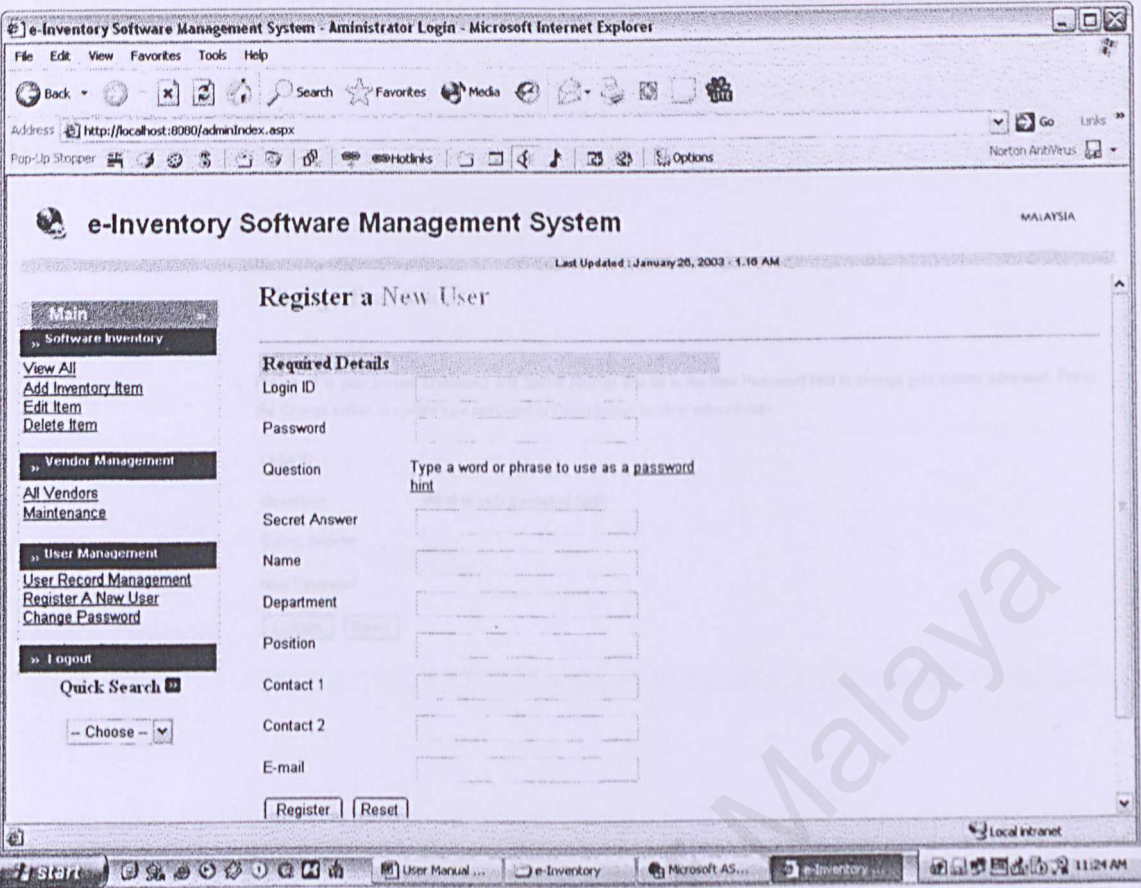
Figure 3.3(b): Register a New User

Figure 3.3(c): Form for changing Password

3.4 Quick Search Function

There are three different type of search function in this module which is search by software inventory, vendor and user. (Figure 3.4)
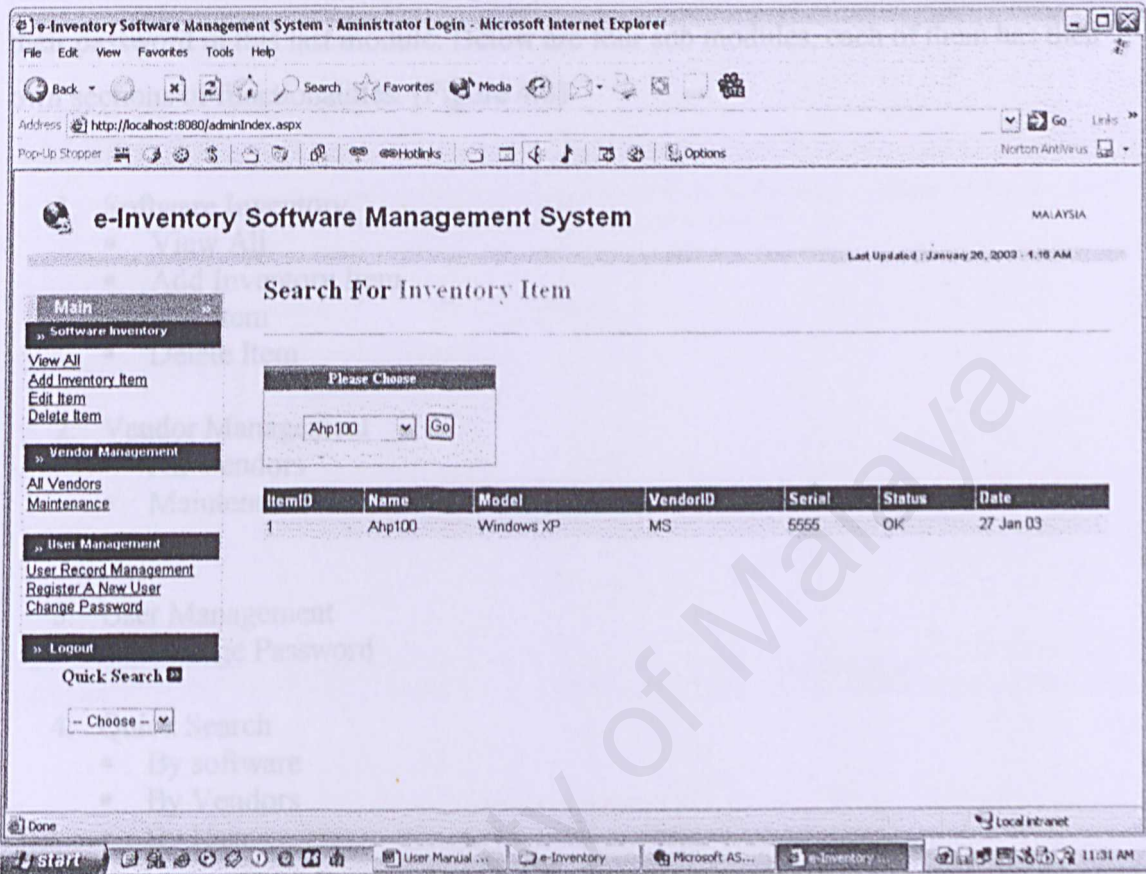
Figure 3.4: Quick Search Function

Chapter 4: Normal User Mode of e-Inventory

Under the module in the normal user, it is all same as administrator mode except in the last module which is User Management. Normal user just can manage to change their password in this last module. Below are four sub modules, each of them has their own sections of functionalities: (Figure 4.0)

1. Software Inventory
   - View All
   - Add Inventory Item
   - Edit Item
   - Delete Item

2. Vendor Management
   - All Vendors
   - Maintenance

3. User Management
   - Change Password

4. Quick Search
   - By software
   - By Vendors
   - By User
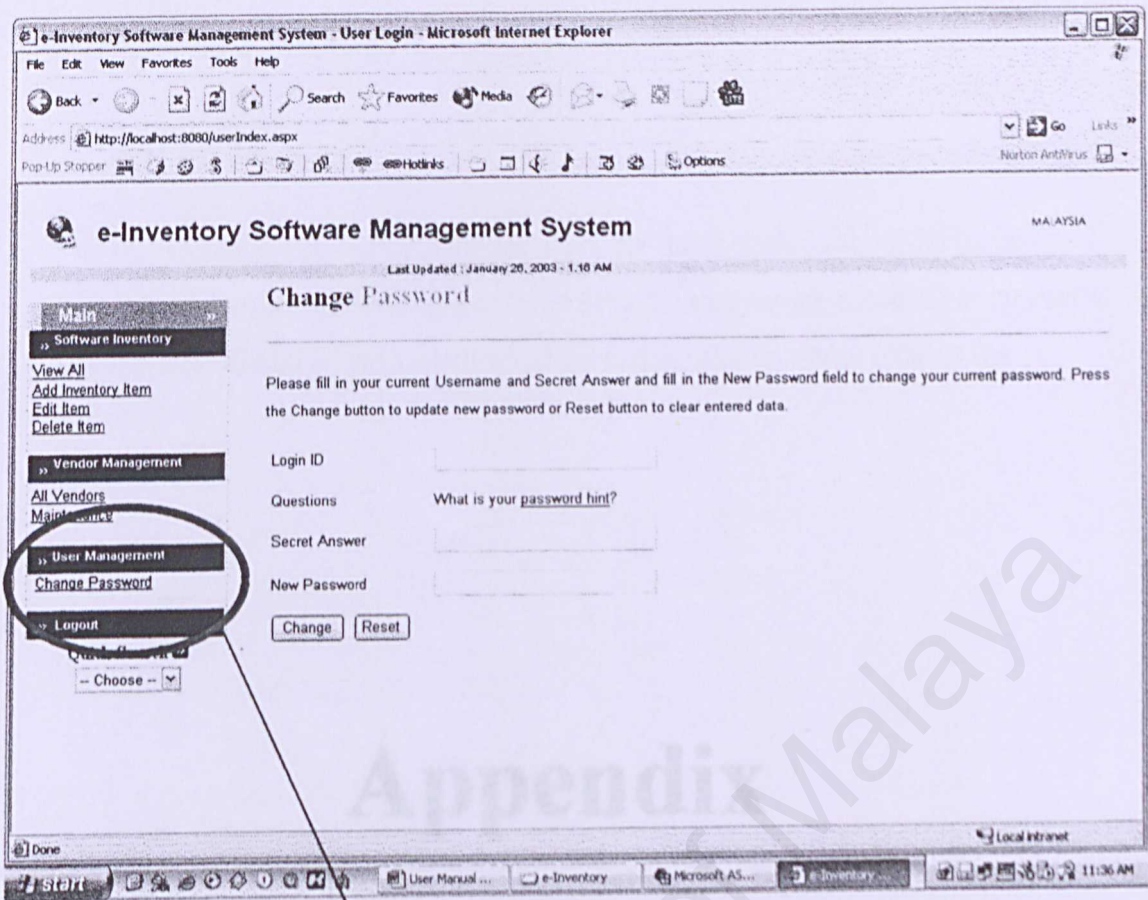
The different in the normal user login and administrator mode.

Figure 4.0: Normal User Mode

Example of Program Coding:

➢ Edit Software Inventory in inventoryEdit.aspx

NOTE: The total number of ASP.net pages for this system is about 70 pages. Thus, I only attach one example of the ASP.net pages program coding as appendix. For the full version of the system program coding please kindly refer to the softcopy.

Appendix

Example of Program Coding:

> ➢ Edit Software Inventory in inventoryEdit.aspx

NOTE: The total number of ASP.net pages for this system is about 70 pages.
Thus, I only attach one example of the ASP.net pages program coding as appendix.
For the full version of the system program coding please kindly refer to the
softcopy.

```
<%@ Page Language="vb" %>
<%@ import Namespace="System.Data" %>
<%@ import Namespace="System.Data.SQLClient" %>
<script runat="server">

    Dim intID as integer
    Dim strID, strName, strModel, strVendorID, strSerial, strStatus, strDate as string
    Sub Page_Load(Source as Object, E as EventArgs)
        if not Page.IsPostBack then
            BindData
        End If
    End Sub

    Sub BindData()
        Dim strConn as string = "server=(local);database=eInventory;Integrated
Security=SSPI"
        Dim sql as string = "Select * from SInventoryRecord"
        Dim conn as New SQLConnection(strConn)
        Dim Cmd as New SQLCommand(sql, conn)
        Dim objDR as SQLDataReader
        conn.Open()

objDR=Cmd.ExecuteReader(system.data.CommandBehavior.CloseConnection)
        list1.datasource=objDR
        list1.DataValueField="ItemID"
        list1.datatextfield="Name"
        list1.databind
        list1.selectedindex=0

    End Sub

    Sub EditRecord(Source as Object, E as EventArgs)
    Dim strConn as string = "server=(local);database=eInventory;Integrated
Security=SSPI"
        Dim sql as string = "Select * from SInventoryRecord Where ItemID = " &
list1.selecteditem.value
        Dim conn as New SQLConnection(strConn)
        Dim objDR as SQLDataReader
        Dim Cmd as New SQLCommand(sql, conn)
        conn.Open()

objDR=Cmd.ExecuteReader(system.data.CommandBehavior.CloseConnection)
        While objDR.Read()
            intID=objDR("ItemID")
            strName=objDR("Name")
            strModel=objDR("Model")
            strVendorID=objDR("VendorID")
            strSerial=objDR("Serial")
            strStatus=objDR("Status")
            strDate=objDR("Date")
        End While
        page.databind()
        Message.text=""
```

```
    End Sub

    Sub doUpdate(Source as Object, E as EventArgs)
        Dim strConn as string = "server=(local);database=eInventory;Integrated
Security=SSPI"
        Dim MySQL as string = "Update SInventoryRecord Set Name=@Name,
Model=@Model, VendorID=@VendorID, Serial=@Serial, Status=@Status,
Date=@Date Where ItemID=@ItemID"
        Dim Conn as New SQLConnection(strConn)
        Dim Cmd as New SQLCommand(MySQL, Conn)
        cmd.Parameters.Add(New SQLParameter("@ItemID", labelID.text))
        cmd.Parameters.Add(New SQLParameter("@Name", tb1.text))
        cmd.Parameters.Add(New SQLParameter("@Model", tb2.text))
        cmd.Parameters.Add(New SQLParameter("@VendorID", tb3.text))
        cmd.Parameters.Add(New SQLParameter("@Serial", tb4.text))
        cmd.Parameters.Add(New SQLParameter("@Status", tb5.text))
        cmd.Parameters.Add(New SQLParameter("@Date", tb6.text))

        Conn.Open()
        cmd.ExecuteNonQuery
        Message.text = "Item has been successfully updated!"
        BindData

list1.SelectedIndex=List1.Items.IndexOf(List1.Items.FindByValue(labelID.text))

    End Sub

</script>
<html>
<head>
</head>
<body topmargin="0">
    <form name="form1" runat="server">
        <p>
            <b><font size="5"> Software Inventory <font color="#808080">Edit
</font></font></b>
        </p>
        <hr />
        <p>
        </p>
        <table style="BORDER-TOP-WIDTH: 0px; BORDER-LEFT-WIDTH: 0px;
BORDER-BOTTOM-WIDTH: 0px; WIDTH: 564px; BORDER-COLLAPSE: collapse;
HEIGHT: 398px; BORDER-RIGHT-WIDTH: 0px" bordercolor="#111111"
cellspacing="0" border="1">
            <tbody>
                <tr>
                    <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="171" bgcolor="#c0c0c0" height="17">
                        <p align="left">
                             <b><font size="2">please select item to
edit:</font></b><font size="2"> </font>
                        </p>
                    </td>
```

```
          <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="364" colspan="3" height="17">
                    <p>
                    </p>
              </td>
          </tr>
          <tr>
              <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="171" height="143" rowspan="7">
                    <div align="left"> 
                        <asp:DropDownList id="list1" runat="server"
Width="160px"></asp:DropDownList>
                    </div>
                    <div
align="left">          &n
bsp; 
                    </div>
                    <div align="center">
                    </div>
                    <div align="center">
                    </div>
                    <div align="center">
                    </div>
              </td>
              <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="113" height="17">
                    <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                        <font face="Arial" size="2">Item ID</font>
                    </p>
              </td>
              <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="4" height="17">
                    <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                    </p>
              </td>
              <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="247" height="17">
                    <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                        <font face="Arial" size="2">
                        <asp:Label id="labelID" runat="server" Text="<%# intID
%>"></asp:Label>
                        </font>
                    </p>
              </td>
          </tr>
          <tr>
              <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="113" height="17">
```

```html
                <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                    <font face="Arial" size="2">Name</font>
                </p>
            </td>
            <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="4" height="17">
                <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                </p>
            </td>
            <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="247" height="17">
                <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                    <asp:textbox id="tb1" runat="server" Width="200px" Text="<%#
strName %>"></asp:textbox>
                     
                </p>
            </td>
        </tr>
        <tr>
            <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="113" height="17">
                <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                    <font face="Arial" size="2">Brand</font>
                </p>
            </td>
            <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="4" height="17">
                <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                </p>
            </td>
            <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="247" height="17">
                <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                    <font face="Arial" size="2">
                    <asp:textbox id="tb2" runat="server" Width="200px" Text="<%#
strModel %>"></asp:textbox>
                     </font>
                </p>
            </td>
        </tr>
        <tr>
            <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="113" height="17">
                <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                    <font face="Arial" size="2">Vendor ID</font>
                </p>
            </td>
```

```html
                    <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="4" height="17">
                        <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                        </p>
                    </td>
                    <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="247" height="17">
                        <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                            <font face="Arial" size="2">
                            <asp:TextBox id="tb3" runat="server" Width="200px" Text="<%#
strVendorID %>"></asp:TextBox>
                             </font>
                        </p>
                    </td>
                </tr>
                <tr>
                    <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="113" height="17">
                        <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                            <font face="Arial" size="2">Serial Number</font>
                        </p>
                    </td>
                    <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="4" height="17">
                        <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                        </p>
                    </td>
                    <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="247" height="17">
                        <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                            <font face="Arial" size="2">
                            <asp:TextBox id="tb4" runat="server" Width="200px" Text="<%#
strSerial %>"></asp:TextBox>
                             </font>
                        </p>
                    </td>
                </tr>
                <tr>
                    <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="113" height="17">
                        <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                            <font face="Arial" size="2">Status</font>
                        </p>
                    </td>
                    <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="4" height="17">
                        <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
```

```
                    </p>
                </td>
                <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="247" height="17">
                    <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                        <font face="Arial" size="2"></font>
                        <asp:TextBox id="tb5" runat="server"
Width="200px"></asp:TextBox>
                    </p>
                </td>
            </tr>
            <tr>
                <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="113" height="17">
                    <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                        <font face="Arial" size="2">Date</font>
                    </p>
                </td>
                <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="4" height="17">
                    <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                    </p>
                </td>
                <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="247" height="17">
                    <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
                        <font face="Arial" size="2">
                        <asp:TextBox id="tb6" runat="server" Width="200px" Text="<%#
strDate %>"></asp:TextBox>
                         </font>
                    </p>
                </td>
            </tr>
            <tr>
                <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="169" height="1">
                    <div align="center">
                    </div>
                </td>
                <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="366" colspan="3" height="1">
                    <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px" align="left">
                        <asp:button id="button1" onclick="EditRecord" runat="server"
Width="40px" Text="Edit"></asp:button>
                             <asp:button id="Button4"
onclick="doupdate" runat="server" Text="Update"></asp:button>
                    </p>
                    <div align="center">
```

```
          </div>
        </td>
      </tr>
      <tr>
        <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="169" height="3">
          <p id="ID" style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
          </p>
        </td>
        <td style="BORDER-RIGHT: medium none; BORDER-TOP: medium
none; BORDER-LEFT: medium none; BORDER-BOTTOM: medium none"
valign="top" width="366" colspan="3" height="3">
          <p id="ID" style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 12px">
            <asp:Label id="Message" runat="server" Font-Names="Arial" Font-
Size="X-Small" ForeColor="Red"></asp:Label>
          </p>
        </td>
      </tr>
    </tbody>
  </table>
  <p id="ID">
  </p>
  <p align="left">
  </p>
</form>
</body>
</html>
```