University of Technology MARA

DEVELOPMENT OF COMPUTER INVENTORY SYSTEM USING AUTOMATIC DATA GATHERING AND BARCODE FOR FTMSK

MOHD HAFIDZI BIN MAT YASIN

Thesis submitted in fulfillment of the requirements for

Bachelor of Science (Hons.) Data Communication and Networking

Faculty of Information Technology and Quantitative Sciences

November 2007

APPROVAL

DEVELOPMENT OF COMPUTER INVENTORY SYSTEM USING AUTOMATIC DATA GATHERING AND BARCODE FOR FTMSK

Research done by:
MOHD HAFIDZI BIN MAT YASIN
2005617427

A project paper submitted to the
Faculty of Information Technology and Quantitative Sciences
University of Technology MARA

In partial fulfillment of requirement for the
Bachelor of Science (Hons) Data Communication and Networking

Approved By:
Project Supervisor

.....
(SALMAH BTE ABD AZIZ)

MARA University of Technology
Shah Alam
November 2007

DECLARATION

I certify that this thesis and the research to which it refers are the product of my own work and that any ideas or quotation from the work of other people, published or otherwise are fully acknowledged in accordance with the standard referring practices of the discipline.

NOVEMBER 09, 2007 …….……

MOHD HAFIDZI BIN MAT YASIN

2005617427

ACKNOWLEDGEMENT

In the name of Allah, the most gracious and the most merciful. Alhamdulillah, thanks to the Almighty for blessing me with strength and courage to complete this thesis. First of all, I would like to address my appreciation and special thanks to my dedicated supervisor, Puan Salmah Bte Abd Aziz for her encouragement, guidance, tolerance, detailed review, constructive criticism and excellent advice during the preparation of this thesis.

I also would like to express my appreciation to my lecturers, especially Encik Adzhar b Abdul Kadir for his priceless guidance, ideas and critics to make the best of this research. Last but not least, I also would like to extend my gratitude and appreciation to my family and friends who really give me the full support, encouragement and motivation during this work.

ABSTRACT

Computer system inventory is an important subject in management. Having reliable computer inventory data will help the staff in managing computer assets, especially when they are going to trace the computers and peripheral and do mass upgrade. The purpose of the system is to help the staffs manage the computers inventory more effective and easy way. Using this Computer Inventory System, staffs can get information about all computers, monitors and other peripherals that currently exist in the faculty. They also can trace where all the computers are located in the faculty whether in the computer labs, classes, offices, in the lecturer's room or anywhere in the faculty. The main feature of this system is using Automatic Data Gathering and Barcode system. The staff no longer needs to key in all the data about the computer specifications into the system but only the serial number and the location of the computer. The staff just can sit in their room waiting for the computer data and manage from their room. The methods that will be used to gather information for this research are done by experiment. The system architectures that are used to develop this system are XAMP which is the combination of Apache Web Server, MySQL Database Server, PHP and PhpMyAdmin, PC and barcode scanner. This system can be accessed using web browser that have internet connection and managed by the Administrator and the staffs. This solution will enable the staff reduce time and can manage the computer inventory effectively and more efficient.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Computers represent major capital investment and assets for most organizations. To manage this investment, it is important to have a system to track the status and location of the assets. Having a reliable computer inventory system will help the staff in managing computer data, especially when they are going to trace the computers and devices.

There are many commercial systems in the market, but a lot of the software, the good ones, always required a high cost to implement. Basically, the system stored only limited data and information of the inventory. Some of the systems are using barcode to keep track of the inventory. Barcode technology quite often is used to keep data such as product information or serial number that is unique.

Faculty of Information Technology and Quantitative Science (FTMSK) currently has no computerized and automated computer inventory process nor does it use barcode or other types of technology to track computer and devices. FTMSK currently relies on manual inventory methods to manage and maintain their computer information in the log book. The manual inventory method relies on traditional checking of items with paper-based physical inventory reports and validation of human readable asset labels. Information on automatic data gathering, label and ID technologies, barcode hardware and scanning technologies, software applications, and a system development are discussed.

This research will study on how an automated and barcode computer inventory system can reduce the amount of time needed compared to perform a physical inventory as well as increase the accuracy of the results.

The implementation of moderate barcode technology and web or network based data collection, would best suit FTMSK's infrastructure. It offers the benefits of both web-based browser access as well as barcode technology without committing to a dedicated centralized inventory staff.

This chapter will explain the background research problem and the flow of the current system. This is followed by problem statement, objective, scope and significant of this research project.

1.2 Problem Statement

Currently in the FTMSK, there is no system is used to manage the computers and other peripherals in the faculty. All computers that have been located in the faculty, whether in the computer lab, in the office or in the lecturer's room is currently using form or log book. The staff is facing hard time to track the location and the owner of the computer and other devices in the faculty.

The inventory data is collected by manually, by key-in all the data by the staff. It required a lot of time to do the data entry job and updating the data for the staff. By using the computer inventory system to collect all the required data using the automatic data gathering feature, this will save more time. Barcode scanner will be used for retrieving the computer data by scanning the computer barcode label.

1.3 Research Question

a. Why barcode technology are widely used and implemented in the inventory system and other system?
b. How to manage the assets (computer and other devices) of faculty in manageable and easy way?
c. What are the expected results of the system that will be developed?

1.4 Objective

The objective of the project has been identified as follows:

a. To study the efficiency of using barcode technology to track the inventory and devices compare to manual inventory method.
b. To develop a web-based inventory system that will store all data inventory (hardware and software) of the computers and other devices in the faculty automatically, and retrieving the computer data by using barcode scanner.
c. To develop a system that will be able to generate reports based on the data inventory that has been collected.

1.5 Scope of the Research

Scope is defined as guidelines to develop and design an “Automated and Barcode Computer Inventory System (ABACIS)” project. The system will be collecting data of the computer hardware and software details using automatic data gathering features, and retrieving the computer data by using barcode scanner. The data that will be collected are the details of computer specification (hardware and software). Then the system will generate a barcode image to the computer to help the staff to track the computer easily using a handheld barcode scanner. The inventory data in the system will be analyzed and then will be generate into a report for future references.

The scope of the research has been further subdivided into smaller scopes such as following:

1.5.1 Location

The location of implementing the project will be located at FTMSK Computer Office, Faculty of Information Technology and Quantitative Sciences, Universiti Teknologi MARA (UiTM) Shah Alam.

1.5.2 System Architecture

i. Software that will be used for this project are PHP programming, Apache Web Server, MySQL Server and PhpMyAdmin (XAMP).

ii. Hardware that will be used for this project are Personal Computer (PC) and barcode scanner.

1.5.3 User

i. The primary user is the administrator that will control and have full access in the entire system. The administrator has the capability to manage the computer inventory system.

ii. The secondary user is lab technician in each computer lab in the faculty. They will only have access for collecting data and updating the computer data in their lab.

1.5.4 Technology

The technology that will be used is barcode technology and JavaScript scripting ability for Automatic Data Gathering feature.

1.6 Significant of Research

a. This project will help the staff to manage all the computers and other devices in a web-based system that can be accessed from other places. The staff can track all the location of the computer inventory in the faculty using this system. This will prevent from loss of the inventory from the being stolen or not in the database.

b. The system will be useful to the staff and make ease for them to manage and take control of all the computer and assets of FTMSK.

c. The staff of the FTMSK computer office will gain benefit from this system as their work can be reduced and in manageable way.

1.7 Summary

Automated and Barcode Computer Inventory System (ABACIS) is develop to help the staff control and manage the computer inventory data in the faculty. This will reduce their time by using automatic data gathering feature that will automatic collect data from the computer and by using barcode scanner to find the specific data in the system.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter will review all related work and technology used in developing the "Automated and Barcode Computer Inventory System" (ABACIS). This chapter will review on the technologies that will be used in the research and the current or related projects that have been studied.

2.2 Definition of Terms/Concepts

2.2.1 Barcode Inventory System

The term "barcode inventory system" is generic and encompasses a variety of different applications and industries. The term is commonly used in warehousing, logistics, and distribution; it can also be used in retail point-of-sale, manufacturing, and various service sector implementations. In each case, at its core, a "barcode inventory system" includes a measurable list of items or quantities and utilizes barcode technology in some fashion.

Barcode technology is a means of interfacing humans to data processing equipment; therefore, barcode technology cannot be utilized independent of data processing equipment. Commonly, the data processing equipment stores the results of a "barcode inventory system" in electronic form.

2.2.2 Automatic Data Gathering

This term refer to the concept of data collection that run on script that will be developed. It is important that the system will automatic, by detection of the computer hardware and software component and then store the data into the database at the same time. The script will be based on JavaScript because it supports the *Windows Management Instrumentation* (WMI) and *Windows Script Host* (WSH). Both two components are required to enable the detection of the computer hardware and software function in this system. The details of how automatic data gathering function developed based on the two components will be explain later.

2.2.3 Windows Management Instrumentation (WMI)

Effective management of computer in an enterprise network benefits from well-instrumented computer software and hardware, which allow system components to be monitored and managed by the administrator.

Web-Based Enterprise System Management (WBEM) is one of the main areas where many corporations spend tons of money to manage their IT infrastructure. WBEM is an open standard specification defined by the independent organization called "*Distributed Management Task Force*" (DMTF). WBEM is an industry standard to manage physical and logical resources such as computers, networks, etc. The WBEM architecture, which is object-oriented in nature, is based on the *Common Information Model* (CIM). CIM is a unified standard way of defining the physical and logical objects via a schema. *Windows Management Instrumentation* (WMI) is the Microsoft implementation of WBEM and mostly all the objects and collections follow CIM standards.

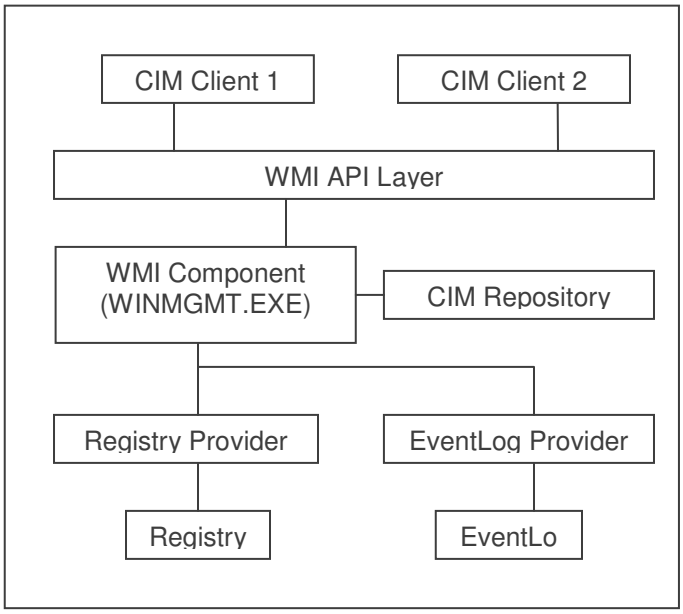


Figure 2.1 : Architecture of WMI

The purpose of WMI is to define a non-proprietary set of environment-independent specifications. These specifications allow management information to be shared between management applications that run in both similar and dissimilar operating system environments.

WMI uses the CMI and Internet protocols for providing management data. WMI provides a constant and richly descriptive model of the configuration, status, and operational aspects of Windows operating systems, assisting management applications in creating solutions that reduce the maintenance and life cycle costs of managing Windows.

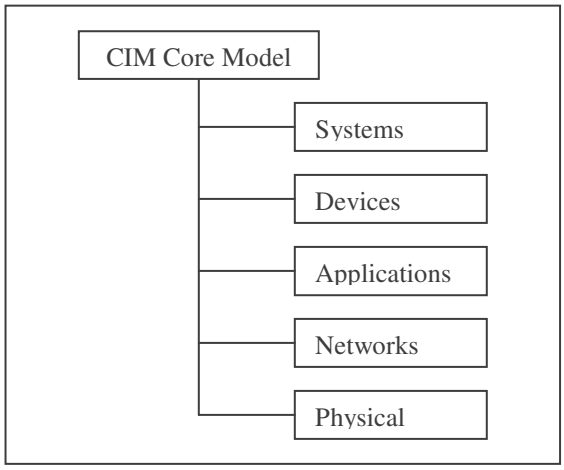


Figure 2.2 : CMI core model

Used in conjunction with other management services such as the *Microsoft Management Console (MMC)*, WMI helps simplify the task of developing well integrated management applications, allowing vendors to provide Windows customers with best-of-breed, enterprise-scalable management solutions. Local and remote events combined with a rich query language for the information model provide the means to create solutions to complex management problems. The ability to easily script these solutions in Visual Basic or using Windows Scripting Host (WSH) adds an often-requested dimension to Windows management.

Microsoft has developed a new set of WMI classes derived from CIM to access Windows OS-specific information. These classes start with *Win32_* as an identifier. *Win32_* classes are grouped under three major categories.

WMI Class	Description
Computer System Hardware Classes	These classes expose hardware related information like, local drives, networking components, etc.
Operating System Classes	These classes expose OS related information such as, Windows User Accounts, Win32 Processes, etc.
Installed Application Classes	These classes expose all the Microsoft Windows Installer related information.

Table 1 : Three major categories of WIN32_ classes

For an example, the following piece of code will list all the Win32 services from the local computer.

```
For Each objWBEM in
GetObject("Winmgmts:").InstancesOf("Win32_Service")
Response.write "DisplayName: " & objWBEM.DisplayName & _
   "<BR>State: " & objWBEM.State & _
   "<BR>Status: " & objWBEM.Status
Next
```

Here is the result of the above code:

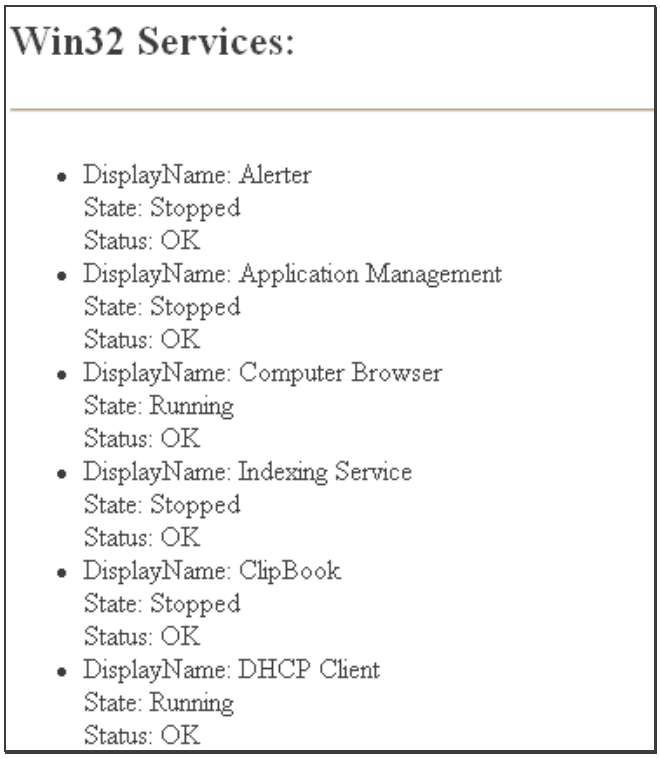


Figure 2.3 : Result of sample code.

2.2.4 Windows Script Host (WSH)

The Microsoft Windows Script Host (originally called Windows Scripting Host) is distributed and installed by default on Windows 98 and later versions of Microsoft Windows. It is also installed if Internet Explorer 5 (or a later version) is installed. It provides scripting capabilities comparable to batch files, but with a greater range of supported features. Beginning with Windows 2000, the Windows Script Host became available for use with user login scripts.

It is language-independent in the sense that it can make use of different Active Scripting language engines. By default it interprets and runs plain-text JScript (.JS and .JSE files) and VBScript (.VBS and .VBE files). Users can install different scripting engines to enable them to script in other languages, for instance Perl. The language independent filename extension WSF can also be used.

The advantage of the Windows Script File (.WSF) is that it allows the user to use a combination of scripting languages within a single file. The researcher has taken the advantage of the scripting ability by developing a JavaScript file to be used as Automatic Data Gathering engine for Computer Inventory System.

2.3 Different Types of Inventory System

There are a number of different inventory systems starting with the most ancient system to the most advanced system.

2.3.1 Physical Counts (Periodic System)

The most basic method of inventory tracking is physical counts. A lot of firms relied on physical count to try and figure approximately where they were at.

2.3.2 Two-Bin System

The two-bin system is only slightly more sophisticated than the physical count system. Using the marvels of modern technology, this system uses two bins of materials. When one bin is empty, it's time to re-order.

2.3.3 Perpetual Tracking

This method of counting is demand driven. Instead of counting how many items are in inventory, we count how many leave inventory. The demand can be tracked by batches of inventory usage, such as demand that is entered once a week, or they can entered in real-time which provides the ability to continuously monitor inventory levels. If we are already in the practice of counting demand, this is a great way to track inventory because it involves little additional effort. It was ideal because there is little variability in the products that are demanded. For a supermarket, this can be more difficult because of the variety of products sold.

2.3.4 Universal Product Code (UPC)

UPC is a system that supermarkets first implemented in the 70s. This requires unique codes to be put on all types of inventory and is usually accompanied by a barcode that can be scanned via infrared scanner. In addition to helping the market determine how much customers should pay, this also gives supermarkets, and other firms, the ability to track and count the movement of any and all inventory with a simple infrared scan.

Example of infrared product tracking is the implementation of package tracking that FeDex and UPS have incorporated into their business processes. Both FeDex and UPS now track all packages at every stage from pickup to delivery. The beauty behind the system these two companies have implemented is that not only does it directly help their operations management, but it also directly improves customer service. Now, customers can log onto a FeDex or UPS website from anywhere in the world, enter in the tracking number they received when they dropped the package off for shipping and know exactly where their package is in real time. Marketing studies have shown that informed customers are typically more satisfied and this process is a wonderful example of how the businesses are able to inform themselves and their customers with one technology.

2.3.5 Radio Frequency ID (RFID)

RFID is another method of tracking inventory. Instead of using technology to track inventory as it is moved, RFID counts inventory automatically from a remote location. This is superior to perpetual inventory tracking or perpetual inventory with the UPC for a couple of reasons. Most notably is that RFID accounts for shrinkage (lost inventory). As much as we might not want inventory to just disappear, the fact of the matter is, things grow legs. Also, inventory is often scrapped. Perpetual counts that lower inventory levels only when there is demand don't account for lowered inventory when a good is stolen. RFID counts what is actually there, and it can tell you exactly where it is.

2.4 Barcode Technology

Barcode printing has greatly risen in popularity due to the speed, ease and accuracy of automatic data collection. Barcode increases productivity and tracking in all different types of industry, and is an important component of inventory control in the supply chain.

2.4.1 Bar Code Application

There are many industries and applications utilizing barcode technology. The following industries are just a few that have implemented barcode technology:

- Healthcare
- Shipping and Receiving
- Compliance Labeling
- Ticketing
- Government
- Enterprise Resource Planning
- Manufacturing
- Retail and Grocery Stores
- Transportation and Logistics
- Supply Chain Management

The following are a few examples of applications using barcode technology within these industries:

- Admissions
- Pharmacy
- Radiology
- Blood Bank
- Inventory Control
- Order Fulfillment
- Shipping
- Parts Marking
- Driver Licenses
- Postal Services
- Auto Registration
- Asset Tracking
- Drum Labeling
- Pallets Containers
- Cargo/Bin
- Baggage Tags

- Boarding Cards
- Work in Progress
- Shelf Labeling
- Food Items
- Clothing
- Electronics Labeling

2.4.2 Barcode Types, Content and Format

Although there are many symbologies used to create barcodes, most symbologies of barcode produce symbols with several aspects in common.

Barcode Content

Bars and spaces are known as the "elements" of the barcode. They are grouped together to make "characters" that each represent a number, letter, punctuation mark or other character. Bars must be dark enough so they do not reflect back a bar code reader light, while the spaces within the barcode and the background around it must be clear and reflective enough to be distinguishable from the bars by the barcode reading device.

The "density" of the barcode is the number of characters that can be represented per linear inch and is usually expressed in characters per inch or CPI. The higher the density, the more information a barcode can represent within a given space.

Barcode Format

Clear spaces, known as "quiet zones" or margins, are placed before and after the barcode symbol. These quiet zones ensure that only the complete barcode is read.

"Start" and "Stop" characters or patterns indicate the beginning and end of the barcode symbol and sometimes even provide clues as to the direction the barcode is being read. "Bi-directional" barcodes can be read from

either of two directions. Most barcodes are arranged in a "linear" format, a single row of bars and spaces that usually allows for Bi-directional reading. These are often referred to as "one dimensional" bar codes. "Two dimensional' barcodes encode information in two directions and they require special readers. These barcodes come in two basic formats called "stacked" symbologies and "matrix" symbologies.

"Check characters" or "check digits" are sometimes used to help determine that the correct information is read from a barcode. "Self checking" barcodes prevent a printing defect from causing similar characters to be substituted for each other.

"Data" or "application" identifiers are sometimes used to indicate the general category or intended use of the information contained within a barcode.

Symbologies that create "discrete" codes separate each character by spaces that carry no information. Discrete characters can be decoded independently and do not require the highest print quality standards. Symbologies that create "continuous" codes use every space within the bar code to carry information. Continuous codes can convey more information per inch than discrete codes but have slightly higher print quality requirements.

Barcodes are most often displayed "horizontally" and in this orientation are known as "Picket Fence" codes due to their resemblance to a picket fence. However they may also be displayed "vertically" and these are referred to as "Rotated" or "Ladder" codes due to their resemblance to the rungs of a ladder.

2.4.3 Common Barcode Symbologies

2.4.3.1 Universal Product Code (UPC)

The Universal Product Code, or UPC, is widely used in retail, packaging, counting and data processing applications. There are several versions of the basic symbology in use, including the EAN standards for international applications.

Figure 2.4 : Universal Product Code (UPC)

Characters : Only numbers from 0 to 9 are represented.
Length : Fixed at 12, 6 digits.
Format : Linear, continuous barcode.
Reading : Bi_directional.
Checking : Self_Checking and Check Digit incorporated into the barcode.
Required size: A 12 digit full sized barcode requires about 1.5 inches horizontally and 1.0 inches vertically.
International : The European Article Numbering, EAN, standard is UPC's international counterpart.

2.4.3.2 Interleaved 2-of-5

Interleaved 2 of 5 is a high-density code used in warehousing, product/container identification general industrial and automotive applications. Its name indicates each barcode character contains five bars, two of which are wide. Both bars and spaces convey information. This symbology is very useful for numeric messages less than 10 digits long.

Figure 2.5 : Interleaved 2-of-5

Characters : Only numbers from 0 to 9 are represented.
Length : May vary but must have an EVEN number of digits.
Format : Linear, continuous barcode.
Reading : Bi-directional.
Checking : Self-Checking and may have a Check Digit incorporated into the barcode.

2.4.3.3 Code 39

Code 39 is widely used in industrial, medical and government applications, including photo finishing, high speed sorting, inventory handling, aluminum, electronics, telecommunications and furniture. It is endorsed by several industry trade groups including the Automotive Industry Action Group AIAS, the Health Industry Business Communications Group HIBCC and the U.S. department of Defense DOD. Its name signifies each barcode character is composed of nine elements, three of which are wide.

Figure 2.6 Code 39

Characters : Represents ALL 128 alphanumeric characters from the ASCII character set.

Length : Variable.

Format : Linear, discrete code.

Reading : Bi-directional.

Checking : Self Checking. May have a Check Digit incorporated into the barcode but is not normally used.

2.4.3.4 Code 128

Code 128 applications include general industrial, inventory control and container markings. It is used as the basis for the international language known as Application Identifiers, or AI. As its name signifies, this high-density code can represent the entire 128 characters ASCII character set, including any character found on a PC keyboard. The code offers high versatility and high data security (reliability). Code 128 is endorsed by the HIBCC and the Uniform Code Council, UCC.

Code 128

Health Industry Bar Code (HIBC)

UCC/EAN Code 128

Figure 2.7 Code 128

Characters : Represents ALL 128 alphanumeric characters from the ASCII character set plus any keyboard character.
Length : Variable.
Format : Linear, continuous code.
Reading : Bi-directional.
Checking : Self Checking and may have a Check Digit incorporated into the barcode.

2.4.3.5 Codabar

Codabar applications include inventory control, libraries, blood banks and photo finishing. Each character is represented by a group of four bars with three included spaces. The ability to use four different start/stop characters at either end of the barcode symbol allows multiple types of information to be encoded.

Figure 2.8 Codabar

Characters : Decimal number digits and several ASCII symbols.
Length : Variable.
Format : Linear, discrete code.
Reading : Bi-directional.
Checking : Self Checking and optional Check Digit.

2.4.3.6 Matrix Symbologies (Data Matrix)

Data Matrix is a relatively recent 2 dimensional barcode development. This code has a high data capacity that is most commonly used for product identification and lot tracking of small components. Several organizations have recommended the use of Data Matrix for specific applications. An example of this is the "EIA" of the Electronics Industry Association, "SEMI"_ the Semiconductor Equipment and Materials International association and the "AIAG"_ Automotive Industry Action Group. All have been coordinating efforts to create an inter industry series of compatible standards for wafer, electronic component and product marking. For these organizations Data Matrix is desirable because of its high data capacity, its variable sizing capabilities and its ability to withstand harsh environments.

The healthcare industry has also found several applications including lot tracking of pharmaceuticals and specimen identification in biomedical testing.

Figure 2.9 Data Matrix

2.4.3.7 MaxiCode

MaxiCode was introduced by the United Parcel Service and developed specifically for high-speed package sortation applications. The purpose behind the creation and use of MaxiCode is to be able to encode all the necessary information about a package and allow that information to travel with the package. In essence, this information gives the package a "personality".

Figure 2.10 MaxiCode

2.4.3.8 Other Barcode Symbol

While the previous symbologies are some of the common types used, there are many more available. Here are a few more samples:

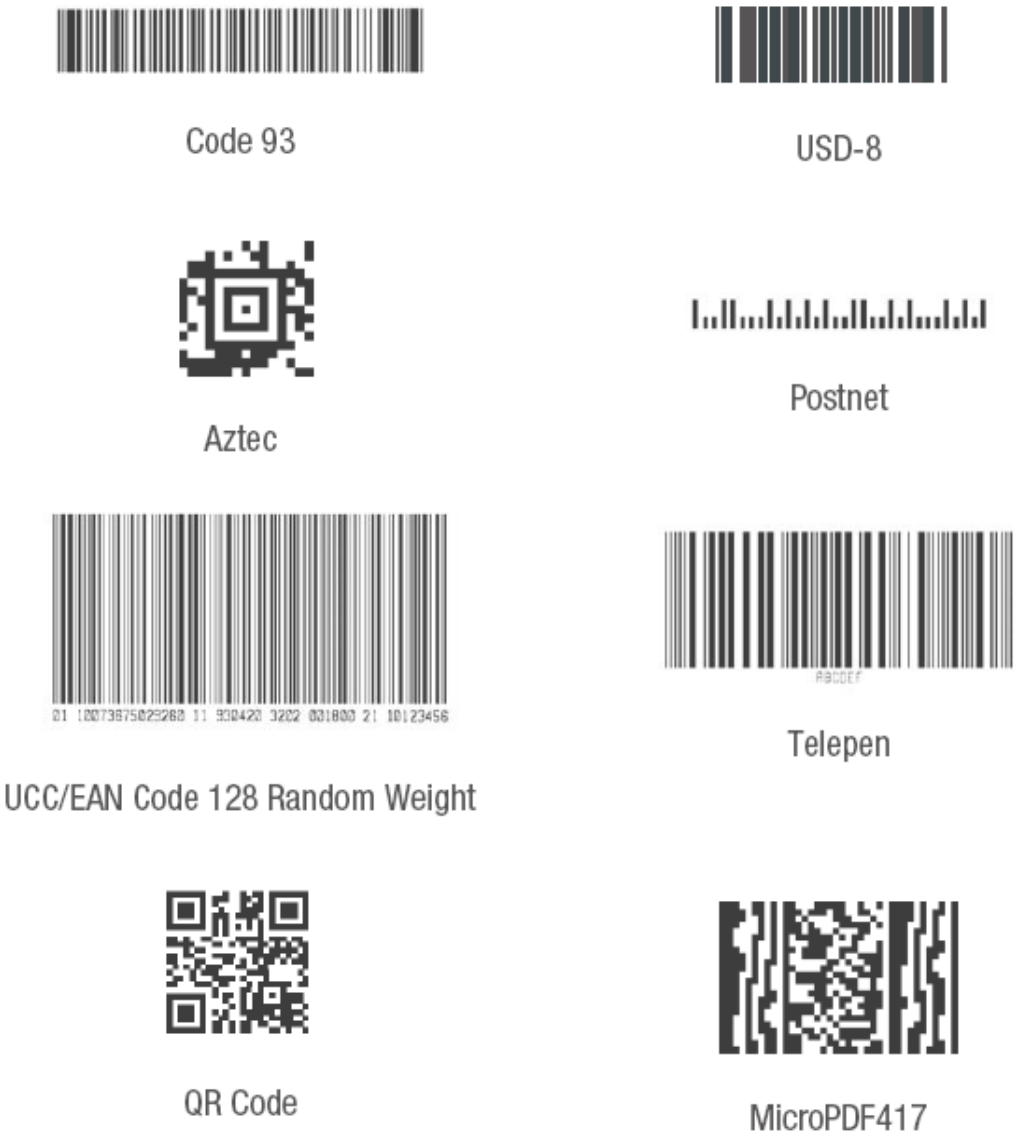


Figure 2.11 Other barcode symbologies

2.5 Advantages of Using Barcode Technology

Improved Data Accuracy

Bar coding provides virtually perfect data accuracy versus manual data entry. Studies have shown manual keying of data has an error rate of 1 error for every 300 characters entered versus 99% accuracy with integrated bar coding systems.

Faster Data Input and Handling

Data moves directly, and immediately, to a computer or other device for processing and storage.

Cost Benefits

Better customer service, capital and inventory management, faster response times, lower equipment and labor costs.

Flexibility

A wide variety of bar code equipment, label materials and experienced companies are available to assist in implementing Auto ID systems.

2.6 Basic Web-Based Technology (Platform)

2.6.1 Apache Web Server

Apache[1] is the most popular Web server on the planet. Netcraft places it as having more than 60% of the market in December 2000, as the most popular server since mid-1995, and as being more popular than all other servers combined since the beginning of 1999.

Apache is written with robustness, performance, security, and standards-conformance in mind. Apache is also extensible - its capabilities can be augmented through the inclusion of other modules that are not part of the core server. Apache's behavior can be tailored to the requirements of individual sites, allowing it to serve the needs of sites small and large. Administrators for modest sites often find the base server perfectly adequate with no modification whatsoever. Larger or more complex sites require individual customization; Apache gives the webmaster the flexibility needed to achieve this.

Two popular extensions for Apache are the mod_perl and mod_php modules that, in effect, embed the Perl and PHP interpreters into Apache so that it can execute scripts written in these languages directly rather than starting up a separate Perl or PHP interpreter to process them. The immediate result is a dramatic increase in script execution efficiency.

- is a powerful, flexible, HTTP/1.1 compliant web server implements the latest protocols, including HTTP/1.1 (RFC2616)
- is highly configurable and extensible with third-party modules

[1] http://httpd.apache.org

- can be customized by writing 'modules' using the Apache module API
- provides full source code and comes with an unrestrictive license
- runs on Windows NT/9x, Netware 5.x and above, OS/2, and most versions of Unix, as well as several other operating systems
- is actively being developed
- encourages user feedback through new ideas, bug reports and patches

2.6.2 PHP and MySQL

PHP[2] is a server-side scripting language for creating dynamic Web pages.

> *"PHP is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML."*

MySQL[3] is a relational database management system (RDBMS) based on SQL (Structured Query Language). First released in January, 1998, MySQL is now one component of parent company MySQL AB's product line of database servers and development tools.

Many Internet startups became interested in the original open source version of MySQL as an alternative to the proprietary database systems from Oracle, IBM, and Informix. MySQL is currently available under two different licensing agreements: free of charge, under the GNU General Public License (GPL) open source system or through subscription to MySQL Network for business applications.

MySQL runs on virtually all platforms, including Linux, Unix, and Windows. It is fully multi-threaded using kernel threads, and provides application

[2] http://www.php.net
[3] http://www.mysql.com

program interfaces (APIs) for many programming languages, including C, C++, Eiffel, Java, Perl, PHP, Python, and Tcl.

MySQL is used in a wide range of applications, including data warehousing, e-commerce, Web databases, logging applications and distributed applications. It is also increasingly embedded in third-party software and other technologies. According to MySQL AB, their flagship product has over six million active MySQL installations worldwide.

> *"The MySQL® database has become the world's most popular open source database because of its consistent fast performance, high reliability and ease of use. It's used in more than 10 million installations ranging from large corporations to specialized embedded applications on every continent in the world."*

2.6.3 Windows or Linux Operating System

This system will be developed under Windows or Linux operating system. These two operating systems can be the platform for the system to be developed.

2.6.4 Modules in Inventory System

There are several modules or features that will develop in this system. This feature will be the main function of the system.

i. *Asset Management*

By using this method, the system will manage all the asset and inventory in the FTMSK. It will record the inventory movement, where it is located and who is currently using the computer.

ii. *Hardware Audit*

This method will manage, scan and discover the specification of the computer hardware and the complete data of the inventory. Examples of data that will be collected are, manufacturer and model, processor, memory, network card, hard drives, audio and video type and other information in detailed.

iii. *Software Audit*

This method will manage the software that has been installed in the computer. All the data will be stored into the database. Examples of data that will be collected are, software licensed expiry date, operating system type, windows ID, and other info.

2.7 Similarities and Differences from Previous Research

2.7.1 Nonpoint Data Management System (Web-Based Inventory System)

E.H. Pechan & Associates, Pechan has developed a new web-based system for developing, storing, analyzing, reporting, and exporting non-point/area source air pollutant emissions data. The system is called the Nonpoint Data Management System (NDMS). NDMS is a data driven application which is supported by a backend database in which data are stored, edited, and retrieved by the user through the web interface. The web interface allows for multiple users to work in the system at the same time.

The similarity with the project is the implementation of web-based inventory system, but the data that is gathered is about air pollutant emissions data. (Manish Salhotra, Dr. Frank Divita, Stephen M. Roe, Randy P. Strait., 2005)

2.7.2 License Plate Inventory System

The License Plate Inventory System is a Scan NT-based database management system that tracks the vehicle inventory of a parking facility using license plate records.

License plate records are entered into the system by parking personnel using a *License Plate Inventory Handheld Computer* (LPI HHC), which accepts information such as license plate number and plate locale (state, province, etc.). This information can be entered either as vehicles enter a facility, or at facility spaces to capture information about where the vehicle is located (sub-lot, row, and space number).

The data stored in the LPI HHC shall be transferred to the Scan NT LPI system, which shall maintain in the Scan NT database LPI records for all lots in the system. When a vehicle exits a facility, an operator shall enter the license plate number of the vehicle into the Auditor PowerPad Fee Computer (with LPI software option). The Auditor PowerPad sends this data to the Scan NT computer. The LPI record for the vehicle in the Scan NT database is updated and is moved from the current inventory to the history inventory.

The similarity with the project is using the handheld computer to acquire the data. This will make the system more portable and the real time data will be captured. (Federal APD, Inc.)

2.7.3 Documentation and Inventory System Based on Four-Tier Architecture

This paper describes a model of designing such multi-tier, multi-user system which deals with large amounts of data. This paper also describes a system implemented, a system for managing telecommunication equipment and services data as a part of Operation and Support System (OSS) project. This system is based on four-tier architecture. Four-tier architecture is built from; database, application server, web server and thin client. Each of these components could be located on different machine. There a three groups of people involved in life cycle of system based on four-tier architecture, end user, operation and maintenance, and developer.

The similarity with the project is that this paper is explaining about system architecture of an inventory system and development of the system that will involved with a large amount of data. (Nenad Dragun and Robertina Jarak)

2.7.4 AssetManage Asset Tracking Software 2007

AssetManage keeps track of expenses related to an asset, creates inventories for multiple locations, and lets the users attach images, documents and hyperlinks to each asset. AssetManage can work with a Microsoft Access database file, or with any ODBC-compatible database, including SQL Server, MySQL and Oracle. With AssetManage 2007, users can organize all of their equipment, tools, furniture, computers, vehicles, and other assets for depreciation, tax, and insurance purposes.

It's easy to locate current assets, and determine what department or person they're assigned to. Users can search for text in any field in the inventory database. They can even search within the free-form notes fields. Scan and print barcode labels directly from the program. It also can track all of leases, warranties, and service agreements, as well as asset maintenance and repair history. By attaching all of these documents to the asset description, the users can ensure that critical information is immediately accessible when they need it.

The similarity about the system is an inventory system that using barcode for tracking the inventory and asset. The system developed to manage all types of asset.

2.7.5 Model Driven Development of Inventory Tracking System

An Inventory Tracking System (ITS) is a control and monitoring system that tracks the flow of goods and manages assets of warehouses. ITS is commonly used in warehouse management systems. The primary goal of an ITS is to provide convenient and reliable mechanisms to manage the movement and flow of inventory in a timely manner. An ITS should enable operators to configure warehouse storage organization criteria, maintain the set of goods known to the system, and track the inventory using GUI-based operator monitoring consoles.

This system discussed about the system model of an inventory system. Basically the system that will be developed have some similarity with this project. The idea of system model architecture is that it can be implemented to manage all type of assets in the system and how to track the asset. (Gan Deng, Tao Lu, Emre Turkay , 2003)

2.7.6 Silver Net Inventory Web-Based System

Silver Net Inventory System (SiNetInSy) is a complete web based inventory management system that performs the functions of Purchases, Sales and payments. This system will guide you through the creation of vendors list, purchase orders, products list, receiving lists, sales orders, invoices, sale and payment receipts. This is in addition to transfer orders between locations, customers and vendors balances and various types of reports for monitoring your business.

This system is a multi-user and multi-warehouse a web-based inventory system that is the feature of this project. (ExecutivePro, 2007)

2.7.7 Barcode Inventory System

This is a manual of procedures and user information barcode inventory system and bar code scanners. This manual explains how to use the bar code scanner in the system and the procedures of using the inventory system.

The similarity with the project is the used of the bar code scanner as a tool to gather data into the system. (UCSC Equipment Management, 2003)

2.7.8 Automated Asset Inventory System

This report was prepared for the *Arizona Department of Transportation* (ADOT), to explore options for implementation of a barcode inventory system to track fixed assets on an organization-wide basis.

ADOT currently has no department-wide automated fixed asset inventory process nor does it use barcode or other types of technology to track fixed assets on an organization wide basis. ADOT currently relies on manual inventory methods to maintain their fixed asset information in their fixed asset accounting software.

Information on label and identification (ID) technologies, barcode hardware and scanning technologies, software applications, and a Pilot implementation are discussed. Sample inventory time data was collected using the current ADOT fixed asset inventory process.

The similarity with the project that is the system is using the bar code hardware and technology in their system.

2.8 Summary

In this chapter, the researcher discussed related information regarding the research. These include explanations in definition of terms and concepts, review of several technologies that will use in the development phase. This chapter also reviewed the similar projects that have been developed.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter will cover the details description of methodology that will be used to develop this project completed and successful. Methodology is a part of the research that really important and will explain how the research is being done. The development of this project is based on the System Development Life Cycle (SDLC) or also known as Waterfall model, to achieve the entire objectives completely in a given period. The phases are project planning, system analysis, system design, implementation, testing, and maintenance. The following sequences of methodology phases are followed to complete the project successfully.

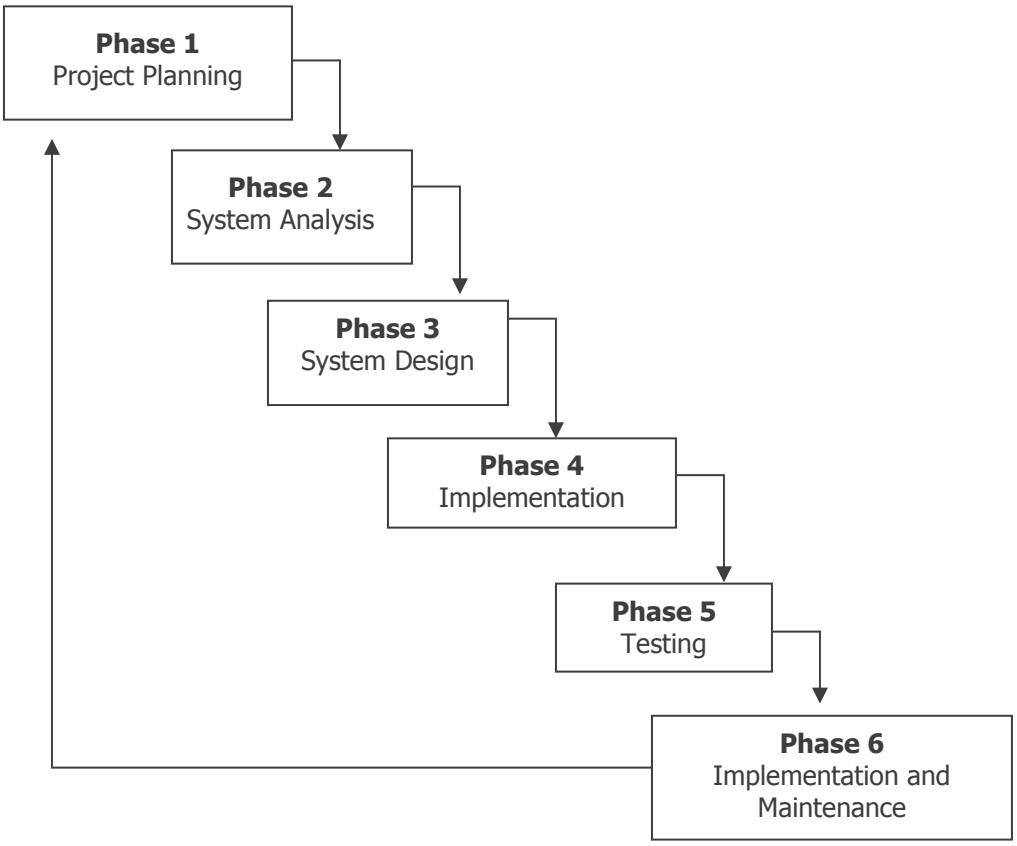


Figure 3.1 : System Development Life Cycle (SDLC) or Waterfall model

3.1.1 Phase 1 : Project Planning

The first phase of the methodology and research method is project overview and project planning. This stage requires researcher to define the scope, significance, objectives, goals and specification of this project as milestones to complete it. All of the objectives must take into consideration in order to make sure that it can be achieved successfully. Furthermore, this phase also needs the researcher to do some feasibility study and determine any possible constraints regarding the research topic. The main idea is to give a brief overview about the project that is going to be developed as a whole.

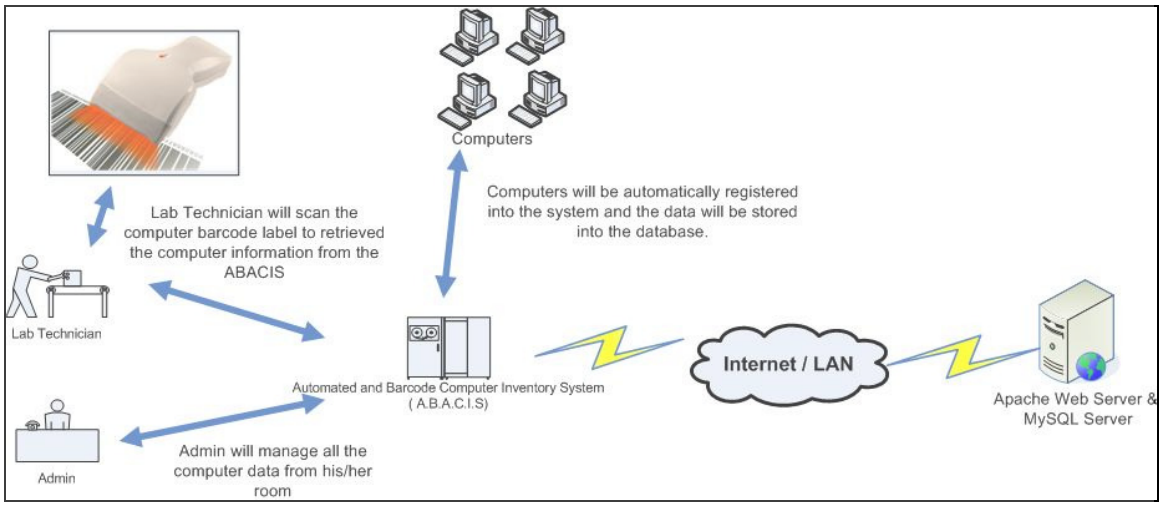


Figure 3.2 : Project planning and overview

3.1.2 Phase 2 : System Analysis

After all of the elements in the Phase 1 are completely defined, then the System Analysis phase needs the researcher to study the problem, deficiency, or new requirements and specifications in more detail. This stage is accomplished by doing the analysis on primary and secondary primary study.

In primary study, all of information that needed is gathered from interview which has been conducted with the user. While in secondary study, journals, articles, books and internet resources have been analyze and understood to get a better view of the problems and the project itself.

3.1.3 Phase 3 : System Design

Analysis gathers the requirements for the system. Then, the System Design phase will be designing the system based on the requirement from the System Analysis phase. Design focuses on high level design like, what programs are needed and how are they going to interact, interface design (what are the interfaces going to look like) and data design (what data will be required and as output from the system). During these phases, the software's overall structure is defined.

Admin and technician need access to the web browser that has internet access or Local Area Network (LAN) access to the web server. Using the computer inventory system for collecting data, and then the data will be store in the MySQL server.

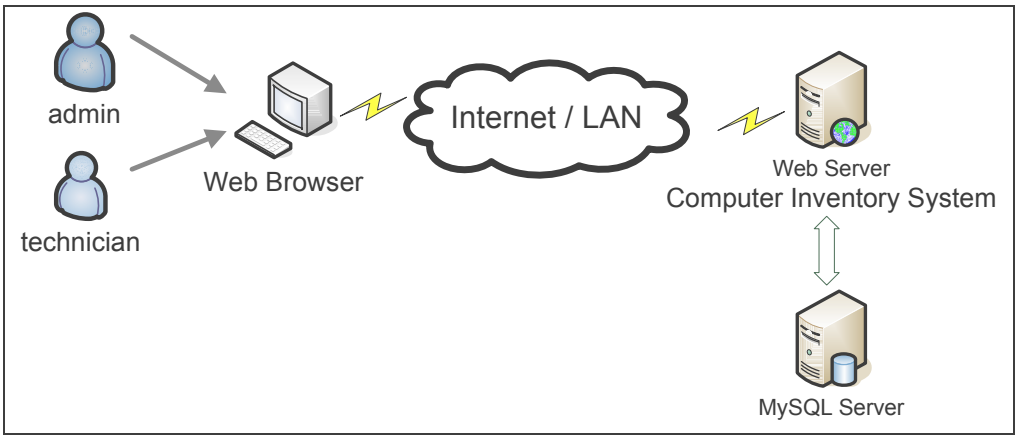


Figure 3.3 : Diagram of System Design

3.1.4 Phase 4 : Implementation

In this phase the designs are translated into code. Computer programs are written using a conventional programming language or an application generator. Programming tools like Compilers, Interpreters, and Debuggers are used to generate the code. Different high level programming languages like C, C++, Pascal, PHP and Java are used for coding. With respect to the type of application, the right programming language is chosen.

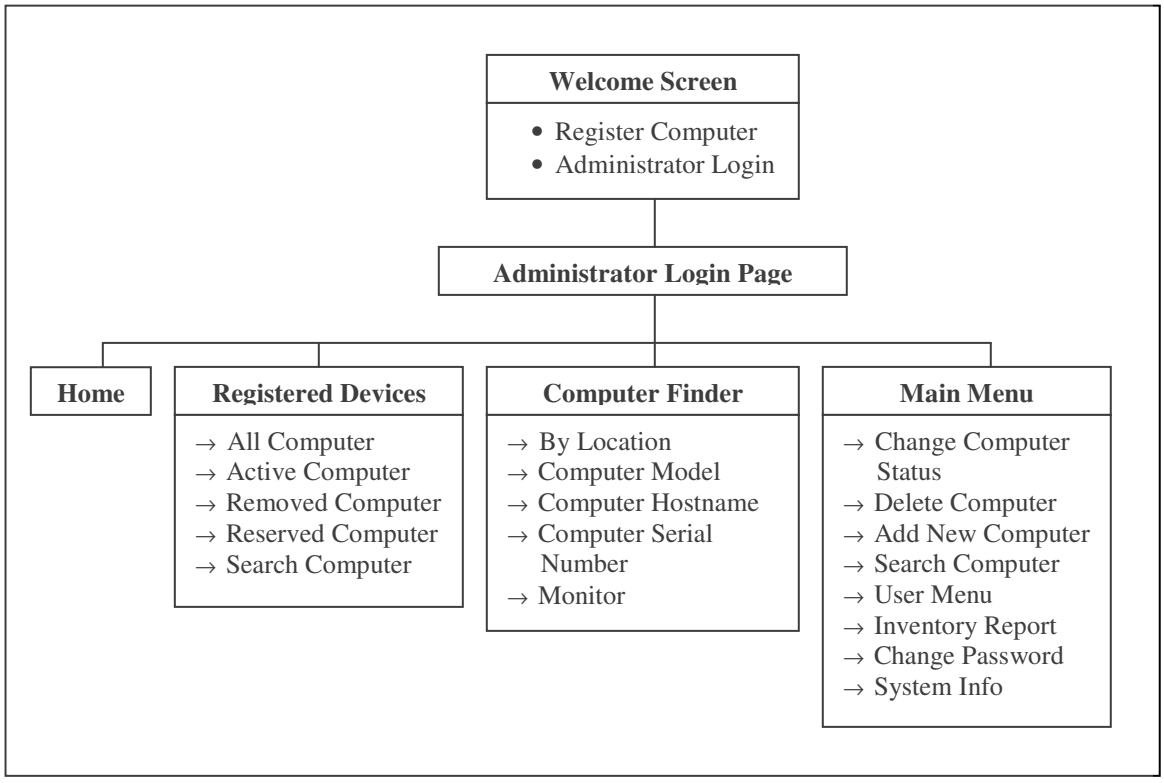


Figure 3.4 : System Flow

3.1.5 Phase 5 : Testing

In this phase the system is tested. Normally programs are written as a series of individual modules, this subject to separate and detailed test. The system is then tested as a whole. The separate modules are brought together and tested as a complete system. The system is tested to ensure that interfaces between modules work (integration testing), the system works on the intended platform and with the expected volume of data (volume testing) and that the system does what the user requires (acceptance/beta testing).

3.1.6 Phase 6 : Implementation and Maintenance

Inevitably the system will need maintenance. Software will definitely undergo change once it is delivered to the customer. There are many reasons for the change. Change could happen because of some unexpected input values into the system. In addition, the changes in the system could directly affect the software operations. The software should be developed to accommodate changes that could happen during the post implementation period.

3.2 Project Requirements

For the project, we need a hardware and software to complete this system. The hardware and software that required is:

3.2.1 Hardware

HARDWARE	TYPE
Processor	Intel Pentium 4 2.8 GHz
Hard Disk	80GB Maxtor
RAM	512 MB
Barcode Scanner	Scanner ARK CCD Barcode USB PS-700E

Table 2 : Hardware requirements

3.2.2 Software

The software that required is:

No	SOFTWARE
1	PHP programming
2	MySQL Database
3	PhpMyAdmin
4	Apache Web Server
5	Macromedia Dreamweaver 8 or Authoring Tools

Table 3 : Software requirements

3.3 Automatic Data Gathering Process

Researcher have study and analyze the component to be used for enabling the system to use the automatic data gathering function by using WMI,WSH and JavaScript. Example of JavaScript using the WMI and WSH component in the system as following in *takedata.js* file;

```
20  //computer
21  v_vendor="NONE";v_model="NONE";v_version="NONE";v_sn="NONE";
22  instances = objService.ExecQuery("SELECT Version,Vendor,Name,IdentifyingNumber FROM Win32_ComputerSystemProduct");
23  e = new Enumerator(instances);
24  if(!e.atEnd()){
25      d=e.item();
26      v_vendor=d.Vendor;v_model=d.Name;
27      v_version=d.Version;v_sn=d.IdentifyingNumber;
28  }
29  if (v_sn.match(/123456789/))    {
30      instances = objService.ExecQuery("SELECT * FROM Win32_BaseBoard");
31      e = new Enumerator(instances);
32      if(!e.atEnd()){
33          d = e.item();
34  //Product not exists in CIM Schema
35          v_vendor=d.Manufacturer;v_model=d.Product;
36          v_version=d.Version;v_sn=d.SerialNumber;
37      }
```

Figure 3.5 : A few line of JavaScript coding for detecting the computer version.

```
//OS
instances = objService.ExecQuery("SELECT * FROM CIM_OperatingSystem");
e = new Enumerator(instances);
    d=e.item();
    writehidden("os",d.Caption);writehidden("hostname",d.CSName);writehidden("os_date",d.InstallDate.substr(0,8));
    v_ram=Math.round(d.TotalVisibleMemorySize/1024);

//RAM
instances = objService.ExecQuery("SELECT Capacity FROM CIM_PhysicalMemory");
e = new Enumerator(instances);size=0;
if(!e.atEnd())  {
    for(e;!e.atEnd();e.moveNext()){
        d=e.item();
        size=size+parseInt(d.Capacity);
    }
    v_ram=(size/1048576);
}

//myparam+="&ram="+v_ram;
writehidden("ram",v_ram);

//processor
instances = objService.ExecQuery("SELECT Name,MaxClockSpeed FROM CIM_Processor");
e = new Enumerator(instances);
i=0;
for(e;!e.atEnd();e.moveNext()){
    d=e.item();
    writehidden("proc["+i+"]",d.Name);writehidden("proc_speed["+i+"]",d.MaxClockSpeed);
    i++;
}

//vga
instances = objService.ExecQuery("SELECT VideoProcessor FROM Win32_VideoController where VideoModeDescription is not null");
e = new Enumerator(instances);
if(!e.atEnd())  {
    d=e.item();
    writehidden("vga",d.VideoProcessor);
}
```

Figure 3.6 : Line of coding for detecting Operating System, RAM speed, processor speed, and VGA Display.

3.4 Summary

This chapter concentrated on research methodology that had been taken in order to carefully planning, analysis, design and implementation of the project. Various fields of researches and analysis were conducted in this methodology phase. The methodology also leads a step by step procedure in making the project a success. All the information gathered was used to develop the project.

CHAPTER 4

RESULTS AND FINDINGS

4.1 Introduction

This chapter provide the result and finding that researcher got based on the requirement gathered in the knowledge acquisition phase and what the outcome from the developed system. This chapter will explain most of the system flow and the feature that have been developed. The result of the experiment that has been done in testing phase also will be explained in this chapter.

4.2 Data Structures

Computer Inventory System's data structure consists of 32 tables with relationships described in table below.

Num.	Column Name	Data Type	Explanation
1.	computerid	unsigned integer, auto increment	Computer identification number
2. (a)	modelid	unsigned integer, auto increment	Component model's identification number
(b)	RAM[modelid]	unsigned integer	RAM size in megabyte
(c)	FLOPPY[modelid]	unsigned tiny integer	1:floppy drive exist; 0:floppy drive not exist
(d)	OWNERSHIP_MONITOR[modelid], OWNERSHIP_PERIPHERAL[modelid]	unsigned integer	Ownership model ID.
3.	name	variable-length character	Component model's name
4.	registerdate	date	Item registration date
5.	removed	date	Item's removal date
6.	serialnumber	31 character	Item serial number
7	reserved	1 character	Marking computers that are reserved. "Y" means reserved
8.	COMPUTER_SYTEM[hostname]	31 character	Computer's hostname

9.	COMPUTER_SYTEM[user]	31 character	Computer's user name
10.	COMPUTER_SYTEM[domain]	unsigned integer	Network domain category ID.
11.	COMPUTER_SYTEM[non-id]	unsigned integer	Identification number for non-branded computers
12.	COMPUTER_SYSTEM[pnpmon]	1 character	Indentification whether the computer support PnP Monitor
13.	MODEL_CATEGORY[vendor]	max. 31 characters	Computer vendor's name
14.	MODEL_CATEGORY[model]	max. 63 characters	Computer model
15.	MODEL_CATEGORY[version]	max. 31 characters	Computer model's version
16.	PROCESSOR_CATEGORY[speed]	unsigned integer	Processor maximum speed in MHz, not current speed
17.	HARDDISK_CATEGORY[size]	unsigned small integer	Harddisk size in GB
18.	MONITOR_CATEGORY[pnpid]	7 characters	Monitor PnP ID that is retrieved from EDID
18.	NIC[mac]	17 characters	MAC Address
19.	NIC[ipaddress]	max. 15 characters	Adapter's first IP Address
20.	MONITOR[snlock]	1 character	Value 'Y' means the PnP monitor is detected automatically from EDID, hence its serial number may not be edited.
21.	LOCATION[location]	50 characters	Location of the computer assigned and placed. The owner of the computer is based on the location.

Table 4 : Data Structures of Computer Inventory System

4.3 System Interfaces

4.3.1 Register Computer

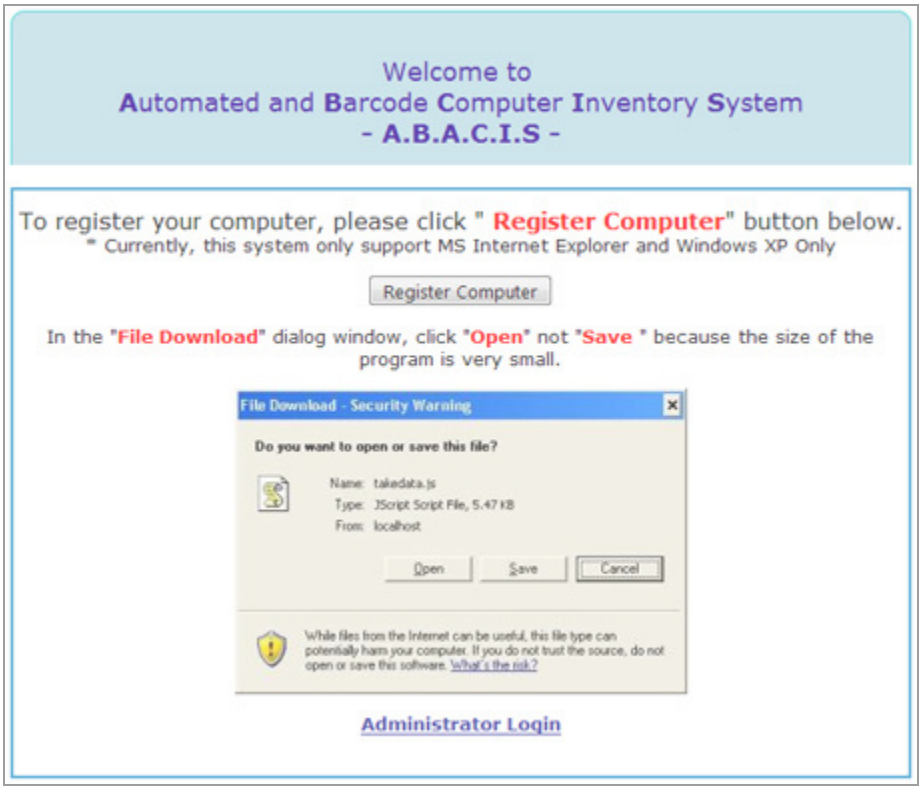


Figure 4.1 : Register Computer Interface

This is the first screen when browse to the system. This will allow the user for automatically, adding the computer data into the database system. User only needs to click the “Register Computer” button and Run the JavaScript file. The script will automatically scan and detect the current computer hardware and software.

4.3.2 Login Page

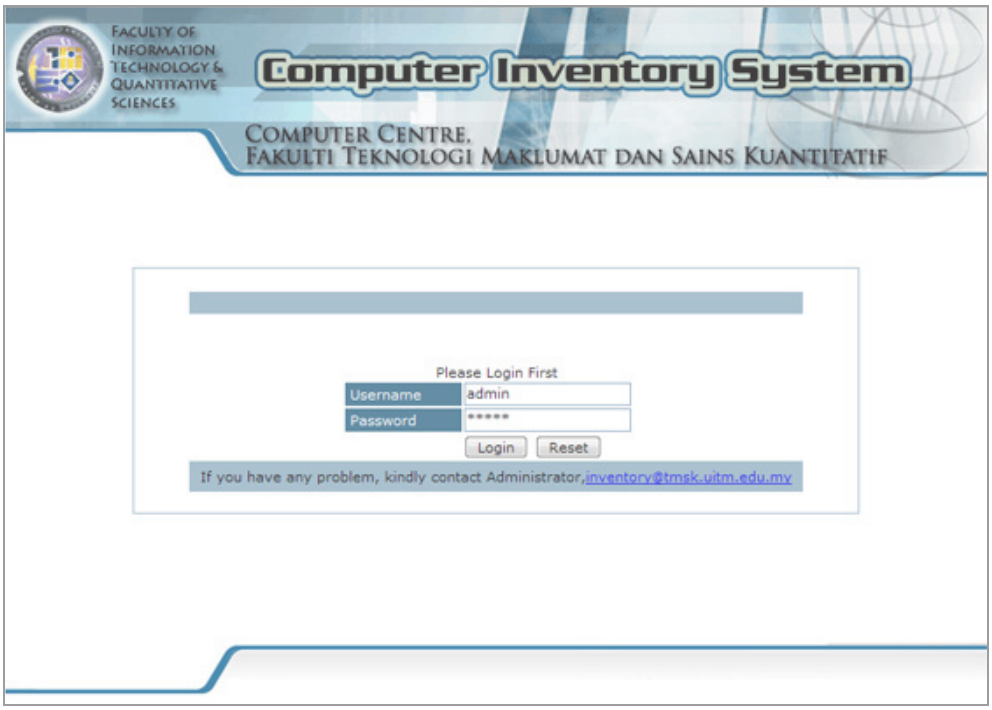


Figure 4.2 : Login Screen

It is not allowed to all the staffs to access the whole network, each staff have their user name and password in the system. For security reason, the staff working with the system, must be login in every time it is used. Only the administrator can access the whole system include the maintaining function.

4.3.3 Main Page

Figure 4.3 : Home Interfaces for Administrator

The system provides admin interfaces after successfully login into the system. Administrator have the privileged to access all registered computer in the system, including all active computers, deleted computers and reserved computers. Administrator also can change the computer or devices status, model name of the specific computer.

4.3.4 Search Computer

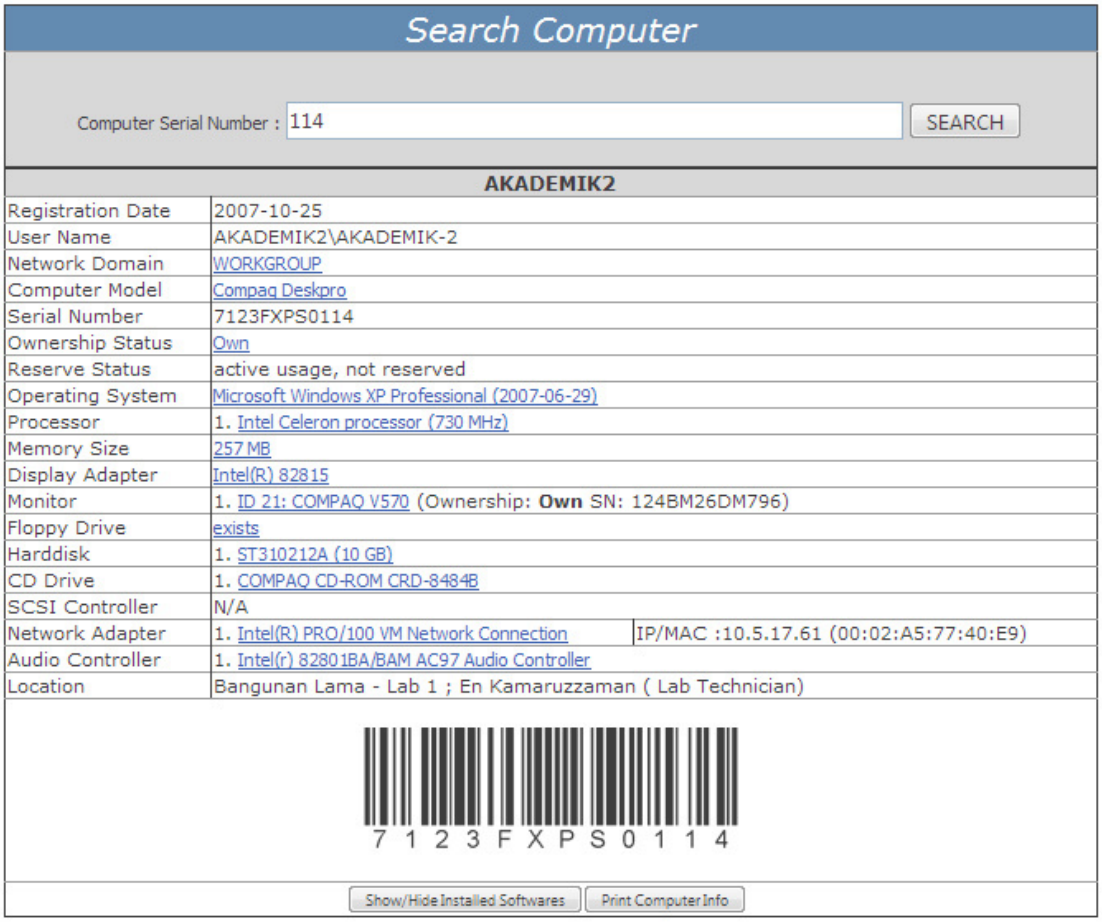


Figure 4.4 : Search Computer Interface, result with barcode label

This is the result interface that has been generated, after the user scans the barcode label on the computer. The system will automatically retrieve the data from the database and display the result.

4.3.5 Management Page

Figure 4.5 : Main Menu for Administration

The Management Page only can be accessed by admin level. Admin can have privileged on the menu, as admin can change the computer status, delete computer info, add new computer, generated inventory report and also user management.

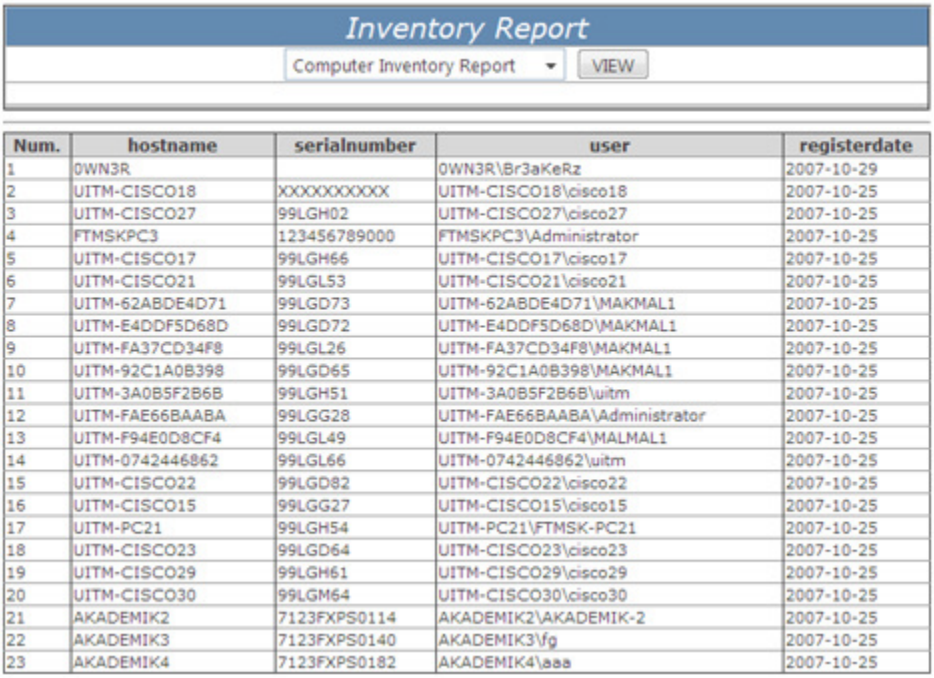
4.3.6 List of Computers Registered

Figure 4.6 : List of the registered computer in the system

The computer that have been registered or automatically add into the database can be display in the system. All the important information such as computer hostname, computer username, domain, date of registration, and other will be displayed.

4.4 Report Generated From System

4.4.1 Computer Inventory Report



Inventory Report

Computer Inventory Report | VIEW

Num.	hostname	serialnumber	user	registerdate
1	0WN3R		0WN3R\Br3aKeRz	2007-10-29
2	UITM-CISCO18	XXXXXXXXXX	UITM-CISCO18\cisco18	2007-10-25
3	UITM-CISCO27	99LGH02	UITM-CISCO27\cisco27	2007-10-25
4	FTMSKPC3	123456789000	FTMSKPC3\Administrator	2007-10-25
5	UITM-CISCO17	99LGH66	UITM-CISCO17\cisco17	2007-10-25
6	UITM-CISCO21	99LGL53	UITM-CISCO21\cisco21	2007-10-25
7	UITM-62ABDE4D71	99LGD73	UITM-62ABDE4D71\MAKMAL1	2007-10-25
8	UITM-E4DDF5D68D	99LGD72	UITM-E4DDF5D68D\MAKMAL1	2007-10-25
9	UITM-FA37CD34F8	99LGL26	UITM-FA37CD34F8\MAKMAL1	2007-10-25
10	UITM-92C1A0B398	99LGD65	UITM-92C1A0B398\MAKMAL1	2007-10-25
11	UITM-3A0B5F2B6B	99LGH51	UITM-3A0B5F2B6B\uitm	2007-10-25
12	UITM-FAE66BAABA	99LGG28	UITM-FAE66BAABA\Administrator	2007-10-25
13	UITM-F94E0D8CF4	99LGL49	UITM-F94E0D8CF4\MALMAL1	2007-10-25
14	UITM-0742446862	99LGL66	UITM-0742446862\uitm	2007-10-25
15	UITM-CISCO22	99LGD82	UITM-CISCO22\cisco22	2007-10-25
16	UITM-CISCO15	99LGG27	UITM-CISCO15\cisco15	2007-10-25
17	UITM-PC21	99LGH54	UITM-PC21\FTMSK-PC21	2007-10-25
18	UITM-CISCO23	99LGD64	UITM-CISCO23\cisco23	2007-10-25
19	UITM-CISCO29	99LGH61	UITM-CISCO29\cisco29	2007-10-25
20	UITM-CISCO30	99LGM64	UITM-CISCO30\cisco30	2007-10-25
21	AKADEMIK2	7123FXPS0114	AKADEMIK2\AKADEMIK-2	2007-10-25
22	AKADEMIK3	7123FXPS0140	AKADEMIK3\fg	2007-10-25
23	AKADEMIK4	7123FXPS0182	AKADEMIK4\aaa	2007-10-25

Figure 4.7 : Computer Inventory Report

In the Computer Inventory Report, all data of the computers that have been registered in the system will be listed in the report.

4.4.2 Software Inventory Report

Num.	Software	Version
1	Adobe Flash Player Plugin	9.0.47.0
2	ATI - Software Uninstall Utility	6.14.10.1014
3	ATI Display Driver	8.282.1-060802a-035868C-ATI
4	C-Media 3D Audio	
5	Canon PIXMA iP1000	
6	Codec Pack - All In 1 6.0.3.0	
7	DU Meter	3.50 Build R2822
8	DynDNS Updater 3.1	3.1
9	Folder Access 2.0.0 Free Version	
10	Kaspersky Anti-Virus 7.0	7.0.0.125
11	Magic ISO Maker v5.4 (build 0251)	
12	Microsoft .NET Framework 1.1	
13	Microsoft .NET Framework 2.0	
14	MozBackup 1.4.7	
15	Mozilla Firefox (2.0.0.8)	2.0.0.8 (en-US)
16	Nero 6 Ultra Edition	
17	Notepad++	
18	SiS 900 PCI Fast Ethernet Adapter Driver	
19	Vista Visual Pack 7.0	
20	VideoLAN VLC media player 0.8.6c	0.8.6c
21	Winamp (remove only)	
22	Windows Media Format 11 runtime	
23	Windows Media Player 11	
24	WinRAR archiver	
25	XAMPP 1.6.3a	
26	Yahoo! Messenger	
27	Macromedia Dreamweaver 8	8.0.2
28	Optimizer XP	3.50.2003.XG
29	DAEMON Tools	3.41.0
30	Java(TM) 6 Update 3	1.6.0.30
31	Macromedia Extension Manager	1.7.240
32	Disc2Phone	1.3.0.106
33	ATI Catalyst Control Center	1.2.2405.30455
34	Microsoft Office Professional Edition 2003	11.0.5614.0
35	Microsoft Office Visio Professional 2003	11.0.3216.5614
36	ATI Parental Control & Encoder	3.0
37	Adobe Reader 8.1.0	8.1.0

Figure 4.8 : Software Inventory Report

In the Software Inventory Report, all data of the software that have been installed in the computer, that have been registered in the system will be also listed in the report. Admin can refer this information to manage and update the latest version of the software.

4.5 Analysis from Implementation Evaluation

During the testing phase and the implementation phase, researcher has done an experiment to compare the efficiency of the system with the manual data collection. In the FTMSK, there are two building that is, existing building and additional building. The existing building has 12 computer labs and the additional building has 10 computer labs. Each lab has approximately 30 computers. The experiment is located in Lab 1 in the existing building in the faculty. The purpose of this experiment is to measure the time taken to collect data from each computer. This will calculate time need for each computer and will be comparing with the time need using the system. In the Lab 1, there are 30 computers but only 10 computers has been used as sample size and each computer were switch on and connected to the internet.

Computer ID #	Manual	A.B.A.C.I.S
1	5	1.1
2	4	0.8
3	5	0.8
4	3	1.0
5	3.5	1.2
6	5	0.9
7	4.5	1.3
8	3	1.1
9	4	0.8
10	5	1.4
Total PC = 10	42 minutes	10.4 minutes

Table 5 : Results from experiment evaluation

Average time need for manual data collection for one computer;

42 minutes / 10 computers (sample size)
= 4.2 minutes

Average time need using the A.B.A.C.I.S data collection for one computer;

10.4 minutes / 10 computers (sample size)
= 1.04 minutes

From the result, researcher has found that the time taken for using ABACIS is faster than the time taken by using the manual data collection. From the result of the experiment, the researcher has concluded the system is more efficient than the manual data collection.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

5.1 Introduction

Generally, this chapter will simplify the entire system development that is being done. The result of the development is viewed roughly and entirely for evaluation process with the project objectives that have been set. The strength and weaknesses of the system is also discussed. Several positive suggestions to improvise system weaknesses have also been find out and defined in this chapter.

5.2 Results and Achievement

The system has successfully developed and tested the efficiency based on the experiment. During the implementation and testing phase, researcher has taken an experiment to test the efficiency of the system compare to the manual data collection. From the result, the time taken for data collection is faster than using the manual data collection. This will save more time in doing the computer inventory and data collection. From here, researcher can conclude that objectives of the system have been achieved.

5.3 System Limitation

Although the system is in the testing phase and most of the function is working, but some of the system still need more adjustment and development plans in term of implementation of the system in the real environment. The system still has limitation based on the current development and the following is the system limitation that has needed more time to develop;

a. **Only Windows-based computer can be detected.**
   Currently, the ABACIS only detect the computers with Windows-based operating system. The UNIX, Linux and Apple computer can not be used by this system. But, there is an alternative solution in ABACIS; admin can store the information of the computer by using the manual Add New Computer function in Management Page.

b. **Only compatible with Microsoft Internet Explorer for automatic data gathering function.**
   There are some browsers that are reading the JavaScript file, not running the file. This is the limitation of the current system and need more time to test.

5.4 Recommendations

The researcher has divided into three different recommendations that can be done to improvise the system. There are:

a. Using network probing technique.

There is another alternative way in collecting data that can be used in computer inventory system. This technique using network probing that will detect any connected computer and devices based on their IP address.

b. Develop small client program

There are some of the computer inventory system that are using client program to track the computer data in real time. This feature will enable the system to monitor the accurate data of the computers.

c. More functions and modules.

There are more functions and modules that can be developed under this system but it need more time and research. This system also can be integrated with the other system, so it can be manage with central system.

5.5 Conclusions

As the conclusion, this web-based computer inventory system will be using barcode scanner and integrate with the system to gather info and managing the asset of the FTMSK with efficient.

By using the system, the staff of the Computer Office in FTMSK will no longer facing hard time to manage and tracking of the computer located in the faculty.

Hopefully, this system will give more benefit to the user and will increase productivity and efficiency of the management in faculty.

APPENDIXES

REFERENCES

Federal Signal Corporation, *License Plate Inventory System*
Retrieved from
http://www.federalapd.com/PRODUCTS/Specifications/LPIspec.pdf

Liberty Street Software, *AssetManage Asset Tracking Software 2007*
Retrieved http://www.assetmanagesoftware.com/

Aniruddha Gokhale, *Model Driven Development of Inventory Tracking System*
Retrieved from http://www.cs.wustl.edu/~schmidt/PDF/ITS_Modeling.pdf

Manish Salhotra, Dr. Frank Divita, Stephen M. Roe, Randy P. Strait,
Pechan Introduces New Web-Based Inventory System
Retrieved from
http://www.epa.gov/ttn/chief/conference/ei15/session7/roe.pdf

ExecutivePro, *Silver Net Inventory Web based system*
Retrieved from http://www.executivpro.com/silvernet.pdf

Nenad Dragun, Robertina Jarak, *Documentation and Inventory System Based on Four-Tier Architecture* retrieved from
http://ieeexplore.ieee.org/iel5/9452/30001/01372390.pdf?tp=&arnumber=1372390&isnumber=30001

Bar Code Inventory System Manual
Retrieved from http://equipment.ucsc.edu/info/BarcodeProcedure.pdf

Andreas Schiffer, April 2006*, Automated Asset Inventory System*
Retrieved from
http://www.azdot.gov/TPD/ATRC/publications/project_reports/PDF/AZ580.pdf