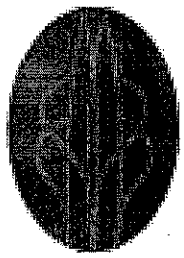


UNIVERSITI
TEKNOLOGI
PETRONAS

SALES VS COSTS (SVC) REPORT GENERATION SYSTEM

LUY PAGNA
7190

INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNOLOGI PETRONAS

JULY 2007

Sales VS Costs (SVC) Report Generation System

by

Luy Pagna

Dissertation submitted in partial fulfillment of
the requirements for the
Bachelor of Technology (Hons)
(Information and Communication Technology)

JULY 2007

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

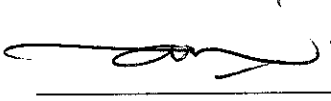
Sales VS Costs (SVC) Report Generation System

By

Luy Pagna (7190)

A project dissertation submitted to the
Computer and Information Sciences Programme
Universiti Teknologi PETRONAS
In partial fulfillment of the requirement for the
BACHELOR OF TECHNOLOGY (Hons)
(INFORMATION AND COMMUNICATION TECHNOLOGY)

Approved by,



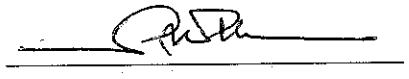
(Pn. Aliza Bt Sarlan)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

July 2007

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.



LUY PAGNA

ABSTRACT

Sale Versus Cost (SVC) Report Generation is a standalone application developed by using Microsoft Visual Basic 2005 to generate Sale and Cost report for MITCO in order to solve the current problem of using Lotus Approach. Currently, Generating the SVC report is monthly work of Financial Department of the company which they have been facing the difficulties. Once data has downloaded from SAP database, they are still required to do "massaging" to get the usable information and good report format. This massaging makes them spend more time and effort, and they still meet difficulties of matching the actual cost with actual in the following month, controlling exchange rate difference between SAP and Maybank and calculating subtotal export/domestic/international sales and so on. It means Lotus approach has been used currently is unstable and there are many steps to follow complicatedly. That why, SVC Report Generation is proposed and developed in order help the daily work of Finance department to increase their performance in generating the SVC report by reducing time of massaging original data from SAP (Lotus123). This document will also describe the process of development life cycle with Prototype methodology in order to understand the process and method of developing this system. Prototype methodology is one popular and effective methodology used to develop, both small and big system, to ensure the system can meet the user requirements and satisfy the user. In this meaning, the user could take more advantages from this system in order to generate the Sale Vs Cost report easily and effectively.

ACKNOWLEDGEMENT

First off I would like to thank ICT/BIS lecturer and FYP supervisor Mrs. Aliza Bt Sarlan, who has taught and guided a lot about the project. I really appreciate her kindness that bring me to this level and give the direction to me to do the next step of this project.

Furthermore, a big thanks PETRONAS, the biggest and excellent company in Malaysia, who has supported the students, especially, Cambodian students, both budgets and equipments. Your kind behaviors have been improving Human Resource in Cambodia that we do need to develop social economy in Cambodia and ASEAN. I am one of PETRONAS scholar can't forget your merit in which kept deeply in my heart.

And finally, I'd like to thank my lovely family in Phnom Penh, Cambodia, for their unwavering encouragement and support; this includes parents and three sisters. I have never felt lonely when I heard your call.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1. Background

SALES VS COST (SVC) Report Generation application is kind of application developed to generate report of sale for MITCO. MITCO is a strong Malaysian International company named Malaysian International Trading Corporation Sdn Bhd. MITCO owns all marketing right to PETRONAS plants which its business is marketing and trading of petrochemicals and general merchandise. For trading arrangement, MITCO takes position and assumes the trading risks. Any margin or loss in absorbed by MITCO. Trading is mostly done with third parties.

This application will help the Finance development to generate sale report dynamically and effectively that currently; SVC report is generated in SAP and then uploading of SVC report into Lotus approach. Therefore SVC Report Generation system is developed in order to help Finance work more easy to manage sale data and generate good quality of sale report

1.2. Problem Statement

1.2.1. Problem Identification

MITCO is one of biggest company in Malaysia which sales products such as Chemicals, Polymers and general Merchandises in many countries. So there are lots of sale transactions and data that need to manage in effective way.

Base on the current implementation, Finance department have to update sales and product cost information in Tempest. Then Operations will arrange for documentation and proceed with loading and forward the cost invoices to Finance. Finance will extract this information from Tempest relating to the cost invoice. Base on sales invoice, Finance will extract sale information from Tempest into SAP. In

order to generate the report, Finance department has to create SVC Report in SAP. Because SVC report in SAP is not well format and structure so this SVC report will be upload into Lotus approach in order to organize data in special format of report they need. There are many steps and complicated works in processing SVC report generation. Those spend more time to generate this report in a required by updating and managing data of SVC report in SAP and do more editing SVC report in Lotus123 in order to get the final of Sale report product. As the result, SVC Report Generation is developed to solve the problem above.

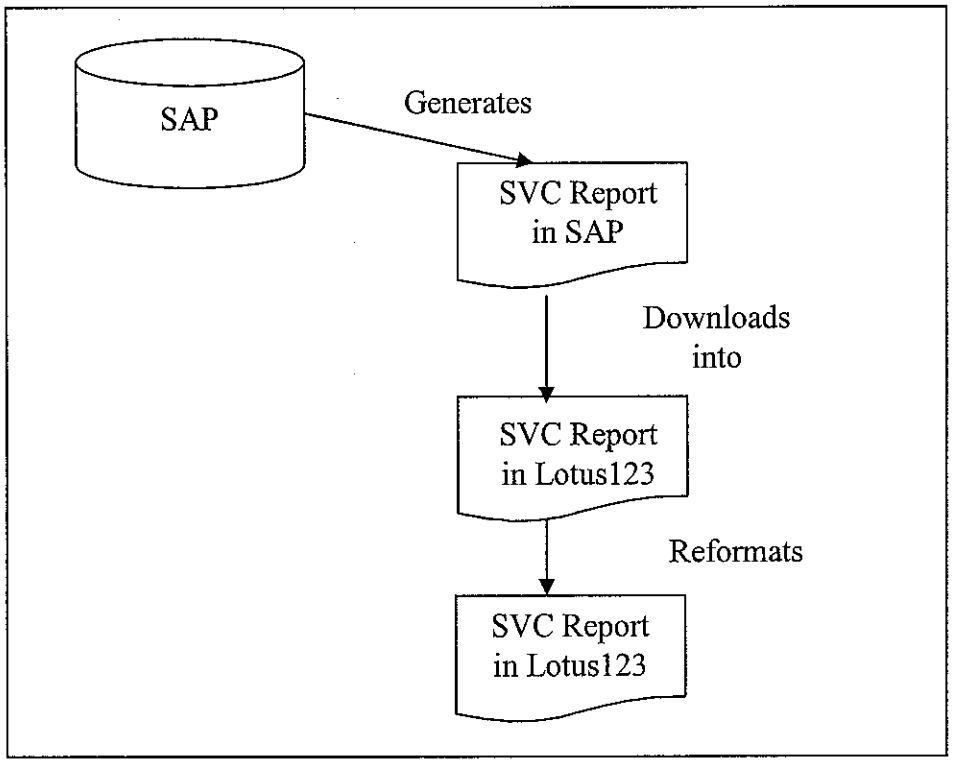


Figure 1: Current Report Generation Process

Current method of Reporting in Summary

- Need to download information from SVC report (Lotus 123) to Lotus Approach
- However, prior to that, the SVC report needs to be further "massaged". Eg:
    - The common secondary costs need to be allocated to each transaction based on sales ratio.
    - If there is cost reflected in SVC report but no sales (because sales already captured in previous month), then need to manually insert the

info on load port etc. this is because each transaction must have a complete information, and to be reflected under one line.

Limitation of Lotus Approach

- Long turnaround time
- Needs further massaging to the numbers prior to download
- Unstable
- Doesn't read any information from Tempest, only from SVC report (Lotus123)

1.2.2. Significance of the Project

SVC Report Generation application aims to provide Finance department of MITCO the effective report generation system in order to help their performances, especially, generate the good quality sale report of company. This project will study the user requirements and try to complement all these requirements by delivering the high interactive system, which is easy to use and performance their work in report generation. Furthermore, this system will provide the convenient method for user to get data from SVC report in SAP into new database system and from this database, user can use to manipulate and generate the report dynamically.

1.3. Objective and Scope of Study

1.3.1. Objective

1. To develop the application that is able to capture data from SAP report to put in MS Access.
2. To develop the high interactive system could generate dynamic report of SVC.

1.3.2. Scope of Study

This project focuses on developing the application that could capture data from SAP report to input into MS Access. Additionally, the user could generate SVC of sale report from this source. In order to achieve these, some studies are mainly required as below:

1. To study and understand the business requirements of user.
2. To find the suitable solution of capturing data from SAP report to input in MS Access.
3. To use the suitable technology to develop this system.
4. To integrate VB2005 to manipulate data and generate a report in MS. Excel

1.3.3. Feasibility of the Project

This project will be developed within the time frame of 28 weeks. The first 14 weeks are allocated time frame will be used to carry out preliminary research, system analysis, and deliver the first prototype of the system, and another 14 weeks are used to develop and implement the project such as coding, testing and installing system. The methodology used and activities involve in each phase will be explain in detail under next chapter.

CHAPTER 2

LITERATURE REVIEW

2.1 Report Generation System

Report is a written document to describe or brief information in the meaning for a specific purpose and format. Report has given different definitions including Hill and Dale stated Report as "a written document describing the findings of some individual or group". According to www.umkc.edu/registrar/sis/glossary.asp, Report is the presentation of a formatted collection of information; can be presented on paper, on the web, on diskette, or online. The printed record of a committee's actions, including its votes, recommendations, and views on a bill or question of public policy or its findings and conclusions based on oversight inquiry, investigation, or other study. In addition, the Report is can be defined as a formatted and organized presentation of data. Most database management systems include a report writer that enables you to design and generate reports. And how about Report Generation System?

Report Generation System is a kind of application developed to generate the report based on corporation' requirements in order to help them to make the report easily and effectively. For instant, SVC Report system is developed for MITCO Company to generate their sale and cost report much faster and easy to massage in a particular format. It is undeniable that report generation is one of the most important tasks in many companies regardless of the size of the company. Hong and Peck (2003) stated "A good report generation mechanism can increase a company's productivity in terms of effort and time." Report Generation systems are so important in company to summarize the results or performances of the company and give the necessary information for company to control and improve the works. Particularly, the system could generate the good quality report with general format and model in order to achieve efficiency and effectiveness of specific task.

Furthermore, this is more obvious in some startup companies, which normally use some in- house report generators. Application development could be complex and thus software developers might require substantial efforts in maintaining application program code. In addition, most of the report generators use a different kind of format to store the report model. Especially, that kind of system may be developed by using different technologies. Some may be developed standalone application and some are web application in order to perform their work in specific purpose and requirement. As this project tries to develop standalone application to generate the dynamic report in high performance and interactive user interface, there are some system might use web application. For example, the Excel report generation system is developed in ASP.NET using VB2005 and DataGrid by Yagulasamy, Software Engineer for Protech Solution Inc. this system could generate the Excel report base the web browser interface. In addition, Dynamic Excel Reports with ASP was developed by Steven Smith in 2003 to convert table data into an Excel spreadsheet format for the user.

On another hand, Hong and Peck (2003), lecturer of University of Malaya, have try to use XML driven and component to generate the report. As they said "An application is no longer considered an enterprise- level product if XML is not being used elsewhere." XML driven and Component-based development approach to report generation with the purpose of promoting portability, flexibility and generality. In this approach, report layout is specified using user-defined XML elements together with queries that retrieve data from different databases. A report is output as an HTML document, which can be viewed using an Internet browser. The report generation mechanism of this approach supports heterogeneous database models, and therefore, reports generated by an application are independent from other database models. The report layout and content can be specified using XML elements, which eventually made up the report schema. The XML report schema can be used to help application developers create reports even much more faster as well as code maintenance can be relatively done much more easier.

In common practice, a big company normally uses more than one report generator to cater for their reporting needs. The lack of a generic format of report model has the impact that reports generated in one report generator very unlikely work on another

report generator due to the proprietary format used by different vendors (Hong and Peck (2003), lecturer of University of Malaya). This is also become a problem in company, that why there are many Report Generation systems or applications were developed both internal and external company in order to improve effectiveness of work. For instance, Software Administration Kit (SAK) is a unique all-in-one solution for customer and order tracking, form letter and e-mail submission, sales report generation, version management and order fulfillment. SAK has been designed to meet the requirements of shareware authors and other e-commerce vendors and Report Management as example. However, all these system are developed base on real requirements and specific company. Thus SVC Report Generation system is developed to answer the requirements of Finance department of MITCO to help them generate good quality report in dynamic, general model and well format, especially, Rich Client Interface of the system would improve the interaction between user and system easily.

2.2 Read Data From Spreadsheet

The main challenging work in developing this system is to take all data in spreadsheet file such as Lotus123 or MS Excel to input in MS Access. This function is important to help user to save time after they have generated the report in Lotus123 format and wanted to input to new database system. It is the convenient way which the system could perform in order to achieve its convenient and effective function.

Nowadays, they are many tools and techniques used to solve this issue. In particular, this project have proposed MS Visual VB 2005 programming language to capture the data from spreadsheet, so it requires finding the connector between these in order to access a spreadsheet file. The system can read data from each cells and pass it input particular field in database designed in MS Access. On the other hand, there are alternative ways to do this, for example, using SAS 6.08 for Windows: transferring SAS data and SAS graphics to Windows-based software applications, stated by S. Nelson (Winter 1994). Dynamic Data Exchange is a method of taking data from one application on a PC and "linking" it with another application. Thus, DDE allows us to read or write data between the SAS System and another

application that also supports DDE. DDE can be thought of as a relationship between two computers -- or in this case, two applications. Like all good relationships, the two partners communicate and can send or receive information from the other. Applications that support DDE may do so to different extents. For example, Microsoft Excel Versions 3.0 and 4.0 and Lotus 1-2-3/G or higher both fully support DDE server mode. The SAS System can read data and send data or system commands to these packages. However, with packages such as Microsoft Access (MS ACCESS) for Windows and Borland's Paradox Version 1.0, you can read data but not send data directly to SAS via DDE. In order for the SAS System to send and receive data from a DDE compliant application, both applications must be currently executing and the desired file(s) must be open in the server application. In SAS the FILENAME statement (with the DDE keyword) points to the DDE application we want to interact with.

The common programming languages are ASP.NET and VB.NET to get data from spreadsheet. It could create the web-based application to capture data from spreadsheet file and generate Excel report as they need.

2.3 MS. Excel Report

There are many ways to generate report and massage it, both manual and automated system, in specific requirement and form. For example, they can use existed report system to generate Report such as PDF Reporting software, Excel Report Builder 5.5, Excel Report and so on. On another hand, Reports, which are more related in financial data, Sale and Cost, and calculation, can be used MS office (MS Excel and MS Access), Lotus123 and others to calculate and summarize data in form that company need to make decision. Because current Lotus approach has limitation that why MS Excel is introduced. The system is developed by using MS VB 2005 programming to generate report in MS Excel format in order to provide the standard format of report and flexible usage. Report generators could able to update, modify or add some necessary information as they want in Excel Report. It is an easy and convenient tool for the creation and customization of reports which takes advantage of the formatting and presentation capabilities of Microsoft Excel 7.0, 8.0, 2000,

2003 (see more in Appendix). It saves cost of development and no need to use the third party tool such as Crystal Report but sometime the company will be unwilling to fork over the money to purchase a reporting tool. So what do you do then? Microsoft Excel is installed on most computers these days since it is part of Microsoft office, and can create detailed reports with Excel (With a market share estimated at anywhere from 60% to more than 90%, Excel is the clear leader in the spreadsheet market, by Beth Frances Cox, 2000).

Microsoft's spokeswoman says Excel stands out because of its easy connectivity to data, which lets users more readily analyze the information needed to make decisions. Excel also provides unprecedented tools that let users share, analyze, and collaborate on information over the Web, according to the spokeswoman. Excel 2000 can save to--and read from--HTML files. HTML is elevated to the same level as .xls, Excel's proprietary file format. It also supports dragging and dropping of table data from the browser directly into Excel.

As the system is developed by using MS Visual Basic 2005 programming, MS Excel is chosen to work effectively with MS VB. There are existed report systems developed by these technologies based on their specific requirements they received good results. For instance, ExcelEverywhere solves the problem using Microsoft Excel, and let ExcelEverywhere generate an ASP or ASP.NET-page. The ASP-page looks like and calculates like the spreadsheet. No programming required. Easy updating, fix spreadsheet and generate. Supports 190 Excel-functions. Code-behind module in C# and VB.NET for backend-integration. Use it for expense reports, surveys, order forms, financial advisor, ROI-calculator, engineering. No Excel needed on server! XLReportGen is a report generator for Microsoft Excel that outputs reports in Microsoft Excel. .NET, VB.NET and SQL get your project started rapidly. And some articles is about developing the system such Generating Microsoft Excel Reports in .NET, By Mark Bourisaw March (2004). Therefore, Report is generated in MS Excel format is flexible and convenient to the company to use, add, update and massage as they want.

CHAPTER 3

METHODOLOGY

3.1 Software Development Methodology

In many ways, building an information system or software is similar to building a house. The process or methodology of developing the system is just the way to transform into a simple drawing that is shown to the customer and refined until the customer agrees that the picture depicts what he or she wants. It is a approach to communicate between developer and his or her customer to understand the system requirements well.

The System Development Life Cycle is a standard approach for developer to follow in order to develop any software programs. There are four of fundamental phases: Planning, analysis, design and implementation. Different projects may use these phase in different process according to their methodology and special project required. In this project, Prototype methodology is applied to develop the effective and efficient software. A prototyping-based methodology performs the analysis, design, and implementation phases concurrently, and all three phases are performed repeatedly in a cycle until the system is completed.

Some companies view prototypes as a waste of time, but a working representation of a project's solution can save time and money and reduce problems at all stages of development. Prototypes help developers gather requirements and demonstrate architecture long before they've locked into actual application code. They also ease clients' anxieties about large projects, and they can help some clients make decisions and commitments. In addition, Framework prototyping uses mock-ups to represent functionality in the same technology as the project solution, with dialogue boxes and screens serving as holders for actual code. This kind of prototype is useful for nailing down requirements or getting a first look at an unfamiliar solution.

Reiterative prototype development methodology is another common approach. The prototype is refined through various stages until it eventually evolves into the desired product. This can be a failsafe method for rapid development, or it can provide planned stopping points in a staged rollout. Each prototype represents vastly different goals and requires varying degrees of investment, and each affects the development methodology in different ways. It's up to the architect, business driver, project manager, and sometimes even the client to decide if a prototype is necessary and what kind will be used.

With these methodologies, the basic of analysis and design are performed, and work immediately begins on a system prototype as shown in Figure 2.

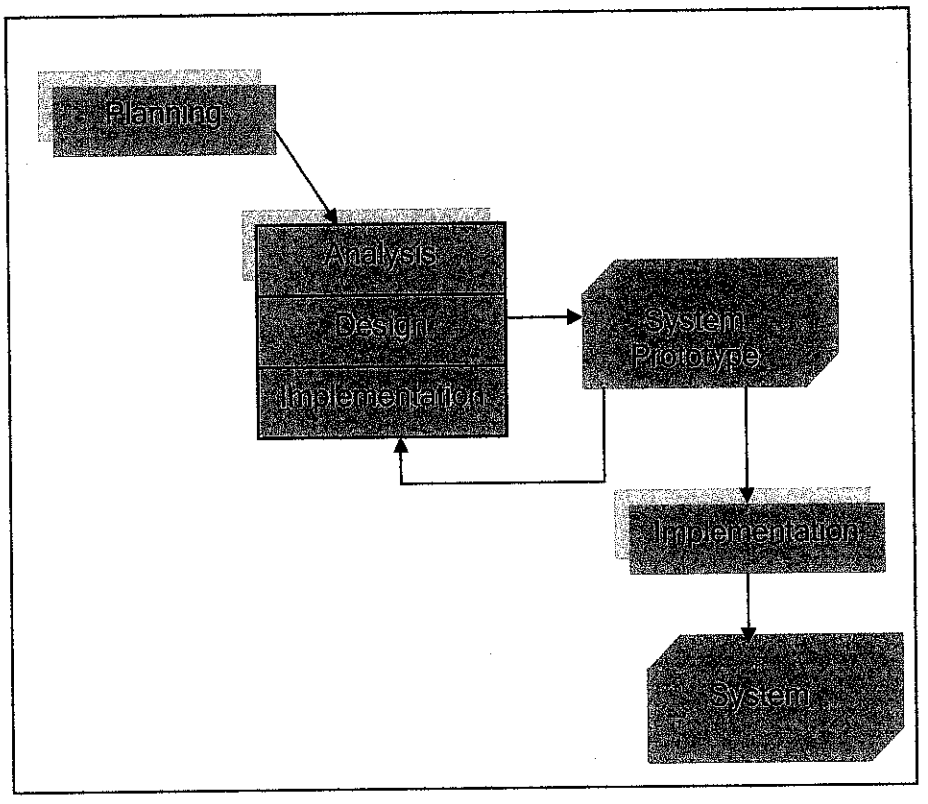


Figure 2: Prototype Methodology

The first prototype is usually the first part of the system that developer will use. This shown to the customers who provide the common, which are used to re-analyze, re-design, and re-implement a second prototype that provides a few more features. In SVC project, developed the first prototypes after we have done the planning and analysis of user's requirements and needs at MICTO Company, KLCC. This process continues in a cycle until the analysts, users, and sponsor agree that the prototype

provide enough functionality to be installed and used in the organization. After the prototype or system is installed, refinement occurs until it is accepted as the new system. Therefore we have done the prototype and presentation to the customer and improve it from day to day until the customer satisfied.

Planning

In this phase, it is the fundamental process of understanding why an information system should be built and determining how the project team will go about building it. It involves all feasibility study about the ability of building this project, what is business value of the project and how it will be used? Furthermore, as the developer, we have to identity the objective and scope of the project, work plan, staffs and how to control the project until the end. For instant, we have done problem identification, objectives and scope of study of this project.

Analysis

This phase will study who are the customer, who will use this system and what the system can do and when. Action should complete in this process is to gather the information about the system requirement and customer's needs in order to help us in defining the system functionality and satisfy the customer's requirement in the system. Analysis strategy should be applied to guide the project team how to identify the project and then ways to design a new system. In real work, Interviewing with real customer or user in MITCO have done to understand and identify the problems and reach the new proposed system which help them in solving all these problems.

Design

The design phase decides how the system will operate, in term of the hardware, software and network infrastructure such as the user interface, form and report, specific programming language, and database. From this point of view, prototype and solution can be introduced in way of developing the system including Architecture Design. SVC Report system will use Microsoft Visual Basic 2005 and MS Access to develop the system by generating the report into MS Excel.

Implementation

The final phase in SDLC is the implementation phase, during which the system is actually built. This is the phase that usually gets the most attention, because for the most system it is longest and most expensive. It involves in three main activities such system construction, installation and user support. This step will produce the real system to the customer. Analysis, Design and Implementation will run concurrently according to each prototype or subsystem until the customer satisfactions.

Prototyping is an exercise in risk management and development facilitation, and it minimizes the chances of producing the wrong functionality or an illogical design. Several factors must be considered when deciding if we should propose a prototype, and if so, which route to follow. Prototypes are excellent, hands-on planning tools that are useful to anyone involved in a development effort. Being able to visualize an eventual solution provides a template for development and an easily grasped overview for business drivers and other non-technical people involved in the project. A proof of concept can facilitate a project in everything from sales to quality assurance. If our solution warrants it, a prototype can help us design, develop, and deliver your product effectively and efficiently.

3.2 Software and Tools

.Net Framework

G. Guerro stated "The .NET Framework is a new development platform that provides consistent and efficient support to distributed enterprise applications over local area networks (LANs) and the Internet. (MSDN)" and it should be clustered of several technologies:

- ***The .NET languages*:** These include C# and VB .NET (Visual Basic .NET), the object oriented and modernized successor to Visual Basic 6.0; these languages also include JScript .NET (a server-side version of JavaScript), J# (a Java clone), and C++ with Managed Extensions.
- ***The CLR (Common Language Runtime)*:** The CLR is the system agent that runs and manages .NET code at runtime. Especially, the CLR is the engine that executes all .NET programs and provides automatic services for these

applications, such as security checking, memory management, and optimization, threading, error control, and type safety.

- ***The .NET Framework class library*:** The class library collects thousands of pieces of prebuilt functionality that you can "snap in" to your applications. These features are sometimes organized into technology sets, such as ADO.NET (the technology for creating database applications) and Windows Forms (the technology for creating desktop user interfaces). Furthermore, this class library is a comprehensive collection of object-oriented types that we can use to develop any application, service, or component. This class library supersedes the Microsoft Foundation Classes (MFC) commonly used in C++ development, and it is designed to be easily extensible to provide object-oriented programming support to other services, such as Microsoft Windows Server System products that currently provide proprietary-only Application Programming Interfaces (API).

- ***ASP.NET*:** This is the engine that hosts web applications and web services, with almost any feature from the .NET class library. ASP.NET also includes a set of web-specific services. (More details in ASP.Net 2.0 topic)

- ***Visual Studio*:** This optional development tool contains a rich set of productivity and debugging features. The Visual Studio setup CDs (or DVD) include the complete .NET Framework, so it won't need to download it separately. Microsoft Visual Web Developer Express 2005 Edition can also used for developing a small or medium web project because it is free.

Sometimes the division between these components isn't clear. For example, the term *ASP.NET* is sometimes used in a narrow sense to refer to the portion of the .NET class library used to design web pages. On the other hand, ASP.NET also refers to the whole topic of .NET web applications, which includes .NET languages and many fundamental pieces of the class library that aren't web-specific. Figure 3 will explain simpler about the Microsoft .Net technology.

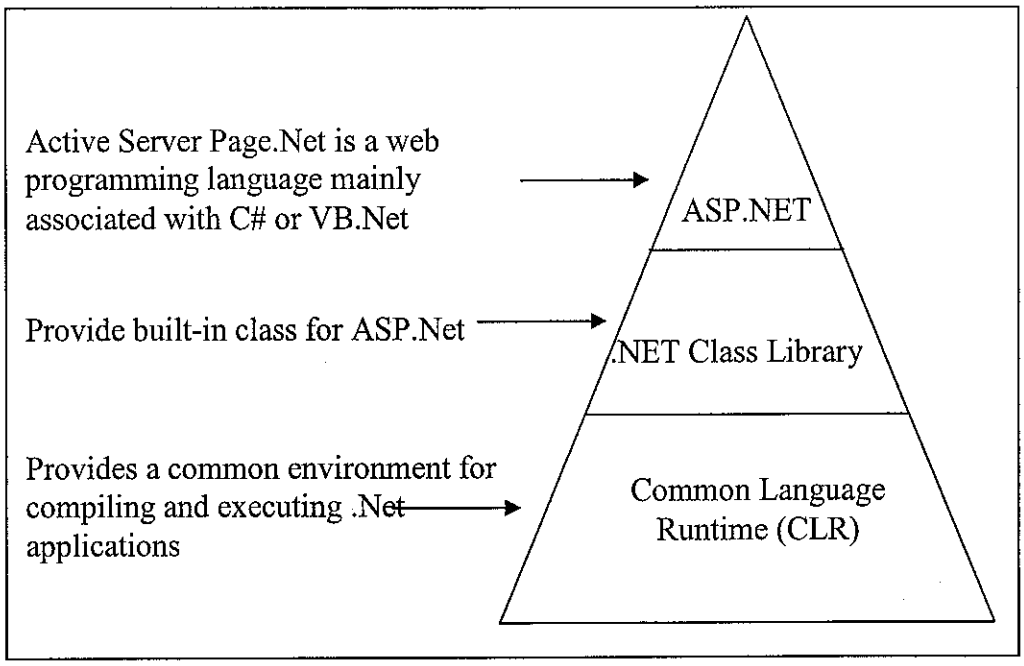


Figure 3: Microsoft .Net Overview

Key features of this new platform include the following:

- Provides a consistent language-independent, object-oriented development environment to leverage the developer's programming knowledge.
- Provides hassle-free software deployment, avoiding versioning problems with related components.
- Is a rich execution model, independent of storage location, where components can be stored and executed locally, or stored remotely and executed locally, or stored and executed remotely from an Internet location.
- Provides safe code execution, with superior security settings to match the security needs of today's organizations.
- Provides a consistent programming environment for both Windows and Web applications.
- Improves execution performance of Windows and Web applications by efficient code compilation in both environments.
- Is compliant with communication standards to ensure that .NET-connected applications can coexist and integrate with other applications and other platforms.

Microsoft Visual Basic 2005

When Visual Basic 1.0 was introduced in the early 1990s, it greatly simplified Windows application development. Visual Basic 2005 continues the tradition by providing a programmer-friendly environment in which you can write powerful desktop, web-based, and mobile applications quickly and easily.

Microsoft Visual Basic 2005 Express Edition

According to MSDN home, Visual Basic 2005 Express provides a powerful WYSIWYG visual design surface to quickly and easily create interactive Windows applications. Microsoft also released new learning content on the MSDN Coding4Fun Web site, providing tools and resources for the community to get up and running quickly. The Visual Studio 2005 Express editions now join the SQL Server 2005 Express editions as a no-cost offering within the Microsoft application platform.

Furthermore, "Software has the potential to transform everyday lives. By making the Visual Studio 2005 Express editions available free of charge, we're putting the power of programming into the hands of an exploding community of recreational programmers," stated by S. Somasegar, corporate vice president of the Developer Division at Microsoft. "This community has asked for it, and we are excited to provide it."

Visual Basic 2005 Express has tools to help developers design an interface for their application and even show them where they need to add VB code.

Microsoft Access 2003

According to Wkipedia, **Microsoft Access** is a relational database management system from Microsoft which combines the relational Microsoft Jet Database Engine with a graphical user interface. Access can use data stored in Access/Jet, Microsoft SQL Server, Oracle, or any ODBC-compliant data container. Skilled software developers and data architects use it to develop application software. In Addition, relatively unskilled programmers and non-programmer "power users" can use it to build simple applications. It supports some object-oriented (OO) techniques but falls short of being a fully OO development tool.

3.3 Hardware Specification

Apart from software, hardware also plays an important role for the development process. A complete development platform with specifications as below:

- Dell Laptop
- Pentium IV 2.8GHz
- 512MB DDRAM
- 40GB HDD
- 10/100 Internal LAN Card

3.4 Project Schedule

Project Name: SVC Report Generation System (PART I)

ACTIVITIES	DATE	WEEK NO													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
Propose IT Project	29th Jan 07														
Proposal Approved Topic by Research Cluster	09th Feb 07														
Initial study of the project and submit Preliminary Report	12th Feb 07														
Study and analyze the user requirements	16th Feb 07														
Study the tools and software are required to develop the this project	23rd Feb 07														
Seminar 1	26th Feb 07														
Start developing first prototype of the system	2nd Mar 07														
Submit Progress Report	19th Mar 07														
Continue developing prototype	26th Mar 07														
Seminar 2	02nd Apr 07														
Continue developing prototype	09th Apr 07														
Submit Interim Report	18th Apr 07														
Oral Presentation and deliver first prototype	30th Apr 07														

Project Name: SVC Report Generation System (PART II)

ACTIVITIES	DATE	WEEK NO													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
Project Work Continue		■	■	■											
Submission of Progress Report I	08th Aug 07				■										
Project Work Continue					■	■	■	■	■						
Submission of Progress Report II	19th Sept 07									■					
Seminar	24th -28th Sept 07										■				
Project work continue										■	■				
Poster Exhibition	03rd Oct 07											■			
Submission of Dissertation (soft bound)	05th Oct 07												■		
Oral Presentation	22nd-26th Oct 07													■	
Submission of Project Dissertation (Hard bound)	2nd Nov														■

■ Process

CHAPTER 4

RESULTS AND DISCUSIONS

4.1 System Functionalities

Based on our first visit and interview with Mr. Armayya Fitri Yusop, Executive of Financial and Accounting and Tax in MITCO, User requirements are studied more detail in order to identify functionalities of SVC Report Generation System. As the result, functionalities are defined as the following:

- Allow user to upload data from MS. Excel to MS Access
- User can generate SVC report easily from MS Access
- Allow user to check and update data before generating the final report
- Generate the SVC report in standard or general format
- Generate or update data based on specific criteria such as date, product code, country and department
- Allow manager to create user accounts and manage the users or staffs who will work with this system

4.1.1 Proposed solution

In order to answer these requirements and develop the system with proper functionalities, the solution of developing the system is proposed. This solution will describe how to the system work and main steps of designing and developing the whole system in the specific criteria and times. Figure 4: shown the SVC Report Generation solution.

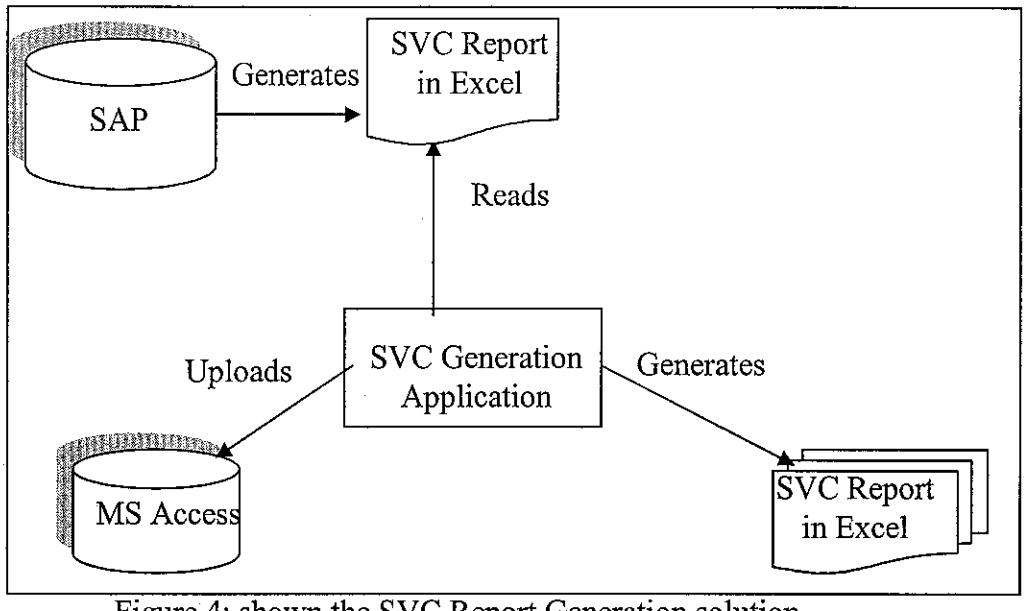


Figure 4: shown the SVC Report Generation solution

Based on the diagram, this solution is divided into three main activities or steps:

1) **Generating SVC Report to Excel Report File**

Firstly, the user has to generate or create SVC report from SAP database into MS Excel report that it will be used later to upload data into the MS Access. So this data is main or original source from SAP database.

2) **Uploading the data from SVC report in MS Excel file into MS Access**

After generating SVC report in MS Excel, the users can the system to upload all the data from MS Excel to MS Access in correspondence. It helps users a lot reduce time to input data from one source to others. Especially, it is one of main functionalities of SVC Report Generation System.

3) **Manipulating and/or generating SVC report in MS Excel by using data from MS Access**

Lastly, users could manipulate data by updating, adding or deleting data based on the real requirements they want. User interfaces are provided to help users easy to manipulate the data and generate the report into MS Excel with standard format. This is also other one of main functionality in the system.

Therefore, there are three steps we have to do in order to develop this system effectively. Firstly, Database Schema and Design is core key that we need to develop first in order to be easy to support other two main functions of system

such as Uploading data into MS Access and Generating SVC Report into MS Excel. After the Database of SVC Report developed, these two functions need to use this to manipulate to design and develop the real system. Base on this, developer is also easy to decide to develop which function first.

Developing SVC report manipulation and generation is the next step of this system. This function can be designed and developed by using the sample data or database in MS Access. It also helps our development to deliver the system fast based on the function by function. The last steps of developing system is developing SVC report uploading function from MS Excel to MS Access which is this function could save customer a lot of time to transfer data from one source to others..

4.1.2 Stakeholders

In order to understand and identify the user requirements well, stakeholders have to define to capture and satisfy their requirements. It could also improve communication and involvement of them in the system in specific time and step. In the first phase, three types of stakeholders or users are identified as the following:

1. **Supervisor:** Mrs. Aliza Bt Sarlan is my FYP supervisor who gives advice and guide the whole project development and follow up the process or progress of developer (student). She could help student to analyze the user and system requirements, especially, to keep communication between user and developer.
2. **Expert Users:** in order to develop this project smoothly, Expert Users are required to evaluate and give comments to the developer about the system before delivering to the customer. They could find out more weakness and improvement points of the system to make sure the system delivered is effect and efficient. It saves time and cost because developer doesn't need to meet the customer directly and often. Expert Users is one is Mrs. Aliza Bt. Sarlan, UTP Lecturer and FYP supervisor, and two internal Lecturers of UTP.
3. **Real Customer or User:** Financial Dept of MITCO is user of this system that they need to identify their requirement with UTP supervisor, Export Users, or/and developer.

4.1.3 Use Case Diagram of SVC Report Generation System

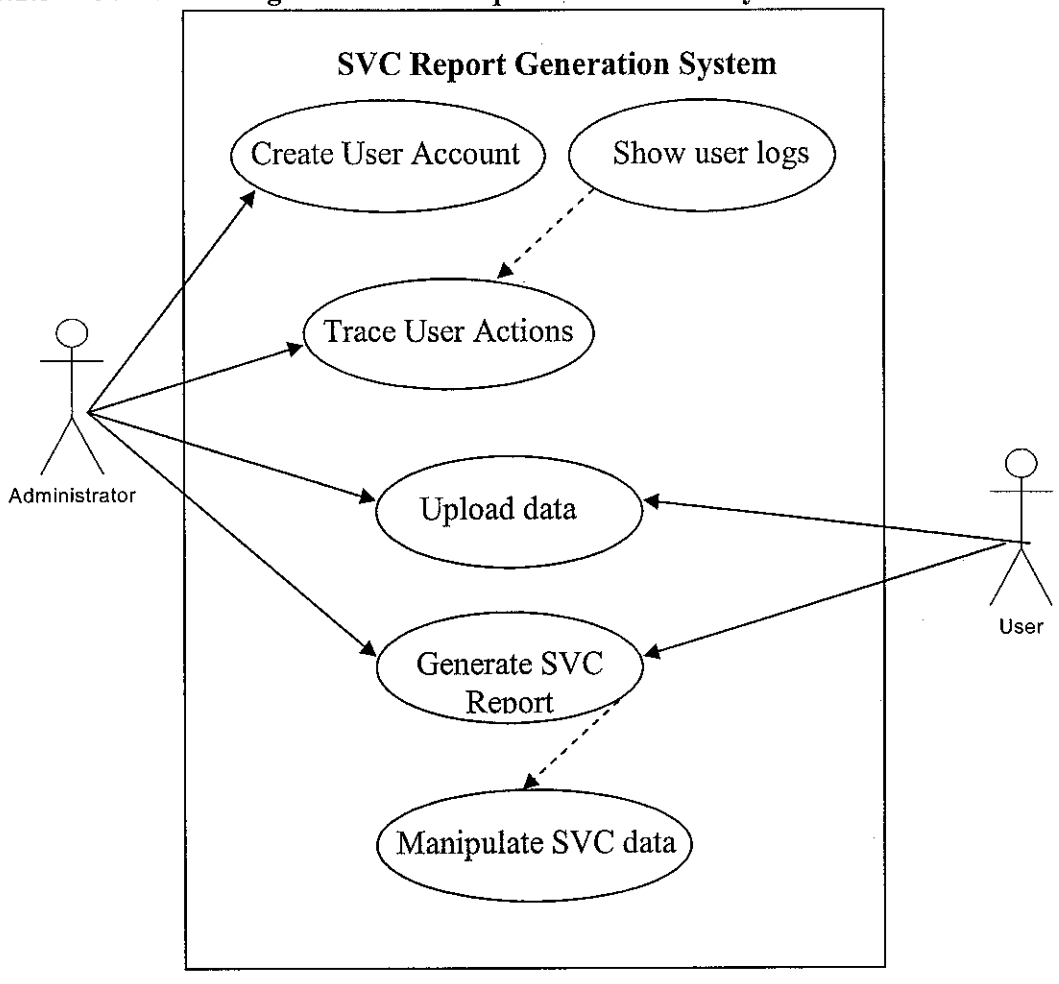


Figure 5: SVC Report Generation System Use Case Diagram

4.1.4 SVC Report Generation System Architecture

Based on the solution and functionalities of system stated above, the system architecture of SVC Report Generation Application could be designed as below:

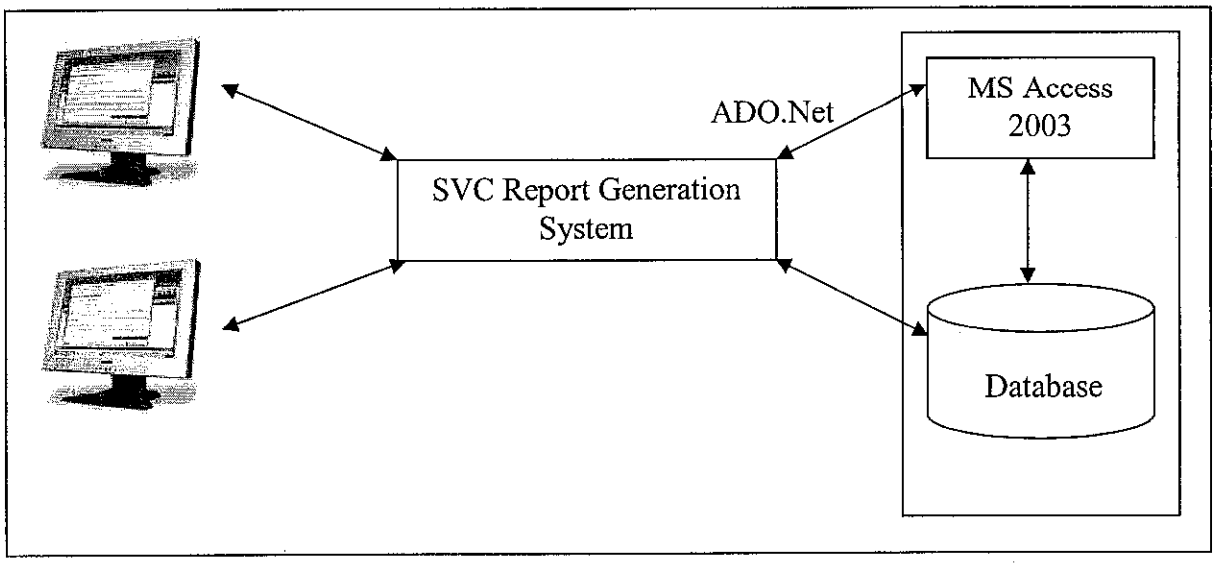


Figure 6: SVC Report Generation Architecture

This system could allow many users to use the system from different location in local network. The SVC Report Generation Application is user friendly interface which users could log in and use to upload data and generate report as they want. Referring to this architecture, MS Access (DBMS and Database) is stored in centre so all data uploaded and manipulated are stored in a single place. This could avoid data duplication and making works complicated.

4.2 Database Design Schema

The first design of SVC Report Generation database, there are five entities as the following:

Table 1: User Account Entity

Entity Name	Attribute	Data Type	Description
User Account	User ID	Number	Describe amount and unique key of individual account
	User Name	Character	Describe the name of account to login the system
	User Password	Character	Describe password of each account
	User Confirmed Password	Character	Confirm the password of user
	U Name	Character	Describe the full name of user
	Gender	Character	Identify gender of each user
	Date of Birth	Date	Keep the date of birth info
	Privilege	Character	Describes the user right of each account
	U Position	Character	Describe the position of user
	Division	Character	Describe the department of user work in
	Telephone	Character	Telephone number of user
	Ext	Character	Extension of telephone number
	Mobile	Character	Hand phone number
	Email	Character	Email address
	Address	Character	Present address of user
	Created Date	Date	Record the date that user created

Table 2: Product Information Entity

Entity Name	Attribute	Data Type	Description
Product Information	Product ID	Number	Identify the amount and unique number of product codes
	Product Code	Character	Describe the code number
	Product Name	Character	Describe the name of each particular product code
	Product Group	Character	Describe the group which each product belongs to
	Department	Character	Describe the department of each product

Table 3: SalesVsCost Entity

Entity Name	Attribute	Data Type	Description
Sales and Cost	Product Code	Character	Identify the product code of each product
	Invoice number	Character	A unique number of each product sold
	Quantity	Number	Number of product sold
	Invoice Date	Date	Date of invoice
	Allocation	Character	Identify the location of the product
	Customer	Character	Name of customer or company
	Country	Character	Name of country that product exported
	Discharge Port	Character	
	Vessel name	Character	Name of vessel or container for liquids
	Spot/Term Sales	Character	Term of Sales
	Price Term	Character	Price Term
	Sales value (MYR)	Number	Amount of sales of each product charge in Malaysian Riggit
	Exchange Rate	Number	Price of exchanging rate to US dollar
	Sales Value(USD)	Number	Amount of sales of each product in US dollar
	Cost Invoice number	Character	A unique of number product cost
	Cost Quantity	Number	Number of product cost
	Cost Invoice Date	Date	Date of invoice
	Cost Allocation	String	Identify the location of product in cost part
	Supplier	Character	Name of supplier or company that sale product to MITCO
	Load Port	Character	
	Cost value (MYR)	Number	Amount of cost of each product charge in Malaysian Riggit
	Exchange Rate	Number	Price of exchanging rate to US dollar
	Sales Value(USD)	Number	Amount of sales of each product in US dollar
	Cost Packing and Handling	Number	Cost of packing and handling of product
	Cost Freight	Number	Freight cost
	Cost Trucking	Number	Truck cost
	Cost Port and Demurrage	Number	Cost of Port and demurrage
	Cost Agency	Number	Cost of agency expense
	Cost Insurance	Number	Cost of insurance
	Financial Cost	Number	Expense on finance
	Other Charges	Number	Eg. Insurance, financing, agency fee
	Service Fee	Number	Eg. Marketing fee
	Price Term	Character	Term of price

Table 4: Log Generation Entity

Entity Name	Attribute	Data Type	Description
Log Generation	User Name	Character	Describe who did the report generation
	Type of Report	Character	What kind of report they generated
	Created Date	Date	When the user generated the report

Table 5: Log Upload Data Entity

Entity Name	Attribute	Data Type	Description
Log Upload Data	User Name	Character	Describe who uploaded data
	File name	Character	Describe name of file uploaded
	Created Date	Date	When the user uploaded the data

Based on the entities above, the entity relationship diagram is created to design the real schema of SVC Report Generation application.

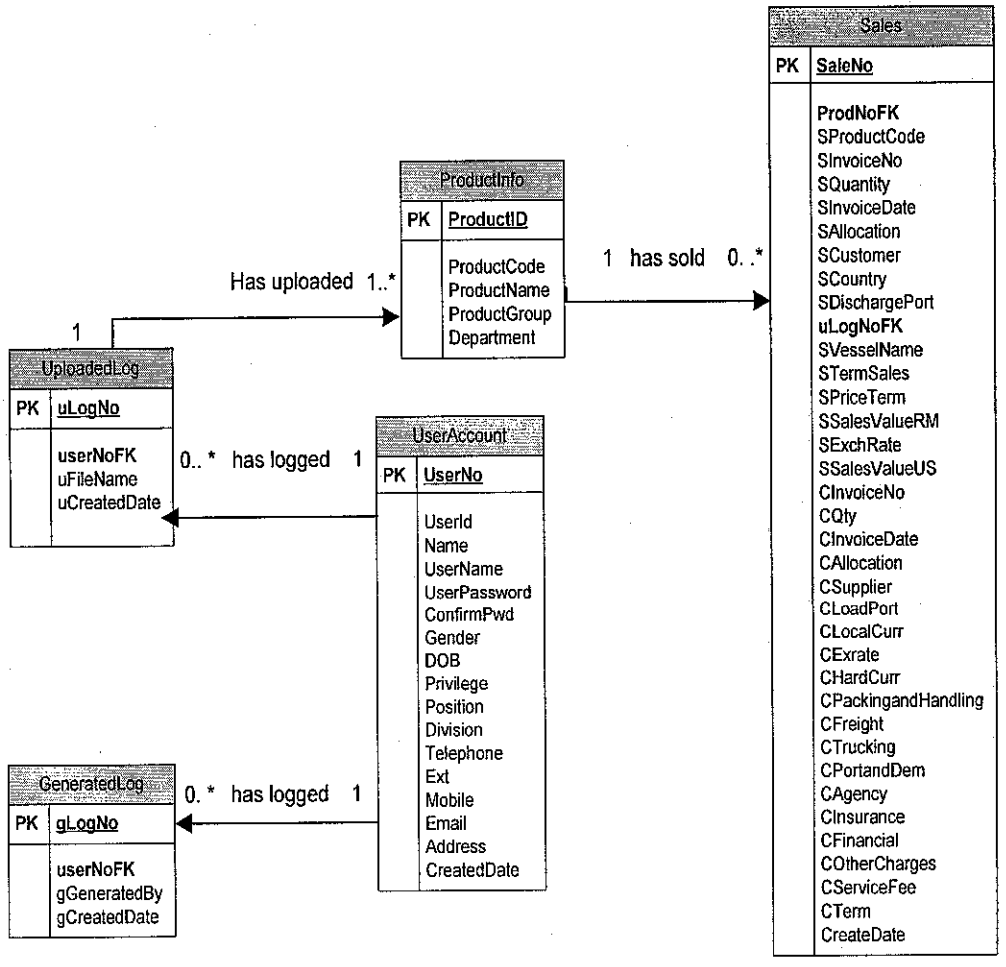


Figure 7: SVC Report Generation Entity Relationship Diagram

Table 6: Tables Description

No	Table Name	Description
1	tblSales	Stores all information related to sale and cost data. It has relationship with Cost, ProductInfo and UploadedLog tables.
2	tblProductCodes	Keeps information of product elements such name, code…etc.
3	tblUserAccount	Stores user account data created and given the privilege to access the system
4	tblUploadedLog	Keep and trace information of files uploaded by specific user in special time
5	tblGeneratedLog	Keeps and traces information of SVC report generated by user.

4.3 Sample Prototypes of SVC System

This topic will show some sample of SVC Report application prototypes that deliver to the customers to see and evaluate fast. From these prototypes, the system will found out more improvement to satisfy the customers.

Below is forms navigation of the whole system which is easy for users and developers to understand and develop the system. More communication between customer and developer will help the system to develop in right way with right services. It will be shown in Figure 9. There are some prototypes of system will be shown such as Login form, Functions form and Upload form. That allows the users to evaluate base on the color, font, and form design and so on. Task Object Event (TOE) Chart is used to help developer to know the functionalities and components in each form. TOE Chart is a table which describes the Task and what Objects should be created to achieve the task, and Event of each object. It is highly recommended to use a system using Visual Basic programming (Diane, Microsoft Visual Basic 2005: Reloaded, second edition, 2007)

a) Login form is used to authorize the user before login into the system so user (Administrator) has to create the user account to all users to use this system base on privilege such as Administrator and Normal user. Administrator has more right which could create user account for each user and normal user could not, he or she just can upload and generate the report.

Table 7: Log in Form (frmUserLogin) TOE Chart

Task	Object	Event
Get username and password of each user who wants to login into the system	**lblUserName** **lblPassword** **txtUserName** **txtPassword**	None
Enter the SVC system	**btnLogin**	Click
Clear the username and password data in the textboxes	**btnClear**	Click
Provide help or user manual for user	**btnHelp**	Click

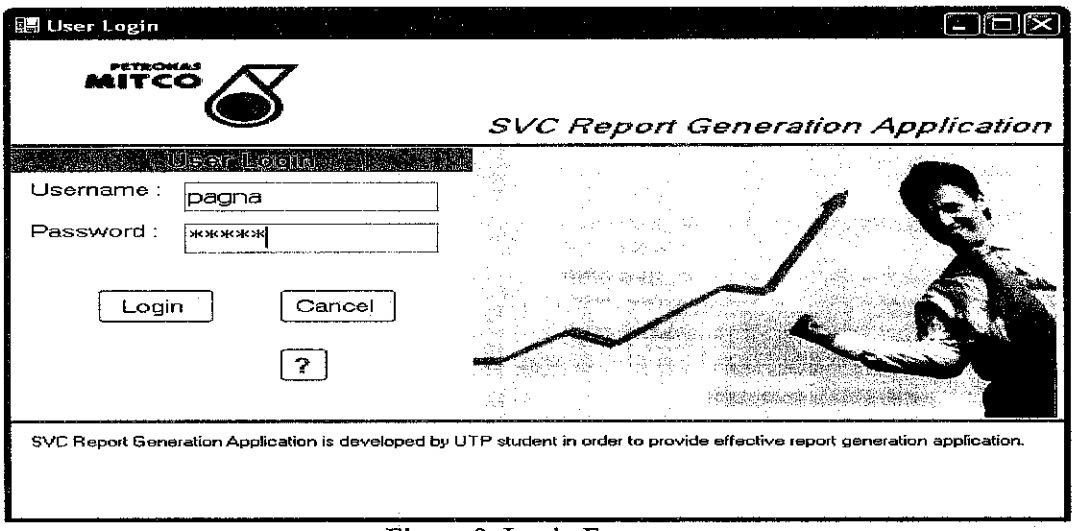


Figure 8: Login Form

b) Main Functions form is form which contains all navigation buttons to link to the main functions of system such User Management, Upload Data and Generate Report.

Table 8: SVC Function Form (frmSVCfunction) TOE Chart

Task	Object	Event
Open the Generation SVC form	**btnGenerateSVCReport**	Click
Open Upload SVC data from	**btnUploadData**	Click
Open User Management	**btnUserManagement**	Click
Exit the system	**btnExit**	Click
Display the current username and privilege of user	**lbluserdesc**	None
Open Upload SVC data from	**mnUploadData**	Click
Open User Management	**mnUserManagement**	Click
Exit the system	**mnExit**	Click
Logoff	**mnLogOff**	Click
Open the tutorial file	**mnTutorials**	Click
Open the about us form	**mnAboutUs**	Click

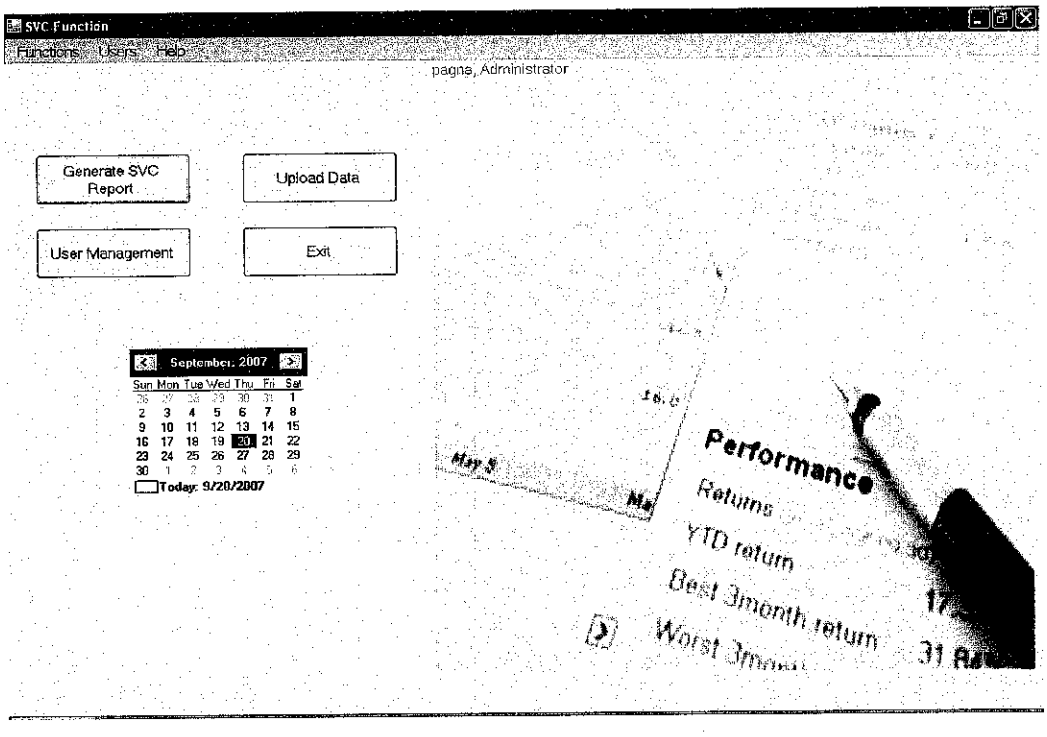


Figure 9: Functions Form

c) Upload Data Form allows user to upload data from MS. Excel to MS. Excel easily by just clicking browse the Excel file and then click on Proceed.

Table 9: Upload data Form (frmUploadData) TOE Chart

Task	Object	Event
Read SVC data in MS. Excel in order to put database	**btnProceed**	Click
Browse or open the SVC Excel file	**btnBrowse**	Click
Display the title of upload form	**lblUploadData**	None
Show the progress or process of data read into Database	**pgbUploadData**	None
Display the file path of SVC data	**lblChooseFile** **txtFilePath**	None
Allow user to determine the monthly report	**cboMonthly** **lblmonthly** **txtyear** **lblyear**	None
Display the current username and privilege of user	**lbluserdesc**	None
To open Upload SVC data from	**mnUploadData**	Click
To open User Management	**mnUserManagement**	Click
To exit the system	**mnExit**	Click
To Logoff	**mnLogOff**	Click
To open the tutorial file	**mnTutorials**	Click
To open the about us form	**mnAboutUs**	Click

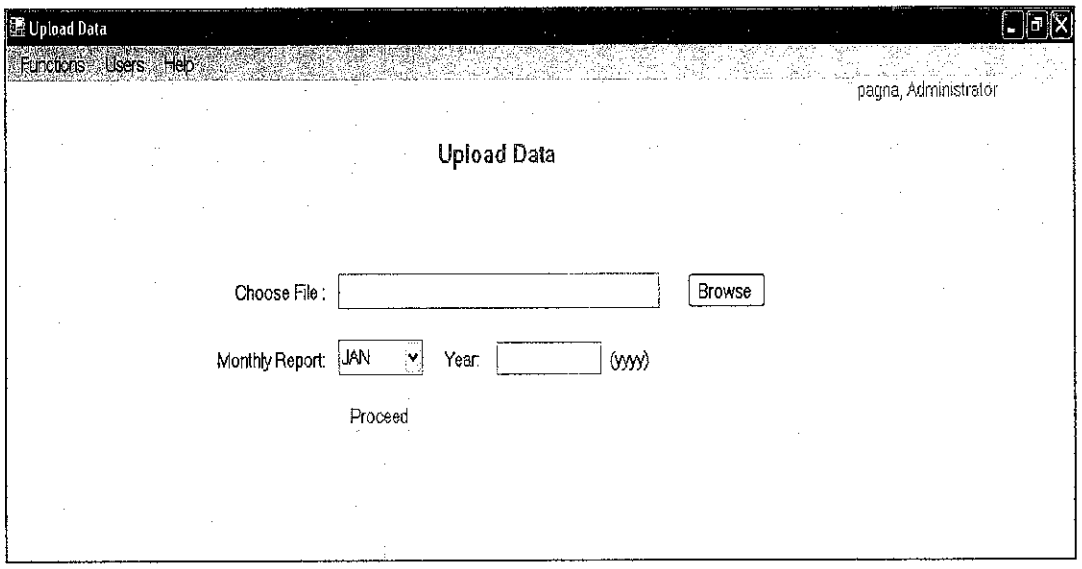


Figure 10: Upload Data Form

d) UserManagement Form is used to retrieve user account data and do manipulate data such as update, delete and show logs file.

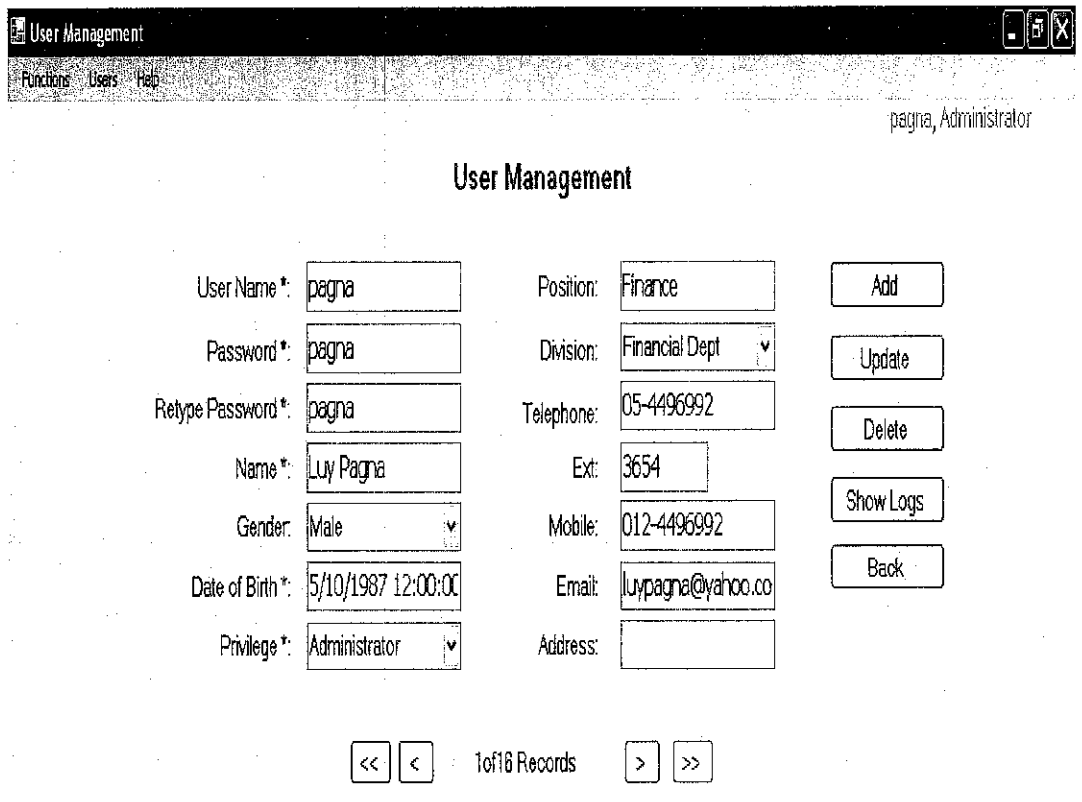


Figure 11: User Management

Table 10: User Management Form (frmUserManagement) TOE Chart

Task	Object	Event
Link or open the Add User Account form	**btnAddUser**	Click
Update the User account data in the current display	**btnUpdateUser**	Click
Delete the user account data in the current display	**btnDeleteUser**	Click
Display the show logs option dialog form	**btnShowLogs**	Click
Go back to SVC Function form	**btnBackUserMgt**	Click
Move or navigate the user records	**btnNavFirst, btnNavPrevious btnNavNext, btnLast**	Click
Display each user account from database	**lblUserName,txtUserName lblPassword,txtPassword, lblRetypePassword, txtConfirmPassword, lblFullName,txtName lblGender,cboGender lblDateofBirth,txtDateofBirth lblPrivilege,cboPrivilege lblPosition,txtPosition lblDivision,cboDivision lblTelephone,txtTelephone lblExtension,txtExt lblMobile,txtMobile lblEmail,txtEmail lblAddress,txtAddress**	None
Display the current username and privilege of user	**Lbluserdesc**	None
Display the title of User Management form	**lblUserManagement**	None
To open Upload SVC data from	**mnUploadData**	Click
To open User Management	**mnUserManagement**	Click
To exit the system	**mnExit**	Click
To Logoff	**mnLogOff**	Click
To open the tutorial file	**mnTutorials**	Click
To open the about us form	**mnAboutUs**	Click

e) AddUserAccount Form is used to add user account when user clicked Add button from UserManagement Form.

Table 11: Add User Account Form (frmAddUser) TOE Chart

Task	Object	Event
Save or add user account data into database	**btnSaveUser**	Click
Clear or reset data have input in the textboxes and combo boxes	**btnAddClear**	Click
Go back user management form	**btnBackToUserMgt**	Click
Get user account data from user input	**lblUserName, txtUserName, lblPassword, txtPassword, lblRetypePassword, txtConfirmPassword, lblFullName, txtName, lblGender, cboGender, lblDateofBirth, txtDateofBirth, lblPrivilege, cboPrivilege, lblPosition, txtPosition, lblDivision, cboDivision, lblTelephone, txtTelephone, lblExtension, txtExt, lblMobile, txtMobile, lblEmail, txtEmail, lblAddress, txtAddress,**	None
Display the current username and privilege of user	**Lbluserdesc**	None
Display the title of User Management form	**lblAddUserTitle**	None
To open Upload SVC data from	**mnUploadData**	Click
To open User Management	**mnUserManagement**	Click
To exit the system	**mnExit**	Click
To Logoff	**mnLogOff**	Click
To open the tutorial file	**mnTutorials**	Click
To open the about us form	**mnAboutUs**	Click

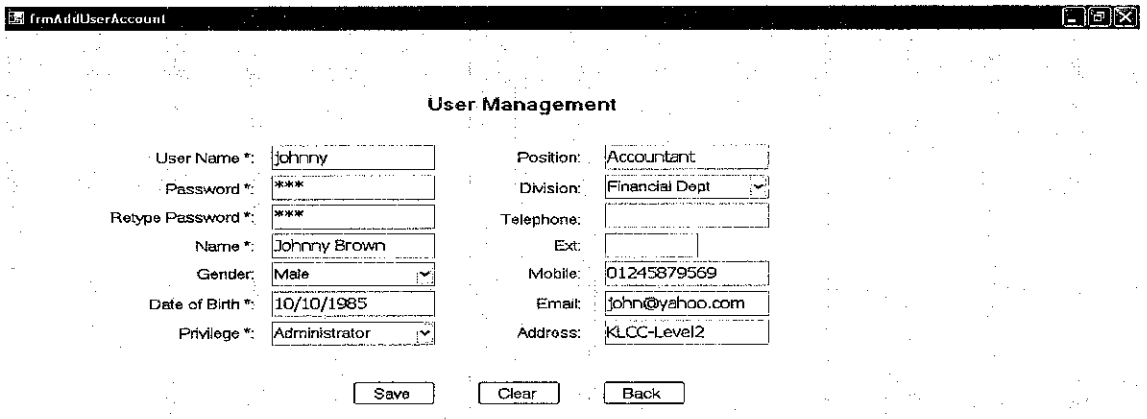


Figure 12: Add User Account

f) ShowLogs Form is used to show historical activities of users when they uploaded data or/and generate reports in order to track the user's perform.

Table 12: Show Logs Form (frmShowLogs) TOE Chart

Task	Object	Event
Display the upload log or generation log base on the user option	**btnOk**	Click
Cancel this action and go back to User Management form	**btnCancel**	Click
Give the user the option to display the log form	**rdoUploadlog** **rdoGenerateLog**	Click

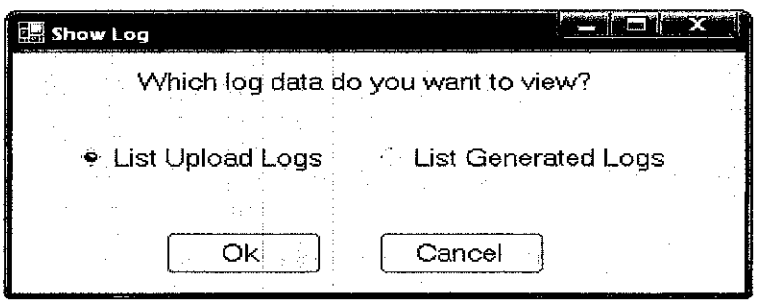


Figure 13: Show Logs

g) List Upload Logs Form is used to show data recorded activities of user when user have uploaded data in any particular time and file.

Table 13: List Upload Logs Form (frmUploadLog) TOE Chart

Task	Object	Event
Go back to UserManagement form	**btnBack**	Click
Go the SVC Function form	**btnMainMenu**	Click
Show the title of form and display the data of users who have uploaded data	**lblUploadlogs** **dgvUploadLogs**	None
Display the current username and privilege of user	**lbluserdesc**	None
To open User Management	**mnUserManagement**	Click
To exit the system	**mnExit**	Click
To Logoff	**mnLogOff**	Click
To open the tutorial file	**mnTutorials**	Click
To open the about us form	**mnAboutUs**	Click

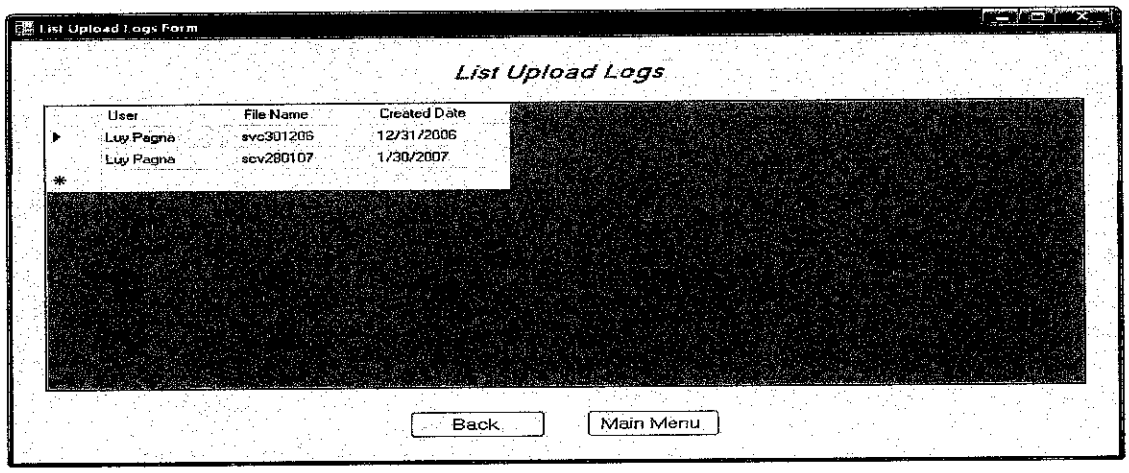


Figure 14: List Uploaded Log

i) List Generation Logs Form is used to show data recorded activities of users when they have generate reports in order to track the user's perform.

Table 14: List Generation Logs Form (frmGenerationLogs) TOE Chart

Task	Object	Event
Go back to UserManagement form	**btnBack**	Click
Go the SVC Function form	**btnMainMenu**	Click
Show the title of form and display the data of users who have uploaded data	**lblGenerationLogs** **dgvGenerationLog**	None
Display the current username and privilege of user	**lbluserdesc**	None
To open User Management	**mnUserManagement**	Click
To exit the system	**mnExit**	Click
To Logoff	**mnLogOff**	Click
To open the tutorial file	**mnTutorials**	Click
To open the about us form	**mnAboutUs**	Click

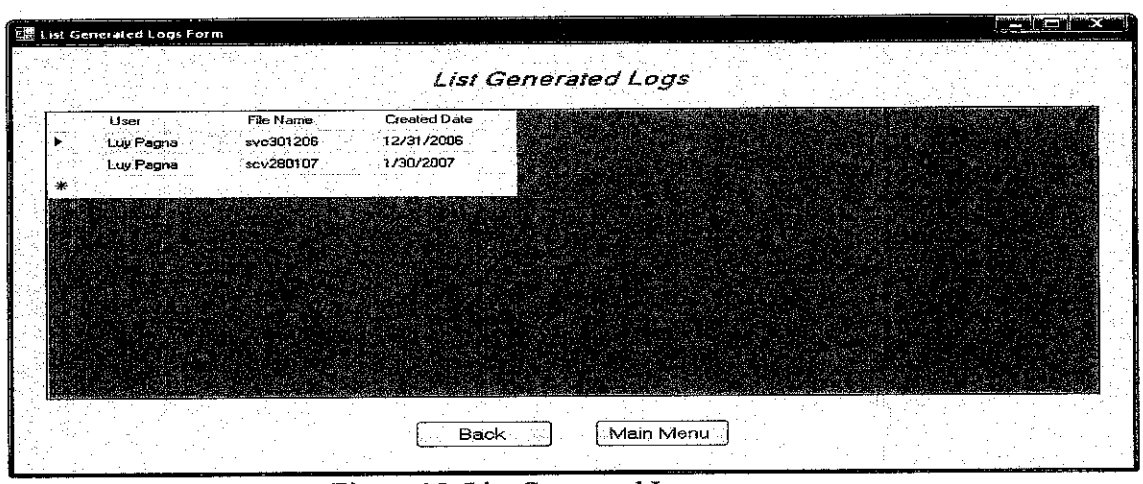


Figure 15: List Generated Logs

j) GenerateReport Form is used to manipulate the data of SVC which have been uploaded to database. Users can edit or add data if it is necessary. In addition, this form can allow the users to generate the report into MS Excel.

Table 15: Generate SVC Report Form (frmGeneratedSVCReport) TOE Chart

Task	Object	Event
Generate SVC Excel report	**btnGenerateSVCReport**	Click
Give the user to make a report by Month, Year, Department, Product Name, Product Code, Allocation, Customer	**cboReportbyDate** **lblReportbyDate** **cboSelectReport** **lblSelectReport** **cboReportBy** **lblReportBy**	Click
Display the title of form	lblGenerateSVCReport	None
Display the current username and privilege of user	**lbluserdesc**	None
To open User Management	**mnUserManagement**	Click
To exit the system	**mnExit**	Click
To Logoff	**mnLogOff**	Click
To open the tutorial file	**mnTutorials**	Click
To open the about us form	**mnAboutUs**	Click

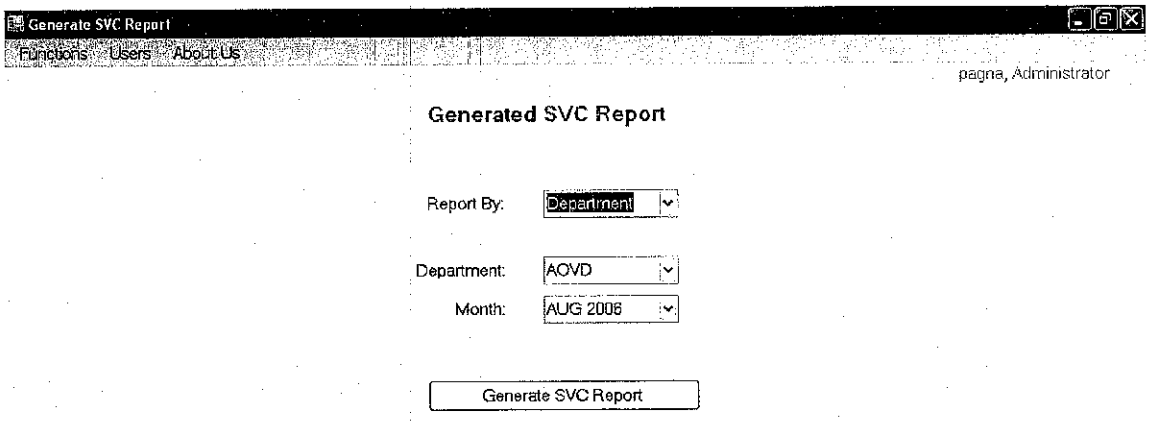


Figure 16: Generated SVC Report Form

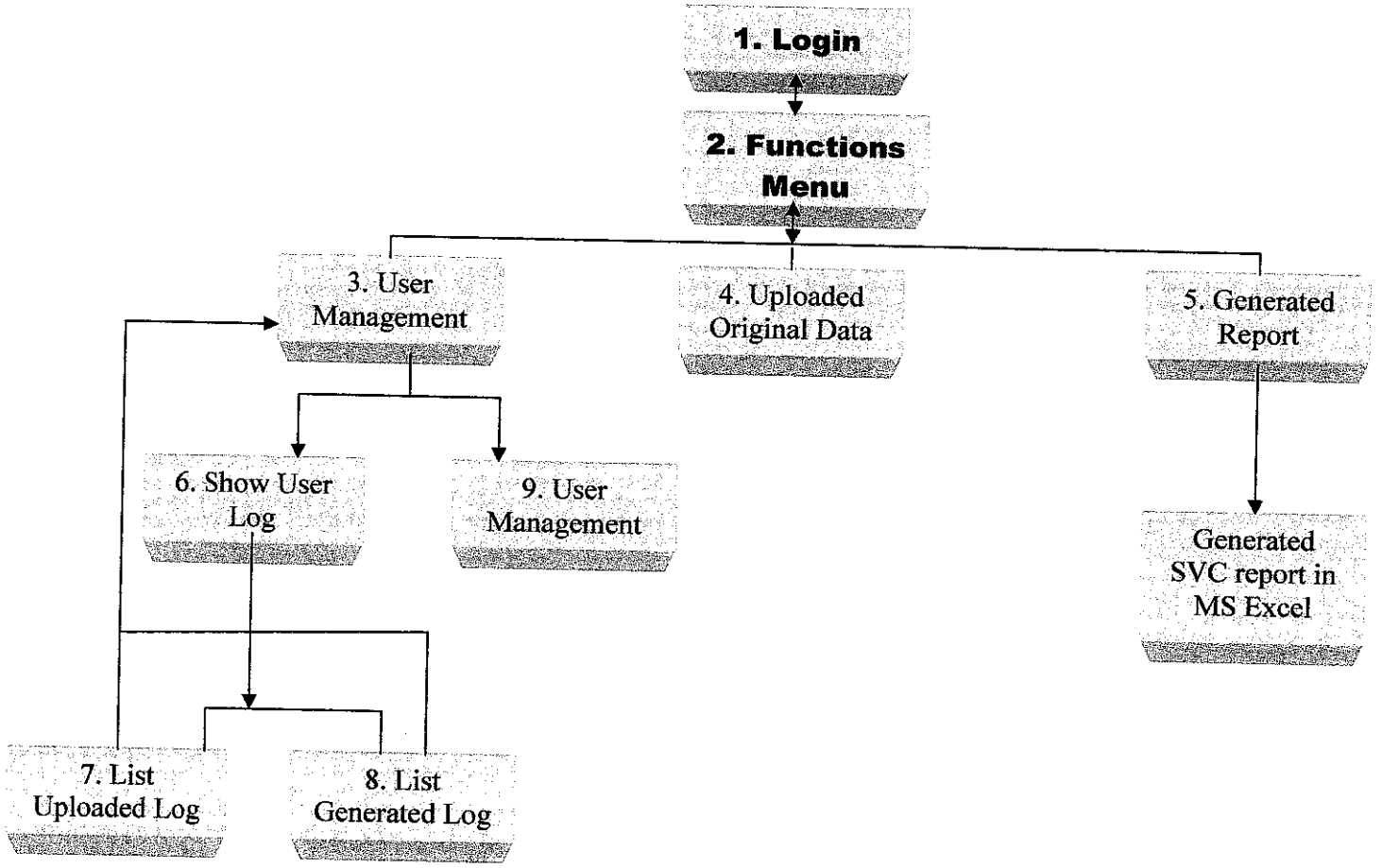


Figure 17: SVC Report Generation Form Navigation

4.5 Implementing the System

In the step, it will describe how the system work and introduce the user to system by input the real data as the following:

i. Run the application
ii. Type user name which have been created in order to login into the system.
For example: user name: pagna and password: pagna

** Normally, the user account have created by Administrator*

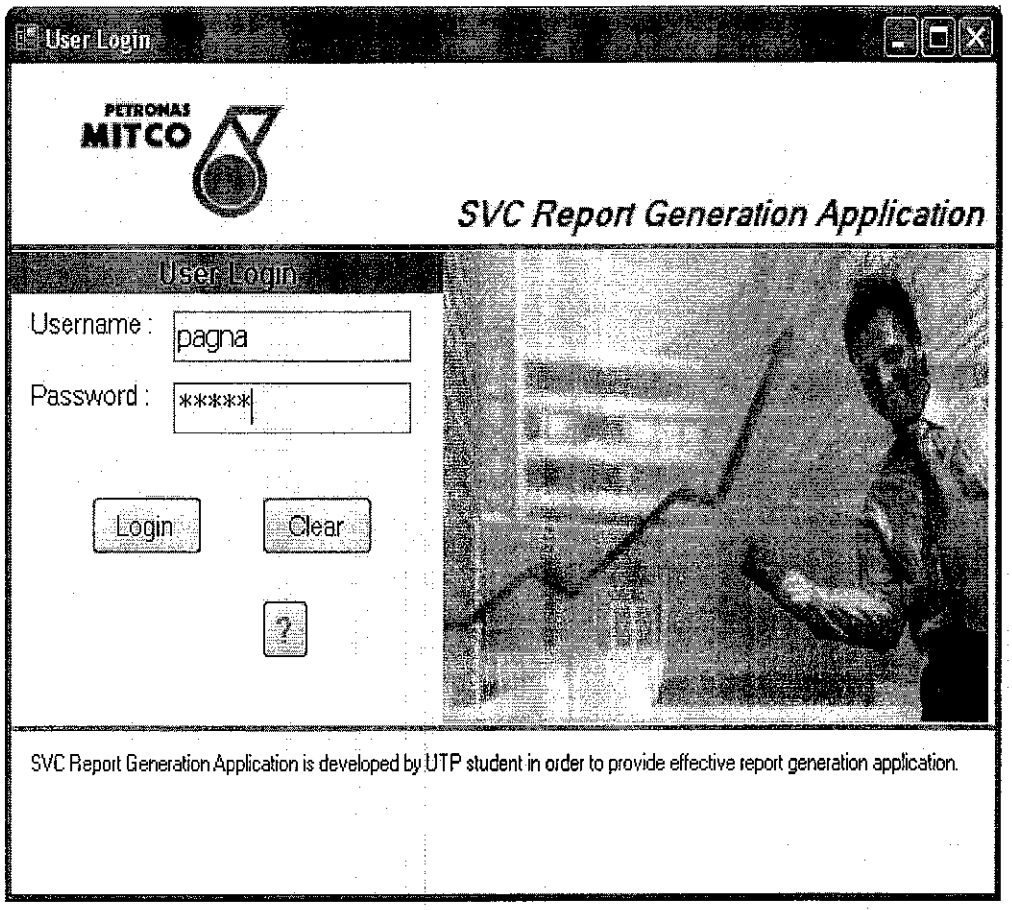


Figure 18: Logging the System

iii. Click Login button
iv. Choose any functions you want to perform by clicking on that Button (See Figure 9: Functions Form)

** If users login as Normal User, they would not perform User Management functions. This function will disable.*

A. <u>To Manage User Account</u>

1. Click on User Management button (See Figure 19)
2. Click on Navigation button below to move the record
3. Click on Add to User Account
4. Fill the data of new user

** All fields with (*) are mandatory field*

5. Click on Save
6. Click Back to go to User Management Main form

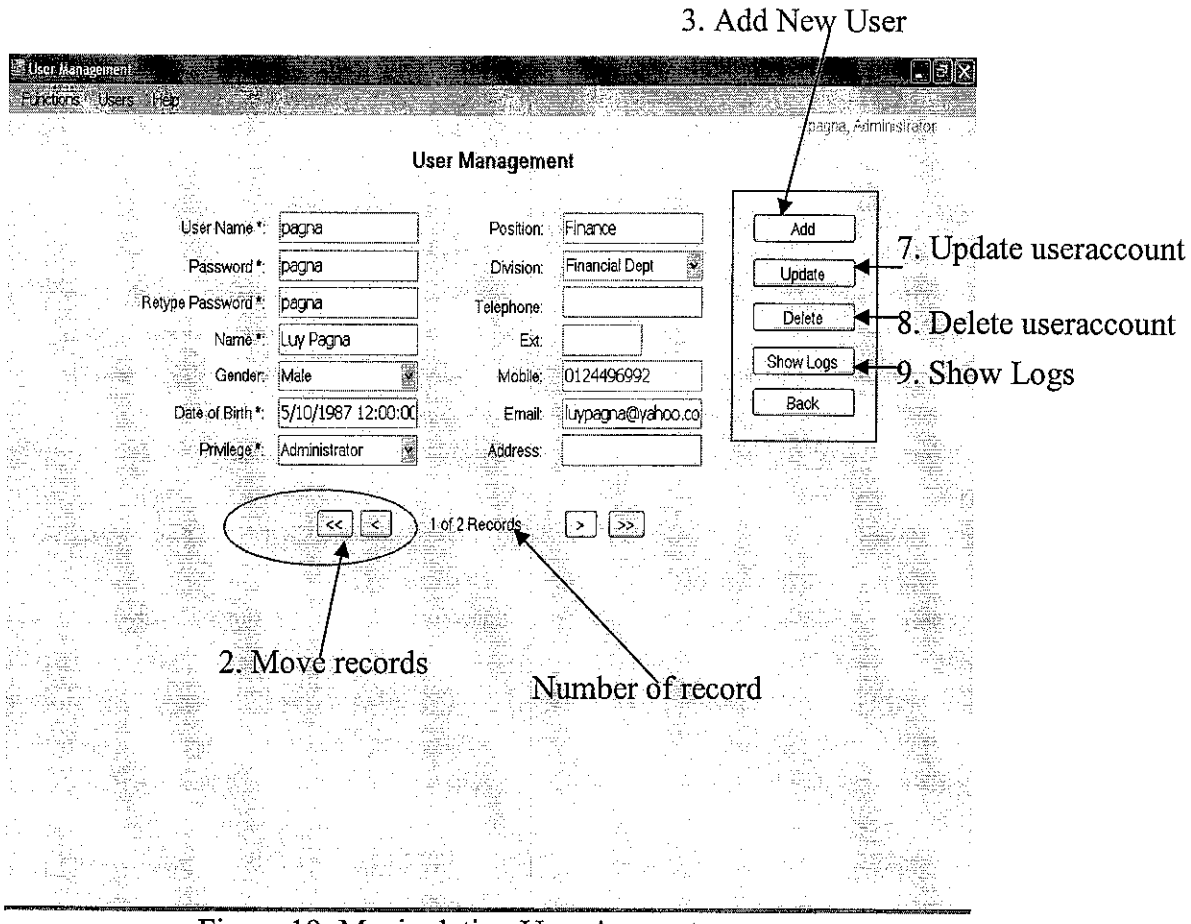


Figure 19: Manipulating User Account

7. Click on Update button if user wants to update any user account
8. Click on Delete button if user want to delete any particular account
9. Click Show Logs Button to show the history of user actions
10. Click Back to go to Main Functions form

B. <u>Show Logs</u>

1. After clicking on Show Logs button, the show dialog will appear, user can choose on the radio button which Logs they want (See Figure 13).
2. Click OK button to continue or Cancel button to go back

C. <u>To Upload Data</u>

1. Click on Upload Data button from Main Function form
2. Browse the file

** Only MS. Excel file can be read*

3. Choose the Month and Year of Report
4. Click on Proceed button

5. The progress bar will run to make sure this process is completed successfully.

* *Make sure Month and Year are chosen*

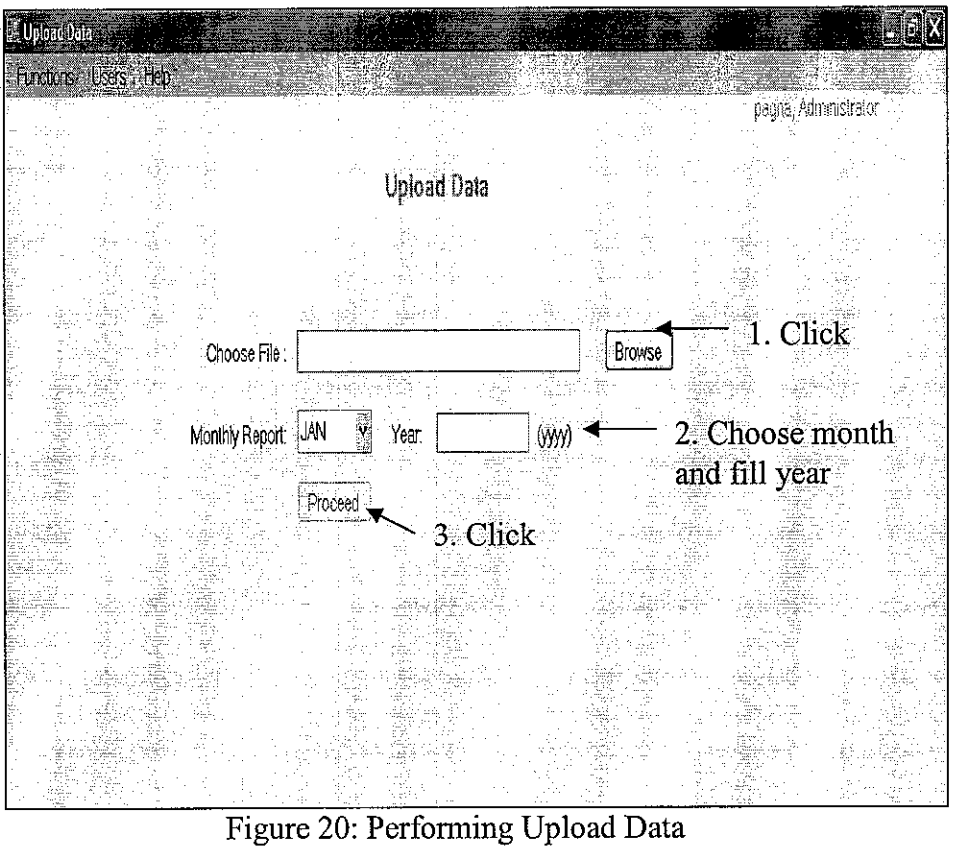


Figure 20: Performing Upload Data

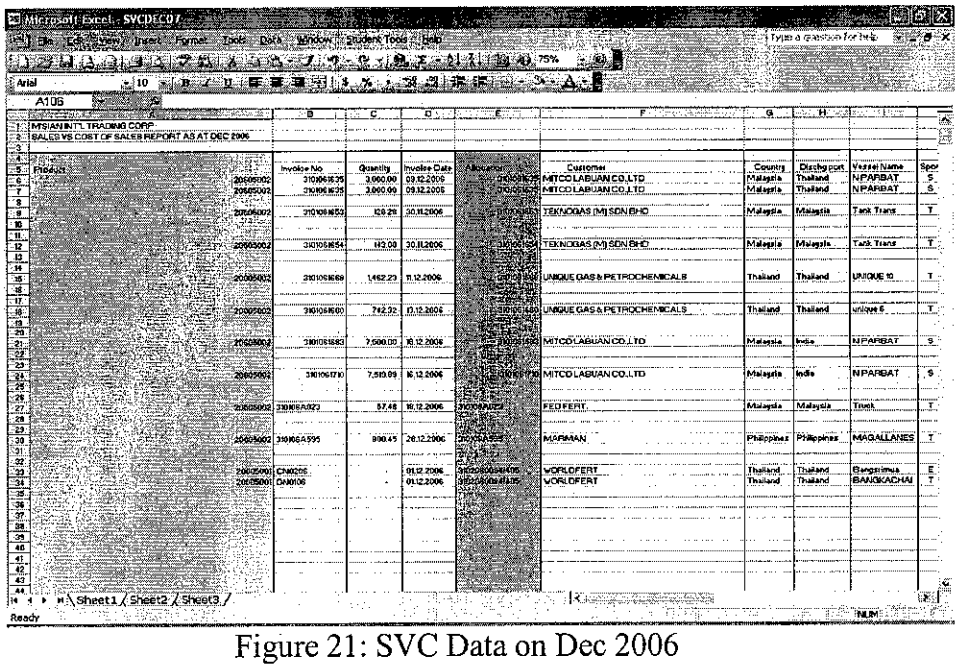


Figure 21: SVC Data on Dec 2006

D. To Generate SVC Report

1. Click on Upload Data button from Main Function form
2. In Report By group, user can chose types of report to generate by

i. Month
ii. Year
iii. Department
iv. Product Name
v. Product Code
vi. Allocation
vii. Customer

3. Click on Generate SVC Report button

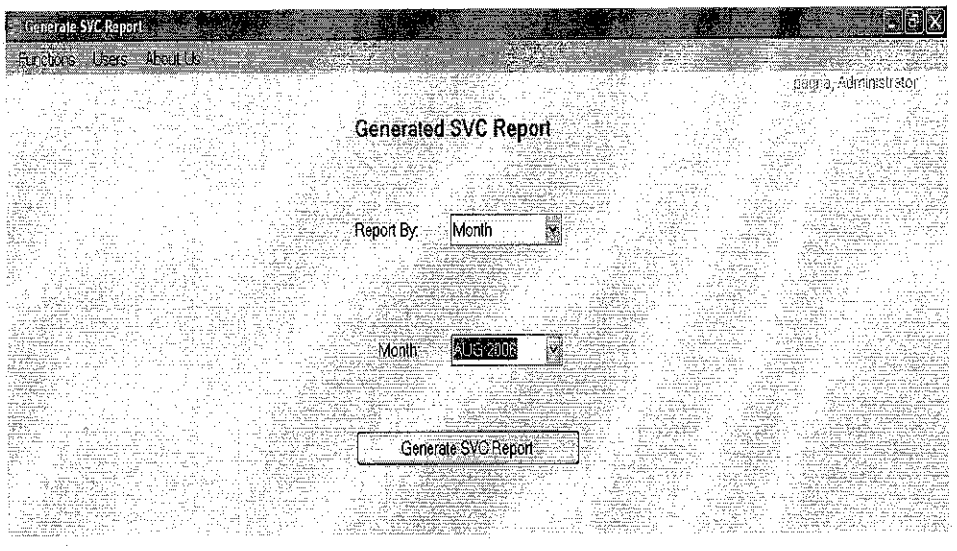


Figure 22: Generate SVC Report on JAN 2007

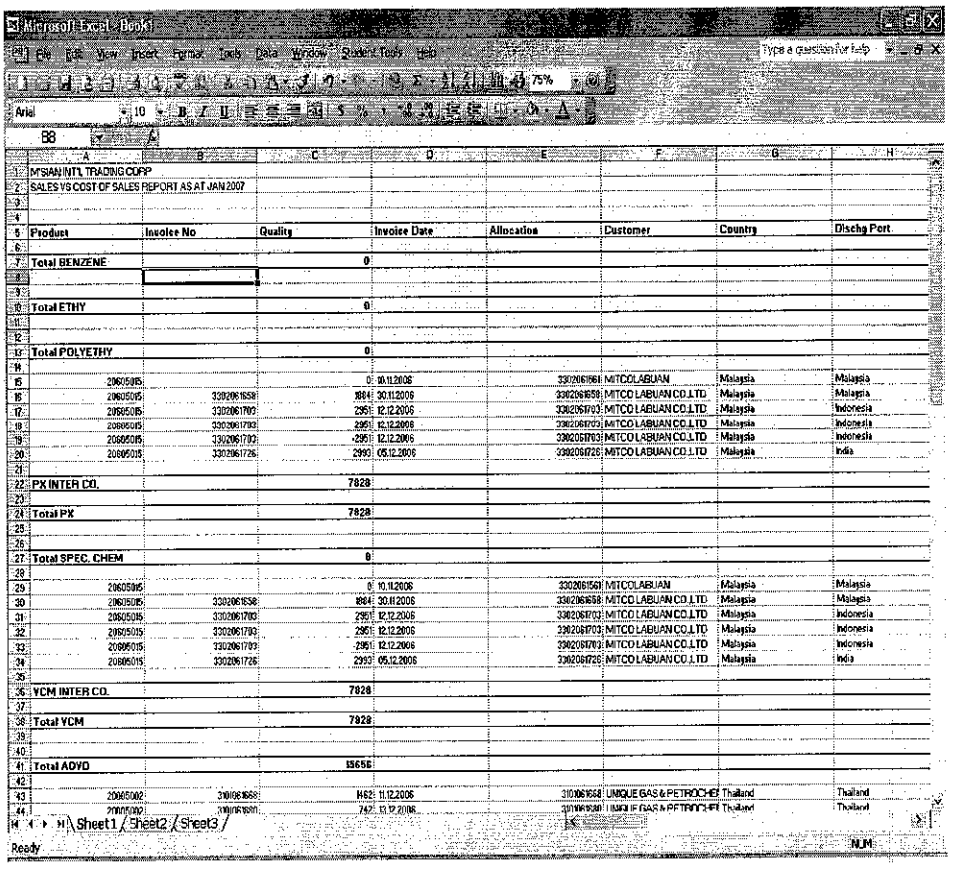


Figure 23: SVC Report After Generated

4.6 Testing and Result

This section will explain the test conducted and the result derived out of these tests and test cases are also developed to help developer to implement the system testing.

4.6.1 Black Box Testing

Black box testing was conducted with the purpose to ensure that the system developed meets the objectives set during the analysis phase. The main objective of the test conducted was to ensure that the system give correct output according to the input or status of the current process state. The correctness of the system flow was also identified and tested during the black box testing. This testing was carried out throughout the project's development phase. The test will be started by allowing the Administrator to create the new user. Then the new user can login into the system and perform upload data and generate the SVC report.

Administrator Create New User Account

The administrator account is the only default account which is automatically created. In here, the administrator account is username: pagna and password: pagna. After the administrator have logged into the system, he or she can change the profile and create new account. The new account is taken to test this time is:

Fields	**Input**
User Name:	syahrul
Password:	Syaharul123
Retype Password:	Syaharul123
Name:	Syahrul Aniza Sharil
Gender:	Female
Date of Birth:	22/05/1985
Privilege:	Administrator
Position:	Accountant
Division:	Finance
Telephone:	Null
Ext:	Null
Mobile:	012 44526988
Email:	Null
Address:	Puchong, Selangor

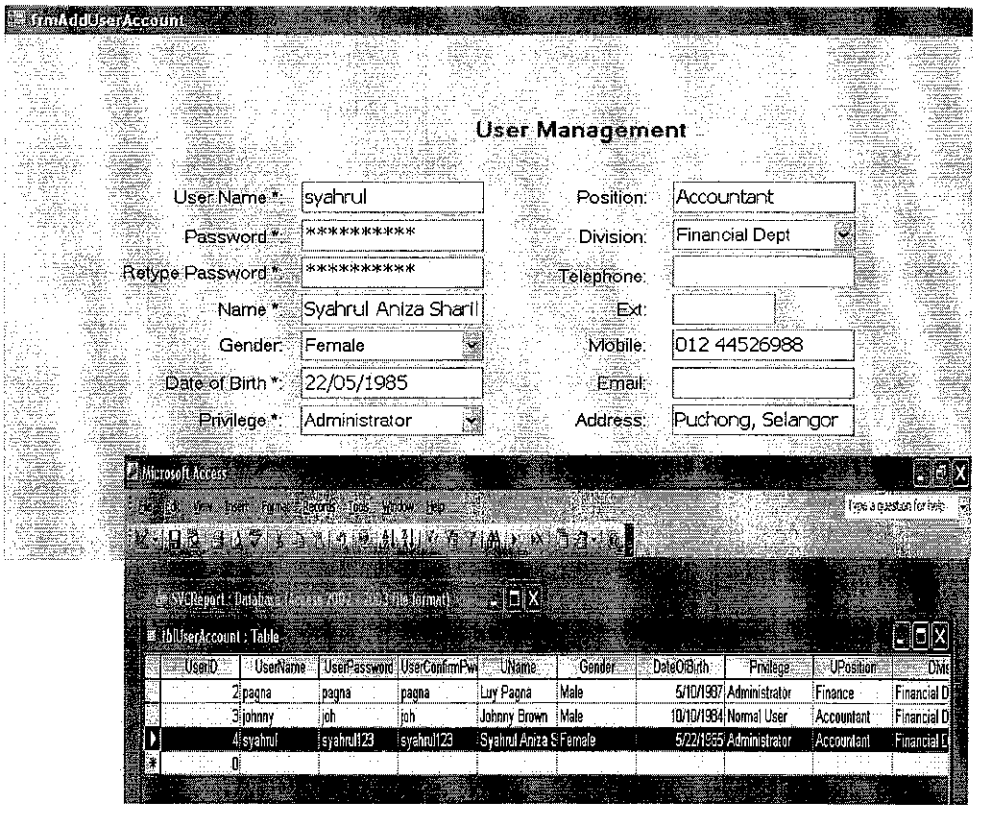


Figure 24: Add syahrul account

After the user click Save button, the data is input into database as shown in Figure 25 in addition, when the user forget to input any fields with (*) sign, the message box will appear to ask use to input it.

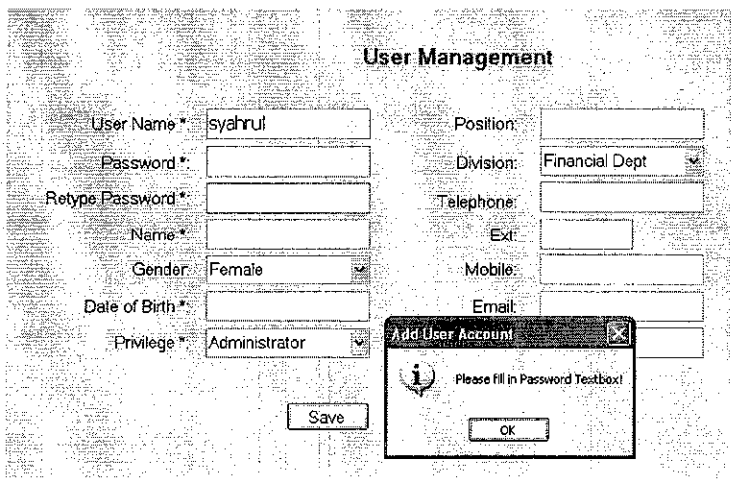


Figure 25: Input Data Confirmation

Log into the system

In order to ensure the login function work properly, testing is applied here with new user just created (syahrul).

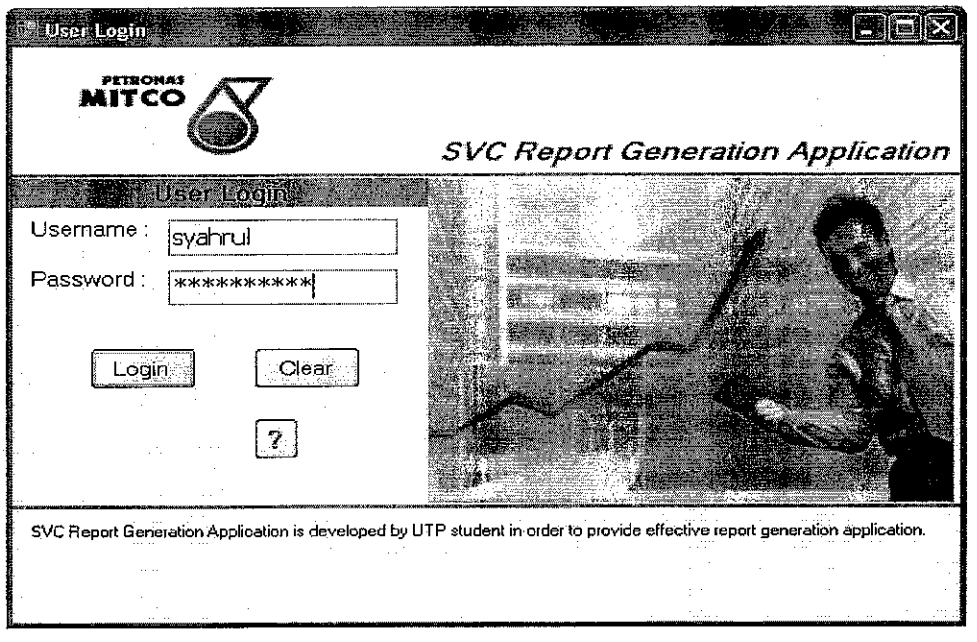


Figure 26: syahrul Login

After click Login button, the system allows this useraccount to proceed and perform any functions in the system.

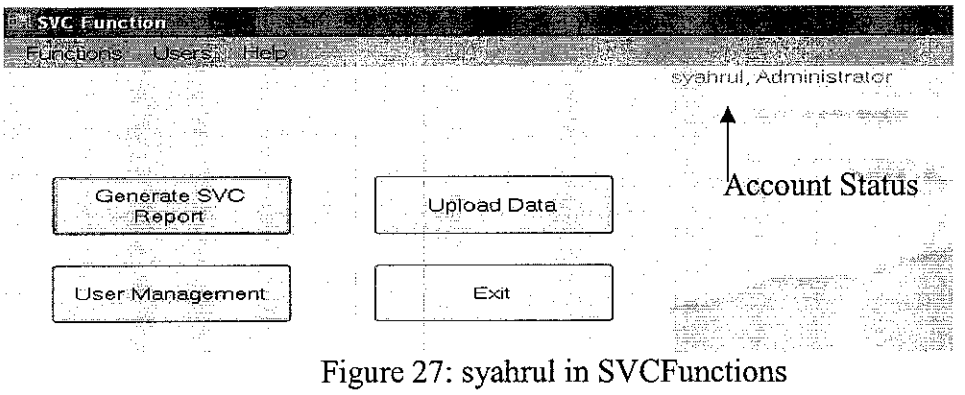


Figure 27: syahrul in SVCFunctions

Upload SVC Data

Upload SVC Data is one of the main functions in this system. It is used to capture or read data from MS. Excel and then pass it into database. To test, the real data of SVC is taken to use. First, browse the SVC Excel file and then select the month and year of report and finally, click on Proceed button.

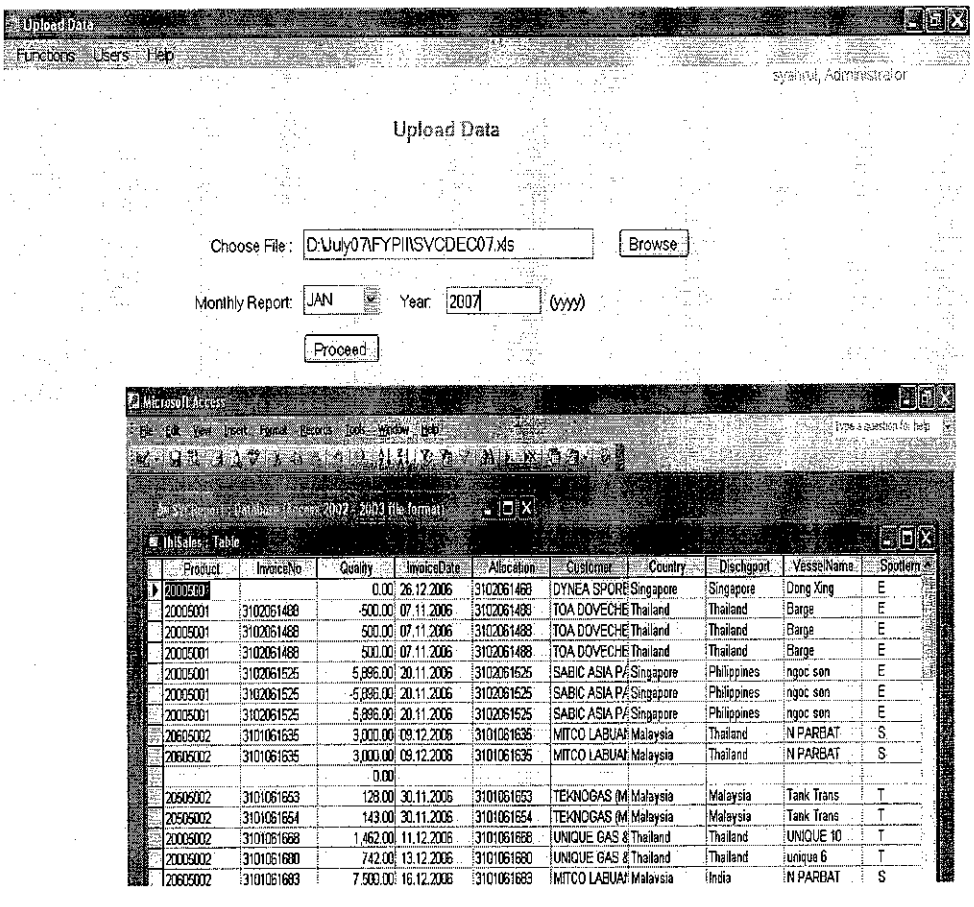


Figure 28: Upload Test

After upload data, all data from a particular Excel will pass into database in MS. Access accordingly.

Generate the SVC Report

In this step, user just simply click on Generate SVC Report button or command in Main Function Form then it will appear the SVC Report Generation form. The user can select the criteria of report which he or she wants to generate base on the Month and Department. By month, the user will allow to choose any months that have uploaded into the database and other is report department which user can select the Department name and month of report.

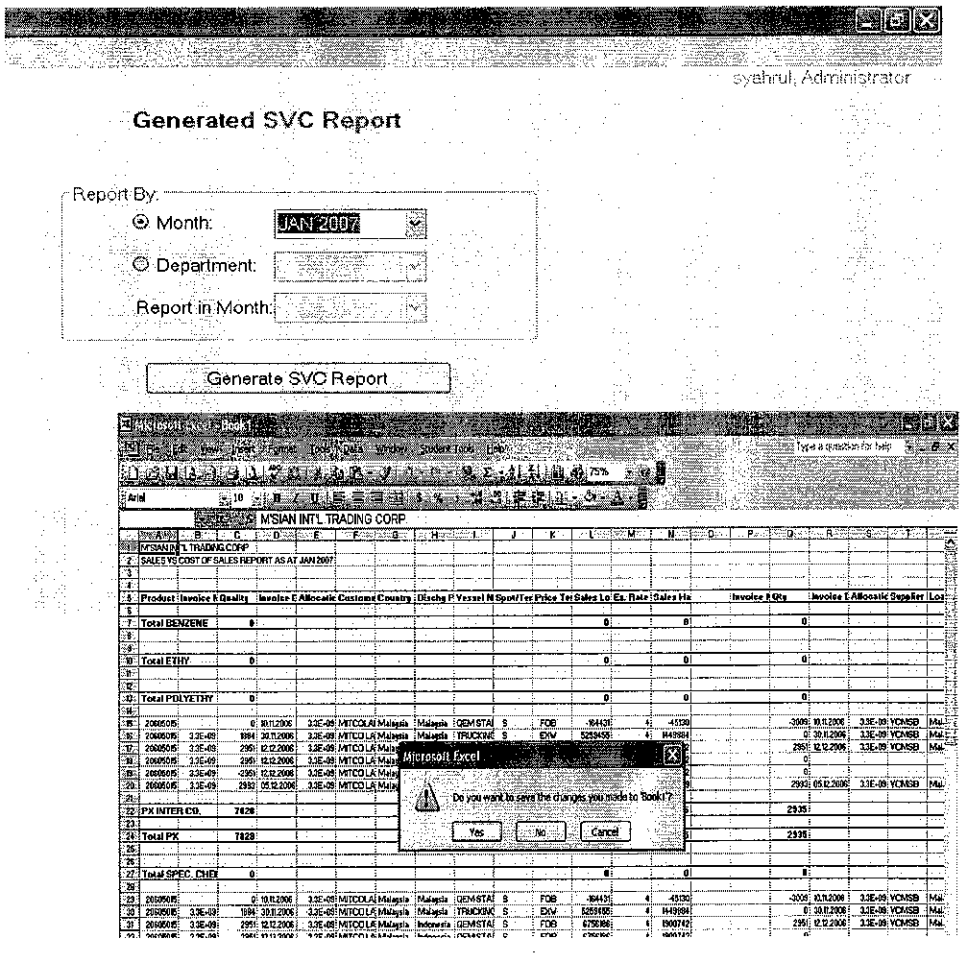


Figure 29: Generate SVC Report Test

As shown in Figure 29, the new report is generated with a specific and standard format that user wants. All data and reports are created just one click. In conclusion, all functions work properly.

4.6.2 Test Cases

Table 16: Create User Account Test Case

No	Test Condition	Operation Action	Performed By	Input Specifications	Output Specifications (Expected Results)	Pass or Fail	Comments
1	New user account can be created by Administrator	User fills in all necessary information required	Administrator	• User Name • Password • Retype Password • Name • Date of Birth • Privilege • Position • Division • Telephone • Ext • Mobile • Email • Address	The account is registered and shows the detail information entered by user	Pass	Normal users can not change their password by themselves
2	Username must be unique	User input duplicated username.	Administrator	• User Name • Password • Retype Password • Name • Date of Birth • Privilege • Position • Division • Telephone • Ext • Mobile • Email • Address	Display error message	Pass	
3	User fail to input data in all	Keep it blank data in	Administrator	• User Name • Password	Display error message and give focus to the	Pass	

	mandatory field	any field which have (*) sign		• Retype Password • Name • Date of Birth • Privilege	field which is blank		
4	Mismatch data in password and retype password field	Input different data in Password and Retype password.	Administrator	• Password • Retype Password	Display error message when user leaved the retype password field	Pass	
5	User can update any data in current showing record	Edit some data shown in User management form	Administrator	• Click on Update button	Display message box to inform that user wants to update a particular data and then display message box show that data updated.	Pass	
6	User can update any data in current showing record	Click delete button to delete a particular data	Administrator	• Click Delete button	Display message box to inform that user wants to delete a particular data and then display message box show that data deleted.	Pass	
7	User can show the logs form	Click on Show Logs button	Administrator	• Click Show Logs button	Display the Show Log form to allow user chooses any logs and then the system will display the user action history of uploading data and generating report	Pass	
8	User can go back to main function	Click on Back button	Administrator	• Click Back button • Click Main Function in Menu	Display Main Function form	Pass	

Table 17: Upload Data Test Case

No	Test Condition	Operation Action	Performed By	Input Specifications	Output Specifications (Expected Results)	Pass or Fail	Comments
1	User can upload SVC data from MS. Excel format	Browse any raw data of SVC in MS. Excel format	All types of User	• Raw SVC data in Excel format	All data with product code are input into database	Pass	Some data may missing because they are not in well format.
2	User can not upload the same file	Upload data by using the previous file	All types of User	• Raw SVC data in Excel format but the same file as one have been input	Display message error, not allow to input these data again	Pass	It can't check this data has input or not if the user changed the file name in different
3	Upload data without input Month and Year	User did not choose the Month and file the year data	All types of User	• Blank in Monthly combo box and year text field	Display the message error then ask user to input those data	Pass	
4	User can upload data in different format and/or types of file	Take different format of data and different file types	Raw SVC data in Excel format but the same file as one have been input	• File with different format from SVC data • Any Files are not MS. Excel	Display error message	fail	

Table 18: Generate SVC Report Test Case

No	Test Condition	Operation Action	Performed By	Input Specifications	Output Specifications (Expected Results)	Pass or Fail	Comments
1	User can generate SVC report to MS. Excel format	Generate the report in a particular month by clicking Generate SVC Report button	All types of User	• Click Generate SVC Report	New Excel file will generated with standard format and based on criteria	Pass	
2	User can generate SVC report without choosing the month	Click Generate SVC Report button to generate report without choosing any particular month	All types of User	• Keep Monthly combo box blank • Click Generate SVC Report	Display message box to inform user to choose any month of report that he or she wants to generate	Pass	
3	User ca n generate SVC report base on the types	Generate report by changing all criteria	All types of User	• Month • Year • Department • Product Name • Product Code • Allocation • Customer	The Excel will generate the result base on the criteria given	Pass	Just only the criteria given. It may need more these.

Table 19: Track User Action Test Case

No	Test Condition	Operation Action	Performed By	Input Specifications	Output Specifications (Expected Results)	Pass or Fail	Comments
1	User can show the logs	Click Show Log button	All types of User	• Click Show Logs button • Select types of Logs • Click Ok	Display the Upload Log or Generation Log form	Pass	
2	User can show both at the same time	Click Show Log button	All types of User	• Click Show Logs button • Select types of Logs • Click Ok	Display only one type of Log form at the time	Fail	

4.6.3 Usability and Acceptance Test

Usability testing was conducted for measuring how well the user can use some human-made object such as Interface and overall appearances of the application for its intended purpose. Usability testing focuses on a particular object or a small set of objects, whereas general human-computer interaction studies attempt to formulate universal principles. In addition, SVC Report Generation was also tested by the Acceptance test in order to confirm that the system is complete, meets the business needs that prompted the system to be developed, and is acceptable to the users. In order to do these tests, the survey was developed for the testing the system purpose. The target audience is the future user of the system. This questionnaire is used to gather and collect information, opinion and feedback on the project the developed prototype system. There are eleven users (UTP students) who have used and played with the SVC Report Generation Application. As the results, the graph results from the testers, supported by www.freeonlinesurvey.com , are shown as the following:

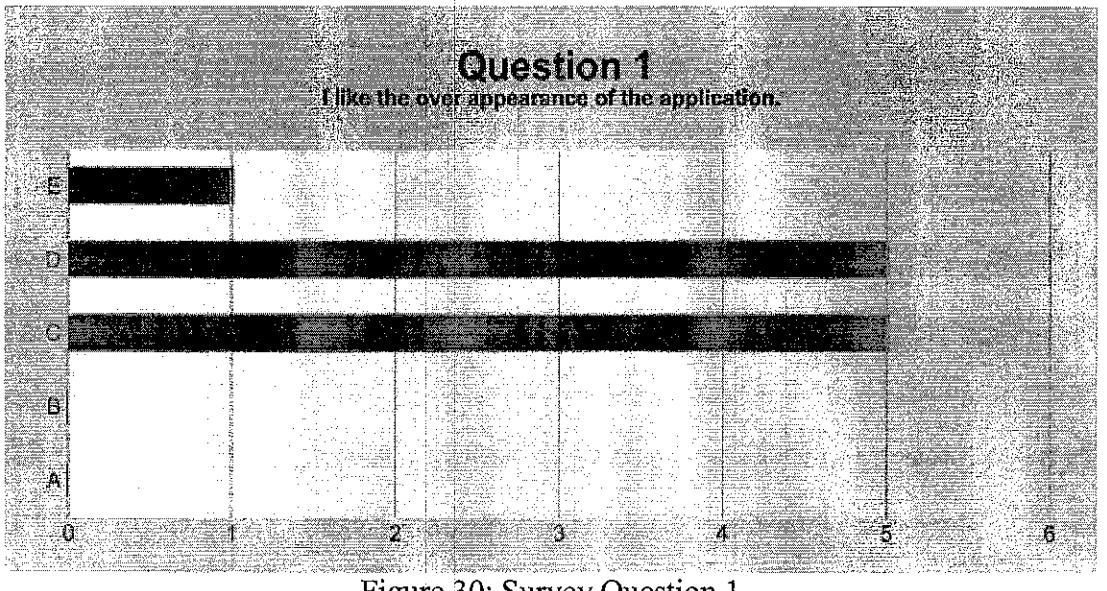


Figure 30: Survey Question 1

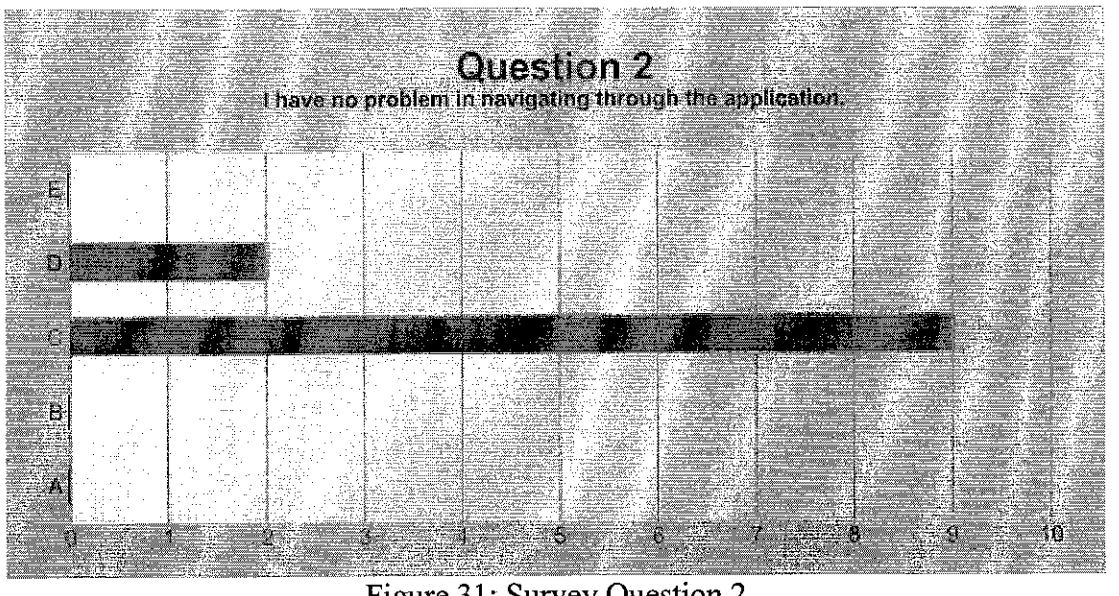


Figure 31: Survey Question 2

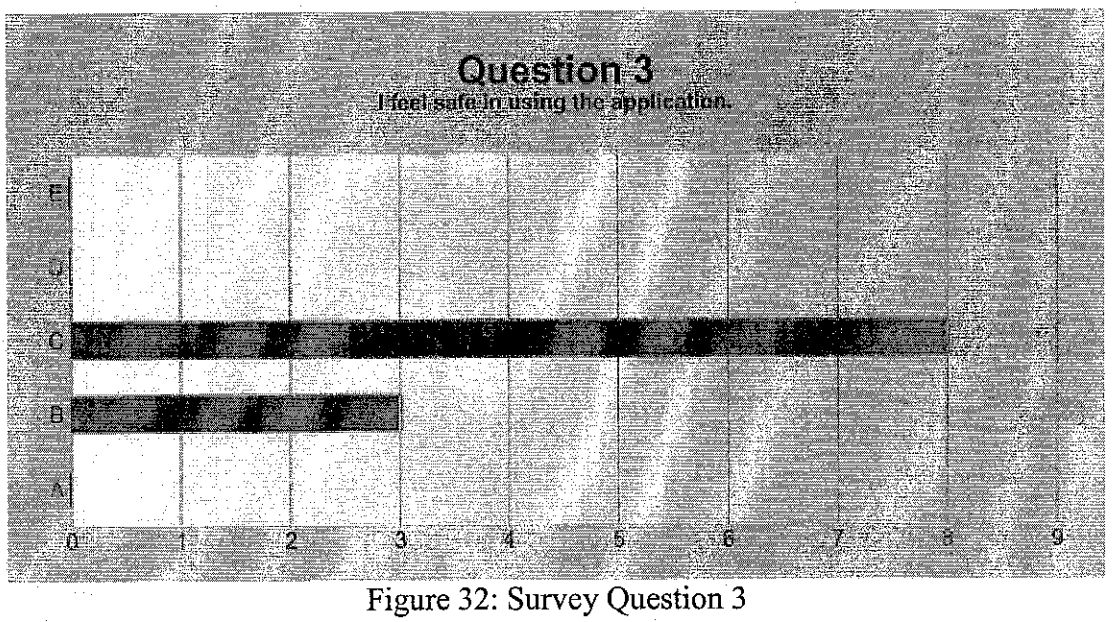


Figure 32: Survey Question 3

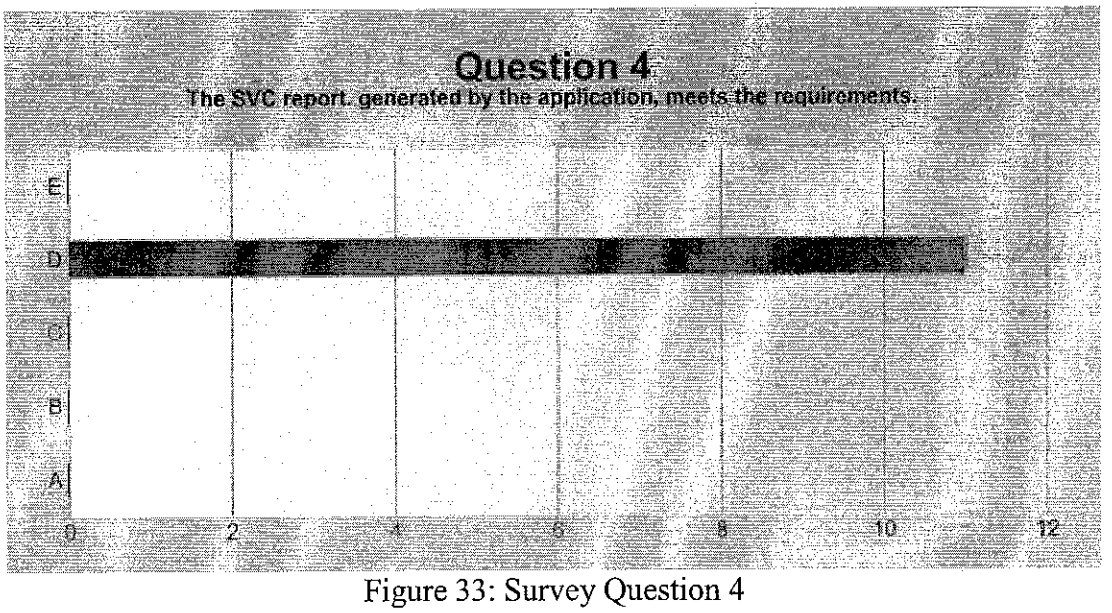


Figure 33: Survey Question 4

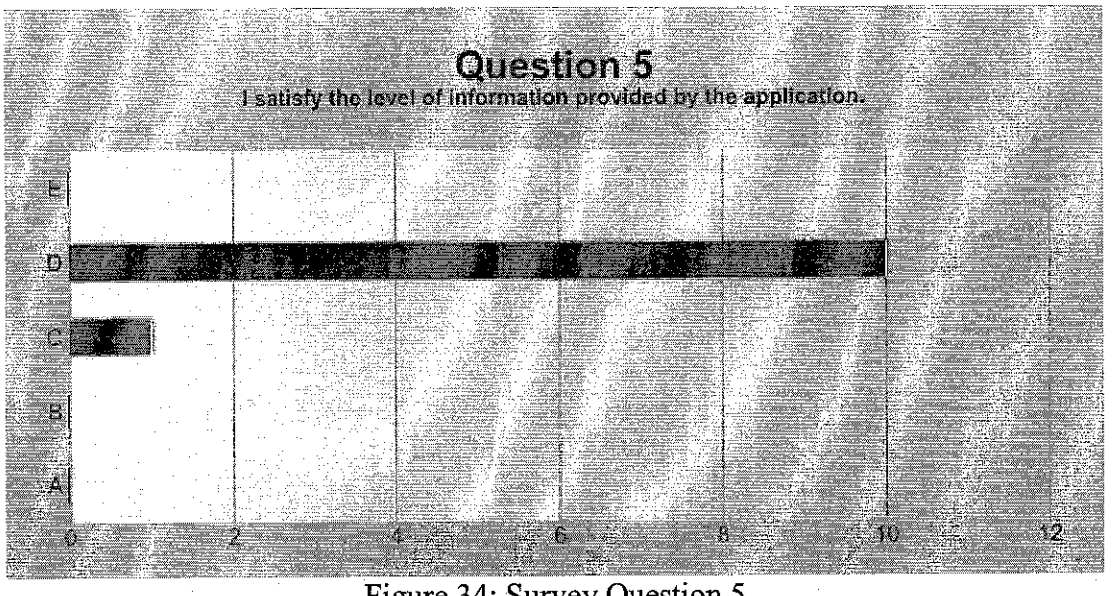


Figure 34: Survey Question 5

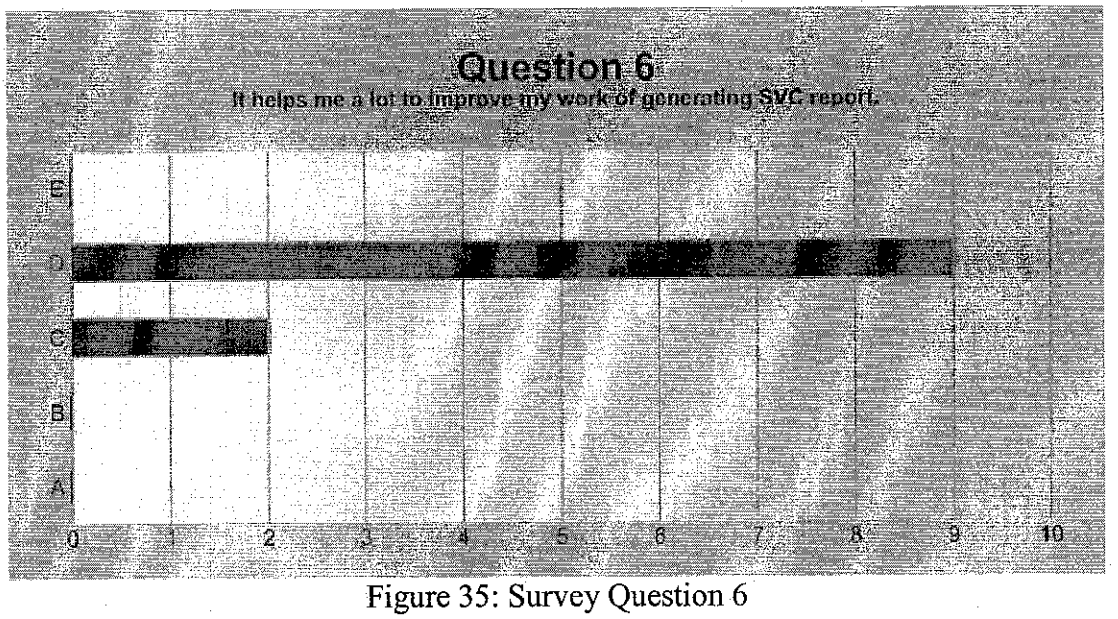


Figure 35: Survey Question 6

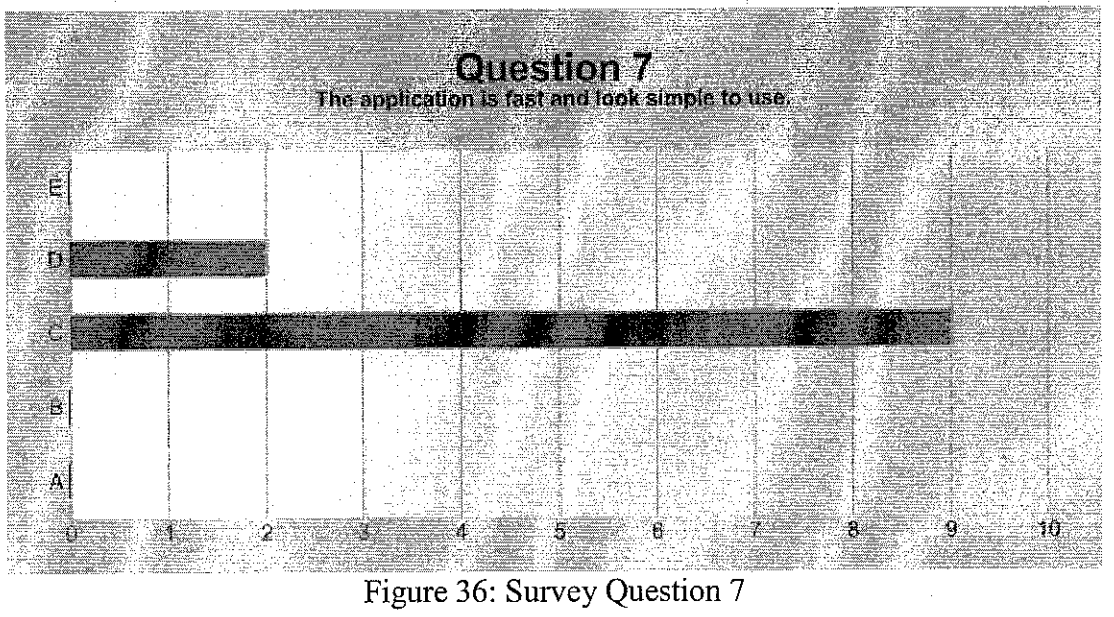


Figure 36: Survey Question 7

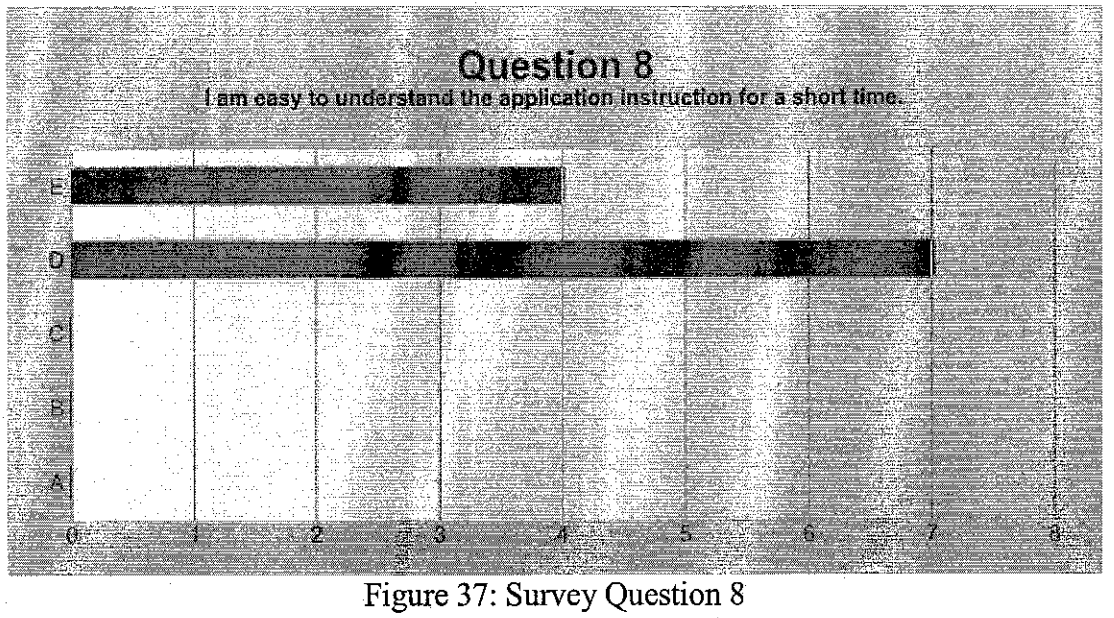


Figure 37: Survey Question 8

The results above have shown that there are more than 80 percents of users agreed that the overall appearance of the SVC Report Generation is good. More than 80 percents of testers agreed that the information or report which generated by the application is satisfied and the application helped them a lot of improve their works in generating Sale and Cost report. However, the results have also shown the level of security need to improve to make the system more effectiveness.

4.7 Limitation

SVC Report Generation system provides a good performance of generating the dynamic report, by the way, it can't capture many kinds of file except MS. Excel. In addition, the format of data in MS. Excel must be the same in order the system can capture or write data in the same structure. Another hand, this system would give the user the ability to generate the SVC report base on the criteria given such as Monthly, Annual, Product Name, Product Code, Department, Allocation, Supplier. If the user needs the other types of this report, it may use other way also. The limitation of system occurred because the system is developed to solve the specific problem of generating report for MITCO. Thus it could be improved more in order to increase the functionalities and its uses.

CHAPTER 5

CONCLUSION

Sales VS Cost (SVC) Report Generation System has been successfully developed by using MS. VB 2005. The system could provide the user the convenient way to generate the report dynamically. It would help Financial Department of MICTO to solve their problem of generating the report using Lotus approach. Especially, they could take full advantages of this system in order to improve their work performance and save time and effort. However, there are some challenging faced during development time. For example, the understand user requirements is one of core success in developing the system because developer can't do anything without knowing the requirements and translate them into functionality of the system. From this point of view, developer and customer or stakeholder must keep close communication to understand the system functionalities well and satisfy the customer's need. To be a developer, the programming skill is also very important in order to develop the system fast and efficiently. From the whole development process of this project, many experiences and lessons have been learnt and there are values used in the future. In conclusion, SVC Report Generation System would provide the suitable solution and technology to satisfy the customer's requirements.

REFERENCES

[1] Chee Hong and Sai Peck: "A Report Generator Component System using XML-driven, Component-Based Development Approach", 2003, University Malaya

[2] Daniel K.C. Chan: "A Document-Driven Approach to Database Report Generation", 1998, Project VERSO

[3] Microsoft Visual Basic 2005: RELOADED, second edition, by Diane Zak, 2007

[4] Beginning Object-Oriented Programming with VB 2005: From Novice to Professional, by Daniel R. Clark, Apress Copyright © 2006.

[5] Beginning Visual Basic®2005 Databases, by Thearon Willis, Copyright © 2006 by Wiley Publishing, Inc., Indianapolis, Indiana, ISBN:0-7645-8894-X.

[6] Effective Prototyping for Software Makers, by Jonathan, Michael, Nevin, 2007, ISBN 13: 978-0-12-088568-8

[7] Mark A., Ph.D (2002): Developing an Online Health Outcome Report Generation System for Asthma in Wisconsin.

[8] http://en.wikipedia.org/wiki/Microsoft_Access

[9] http://www.ujihara.jp/iTextdotNET/en/index.html

[10] www.microsoft.com/presspass/press/2006/apr06/04-19VSExpressFreePR.mspx

[11] http://www.ujihara.jp/iTextdotNET/en/

[12] Generating Microsoft Excel Reports in .NET, by Mark Bourisaw March, 2004

[13] Beth Frances Cox (August 2000): Spreadsheets: Bigger And Better

[14] A Brief History of Spreadsheets, by D. J. Power Editor, DSSResources.COM

[15] www.umkc.edu/registrar/sis/glossary.asp

[16] http://www.webopedia.com/TERM/R/report.html

[17] David Jeavons (December 2007): ADO.NET For Beginners

[18] Ganesh (June, 2003): Creating an Excel Spreadsheet Programmatically Using VB.Net

[19] Microsoft Knowledge Base (July, 2004), Article ID: 316934, "How To Use ADO.NET To Retrieve and Modify Records in an Excel Workbook with Visual Basic.Net

[20] John Macomber (June, 2006): Automate Excel VB 2005 Express

APPENDIX A

Manipulate Data with ADO.NET

This section will describe briefly how to manipulate data such retrieve, update, and delete data by using ADO.NET.

An Introduction to the System.Data object

In order to start working with databases developer first need to ensure that he or she has a reference to the System.Data object. To do this:

1. Select the *Add Reference* menu option from the *Project* menu.
2. Highlight COM Tab
3. Choose Microsoft ActiveX Data Objects 2.8 Library

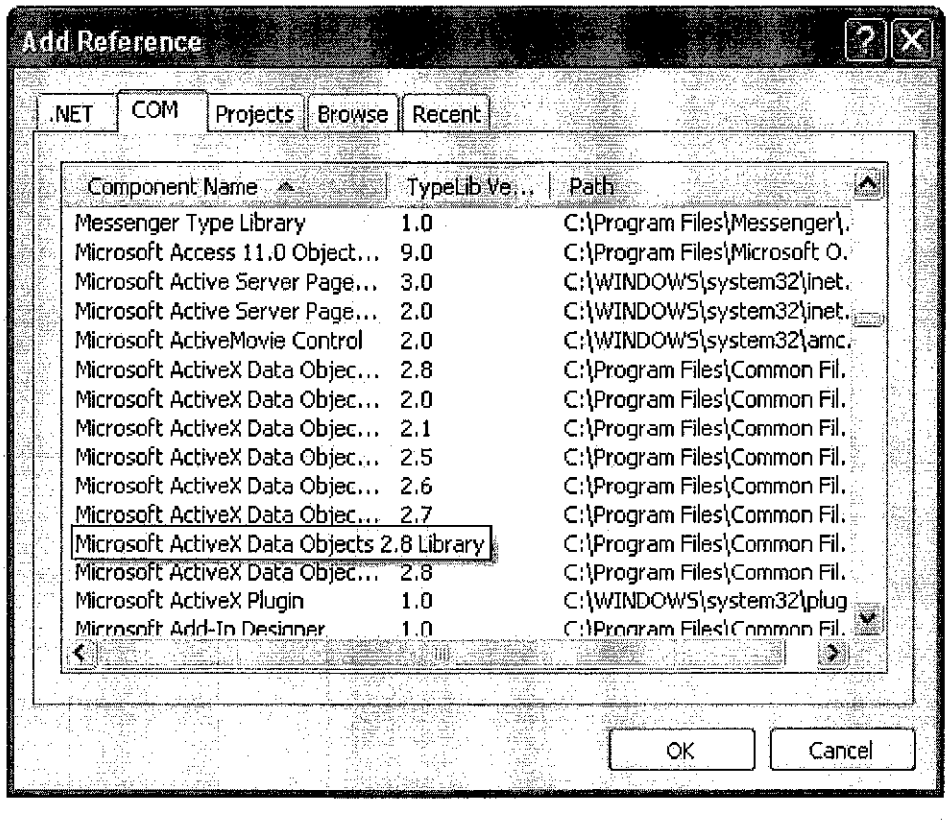


Figure 38: Add ActiveX Data Objects 2.8 Library

4. Click Ok

Retrieving Data from the database

1. Imports System.Data.OleDb

```
1  Imports System.Data.OleDb
2
3  Public Class Form1
4
5      Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
6
7      End Sub
8
9  End Class
```

Figure 59: Import OleDb

2. Create The ConnectionString. For example:

"Provider=Microsoft.Jet.OleDb.4.0;Data Source=" & Application.StartupPath & "\SVCReport.mdb"

** This specifies that we will be using the Microsoft.Jet.OleDb.4.0 provider and the location of the database is in the StartupPath of the application (should be your bin\Debug folder of your project) and the database is called SVCReport.mdb.*

3. Create DataAdapter and fill it to database table

Sample Code:

```
        Dim connStr As String = "Provider =
        Microsoft.Jet.OLEDB.4.0;Data Source = C:\Luy Pagna Final
        Project\SVCReport.mdb"
        Dim sqlStr As String = "SELECT UserName,Privilege FROM
        tblUserAccount"
        Dim dataAdapter As New OleDb.OleDbDataAdapter(sqlStr,
connStr)
        dataAdapter.Fill(dt)
        dataAdapter.Dispose()
        Dim i As Integer = frmUserLogin.index
        lbluserdesc.Text = dt.Rows(i)("UserName") & ", " &
        dt.Rows(i)("Privilege")
```

Adding data to the Table

1. Create The ConnectionString. For example:
"Provider=Microsoft.Jet.OleDb.4.0;Data Source=" & Application.StartupPath &
"\SVCReport.mdb"
2. Declare OleDbCommand and Execute the command by ExecuteNonQuery
3. Close Connection

Sample Code

```
sql = "INSERT INTO tblListUploadLogs
(UserName,FileName,CreatedDate) VALUES ('" _
& "Luy Pagna','" & FilePath & "',#" & System.DateTime.Now & "#)"

conn.Open()
Dim command As New OleDb.OleDbCommand(sql, conn)
command.ExecuteNonQuery()
conn.Close()
```

Updating and Delete Data from Table

In order to update or delete data from specific table, the codes are changes only the SQL statement.

Update SQL statement

```
UPDATE tablename SET fieldname=Value, secondfieldname=Value WHERE fieldname=Value
```

Sample Code

```
Dim conn As New OleDb.OleDbConnection(connStr)
Dim sqlUpdateUser As String = String.Empty
            ''''''''''''
Dim UserID As Integer = 10
sqlUpdateUser = "UPDate tblUserAccount SET UserName='" &
txtUserName.Text & "',UserPassword='" & txtPassword.Text & "'," _
& "UserConfirmPwd='" & txtConfirmPassword.Text &...
& "WHERE UserID=" & UserID
conn.Open()
Dim command As New OleDb.OleDbCommand(sqlUpdateUser, conn)
command.ExecuteNonQuery()
conn.Close()
```

Delete SQL Statement

```
DELETE FROM tablename WHERE fieldname=Value
```

Sample Code

```
Dim conn As New OleDb.OleDbConnection(connStr)
Dim sqlDeleteUser As String = String.Empty
''''''''''
UserID As Integer = 10

sqlDeleteUser = "DELETE FROM tblUserAccount WHERE UserID=" &
UserID
conn.Open()
Dim command As New OleDb.OleDbCommand(sqlDeleteUser, conn)
command.ExecuteNonQuery()
```

APPENDIX B

Creating an Excel Spreadsheet Using VB 2005

1. Start Visual Basic 2005 Express and create a new Windows Application project
2. Right Click on Project
3. Highlight Add Reference
4. From the COM tab, select the Microsoft Excel 11.0 or 11.0 Object Library

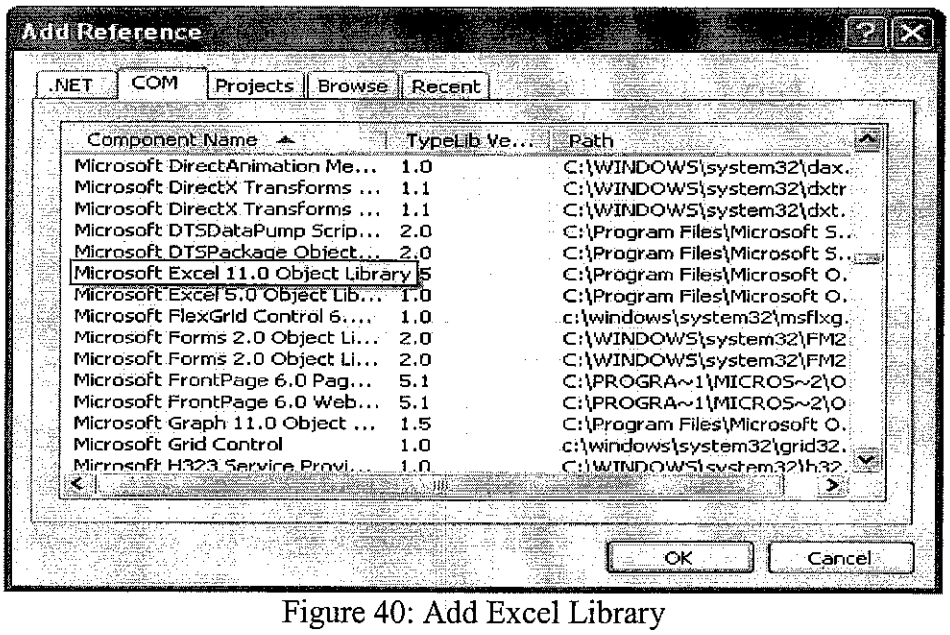


Figure 40: Add Excel Library

Then check in the project folder, it will be shown

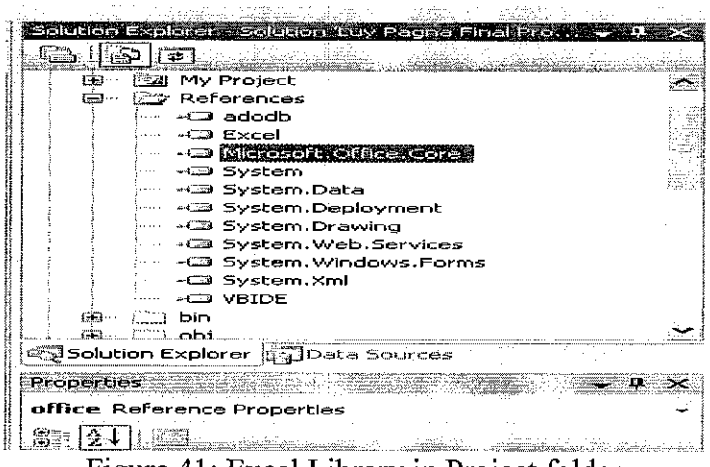


Figure 41: Excel Library in Project folder

Sample code

```
Public Class Form1
Dim objExcel As New Microsoft.Office.Interop.Excel.Application
Private Sub btnGenerateReport_Clcik(..)
        With objExcel
                .Visible = True
                .Workbooks.Open(My.Application.Info.DirectoryPath
            & "\PCSample.xls")
        End With
        With objExcel
                .Visible = True
                .Workbooks.Add()
                .Range("A1").Value       = "A1 Construction, Inc."
                .Range("A1").ColumnWidth = 20
                .Range("A1").Font.Bold   = True
                .Range("A2").Value       = "So. Main St."
                .Range("A3").Value       = "Hartford" &", " & "CT"
        End With
End Sub
End Class
```

APPENDIX C

A Brief History of Spreadsheets

by D. J. Power

Editor, DSSResources.COM

Spreadsheets have been used by accountants for hundreds of years. Computerized or electronic spreadsheets are of much more recent origin. Information Systems oral history and some published newspaper and magazine stories celebrate Dan Bricklin as the "father" of the electronic spreadsheet. In 1978, Harvard Business School student, Daniel Bricklin, came up with the idea for an interactive visible calculator (see email from Frankston, 4/15/1999a). Bricklin and Bob Frankston then co-invented or co-created the software program VisiCalc. We can look back and recognize that VisiCalc was the first "killer" application for personal computers.

l-r Dan Bricklin and Bob Frankston approx. 1980

What is a spreadsheet?

In the realm of accounting jargon a "spread sheet" or spreadsheet was and is a large sheet of paper with columns and rows that organizes data about transactions for a

business person to examine. It spreads or shows all of the costs, income, taxes, and other related data on a single sheet of paper for a manager to examine when making a decision.

An electronic spreadsheet organizes information into software defined columns and rows. The data can then be "added up" by a formula to give a total or sum. The spreadsheet program summarizes information from many paper sources in one place and presents the information in a format to help a decision maker see the financial "big picture" for the company.

Beginnings and the "Tale of VisiCalc"

In 1961, Professor Richard Mattessich pioneered the development of computerized speadsheets for use in business accounting. Some historical information on the computerization of accounting spread sheets using mainframe computers is discussed on Mattessich's web page "Spreadsheet: Its First Computerization (1961-1964)". Rene Pardo and Remy Landau co-invented "LANPAR" LANguage for Programming Arrays at Random in 1969. This electronic spreadsheet type application was used for budgeting at Bell Canada, AT&T, Bell operating companies, and General Motors. They received a US patent (no. 4,398,249) for LANPAR in August 1982 after 12 years of litigation. Mattessich, Pardoe and Landau's work and that of other developers of spreadsheets on mainframe computers probably had no influence on Bricklin and Frankston. Therefore, a history of the modern era of microcomputer-based electronic spreadsheets should begin with the "Tale of VisiCalc".

The tale of VisiCalc is part myth and part fact for most of us. The story is that Dan Bricklin was preparing a spread sheet analysis for a Harvard Business School "case study" report and had two alternatives: 1) do it by hand or 2) use a clumsy time-sharing mainframe program. Bricklin thought there must be a better way. He wanted a program where people could visualize the spreadsheet as they created it. His metaphor was "an electronic blackboard and electronic chalk in a classroom."

By the fall of 1978, Bricklin had programmed the first working prototype of his concept in integer basic. The program helped users input and manipulate a matrix of five columns and 20 rows. The first version was not very "powerful" so Bricklin

recruited an MIT acquaintance Bob Frankston to improve and expand the program. Bricklin calls Frankston the "co-creator" of the electronic spreadsheet. Frankston created the production code with faster speed, better arithmetic, and scrolling. He also expanded the program and "packed the code into a mere 20k of machine memory, making it both powerful and practical enough to be run on a microcomputer". For more details check Dan Bricklin's email from May 12, 1999.

During the fall of 1978, Daniel Fylstra, founding Associate Editor of Byte Magazine, joined Bricklin and Frankston in developing VisiCalc. Fylstra was also an MIT/HBS graduate. Fylstra was "marketing-oriented" and suggested that the product would be viable if it could run on an Apple micro-computer. Bricklin and Frankston formed Software Arts Corporation on January 2, 1979. In May 1979, Fylstra and his firm Personal Software (later renamed VisiCorp) began marketing "VisiCalc" with a teaser ad in Byte Magazine. The name "VisiCalc" is a compressed form of the phrase "visible calculator" (see email from Frankston, 4/15/1999b).

VisiCalc became an almost instant success and provided many business people with an incentive to purchase a personal computer or an H-P 85 or 87 calculator from Hewlett-Packard (cf., Jim Ho, 1999). About 1 million copies of the spreadsheet program were sold during VisiCalc's product lifetime. Dan Bricklin has his version of the history of Software Arts and VisiCalc on the web at www.bricklin.com/history/sai.htm. Bricklin includes early ads and reviews and pictures of the VisiCalc packaging and screenshots.

What came after VisiCalc?

The market for electronic spreadsheet software was growing rapidly in the early 1980s and VisiCalc stakeholders were slow to respond to the introduction of the IBM PC that used an Intel computer chip. Beginning in September 1983, legal conflicts between VisiCorp and Software Arts distracted the VisiCalc developers, Bricklin and Frankston. During this period, Mitch Kapor developed Lotus and his spreadsheet program quickly became the new industry spreadsheet standard.

What is Lotus 1-2-3?

Lotus 1-2-3 made it easier to use spreadsheets and it added integrated charting, plotting and database capabilities. Lotus 1-2-3 established spreadsheet software as a major data presentation package as well as a complex calculation tool. Lotus was also the first spreadsheet vendor to introduce naming cells, cell ranges and spreadsheet macros. Kapor was the VisiCalc product manager at Personal Software for about six months in 1980; he also designed and programmed Visiplot/Visitrend which he sold to Personal Software (VisiCorp)for $1 million. Part of that money along with funds from venture capitalist Ben Rosen were used to start Lotus Development Corporation in 1982. Kapor cofounded Lotus Development Corporation with Jonathan Sachs. Before he cofounded Lotus, Kapor disclosed and offered Personal Software (VisiCorp) his initial Lotus program. Supposedly VisiCorp executives declined the offer because Lotus 1-2-3's functionality was "too limited". Lotus 1-2-3 is still one of the all-time best selling application software packages in the world (see email from Mitch Kapor, 04/15/1999).

Kapor served as the President and Chief Executive Officer of Lotus from 1982 to 1986 and as a Director until 1987. In 1983, Lotus' first year of operations, the company reported revenues of $53 Million and had a successful public offering. In 1984, Lotus tripled in revenue to $156 Million. The number of employees at Lotus grew to over a thousand by 1985. This rapid growth led to a shakeout in the spreadsheet segment of the personal computer software industry.

In 1985, Lotus Development acquired Software Arts and discontinued the VisiCalc program. A Lotus spokeperson indicated at that time that "1-2-3 and Symphony are much better products so Visicalc is no longer necessary."

What about Microsoft Excel and Bill Gates?

The next milestone was the Microsoft Excel spreadsheet. Excel was originally written for the 512K Apple Macintosh in 1984-1985. Excel was one of the first spreadsheets to use a graphical interface with pull down menus and a point and click capability using a mouse pointing device. The Excel spreadsheet with a graphical user interface was easier for most people to use than the command line interface of PC-DOS spreadsheet products. Many people bought Apple Macintoshes so that they

could use Bill Gates' Excel spreadsheet program. There is some controversy about whether a graphical version of Microsoft Excel was released in a DOS version. Microsoft documents show the launch of Excel 2.0 for MS-DOS version 3.0 on 10/31/87.

When Microsoft launched the Windows operating system in 1987, Excel was one of the first application products released for it. When Windows finally gained wide acceptance with Version 3.0 in late 1989 Excel was Microsoft's flagship product. For nearly 3 years, Excel remained the only Windows spreadsheet program and it has only received competition from other spreadsheet products since the summer of 1992.

By the late 1980s many companies had introduced spreadsheet products. Spreadsheet products and the spreadsheet software industry were maturing. Microsoft and Bill Gates had joined the fray with the innovative Excel spreadsheet. Lotus had acquired Software Arts and the rights to VisiCalc. Jim Manzi had become CEO at Lotus in April 1986 and in July 1986 Mitch Kapor resigned as Chairman of the Board. The spreadsheet entrepreneurs were moving on ...

Legal Battles

In January of 1987, Lotus Development filed suit against Paperback Software and separately against Mosaic Software claiming they had infinged on the Lotus 1-2-3 spreadsheet software. In a related matter, Software Arts, the developerof the original VisiCalc spreadsheet software filed a separate action against Lotus claiming that Lotus 1-2-3 was an infringement of VisiCalc. Briefly, Lotus won the legal battles, but lost the "market share war" to Microsoft. According to Russo and Nafziger (1993) "The Court granted Lotus' motion dismissing the Software Arts' action and confirming that Lotus had acquired all rights, including all claims, as part of the earlier transaction."

Most people have probably forgotten the Lotus clones, TWIN and VP Planner. Twin was designed to work like Lotus' 1-2-3 and advertising proclaimed it "offers you so much more, for so muchless." Paperback Software published a spreadsheet software product called VP Planner.

Russo and Nafziger note "Both Mosaic's TWIN and Paperback's VP Planner had most of the same features, commands, macro language, syntax, organization and sequence of menus and messages as Lotus' 1-2-3. Their visual displays were not however identical to 1-2-3 or to each other. Both TWIN and VPPlanner reorganized and placed their respective menus, sub-menus, prompts and messages on the bottom of the screen."

On June 28, 1990, Judge Keeton of the Federal District Court in Boston upheld the copyright of the Lotus 1-2-3 user interface. The Court ruled that "[t]his particular expression of a menu structure is not essential to theelectronic spreadsheet idea, nor does it merge with the somewhat less abstract idea of a menu structure for an electronic spreadsheet....the overall structure, the order of commands in each menu line, the choice of letters, words, or 'symbolic tokens' to represent each command, the presentation of these symbolic tokens on the screen, the type of menu system used, and the long prompts -- could be expressed in a great many if not literally unlimited number of ways." Lotus Dev. Corp. v. Paperback Software Int'l, 740 F.Supp. 37, 67 (D.Mass. 1990).

What about recent history?

In the late spring of 1995, IBM acquired Lotus Development and Microsoft Excel is the spreadsheet market leader.

In October 2003, Dan Bricklin is working at Interland, Inc. at interland.com and he is maintaining an interesting Web Site at URL www.bricklin.com. Dan has VisiCalc at his site. Lotus gave him permission to post a working copy of the 1981 IBM PC version of the VisiCalc spreadsheet program on his web site. You can download it and run it on a PC using MSDOS in Windows 95 or 98.

Bob Frankston is "pursuing a number of projects ..." at www.frankston.com.

According to a Red Herring Profile, Mitch Kapor "gradually traded in his position as an entrepreneur searching for the next big technology idea for the long-term advisory role of angel investor". In January, 1999, Mitch Kapor joined Accel Partners, a venture capital firm based in Palo Alto, California (URL http://www.accel.com/). Mitch's web site is Kapor Enterprises, Inc. at http://www.kei.com/.

Currently, Dan Fylstra is president of PC software vendor Frontline Systems, Inc. at www.frontsys.com. Frontline Systems Inc. is a developer of spreadsheet solver add-ins for Excel, Lotus 123 and other spreadsheet programs. A solver add-in can be used for both equation-solving (often called goalseeking) and for constrained optimization using linear programming, nonlinear programming, and integer programming methods.

Professor Richard Mattessich is retired and an emeritus Professor of Commerce and Business Administration at the University of British Columbia (email: richard.mattessich@commerce.ubc.ca).

APPENDIX D

Visual Basic .NET

From Wikipedia, the free encyclopedia

Visual Basic .NET (VB.NET) is an object-oriented computer language that can be viewed as an evolution of Microsoft's Visual Basic (VB) implemented on the Microsoft .NET framework. Its introduction has been controversial, as significant changes were made that broke backward compatibility with VB and caused a rift within the developer community.

The great majority of VB.NET developers use Visual Studio .NET as their integrated development environment (IDE). Sharp Develop provides an open-source alternative IDE.

Like all .NET languages, programs written in VB.NET require the .NET framework to execute.

Versions of Visual Basic .NET

As of November 2006 there are three versions of Visual Basic .NET.

Visual Basic .NET

The original Visual Basic .NET was released alongside Visual C# and ASP.NET in 2002. C# — widely touted as Microsoft's answer to Java — received the lion's share of media attention, while VB.NET (sometimes known as *VB7*) was not widely covered. As a result, few outside the Visual Basic community paid much attention to it.

Those who did try the first version found a powerful but very different language under the hood, with disadvantages in some areas, including a runtime that was ten times as large to package as the VB6 runtime and an increased memory footprint.

Visual Basic .NET 2003

Visual Basic .NET 2003 was released with version 1.1 of the .NET Framework. New features included support for the .NET Compact Framework and a better VB upgrade wizard. Improvements were also made to the performance and reliability of the .NET IDE (particularly the background compiler) and runtime.

In addition, Visual Basic .NET 2003 was also available in the *Visual Studio .NET 2003 Academic Edition* (VS03AE). VS03AE is distributed to a certain number of scholars from each country for free.

Visual Basic 2005

Visual Basic 2005 is the next iteration of Visual Basic .NET, Microsoft having decided to drop the .NET portion of the title.

For this release, Microsoft added many features, including:

- *Edit and Continue* - probably the biggest "missing feature" from Visual Basic, allowing the modification of code and immediate resumption of execution
- Design-time expression evaluation
- The *My* pseudo-namespace (overview, details), which provides:
    - easy access to certain areas of the .NET Framework that otherwise require significant code to access
    - dynamically-generated classes (notably *My.Forms*)
- Improvements to the VB-to-VB.NET converter
- The *Using* keyword, simplifying the use of objects that require the Dispose pattern to free resources
- *Just My Code*, which hides boilerplate code written by the Visual Studio .NET IDE
- Data Source binding, easing database client/server development

The above functions (particularly *My*) are intended to reinforce Visual Basic .NET's focus as a rapid application development platform and further differentiate it from C#.

Visual Basic 2005 introduced features meant to fill in the gaps between itself and other "more powerful" .NET languages, adding:

- .NET 2.0 languages features such as:
  - generics
  - *Partial classes*, a method of defining some parts of a class in one file and then adding more definitions later; particularly useful for integrating user code with auto-generated code
  - Nullable Types
- XML comments that can be processed by tools like NDoc to produce "automatic" documentation
- Operator overloading
- Support for unsigned integer data types commonly used in other languages

IsNot Patent

One other feature of Visual Basic 2005 is the conversion of `If Not X Is Y` to `If X IsNot Y` which gained notoriety when it was found to be the subject of a Microsoft patent application.

Visual Basic 2005 Express

Visual Basic 2005 Express - Microsoft's free development application.

As part of the Visual Studio product range, Microsoft has created Visual Studio 2005 Express Editions for hobbyists and novices. One of these editions is Visual Basic 2005 Express Edition. It is available for free from Microsoft.

The Express Editions are targeted specifically for people learning a language. They have a streamlined version of the user interface, and lack more advanced features of the standard versions. On the other hand, Visual Basic 2005 Express Edition *does* contain the Visual Basic 6.0 converter, so it is a reasonable way to evaluate feasibility of conversion from older versions of Visual Basic.

Relation to Visual Basic

Whether Visual Basic .NET should be considered as just another version of Visual Basic or a completely different language is a topic of debate. This is not obvious, as once the methods that have been moved around and which can be automatically converted are accounted for, the basic syntax of the language has not seen many "breaking" changes, just additions to support new features like structured exception handling and short circuited expressions. One simple change that can be confusing to previous users is that of Integer and Long data types, which have each doubled in length; a 16-bit integer is known as a Short in VB.NET, while Integer and Long are 32 and 64 bits respectively. Similarly, the Windows Forms GUI editor is very similar in style and function to the Visual Basic form editor.

The things that *have* changed significantly are the semantics — from those of an object based programming language running on a deterministic, reference-counted engine based on COM to a fully object-oriented language backed by the .NET Framework, which consists of a combination of the Common Language Runtime (a virtual machine using generational garbage collection and a just-in-time compilation engine) and a far larger class library. The increased breadth of the latter is also a problem that VB developers have to deal with when coming to the language, although this is somewhat addressed by the *My* feature in Visual Studio 2005.

The changes have altered many underlying assumptions about the "right" thing to do with respect to performance and maintainability. Some functions and libraries no longer exist; others are available, but not as efficient as the "native" .NET alternatives. Even if they compile, most converted VB6 applications will require some level of refactoring to take full advantage of the new language. Extensive documentation is available to cover changes in the syntax, debugging applications, deployment and terminology.

Comparative samples

The following simple example demonstrates similarity in syntax between VB and VB.NET. Both examples pops a message box saying "Hello, World" with an OK button.

Classic VB example:

```
Private Sub Command1_Click()
    MsgBox "Hello, World"
End Sub
```

A VB.NET example:

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    MessageBox.Show("Hello, World")
End Sub
```

- Note that all procedure calls must be made with parentheses in VB.NET, whereas these were only required for function calls (however in VB6 they could be used in procedure calls as well by using the `Call` keyword)
- Also note that the names `Command1` and `Button1` are not obligatory. However, these are default names for a command button in VB6 and VB.NET respectively.
- Actually, there is a function called `MsgBox` in the `Microsoft.VisualBasic` namespace, but the `System.Windows.Forms.MessageBox` class is a preferred way of displaying message boxes since it has more features and is less language-specific.

The following example demonstrates a difference between VB6 and VB.NET. Both examples unload the active window.

Classic VB Example:

```
Private Sub cmdClose_Click()
    Unload Me
End Sub
```

A VB.NET example:

```
Private Sub btnClose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnClose.Click
    Me.Close()
End Sub
```

Note the 'cmd' prefix being replaced with the 'btn' prefix, conforming to the new convention previously mentioned.

Visual Basic 6 did not provide common operator shortcuts. The following are equivalent: VB6 Example:

```
Private Sub Timer1_Timer()
    Form1.Height = Form1.Height - 1
End Sub
```

VB.NET example:

```
Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
    Me.Height -= 1
End Sub
```

Controversy

Many long-time Visual Basic programmers have complained about Visual Basic .NET, because initial versions dropped a large number of language constructs and user interface features that were available in VB6 (which is now no longer sold), and changed the semantics of those that remained; for example, in VB.NET parameters are (by default) passed by value, not by reference. Detractors refer pejoratively to VB.NET as *Visual Fred* or *DOTNOT*. On March 8, 2005, a petition was set up in response to Microsoft's refusal to extend its mainstream support for VB6 at the end of that month.

VB.NET's supporters state that the new language is in most respects more powerful than the original, incorporating modern object oriented programming paradigms in a more natural, coherent and complete manner than was possible with earlier versions. Opponents tend not to disagree with this, instead taking the position that although VB6 has flaws in its object model, the cost in terms of redevelopment effort is too high for any benefits that might be gained by converting to VB.NET. Independent developers producing software for Internet distribution have also taken issue with the size of the runtime.

It is simpler to decompile languages that target Microsoft Intermediate Language, including VB.NET, compared to languages that compile to machine code. Tools like .NET Reflector can provide a close approximation to the original code due to the large amount of metadata provided in MSIL.

Microsoft supplies an automated VB6-to-VB.NET converter with Visual Studio .NET, which has improved over time, but it cannot convert all code, and almost all non-trivial programs will need some manual effort to compile. Most will need a

significant level of refactoring to work optimally. Visual Basic programs that are mainly algorithmic in nature can be migrated with few difficulties; those that rely heavily on such features as database support, graphics, unmanaged operations or on implementation details are more troublesome.

However in 2005 ArtinSoft, the company that developed the VB6-to-VB.NET converter for Microsoft that comes with Visual Studio .NET, developed a migration tool called the ArtinSoft Visual Basic Upgrade Companion. This tool expands upon the migration wizard included in Visual Studio .NET by providing some automated code refactoring, such as type inference for late-bound variables—producing explicitly typed variables—and conversion to structured error handling, among many other tweaks that improve code quality.

Using artificial intelligence algorithms, it is possible for this new tool to recognize certain code patterns that can be reorganized into more structured versions, yielding a higher quality .NET code. For example, the tool is able to automatically recognize commonly used patterns of "On Error GoTo", analyze them, and convert them to code blocks that use "Try Catch" instead of the legacy error handling model—in many cases with no human intervention.

In addition, the required runtime libraries for VB6 programs are provided with Windows 98 SE and above, while VB.NET programs require the installation of the significantly larger .NET Framework. The framework is included with Windows Vista, Windows XP Media Center Edition, Windows XP Tablet PC Edition and Windows Server 2003. For other supported operating systems such as Windows 2000 or Windows XP (Home or Professional Editions), it must be separately installed.

Microsoft's response to developer dissatisfaction has focused around making it easier to move new development and shift existing codebases from VB6 to VB.NET. Their latest offering is the VBRun website, which offers code samples and articles for:

- completing common tasks in VB6, like creating a print preview
- integrating VB6 and VB.NET solutions (dubbed *VB Fusion*)
- converting VB6 code to VB.NET

Cross-platform and open-source development

The creation of open-source tools for VB.NET development have been slow compared to C#, although the Mono development platform provides an implementation of VB.NET-specific libraries and is working on a compiler, as well as the Windows Forms GUI library.

Hello World Example

The following is a very simple VB.Net program, a version of the classic "Hello world" example:

```
Public Class ExampleClass

        Public Shared Sub Main()
                System.Console.WriteLine("Hello, world!")
        End Sub

End Class
```

The effect is to write the text *Hello, world!* to the output console. Each line serves a specific purpose, as follows:

```
Public Class ExampleClass
```

This is a class definition. It is *public*, meaning objects in other projects can freely use this class. All the information between this and the following `End Class` describes this class.

```
Public Shared Sub Main()
```

This is the entry point where the program begins execution. It could be called from other code using the syntax `ExampleClass.Main()`. (The **Public Shared** portion is a subject for a slightly more advanced discussion.)

```
System.Console.WriteLine("Hello, world!")
```

This line performs the actual task of writing the output. *Console* is a system object, representing a command-line console where a program can input and output text. The program calls the *Console* method *WriteLine*, which causes the string passed to it to be displayed on the console.

APPENDIX E

MAGIN FORMULA

Margin = Sale – Cost –Other Charges + Service Fee

- END -