FOCT- A COLLABORATIVE SYSTEM FOR CASE TOOLS USERS

ZHU LIZHONG

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY, UNIVERSITY OF MALAYA
KUALA LUMPUR

JANUARY 2007

FOCT- A COLLABORATIVE SYSTEM FOR CASE TOOLS USERS

ZHU LIZHONG
WGA020030

DISSERTATION SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF COMPUTER SCIENCE

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY, UNIVERSITY OF MALAYA
KUALA LUMPUR

JANUARY 2007

Abstract

CASE tools have already been used for many years in IT organizations. Although CASE tools bring many benefits to organizations there are many issues during the implementation of CASE tools.

This research aims to investigate the utilization of CASE tools in both educational and industrial environments in Malaysia, followed by success factors and obstacles affecting the use of CASE tools and create a web-based system called the Forum of CASE Tools (FOCT). To meet these three objectives, the researcher conducted interviews with IT professionals from the industry; and conducted surveys through questionnaires among IT students and lecturers from colleges and universities. After analyzing the data collected from the target groups, the researcher is clearer on the status of CASE tools utilization in both the educational and industrial sectors. Some success factors and obstacles were identified from the target groups. Based on the data collected, the researcher developed a web-based system called Forum of CASE Tools system (FOCT). This allows the CASE tool users to share their knowledge and experience in using CASE tools and promote the utilization of CASE tools in Malaysia. By doing this research, a better understanding of using CASE tools in Malaysia has been achieved. More importantly, this research would be able to help educators train knowledge workers for this information age.

Acknowledgement

A Chinese saying goes, "If people help you, you should do more for them". During the process of doing my dissertation, many people have helped me out. Without their assistance, I would not have been able to complete the research. I am very glad to have an opportunity to say 'thank you' to those who have helped me in my study life in Malaysia.

My first praise goes to my supervisor, Dr. Zaitun Abu Bakar, for giving significant help in my dissertation. As a foreign student, I faced a lot of difficulties and discouragement while doing this dissertation. With great patience she always encouraged me and advised me. Without her help, my efforts would have come to nothing.

I would like to thank Kelvin, Siva N Kumaram, Tengku Omar, Cheah Kam Yen, Tam and Professor. Selamat, M H. They provided valuable information and support during my interviews and surveys. I was fortunate to have friends from DUMC church and Calvary church. They have been praying for my study life. They gave me help directly or indirectly with God's blessing.

Finally, the most important contributors to this dissertation were my parents and my wife, Yangmin. They really sacrificed a lot during my term of study, and supported me financially and spiritually. Their support was a moral booster.

Table of Contents

Table of Contents (continued)

List of Figures

List of Figures (continued)

List of Figures (continued)

List of Tables

Chapter One – Introduction

1.1 Introduction

The development of software is a complex process. Managing the process is a challenge to software developers. It not only requires that software developers have much experience in software design, but also requires them to master some tools to improve the quality of software. In order to be able to achieve a better quality of software and higher productivity, Computer Aided Software Engineering tools (CASE tools) should be used by software developers to help them in software development.

This chapter gives an overview of the research on the use of CASE tools carried out by the researcher. Firstly, the definition and history of CASE tools are discussed. This is followed by a discussion on the advantages and disadvantages of using CASE tools as well as the use of these tools in Malaysia. The researcher then discusses the research objectives, scope of the research, methodologies employed, the expected outcomes, and the research limitations. Lastly, the organization of the thesis is listed.

1.2 Definition of Computer Aided Software Engineering

Computer Aided Software Engineering (CASE) was introduced in the 70's. Some experts have defined Computer Aided Software Engineering as the automation of systems development. CASE gives software developers help in building software system by using tools, techniques and methodologies in system development. Traditionally, researchers have classified CASE tools into four categories (David, 1997). These are:

1) Upper CASE

   Upper CASE tools are used to handle high-level designs such as object and database modeling and also check limited repository information. Upper CASE tools are used in the methodology level.

2) Lower CASE

   Lower CASE tools can be used in application development such as code generation and interface design.

3) Integrated CASE

   Integrated CASE tools are a combination of the capabilities of Upper and Lower CASE tools and also have additional features such as database design.

4) Meta CASE

   Meta CASE tools allow the system developers to innovate customized methodologies into particular CASE tools. For example, if a developer finds that there are no suitable methodologies for his or her project, he or she could develop a new methodology using Meta CASE tools.

1.3 History of CASE Tools

Initially, in the 70's, the growing size and the complexity of software systems created the need for a tool to help developers break large and complex software into smaller and simpler modules and convert the software from an abstract level into a design level. During that time, many of the graphical notations available were used to help software developers model problems and solutions (Alfred, 1991). The CASE tools available at that time provided some basic functions such as diagram editions and system checking.

In the 80's, CASE tools became more powerful as many researchers added more functions such as documentation generation, diagram, consistency checking, and code generation to them.

In the last ten years, due to significant changes in information technology, the CASE tools had additional new features added to them. These included Object-Oriented Techniques, Component-Based design, 4th generation language support and Reverse-Engineering.

At present, many CASE tools are available in the market. Examples are: RationalRose, SystemArchitect, Erwin and DOORS. These tools provide many functions such as: diagramming, repositories, interface design, schema object generation and reverse engineering.

1.4 Advantages and Disadvantages of Utilizing CASE Tools

Many software companies use CASE tools today. The CASE tools are playing an essential role in system development, especially in today's analysis and design stages of system development. Generally, CASE tools help developers make their jobs easier, faster and better. Some benefits of using CASE tools are:

1) Faster coding (Joan, 2003)

    CASE tools can generate source code automatically, and programmers do not have to write the code manually. Therefore the schedule of software development would become shorter.

2) More flexible ways for modification and maintenance

CASE tools have the ability to allow developers to maintain and modify the code. For example, if a developer wants to change an attribute of a class, he or she needs to only make the change once and the system automatically changes other factors related to this change.

3) Better communication between users and developers

CASE tools provide many notations to present different meanings, which are standardized and easy for users and developers to understand. Some standards such as IEEE are used to keep diagrams and the styles of naming are consistent. This benefit of CASE tools can reduce miscommunication between users and developers.

4) Better quality of software using reuse components

CASE tools also support re-usability in software design. In software design, some modules and components (e.g. some class diagrams and entities) can be used again for other projects. If later, developers want to design a new system that makes use of similar classes in the previous project, the developers can use these components without any changes. Developers need not worry about problems in this component as these components have been tested in the previous project.

5) Automatic documentation

Documentation is very helpful to both end users and developers. CASE tools can automatically generate documentations for end users and developers.

6) Powerful features for consistency checking

   The consistency-checking features can ensure that developers are able to make changes to parts of the software without causing any conflict with others parts. Thus, these consistency-checking features reduce the risk of change.

7) Teamwork support

   Nowadays, most software is developed by teams instead of individuals. CASE tools provide the capability of supporting software development by teams. CASE tools allow members of a development team to share designs and documentations in a software development project.

However, CASE tools are not perfect. Many IT companies have encountered problems in using CASE tools. Some problems resulted from the cost of installing CASE tools while others appeared because of the complexity of CASE tools. When using CASE tools, some issues to be considered are:

1) Education and training cost of using CASE tools

   In order to ensure that the users know how to use CASE tools properly, companies have to spend much money in training and education. In many cases, this is a big cost to the companies.

2) Complexity of CASE tools

   Some CASE tools are very complex. Users cannot master them easily.

3) Ongoing usage fees

   Once companies have installed CASE tools, they likely continue to upgrade the version of CASE tools, and hence the extra money will be spent.

4) Simplistic code

CASE tools can generate code for programmers. But the codes are very simple. It normally cannot be used without any modifications. Programmers have to add more code to make sure the program runs correctly. Therefore when CASE tools are used, programmers sometimes spend a lot of time modifying the code.

5) Inadequate integration with other tools (Alan, 2002)

Integration of CASE tools is very important to ensure that they can be used in other projects. Unfortunately, some CASE tools cannot integrate with others.

6) Personal skills

Using CASE tools requires users to have strong skills and knowledge in methodologies and notations. If the users are new to such things, they will find it quite difficult to use CASE tools.

1.5 Usage of CASE Tools in Industrial and Educational Sectors in Malaysia

CASE tools are very powerful tools to help system developers in analysis, design and code and so on. According to a report done by Mohd (1989), in Malaysia, there has also been a significant advance in the development of software using CASE tools. The researcher believes that there will be an increasing trend in the use of CASE tools in both the industrial and educational fields.

1) Usage of CASE tools in the industrial sector

In order to speed up software development, companies in Malaysia are using CASE tools to improve software development. Selamat (1994) said that "Malaysia, a member of ASEAN, is experiencing a dramatic growth in the IT industry. Many enterprises are investing substantial resources into CASE technology to fulfill the growing demand of IT applications". Some other companies use complex CASE

tools in analysis, design, code generation and documentation generation. Companies use the CASE tools based on different requirements such as business reasons and staff skills.

2) Usage of CASE tools in the educational sector

Software companies benefit from educators because the colleges and universities provide qualified new staff for them. Basically, before entering the companies, most IT students have learned some fundamental knowledge about software development. In order to ensure that IT students know about CASE tools, colleges and universities must teach and deliver new knowledge to IT students accordingly. These universities use CASE tools in their teaching areas that include System Development, Software Engineering and Principles of Database.

1.6 Problem Statement

More and more technologies are becoming available to support software development. To remain competitive, Software companies must be able to use these new technologies. Barbara (2004) pointed out that choosing a CASE tool for learning and teaching systems is an issue within an academic institution. CASE tools are becoming more important for software designers. In implementing CASE tools, both companies and educators face the following problems:

1) Companies that use CASE tools in software development sometimes do not have staff with the required skills (Duska, 1994).

2) CASE tools are becoming so powerful and complex that it is increasingly becoming difficult for developers and student to use (Paul, 1999).

3) Cost of using CASE tools is very expensive (Premkumar, 1995). Therefore, universities and colleges cannot afford to use CASE tools to teach their students.

4) There is limited time for teaching the use of CASE tools (Barbee, 1990). Therefore, students can only learn basic features of CASE tools.

To what extent are the above factors affecting universities and colleges in Malaysia and what can be done to alleviate the problems?

1.7 Research Objectives

Most of us agree that sometimes, what employees actually use in companies is different from what they learn in universities. A similar relationship can be made between software companies and IT students. The usage of CASE tools is a good example to be studied by the researcher. The research attempts to clarify whether there is any mismatch between required skills of software companies and knowledge produced by educators attempt to. The researcher will explore three objectives in this study. The research objectives are:

1) To investigate what CASE tools are taught in colleges and universities and compare with that required from IT organizations.

2) To identify the factors and obstacles in implementing CASE tools in educational and industrial sectors.

3) To provide a collaborative CASE tools system that can help to promote the usage of CASE tools.

To cover the three objectives, the researcher developed some research questions, which were used to help the researcher to meet the objectives accordingly.

1) Utilization of CASE tools in some companies and some universities

   Do companies use CASE tools in software development?

   Do universities use CASE tools for teaching purposes?

   Why do the companies and universities use CASE tools?

   What are the CASE tools used in both universities and companies?

8

2) Factors and obstacles that affect the usage of CASE tools in companies and universities

   Is there a business consideration to use CASE tools in companies and universities?

   What resources are lacking in trying to learn how to use CASE tools by employees and students?

   Are the CASE tools too difficult or too complicated to use?

   Is the time for training and teaching enough to master the knowledge of using the CASE tools?

3) Comparison between students' knowledge and a company's usage of CASE tools

   What are the necessary requirements for using CASE tools?

   What do students know about using CASE tools?

   Is there any mismatch between job requirements and student's skills?

4) System development of Forum of CASE Tools (FOCT) System

   Does FOCT provide a useful and helpful platform for promoting the usage of CASE tools?

1.8 Significance of the Research

The current problems in the educational and industrial sectors have been described in the previous section. After studying other similar and related research work, the researcher is claiming that this research work is significant for the following reasons:

1) Clarification of problems in educational and industrial sectors

   The research problems have been described in the previous sections. The main problems have already existed for a few years. However, no

9

documented research work has been carried out to investigate the problem situation. Therefore such a study will benefit both the educators and IT community to understand the problems and the current scenario. To understand the problems better, the researcher will conduct surveys in the educational and industrial sectors. From the data gathered, an analysis can be conducted and recommendations formulated. This finding will contribute towards deriving at a solution to the problems.

2) More statistical findings

Although other researchers have conducted research in the usage of CASE tools in Malaysia, they have limited their studies to the industrial sector only. However, this study covers both educational and IT industrial sectors with different views. The data collected will be analyzed and presented in different formats. Therefore richer and more meaningful findings can be explored and generated.

3) Improvement of students skill

This research project is focused on investigating the usage of CASE tools in universities and colleges. The result of this research is more accurate and reliable for IT educators and IT professionals compared to other similar research. Therefore, the educators are able to identify the problems that exist in the colleges and universities and minimizing the gap between educational and industrial sector. Employers will be able to train new employees to improve the software skill with minimum cost.

These significant impacts encourage the researcher to carry out the study. This research will provide a firm basis for future and more detailed study on the usage of CASE

tools. The researcher hopes that the finding of the study will be beneficial to both educators and IT professionals to improve the productivity of employees in Malaysia.

1.9 The Scope of the Research

In order to closely focus on the research objectives, the researcher narrows the scope of this topic because of some factors including time, cost and researcher's limitations. The scope of the research was defined as follows:

1) The research's target area is Malaysia.

   The researcher will focus on the research area in Malaysia only because as a student, it is impossible to go overseas to conduct interviews and surveys.

2) Target educators are from some universities and colleges in Malaysia.

   The researcher will focus the study on colleges and universities in Malaysia, because these colleges and universities offer many IT and computer subjects that involve the use of CASE tools.

3) Target students are those students who are studying in colleges and universities. The researcher will select fifty IT students as a target group and administer questionnaires to them. From the completed questionnaires, the researcher will be able to collect data and information, which are the primary data for this research because these students can give some feedback on the usage of CASE tools in their study programs.

4) Target lecturers are in the universities and colleges. The researcher randomly selects fifty lecturers as respondents. The lecturers would be able to produce valuable feedback because they are using CASE tools in their teaching.

5) Target companies are IT organizations, which are using CASE tools to develop software in Malaysia. Twenty-five companies will be invited for this research. From interviewing the software companies, the researcher will be able to know what the real requirements are from IT companies and which areas are important for students.

1.10 Research Design
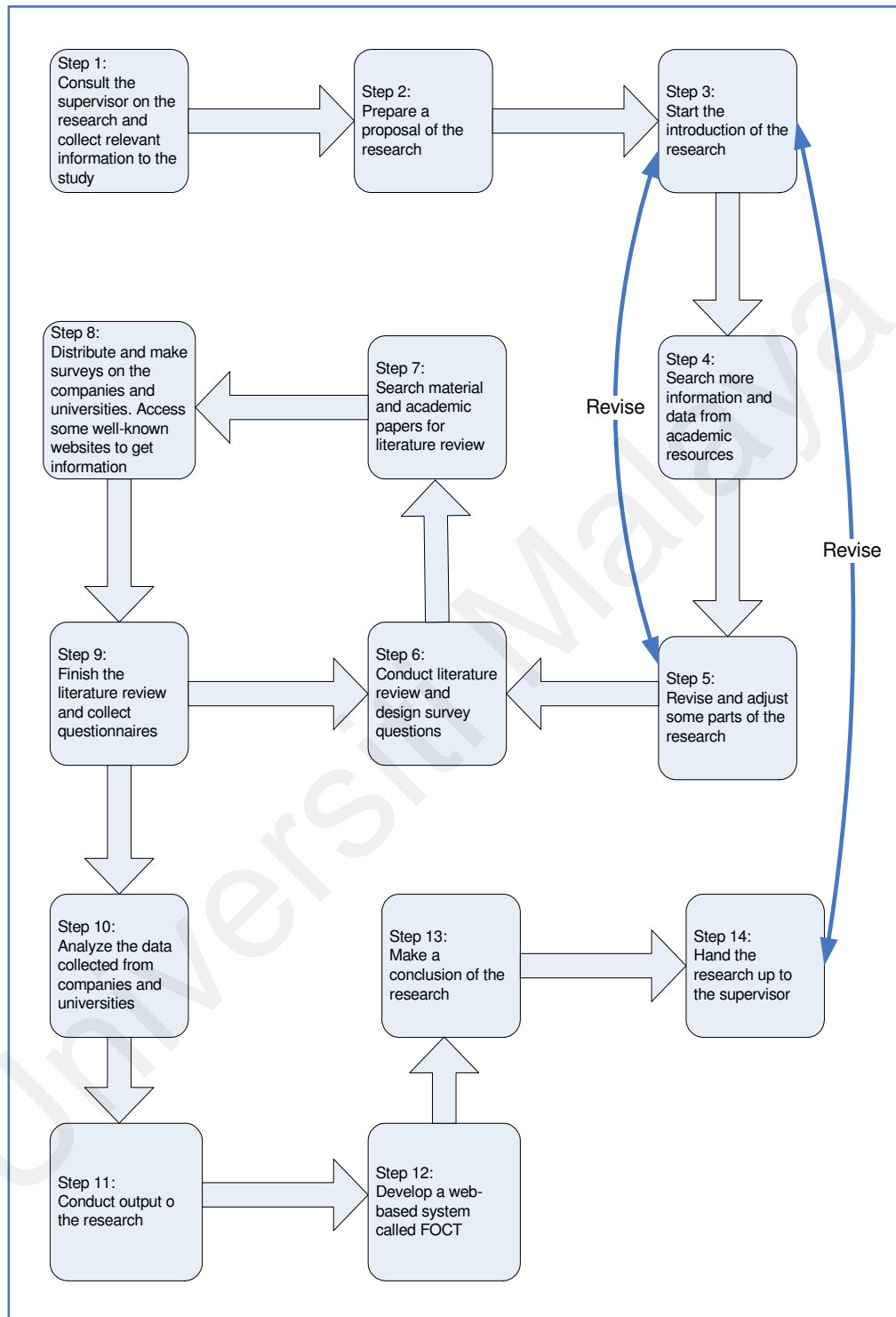
Figure 1.1: Research Design

To complete this research in an effective way, the researcher has come with a research design as shown in Figure 1.1. There are fourteen steps to complete this research.

1.11 Research Methodologies

To ensure that the researcher obtains the appropriate data from this research and achieves accurate results, the researcher must choose the right methodologies. The research instruments used in the collection of data are shown below:

1) Questionnaires

The researcher plans to distribute questionnaires to students and lecturers in some colleges and universities in Malaysia. Questionnaires are an effective way to collect data from a large number of people. The researcher plans to distribute at least two hundred questionnaires among the students. The researcher further needs to send out questionnaires to lecturers who are teaching some subjects that use CASE tools.

2) Interviews

The researcher will conduct interviews with IT organizations. Interviewing some IT companies or IT departments is a main research approach to get the primary data about using CASE tools from IT organizations. By interviewing IT people, the researcher will be able to obtain details about how CASE tools are used in their companies. A similar approach applied in a CASE tools study has been used by Selamat (1996). This researcher made surveys and interviews in IT Malaysian companies.

3) Literature analysis which include the following sources:

a) Job advertisements

Job advertisements are an economical and effective way to collect secondary data. From the job advertisements, the researcher will be able to find out what

14

skills are needed by IT graduates. Generally, the employee will list out skills that the candidates must have, such as knowledge on Rational Rose and UML. Job advertisements are published in newspapers and some well-known websites such as www.JobStreet.com.my in Malaysia. The researcher will frequently check these resources as references.

b) Journals and magazines

Journals and magazines are very important sources to find helpful and reliable secondary information. University libraries subscribe many journals related to CASE tools, such as CAM and IEEE. The researcher will search Universiti Malaya (UM) library or other research centers to get these journals and magazines. The papers obtained from the journals and magazines give the researcher the latest information and issues about CASE tools. After reading and analyzing these papers, the researcher will be able to get accurate and very clear knowledge about the study of CASE tools.

c) Academic reference books

Reference books are a main resource to the researcher. Reference books give more complete and accurate definition about this study. The researcher will select some references as secondary information.

d) Websites

Some professional and well-known websites such as IBM's website issue many white papers and provide forums for researchers to explore more knowledge. The researcher will access these websites to search for helpful information.

1.12 Expected Research Outcomes

The expected research outcomes are as follows:

1. From interviewing IT organizations, a status report on the utilization of CASE tools in the industrial sector will be obtained.

2. From surveying students and lecturers from colleges and universities, a report on the usage of CASE tools in the educational sector will be reported.

3. The main success factors and obstacles will be identified.

4. From the above findings, the Forum of CASE Tools (FOCT) system will be developed. This system will help in promoting the utilization of CASE tools.

1.13 Limitations of the Research

The main limitations are the sample size of companies, students in colleges and universities and language.

o *Size of companies*

The researcher plans to interview about twenty-five companies in Malaysia because of reasons such as time frame and cost. The researcher would have liked to spend about three or four months to conduct interviews with many IT organizations.

o *Size of survey groups*

The researcher plans to distribute about two hundred sets of questionnaires among some private and government universities in Kuala Lumpur and Selangor such as KDU, APPIT, UM, UTM and UPM. The researcher is not able to distribute the questionnaires to USM, due to distance and cost. Maybe, a web-based survey is an alternative way to collect data from USM. But the time and reliability of data are concerns.

16

o *Language*

Language is another limitation for the researcher. For example, the researcher wants to access some websites of universities to get information about IT subjects, but many websites are presented in Bahasa Malaysia, which is the Malaysian official language. It is difficult for the researcher to understand. Very often, the researcher has to ask friends to interpret the content, which is presented in Bahasa Malaysia.

1.14 Organization of the Thesis

According to the guide on dissertations and thesis in UM, the researcher will arrange the order of the research under the following headings:

1. Preface

   This chapter contains the title page, abstract, acknowledgements, table of contents, list of figures, and list of symbols and abbreviations.

2. Introduction

   This chapter contains the introduction to the issues of CASE tools, brief history, research objectives, research questions, methodology, scope of the research, research design, limitations of the research and organization of the thesis.

3. Literature Review

   This chapter will explore the relevant academic issues of using CASE tools done before. Some issues will be studied such as the support of CASE tools to software engineering and methodologies and problems in using CASE tools.

4. Survey and Data Analysis

   From findings, some figures and tables will be generated.

5. System Development of Forum of CASE Tools (FOCT) system

System development of FOCT is described in this chapter. All necessary information and documentation will be described in detail.

6. Conclusion

This chapter interprets the tables and figures derived from data analysis, and also compares data with previous expected output and literature review.

7. References

8. Appendix

1.15 Conclusion

Generally, from the brief introduction of CASE tools, the researcher has presented a basic understanding of utilization of CASE tools. The researcher discussed the history of CASE tools, advantages and disadvantages of CASE tools' utilization, factors and obstacles in using CASE tools in both the industrial and educational environment. Furthermore, the researcher described the research methodologies applied in this research, the expected research outcomes, limitations of the research and lastly, the researcher set the organization of thesis according to the thesis format.

Chapter Two - Literature Review

2.1 Introduction

Software development is a complex process. During the development process, there are many tasks to be done such as identifying users' requirements and developing applications. Computer-Aided Software Engineering (CASE) tools are able to give developers much help in managing these complex tasks more effectively and efficiently in software development. As such more and more commercial companies require software developers to know how to use CASE tools. In order to fulfill the requirements of the job market, colleges and universities have the responsibilities to produce graduates with the right skills. To this end, some colleges and universities offer courses that use CASE tools so that students gain knowledge and awareness about using CASE tools.

In reviewing the literature on CASE tools, it is evident that the use of CASE tools presents some issues in both the industrial and the educational sectors. The question is: "Does this apply to the Malaysian sectors and if so, what are the issues"? To answer the above question, and to improve the effectiveness of the use of CASE tools in both the industrial and educational sectors in Malaysia, it is necessary to conduct a deeper research on these areas. In this chapter, there are several main points to be discussed. They are:

1) Introduction to CASE tools

2) Current status of the utilization of CASE tools in Malaysia and other countries

3) Importance of using CASE tools for both industrial and educational sectors

4) Issues arising from the use of CASE tools in the industrial and educational sectors

5) Similar Web-Sites for FOCT System Development

2.2 Introduction to CASE tools

In this section, the main points regarding CASE tools will be explained.

2.2.1 Definition of CASE Tools

CASE tools stands for Computer-Aided Software Engineering tools. It refers to automated software development. Carma (1989) defined that "The CASE technology is a combination of software tools and methodology". However, another expert gave a definition of CASE tools as, "Computer-Aided Software Engineering (CASE) encompasses a collection of automated tools and methods that assist software engineering in the phases of the software development life cycle" (Richard, 1997). According to Carnegie Mellon Software Engineering Institute, a CASE tool is a computer-based product aimed at supporting one or more software engineering activities within a software development process (CMU).

2.2.2 Evolution of CASE Tools

CASE tools have been used for many years in software development organizations. Today, the market for CASE tools is growing. A survey of the CASE tool market showed that the annual worldwide market for CASE tools was USD 4.8 billion in 1990 and grew to approximately USD 12.11 billion in 1995 (Stan, 1998). From the time CASE tools were introduced, they have been enhanced with many new features. Carma (1989) gave a brief history of CASE tools before 1990s as shown in Table 2.1.

Table 2.1: CASE Technology Evolution

Early 1980s	Mid-1980s	Late 1980s	Early 1990s
Computer-aided documentation Computer-aided diagramming Analyzing an design tools	Automatic Design Analysis and Checking Automated system information reposting	Automatic code generation from design spec Linking design automation and program automation	Intelligent Methodology driver Habitable user interface Reusability as development methodology

Source (Carma, 1989)

From the above table, it can be seen that there have been four main phases in the development of CASE tools during the period from the early 80's to the early 90's. After 30 years of development, CASE tools have become more powerful and more functional. These CASE tools were developed to support existing software development methodologies. Alan (1991) defined a clear relationship between methodology and tools. Figure 2.1 shows that methodologies became more advanced as a guide in software development.
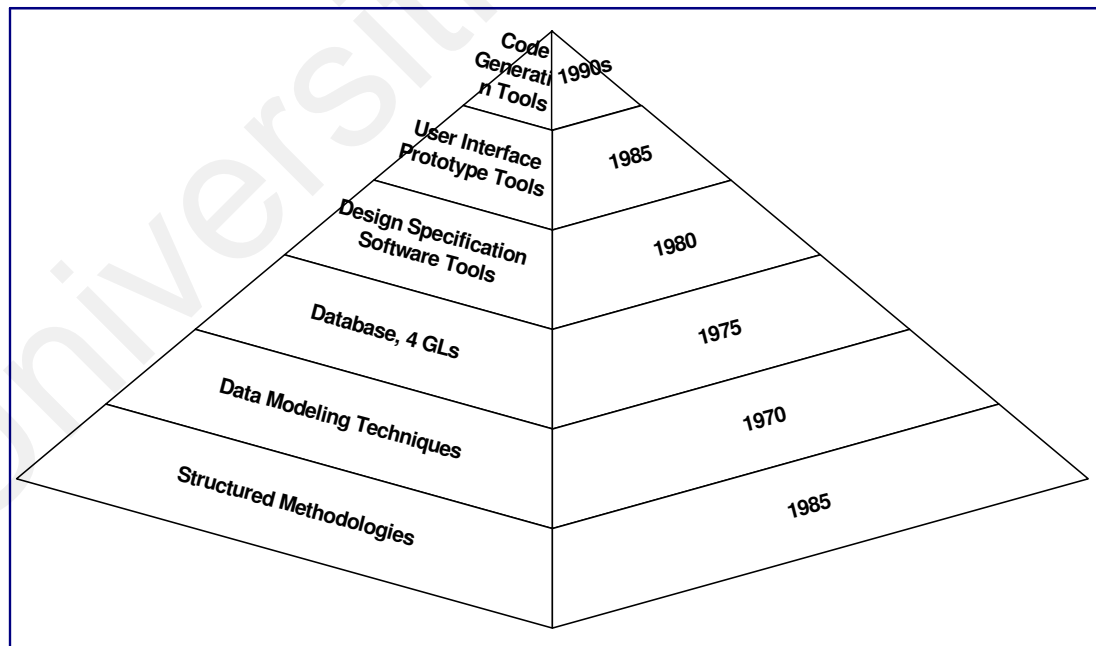
Figure 2.1: Relationships between Methodologies and Tools

Source (Alan, 1991)

2.2.3 Features of CASE Tools

CASE tools provide many features for software development. Ian (2001) provides an alternative classification of CASE tools that can be used in different stages of software development. This is illustrated in Table 2.2.

Table 2.2: Alternative Classification of CASE tools

	Specification	Design	Implementation	Verification and validation
Re-engineering tools			●	
Testing tools			●	
Design tools			●	●
Programming Analysis tools			●	●
Language Process tools		●	●	●
Methods support tools	●	●	●	
Prototyping tools	●			
Configuration tools		●	●	●
Change management	●	●	●	●
Documentation tools	●	●	●	●
Editing tools	●	●	●	●
Planning tools	●	●	●	●

Source (Ian, 2001)

We can see from Table 2.2 that CASE tools can be utilized in many phases of system development. In general, CASE tools include the following categories and features (Richard, 1997):

1) Editing tools. The editing tools have two main features: textual editors and graphical editors.

2) Programming tools. These tools can be used to generate code and execute compilation.

3) Verification and validation tools. These tools are used to support program verification and validation. These tools aim to ensure that the final products meet the user requirements.

4) Configuration management. These tools include version management, change control, item identification and library management. During the

software development process, there are bound to be many changes due to changes in user requirements or additional requests. Therefore software requirements have to change frequently. These tools can be implemented to control the changes and ensure the changes are under control.

5) Metrics and measurement. These tools aim to analyze and monitor the source code.

6) Project management. These tools include features such as project planning, cost estimation and communication tools.

7) Miscellaneous. These tools are actually integrated with some other software such as Excel and accounting systems.

2.2.4 Related Topics

CASE tools cannot exist without other support or influence such as methodologies. Some necessary topics will be discussed because these topics are associated with CASE tools. Simon (2002) defined "A methodology as a collection of many components. Typically, every methodology has procedures, techniques, tools and documentation aids that are intended to help a system developer in his or her efforts to develop an information system". Methodologies have been used for many years. These methodologies give a lot of help in system development and overcome many problems for developers. According to Simon (2002), the advantages of using methodologies are:

1) Produce a higher quality product in terms of documentation standards, acceptability, maintainability and consistency of software.

2) Ensure the user requirements are fully met.

3) Help project managers handle the cost of project well.

4) Provide a better communication between the developers and end users.

5) Standardize the development process.

Besides methodologies, Software Development Life Cycle (SDLC) and Object-Oriented Design (OOD) will be discussed. Software Development Life Cycle (SDLC) is an iterative cycle process that developers should follow to accomplish the objectives in each of the development stages. Ian (2001) suggested a well-known model called the Waterfall model. Generally, there are five steps in this model and there are many specific tasks in each step. Specific features of CASE tools are able to fit into these stages.

Object-Oriented Design (OOD) is used to define problems and processes in a way that allow developers to separate the entire system into the some modules and components. It also promotes reusability in software design. Some benefits are defined by Pierre (1997):

1) More stable models. These models can reflect real entities clearly.

2) Iterative construction. In OOD, some components have weak relationship with each other. Therefore, the developers can modify the relationship easily and make it flexible without affecting other models.

3) Gives more reusability in design. Since, some component designs are object-based, similar objects can be used in other designs or systems.

CASE tools also feature the ability to cater for OOD such as supporting Unified Modeling Language (UML). Grady (2001) explained that the purposes of UML are to provide visualizing, specifying, constructing and documenting object-oriented system.

2.3 Current Status of Utilization of CASE Tools in Malaysia and Other Countries

CASE tools are not really new to Malaysian companies and other overseas companies. The situation of using CASE tools is optimistic in some countries. But the companies have some issues such as the lack of adequate training and the cost of CASE tools. In Finland, Jari (1999) conducted two surveys during 1993 and 1996 and in both surveys he found that "CASE tools brought improvements primarily in working procedures and in the use of standards, thereby enhancing the growth of the process and products". In Singapore, Danny (1998) conducted a research on utilization of CASE tools in 1997. In this research, only 29.6% of Singaporean organizations used CASE tools. Different countries show different rates of utilization of CASE tools. But most of the researchers agree that CASE tools brought benefits to organizations. In the Netherlands, Rob (1993) conducted a survey. In this research, only 16% of the responding organizations were of the opinion that CASE-tools are of limited significance. Twenty-two percent considered the significance to be 'reasonable' and 47% thought that the importance was 'significant'. Only 15% considered CASE-tools to be essential for systems development. In New Zealand, Stepthen (1993) found that the utilization of CASE tools had increased significantly over last five years.

In Malaysia, Selamat (1996) said that, "A recent figure released by Computer Systems Advisers (M) Sdn Bhd put the sales of the CASE tool at RM1.5 million (USD1= RM2.6) in 1991, a 60 percent increase from the previous year". The data was obtained in 1991. Selamat also conducted a survey in 1996 in Malaysia. Table 2.3 shows that only 5% of organizations used CASE tools for more than five years in Malaysia, and ten percent of applications were developed with CASE tools. They had intentions to use CASE tools in software development. There should be an increase in the utilization of CASE tools since then because in the last decade, there was a significant increase in software development.

Table 2.3: Profile of CASE Tools

	Number	Percentage
Types of CASE Software		
Upper	**33**	**82**
Lower	**7**	**18**
Total	**40**	**100**
Use of Upper CASE		
Application Development	**17**	**51**
Application in progress	**4**	**13**
Documentation Purposes	**3**	**9**
Abandoned	**9**	**27**
Total	**33**	**100**
Use of Lower CASE		
Application Development	**6**	**86**
Application in progress	**0**	**0**
Documentation Purposes	**0**	**0**
Abandoned	**1**	**14**
Total	**7**	**100**
Years of CASE Adoption		
Less than 1 year	**21**	**53**
Between 1 to 3 years	**15**	**37**
Between 4 to 5 years	**2**	**5**
Over 5 years	**2**	**<u>5</u>**
Number of applications developed with CASE		
None	**13**	**33**
One in progress	**4**	**10**
One only	**11**	**27**
Between 2 to 5	**8**	**20**
Over 5	**4**	**<u>10</u>**
Verified from 0 to 3, where 0 means no effect, 1 means slight effect, 2 means moderate effect, while 3 implied significant effect.		

Source (Selamat, 1996)

In 1997, Ruhana (1997) conducted a research on the usage of CASE tools and techniques in Malaysia. All respondents in this research agreed that CASE tools have been extensively used in supporting various activities in information system application development. Table 2.4 indicates that more than 70 percent usage goes to diagramming, screen/report painters, documentation, database and file generation, testing, estimating, scheduling and task assignment/tracking. The two highest usage ratings of CASE tools are for documentation and scheduling. From the studies conducted by Selamat in 1996 and

made by Ruhana in 1997, it can be seen that the usage of CASE tools has increased for different purposes in system development in Malaysia.

Table 2.4:

Usage Rating of CASE tools in Supporting IS Development Activities

Activities	Usage Rating (%)
Analysis/Design	
Diagramming	74.6
Screen/Report Painters	74.66
Analyses	57.54
Documentation	76.71
Simulators	50.00
Specification Languages	50.68
System Information Management	
Repository	60.27
Info Management System	65.07
Implementation	
Code generation	66.44
Database/File Generation	73.97
Testing	73.29
Maintenance	
Reformatting	62.33
Restructuring	63.70
Program Analysis	67.81
Project Management	
Estimating	70.55
Scheduling	76.03
Task Assignment/Tracking	71.23
Methodology Enforcers	58.22

Source (Ruhana, 1997)

2.4 Importance of Using CASE Tools in Industrial and Educational Sectors

Software development became more and more complex as the user requirements increased significantly. Having a CASE tool to support software development is very important for developers. In universities, CASE tools are becoming a part of the core courses for IT students. In general terms, CASE tools bring several benefits to both industrial and educational sectors.

2.4.1 Benefits to Industrial Sector

CASE tools bring many benefits to software developers. Basically, these benefits are productivity improvement, better quality software, cost saving of software development, re-usability and OOD support.

1. Productivity improvement

Generally, there is a common expectation that CASE tools ensure high productivity. Productivity can be defined as the ratio of quality work products completed per unit of time (Selamat, 1996). High productivity can be obtained through faster coding. Selamat (1996) found that productivity indicated in Figure 2.2 increased dramatically during the coding and testing phases compared to early stages of the SDLC. He further explained that this was because of automatic generation of source code when using lower-CASE tools.
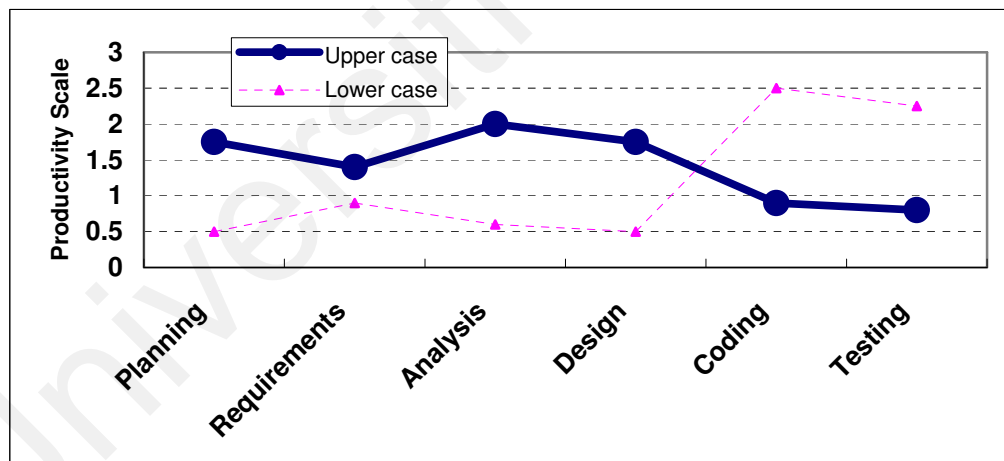
Figure 2.2: Evaluation of CASE Tools

Source (Selamat, 1996)

Productivity can also be achieved by sharing information between developers and end users. Software development is definitely not an individual task. Many people are needed to be involved to handle the different tasks. To ensure that the tasks can be done successfully, CASE tools may be a good choice to developers. For example,

28

CASE tools allow the team members to access user requirements and changes the requirements. If the developers do not use the CASE tools, some changes cannot be under the control of developers because during development period, there will be many changes. If the developers cannot manage these changes properly, some developers will not know which change is a new one, and which one the developers should follow. These problems will reduce the productivity of developers. The CASE tools can help developers minimize and overcome these problems. Elliot (1988) found that after using CASE tools, organizations experienced various degrees of improvement in productivity. His report said that on average, these organizations achieved 30% to 40% improvement in productivity in the analysis and design of life cycle.

2. Better quality software

Before discussing quality of software, it is necessary to define software quality. Richard (1997) defined software quality in terms of five aspects. They are as follows:

1) The totality of features and characteristics of a software product that bears on its ability to satisfy given needs.

2) The degree to which software processes a desired combination of attributes.

3) The degree to which a customer or user perceives that software meets his or her composite expectations.

4) The composite characteristics of software that determine the degree to which the software in use will meet the expectations of customers.

5) Attributes of software that affect its perceived value, for example, correctness, reliability.

Ian (2001) agreed that higher quality software should have some attributes such as reliability and security. To get better quality software, developers must understand the user needs well. Usually, one project has many user requirements. Managing user requirements correctly is a first step to ensuring higher quality software. CASE tools offer such features to help developers manage the user requirements in a better way. One of the features that can be used is consistency checking. Carma (1989) pointed out that one of the benefits of CASE tools is improvement of software quality and it can be obtained through automated checking. Traceability of using CASE tools is another feature that minimizes the rate of error occurring. In one project, the requirements or design can change very often. Subsequently, the possibility of errors occurring will increase. If there are so many errors, the software quality will not be as good as expected. Tracing errors by using CASE tools is a good way to control the quality of software.

3. Cost saving of software development

CASE tools provide many features to speed up software development. Two of the features are code generation and documentation generation. These features are able to reduce the time on writing code and documentation. CASE tools have an ability to standardize the format of documentation. By doing so, the time spent on writing documentation will be decreased. Therefore, the cost of software development will be reduced.

4. Re-usability

Re-usability is a big advantage of Object-Oriented Programming (OOP). CASE tools promote re-usability in software development. One of the examples of re-use is code re-use. CASE tools have an ability to store the previous code into a CASE repository. Later, if the developers want to develop a new project, which is very similar with the

previous one, the developers can use the previous code for the new project. Another example of re-use is the re-use of user requirements. Sometimes, due to the same business process, some user requirements can be used again in other projects. Such user requirements will benefit the developers for future use. Developers can refer to the previous user requirements as references and get an understanding of the project quickly.

5. OOD support

Currently, more and more IT organizations prefer to use OOD in their projects. The definition of OOD given by Ian (2001) as "Object-Oriented Design is a means of designing software so that the fundamental component in the design represent objects with their own private state and operation rather than functions". One of ways of presenting objects is to use Unified Model Language (UML). Grady (2001) gave a clear definition of UML as "The Unified Modeling Language (UML) is a standard language for writing software blueprints. The UML may be used to visualize, specify, construct and document the artifacts of a software-intensive system". CASE tools support UML and OOD. Objects can be presented by using UML diagrams and relationship between objects or entities can be described with UML notations.

2.4.2 Benefits to Educational Sector

CASE tools have been accepted in most IT organizations. As CASE tools bring many benefits to them, knowing CASE tools is necessary to the employees in IT organizations. In order to match this requirement, colleges and universities have to teach CASE tools in IT or computer courses. Mali (2005) said that, "some of the important benefits of using CASE tools in teaching Systems Analysis and design courses are: its role as a pedagogical instrument in teaching and learning a systems development methodology and to provide

31

support in teaching tools and techniques which are popular in the workplace and putting the students on the forefront of new and leading edge technologies".

Mary (1991) suggested "Based on the results of this research, we can recommend introduction of CASE not only into a software engineering course, but also into other courses in an Information Systems or Computer Science curriculum. Since less human and computer resources were used during the coding phase of the pretty printer, incorporation of CASE throughout the curriculum should have the same effect in other courses that require either coding or system development. The students could do more realistic course projects in the same amount of time. Additionally, the students are exposed to leading edge technologies; the classroom setting becomes more like the industrial environment".

Barbee (1990) said that "Student attitudes were positive and the quality of their project work appeared to be somewhat better during the course". Some educators believe that benefits can be obtained from teaching CASE tools. Some of benefits are:

1) Having a better understanding in system development because CASE tools support many methodologies. Students can experience these methodologies by using CASE tools. For example, the students can use CASE tools to conduct user requirements analysis. Further, the students can use the CASE tools to define objects, entities and their relationships based on the requirements collected previously. By doing so, the students will have a deeper understanding in more specific tasks.

2) Learn more about OOD by using CASE tools. OOD is a new trend of software development. CASE tools support this. Using CASE tools will enable students understand OOD better because CASE tools have many features that allow the students to design a system based on OOD methodology.

3) Be competitive in the job market.

In the current job market, there is a new requirement for new employees. Usually, the organizations request candidates to know more about tools such as RationalRose. Barbara (2005) survey in UK indicated that CASE tool skills are important knowledge in software courses. If the students are exposed to CASE tools, then the students will be more competitive in the job market.

2.5 Issues of Using CASE Tools in Industrial and Educational Sectors

2.5.1 Issues in Industrial Sector

There are some issues in the use of CASE tools. Duska (1994) reported that 80% of UK firms who tried CASE tools have been disappointed by the results. These problems did not only occur in the UK, but Malaysian IT organizations too have some problems in using CASE tools (Selamat, 1994). These issues are concerned with people, cost and implementation.

1. Lack of Training for People

CASE tools are designed for end users or developers. Giving training to developers is the best way to get the skills. Alan (1991) suggested that project members should attend training courses prior to the beginning of project's launching. Developers should have opportunities to experience and learn how to use CASE tools. Selamat (1994) conducted a research on using CASE tools in Malaysia. Figure 2.3 shows that only 10% of the organizations made some effort to introduce CASE tools before they are being used.
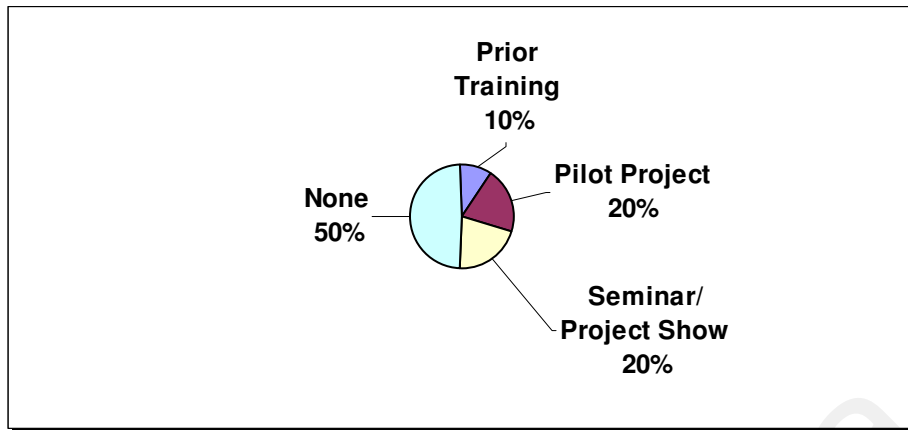
Figure 2.3: Effort to Introduce CASE Tools

Source (Selamat, 1994)

If developers can use CASE tools properly, then the productivity of software development should increase. However, if the developers used CASE tools inappropriately the CASE tools will bring a negative impact on the software development. Selamat (1996) pointed out that if CASE tools users were not trained, then the possibility of companies not using of CASE tools would increase. At the end of his research, he concluded that in Malaysia, there were four key reasons for companies not utilizing CASE tools. The key reasons were: high learning curve, little training, lack of expertise and lack of management. Later in 1996, he did another study in using CASE tools. He gave the reasons for companies abandoning the use of CASE tools in three countries and this is shown in Table 2.5.

Table 2.5: Reasons for Abandoning CASE Tools in Three Countries

Reasons	Rank		
	Malaysia	**USA**	**UK**
Lack of adequate training	**1**	**2**	
Lack of CASE expertise	**2**	**3**	
Poor management	**3**	**4**	**1**
High cost of tool			**2**
Satisfaction with current methods			**3**
Lack of involvement of IS personnel in selecting CASE tools		**1**	

Source (Selamat, 1996)

From Table 2.5, it can be seen that the lack of adequate training is the most important reason for abandoning CASE tools in Malaysia. Therefore, IT organizations should put more effort on training employees in using CASE tools.

2. Cost

Cost is always a concern when buyers purchase products. The cost of CASE tools is also an issue among IT organizations. If the cost of CASE tools is too high, the companies will not be able to use them. Premkumar (1995) stated that the initial purchase price of a CASE tool system would be greater than its benefits and the cost of training for CASE tools was high. Selamat (1996) concluded that in UK, the main reason in rejecting the use of CASE tools was high cost. The same case happened in Singapore, and Danny (1998) found that the main barrier to use of CASE tools was the cost of CASE tools shown in Table 2.6.

35

Table 2.6: Barriers to Use of CASE Tools

Rank	Factors	Response	Percent(%)	Mean	S.D
1	High cost of implementing CASE tools	Average Neutral Disagree (Missing)	46.3 14.8 7.4 31.5	3.784	0.917
2	Long learning curve to use CASE tools effectively	Average Neutral Disagree (Missing)	48.1 11.1 9.3 31.5	3.676	0.852
3	Limited capability of CASE tools	Average Neutral Disagree (Missing)	35.2 25.9 7.4 31.5	3.595	0.927
4	Lack of fit between system development methodology and CASE tools	Average Neutral Disagree (Missing)	38.9 20.4 11.1 29.6	3.526	0.922
5	Using CASE tools without knowledge of underlying software engineering methods & techniques	Average Neutral Disagree (Missing)	35.2 18.5 13.0 33.3	3.472	1.055
6	Uncertainty over benefits of using CASE tools	Average Neutral Disagree (Missing)	31.5 9.3 27.8 31.5	3.135	1.058
7	Resistance to system development	Average Neutral Disagree (Missing)	22.2 16.7 29.6 31.5	2.946	0.970
Ranking is based on the mean Responses are grouped as follows: Disagree 1, 2:Netrual, 3: Agree; 54 cases-100%					

Source (Danny, 1998)

Simon (2002) argued that there were additional costs related to the use of CASE tools, such as installation of CASE tools and manual. These costs can discourage the use of CASE tools.

3. Implementation

Implementing a CASE tool involves a lot of things such as methodologies and management of CASE tools. If companies cannot manage the implementation of CASE tools properly, the companies will suffer many problems in using the CASE

tools. Usually, these problems come from methodology, management and organization.

1) Methodology

Methodlogy is a very important factor in implementing a CASE tool. Like any other technology, CASE tools have some disadvantages in its design. Charies (1988) argued that CASE tools should provide two kinds of flexible functions. First, the CASE tools must be able to tailor a general methodology to a specific application. Second, CASE tools must be flexible enough to allow developers to customize them in different techniques. Some CASE tools cannot provide methodologies that developers want. Some developers may use one of the methodologies that they are familiar with. Sometimes, however, the developers are requested to change and use other methodologies, which they are not familiar with. This will result in problems due to unfamiliarity of different CASE tools.

2) Management Support

In order to implement CASE tools successfully, obtaining support from management is very necessary. Clifferd (1992) gave advice to managers for implementing CASE tools and he said that gaining approval and sustaining sponsorship are key factors in the long-term success of CASE adoption efforts. Financial support must be gained from management because CASE tools are usually expensive. Before giving the support, the management should have a good understanding on the benefits that the organization will get from CASE tools. On the other hand, project managers should set a short-term plan and a long-term plan on using CASE tools. After implementing the CASE tools,

37

project managers must audit the performance of using the CASE tools in terms of the productivity of developers and the quality of software produced.

3) Organizations

To implement a CASE tool successfully, the organizational issues must be considered seriously because these issues will affect the success of CASE tools implementation. The three major issues are: attitude of employees towards change, organizational culture and size of organization.

The attitude of employees towards change is one of the main criteria to ensure the success of the implementation of CASE tools. The employees must accept the changes resulting from using different CASE tools. Initially, the employees fear changes because they have to learn new technologies such as CASE tools. If the developers do not want to use CASE tools during development, then it will be difficult to implement CASE tools in organizations.

Organizational culture is also important in implementing CASE tools. Managing teamwork has been an issue for many years. In many projects, the developers work more independently rather than work as a team. Now software development is becoming more complex, it is impossible for individuals to finish a big project. CASE tools can promote sharing of information within a team during software development processes. If CASE tools are correctly used in a team, developers will be able to exploit the capability of the CASE tools to the maximum. If a team cannot work closely with each other, the CASE tools will not be able to give much help for the team.

The size of an organization is also a consideration in the implementation of CASE tools. If a company is very small, the cost of CASE tools is a big burden

for it. Beside this, in a small-size company, one developer plays many roles in software development. It is too difficult for him or her to handle the cycle of software process by using CASE tools. Hence, the features of CASE tools cannot be fully utilized.

2.5.2 Issues in Educational Sector

Table 2.7 indicates that in 2005, more than 93,292 students graduated and entered the job market in Malaysia. The students with strong skills will become very competitive in job market. The report said that the focus on ICT in education would ensure the creation of a large pool of relevant-skilled expertise. Development of skilled manpower, in fact, is one of the focus areas for the education and industry sectors in Malaysia.

Table 2.7: Manpower Qualifications -

IT and Engineering Manpower Available for the ICT Sector

Qualification	2001	2002	2003	2004	2005	Total
Diploma	33,395	37,098	46,791	56,227	63,932	237,442
Bachelor	17,741	19,767	23,265	25,466	27,686	113,925
Postgraduate	847	1,043	1,207	1,384	1,675	6,156
TOTAL	51,983	57,908	71,263	83,077	93,292	357,532

Source (http://www.asocio.org/resources/profiles/malaysia.pdf)

According to Judy (1986), "Students who have received a computer science degree from a programme that emphasizes the 'nuts and bolts' of learning to use multiple programming languages and several fourth generation languages many find that the available jobs require a higher-level perspective of system development". To fully meet the industrial requirements, in the past few years, many colleges and universities offered

software courses. In the software courses, learning CASE tools became compulsory for students. When using CASE tools as learning tools, the educators and students faced some problems. James (1992) defined four specific requirements, which are: visual approaches, program generator, desired lab and environment. Mali (2005) also explained that "the issues related to the difficulties in learning CASE tools and the perceived complexity of learning CASE tools is recognized by educational practitioners, who complain that most commercial CASE tools are not adaptable to the learning requirements of student projects and some of the more comprehensive ones suffer from the problems of a long learning curve". In implementing CASE tools in the educational environment, Donald (1999) pointed out four main factors as follows:

1) Faculty

 A high-quality faculty and staff is the single most critical element in the success of a program. Faculty needs both advanced education in computing and experience in software engineering practice. Because of the dynamic nature of computing, it is essential that faculty continue to engage in professional development such as research, participation in professional societies, consulting, and technical training.

2) Infrastructure

The program must provide adequate infrastructure and technical support. This includes well-equipped laboratories and classrooms, modern CASE Tools, and sufficient reference and documentation material.

3) Industry Participation

A critical element in the success of a software engineering curriculum is the involvement and participation of industry. Industrial advisory boards and industry-academic partnerships help maintain curriculum relevance and currency.

4) Student Involvement

Interaction with students about curriculum development and delivery provides valuable information for assessing and analyzing a curriculum. Involvement of students in professional organizations and activities extends and enhances their education.

Barbee (1990) pointed out that there were some issues for teaching CASE tools in the educational environment. These issues included the student's learning time, cost of CASE tools, lecturer's support, and facilities. Besides these issues, Barbee further defined some other issues such as the CASE tools themselves, assignments, and vendor's support. Some important issues will be explored in detail below:

1) Student's learning time

Student's learning time is one of the main issues faced by educators and students. Usually, lecturers give a one-hour laboratory session for experiencing CASE tools. During this period, the students spend much time just to login the system. If the system has some security procedures, the students will have to navigate to open the proper files. Actually, this repeated process occupies much time. The actual time spent on learning CASE tools will be very little in the end.

2) Costs of CASE tools

According to a report given by Barbee (1990) in 1988, the cost of one component of CASE tools was approximately $500 and the cost of a complete system was $12500. Paul (1991) conducted a research in USA. This research shows that the start-up and ongoing costs of CASE tools are rather high; as shown in Figure 2.4 and Figure 2.5. The high cost of CASE tools becomes a barrier in teaching CASE tools for educators and students.

Workstation (Personal Computer Hardware) 75 at $ 5,000	--------	$375,000
CASE Tools Software:		
System plan tools (PC) at 5 $5000	--------	$25,000
System design tools (PC) at 75 $ 5000	--------	$375,000
Management tools (PC) 15 at $5000	--------	$75,000
Implementation tools (maintenance)	--------	$25,000
Upgrades and interface software development		
Consultants 50 days at $ 1000/day	--------	$50,000
Staff Training		
Trainers: 100 days at $1000/day	--------	$100,000
Staff Time: 10 days at $175/day	--------	$1750
Total investment	--------	$951,750

Figure 2.4: Start-Up CASE Investment

Software Engineering Group:		
Three people at $ 5000 (salary & benefits)	----------	$150,000
Hardware Maintenance 75 at $ 500	----------	$37,500
Software Upgrades and maintenance		
Estimated at 10% a year for PCs	----------	$47,500
Estimated at 15% a year for mainframes	----------	$37,500
On-going staff training (done by engineering)		
Two days per person for staff of 150 at $175	----------	$52,500
Attendance at user meetings, symposia		
Tow people to attend four meetings at $1500	----------	$12,000
Miscellaneous	----------	$10,000
Books and publications		
Total annual ongoing cost	----------	$347,000

Figure 2.5: Ongoing Cost of Using CASE Tools

3) Lecturer's support

Support from the lecturers is also very important in learning CASE tools. Lecturers who have to teach CASE tools should be very proficient in their use before they can teach students as this has an impact on the effectiveness of student's learning. If the lecturer does not guide the students well when teaching CASE tools, the students will not be able to get much knowledge in using CASE tools.

4) Facilities

Generally, facilities include the number of laboratories, PCs and other software available. There must be enough laboratories to allow the students to practise using the CASE tools. If there are not enough PCs, some students will have to wait for their turn or share PCs with others. The students may be discouraged in such a situation and lose interest in learning CASE tools.

5) The CASE tools themselves

Most of the CASE tools are made for use in the industrial environment and have many advanced features. Therefore, students feel that it is quite difficult to learn and use the CASE tools. To give sufficient support to students, the vendors should provide a version of CASE tools that are suitable for learning purposes. For example, a CASE tool with simple features.

6) Materials and assignments

Materials and assignments are very essential in learning and practicing to use CASE tools. Before starting to learn CASE tools, sufficient materials should be given to students. It is very important for the students to get vendor-supplied materials. Jeffery (1993) stated that "There is a need for the development of educational material which falls somewhere between the short, step-by-step tutorial and tool-independent text books presentation of methodologies". The students will be able to learn to use CASE tools from textbooks. When they use the CASE tools in laboratory, they will feel that learning to use CASE tools is easier. Assignments given to students should be close to the real world. After finishing the assignments, these students can really experience the CASE tools. If the assignments are too simple, the students will just use a few features of the CASE tools to do it. Then, the CASE tools will not give much benefit to them.

7) Vendor's support

As said before, many CASE tools are made for use in the industrial environment. Some students do not know what CASE tools are and the features that a CASE tool has. Then, vendors should promote the awareness of the use of CASE tools in the educational environment. Alan (1993) specified few recommendations for vendors and agencies. They should:

1. Raise awareness of the range of CASE tools currently available in the marketplace. For example, vendors make presentations at the agencies.

2. Build up an accessible body of knowledge on CASE tools for use within the agencies, so that tool users have access to information on the availability, use, costs, and benefits of CASE tools.

3. Provide greater encouragement and promotion of CASE tool user groups. This perhaps requires the agency resources and management commitment to ensure they are attended and successful.

4. Ensure that the vendors are aware of the agency's particular needs with regard to CASE tool support.

Vendors must provide very effective support to promote CASE tools. They should know what students really need for using CASE tools and show the benefits of using CASE tools in a real work environment. Ideally, the vendors should offer an educational version of CASE tools at no charge or at significantly reduced prices because students who gain experience using CASE tools will be potential customers in the future.

2.6 Similar Web Sites for FOCT System Development

In order to promote the usage of CASE Tools, the researcher searched more than 30 web-based systems, which include CASE tools vendors, and research organizations. These websites provide some kinds of features for CASE tools users. Some main helpful features are shown in Figure 2.6.
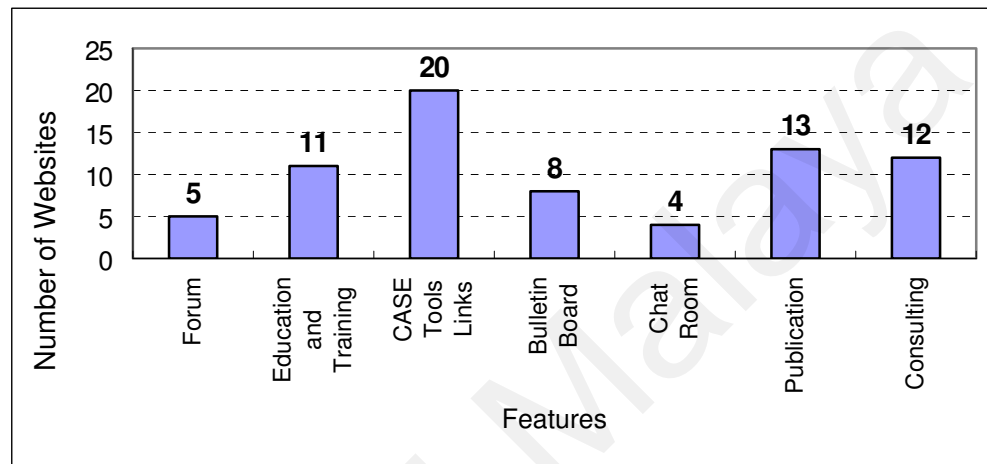

Figure 2.6: Features of CASE Tools Websites

In Figure 2.6, it can be seen that twenty websites provide helpful CASE tools links. Thirteen websites in this research provide publications for readers and twelve websites provide consulting services. Eleven websites give education and training for CASE tools users. Eight websites have bulletin boards and five companies have forums. Four companies have chat rooms. Through observing these websites, the researcher found that there were some weaknesses. They are discussed below.

1) Most of features are designed for IT professionals only, not for educational sectors.

   After accessing these websites, the researcher found that most of web sites were designed for IT professionals. There were only few vendors, who have educational version of CASE tools. According to a report from faqs (1997), some CASE tools such as Schemacode and RationalRose provide an educational version for teaching purposes. To promote the usage of CASE tools, the vendors targeted not only IT

professionals, but also educators. Therefore, there should be a channel that lets vendors, IT professionals and educators communicate with each other.

2) There is a gap between IT professionals and educators.

CASE tools users should include IT professionals, lecturers and students. It is not easy for students to master the skills in using these CASE tools because of reasons mentioned in the previous sections of this chapter. To improve students' skills, the researcher will create a platform where IT professional and students can interact with each other.

3) It is difficult to find a platform for both of lecturers and students to evaluate the effectiveness of using CASE tools and relevant subjects.

Some websites provided many articles to introduce CASE tools. But only few of them provide resources that guide lecturers and students how to learn CASE tools effectively and measure the result of using CASE tools. To support the learning of CASE tools, the researcher will develop a feature to evaluate the subjects and problems in learning CASE tools.

Generally, the websites accessed in this research have some common features for CASE tools users. These web sites only provide some basic information for CASE tools users. CASE tools users do not get sufficient support to learn CASE tools. To improve CASE tools usage, the researcher will develop a collaborative system for CASE tools users.

2.7 Conclusion

In this chapter, the researcher firstly explored the history of CASE tools with some supportive topics. These topics include definition of CASE tools, features of CASE tools, supportive methodologies and system development.

Secondly, the researcher discussed the current status of utilization of CASE tools in Malaysia and other countries according to some literature published by some experts. These experts conducted studies in Malaysia and other countries such as Singapore and Finland. They found that in the past few years the percentage of people using CASE tools in some countries was low because of reasons such as high implementation costs, training costs and the complexity of CASE tools.

Thirdly, the researcher discussed the importance of using CASE tools in both the industrial and educational sectors in Malaysia. IT organizations obtained much benefit such as achieving better quality of software, faster development process and standardization of software designs from using CASE tools. The colleges and universities also had positive effects on their teaching environment. The students could have hands-on experiences in using CASE tools and be competitive in the market place.

Fourthly, some issues were pointed out in implementing CASE tools in the two sectors. After reviewing the literature, the researcher found that there were some issues and concerns in implementing CASE tools. IT organizations and educators faced certainty that include the cost of using CASE tools, people's skills, learning time, support from vendors and lecturers and infrastructure.

Finally, some similar CASE tools web sites were explored to help the researcher to identify the features of this system. These websites gave the researcher more information and ideas about the usage of CASE tools. The researcher has a better understanding of

developing an adequate and collaborative system to provide a platform to both CASE tools users and vendors for using CASE tools.

After reviewing the literature, the researcher has an in-depth understanding in CASE tools' utilization in Malaysia. Based on the study described in this chapter, the researcher will conduct interviews and surveys in Malaysia. This work is described in the next chapter.

Chapter Three - Survey and Data Analysis

3.1 Introduction

In this chapter, the researcher will discuss the survey methodologies applied for data collection, followed by design of the survey, and the pilot test. Finally, the researcher will analyze the data collected from IT organizations, universities and colleges.

3.2 Survey Methodologies

The researcher administered questionnaires to lecturers and students in some colleges and universities in Malaysia. The researcher spent about one month to distribute the questionnaires randomly and collect the questionnaires from colleges and universities. The target lecturers were those teaching some subjects that involve using CASE tools such as System Analysis and Design, Software Engineering and Object-Oriented Techniques. The researcher went to most of the public universities and many private colleges around Kuala Lumpur and Selangor and distributed a hundred and twenty-one questionnaires personally among eight public universities as shown in Figure 3.1. The researcher also sent E-Mails to lecturers in USM and UMS because the two universities are not located around KL.
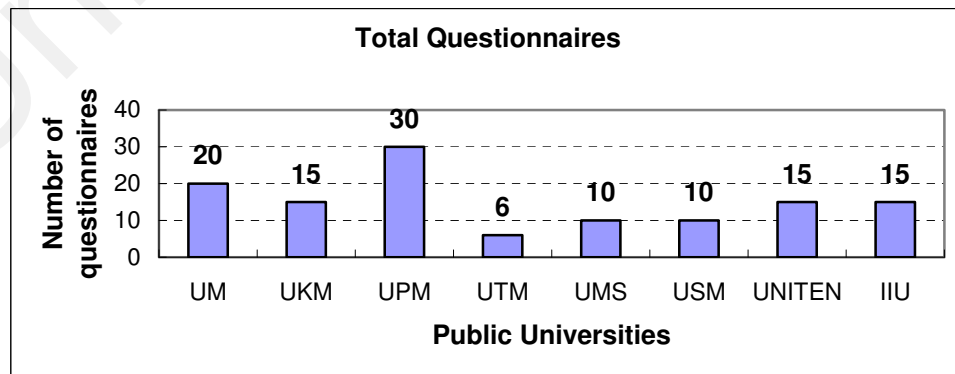
Figure 3.1: Total Questionnaires Distributed among Public Universities

The researcher distributed questionnaires among private universities and colleges as shown in Figure 3.2. One hundred and twenty-one copies were distributed to lecturers.
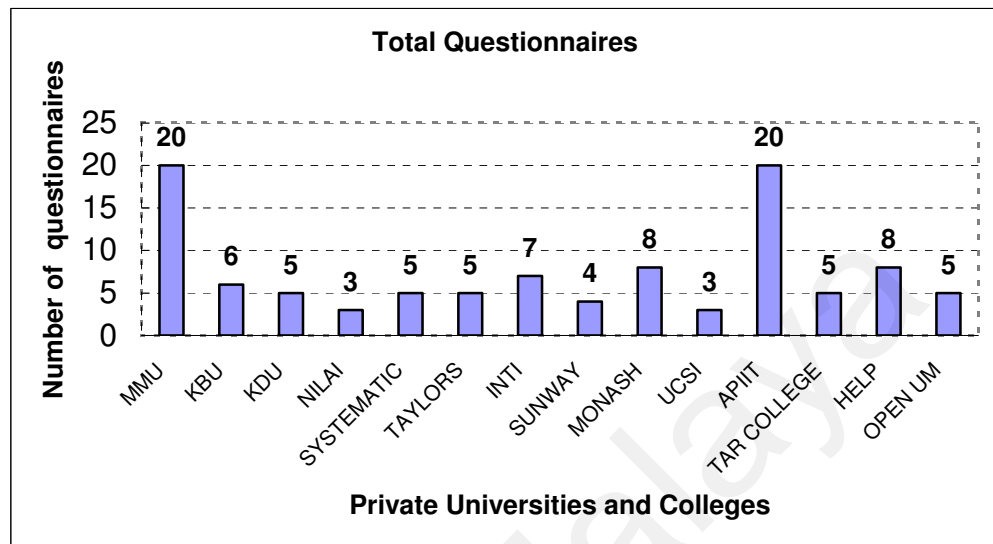
Total Questionnaires

A bar chart titled "Total Questionnaires" with y-axis labeled "Number of questionnaires" (ranging from 0 to 25) and x-axis labeled "Private Universities and Colleges". The bars show: MMU = 20, KBU = 6, KDU = 5, NILAI = 3, SYSTEMATIC = 5, TAYLORS = 5, INTI = 7, SUNWAY = 4, MONASH = 8, UCSI = 3, APIIT = 20, TAR COLLEGE = 5, HELP = 8, OPEN UM = 5.

Figure 3.2 Total Questionnaires Distributed among Private Universities and Colleges

In order to get good response, usually before distributing the questionnaires, the researcher sent e-mails to them with the softcopy of the questionnaire. Then the researcher made appointments with them and distributed the printed questionnaires to them and some lecturers asked for a softcopy rather than hardcopy of questionnaires. The researcher distributed about two hundred questionnaires among colleges and universities. However, the researcher only got fifty-three completed sets back. The questionnaires covered the following areas:

1) Background and experiences of lecturers

These questions ask about the lecturer's experience in teaching certain subjects.

50

2) Some subjects taught that include CASE tools

   These questions are to define how much experience the lectures have in the subjects they teach using CASE tools.

3) Awareness of teaching CASE tools

   The question is about the lecturer's exposure to CASE tools.

4) The time spent on teaching CASE tools for each subject

5) The lecturers were asked about the teaching time for each subject related with CASE tools.

6) Ease of teaching CASE tools

7) The lecturers were asked about how the lecturers feel about teaching the CASE tools.

8) Having sufficient time and infrastructure for teaching CASE tools

   These questions are to define whether the time and infrastructure are adequate for teaching CASE tools.

9) Problems faced in teaching CASE tools and suggestions to improve student's skills in CASE tools

   These questions are to investigate the issues and problems faced by lecturers during the time they teach the subjects that involve the use of CASE tools.

The target students include degree students and masters students in public universities and private educational colleges. The researcher sent out the questionnaires to students from UM, APIIT, KDU, SYSTEMATIC. The questionnaires covered the following areas:

1) Understanding on the background of CASE tools and methodologies

These questions are to investigate the students' level of understanding of CASE tools concepts and methodologies. From the answers given by the students, the researcher can know whether the students have solid fundamental knowledge about using CASE tools and methodologies.

2) Usage of CASE tools to these students

These questions are to ask the students on type of tools they are using, and which features they use.

3) Kinds of CASE tools being used in their colleges and universities

This question is to ask the students to give the name of CASE tools used in colleges and universities.

4) Feedback from students

By asking this question, the researcher will be able to get students' responses such as the acceptance of CASE tools, the benefits of using CASE tools and the problems in using CASE tools.

3.3 Design of Survey

Section A describes the survey results for IT organizations. The researcher sent eighty request letters for interviews. Twenty-five companies agreed to participate in the interviews. The researcher spent about one hour to conduct an interview in each IT organization. There were fourteen questions to be asked and these questions covered all research objectives. Through interviewing the IT professionals, the researcher got a better understanding of the kind of CASE tools utilized in Malaysia.

Section B describes the survey results for students. The students are either from public universities or private colleges. There were eighteen questions to be asked to the students. These questions try to find out all the objectives described in chapter one.

Section C describes the survey results for lecturers in universities and colleges. The researcher estimated the sample size in each university or college, and tried to cover all target people. There were thirteen questions to be asked. These questions covered many areas that were required for the research objectives in chapter one.

The researcher used the SPSS tool to analyze the data. The version of SPSS is 9.0 for Windows. It was used to create the frequency information as well as the charts.

3.4 Pilot Test for the Surveys

Before the administering the survey, the researcher did pilot tests with selected lecturers and students. The researcher gave five sets of questions to lecturers, and asked them give feedback about the survey. One of the lecturers from UM pointed out a problem on how the period of teaching CASE tools was to be defined. Five students were involved in the pilot test. The majority of the students agreed on the format of the survey. Two masters students from UM suggested giving specific names of the products of CASE tools.

3.5 Analysis of Data

Section A: Findings from IT organizations

The researcher interviewed twenty-five organizations for the research. After interviewing these organizations, the researcher found some facts for using CASE tools in software development. The facts covered most of problems and issues faced by IT organizations. All the interview questions are discussed below:

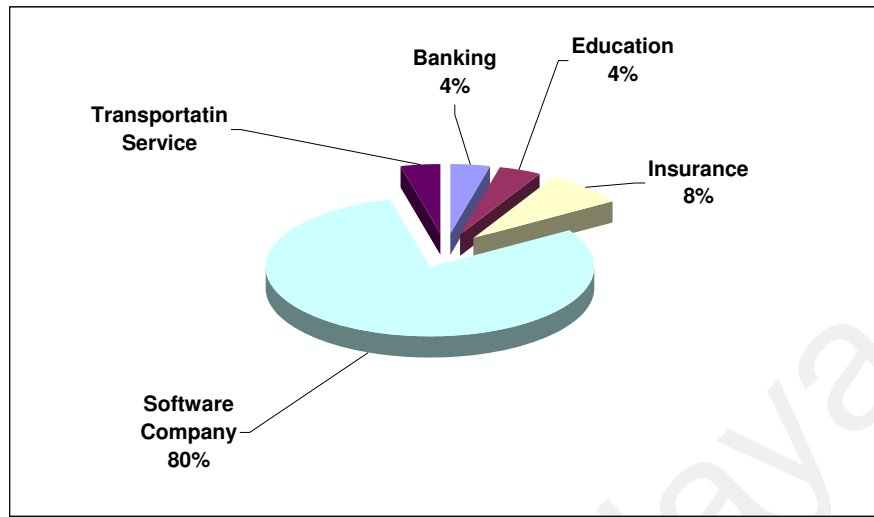**Responses for question 1: What is your organization's main business?**

Figure 3.3: Percentages of Sectors

Figure 3.3 shows that over 80% of IT organizations are software companies. The rest of IT organizations are from banking, education and insurance.

Responses for question 2: Could you describe the level of computerization in your organization in detail?

The interviewees described that these companies are highly computerized. They further described that 90% of jobs is processed by using computers.

Responses for question 3: **How about the utilization of CASE tools in your organization?**

These companies used different kinds of CASE tools for different purposes in software development phases. Mr.Darren Heng, Senior System Analyst in Ingenuity MicroSystems, explained that his company used Visio Professional for design and reverse engineering and also developed a tool named Test CASE and Bugs Log for testing purposes.
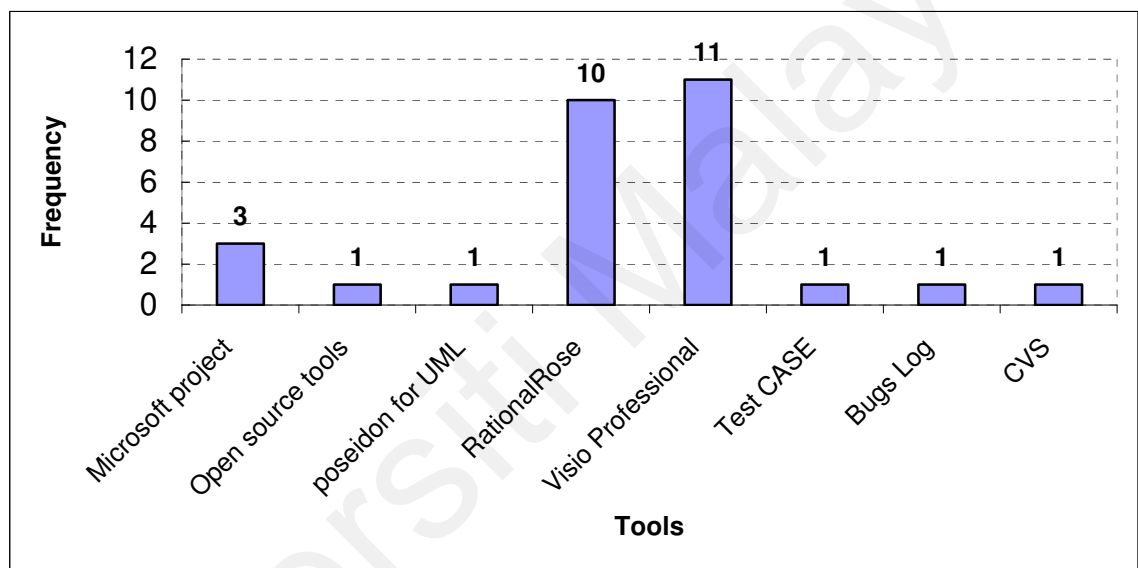

Figure 3.4: Tools Used in Software Development among these IT Organizations

The IT organizations selected tools based on real situations in their companies. Figure 3.4 shows that the most often used tool is Visio Professional. Eleven out of fifty-three companies used Visio Professional; ten companies used RationalRose, and three companies used Microsoft Project. Therefore, the conclusion can be made that the Visio Professional and RationalRose are the commonly used CASE tools in IT organizations in this study.

Responses for question 4: What skills and ability in using CASE tools do you expect from a fresh IT graduate before entering your company?

Through interviewing the IT professionals, most of them said that as professionals, they expected that fresh graduates have basic skills in software development.

1) Know methodologies in software development

    Most of the interviewees pointed out that it is very important for fresh graduates to understand the system methodologies. For example, they must know SDLC in software development. If the students do not know the SDLC, they would have no idea about each step in SDLC.

2) Know how to use CASE tools in real business

    Some interviewees said that fresh graduates should know how to use CASE tools like RationalRose. Mr.Cheah, Senior System Analyst in eGENTING, explained that if the students knew how to use CASE tools, the students would be competitive and could handle more work and tasks with minimum training for using CASE tools. Only a few interviewees said that they did not expect too much from fresh students since they just left school and they did not know much about real business.

3) In-depth understanding of UML and software modeling

    Some interviewees pointed out that the students should know UML, and be able to design simple models because UML is becoming very important for communication among team members. Having knowledge in Software modeling is very necessary for IT graduates. Mr.Cheah, Senior System Analyst in eGENTING, said that students should know the models in software development. For example in eGENTING, they usually use RationalRose to

design database first. Therefore, the students must know UML as it is used to represent the models.

From the IT professional's comments, the researcher concluded that SDLC, usage of CASE tool and UML are very important skills for the student.

Responses for question 5: Do you think that the vendor can provide adequate training for using a CASE tool after your company bought the CASE tool?

More than 84% of respondents agreed that the vendors provided enough training. However, 16% of interviewees said that the vendors did not give enough help to customers. The data is showed in Figure 3.5.
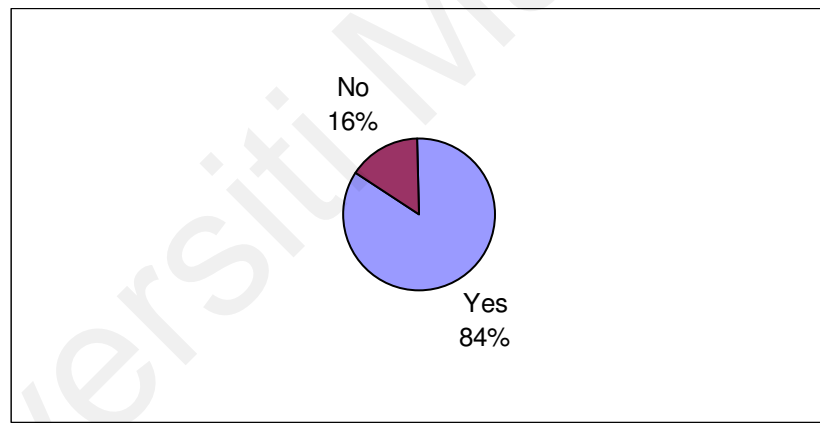
Figure 3.5: Percentage of Respondent for Vendor Training

The interviewees further explained that it depended on the skill level of users. If the users are new to CASE tools, then the vendor training is enough for basic level. However, if the users are in intermediate or advanced level of using the CASE tools, then the vendors cannot give much help to them because they expect much more knowledge in real business than using the basic features. The rest of interviewees agreed that the vendors did not give sufficient support for using CASE tools. One of the reasons is that the vendors

only gave few examples of using the features of the CASE tools, and these examples were not really relevant to real business situations. Another reason is that most projects are team-based projects. Before starting a project, the team must make sure that all members in this team must know how to use the CASE tools. Sometimes, a company sends one or two staffs for training, and the rest of the staff did not get the opportunity to learn the tools. They will have different understanding about the CASE tools in this team, and eventually they will have a gap within them. With regards to vendor training, it can be concluded that the vendors provide sufficient basic training for IT professionals.

Responses for question 6: How long will employees spend on learning how to use a CASE tool?

Most of the interviewees said that one or two weeks were enough to learn the CASE tools. The time required for learning CASE tools depends on staff's experience in using the CASE tools.

Responses for question 7: Do you think that the universities and colleges are teaching students the CASE tools that the industry requires?
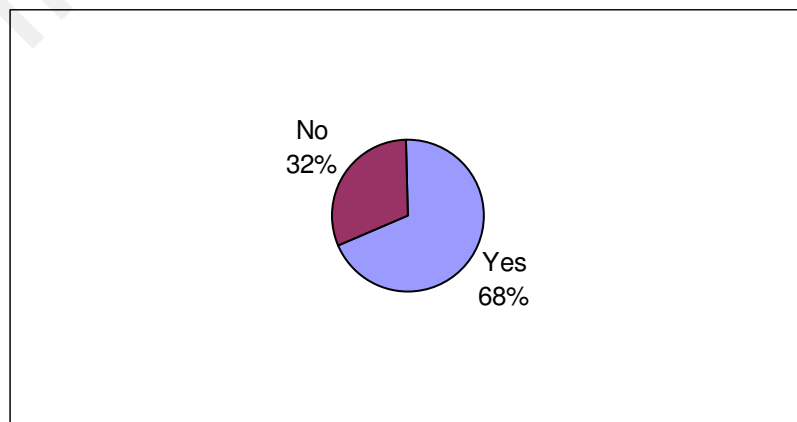
Figure 3.6: Agreement of Educators Providing Courses Using CASE Tools

The data in Figure 3.6 shows that 68% of the interviewees said that educators should provide some courses using CASE tools whilst 32% of the interviewees said that the educators did not really need to offer any subjects using the CASE tools. From the above, most interviewees felt that educators should teach and encourage students to learn the CASE tools. They advised lecturers to give more projects and assignments, which closely resembles the real world, and not just theory-based. The lecturers should teach the students the principles of using the CASE tools, and not just how to draw diagrams. The researcher concluded that most IT professionals agreed that it is really necessary to teach CASE tools for students in colleges and universities.

Responses for question 8: Do you think that it is necessary to use CASE tools to develop software?
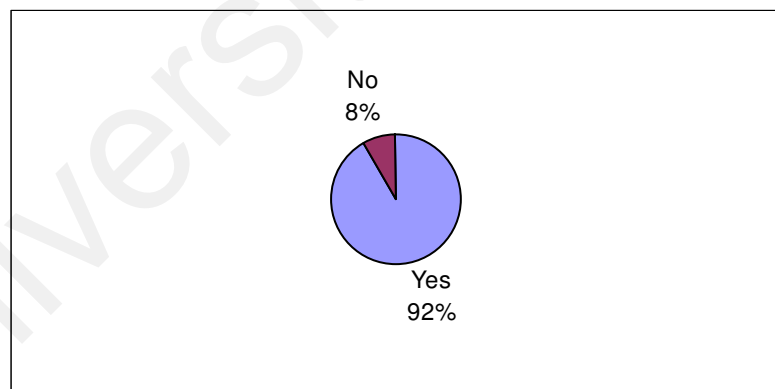
Figure 3.7: Necessity of Using CASE Tools

According to the interviews as shown in Figure 3.7, 92% of companies agreed that using CASE tools is very necessary. They listed out some benefits of using CASE tools. CASE tools could promote better communication among a team; CASE tools enforced standardization on software development. Only 8% of interviewees said that it is not really

necessary to use CASE tools in software development. They argued that the cost of using CASE tools would be a burden for companies since the costs of CASE tools are very high.

Responses for question 9: what are the main purposes of using CASE tools in software development?

After the interviews, the researcher found that the main purposes of using CASE tools were: visualization of applications, analysis of risks, cost estimation of projects and documentation. Some companies used CASE tools in the design stage and to draw diagrams. They did not use the full features of CASE tools.

Responses for question 10: Do you think there will be a growing trend of CASE tools utilizing in IT organizations in future?

Over 90% of the interviewees believed that there would be a growing trend in the use of CASE tools in IT organizations but this would take time. The majority of the interviewees agreed that obstacles of using CASE tools were the cost and complexity of the CASE tools. Some big companies experienced the benefits of using the CASE tools, so they do believe that there will be a growing trend of using CASE tools. Mr. Tengku Omar, Analyst Programmer in Teras Teknology Sdn Bhd, has been using RationalRose for few years. He explained that some designs could be re-used for other projects and it would be easy to manage the documentation. This would speed up the development process. He is very confident that the use of CASE Tools will increase significantly in the future.

Responses for question 11: Do you think it is very important that companies send their employees to get training for using the CASE tools?

More than half of the interviewees said that it was necessary that companies should send employees for training. However, they also said that the training period was too short for the students to master the CASE Tools.

Responses for question 12: In your opinion, what benefits can be obtained from using CASE tools?

Most of the interviewees strongly agreed that they obtained much benefit from using CASE tools. The tools helped them to generate documentation, improve communication within team members and promote knowledge sharing in their working places. Mr. Kan Chow Keat, Technical Consultant of Silverlake System Sdn. Bhd. added that the CASE tools could help the management to measure the maturity of software development in his company. He gave an example of how RationalRose enforces developers to follow certain standards.

Responses for question 13: What are the problems and issues in CASE tools' utilization that needs to be considered if you want to use CASE tools?

Interviewees have some problems and issues in using CASE tools. The main problems are:

1) Cost of CASE tools

The cost of CASE tools is a big problem in implementing the CASE tools. The companies could not afford the tools. They did not invest on it with a big amount of money. They were interested in seeing reliable or tangible results.

2) Complexity of CASE tools

Most of the interviewees agreed that the complexity of tools is an issue for developers. They felt the interfaces of the tools and some terms in CASE tools are quite difficult to understand. They hoped that the tools would have better interface designs in future.

3) Integration with different tools

According to Mr. Zarrella, "Some vendors have developed import/export tools extensions to deal with the problems of data dictionaries that could not be shared by different tools or among multiple users". Integration with different tools is a problem. Some companies used different tools for different purposes. Mr. Kan Chow Keat suggested that there should be more consolidated CASE environments instead of too many specific purpose independent CASE tools, which did not provide an integrated environment. In such CASE environment, the tools could not be fully utilized and maybe have negative effects in software development.

Responses for question 14: What are your suggestions on promoting the usage of CASE tools?

To promote the usage of CASE tools, some interviewees suggested using Open-Source tools in the work place and thereby improve the utilization of CASE tools in organizations. Mr. Kelvin Yap explained that using Open-source tools would significantly reduce the cost of the tools as Open-source tools are free. The management should realize the benefits of using the tools on long-term, and not on short-term because implementing the CASE tools takes longer time. Probably, after implementing for two or three years, the management would be able to see the benefits of using the tools.

Section B: Findings from students' survey

There are two parts in section B. Part one deals with the background of the students. Part two covers the use of CASE tools in their colleges and universities. Areas to be discussed include background of respondents, relevant subjects, the CASE tools applied in subjects, sufficiency of facilities and guidance, ease of using the CASE tools, time, and purposes of using the CASE tools and factors affecting the use of CASE tools.

Fifty students were involved in the survey. Figure 3.8 shows that the students were from six different universities and colleges. Sixteen students were from UM. One student was from UTM; ten were from APIIT; thirteen were from KDU; and three were from SYSTEMATIC. Seven students did not indicate the college or university they were from.
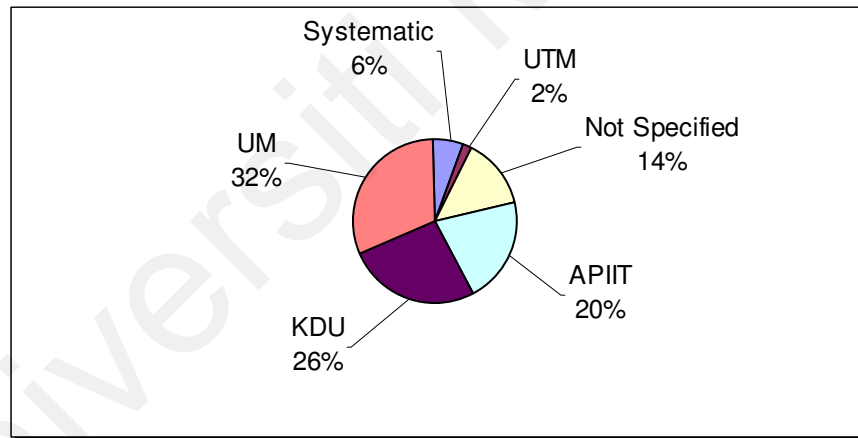
Figure 3.8: Total Respondents from Universities and Colleges

Responses for question 1: Are you a degree student or masters student?
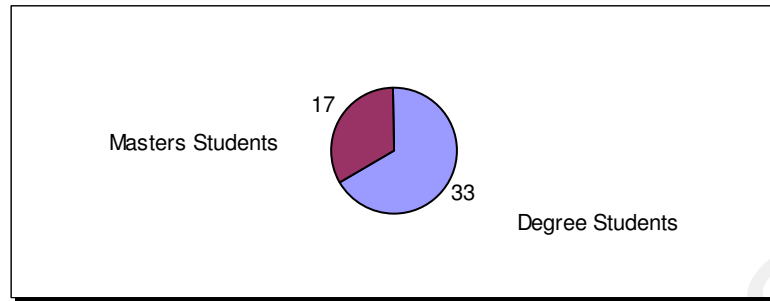
Figure 3.9: Levels of Students

Figure 3.9 shows that thirty-three out of fifty participants were degree students, and seventeen were master students.

Responses for question 2: What subjects have you studied before?
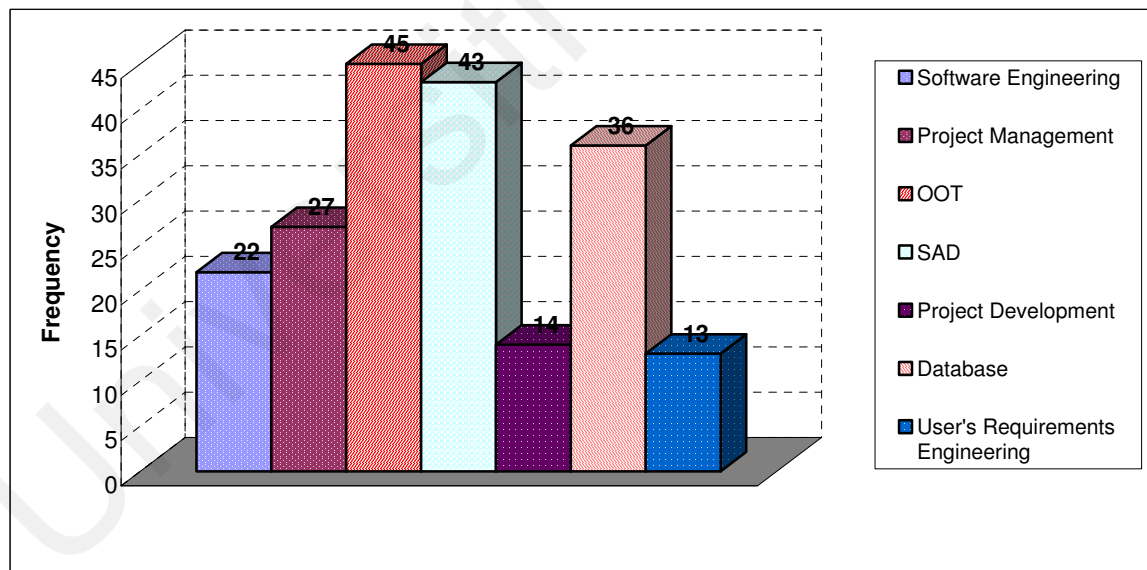
Figure 3.10: Subjects Learnt by Students

To give more CASE tools knowledge to IT students, many educational institutions offered subjects that involved the use of CASE tools. In this survey, the researcher wanted to

know what subjects the students learned. The researcher found that forty-five students have learnt OOT, forty-three students have learnt SAD, and only thirteen students have learnt User Requirements Engineering. Twenty out of fifty students learned Software Engineering. From the responses to this question the researcher concluded that most of the students have a basic understanding of OOT and SAD.

Responses for question 3: How well do you know the System Development Life Cycle (SDLC)?
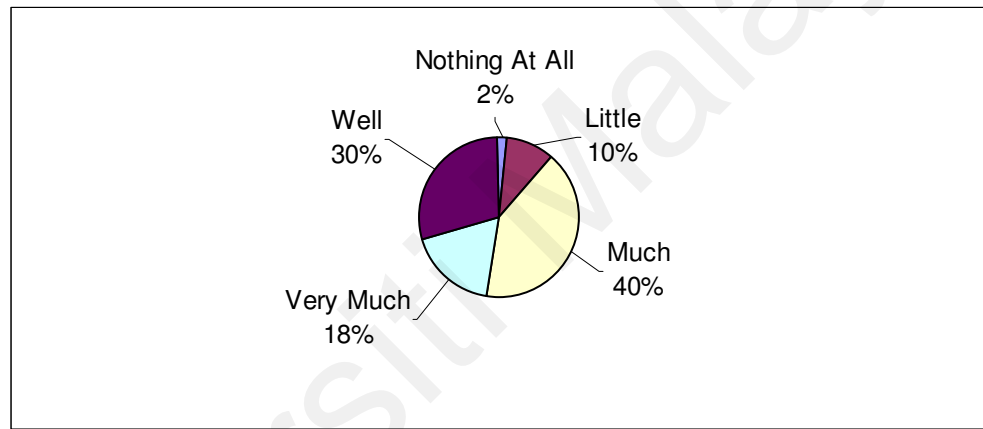
Figure 3.11: Knowing SDLC

SDLC is a very important and common methodology in software development. IT students should have strong knowledge in SDLC. Figure3.11 shows that 98% of the students had different levels of knowledge in SDLC. Thirty percent of the respondents knew SDLC well. Eighteen percent of the respondents knew SDLC very much. Forty percent of the respondents knew SDLC much. Ten percent of the respondents knew only a little. Only 2% of students did not know SDLC at all. Therefore, conclusion is that most students knew SDLC.

Responses for question 4: How often do you use SDLC in your projects or assignments?
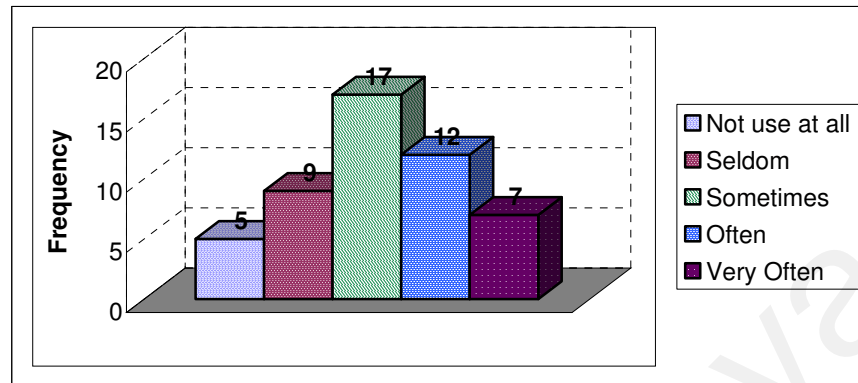
Figure 3.12: Frequency of Using SDLC

To get a better understanding of SDLC, students must apply SDLC in their projects or assignments frequently. Figure 3.12 shows that seventeen students used SDLC in their assignments sometimes. Out of fifty students, twelve used SDLC often; seven used SDLC very often; nine seldom used SDLC; and five did not use SDLC at all. The finding of this question is that majority of students knew SDLC and fully applied SDLC in their assignments.

Responses for question 5: Are you familiar with Object-oriented Techniques?
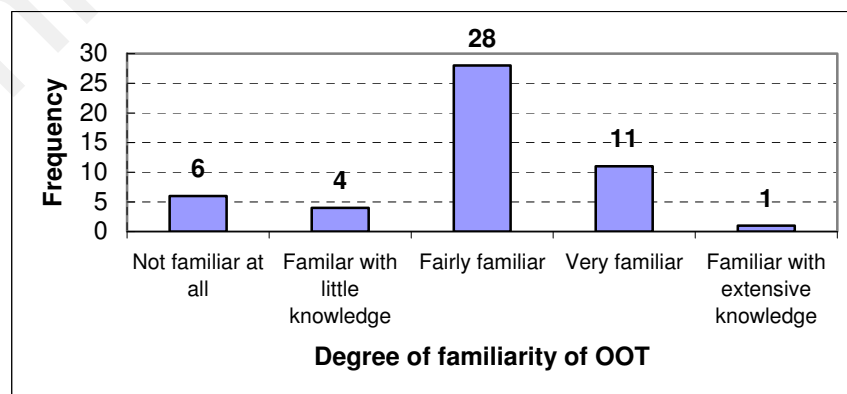
Figure 3.13: Familiarity of OOT

Figure 3.13 indicates that twenty-eight students out of fifty are familiar with OOT. Eleven students are very familiar with OOT. One student has extensive knowledge in OOT. Four students are familiar with little knowledge and six students out of fifty are not familiar with OOT at all. The conclusion is that the majority of respondents are familiar with OOT.

Responses for question 6: Did you learn how to use CASE tools?
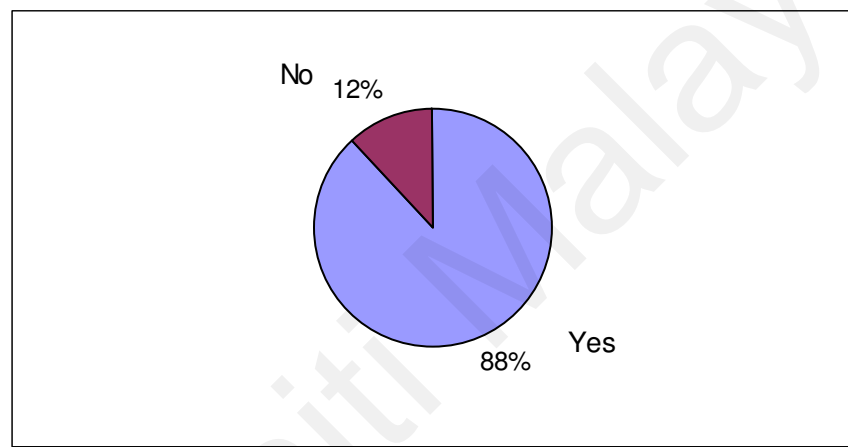
Figure 3.14: Whether Learnt CASE Tools Before

With reference to Figure 3.14, the researcher found that 88% of students said that they learned CASE tools before, and 12% of students did not. Therefore, most of the students have learned CASE tools before.

Responses for question 7: Do you think CASE tools are really useful to help you improve your skills in software development?
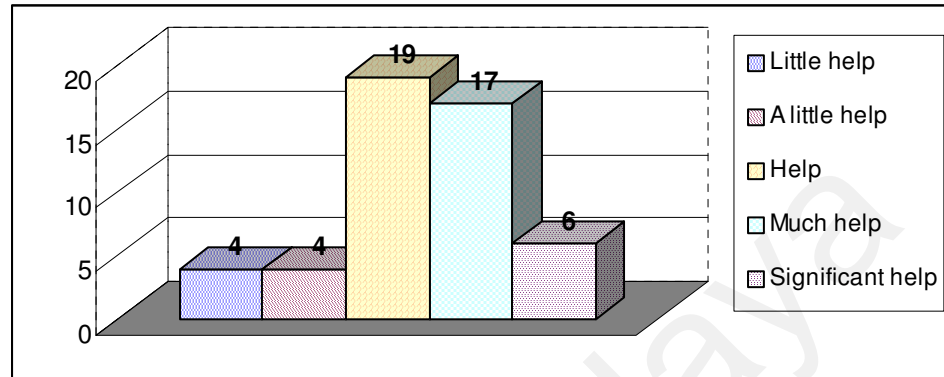
Figure 3.15: Degree of Help in Improving Software Skill

Figure 3.15 shows that nineteen students out of fifty felt that CASE tools could help them to improve software skills. Seventeen students felt that they could get much help for software skills. Six students felt that they got significant help. Four students felt they got little help in software skills and four students obtained a little help in software skills. The researcher concluded that most of them felt that CASE tools were helpful in improving software skills.

Responses for question 8: Which of the CASE tools have you learnt?

CASE tools must be applied in subjects teaching. Different subjects maybe used different tools.
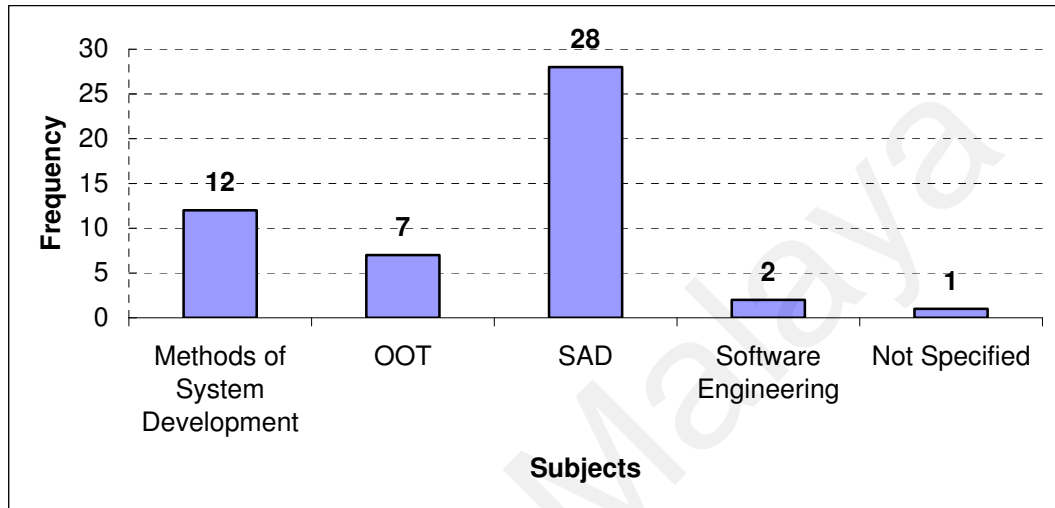
Figure 3.16: Subjects Learnt Using RationalRose

RationalRose is a very common tool. Figure 3.16 shows that out of fifty students, twenty-eight learned SAD using RationalRose. Twelve students learned Method of System Development using RationalRose. Two students learnt Software Engineering using RationalRose. One of the students did not indicate whether he/she used RationalRose. The researcher concluded that RationalRose is commonly used in SAD.
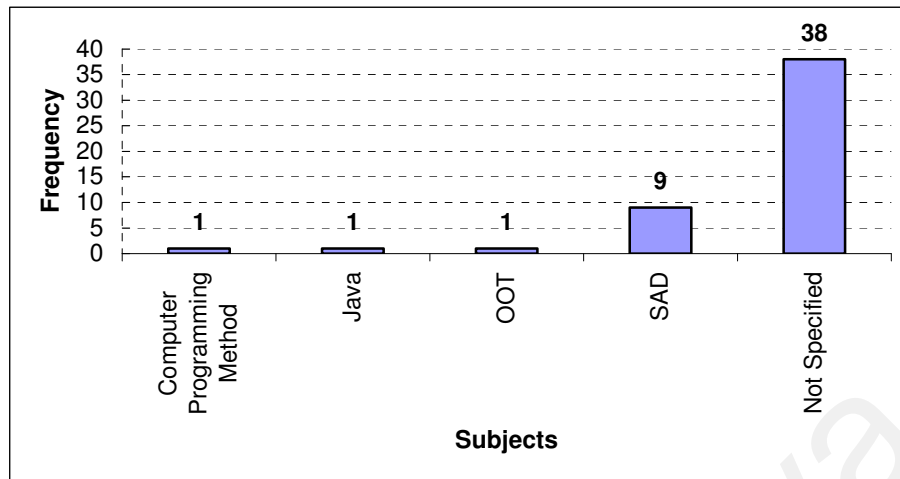
Figure 3.17: Subjects Learnt Using SystemArchitect

Figure 3.17 shows that there were nine students who learned SAD using SystemArchitect. Thirty-eight students did not use SystemArchitect at all. Nine students out of fifty learnt SAD using SystemArchitect.
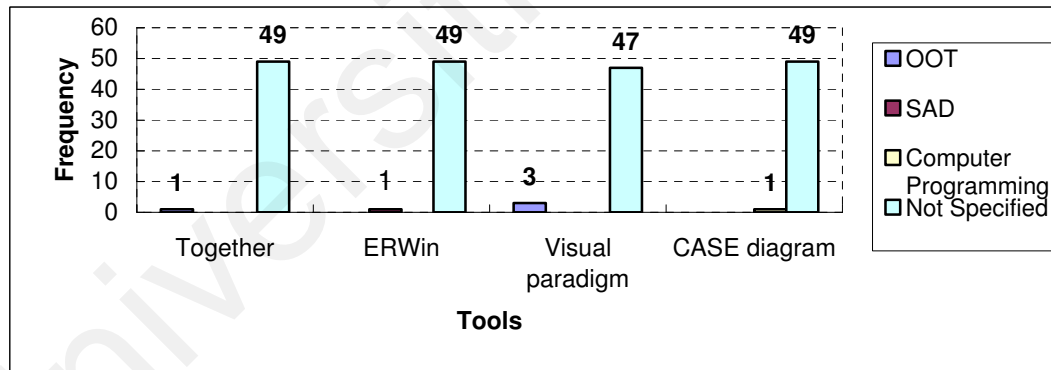

Figure 3.18: Subjects Learnt Using Other Tools

Figure 3.18 shows that some other tools were used in teaching. Three students out of fifty used Visual paradigm to learn OOT. One used Together to learn OOT and one respondent used ERWin to learn SAD. One respondent used CASE diagram to learn Computer Programming. These tools were rarely used in the learning environment. Most of students did not specify the selection. The researcher concluded that most of the students used RationalRose and SystemArchitect in SAD.

Responses for question 9: Do you think your university or college provides sufficient facilities to use CASE tools?
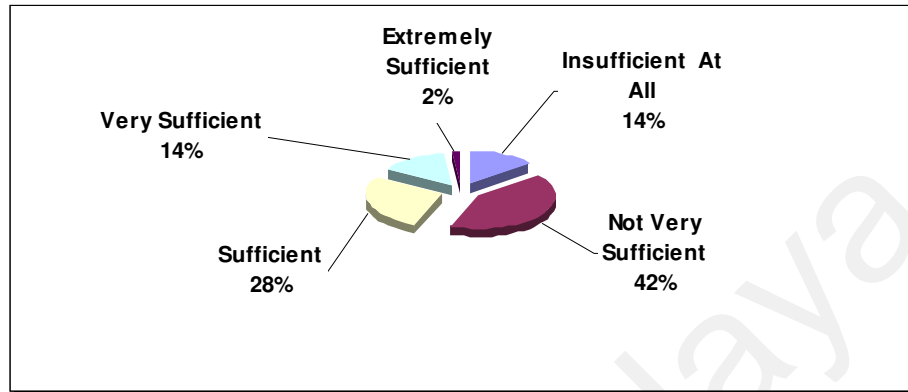

Figure 3.19: Sufficiency of Facilities

Figure 3.19 shows that 42% respondents said that the facilities were not sufficient. 14% respondents agreed that the facilities were not sufficient at all. 28% respondents commented that the facilities were sufficient. 14% respondents said that the facilities were very sufficient. Only 2% respondent said that the facilities were extremely sufficient. Therefore, the results of this finding is that only 44% respondents agreed that the facilities are sufficient and more than 56% respondents indicated that the facilities are not sufficient for students to learn the CASE tools. The conclusion in this question is that the facilities are not sufficient for student's learning.

Responses for question 10: How do you find learning to use the following CASE tools?

Ease of using CASE tools is very important for students to master the CASE tools. In this question, the researcher wanted to find out which tool was the easiest to use.
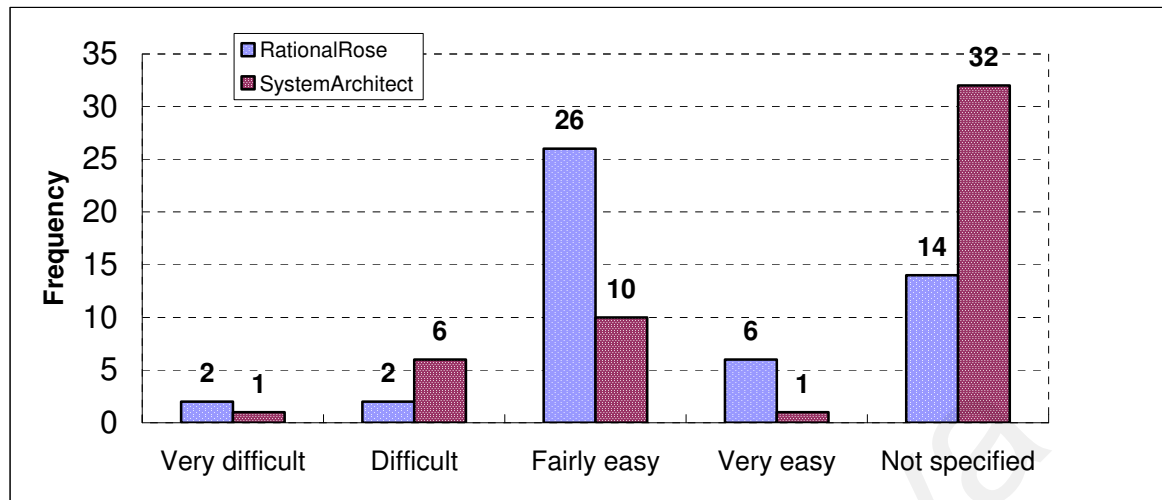
Figure 3.20: Degree of Ease of Using RationalRose and SystemArchitect

Figure 3.20 shows that thirty-six respondents evaluated RationalRose and fourteen respondents did not evaluate RationalRose. Among thirty-six respondents, twenty-six respondents said that RationalRose is fairly easy to use for their studies. Eighty-one percent of the respondents felt that RationalRose was fairly easy to use. Eighteen respondents participated in the evaluation of SystemArchitect. Out of these, ten respondents commented that SystemArchietect is fairly easy to use. Generally, RationalRose is more acceptable than SystmArchitect in terms of the degree of the ease of use.
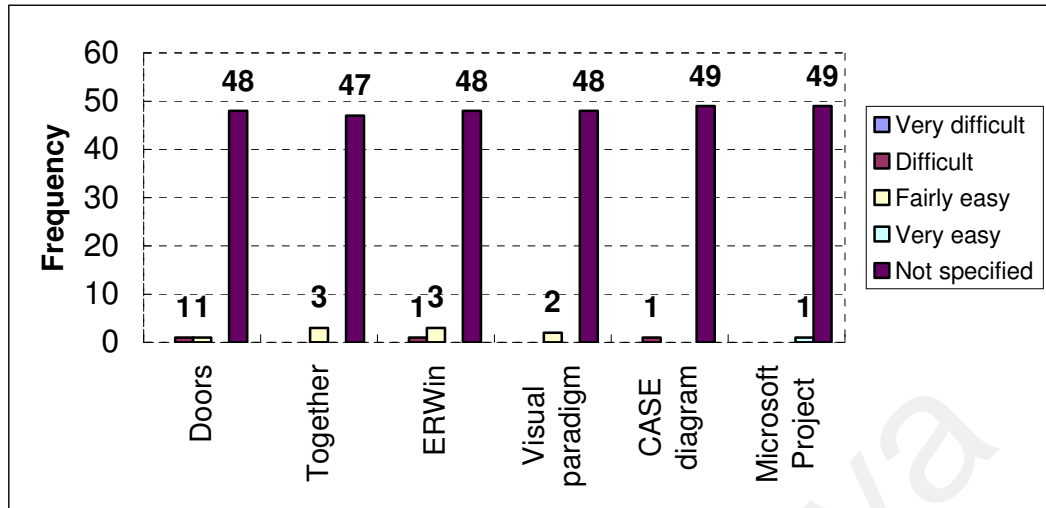
Figure 3.21: Degree of Ease of Using other Tools

Figure 3.21 shows some other tools used by students. Out of fifty students, three said that it is fairly easy to use Together. One of the students said that ERWin is difficult to use. Three of the students said that ERWin is fairly easy to use. Two students said that Visual paradigm is fairly easy to use. Compared with Rational and SystemArchitect, the conclusion is that the main tools that are accepted by students are RationalRose and SystemArchitect.

Responses for question 11: How long did you spend learning the CASE tools?

There are two sub-questions in this question for each CASE tool. One was to ask students how many hours they actually spent and another one was to ask how many hours they thought they should have spent.
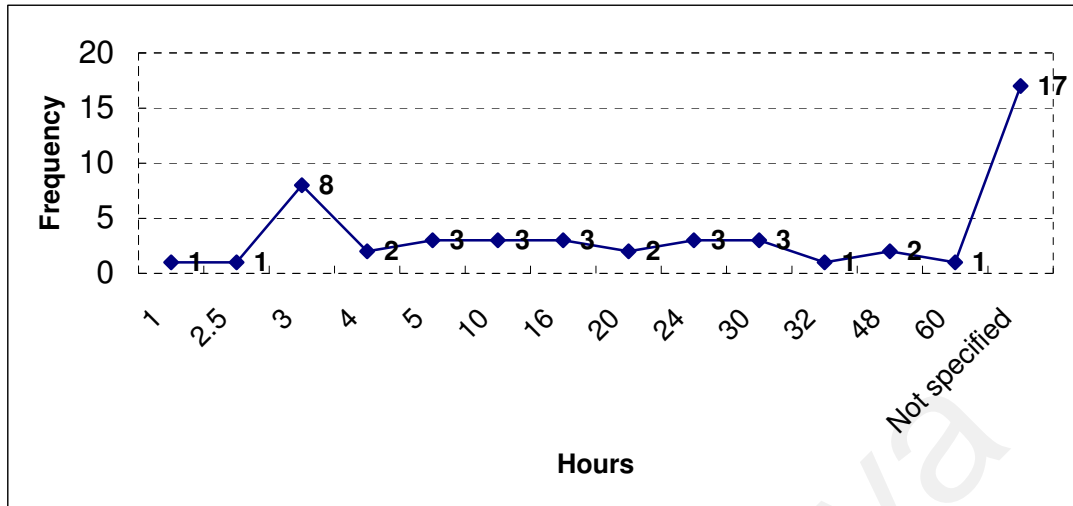
73

Figure 3.22: Hours Spent in RationalRose

Figure 3.22 shows there were seventeen respondents who did not specify any values, which probably means they did not use RationalRose in their studies. More than thirty-three respondents used RationalRose in their studies. Eight students spent three hours each learning RationalRose. Three students or fewer than three students spent between one to sixty hours each to study RationalRose. Figure 3.22 shows that the majority of the students did not define a general duration of learning the CASE tools. From this figure, the possible explanation is that the students did not know how many hours they spent learning the CASE tools.
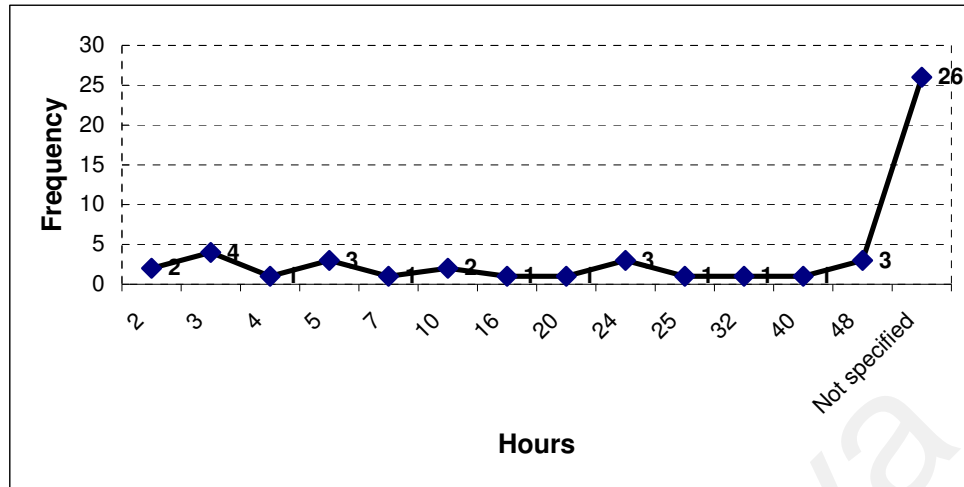
Figure 3.23: Hours Expected in RationalRose

Regarding the hours expected in learning RationalRose, there were many students who spent different hours as shown in Figure 3.23. Perhaps, because the students did not know how many hours they actually spent in RationalRose, they would not know how many hours they expected to spend. There is not much different between hours spent and hours spent in learning RationalRose.
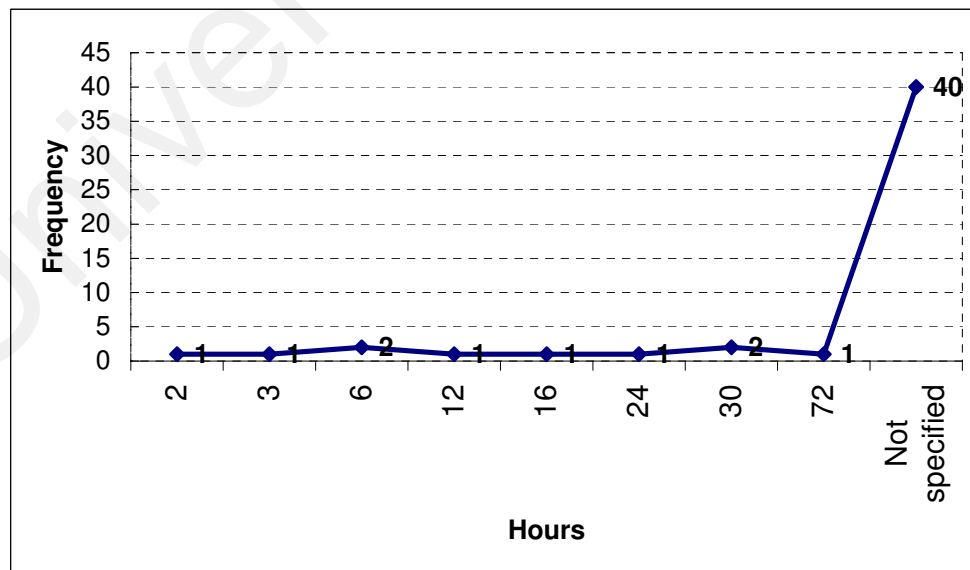

Figure 3.24: Hours Spent in SystemArchitect

Ten respondents said that they spent one or two hours learning SystemArchitect. Forty students did not learn SystemArchitect. In Figure 3.24, forty students did not specify any time duration.

Probably, the respondents commonly used RationalRose rather than SystemArchitect. One student out of ten spent one hour to learn SystemArchitect. One of the students spent seventy-two hours to learn SystemArchitect. The similarity of the hours spent in RationalRose and SystemArchitect is that most students spent three or less than three hours to learn the two tools with wide ranges from one hour to seventy-two hours. The respondents were not sure how many hours they should spend learning the two tools.

In Figure 3.25, forty-one respondents did not specify any time duration. Only nine students had expectation on the time of learning SystemArchitect. The hours expected in SystemArchitect ranged from three hours to seventy hours. Compared to hours spent in SystemArchitect, the nine expected two or less than two hours to learn SystemArchitect. On the average, the time durations of learning SystemArchitect are comparatively short. Probably, the respondents felt that SystemArchitect is easy to use for them, and they did not need three hours to learn SystemArchitect. Regarding this question, the researcher found that only few respondents spent one or two hours learning RationalRose and SystemArchitect. Other respondents gave different time durations. Generally, the respondents did not really know how many hours they should spend learning the CASE tools.
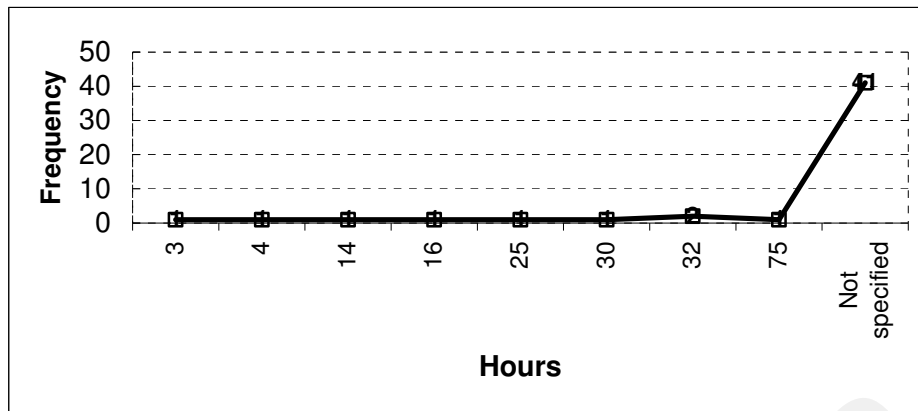
Figure 3.25: Hours Expected in SystemArchitect

Responses for question 12: So far, what can you do by using the CASE tools in your assignments?
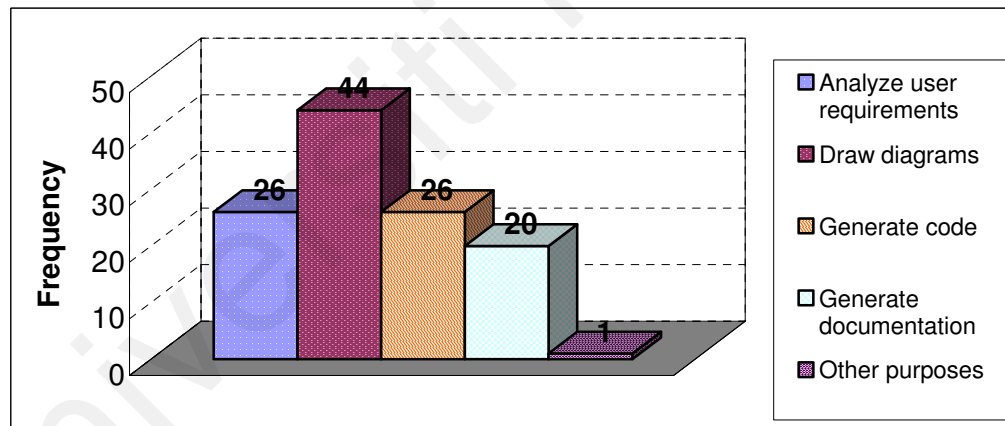

Figure 3.26: Purposes of Using CASE Tools

CASE tools can be used for different purposes in software development. More than forty-four respondents out of fifty used the CASE tools to draw diagrams; twenty-six respondents used the CASE tools for the purposes of analyzing user requirements and generating code; twenty of them used the tools to generate documentations. One of the respondents specified the tools for other purpose. This respondent defined the purpose of

77

using the tools is to plan projects. From Figure 3.26, the researcher found that the respondents used the main functions of the CASE tools such as drawing diagrams, analyzing user requirements and generating code and documentation in software development process. According to a study done by Augustin (1999), more than 74.66% of companies used CASE tools to draw diagrams; more than 66.44% of companies used CASE tools for code generation and 76.71% of companies used CASE tools for documentation. In this study, 88% of respondents commented that the main purpose was to draw diagrams. 56% of respondents used CASE tools for analyzing user requirements and generating code. 20% of respondents used CASE tools for generating documentation. Basically, the purposes of using the CASE tools in this survey met the usage of the CASE tools in Software companies.

Responses for question 13: Do you think that your lecturers have given you sufficient guidance or assistance in using CASE tools?
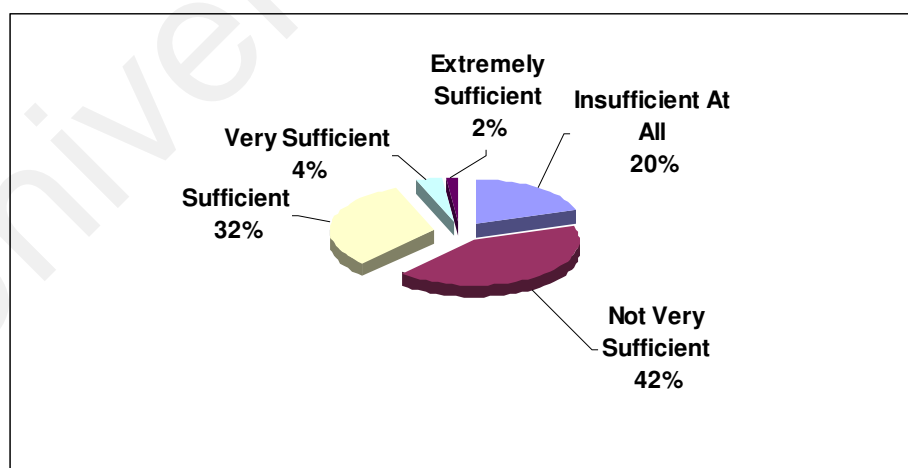

Figure 3.27: Sufficient Guidance

In Figure 3.27, 20% of respondents said that the lecturer's support was insufficient for learning CASE tools. 42% of respondents agreed that the lecturer's support was not very

78

sufficient. 32% of students in this survey commented that the lecturer's guidance was sufficient to teach CASE tools. Only 4% of respondents said that the lecturer's support was very sufficient and 2% of respondents said that the lecturer's guidance was extremely sufficient to teach CASE tools. The general finding is that 62% of respondents agreed the guidance was not sufficient and 38% of respondents said that the lecturers gave sufficient guidance in using CASE tools.

Responses for question 14: Please elaborate on your response to question 13.

Based on the result in Figure 3.27, the possible reasons are that lecturers did not give them much help in learning the CASE tools and the lecturers did not have experience in the CASE tools. Most of the respondents explained that lecturers did not give much help in using the CASE tools, and some lecturers did not have much knowledge in CASE tools.

Responses for question 15: In your opinion, what prerequisite knowledge is necessary before you learn how to use CASE tools?
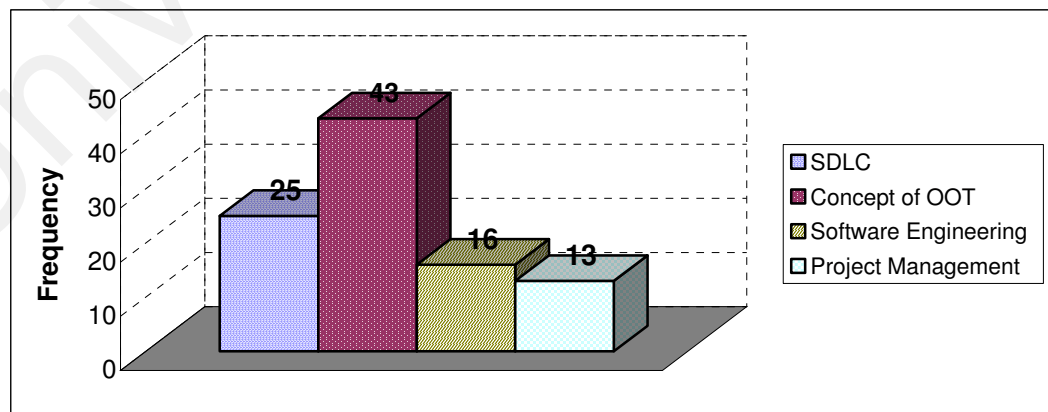

Figure 3.28: the Necessary Knowledge for Learning CASE Tools

Learning CASE tools needs some knowledge from different areas. Generally, some subjects are required for students to learn CASE tools. In Figure 3.28, more than forty-three respondents out of fifty said that having the concept of OOT was the most important knowledge. Twenty-five respondents agreed that SDLC was a necessary subject to learn CASE tools. Sixteen respondents commented that knowledge on Software Engineering was necessary before learning the CASE tools. Only thirteen respondents said that Project Management was a necessary subject to learn the CASE tools. Generally the knowledge in OOT and SDLC followed by Software Engineering and Project Management were necessary for learning CASE tools.

Responses for question 16: What are the limitations you experienced in learning and using CASE tools in your university or college?
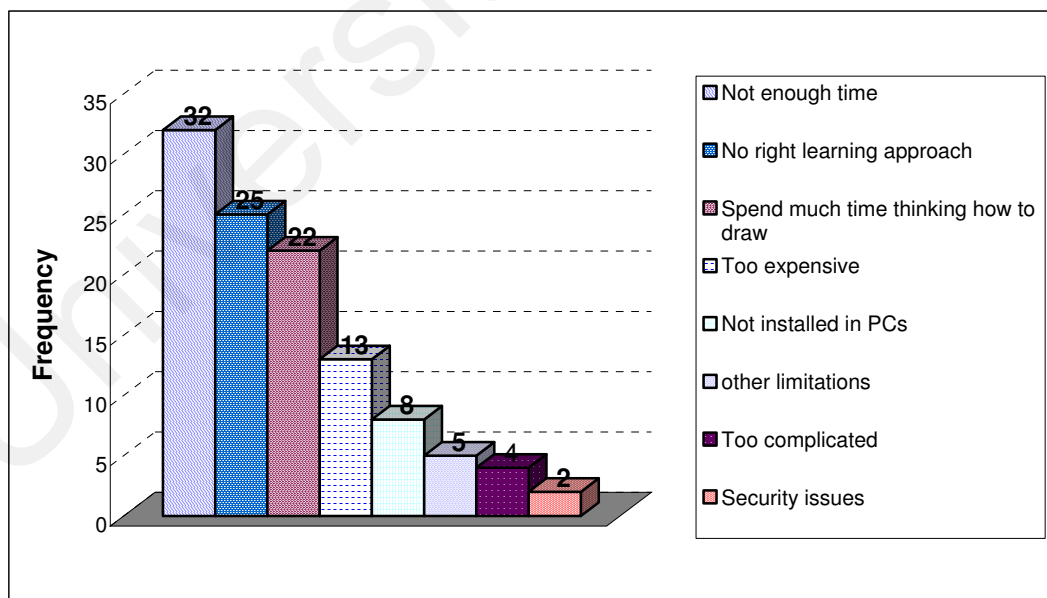

Figure 3.29: Factors Affecting Learning CASE Tools

Some factors affected the students on learning CASE tools. The researcher listed out some possible factors of using the CASE tools in universities and colleges. In Figure 3.29, more

than thirty-two of respondents said that they did not have enough time to learn the CASE tools. The respondents further said that the lecturers did not give much time to learn the CASE tools and there were no specified time periods in syllabuses. Twenty-six respondents agreed that there were no appropriate learning approaches to learn the CASE tools. Most of the respondents explained that the lecturers were not helpful. They further said that the lecturers gave assignments and projects to them, and asked them to learn the CASE tools themselves. Therefore, they had to learn themselves the use of CASE tools for completing their assignments and projects. Eventually, the students came out with too many versions of assignments and different diagrams with different notations. Most of the students did not know which one was right or most suitable in their assignments. Furthermore, the lecturers did not give back the assignments to them. The students did not know the results of assignments. Twenty-two respondents thought they spent too much time thinking how to draw diagrams. Subsequently, because respondents did not learn much about the CASE tools, they would spend much time trying out functions of the tools. The students would spend half an hour to find right the function for the basic diagrams. Thirteen students commented on the cost of CASE tools. Only eight respondents considered the factor of installation of CASE tools in PCs. Four respondents concerned about the complexity of the tools. Five respondents defined other reasons for limitation of using CASE tools. The researcher concluded based on the finding in Figure 3.29 more than half of the respondents have limitations such as time and learning approach.

Responses for question 17: Please elaborate on your responses to question 16.

Most of the respondents argued that they did not have enough time to learn CASE tools. Most of the students spent one hour per week learning CASE tools. Sometimes, there was inadequate time for the students to try CASE tools. Some students also explained that lecturers should give them right learning approaches.

Responses for question 18: What is your suggestion to improve the ease of learning CASE tools for you?

Most of the students in this study suggested that the lecturers should update themselves with CASE tools knowledge before teaching the students. They further said that there should be more PCs in labs and they should be given a longer time for mastering the CASE tools.

Section C: Survey from Lecturers

Section C is one of three surveys. In this section, the researcher asked lecturers some questions that covered the objectives described in chapter one.
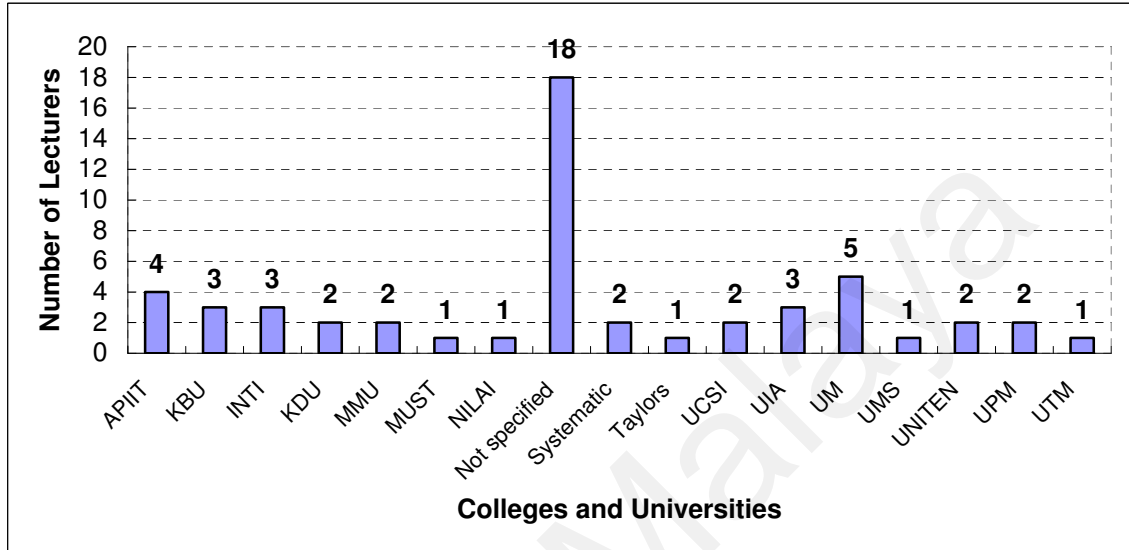

Figure 3.30: Number of Respondents from Colleges and Universities

Initially, to get more accurate results of the survey, the researcher distributed about two hundreds and forty copies of questionnaires to lecturers. Only 22% of questionnaires were returned compared to the number of distributions. In Figure 3.30, the higher respondents were from UM and APIIT. Eventually, the researcher collected fifty-three respondents from different colleges and universities. The respondents covered most of the colleges and universities in Malaysia. Fourteen respondents were from public universities and twenty-one respondents were from private universities and colleges. Eighteen respondents did not specify the names of their universities or colleges.

Responses for question 1: How long have you taught the following courses?

The lecturers' experience is important for students to gain better understanding of using CASE tools. In this survey, the researcher wanted to know the lecturer's experience in some subjects, which are related to the usage of CASE tools.
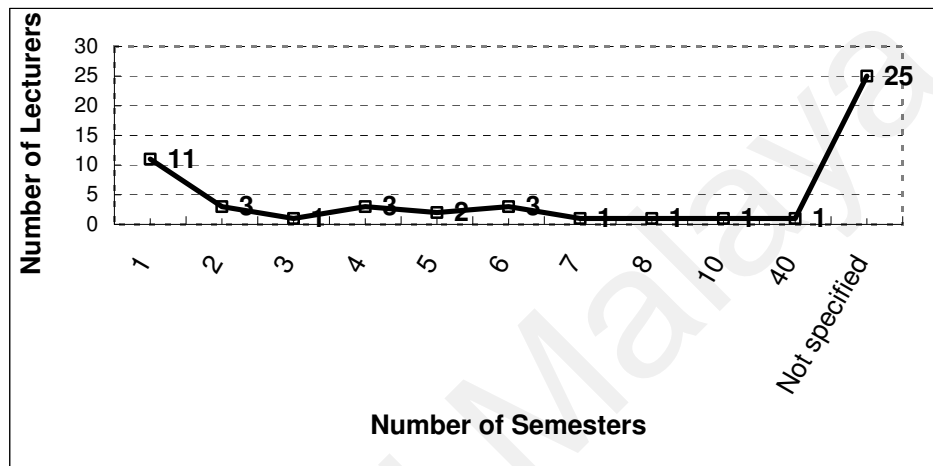

Figure 3.31: Number of Lecturers who Taught Software Engineering

In Figure 3.31, twenty-eight lecturers out of fifty-three taught Software Engineering. Twenty-five respondents did not have any teaching experience in Software Engineering. Eleven of the twenty-eight respondents taught Software Engineering for one semester. One respondent taught Software Engineering for forty semesters. Twenty of twenty-eight respondents have six or less than six semesters experience in Software Engineering. Four of twenty-eight respondents have seven or more than seven semesters experience in Software Engineering.
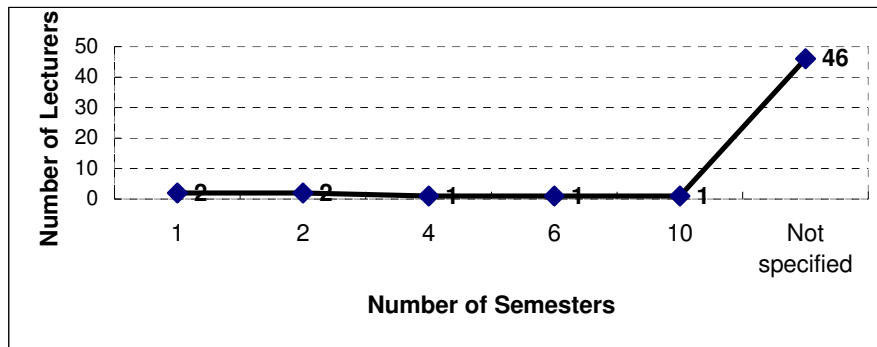
Figure 3.32: Number of Lecturers who Taught Project Management

In Figure 3.32, forty-six respondents out of fifty-three respondents did not have any experience in teaching Project Management. Seven respondents had teaching experience in Project Management. Two respondents had one semester experience in Project Management. One respondent had a ten semester teaching experience in Project Management.
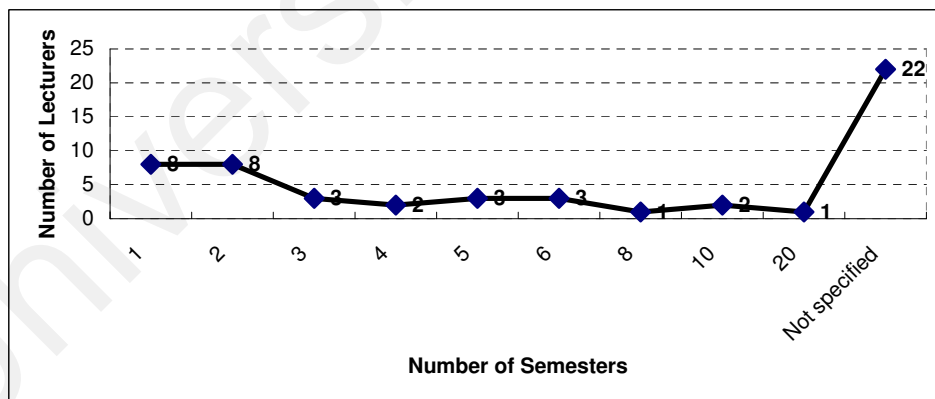

Figure 3.33: Number of Lecturers who Taught OOT

In Figure 3.33, twenty-two respondents did not have any experience in teaching OOT. Thirty-one respondents had teaching experience in OOT, which means that 62% of respondents had teaching experience in OOT. Twenty-seven of the thirty-one lecturers had

less than six semesters in teaching OOT. Four of thirty-one lecturers had more than six
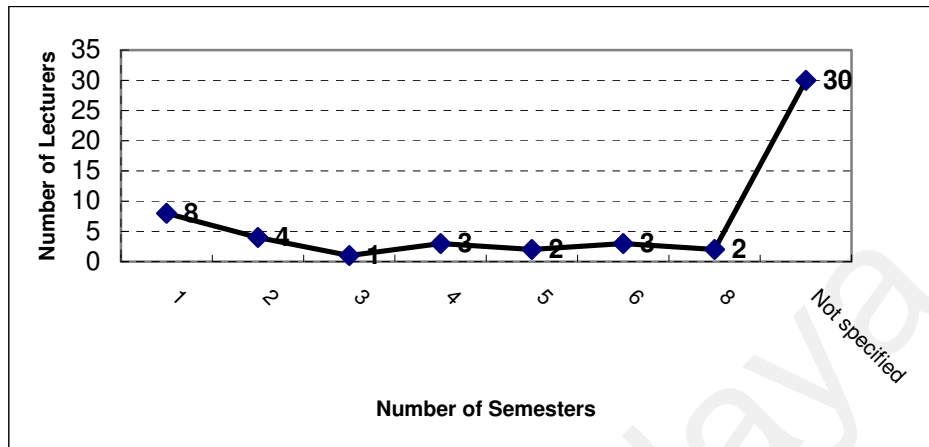
semesters teaching experience in OOT.

Figure 3.34: Number of Lecturers Who Taught Database

In Figure 3.34, thirty respondents did not specify any semester teaching Database. Twenty-three respondents had taught Database. Among respondents who had teaching experience, twenty-one respondents had less than six semesters teaching experience in Database. Only two respondents had more than eight semesters teaching experience in Database.
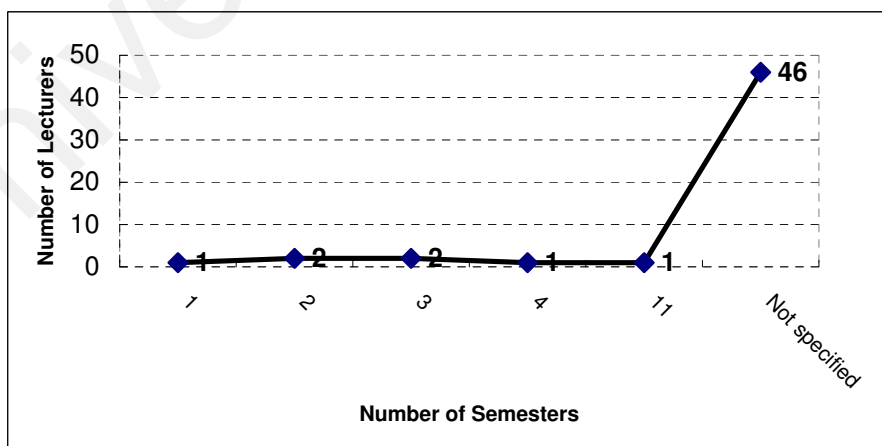

Figure 3.35: Number of Lecturers who Taught User Requirements Engineering

In Figure 3.35, forty-six respondents did not specify any semesters teaching User Requirements Engineering. Only seven the respondents had teaching experience in User Requirements Engineering. Among the respondents who had taught User Requirements Engineering, six respondents out of seven had four or less four semesters teaching experience in User Requirements Engineering. The majority of respondents did not have teaching experience in this subject. Perhaps, this subject is a relatively new subject for lecturers.
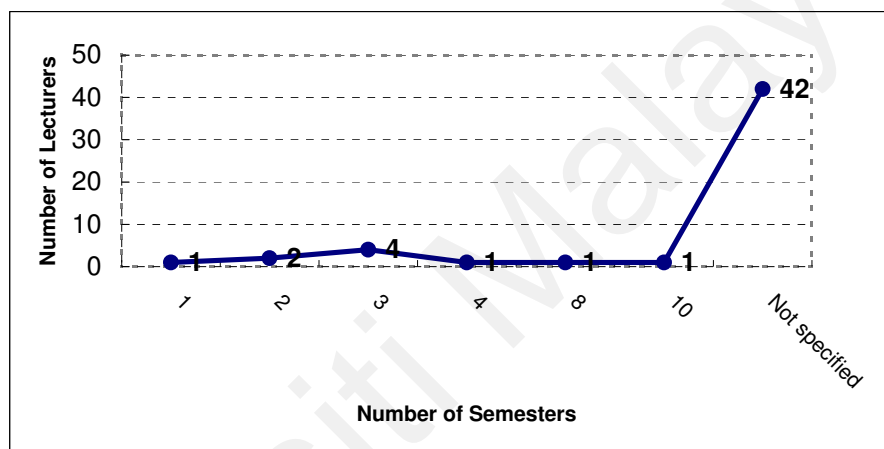

Figure 3.36: Number of Lecturers who Taught Project Management

As shown in Figure 3.36, forty-two respondents did not have teaching experience in Project Development; eleven of respondents taught Project Development. Among respondents who taught Project Development, nine respondents had less than eight semesters teaching experience in Project Development. One of respondents had taught for ten semesters.
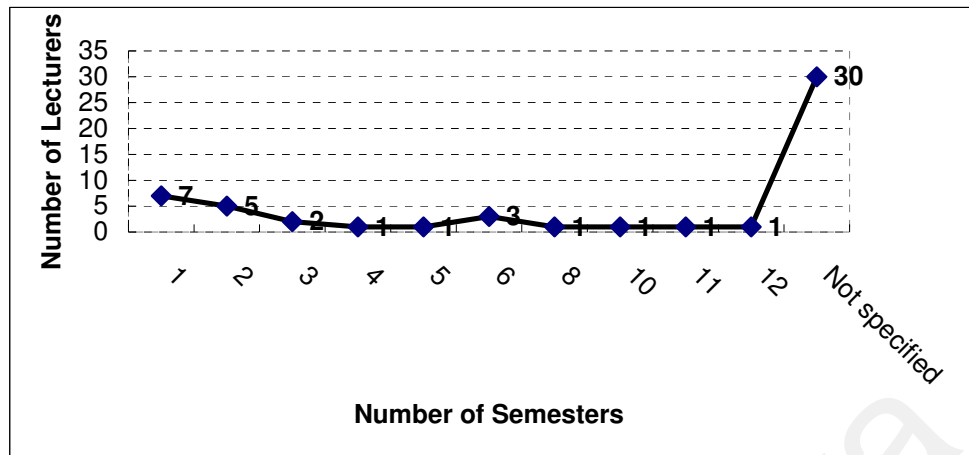
Figure 3.37: Number of Lecturers who Taught SAD

In Figure 3.37, twenty-three respondents had taught SAD before. Thirty respondents did not teach SAD. Seven respondents taught SAD for one semester. Five respondents taught SAD for two semesters. One respondent has taught SAD for twelve semesters.

Responses for question 2: Which of the CASE tools do you use in your course?

Actually, there are two sub-questions here. One is to ask the CASE tools used in this course and another one is to ask lecturers to give the resources for teaching CASE tools.
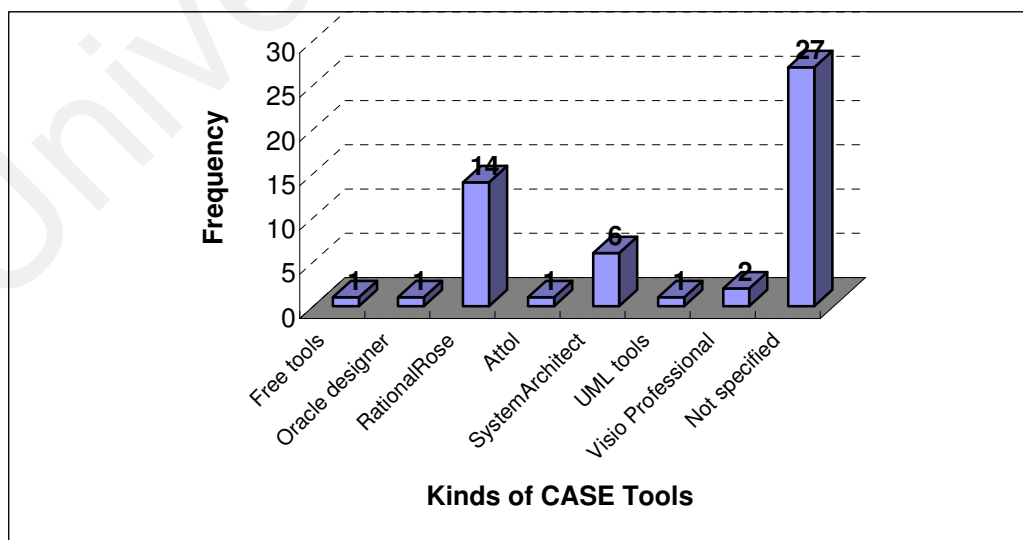

Figure 3.38: CASE Tools Used in Software Engineering

Figure 3.38 shows that there are twenty-seven respondents who did not use any CASE tools in Software engineering. Fourteen respondents used RationalRose in Software Engineering. Six respondents used SystemArchitect, and two respondents used Visio Professional in Software Engineering. Four respondents chose other four tools. The researcher concluded that RationalRose and SystemArchitect were commonly used tools in Software Engineering.
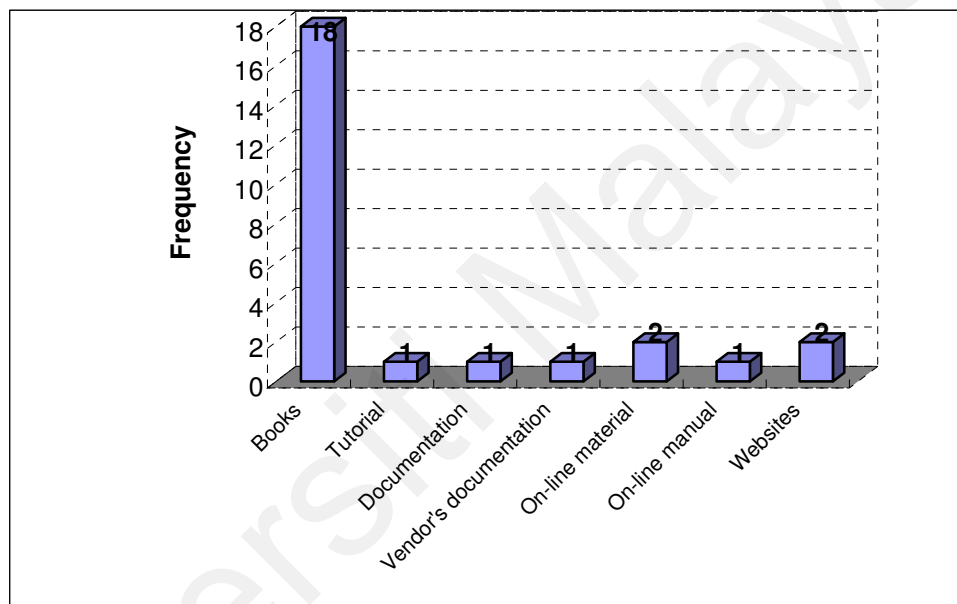

Figure 3.39: Resources for Teaching CASE Tools Used in Software Engineering

Study resources are important for both lecturers and students. There are many resources, which are available for students' learning. The respondents listed out seven different resources, which are shown in Figure 3.39. Among these resources, eighteen respondents selected books for teaching CASE tools in software Engineering. The books were the most important resource for respondents to teach CASE tools.
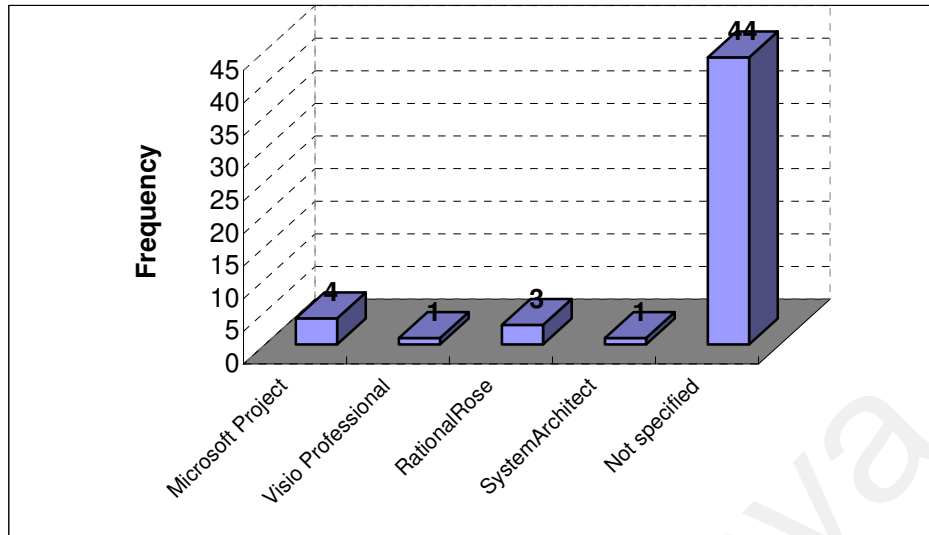
Figure 3.40: CASE Tools Used in Project Management

In Figure 3.40, nine respondents used CASE tools in Project Management; forty-four respondents did not specify any tools in Project Management. Four respondents preferred using Microsoft Project, and three respondents chose RationalRose. One respondent used Visio Professional and another one used SystemArchitect. Most of the respondents who taught Project Management did not use any tool.
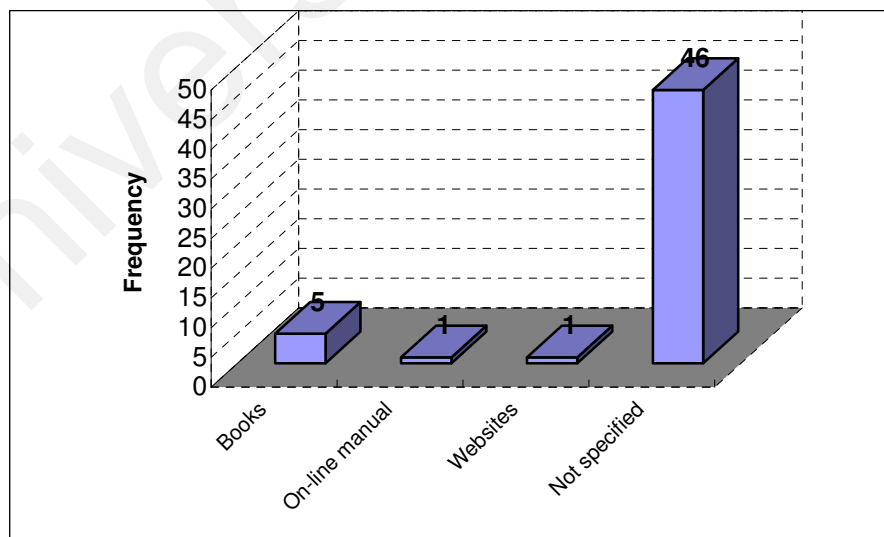

Figure 3.41: Resources for Teaching CASE Tools Used in Project Management

In Figure 3.41, only six respondents utilized resources to teach the tools in Project Management. Five respondents just used books as main resource to teach CASE tools in Project Management. One respondent chose on-line manual and one respondent used websites to search for information. Therefore, the researcher can say that books are the main study resource in teaching CASE tools in Project Management.
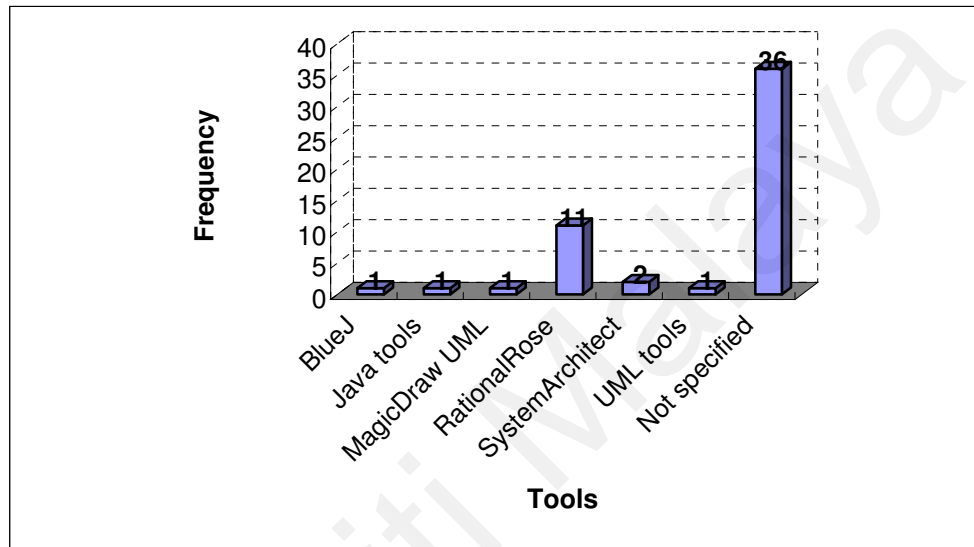

Figure 3.42: CASE Tools Used in OOT

There are more than six alternatives for teaching CASE tools in OOT shown in Figure 3.42. Thirty-six did not use any CASE tool in OOT. Eleven respondents used RationalRose to teach OOT. Two respondents used SystemArchitect in OOT.
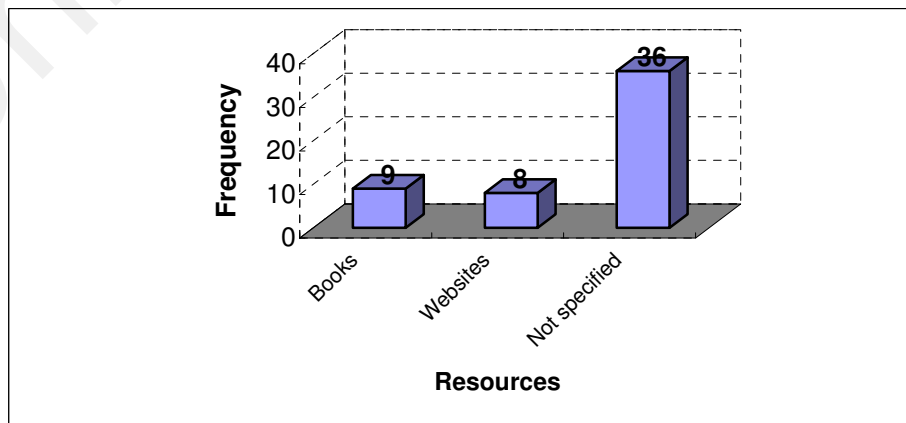

Figure 3.43: Resources for Teaching CASE Tools Used in OOT

In Figure 3.43, there are nine respondents who used books as the main resource to teach CASE tools in OOT. Eight respondents chose websites as the main resource. Books were the first option to teach CASE tools in OOT.
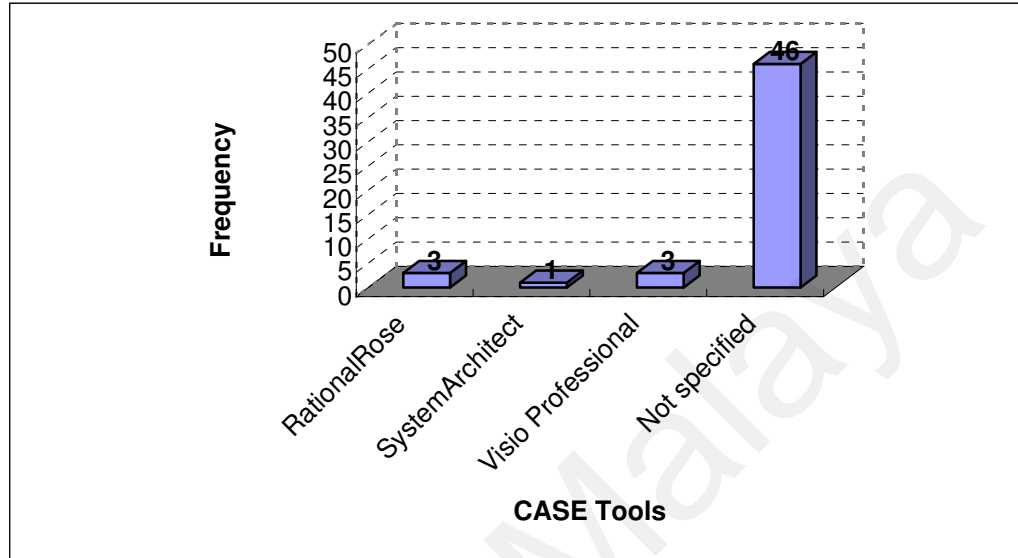

Figure 3.44: CASE Tools Used in Database

In Figure 3.44, seven respondents used CASE tools in Database; three respondents chose RationalRose, three respondents used Visio Professional and one respondent used SystemArchitect. From Figure 3.44, it appears RationalRose and Visio Professional that are the most common tools in teaching Database.
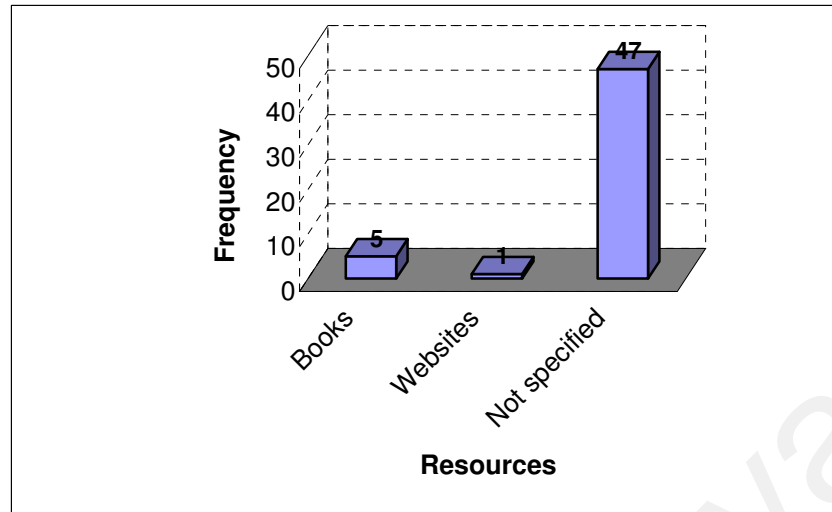
Figure 3.45: Resources for Teaching CASE Tools Used in Database

It can be seen in Figure 3.45 that books were the most important resource for teaching CASE tools in Database. Only one respondent chose websites to search for information. Forty-seven respondents did not define any resource.
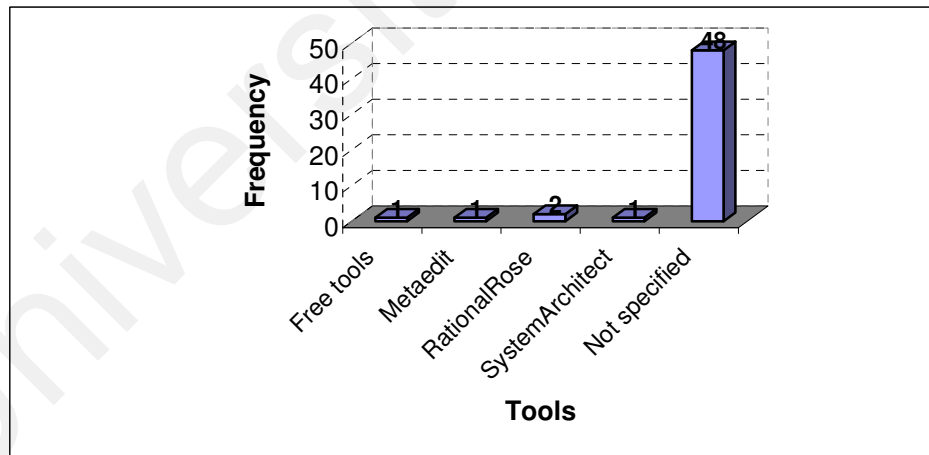
Figure 3.46: CASE Tools Used in User Requirements Engineering

Figure 3.46 shows only five respondents out of fifty-three used CASE tools in User Requirements Engineering. Among the five respondents, two respondents selected RationalRose to teach User Requirements Engineering. One respondent chose free tools.

Here, the respondent did not give the specific names of the free tools. One respondent used

Metaedit and one respondent selected SystemArchitect.
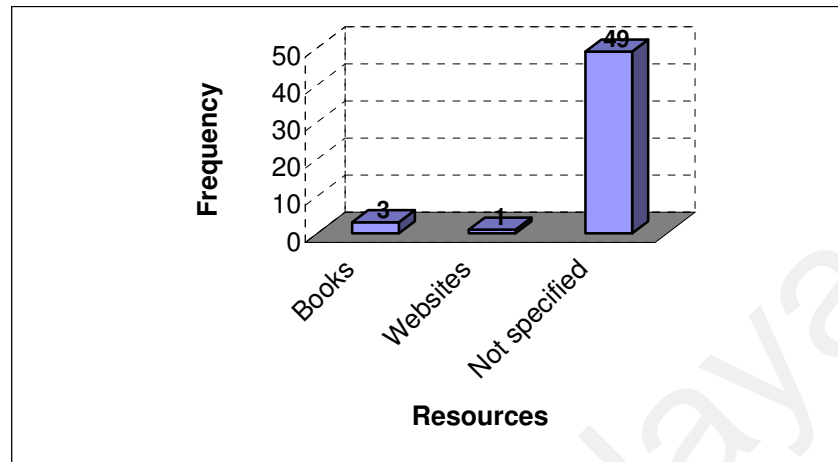
Figure 3.47: Resources for Teaching CASE Tools Used in User Requirements Engineering

In Figure 3.47, three respondents chose books as teaching resource. Only one respondent chose websites to search for information. Forty-nine respondents did not indicate any resource to teach CASE tools.
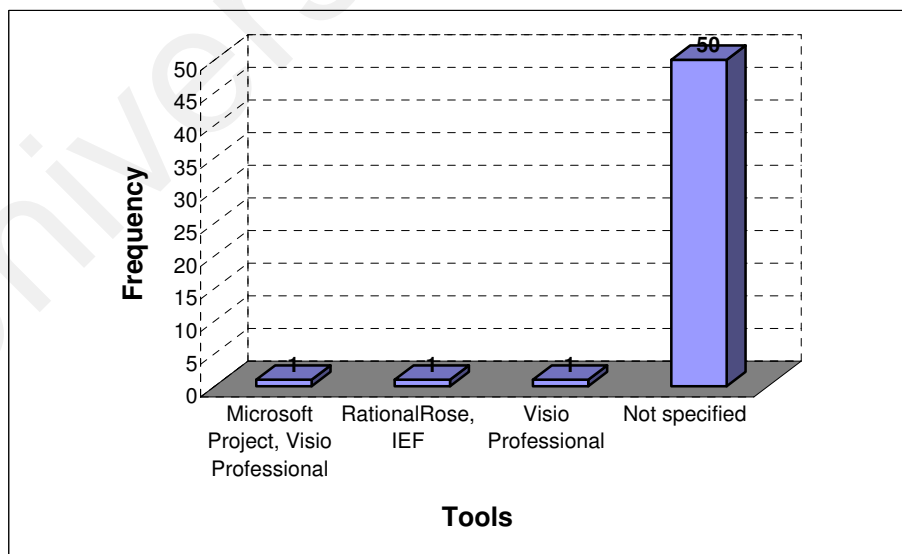
Figure 3.48: CASE Tools Used in Project Development

In Figure 3.48, only three respondents used CASE tools in Project Development. One respondent used Microsoft Project and Visio Professional to teach Project Development. One respondent used RationalRose and IEF to teach Project Development. Fifty-three respondents did not give any name of tools.
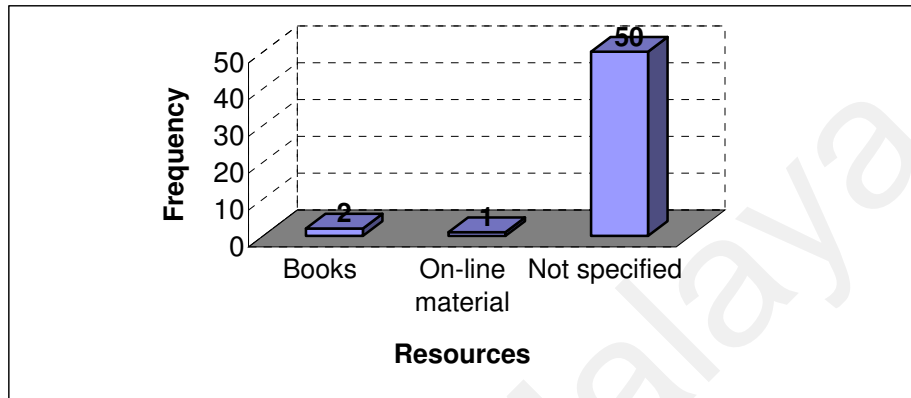
Figure 3.49: Resources Used for Studying CASE Tools in Project Development

Figure 3.49 shows only two respondents used books to teach CASE tools in Project Development. One respondent used on-line materials to teach CASE tools. Fifty respondents out of fifty-three did not provide resources for teaching CASE tools in Project Development.
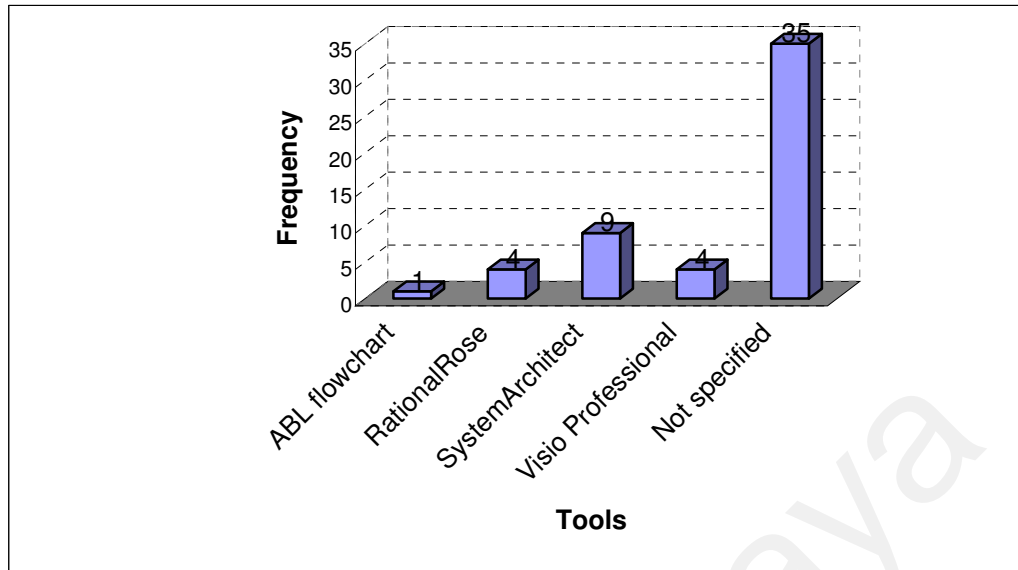
Figure 3.50: CASE Tools Used in SAD

In Figure 3.50, one respondent used ABL flowchart to teach SAD; four respondents used RationalRose to teach SAD; nine respondents used SystemArchitect and four respondents used Visio Professional. Thirty-five respondents did not use any CASE tool in SAD. SystemArchitect is mostly used in SAD.
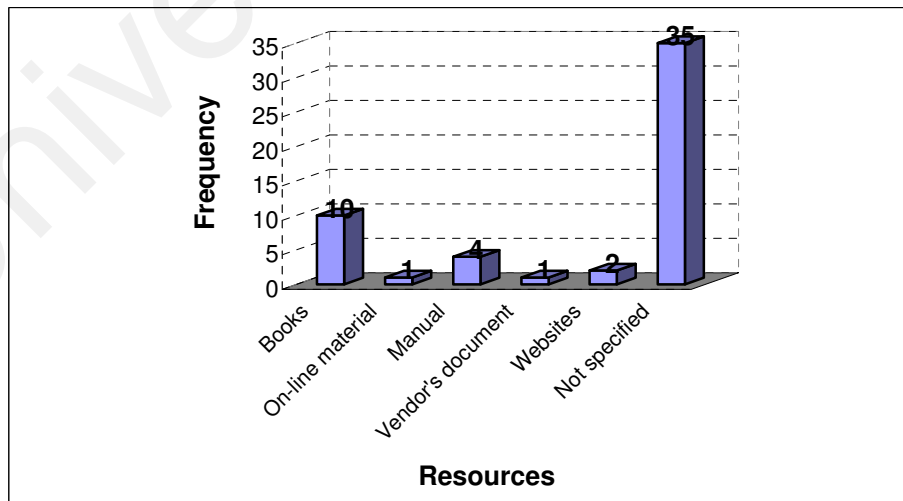
Figure 3.51: Resources for Teaching CASE Tools Used in SAD

In Figure 3.51, eighteen respondents used various kinds of resources to teach CASE tools in SAD. Among the eighteen respondents, there were ten respondents who used books as their main resources to teach CASE tools. Four respondents used on-line manual for teaching purposes in SAD; two respondents searched websites to get information about CASE tools; one respondent used on-line material and another respondent used vendor's documentation for teaching CASE tools.

The researcher summarized the main CASE tools that respondents used to teach their subjects in Figure 3.52.
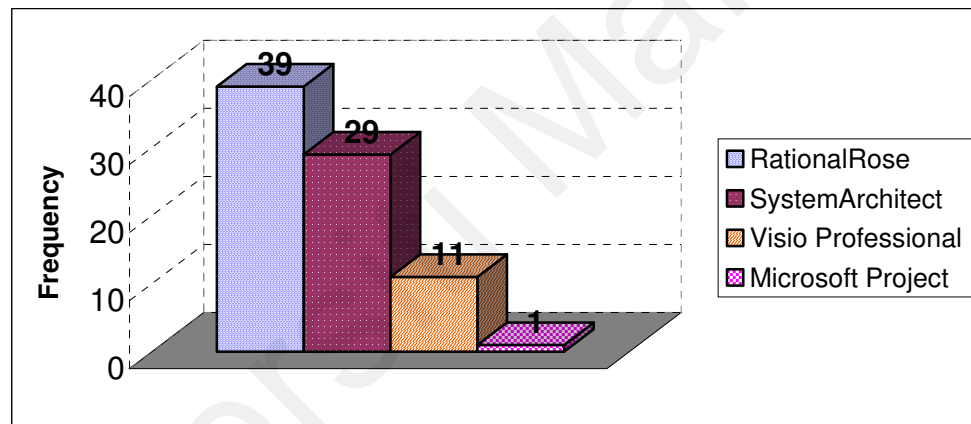

Figure 3.52: Usage of Main CASE Tools Used in Subjects

In Figure 3.52, thirty-nine respondents used RationalRose in their subjects; twenty respondents used SystemArchitect; eleven respondents chose Visio Professional; one respondent used Microsoft Project in their subjects. RationalRose is the most commonly used CASE tool in teaching subjects and then is followed by SystemArchitect.

Respondents gave many resources as shown in Figure 3.53. In this figure, books are most used resource to teach CASE tools followed by websites and on-line material. Six

respondents chose Vendor's documentations. Here, it is quite hard to define or give the exact definition of resources such as on-line material and websites for respondents.
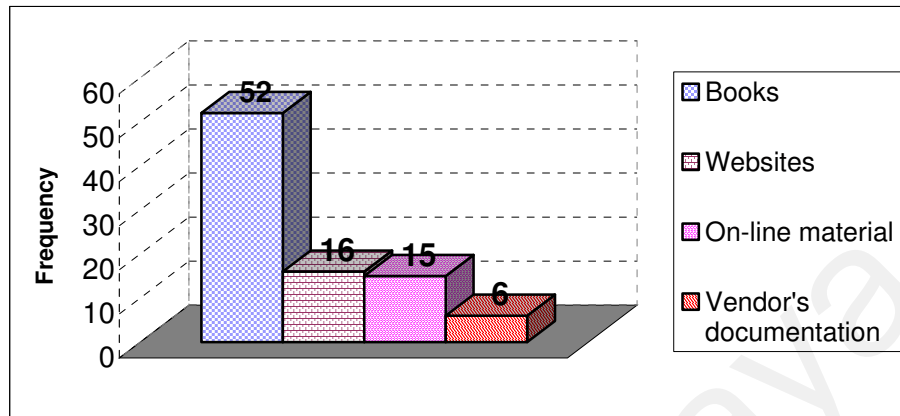
Figure 3.53: Resources for Teaching CASE Tools

Resources are very helpful for the lecturer. In Figure 3.53, there are fifty-two respondents who used books as the main resource to teach CASE tools. Sixteen respondents used websites to get information. Fifteen respondents used on-line material to teach CASE tools. Actually, it is not a good practice for the respondents only to use the books or websites to teach the CASE tools. Lecturers ought to use both books and websites.

Responses for question 3: Do you agree that teaching using CASE tools is necessary?
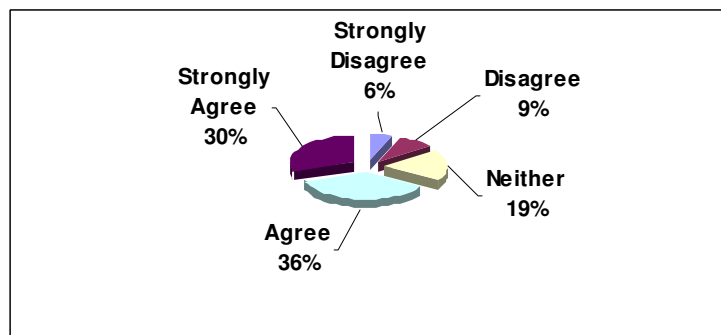
Figure 3.54: Necessity of Teaching CASE Tools

The researcher asked about necessity of teaching CASE tools in colleges and universities. In Figure 3.54, fifty-three respondents gave their comments to the teaching of the CASE tools. 30% of respondents strongly agreed that it is necessary to teach the CASE tools. 36% of respondents out of fifty-three agreed that it is necessary to teach the CASE tools. 19% of respondents were neutral. 9% of respondents agreed that it is not necessary to teach the CASE tools, and only 6% of respondents strongly disagreed that it is not necessary to teach the CASE tools. 66% of respondents agreed that it is necessary to teach the CASE tools.

Responses for question 4: Please elaborate on your responses to question 3.

Some respondents explained that through using the CASE tools, students could have better understanding on the CASE tools and methodologies. Others said that the students benefited from learning the CASE tools because using the CASE tools was a requirement from IT organizations. Therefore, the respondents agreed that teaching the CASE tools was necessary.

Responses for question 5: How much time do you spend on using the CASE tools in each subject?

In question 5, the researcher wanted to know the period of teaching CASE tools in subjects given by the researcher. Software Engineering is a core subject to IT students. Different educators set different time durations based on the syllabus of subjects. In Figure 3.55, there are twenty-seven respondents who spent at least one hour in teaching CASE tools in Software Engineering.
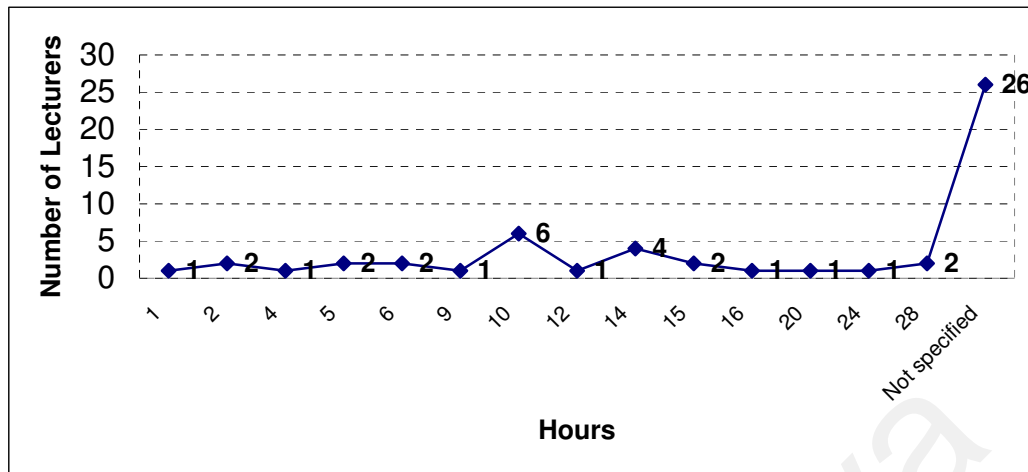
Figure 3.55: Hours Spent Teaching in Software Engineering

Twenty-six respondents did not specify the time of teaching the CASE tools in Software Engineering. Six respondents spent ten hours on teaching the CASE tools in Software Engineering. This is the longest time to teach the CASE tools compared to other group of respondents. Four respondents spent fourteen hours to teach the CASE tools in Software Engineering. The researcher found that there were seven respondents who specified hours ranged from one to twenty-eight hours. Different respondents have different time schedules of teaching the CASE tools in Software Engineering. There is no standard to identify how much time they should spend in teaching CASE tools in Software Engineering.
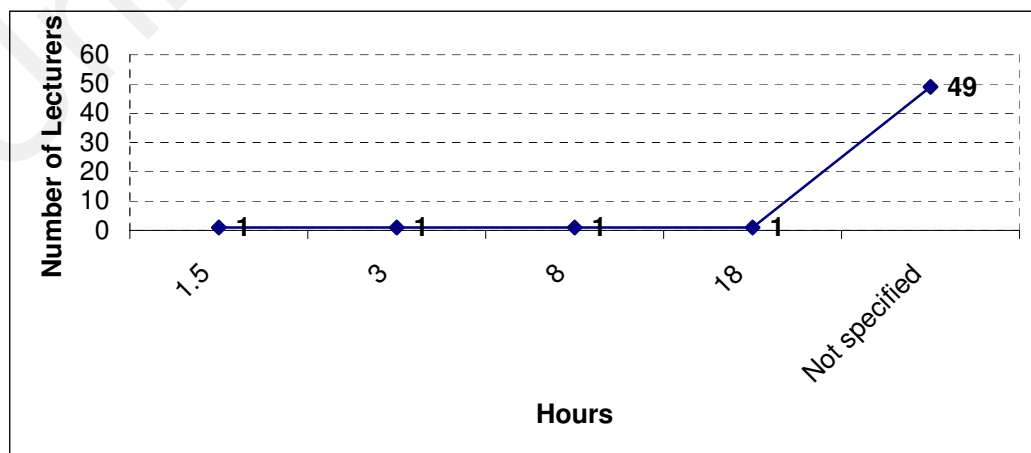
Figure 3.56: Hours Spent Teaching in Project Management

In Figure 3.56, there are only four respondents who taught Project Management. Forty-nine respondents did not define the hours of teaching in Project Management. One of the respondents spent eighteen hours to teach CASE tools in Project Management. One the respondent used eight hours to teach the CASE tools in Project Management.
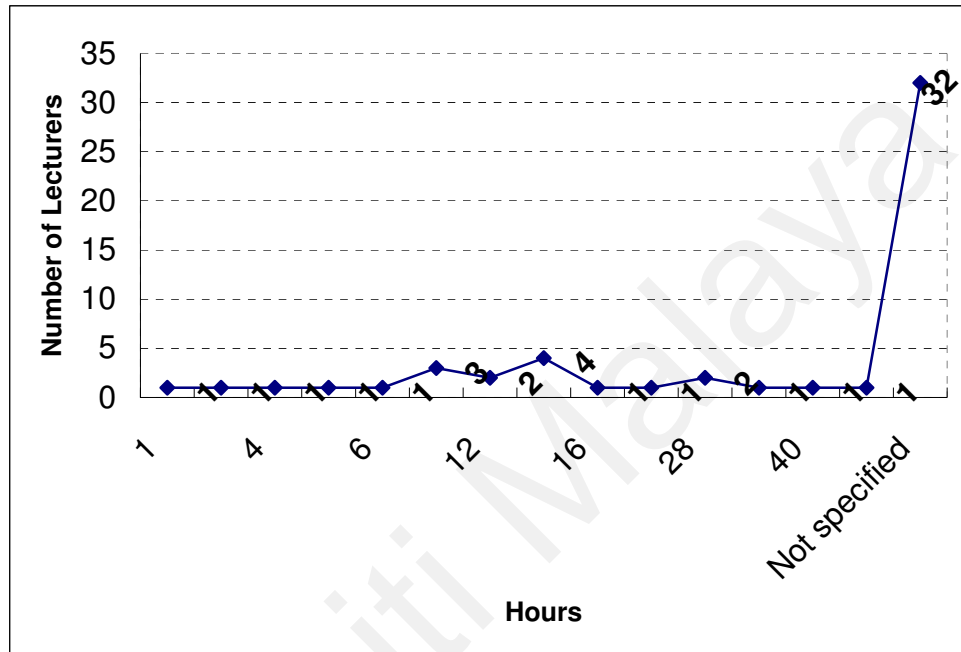
Figure 3.57: Hours Spent Teaching CASE Tools in OOT

More and more colleges and universities offered OOT to students. Students are given more practice using CASE tools in OOT. In Figure 3.57, twenty-one respondents spent certain time teaching CASE tools in OOT. Four respondents spent twelve hours teaching CASE tools in OOT, and three respondents spent ten hours teaching CASE tools in OOT. Among twenty-one respondents, fourteen respondents spent two or less than two hours teaching CASE tools in OOT. Thirty-two respondents did not allocate their time into CASE tools in OOT.
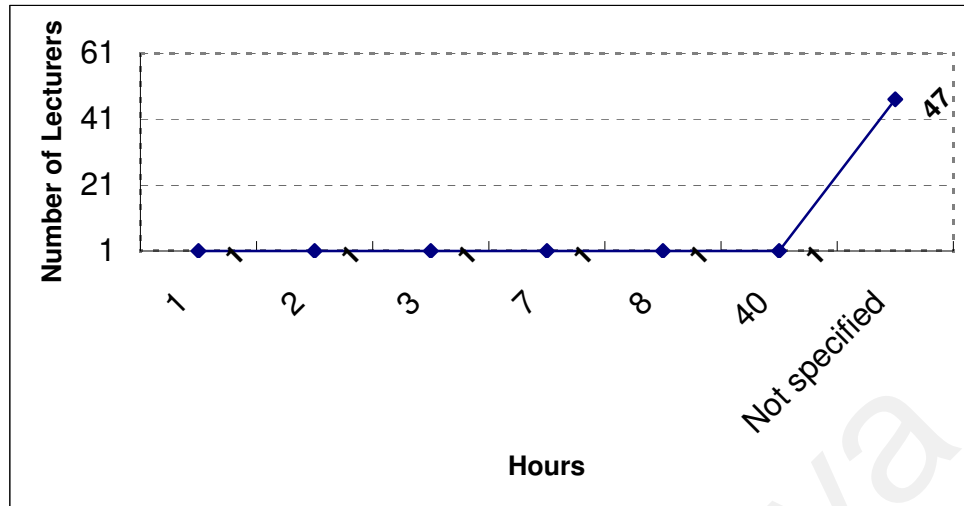
Figure 3.58: Hours Spent Teaching CASE Tools in Database

CASE tools are used in designing a Database. CASE tools help students to design entities and classes in Database design. In Figure 3.58, six respondents used CASE tools in Database with a range from one to forty hours.
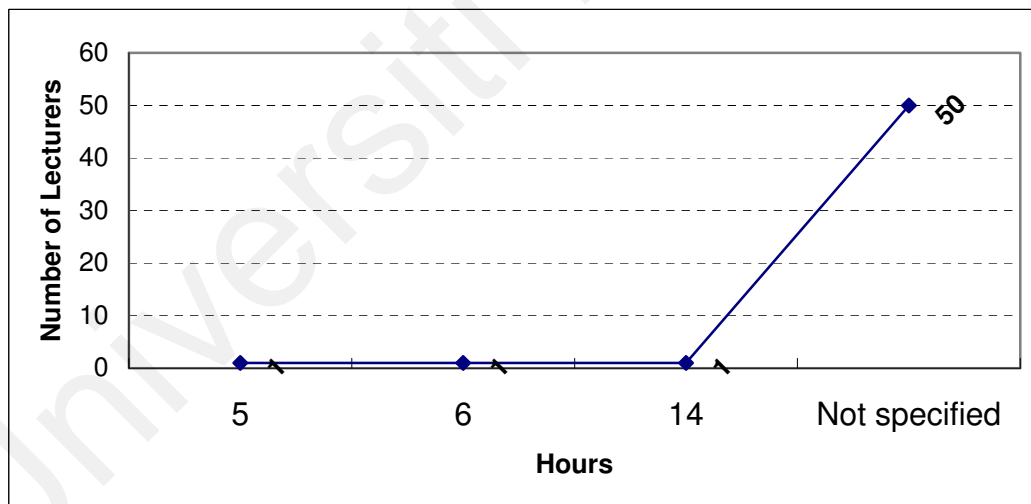
Figure 3.59: Hours Spent Teaching CASE Tools in User Requirements Engineering

According to requirements from IT organizations, Software developers commonly used CASE tools to analyze the user requirements. In teaching places as shown in Figure 3.59, the researcher found that there were three respondents who used the CASE tools in User Requirements Engineering in this studies. One respondent spent fourteen hours in

teaching the CASE tools in User Requirements Engineering and other two spent five and six hours in teaching the CASE tools.
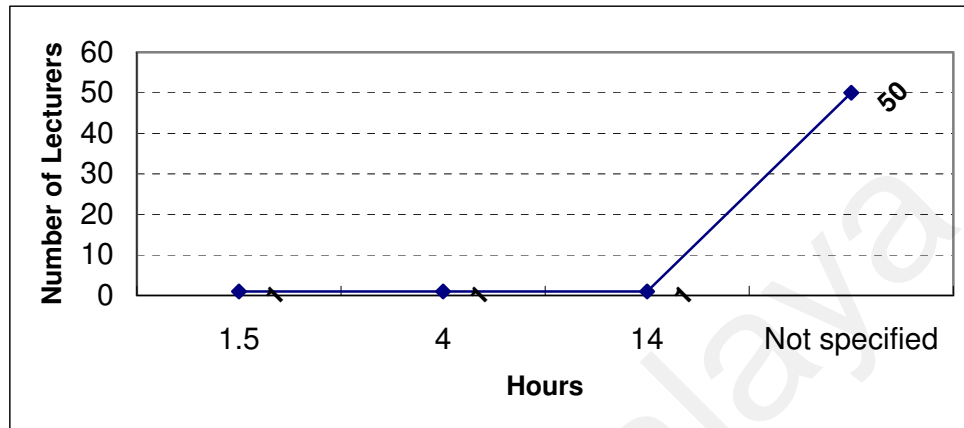
Figure 3.60: Hours Spent Teaching CASE Tools in Project Development

Project Development involves many areas such as analysis, design and implementation. CASE tools can be implemented into many phases to help developers to improve work effectively. Lecturers encourage students to use CASE tools in their assignments and projects. In Figure 3.60, one respondent spent one and half hours to teach CASE tools; another respondent spent four hours to teach CASE tools; one respondent spent fourteen hours to teach CASE tools. Fifty respondents out of fifty-three did not specify how many hours they spent in CASE tools in Project Development.
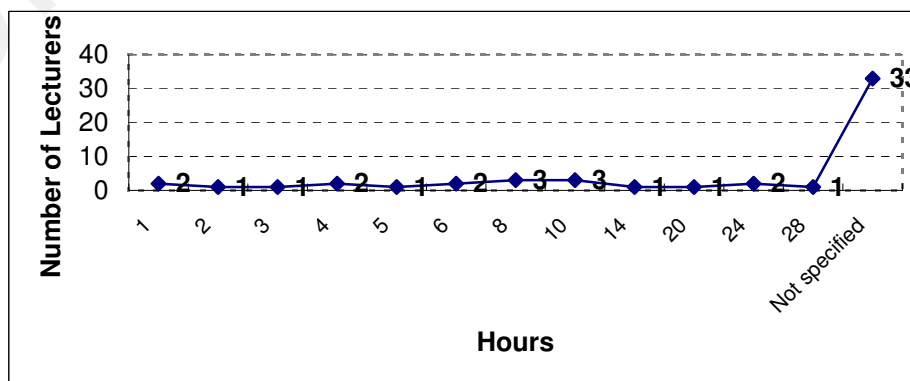
Figure 3.61: Hours Spent Teaching CASE Tools in SAD

SAD is a core subject in an IT course and is to give students a basic understanding of software development process. Most educators provided this subject for the students. Some colleges and universities used CASE tools in SAD. In Figure 3.61, there are twenty respondents out of fifty-three taught the CASE tools in SAD. Three respondents taught the CASE tools for fourteen hours followed by three respondents with fourteen hours in SAD. One respondent taught the CASE tools in SAD for twenty-eight hours. The rest of the respondents spent hours ranging from one to twenty-four hours.

Table 3.1: Hours Spent in Subjects

Subjects with spent hours	N	Minimum	Maximum	Mean	Std Deviation
OOT	21	1	42	**16.00**	11.70
Database	6	1	40	10.16	14.87
User Requirements Engineering	3	5	14	8.33	4.93
Project Development	3	1	14	6.5	6.61
SAD	20	1	28	9.8	8.1
Software Engineering	27	1	28	**11.63**	7.21
Project Management	4	1	18	7.62	7.45
Valid N	0				

The researcher has summarized the hours spent in subjects in the table above. From Table 3.1, it can be seen that mean of hours spent in OOT is 16,which is highest mean hours spent in all subjects followed by eleven hours spent in Software Engineering. Therefore, the researcher concluded that the lecturers spent more than ten hours teaching using CASE tools in OOT and Software Engineering.

Responses for question 6: How well are you aware of the industry's requirements for using CASE tools?
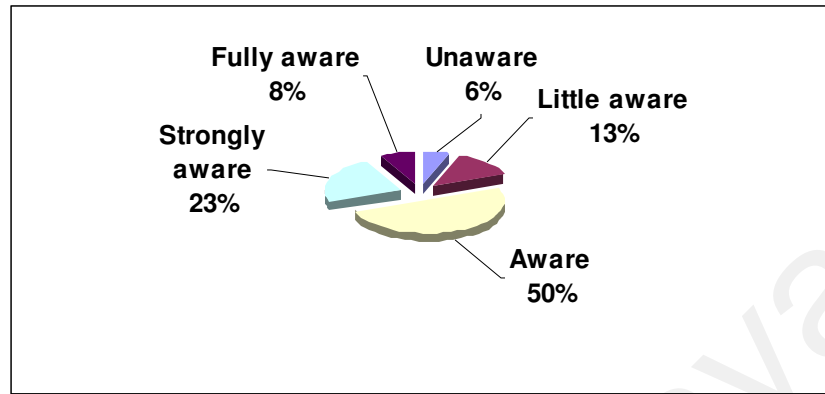
Figure 3.62: Awareness of Using CASE Tools

CASE tools have been used for many years in IT organizations. In some organizations, having skills on how to use the CASE tools is necessary. Some colleges and universities have realized the importance of the CASE tools in software development. In Figure 3.62, 8% of respondents were fully aware of usage of the CASE tools in IT organizations. 23% of respondents were strongly aware of the importance of using the CASE tools in IT organizations. 50% of respondents were aware of importance of using the CASE tools required from IT organizations. 13% of respondents had little awareness of the importance of using the CASE tools in IT organizations. Only 6% of respondents were not aware of using the CASE tools in industrial line. Generally, the researcher can conclude that 81% of respondents have awareness on the importance of using the CASE tools in IT organizations.

Responses for question 7: Is it easy to teach students how to use CASE tools?

In this question, the researcher evaluated the usage on kinds of CASE tools. In Figure 3.63, the researcher summarized the two main tools, which were RationalRose and SystemArchitect in colleges and universities. Twenty-two respondents agreed that RationalRose was fairly easy. Ten of respondents said that RationalRose was very easy. Six respondents out of forty-three commented that RationalRose was easy and four respondents said that RationalRose was not very easy. Ten respondents did not define the ease of usage of RationalRose.

Thirty-two respondents did not give their comments on SystemArchitect. Six respondents out of twenty-one said that SystemArchitect was easy, followed by six respondents with fairly easy comments. Seven respondents commented SystemArchitect was very easy to use. One respondent agreed that SystemArchitect was not very easy; another respondent thought SystemArchitect was extremely easy to use.
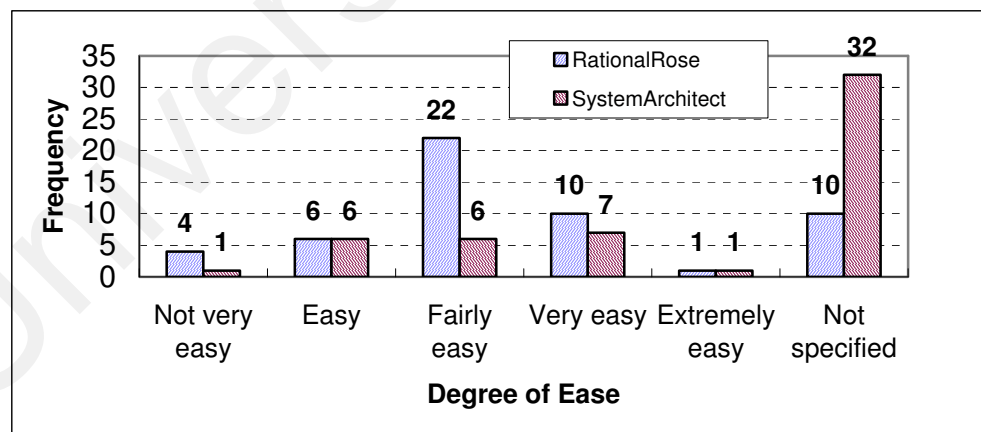
Figure 3.63: Ease of Usage of CASE Tools

Responses for question 8: Do you think that you have sufficient time to deliver the knowledge of CASE tools to students?

Some factors affected the effectiveness of teaching CASE tools in colleges and universities. Generally, time is one of main factors to teach the CASE tools for lecturers. In Figure 3.64, 8% of respondents said that the time of teaching the CASE tools was insufficient at all. 28% of respondents commented that the time was not very sufficient; 45% of respondents agreed that the time was sufficient. 13% of respondents said that the time was very sufficient and 4% of respondents said that the time was extremely sufficient.
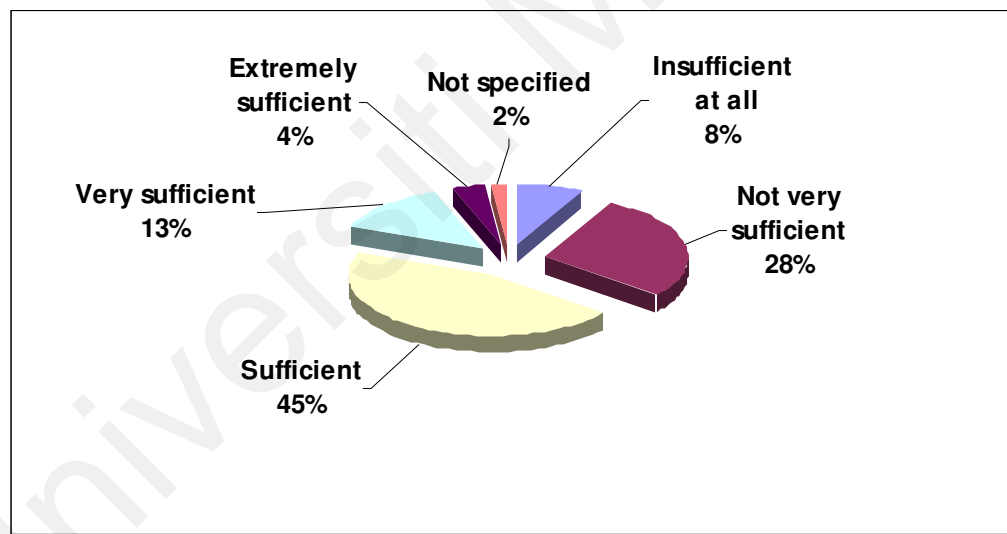
Figure 3.64: Sufficiency of Time

One respondent did not specify any scale. Therefore, the finding of this question was that 36% of respondents agreed that the time was not sufficient to teach the CASE tools and more than 62% of respondents were satisfied with the time for teaching the CASE tools.

Responses for question 9: In your opinion, what are the basic requirements to teach CASE tools to students?
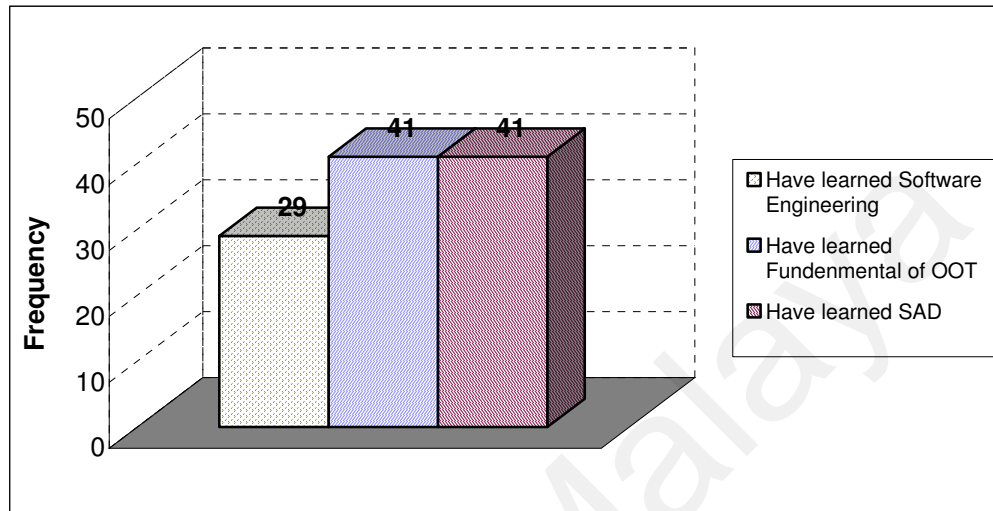
Figure 3.65: Basic Requirements to Learn CASE Tools

To make sure that students have good understanding about CASE tools and apply into working places later, the student must have basic knowledge before learning the CASE tools. In this survey, lecturers were asked to give suggestions on some areas. In Figure 3.65, forty-one respondents said that having knowledge on OOT and SAD was necessary. Twenty-nine respondents said that having knowledge in Software Engineering was necessary to learn the CASE tools. In general, most of the respondents agreed that having knowledge on OOT was a basic requirement to learn the CASE tools, followed by SAD and Software Engineering.

Responses for question 10: Do you think that universities or colleges have provided the necessary infrastructures to teach CASE tools?
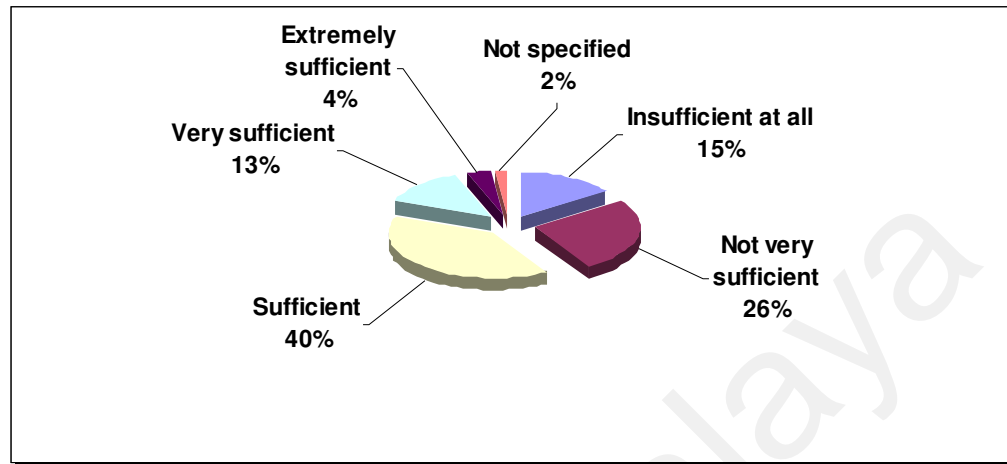
Figure 3.66: Sufficiency of Infrastructure

Infrastructure is one of factors that enable the teaching of the CASE tools for lecturers. Enough seats in labs and PCs will ensure that students can fully utilize the resources for practising the CASE tools. Figure 3.66 shows 40% of respondents agreed that the infrastructure is sufficient to teach CASE tools. 4% of respondents agreed that infrastructure was extremely sufficient; 13% of respondents said that infrastructure was very sufficient; 26% of respondents out of fifty-three indicated that the infrastructure was not very sufficient and 15% of respondents said that the infrastructure was not sufficient at all. 2% of respondents did not give comment on this. Therefore, the conclusion can be made that 57% of respondents agreed that the infrastructure was sufficient and 41% of them commented that the infrastructure was not sufficient to teach the CASE tools.

Responses for question 11: What are your expectations of students' ability in using CASE tools after graduating from the universities or colleges?

In this survey, regarding the students' skills, some of the lecturers hoped that the students would be able to have basic understanding in CASE tools and use them to handle the tasks in working places. Some lecturers agreed that the students should have skills in SDLC.

Response for question 12: What problems did you have in teaching CASE tools?

Most of the respondents commented that the time was a main problem. Some of the respondents agreed that the cost of CASE tools could limit the use of CASE tools.

Response for question 13: what are your suggestions in improving students' skills in using CASE tools?

Some respondents suggested that the vendors should provide an educational version of CASE tools with low prices. Some respondents argued that the colleges and universities should provide adequate infrastructure to teach CASE tools. Most of the respondents expected that more practical assignments and projects should be given to students.

3.5.1 Discussion of Some Factors Found from the Different User Groups

The researcher analyzed each question in the previous sections. The researcher found that some other areas should be discussed. In order to give readers a deeper understanding about the usage of CASE tools, in this section, the researcher wanted to discuss some critical factors, which are involved in three groups of people.

1) Time of Learning CASE Tools

   In this research, the researcher found that the time of learning CASE tools was a main concern from lecturers and students. The students said the time of learning

CASE tools was a main limitation compared with other factors. Based on the finding from previous sections, Figure 3.67 indicates that more than 64% of respondents said that they did not have enough time to learning the CASE tools. However, the lecturers in this survey said that 62% of lecturers were agreed that the tine of teaching the CASE tools was sufficient. There is a quite difference between student's perceptions and lecture's perceptions. The researcher suggested that educators should set reasonable time schedules for teaching the CASE tools based on the complexity of the CASE tools.
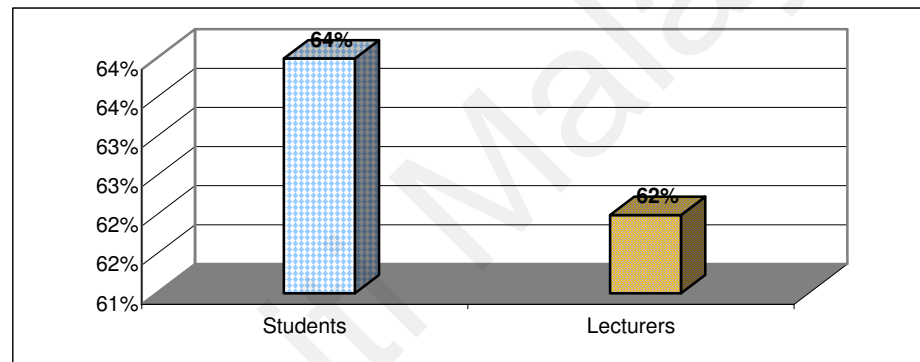
Figure 3.67: Comparison of Time Factors from Students and Lecturers

2) Lecturer's Experience in Teaching Subjects

The lecturer's experience is a critical factor for students. The researcher made an analysis on the lecturer's experience in teaching some main subjects in this research. Table 3.2 indicates that fifty-three lectures were asked in this research. The highest mean in Table 3.2 is 4.82. Most probably, the lectures had more experience in teaching Software Engineering. But most of the lecturers only taught less than five semesters. The lowest mean is 3.26. The lecturers did not have much more experience in teaching Database than other subjects. Therefore, the conclusion can be made that the lecturers had experience in teaching some core

subjects. Based on the finding in this research, the research suggested that lecturers should have more opportunities to obtain experiences.

Table: 3.2: Lecturer's Experience in Teaching Subjects

	N	Minimum	Maximum	Mean	Std. Deviation
Lecturers	53	1	53	27.00	15.443
Semesters Taught in Software Engineering	**28**	**1**	**40**	**4.82**	**7.454**
Semesters Taught in Project Management	7	1	10	3.71	3.302
Semesters Taught in OOT	31	1	20	3.94	3.941
Semesters Taught in Database	**23**	**1**	**8**	**3.26**	**2.359**
Semesters Taught in User's requirements	7	1	11	3.71	3.352
Semesters Taught in Project Development	11	1	10	4.09	2.773
Semesters Taught in SAD	23	1	12	3.96	3.457
Valid N (listwise)	0				

3) Purposes of Using CASE Tools

According to the study, the researcher found that basically, the students were familiar with most of the features of CASE tools, which were used in IT companies.

Table 3.3: Features Used in Two Different Groups

Features of CASE Tools	IT Professionals	Students
Analysis of Risk	3	
Cost Estimation	2	3
Generation of Documentation	4	2
Diagrams	1	1

1: Most often used resources, 5: Least often used resources

Table 3.3 indicates that in this study, the IT professionals used some features of the CASE tools in software development. On the other side, the students used the similar features in their study. Therefore, the conclusion can be made that the students should be able to use the basic features of the CASE tools.

4) Resources of Learning CASE Tools

After having analyzed the result collected from students, lecturers and IT professionals, the researcher found that there were much difference between students and IT professionals in term of resources of learning CASE tools. In Table 3.4, the IT professionals have more alternatives to search for information about case tools. However, most of the students learned CASE tools by reading textbooks given by their lectures.

Table 3.4: Resources of Learning CASE Tools

Resources	IT professionals	Students
Books	5	1
CASE tool community websites	2	2
Manuals of CASE tools	1	
Forums of CASE tools	3	
Training courses	4	

1: Most often used resources, 5: Least often used resources

From Table 3.4, it can be seen that the most often used resource was the manuals of CASE tools. Second resource was the CASE tool's communities. Compared with students, the IT professionals were able to learn the CASE tools faster than the students. The most of CASE tools vendors have their own different methodologies to present the software development process. The most effective and flexible way to learn the CASE tools is to learn their manuals. However, the books are not able to give users a specific guideline for a particular CASE tool. Compared to the student's learning approach, the IT professionals will be able to have more resources and much help in learning the CASE tools. To improve the skills of using the CASE tools, the researcher suggested that:

1. Students should improve learning skills and know how to explore the information about CASE tools through Internet.

2. Lecturers should give sufficient help in learning the CASE tools.

3. More ways should be used to learn CASE tools.

3.6 Conclusion

In this chapter, the researcher analyzed all data collected from interviewees and respondents.

In section A, the researcher asked some questions about the background of IT organizations, comments to students' skills, suggestions about using CASE tools and CASE tools' utilization in their companies. Through interviewing the IT professionals, the researcher first found that most of the participants believed that CASE tools were very helpful in software development in terms of better communication, standardization, faster coding and documentation generation. Second, interviewees also pointed out the issues and concerns about using CASE tools such as cost of CASE tools, extra training cost, complexity of CASE tools and people's skills. Third, the IT professionals commented that as fresh IT graduates, they should know some basic concepts of OOT and SAD, have basic skills in using CASE tools. These are really practical and relevant to industrial requirements. Fourth, most of the interviewees were confident that more and more IT organizations would use CASE tools in future.

In section B, the researcher distributed questionnaires to students. After analyzing the collected data, the research discovered some facts about using CASE tools in colleges and universities. First, most of the respondents have basic knowledge in OOT and SAD. The respondents also applied the OOT and SAD in their assignments and projects frequently. Second, the majorities of the respondents have learnt CASE tools before, and did believe that CASE tools were helpful in improving software skills. They used CASE tools to

analyze the user requirements and draw diagrams. Third, the main tools implemented in colleges and universities were RationalRose and SystemArchitect. Other tools were seldom used. Fourth, more than half of the respondents were not really satisfied with the infrastructure and lecturer's guidance. Finally, the respondents commented that time, learning approach and complexity of CASE tools were factors that affected the CASE tools learning and expected that lecturers should give them more assignments and projects that were closed to real businesses.

In section C, the researcher investigated the lecturers who used CASE tools to teach some subjects. After completing the investigation, the researcher found that most of the respondents had enough teaching experience in some subjects using CASE tools, and they have realized the importance of using CASE tools in industrial environment. Most of the respondents were satisfied with the time and infrastructure regarding teaching CASE tools in their colleges and universities. The researcher discovered that there was a difference among the respondents in terms of teaching time. Furthermore, the respondents pointed out that the cost of CASE tools, ease of tools, lab space and lack of resources are main issues and problems in implementing CASE tools. Finally, they suggested giving students more assignments and hands-on practice.

To get more comprehensive analysis, in the final section, the researcher analyzed the three groups with a multi-dimension approach. Through comparing and analyzing the three groups people, the readers are able to clarify some certain issues in terms of time, learning resources and features used in different environments.

Based on the finding in this chapter, in the next chapter, the researcher will develop a collaborative system to enable the CASE tools users and vendors to share the CASE tools knowledge and promote the usage of CASE tools.

Chapter Four – The Development of FOCT

After analyzing data collected from respondents described in chapter three, the researcher found that it was necessary to provide a collaborative system for CASE tool users. This chapter describes the development of a system called Forum of CASE tools (FOCT). It allows CASE tool users to share knowledge and information about the usage of CASE tools.

4.1 Introduction

Firstly, this chapter describes the system overview, system objective, system scope, users of the system, system requirements, system installation, supporting languages, system design and how the system works. Then the researcher continues to discuss the system methodology, software requirements and installation, system design, database design, interface design, system development, system testing, user manual and finally the limitations of the system.

4.2 System View

FOCT is a web-based application. The features of FOCT are based on the findings described and documented in chapter three. FOCT is designed specifically for the following five groups of people: IT Professional, lecturer, student, vendor and system administrator. This is shown in Figure 4.1.

FOCT has some useful and helpful features for these five groups of people. These features were developed to let CASE tool users figure out the issues and problems described in chapter three, and to share their knowledge of CASE tools. Each group has different password for system access. Firstly, the students can evaluate the usage of case

116

tools in their colleges and universities. They can further post their skills in using CASE tools. The IT professionals can post comments on their skills in using CASE tools. The IT professionals also have rights to post their suggestions from the industrial perspective. The lecturers are given privileges to post messages to the users. Vendors can share CASE tools information and promote the tools by posting messages.
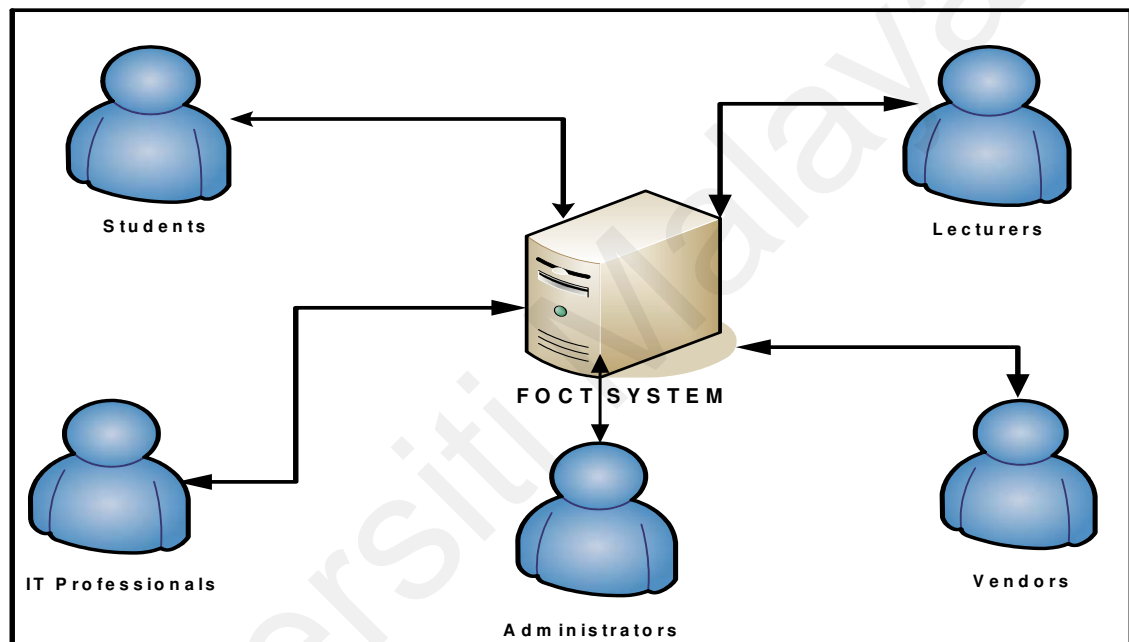

Figure 4.1: An Overview of FOCT System

4.3 System Objectives

Sharing knowledge is necessary for both IT staff and people in colleges and universities to reduce the gap between theory and practice. This system is to establish a platform for CASE tool users from IT organizations and universities in Malaysia. Through using this system, they can interact with each other and improve the skills in using CASE tools. The system objectives are as follows:

    1) Check the results of evaluation of CASE tools usage in universities.

2) Provide helpful news for users.

3) Allow the students to evaluate the effectiveness of learning of CASE tools in terms of the sufficiency of facilities, support of lecturers, and the time allocated for learning.

4) Evaluate the students' skills by posting the skills in using CASE tools.

5) Allow IT professionals to register, edit and evaluate a particular student and add comments on the student.

6) Give CASE tool vendors the capability to post messages to FOCT.

4.4 System Scope

The purpose of this system is to provide a place where students, lecturers, IT professionals and vendors can contribute their knowledge to others. The scope of the system is described in two main areas, namely, application area and user groups.

4.4.1 Application Area

The main scope of this system is the application area as shown in Table 4.1. In this system, there are some sub-applications that link the entire system.

Table 4.1: Application Area of FOCT

Functions	Descriptions
Events	Users can add events
Evaluation of Learning CASE tools	After login successfully, students can evaluate the learning of CASE tools.
Skill Evaluation	Students can post skills and get the results from IT professionals.
Chat room	Users can chat with other users.
Alumni	The IT professional are able to see the schoolmates.
CASE Tools Consultant	Vendors can provide CASE tools and consult services through the system
Job Offers	The IT companies post the job positions on this system and students are able to apply.
Forum	All users can update their records.

4.4.2 User Groups

In this system, the main user groups are IT professionals, lecturers, students, vendors and administrators. They are given password to access certain features based on the level of authority.

1) IT Professionals

   IT professionals are a group of people who use CASE tools in IT organizations. Their comments are valuable for lecturers and students. They perform the following functions: register records, login the system, edit records, evaluate student's skills, post job offers, meet alumni and update own records.

2) Lecturers

   Lecturers have authority to access the website to post messages and manage the topics in the chat room.

3) Students

   The students have authority to access certain features. The features are: login, register records, evaluate CASE tools, evaluate their skills, apply jobs, view the access numbers of student's personal information, and edit own records.

4) Vendors

   The vendors of CASE tools have the capability to post messages about CASE tools. This gives them a good opportunity to promote their CASE tools. The vendors can perform functions such as register, login and post messages, provide consultant service and publish the seminar and training courses.

5) Administrator

   The administrator will be able to handle the News posted from both lecturers and vendors.

4.5 Software Requirements

The software required to build the system I shown in Table 4.2

Table 4.2: Software Required in the System Development

Software Names	Description
Tomcat3.2	Tomcat 3.2 is a web server for running JSP.
Mysql	MySQL is open-source database.
JDeveloper	It is used to write java code and compile the code.
Drameweaver	It can be used to write JSP and HTML code.
JDK1.4	JDK is used to compile java code into class files.

➢ Tomcat 3.2

James defined Tomcat as: "The Jakarta Tomcat server is an open source, Java-based Web application container that was created to run servlet and JavaServerPage Web application (James, 2002)". Tomcat is stable and equipped with many features that a commercial web application container has. The researcher used it to run JSP and servlet.

➢ MySQL

MySQL is a very robust relational database. According to Mark (2000) "It provides speed and flexibility that no other database in its class can match". The researcher used it to store the data from the five groups of people.

➢ JDeveloper

JDeveloper is used to write Java code and compile the code in class files. It can support servlet and integrate with MySQL. The researcher used it to write JavaBean.

➢ Dreamweaver

Dreamweaver is a tool that is used to generate JSP source and HTML code. It can be used to design interfaces.

➢ JDK1.4

JDK 1.4 is short form of Java Development Kit. To run Tomcat, the researcher had to install JDK to support Tomcat.

4.6 System Installation

To make sure the FOCT executes successfully, there are six steps that need to be taken.

4.6.1: Download MySQL, Tomcat, JDK and Thirty Party Applications

The first step is to download some necessary software from authoritative websites. In this project, to run FOCT, the two main tools were needed. These downloads are free and open to the public. The researcher also used some open source applications such jchatbox, web calendar to enrich the features in this system. The URLs are listed in Table 4.3.

Table 4.3: URLs of Tools

Names of Software	URLs
JDK 1.4	www.java.sun.com
MySQL	www.mysql.com
Tomcat Web server	http://jakarta.apache.org/tomcat/index.html
Jchatbox	http://www.javazoom.net/jzservlets/jchatbox/jchatbox.html
Web Calendar	http://www.dynamicdrive.com/dynamicindex7/index.html

121

4.6.2: Extract all the Downloaded Files into the Right Directories

After downloading these tools, the researcher extracted the tools into the D directory as shown in Figure 4.2. Here, in order to run Tomcat, the researcher used a tomcat server named etw. MySQL and JDK1.4 are put into the D directory.
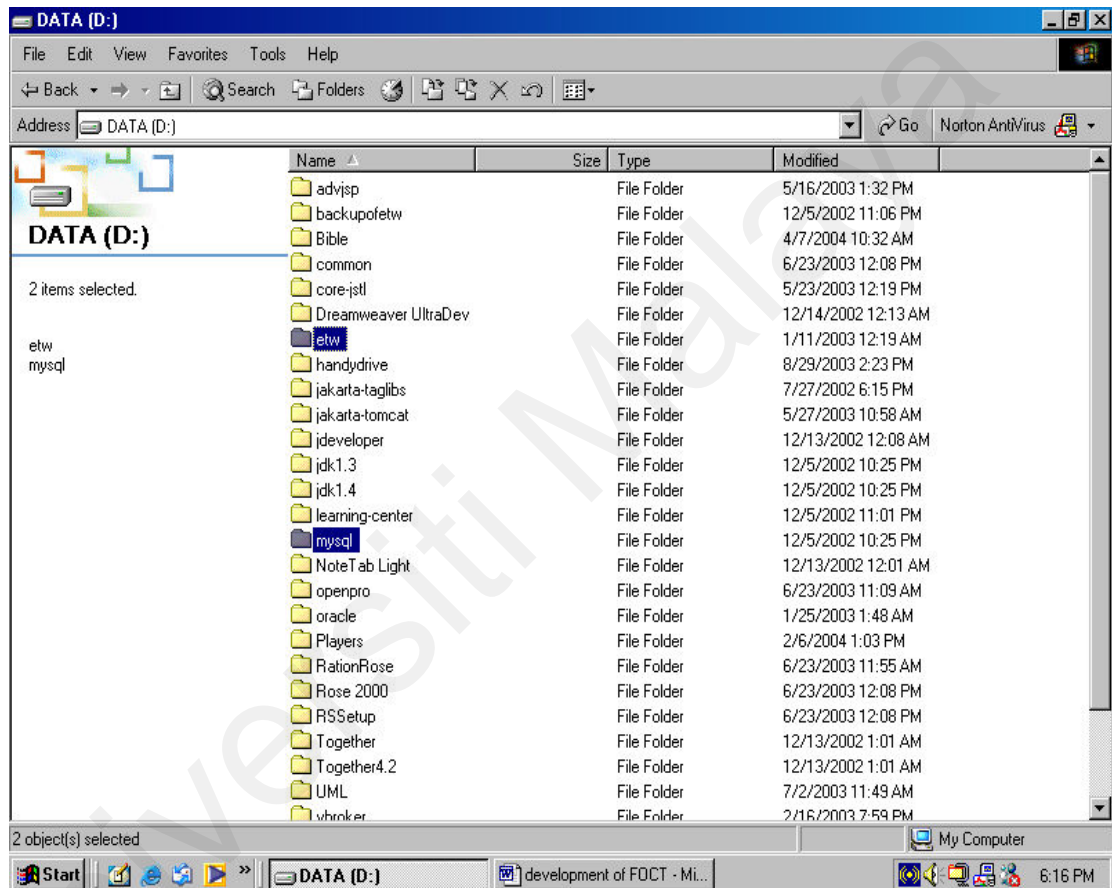

Figure 4.2: Tools in D Directory

4.6.3: Configure the Environmental Setting

    o   Configure JDK environment

To run JDK1.4, users just need to set the java_home in Environment variables as
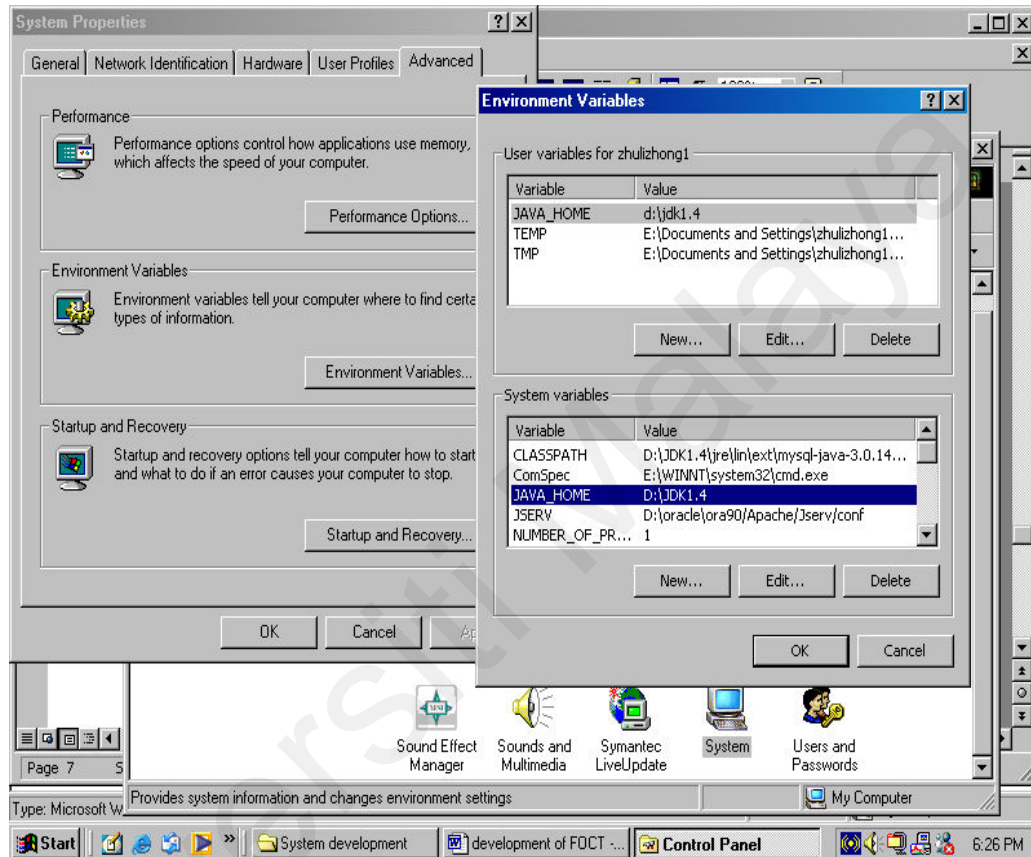
shown in Figure 4.3.

Figure 4.3: Environment Variables Setting for JDK1.4

    o   Configure Tomcat Web server

The researcher needed to set the Tomcat_Home for Tomcat web server. Setting up

Tomcat_Home was very similar to setting up JDK_Home. The only difference is

that the researcher had to add Tomcat_Home to D:\etw-tomcat.

o Configure MySQL

　　　To run MySQL, the researcher just modified the my.ini file as shown in Figure 4.4.
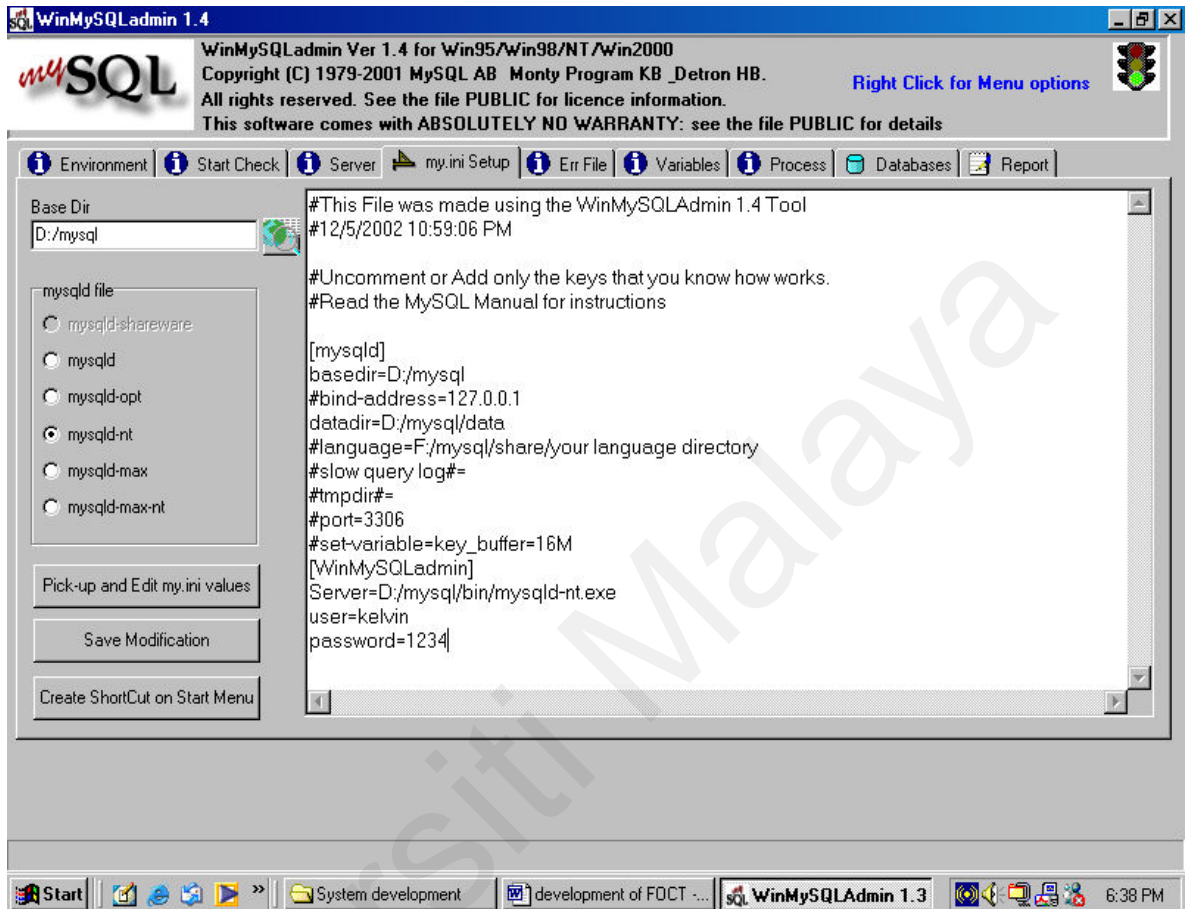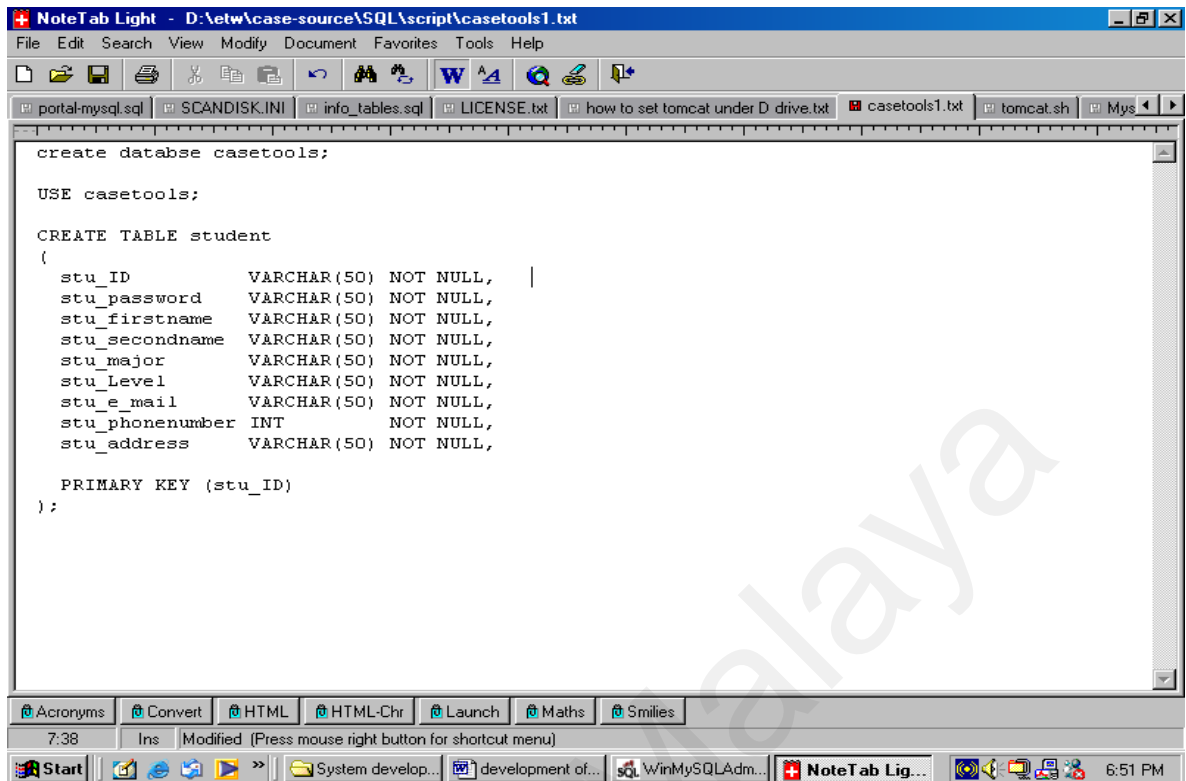

Figure 4.4: MySQL Setting

The researcher moved MySQL into Windows XP and modified the my.ini file, which will

ensure the database can be connected successfully.

4.6.4: Create Database and Tables

After setting all the necessary steps, the researcher accessed the system by using a user

name and password. The researcher created the database and tables for FOCT. The sample

code is shown in Figure 4.5.

Figure 4.5: Commands for Creating Database and Tables

4.6.5: Start MySQL and Tomcat Web Server

To start the MySQL database and Tomcat web server, readers can refer to Appendix C.

4.6.6: Test the Installation

To test whether Tomcat is running successfully, users can open a web browser at the following URL: http://localhost:82/foct/jsp/index.jsp. The users should see a page similar shown in Figure 4.6.
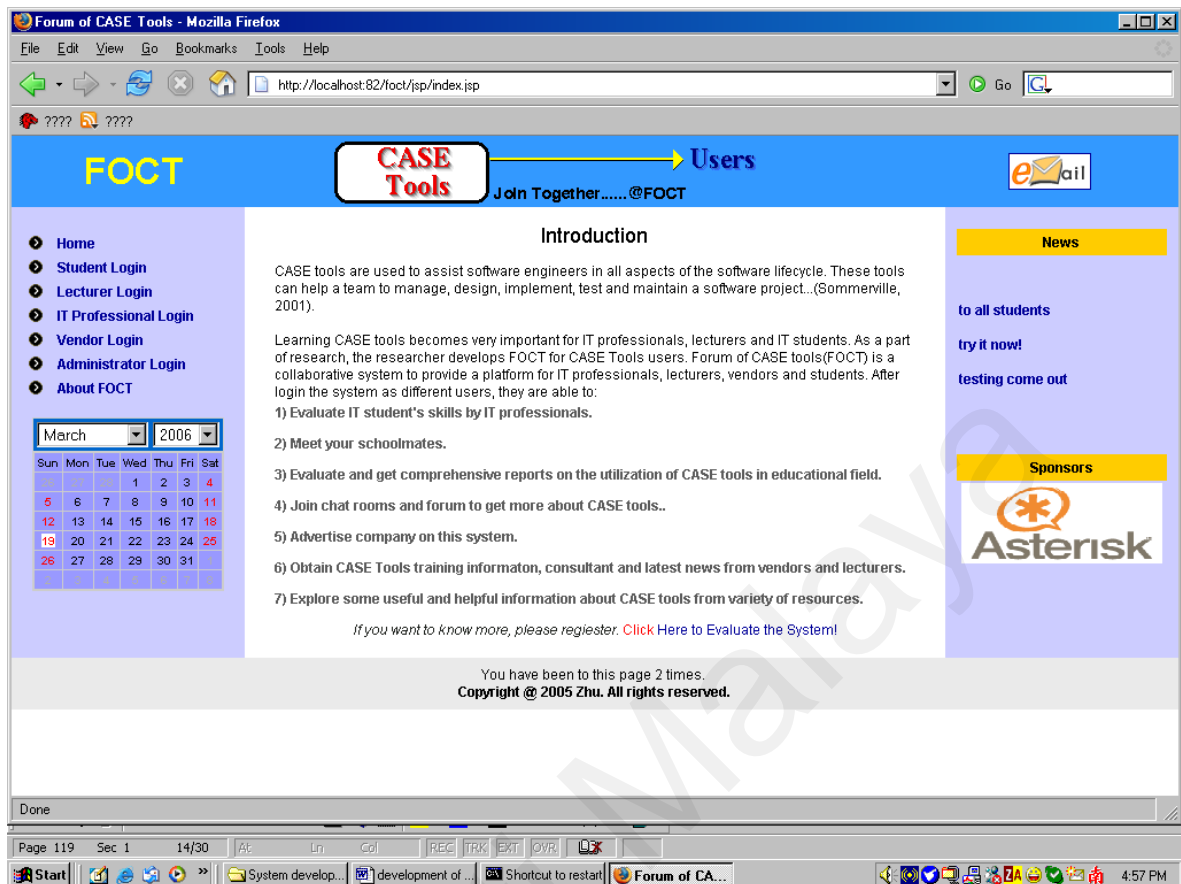
Figure 4.6: Accessing FOCT

4.7 Supporting Language

The researcher used some robust and flexible languages to build the system. A framework

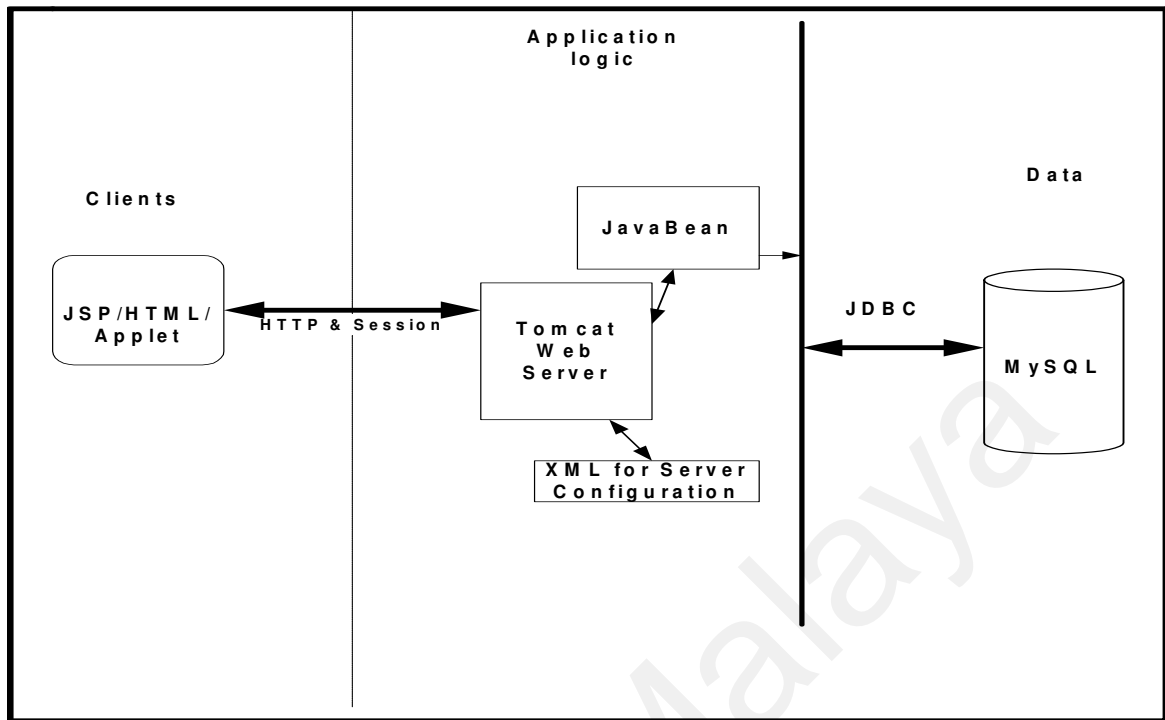that is given by Karl (2000) is shown in Figure 4.7.

Figure 4.7: Supporting Languages Used in System Development

Source (Karl, 2000)

✓ JavaBean

JavaBean is a part of the core Java technology. There are many benefits for using JavaBean such as reusability, and flexibility. The researcher used it in this system for database connection. A connection Bean is developed to connect the MySQL. This component can be reused and can be modified easily.

✓ Java Server Page (JSP)

Damon (2001) described JSP as "JSP is a technology using server-side scripting that is actually translated into Servlets and compiled before they are run". In this system, the researcher used JSP and HTML to present the data received from data source based on the request of user.

127

✓ Applet

Applet is a program that is embedded in a web page. Hence, it can be executed using a web Brower. In FOCT, the researcher used Applets to present the charts of the evaluations.

✓ JavaScript

JavaScript is a scripting language that runs on the client-side. The researcher used JavaScript to open new windows.

✓ XML

Extensible Markup Language (XML) is becoming a spotlight for supporting other languages such as JSP for various purposes. In FOCT, XML is used for configuring the server file of Tomcat.

4.8 System Design

The structure of FOCT is made up of five different groups with different features. To access the system, each group has to login first. The first group is the students group as shown in Figure 4.8. The second group is the IT professionals group as shown in Figure 4.9. The third group is the lecturers group as shown in Figure 4.10. The fourth group is the vendors group as shown in Figure 4.11 and the fifth group is administrator group as shown in Figure 4.12.
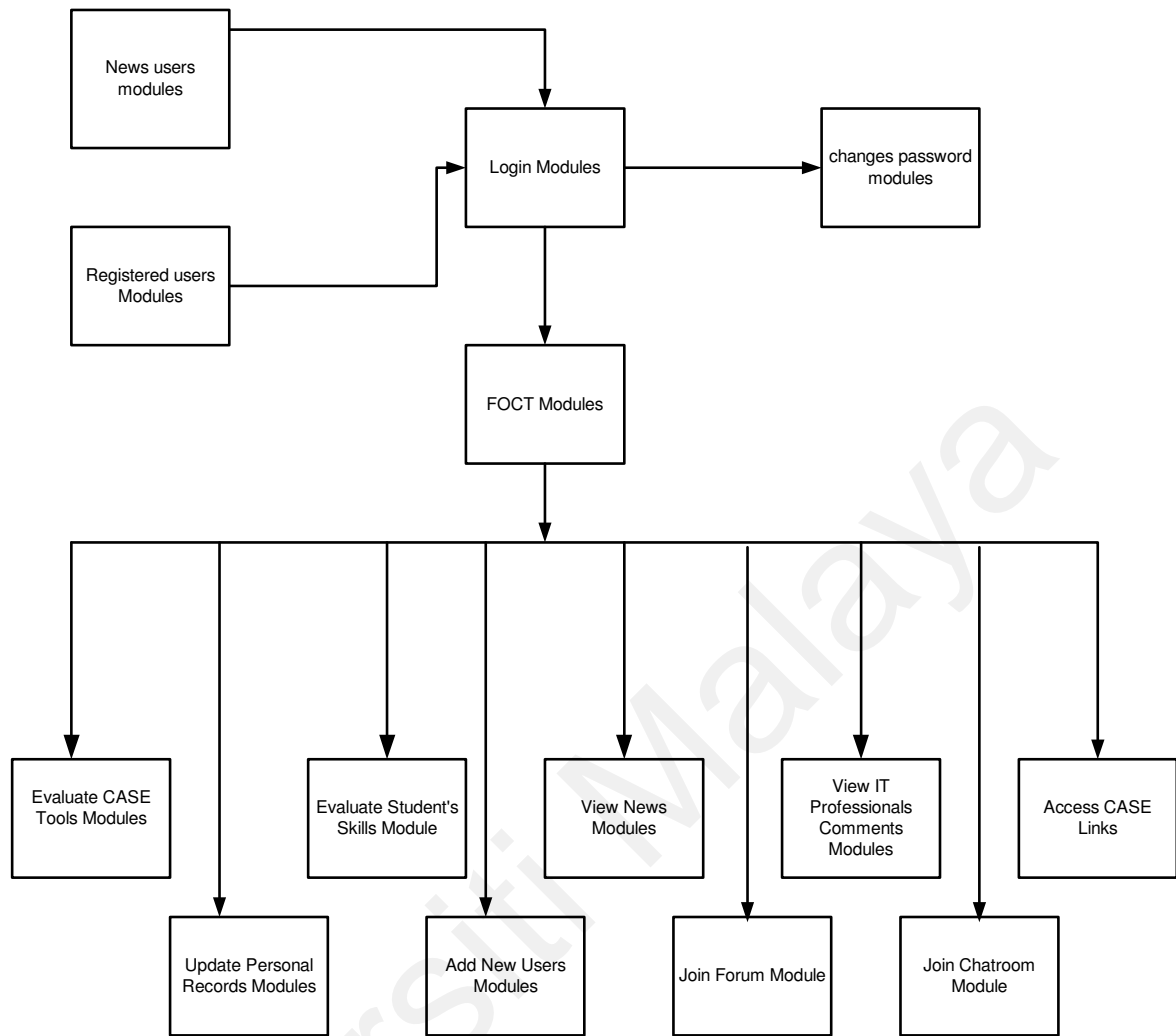
Figure 4.8: System Structure for Students Module

In this module, the students can login to the system using a student ID and password. However, if the student is new, he or she will be asked to register first. After the student has successfully logged into FOCT, there are seven modules available to him or her as shown in Figure 4.8. The CASE links module, View news and View IT professional's Comments are common features for all users. Once, students login, they can evaluate the CASE tools in terms of time, infrastructure, lecturer's support and so on. The students also have authority to view and post their skills. The students are also able to edit their personal records.
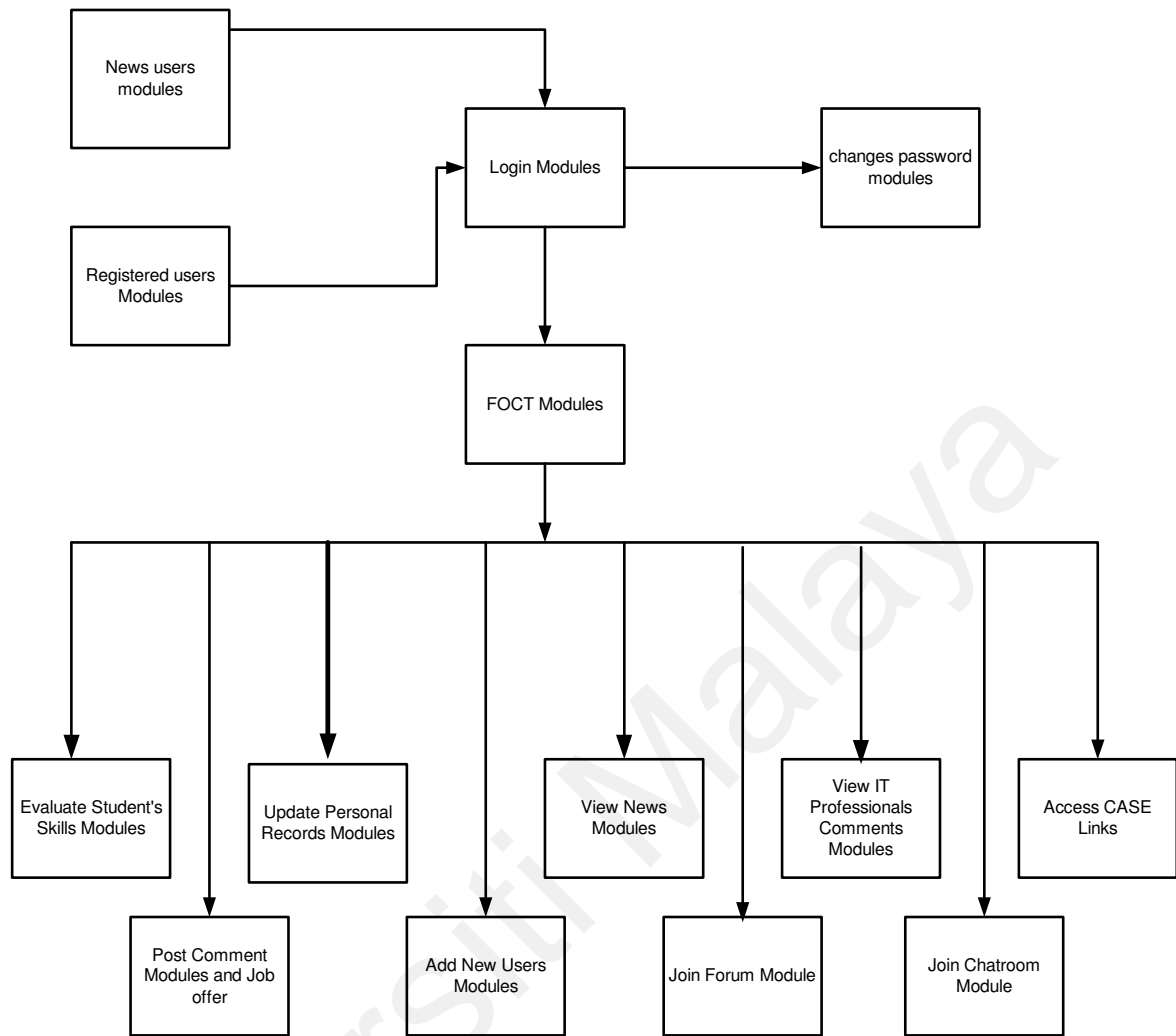
Figure 4.9: System Structure for IT Professionals Module

IT professionals can login using their name and password as shown in Figure 4.9. They also can access three common functions without having to login. These are Access CASE links module, View news and View IT professional's Comments. Once IT professionals login successfully, they can also access Evaluate Student's skills module, Update Modules and Post Comment module. The IT professionals can select a particular student to evaluate the student's knowledge. They can also post their own comments about CASE tools and edit their own records.
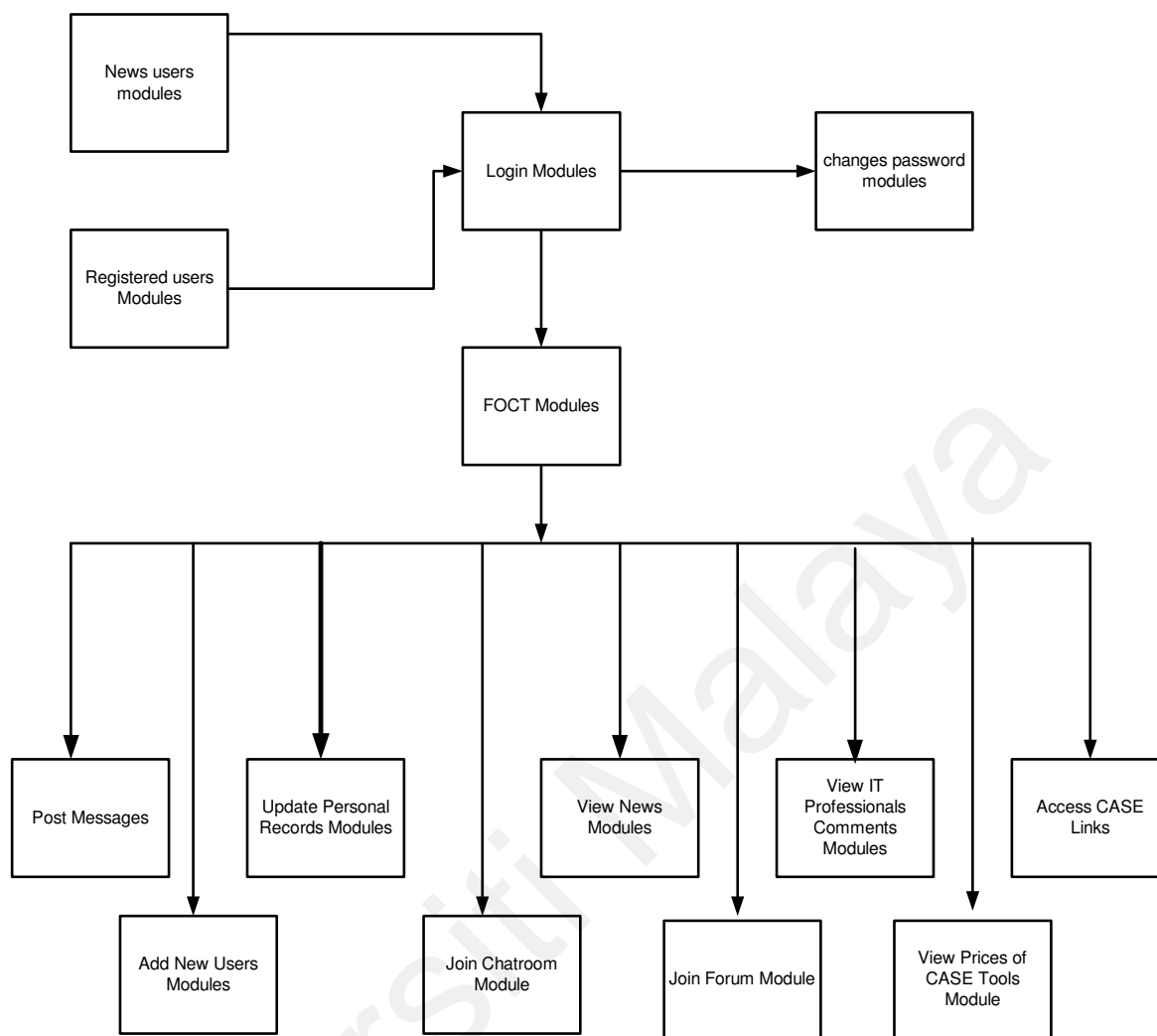
Figure 4.10: System Structure for Lecturers Module

As shown in Figure 4.10, lecturers can access the three common features as described previously without login. If the lecturer is new to the system, he or she needs to register first. Once the lecturers login using their university's names, lecturer ID and password, they will be able to access some modules such as Update Personal Record module and Post Message module. The Post Message module gives lecturers privileges to post relevant messages to the users.

Figure 4.11: System Structure for Vendors Module

Vendors have similar features like lecturers. In Figure 4.11, the vendors have rights to post news and messages. This will expose students and lecturers to new information about CASE tools.
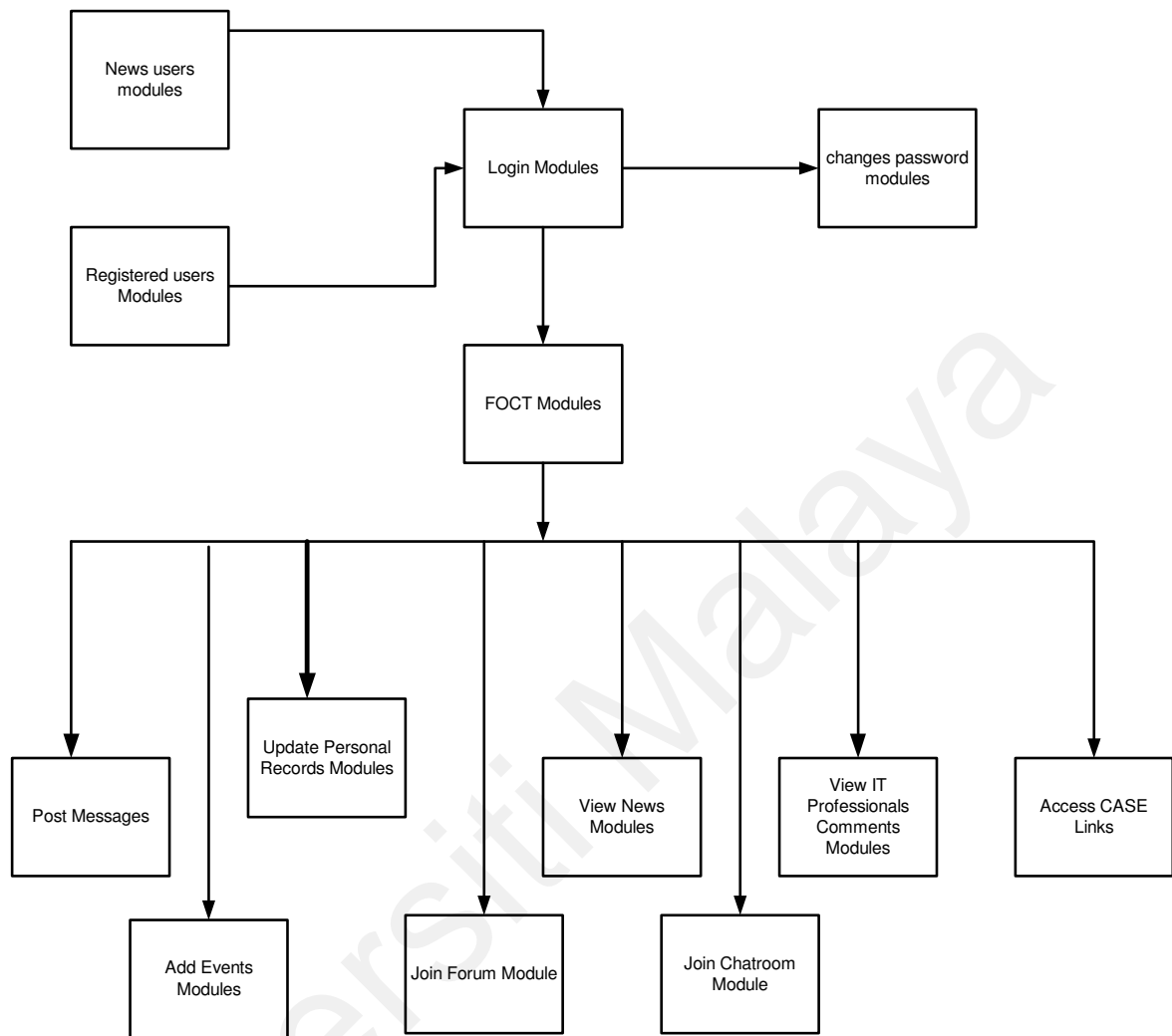
```
┌──────────────┐          ┌──────────────┐          ┌──────────────┐
│ Registered   │─────────▶│ Login        │─────────▶│ changes      │
│ users Modules│          │ Modules      │          │ password     │
│              │          │              │          │ modules      │
└──────────────┘          └──────┬───────┘          └──────────────┘
                                 │
                                 ▼
                          ┌──────────────┐
                          │ FOCT Modules │
                          └──────┬───────┘
                                 │
                                 ▼
```
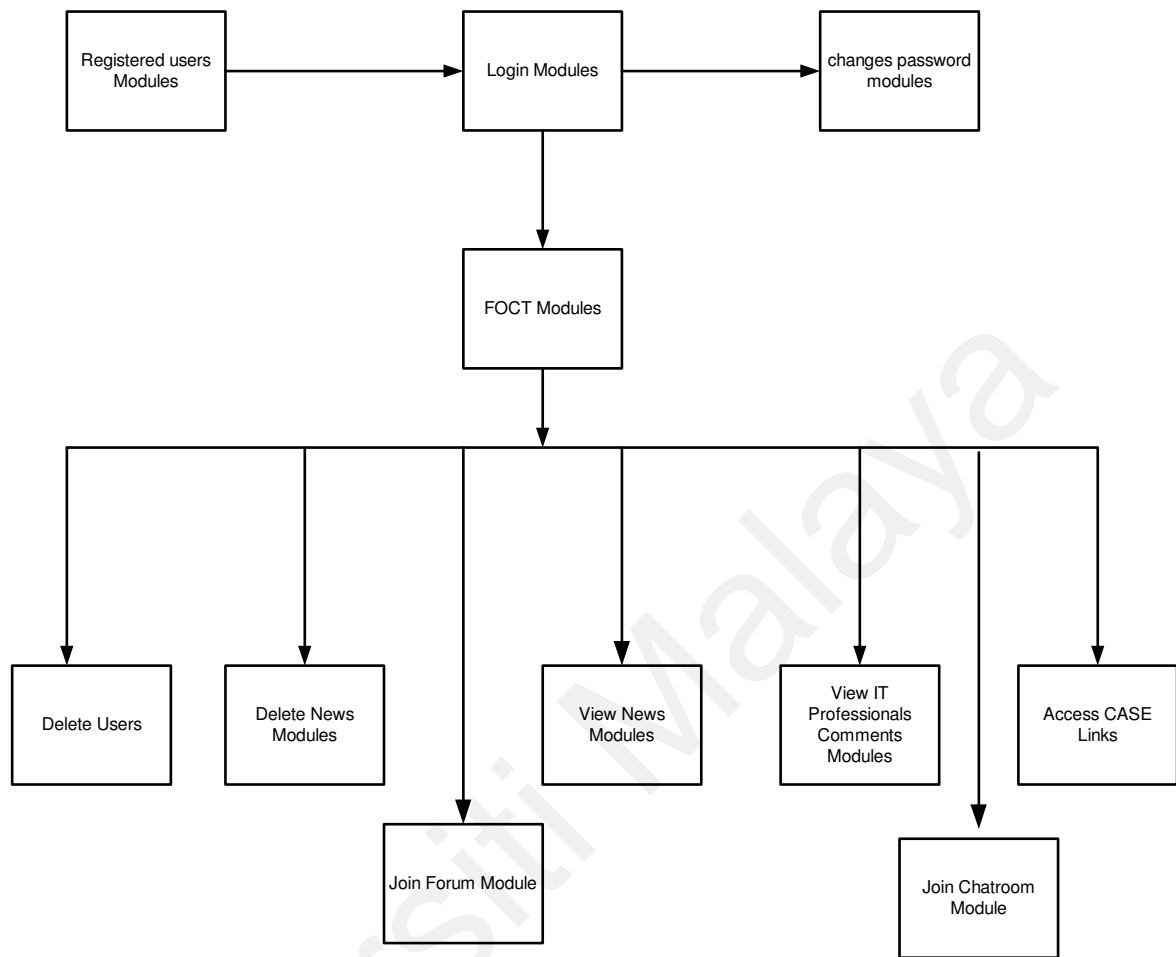
Figure 4.12: System Structure for Administrator Module

Besides having access to the three common functions, administrators are also able to allow or delete the messages posted by either lecturers or vendors as shown in Figure 4.12.

4.9 How the System Works

FOCT has many features like user profile update to help the users more effectively. The entire system is described using a Data Flow Diagram (DFD). The DFD of FOCT is divided into three levels ranging from level 0 to level 2 as shown in Figure 4.13 and Figure 4.14.
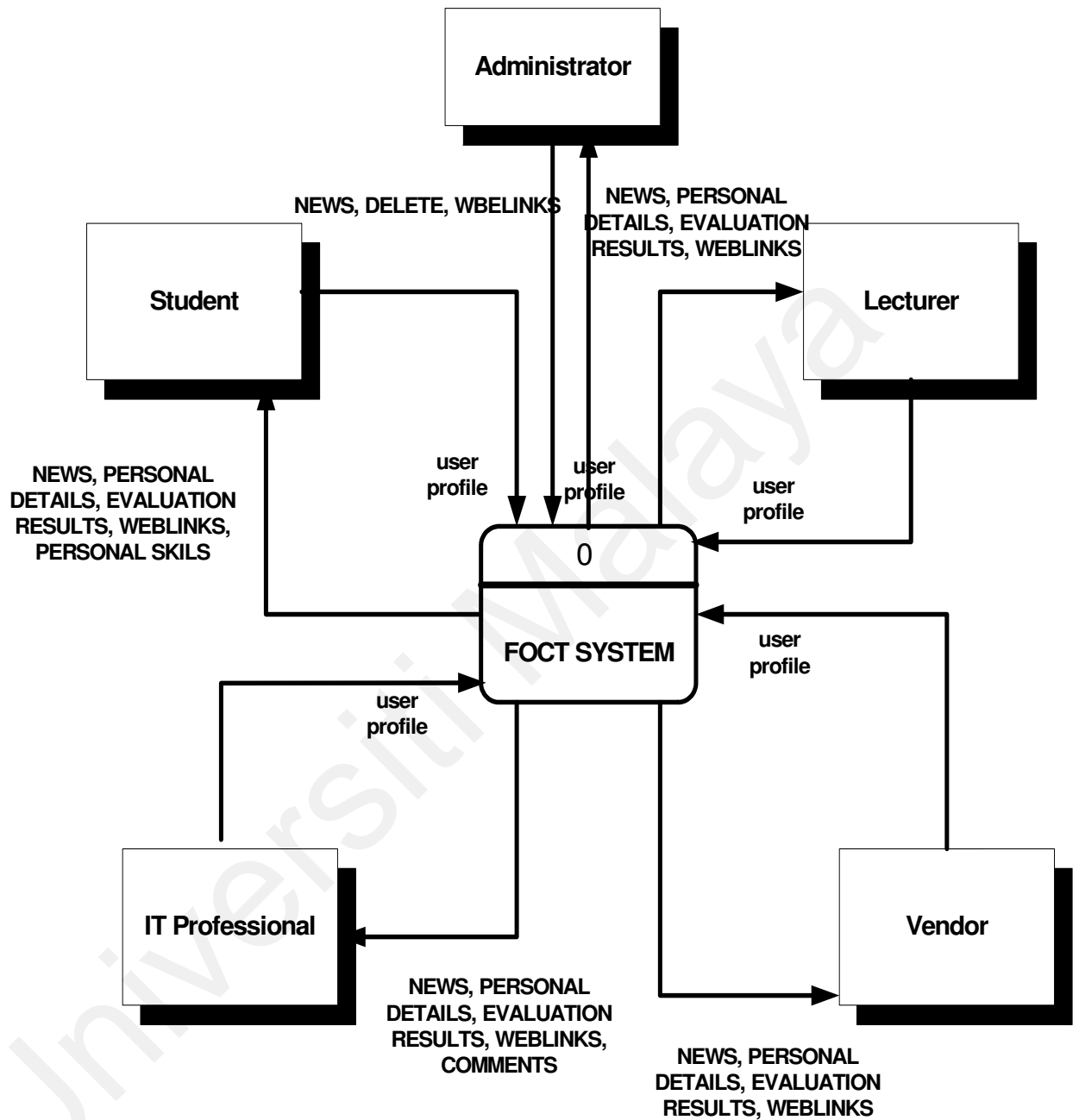
133

Figure 4.13: Context Diagram and Diagram 0 for the FOCT System
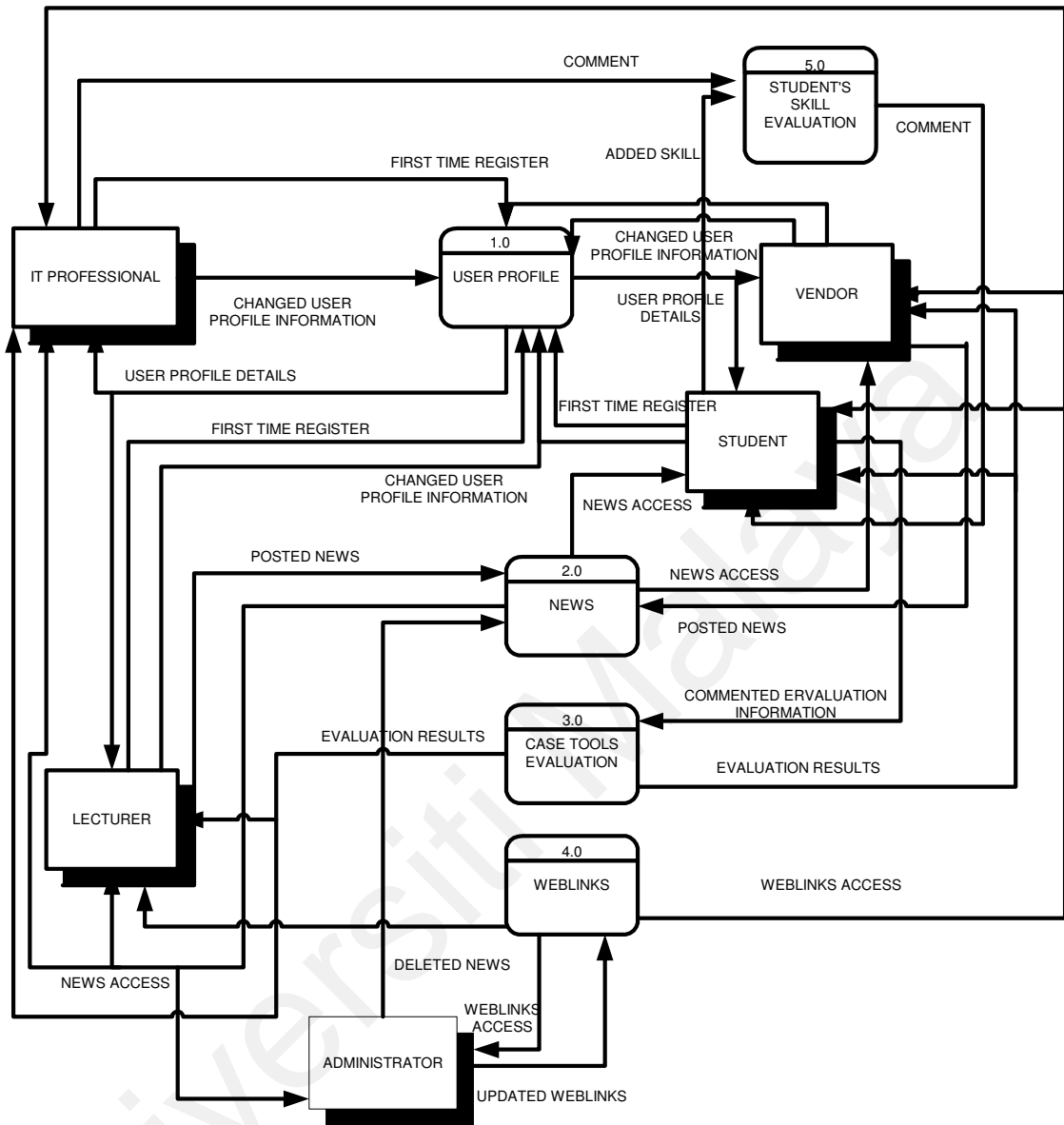
Figure 4.14: Level 1 and DFD for FOCT System

1. User Profile

   Under the user profile in Figure 4.15, users can add new users. Validated users can update their personal records or information.
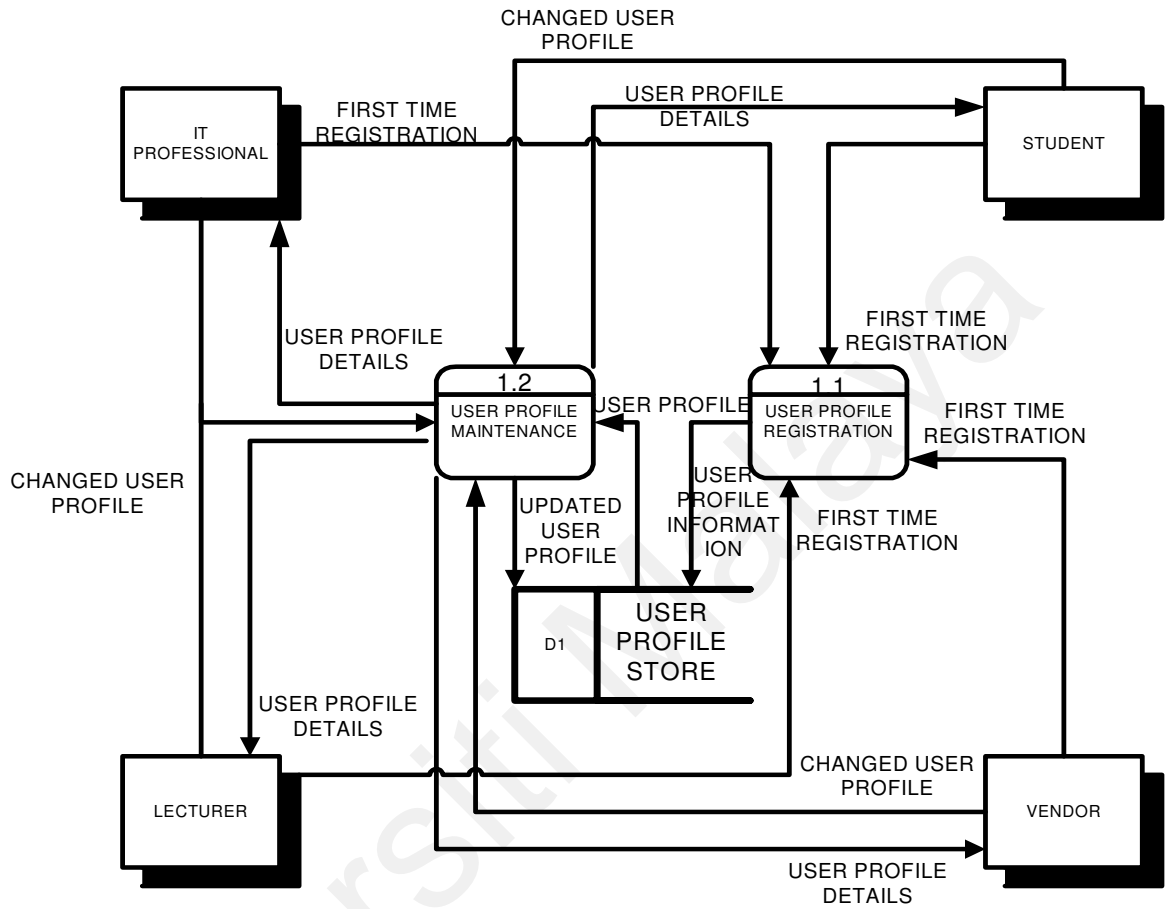
Figure 4.15: DFD for User Profile in FOCT System

2. Skill Evaluation

Using the skill evaluation feature, a student can fill in a CASE tool evaluation form to evaluate his or her skill and IT professionals can give comments on what skills the student has. This is shown in Figure 4.16.
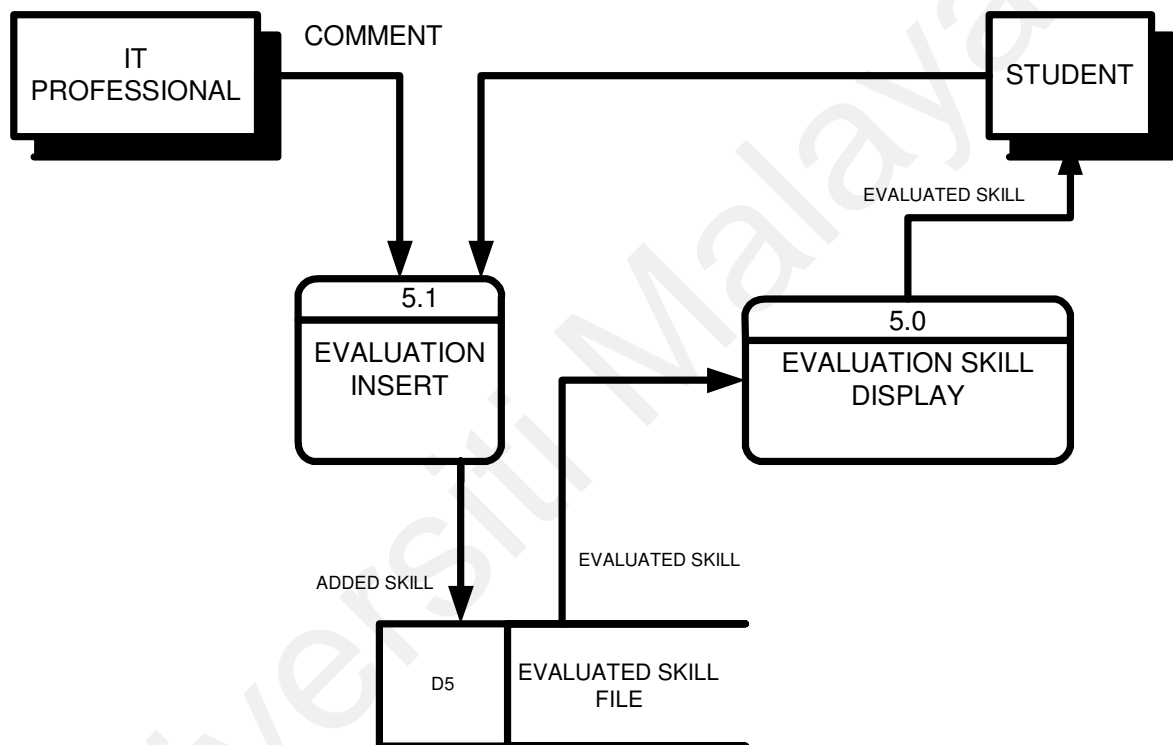
Figure 4.16: DFD for Skill Evaluation in FOCT

3. CASE Tools Evaluation Result

    Using the evaluation result feature, all users can view the results of the CASE
    tools evaluation. Only students are able to add comments to CASE tools
    evaluation. This is shown in Figure 4.17
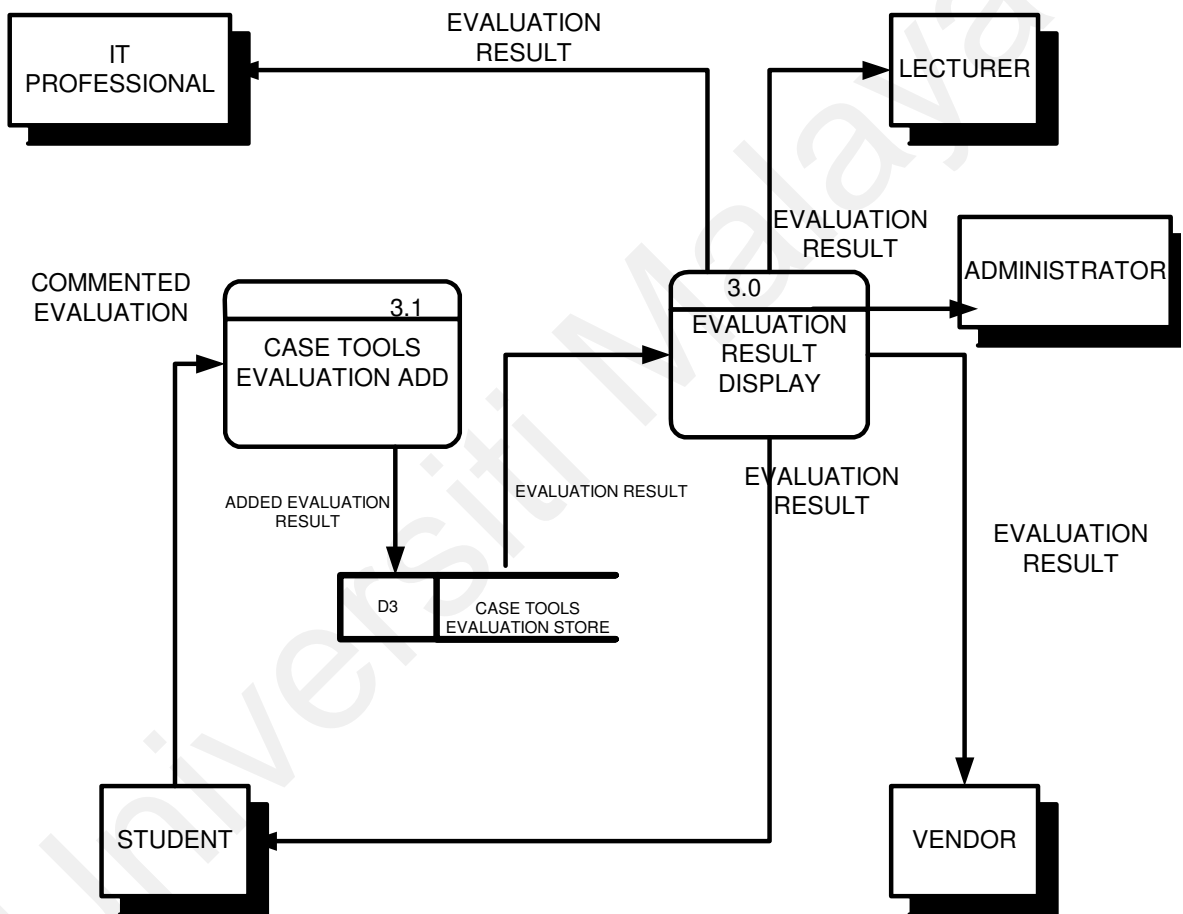
Figure 4.17: DFD for CASE Tools Evaluation in FOCT

4. News Access

   Using this feature, all users can read the news. Only validated vendors and lecturers can post news to FOCT. This is shown in Figure 4.18.
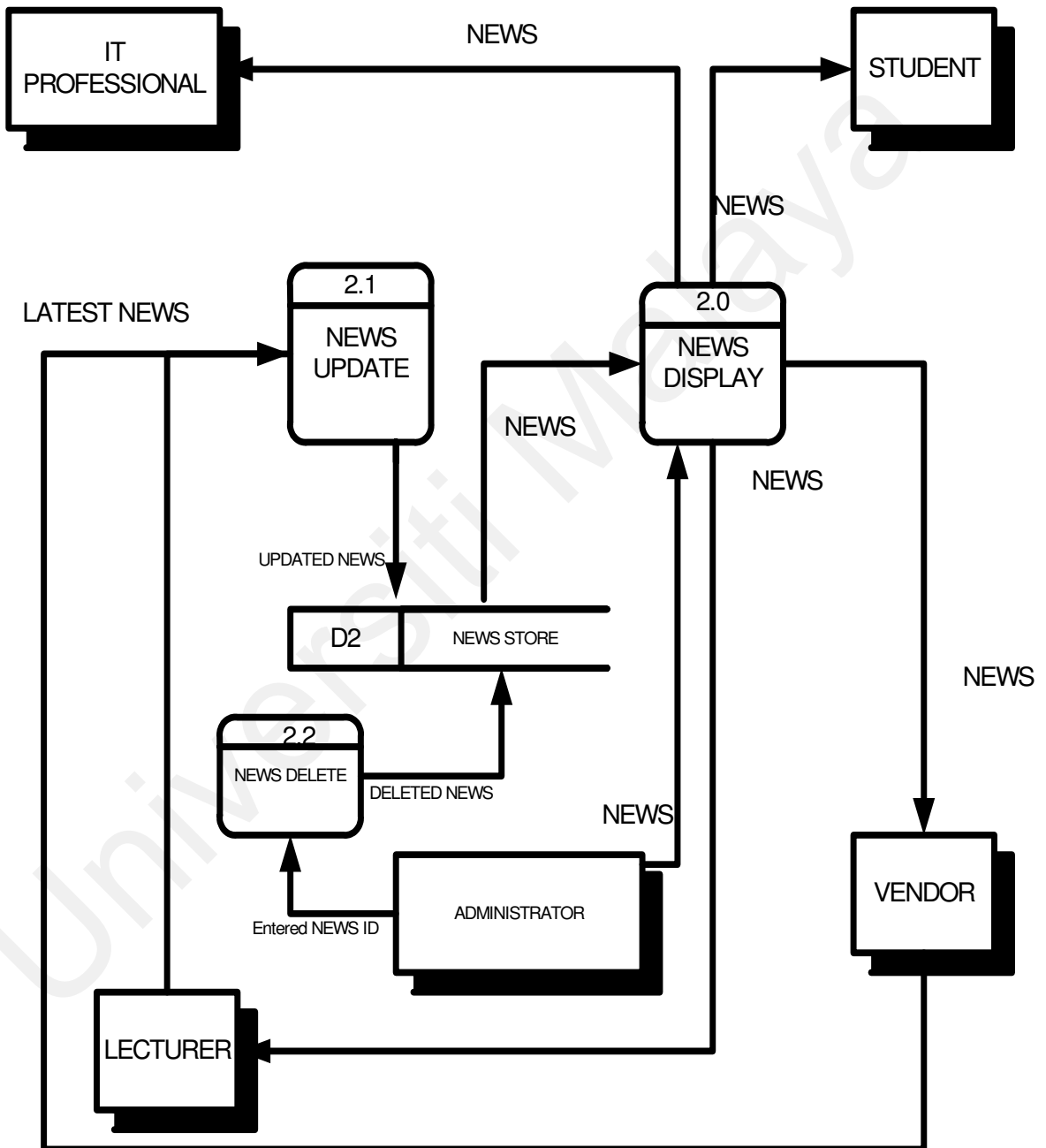
Figure 4.18: DFD for NEWS in FOCT System

5. Web Links

Using the weblinks, all users can click some useful links to direct to other useful

websites and resources of CASE tools research. This is shown in Figure 4.19

Figure 4.19: DFD for Weblinks in FOCT System

4.10 Modules and Functions

The FOCT consists of several modules for five user groups. These modules perform different functions.

1) Students

Students are the main users for this system. There are many modules for students as shown in Table 4.4.

Table 4.4: Students Module and Functions

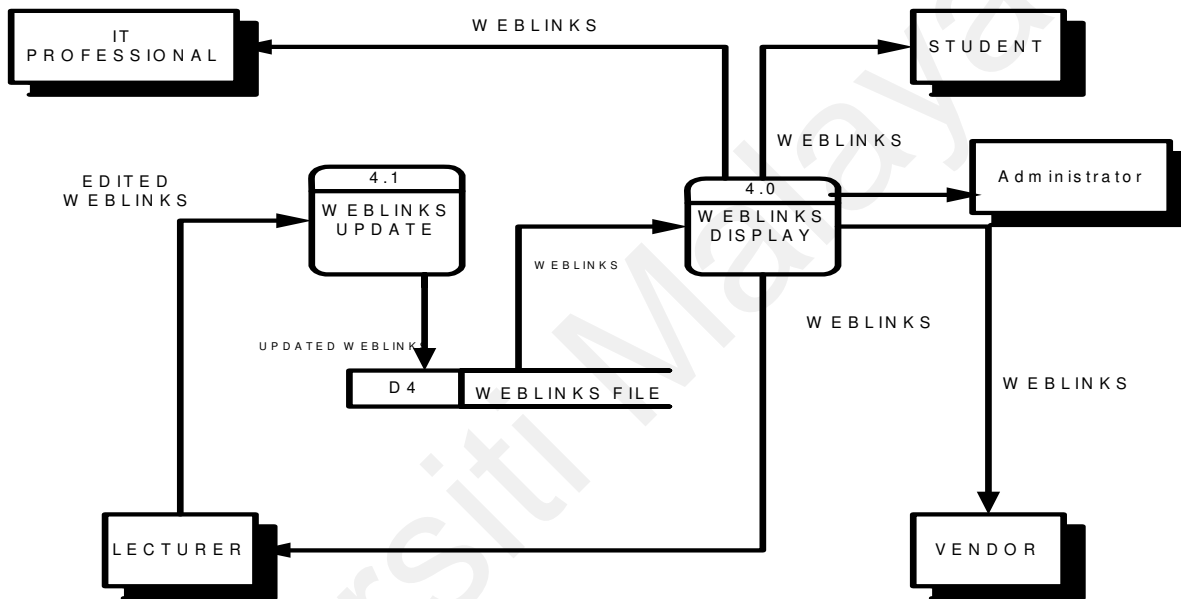| Module	Functions
Login	Let students access the system.
Evaluate the learning of CASE tools	Enable students to evaluate the usage of CASE tools.
Update records	Allow students to update personal records after edited the records.
Register new students	Add new student's records into system.
Post skills	Add student's skill into the system.
View the result of evaluation	Enable students to view the result of evaluation.
View the comment on student's skills from IT professionals	Allow students to see what IT professional commented on their skills using CASE tools.
CASE tool links	Enable students explore more sources for CASE tools.
Evaluate skills	Allow students to evaluate their skills.
Access the news	Enable students read the news.
Join forum	Allow students to post questions.
Join chat room	The students can chat with other users through chat room.

2) IT Professionals

Table 4.5: IT Professionals Module and Functions

Module	Functions
Login	Let IT professionals access the system.
Update record	Allow IT professionals to update personal records after edited the records.
Register new users	Add new IT professional's records into system.
Post comment and job positions	Add IT professional's comment and job offers into the system.
View the result of evaluation	Enable IT professionals to view the result of evaluation.
CASE tool links	Enable IT professionals explore more sources for CASE tools.
Evaluate students	Allow IT professionals to evaluate a particular student's skills.
Access the news	Enable IT professionals read the news.
Join forum	Allow IT people to post questions.
Join chat room	The IT professionals can chat with other users through chat room.

3) Lecturers

Table 4.6: Lecturers Module and Functions

Modules	Functions
Login	Let lecturers access the system.
Update records	Allow lecturers to update personal records after editing the records.
Register new users	Add a new lecturer's records into system.
Post messages	Add lecturer's comment into the system.
View the result of evaluation	Enable lecturers to view the result of evaluation.
CASE tool links	Enable lecturers explore more sources for CASE tools.
Access the news	Enable lecturers to read the news.
Join forum	Allow lecturers to post questions to the forum.
Create and join chat room	The lecturers can chat with other users through chat room.
View the prices of CASE tools	Lecturers view all the prices of CASE tools posted by vendors.

4) Administrator

Table 4.7: Administrator Module and Functions

Modules	Functions
Login	Let Administrator access the system.
View the result of evaluation	Enable Administrator to view the result of evaluation.
CASE tool links	Enable Administrator to explore more sources for CASE tools.
Access the news	Enable Administrator to read the news.
Delete the News	Allow Administrator to delete the News.
Join forum room	The administrator can chat with other users through chat room.
Join forum	The administrator is able to join the forum.

5) Vendors

Table 4.8: Vendors Module and Functions

Modules	Functions
Login	Let vendors access the system.
Update	Allow vendors to update personal records after editing the records.
Register	Add a new vendor's records into system.
Post messages	Add vendor's news into the system.
View the result of evaluation	Enable vendors to view the result of evaluation.
CASE tool links	Enable vendors to explore more sources for CASE tools.
Access the news	Enable vendors to read the news.
Post events and consultant service.	Allow vendors to publish their events and consultant service in this system.
Join forum	Vendors post the CASE tools information.
Join chat room	The vendor can chat with other users through chat room.

4.11 Database Design

casetools Database consists of seventeen tables in the FOCT system. The seventeen tables

are designed to handle different information as shown in the tables below.

Table 4.9: Database and Tables in FOCT System

Database: casetools	
Table Name	Functions
bbs	Contains news data for reading news.
evaluation	Contains CASE tool evaluation information
evastu	Contains result of evaluating student's skill.
itcomment	Contains comment from IT professionals.
itprofessional	Contains IT professional's profiles.
lecturer	Contains lecturer's records
student	Contains student's records
vendor	Contains vendor's records
vencom	Contains comment posted by vendors
admin	Stores admin records
forum	Contains the records for forum
job	Stores the job information posted by IT professionals
seminar	Stores the seminar information posted by vendors
trainingCourse	Stores the training course records posted by vendors.
consultant	Contains the consultant service information posted by vendors
tools	Stores the CASE tools software information
alumni	Stores the alumni information

Table 4.10: bbs Table and Fields

bbs table		
Field Name	Data type	Descriptions
bbs_id	Integer	Describes id as primary key.
bbs_sendername	String	Describes sender name.
bbs_senderid	Integer	Defines sender id.
bbs_senderemail	String	Defines sender E-mail.
bbs_senddate	Date	Defines the date of sending bbs.
bbs_sendtitle	String	Defines the title of content.
bbs_sendcontent	Text	Defines the content.

Table 4.11: evaluation Table and Fields

evaluation table		
Field Name	Data type	Descriptions
eva_id	Integer	Describes id as primary key.
eva_posterid	Integer	Describes poster id.
eva_postdatetime	Date	Defines the date of posting evaluation form.
eva_que1	String	Defines question 1 in evaluation form
eva_que2	String	Describes question 2 in evaluation form
eva_que3Rat	String	Describes question 3 about RationalRose in evaluation form
eva_que3Sys	String	Defines question 3 about SystemArchitect in evaluation form
eva_que3Tog	String	Describes question 3 about Together in evaluation form
eva_que3Pro	String	Defines question 3 about Microsoft Project in evaluation form
eva_que3Vis	String	Describes question 3 about Visio Professional in evaluation form
eva_que4Hour	String	Describes question 4 about hours in evaluation form
eva_que5Dra	String	Describes question 5 about using purposes in evaluation form
eva_que5Cod	String	Describes question about using purposes in evaluation form
eva_que5Dou	String	Describes question about using purposes in evaluation form
eva_que5Use	String	Describes question about using purposes in evaluation form
eva_que5Oth	String	Defines question about using purposes in evaluation form
eva_que6Dif	String	Defines question 6 in evaluation form
eva_que7Fac	String	Describes question 7 about facilities in evaluation form
eva_que8Gui	String	Describes question 8 about guidance in evaluation form
eva_que8Pc	String	Describes question 8 about factors of using PCs in evaluation form
eva_que8Exp	String	Describes question 8 about factors of price in evaluation form
eva_que8Com	String	Describes question 8 about complication in evaluation form
eva_que8Sec	String	Describes question 8 about security in evaluation form
eva_que8Lear	String	Describes question 8 about learning approach in evaluation form
eva_que9Com	String	Describes question 9 in evaluation form

Table 4.12: evastu Table and Fields

evastu table		
Field Name	Data type	Descriptions
eva_id	Integer	Describes id as primary key.
eva_posterid	Integer	Describes sender id.
eva_learnedtools	String	Defines the tools the student learned.
eva_tool	String	Defines the name of the tool.
eva_suboot	String	Defines the subject OOT.
eva_subsad	String	Defines the subject SAD.
eva_subse	String	Defines the subject Software engineering.
eva_subuser	String	Describes the subject user requirement engineering.
eva_subpro	String	Describes the subject Project Management.
eva_data	String	Describes the subject Database
eva_projava	String	Describes the programming the student learned.
eva_procplus	String	Describes the programming the student learned.
eva_procsharp	String	Describes the programming the student learned.
eva_proc	String	Describes the programming the student learned.
eva_time	String	Describes the time the student spent.
eva_purdraw	String	Describes purposes of using CASE tools.
eva_purcode	String	Describes purposes of using CASE tools.
eva_puruser	String	Describes purposes of using CASE tools.
eva_puroth	String	Describes purposes of using CASE tools.
eva_des	String	Describes suggestions of using CASE tools.

Table 4.13: itcomment Table and Fields

itcomment table		
Field Name	Data type	Descriptions
com_id	Integer	Describes id as primary key.
com_toolsname	String	Describes the name of tools used.
com_year	Integer	Defines how many years they used the tools.
com_purpose	String	Defines purposes.
com_comment	String	Defines comment about casetools.
com_suggestion	String	Defines the suggestion about using case tools.
com_proid	String	Defines IT professional ID.
com_date	String	Defines the date of sending the comment.

Table 4.14: itprofessional Table and Fields

itprofessional table		
Field Name	Data type	Descriptions
pro_id	Integer	Describes id as primary key.
pro_name	String	Describes professional's name.
pro_position	String	Defines sender id.
pro_email	String	Defines sender E-mail.
pro_companyaddress	String	Defines the date of sending bbs.
pro_companyname	String	Defines the title of content.
pro_contactnumber	String	Defines the content.
pro_website	String	Describes the company website.
pro_password	String	Describes the password used to login.

Table 4.15: lecturer Table and Fields

lecturer table		
Field Name	Data type	Descriptions
lec_id	Integer	Describes id as primary key.
lec_first name	String	Describes lecturer's first name.
lec_email	String	Defines lecturer's email.
lec_phonenumber	String	Defines lecturer's contact number.
lec_secondname	String	Defines lecturer's second name.
lec_address	String	Defines lecturer's address.
lec_password	String	Defines lecturer's password.

Table 4.16: student Table and Fields

student table		
Field Name	Data type	Descriptions
stu_id	Integer	Describes id as primary key.
stu_firstname	String	Describes student's first name.
stu_secondname	String	Defines student's second name.
stu_major	String	Defines the major.
stu_level	String	Defines the level of education.
stu_phonenumber	String	Defines the phone number.
stu_email	String	Defines the email.
stu_password	String	Defines the password used to login
stu_address	String	Defines the student's address.

Table 4.17: vencom Table and Fields

vencom table		
Field Name	Data type	Descriptions
ven_id	Integer	Describes id as primary key.
ven_posterid	String	Describes vendor id.
ven_content	Text	Defines content.
ven_toolname	String	Defines the name of tool.
ven_senddate	Date	Defines the date of sending comment.

Table 4.18: vendor Table and Fields

vendor table		
Field Name	Data type	Descriptions
ven_id	Integer	Describes id as primary key.
ven_name	String	Describes vendor name.
ven_password	String	Defines vendor password.
ven_senderemail	String	Defines vendor's E-mail.
ven_productname	String	Defines the product name.
ven_companyaddress	String	Defines the address of company.
ven_companyname	String	Defines the name of company.
ven_website	String	Defines the website the company has.
ven_contact number	String	Defines the contact number.
ven_email	String	Defines the vendor's e-mail.

Table 4.19: forum Table and Fields

forum table		
Field Name	Data type	Descriptions
MessageID	Integer	Describes id as primary key.
TopicID	String	Describes topic id.
PremessageID	String	Defines previous message id.
FolmessageID	String	Defines following message id.
PosterID	String	Defines the poster id.
Post_date	Date	Defines the date when poster posted.
Message	Text	Defines message.
title	String	Defines the title of message.

Table 4.20: job Table and Fields

job table		
Field Name	Data type	Descriptions
Job_id	Integer	Describes id as primary key.
Job_position	String	Describes the name of position.
Job_responsibility	String	Defines the responsibility of job.
Job_requirements	String	Defines job requirements.
Job_sender	String	Defines the sender id.
Job_closingdate	Date	Defines the closing date.

Table 4.21: seminar Table and Fields

seminar table		
Field Name	Data type	Descriptions
Sem_id	Integer	Describes id as primary key.
Sem_desc	String	Describes the seminar.
Sem_email	String	Defines the email for seminar.
Sem_contactnumber	String	Defines contact number.
Sem_speaker	String	Defines speaker name.
Sem_venue	String	Defines the venue.
Sem_fees	Double	Defines the fees of seminar.
Sem_conductdate	Data	Defines the date.
Sem_time	Date	Defines the time.
Sem_vendorid	Integer	Defines the vendor id.
Sem_topic	String	Defines the name of topic.

Table 4.22: tools Table and Fields

tools table		
Field Name	Data type	Descriptions
Tool_id	Integer	Describes id as primary key.
Tool_name	String	Describes the name of tool.
Tool_price	Double	Defines the tool price.
Tool_license	Integer	Defines license.
Tool_version	String	Defines the version of tool.
Tool_features	String	Describes the features the tool has.
Tool_description	String	Describes the overall information.
Tool_contact	String	Defines the vendor contact.
Tool_vendorid	Integer	Defines the vendor id

Table 4.23: consultant Table and Fields

consultant table		
Field Name	Data type	Descriptions
Consultant _id	Integer	Describes id as primary key.
Consultant _companyname	String	Describes the company name.
Consultant _solution	String	Describes the solution.
Consultant_desc	String	Defines general information about consultant.
Consultant_email	String	Defines email.
Consultant_companyurl	String	Describes the company URL.
Consultant_contactnumber	String	Describes contact number.
Consultant_vendorid	Integer	Defines the vendor id.

Table 4.24: alumni Table and Fields

alumni table		
Field Name	Data type	Descriptions
Alumni_id	Integer	Describes id as primary key.
alumni_universityname	String	Describes the name of university.
Alumni_studentid	String	Defines the student ID.
Alumni_email	String	Defines email.
A;umni_name	String	Defines the name of student.

Table 4.25: trainingcourse Table and Fields

Trainingcourse table		
Field Name	Data type	Descriptions
Trainingcour_id	Integer	Describes id as primary key.
Trainingcour_name	String	Describes the name of university.
Trainingcour_fees	Double	Defines the student ID.
Trainingcour_instructor	String	Defines email.
Trainingcour_conductdate	Date	Defines the name of student.
Trainingcour_desc	String	Defines general information.
Trainingcour_email	String	Define the email.
Trainingcour_contactnumber	String	Define the contact number.
Trainingcour_vendorid	Integer	Defines the vendor id.
Trainingcour_venue	String	Defines the venue.

Table 4.26: admin Table and Fields

admin table		
Field Name	Data type	Descriptions
Admin_id	Integer	Describes id as primary key.
AdminTpassword	String	Describes the password of administrator.
Admin_name	String	Defines the name of administrator
Admin_email	String	Defines the email of administrator.

4.12 Interface Design
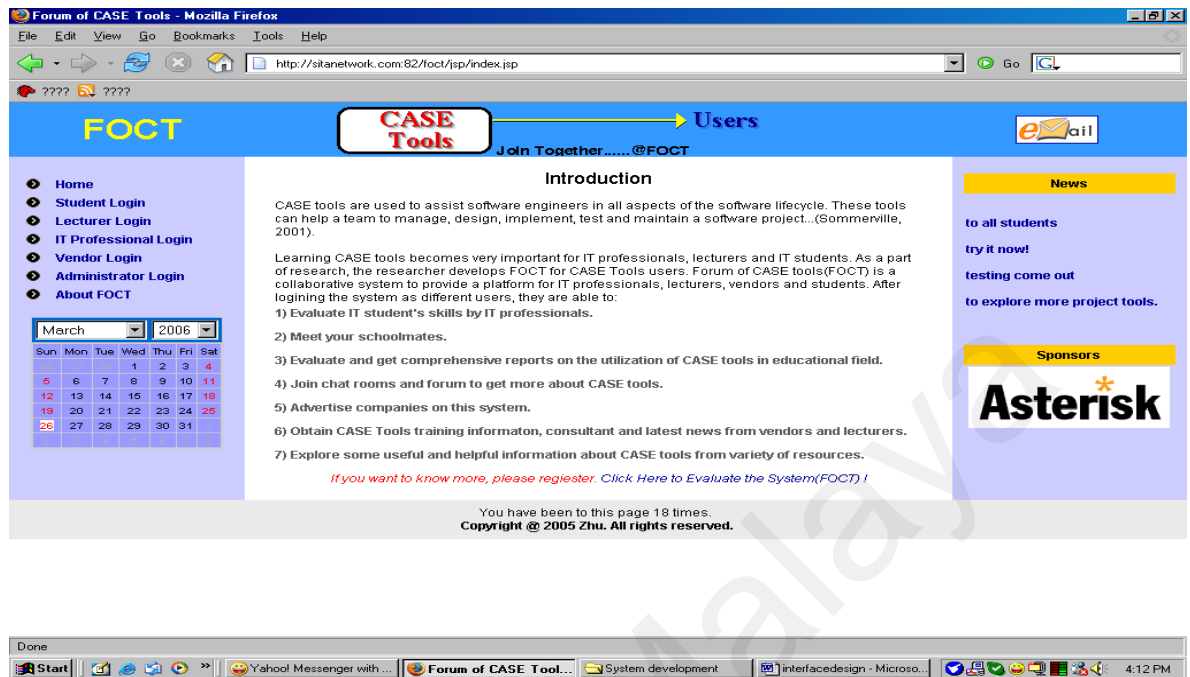
Figure 4.20: Homepage of FOCT
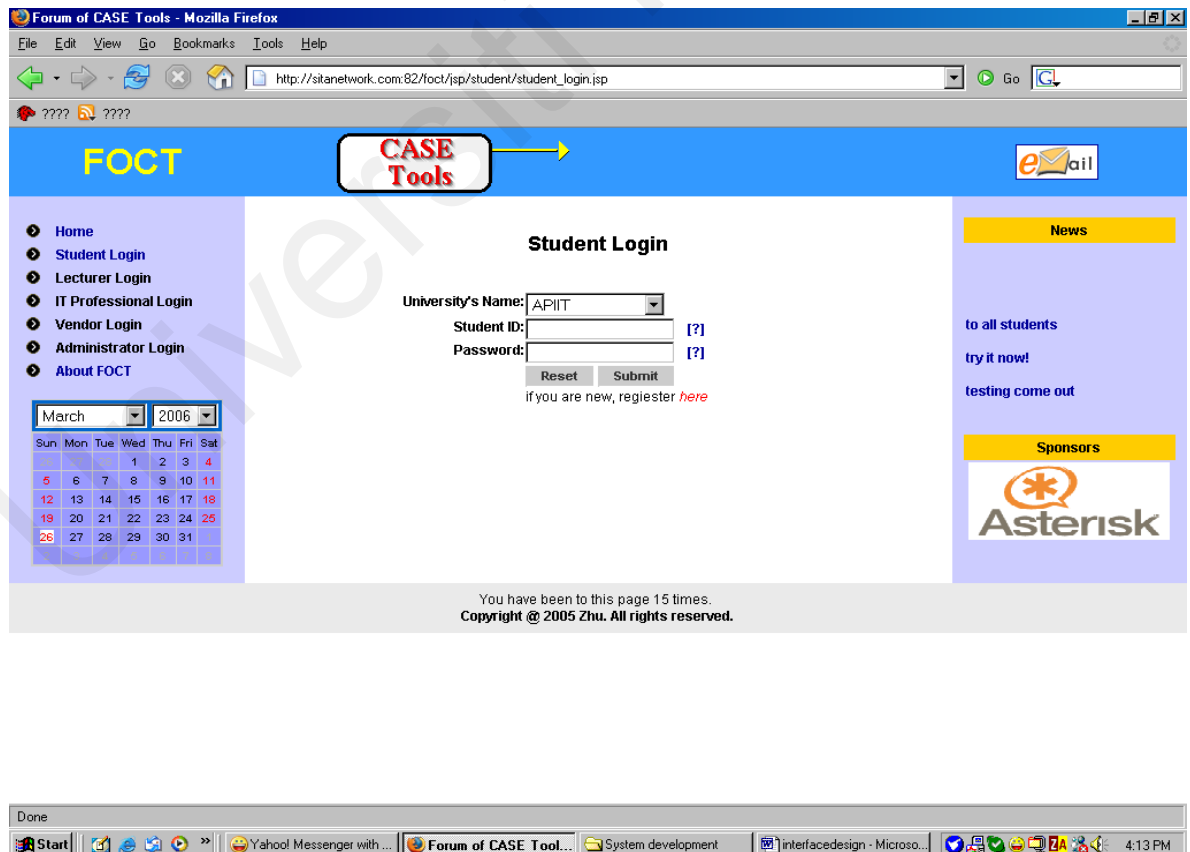
Figure 4.21: Student Login

149
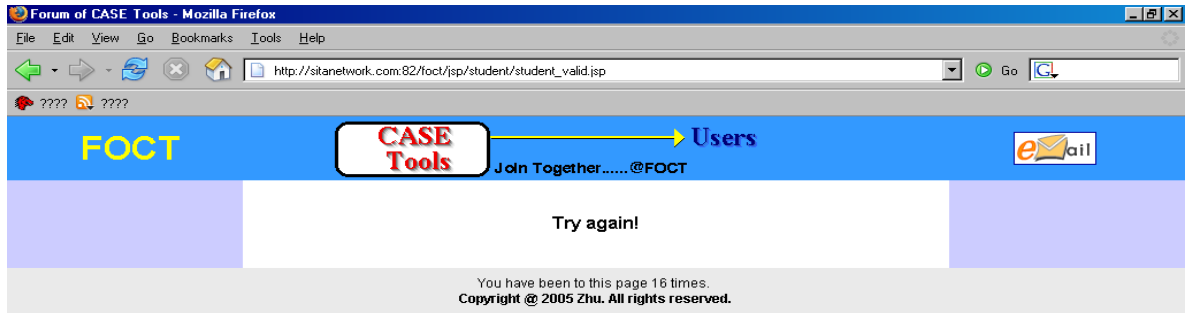
Figure 4.22: Error Checking for Student Login
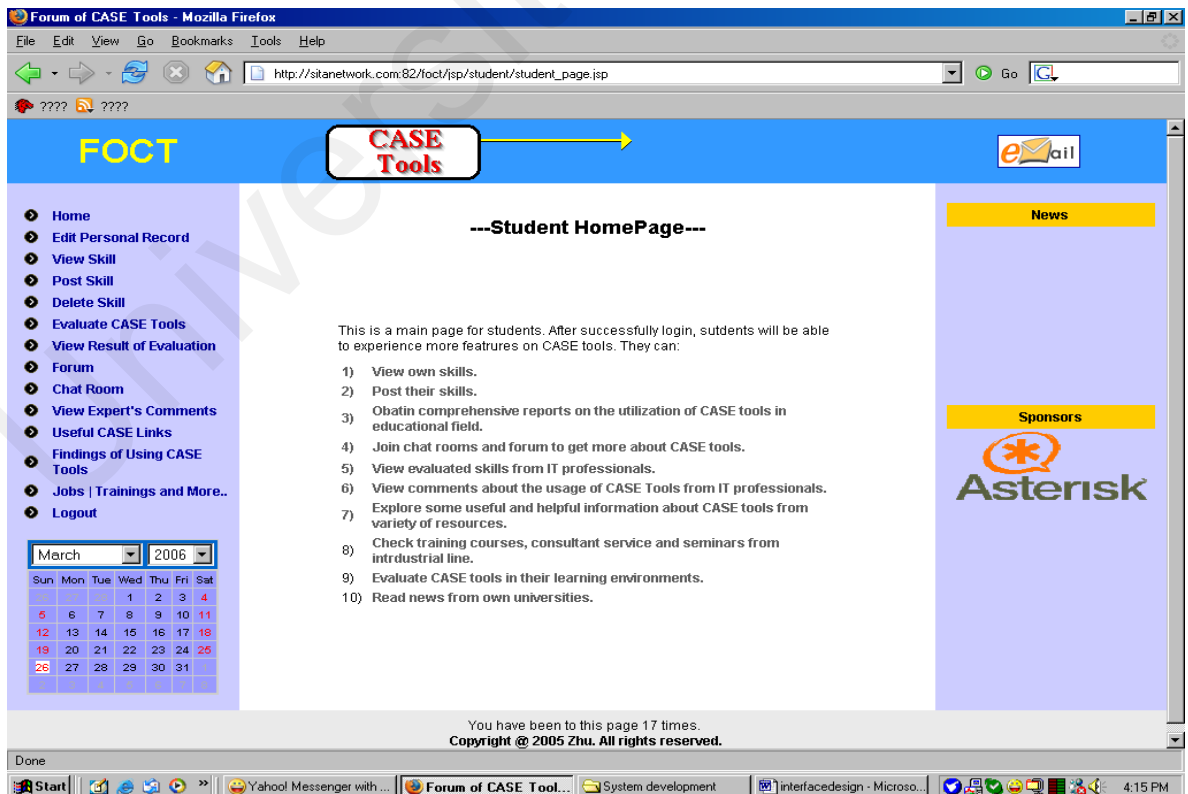
Figure 4.23: Entered Validated Values

Figure 4.24: Form of Evaluation of CASE Tools
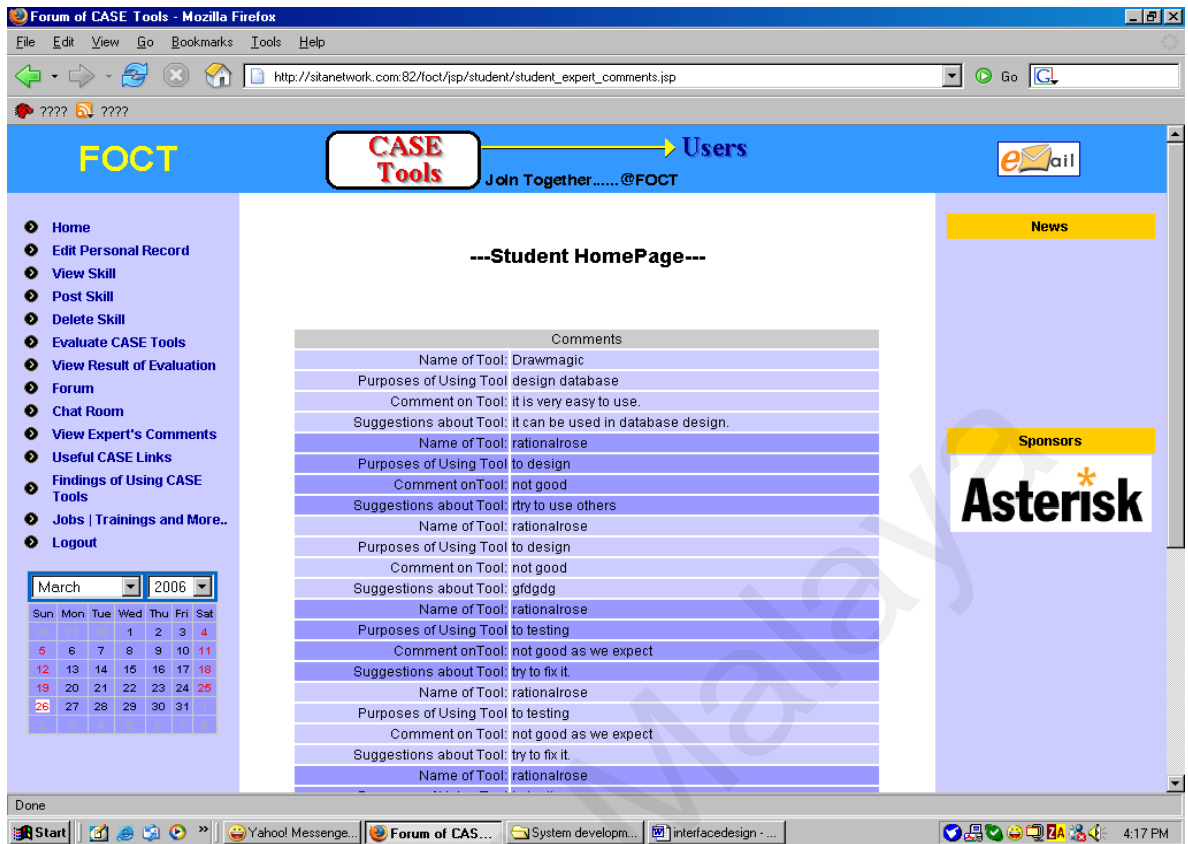

Figure 4.25: Forum

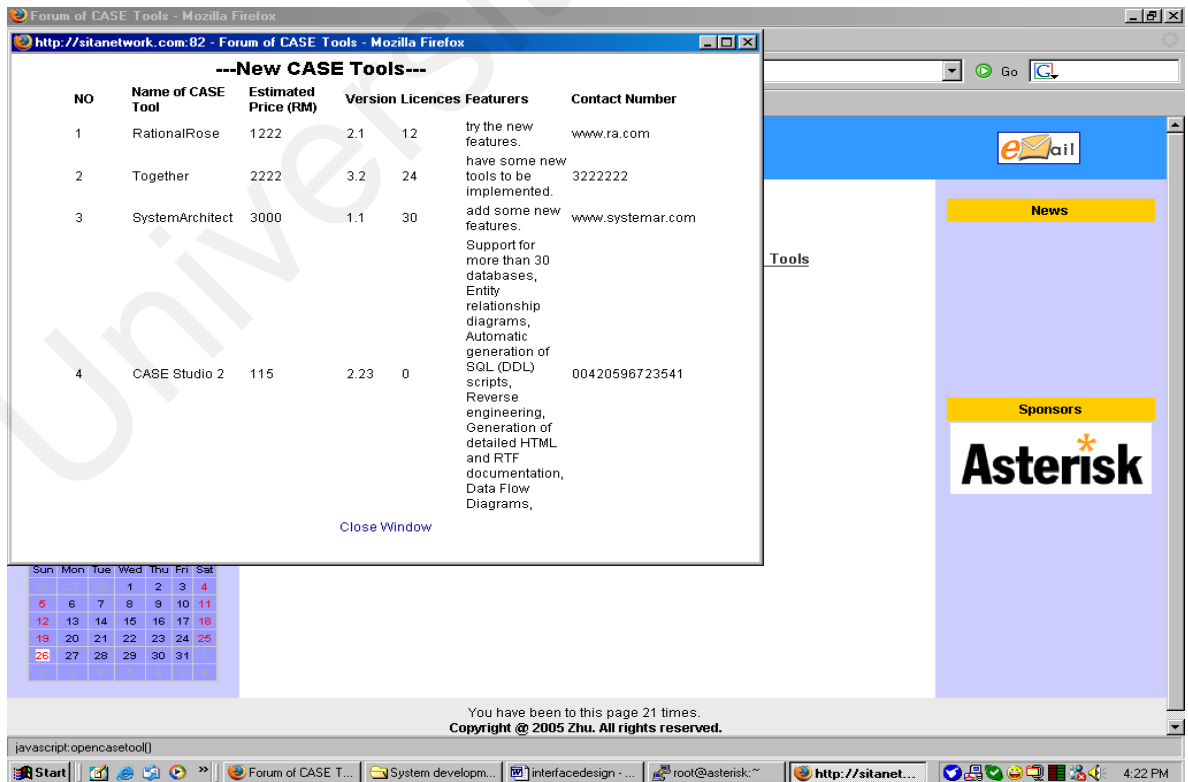**Figure 4.26: Viewing IT Professional's Comments about CASE Tools**
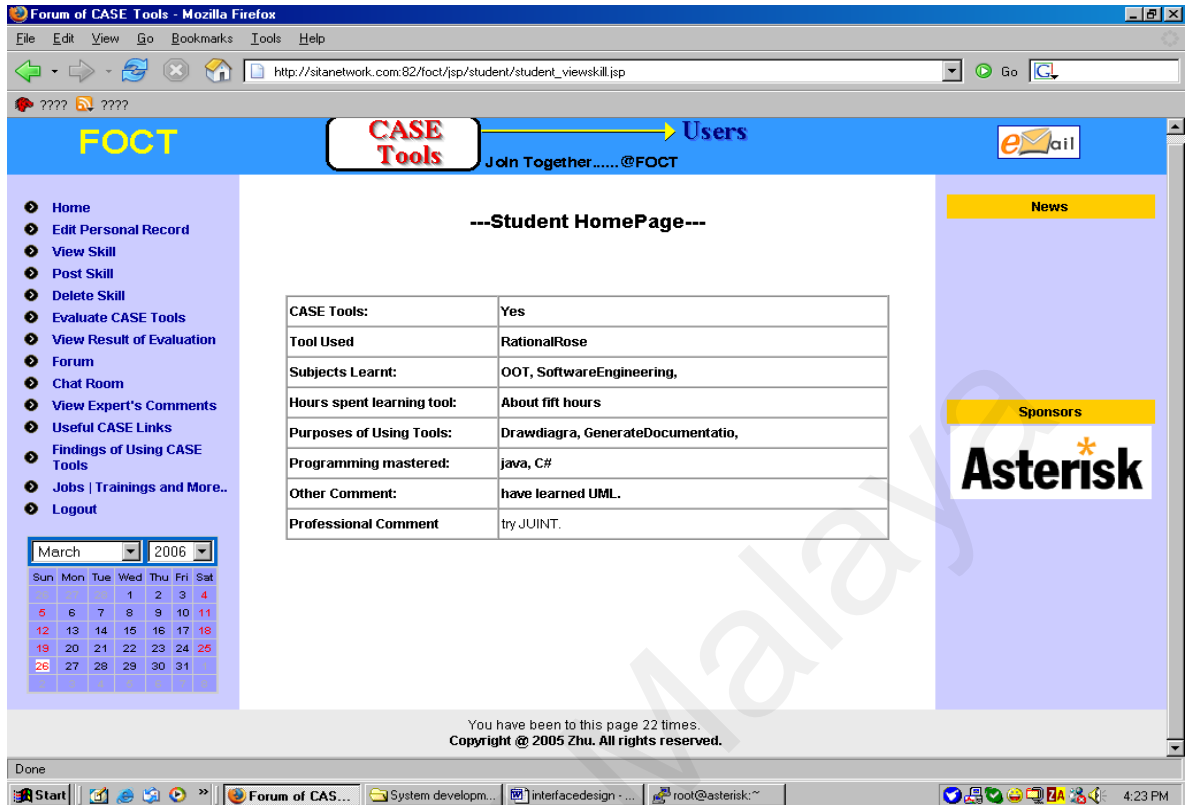

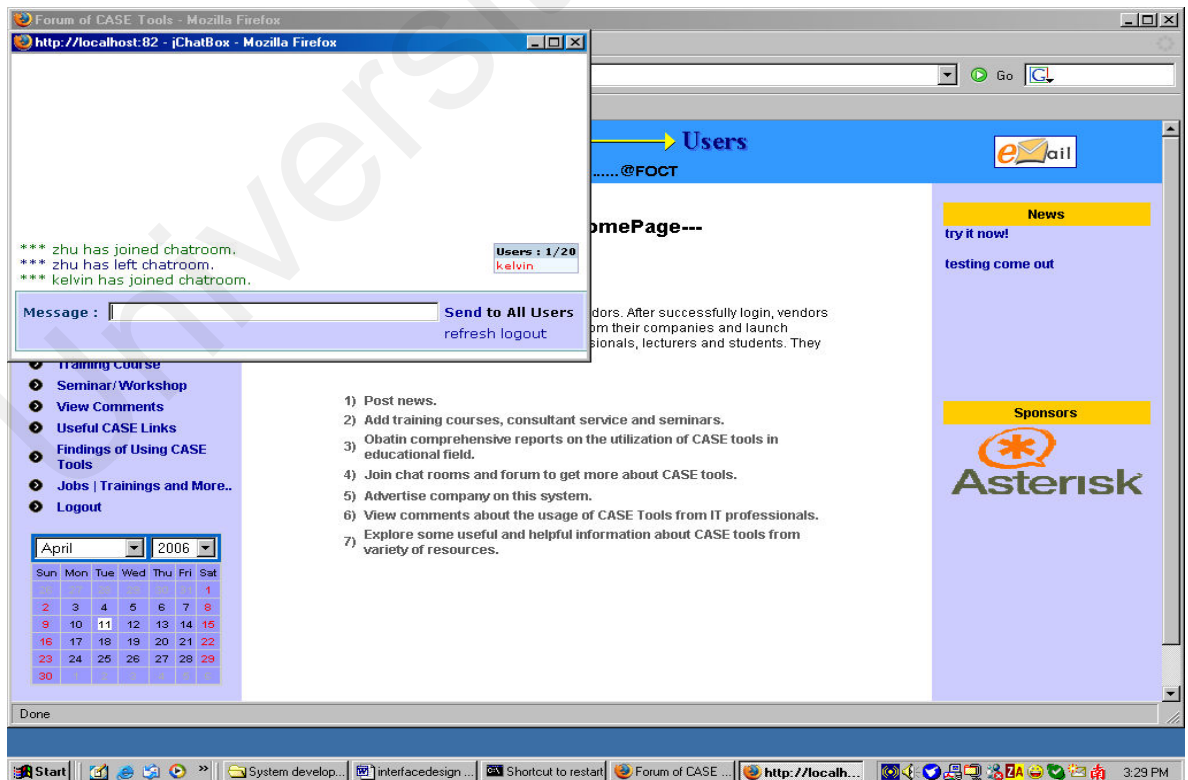**Figure 4.27: Checking CASE Tools**

Figure 4.28: Viewing Student's Skills

Figure 4.29: Joining Chat room
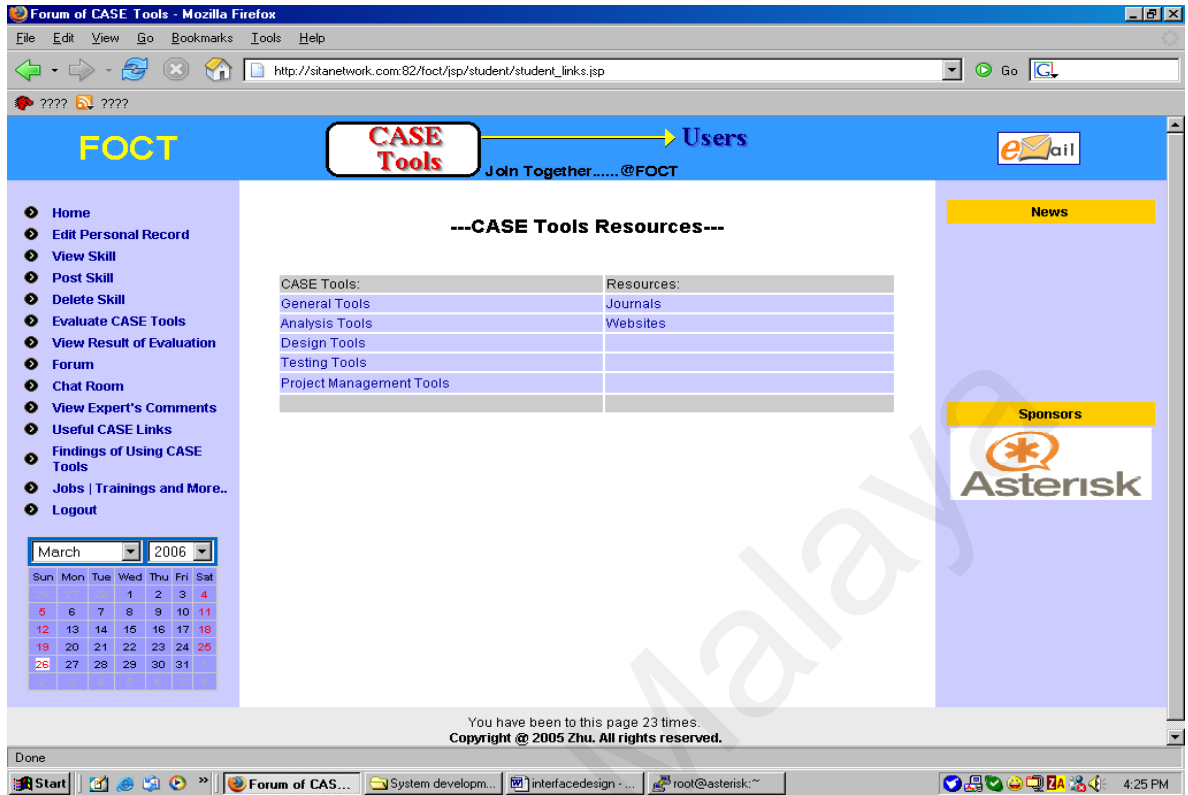
Figure 4.30: Showing CASE Tools Resources
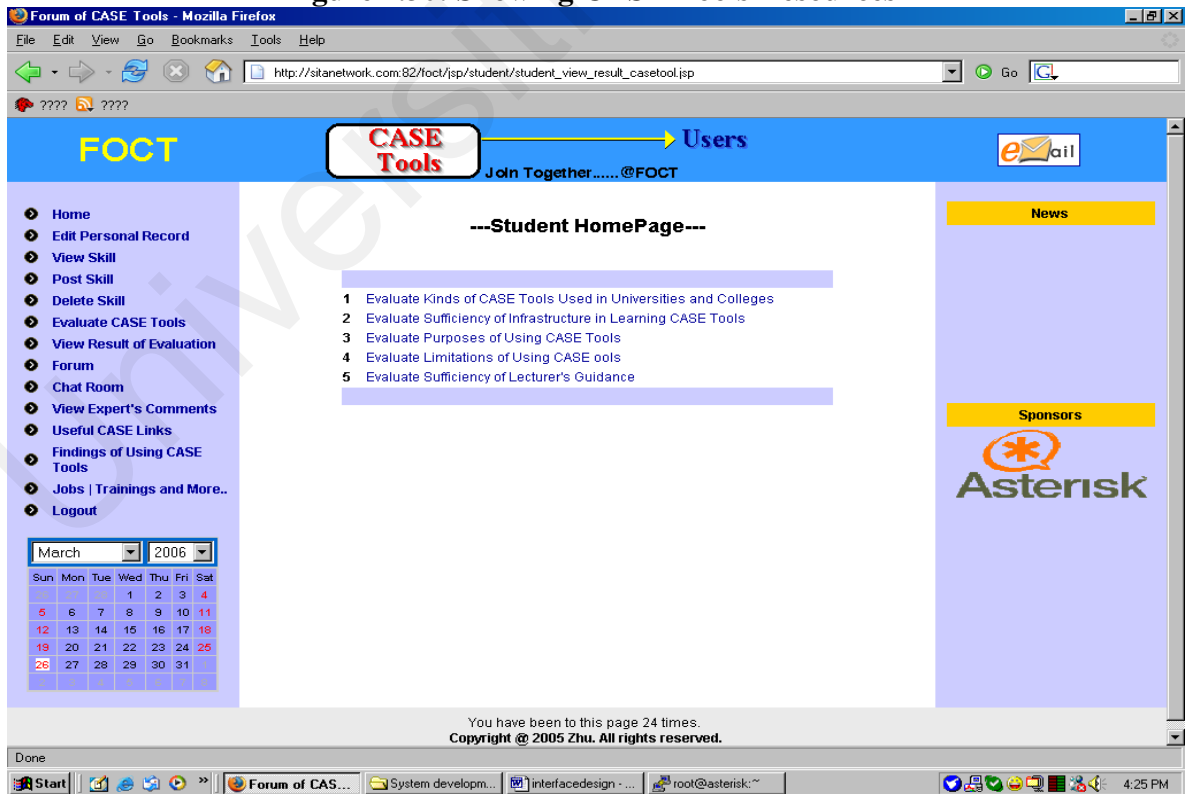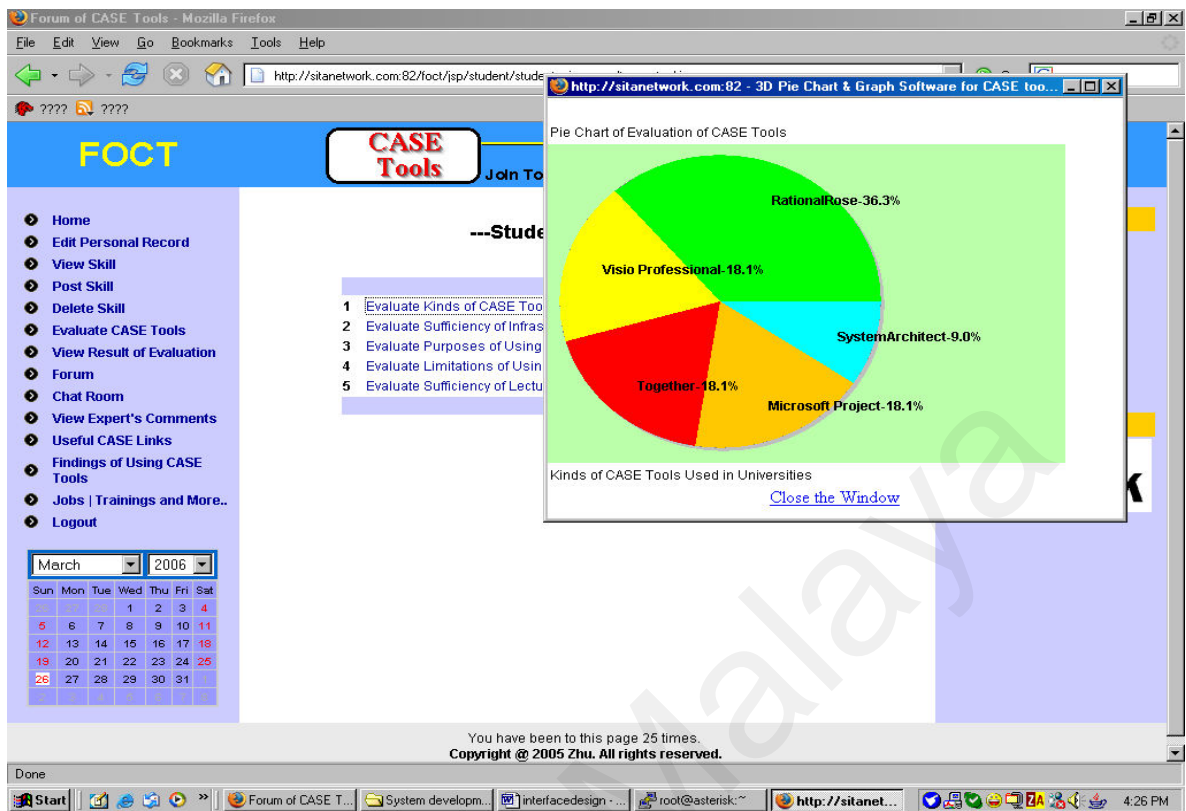

Figure 4.31: Viewing the Result of Using CASE Tools

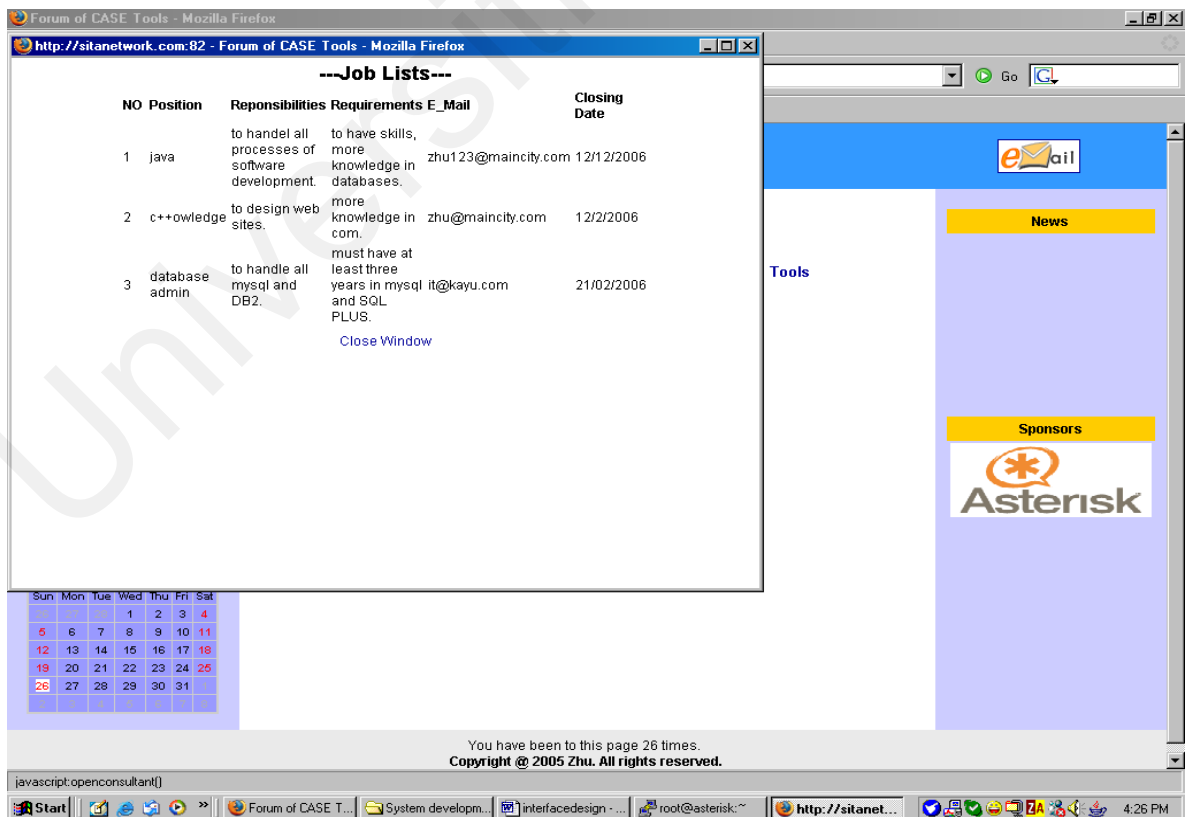**Figure 4.32: Showing the Result of Using CASE Tools**
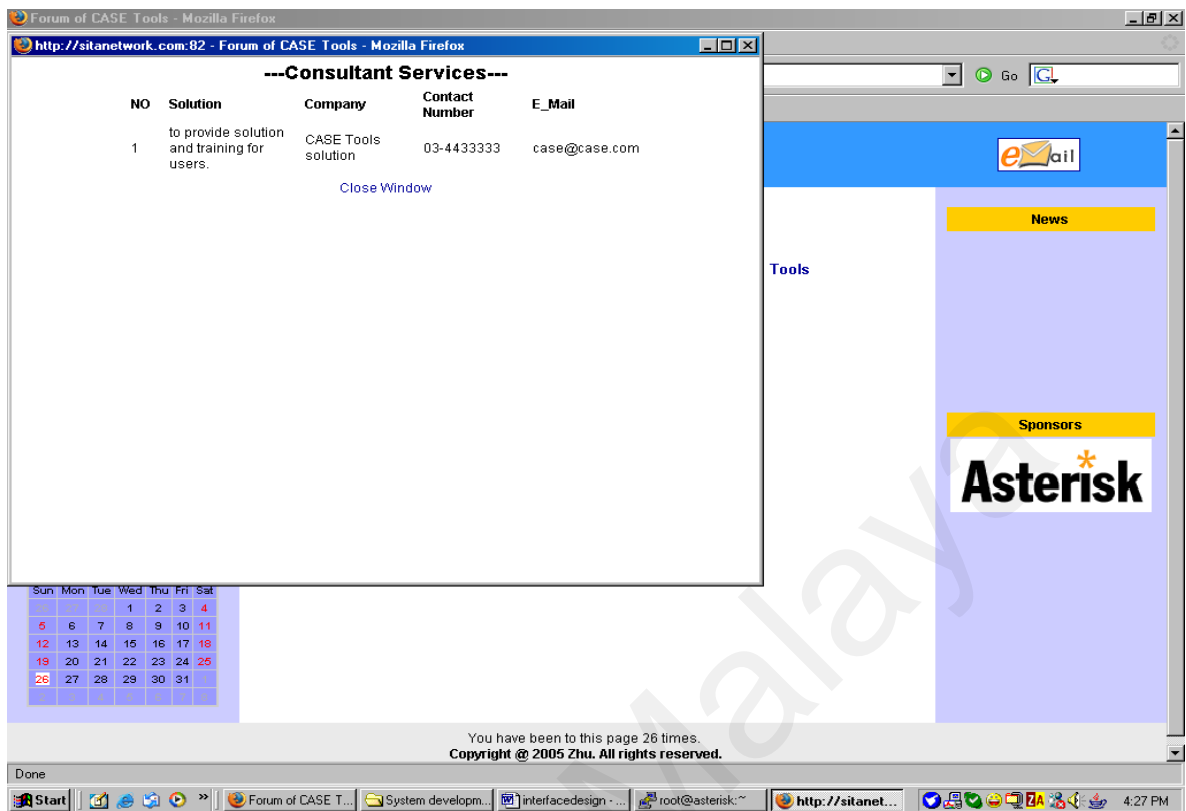

Figure 4.33: Showing Job lists
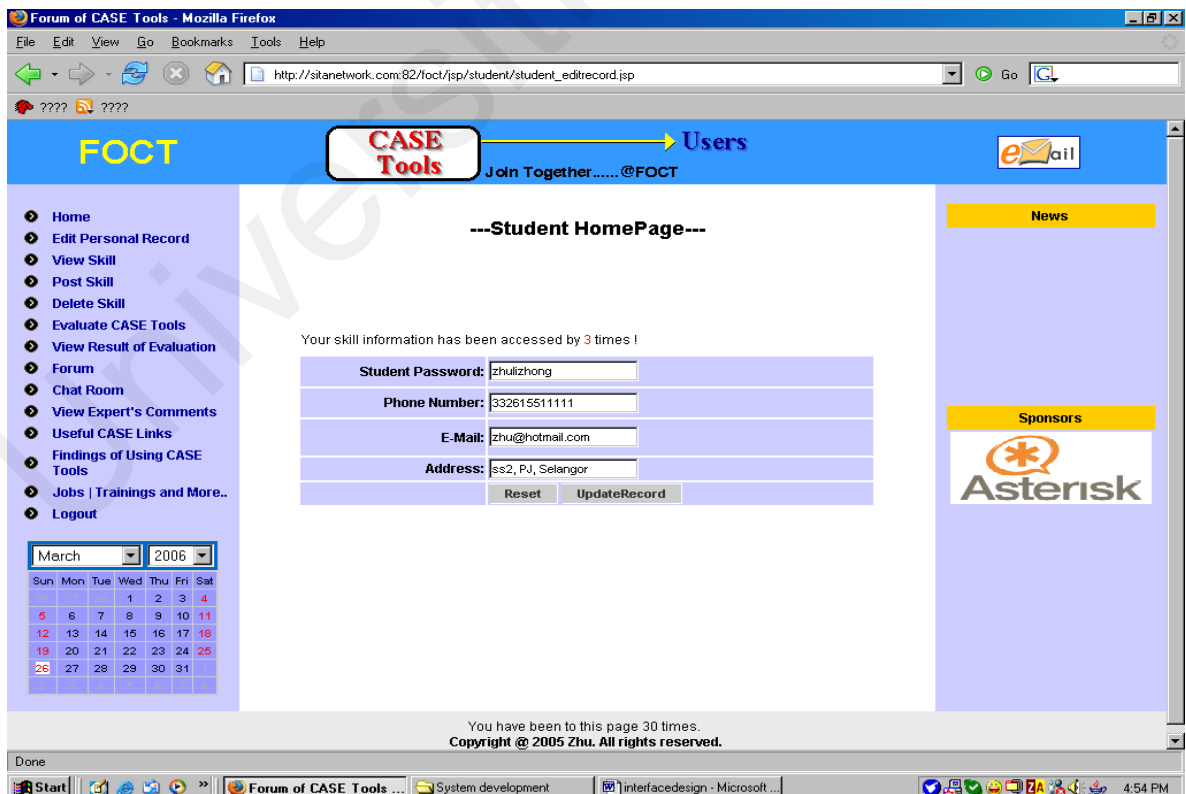
Figure 4.34: Viewing Consultant Service
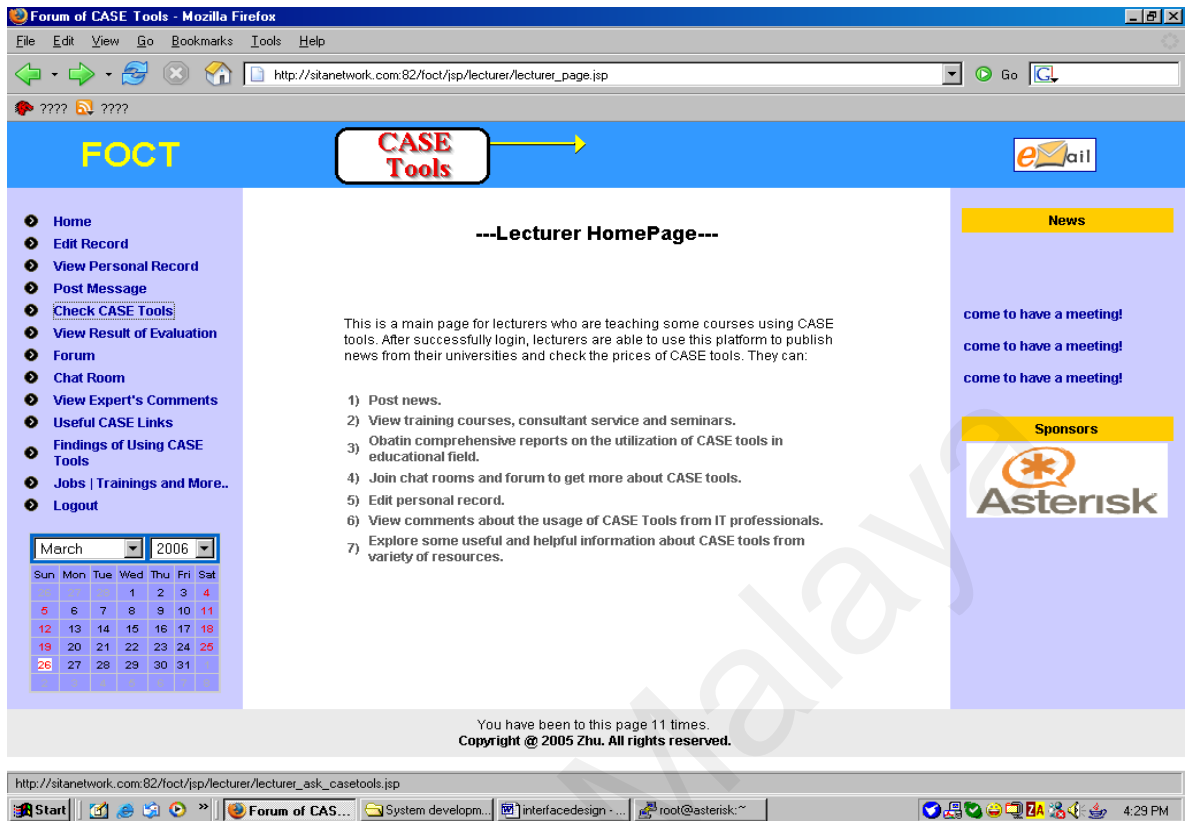

Figure 4.35: Viewing Student's Own Records

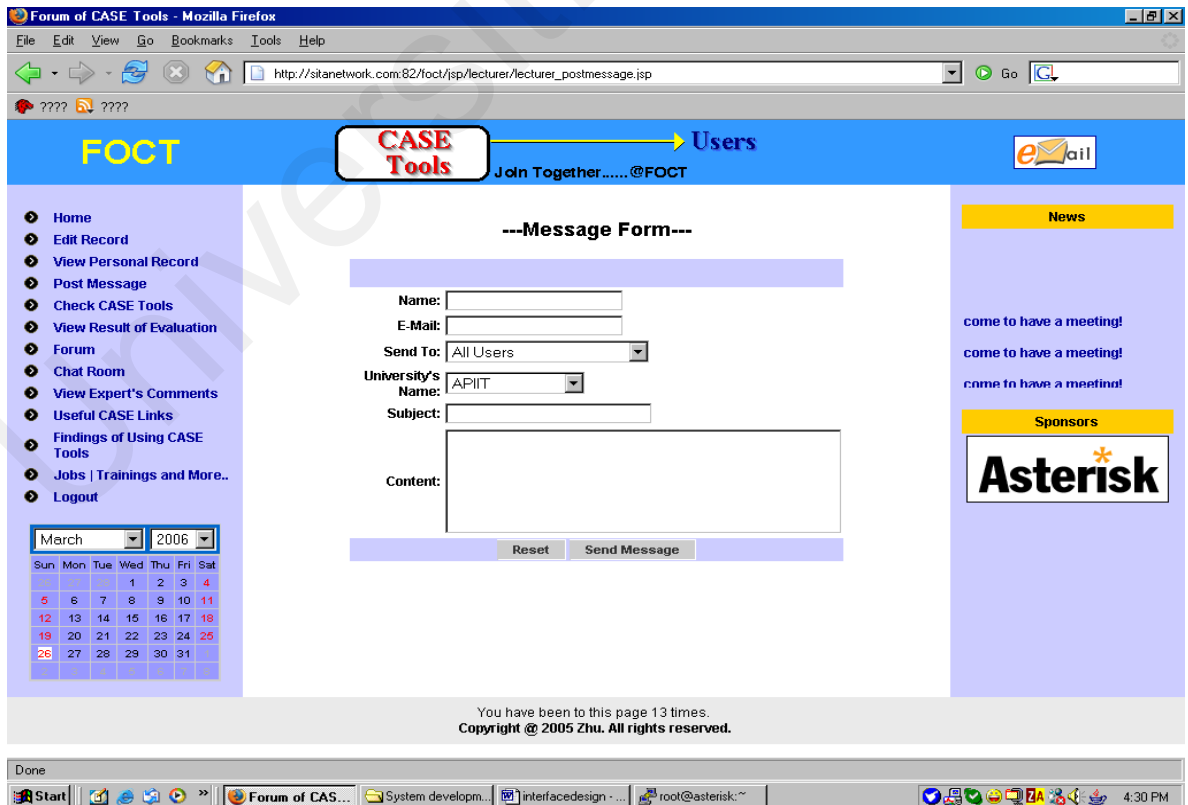**Figure 4.36: Evaluating Lecturer's Homepage**

Figure 4.37: Adding a New Message
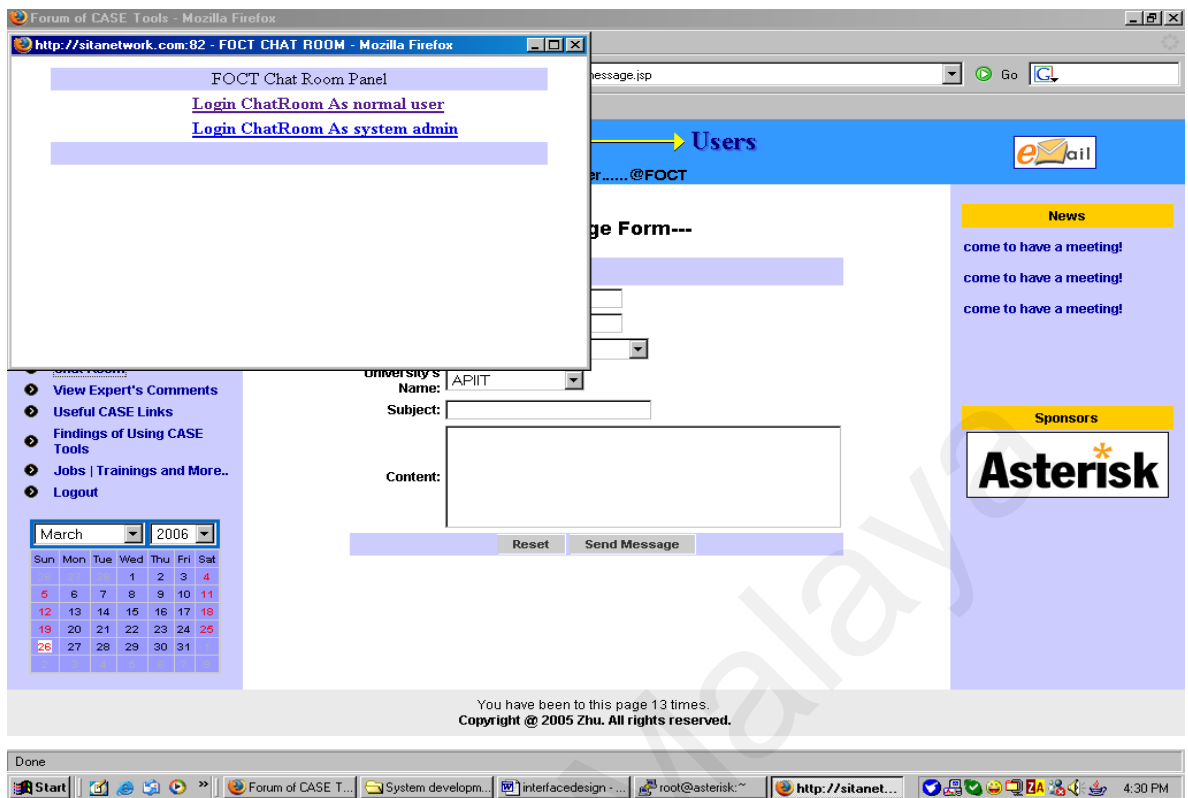
Figure 4.38: Login Chat Room
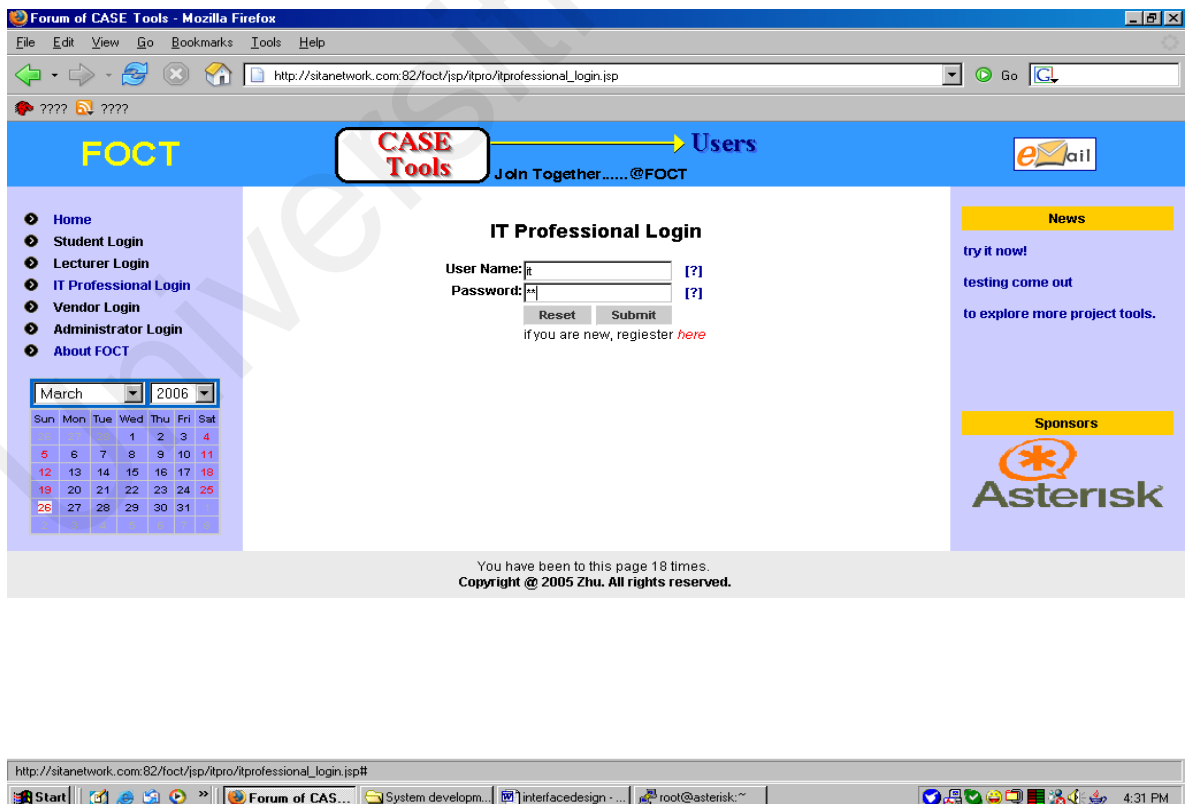
Figure 4.39: IT Professional Login
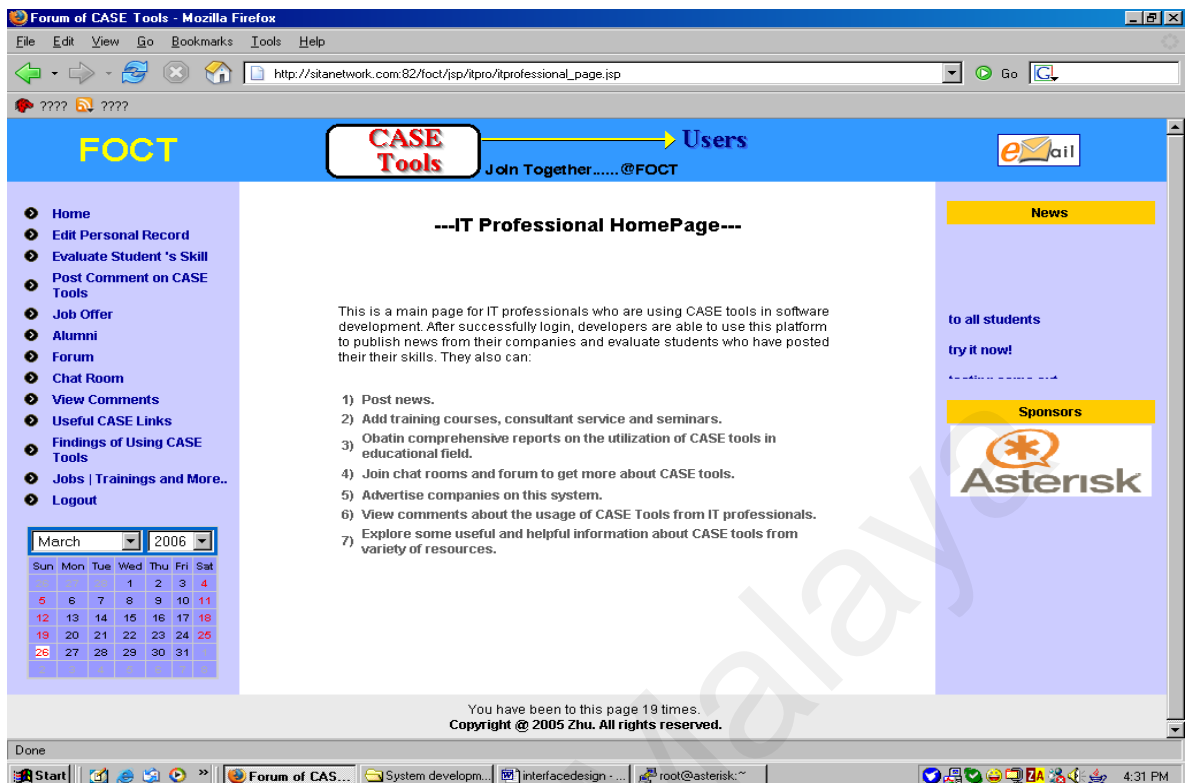
Figure 4.40: IT Professional Page

Figure 4.41: Posting a New Comment

Figure 4.42: Evaluate and Viewing Potential Employees

Figure 4.43: Adding a New Job

Figure 4.44: Vendor Login

Figure 4.45: Vendor Homepage

Figure 4.46: Adding a New Tool
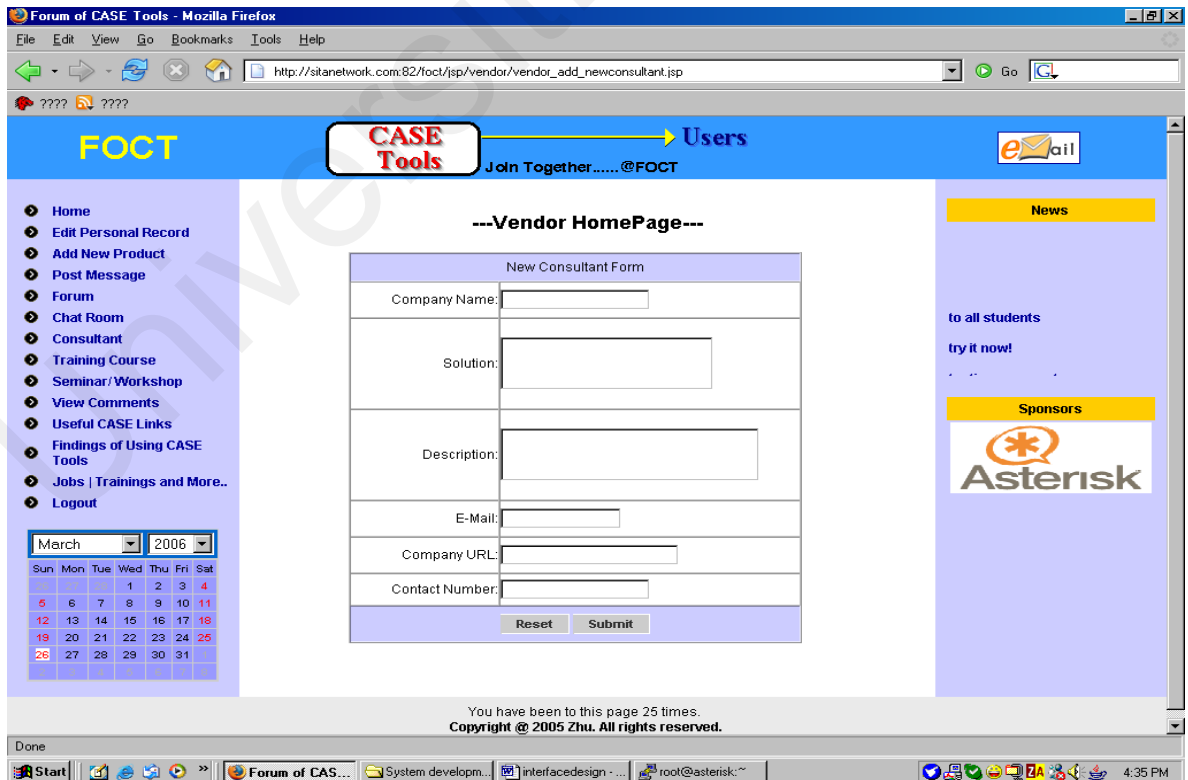
Figure 4.47: Adding a New Service
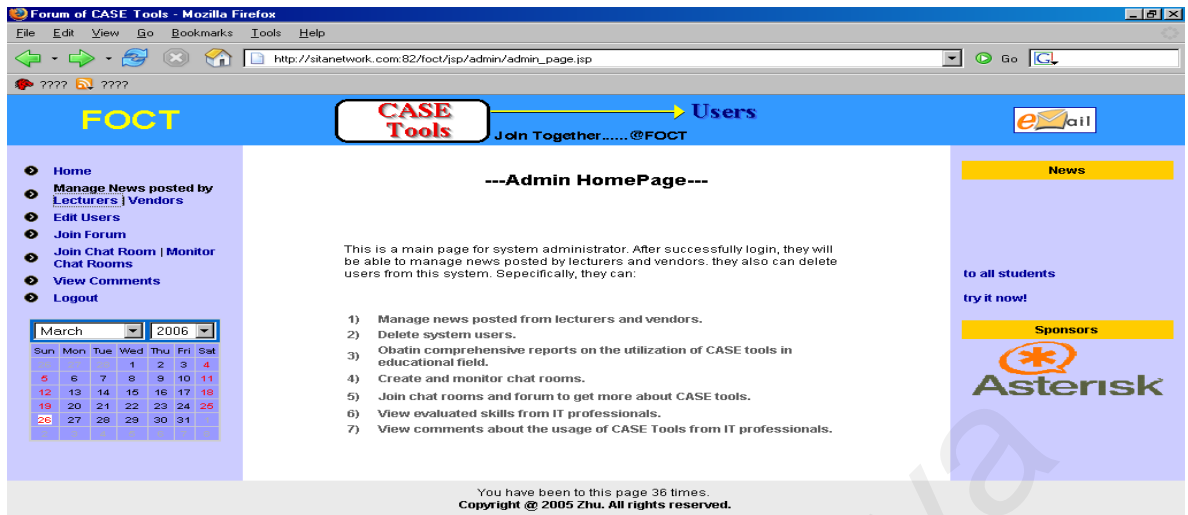
Figure 4.48: Administrator Homepage
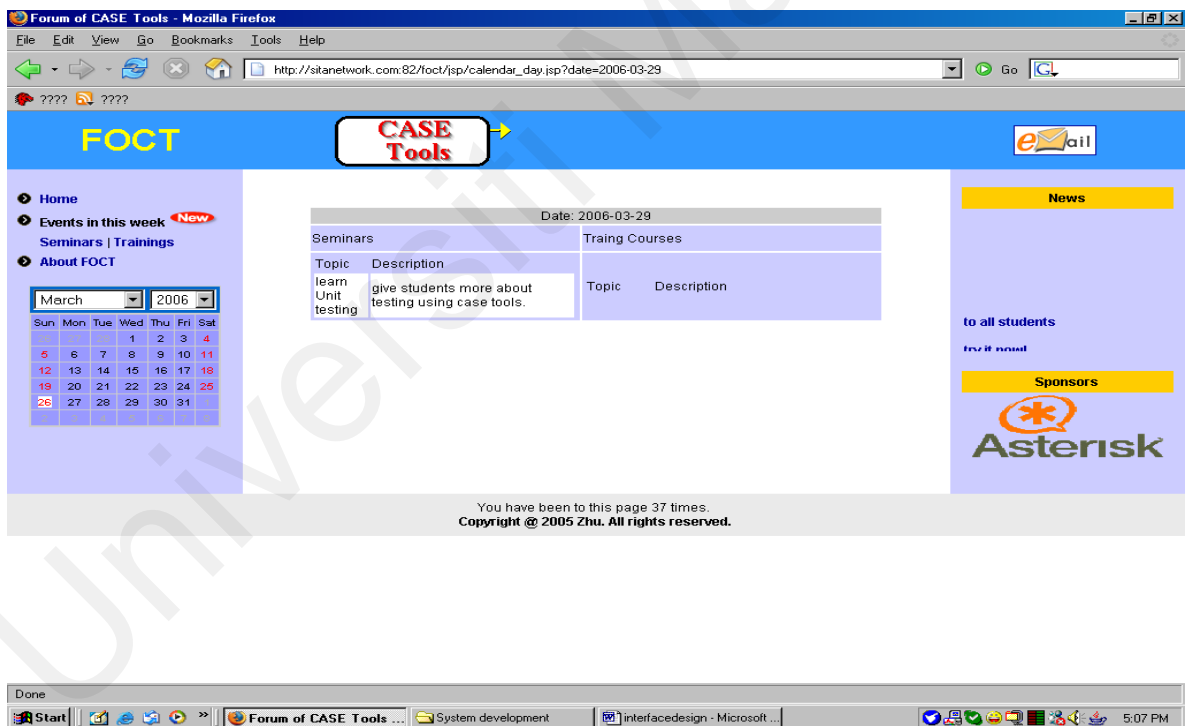
Figure 4.49: Checking Events Using WEB Calendar
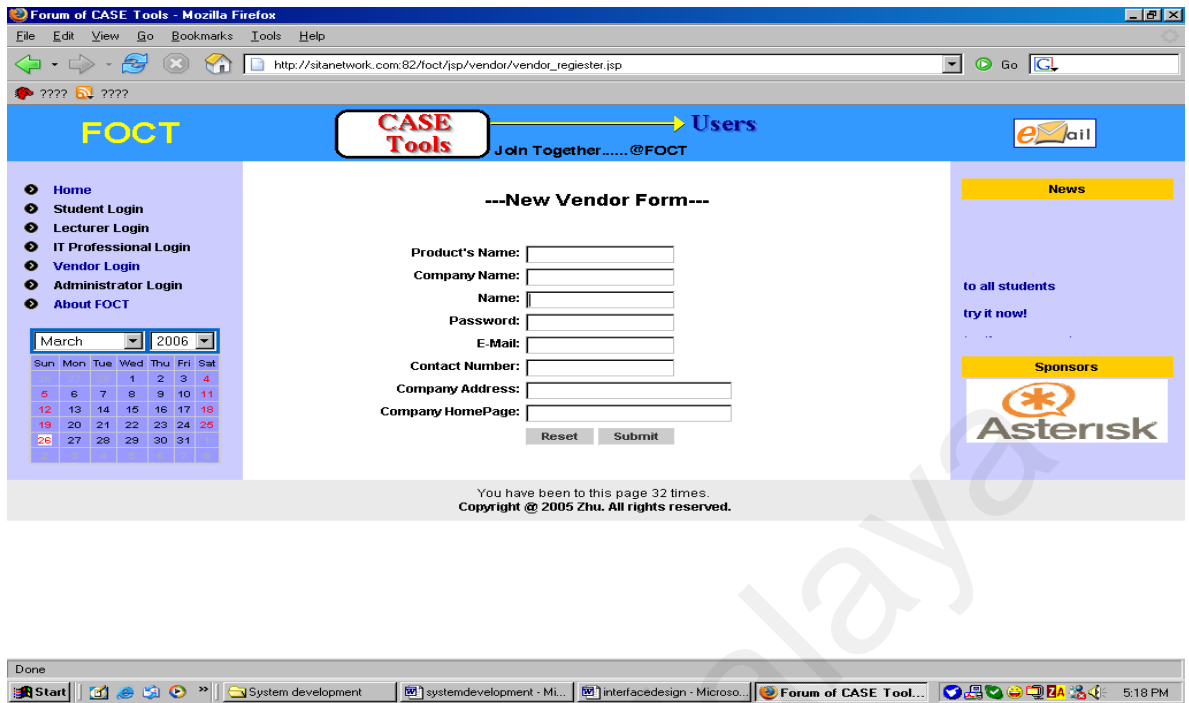
Figure 4.50: New Vendor Form

4.13 System Development

FOCT system is to provide a forum where lecturers, students, IT professionals, system administrator and vendors can contribut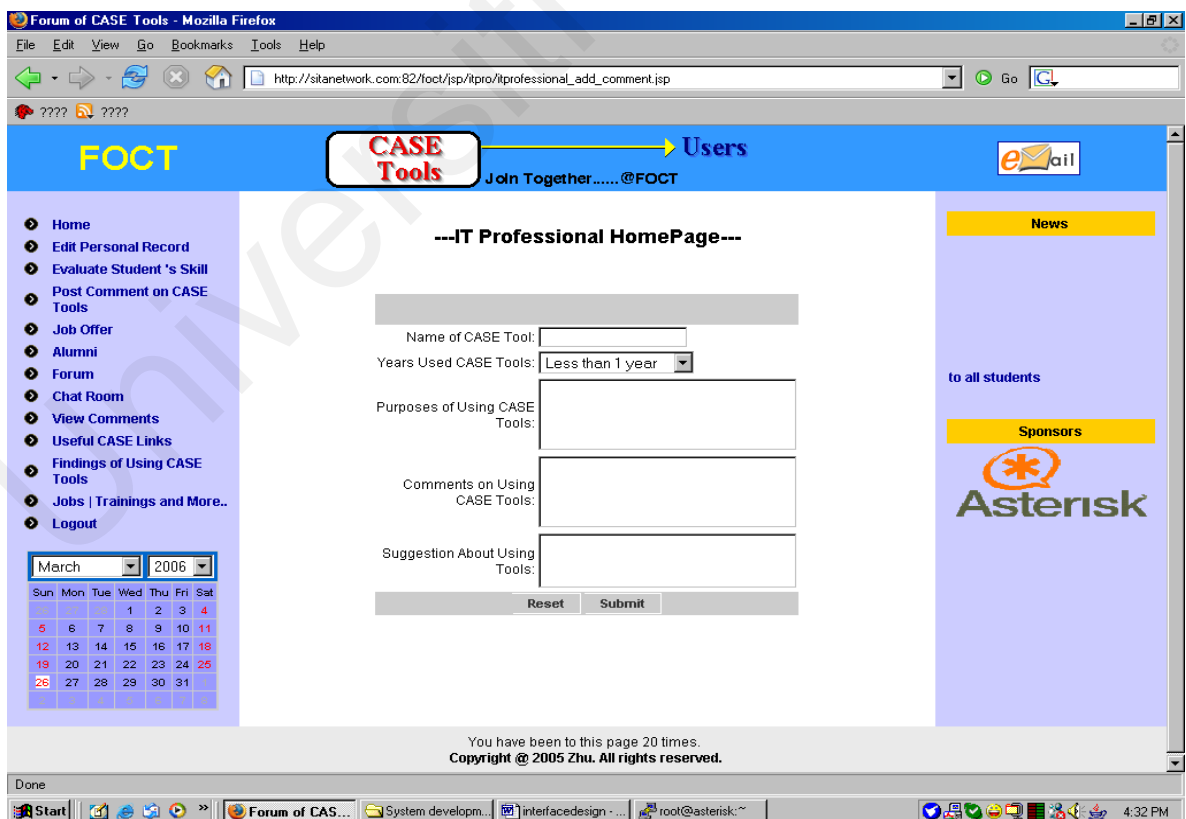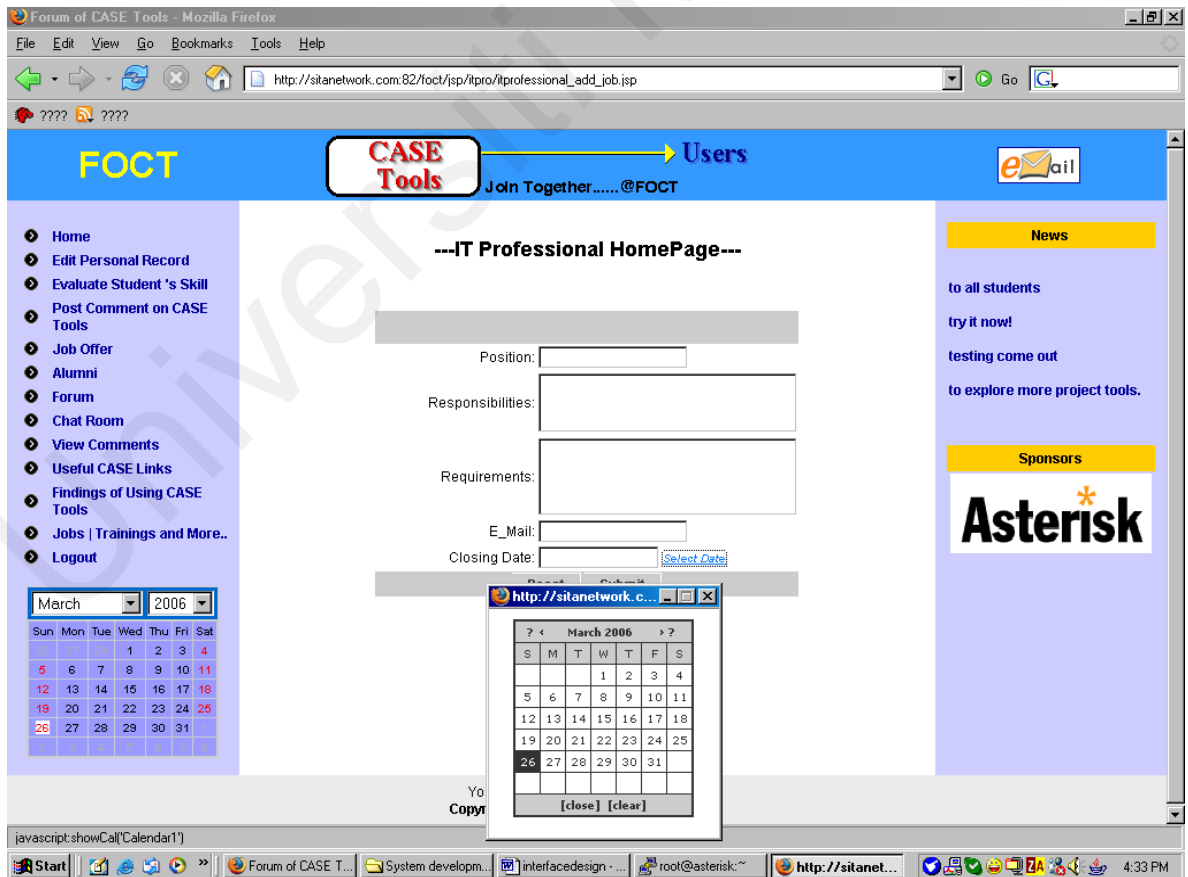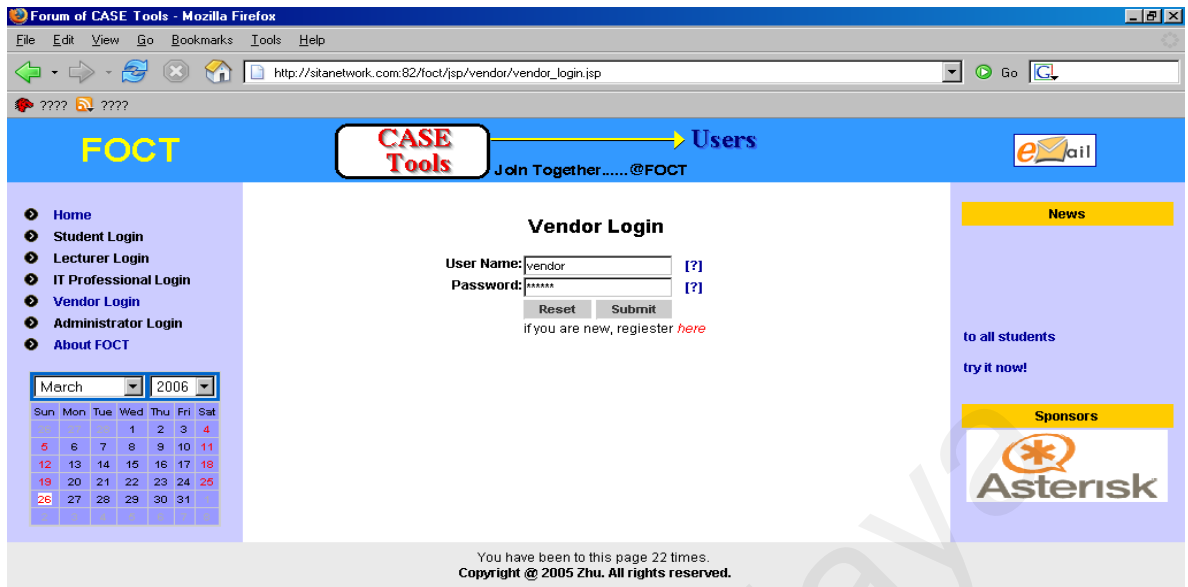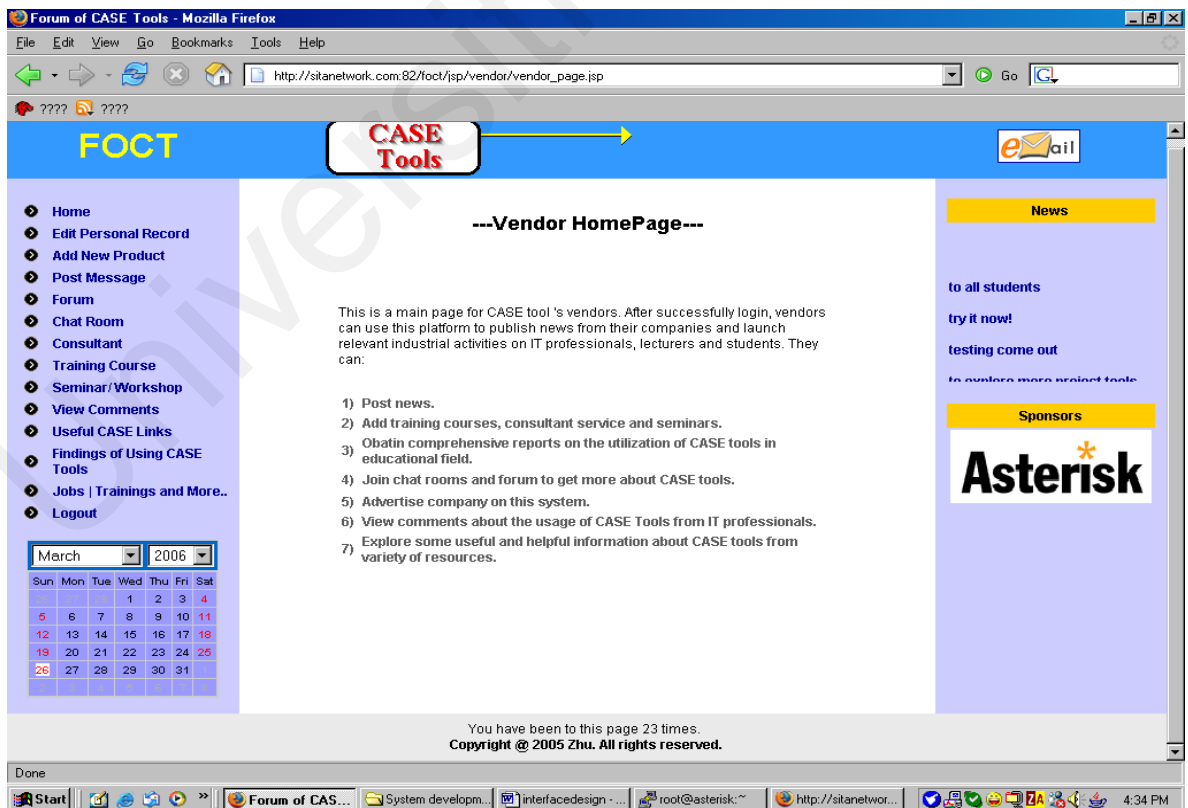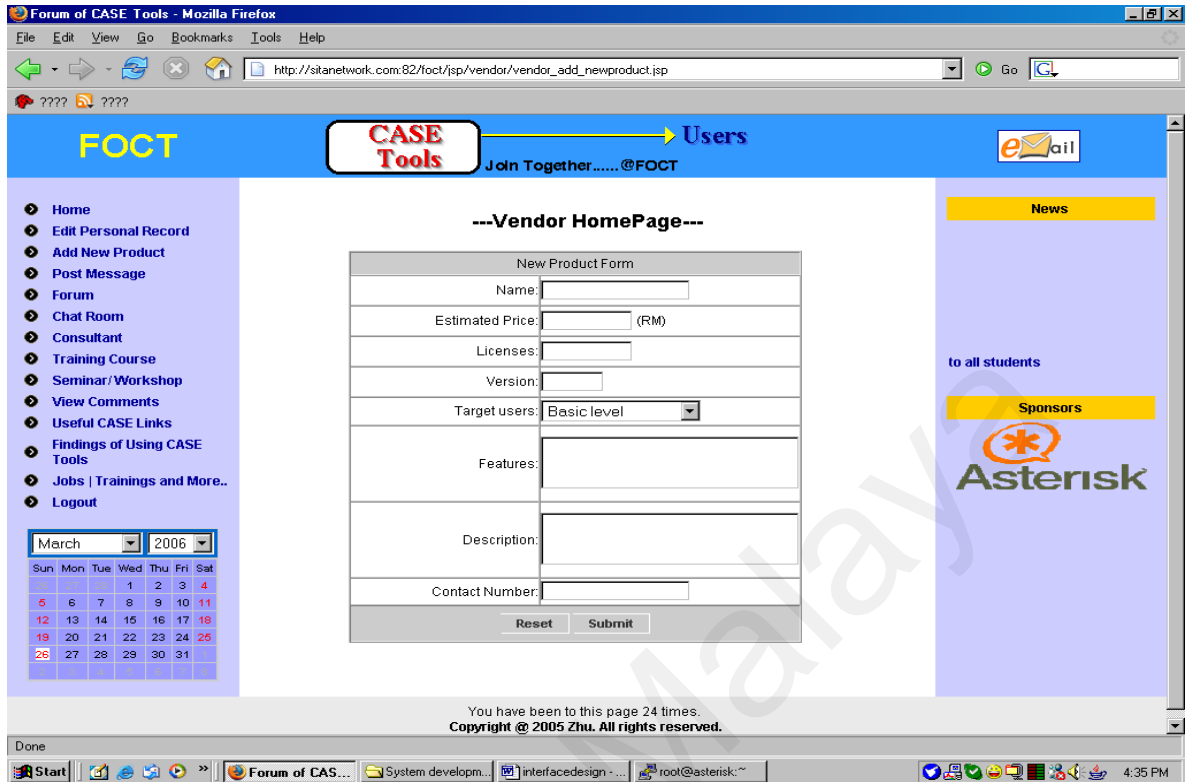e their knowledge in using CASE tools in Malaysia. In order to meet all the requirements from the five groups of people, the researcher needed to make a research on system development. The researcher collected data and necessary information through surveys, interviews and literature review.

4.13.1 Surveys

To obtain the data and information from both lecturers and students, the researcher used the data described in chapter three. The data obtained was sufficient for the researcher to develop the FOCT system.

4.13.2 Interviews

The researcher interviewed Mr. Kelvin who is an IT professional in using open source CASE tools. He gave many valuable suggestions about the system design. The researcher also interviewed a few CASE tools vendors via e-mail.

4.13.3 Literature Review

There had been much literature review done in CASE tools before. The researcher referenced the literature described in chapter two. It gave the researcher a clear picture on utilization of CASE tools. Through reading these papers, the researcher received a lot of helpful information and data about CASE tools. This helped the researcher to develop FOCT system.

4.14 Testing

Testing is a very important step to ensure that FOCT runs successfully. Shelly (2003) gave

a guideline for testing. This is shown in Figure 4.56.
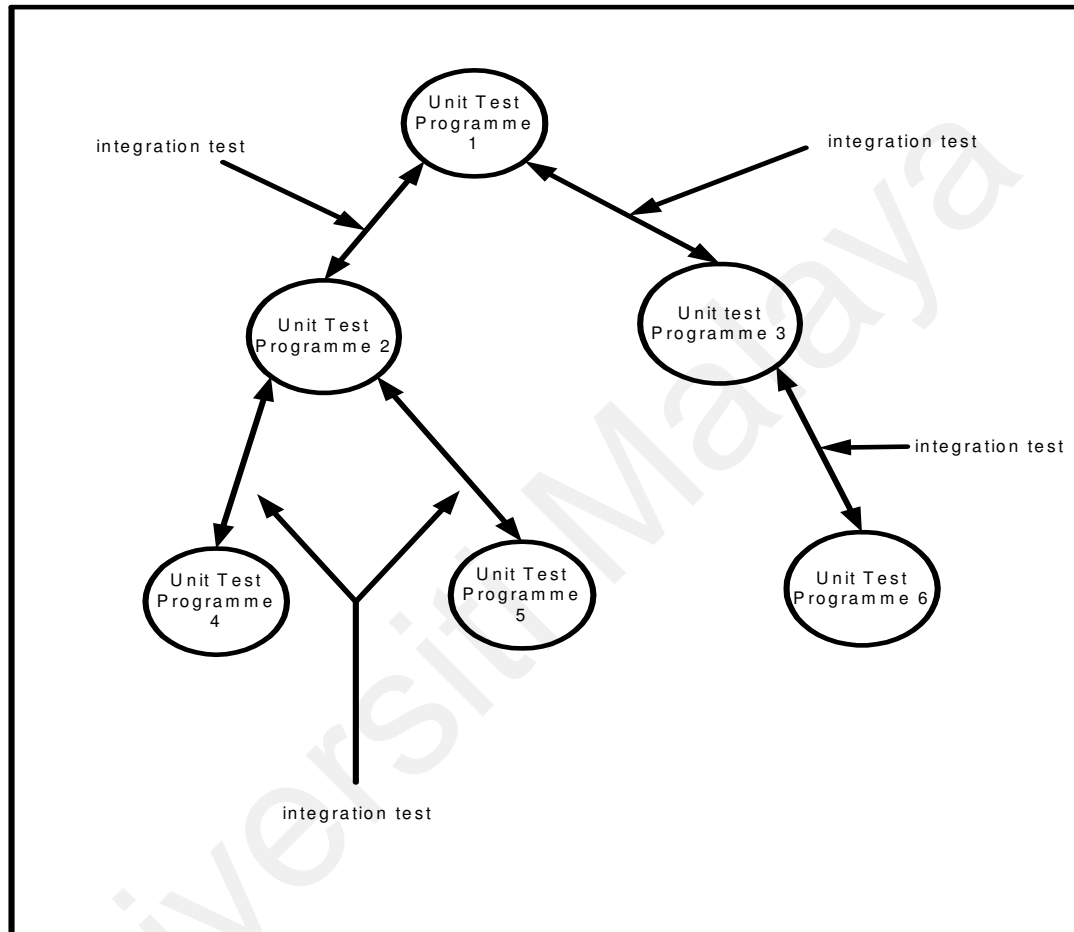
Figure 4.51: Testing Guidelines

Resource (Shelly ,2003)

1) Unit Testing

Unit testing is used to test an individual programme or module and identify errors.

The test data should contain correct data and incorrect data. In FOCT, there are

many programmes to be tested. Here, the researcher tested the login program as an

example to see whether the programme could run correctly.

166
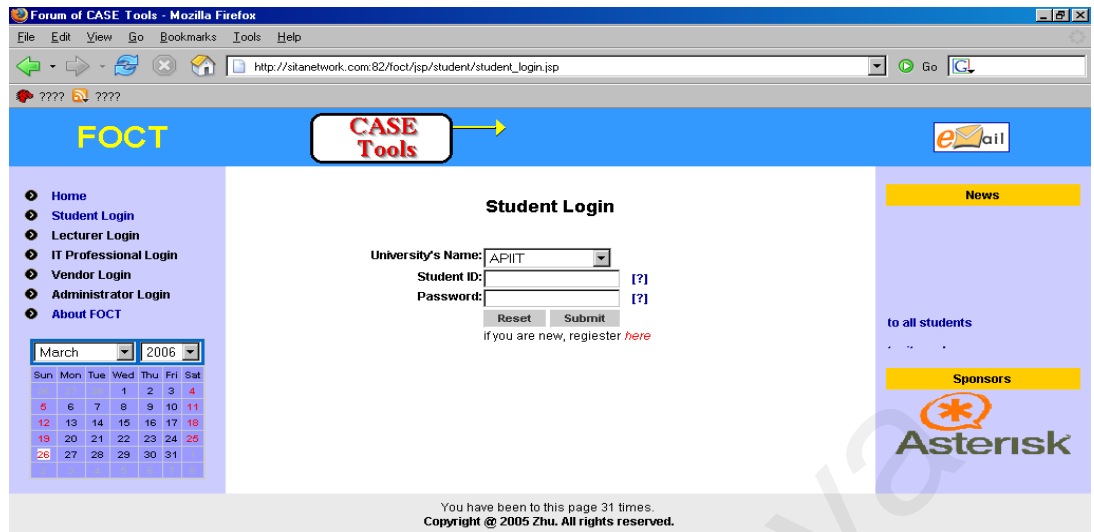
Figure 4.52: Login Checking Without Value

After the user enters the student number and password, the system will check

whether one of the fields is missing or one of the fields is empty. If that happens

the system will give an error message as shown in Figure 4.53.
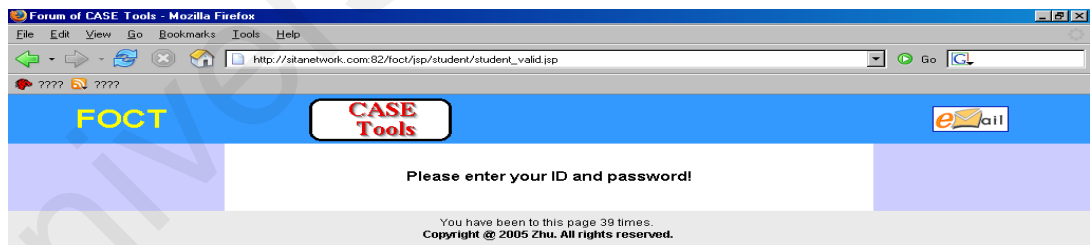
Figure 4.53: Error Message Displayed
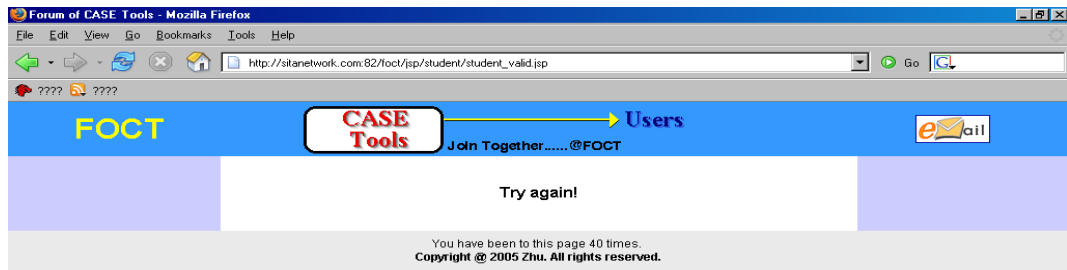
Figure 4.54: Enter Incorrect Values

If the user entered incorrect values, the system would tell the user to check the

input and request the users to try again as show in Figure 4.54 and Figure 4.55.
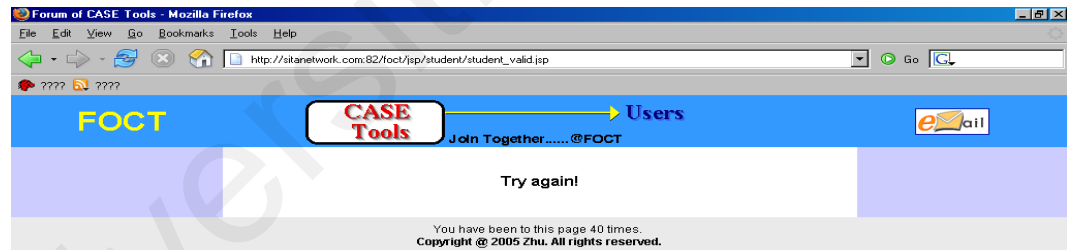
Figure 4.55: Error Messages

If the user entered correct values, the system will bring them into the right

pages as shown in Figure 4.56.
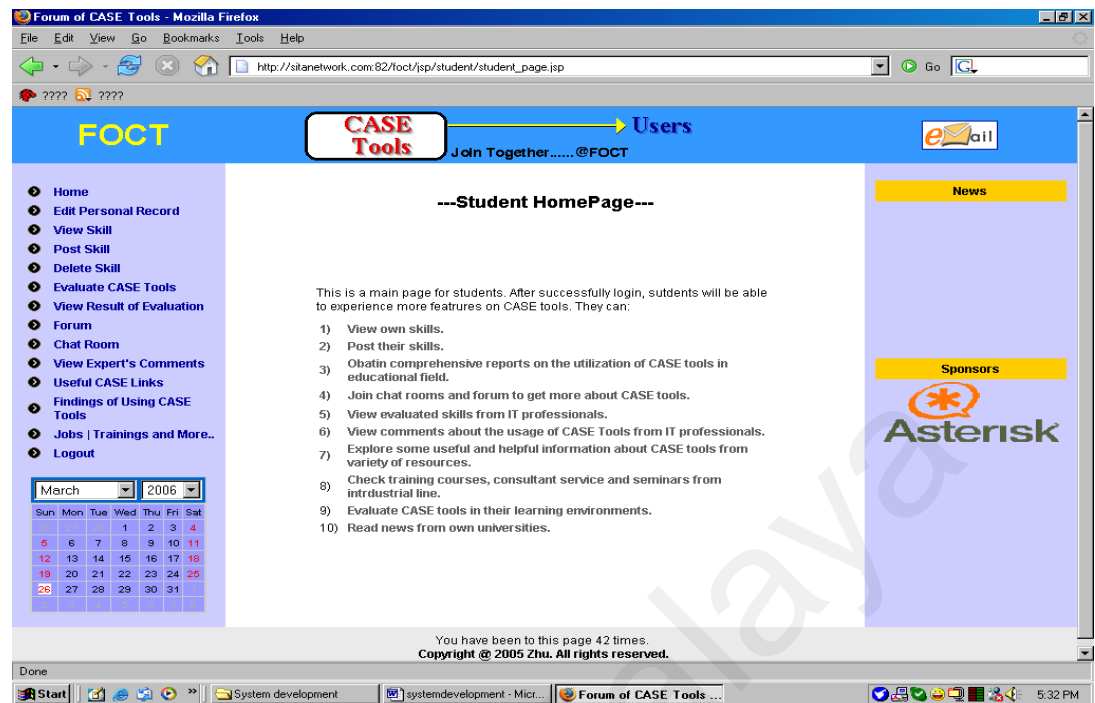
Figure 4.56: Entered Correct Values

2) Integration Testing

After the unit testing, the researcher executed the integration testing. Integration testing is to test the dependency between the programmes. Here, the researcher tested the edit record and update record programmes. Once the user login into FOCT, the user can edit and update his or her data as shown in Figure 4.57.
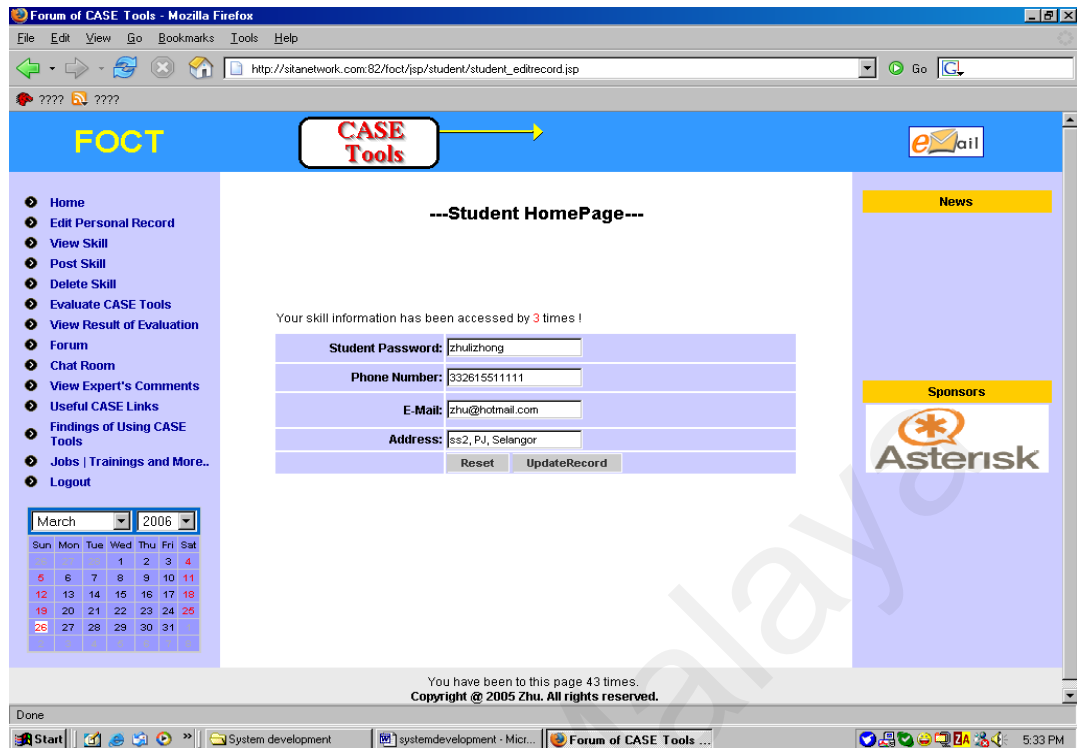
Figure 4.57: Display Changed Records in FOCT

3) System Testing

After the integration testing, the final step is system testing. In the system testing, the researcher performed all the features under different conditions. Eventually, the FOCT ran successfully.

4) End User Testing

The researcher invited thirty-one users to test this system. The breakdown of the users is shown in Figure 4.58. After the testing, the end users filled an on-line evaluation form as shown in Appendix B. The users were asked to answer ten questions. The ten questions covered various aspects of the system including GUI, usability, security, performance, satisfaction and helpfulness. A summary of the feedback is presented below.
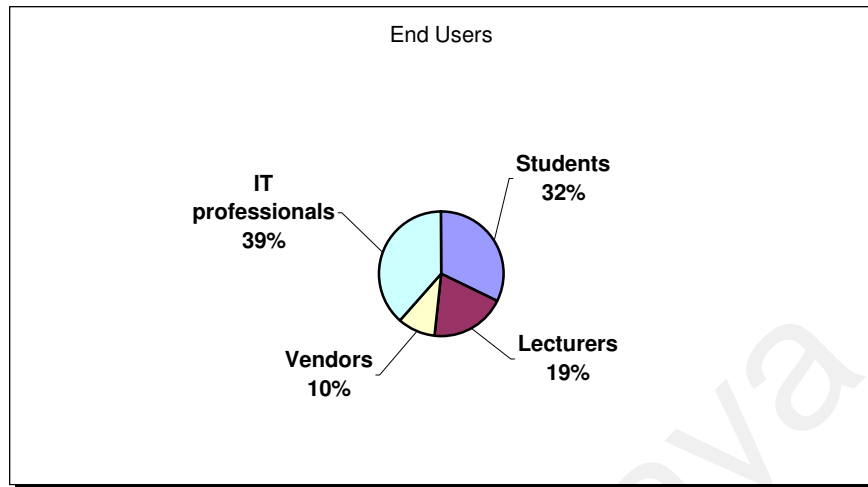
Figure 4.58: Overall of End Users

1) Interface Design

The overall of GUI design is good. The results are shown in Figure 4.59.
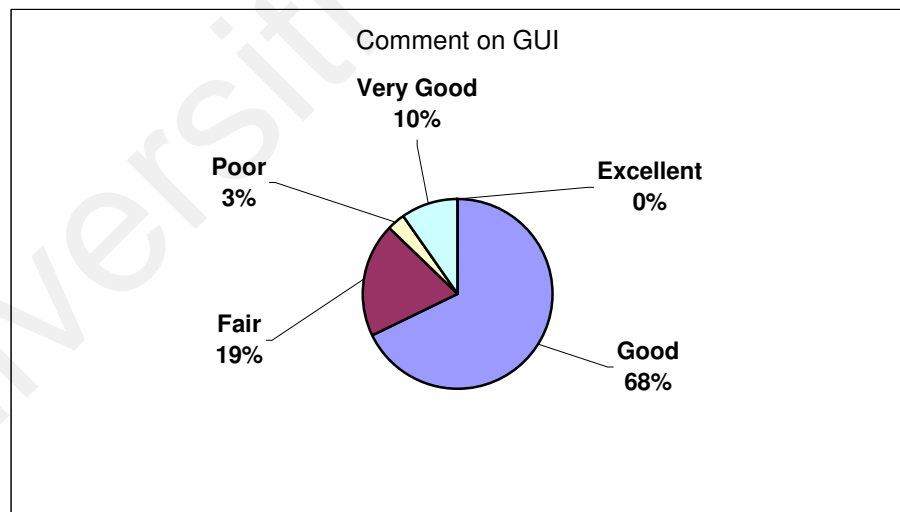
Figure 4.59: Comment on System GUI

2) Usability

In this section, the features tested by the users include forum, chat room,

messages, activities, case tool evaluation and skill evaluations.

Table 4.27: Results of Usability

Features	Scales				
	Poor	Fair	Good	Very Good	Excellent
Chat room	4	12	11	4	0
Forum	7	5	11	8	0
Expert's comment	3	4	13	10	1
CASE links	4	2	11	14	0
Event-training	1	7	10	8	5
Event-consultant	1	6	14	6	4
Event-job	2	7	8	9	5
Event-seminar	1	3	16	10	1
Event-product	0	6	15	6	4
Evaluation of skills	2	4	13	6	6
Alumni	1	5	16	6	3
Survey of CASE tools	1	2	22	6	0
Report of usage of CASE tools	0	8	16	7	0
Web Calendar	0	4	18	8	1

The data in Table 4.27 indicates that the majority of users rated that the system features were good.

3) Security

Security is a key issue in software development. Figure 4.60 indicates that the majority of the end users said that this system was secure.
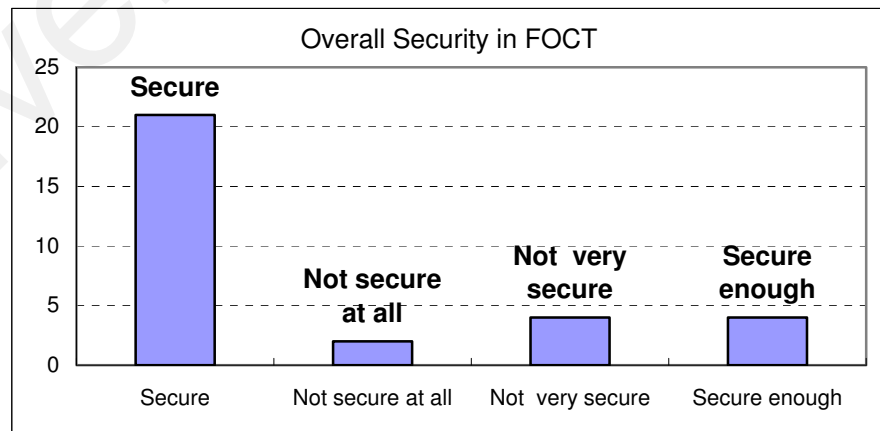
Figure 4.60: Overall of Security of FOCT

172

4) Performance

The system performance was tested by end users. Figure 4.61 shows that sixteen users said that the performance of the system was good.
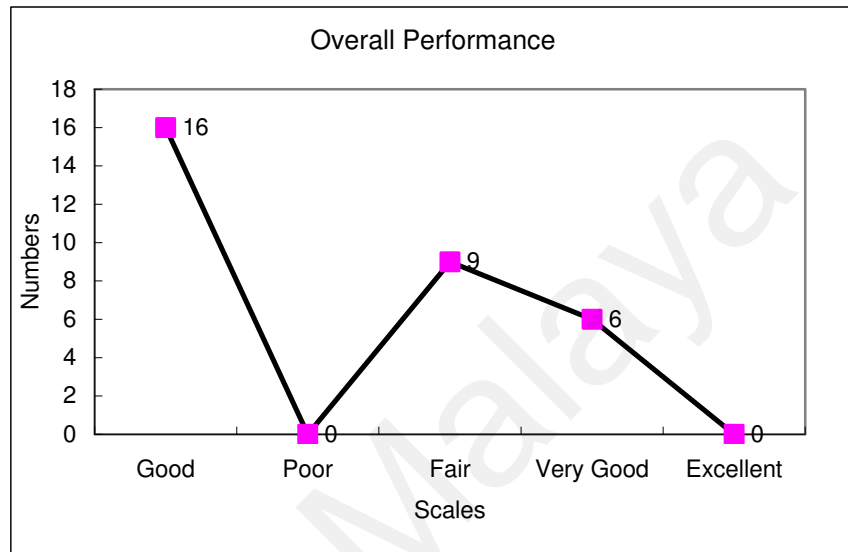
Figure 4.61: Performance of FOCT

5) Satisfaction

Satisfaction is very important for end users. The data in Figure 4.62 indicates that 94% of the end users were satisfied with the system with only 6% of the end users being not satisfied.
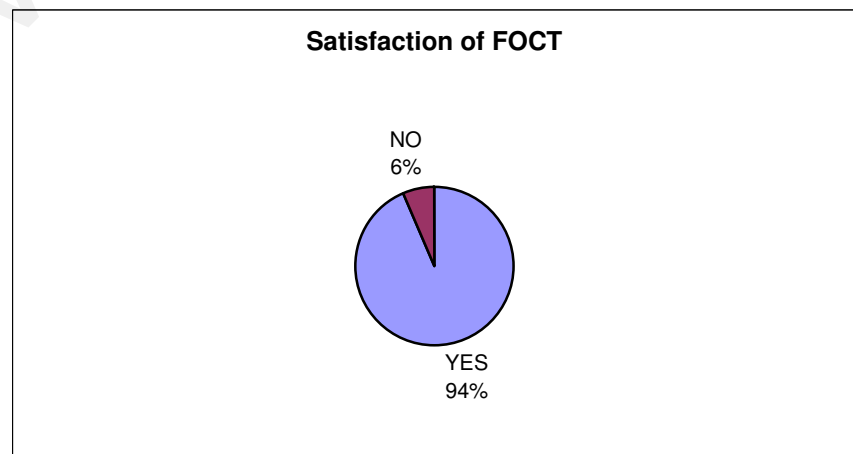
Figure 4.62: Satisfaction of FOCT

173

6) Helpfulness

The main purpose of this system is to help end users to know more about CASE Tools. Figure 4.63 shows that 90% of end users said that the system was helpful to them in learning CASE tools. Only, 10% of users said that the system was not helpful to them in learning CASE tools.
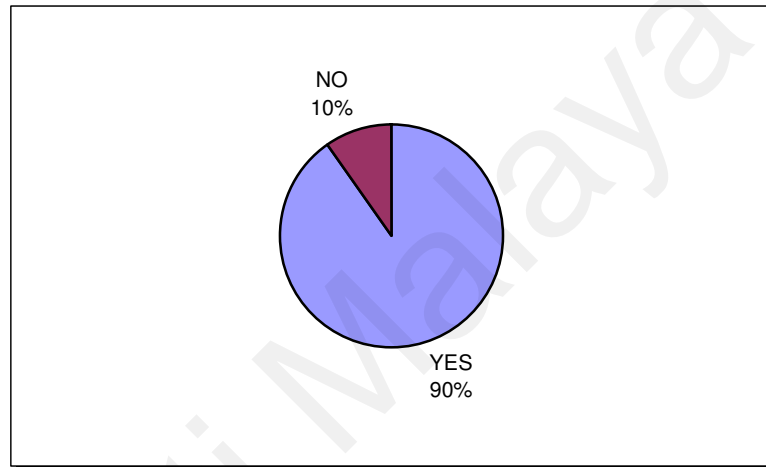
Figure 4.63: Helpfulness of FOCT

4.15 Limitations of the System

The researcher felt that FOCT had two main limitations. These are:

1. Lack of product demonstrations of CASE tools

   The researcher needs to add more product demonstrations of CASE tools in FOCT. In this way, users can experience how the CASE tools can be used in the real world.

2. Lack of flexibility in system modifications

   Design is always an issue for developers. James (2004) mentioned that "If you find that you have a lot of links to JSP pages from other pages, you many not

understand MVC very well". The researcher used too many JSP code in FOCT. Although using JSP is relatively easy to develop a system, it is quite hard to modify the system later.

4.16 User Manual

To ensure users to access the system correctly, the researcher developed a user manual for the five groups of people. Users can refer to the user manual when they access the system. In the user manual, the researcher described each step in detail as shown in Appendix C.

4.17 Conclusion and Future Enhancement

Through successfully developing FOCT system, the researcher concluded that:

1. FOCT system fully meets the objective that the researcher defined in chapter one.

2. FOCT is well developed with accessible functions for all kinds of users.

3. FOCT gives users a helpful and useful platform to promote the usage of CASE tools in the educational sector in Malaysia.

Although FOCT has been developed, there is a gap between the real system and the system the researcher expects. FOCT can be enhanced in the future as follows:

1. Add more features in other areas

Although the researcher felt that the features of FOCT are sufficient to support the user needs, one of the lecturers from end user testing, suggested that some new features should be added for better support. Examples are diagram drawing and assignment management.

2. Improve the system design by using Struts

   Struts is a new framework that helps developers create web applications with a robust design. The design of FOCT can be improved by using Struts.

3. Expand the research areas in CASE tools

   Usage of CASE tools should be from software development. The contents of FOCT should be enriched with other knowledge in software development.

4. Support more Internet Browsers

   So far, the researcher tested FOCT using Mozilla Firefox and Microsoft Internet Explorer. All features can be accessed successfully by using Mozilla Firefox. However some users informed the researcher that some of the features could not be accessed properly when using Microsoft Internet Explorer. Therefore, the researcher should consider this issue in the future.

In conclusion, the researcher has started the first step of creating FOCT as a platform for usage of CASE tools in Malaysia. The researcher really believes that it is a good start for further study on CASE tools and FOCT will bring much benefit to CASE tools users from both the educational and industrial sectors in Malaysia.

Chapter Five – Conclusion

5.1 Introduction

In this chapter, the researcher will give a conclusion based on the data found in chapter three and chapter four. Outcomes, limitation of the research and future work are further elaborated in this chapter.

The first aim of this research was to investigate what CASE tools are taught in educational institutions and compare that which is required by the industry in Malaysia. The second aim was to identify the factors and obstacles in implementing CASE tools in both these two sectors. The third aim was to develop a Forum of CASE Tools (FOCT) system, which enables CASE tool users to contribute their knowledge about the usage of CASE tools after having some experience using them.

Interviews were conducted to collect the necessary data from IT organizations. The researcher interviewed twenty-five IT organizations in Selangor and Kuala Lumpur. On the average, the researcher spent slightly over an hour in each session. During the interviews, the researcher asked about the background of these companies and about the usage of CASE tools in these companies. The data were analyzed and summarized in chapter three.

Questionnaires were distributed among both students and lecturers in colleges and universities. The researcher only managed to collect back fifty questionnaires from the students and fifty-three questionnaires from the lecturers. The output of this survey was analyzed by using SPSS and had been discussed in chapter three.

After completing all the necessary steps from literature review, interviews and surveys, the researcher developed FOCT to assist CASE tool users to promote the usage of CASE tools.

5.2 Outcomes of the Research

First Objective: <u>**To investigate what CASE tools are taught in colleges and universities and compare with that required by IT organizations**</u>

1) Utilization of CASE tools in educational and industrial sectors

CASE tools have been used for few years in Malaysia. There are many tools available in the market. The researcher found that Visio Professional and RationalRose were the most commonly used tools in the IT organizations interviewed. More than 40% of IT organizations chose Visio Professional and RationalRose as development tools. This is described in chapter three. Furthermore, the researcher discovered that the majority of big companies used RationalRose as the main development tool. These companies have used RationalRose for one or two years and used RationalRose in many steps in the software development processes. They confirmed that using RationalRose brought some benefits to system development such as fully supporting SDLC, better integration with other tools, promoting standardization, faster system development, and improved communication among team members.

The researcher conducted surveys through questionnaires in colleges and universities in Malaysia. From this survey, the researcher found that two main tools, namely RationalRose and SystemArchitect were frequently used in colleges and universities. The lecturers agreed that the two tools have powerful features. Most of the lecturers said that the vendor of RationalRose provided an educational version for students' use at lower costs. Some lecturers, however, preferred using SystemArchitect because SystemArchitect could support more methodologies and it is suitable for teaching purposes.

2) Comparison of student's knowledge and job requirements from the job market

To meet job requirements, students should have strong knowledge in using CASE tools. Interviewees said that students should have enough training in colleges and universities, know the concept of SDLC and OOT, and know how to use some basic features of CASE tools.

There are two factors that affect students' learning of CASE tools. The lecturer's knowledge of CASE tools is one of the factors affecting the learning effectiveness of students. Fifty-eight percent of the lecturers in this survey have teaching experience in OOT and 52% of the lecturers have taught Software Engineering. Forty percent of the lecturers have taught SAD. This survey shows that on the average, most lecturers have more than three, but less than five semesters of teaching experience in relevant subjects. The researcher believes that lecturers have enough teaching experience in relevant subjects.

The second factor affecting the learning of CASE tools is the time that students spent in learning them. In this study, most students who used the two main tools spent more than fifteen hours learning RationalRose and another twenty hours learning SystemArchitect.

Interviewees said that one or two weeks is enough to learn CASE tools in IT organizations.

The survey shows that most students have learnt and have knowledge about SLDC and OOT as described in chapter three. More than 88% of them used tools to draw diagrams and 52% of them used tools to analyze user requirements. Some of them used tools to generate documentation and codes. Basically, the students have the necessary fundamental skills to use CASE tools at the workplace.

From the results obtained above, the researcher concluded that students have sufficient support from lecturers and enough time to learn CASE tools. They also have a good understanding in SDLC and OOT. They have skills in using CASE tools. Based on job requirements given by interviewees in this in this study, the researcher concludes that the students' skills in using CASE tools matches job requirements in the job market.

Second Objective: <u>To identify factors and obstacles in implementing CASE tools in educational and industrial sectors</u>

The results of the interviews are described in detail in chapter 3. Primarily, the IT organizations and universities agreed that some factors are very important for CASE tools users to learn CASE tools. These factors are described as follows:

1) Necessary knowledge

Before learning CASE tools, users should have some basic knowledge. Most of interviewees felt that knowing SDLC was necessary. Only a few interviewees felt that developers should know methodologies. Seventy-seven percent of the lecturers interviewed agreed that in learning CASE tools, students should know SAD and OOT. On the other hand, from the survey, 88.6% of students knew OOT and 98% of students knew SDLC. The researcher concludes that the students had the basic knowledge to learn CASE tools.

2) Time

In this survey, more than 52% of students in colleges and universities agreed that there was not enough time to learn CASE tools. Most of the students spent more than three hours learning RationalRose and SystemArchitect. Most of the

180

interviewees in IT companies said that they would spend one or two weeks to learn CASE tools. However it depends on what the level of learning CASE tools they want to achieve. The researcher would like to conclude that the students should set longer time to learn CASE tools.

3) Learning approaches

Learning approaches will affect the effectiveness of learning CASE tools. More than 52% of students commented that there was no right learning approach to learn CASE tools. Some students said that the lecturers just simply gave assignments requiring the use of CASE tools. They further explained that the lecturers did not give much explanation about how to use the CASE tools and they did not really know what the results of the assignments were. The researcher discovered that this was a weakness in the teaching of CASE tools.

4) Necessity of using CASE tools in companies and universities

According to the results of this study described in chapter three, more than 68% of interviewees agreed that colleges and universities should provide some courses that use CASE tools. They explained that colleges and universities have the responsibility of giving students updated knowledge on the usage of RationalRose because these tools were used in IT organizations. Meanwhile, 92% of interviewees said that it is necessary to use CASE tools in software development. They said that CASE tools bring some benefits to them. They said that CASE tools should have been in Malaysia earlier. Furthermore, more than 81% of lecturers in this survey have awareness in using CASE tools and the lecturers have realized that using CASE tools is necessary in industry.

5) Sufficiency of facilities and guidance of lecturers

Facilities and guidance are very important for students to learn CASE tools. From this survey the researcher found that more than 56% of students said that the infrastructure in colleges and universities were not sufficient for them to learn CASE tools effectively. The students complained that there was not enough PCs in labs. Sixty-two percent of the students in this survey said that the lecturers did not give sufficient help or guidance in their studies. Most of the students explained that the lecturers did not have in-depth knowledge of CASE tools. Based on the survey data, 65% of lecturers felt that the time allocated for teaching CASE tools was enough. Fifty-nine percent of the lecturers in the survey commented that the facilities were not very sufficient to teach CASE tools.

In this study, the obstacles of using CASE tools were analyzed. Generally, the cost of CASE tools, extra cost for training and the complexity of CASE tools affected the usage of CASE tools negatively.

1) Cost of CASE tools

Most of the interviewees agreed that the cost of CASE tools was a big obstacle that limits the usage of CASE tools. Some lecturers in this survey pointed out that as educators, they felt that the cost of CASE tools is a barrier.

2) Extra cost for training

In this study, the researcher found that training cost was another obstacle to the use of CASE tools. Most of the interviewees in IT organizations felt the training cost of CASE tools was very high. Educators commented that they have to send the lecturers out for training. Therefore, most of the respondents agreed that the extra cost of training would create a barrier in using CASE tools.

3) Complexity of CASE tools

Most of the interviewees said that CASE tools were very difficult to use. Only a few interviewees said that CASE tools were easy to use. More than 44% of student surveyed said that they had to spend much time thinking on how to use some of the features.

Third Objective: <u>To provide a collaborative CASE tools system that can help to promote the usage of CASE tools.</u>

FOCT was completed to help CASE tool users. It allows the users to share their knowledge in using CASE tools and provides relevant information about CASE tools. The system development of FOCT was explored in chapter four. Based on the evaluation of the system, the majority of users are satisfied the system, and commented that the system gave them much help in learning and using CASE tools.

5.3 Limitations of the Research

Although many areas are covered in this research, the detailed or visible benefits of using CASE tools in IT organizations were not investigated. Furthermore, this research has limitation between students' skills and requirements from industry. The limitations of FOCT are described in chapter four.

5.4 Future Work of the Research

After completing this research, the researcher suggests that two tasks should be done in the future. One of the tasks is to define the correct amount of time to allocate for learning CASE tools to students with different educational levels. Through this study, the

researcher felt that there was no standard time period for learning CASE tools in colleges and universities. The time period should be formulated based on feedback from IT professionals. Another task is to enhance the features of FOCT. This enhancement was described in chapter four.

5.5 Conclusion

Through this study, the researcher not only has an in-depth understanding of utilization of CASE tools in the educational and IT sectors in Malaysia, but also has memorable experiences doing this research in Malaysia.

This research explored the realization of using CASE tools in the real world and would give CASE tools users a clear and right track of using CASE tools in industry. The study revealed to the educators that there is inadequate support from lecturers and facilities are lacking in colleges and universities.

By doing this research, an in-depth understanding of the investigation of CASE tools' utilization in educational and industrial sectors in Malaysia, factors, obstacles of using CASE tools can be achieved, and FOCT is developed to fulfill the objectives of this research.

References

Alan M. Christie, Alan W Brown, Susan Dart, 1993, A Case Study in Software Maintenance. [online]. Available from: http://www.sei.cmu.edu/legacy/scm/abstracts/abssw_maint_case_study_TR08_93.html. [cited on 10 Nov 2005] .

Alan W.Brown, David J.Carney, 2002, Principles of CASE Tool Integration. [online]. Available from:http://interactive.sei.cmu.edu/publications/bboks/otherbooks/princ.of. case.tools.int.html. [cited on 8 Nov 2005].

Alan S. Fisher (1991), *CASE Using Software Development Tools,* 2nd Edition, USA, John Wiley & Sons,Inc.

Alfred L. McKinney (1991), *A Report on Computer-Aided Software Engineering (CASE),* USA, Consortium for Computing in Small Colleges (p.85).

Asocio, ICT Industry in Malaysia. [online]. Available from:http://www.asocio.org/ resources/profiles/malaysia.pdf. [cited on 8 12 2006].

Augustin Aegidius Sitiol, Sai Peck Lee, An Approach to the development of hybrid Object-oriented design Tool, *Malaysian Journal of Computer Science*, **Vol12**, No1, June1999, p. 47-54.

Barbee T. Mynatt, Laura Marie Leventhal (1990), An Evaluation of a CASE-Based Approach to Teaching Undergraduate Software Engineering, *ACM*, **Vol2**, p.122-p.125

Barbara Howell, 2004, Making a CASE for Using the Students Choice of Software or Systems Development Tools. [online]. Available from: http://proceedings.informing-science.org/InSITE2004/034howel.pdf. [cited on 11 March 2006].

Barbara Howell, 2005, Effective Organizations need Effective IS Graduates. [online]. Available from**:** http://uxisweb1.brunel.ac.uk/iseingsites/EMCIS/EMCIS2005/pdfs/-44.pdf. [cited on 8 January 2006].

Carma McCure (1989), *CASE is Software Automation*, USA, Prentice-Hall International, Inc (p.5).

CMU, What is a CASE Environment? [online] (Updated: 9 August 2004). Available from: http://www.sei.cmu.edu/legacy/case/case_whatis.html. [cited on 12 September 2005].

Charies F. Martin (1988), Second-Generation CASE tools: A Challenge to Vendors*, IEEE Software*, **March**, p.46.

Clifferd C. huff (1992), Elements of a Realistic CASE Tool Adoption Budget*, Communication of The ACM*, April **vol35**, No.4, p.45-p.54.

Danny C. C. Poo, Mui Ken Chung (1998), CASE and Software Maintenance Practices in Singapore, *The Journal of System and Software*, Vo**l449**, p.99.

David, Linthicum (1997), *Client/Server and Intranet Development*, USA, John Wiley & Sons.

Damon Hougland, Aaron Tavistock (2001), Core JSP, USA, Prentice Hall.

Donald J. Bagert, Thomas B. Hilburn, Greg Hislop, Michael Lutz, Michael McCracken, Susan Mengel, 1999, Guidelines for Software Engineering Education Version 1.0, Vol3, p.343-352. [online]. Available from: http://www.sei.cmu.edu/pub-lications. [cited 9 November 2005].

Duska Rosenberg, Chris Hutchison (1994*), Design Issues in CSCW, Computer Supported* C*ooperative Work*, UK, Springer-Verlag London Limited (p.269).

Elliot J. Chikofsky, Burt L. Rubenstein (1998), CASE: Reliability Engineering for Information Systems, IEEE computer society Press technology series, Software development, Computer-Aided Software Engineering (CASE), *IEEE* March**,** p.11-16.

Faqs, [online]. Available from: http://www.faqs.org/faqs/software-eng/part2/section-5.html. [cited on 12 November 2005].

Grady Booch, James Rumbaugh, Ivar Jacobson (2001), *The Unified Modeling Language User Guide*, USA, Addison-Wesley (p.15).

Grodon C.Everest (1992), Accessing User Experence With CASE Tools: An exploratory Analysis System Sciences*, IEEE,* January.

Ian Sommerville (2001), *Software Engineering* 6th Edition, USA, Addison-Wesley Inc (p.545).

Jams Goodwill, Richard Highttower (2004), *Professional Jakarta Struts*, USA, Wrox.

James Kiper , Michael J. Lutz , Henry A. Etlinger (1992), Undergraduate Software Engineering Laboratories: A Progress Report from Two Universities, *ACM SIGCSE Bulletin,* Vol.24 No.1, p.57-62.

James Goodwill (2002), *Apache Jakarta-Tomcat*, USA, Apress.

Jari Maansaari, Juhani Iivari (1999), *Information & Management* Vo**l36**, p.37-53.

Jeffery J. Joyce (1993), *Integration of CASE into Undergraduate Education*, p.136, September 4.

Joan Peckham, Scott J. Lioyd (2003), *Practicing Software Engineering in the 21st Century*, USA, Idea Group.

Judy Bamberger, Richard Fairley, Charles Mekay (1986), *Are the universities meeting the needs of industry? Is industry holding up its end of the deal,* ACM, p.381-384

Karl Avedal, Danny Ayers (2000), *Professional JSP*, USA, Wrox.

Mali Senapathi (2005), A Framework for the Evaluation of CASE Tool Learnability in Educational Environments, *Journal of Information Technology Education,* Vol4.

Mark Maslakowski (2000), *Teach Self MySQL in 21 Days*, USA, SAMS.

Mary J. Granger, Roger Alan (1991), Proceedings of the Twenty-Second SIGCSE Technical Symposium on Computer Science Education*, ACM SIGCSE Bulletin, Vol23 Issue 1.*

Mohd. Shazali OTHMAN, 1989, Status of Standardization on Information Technology in Malaysia. [online]. Available from: http://www.cicc.or.jp/english/hyoujyunka/af02/-2-09.html. [cited on 10 January 2006].

Paul F. Zarrela (1999), *CASE tools integration and standardization*, USA, Software Engineering Institute, Carnegie Mellon University.

Paul F. Zarrella, Dennis B. Smith, Edwin J. Morris, 1991, Issues in Tool Acquisition. [online].Available from: ttp://www.sei.cmu.edu/pub/documents/91.reports/pdf/tr08.9-1.pdf. [cited on 9 October 2005].

Pierre. Alain Muller (1997*), Instant UML*, USA, Wrox Press, LTD (p.15).

Premkumar. G, Micheal. Potter (1995), Adoption Computer Aided Software Engineering (CASE) Technology: An Innovation Adoption Perspective, *DATABASE Advances,* Vol.26, Nos.2,3, May/August.

Richard H. Thayer, Merlin Dorfman (1998), *Software Requirement Engineering*, USA, IEEE Computer Society Press (p.515).

Richard H. Thayer, Merlin Dorfamn (1997), *Software Requirements Engineering* 2nd Edition, USA, IEEE Computer Society Press (p.399).

Rob J. Kusters and Gerard M. Wijers (1993), On the Practical use of CASE-tools Results of a Survey*, IEEE,* July, p2-10*.*

Ruhana Ku, Mahamud, Nafishah Othman, Norita Md. Norwawi (1997), Current Practices in Information System Development Tools and Techniques, *Malaysian Journal of Computer Science,* Vol. **10,** p.27-35

Sadahiro Isoda, Shuichiro Yamamoto, Hiroaki Kuroki, Atsuko Oka (1993), *Nippon Telegraph and Telephone Corporation NTT Shinagawa TWINS*, Vol6, p.48 – 59.

Selamat, M H, Choog C Y, Othman A T (1994), Non-use phenomenon of CASE tools: Malaysian experience*, Information and Software Technology*, Vol36, p.531-537.

Selamat. M. H, M M Rahim (1996), Evaluation of Computer-Aided Software Engineering tools: Experience from Malaysian Organizations*, International Journal of Information Management,* Vol.16, No.4, p.299-313.

Shelly, Cashman, Rosenblatt (2003), *System Analysis and Design,* 5th edition*,* USA, Thomson Course Technology.

Simon Bennett, Steve McRobb, Ray Farmer (2002), *Object-Oriented System Analysis and Design Using UML*, 2nd Edition, USA, McGraw-Hill (p.57, p.61, p.556).

Stan Jarzabek and Riri Huang (1998), The Case for User-Centered CASE Tools*, Communication of the ACM,* Vol. 41, No. 8

Stephen G. MacDonell, 1993, Software Development, CASE Tools and 4GLs-A Survey of New Zealand Usage. [online]. Available from: http://divcom.otago.ac.nz/infosci/publictns/complete/papers/dp9304sm.pdf.gz. [cited on 11, March, 2006].

Reference of Sample Websites

Analysttool. [internet]. Available from: http://www.analysttool.com/.[cited on 11 November 2005].

Borland. [internet].Available from: http://www.borland.com/us/. [cited on 11 November 2005].

Construx Software Builders, Inc.[internet]. Available from: http://www.construx.com/professionaldev/industry/. [cited on 11 November 2005].

Cser.[internet].Available from: http://www.cser.ca/index.php. [cited on 11 November 2005].

Case. [internet]. Available from: http://www.umsl.edu/~sauter/analysis/analysis_links.html#case. [cited on 20 September 2005].

Casewise.[internet]. Available from: www.casewise.com. [cited on 11 November 2005].

Dacs.[internet]. Available from: www.thedacs.com. [cited on 10 November 2005].

Embry Riddle Aeronautical University. [internet].Available from: http://faculty.erau.edu/hilburn/se-educ/. [cited on 8 September 2005].

Ethics.[internet].Available from: http://www.cs.utexas.edu/users/ethics/index.html. [cited on 14 September 2005].

Fase.[internet].Available from: http://www.cs.ttu.edu/fase/. [cited on 4 November 2005].

ISPRAS. [internet]. Available from:http://www.ispras.ru/index.html. [cited on 2 November 2005].

Itmweb. [internet]. Available from: http://www.itmweb.com/index.html. [cited on 3 November 2005].

Kenneth L. Modesitt, Don Bagert, Laurie Werth. Overview of Survey of International Software Engineering Academic Programs. [internet]. Available from:http://www.en-gin.umd.umich.edu/CIS/Survey_ISEAP_2001/overview/Over-view.html, [cited on 20 September 2005].

peterindia.[internet].Available from : http://www.peterindia.net/index.html. [cited on 13 September 2005].

The CASE Tools. [internet]. Available from: http://kmh.ync.ac.kr/comScience/oop/mode-ling/case2/tools.html#A. [cited on 12 August 2005]

tellogic.[internet]. Available from: http://www.telelogic.com/corp/products/system-architect. [cited on 12 August 2005]

Rit.[internet]. Available from: http://www.se.rit.edu/forms.php. [cited on 10 November 2005].

Rspa.[internet].Available from: http://www.rspa.com/main.html.[cited on 13 November 2005].

Stgcase. [internet]. Available from: http://www.stgcase.com/index.html. [cited on 20 September 2005].

Software Engineering. [internet]. Available from: http://griffin.se.rit.edu/sections/. [cited on 11 October 2005].

Software Eng.[internet]. Available from: http://www.faqs.org/faqs/software-eng/. [cited on 1 November 2005].

SEI Areas of Work: Engineering. [internet]. Available from: http://www.sei.cmu.edu-/engineering/engineering.html. [cited on 1 November 2005].

Software engineering. [internet].Available from: http://www.netnam.vn/unescocourse /se/soft_frm.htm. [cited on 4 November 2005].

Threesl.[internet].Available from: www.threesl.com. [cited on 11 November 2005].
The Software Engineering Lab. [internet]. Available from: http://www.dse.doc.ic.ac.-uk/sel/index.html. [cited on 25 October 2005].

The DoD Software Tech News.[internet].Available from: http://www.softwaretec-hnews.com/listing.shtml. [cited on 2 October 2005].

Toolprice.[internet].   Available   from:http://www.visual-paradigm.com/product/vp-uml/vpumlpricelist.jsp. [cited on 13 November 2005]

Vendor. [internet]. Available from: http://www.qucis.queensu.ca/SoftwareEnginee-ring/vendor.html. [cited on 5 November 2005].

Visual-paradigm. [internet]. Available from: http://www.visual-paradigm.com/. [cited on 23 October 2005].

Visible.[internet].Available from: http://www.visible.com/index.htm. [cited on 11 November 2005].