MULTIMEDIA-ENHANCED EXPERT SYSTEM FOR C++ PROGRAMMING

By

SITI MURNI MUSTAFA

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

Universiti Teknologi Petronas
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

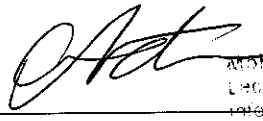
MULTIMEDIA-ENHANCED EXPERT SYSTEM FOR C++ PROGRAMMING

by

Siti Murni Mustafa

A project dissertation submitted to the
Information Technology Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Technology (Hons)
(Information Technology)

Approved:



Mr. Nordin Zakaria
Project Supervisor

Mohamed Nordin Zakaria
Lecturer
Information Technology/Information Systems
Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan, MALAYSIA.

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

June 2005

CERTIFICATION OF ORIGINALITY

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

SITI MURNI MUSTAFA

ABSTRACT

All final year students in UTP are required to undertake a final year project (FYP) paper, which is a design and/or research-based subject. It requires student to do research; design and/or development work in each discipline, especially on real-world problems which would motivate student to produce practical solutions. For this project, specific title has been proposed and approved which is "A Multimedia-Enhanced Expert System for C++ Programming (Cplus2ES)". It is a research and development work project. The objective of the project is to make sure student do a research in the area that relevant with their approved title. Beside, student also needs to design a system using the information from the research in order to translate them into the real working system. The system must be efficient, convenience and user friendly compare to the other system available in the market.

ACKNOWLEDGEMENTS

First and foremost, thank God for this opportunity and seeingme through some truly difficult times especially through the journey of completing this project. I would like to express my warmest gratitude and appreciation to all parties who have contributed towards the success of this Final Year Course Project.

I would like to express my appreciation especially to the following people:

- Mr. Nordin Zakaria
  Supervisor
- Mr. Mohammed Noor Ibrahim
  IT/IS Final Year Project Coordinator
- Mrs. Vivian Yong Suet Peng
  IT/IS Final Year Project Coordinator
- Mr. Shuib Basri
  IT/IS Final Year Project Coordinator

Also my utmost gratitude to my father Mustafa Che Harun and mother Minah Mat Saman which has brought me up and help in many ways unimaginable. Only Allah may repay them.

This special thanks and appreciation also dedicated to all UTPian and colleagues for their continued support, guidance and contribution to the success of this Final Year Project. With the full cooperation from the various people above, I have successfully achieved the objective of this project. Thank You.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

UTP	– Universiti
AI	– Artificial Intelligent
Cplus2ES	– C++ Expert System
CAI	– Computer Aided Instructions
CBT	– Computer Based Training
IDDS	– Intelligent Decision – Support System
CORVID	– Exsys CORVID
ESDLC	– Expert System Development Life Cycle
VB	– Visual Basic

CHAPTER 1

INTRODUCTION

1.1 Background of Study

Artificial Intelligence, or AI for short, is a combination of computer science, physiology, and philosophy. AI is a broad topic, consisting of different fields, from machine vision to expert systems. The element that the fields of AI have in common is the creation of machines that can "think".

In order to classify machines as "thinking", it is necessary to define intelligence. To what degree does intelligence consist of, for example, solving complex problems, or making generalizations and relationships? And what about perception and comprehension? Researches into the areas of learning, of language, and of sensory perception have aided scientists in building intelligent machines. One of the most challenging approaches facing experts is building systems that mimic the behavior of the human brain, made up of billions of neurons, and arguably the most complex matter in the universe. Perhaps the best way to gauge the intelligence of a machine is British computer scientist Alan Turing's test. He stated that a computer would deserve to be called intelligent if it could deceive a human into believing that it was human.

Artificial Intelligence has come a long way from its early roots, driven by dedicated researchers. The beginnings of AI reach back before electronics, to philosophers and mathematicians such as Boole and others theorizing on principles that were used as the foundation of AI Logic. AI really began to intrigue researchers with the invention of the computer in 1943. The technology was finally available, or so it seemed, to simulate intelligent behavior. Over the next four decades, despite many stumbling blocks, AI has

grown from a dozen researchers, to thousands of engineers and specialists; and from programs capable of playing checkers, to systems designed to diagnose disease.

AI has always been on the pioneering end of computer science. Advanced-level computer languages, as well as computer interfaces and word-processors owe their existence to the research into artificial intelligence. The theory and insights brought about by AI research will set the trend in the future of computing. The products available today are only bits and pieces of what are soon to follow, but they are a movement towards the future of artificial intelligence. The advancements in the quest for artificial intelligence have, and will continue to affect our jobs, our education, and our lives.

Expert Systems are the most mature and widely used commercial application coming out of AI. In programming domain, expert systems are capable of integrating the perspectives of individual disciplines (e.g. system analysis, knowledge acquisition and system design) into a framework that best addresses the type of ad hoc decision-making required of programmer. Expert systems can be one of the most useful tools for accomplishing the task of providing growers with the day-to-day, integrated decision support needed to grow their knowledge

The expert system discussed in this project, namely the *"Multimedia-Enhanced Expert System for C++ Programming (Cplus2ES)"*, is aimed at capturing expertise knowledge in the area of C++ language, making it available to decision makers who need answers quickly at the right place at the right time.

1.2 Problem Statement

1.2.1 C++'s Expert vs. C++'s Demand

According to the *Pluralsight LLC(a group of experienced system developer)*, there are encouraging trends of using C++ programming whereby:

- Universities are switching back to C++ from Java as their primary undergraduate language.
- C++ book sales are up in a difficult market
- By most measures, C++ use in industry never declined and even though it's long been the dominant language it is *still* slowly growing (Java has peaked and seems to have started slowly declining by many measures).
- And C++-specific conferences are returning for example in 2001 and 2002 Pluralsight LLC held three successful *The C++ Seminar (TCS)* events, and in 2004 the *Stroustrup & Sutter on C++ (S&S)* fest sold out at double the size of the TCSes.

In another study taken from *Association of Cand C++ Users (ACCU)* - January 2001 (Vol. 2 No. 1), a survey was drawn from 129 000 IT related job offers. The six most often skill required by the employer are C++ (19951 offers), Java (16920 offers), SQL (9121 offers), VB (9003 offers), HTML (8788 offers), and Java Script (4793 offers). From this survey shown that C++ skill still sit at the first requirement in the job offers. However, there is no enough expertise (C++) to cattle this demand. Nowadays, students as well as programmer (expert/novice) prefer to study and jump to the other programming languages which is offers more GUI features compared to C++. This trend leads to the decreasing of C++ expertise in the market.

According to the H. M. Deitel (author of C++ How to Program), the main reason why less students or programmer interested to use C++ because it is a difficult language that is normally taught only to experienced programmers. In order for students or novice programmer to learn this language, it required an expert to guide and teach them.

Therefore, this expert system (An-Enhanced Multimedia Expert System for C++ Programming) serves as an efficient and effective aid in this narrow problem area. There are conditions where the expert is not accessible and C++ expert system provides an electronic gateway to address problems normally thought to require a human expertise. By developing the expert system for C++ learners, it can help to increase the number of C++ expertise as well as to full fill the demand in the market.

Beside that, A Cplus2ES is designed to improve the old version of C++ compiler existed in the market. Nowadays, C++ compiler that was used by student or programmer still cannot provide them with a result as good as they expected. The compiler just inform user that their code having mistakes but it do not tell the user what the mistakes are? So, it is hard for user to discover their problem.

In order to solve this problem, an expert system for C++ is designed for answering the question stated above. When user compiles the code, the compiler will provide user with the report. If the report record that the program having error and user do not know what exactly the error is, the system will inform the user. For example, they forgetting to include the ***iostream*** file in a program that inputs data from the keyboard or outputs data to the screen after they answering a series of question provided by the system.

1.2.2 Significance of the Expert System in C++ Programming

C++ is a difficult language that is normally taught only to experience programmers, so this project is unique and significance to do because:

- It is design appropriate for technically oriented people with little or no programming experience
- It is design appropriate for experienced programmers who want a deeper treatment of the language

How can this project appeal to both groups? The answer is that the common core of the project emphasizes achieving program clarity through the proven series of question ask

to the users in order to find out their main problems. This system try to help user by attempting provide information in a clear and straightforward manner.

1.3 Objectives and Scope of Study

1.3.1 Objectives

The objectives of this project to be achieved are as follows:

i) To understand the underlying concepts of AI in the Learning Field

ii) To do a research on learning and teaching expert system, its engineering and development

iii) To examine the rule based expert system approach in constructing a computer-based C++ expert system

iv) To develop a C++ expert system, focusing on the diagnosis of the common problem errors done by user in C++ programming. This system will capture a C++ expert's knowledge and make it available to other decision makers like novice programmers and students, who need answers quickly in order to create a good program. By answering a few questions presented by the system, a non-expert will be able to obtain the results of the possible error they had done.

1.3.2 Scope of the Study

This study will focus on development of providing knowledge and information expert system to diagnose common programming errors done by the user, as how it would be undertaken by a C++ expert.

This project will focus on developing an expert system that poses questions on observable errors always done, to user, and based on the answers provided, will make a diagnosis on the possible C++ errors contracted by the user. Emphasis will be given to knowledge acquisition to develop the knowledge base and inference methods to deduce the diagnosis.

1.3.3 Feasibility of the Project Within Time and Scope

This project can be deemed as technically feasible to complete within the given time and scope. There is no relative cost related to the projects as the expert system can be developed using software that is already available in the lab. There are also adequate resources available to support the project such as books and online resources.

The time frame given to complete the project is also sufficient. The Gantt chart (Gantt chart is shown in Appendix II) produced indicates the time allocated for each task and its serves as guidance for project execution. Please refer at the appendices for the Gantt chart.

CHAPTER 2

LITERATURE REVIEW AND THEORY

2.1 Application of Artificial Intelligent (Expert System) in Education

2.1.1 Computer Aided Instruction (CAI)

Individually paced instruction and frame-based, computer-aided instruction comprised early attempts to provide adaptive instruction and, although successful for some types of learning, fell short because their learning environments had low fidelity and their ability to adapt was limited to branching between static screens (Murray 1998).

In the sixties the first attempts to use computers in educations were based on rather behaviorist theories with emphasis on feedback and reinforcement actions (Gazzaniga & Scarafiotti 1997).

- The teaching path was fixed and linear.
- The communication style was mono-directional (from the computer to the student) and imperative.
- Individuality was restricted to the amount of time spent in the learning process.
- The CAI programs proved useful above all for training.
- Most severe criticism: the rigidness based on the action/reaction principle.

CAI refers to computer programs that provide drill and practice exercises while CMI refers to programs that evaluate and diagnose students' needs, guide them though the next step in their learning, and record their progress. Both CAI and CMI can be used with little teacher intervention. CEI, on the other hand, requires the teacher to be involved in planning and helping to carry out learning activities.

2.1.2 Computer-Based Training (CBT)

CBTs are "conventional" systems that support learning and training which do not adapt to individual users. A CBT is more "traditional" than an ITS: "despite providing ready access to learning materials and on-on-one interaction, lacks the flexibility and learner-centered orientation of ITS that adds a dynamic and adaptive dimension to self-paced instruction" (Bell & Redfield 98). See ITS.

2.2 Intelligent Decision-Support System (IDDS)

The evolution and maturity of decision support systems in their various forms highlights how the focus of information systems has moved from transaction oriented operational systems to managerial systems that support decision making and knowledge processing. Information systems need to be viewed as tools used by organizations to improve their efficiency and effectiveness. It is also the organization which provides the context from which the success of this system can be evaluated. The focus of support systems needs to be their ability to achieve organizational outcomes.

Concurrently, organizations have also been undergoing significant changes. The globalization of markets and enterprises, the rapidly changing environment in which organizations operate, and the shift of focus from physical production to knowledge work are some of the elements that have led organizations to adopt different structures and consequently change their work environments and practices. The modern organization can be characterized as having a flat structure, being able to react flexibly and rapidly to changing circumstances and as having a workforce able to be effective in such an environment (Huber, 1990, Scott Morton, 1991).

Our research is aimed at developing a framework for the construction of systems that play an active role in supporting both knowledge processing and task performance. We view such systems as targeted intelligent decision support without relying, necessarily, on Artificial Intelligence techniques and technology. We are conducting an exploratory research study to define the framework through its practical application.

The approach adopted in our work differs from traditional approaches in decision support systems (DSS) in that it is not focusing merely on managerial decision making but, we believe, attempts to reflect organizational realities. In adopting an organizational perspective, we see knowledge processing as an integral part of work practices in a modern organization and not the exclusive prerogative of managerial work. This position is consistent with the concept of post-Fordist work discussed in industrial relations circles (Amin, 1994). This position, however, entails that workers are engaged in both knowledge processing and task performance, in contrast to traditional managerial work that focuses only on the former. As a consequence, the proposed framework intends to integrate and support both task performance and knowledge processing.

The framework proposed in this study is two layered. It comprises a *Pragmatic* layer in which the task is modeled, and a *Conceptual* layer which documents meta models of the process and structure of the task. The two layers are presented as recursive rather than as a hierarchical structure in that the *Conceptual* layer represents the ontology and epistemology of the task. This is consistent with the position adopted in CommonKADS approach to knowledge based systems development.

The significance of our research lies in the proposition that application of the framework provides the mechanism for organizational learning. The models, and their annotations, in both the *Pragmatic* and *Conceptual* layer, represent a task based organizational memory. These, together with the facilities that are needed to support knowledge processing at the *Conceptual* layer, provide the necessary components for organizational learning. In addition, the framework supports knowledge reuse by enabling workers to define tasks in terms of existing meta models and their ancillary data. Both learning and reuse contribute to the framework's role as a dynamic organizational memory (Burstein, Linger, Zaslavsky and Crofts, 1997).

The exploratory research study is being conducted in collaboration with the Epidemiological Research Unit of Macfarlane Burnett Centre for Medical Research. The major work of the Unit is to conduct survey based research on public health issues

relating to blood borne disease. The choice of a scientific research organization was deliberate. Work performed by scientists in research organization is, by any definition, post-Fordist thus alleviating the need to demonstrate the existence of such work practices. Another important consideration was that review and exploration processes are embedded in the work practices of research organizations, as well as within the scientific community as a whole. The proposed framework enables knowledge to be shared and communicated within the community. Knowledge within the framework is organized, structured and documented in a consistent manner facilitating such communication. Our involvement with the Unit arose from the perceived need, on the part of our collaborators, for more sophisticated support to design, administer and analyze surveys. The IDS approach adopted in this work was facilitated by the Unit's willingness to accept innovative approaches to the provision of such support.

The significance of the proposed framework is that it provides the basis for implementing intelligent decision support systems that not only address individual needs, as in traditional DSS, but also have significant organizational outcomes. These outcomes stem from the dynamic task based organizational memory that the framework represents, and the organizational learning that is an integral part of knowledge processing facilities within the *Conceptual* layer of the framework. Moreover, knowledge reuse is achieved when tasks can be defined by instantiating the meta models in the *Conceptual* layer. We believe that the proposed framework is not limited to the case study task but is generic to any task in a context where workers are empowered to reflect, explore and learn

2.3 Exsys Corvid & Knowledge Automation Expert System

Exsys CORVID represents a proven knowledge automation expert system paradigm resulting from a close examination of what is needed to build and implement decision-making systems in today's Internet-oriented world. CORVID is designed to allow system developers to easily capture knowledge, build powerful interactive advisory applications and deliver them online (Internet, Intranet, wireless) - quickly and effectively producing significant return on investment.

An expert makes a decision by considering many things. Through experience and "know-how" an expert learns which factors are potentially relevant, the meaning of certain elements or circumstances (who, what, where, why, when), and how they should be combined and weighted to reach a recommendation. Knowledge Automation Expert Systems are programs that emulate this decision-making process. They allow the individual steps in solving problems to be described in rules. An Inference Engine then uses the rules to automatically determine what information is needed, the implications of various facts, and arrives at a logically reasoned conclusion. ***Exsys CORVID's*** knowledge Automation is a unique technology of directly delivering expert knowledge via expert systems that can be accessed via a Web Browser. It provides a way to interact with a site visitor in a way that emulates the conversation they would have with a human expert to answer their questions. This is the only practical way to directly deliver expert decision-making knowledge to Web site visitors. Knowledge Automation is conceptually very different from search engines or "case-based" systems that use key words to guess at what information might be relevant to a user. A Knowledge Automation Expert System has logical rules that enable it to systematically and accurately determine the best recommendation. The user is provided with specific answers tailored to their situation, with all relevant factors considered.

CHAPTER 3
METHODOLOGY / PROJECT WORK

3.1 Procedure Identification

For this project, the development methodology used is the expert system development life cycle (ESDLC) that incorporates prototyping during its system development. The process of building an expert system is an integrated six-phase process:

1. Phase 1: Problem assessment
2. Phase 2: Knowledge acquisition
3. Phase 3: System design
4. Phase 4: Testing
5. Phase 5: Documentation
6. Phase 6: Maintenance

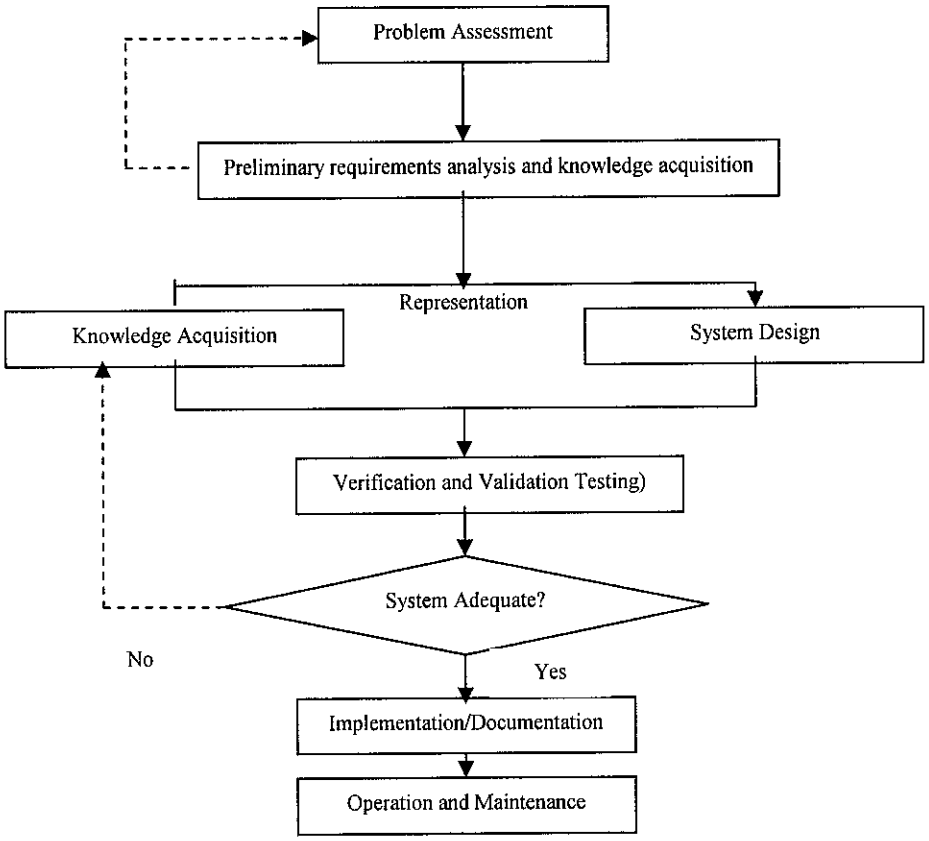


Figure 3.1: Expert System Development Life Cycle

3.1.1 Problem assessment

The first step in building an expert system is the identification and the assessment of a unique problem potentially solvable with an expert system. A review of the problem domains that may be candidates for expert system development is necessary.

This process is structured according to the following five tasks; 1) Identify candidate problems, 2) Perform feasibility study, 4) Perform cost/benefit analysis, 5) Select the best project and 6) Write the project proposal.

Some of the candidate problems identified is:

- Shortage of C++ expert
- Inaccessibility of programmers
- Increasing number of people that having interest to learn C++
- Increasing demand from user to have a program that can assist them at the right place at the right time

3.1.2: Knowledge Acquisition

Knowledge acquisition is defined as [Buchanan et al., 83]:

"The transfer and transformation of potential problem-solving expertise from some knowledge source to a program".

Essentially knowledge acquisition is the process of acquiring knowledge from a human expert (or group of experts). Knowledge acquisition involves elicitation of data from the expert, interpretation of the data to deduce the underlying knowledge and the creation of a model of the expert's domain knowledge in terms of the most appropriate knowledge representation mechanism. For this, knowledge engineers must familiarize themselves with the domain of the expert and represent the knowledge in a form that can be computerized.

Although there have been several moves to use software tools for knowledge elicitation, a UK survey [Smith et al., 94] revealed that 77% of KBS had used some form of unstructured interview to obtain information. Most of these started out with informal discussions to explain the project and to gather preliminary information, followed by more formal structured interviews.

Below are a number of used knowledge elicitation methods for this project:

• **Printed / Electronic Sources** - The simplest form of knowledge acquisition is from printed sources. This includes searching through documents, books and other items of printed material to find the knowledge necessary to build a knowledge base.

• **Interviews** - Informal interviews consist of the knowledge engineer asking spontaneous questions, where there has been little or no planning prior to the interview.

• **Questionnaires** – Expert engineer designed a questionnaire and distributed it to some selected participants in order to capture participant's knowledge and experience

• **Observational Studies** – Expert engineer observed the condition in UTP. What happen around here and what consequences she could see as the impact from the situation happened.

For the purpose of this study, the particular method of knowledge acquisition employed is not a primary concern. The chosen method should elicit a sufficient quantity of useful data which can in turn be represented by Exsys CORVID in the proposed system, thereby demonstrating the usefulness of the Cplus2ES system. Further consideration should be given to the appropriateness of any given method after the demonstration of a prototype system. For this reason, a combined choice of printed / electronic sources and interviews will be used most for the purposes of this study. Opinions about the use of interviews vary widely. [Kawaguchi et al., 91] consider they are *"essential in eliciting new knowledge from domain experts*", while [Cooke and McDonald, 86] refer to them as *"a less than optimal knowledge acquisition technique*". However the interview process remains the most frequently used method for obtaining domain knowledge from human experts. This, coupled with printed / electronic sources, should provide an adequate mechanism to elicit knowledge for the proposed system.

Beside those two methods, others method which is questionnaire (Sample of questionnaire is shown in Appendix I) still be used in order to gather as much as information. First, expert engineer conduct a small scale of survey to a number of 30 UTP students who are taking C++ course. The set of questionnaire will be the combination of Likert scale. Participants are given the set of questionnaire and they have to respond to the questions.

Knowledge acquisition is a cylindrical process. It has three main tasks:

1. *Collection*: The task of acquiring knowledge from the expert, book, journal, internet and user
2. *Interpretation*, of the collected information involves the identification of key pieces of knowledge, such as concepts, rules, strategies, etc.
3. *Analysis,* involve the study of the key pieces of knowledge uncovered during the interpretation tack.

3.1.3 System Design

The design process for this project is structured according to the following four tasks:

3.1.3.1 Select Control Technique

Decision-making on control will center on the choice of the inference technique and the goal agenda. It is important for expert engineer to choose an appropriate control technique in order to make sure this project will complete in the given scope and time.

3.1.3.2 Select Expert System Development Software

During this effort, attempts are made to match the features of the problem with the capabilities of available software. For the purpose of this project, Exsys CORVID shell has been chosen as a software tool. CORVID is based on the VB model. The reason CORVID is chosen are:

- CORVID offers a revolutionary way to deploy interactive expert system on Web sites. This brings our top-levels expert system to the fore-front to help site visitors make quick, correct and consistent decisions. CORVID systems run in a window from within an existing Web page so the site visitors is not linked away or distracted.
- CORVID applications are delivered by a small (~100k) applet that allows robust interface design options. Expert engineer just select how a question is to be asked and what text or graphics to associate with it. Questions can be asked through a variety of controls, styles, and can include JPG or GIF images. These features shown to us that CORVID provide expert engineer a convenience way to present the information to the user through the expert system.
- Since the system that designed using CORVID is running as an applet, expert engineer can use HTML for the rest of the page surrounding the applet. This features enable expert engineer to customize the interface of the system

3.1.3.3 Develop the Prototype

A prototype is a model of the final system. Its basic structure, in terms of the way it represents and processes the problem's knowledge, is the same as expected in the final system. This prototype is designed based on appropriate system architecture and using selected development software which is *EXSYS CORVID*.

3.1.3.3.1 Knowledge Representation and Design

Before starting design the system prototype, expert engineer needs to organize and represent the expert's knowledge information flow into formal representations. The program's logic is designed at this stage. The knowledge collected is grouped into their categories based on common C++ programming errors contracted by C++ users. Then, the knowledge is organized in the form of decision tree.

One widely used representation is the *production rule*, or simply *rule*. A rule consists of an IF part and a THEN part (also called a *condition* and an *action*). The IF part lists a set of conditions in some logical combination. The piece of knowledge represented by

the production rule is relevant to the line of reasoning being developed if the IF part of the rule is satisfied; consequently, the THEN part can be concluded, or its problem-solving action taken. Expert systems whose knowledge is represented in rule form are called *rule based-systems*.

Basically, these production rules are used to insert all the knowledge based on the decision tree that is constructed into the Knowledge Base of the system. This is accomplished through the interface and the knowledge base editor of CORVID. Knowledge Base is the repository for rules and facts. These production rules that were inserted into the Knowledge Base are responsible to specify what action to be taken if certain condition exists. Beside that, CORVID has the CORVID Inference Engine which is able to match rules against the current state of the system to apply the corresponding actions. In other word, both Knowledge Base and Inference Engine work together in order to direct Cplus2ES user to the final solution of their problem.

All of the Expert System's data, facts and results that user capture for the specific problem during the consultation session are stored in the Working Memory. It is necessary for the working memory to store the information such as user's answer in order for the system to determine the following question to be asked to the user. The process of answering the question is keep repeated until the CORVID interference engine comes to the final solution, which is the problem solution of the specific C++ programming error.

Beside that, additional knowledge and information about C++ programming are stored in the database server and save in text and HTML file. Both of this file formats are ready for providing user with additional information once they was called by the system.

To easily represent the knowledge to the user, system interface need to be designed in an appropriate manner. Using CORVID software, user-friendly interface was built using JAVA CORVID runtime applet and interface command block. Beside that, some elements of HTML and PHP are used in order to make the system interface becomes

more attractive. All the working process on how this system arranges the knowledge representation to the user can be seen in the Cplus2ES System Architecture:

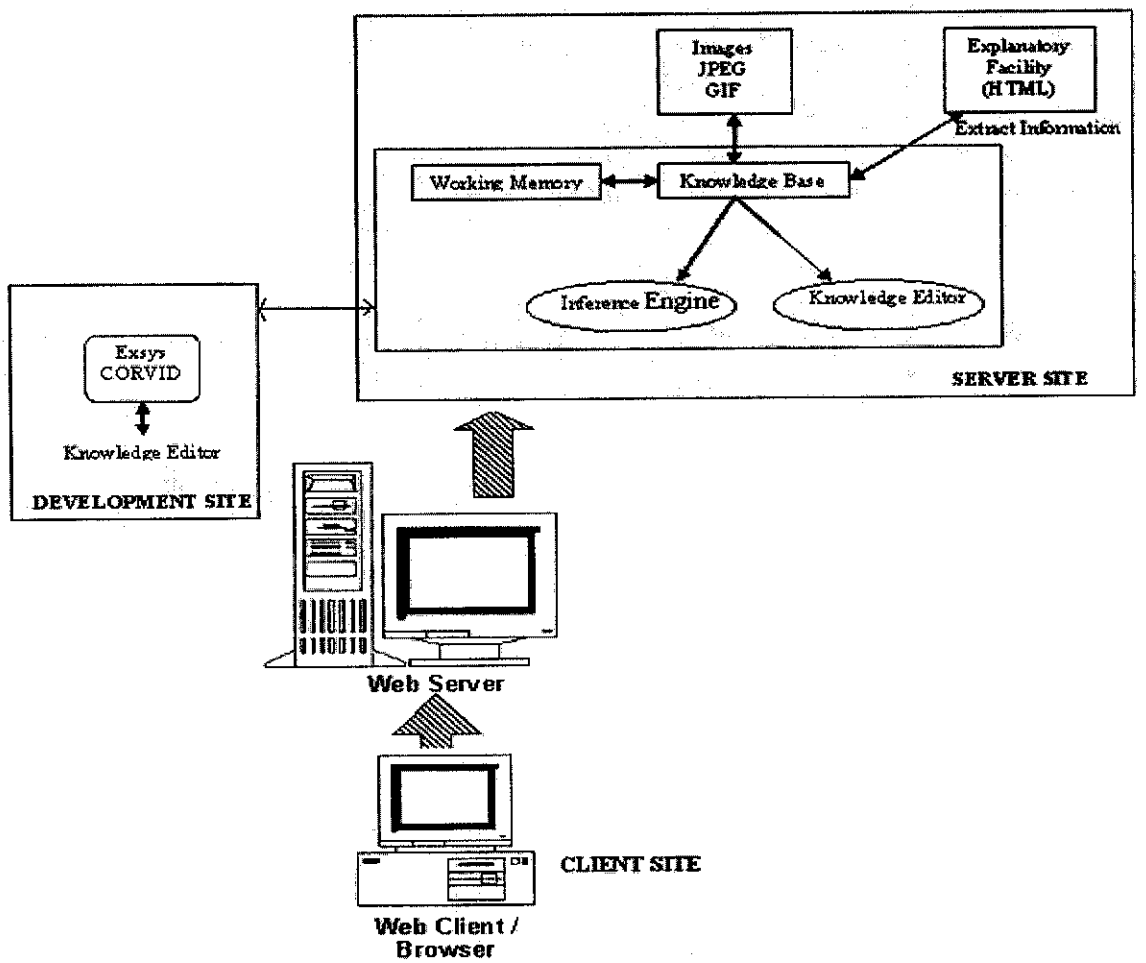


Figure 3.2: Cplus2ES System Architecture

3.1.3.3.2 Formalization of Cplus2ES's Knowledge

A large part in building an expert system is identifying the individual decision steps and converting them into a form that a computer can use. One of the most effective and efficient way in translating individual decision making into the computer language is IF/THEN rule as being used by the CORVID software. This is a rule where there is an IF part that can be tested to be true or false based on the date for a specific case or situation. When the IF part is true, the statements in the THEN part are also considered true. For this system, the THEN part the proposed solution of specific C++

programming error and its associated with CF. CF indicates the level of confidence of the proposed solution based on user input. The level of confidence in this system is 10. Below is how production rules are written in this system:

IF:

C++ is a different breed of programming language. A C++ program consists of syntaxes, identifiers, operators, functions and variables. Please choose one of the C++ problems listed below which I can help most?

AND: Semicolon is one of the important elements in C++ programming. Choose the error message at the debug window:

AND: Are you sure you are compiling a C++ file?

AND: Double click at the specified error message, the cursor will show which statement of the program that having an error. Do you notice any missing semicolon at the one of the place listed above?

AND: Double click at the specified error message, the cursor will show which statement of the program that having an error. Do you notice any variable declaration in the middle of a {} block as shown in the example above?

THEN:

You need to move the variable declaration to the top of the block, or add a new block to contain that variable declaration as shown in the example above:

Using CORVID's Logic Block, this production rule is expressed as follows:

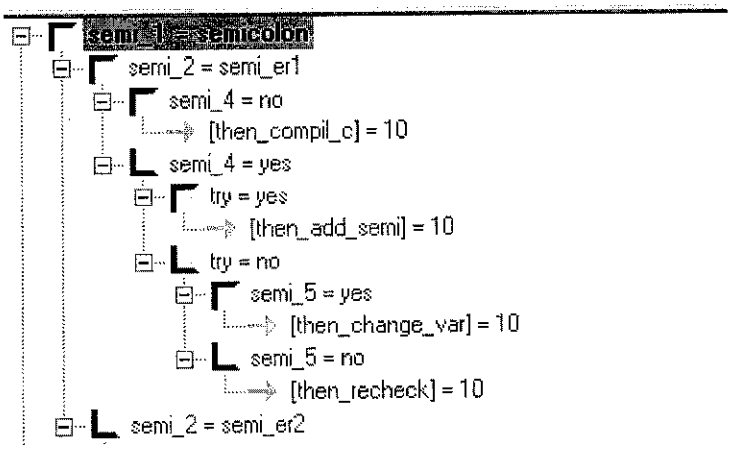


Figure 3.3: Logic Block in Cplus2ES

The Rule View in CORVID illustrates the full Variable prompt text as shown below:

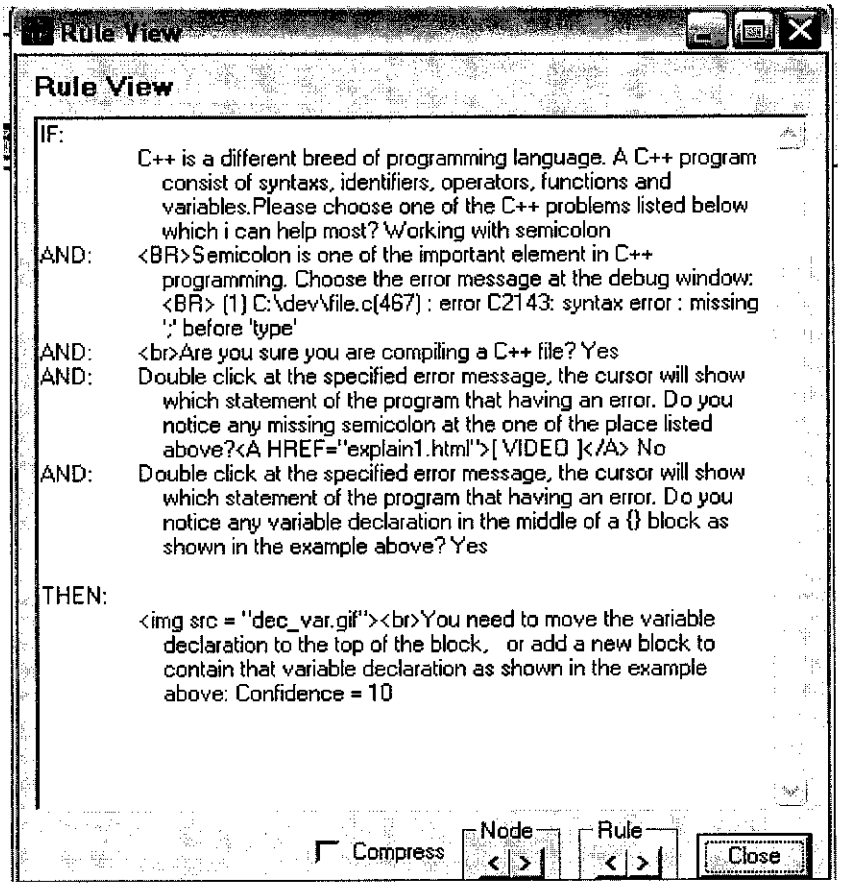


Figure 3.4: Rule View in Cplus2ES

3.1.3.4 Develop the Interface

Development of the interface should begin with the prototype development of the expert system. The interface should be an afterthought, but an integral part of the expert system development process. As mention in the Knowledge Representation part, the interface for this system is built using JAVA CORVID runtime applet and interface command block (Figure 6).

The CORVID Runtime program communicates with the end user to display the system title, ask questions and display message or result. CORVID provides a set of Interface commands that allow text and graphics to be formatted and included these displays. The

Interface commands also support ways to link text and graphics to other URLs and HTML pages.

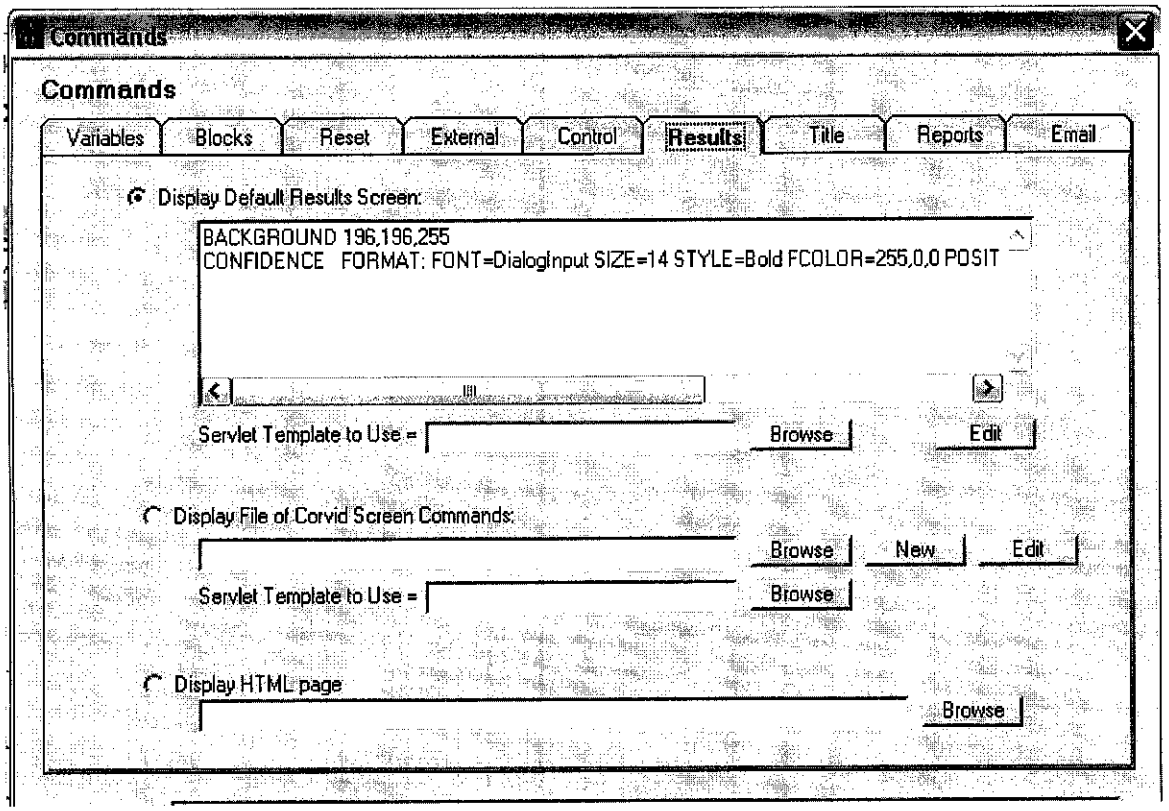


Figure 3.5: Interface Commands in CORVID

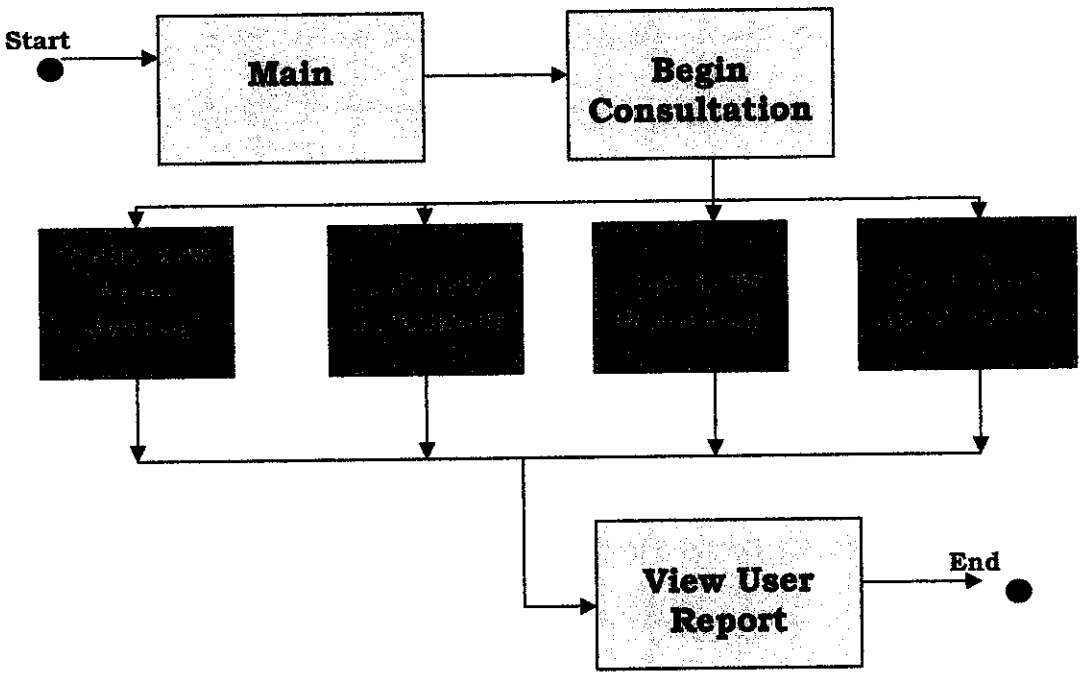


Figure 3.6: Cplus2ES System Flow

As shown in the system flow at Figure 5, once user enters the system, the main menu will be displayed. User need to click at the 'Begin Consultation' button in order to enter into the consultation session. This consultation session will diagnose user history and experience in using C++ software. Based on the information gathered during the consultation session, the Cplus2ES will determine user level of expertise.

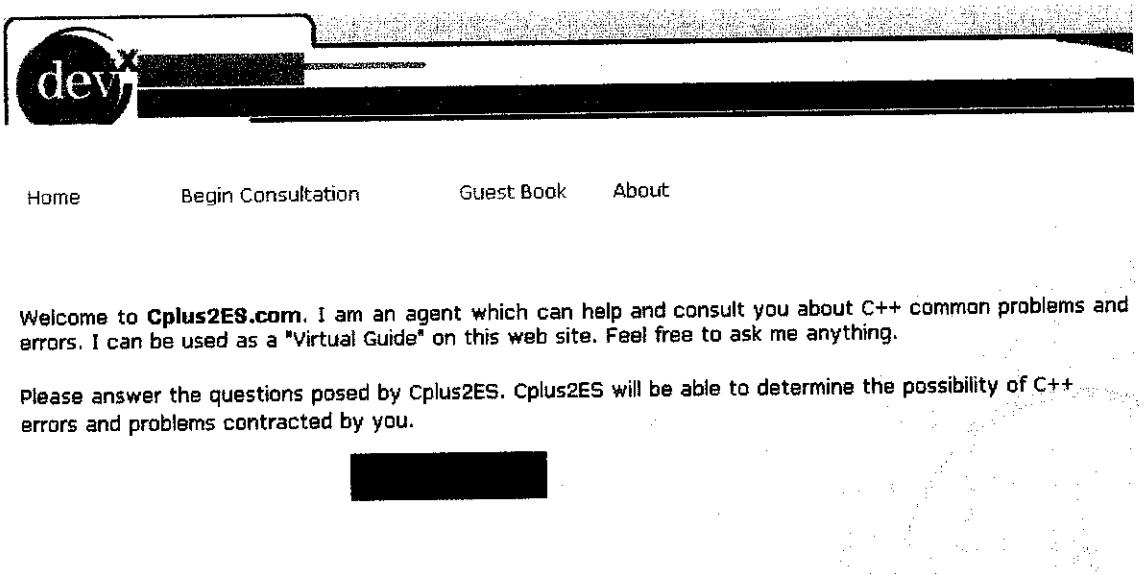


Figure 3.7: Cplues2ES Main Menu

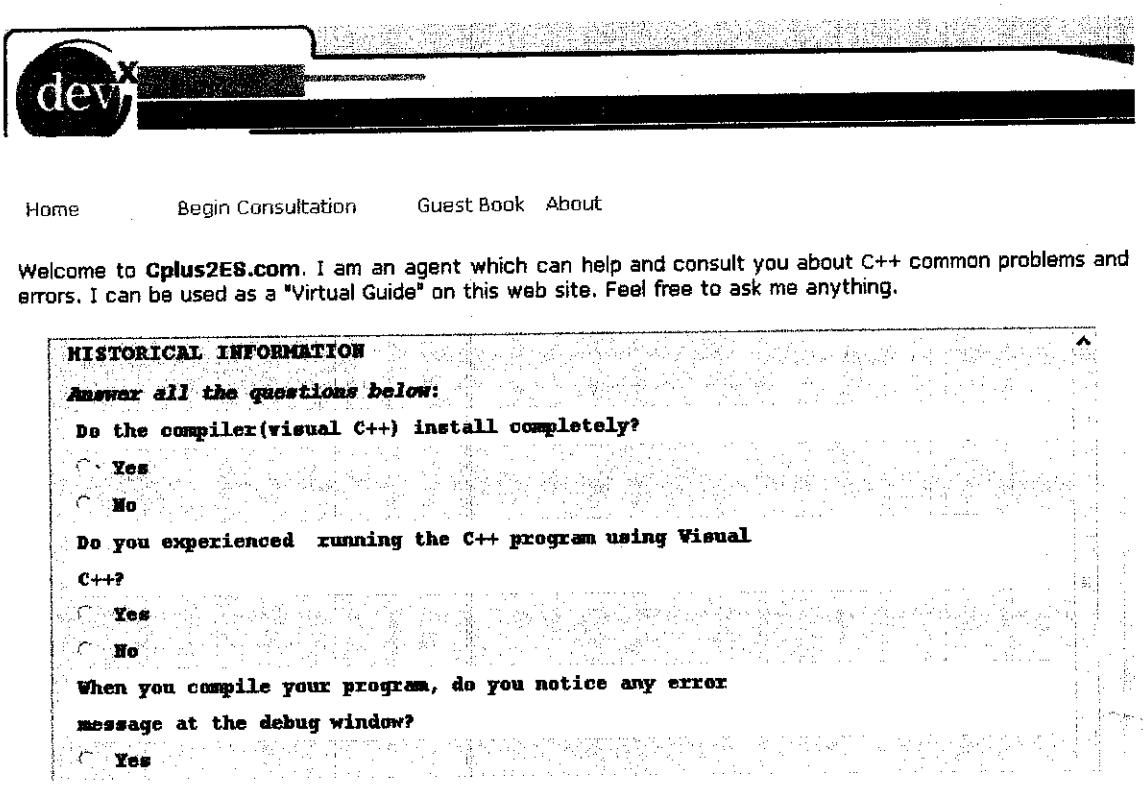


Figure 3.8: Cplus2ES Expert System

Cplus2ES Expert System is designed for the user who has an experience in using and running a program using C++ software. Basically, this is a main part in this system that

responsible to help user to find a solution regarding to common C++ programming errors. Users are required to answer the questions prompted in the form of text and also graphic. Beside that, link for the additional information for each question are provided. This additional information is stored in HTML file and VIDEO file. Based on the input gathered from the user, the Cplus2ES will come out with the suitable solution.

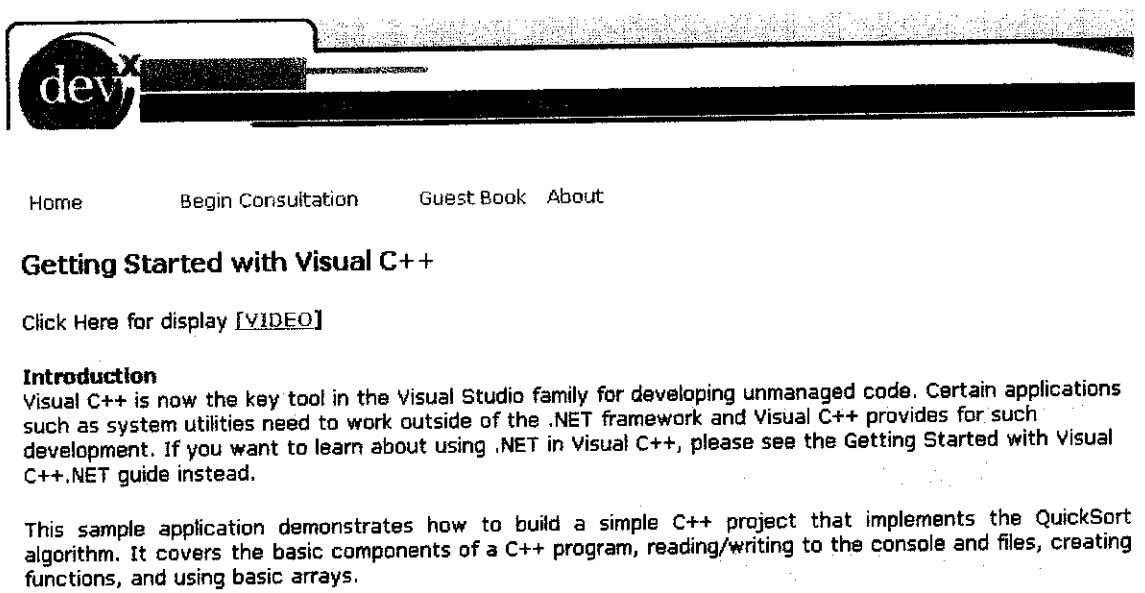


Figure 3.9: Cplus2ES Visual.net

Cplus2ES Visual.net is designed for the user who is new with the C++ software. At this part, user is provided with the step-by-step on how to run a program using C++ software. There two method available which is demonstration using combination of text and graphic and also video.

TYPE OF COMMON ERROR IN C++ PROGRAMMING THAT WAS DISPLAYED AT THE DEBUG WINDOW:

1. MISSING SEMICOLON

i. error C2146: syntax error : missing ';' before identifier glutSwapBuffers'
--->Expalanation: In this example, the identifier used is 'glutSwapBuffers'.

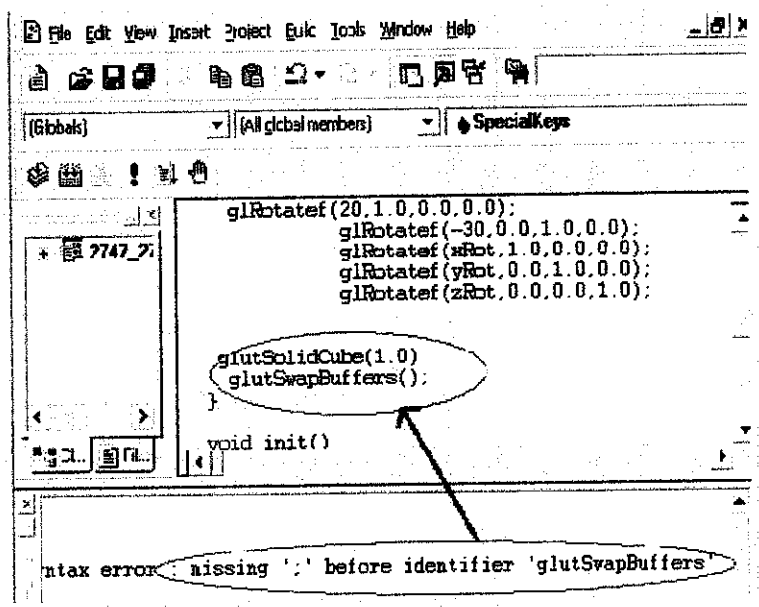


Figure 3.10: Cplus2ES Error Msg

Cplus2ES Error Msg is designed to help user who has a little knowledge about common C++ error messages that appear at the debug window of C++ software.

Cplus2ES Installation is designed in order to teach a new user on how to install C++ software into their workstation.

3.1.4 System Testing

As the project proceeds, the expert system will need to be periodically tested and evaluated to assure that is performance is converging towards establishing goals. For an expert system, the evaluation process is more concerned with system validation and user acceptance. Validation efforts determine if the system satisfactorily performs the intended tasks. User acceptance efforts are concerned with issues that impact how well the system addresses the needs of the user.

3.1.5 System Development

With each refinement, the system capability is improved. In an evolutionary fashion, the prototype system begins to take on the form of the final system.

3.1.6 Maintenance

The final phase of the project is system maintenance. This task follows the deployment of the system into the field where it is used routinely by the users. Most expert systems contain knowledge that is evolving over time. The changing state requires appropriate modifications to the system.

3.2 Tools

Tool required for this project are:

- ✓ Windows 98/ME/2000/XP
  500MHz processor
  128 MB RAM
- ✓ EXSYS CORVID Software
- ✓ HTML/PHP
- ✓ MySQL

CHAPTER 4

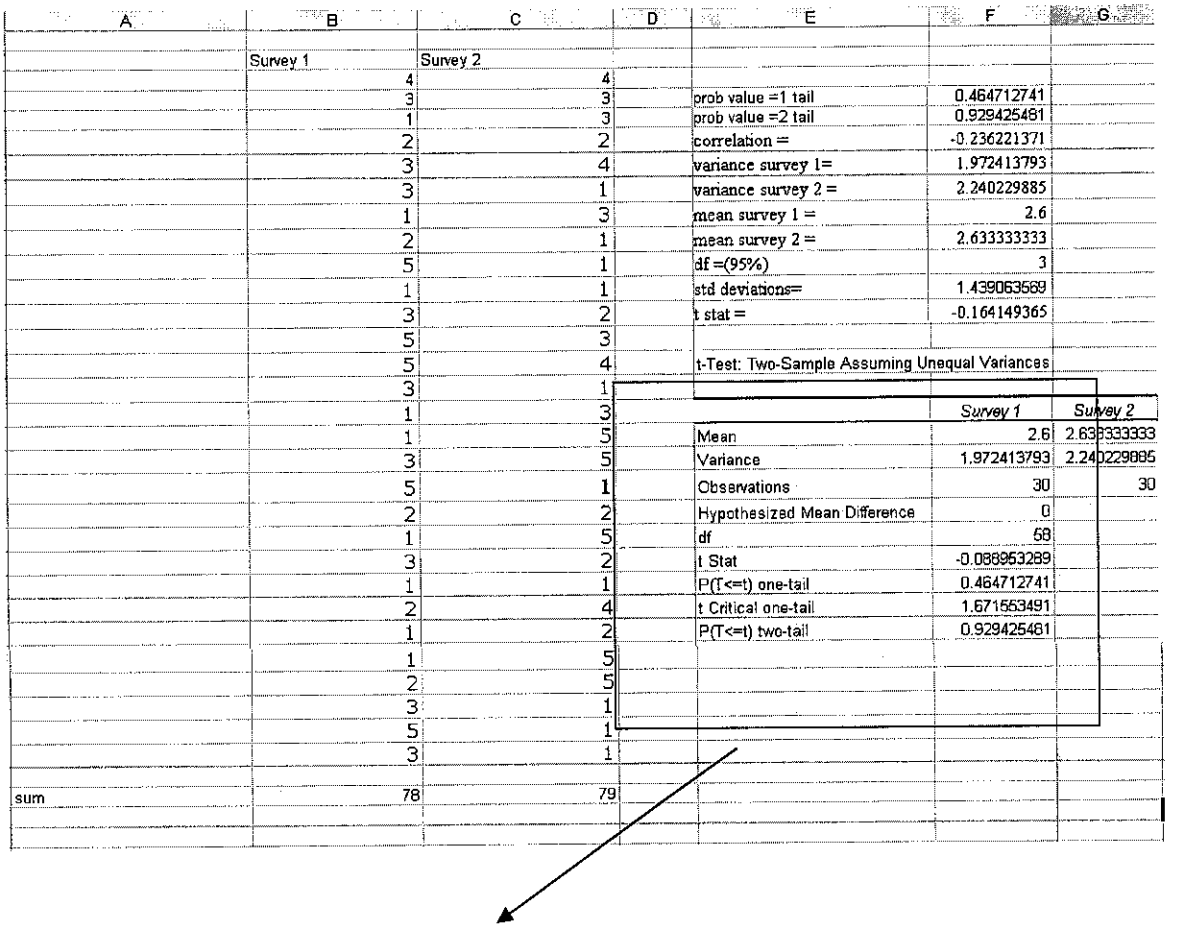
RESULTS AND DISCUSSION

4.1 Findings and Discussion

4.1.1 Data Analysis from the Questionnaire

Inferential statistics are used to make inferences about an unknown variable based on known descriptions. In this project, expert engineer used a t-test which is a simple procedure used to compare the means of two groups. Expert engineer also discussed alternative and null. If the probability of error is greater than accepted error, no matter what the numbers look like, null hypothesis must be accepted. In this case, the difference that may appear in the raw data is not significant enough to infer a difference in the overall population. Conversely, if the accepted error is less than the probability of error, the null hypothesis must be rejected. There is no gray area in this aspect as there are only two outcomes of any inferential procedure: reject or accept the null.

Below are results gathered from the evaluation of questionnaire that expert engineer conducted. There were 30 participants for each survey, which they need to answer a question regarding to the level of their satisfaction towards Cplus2ES expert system features and learnability.



	A	B	C	D	E	F	G
		Survey 1	Survey 2				
		4	4				
		3	3		prob value =1 tail	0.464712741	
		1	3		prob value =2 tail	0.929425481	
		2	2		correlation =	-0.236221371	
		3	4		variance survey 1=	1.972413793	
		3	1		variance survey 2 =	2.240229885	
		1	3		mean survey 1 =	2.6	
		2	1		mean survey 2 =	2.633333333	
		5	1		df =(95%)	3	
		1	1		std deviations=	1.439063569	
		3	2		t stat =	-0.164149365	
		5	3				
		5	4		t-Test: Two-Sample Assuming Unequal Variances		
		3	1				
		1	3			*Survey 1*	*Survey 2*
		1	5		Mean	2.6	2.633333333
		3	5		Variance	1.972413793	2.240229885
		5	1		Observations	30	30
		2	2		Hypothesized Mean Difference	0	
		1	5		df	58	
		3	2		t Stat	-0.088953289	
		1	1		P(T<=t) one-tail	0.464712741	
		2	4		t Critical one-tail	1.671553491	
		1	2		P(T<=t) two-tail	0.929425481	
		1	5				
		2	5				
		3	1				
		5	1				
		3	1				
	sum	78	79				

t-Test: Two-Sample Assuming Unequal Variances

	tech	*Eng*
Mean	2.6	2.633333333
Variance	1.972413793	2.240229885
Observations	30	30
Hypothesized Mean Difference	0	
df	58	
t Stat	-0.088953289	
P(T<=t) one-tail	0.464712741	
t Critical one-tail	1.671553491	
P(T<=t) two-tail	0.929425481	
t Critical two-tail	2.001715984	

Figure 4.1: Inferential Statistic using Microsoft Access

Steps in hypothesis testing using Inferential Statistics:

1. State the null and alternative hypothesis

$\mathbf{H_O}$ – represent the null hypotheses

$\mathbf{H_A}$. represent the alternate hypotheses

μ_R – represent the features of Cplus2ES expert system

μ_S – represent the student level of satisfaction

The null hypothesis:

There will be no increase of student satisfaction if the Cplus2ES features are not user-friendly

$\mathbf{H_O}$: $\mu_R = \mu_S$

The alternate hypotheses:

There will be an increase of student satisfaction if Cplus2ES features are more user-friendly

$\mathbf{H_A}$: $\mu_R \neq \mu_S$

2. Set the significance level

Significance Level = 0.05

3. Obtain the "probability value" based on result in figure 2 above.

t = -0.088953289

4. Compare the probability value to the significance level and make the statistical decision

The results of the t-Test are shown in Table 1. The t-Test is used because there are two samples that assume unequal variance. The alternate hypothesis $\mathbf{H_A}$ is directional. Thus, the one-tail statistical test is used.

The t value of -0.088953289 is significant as the value falls within the $\mathbf{H_O}$ rejection region on both tails, thus the null hypothesis is rejected. Thus, H can be substantiated. In other words, when the Cplus2ES are designed with the user-friendly features, there will be an increased of student satisfaction.

4.1.2 Data Analysis from Observational Studies

Regarding to observable done to a group of C++ user, which is student who taking C++ course at Universiti Teknologi Petronas, shown that they face a problem in doing their assignment and project given by their lecturer because of lack of knowledge to capture and interpret the error message display at the debug window of compiler. Even they are alerted that their program having error, but they do not know what and where the error are?

Common problem faces by UTP student in interpreting their program using C++ compiler:

- Lack of knowledge to understand and interpret the error message at the debug window
- Unreachable of expert (UTP's lecturer/tutor) in order to assist them
- C++ compiler manual / book that use language that not appropriate for people with little or no programming experience
- Limited of time and budget

In addition to help the student, Cplus2ES try to come out with a system that enables to help them solve their problem. It is available to the student at any time at

any place because it is a web-based system. Student who face problem such as do not fully understand the message given at the debug window by the compiler, can use this system to capture and interpret the root of their problem. It is because, this system will provide student with the appropriate solution based on the observable done when student answer a series of questions provided.

4.1.3 User Testing Analysis

Descriptive statistics is the simplest form of statistics. It lays the foundation for all statistical knowledge. Descriptive statistics are summary values that describe features of the distribution based on the data sample. These include statistics of location (mean, median, and mode), statistics of scale (variance), statistics of distributional shape (test of normality), quantiles (minimum, maximum) and counts of the data (sample size). Thus under this section, we will include and describe all the data that had been obtained from the survey.

The mean value, or is also known as the average value, is actually the sum of all the sample values divided by the sample size. It is the best estimate of the expectation (mean) of the underlying population. It is also the center of gravity of the histogram of the sample. Because the mean is calculated from all the sample values, it makes the maximum possible use of the available data. On the other hand, it can be influenced by any extreme value, which means it is not resistant. Median value in the other hand is representing the value(s) that are located in the middle of the sample. The sample mode is the single most frequently occurring data value. Samples from a continuous distribution may not have any repeated data values, so the mode is generally more informative with samples from discrete distributions.

Below are results gathered from the evaluation through questionnaire that conducted. There were 30 participants, which they need to answer a question regarding to their satisfaction of C++ software's features and environment. From the results, the minimum, maximum, mean, median, variance and standard deviation values had been calculated

30 data points were entered:
1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 2.00 2.00 2.00 2.00 2.00 3.00 3.00 3.00 3.00 3.00 3.00 3.00 3.00 3.00 3.00 3.00 3.00 4.00 5.00 5.00 5.00 5.00 5.00

Mean = 2.60

95% confidence interval for actual Mean: 2.076 thru 3.124

Standard Deviation = 1.40

Hi = 5.00 Low = 1.00

Median = 3.00

Average Absolute Deviation from Median = 1

The survey, shown the mean for the level of student satisfaction to this Cplus2ES expert system features and environment is 2.60. From this mean, we can conclude that the level of student satisfaction is average.

CHAPTER 5

RECOMMENDATION & CONCLUSION

An 'Expert' is one that can draw upon a comprehensive knowledge about a very specialized field or domain (that is, a restricted but reasonably complete Universe of Discourse).

An expert is also very good at solving particular types of problems, restructuring information in such a way that usually divides a problem into smaller, usually more easily solvable parts. An expert can also find multiple solutions, if they are appropriate, and can justify, verify or at least attach some level of certainty to its solution.

A Cplus2ES is a computer expert system that can work continually (24hrs a day), can be duplicated (thus creating many experts), never dies (taking knowledge with it), learns indefinitely (so long as new information is added to the system), always operates at peak performance, and does not suffer form personality incompatibilities. This expert system, however, is NOT INTELLIGENT. It may appear to be working intelligently, but that is because it is programmed to emulate human intelligence. All responses and prompts have been painstakingly designed, allowing for near 'natural' interaction. All decisions and conclusions are based on hard and fast rules, and heuristics (rules of thumb).

With the rules of thumb provided by the expert engineer within the Cplus2ES, it can bring user to the new style of solving C++ programming problem without facing and wasting time to see C++ expert. They can be consulted by the Cpuls2ES directly from the site.

REFERENCES

[1] Deitel & Deitel , *C++ How to Program*, Prentice Hall

[2] Deitel & Deitel, Internet & World Wide Web How to Program, Pearson Prentice Hall

[3] Cheang Azu Ann. 2004, *Feline Dermatology Expert System, Degree Thesis*, Universiti Teknologi Petronas, Malaysia

[4] http://www.accu-usa.org/2001-01.html

[5] Webopedia Online Dictionary, http://www.webopedia.com

[6] http://pluralsight.com/blogs/hsutter/archive/2005/03/14/6714.aspx

[7] http://msdn.microsoft.com/library/default.asp?url=/library/en-s/vccelng/htm/tions_31.asp

[8] http://www.hec.unil.ch/archives/isdss97/papers/84.htm

[9] http://64.233.179.104/search?q=cache:bqWO63B4OnEJ:www.exsys.com/pdf/AboutCORVID.pdf+exsys+corvid+facts&hl=en

[10] http://www.eng.upm.edu.my/~mdaud/kbp5612/m9.htm

APPENDIX I

Preface:

UTP has moved forward to be a world class university by providing first class facilities and accommodations. As an institute of higher learning, the students are given the freedom to express and protect their rights, together with UTP to generate a sophisticated and secure campus atmosphere.

The objective of this survey is to study the C++ programming problems faced by the UTP students .The result from this survey will be used to develop a system named "A Multimedia Enhanced Expert System for C++ Programming". This survey consist of two sections namely section A and section B. Respondents are required to answer all sections.

SECTION A : RESPONDENT'S BACKGROUND

Instruction :

*Please **bold** your answers.*

1. Gender

 Male Female

2. Year of study in UTP

 Foundation
 year 1st
 year 2nd
 year 3rd
 year 4th
 year 5th
 More than five years

3. Programme

 Information Technology
 Information System
 Mechanical Engineering
 Chemical Engineering
 Electrical and Electronic Engineering
 Civil Engineering

SECTION B : ***C++ Programming Problems using C++ compiler (Microsoft Visual.ne***
Instruction:

*Please **bold** your answers.*

1. **Do you know how to install C++ compiler (Visual.net)?**
   - *Yes*
   - *No*

2. **Do you experience running the C++ program using Visual.net?**
   - *Yes*
   - *No*

3. **When you compile C++ program, do you understand what the error message debug window mean?**
   - *Yes*
   - *No*

4. **Indicate the favorite reason for having problems with the C++ co by distributing a total of 5 point among them**

   - Misunderstood the message at the debug window _____
   - Unreachable of C++ expert _____
   - Language used in C++ book/manual is hard to understand _____
   - Limited of time and budget _____
   - *Others (please justify)* : _____

5. **Indicate the common C++ errors that you encountered by distributing a tota point among them**

   - Identifier error _____
   - Semicolon error _____
   - Keyword error _____
   - Header file error _____
   - Operator error _____
   - *Others (please justify)* : _____

5. **Do you satisfy with the environment / features provide by C++ software?**

Dissatisfied	1	2	3	4	5	Fully satisfied

(Please justify your reason)_____

6. Learnability

(The performance support system should be easy to learn for users)

	Always	Most of the time	Some of the time	Never	Not appli-cable
Did the C++ compiler behave in the way you expected?					
When starting the C++ compiler was it clear what you were expected to do?					
Was each screen clearly identified with an informative title or description?					
Was important information highlighted on the screen?					
Was it easy to find the required information on a screen?					
Were instructions presented in a clear way?					
Could you follow the instructions easily?					
Did you understand the instructions first time?					
Were parts of the C++ compiler difficult to use?					
Did the C++ compiler have distracting features?					
Were you aware throughout how the C++ compiler software fit into your personal educational scheme?					

Do you have any further comments about the learnability of the system?

7. Functionality

(The compiler should meet the needs and requirements of users when carrying out tasks)

	Always	Most of the time	Some of the time	Never	Not applicable
Were the tasks within the C++ compiler aimed at your intellectual level?					
Was the information presented appropriate for the task?					
Could you access all the information needed to perform the task?					
Did the C++ compiler allow you to do what was necessary to perform the task? Where task sequences were long, were they broken down?					
When you were presented with a list of options was each option clear?					
Did you get relevant feedback when necessary?					

Do you have any further comments about the functionality of the system?

8. Helpfulness of the system

(Informative, easy to use, relevant guidance and support should be provided by the compiler)

	Alwa ys	Most of the time	Some of the time	Never	Not appli- cable
Were appropriate help facilities available?					
Were the help facilities related to the tasks performed?					
Was it clear what actions you could take at any stage?					
Did the C++ compiler inform you of delays, making clear it were still working?					
Was there a cancel key/function for the user to reverse an error situation?					
Were there sufficient instructions when needed on the screen?					
Did you feel the C++ compiler helped you if you got confused?					
Were error messages explained positively?					
Did system instructions and error messages indicate what to do?					

Do you have any further comments about the helpfulness of the system?

9. Quality of the system interface

(The interface should be sufficiently flexible in structure, in the way information is presented and in terms of the user can do)

	Always	Most of the time	Some of the time	Never	Not applicable
Did you find the information was presented attractively?					
Did you find that the information was presented consistently?					
Were the icons and symbols easy to recognise and understand?					
Were the icons and symbols consistent?					
Was the language clear?					
Was the language consistent?					
Where jargon was used, did you understand it?					
Were the multimedia components (such as graphics and text) complementary?					
Did the use of colour help to make the C++ compiler clear?					
Was the sound clear?					
Was the video sequencing clear?					
Was the graphics clear?					
Was it possible to print certain parts of information you wanted to keep?					
Do you feel the video and sound features, throughout this C++ compiler software helped you to learn?					

Do you have any further comments about the quality of the system interface?

10. Controllability of the system by the user

(The user should feel in control of the compiler and the tasks to be performed)

	Always	Most of the time	Some of the time	Never	Not applicable
Did you have control over the order in which you carried out activities?					
Did you feel that you could decide what you wanted to do?					
Did the C++ compiler help you to set your own learning goals?					

Do you have any further comments about the controllability of the system by the user?

11. Workload imposed on the user

(The system should not put too much workload on the user)

	Alwa ys	Most of the time	Some of the time	Never	Not appli- cable
Were there too many instructions to read?					
Did you feel you had to remember much information?					
Did the C++ compiler prompt you if you were confused?					

Do you have any further comments about the workload imposed on the user?

Likeability of the system

(The user should find using the compiler an enjoyable, stimulating experience)

	Always	Most of the time	Some of the time	Never	Not applicable
Was the C++ compiler software entertaining to use?					
Was the C++ compiler challenging?					
Did the C++ compiler have an attractive presentation?					

Do you have any further comments about the likeability of the system?

APPENDIX II : GANTT CHART

No	Detail / Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	Topic Confirmation														
	Requirement Definition														
	• Introduction														
	• Objective														
	• Literature Review / references														
2	System and Software Design														
	• Project Planning														
	• Choose Tools														
	• Schematic Diagram														
	• Develop Expert System Rules														
3	Submission of Preliminary Report														
4	Develop Prototype														
	• Convert Rules														
	• Design System Interface														
	• Review with Supervisor														
5	Submission of Progress Report														
6	Enhance Prototype														
	• Pre EDX														
7	Submission of Dissertation Final														
8	Oral Presentation														
9	Submission of Project Dissertation														

Development phases

Dateline for report and presentations