Chapter 3 System Analysis

3.1 System Development Methodology

Professional system developers and the customers they serve, share a common goal of building information systems that effectively support business process objectives. In order to ensure that cost-effective, quality systems are developed which address an organization's business goals, developers employ some kind of system development Process Model to direct the project's life cycle. An expert system emulates the behavior of a human expert within a specific domain of knowledge. There is a clear distinction between an expert system and conventional software. This clear distinction clearly manifests in the way problems are resolved in the two systems. The systematic approach will ensure availability of usable, stable, scalable, upgradeable and maintainable system. Typical activities performed in developing a system include the following:

- System conceptualization
- System requirements and benefits analysis
- Project adoption and project scoping
- System design
- Specification of software requirements
- Architectural design
- Detailed design
- Unit development
- Software integration & testing
- System integration & testing
- Installation at site
- Site testing and acceptance

- Training and documentation
- Implementation
- Maintenance

While nearly all system development efforts engage in some combination of the above tasks, they can be differentiated by the feedback and control methods employed during development and the timing of activities. Badly planned which does not meet the deadline and requirements would only bring frustrations among users. This chapter will discuss three different approaches to software development such as System Development Life Cycle's SDLC Waterfall Model, Linear Model of Expert System Development and Evolutionary Development's Prototype.

3.1.1 Waterfall Model

This is a classical model of system development. An alternative name for this model is the 'one-shot' approach [5]. As can be seen from the Figure 3.1, there is a sequence of activities working from top to bottom. The output for the each stage is the input for the next stage. This model allows developer to review back each phase base on the new idea of the system. The diagram shows some arrows pointing upwards and backwards. This indicates that the later stage might reveal the need for extra work at an earlier stage but this should definitely be the exception rather than rule. After all, the flow of waterfall should be downwards with the possibility of just little splashing back. The limited scope or iteration is in fact one of the strengths of this process model.

The waterfall model defines each development stage sequentially. Development process is more visible because of this cascade from one phase to another. In practice, these phases overlap and feed information to each other. Therefore, the model doesn't depict a linear sequence of processes but iterations of activities.

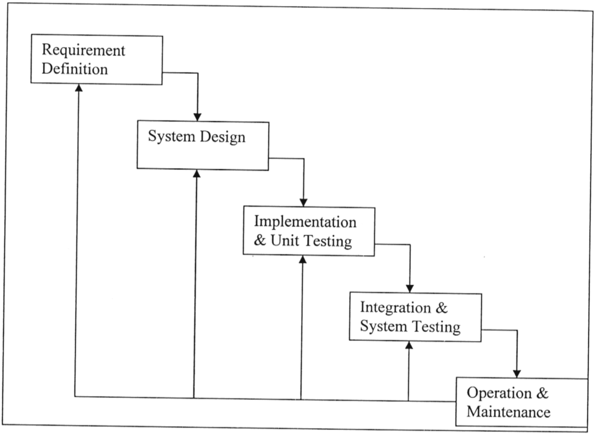

Figure 3.1: Waterfall Model

3.1.2 Linear Model of Experts System Development

Knowledge engineering has been accepted as one of the best approaches for developing an expert system. Most of the books on expert systems define knowledge engineering as the 'process of building an expert system' [4]. Knowledge engineering is a highly iterative process whereby the interest of the system designers lies in the problem knowledge. The designers partially build the system, test it and then modify it. This process is repeated throughout the project where the system's knowledge and designers understanding grows with each test.

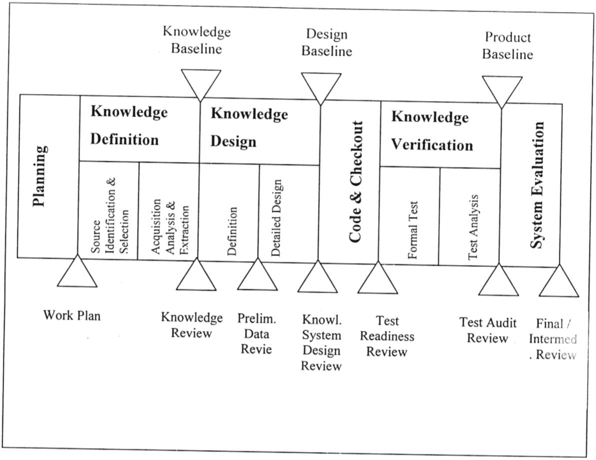

Figure 3.2: Linear Model Diagram For Expert System Development

The above figure depicts the main processes in developing an expert system. There are six main steps involved in the development of an expert system such as:

- Planning
- Knowledge Definition Knowledge
- Knowledge Design
- Code & Checkout
- Knowledge Verification
- System Evaluation

Planning

In the planning phase, basically studies are conducted to determine the feasible and justification of the candidate problem. Based on the feasibility study, the problem is further examined to determine overall goals of the project. At the same time, resource and schedules are planned and later preliminary functional layout is designed.

Knowledge Definition

In this phase, there are two main things to be done. Firstly, knowledge source identification and selection and secondly is the knowledge acquisition, analysis and extraction. Source identification, availability and selection are the important tasks to be taken into count in the knowledge source identification and selection whereas in the knowledge acquisitions, analysis and extraction the most important tasks involved are analysis of acquisition strategy, knowledge element identification, detailed functional layout, preliminary control flow and knowledge baseline. At this stage techniques like Data Flow Diagram (DFD), Entity Relationship Diagram (ERD), Structural Chart and State Transition Diagram (STD) can be used to understand the behavior of the system graphically. At this stage, both the functional and non-functional requirements is identified which are important to make sure the success of a system.

Knowledge Design

Knowledge Design is divided into two main tasks such as knowledge definition and detailed design. Once a thorough analysis has been performed the knowledge design will take over where the knowledge is defined in terms of knowledge representation, internal fact structure, preliminary user interface and initial test plan and then a detailed design is done such as designing the structure, implementation strategy, detailed user interface and detailed test plan.

Code & Checkout

Oncc the structure of the planned system has been designed, the code and checkout phase will take place. In this phase, an initial prototype is built to serve as vehicle for obtaining a better understanding of the problem. By building a small system and reviewing the test results with a domain expert, insights are gained into additional system requirements. The prototype also serves as the focal point for further interviews with the expert. At this stage user manuals, installation and operation guide are prepared as well.

Knowledge Verification

This is a testing phase. There are two types of testing will take part here such as formal tests, which include knowledge elicitation session with the domain expert to obtain further information that might have been overlooked in the earlier sessions, and refinements are made to the system. At this stage a test procedure will be used to examine the system and test reports are produced at the end of this phase. Next, test analysis is used to evaluate the test results and feeds recommendation to future enhancements.

System Evaluation

Finally, the verified system is then presented to the user to confirm meets their requirements. Once finalized, the completed system will be validated and final report or documentation is produced. The Figure 3.3 shows how the above is done phase-by-phase.

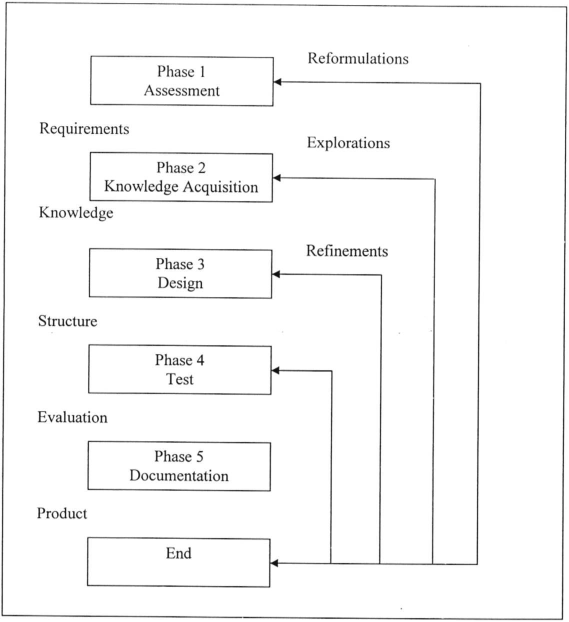

Figure 3.3: Phases in Knowledge Engineering

3.1.3 Evolutionary Development Model

Evolutionary Development Model as depicted in the Figure 3.4 is another popular development model available for our usage and reference. There are two types of evolutionary developments such as Exploratory Prototyping and Throwaway Prototyping. The exploratory prototyping is to work with the customers and to evolve a final system from an initial outline specification. This method should start with well-understood requirements. The development evolves by adding features to the prototypes as suggested by the users. By the end of the session, a final system is produced. The throwaway prototyping method's objective is to understand the system requirements. This type constructs the prototype just to understand and learn more about the problem or its solution and later it is discarded after the desired output is gained. This approach drags the requirements analysis phase but at the bright side the system developers would gain a clear and more clarified requirement specification.

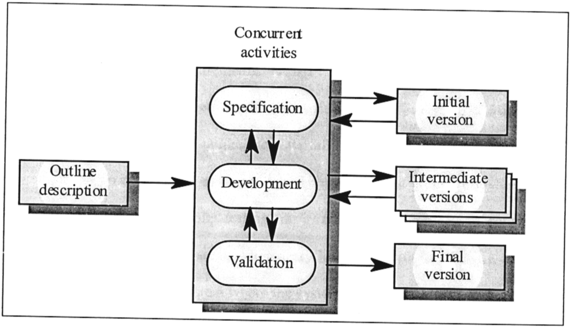

Figure 3.4: Evolutionary Development Model (Exploratory Prototyping)

3.1.4 Analysis and Synthesis

There are many Process Model or Life Cycle models available to choose as a methodology to develop our system but at the same time, the nature of the planned system is very important when selecting the process model. While nearly all system development efforts engage in some combination of the main four tasks such as analysis, design, coding testing and maintenance, they can be differentiated by the feedback and control methods employed during development and the timing of activities. Problem solving in expert system is predominantly heuristics based on trial and error and guided by some reference to a predetermined goal. Problem solving in conventional software is usually through the algorithmic methods, which is procedural in nature. In, summary expert system encodes the domain dependent knowledge of everyday practitioners in some field and uses this knowledge to solve problems, instead of using comparatively domain-independent methods derived from Computer Science.

Compared to Waterfall Model, Knowledge engineering can actually reduce the shortcomings with an initial system prototype built to retain both of the domain expert's and system requirements. Apart from that, knowledge engineering development model does not require any of its phases to remain frozen before proceeding to the next phase unlike the waterfall model. The model allows the system developers to go back to any of its preceding phases without going through each phase in between. As an example testing is a continuation process throughout the development of an expert system. After every interview with the domain expert, new knowledge is added to the system where the system's knowledge is modified or added a number of times.

Knowledge engineering's prototype is built as a means to capture the problem domain correctly rather than as a tool for requirement analysis as what in the prototyping model make use of the prototype. Knowledge engineers only interact with the domain expert to capture possible system knowledge organization and structure and the intended users only interact with the prototype after the completed system is released to them unlike the prototyping model where the users are allowed to interact with it to assess further the user requirements.

As a conclusion, Linear Model of Knowledge engineering suits the best for developing Ctutorial4u since it's a Rule Based System because it is better to adopt a system development strictly designed for expert system. In that case, let's look at how the main six phases in Linear Model of Knowledge Engineering helped to mould CTutorial4u to be a successful knowledge based system.

Planning

During the planning phase of CTutorial4u, studies were conducted to determine the feasibility of the project through surfing the internet for similar systems, discussion with supervisor and questionnaire analysis among the higher learning students in Malaysia. Based on the feasibility study, the problem is further examined to determine overall goals of the project such as confirmation of intended users of the system, scope and objective of the project. At the same time, resources and schedules are planned and later preliminary functional layout is designed. This phase is actually equivalent to the System Conceptualization of typical system development methodology.

Knowledge Definition

As stated earlier, knowledge definition phase is divided into two sub phases which are knowledge source identification and selection and secondly is the knowledge acquisition, analysis and extraction. Knowledge identification phase is where all the sources, basically the rules to develop the system are identified. Through the literature review objects or entities, classes and use cases are identified. The knowledge acquisition phase helps to identify the relationship among the entities which visualizes the system in real world. The UML diagram is used to understand the behavior of the system graphically. The flow chart and structure chart were used to visualize the flow of the system. This is a combination of system requirement and project adoption and project scoping phase of typical system development methodology. At this stage, both the functional and non-functional requirements is identified which are important to make sure the success of a system which will be explained further in chapter 4.

Knowledge Design

Knowledge Design is how we convert the defined knowledge at knowledge definition phase into the form of system. All the classes in the class diagram were converted to be tables in Microsoft Access 97 whereas based on the flow chart and structure chart, user interfaces were designed in Visual Basic 6.0. System design, specification of software requirements, architectural design and detailed design phases of typical software development methodology were done collectively at this phase of Linear Model of Knowledge Engineering.

Code & Checkout

In this phase, an initial prototype was built to serve as a vehicle for obtaining a better understanding of the problem. By building a small system and reviewing the test results with 5 students with intention to gain additional system requirements. The prototype also serves as the focal point for further interviews with the domain users (students). At this stage user manuals and installation were prepared. When compared to typical system development methodology it is same as system integration & testing phase.

Knowledge Verification

Formal test done to verify all requirements were met and include knowledge elicitation session with the domain user to obtain further information that might have been overlooked in the earlier sessions, and refinements are made to the system. User Acceptance Test (UAT), a test procedure was used to examine the system and test reports are produced at the end of this phase. Next, test analysis is used to evaluate the test results and feeds recommendation to future enhancements. It is same stage as site testing and acceptance of Linear Model.

System Evaluation

Finally, the verified system is then presented to the user to confirm meets their requirements. The completed system was validated, implemented and final report or documentation is produced. Students were produced with user manual and tutors and lecturers were produced with installation guide. The system will be maintained by the tutors and lecturers in the future. So, this phase is equivalent with training and documentation, implementation and maintenance of the Linear Model of Knowledge Engineering.

3.2 Requirement Analysis

Preliminary study and analysis was done to collect data and information. Requirement collection and analysis is the process of collecting and analyzing information about the system that will be developed and to identify the user's requirements of the new system. A well collected data and information is vital to meet the objective of the project.

3.2.1 How to elucidate user requirement

There are many ways to collect user requirements such as interview, survey, study similar system or project and so on. Actually, information required for this project was gathered through discussion with the project supervisor, survey using questionnaires to gather student's perception of C programming taught in their institutes and expectation about the knowledge based tutorial system and by surfing the internet for more information on the similar system available.

3.2.1.1 Discussion with the project supervisor

Discussion with the project supervisor, Puan Nazean Jomhari was done earlier to determine objective and scope of the project, the system requirements, information and data needed and the expected result of the project. Apart from that, from time to time clarification on the requirements is discussed with Puan Nazean. Other than that, she also gave some ideas on how the system should look like and through that her expectation was notified. The discussion with the supervisor helped a lot in designing the system.

3.2.1.2 Survey about C among students using Questionnaires

The questionnaire as attached in the Appendix A was distributed to a few local universities and private colleges in Klang Valley area in order to investigate the way C programming subject being taught in higher learning institutes of Malaysia. The survey results were then evaluated on how to improve the passing rates among the students using knowledge based C tutorial system designed in rule-based system architecture. The next section which is the functional requirements depicts all the analysis with the graphs from the questionnaire.

Firstly, a set of questions were prepared based on the requirements and information needed in developing the system. The questionnaire is divided into 3 parts; part A is the personal information, part B is general question about C programming and part C is about the proposed rule-based C tutorial system. Then, the higher learning institutes in Klang Valley were identified to take part in the survey. The questionnaires were distributed randomly to the students who are taking C Programming subject in every particular institute. The students were asked to answer the questionnaire forms. All the answered questionnaires were collected and analyzed. There are 112 students from 15 higher learning institutes involved in the survey. They are from Universiti Putra Malaysia (UPM), Universiti Malaya (UM), Universiti Islam Antarabangsa (UIA), Universiti Kebangsaaan Malaysia (UKM), Universiti Teknologi Mara (UITM), Universiti Multimedia (MMU), Univeristi Tenaga Nasional (UNITEN), Politeknik Shah Alam, Cosmopint College, Informatics College, Sedaya College, Kolej Yayasan Mara, Kolej PTPL Shah Alam, Stamford College and Federal Institute of technology (FIT).

Part A of the questionnaire helps to identify background of the student. Question 1 identifies the student's higher learning institute and whereas question 2 helps to find out the semester and the year of the student in their respective institutes and question 3 is to find out the field or program the students are persuading in that institutes. This will help in identifying the standard of the system whether to be easy or higher level.

Basically part B of the questionnaire helped to identify when exactly the C Programming subject is being offered in all the institutes in Malaysia. The questionnaire's core motive is to further seek the students' perception towards this subject. Question 1 of the Part B is to find out when the C programming subject being offered in their university and question 2 will clarify are there are any prerequisites required to register for this subject. Then the students' opinion about the subject was identified in question 3 where some of them think that it is very easy, moderate and there are also students think that it is very difficult. The intention of this question is to find out how many students think that C programming is difficult. Their perception differs based on their family background and whether they have done programming subjects in their school. For those who never had any experience in programming in their secondary or primary school this subject looks like quite difficult to be adopted. Question 4 is to find out whether the students agree or disagree with the statement that assignment and lecture materials prepared by the lecturers are good. From this question reasons for the students failing or getting poor marks for this subject can be identified where if many disagree then improvement is required in teaching materials.

Question 5 is to identify how this subject is being taught in their institutes so that this system will be designed to be more helpful rather than normal teaching materials such as lecture notes, lab session and assignments. Question 6 is to verify whether the lecturer who is teaching this subject has all the characteristics listed. This is to make sure whether the university is selecting a qualified person as the lecturer to teach C programming. The average mark of the students in all their tests or quizzes is asked in question 7 to seek ways to improve and deliver a better tutorial system if most of them get low marks and adopt the lecturer's style of teaching if most of the students from those institutes that get high marks. Question 8 is to find out students problems with their current lectures. This question will help the developer to cater and eliminate those problems. At the same time, student's suggestion and recommendation is captured as well in the Question 9 on how to improve the course.

Apart from identifying the student's opinion about programming, there is also another part, Part C designed to identify the student's expectations of the proposed system. This section rendered a helping hand in designing the flow and features of the system. Question 1 of Part C to capture the features expected by the students such as whether the students wants the tutorial to cover all topics from a programming book, short and long quizzes with answers, consultation session, sample sources code and search function. If the features given have more than 50% respondents who favor it then that feature will be included in the new rule-based tutorial system. Question 2 seeks the student's preferences on how the system to be designed whether by selected topics, by application or by all chapters following a reference book. Question 3 is to find out how long a student needs to finish a quiz, which consists of 10 simple questions. From this question the short and long quizzes can be designed and the time out can be set based on the majority of the answers. Question 4 is to find out how to organize the quizzes whether after every chapter or mixed questions from all chapters or both ways. Moreover, question 5 helps to highlight what are the C programming topics to be given more emphasis and consideration. Lastly, Question 6 was designed to seek the students preferences whether the system need to be standalone system or an online system.

3.2.1.3 Surf through the Internet and research on reading materials

Internet is a vast network of interconnected computers. The deluge of web pages has generated a strong commentary on the tragedy of the flood of information. Surf the Internet on existing systems to adopt some ideas and to collect relevant pictures and graphics. So, books on developing an expert system were read. Then, Microsoft Image Composer and Adobe Photoshop 7.0 were explored on how to manipulate the pictures. The reason to use some multimedia tools is to develop a system, which can attract the users and make the system interesting and attractive.

3.2.2 Functional requirements

Functional requirements describe the interaction between the system and its environment. Functional requirements capture the tasks that a system must perform and as such does not include implementation details such as what hardware or software the system must use. The functional requirements of CTutorial4u include administration session, lecture notes session, tutorial session and quiz session. The features planned to be embedded in the system in the beginning is as in the Figure 3.5 but based on the visibilities of all the features the real features are administration session, lecture notes, tutorial session and quiz session.

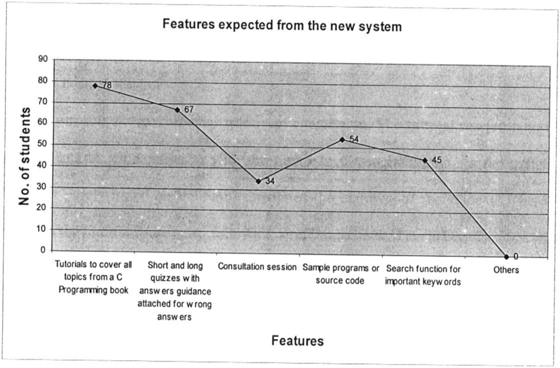

Figure 3.5: Line graph shows the features expected from CTutorial4u

3.2.2.1 Administration session

The main important function is the administration session where the system administrator has extra authority that allows them to maintain the system functionality such as add and modify the queries in the knowledge base of tutorial and quiz session. Any modification to the system knowledge base could occur when an error is detected or if the students decide to add more queries to the quiz session to enhance its usability because only the lecturer or tutors who act as the administrator is allowed to modify the system.

3.2.2.2 Lecture notes session

Lectures notes are an essential session to the students to explore before they try the tutorial session and quiz session. Lecture notes contains ten chapter such as Introduction, Condition, Program Controls, Functions, Arrays, Pointers, Characters and strings, Formatted input/output, Structures and Unions and File processing. These chapters are all in power point format.

3.2.2.3 Tutorial session

Tutorial session is the most vital feature of Ctutorial4u. The tutorial session is to help the students to get familiarize with the topics in C programming. As the system is designed purely based on the requirements of the majority of the students where the topics in the tutorial is designed following a reference book. This is because 80% which is 90 of 112 students chose the third option for question 2 of part C. The Figure 3.6 depicts the analysis of on how to design the tutorial session which is captured through question 2 of part C of the questionnaire.

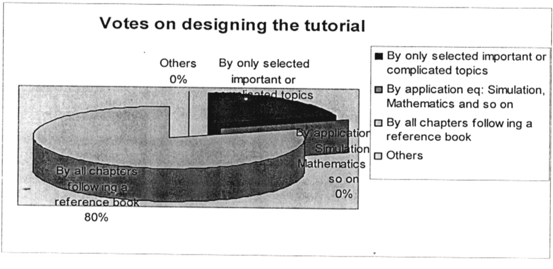

Figure 3.6: Pie Chart shows votes on ways to design the tutorial

At the same time, there will be more solutions and examples available for the arrays and sorts then followed by pointers, counters, control structures and string functicns and loops. Most of the students, 45 chose arrays and sorts as the toughest topic and followed by 42 students who answered pointers as the most difficult chapter to understand. Subsequently, only 12 students say counters and only 8 students chose strings and loops as the most difficult chapter in C Programming. The least of the students where only 5 chose control structures as the most difficult topic in C programming. This information was captured from question 5 of Part C of the questionnaire and Figure 3.7 shows the results for that question.

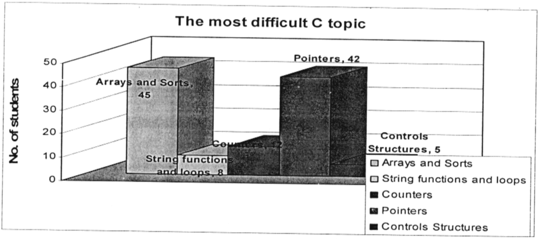

Figure 3.7: Bar chart shows students' choice of the most difficult C topic

Tutorial session covers topics such as:

- Arithmetic - Multiplication, Division, Addition, Subtraction & Average.
- Control Structures - Selection structure (If, If/Else, Switch)
  - Repetition Structure (Do, Do-While, For)
- Functions - Maximum & Minimum
- Array - Size of array from 1 to 10 & with or without histogram.
- Pointers - Call by reference & call by value
- File Processing - Create a Sequential file

- Writing to a Random Access file
- Reading a Random Access file

The purpose of the session is to display the source code based on the user's selection from the options provided for each chapter. Then, when the users select to execute the code, the system will execute the code and show the output. The details of this session can be clearly seen in the user manual in Appendix C.

Firstly, arithmetic topic is not planned to be covered in this session but at the end advised by the supervisor to cover it since it would be a stepping stone for the users to go further to a more tricky topics like arrays. Arithmetic calculation is the most imperative function to introduce to the beginners because the users will understand better, how day-to-day arithmetic calculation being applied in C programming rather than straight away jump to the programming side. The Deitel and Deitel book also do the same method where the first chapter of the book is arithmetic calculation. Multiplication, division, addition, subtraction and average are the arithmetic types that will be taught in this session. This chapter will actually show example of codes based on the user input for arithmetic types, variable type (integer, double and float) and value of variable A and B. Once user gives inputs for all these mandatory fields, the source code will be displayed in the list box on the right panel. After that, the user can select whether to execute the code or not. The beauty of this system is to show how the real output in DOS prompt when source code is being executed in Visual C++ compiler. Please refer to Appendix C for more details on this topic at Figure C.9 and C.10.

Control structure is divided into two types which are selection structure (if, if/else & switch) and repetition structure (so, do-while & for). The source code will be displayed based on the user's input for the test grade. The selection structure will be based on a scenario to print whether the student has passed the examination or failed based of the marks or grade entered where the pre-condition set in the working memory is 60 for passing grade. So if the grade is more than 60, then message "Passed" will appear and the other way round if it is less than 60, where "Failed" message will appear as in Figure C.12 of Appendix C. The repetition structure is to calculate the average marks for a student based on five tests grades.

The function topic, actually introduces maximum and minimum function. The system will display the source code to identify the maximum and minimum value based on the user's input for the integer 1, 2 and 3. When the source code being executed, the integer values keyed in by the user will be verified and the system will show the maximum or minimum value based on the selection for the function type selected by the user. The screen design of this topic is depicted in Figure C.13 and C.14 of Appendix C.

The topic array is introduced in this system to show how array as a group of memory locations related by the fact that they all have the same name and the same type and users needs to specify the type of each element and the number of elements required by each array so that the computer may reserve the appropriate amount of memory. Moreover the Deitel and Deitel book introduces histogram in conjunction with every value of the element in the group of array. Based on the array size selected by the user (minimum 2 and maximum 10 array size) and value of each element also captured from the user's input and then the source code will be displayed. So, when the source code being executed, the output will be ("Element, Value, Histogram") as in the Figure C.16 of Appendix C.

The topic pointer covers ways to pass argument such as call value and call by reference. Call by value uses direct passing argument based on the value integers keyed in, the system will cube the value and square the value based on the user's selection. In call by reference the concept of array plays a major role. The pointer will be pointing to the variable in the array and call the function cube by reference or square by reference. Then there is an option whether to sort the integer variables in the array or not. If the user selects to sort, the system will display the code for bubble sort. Please refer to Figure C.17 and C.18 in Appendix C for more details.

The file processing topic contains processes like Create a Sequential file, Writing to a Random Access file and Reading a Random Access file. Based on the variable type, variable name and file name, there will be source code displayed to create or to write to a sequential file as in Figure C.19. Whereas for reading a random access file, the system will display pre-coded program saved in the working memory.

3.2.2.4 Quiz session

There will be quiz organized in both after every chapter and mixed question from all chapters because 57% of the students chose to have the quiz both ways after every chapter and as well as have mixed questions from all chapters at the end of the course. This user needs was captured from question 4 Part C and this is depicted in Figure 3.8. The quiz session covers 10 simple questions from chapter arithmetic, control structures, functions, arrays, pointer, file processing, and mixed questions from all chapters.

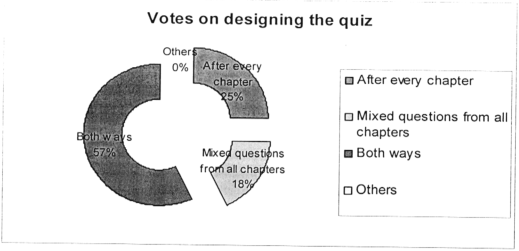

Figure 3.8: Doughnut chart shows the responses on how to design the quizzes

The time limit to finish ten simple questions is fixed to 15 minutes just to meet requirements of the majority users where 67% of the students chose 15 minutes for the question 3 of part C and the result of the survey is shown clearly in the pie chart Figure 3.9.

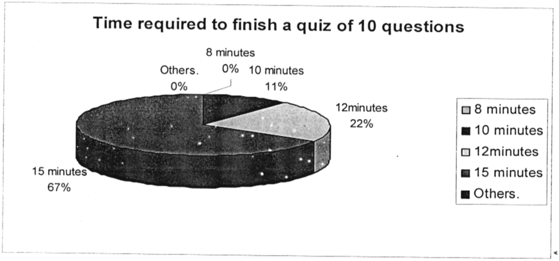

Figure 3.9: Pie chart on time to finish a quiz consists of 10 simple questions

3.2.3 Non-Functional Requirements

The non-functional requirement is the added-features of a system where it can't be seen directly but it is embedded in the system itself. Among the non-functional requirements for the Ctutorial4u project includes:

- **Reliability & Usability**

The Ctutorial4u application should be able to derive the correct and reliable output for each of the functions available. The Ctutorial4u interface should also be easy to use and understand.

- **Security**

The security function or the authorization and authentication function prevents unauthorized users from gaining access to the system and looking at others scores and information. This function will provide the system a sense of security in an electronic based environment. When a user login to the system a special user login request is sent to the system to validate the user and establish a user type (admin and user). If the password and user ID provided by the user is incorrect then the system will reject the request and ends the session. Only the administrator can modify the knowledge base and inference engine with clarification from the domain expert.

- **Services**

Ctutorial4u will be able to provide a 'Help' function where the users can refer to in times of ambiguity. The help section serves to provide adequate explanation to the students and lecturers as to how and why it arrives to a solution after a query session with Ctutorial4u consultation has ended. At the same time, step by step on how to use the system is given in this section. This requirement is not gathered from the questionnaire but by surfing the internet and looking in other similar products.

- **Consistency & Efficiency**

The interfaces for all the pages arc consistent in the use of colors, font sizes, text structure, available graphics and menu bar. Consistent and efficient screens will allow the users to use the features provided without having to memorize all the commands to perform certain tasks.

3.3 Development Environments

Development requirements include hardware specifications and software requirements required to develop the system.

3.3.1 Hardware Requirements

Ctutorial4u would interface with the following hardware requirements:

- Pentium II 100 MHz processor or higher
- At least 2 GB of hard disk
- At least 32 MB of RAM
- Other standard computer peripherals – monitor, keyboard, mouse printer and CD-ROM.

3.3.2 Software Requirements

The Table 3.1 provides the minimum software requirements required to develop the CTutorial4u.

Table 3.1: Software requirements for Ctutorial4u

Software Type	Required Software	Remarks
Operating System	Windows 98	Minimum OS required is Win 98
Database	MS Access 2000	MS Access is required to develop the database
Software Development Tool	Visual Basic 6.0 (VB6) Adobe Photoshop 7.0 MS Image Composer MS Office 2000 MS Visual C++ 6.0	VB6 and will be used to develop the user interface. MS Visual C++ 6.0 is to compile and run the code before adding to the knowledge base. Adobe Photoshop 7.0 and MS Image Composer are required to design the image needed whereas MS Office 2000 is required to produce the documentation of the system.

Operating System

Microsoft Windows 98 was chosen as the development platform because it is easy to use and flexible. It is also supports Visual Basic as well as other development tool. Windows 98 is also very user-friendly, safe and highly reliable.

Database

The database chosen for the CTutorial4u is Microsoft Access 2000.

Software Development Tool

Visual Basic 6.0 is chosen as the software development toll to develop the user interface and at the same time code is developed in VB6.0 development environment. Adobe Photoshop and Microsoft Image Composer helps to design the image needed to use in designing the interface of the system. Microsoft Word 97 is used develop the documentation of the system. MS Visual C++ 6.0 is to compile and run the code before adding to the knowledge base.