CHAPTER 4

Methodology in Developing VECITS

Tutoring and learning theories are very complex which involve complication in research. Methodology may help researcher to analyze information in order to present it to the system. The surveys include finding out what are the most important sub topics that students should know in studying vector and what type of questions and answers they usually encounter.

The types of the wrong answers that are usually given by students in answering the question are also considered in the study. This is due to the fact that ITSs stress on what the wrong answers that are usually given by the student in order to guide them how to arrive at the correct answer rather than giving only the "correct" or "wrong" answer. In developing VECITS, the design models, the components and the format used are considered. These methods are derived from the existing ITSs which has been developed.

4.1 Design of the Study

VECITS is an ITS system, which emulates a human instructor for teaching a topic. It consists of an expert model, which knows what to teach (domain content), an instructional model which knows how to teach (instructional strategies) and a student model which simulates the student who is being taught. Through the interaction of these models, VECITS are able to make judgements about what the student knows, and how well the student is progressing.

4.1.1 The Expert Knowledge Model

The Expert Knowledge Model in VECITS only use two types of knowledge architecture from the various models discussed in 2.2.2.1. The models are Black Box Model and Knowledge Structure Model.

Black Box Model

The modeling technique mainly used in this study is **black box model** with the features of other models implied indirectly. The definition of black box model has been discussed in 2.2.2.1. The structure of the model is shown in the figure below [Vladimir A. Goodkovsky].

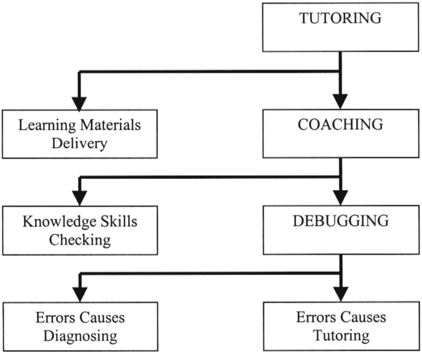


Figure 4.1: The Structure of Black Box Approach

The model uses step by step process beginning with Tutoring and ending with Debugging. Tutoring provides the basic knowledge of the subject by delivering the learning material to student. Coaching will provide a knowledge skill checking of mastery to the system and also detect if there is any error that occurs. Knowledge skill can be check by providing questions. Debugging will allow system to detect the error done by the student and give some hints or help to give student a clue. These are the basic steps in this system development.

Knowledge Structures Model

Knowledge Structures model provides the rule base in an **IF-THEN** or **WHEN-THEN** relationships. The study focused on the rule-based **IF-THEN** relationship that analyzes what would happen if one condition couldn't be met or wrong. For example

```
to handle evaluate
        if not ASYM_CompareByCase(_lastText of self, text of self)
                if _locked of self = TRUE
                        send ASYM_RegisterResponse self,text of self to self
                        text of self = _lastText of self
                else
                        _lastText of self = text of self
                        send ASYM_RegisterResponse self,text of self to self
                end
        end
end
```

Figure 4.2: Coding for Rule Based

Figure above shows the example of the coding of the Asymetrix ToolBook II used in this study. The coding shows a sub procedure called evaluate. This procedure is used to compare the answer input by the student with the existing answer stored in the system. If the answer is true, (same as the stored answer), a respond message box for a correct answer will appear. If the answer is not true, the respond message box for incorrect answer will appear.

4.1.2 The Student Model

ITS compare the student's performance with the student models in order to determine if the student has mastered the content domain. The student model affects many aspects of the tutor such as topic selection, problem generation and hint selection. It depends on the level of the students. More complicated problems are given to the expert compared to the novice. More hint or help is provided for the lower level students in order to guide them in solving a particular problem. Students will benefit from viewing their own student models [Cook and Kay,1994]. In order to view their own performance, the domain knowledge and exercises are given. The interface allows students to choose either to begin with the domain knowledge section or skip to the exercise section if they have already mastered the topic.

VECITS has the ability to analyze the problem with diagnosis, guidance and rectification. The system which emulate the tutor will predict the student performance and if the actual performance does not meet the prediction, the system must determine the source of deficiency is either due to a misconception or a missing conception. Once the system recognizes the cause of deficiency, it makes a diagnosis and prescribes instructional remedy.

4.2 Features in VECITS

4.2.1 Hypertext Feature.

Nelson (1965) claimed that "Systems of paper have grave limitations for either organizing or presenting ideas. A book is never perfectly suited to the reader; one reader is bored, another confused by the same pages. No system of paper - book or programmed text - can adapt very far to the interests or needs of a particular reader or student."

In contrast to a paper representation, a hypertext representation provides a non-sequential, non-linear method of representing and accessing information. In a hypertext document information is stored as a network of nodes linked together by hotspots. The selection of one of these hotspots, or hotlinks, allows a jump to another page within a document or even to another document.

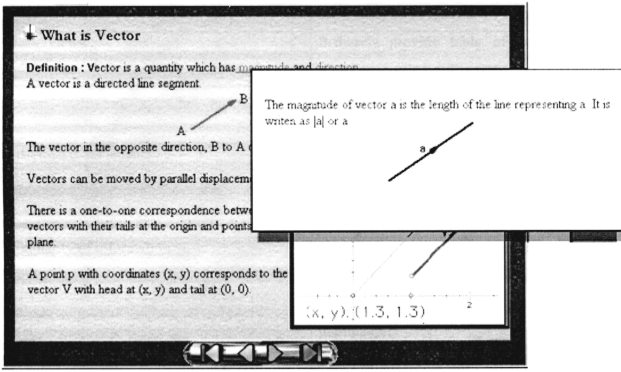


Figure 4.3: Example of Hypertext

VECITS provide a hypertext features to give an additional information to student. Thus, student can click to the hypertext if they think that they did not know what is the meaning of that particular word. Figure 4.3 above shows one of the hypertext provided to student if they did not know what magnitude is. Hypertext provides many advantages as shown in Table 4.1 below [Conklin,1987].

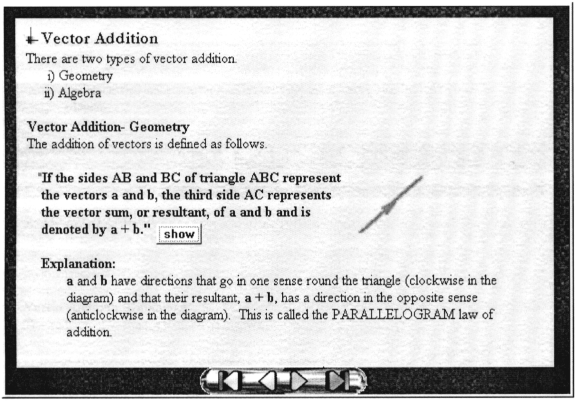
Table 4.1: Advantages of Hypertext

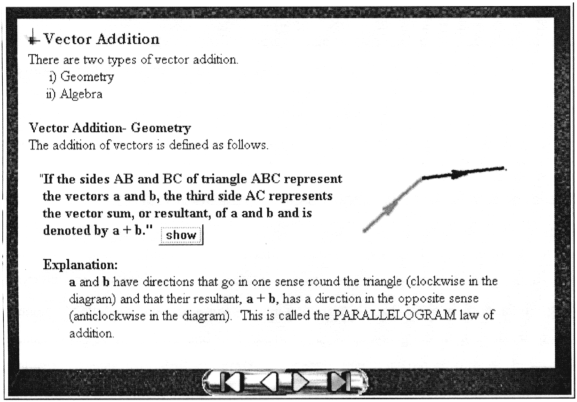
Advantages	**Description**
Ease of tracing references :	Machine support for link tracing means that all references are equally easy to follow forward or backward.

Advantages	**Description**
Information structuring :	Both hierarchical and non-hierarchical organizations can be imposed on unstructured information; even multiple hierarchies can organize the same material.
Global views :	Browsers provide table of contents style views supporting easier restructuring of large and complex documents. Global and local views can be mixed effectively.
Customized documents :	Text segments can be threaded together in many ways allowing the same document to serve multiple functions.
Modularity of information :	Since the same text segment can be referenced from several places an idea can be expressed with less overlap and duplication.
Consistency of information :	References are embedded in their text and if the text is moved, even to another documents, the link information still provides direct access to the reference.
Task stacking :	The user is supported in having several paths of inquiry active and displayed on the screen at the same time. Any given path can be unwound to the original task.
Collaboration :	Several authors can collaborate with the comments about the document being interwoven.

4.2.2 Animation Features.

Animation is simply a series of pictures that are seen one after the other and give the illusion of movement. Animation is quite important to multimedia system because it will attract a student attention to the topic. VECITS provides animation for the direction movement of vector. Using the animation, student will see how to get an addition of vector. It is shown in the figure below.





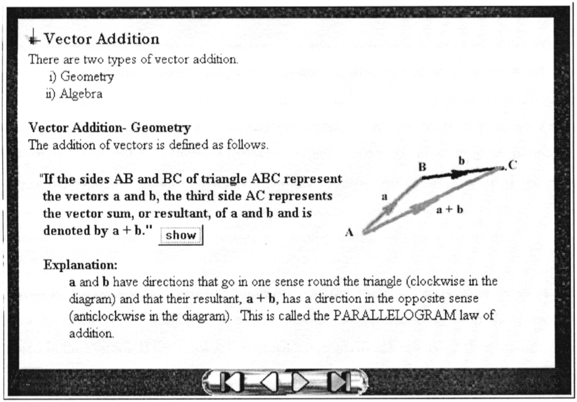


Figure 4.4: Animation for Vector Addition

This screen shows an animation of how we will get a value of vector '**a** + **b**'. When a student push a 'show' button, an arrow will be appear one by one to show the direction of vector and the direction of '**a** + **b**' value.

4.3 Model of Design

As mentioned in 2.2.1, there are 4 types of CAI model, which are drill-and-practice, tutorial, simulation and game software. In this study, VECITS was developed using drill-and practice and tutorial models. Even drill-and-practice technique is one of the traditional CAI models, VECITS will use this techniques as the basic reference concept. Drill-and-practice software is suitable for students who have lower level cognitive skills [C. Floyd Richmond, 1994]. This means that they need help and hints in order to help and guide them solving a particular problem. Besides that, drill-and-practice assumes that the student already has some prior knowledge of the content of the subject and seeking primarily to refine existing skills.

The tutorial model usually assumes that the students have no previous knowledge of the content being taught and attempts to present the material in a logical sequence which fosters understanding [C. Floyd Richmond, 1994]. The model is like a workbook and requires too little interaction between the student and the computer. It may result the software becoming too passive. But VECITS attempts to overcome this problem by presenting animation and two-way interaction between the computer and the student. Animation provided will show the student of how the movement of the direction occurs. It gives student an overview of how the calculation such as addition, subtraction and multiplication of vector are made.

Two-way interaction between the computer and the student means when student input the answer, VECITS will diagnose it by comparing the input answer with the correct answer stored in the system. If the answer is correct, system will acknowledge them that the answer is correct. If the answer is incorrect, it will tell that the answer is incorrect and why it is incorrect. In addition system will tell of how to correct it and gives a related page as reference when student press the 'note' button. So that, this two-way interaction will help student to try to solve the problem until they get the correct answer.

The aim of the system is to help student to review the topic after studying it in class. They have some knowledge of the topics before using the system. They should have studied what is vector, the rules of vector and the procedures in the calculation involving either addition, subtraction or multiplication. Even if they already have the basic theory of vector, the system still provides the prerequisite knowledge of vector in order to give them an overview in case they cannot recall what they have studied. Drill-and-practice model also allows student to answer the question more than once. They can use the system anytime and can attempt the same question if they want to gain the skill in answering that question before attempting the next question. Most of the drill-and-practice software uses the following design [C.Floyd Richmond,1994].

1. The computer selects and presents a problem.
2. The student responds.
3. The computer evaluates the student's response and provides positive or negative feedback based on the student's answer.
4. Steps one through three are repeated until the student is ready to stop.

In this study, the third design was enhanced by providing not only 'correct' or 'incorrect' answer but the system also give a guide or response what they have answered. If the answer given by the student is incorrect, system will tell which part is wrong and how to solve it. This will help students to gain a hint or clue of how to solve the problem. Students must get the correct answer if they want to go to the next step. This is because, the system will not allow user to proceed with the next step before finishing the previous. The system controls the input and provides guidance to student.

4.4 VECITS Feedback Characteristics

In VECITS, only three feedback features from the various features discussed in 2.3 are used. There are Hint format, Help format and Coaching format.

4.4.1 Hint Format

The aim of Hint in VECITS is to help student get a clue of how to solve the particular problem. Hint is provided at each problem and only activated when the student requests. Means, student may press the button provided at each problem, and pop up window containing a clue will appear. There is only one hint for one problem provided. If a problem given needs to be solved in more than one step, hint will give the step by step clue. The example is shown in figure 3.10, a pop-up screen for hint.

4.4.2 Help Format

The Help format allows the student to request assistance when they need assistance or when they have an error. In VECITS, help is labeled as 'Tutor' in button form. The Tutor button is activated when the student requests. Each level has its own tutor button and differs from one another. Means, the contents of the help is depend on the step of the solution.

4.4.3 Coaching Format

This format simulates as human coach and VECITS will coach the student by providing domain knowledge of vector as notes. Besides that, a set of questions is also provided in order to test the understanding of domain knowledge. This format provides a knowledge skill checking of mastery to domain knowledge and also detects if there is any error that occurs during answering the questions.