Name : **Wilhelmina Ngelambai**

Matrix No : **WEK 98145**

Project Title : **First Aid Advisor**

Supervisor : **Cik Norisma**

Moderator : **Prof. Madya Dr. Rodziati**

Abstract

This report is set out to discuss the development of intelligent system in medical domain. This report includes the introduction to the proposed system, the literature review, the methodology, system design, the development and implementation, testing and evaluation. The area of application is medical first aid. The report begins with the objectives and scope of the system. Literature review explores the structure of the intelligent system, the knowledge representation and techniques and also the existed intelligent system in medical domain. Case-based reasoning is a methodology that is proposed for knowledge acquisition for First Aid Advisor system. Visual Prolog is the programming tool for the First Aid Advisor system. The system design shows the data flow of the proposed system. The development is focused on the knowledge engineering process. And testing is to debug the system and this is includes the maintenance. The evaluation is to measure the achievement of the system towards the desired performance.

Acknowledgement

I would like to express my gratitude to all the people who contribute the ideas and valuable information in completing this thesis. Their suggestions and criticisms help me to improve the thesis immensely.

In particular, I want to acknowledge the firm commitment and guidance of my supervisor, Ms. Norisma. Her insight into the field of intelligent system makes it easier to understand the technology when everyone trying to make it difficult. Also, I would like to thank my moderator Professor Madya. Dr. Rodziati, for her suggestions and criticism.

In addition, I would like to express my appreciation to the staff of Emergency Medical Unit of university Malaya Medical Center. Thanks to Dr. Salleh for taking time to provides all the information in the form that is easy to use in knowledge acquisition process. Also, to the medical students who help me in anyway in knowledge acquisition process.

As always, I owe a deep debt of gratitude to my family – mom, dad, my brothers and sisters who have improved the report in the countless way.

wilhelmina

Chapter Organization

Part I : WXES 3181

Chapter 1 introduces the First Aid Advisor system. It explains what is First Aid Advisor and why the system is so valuable. It includes the objectives, scope, limitation and the target user. Later chapter provides the design of the FAA system, while this chapter provides the motivation of the effort.

Chapter 2 is concerns with the literature review. It explores the major characteristics of the intelligent system. It compares the existed system with the proposed system. Also it discusses the knowledge representation and reasoning techniques.

Chapter 3 presents the methodology for the software engineering to be used in developing the FAA system. It reviews the waterfall technique and the case-based reasoning. The programming tools also reviewed in this chapter.

Chapter 4 shows the system design. This is includes the data flow diagram, the operation and user-interface design.

Part II WXES 3182

Chapter 5 brings together the key ideas raised throughout the chapters in Part I (Chapter 1 through 4). It related with the development and implementation of the system.

Chapter 6 covers the various tests that have been carried out through the verification during the development of the system. This is also includes the documentation.

Chapter 7 is the final chapter. This chapter is concerned with the system evaluation and maintenance.

Table Of Content

Tables and Figures

Chapter 1:

INTRODUCTION TO FAA SYSTEM

1.1 Motivation of FAA

What is First Aid? American Red Cross defines first aid as the immediate and temporary care given to a victim of injury or sudden illness until more advanced care can be obtained [13]. Therefore, first aid is an important skill. By performing simple procedures and following certain guidelines, it may be possible to saves lives by giving basic treatment until professional medical help arrives. In an emergency situation, there is no time to seek and ask for help from the people around for treatment to be treated to the victim.

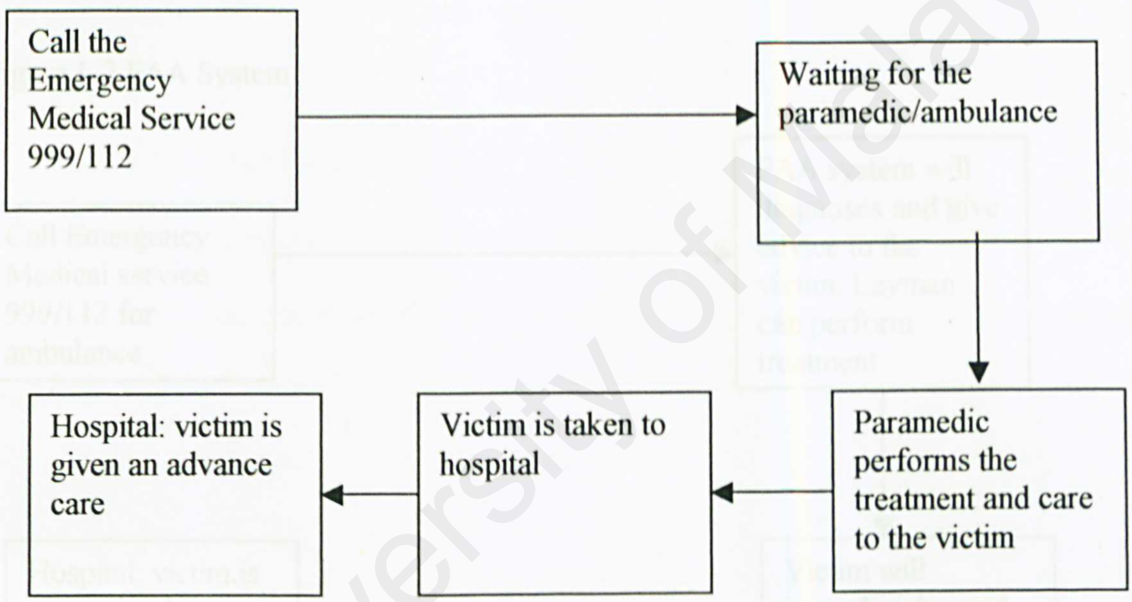
In reality, during the emergency, people will call the emergency medical service's number **999** for ambulance, and **112** (cellular phone's user). The treatment will only be given until the paramedic team arrives at the scene. The paramedic may be takes time to arrive at the scene, this may cause by the flow of the heavy traffic or the because of the distant between the scene from hospital. The traditional procedure illustrated in Figure 1-1. The case can be life-threatening if no immediate treatment is given to the victim.

In order to response to the unexpected emergency situation, lay-man (non-professional, especially referring to medicine) must have a basic knowledge of emergency treatment procedures and guides. Generally, only those who are in medical field can perform this procedure.

With the rapid growth of artificial intelligent technology, and the demand of this emergency situation, First Aid Advisor will be developed. It is a computer program that capturing and applying a medical expert's knowledge to diverse tasks from performing medical diagnosis to giving an advice the medical case.

The FAA system caters for emergency treatment advice, and procedures and guides. This system works as intelligent medical kiosk.

Figure 1-1 Traditional Emergency Procedure

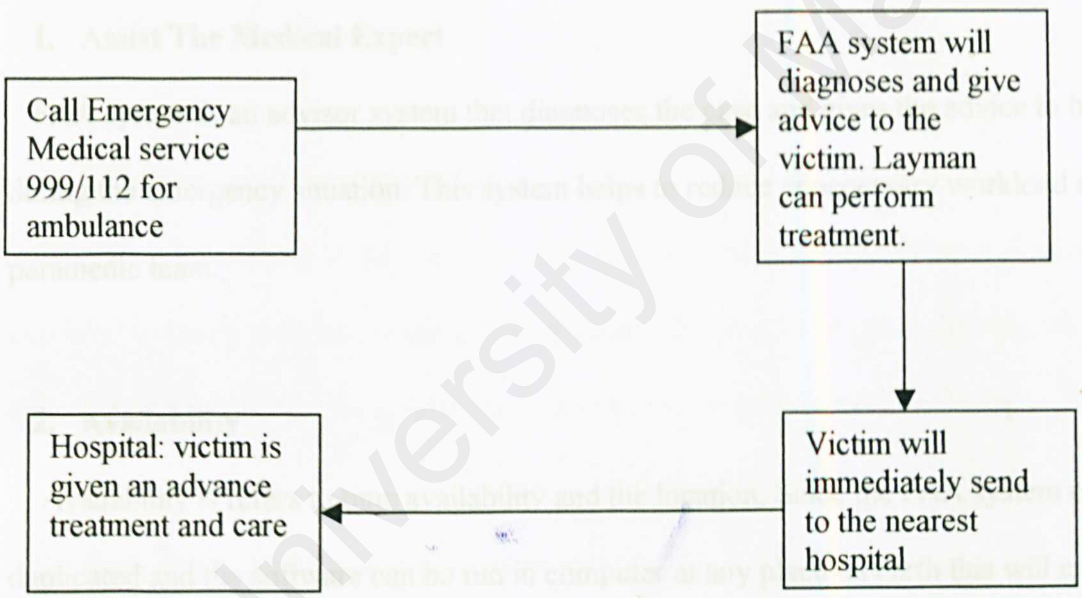


1.2 Introduction To First Aid Advisor (FAA)

The field of medical deals with complex cases and tasks. One of the tasks is to deal with the emergency situation. The normal procedure is, once the emergency department receive the call, the will get the detail of the emergency such as the condition of the victim, the situation of the scene and location. It takes a few minutes to get the detail and to transport to the location. Once the paramedics arrive at the scene, they will check the victim to diagnose the situation, then they give the treatment to the victim. The process is illustrated in figure 1-2.

FAA system aids the users in handling the emergency situation and giving the advice for further treatment and care of the immediate action to the victim. It response to major domestic emergency situation likes wound, burns, broken bones, choking, poison and etc.

Figure 1-2 FAA System



1.3 Objective of FAA

FAA is software package runs on personal computer. While the computer is running the program, it totally dedicated to it. The FAA system has two main objectives, technical objective and general objective.

1.3.1 Technical Objectives

The following are six technical objectives of FAA:

1. **Assist The Medical Expert**

FAA system is an advisor system that diagnoses the case and gives the advice to be treated during the emergency situation. This system helps to reduce unnecessary workload of the paramedic team.

2. **Availability**

Availability is refers to time availability and the location. Since the FAA system can be duplicated and the software can be run in computer at any place on earth this will makes the system is available at anytime 24 hours a day, and 7 days a week. It really helpful especially in outskirt area where is lack of medical services.

3. **Reliability**

This system is reliable because all the knowledge base is draws upon the respected resources from the interview with the Medical Emergency Unit of University Hospital, Kuala

Lumpur, the doctors and the medical students of various universities and literature review from medical books and Internet sources. The knowledge base part gives the description of the cases and diagnosis techniques from various sources. Knowledge base is updated frequently to make sure the new case can be store in the knowledge base and new knowledge can be stored.

4. **Portability**

In addition, FAA system is not tied to one operating system. Since the system is writing in Visual Prolog, the FAA system can run its applications across the MS-Windows 3.x, 95/98/2000,NT and OS/2 platforms.

5. **Efficiency and Response Time**

Efficiency is referring to the response time and the accuracy of the solution. This system is expected to timely response to the problem, means the process to get to the solution is expected to be fast. And the solution provided by the system must be accurate.

6. **Ease of Use and Ease of Maintenance**

FAA will be presented with a display screen in a user-friendly manner. The text will be presented in a natural language. The user interface is made simply, and require user to insert input by clicking checkbox and options buttons. The system is easy to maintain by its programmer because it separate knowledge from control. Any changes to the knowledge or control will not ruin the whole program.

1.3.2 General Objectives

Following are the general objectives of FAA:

1. **Increase the quality of Life**

By using this system, humans no longer completely rely on the paramedic team to give the first aid to the victim. This intelligent system helps the medical expert to make available their knowledge. In a way, this will increase the quality of life.

2. **Support the Growth of Technology**

FAA system is a computerized system, the uses of FAA shows that the users support the growth of artificial intelligent technology. Humans no longer rely on the traditional emergency case's procedures; which is call the emergency numbers and waiting for the rescue team.

1.4 Scope of the System

This system only focuses on Trauma cases, and specifically for home emergency case. Five major cases are burns, bleeding, broken bones, choking and wounds. These cases are more likely occur in day-to-day life.

The target group of FAA is system is the layman and medical user. FAA system will be placed at domestic area such as office building, residential area and etc, accessible by general public. For medical user, the system is designed for medical student as a 'textbook' or reference resource for domestic emergency case.

1.5 Limitation

Medical emergency advice and treatment is a matter of life and death. The following are the limitation of the FAA system:

1. **Cases is Limited**

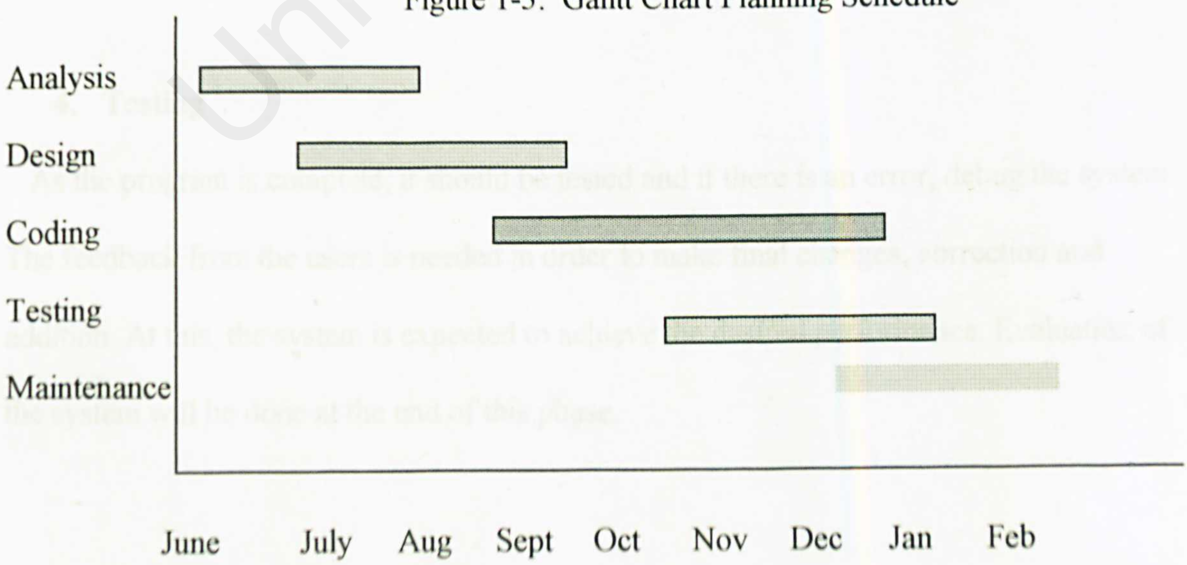
The wrong advice and treatment to the victim may lead to death. Therefore, the cases that are included in the FAA are the most common cases.

2. **Validation of the New Case with Medical Expert**

Since that this system can adapt the new case to the old cases, still, it needs to confirm with the medical expert to certain the treatment, otherwise it may leads to another dangerous medical case.

1.6 Planning Schedule

Figure 1-3: Gantt Chart Planning Schedule



1. **Analysis**

Analysis is the first phase of designing the system. At this phase, the problem will be identified and also will be determine whether it is suitable to build a system. Alternatives will take into consideration. As the problem is found suitable, the development tools will be selected. The literature review is done to determine possibility of building the system base on the previous systems. Knowledge acquisition and documentation also included in this phase.

2. **Design**

By using the knowledge acquired and tools that has been selected, the outline of the system is designed. Data will be organized in a format to understand its knowledge. The knowledge will be converted to IF THEN rules. At the end of this phase, the basic design is completed.

3. **Coding**

Coding is a phase to do the knowledge engineering. The prototype segments of the system will be created. A portion of knowledge will be translated into rules and tested. This task is performed to test the concept before proceed with the entire program.

4. **Testing**

As the program is complete, it should be tested and if there is an error, debug the system. The feedback from the users is needed in order to make final changes, correction and addition. At this, the system is expected to achieve the desired performance. Evaluation of the system will be done at the end of this phase.

5. Maintenance

The system will be updated periodically to make sure the solution in the case library is reliable. If the knowledge is no longer applicable, it will be remove from the knowledge base. The documentation is done at every phase.

1.7 Conclusion

The FAA is a software to diagnose and giving advice for medical first aid cases. The FAA system compilers process knowledge bases into more efficient form by representing the knowledge in appropriate form and specify the inference technique, so that the application can proceed more quickly.

Chapter 2:

LITERATURE REVIEW

2.1 Overview of the Intelligent System

The line dividing intelligent from non-intelligent system keeps shifting as computers and their software become more capable. What is meant by intelligence is broadly characterized by what are human beings are able to do. As human beings, they have an intuitive feel for what that is. Looking at the human brain from the outside, humans have the considerable knowledge of what it can do. From the perspective of 'inside-the-brain,, however human have taken only few step toward understanding how the neuron operates.

Intelligent system was first introduced with the concept of artificial intelligent [4] .A working concept of artificial intelligence is, it is considered as software, which operates on a sophisticated system like a human expert. In other word, intelligent system is a computerized program tools that mimic the humans mind. It uses the knowledge from education and experience to solve the problem. Although a computer cannot have experiences or study and learn humans as the human mind do, but it can apply the knowledge given to it by human experts. It differs from expert system because expert system is very domain specific.

2.2 Human Expert vs. Intelligent System

Human expert is an individual who is very knowledgeable in his field of expertise. Expert represents a valuable resource for any organization. They can offer creative ideas, solve difficult problems, or efficiently perform routine tasks. Their contribution can enhance the productivity of the organization. In this proposed system the human expert is an experienced doctor.

Expert system is a subset of intelligent system [7]. Expert system is a computer program designed to model the problem-solving ability of human expert [2]. The reliance on the knowledge of a human domain expert for the system's problem solving strategies is a major feature of expert system. What is the value of capturing the talent of expert? This will be answered in the table below.

Table 2-1 Comparison of Human expert and Intelligent System

Factor	Human expert	Intelligent System
Time availability	Workday	Always
Geographic	Local	Anywhere availability
Safety	Irreplaceable	Replaceable
Perishable	Yes	No
Performance	Variable	Consistent
Speed	Variable	Consistent
Cost	High	Affordable

Like any machine, intelligent system continues to perform service after the working days of the expert. Since the computer program can be duplicated and it widely disperse in other locations lacking the expert's talent.

Human knowledge is perishable. Through the job transfer, retirement and death, an organization can lost the talent of the expert. Once the expertise captured in a computer program, it permits the continued support in his field.

Also, the intelligent system produces more consistent result than human expert. It this due to humans nature that their decision making process is influenced by many factor such as feeling and the environment. Intelligent system does not support emotion.

For human expert, the speed of problem solving is influenced by many factors. One of the factors is cognitive process. Humans may take time to retrieve the data in the long-term memory whereas, intelligent system is consistent, and in fact some are faster. This is happened because intelligent system will do the matching of the problem to the existed rule in the knowledge base.

Another important factor is, human expert tend to be expensive. But the cost of intelligent system is affordable.

2.3 Types of Problem solve by Intelligent System

Intelligent system simulates a human expert, consultant or advisor. Such a person has extraordinary knowledge, education and experience in a particular domain. Regardless of the application are, given the type of the problem, the expert collects and reason information in similar ways. Intelligent system likewise is designed to accomplish generic tasks on the basis of problem type. Table 2.2 illustrates the types of problem solved intelligent system.

Table 2-2 Types of Problems Solved by Intelligent Systems

Problem-solving Paradigm	**Description**
Control	Governing system behavior to meet specification
Design	Configuring objects under constraint
Diagnosis	Inferring system malfunctions from observation
Instruction	Diagnosing, debugging, and repairing behavior
Interpretation	Inferring situation description from data
Monitoring	Comparing observations to expectation
Planning	Designing actions
Prediction	Inferring likely consequences of given situations
Prescription	Recommending solution to system malfunction
Selection	Identifying best choice from a list of possibilities
Simulation	Modeling the interaction between system components

1. **Control**

Control system adaptively governs the behavior of a given system to meet its specification. For instance, controlling a manufacturing process. The control system obtains data on the system's operation, interpret data to a form an understanding of the state of the system or prediction its future state, determines and executes needed adjustments. It also performs monitoring and interpretation tasks to track system's behavior over time.

2. Design

Design system configures objects under a set of problem constraints. The system usually performs the its tasks following a series of steps, each with its own specific constraints.

3. Diagnosis

Diagnosis system infers system malfunctions or faults from observable information. Normally the diagnosis system has knowledge of possible fault conditions with means to infer whether the fault exists from information on the system observable behavior.

4. **Instruction**

Instruction system guide the education a student in a given topic. They treat the student as a system that must be diagnosed and repaired. Typically it is an interactive topic to form the understanding of the behavior.

5. **Interpretation**

Interpretation system produces an understanding of a situation from available information. Typically this information consists of data from such sources as sensors, instruments, test result, etc.

6. Monitoring

Monitoring system compares observable information on the behavior of a system states that are considered crucial for its operation. Usually the monitoring system interprets signals from sensor and compare the information with known crucial state then later it will establish a sequence tasks to be performed.

7. Planning

Planning system form actions to achieve a given goal under problem constraints. Planning system must have the flexibility to change the series of planned tasks when new problem information. Planning system usually require non-monotonic reasoning.

8. Prediction

Prediction system infers likely consequences from a given situation. This system attempts to predict future events using available information and a model of the problem.

9. Prescription

Prescription system recommends solution to a given system malfunction. This system usually first incorporate a diagnostic task to determine the nature of malfunction.

10. Selection

Selection system identifies the best choice from a list of possibilities. It work from problem specifications defined by the user and attempts to find solution that most closely matches the specifications. It is used for decision support system.

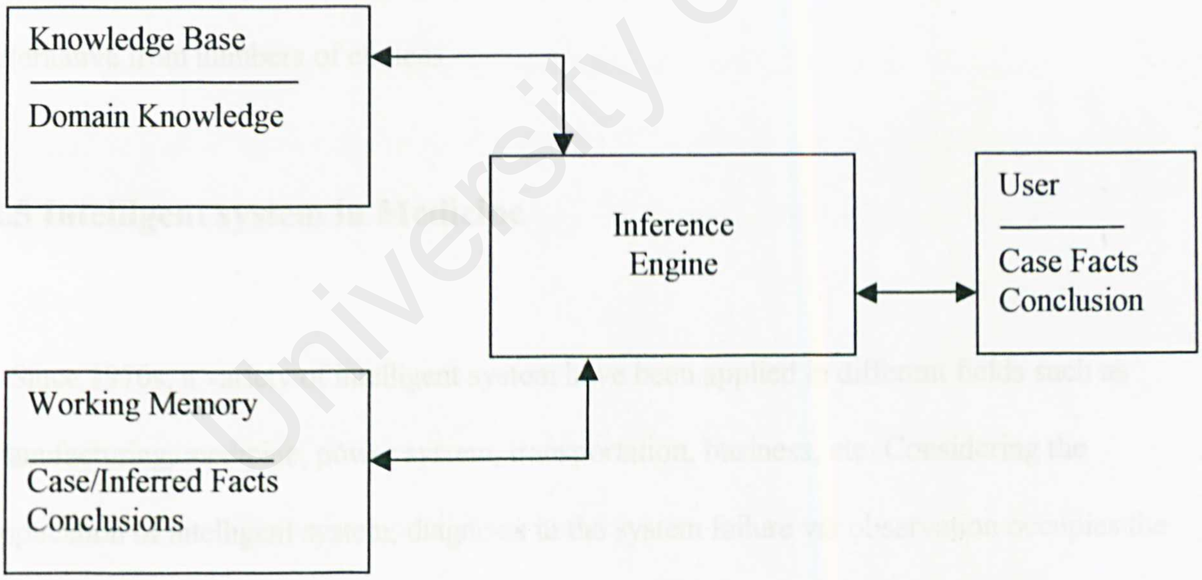
11. Simulation

Simulation systems model a processor or system to permit operational studies under various conditions. They model the various components of the system and their interaction.

2.4 Structure of Intelligent System

Intelligent system consists of three major components: a knowledge base, an inference engine, and user interface. These are illustrated in Figure 2-4. The knowledge base contains all the facts, ideas, relationships, and interactions of narrow domain. The inference engine analyzes the knowledge and draws conclusions from it. The user interface software permits new knowledge to be entered into the knowledge base and implements communication with user.

Figure 2-1 General block diagram of intelligent system



An intelligent system consists of:

a. **Knowledge base** of domain facts and heuristics associated with the problem;

b. **Working Memory** –" a global data-base" – for keeping track of the problem status, the input data for the particular problem, and the relevant history of what has thus far been done.

c. **Inference Engine** is a processor in the system that matches the facts contained in the working memory with domain knowledge contained in the knowledge base, to draw conclusions about the problem.

To use the intelligent system, the user boots up the intelligent software. The system then asks the users a set of questions to gather some initial information about the problem to be solved. The user can key in the information requested or select it from alternatives presented in the menu form. Once the system has the input needs, it proceeds to search for a solution and reach conclusion. The output may recommend the action or the selection of the best alternative from numbers of choices.

2.5 Intelligent system in Medicine

Since 1970s, a variety of intelligent system have been applied in different fields such as manufacturing, medicine, power system, transportation, business, etc. Considering the application of intelligent system, diagnosis in the system failure via observation occupies the important field. Medical diagnosis is the most widely used among these. This review has aimed to investigate the diagnosis system designed for medical field. The existed system are MYCIN, 5GL-Doctor, CASEY and First Aid Parasol.

2.5.1 MYCIN

MYCIN [2] is probably the best known of all the system in medical diagnosis. Edward Shortliffe of Stanford University designs the system in the mid-1970s. This system build to aid physicians in diagnosing and treating patients with infectious blood disease caused by *bacteremia* (bacteria in the blood) and *meningitis* (bacteria disease that causes inflammation of the membrane surrounding the brain and spinal cord) and cystitis. To verify the presence and identify of *bacteremia*, blood samples are taken and tests are conducted. Some tests may take 2-3 day to provide the accurate result. Due to the limited information, physicians need to seek advice from the expert on the problem. In order to solve this problem, MYCIN is built.

Here's how MYCIN helps a doctor identify patients with bacteremia, meningitis and cystitis. The physician sits down at the computer and enters the patient's age and medical history, result of laboratory tests and any additional relevant information. With this information plus its knowledge of the domain, MYCIN attempts to diagnose the cause of the infection. Then it recommends the type of drug and the dosage that might be used to cure the infection.

Major Features of MYCIN

1. **Utilizes a Backward-Chaining System**

MYCIN has about 500 rules and works in backward chaining fashion to identify the nature of infection. [2] It can also identify the suspected invading organisms in their classes. The system employs a backward chaining technique when forming a suggested therapeutic remedy. To draw the conclusion of the rule, MYCIN works backward through the rules that support each premise, searching for conforming evidence.

2. **Separates Knowledge From Control**

Separates knowledge from control is a trademark of the expert system. By maintaining the knowledge separate from its control, the MYCIN development team found that they could easily modify the system's knowledge. To add or modify the existed knowledge, changes only require at the knowledge base. This is possible because rules are independent pieces of knowledge that are used on an as-needed basis. Therefore, the addition or deletion of the rule does not require changes to other rules in the system.

3. **Incorporates Meta-Rule**

MYCIN employs meta-rule to redirect its search. The following is an example meta-rule from MYCIN:

METARULE001[4]	
IF	1) the culture was not obtained from a sterile source, and
	2) there are rules which mention in there premise a previous organism, which may be the same as the current organism
THEN	it is definite(1.0) that each of them is not going to be useful.

It states that when trying to identify organisms from a non-sterile site, rule which base their identification on other organisms found at the site are not going to be very useful.

4. **Employs Inexact Reasoning**

Each rule has an associated 'certainty factor' (CF) in the range [-1,+1]. This reflects a degree of confidence in the conclusion drawn, given the evidence of premises. A positive CF indicates a degree of belief in the answer [2]. –1 represents definitely false and +1 definitely true. For example, if the physician believes some evidence may be true, then he can enter a value of 0.7. Following is the rule for certainty factor

IF Evidence **THEN** Conclusion CF 0.7

In this fashion, given evidence is known, then the conclusion is asserted with a degree of belief reflected in the rule's CF value.

5. **Remember Prior Session**

MYCIN remembers information from a prior session concerning a given patient. [6]This includes data provided by the patient, conclusion drawn by MYCIN, and finding such as the identities of the suspected invading organisms and the therapeutic recommendations.

6. **Accommodates the User**

To perform the function in the acceptable manner, MYCIN is designed to be easy to use and present itself in a manner that is natural to physician. [6]. MYCIN interacts with the physician in English. It also support spelling checker. MYCIN gives explanations for questions like *why*, and *how*.

MYCIN'S Problem Solving Approach

1. **Diagnosis**

MYCIN is knowledgeable about being possible infections and infecting organisms.[2] The first part of MYCIN is diagnosis phase. During the diagnosis phase, first, MYCIN will asks a set of questions to obtain general background information on the patient including the patient's age, sex and symptoms.

MYCIN then asks for available results from laboratory tests. With the answer given by the patient, MYCIN directs the line of questioning toward the suspected infection or infecting organisms. This system will identify the nature of infection. Later it will identify what are the organisms are causing the infection.

The main goal of diagnosis phase is to get the evidence of the infection. This goal will be used for the next phase, Prescription phase.

2. **Prescription**

Prescription phase is a task to prescribe a set of drugs to give to the patient that will eliminate each organism identified during diagnosing session.[2] It is not an easy task for MYCIN system prescribe the drugs because the combination of various drugs can create a toxic mixture.

MYCIN and FAA

In general, MYCIN is a system that motivated the development of FAA system. The FAA problem solving approach is almost similar to MYCIN. MYCIN is doing the diagnosis and giving prescription whereas FAA diagnose case and give advice to the user. MYCIN utilizes backward chaining to prove the goal but the FAA utilizes forward chaining to solve the problem. FAA system will reason the problem by using case-base reasoning model. In MYCIN it support the spelling checker, in the FAA system it does not support the checker because user only input the information by clicking the check box. Both systems are separate knowledge and control, has the meta-rules, and accommodate user by support the explanation facility.

2.5.2 CASEY

CASEY (Koton 1988a, 1988b, 1989) is a case-based diagnostician. It is used in medical field. CASEY is built base on The Heart Failure Program (Long et al. 1987)[6], a model-based diagnostic program that diagnoses heart failure with unprecedented accuracy. To perform its function, CASEY will first get the general description of its new patients. The description includes patient's normal signs and presenting signs and symptoms. As the output CASEY will gives the explanation of the causal of disorder. The causal explanation connects together symptoms and internal states.

CASEY diagnoses patients by applying model-based matching and adaptation heuristics to the cases it has available. In its knowledge base has a case library of approximately twenty-five cases, all which are diagnosed by the Heart Failure Program [2]. CASEY's model-based matching and adaptation heuristic are domain-independent and are as accurate as the domain model they are applied to.

Major Features of CASEY

1. Matching and Ranking Cases

The ability to distinguish which of several partially matching cases have the potential to be more useful than others is the key of case-based reasoning work. CASEY will search the case library for partially matching cases, when search processes ask matching functions to compute the match along certain dimension represented as index. Based on this series of dimensional matches, search functions collect a set of cases that partially match the situation.

After this set has been collected, a more comprehensive evaluation of degree of match is done, this taking into account the importance of match along each dimension. This process is called ranking. Ranking procedures choose those cases that can best address the reasoner's purpose.

2. Transformation

CASEY uses model-guided repair for diagnosis task: to adapt the causal explanation of a previous patient's heart ailment to a specific of a new situation [6]. CASEY's evidence heuristics evaluate the differences between an old and new problem to determine if two cases could be match. CASEY has repair heuristics associated with each evidences heuristic that know how to carry out the repair that is necessary to make an old solution to fit a new solution. In a way it transform it knowledge of old cases to create a new case.

Problem Solving Approach

1. Diagnosis

CASEY is a simple program to diagnose heart problems by adapting the diagnoses of previous heart patients to new patients. By using a case-based approach, CASEY can explain phenomenon by remembering a similar phenomenon, borrowing its explanation, and adapting it to fit. In diagnosis, a problem solver is given a set of symptoms and asked to explain them.

CASEY and FAA

CASEY is an intelligent system that diagnosis for heart disorders. It is a case-based reasoning model, it matches the problem to the case in its case library. The reasoning model in CASEY will be adapted to the FAA system. The knowledge base in FAA will be kept in the case library. The FAA system will be implemented uses case-based reasoning model. CASEY is determined the causes of the heart failure and it is very domain specific to the heart problem but, FAA is cater for several major trauma cases. CASEY is differs from FAA because it only giving the causes of heart disorder, but FAA is diagnose and giving the advice to the emergency case. The case in the case library contains the information about the situation, and the result using the solution.

2.5.3. 5GL-Doctor

5GL-Doctor[10] stands for Fifth Generation Doctor. It is multi-functional, medical diagnosis software that matches the symptoms, or signs and test result with know health disorder. Lisa Development Pty. Limited of Australia develops the system. The reviewed edition is a standard edition, which is available in the Internet at www://members.ozemail.com.au/ lisadev/sftdoc.htm.

5GL-Doctor is medical diagnosis assistance software for windows. It comes in dual editions: Personal Edition for non-medical user and Medical Edition medical professional. Personal Edition is self-diagnosis software for window. It has about 4000 long symptoms and sign lists. To use the software, the user only has to choose whether to click the displayed option or use plain English for interaction. The system includes the information about the diseases, drugs, medical procedures and medical test. Also, it provides the explanation facility. This system caters for home user.

Medical Edition is caters for medical professional. Since the medical professionals are not going to miss clear symptoms and signs of the ordinary disease, 5GL-Doctor will help to diagnoses the rare disease or disorder. This edition includes a big number of rare disorders. It also includes lots of case studies, for instance, a presenting symptom such as 'cough' in elderly person could signal anything from heart problems to the common cold. In order word, 5GL-Doctor works as a reference source for the medical professionals.

Features of 5GL-Doctor

The following are the specific features of 5GL-Doctor:

- Rapid matching of symptom, signs or others indicators with about 3000-5000 known medical disease, conditions and case studies

- Inquiry supports both common sense and medical vocabularies
- Approximately 6000 predefined symptoms, signs, and test results- select from the list and then click on inquiry (added entries automatically become part of the symptoms selection lists).
- Six different analysis/diagnosis functions
- Short lists of disorders are hierarchically produced. The most likely outcomes are listed first.
- Multifunctional-designed to meet the needs of diagnosis, and medical information storage.
- Information is derived from reliable medical sources (Harrison's, Merck's and WHO etc)
- The editions of 5GL-Doctor are able to be updated, images can be concluded, flow charts can be defined.
- A Help feature is built into the software to provide quick explanations of terminology.
- The medical procedures 'knowledge pool' suggests tests, which may be performed to conform the diagnosis.
- The medical algorithm compiler facilitates the entry and execution of the algorithms. Once they are executed, further questions are asked and points allocated. The results are interpreted according to the choice criteria.
- Sections on drug (about 800 generic names); tests and procedures (about 400, general and specific information); dictionary
- American/International English or Australian/English spelling
- Easy conversion of units

Problem Solving Approach

To perform its function, 5GL-Doctor will diagnose the patient, later it will give the prescription of the drug.

1. **Diagnosis**

5GL-Doctor uses Window's based interaction. Therefore, the diagnosis will start by clicking the display option. First during the installation process, the user may specify the language preferred, for instance International English.

Using the input from keyboard and mouse, the system will match the symptoms and signs with it database. It will go through some procedures in order to identify the disease. Once the disease is found, the diagnosis part is completed. Next the system will give the prescription to the patient.

2. **Prescription**

Since the early diagnosis is focus, the information of each disorder is readily available in the database. The prescription of the disorder will be displayed in a window. The prescription part is the detail of the entry, it includes the description the disorder, the typical of disorder and option for the drug prescription, together with its picture.

Sample Session of 5GL-Doctor

The following are the snapshot of the session in 5GL-Doctor.

Figure 2-2: Diagnosis in 5GL-Doctor

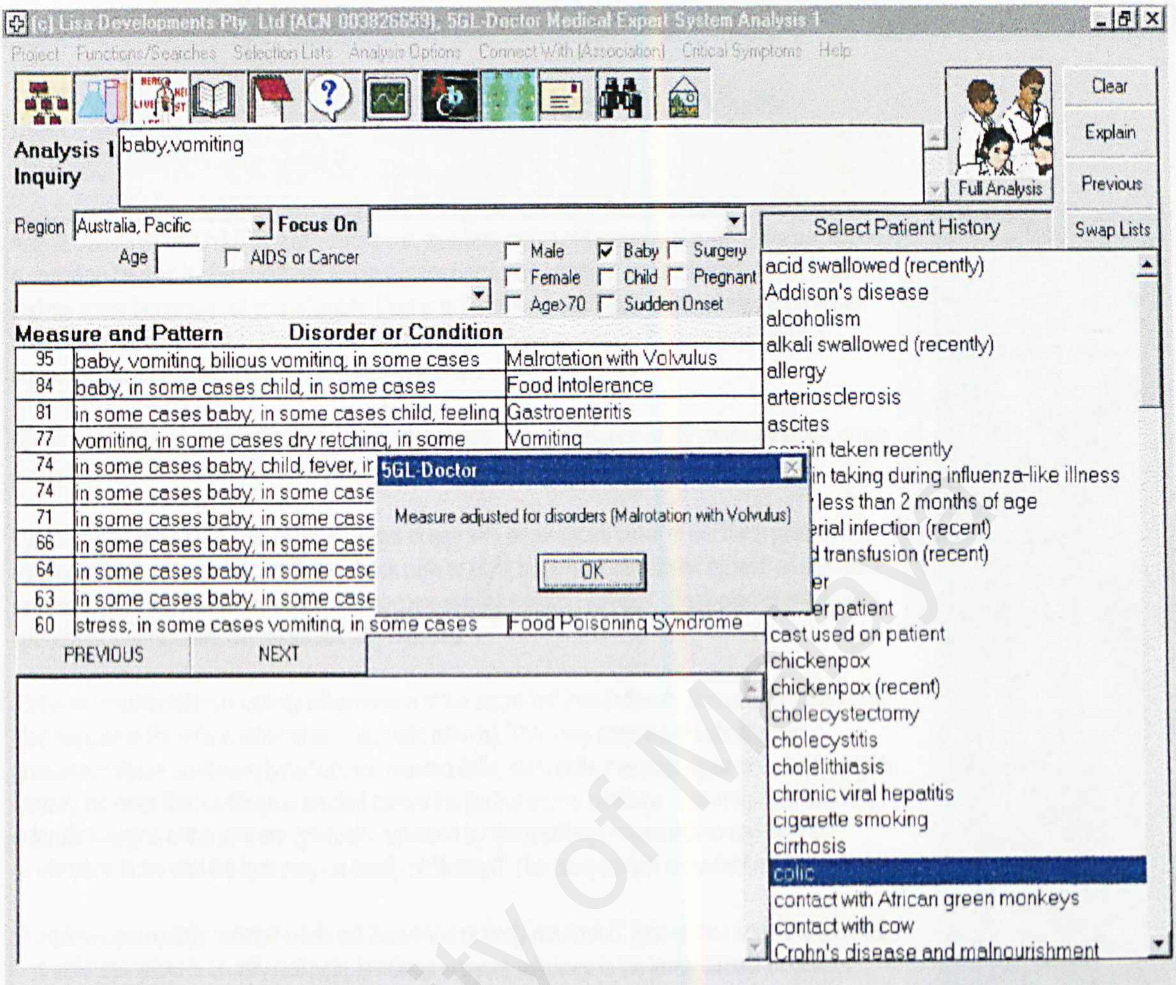
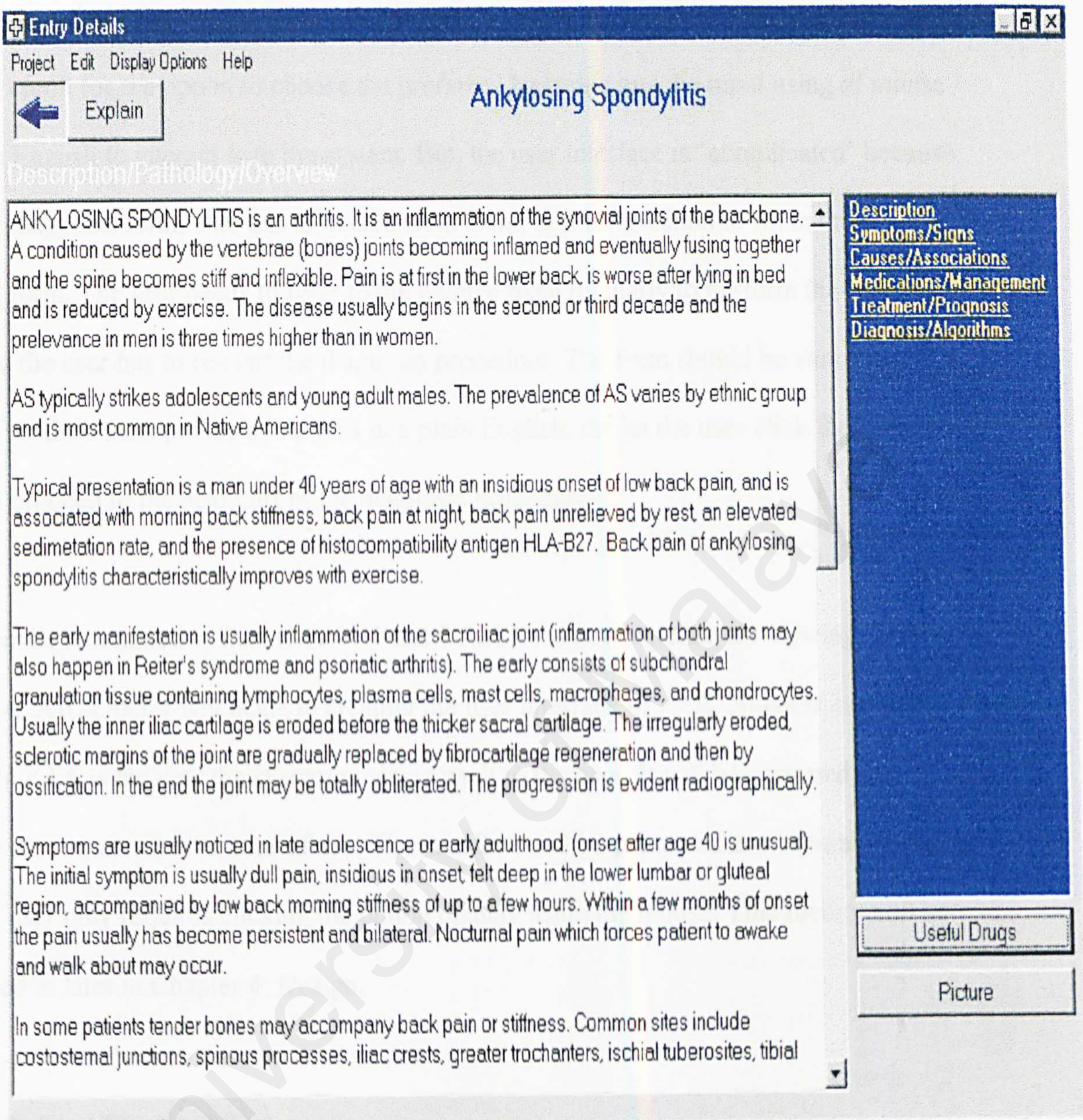


Figure 2-3: Conclusion screen of 5GL-Doctor



The snapshots of the session show the partial session of the diagnosis and prescription in 5GL-Doctor software. The interaction is user input the situation by clicking the option and the user could ask the system give explanation about the symptoms and also provides pictures of the drugs.

5GL Doctor and FAA

5Gl-Doctor is medical software that diagnoses the medical disorder. In term of ease of use, give credit for the option to choose the preferred language and the input using of mouse and plain English to interact with the system. But, the user interface is 'complicated' because to fill the diagnosis form, user has to decide what is the first information to fill in the provided form. The user needs to refer to the manual from regularly to perform the right step otherwise the user has to re-start the diagnosis procedure. The form should be simple. For instance, let the user input the symptoms in a plain English, the let the user click the 'Next' button to proceed to the next part pf the diagnosis procedure.

Since the mentioned system (MYCIN and CASEY) is not available for viewing the user interface, 5GL-Doctor will helps in creating the user interface of FAA. Another important aspect in FAA is the way the system presents itself to the user. It will be designed to achieve the requirement of ease of use. The interface will be presented in a natural language and the user only requires clicking the option display using the mouse. This facility will be discussed in later in Chapter 4: Design.

2.5.4 Existed First Aid System:

First Aid with PARASOL EMT

There are a lot of existed First Aid systems available. But all these system are non-intelligent system. All the existed systems are either online system. For the sake of discussion, one medical information system has been chosen. It available at http://www.parasolemt.com.au/ . Figure 2-4 show the fist page of the First Aid with PARASOL EMT [1]

Figure 2-4 First Aid with PARASOL EMT [11].

First Aid & Emergency Medical Training

- CPR & Basic Life Support
- Senior First Aid
- Advanced First Aid
- Remote Area First Aid
- Oxygen & Defibrillation

Course Dates **Updated** :

- Canberra, ACT
- Sydney, NSW
- NSW South Coast
- Richmond, NSW

Mackay, QLD

First Aid Kits & Supplies

- Personal, Domestic & Commercial Kits
- Soft Packs & Hard Cases
- Emergency Medical Equipment
- Authorised Laerdal™ Distributor

Laerdal

Occupational Health & Safety Solutions

- OH&S Awareness Training
- OH&S Workplace Representative Training
- OH&S Workplace Health & Safety Officer Training
- Confined Space Training
- Fire Extinguisher/Fire Warden Training

OH&S inspections & audits

On the first Aid and Emergency Medical Training, it will provides the information and guides about the CPR & Basic Life Support, Senior First Aid, Advanced First Aid, Remote Area First Aid and Oxygen Defibrillation.

For First Aid Kits and Supplies, it will describe the Personal Domestic & Commercial Kits, Soft Packs and Hard Cases, Emergency Medical equipment and Authorised Laerdal™ Distributor.

Occupational Health and Safety Solutions will provide the information about the specific training. The training are comprises of OH &S (occupational health and safety) Awareness Training. OH& S Workplace Representative Training, OH& S Workplace Health & Safety Officer Trading, Confined Space Training, Fire Extinguisher/ Fire Warden training, and OH& S Inspection and Audit. Once the user clicks the subtitle it will link to the specific page .

This page also provides information of its company profiles, the regional information. Give credits to the almost complete knowledge in this system. But, this system is only an information base First Aid system, the knowledge is static. The only base will only be expanded if the content manager and the programmer update the website. The down side of this information system, it can only be accessed only if the Internet service is available. It is not practical for the area where the internet service is not available or poor internet service especially in a rural area.

By reviewing and analyzing the pros and cons of the existed system, FAA system will be developed to meet the demand of technology in handling the emergency situation.

For further information, the existed information system in first aid can be obtained at:

1. http://www.redcross.org/ca/svc/medprot.html# INTRODUCTION
2. http://vnh.org/standardFirstAid/Chapter1.html
3. http://www.redcresent.org.my/campaingns/firstaider.php3

2.6 Types of Knowledge

Knowledge is an abstract term that attempts to capture individual's understanding of a given subject [2]. To define knowledge in a more practical perspective, there are five types of knowledge: procedural knowledge, declarative knowledge, meta-knowledge, heuristic knowledge and structural knowledge.

a) Procedural knowledge

Procedural Knowledge is often referred to as knowing how the problem is solved. The typical type of procedural knowledge is rules, strategies, agendas and procedures. For example: How to bake a cake?

b) Declarative knowledge

Declarative knowledge refers to knowing that something is true or false. Declaratives knowledge associates with concept, objects and fact. Example: Eagle is a bird.

c) **Meta-Knowledge**

Meta-knowledge is knowledge about knowledge. The knowledge has layer. The knowledge is about the other types of knowledge and how to use them. Example: Eagle is a bird if eagle can fly. Eagle can fly if it has wing. Eagle has wing.

d) **Heuristic Knowledge**

Heuristic knowledge describes a rule-of-thumb that guides the reasoning process. It is often called shallow knowledge [2]. It is empirical and represents the knowledge compiled by an expert through the experience of solving past problem. Example: In order to reach the KLCC, just walk straight to the twin tower building.

e) **Structural Knowledge**

Structural knowledge describes knowledge structures. It associates with the rule sets, concept relationships and concept to object relationships. Normally the knowledge is represented in a' tree' form.

2.7 Knowledge Representation Techniques

Most of the artificial intelligent systems are made up of two basic parts; a knowledge base and inference mechanism. But what is knowledge? Knowledge is an abstract term that attempt to capture an individuals' understanding of a given subject. in the world of intelligent system we call it as domain-specific knowledge.

What is knowledge representation? Knowledge representation is defines as the method used to encode knowledge in the intelligent system's knowledge base. Cognitive psychologists have formed a number of theories to explain how humans solve problem. This work uncovered the types of knowledge humans commonly use, how they mentally organize this knowledge, and how they use it efficiently to solve the problem. Researchers of artificial intelligence have used the results of these studies to develop techniques to best represent these different types of knowledge.

Three knowledge representation techniques will be used in the FAA system. There are rules, predicate logic and certainty factor.

Rules

Rule is defines as a knowledge structure that relates some known information to other information that can be concluded of inferred to be known [2]. Rule is associates with procedural knowledge. The given conclusion or goal is associates with some action. In this sense, a rule describes how to solve problem.

The rule's structure logically connects premises and conclusion. The premises (antecedents) contain 'IF' part, and conclusion (consequents) contain 'THEN' part. Sometimes it comes with conjunction 'AND' to combine the statement. Also, the rule can contain an 'ELSE' statement, that is inferred to be TRUE or FALSE. The following are the example of knowledge base that utilizes rule:

IF	Patient is pale
AND	Coughing during the night
OR	Severe asthma attack
THEN	Patient has asthma
ELSE	It is not the symptom of asthma

This type of rule will be used in the database to categorize the emergency case. There are few types of rule will be used in the FAA system, there rules are relationship rule, recommendation rule, directive rule, strategy rule and heuristic rule.

These five rules can be combined through the meta-rule. By using the meta-rule it will enhance the system's efficiency by directing its reasoning into the promising area.

The principal advantage of using rule is it design that is represents a natural approach to solving complex problem. Generally humans tend to dissect a complex problem into a smaller, manageable sub-problem. Also, it eases the system's development and maintenance. During the maintenance, the knowledge engineer can focus on one module at a time. Technical advantage of this approach is, it allows the knowledge engineer to integrate different knowledge representation techniques and inference strategies into system.

Predicate Logic

Perhaps the oldest form of knowledge representation is logic, the scientific studied the process of reasoning and system of rules and procedures that aid in the reasoning process.

The ones most often linked with intelligent system have been propositional logic and predicate calculus. Both techniques use symbols to represent knowledge and operators applied to the symbols to produce logical reasoning.

A proposition is nothing more than a statement that is either true or false, a premise that can be used to derive new propositions or inferences. Rules are to determine the truth (T) or falsity (F) of new proposition. To form more complex premises can be combined using logical connectiveness. This connective operators are designated as AND, OR, NOT, IMPLIES and Equivalent.

Certainty Factor

Uncertainties arise constantly in practical situation. Real life problem solving demands an acceptance of uncertainty in order to minimize the difficulties it poses. In the intelligent system, certainty factor (CF) is used to help deal with uncertain situations.

A certain factor is simply a number that indicates how confident the system to a particular fact is true. Based on experience or evidence, system able to choose a certainty factor that indicates the level of truth. A typical range of certainty factor might be –1 .0 to +1.0, where –1.0 indicates certainty the factor is false and +1.0 indicates complete certainty that the information is true. Zero (0), indicates the certainty is unknown.

A production rule using the certainty factor might look like this:

IF	the patient has wound,

THEN	the patient has incision (certainty 0.3).

The certainty, or confidence only 0.3 because the symptoms may cause to laceration. Therefore, in the proposed system certainty factor may be used to show the level of confidence to the respective solution.

2.8 Inference Strategies

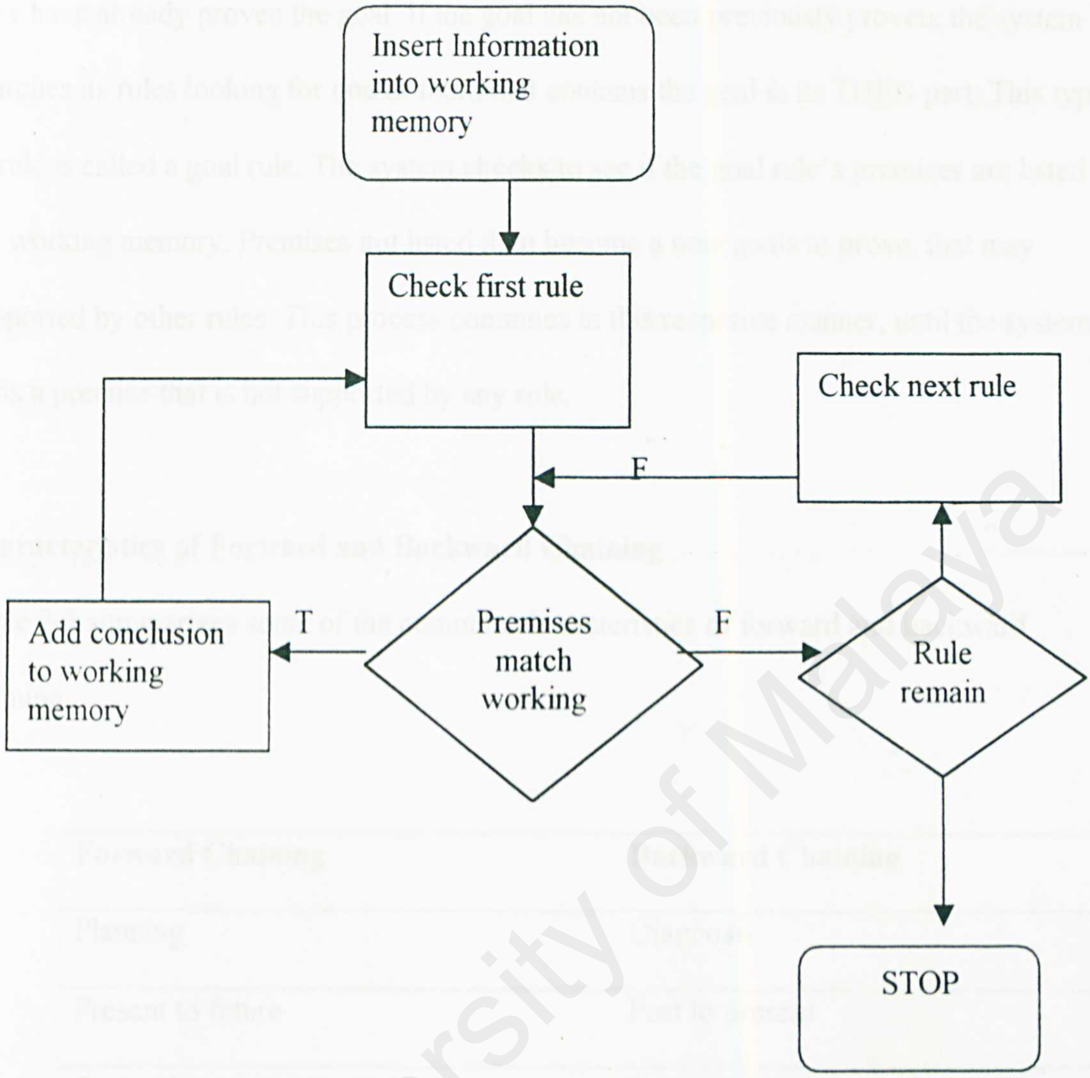
The object of a search procedure is to discover a path through a problem space from an initial configuration to a goal state. A group of multiple inferences that connect a problem with its solution is called a chain [3]. In PROLOG, searches only begins with a goal state, there are actually two directions in which such a search could proceed:

- Forward Chaining
- Backward Chaining

Forward Chaining

To achieve the goal, the solution process for some problem naturally begins by collecting information. The collected information is then reasoned with to infer logical conclusion. This is data driven searching technique. It is also called **forward chaining.** Forward chaining is defines as an inference strategy that begins with a set of known facts, derives new facts using rules whose premises match the known facts, and continues this process until a goal state is reach or until no further rules have premises that match the known or derived facts [2]. Figures 2-5 shows the processes of forward chaining.

Figure 2-5 Forward Chaining inference process



Backward Chaining

Backward Chaining is called top-down reasoning because it reasons from the higher-level constructs such as hypotheses down to the lower-level facts, which may support the hypotheses [3].

A backward-chaining system begins with a goal. First, it checks the working memory to see if the goal has been previously added. If this step is necessary since another knowledge base have already proven the goal. If the goal has not been previously proven, the system searches its rules looking for one or more that contains the goal in its THEN part. This type of rule is called a goal rule. The system checks to see if the goal rule's premises are listed in the working memory. Premises not listed then become a new goals to prove, that may supported by other rules. This process continues in this recursive manner, until the system finds a premise that is not supported by any rule.

Characteristics of Forward and Backward Chaining

Table 2-3 summarizes some of the common characteristics of forward and backward chaining.

Forward Chaining	**Backward Chaining**
Planning	Diagnosis
Present to future	Past to present
Antecedent to consequent	Consequent to antecedent
Data driven, bottom up reasoning	Goal driven, top down reasoning
Work forward to find solutions follow from the fact	Work backward to find facts that support the hypothesis
Breadth-first search facilitated	Depth-first search facilitated
Antecedents determine search	Consequent determine search
Explanation not facilitated	Explanation facilitated

2.9 Conclusion

This chapter reviewed the design and architecture of the intelligent system. It shows how the intelligent system works. Also in this chapter it analyses three existed medical intelligent systems and one non-intelligent system. And the non-intelligent system is one of the motivations in developing the intelligent FAA system. In other words, FAA is developed to replace the information system to intelligent system. The review of the knowledge representation and reasoning techniques will helps in managing the knowledge base and database in inference engine.

Chapter 3:

METHODOLOGY

3.1 The FAA System Life Cycle (Waterfall model)

One of the key methods of software engineering is the life circle. This software circle is the period of time that starts with the initial concept of the software and ends with retirement from use. Rather than thinking of development and maintenance separately, the life circle concept provides a continuity that connects all stages. Planning for maintenance and evaluation early in the life cycle reduces the cost stages later. FAA system utilizes the Waterfall model for software engineering process.

Waterfall Model

There are numbers of life cycle models have been developed for conventional software. The models are Code-and-Fix Model, Incremental Model, Spiral Model and Waterfall Model. For the sake of discussion, Waterfall Model is discussed because this model is use by FAA system life cycle. The classic waterfall model was the original life cycle model [3] and is illustrated in Figure 3-1.

In the waterfall model, each stage ends with a verification and validation (V & V) activity to minimize any problem in that stage. The arrows go back and forth only one stage at a time. This represent the iterative development between two adjacent stages in order to minimize the cost compared to the higher cost of iterating development over several stages.

The life cycle is actually a meta-methodology [3]. This is because it determines the order and duration that the common software methods are applied. The common software development methodologies are:

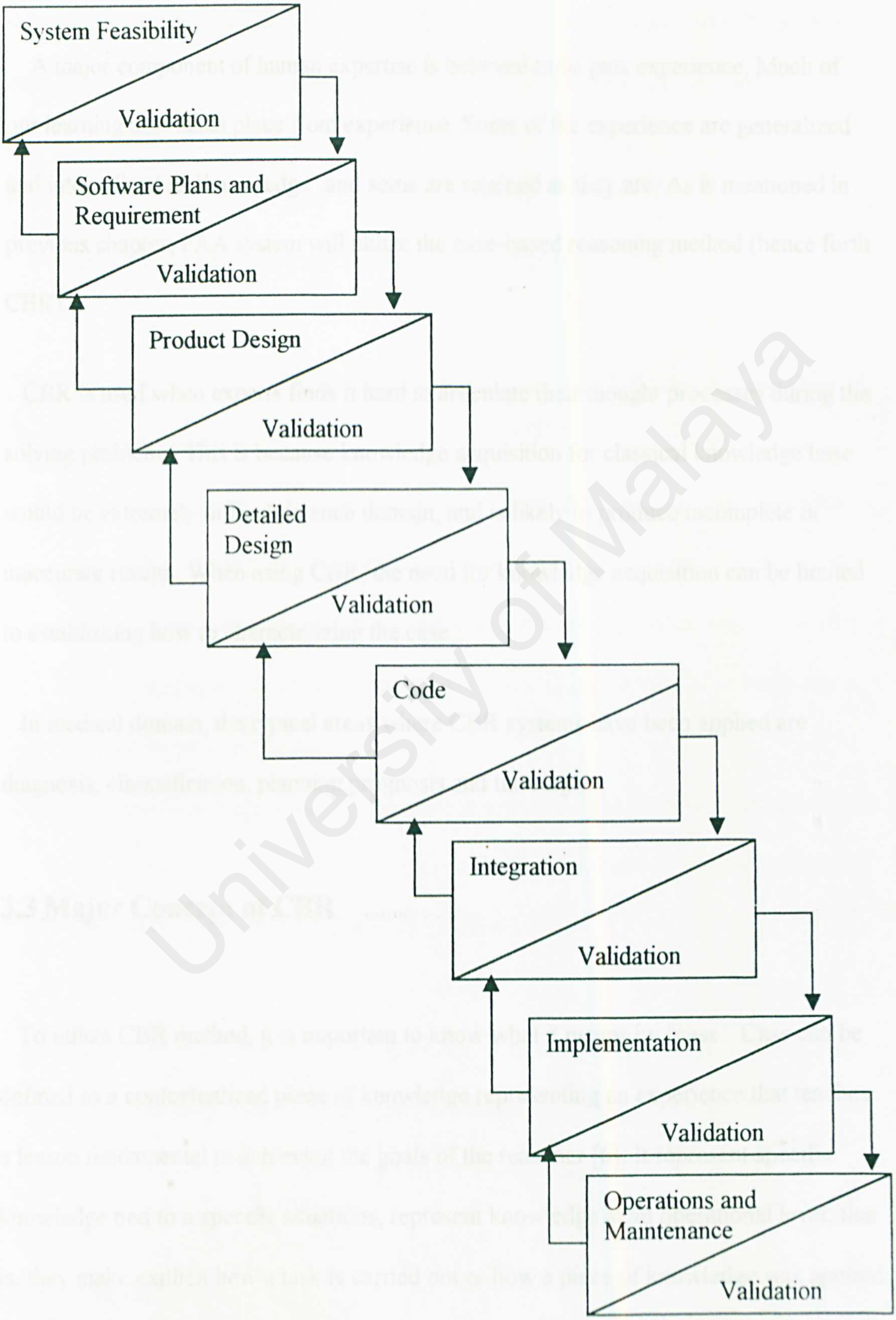
1. Specific methods to accomplish a stage

- Planning
- Requirement
- Knowledge acquisition
- Testing

2. Representation stage

- Documentation
- Code
- Diagrams

Figure 3-1 Waterfall Model



3.2 Introduction to Case-Based Reasoning

A major component of human expertise is believed to be past experience. Much of our learning also takes place from experience. Some of the experience are generalized and internalized as 'knowledge' and some are retained as they are. As is mentioned in previous chapter, FAA system will utilize the case-based reasoning method (hence forth CBR).

CBR is used when experts finds it hard to articulate their thought processes during the solving problems. This is because knowledge acquisition for classical knowledge base would be extremely difficult in such domain, and is likely to produce incomplete or inaccurate results. When using CBR, the need for knowledge acquisition can be limited to establishing how to characterizing the case.

In medical domain, the typical areas where CBR systems have been applied are diagnosis, classsification, planning,prognosis and tutoring.

3.3 Major Concept of CBR

To utilize CBR method, it is important to know what it means by 'case'. Case can be defined as a contextualized piece of knowledge representing an experience that teaches a lesson fundamental to achieving the goals of the reasoner [6]. It represent specific knowledge tied to a specific situations, represent knowledge at an operational level; that is, they make explicit how a task is carried out or how a piece of knowledge was applied or what a particular strategies for accomplishing a goal were used.

The knowledge base in a CBR system is the set of *cases* (ie, the past experiences). When a problem is to be solved, the case library is searched for cases which *match* the current problem. The solution proposed in that case is examined, and applied to the current problem with suitable *modifications*

The following are the summary of the principle of cases

- A case represent specific knowledge tied to a context. It records knowledge at an operational level.
- Cases can come in many shapes and sizes, covering large or small time slices, associating solutions with problems, outcomes with situations or both.
- A case records experiences that are different from what is expected. Not all differences are important to record, however. Cases worthy of recording as cases teach a useful lesson.
- Useful lesson are those that have the potential to help a reasoner achieve a goal or set of goals more easily in the future or that warn about the possibility of a failure or point out an unforeseen problem.

CBR suggests a model of reasoning that incorporates problem solving, understanding, and learning and integrates all the memory processes. It also suggests that the quality of a case-based reasoner's reasoning depends on five things [6]:

- The experiences it has.

- Its ability to understand new situations in term of those old experiences.
- Its adeptness at adaptation.
- Its ability to understand at evaluation and repair.
- Its ability to integrate new experiences into its memory appropriately.

3.4 Processes in CBR

A CBR tool should support the four main processes of CBR: retrieval, reuse, revision and retention. All case-based reasoning methods have in common the following process:

- Retrieve the most similar case (or cases) comparing the case to the library of past cases;
- Reuse the retrieved case to try to solve the current problem;
- Revise and adapt the proposed solution if necessary;
- Retain the final solution as part of a new case.

1. Retrieving Case

Retrieving a case **starts with a (possibly partial) problem description and ends when a best matching case has been found. The subtasks involve:**

- Identifying a set of relevant problem descriptors;
- Matching the case and returning a set of sufficiently similar cases (given a Similarity threshold of some kind); and
- Selecting the best case from the set of cases returned.

Some systems retrieve cases based largely on superficial syntactic similarities among problem descriptors, while advanced systems use semantic similarities.

2. Reuse Case

Reusing the retrieved case solution in the context of the new case focuses on: identifying the differences between the retrieved and the current case; and identifying the part of a retrieved case, which can be transferred to the new case. Generally the solution of the retrieved case is transferred to the new case directly as its solution case.

3. Revise

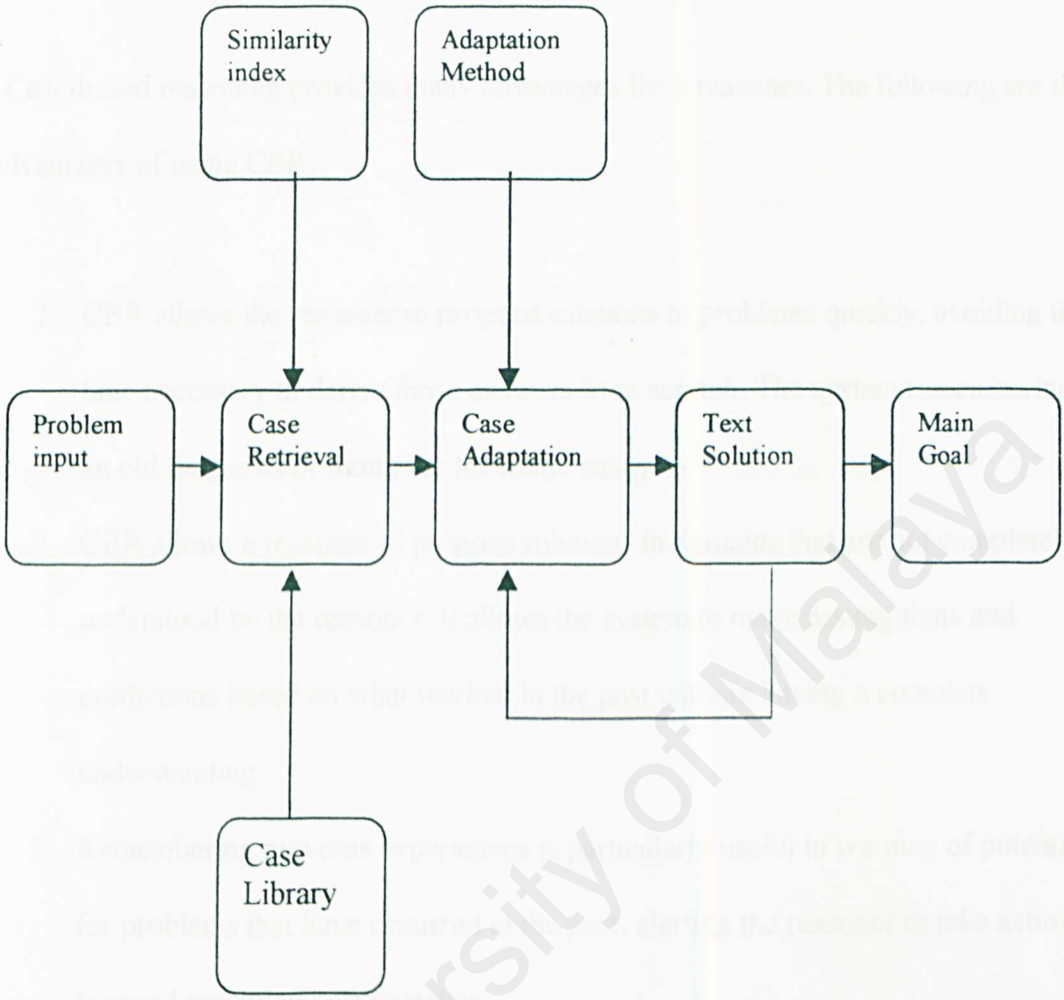
Revising the case solution generated by the reuse process is necessary when the solution proves incorrect. This provides an opportunity to learn from failure.

4. Retain

Retaining the case is the process of incorporating whatever is useful from the new case into the case library. This involves deciding what information to retain and in what form to retain it; how to index the case for future retrieval; and integrating the new case into the case library

A good tool should support a variety of retrieval mechanisms and allow them to be mixed when necessary. In addition, the tool should be able to handle large case libraries with retrieval time increasing linearly (at worst) with the number of cases. The figure 3-2 shows the flow diagram of solution structure in CBR.

Figure 3-2 Nominal Processes in Case-Based Reasoning



The process of CBR involves two primary step: (1) find those cases in storage memory (case library) that have solved problem similar to current problem and; (2) adapt the previous solution to fit the current problem context.

3.5 Advantages of Using CBR

Case-based reasoning provides many advantages for a reasoner. The following are the advantages of using CBR:

1. CBR allows the reasoner to propose solutions to problems quickly, avoiding the time necessary to derive those answers from scratch. The system remembering an old diagnosis or treatment for future cases.
2. CBR allows a reasoner to propose solutions in domains that are not completely understood by the reasoner. It allows the system to make assumptions and predictions based on what worked in the past without having a complete understanding.
3. Remembering previous experiences is particularly useful in warning of potential for problems that have occurred in the past, alerting the reasoner to take actions to avoid repeating past mistakes.

3.6 Disadvantages of Using CBR

Also, there are disadvantages in using CBR to reason:

1. A case-based reasoner might be tempted to use old cases blindly, relying on previous experience without validating it in the situation.

2. A case-based reasoning might allow cases to bias him or her or it too much in solving a new problem.
3. Relying on previous experience without doing validation can result in inefficient solution.

3.7 Suitability of CBR with FAA System

The diagnoses of FAA systems try to retrieve past cases whose symptom lists are similar in nature of that of the new case and suggest diagnoses based on matching retrieved cases These cases store the symptom and solution to the specific problem. Therefore CBR is a useful tool because FAA domain needs:

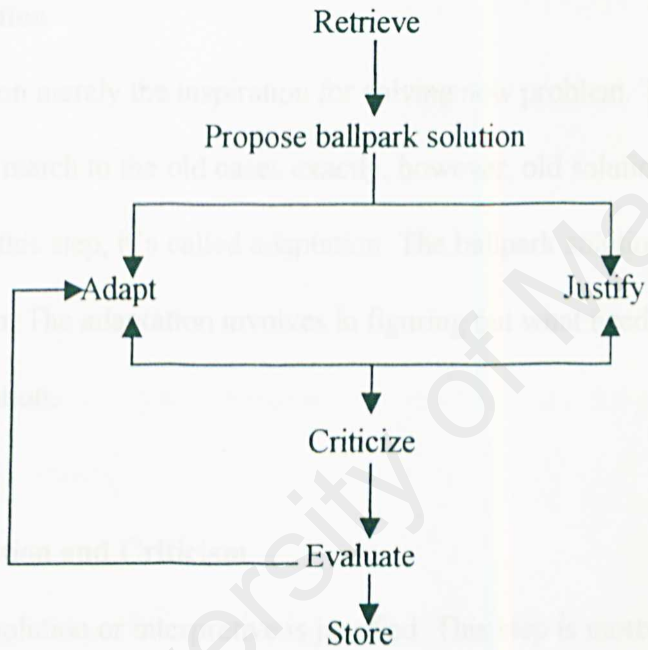
1. Records of previously solved problems exists;
2. Historical cases are viewed as an asset which ought to be preserved
3. Remembering previous experiences
4. Experience is valuable as reference

Case-based reasoning allows the case-base to be developed incrementally, while maintenance of the case library is relatively easy and can be carried out by domain expert.

3.8 Case-based Reasoning Cycle

In order to update the cases in the case library, it is important to know the case-based reasoning life cycle. Figure 3-3 shows the case-based reasoning life cycle [6].

Figure 3-3 The Case-Based Reasoning cycle



Here is the overview of the relationship of the CBR cycle:

- **Case Retrieval**

Case retrieval consists of two parts, recall previous cases and select the best subset. Retrieval is a step to retrieve ' good' cases that can support reasoning that comes in the next steps. Select the best subset, is a step to select the most promising cases to reason with from generated in step retrieval.

- **Proposing a ballpark Solution**

In this step, the system extract the relevant portion of old cases to form a ballpark solution to a new case. In this case, it involves partitioning the retrieved cases according to the interpretations or solution to predict and, based on that, assigning an initial interpretation to the new problem.

- **Adaptation**

The old solution merely the inspiration for solving new problem. This is because the new case rarely match to the old cases exactly, however, old solutions must be fixed to fit new case. In this step, it's called adaptation. The ballpark solution is adapted to fit the new situation. The adaptation involves in figuring out what needs to be adapted and doing the adaptation.

- **Justification and Criticism**

In this step, a solution or interpretive is justified. This step is more likely to be validation step. It can criticize solutions using all the techniques of interpretive CBR. Criticism may require retrieval of additional cases and may result in the need for additional adaptation, this time, it is called repair. This step is giving the assignment of blame or credit to the old cases.

- **Evaluative Testing**

This is the most important part of case-based reasoner. This step gives the reasoner way to evaluate its decisions, and allowing it to collect feedback that enables it to learn.

Evaluation is a process of judging the effectiveness of the proposed solution. It includes explaining differences (e.g. between what is expected and what is actually happen), justifying differences (e.g. between a proposed solution and the old cases), predicting outcomes, and comparing and ranking alternatives possibilities.

- **Memory Update**

Memory update is choosing the ways to 'index' the new case in the memory.

3.9 Programming Tools

Another important issue of designing an intelligent system is the programming tools. As the result of some research, it is found that Visual Prolog is the most suitable tool for building the FAA system.

It is chosen to be the programming language because of its characteristic, all in one. With Visual Prolog, applications such as customized knowledge bases, intelligent systems, and natural language interfaces, smart information management systems are within the programming domain. It allowing the programmers to model the logical relationships among objects and processes, complex problems are inherently easier to solve, and resulting program is easier to maintain through it lifecycle.

Visual Prolog is chosen because it is portable and wealth of facilities. With visual Prolog, the system can be run with many platform across the MS-Windows 3.x, 95,98, and 2000; NT and OS/2 platforms. It is wealth of facilities because all the components that make up the integrated environment can be used in it owns program. A wide range of standard of predicates provides with all the features that are expected from a professional programming language. These include:

- Input/output procedures
- Arithmetic and string manipulation functions
- External database with binary trees and network support
- Operating system functions

3.10 Introduction to Visual Prolog

Prolog is a derived from **Pro**gramming **Log**ic. It is a programming language for symbolic, non-numeric computation. It especially well suited for solving problems that involve objects and relations between objects [1]. Prolog program is a collection of facts together with rules for drawing conclusions from those facts. Therefore, Prolog known as a declarative language.

In Prolog, it includes an inference engine, which is a process for reasoning logically about information. This inference engine includes a pattern matcher, which retrieves stored information by matching answers to questions.

The important feature of visual prolog is that, in addition to logically finding answers to the goal question, it can deal with alternatives and find all possible solutions rather then only one. Instead of just proceeding from the beginning of all programs to the end, Visual Prolog can back-up and look for more than one way of solving each part of the problem.

3.11 Conclusion

This chapter reviewed the reasoning technique uses in FAA system. The waterfall model is a method for software engineering cycle. FAA system will utilizes the case-based reasoning model to reason the problem. Also, this chapter discussed the programming tools. The FAA system will be developed using Visual Prolog.

Chapter 4:

DESIGN

4.1 FAA System Data Flow

As discussed in Chapter 1, the FAA system consists of two parts, diagnosis and advice. The diagnosis is the primary goal of the system, which is diagnose the emergency situation to determine the specific case. The secondary goal is to give the advise to the user. For the existed case, the system will presents, the existed solution, and for the new case, the system will need to find the 'almost similar' case and propose the ballpark solution for the new solution.

To get the clear picture of the operation of the system, the following are the summary of process of the FAA system.

1. User will interact with the FAA system by providing the initial state of the emergency situation. User will click the option on the main menu screen.

2. By using the forward chaining approach, the system will prompt the intermediate finding screen. The intermediate finding screen also requires user to provide input using the check box. Intermediate screen will display a list of symptom that requires users to click any symptoms that is found occur to the victim. This facility is to make the system user-friendly. Input using text is not recommended because in panic situation, input using text is inefficient. This is because to build a sentence or to find a suitable description is almost impossible.

3. With the information provided by the user, the system will extract all the case that has the same index to the current emergency situation. The inference engine will try to match the symptoms with all the premises for each case. If the case is found, the primary goal is achieved.

4. If the system fails to find the exact similar case in the case library it will find the 'almost similar' case in order to create new solution for a new case.

5. If there is a new case, knowledge base will be updated by the system itself.

6. Secondary goal is final finding, the advice part. The system will present the advice in a conclusion screen of the system.

Figure 4-1 and 4-2 will show the logical and physical data flow diagram of the FAA system. The logical data flow diagram illustrates the operation of the system. The flow process is more to the system activity. On the other hand, the physical data flow illustrates how the system will be implemented. The processes represent the program modules and procedures. Type of data stores is the master file. Any processes that operate at two different times must be connected by a data store. This is also includes the status in the working memory.

Figure 4-1 Logical Data Flow of FAA System

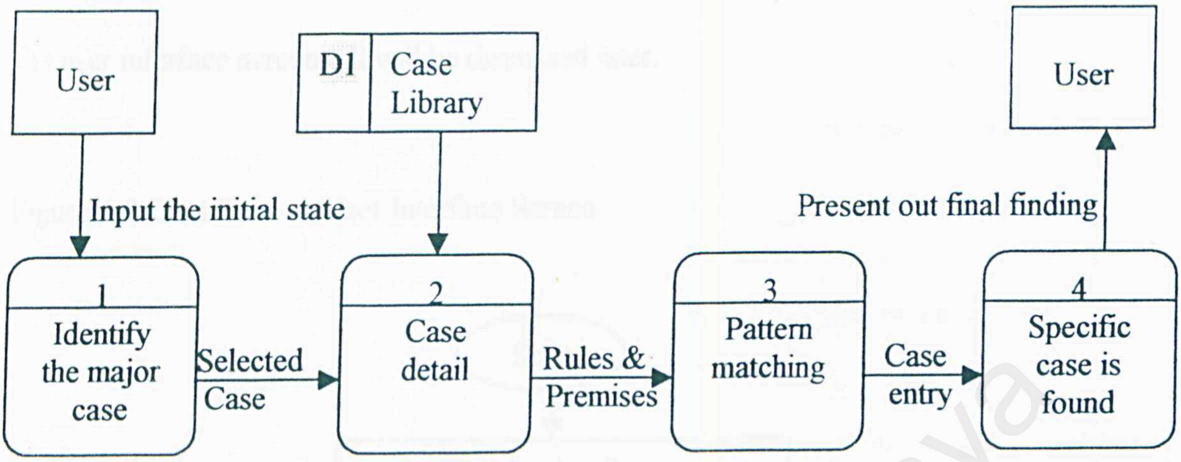
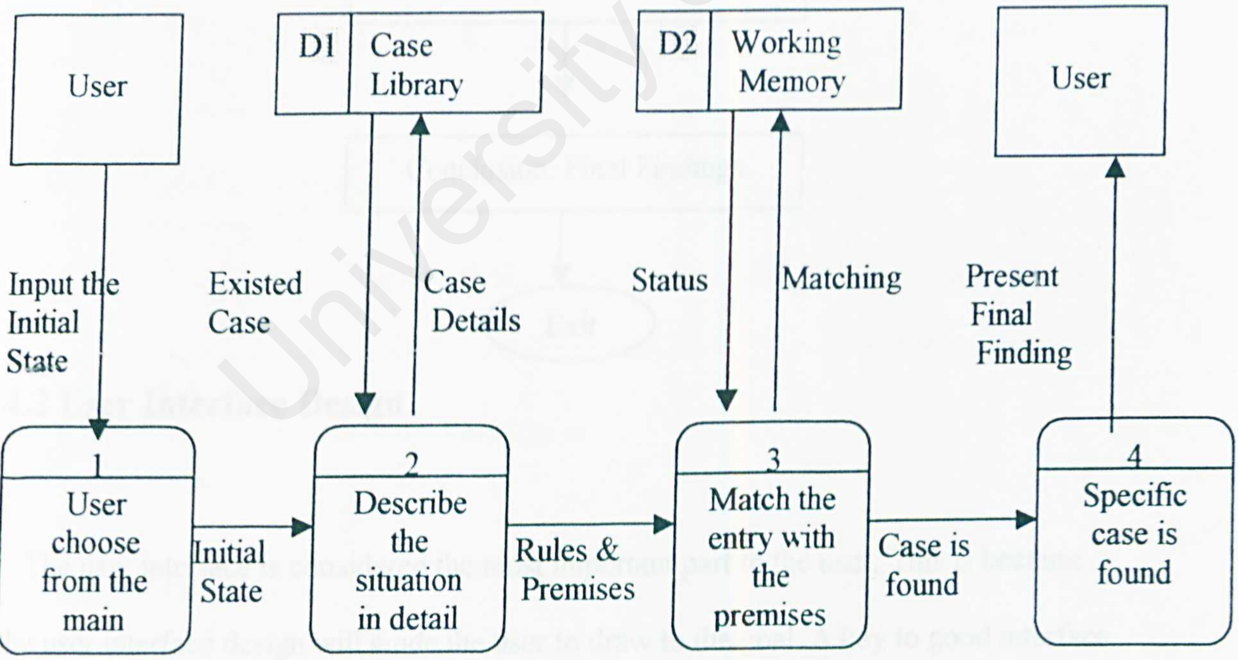
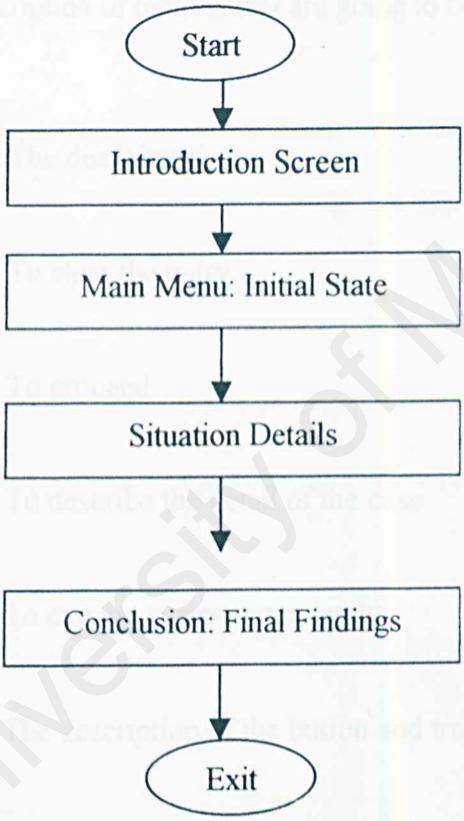


Figure 4-2 Physical Data Flow of FAA System



The processes start will the introduction and end with the conclusion which is the final findings. The user interface screens guide these processes. Figure 4-3 shows the flow the user interface screens. It will be discussed later.

Figure 4-3 Data Flow of User Interface Screen



4.2 User Interface Design

The user interface is considered the most important part to the user. This is because the user interface design will guide the user to draw to the goal. A key to good interface design is the consistency. For each screen, similar material should be consistently placed in the same locations. For instance, the location of question button should be placed in the same spot for each screen.

FAA system is design to interact with the user using text. The system displays the questions and the user responded by selecting from a menu of possible answers.

There are two basic types of interface screen: display screens and question screens. Each screen type contains information specific to its need.

The following are the description of buttons that are going to be used in the screens.

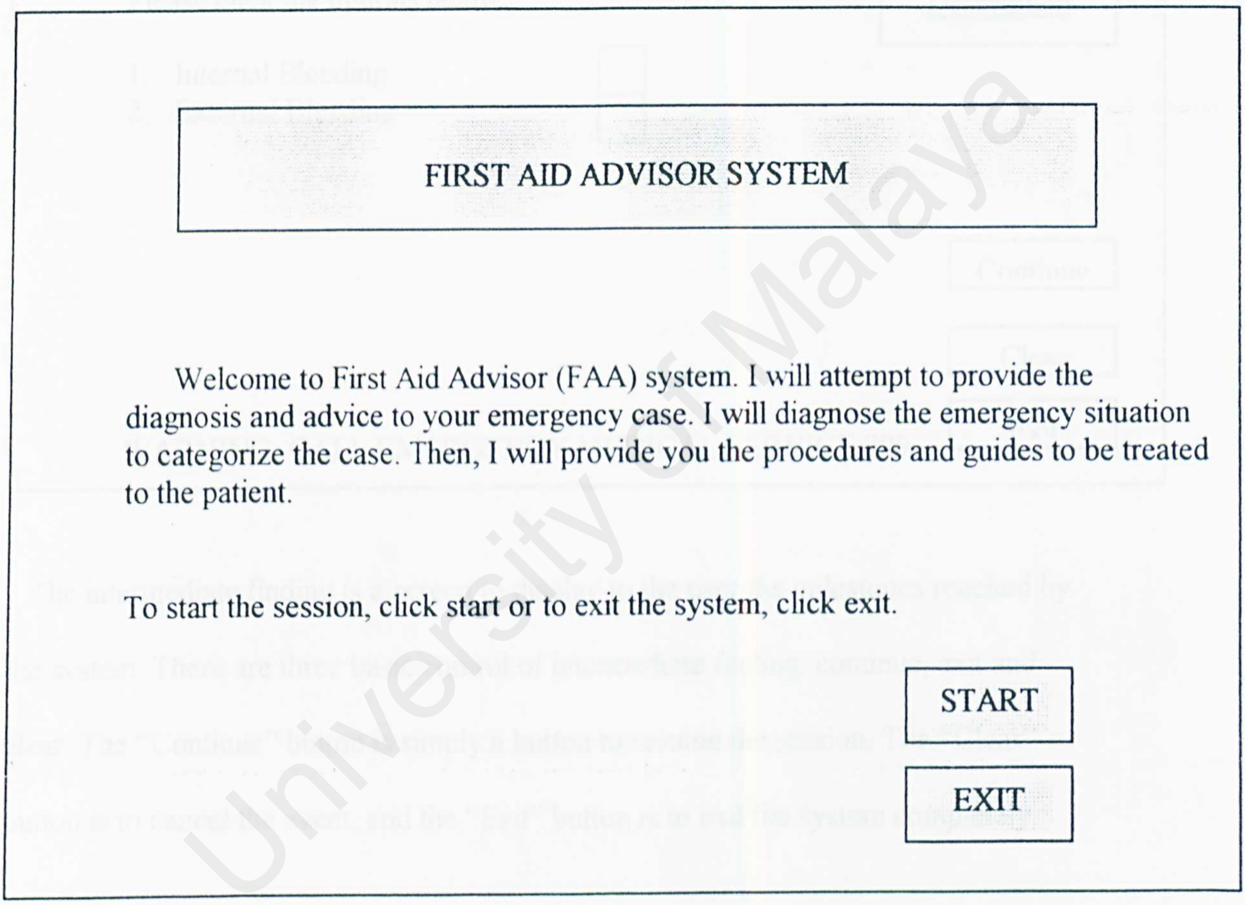
About	The disclaimer
Clear	To clear the entry
Continue	To proceed
Description	To describe the detail of the case
Exit	To exit the system completely
Help	The description of the button and trouble shoot the searching
Start	To start the program

Display Screen

The purpose of display screen is to present information to the user. There are three types of display screens: introduction screen, intermediate finding and final display screen. Each screen consists of two principal part, text portion and control section. Text portion is written to prompt the specific needs of the type of display screen. The control section provides the user with options for controlling the operation of the system. The title and header is optional.

Introduction Screen

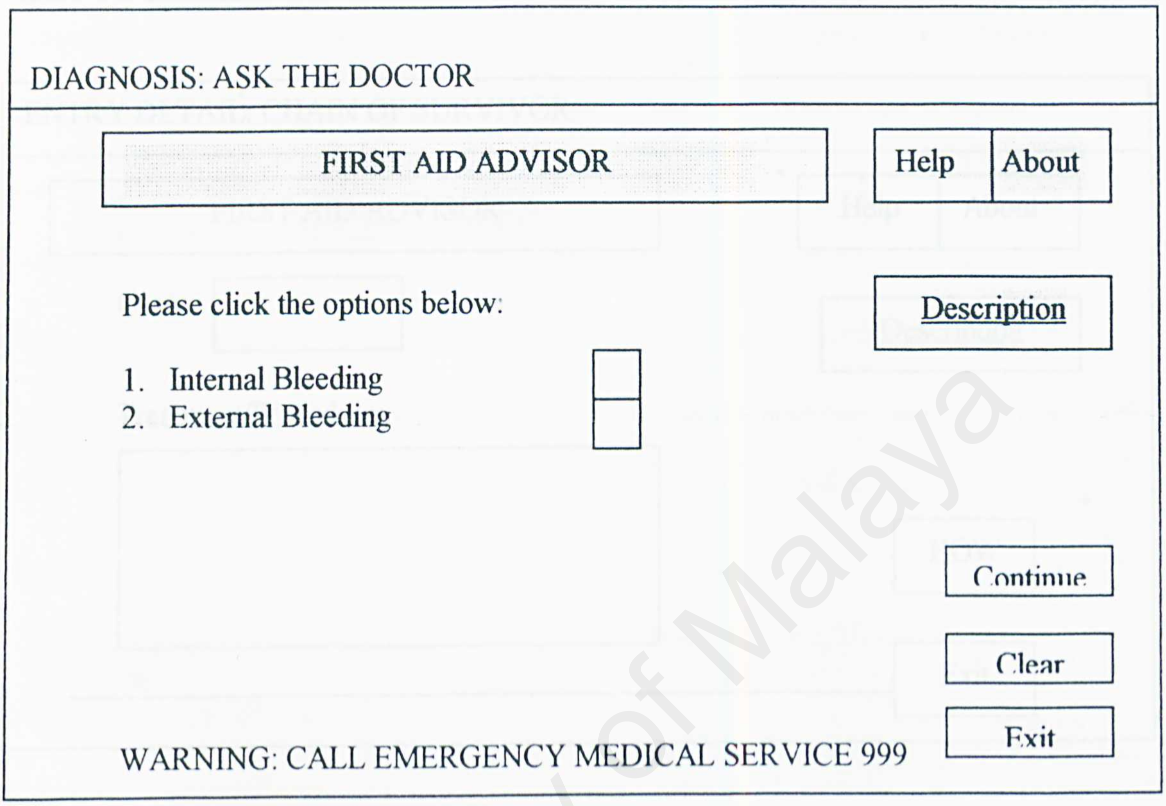
Figure 4-4: Introduction screen



Above is the introduction screen for the FAA system. Also it includes the instruction to handle the system. It has two control buttons, Start and Exit button. Start button is to starts the session and Exit is to close the session.

Intermediate Findings

Figure 4-5 Diagnosis Screen

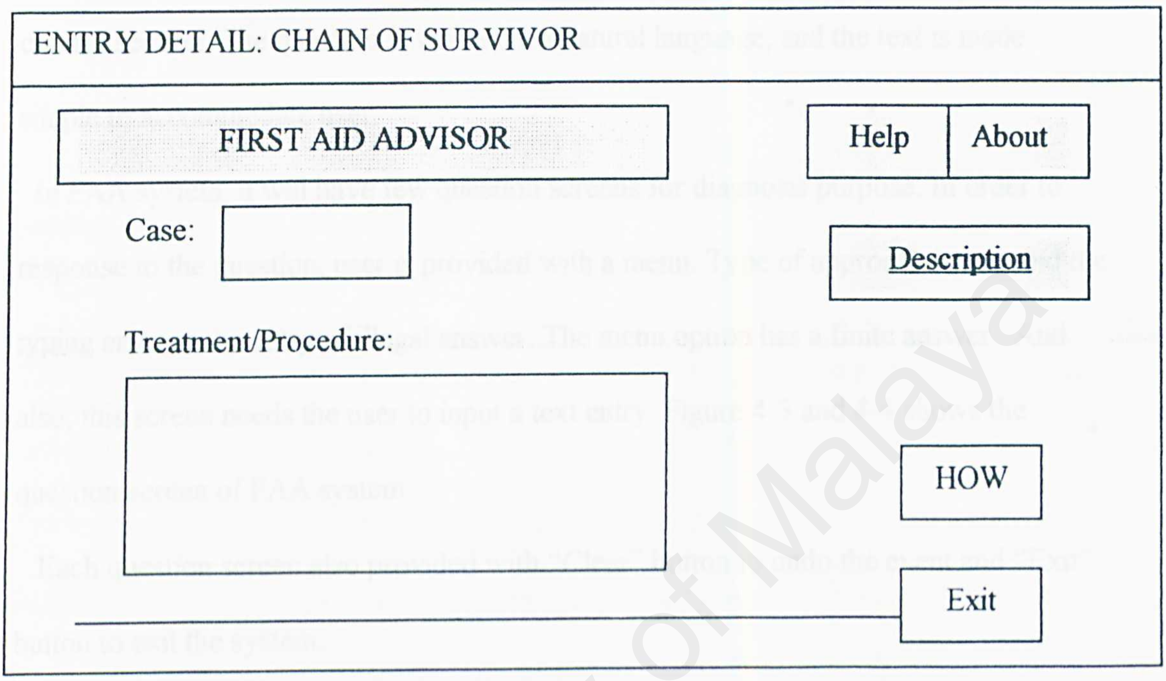


The intermediate finding is a screen to display to the user the milestones reached by the system. There are three basic control of intermediate finding: continue, exit and clear. The "Continue" button is simply a button to resume the session. The "Clear" button is to cancel the event, and the "Exit" button is to exit the system completely.

Consider the example of FAA system; during the process to determine the category of the case, the system may first attempts to gather all the relevant cases. Take for instance; the case is fall under the category of bleeding. Figure 4-5 shows the intermediate finding for case bleeding. Help button is to trouble shoot the searching and About is the disclaimer of the software. Description is to describe the findings.

Conclusion Screen

Figure 4-6 Conclusion Screen



The conclusion screen of FAA will shows to the user with the system's final finding, the recommendation of treatment and care to the emergency victim. This screen will present a set of guide and procedure to be preformed to the victim before the victim gets the advance treatment from the hospital.

This screen also provides a "HOW" explanation to the user. The system will display the rule for the specific case. Also, it includes the explanation facility for the description of the case. This can be done if the user clicks the "Description" button. Figure 4-3 shows the example of Conclusion screen.

This screen must have the "Exit" button in order to exit the system completely.

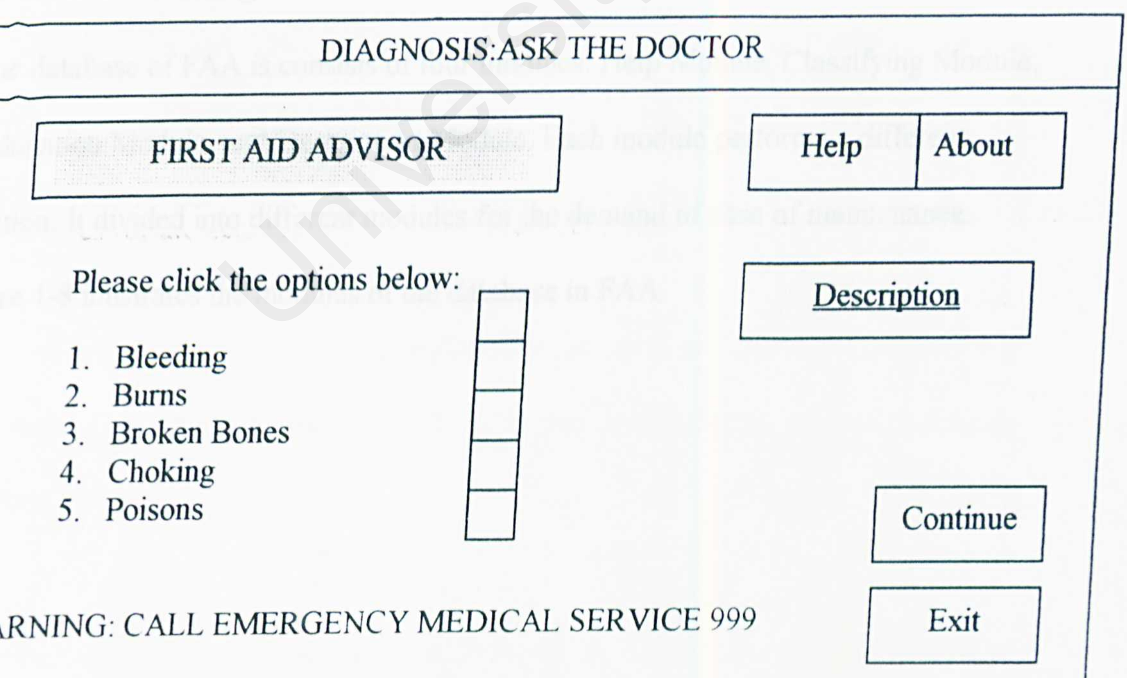
Question Screen

Basically the question screen is to obtain information about the user. This screen normally has three part; text portion containing the question, the answer entry and the control section. The question is written in a natural language, and the text is made simple to accommodate user.

In FAA system, it will have few question screens for diagnosis purpose. In order to response to the question, user is provided with a menu. Type of approach is to avoid the typing error or the entry of illegal answer. The menu option has a finite answer. And also, this screen needs the user to input a text entry. Figure 4-3 and 4-4 shows the question screen of FAA system.

Each question screen also provided with "Clear" button to undo the event and "Exit" utton to exit the system.

igure 4-7 Question Screen



4-1 Summary of User Interface Design

Question Screen

Basically the question screen is to obtain information about the user. This screen normally has three part; text portion containing the question, the answer entry and the control section. The question is written in a natural language, and the text is made simple to accommodate user.

In FAA system, it will have few question screens for diagnosis purpose. In order to response to the question, user is provided with a menu. Type of approach is to avoid the typing error or the entry of illegal answer. The menu option has a finite answer. And also, this screen needs the user to input a text entry. Figure 4-3 and 4-4 shows the question screen of FAA system.

Each question screen also provided with "Clear" button to undo the event and "Exit" button to exit the system.

Figure 4-7 Question Screen

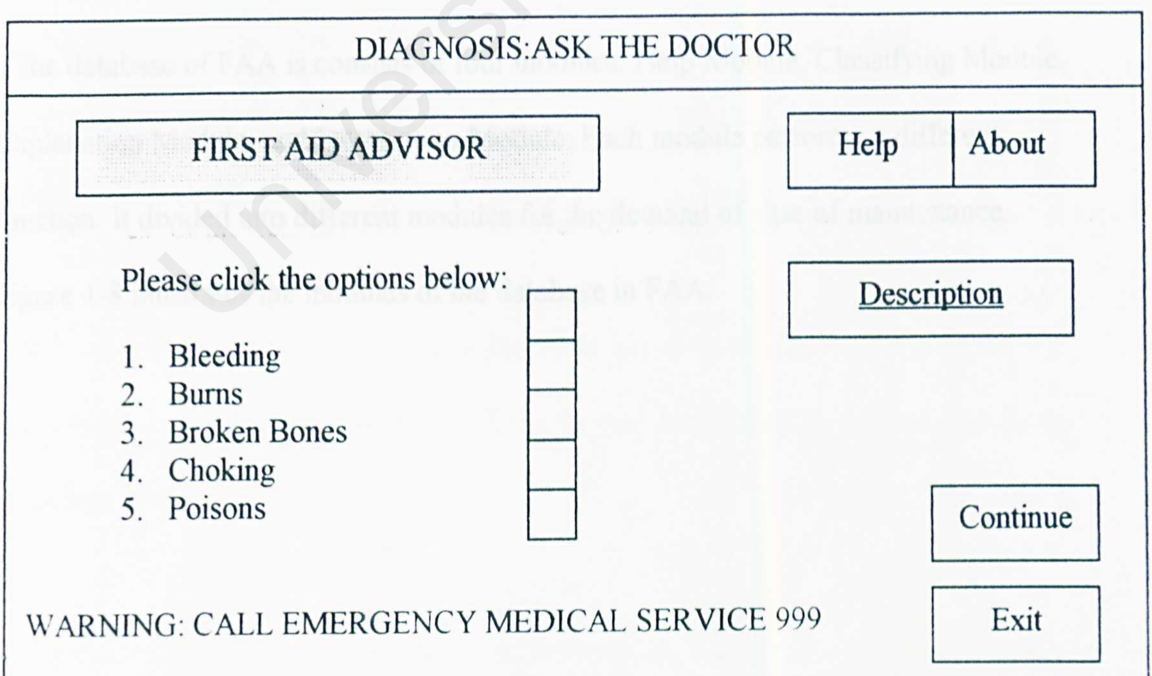


Table 4-1 Summary of User Interface Design

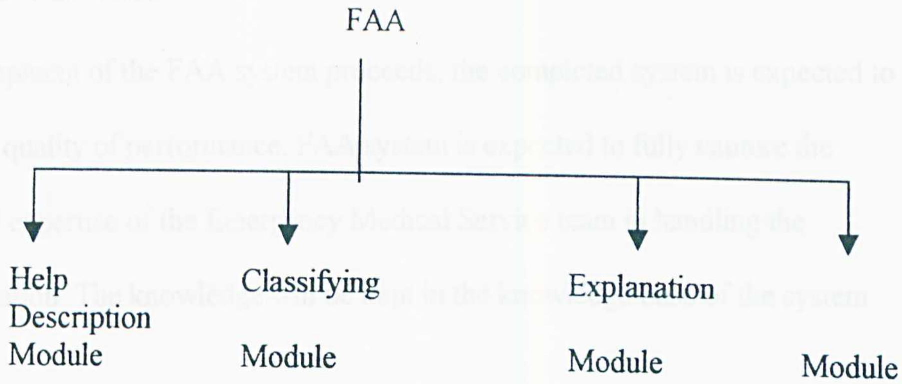
Screen	Issues	Control
Introduction	System Objective Problem Discussion Session Needs	Start Exit
Intermediate Findings	What Was Found Major Reasons Why What Will Be Done	Continue Clear Exit
Conclusion	What Was Found Major Reason Why	Exit
Question	Why Question Natural Language Menu vs. Text	Clear Exit

From the review of the system interface, it shows that the designed system is using the forward chaining intelligent system. The design process is highly iterative. Forward chaining is used because the knowledge base is large. This system will define the initial data at the starting point process. Also, rules in forward chaining are used in order to maintain better control over the interface process.

4.3 Database Design and Modules

The database of FAA is consists of four modules. Help Module, Classifying Module, Explanation Module, and Description Module. Each module performs a different function. It divided into different modules for the demand of ease of maintenance. Figure 4-8 illustrates the modules of the database in FAA.

Figure 4-8 Database Modules



Help Module is consists of the button description and the trouble shooting of the searching. The button description is describing the function of each button. Next, Help module also includes the trouble shooting for searching. This function is caters for the user who is not familiar with the system. The system will display the manual or displays the guideline to get to the wanted information.

The Classifying Module consists the case library. In case library, the cases divided into five major cases, which are the cases in the main menu; Bleeding, Burns, Broken Bones, chocking and Poisons. The main menu is the initial state of the searching. Each case has it specific case (sub-case). In each sub-case it includes the all the symptoms and its solution. The solution is the treatment and advice for the specific case.

Explanation Module works as the explanation facility in this system. It consists the case rules. This module is caters for 'HOW' button. It will display the rule on how to conclude the case.

Another module is Description Module. This module consists the description of the symptoms. It will describe in case detail.

4.4 Expected Outcome

As the development of the FAA system proceeds, the completed system is expected to meet to certain quality of performance. FAA system is expected to fully capture the knowledge and expertise of the Emergency Medical Service team in handling the emergency situation. The knowledge will be kept in the knowledge base of the system.

Also, the FAA system is expected to achieve six technical objectives and three general objectives. The technical objectives (describes in Chapter 1) are assist the medical expert, availability, portability, reliability, efficiency, ease of use, and ease of maintenance. The general objectives are to increase the quality of life and support the growth of technology.

Next, the system is expected to meet the requirement of ease of use and ease of maintenance. Ease of use is mainly for the user. The system is designed to be user friendly. And ease of maintenance is the ease to add and update the knowledge and control in the system by the programmer. Since that the intelligent system is separate its knowledge from control, therefore by updating any of the data, it won't ruin the whole program. Also, the response time is expected to be less than 60 second to come up with the conclusion.

By developing this system it is hoped it will improve the growth of artificial intelligence technology.

4.5 Conclusion

Chapter 4 is a phase of designing the system. At this phase the system has completed the task of selecting the software, selecting the knowledge representation technique and selecting the control technique. This chapter also includes the process flows for both logical and physical. Following the process flow, this chapter also includes the prototype of the user interface. The user interface screens will guide the process of obtaining the goal. The next phase is knowledge engineering. This will be discussed in the next chapter.

Chapter 5:

DEVELOPMENT/ IMPLEMENTATION

5.3 Hardware Requirement

System:

- Personal computer
- Processor Genuine Intel® Celeron™ Processor 366Mhz
- 56.0 RAM
- Bus type PCI
- Port 1 Parallel, 2 Serial
- Floppy Disk 1.44 MB
- Hard disk 3.02 GB (at least)

Input:

- Keyboard Microsoft Natural Keyboard
- Standard serial Mouse

Monitor

- Plug and Play monitor (1600 x 1200 Max resolution)
- Sis 620 (800 x 600 in 64k Colors)
- 14" Monitor (at least)

Multimedia

- Sound and CD Rom

System Development

The system has divided into modules, the interface module, the knowledge base module, database module, the inference engine and the control module .The modules have been built independently and later are bind together as a whole to meet the system requirement.

5.4 End User Interface module

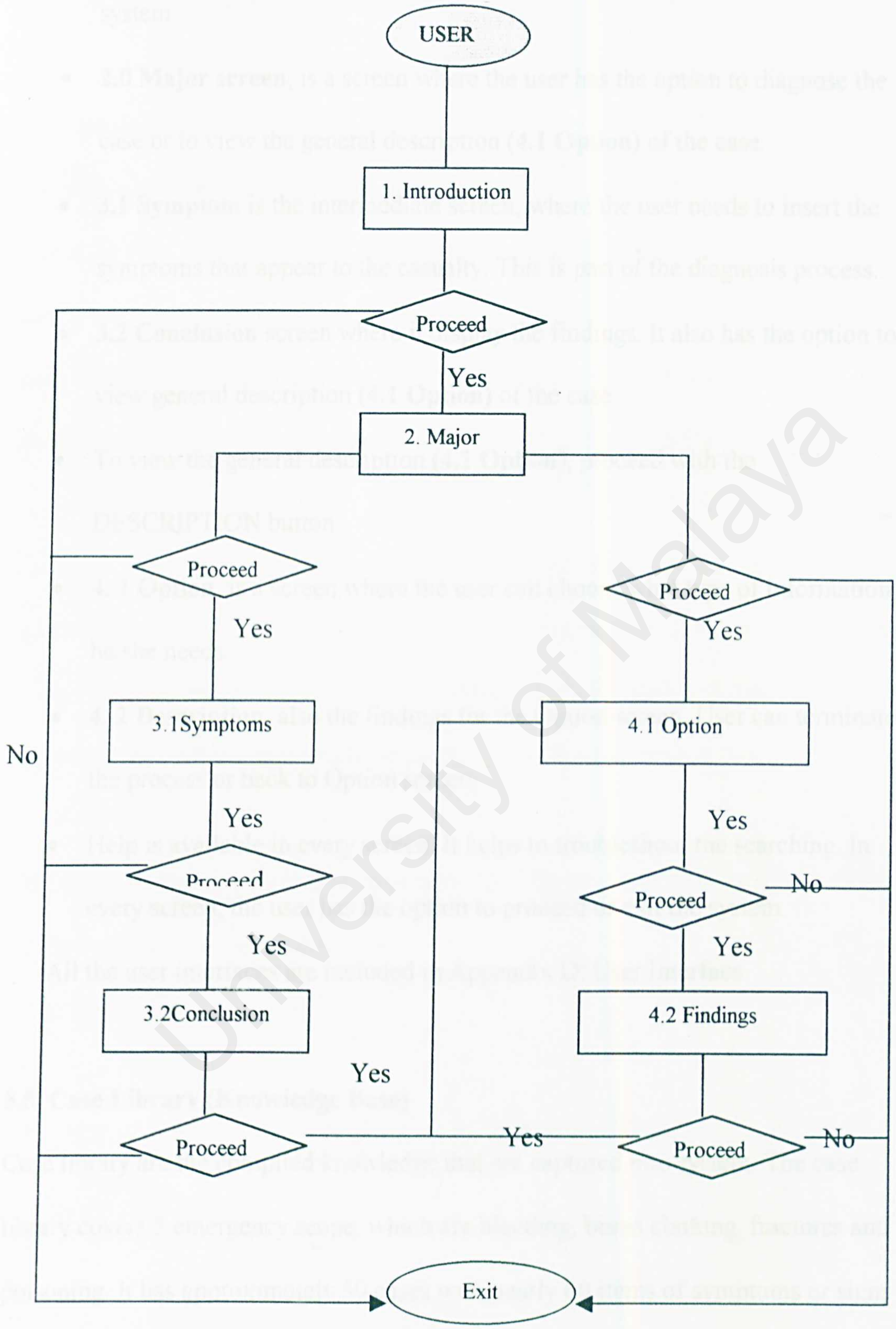


Figure 5.1 Data flow of the User Interface Screen

- **1.0 Introduction** screen will give the user information about the whole system..
- **2.0 Major screen**, is a screen where the user has the option to diagnose the case or to view the general description **(4.1 Option)** of the case.
- **3.1 Symptom** is the intermediate screen, where the user needs to insert the symptoms that appear to the casualty. This is part of the diagnosis process..
- **3.2 Conclusion** screen where it display the findings. It also has the option to view general description **(4.1 Option)** of the case..
- To view the general description **(4.1 Option)**, proceed with the DESCRIPTION button.
- **4. 1 Option**, is a screen where the user can choose what type of information he/she needs.
- **4. 2 Description**, also the findings for the Option screen. User can terminate the process or back to Option screen.
- Help is available in every screen. It helps to troubleshoot the searching. In every screen, the user has the option to proceed or exit the system.

All the user interfaces are included in Appendix D: User Interface

5.5 Case Library (Knowledge Base)

Case library are the compiled knowledge that are captured into system. The case library covers 5 emergency scope, which are bleeding, burns choking, fractures and poisoning. It has approximately 50 cases with nearly 60 items of symptoms or signs for emergency case. The knowledge has been written in Visual Prolog source code (can be found in Appendix B: Knowledge Base). It contains the rule and procedures that will conclude the emergency case. It is called procedural knowledge. The documentation of the knowledge can be found in the Appendix A: Form.

5.6 The database

The database contains information about the current status of the problem being solved. In FAA system database is in "**temp.pro** (can be found in 'exe' folder during run time)" file. It records fact about the problem. Known facts are stored there initially. Then, the new facts are added as they go from inference process. The fact keeps track of all that is known during the inferencing operation.

The inference engine begins its searching, matching the rules in the knowledge base against the information in the database. As the rule is examined, action caused when a rule fires may change the content of the database, then updating the status of the problem. New fact become available to use in the decision making process.

Here is the partial coding to save the known fact input by user into the database and match it against the knowledge base:

Case: Burns,

```
%BEGIN Burns, idc_ok _CtlInfo
 dlg_burns_eh(_Win,e_Control(idc_ok,_CtrlType,_CtrlWin,_CtlInfo),0):-!,
        save("temp.pro",appearance),
        kes(X),
        win_destroy(_win),
        dlg_conclusion_Create(_Win),
        !.
%END Burns, idc_ok _CtlInfo

 dlg_burns_eh(_,_,_):-!,fail.

%END_DLG Burns
```

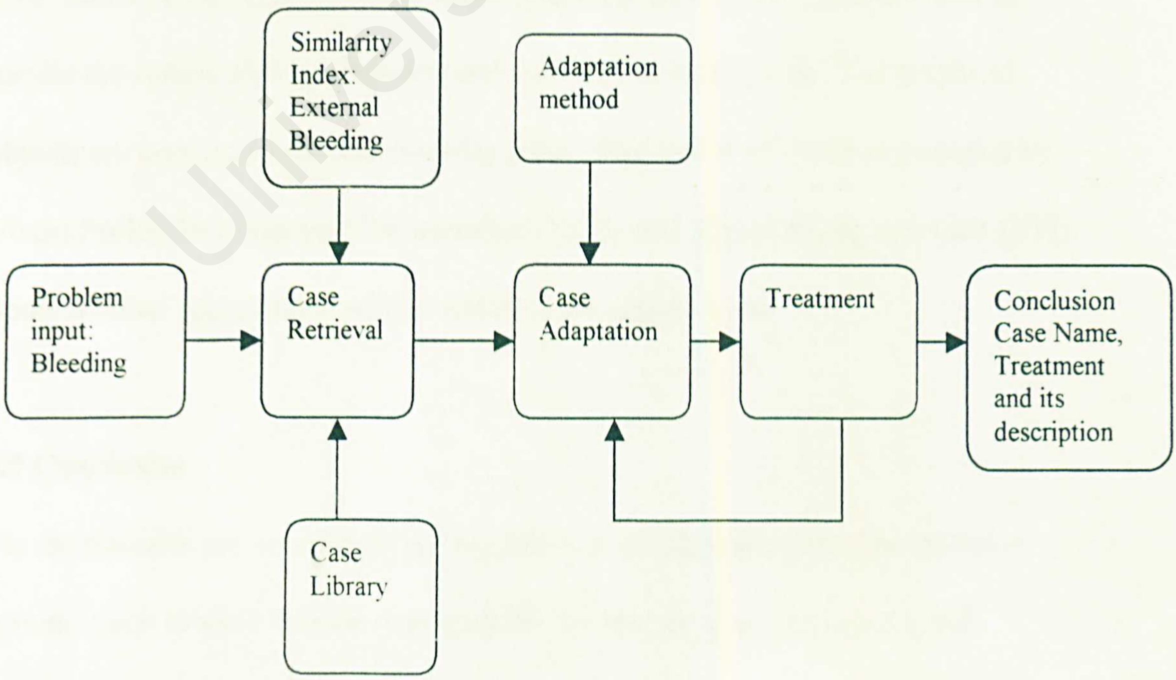
Note:

- save("temp.pro",appearance), is to save the symptoms that are inserted by the user during the diagnosis process. The symptoms will be kept in ***FACT-appearance***. When the user click the CONTINUE button, it will scan all the retrieved cases from the case library against the fact in the database.
- kes(X), is a function to try all the possibility.

5.7 Inference Engine Module

Inference engine will implement the search and pattern matching operation. It will look at each line of the code and then implements the operation specified. It will examines the rules in a particular sequence looking for matches to the initial and current conditions given in the database. As rules matching these condition are found, the rules are fired, then initiating the action specify.

Figure 5.2 Sample: Reasoning Process for Case Puncture



- User will input the problem in the diagnosis screen.
- The system will gather all the cases that are associated with bleeding.
- The cases that has the similarity index : External Bleeding , will be retrieved from the case library.
- The system will try to match the facts in the database against the cases that are retrieved from the case library.
- If the facts match perfectly with one of the cases in the library, therefore it will use the same text solution as the conclusion (finding).
- If none of the cases match to the fact, there the system has to find the most similar case to the input problem.
- Then it will save the new case, this will be added to the case in the case library.

5.8 Control Module

Control is implemented by inserting the graphical object such as push button, dialog note, check box and radio button. These graphical objects will guide the user to handle the system during diagnosis and viewing the information. The graphical objects are inserted to the interfaces by using 'drag and drop' method provided by Visual Prolog Development Environment (VDE) and Visual Prolog Interface (VPI). Some internal source code will be added to the default code.

5.9 Conclusion

As the modules are completed, the modules are combined to form the intended system. Each module is built independently to ease the implementation and debugging. Now the system is ready for testing.

Chapter 6:

TESTING

System Testing

As the project proceed, the system will need to be periodically tested and evaluated to assure its performance is converging towards established goals. The system is tested thoroughly. The testing can determine whether the system does solve the problems for which it constructed.

6.1. Module Testing

Module testing is individual testing for the modules. This is to keep track of the problems in the system and the deficiencies. The adjustment is made if the problem found.

For the Interface module, the testing is done by the using 'test mode' utility, it is the utility provided by the Visual Prolog.

For the Control module, it only can be tested after inserting the source code to invoke the event. This is done after the Interface testing. This includes examine the code.

The knowledge base test has to be together with the Control and Inference engine modules. This is because it needs the response from the user, and the response will be save in the working memory before system search for the conclusion. The test is includes reading the source code, checking the syntax and algorithm. It is found that the system is able to diagnose the emergency case and able to give the advise to the user.

6.2. Integration Testing

After satisfied with the module test, the modules must be combined to into a working system. The collection of modules is tested to make sure the system working correctly. This integration is planned and coordinated so that when the failure occurs the cause can be identified and the code will be edited. This is includes the flow of the interfaces till it reach the conclusion.

6.3. System Testing

The system testing process includes function testing, performance testing and acceptance testing.

A function test checks that the integrated system performs its functions as specified in the requirement. The for example, the diagnosis function to diagnose the medical case and later give the advice to the user.

Next, the performance test, it compares the integrated components with the nonfunctional system requirement. These requirements, including accuracy of the conclusion, speed (efficiency and timely response), reliability, availability and maintainability.

At this point, the system has operates the way it intended to. This is called verified system; the designer's interpretation of the requirements specification. As it is satisfy the requirement, the system is validated.

Another test is, acceptance test, this is to assure system is manages to solve the user's problem. This test needs the involvement from the actual user (refer to User acceptance form in Appendix A: Form)

6. 4 Conclusion

After the testing, some of the issues are covered (the testing result is included in the A: Form), the issue of whether the system meets its original goals when applied to real problems. After the testing by the developer, the test later includes the control group of user to determine the user acceptance.

Chapter 7:

EVALUATION AND MAINTENANCE

Evaluation and Maintenance

7.1 Evaluation

Evaluation phase is to evaluate the performance of the FAA system. For evaluation purpose, the emergency case were diagnose the FAA and by ten individuals from different background. These individuals included, one doctor, three medical student, three faculty members, a teacher, high school student and a layman. These individual are called 'evaluators', were asked to test the system and give recommendation for the system. The recommendation is recorded in a form (refer to the User Acceptance form in Appendix A: Form). From the evaluation, it is find that the system is success.

7.2 System Strength

After the system is complete, it is found that the system is success. Here are the summaries of the system evaluation:

- **Assist The Medical Expert**

It is found that the system manages to assist the medical expert to diagnose the emergency case and give advice to be performed to the casualty.

- **User Friendly**

The graphical user interface is made the system easy to use. It uses the natural language and the graphical object likes buttons and checkbox. At the time of emergency, user prefer to use the checkbox interaction rather than input text. This is because people have the difficulty in describing the symptom if they are panic.

- **Reliable**

The compiled knowledge is captured to the system and has been approved to be accurate by a doctor and the medical students.

- **Efficient and timely response**

The system is timely response and the finding is accurate.

- **Portable**

The system can runs on various operating system.

- **Ease of expansion**

The backend is easy to maintain, because the system separates its knowledge and control.

- **Intelligent User**

The user can diagnose the emergency situation when there is no medical expert is available.

7.3 Limitation

- **Cases are limited**

Cases in the system are limited to the 5 scopes. Therefore, it needs to expand it scope to serve a better system.

- **Case Validation**

The wrong advice may lead to another emergency case, therefore, the solution for new case has to be validate by the medical expert, otherwise the system will not be reliable. The system has to be maintained frequently.

- **Unable to Loop the Diagnosis Process**

The system is unable to give good conclusion for the next diagnosis if the user not exit the system completely. In order the get a good diagnosis, the user has to start all over again, so that the database will retract everything from its memory.

- **Still Unstable**

The system still have bug. Therefore Bleeding radio button has been disabled for the system can run smoothly.

7.4 Maintenance

Maintenance is the final phase of the FAA project. This task follows the deployment of the system. As the FAA system is used, deficiencies may also be discovered. Therefore the maintenance is done periodically.

The maintenance involves the knowledge and control of the system. The knowledge maintenance is changes made to the knowledge, whether to delete or add knowledge. The control can be maintained by edited, by adding or deleting the graphical control (push button, radio button).

For security purposes, the developer performs the maintenances. This is because; the developer is the individual has both knowledge engineering skills to perform the task and experience with the project.

7. 5 Future Enhancement

- **Text Input**

Besides using the checkbox, it is recommended that the user input the symptoms using the text. This is applied when the symptoms is not in the list at the checkbox. Text input will caters for user who wants to learn about the first aid.

- **Explanation**

Add the explanation facility to the system, so that the system can explain their reasoning.

- **Facility**

Add the facility of that the user can print the conclusion or system's finding.

7.6 Discussion

- **Lack of experience**

Due to the lack of experience in handling Visual Prolog, the process of integrating the interfaces is taking a very long period. Integrating the whole system is very tedious because the source code is very 'sensitive' – one syntax error can makes the system become unstable or the system unable to perform its function.

- **Lack of technical support**

In the university itself, the reference book for Visual Prolog is not available. All the technical supports are get form the Prolog Development center through the Internet.

- **Suggestion**

It is suggest that university of faculty to provide a proper reference book for Visual Prolog. Faculty should provide a lab exclusively for students majoring in Artificial Intelligent, and equips with the hardware and software that are design to develop intelligent system or software (Prolog, MatLab, etc).

7.7 Conclusion

The lesson learn from the whole process:

- There are six major phases followed when developing the FAA system, which are often performed in parallel and cycled through several times before the project is complete.
- Problem, people and deployment issues are important when assessing the project's feasibility.
- The testing is involves the validation of the system results and its reasoning process. The testing is conducted throughout the project, becoming increasingly more formal as the project proceeds.
- The system should be designed to ease the later maintenance task.

Reference

1. Bratko,Ivan.PROLOG Prolog for Artificial Intelligence. Addison-wesler,3rd Ed,2001.
2. Durkin,john.expert Systems DESIGN AND DEVELOPMENT,New York: Macmillan Publishing company,1994.
3. Giarrantano, joseph.EXPERT SYSTEMS Principles and Programming. Masuchett: PWS-KENT Publishing Company,1989.
4. Jackson,Peter. INTRODUCTION TO EXPERT SYSTEMS.ADDISON WESLEY,1986.
5. KNOWLEDGEBASE SYSTEMS Advanced Concepts techniques & Application. Ed by Tzafestas, SG, World Scientific Publishing,1997.
6. Kolodner, Janet. Case-Based Reasoning.san Mateo: Morgan Kaufmann Publisher,1990.
7. Raeth,PG. EXPERT SYSTEMS Software Methodology for Modern Applications. Los Alamtos:IEE Computer society Press, 1990.

Internet

8. http://www.aiai.ed.ac.uk/links/cbr.html
9. http://www.ai-cbr.org/projects.html
10. http://www.members.ozemail.com.au/~lisadev/sftdoc.htm
11. http://www.parasolemt.com.au/
12. http://www.redcresent.org.my/
13. http://www.redcross.org/ca/scul/medprot.htm/#introduction

Journal

14. Deepak Kehemani,Md. A.K Sadiq,Satanu Chakraborti. CBR in the Medical Domain: Homeopati, IC-Ai'01 International conference. Las Vegas, June 25-28(2001): 1247-1251

Personal interview

15. Dr. Mohd. Salleh bin Yahya, Senior Mdiacl Officer, Trauma and emergency Center, University Malaya Medical Center
16. Reina Lim (Medical student), 4th yr, MBCHB, University of Glasgow.
17. Richard Ang (Medical Student),5th Yr,MBBS, Barts & The London School of Medicine & Dentistry Queen Mary, University of London.
18. Winston Erng(Medical Student),5th Yr, MBCHB, Univeristy of Dundee.

APPENDICES

1. Appendix A: Forms

2. Appendix B:

Knowledge base

3. Appendix C:

User Manual

Appendix A:
FORMS

1. Sample: Rule Documentation Form for Emergency Case (Natural Language Version)

Project: First Aid Advisor

Case: Burn

Rule Name: First Degree Burn **Rule Set:**

Description: First Degree Burn is the mildest of three types of burn, are generally caused by brief skin contact with hot water, steam, or hot objects or overexposure to the sun. It caused some blistering, swelling, redness and pain.

Symptoms/sign:

1. Reddening (like sunburn)
2. Outer Layer of skin only
3. Caused by hot water or hot object or over exposure to the sun.

Treatment:

1. Remove clothing from burned area immediately.
2. Run cool water over the burned area or hold a clean, cold compress on the burn until the pain subside.
3. If the burn area is small, loosely cover it with a sterile gauze pad or bandage.
4. Call the doctor if the victim is a child even if the burn seems mild.

WARNING:

1. Do not use ice
2. Do not apply butter, grease, powder or any other remedies to the burn

2.Sample: Rule Documentation Form for Emergency Case (Coded Version)

Project: First Aid Advisor
Case: Burn
Rule Name: First Degree Burn (first_degree_burn) **Rule Set:** burn
Description: First Degree Burn is the mildest of three types of burn, are generally caused by brief skin contact with hot water, steam, or hot objects or overexposure to the sun. It caused some blistering, swelling, redness and pain.
RULES: IF Burns THEN First degree burn IF Skin reddening AND Burn at outer layer only AND Caused by hot water OR Caused by hot object Or Overexposure to the sun THEN First Degree Burn
Treatment: 1. Remove clothing from burned area immediately. 2. Run cool water over the burned area or hold a clean, cold compress on the burn until the pain subside. 3. If the burn area is small, loosely cover it with a sterile gauze pad or bandage. 4. Call the doctor if the victim is a child even if the burn seems mild. **WARNING:** 1. Do not use ice 2. Do not apply butter, grease, powder or any other remedies to the burn

2.Sample: Rule Documentation Form for Emergency Case (Coded Version)

Project: First Aid Advisor
Case: Bleeding
Rule Name: Puncture   **Rule Set:** bleeding(puncture)
Description: First Degree Burn is the mildest of three types of burn, are generally caused by brief skin contact with hot water, steam ,or hot objects or overexposure to the sun. It caused some blistering, swelling, redness and pain.
RULES: IF Burns THEN First degree burn IF Skin reddening AND Burn at outer layer only AND Caused by hot water OR Caused by hot object Or Overexposure to the sun THEN First Degree Burn
Treatment: 5. Remove clothing from burned area immediately. 6. Run cool water over the burned area or hold a clean, cold compress on the burn until the pain subside. 7. If the burn area is small, loosely cover it with a sterile gauze pad or bandage. 8. Call the doctor if the victim is a child even if the burn seems mild. **WARNING:** 3. Do not use ice 4. Do not apply butter, grease, powder or any other remedies to the burn

Sample: Documentation for rule (Natural Language Version)

Project: First Aid Advisor
Case: Bleeding
Rule Name: Puncture **Rule Set:**
Description: A perforation of the skin and tissue made by a sharp object.
Symptoms/sign: 1. Dog bites 2. Bee sting 3. Stepping on a rusty nail 4. Resulting from corkscrew
Treatment: 1. Check the wound – do not remove any penetrating object 2. Apply pressure to stop bleeding 3. Scrubbed thoroughly but gently with soap and water 4. Cover with dressing

3. Sample Form for Fractures(Natural Language Version)

Project: First Aid Advisor

Case: Fractures

Rule Name: Fractures forearm **Rule Set:** fractures(forearm)

Description: Fractures Forearm

Symptom/Sign:

1. Pale, cool, clammy skin
2. Rapid weak pulse
3. Tenderness
4. Loss of power to limb
5. Associated with blood loss
6. Associated with organ damage
7. Nausea
8. Deformity

Treatment:

1. Check for distal pulse, if none – gently traction until pulse return
2. Treat any wound
3. Pad bony prominences
4. Apply adequate splint
5. Secure above and below fracture, secure wrist
6. Reassess pulse of capillary return
7. Elevate injury with arm sling
8. Call 999

Sample Form for Fractures(Coded Version)

Project: First Aid Advisor

Case: Fractures

Rule Name: Fractures forearm **Rule Set:** fractures(forearm)

Description: Fractures Forearm

RULE:

IF Fractures
THEN Forearm

IF Fractures Forearm
AND Deformity
THEN forearm

Treatment:

1. Check for distal pulse, if none – gently traction until pulse return
2. Treat any wound
3. Pad bony prominences
4. Apply adequate splint
5. Secure above and below fracture, secure wrist
6. Reassess pulse of capillary return
7. Elevate injury with arm sling
8. Call 999

4. Sample Form for Choking (Natural Language Version)

Project: First Aid Advisor
Case: Choking
Rule Name: Choking **Rule Set:**
Description: Choking is due to lodgment of a foreign object in the casualty's airways(trachea). In some instances, the object lodges at the epilogittis, but does not actually enter the trachea. Both cases cause initial coughing, the body's reflex action to the dislodge the object.
Symptom/Sign: 1. Difficulty or absence of breathing 2. Inability to speak 3. Grabbing the throat 4. Cyanosis 5. Eventual collapse
Treatment: 1. Position the casualty 2. Deliver four firm slaps between the shoulder blades. 3. Check mouth and clear any obstructions that may have come loose. 4. Repeat four firm slaps between the shoulder and blades 4 if the blockage has not been clear call 99 for an ambulance

Sample Form for Choking (Natural Language Version)

Project: First Aid Advisor

Case: Choking

Rule Name: Choking **Rule Set:**

Description: Choking is due to lodgment of a foreign object in the casualty's airways (trachea). In some instances, the object lodges at the epilogittis, but does not actually enter the trachea. Both cases cause initial coughing, the body's reflex action to the dislodge the object.

Rule:

IF Choking
THEN kes Choking

IF Absence of breathing
AND Inability to speak
And Grabbing the throat
AND Collapse
THEN kes Choking

Treatment:

5. Position the casualty
6. Deliver four firm slaps between the shoulder blades.
7. Check mouth and clear any obstructions that may have come loose.
8. Repeat four firm slaps between the shoulder and blades 4 if the blockage has not been clear call 99 for an ambulance

5. Sample: Module Testing

Project Title: FIRST AID ADVISOR
Test: Module Testing
Date: 10 January 2001 Location: Lab
Attendees: Wilhelmina (developer)
Objective: To test all the modules and to ensure its works
Problem Discussion:
Session Trace: - It works well. Repair the control for Dialog Bleeding and Fractures
System Evaluation:
Interface Evaluation: 1. Easy to use 2. User friendly
Comment: The coding needs to be updated therefore the system can be more stable. Check the coding for control and searching. Check the syntax.

6 Sample: Integration Testing

Project Title: FIRST AID ADVISOR

Test: Integration

Date: 12 January 2001 Location: Lab

Attendees: Wilhelmina (developer)

Objective:

1. To makes sure the modules can be integrated

Problem Discussion:
Linking the form dialog to dialog.

Session Trace: -
Syntax error in the some or the dialog. The control button

System Evaluation:

Interface Evaluation:

1. Easy to handle
2. User friendly

Comment:
Recheck the coding for syntax error. Debug the system.

7. Sample: System Testing

Project Title: FIRST AID ADVISOR
Test: System Testing
Date: 17 January 2001 Location: Lab
Attendees: Mastura, Sharizal and Moktar (members of the faculty)
Objective: 2. To test whether the system is user friendly 3. To test whether the knowledge is reliable 4. To test whether the system is timely response
Problem Discussion: The knowledge is limited to the scope. System still unstable.
Session Trace: - The knowledge base needs to be expand. Repair the coding.
System Evaluation: 1. System is reliable
Interface Evaluation: 3. Easy to use 4. User friendly
Comment: The coding needs to be updated therefore the system can be more stable. Check the coding for control and searching.

4. Sample of User Acceptance form

Feature	Evaluation	Comment
Ease of use		
Starting the system	Yes	
Obtaining the explanation	Need to be improved	Has to go through intermediate interface
Help facilities	Yes	
Interface technique	Yes	
Exiting the system	Yes	
Nature of questions		
Clarity of terms	Yes	
Answer complete	Yes	Insert yes or not in the checkbox
Clarity of questions	Yes	Clear
Presentation of result		
Easy to follow	Yes	
Complete	Yes	
System utilities		
Easy to access	Yes	
Complete	Yes	
General consideration		
Speed of the system	Yes	
System is useful	Yes	

Provide any general comments about the system: Add some feature like utility to print the knowledge

Appendix B: Knowledge base

```
%**************************************************
% Wilhelmina Ngelambai
% WEK 98145
% WXES 3182
% This is knowledge base for FIRST AID ADVISOR v1.0
%**************************************************

% list of cases and symptoms

kes(bleeding).]
kes(burns).
Kes(choking).
Kes(poisoning).
Kes(fractures).
%category: Bleeding

bleeding(puncture).
bleeding(incisions).
bleeding(laceration).
bleeding(amputation).
bleeding(is_nosebleed).

% category: Fractured
fractures(case_fractured_forearm).
fractures(case_fractured_upperarm).
fractures(case_fractured_pelvis).
fractures(case_fractured_leg).

% category: burns
burns(first_degree_burn).
burns(second_degree_burn).
burns(third_degree_burn).

% category: choking
choking(choking).

% category: poissoning
poisoning(poisoning).

% here are the symtoms and sign for every case

% symptoms for puncture

punture(external_bleeding,it_is_a_dog_bites).
punture(external_bleeding,it_is_a_bee_sting).
punture(external_bleeding,stepping_on_a_rusty_nail).
```

punture(external_bleeding,wound_are_perforation).
punture(external_bleeding,wound_are_perforation,due_to_anything_from_corkscrew_to_bullet).

% systoms for incisions

incisions(external_bleeding,slicing_by_a_sharp_knife).
incisions(external_bleeding,slicing_by_a_sharp_metal).
incisions(external_bleeding,cut_cause_by_a_broken_glass).

% symtoms far laceration

laceration(external_bleeding,tearing_tissue,it_is_a_deep_cut,wound_cause_by_barbed_wire).
laceration(external_bleeding,tearing_tissue,it_is_a_deep_cut).
laceration(external_bleeding,tearing_tissue).
laceration(external_bleeding,wound_cause_by_barbed_wire).

% symtoms for abrasion

abrasion(external_bleeding,tearing_issue,contain_foreign_matter,skin_scraped_away,result_by_highway_accident).
abrasion(external_bleeding,tearing_tissue,contain_foreign_matter,skin_scrapped_away,result_by_street_accident).
abrasion(external_bleeding,it_is_a_large_cut,skin_scrapped_away).

abrasion(external_bleeding,skin_scrapped_away).
abrasion(external_bleeding,tearing_tissue,result_by_street_accident,contain_foreign_matter).
abrasion(external_bleeding,result_by_street_accident).
abrasion(external_bleeding,tearing_tissue,result_by_street_accident).
abrasion(external_bleeding,tearing_tissue,result_by_highway_accident).
abrasion(external_bleeding,skin_scrapped-away,result_by_highway_accident).

% symtoms for amputation

amputation(external_bleeding,loss_any_part_of_the_body).

% symtoms for nosebleed

is_nosebleed(nosebledd).
is_nosebleed(internal_bleeding,nosebleed).

% symtoms for fractured forearm

```
case_fractured_forearm(fractured_forearm,pale_skin,loss_of_power_to_limb,wound_and_blood_loss,deformity,pain_at_the_site,associated_with_organ_damage,nausea).
case_fractured_forearm(fractured_forearm).
case_fractured_forearm,deformity,pain_at_the_site_associated_with_organ_damage).
case_fractured_forearm(fractured_forearm,deformity,pain_at_the_site).

% symptoms for fractured upperarm

case_fractured_upperarm(fractured_upperarm,pale_skin,loss_of_power_to_limb,wound_and_blood_loss,deformity,pain_at_the_site,associated_with_organ_damage,nausea).
case_fractured_upperarm(fractured_upperarm).
case_fractured_upperarm(fractured_upperarm,deformity,pain_at_the_site,associated_with_organ_damage).
case_fractured_upperarm(fractured_upperarm,deformity,pain_at_the_site).

% symptoms for fractured pelvis

case _factured_pelvis(fractured_pelvis,pale_skin,loss-of_power_to_climb,wound_and_blood_loss,deformity,pain_at_the_site,associated_with_organ_damage,nausea).
case_fractured_pelvis(fractured_pelvis).
case_fractured_pelvis(fractured_pelvis,deformity,pain_at_the_site,
associated_with_organ_damage).
case_fractured_pelvis(fractured_pelvis,deformity,pain_at_the_site).

% Symptoms for fractured leg

case _factured_leg(fractured_leg,pale_skin,loss-of_power_to_climb,wound_and_blood_loss,deformity,pain_at_the_site,associated_with_organ_damage,nausea).
case_fractured_leg(fractured_leg).
case_fractured_leg(fractured_leg,deformity,pain_at_the_site,
associated_with_organ_damage).
case_fractured_leg(fractured_leg,deformity,pain_at_the_site).

%symptoms for first degree burn

first_degree_burn(skin_reddening,burn_at_outer_layer_only,cause_by_hot_water).
first_degree_burn(skin_reddening,burn_at_outer_layer_only,over_exposure_to_the_sun).

%symptoms for second_degree_burn
```

second_degree_burn(contact_with_chemical,skin_blistering,damage_at_the_de
eper_layer).
second_degree_burn(contact_wit_chemical,skin_blistering).
second_degree_burn(contact_with_chemical,damage_at_the_deeper_layer).
second_degree_burn(contact_with_chemical,skin_reddening,skin_blistering,da
mage_at_the_deeper_layer).

%symptoms for third degree burn

third_degree_burn(contact_with_chemical,skin_clor_is_whitish_or_blackened).
third_degree_burn(skin_color_is_whitish_or_blackened,damage_at_the_large_
area).
third_degree_burn(contact_with_chemical,damage_at_the_large_area).

%symptoms for choking

choking(difficulty_in_breathing,inability_to_speak,grabbing_the_throat,victim_co
llapse).
choking(difficulty_in_breathing,grabbing_throat).
choking(inability_to_speak).

%symptoms for poisoning

poisoning(victim_is_vomiting,burns_around_mouth).
poisoning(victim_is_vomitting,burns_around_mouth,blurred_vision).
poisoning(victim_vomitting).
poisoning(drowsiness_or_unconcsious,breathing_difficulty).

Appendix C:
User Manual

First Aid Advisor Statement

This system has been build using Visual Prolog Personal Edition Version 5.2, under freeware license, and Prolog Development Center (PDC) grants the license for noncommercial and educational purpose only. The use of the Personal Edition is restricted to learning the Visual Prolog on a private basis or for teaching Prolog on Universities and other educational establishments

PDC allows the user to:

- use the Personal Edition at home for learning Visual Prolog.
- use the Personal Edition at educational institutions for teaching and learning Visual Prolog.
- make as many copies of the Visual Prolog Personal Edition as you wish; give unmodified copies of the Visual Prolog Personal Edition installation program to anyone; and distribute the Visual Prolog Personal Edition installation program in its unmodified form via electronic means.

Therefore, PDC not permitted user to:

- distribute or give away any executables created with the Personal Edition.
- use the Personal Edition for any (direct or indirect) commercial purpose.
- remove or attempt to remove the Personal Edition banner from the Prolog system or any executable it creates.

System Requirement:

- Personal computer
- Operating system:

  Windows® XP Home Edition or XP Professional, 200 Professional, Millennium Edition, Windows® 9x.

- Winzip software
- Intel Pentium® 133 MHz or higher processor for Windows® 9x,NT
- Intel® Celeron 366 or higher for Windows 9x
- 32 MB or RAM at least
- 10 MB of available hard disk space (at least)
- CD-ROM drive or Floppy Disk 1.44 MB

Input:

- Keyboard Microsoft Natural Keyboard
- Standard serial Mouse

Monitor

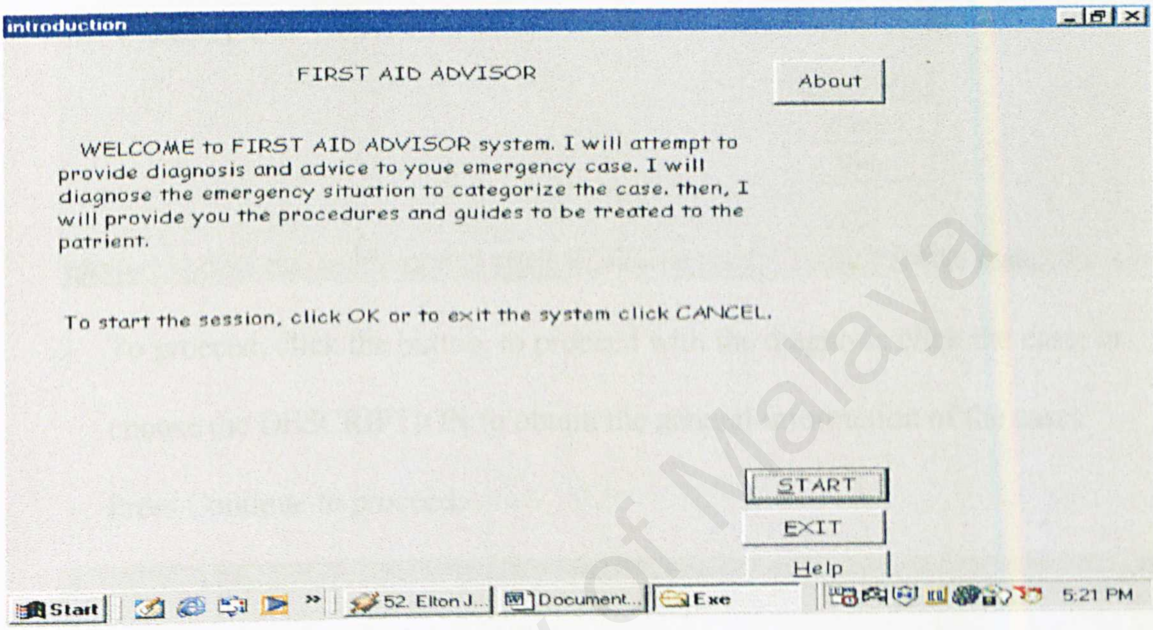
- Plug and Play monitor (1600 x 1200 Max resolution)
- Sis 620 (800 x 600 in 64k Colors)
- 14" Monitor (at least)

Software Installation:

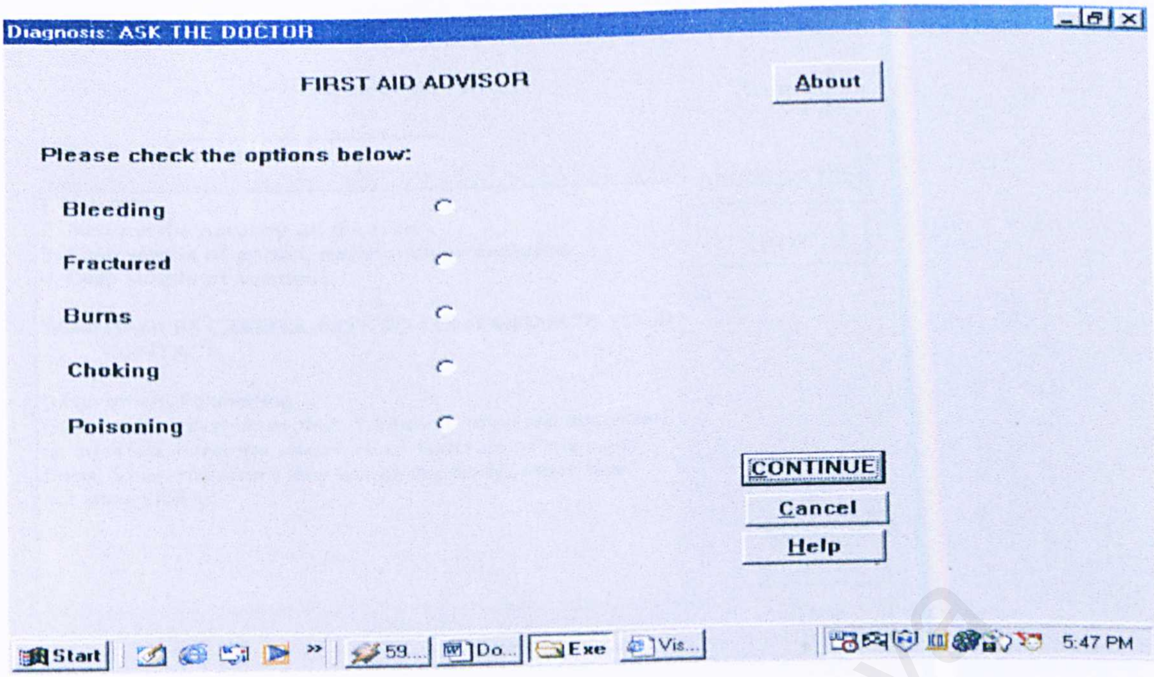
- Just start up the pc, create a folder, extracts the "**advisor.zip**" to the created folder.Double click the theses folder, then find the Exe folder. Open the exe folder. Double click **faa.exe** to run the system.

Using the system:

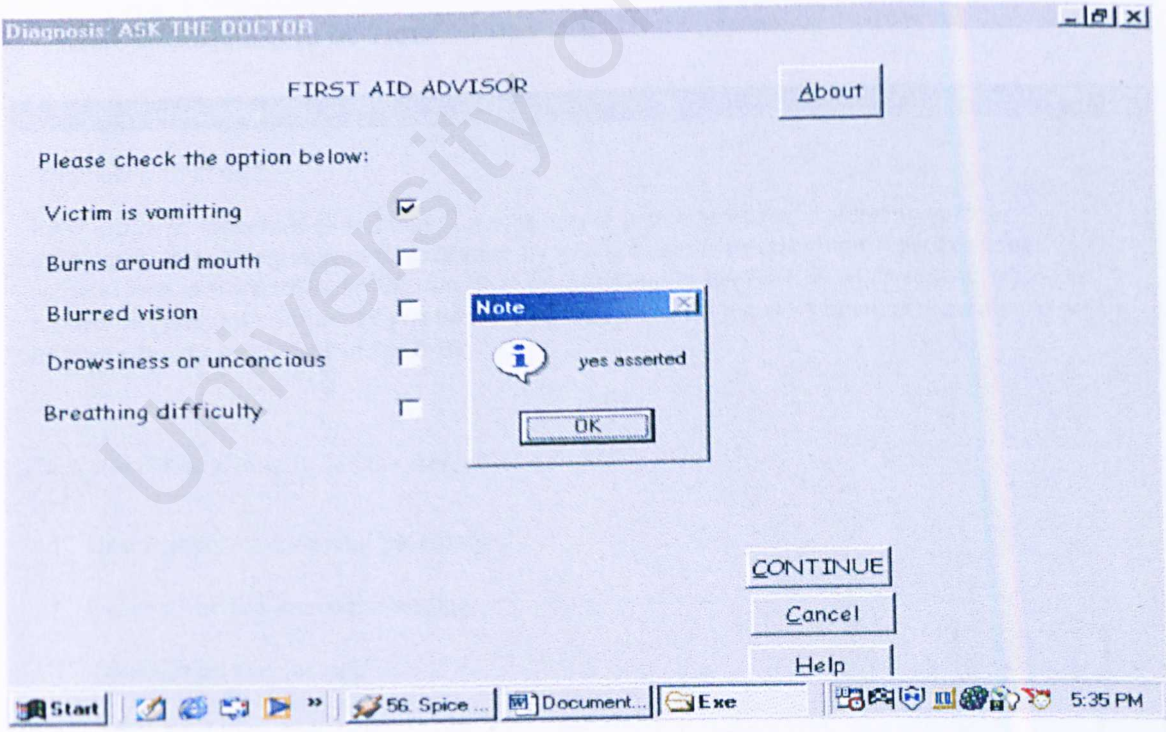
1. The introduction screen will appear. It will give the instruction to guide the user on how to use the system.



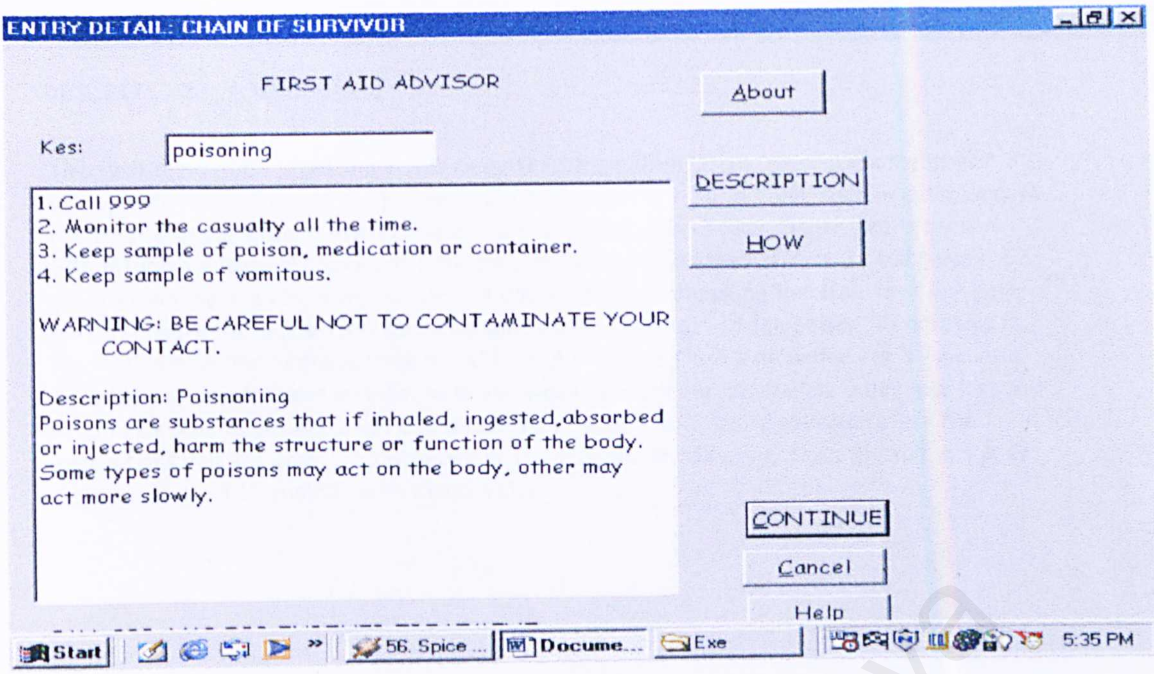
2. The second screen is the intermediate screen, is where the user has to click the option on what to do, choose START to start the session and EXIT to exit the system completely.



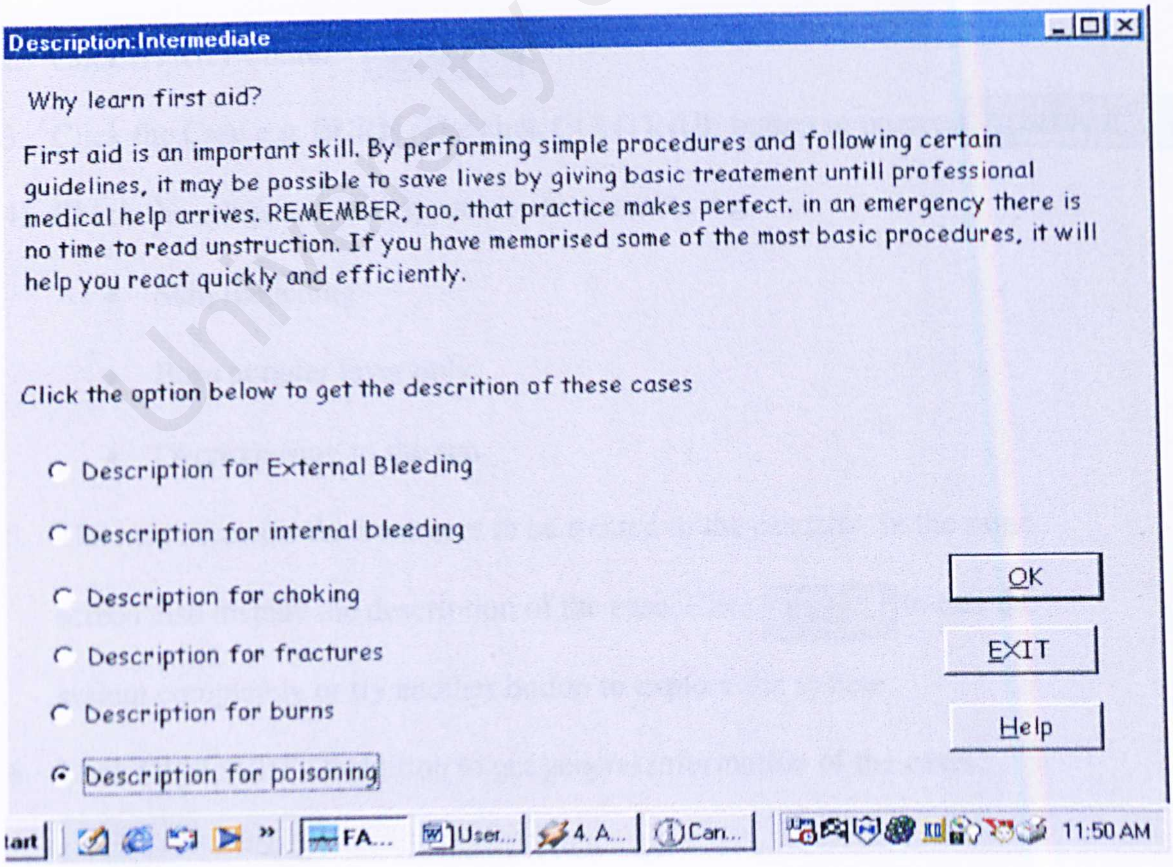
3. To proceed, click the button, to proceed with the diagnosis click the case; or choose the DESCRIPTION to obtain the general information of the cases. Press Continue to proceed.



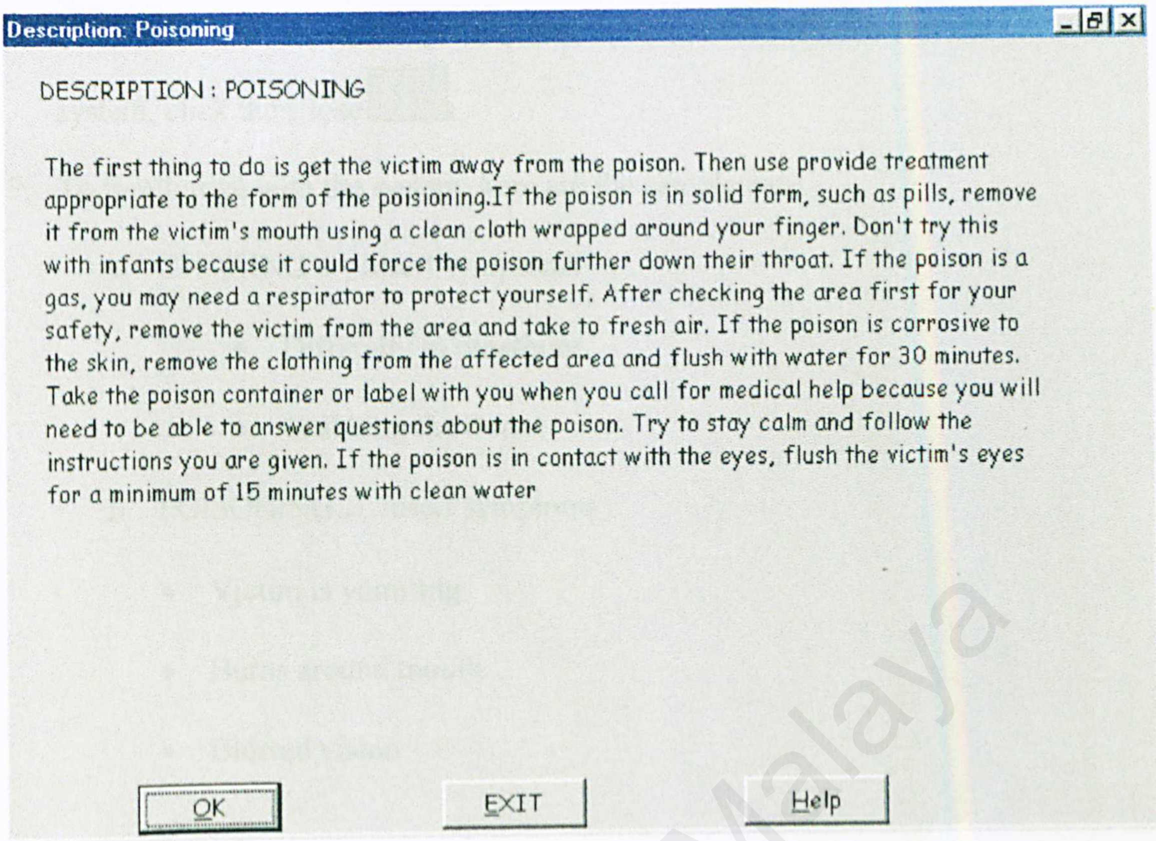
4. Click the symptoms that appears to the casualty, and the conclusion will appear.



5. To view the general information, click DESCRIPTION button, The choose what information to be view:



2. Click OK to view the information.



Tutorial:

1. Run the system. "FAA.exe"
2. Click START button START
3. Click the Case e.g. BURN. The click CONTINUE button to proceed CONTINUE
4. Check the symptoms that appear to the casualty, e.g.
    - Skin reddening
    - Burn at outer layer only
    - Overexposure to the sun
5. The next screen is the treatment to be treated to the casualty. In the same screen also display the description of the case. Click EXIT to exit the system completely or try another button to explore the system.
6. Click DESCRIPTION button to get general information of the cases
   DESCRIPTION

7. Choose the option BURNS.

8. The screen will display the general description of BURNS. To exit the system, click the Close [X].

9. To familiarize with the system, here are the sample case

    a. CHOKING ↵ , insert symptoms

        - Difficulty in breathing
        - Grabbing the throat

    b. POISONING ↵, insert symptoms

        - Victim is vomiting
        - Burns around mouth
        - Blurred vision