CHILI DISEASE SYSTEM (CDS)

NUR SYAHKINA BINTI SAHABE MARUBI

A report submitted in partial fulfillment

of the requirements for the award of the degree of

Bachelor of Computer Science (Software Engineering)

Faculty of Systems Computer & Software Engineering

University Malaysia Pahang

JUN, 2012

DECLARATION

I declare that this thesis entitled "Chili Disease System (CDS) "is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : .....

Name : Nur Syahkina Binti Sahabe Marubi

Date : .....

SUPERVISOR'S DECLARATION

"I hereby declare that I have read this thesis and in my opinion this thesis is sufficient in terms of scope and quality for the award of the degree of Bachelor of Computer Science (Software Engineering)"

Signature : .....

Supervisor : Wan Muhammad Syahrir Bin Wan Hussin

Date : .....

ACKNOWLEDGEMENT

Firstly, I would like to thank to the greatest Allah for giving me the strength and solutions to all the problems I faced throughout the accomplishment of this undergraduate project thesis. With Allah's grace, my project is now complete and has been fully documented in this thesis.

I would like to express my sincere appreciation to my project supervisor, Mr. Wan Muhammad Syahrir bin Wan Hussin who had contributed so much of time, ideas and guidance in order to help me completing the whole undergraduate project.

This project would be impossible without the help and support from many peoples, especially my dearest friends who were willing to share their knowledge and ideas that turned out well in this project.

Finally, my deepest gratitude goes to my whole family, which has given me support mentally and financially, and for their understanding, encouragement and support, towards the completion of my project.

ABSTRACT

Chili disease system is a web-based expert system that is developed to help staff of Jabatan Pertanian Jubli Perak identify and diagnose the disease of chili. With the useful information provided that is symptoms, and the pesticide to help prevent the diseases. Chili Disease System (CDS) help user to get the information about the disease that may attack the chili, and the symptoms. There are various types of disease of this crop that affected the product of chili. Naturally, the disease may cause by the insect pests, or maybe other symptoms. So, the changes that happened at the fruits, stem, or leaf can recognize the diseases. Current system to identify the problem was actually by doing observations regularly. The data uses are based on written document and logical thinking, but the data will probably lose if the documents are not sorted properly. By using forward chaining, the system implements a rule-based expert system technique to match and fired the fact with user input to get information about the disease of chili. This system applied rapid application development methodology for the system development process. Other than that, this application can imitate human thought and making decision almost accurate like the expert.

ABSTRAK

Sistem pengesanan penyakit cili merupakan sebuah sistem yang berasaskan web dimana ia dibangunkan untuk membantu kakitangan Jabatan Pertanian Jubli Perak bagi mengenal pasti penyakit cili yang dihadapi. Dengan sedikit bantuan melalui soalan-soalan yang di kemukakan dalam sistem ini, Penyakit Sistem Cili (CDS) ini dapat membantu pengguna mengetahui sebab-sebab serangan penyakit tersebut berlaku. Terdapat pelbagai jenis penyakit yang boleh menjejaskan produk cili ini. Contohnya, penyakit ini disebabkan oleh serangga perosak, atau mungkin gejala-gejala lain. Tidak lain juga, perubahan-perubahan yang berlaku pada buah-buahan, batang, atau daun juga dapat membantu mengenalpasti penyakit meyerang tanaman cili tersebut. Di samping itu, cara pengesanan penyakit yang sedia ada untuk mengenal pasti masalah ini adalah dengan melakukan pemerhatian yang kerap terhadap tanaman tersebut. Penggunaan data adalah berdasarkan dokumen bertulis dan pemikiran logik, tetapi data tersebut berkemungkinan akan hilang atau tidak tersusun jika dokumen tidak disimpan dengan betul. Dengan adanya teknologi yang semakin berkembang, sistem ini menggunakan teknik pangkalan sistem pakar dengan meggunakan perantaian ke hadapan dimana peraturan teknik ini adalah dengan memadankan dan menyesuaikan fakta melalui input pengguna untuk mendapatkan maklumat yang tepat mengenai penyakit cili. Tambahan pula, sistem ini boleh meniru pemikiran manusia dan membuat keputusan yang hampir tepat seperti pakar.

TABLE OF CONTENTS

2. LITERATURE REVIEW

3. METHODOLOGY

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Chili *(Capsicum frutescens)* is a species from a genus of flowering plants in the family of Solanaceae Family and popular in Malaysian Cuisine. This *Capsicum frutescens* can be annual or short-lived perennial plants [1].

Chili was originated from Mexico and South America but adopted to Malaysian Tropical Climate many decades ago. About 14,560 hectare of chili was grown in Malaysia annually in major producing state such as Johor, Perak and Kelantan in Commercial Scale. Chili was suitable in temperature from 20-30 Celsius with rainfall between 1,500 mm - 2,000 mm monthly and it can adapt to pH between 5.5 - 6.8 on mineral soil, peat or Bris Soil (with proper irrigation system). Red Chili or Green Chili is the product depends on harvesting period and different market needs. Study shows that Malaysian took 33,000 ton/year of fresh chili annually whereby the local production around 23,000 tan/year. Malaysia also import chili from Thailand (Cili Padi Variety) about 5,000 kg daily through Bukit Kayu Hitam Border.

Nowadays, there are million acres of red peppers are grown annually and one-quarter population of world uses it daily, primarily for culinary purpose. For some of chili farming, there are diseases that exist in this chili plants in which it

cannot know the cause and solution. A special attention must be given to these pest and disease problems. If the cause is known, the possibility of the solution is found relatively late. It becomes an inconvenience to identify what type of disease had attack this crop. The farmers need to know the disease as soon as possible so that they can handle it immediately.

1.2 Problem Statement

There is various type of disease of this crop that affected the product of Capsicum f. or chili. Naturally, the disease may cause by the insect pests, or maybe other symptoms. So, the changes that happened at the fruits, stem, or leaf can recognize the diseases.

Current system to identify the problem was actually by doing observations regularly. The data uses are based on written document and logical thinking, but the data will probably lose if the documents are not sorted properly.

By using expert system based on rule based, all the data will kept in database. This data can be update depends on the expert request. Other than that, this application can imitates human thought and making decision almost accurate like the expert.

1.3 Objectives

In this system development, the objectives are:

i. To develop a web-based prototype application in finding the Capsicum f. diseases.
ii. To implement the rule based expert system in the application that capable to detect diseases of Capsicum frustences.

1.4 Project Scopes

The scope of the system development:

i. This system is developed for Pejabat Pertanian Jubli Perak, Indera Mahkota, Kuantan Pahang.
ii. The users of this system are the farmers and expert officers from Pejabat Pertanian Jubli Perak, Indera Mahkota, Kuantan Pahang.
iii. This system is web-based application with rule based methodology applied.
iv. All the data and information from department of Agriculture and farmers who own and works at chili orchard.
v. The system is developed for identifying the type of disease based on data collected.

1.5 Thesis Organization

1.5.1 Chapter One: Introduction

Chapter One will further explain about the background of the problem, problem statement, project scope and significant. This is important to give a big representation about the project. It also elaborates on the main objectives of the project to achieve. At the end of this chapter few points is highlighted as the benefits of the project.

1.5.2 Chapter Two: Literature Review

Chapter Two will review the technique and existing similar project for use as a guidance or guidance in developing the Capsicum disease prototype. The literature review is

essential in this project as it will provide evidence and a proven conclusion to support this project.

1.5.3 Chapter Three: Methodology

Methodology used in this research will be explained in Chapter Three. This chapter will describe overall of the process in developing the project. It has eight phases in methodology within time given that need to follow.

1.5.4 Chapter Four: Implementation

Implementation describes the architecture of the system development such as the database structure and the table's design which are used SQL command to insert data into the database that has developed.

1.5.5 Chapter Five: Result and Discussion

In the Chapter Five, it discussed the result and the constraint in completing the development for the project. In this chapter also provide a comparison between these projects with the available Capsicum Disease prototype.

1.5.6 Chapter Five: Conclusions

Finally, Chapter Five divided into three main sections which are conclusion, recommendation and summary. It also briefly concludes remarks on what may be considered a successful thesis project.

CHAPTER 2

LITERATURE REVIEW

2.0 Introduction

In this chapter, the discussion will highlight several literature reviews about the current application of expert system, and a chili or *Capsicum* disease that is related with this area.

2.1 Expert System

Expert system is a program that attempts to mimic human expertise by applying inference methods to a specific body of knowledge [2]. This body of knowledge is called the domain of Expert System(ES). The three major components of ES are: Knowledge base (KB), inference engine (IE), and user interface (UI). For better interaction with users, an ES should preferably contain an explanation subsystem component or justifier [3]. Expert systems are the most widely use commercial application coming out of artificial intelligence [4]. Artificial Intelligent is the system that converts the knowledge of an expert trough a computer by applying a software code. AI programs that achieve expert-level

competence in solving problems in task areas by bringing to bear a body of knowledge about specific tasks are called knowledge-based or expert systems. Often, the term expert system is reserved for programs whose knowledge base contains the knowledge used by human experts, in contrast to knowledge gathered from textbooks or non-experts [5]. Expert systems are different from conventional computer programs because the tasks of expert systems have no algorithmic solutions and expert systems make conclusions based on not complete or uncertain information.

There are several expert system development processes [6]:

- Determining requirements
- Identifying experts
- Constructing expert system components
- Implementing results
- Maintenance and review

Expert systems can play an important role in developing area for a variety of reasons [7]:

- Expert systems are always and instantly available and always perform at the same level of expertise.
- Expert systems have direct and instantaneous access to the necessary databases and are not limited, biased, and imperfect recollection of human experts.
- Expert systems are logical, objective by emotional arguments that might influence a human expert.

- Expert systems do not forget or make mathematical errors. There are three types of expert system devised in solving real world problems. They are:
    i. Rule-based expert system
    ii. Model-based expert system
    iii. Case-based reasoning

2.2.1 Rule-Based Expert System

Rule-based system is a set of "if-then" statements that uses a set of assertions, to which rules on how to act upon those assertions are created. In software development, rule-based systems can be used to create software that will provide an answer to a problem in place of a human expert. Other than that, it consists of a *rule-base* (permanent data), an *inference engine* (process), and a workspace or working memory (temporary data). Not part of the basic reasoning process, but essential to applications, is the *user interface* [8].

Rule-Based is a simple model that can be adapted to any number of problems [9]. A rule-based system has its strengths as well as weaknesses that must be considered before deciding if it is the suitable technique to use for a given problem.

Overall, rule-based systems are only feasible for problems, which any knowledge in the problem area can be written in the form of if-then rules.

2.2.2 Component of Expert System Development Team

Figure 2.1 shows the person involved in expert system development. There are five members or component in the expert system development team. There were domain expert, knowledge engineer, programmer, project manager and end-user.

The domain expert is a knowledge and skilled person capable of solving problems in a specific area or domain. This person has the greatest expertise in a given domain. An expert must be able to communicate his or her knowledge and willing to participate in the expert system development and also commit a substantial amount of time to the project [10].

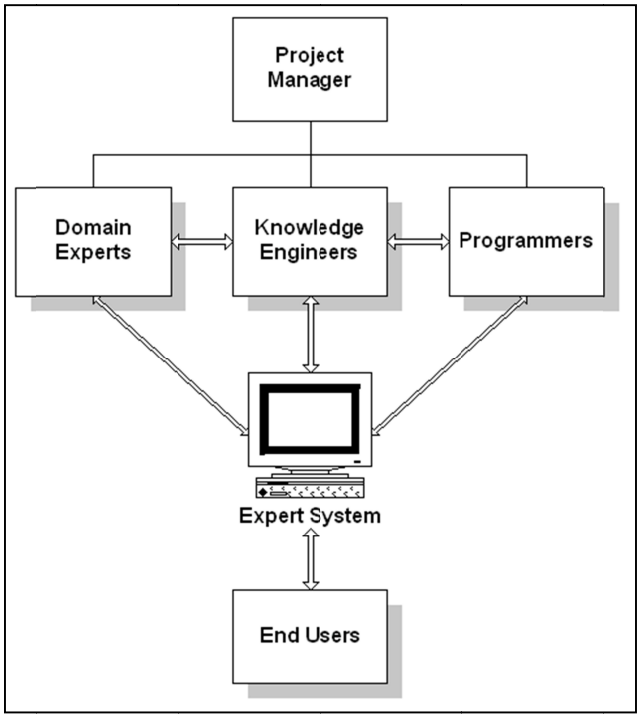


Figure 2.1: Expert System Development Team

A knowledge engineer is someone who is capable of designing, building and testing an expert system. A knowledge engineer need for selecting an appropriate task for the expert system as their responsibility. This person will interview the domain expert to gain knowledge solution on how the problem being solved. It is also known as artificial intelligence specialist responsible for the development of knowledge based application. Duties usually include knowledge elicitation, representation, and coding [11].

The programmer is responsible for actual programming describing the domain knowledge in terms that a computer can understand. This person needs some skills symbolically in programming language such as Prolog, LISP, and OPSS and also some experience in the application to be built.

The project manager is the leader of the expert system development team, responsible for keeping the project on track. He or she makes that all deliverables and milestones are met, interacts with the expert, knowledge engineer, programmer and end-user.

The end-user, often called just the user, which a person who interact or use the application of expert system after developed.

2.2.3 Structure of a Rule-based Expert System

Rule-based systems are a relatively simple model that can be adapted to any number of problems. The rule-based system itself uses a simple technique: It starts with a rule-base, which contains all of the appropriate knowledge encoded into If-Then rules, and a working memory, which may or may not initially contain any data, assertions or initially known information. The system examines all the rule conditions (IF) and determines a subset, the conflict set, of the rules whose conditions are satisfied based on the working memory. Of this conflict set,

one of those rules is triggered (fired). Which one is chosen is based on a conflict resolution strategy. When the rule is fired, any actions specified in its THEN clause are carried out. These actions can modify the working memory, the rule-base itself, or do just about anything else the system programmer decides to include. This loop of firing rules and performing actions continues until one of two conditions are met: there are no more rules whose conditions are satisfied or a rule is fired whose action specifies the program should terminate [12]. Szolovits (1988) stated that many of the early efforts to apply artificial intelligence methods to real problems, including medical reasoning, have primarily used rule-based systems. Such programs are typically easy to create, because their knowledge is catalogued in the form of "if ... then..." rules used in chains of deduction to reach a conclusion, hi many relatively well-constrained domains rule based programs have begin to show skilled behavior Rule-based system is one of the techniques that always been used for medical diagnosis. The artificial intelligence now is rapidly growth where a lot of approaches that combines with the ruled-based system such as Neural Network or Fuzzy Logic to produce a more complex diagnosis support system. There are also the upgraded versions of rule-based system such as Case-Based and Model-Based which are similar to rule-based. For example, computer-implemented Pain Management Advisor (PMA) based on an interactive knowledge-based expert system. In the abstract of the journal, Brynjestad (1999) wrote that the PMA facilitates the diagnosis and treatment of acute and chronic pain. The PMA system includes three key components: a knowledge-based component for use by the primary care providers; an inductive learning component and case-based reasoning technology for use by the chronic pain patient; and a third component used by primary care providers to consult with pain specialists and to update/modify pain management algorithms.

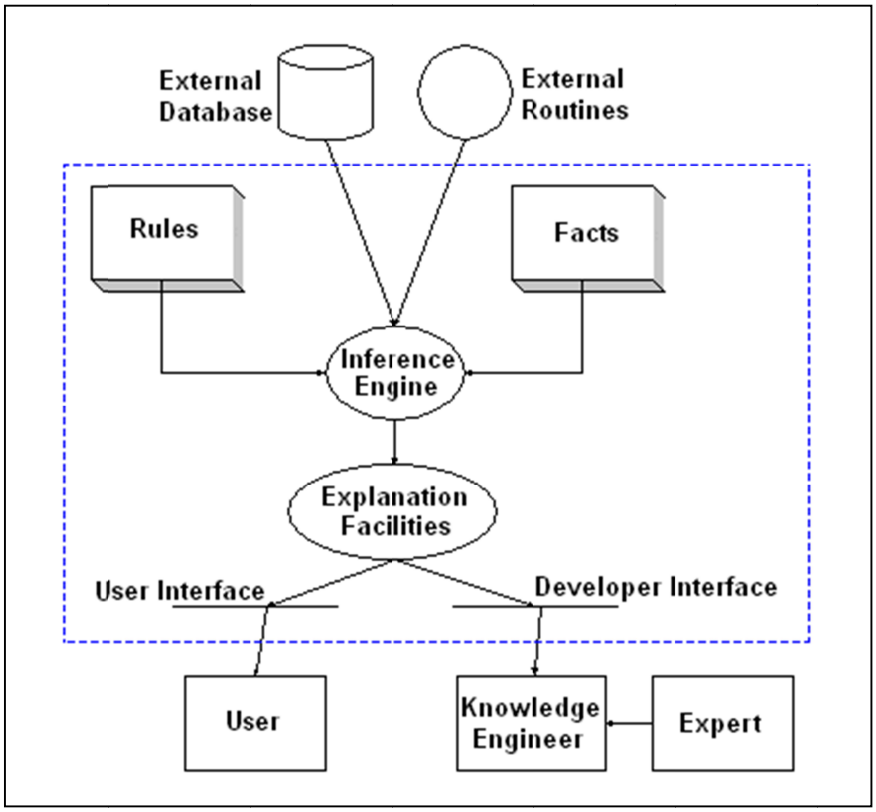


Figure 2.2: Basic Structure Rule-Based Expert System [10]

In figure 2.2, it showed the important components that make up rule-based expert system. The first component is knowledge base here it is contains domain knowledge useful for problem solving. In a rule-based expert system, the knowledge is represented as a set of rules. Each if-then rule specifies a relation, recommendation, irective, strategy or heuristic. When condition (if) part of a rule is satisfied, rule is said to fire and the action part is executed. The second component is database includes a set of facts used to match against the IF condition) parts of rules stored in the knowledge, base. The next component is

inference engine that can carries out reasoning to reach a solution. It links rules in knowledge base with facts in database. There are two types of inference method which are forward chaining and backward chaining. The fourth component is explanation facilities. The explanation facilities showed how a particular conclusion is reached and why a specific fact is needed. The last component is user interface where the communication between a user seeking a solution to problem and an expert system. According to [12], to create a rule-based system for a given problem, you must have (or create) the following:

i. A set of facts to represent the initial working memory. This should be anything relevant to the beginning state of the system,
ii. A set of rules. This should encompass any and all actions that should be taken within the scope of a problem, but nothing irrelevant. The number of rules in the system can affect its performance, so you don't want any that aren't needed.
iii. A condition that determines that a solution has been found or that none exists. This is necessary to terminate some rule-based systems that find themselves in infinite loops otherwise

2.2.4 Forward Chaining

Rule-based systems, as defined above, are adaptable to a variety of problems. In some problems, information is provided with the rules and the AI follows them to see where they lead. An example of this is a medical diagnosis in which the problem is to diagnose the underlying disease based on a set of symptoms (the working memory). A problem of this nature is solved using a forward-chaining, data-driven, system that compares data in the working memory against the conditions (IF parts) of the rules and determines which rules to fire [12].

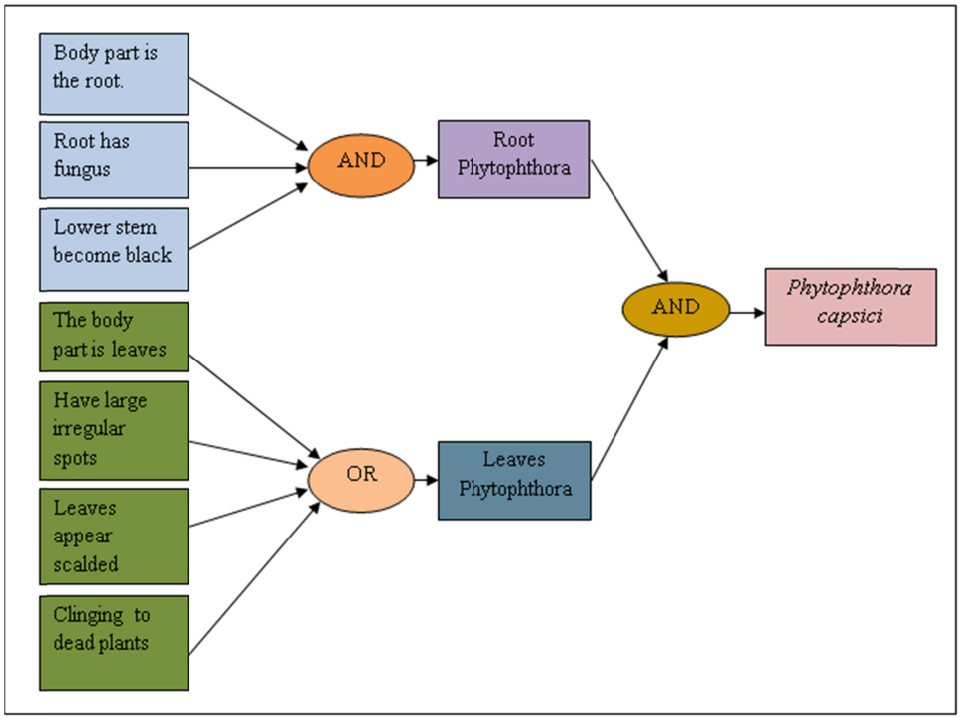


Figure 2.3: Forward chaining process

It starts with the known data and works forward, matching the facts from the database with production rules from the rule base until no further rules can be fired [10].

2.2.5 Backward Chaining

In other problems, a goal is specified and the AI must find a way to achieve that specified goal. For example, if there is an epidemic of a certain disease, this AI could presume a given individual had the disease and attempt to determine if its diagnosis is correct based on available information. A backward-chaining, goal-driven, system accomplishes this. To do this, the system looks for the action in the THEN clause of the rules that matches the specified goal. In other words, it looks for the rules that can produce this goal. If a rule is found and fired, it takes

each of that rule's conditions as goals and continues until either the available data satisfies all of the goals or there are no more rules that match [12].

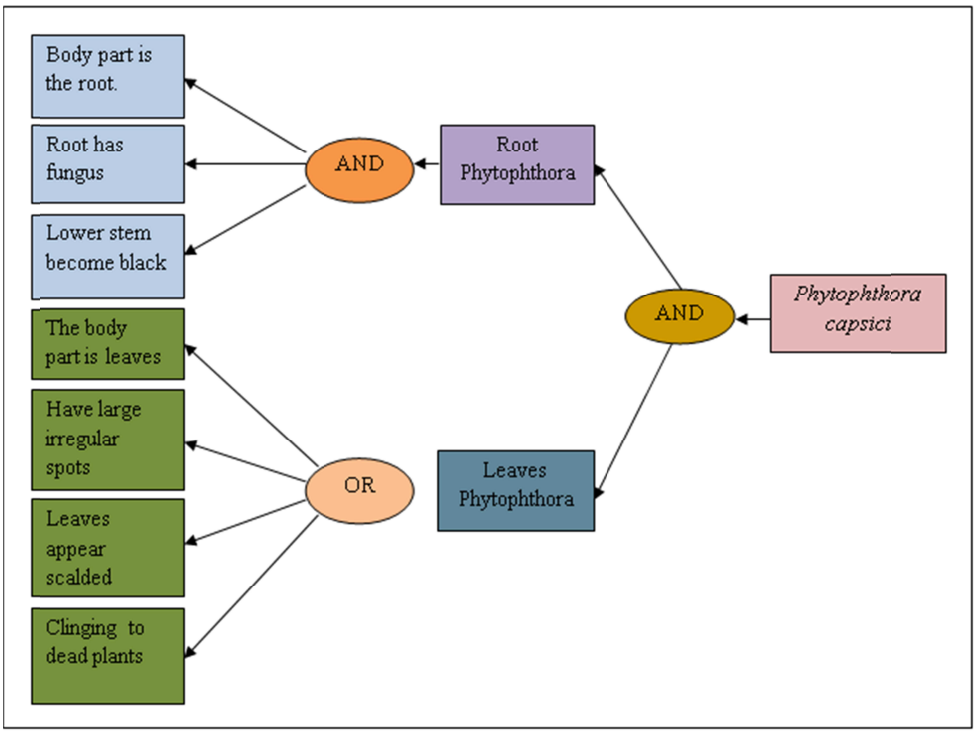


Figure 2.4: Backward chaining process

2.3 Chili or Capsicum f.

Chili rich with nutrient content and reported that every 100 gm chili contain Carotene Beta (2,730 microgram), Phosphorus (80 mg), Niacin (0.7 mg), Vitamin B1, B2, C and others as shown on table 2.0. The most popular lowland grown chili in Malaysia was MC 10 Variety. Farmers grow this variety commercially as mixed crop with water melon.

Chili needs seedling preparation in nursery for 40 days before transplanting (About 8,600 seeds/ha).

Table 2.1: MARDI

Content	Total
Protein (gm)	2.8
Carbohydrate (gm)	9.5
Fat	0.7
Calcium (gm)	15.0
Iron (mg)	1.8
Phosphorus (mg)	80.0
Niacin (mg)	0.7

Pest and diseases are common in chili and need to control from early stage. Among regular pests are Leaf Mites (Aphis spp) which can cause damage to the leaves with a virus carrier and to controlling it by spraying Chloropyrifos up to 2.5 month period. Trips spp, is a virus carrier that cause damage at leaves to the tree. Other pests are Fruit borer (Helicoverpa armigera), Fruit Fly (Bactocera spp) Red Mites (Trtranychus spp) and Snail (Achantina filica). Common diseases in chili are Fruit Antracnose (Collectotricum capsici), Soft, Choanephora cucurbitarum, Leaf Spot (Cercosporrot (Sclerotium rolfsii) a capsici), bacterial

Wilt (Ralstonis solanacerum), Leaf Mozaic (Virus Mozaic). All these type of disease and pests can be control by chemical or IPM (Integrated Pest Management).

IPM (Integrated Pest Management) is basically a strong pressure on the process of integrating or uniting more than one pest control methods. Integration is not only with two or three ways of control. In theory, the integration process can produce positive effects, negative or no effect was dependent on environmental conditions and the characteristics of the control itself. Total number of pests will be lower than the amount expected after the integrated two-way control when using one of these two ways. This integration process has resulted in positive effects and negative effects of pesticides where the total number is much higher than the control results alone. As a result of this integration process, it was found that the number is equal to the number of pests if the negative and positive effects do not occur. Furthermore, the study said that the integration of three or four way control can produce a variety of ways are more complicated.

At University Putra Malaysia, IPM research project has successfully demonstrated that the integration of three way control can control aphid and viruses without the use of pesticides. How kultura, biological and physical successfully integrated with the planting of corn and chili can reduce the population of aphid.

2.3.1 Chili or Capsicum f. diseases [13]

As shown at Table 2.2, it is some common diseases that may attack chili crops either at the fruit, leaves, and roots part. From the symptom given, we can know what type of disease that attacks the crops.

Table 2.2: The Chili (Capsicum f.) symptom

Diseases	Sample Picture	Symptoms
Blossom End Rot		the fruit is discolored, sunken have blossom end with dark have water soaked region have terminal cells die on developing pod
Herbicicde Injury		The leaves form narrowed laminae have prominent raised veins have leaf margin become waxy have flower drop have adventitious roots occur
Erwinia carotovora pv. carotovora		IF cells a watersoaked AND cells are soft AND cells watery AND cells slimy gray AND cells slimy brown AND have foul odor
Anthonomous eugenii		The fruit have excrement have weevil larvae have larvae is brown heads have fruit are dropping extreme

Phytophthora capsici		The roots has fungus have roots become rot have lower stem become black have branches become brown have leaves have large irregular spots have appear scalded have clinging to dead plants
Oidopsis taurica		The leaf surface appear spots have chlorotic blotches have develop a necrotic flacking at lower leaf have covered with a white powdery growth have grey powdery growth have prominent symptom is shedding of the leaf foliage

2.3.2 Chili or Capsicum f. diseases solution [13]

Table 2.2 shown some solution for common types of disease which can be apply at the crops.

Table 2.3: The Chili (Capsicum f.) disease solution

Diseases	Sample Picture	Treatment
Blossom End Rot		Schedule irrigations so that plants will not be stressed for water. Control plant growth to a steady rate.
Herbicicde Injury		If Injury Is the result of drift on foliage, immediate syringing with liberal amounts of water may reduce damage. If syringing is delayed, even an hour or so, the remedial effects will be lessened, particularly with chemicals that absorb quickly into foliage such as the phenoxy-type herbicides.
Erwinia carotovora pv. carotovora		Disease severity increases when fruit is damaged by hail. For controlling, careful handling of pods after harvest. Control of temperature and relative humidity in storage. Rotation with grass crops, alfalfa, and beans will reduce the bacteria population in the soil.

Anthonomous eugenii (pepper weevil)		Severe pest in many chili growing regions, decreases yield and quality. Damage caused by pepper weevil can be reduced with weekly sprays of insecticides, alone or in combination. However increasing regulatory restrictions and increased incidence of insect resistance to conventional pesticides have limited the alternatives available to growers.
Phytophthora capsici		This is major disease of chili in the world. To control it, plant in furrow-ridge method, provide drainage, and do not allow water to stand in furrow longer than 6 hours. Fungicide may be beneficial for foliar blight phase of the disease.
Oidopsis taurica		This can cause in reduction in size and number of fruit. Can lead to sunscald. To control it, using fungicides may be necessary to manage the disease during periods of heavy disease pressure.

2.3.3 Existing Disease Control Method [14]

Crop fertility is often threatened by several of pest type. Therefore, this will affect the quality and the chili products from being healthy. Due to this, a several method of pest infestation and disease control are being applied to keep the plant freshness.

A common disease is caused by bacteria and fungus. The common ways to control it is by using fungicide such as Captan or Benlate. The spraying activity is carried out twice a week. There are the other ways how to control the disease and pest.

i. Disease control:

- To reduce microorganism pest, alternate the crop.
- Erase the weed which became the parasite host.
- Use fertilizer at a specific time by scheduling it. Do not using it exceeding or less from ordinary use cause it can give impact to the tree.
- Make sure the drainage always perfect.
- Ploughing the earlier crop balance.
- Use a good and healthy seed, immerse it in fungicide and insect before being planted.
- Use perennially disease clone
- Burn and pulled out if the tree is ill to avoid it from being spread
- Isolate the ill tree and destroy it.

ii. Chemical control

a. Dipping
   - Seedling is immersed inside fungicide mixture and insecticide before being planted
   - Tendril pineapples example is immersed inside bordeaux mixture to avoid rot heart disease.

b. Dusting
   - Part of affected disease are scattered by using fungicide dust
   - Using sulphur dust in rubber leaves to treat powdery Mildew

c. Painting
   - Circle the part that get attack with a chemical substance to avoid disease from spread
   - Fylomax 90 poison to control red or pink rubber disease

d. Fumigating
   - Used methyl bromide gas such as at the site of tobacco nursery

2.4 Application of Expert System

2.4.1 Fish – web based expert system for fish disease diagnosis

This prototype application was made by Daoliang Li, Yanqing Duan, and Zetian Fu, from Agricultural Information Technology Institute, China Agricultural

University, Beijing. This application has been developed for fish disease diagnosis.

This application consist of a database, knowledge base, an inference engine, a knowledge elicitation tool, an explanation subsystem, fish farming information systems, and a user interface. It was developed using a mixture of Internet techniques and SQL programming languages. GHTML (Dynamic Hypertext Markup Language), Java Script, Java, VB Script and ASP (Active Server Page) were used in the programming.

This online fish farming information system provides comprehensive information on fish farming, fish disease prevention measures, environmental factors, and contact information of fish disease experts.

System tests, such as a logic tests, debugging, rule checking and sample field tests were carried out by system developers. This web-based expert system provides an easy access for fish farmers, and able to mimic the real practice of fish disease diagnostic processes by focusing on the analysis of etiology and pathogen by matching combinations of, microscopic examinations and water quality inspections.

2.4.2 Expert system for integrated plant protection in pepper (*Capsicum annum L.*)

The study has been made by L. Gonzalez-Diaz, P. Martinez-Jimenez, F.Bastida, J.L. Gonzalez-Andujar, from Instituto de Agricultutura Sostenible-CSIC, Crop Protection, Apdo. Cordoba, Spain.

In this study, an expert system was developed with the aim of improving decision- making by pepper growers. It was represented in the knowledge base of the expert system in a series of IF-THEN rules. This system is supported by a data base that containing information for the identification of 11 weeds, 20 insects, 14 diseases, three abiotic factors and control measures.

2.4.3 Expert system for pests, diseases and weeds identification in olive crops

This paper was done by J.L Gonzalez-Andujar form Instituto de Agricultura Sostenible-CSIC, Alameda del Obispo, Apdo, Cordoba, Spain.

This expert system was developed with the aim of improving decision-making by olive oil growers by obtained a knowledge from the literature and from experts. The knowledge was then represented in the knowledge base of expert system in a series of IF-THEN rules.

As usual, the system is supported by a data base containing information for the identification of 9 weeds, 14 insects, and 14 diseases and 150 digital photos and drawings.

CHAPTER 3

METHODOLOGY

3.1 Introduction

Methodology is generally for solving a problem, with specific components such as phases, tasks, methods, techniques and tools [13]. Methodology also is the way how the data being collected, analyzed, designing user interfaces and implementing system based on the guidelines from collected data. While software development is the structure and technique of description, which is the task is to create descriptions of complex systems in which many domains meet and interact [15].

Other than that, software development also describe as a process of inventing, improving, and selecting among alternative solution, and then describing computer programs that meet user's requirements within the constraints of the environment and based on relevant criteria. A good software design is one that describes a system that will meet all its requirements.

At this chapter, it will discuss about the software development activity or method that will be applying for Capsicum Disease Diagnostic System (CDDS). The method used in developing CDDS expert system was based on Rapid Application Development (RAD) approach. The RAD is a software development methodology that involves methods like iterative development and software prototyping. It is a merger of various structured techniques, especially data-driven Information Engineering, with prototyping techniques to accelerate software systems development [16].

3.2 Background of Software Process Model

The primary functions of a software process model are to determine the order of the stages involved in software development and evolution and to establish the transition criteria for progressing from one step to the next. These include completion criteria stage plus choice criteria and entrance criteria for the next stage. Thus, a process model addresses the following software project questions of what we will do next, and how long shall we continue to do it.

Consequently, a process model differs from a software method which often called as software methodology in that a method's primary focus is on how to navigate through each phase on determining data, control, or allocating requirements and how to represent phase products.

Software process models are important because they provide guidance on the phases, increments, prototypes, and validation task in which a project should carry out its major task.

3.3 Rapid Application Development (RAD) Method

Rapid Application Development (RAD) is a development lifecycle designed to give much faster development and higher-quality results than those achieved with the traditional lifecycle. It is designed to take the maximum advantage of powerful development software that has evolved recently [17]. Rapid Application Development as "an approach to building computer systems which combines Computer-Assisted Software Engineering (CASE) tools and techniques, user-driven prototyping, and stringent project delivery time limits into a potent, tested, reliable formula for top-notch

quality and productivity. RAD drastically raises the quality of finished systems while reducing the time it takes to build them."(Professor Clifford Kettemborough of Whitehead College, University of Redlands, 2000).

Online Knowledge defines Rapid Application Development as a methodology that enables organizations to develop strategically important systems faster while reducing development costs and maintaining quality. RAD takes advantage of automated tools and techniques to restructure the process of building information systems.

RAD features which overriding characteristics[18]:

- The focus is on delivering projects in small pieces. If you have a large project, you need to look at ways to break it into smaller projects, each of which can be planned and delivered individually. With a series of smaller projects, you can deliver each one more quickly and in a less structured manner.

- Deliverables, including the final solution, are created using a repeating process of analysis, design, construction, and testing. Prototypes are created early and evolved to include more detail over time.

- RAD emphasizes reuse. This includes the reuse of code, processes, templates, and tools. It is usually faster to assemble prebuilt components than to build everything from scratch.

3.3.1 Rapid Application Development Phase

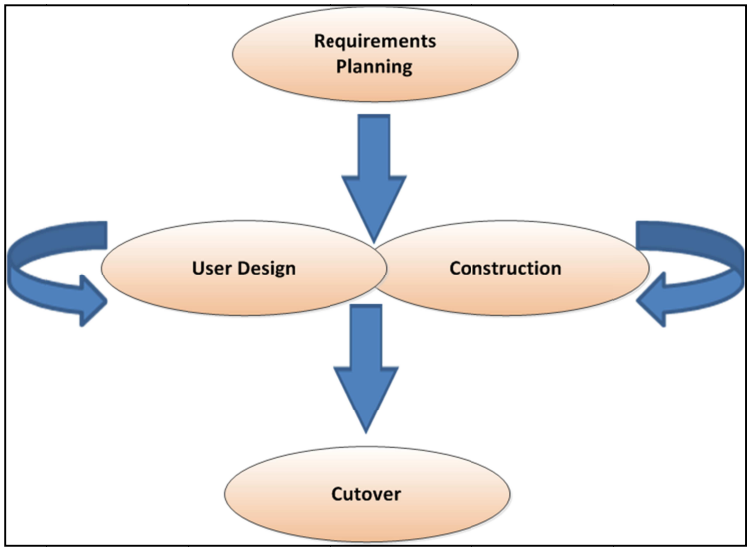


Figure 3.1: RAD model of the software process

The structure of the RAD lifecycle is thus designed to ensure that developers build the systems that the users really need. This lifecycle, through the following four stages, includes all of the activities and tasks required to scope and define business requirements and design, develop, and implement the application system that supports those requirements.

- Requirements Planning
    - Also known as the Concept Definition Stage, this stage defines what the Chili Disease System functions, support and determines the system's scope.

- User Design
    - Also known as the Functional Design Stage, this stage uses Dreamweaver CS3, rational Rose software to model the system's data and processes and to build a working prototype of critical system components.
- Construction
    - Also known as the Development Stage, this stage completes the construction of the physical application system, by develop a PHP coding to a system so that CDS can work properly.
- Cutover
    - Also known as the Deployment Stage, this stage includes final user testing and training, data conversion, and the implementation of the application system.

RAD (rapid application development) proposes that products can be developed faster and of higher quality by Vic Berry and Anja Naumann:

- Using workshops or focus groups to gather requirements.
- Prototyping and user testing of designs.
- Re-using software components.
- Following a schedule that defers design improvements to the next product version.
- Keeping review meetings and other team communication informal.

3.3.1.1 Requirement Planning Phase

All the requirements planning combined the elements of traditional planning and analysis phases.

The requirements, data of the diseases had been collected by making visits to the Pejabat Pertanian Jubli Perak in Indera Mahkota city in Kuantan, Pahang and some reference book about Capsicum Disease attack borrowed from their library. The interview was conducted by discussing the symptoms; types of disease attack among the Capsicum from the buds until it grow with the manager and some of the officer of the Pejabat Pertanian Jubli Perak.

3.3.1.2 User Design Phase

At this phase, a design or models of the required system are prepared based on forward chaining inference technique as shown below at figure3.2. Besides that, developer must make a database based on the system requirement.

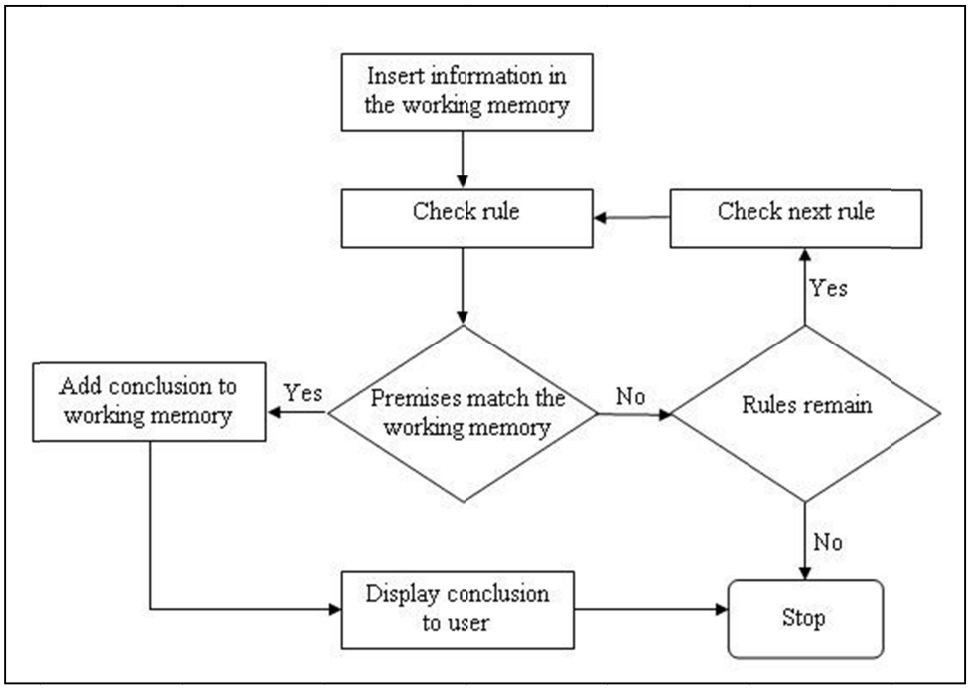


Figure 3.2: Forward chaining inference processes

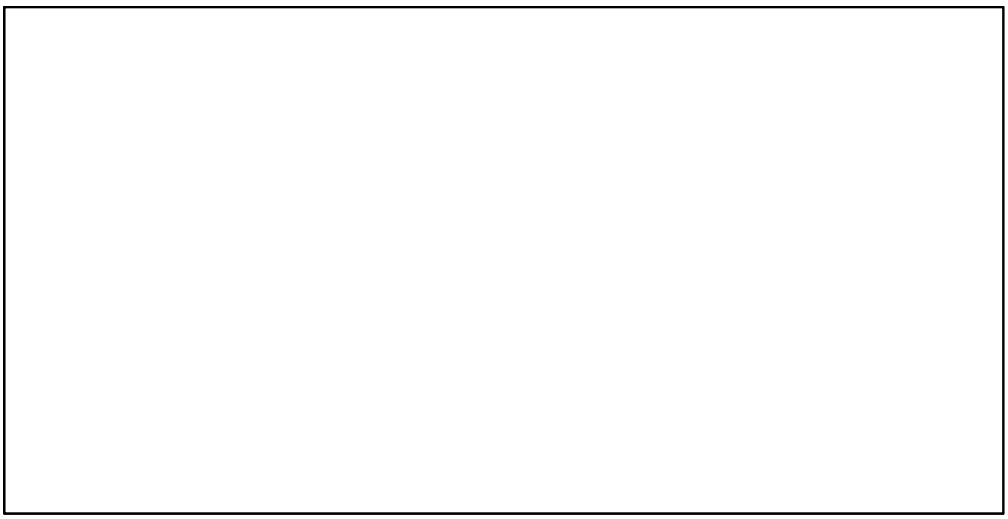


Figure 3.3: Sample rule based of Collectrichum gloeosporiorides diseases

The software requirement specifications for Capsicum f. System are studied again to match them with the design. A model and prototypes will be developing to represent all system processes, inputs, and outputs.

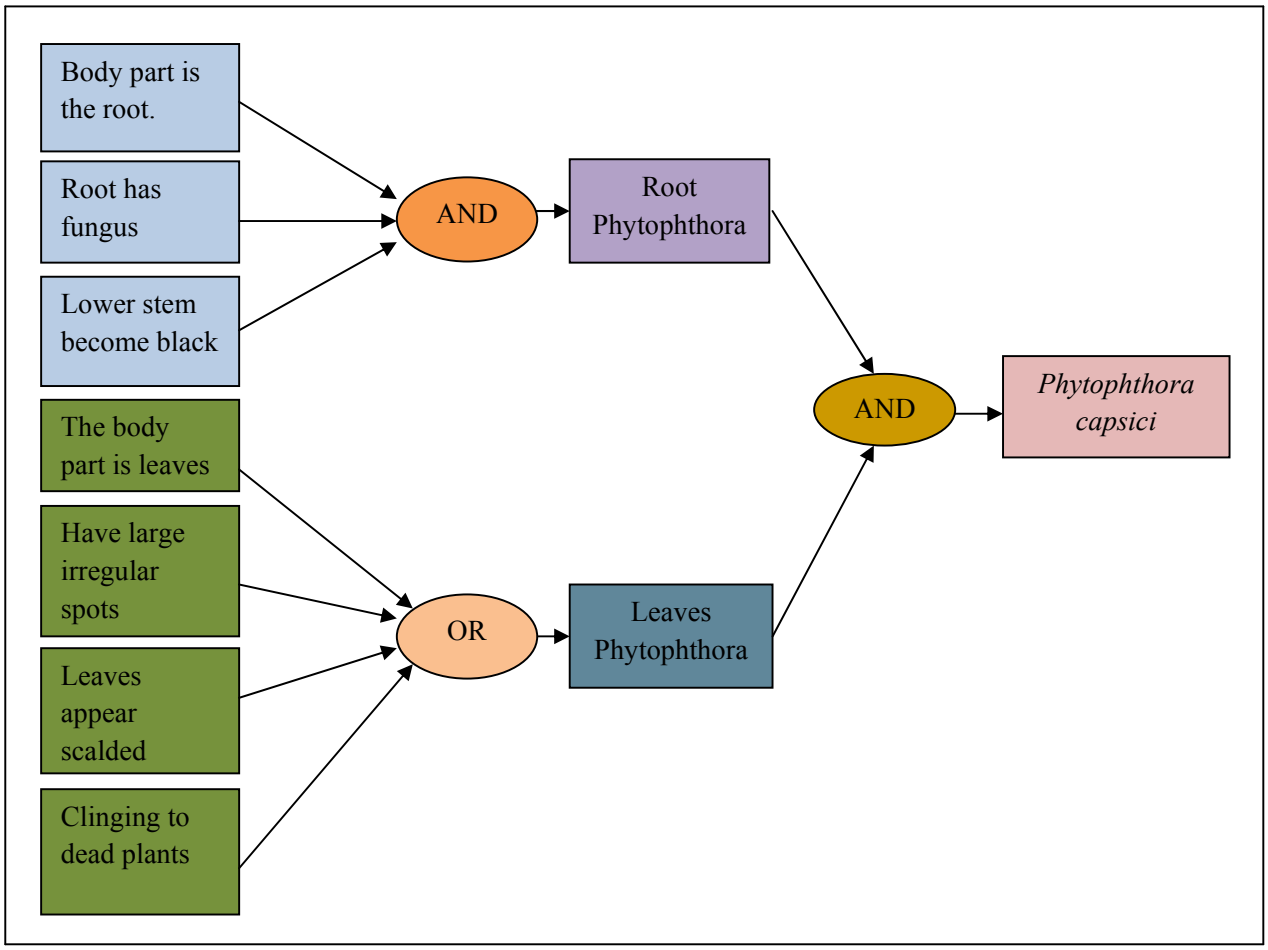


Figure 3.4: Forward chaining inference techniques.

The figure 3.4 above shows how the forward chaining works. When all the desired facts matched, then it will fire to the disease in the database.

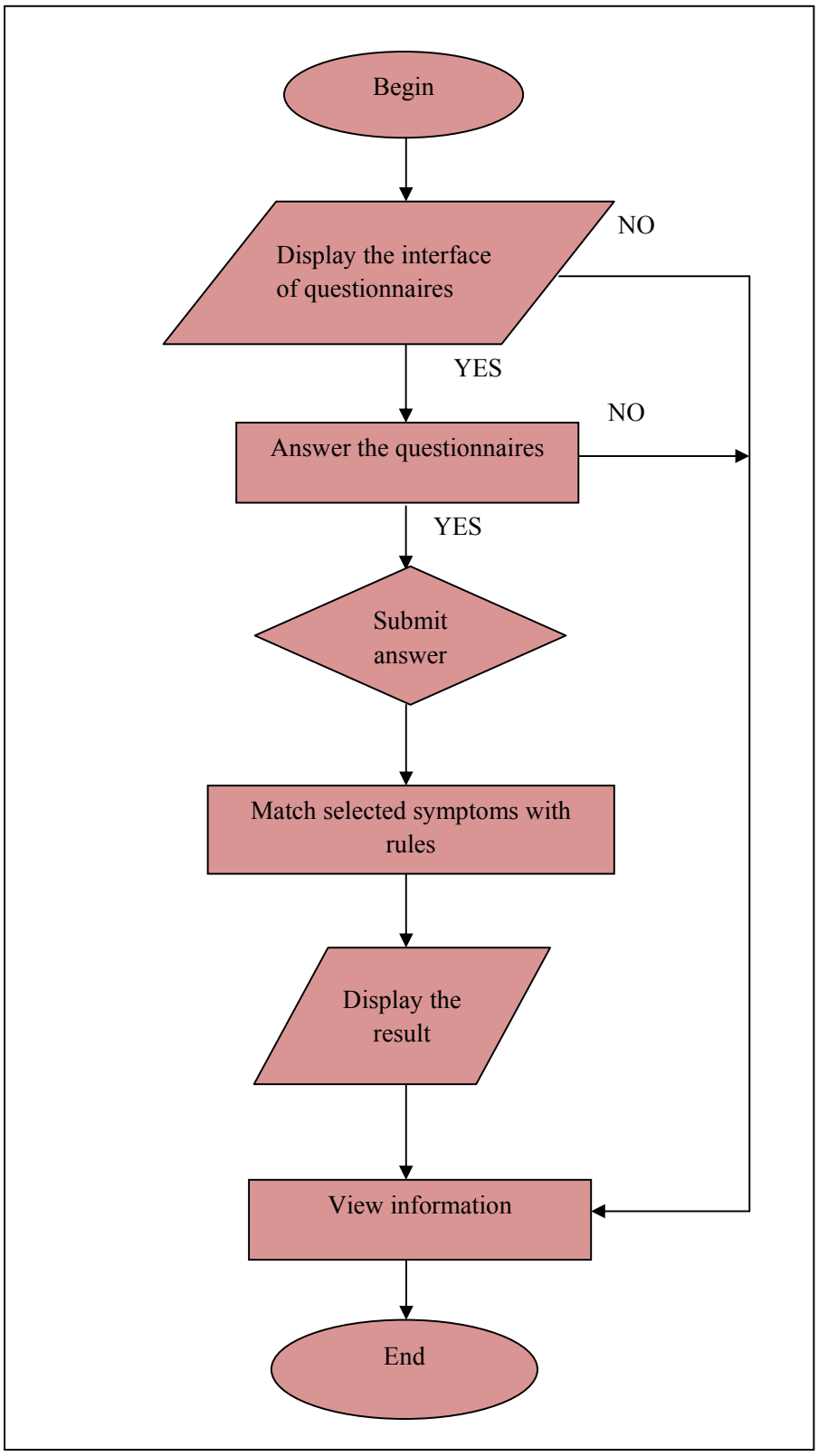


Figure 3.5: Flow chart for Chili Diagnostic System

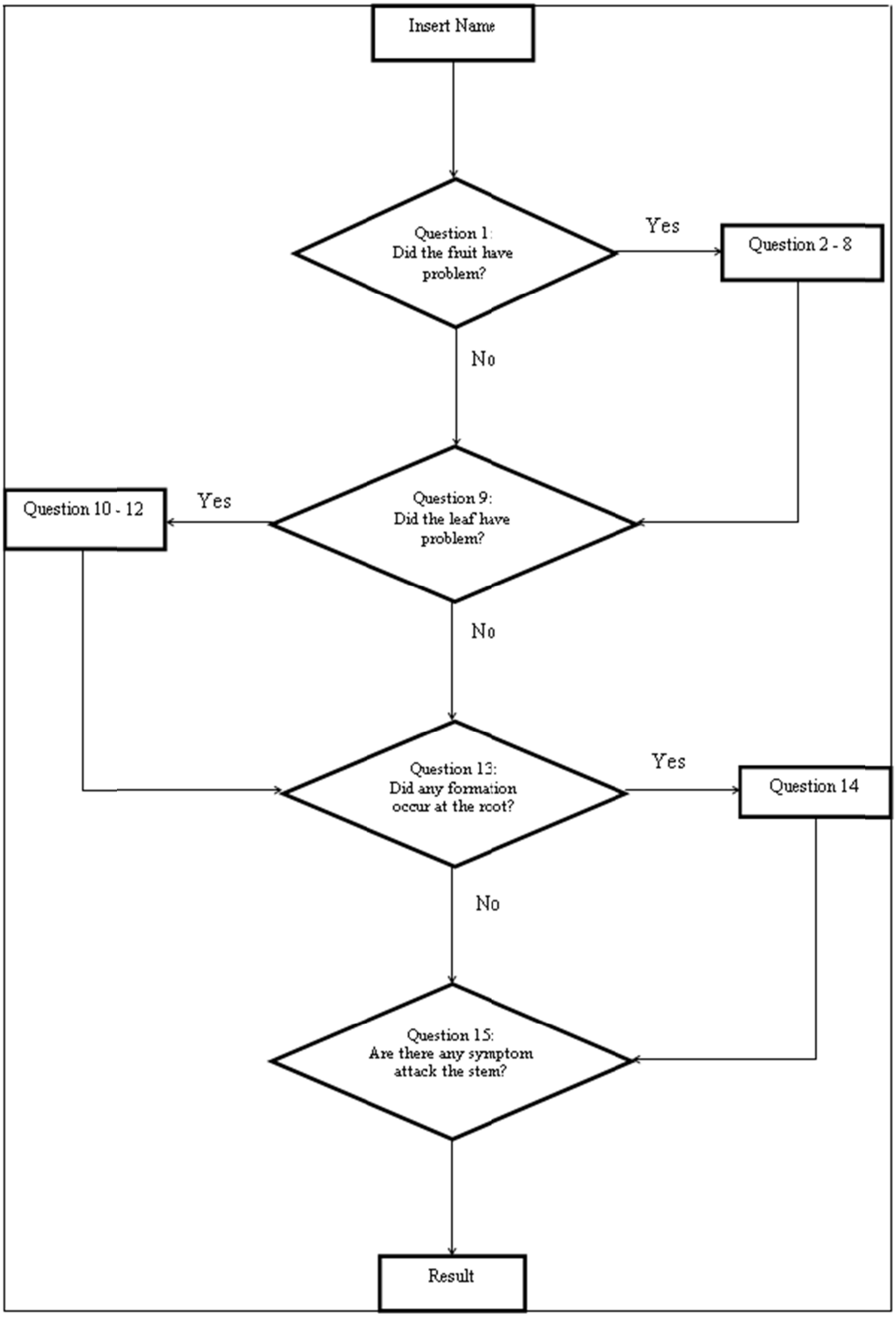

Insert Name
Question 1:
Did the fruit have
problem?
Yes
Question 2 - 8
No
Question 9:
Did the leaf have
problem?
Yes
Question 10 - 12
No
Question 13:
Did any formation
occur at the root?
Yes
Question 14
No
Question 15:
Are there any symptom
attack the stem?
Result

Figure 3.6: Flow chart for Chili Diagnostic Question System

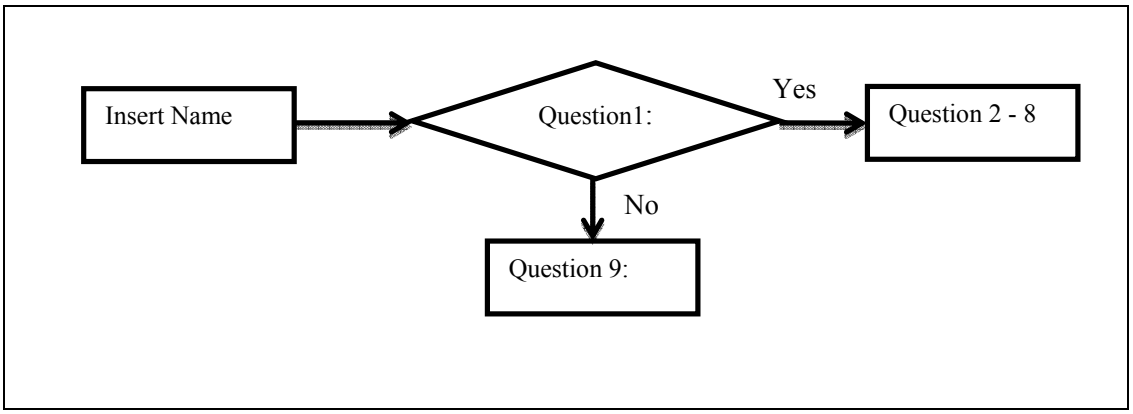


Figure 3.7: Forward chaining inference techniques CDS system.

Figure 3.6 and Figure 3.7 show CDS forward chaining technique:

IF Question1 answered Yes
THEN forward to Question2-8.
ELSE IF answered No

IF Question2-8 are answered
THEN forward to Question9.

IF Question9 answered Yes
THEN forward to Question10-12
ELSE IFanswered No
THEN forward to Question13.

IF Question10-12 are answered
THEN forward to Question13.

IF Question13 answered Yes
THEN forward to Question14
ELSE IF answered No

THEN forward to Question15.

IF Question1 answered Yes
AND forward to Question2-8.

AND forward to Question9
AND answered Yes
AND forward to Question10-12
AND forward to Question13
AND answered Yes
AND forward to Question14.

AND forward to Question15
AND symptom are match
THEN result are show.

3.3.1.2.1 Use case Diagram

Figure 3.8 shows the use case diagram for *Capsicum* Disease System. The use case diagram is the overall view of the system. In this diagram, there is the actor who represents as user that interacts with this application system. Another actor was an administrator who managing and maintaining the database system. Users communicate with the system through user interfaces that act as medium for communication.

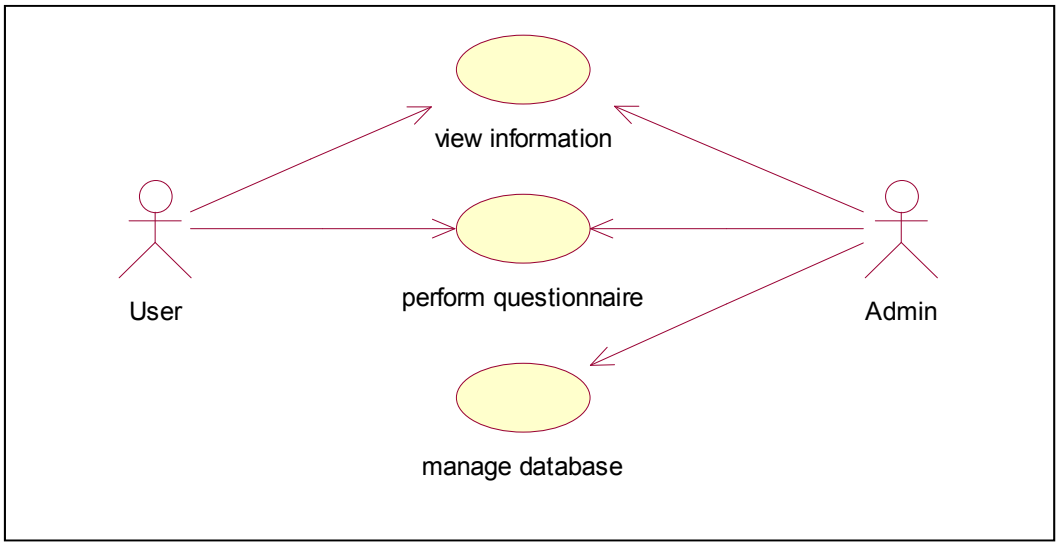


Figure 3.8: Use Case diagram for Chili (Capsicum f.) Disease System

i. User

User is the people who will interact with the Chili (Capsicum f.) Disease System. User may answer a question and view information about chili that provided in the system.

ii. Admin

Admin is the person who is responsible for the Chili Diseases System, user data and maintenance of the system.

iii. perform questionnaire

This event is initiated by user by answering the symptom questionnaire given at the system so that user can know what type of disease that attack their crops.

iv. view information

This event is initiated by user or admin if they want to know some information about chili disease by viewing information given at the system.

v. manage database

This use case is initiated by admin to maintain the system database, user data, and update information of the system.

3.3.1.2.2 Sequences Diagram

The sequence diagram defines how the systems operate and show the interactions between objects in the sequential order that those interactions occur. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams typically are associated with use case realizations in the Logical View of the system under development (OBM (2005). FEA Consolidated Reference Model Document. May 2005).

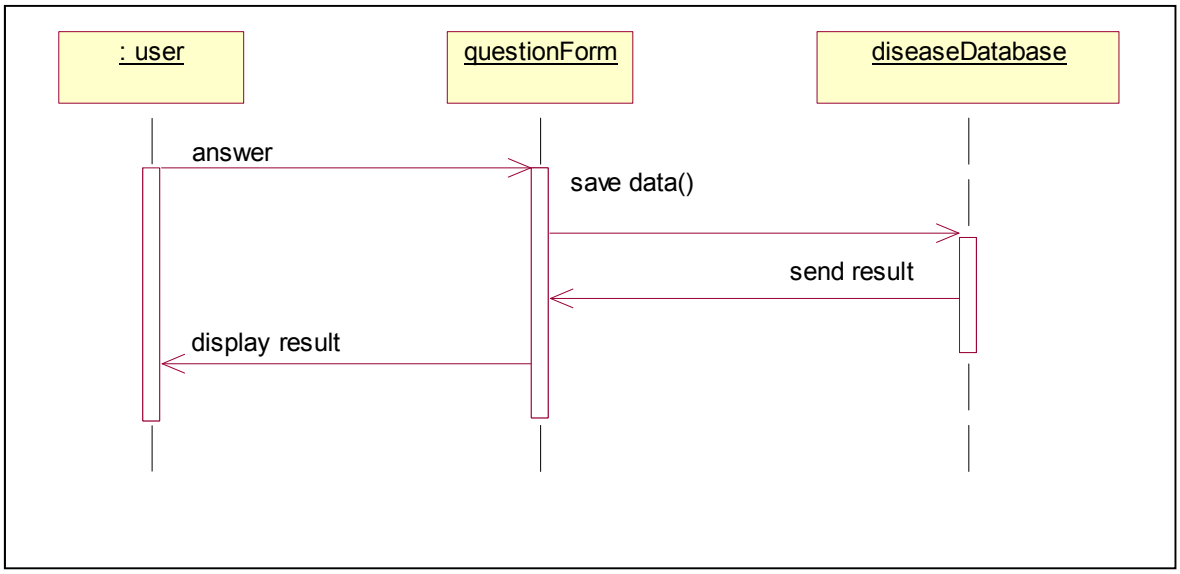


Figure 3.9: Sequences Diagram for perform question

Figure 3.9 shows the sequences diagram in performing a question. The user needs to answer the questionnaire then system will save the answer and send it to the database to see the results. After the answer have match at the database, system will display the result to the user.

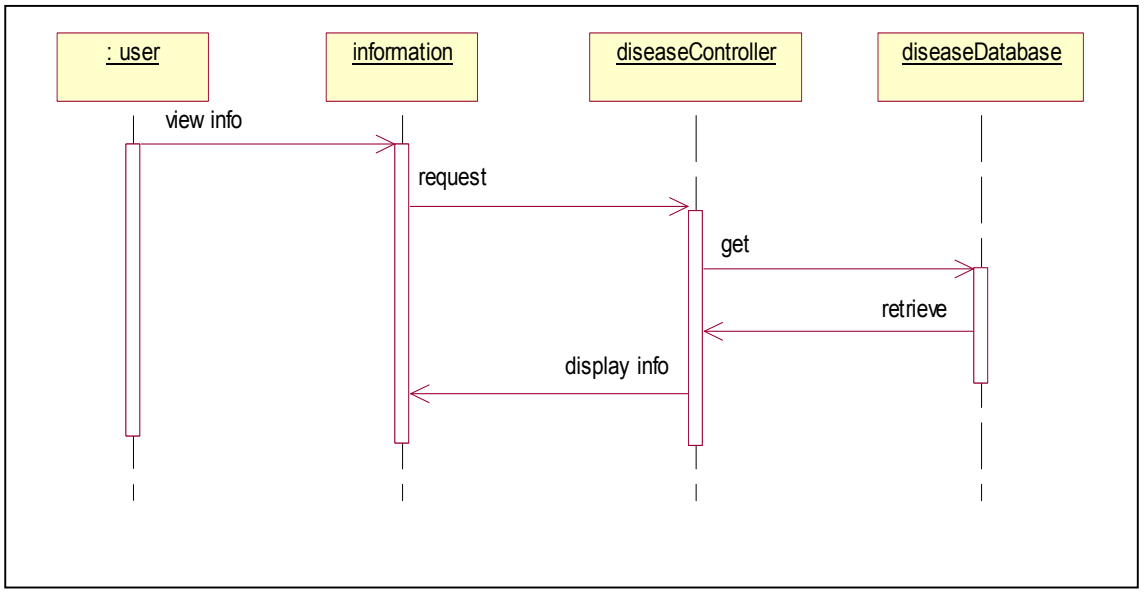


Figure 3.10: Sequences Diagram for view information

As shown at figure 3.10, user can view the chili disease information to gain more knowledge about chili disease that always attacks this type of crops.

3.3.1.2.3 Database Design

A database is an organized collection of data for one or more purposes, usually in digital form. A database system is a database data collection with DBMS (database management system) which will be applied in this Chili Disease System. Database design is needed to show the process of producing a detailed data model of this system database. A decisions making is needed regarding to take some system in the real world and model it in database.

3.3.1.2.4 Data Dictionary

Table 3.0: Data dictionary for admin

Field Name	Data Type	Size	Description
member_id	Int(11)		Primary key of database
username	Varchar(50)	10	username
password	Varchar(50)	10	Authentication key to access to database

Table 3.1: Data dictionary for disease

Field Name	Data Type	Size	Description
dIndex	Int(11)		Primary key for disease description
dDesc	Varchar(20)	20	Disease description

Table 3.2: Data dictionary for disease type

Field Name	Data Type	Size	Description
Blossom End Rot	Varchar	100	Result of Blossom End Rot
Herbicide Injury	Varchar	100	Result of Herbicide Injury
Erwinia carotovora pv. Carotovora	Varchar	100	Result of Erwinia carotovora pv. Carotovora
Anthonomous eugenii	Varchar	100	Result of Anthonomous eugenii
Phytophthora capsici	Varchar	100	Result of Phytophthora

			capsici
Oidopsis taurica	Varchar	100	Result of Oidopsis taurica

Table 3.3: Data dictionary for symptom

Field Name	Data Type	Size	Description
symptomIndex	Int(11)		Symptom description primary key number
symptomDesc	text		Symptom description

Table 3.4: Data dictionary for disease answer rule

Field Name	Data Type	Size	Description
daNo	Int(11)		Primary key for table diseaseanswer
dqIndex	Int(11)		Foreign key from diseasequestion table
symptomIndex	Int(11)	100	Symptom index number to be shown from symptom table
dqPrevAns	Int(11)	100	Symptom previous index that answer by user
dqNextindex	Int(11)	100	Next question index number

Table 3.5: Data dictionary for temporary user

Field Name	Data Type	Description
tIndex	Int(11)	Primary key for table

tUser	Text	Temporary user name
tAnswer1	Int(11)	Answer for question 1
tAnswer2	Int(11)	Answer for question 2
tAnswer3	Int(11)	Answer for question 3
tAnswer4	Int(11)	Answer for question 4
tAnswer5	Int(11)	Answer for question 5
tAnswer6	Int(11)	Answer for question 6
tAnswer7	Int(11)	Answer for question 7
tAnswer8	Int(11)	Answer for question 8
tAnswer9	Int(11)	Answer for question 9
tAnswer10	Int(11)	Answer for question 10
tAnswer11	Int(11)	Answer for question 11
tAnswer12	Int(11)	Answer for question 12
tAnswer13	Int(11)	Answer for question 13
tAnswer14	Int(11)	Answer for question 14
tAnswer15	Int(11)	Answer for question 15

Table 3.6: Data dictionary for disease question

Field Name	Data Type	Size	Description
dqIndex	Int(11)		Primary key for disease question table
dqDescription	text		Disease question
dgTag	Varchar(20)	25	Disease question part

3.3.1.3 Construction Phase

During the construction phase, a prototype is built using the software tools. The initial prototype consists of screens, forms, reports, and other elements of the user interfaces, and the underlying logic is added to the prototype only after the user interface is stabilized. In the Capsicum Diagnostic Disease System, prototype of interfaces is build according to the system requirement.

3.3.1.4 Cutover Phase

The cutover phase is similar to implementation, a variety of actions are needed. Key activities include training the users, converting or installing the system, and completing the necessary documentation.

3.3.2 Hardware and Software

Hardware, in the computer world, refers to the physical components that make up a computer system. There are many different kinds of hardware that can be installed inside, and connected to the outside, of a computer. Table 3.1 is the computer hardware facilities that use during development process.

Software are organized information in the form of operating systems, utilities, programs, and code written by programmers in any of various special computer languages. Table 3.2 is the computer software application that use during development process. Software is divided commonly into two main categories:

- System software: controls the basic (and invisible to the user) functions of a computer and comes usually preinstalled with the machine. See also BIOS and operating System.
- Application software: handles multitudes common and specialized tasks a user wants to perform, such as accounting, communicating, data processing, and word processing.

Table 3.7 The list of hardware used during development

Computer Hardware	Quantity	Purpose
Notebook Lenovo 3000 G430	1	Coding and implementation researching and documenting.
Processor – Intel Pentium	1	Programmed hardware in notebook.
Thumb drive	1	Storage medium
CD writer	1	Backup files and data
Printer	1	Printing.

Table 3.8 The list of software used during development

Computer Software	Purpose
Windows Vista Ultimate	As the operating system of development phase.
Dreamweaver CS3	For design an interface and some coding
Aptana Studio 3	Storage medium
XAMPP	Support the MySQL database

MySQL Database	Store some information of chili symptoms
PhpMyAdmin	A database management system
Microsoft Word 2010	For documentation
Microsoft Powerpoint 2010	For presentation
Google Chrome	Information searching
Mozilla Firefox 3.6.1	Display prototype design

CHAPTER 4

IMPLEMENTATION

4.1 Introduction

These chapters will discuss on the development and implementation process of the Chili Disease System. The objectives of this chapter is to record all phases on developing the system for better future improvements and enhancements.

4.2 Result of Chili Disease System

The end of the development phase of the system includes three (3) categories which is the database, the system interfaces and the rule base implementation. The database has two more categories; database construction and database connector. The database categories will show how this database is created. The system output will demonstrate the system functionalities and it roles based on its requirements. The rule base implementation is the engine in real world before transform into system coding.

4.2.1 Database Construction

Chili Disease System (CDS) has three (3) main databases; the admin database, fact database, and the rule database. The database was constructed by using phpMyAdmin MySQL.

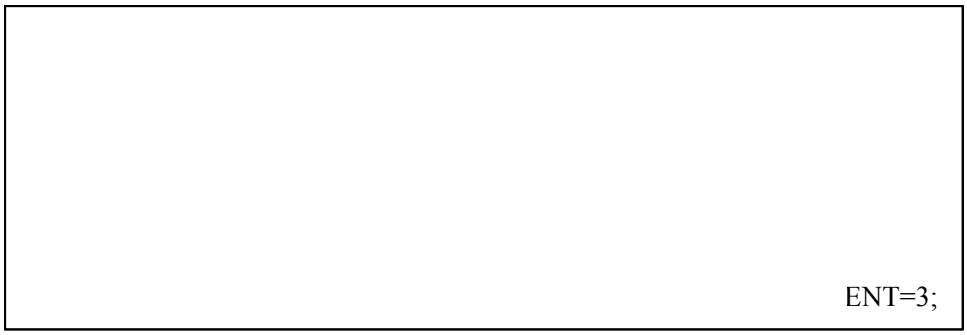
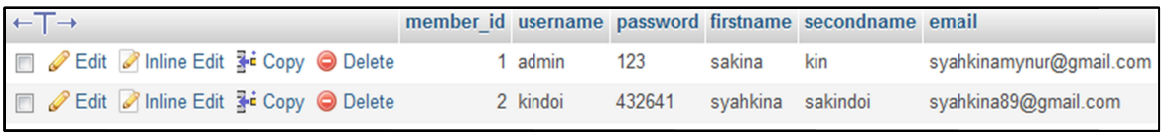


Figure 4.0: SQL query for admin table



	member_id	username	password	firstname	secondname	email
Edit Inline Edit Copy Delete	1	admin	123	sakina	kin	syahkinamynur@gmail.com
Edit Inline Edit Copy Delete	2	kindoi	432641	syahkina	sakindoi	syahkina89@gmail.com

Figure 4.1: Data that recorded at members table

Figure 4.0 shows about the creation of members table where it is for members data to be record using MySQL source code. In the members table, there are six key created as shown at Figure 4.1.

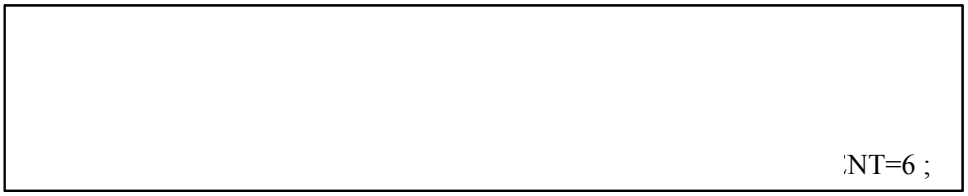
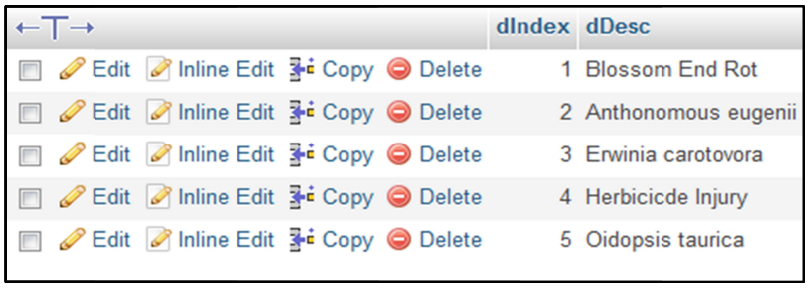


Figure 4.2: SQL query for chilidisease table

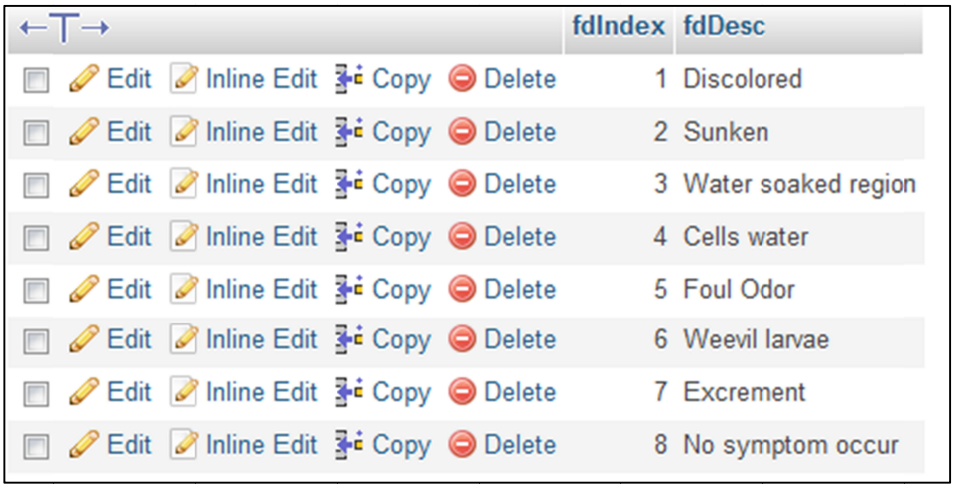


←T→	dIndex	dDesc
Edit Inline Edit Copy Delete	1	Blossom End Rot
Edit Inline Edit Copy Delete	2	Anthonomous eugenii
Edit Inline Edit Copy Delete	3	Erwinia carotovora
Edit Inline Edit Copy Delete	4	Herbicicde Injury
Edit Inline Edit Copy Delete	5	Oidopsis taurica

Figure 4.3: Data that recorded at chilidisease table

Figure 4.2 shows about the creation of chili disease data at dfact database using MySQL source code. At this table, there are 2 key created which is dIndex the index of chili disease name, the dDesc as shown at Figure 4.3.

Figure 4.4: SQL query for fruit table



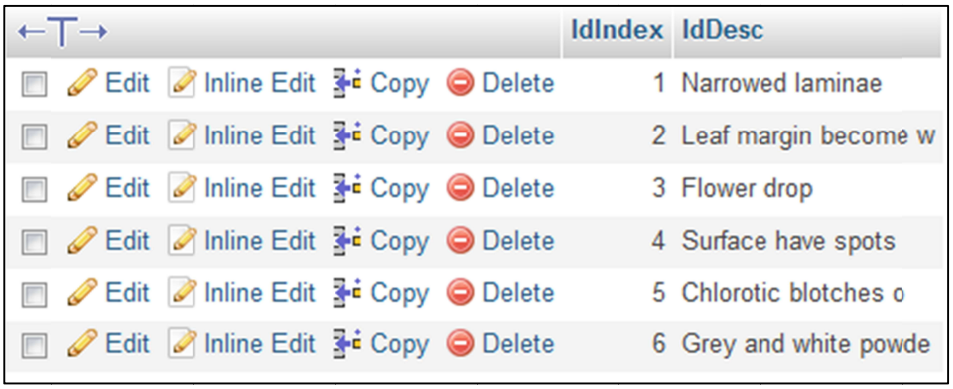
←T→	fdIndex	fdDesc
Edit Inline Edit Copy Delete	1	Discolored
Edit Inline Edit Copy Delete	2	Sunken
Edit Inline Edit Copy Delete	3	Water soaked region
Edit Inline Edit Copy Delete	4	Cells water
Edit Inline Edit Copy Delete	5	Foul Odor
Edit Inline Edit Copy Delete	6	Weevil larvae
Edit Inline Edit Copy Delete	7	Excrement
Edit Inline Edit Copy Delete	8	No symptom occur

Figure 4.5: Data that recorded at fruit table

Figure 4.4 shows about the creation of fruit table using MySQL source code. At this table, there are 2 key created. There are fdIndex, and fdDesc as shown at figure above.

```
T=7 ;
```

Figure 4.6: SQL query for leaf table



←T→	ldIndex	ldDesc
Edit Inline Edit Copy Delete	1	Narrowed laminae
Edit Inline Edit Copy Delete	2	Leaf margin become w
Edit Inline Edit Copy Delete	3	Flower drop
Edit Inline Edit Copy Delete	4	Surface have spots
Edit Inline Edit Copy Delete	5	Chlorotic blotches o
Edit Inline Edit Copy Delete	6	Grey and white powde

Figure 4.7: Data that recorded at leaf table

Figure 4.6 shows about the creation of leaf database and the data at dfact database using MySQL source code. At this table, there are 2 key created which is ldIndex the index of leaf symptom and the description, ldDesc as shown at Figure 4.7.

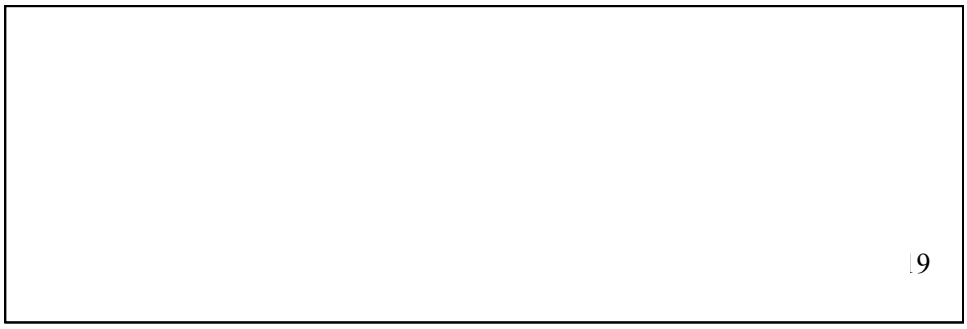


Figure 4.8: SQL query for diseaseanswer table

daNo	dqIndex	fdIndex	dqPrevAns	dqNextIndex
1	1	1	0	2
2	1	9	0	2
3	2	2	1	3
4	2	9	1	3
5	2	2	9	3
6	2	9	9	3
7	3	3	2	0
8	3	9	2	4
9	3	3	9	4
10	3	9	9	4
11	4	4	3	5
12	4	9	3	5
13	4	4	9	5
14	4	9	9	5
15	5	5	4	0
16	5	9	4	6
17	5	5	9	6
18	5	9	9	6
19	6	6	5	7
20	6	9	5	7

Figure 4.9: Example rules in chili disease

Figure 4.8 shows about the creation of leaf database and the data at drule database using MySQL source code. At this table, there are 5 key created as shown at Figure 4.9. There are daNo, dqIndex, fdIndex, dqPrevAns, and dqNextIndex. From Figure4.8, dqIndex refer to disease question at diseasequestion table, fdIndex refer to the answer that user will choose at question form at fruit table, dqPrevAns refer to the answer that user choose at previous question, and dqNextIndex refer to the next question which the rule will fire to.

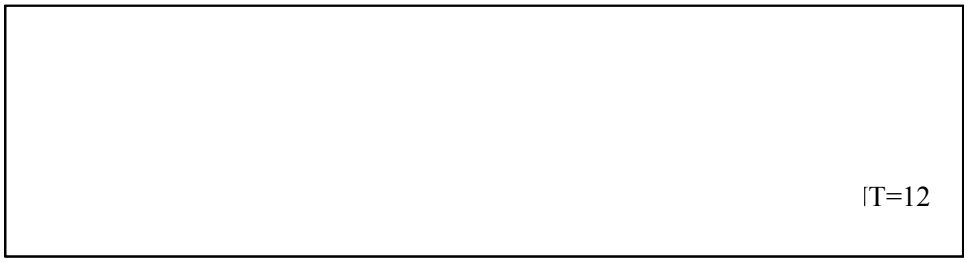


Figure 4.10: SQL query for diseasequestion table

dqIndex	dqDescription	dgTag
1	Did the fruit have a problem?	fruit
2	Did the fruit color become dis	fruit
3	Did the fruit have weevil larv	fruit
4	Did the fruit have other sympt	fruit
5	Did the leaves structure norma	leaf
6	Did the leaves covered with ap	leaf
7	Are the leave grow normally?	leaf
8	Did any formation occur at the	leaf
9	Did the flower had side effect	flower
10	Did the seed collapse?	seed
11	Did the cell have this kind of	cell

Figure 4.11: Data inserted at diseasequestion table

Figure 4.10 shows about the creation of diseasequestion database and the data at drule database using MySQL source code. At this table, there are 3 key created which is dqIndex the index of leaf symptom and the description, dqDescription and dgTag as shown at Figure 4.11.

Figure 4.12 shows about the creation of fruitanswer database and the data at drule database using MySQL source code.Figure 4.13 shows that for question number 1 to question number 8, if the answer has three or more than are true, it will go to disease result. While if the answer are less than three are true, it will continue the question.

```
CREATE TABLE IF NOT EXISTS `fruitanswer` (
  `faNo` int(11) NOT NULL AUTO_INCREMENT,
  `dIndex` int(11) NOT NULL,
  `fdIndex1` int(11) NOT NULL,
  `fdIndex2` int(11) NOT NULL,
  `fdIndex3` int(11) NOT NULL,
  `fdIndex4` int(11) NOT NULL,
  `fdIndex5` int(11) NOT NULL,
  `fdIndex6` int(11) NOT NULL,
  `fdIndex7` int(11) NOT NULL,
  `fdIndex8` int(11) NOT NULL,
  PRIMARY KEY (`faNo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

Figure 4.12: SQL query for fruitanswer table

faNo	dIndex	fdIndex1	fdIndex2	fdIndex3	fdIndex4	fdIndex5	fdIndex6	fdIndex7	fdIndex8
1	1	1	2	3	9	9	9	9	9
2	2	9	9	3	4	5	9	9	9
3	3	9	9	9	9	9	6	7	8
4	1	1	2	3	4	9	9	9	9
5	2	9	9	3	4	5	6	9	9
6	3	9	9	9	9	5	6	7	8
7	1	1	2	3	9	5	9	9	9
8	2	9	9	3	4	5	9	7	9
9	3	9	9	9	4	9	6	7	8
10	1	1	2	3	9	9	6	9	9
11	2	9	9	3	4	5	9	9	8
12	3	9	9	3	9	9	6	7	8
13	1	2	3	9	9	9	9	7	9
14	2	9	2	3	4	5	9	9	9
15	3	9	2	9	9	9	6	7	8
16	1	1	2	3	9	9	9	9	8
17	2	1	9	3	4	5	9	9	9
18	3	1	9	9	9	9	6	7	8

Figure 4.13: Data inserted at fruitanswer table

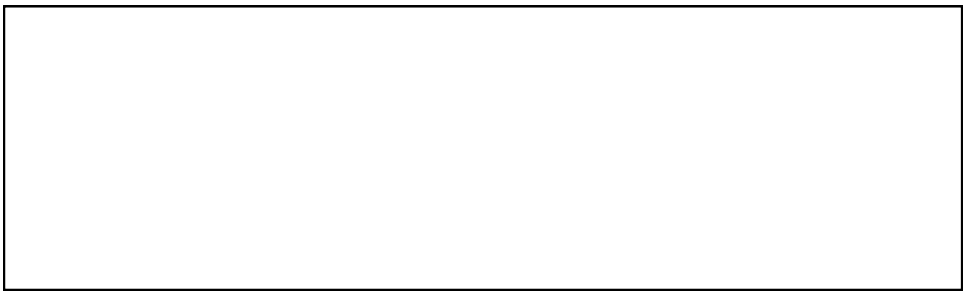


Figure 4.14: SQL query for leafanswer table

laNo	dIndex	ldIndex1	ldIndex2	ldIndex3
1	4	1	2	3
2	5	4	5	6
3	4	1	2	6
4	5	4	5	3
5	4	1	5	3
6	5	4	2	6
7	4	4	2	3
8	5	1	5	6

Figure 4.15: Data inserted at leafanswer table

Figure 4.14 shows about the creation of leafanswer database and the data at drule database using MySQL source code. Figure 4.15 shows the solution of chili disease according to the leaf symptom.

4.2.2 Database Connector

All figure shows the source code that will connect the Chili Disease System with the database and allow the data to be saved in the database.

```
<?php
$host = "localhost";
$user = "root";
$pass = "";
$dbse = "infodb";

$connect = mysql_connect($host,$user,$pass) or die ("Cannot
connect to the database");
mysql_select_db($dbse) or die("Cannot select the database");
?>
```

Figure 4.16: Connector to infodb database

Figure 4.16 show a connector to a localhost where the database are infodb which the table are used for a temporary user and connecting to the rules.

```
<?php
$dbHost = "localhost";
$dbUser = "root";
$dbPass = "";
$dbDatabase = "cb09073";

$db = mysql_connect("$dbHost", "$dbUser", "$dbPass") or die
("Error connecting to database.");
mysql_select_db("$dbDatabase", $db);

// set the default time zone to use in Malaysia
date_default_timezone_set('Asia/Kuala_Lumpur');
?>
```

Figure 4.17: Connector to cb09073 database.

Figure 4.17 show a connector to cb09073 database, where the main connection used by the system.

```
<?php
$dbHost = "localhost";
$dbUser = "root";
$dbPass = "";
$dbDatabase = "dfact";

$db = mysql_connect("$dbHost", "$dbUser", "$dbPass") or die
("Error connecting to database.");
mysql_select_db("$dbDatabase", $db);

// set the default time zone to use in Malaysia
date_default_timezone_set('Asia/Kuala_Lumpur');
?>
```

Figure 4.18: Connector to dfact database.

Figure 4.18 show a connector to dfact database. Databse 'dfact' consist two important table where it include the fact of the chili disease.

```
<?php
$dbHost = "localhost";
$dbUser = "root";
$dbPass = "";
$dbDatabase = "drule";

$db = mysql_connect("$dbHost", "$dbUser", "$dbPass") or die ("Error
connecting to database.");
mysql_select_db("$dbDatabase", $db);

// set the default time zone to use in Malaysia
date_default_timezone_set('Asia/Kuala_Lumpur');
?>
```

Figure 4.19: Connector to drule database.

Figure 4.19 show a connector to drule database. Databse ‘drule’ consist three important table which the table have a rule to match with the fact table.

4.2.3 CDS Database Table

←T→	dIndex	dDesc
Edit Inline Edit Copy Delete	1	Blossom End Rot
Edit Inline Edit Copy Delete	2	Anthonomous eugenii
Edit Inline Edit Copy Delete	3	Erwinia carotovora
Edit Inline Edit Copy Delete	4	Herbicicde Injury
Edit Inline Edit Copy Delete	5	Oidopsis taurica

Figure 4.20: Data that recorded at chilidisease table

←T→	fdIndex	fdDesc
Edit Inline Edit Copy Delete	1	Discolored
Edit Inline Edit Copy Delete	2	Sunken
Edit Inline Edit Copy Delete	3	Water soaked region
Edit Inline Edit Copy Delete	4	Cells water
Edit Inline Edit Copy Delete	5	Foul Odor
Edit Inline Edit Copy Delete	6	Weevil larvae
Edit Inline Edit Copy Delete	7	Excrement
Edit Inline Edit Copy Delete	8	No symptom occur

Figure 4.21: Data that recorded at fruit table

←T→		IdIndex	IdDesc
Edit	Inline Edit Copy Delete	1	Narrowed laminae
Edit	Inline Edit Copy Delete	2	Leaf margin become w
Edit	Inline Edit Copy Delete	3	Flower drop
Edit	Inline Edit Copy Delete	4	Surface have spots
Edit	Inline Edit Copy Delete	5	Chlorotic blotches o
Edit	Inline Edit Copy Delete	6	Grey and white powde

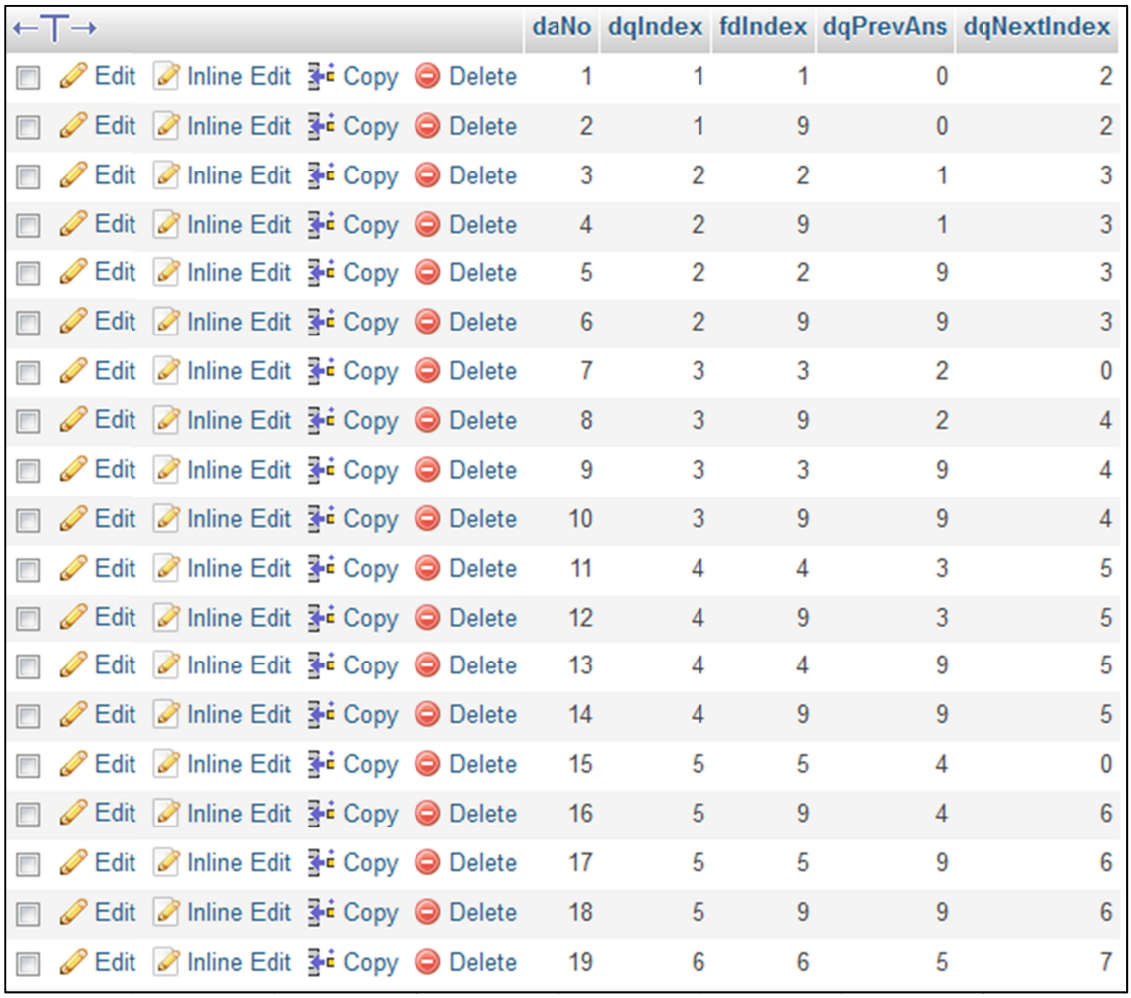
Figure 4.22: Data that recorded at leaf table

←T→		dqIndex	dqDescription	dgTag
Edit	Inline Edit Copy Delete	1	Did the fruit have a problem?	fruit
Edit	Inline Edit Copy Delete	2	Did the fruit color become dis	fruit
Edit	Inline Edit Copy Delete	3	Did the fruit have weevil larv	fruit
Edit	Inline Edit Copy Delete	4	Did the fruit have other sympt	fruit
Edit	Inline Edit Copy Delete	5	Did the leaves structure norma	leaf
Edit	Inline Edit Copy Delete	6	Did the leaves covered with ap	leaf
Edit	Inline Edit Copy Delete	7	Are the leave grow normally?	leaf
Edit	Inline Edit Copy Delete	8	Did any formation occur at the	leaf
Edit	Inline Edit Copy Delete	9	Did the flower had side effect	flower
Edit	Inline Edit Copy Delete	10	Did the seed collapse?	seed
Edit	Inline Edit Copy Delete	11	Did the cell have this kind of	cell

Figure 4.23: Data recorded at diseasequestion table

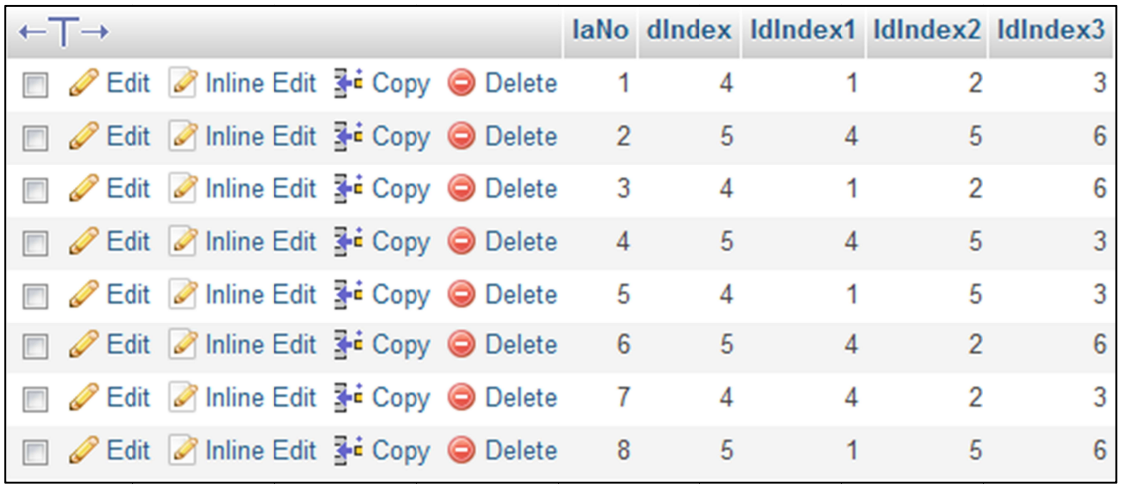
←T→	faNo	dIndex	fdIndex1	fdIndex2	fdIndex3	fdIndex4	fdIndex5	fdIndex6	fdIndex7	fdIndex8
Edit Inline Edit Copy Delete	1	1	1	2	3	9	9	9	9	9
Edit Inline Edit Copy Delete	2	2	9	9	3	4	5	9	9	9
Edit Inline Edit Copy Delete	3	3	9	9	9	9	9	6	7	8
Edit Inline Edit Copy Delete	4	1	1	2	3	4	9	9	9	9
Edit Inline Edit Copy Delete	5	2	9	9	3	4	5	6	9	9
Edit Inline Edit Copy Delete	6	3	9	9	9	9	5	6	7	8
Edit Inline Edit Copy Delete	7	1	1	2	3	9	5	9	9	9
Edit Inline Edit Copy Delete	8	2	9	9	3	4	5	9	7	9
Edit Inline Edit Copy Delete	9	3	9	9	9	4	9	6	7	8
Edit Inline Edit Copy Delete	10	1	1	2	3	9	9	6	9	9
Edit Inline Edit Copy Delete	11	2	9	9	3	4	5	9	9	8
Edit Inline Edit Copy Delete	12	3	9	9	3	9	9	6	7	8
Edit Inline Edit Copy Delete	13	1	2	3	9	9	9	9	7	9
Edit Inline Edit Copy Delete	14	2	9	2	3	4	5	9	9	9
Edit Inline Edit Copy Delete	15	3	9	2	9	9	9	6	7	8
Edit Inline Edit Copy Delete	16	1	1	2	3	9	9	9	9	8
Edit Inline Edit Copy Delete	17	2	1	9	3	4	5	9	9	9
Edit Inline Edit Copy Delete	18	3	1	9	9	9	9	6	7	8

Figure 4.24: Data recorded at fruitanswer table



←T→	daNo	dqIndex	fdIndex	dqPrevAns	dqNextIndex
Edit Inline Edit Copy Delete	1	1	1	0	2
Edit Inline Edit Copy Delete	2	1	9	0	2
Edit Inline Edit Copy Delete	3	2	2	1	3
Edit Inline Edit Copy Delete	4	2	9	1	3
Edit Inline Edit Copy Delete	5	2	2	9	3
Edit Inline Edit Copy Delete	6	2	9	9	3
Edit Inline Edit Copy Delete	7	3	3	2	0
Edit Inline Edit Copy Delete	8	3	9	2	4
Edit Inline Edit Copy Delete	9	3	3	9	4
Edit Inline Edit Copy Delete	10	3	9	9	4
Edit Inline Edit Copy Delete	11	4	4	3	5
Edit Inline Edit Copy Delete	12	4	9	3	5
Edit Inline Edit Copy Delete	13	4	4	9	5
Edit Inline Edit Copy Delete	14	4	9	9	5
Edit Inline Edit Copy Delete	15	5	5	4	0
Edit Inline Edit Copy Delete	16	5	9	4	6
Edit Inline Edit Copy Delete	17	5	5	9	6
Edit Inline Edit Copy Delete	18	5	9	9	6
Edit Inline Edit Copy Delete	19	6	6	5	7

Figure 4.25: Example rules in chili disease answer



	laNo	dIndex	ldIndex1	ldIndex2	ldIndex3
Edit Inline Edit Copy Delete	1	4	1	2	3
Edit Inline Edit Copy Delete	2	5	4	5	6
Edit Inline Edit Copy Delete	3	4	1	2	6
Edit Inline Edit Copy Delete	4	5	4	5	3
Edit Inline Edit Copy Delete	5	4	1	5	3
Edit Inline Edit Copy Delete	6	5	4	2	6
Edit Inline Edit Copy Delete	7	4	4	2	3
Edit Inline Edit Copy Delete	8	5	1	5	6

Figure 4.26: Data inserted at leafanswer table

4.3 System Interfaces

The most important deliverable is the system output itself, which the user will interact with – Chili Disease System.

Figure 4.27: Chili Disease System homepage

Figure 4.27 shows chili disease system home page where there are includes a menu bar at the top of the page.

Figure 4.28: Chili disease type information page

Figure 4.28 shows the chili disease at the fruit part. It includes the symptoms, and how to prevent the disease.

Figure 4.29: Chili Disease System questionnaire

Figure 4.29 above shown the example of the questionnaire that user need to answer to know what type of disease that attack their crops.

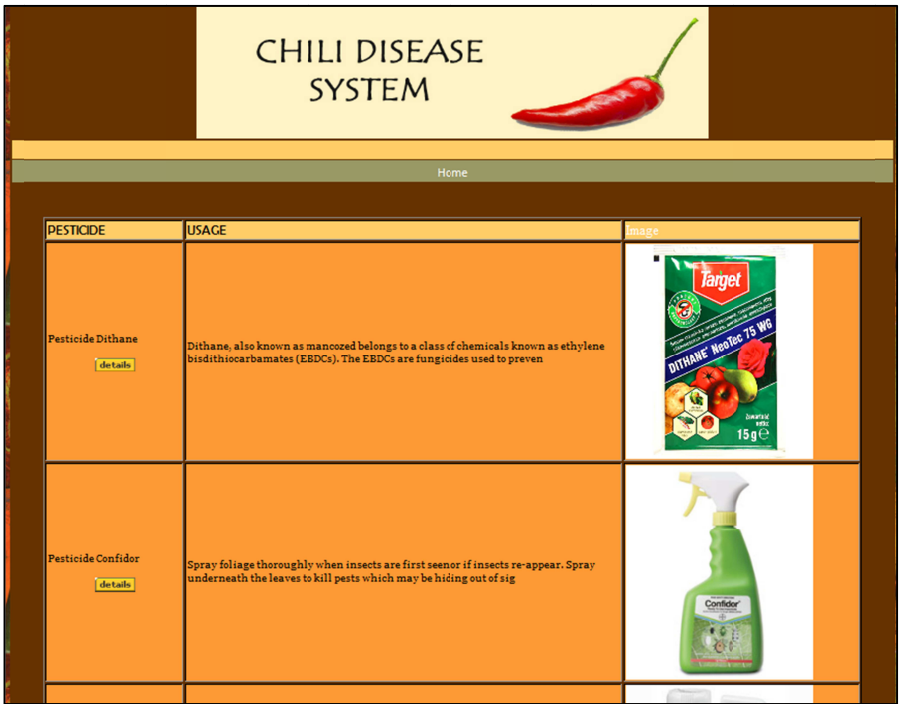


Figure 4.30: Chili Disease System pesticide page

Figure 4.30 show pesticide information that can be applied by user to prevent the disease from getting worse.

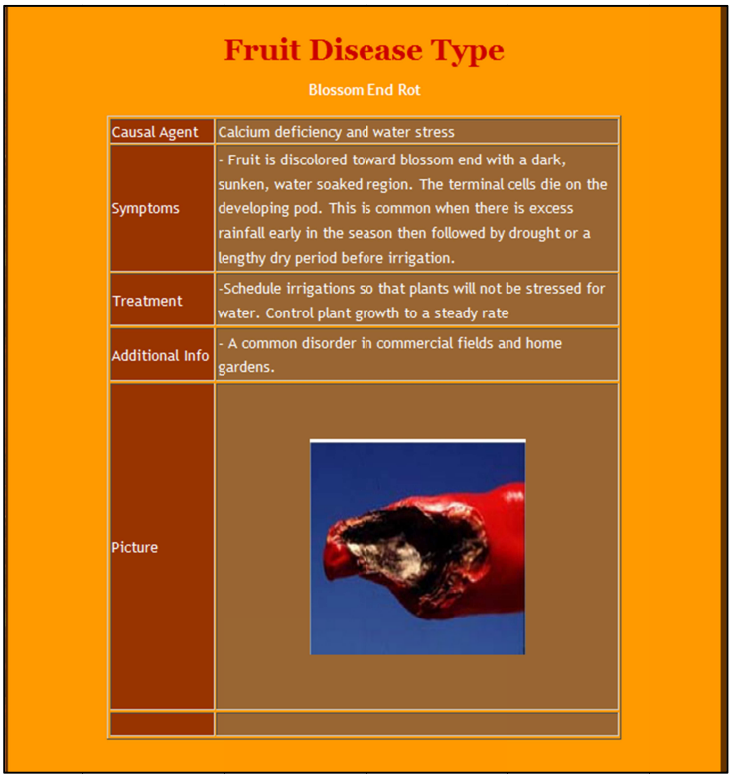


Figure 4.31: Fruit disease type information page

Figure 4.31 is the information details from Figure 4.27.

4.4 Rule-Based Implementation

The rule based model is one of the artificial intelligence models that widely used in many different fields today. The ability of the rule-based to predict the user answer and get the result is tracing by using user answer in rules. The set of rules for these systems are shown in Appendices attach.

4.4.1 Rule-Based Coding Implementation

The structure of the rule-based consist knowledge base that has a rule, database includes the fact, inference engine, explanation facilities and user interface. For this rule-based expert system, each of types that have in rule-based structure is fully connected. The user interface will make a communication with user and the system itself. The answer that user will answer will be process according to the rule that already be set at the database which called as inference engine. Then the inference engine carried out

the reasoning whereby the expert system reaches the solution. It links the rules given in the knowledge base with the facts provided in the database. The result will be tracing in rules by using user answer. If the rules are match with the user answer, the result will appear.

```
<?php
	function updateTemp()
	{
		include_once("infoConfig.php");
		$prev_id = $_GET['pId'];
		$prev_Ans = $_GET['pAns'];
		$uName = $_GET['user'];
		$update = "UPDATE temp SET tAnswer".$prev_id."=".$prev_Ans."
WHERE tUser='$uName' ";
		mysql_query($update);
	}

	updateTemp();
	include_once("ruleConfig.php");
	$curr_id = $_GET['id'];
	$uName = $_GET['user'];

	$qry1="SELECT * FROM diseasequestion WHERE dqIndex=$curr_id";
	$result1=mysql_query($qry1); //to pass a sql query to mysql database.
	if($result1) {
		if(mysql_num_rows($result1) == 1) { // returns number of rows in
result.
			$question = mysql_fetch_assoc($result1);
	?>
```

Figure 4.32: Rule source code to display a question to a system

From figure 4.32, the system will display each of the question by user id and display the question one by one.

```
<?php
                }
        }
        function clean($str)
        {
                $str = @trim($str);
                if(get_magic_quotes_gpc())
                {
                        $str = stripslashes($str);
                }
                return mysql_real_escape_string($str);
        }
        function getAnswer($ch) {
                $curr_id = $_GET['id'];
                include_once("factConfig.php");
                if($curr_id<16)
                {
                        $query="SELECT * FROM fruit WHERE
symptomIndex=$ch";
                        $result=mysql_query($query); //to pass a sql query to mysql
database.
                        if($result)
                        {
                                if(mysql_num_rows($result) == 1)
                                {
                                $sc = mysql_fetch_assoc($result);
                                        $str=$sc["symptomDesc"];
                                }
                        }
                }
                return $str;
```

Figure 4.33: Rule-based source code to shows a symptom to be answer

From the source code of Figure 4.33, the answer will be display according to the fact that stored at database.

```
	$prev_answer = $_GET['pAns'];
	$qry2="SELECT * FROM diseaseanswer da, diseasequestion dq WHERE
dq.dqIndex=$curr_id AND da.dqIndex = dq.dqIndex AND
da.dqPrevAns=$prev_answer";

	$result2=mysql_query($qry2); //to pass a sql query to mysql database.

	if($curr_id<15)
	{
		while($choice = mysql_fetch_array($result2))
		{
		echo"@<a
href='rrrrr.php?id=".$choice['dqNextIndex']."&pId=".$curr_id."&pAns=".$choice['symp
tomIndex']."&user=$uName'> ".getAnswer($choice['symptomIndex'])."</a><br/>";

		}
	}
	else
	{
		while($choice = mysql_fetch_array($result2))
		{
			echo "@<a
href='solution.php?&pId=".$curr_id."&pAns=".$choice['symptomIndex']."&user=$uNa
me'> ".getAnswer($choice['symptomIndex'])."</a><br/>";
		}
	}
?>
```

Figure 4.34: Rule-based source code to trigger a rule from database

CHAPTER 5

RESULTS AND DISCUSSION

5.1 Analysis of Result

The purpose of this chapter is to identify and discuss the result of the Chili Disease System (CDS). The analysis of the result can be done by analyze the objective of the system, development constraint. The purpose of the analysis is to ensure the development of the CDS has meets the user requirement. The developments of CDS are based on the two objectives which have been met in this system:

iii. To develop a web-based prototype application in finding the Capsicum f. diseases.
iv. To implement the rule based expert system in the application that capable to detect diseases of Capsicum frustences.

The system already meet the (i) and (ii) objective by create a website and providing the information to the user about the disease of chili. The information covers by the information about the type of chili disease and symptom of the disease. Other than that, it also provides type of pesticide that can be used to prevent the disease. This

system was developed by using web-based programming language such as HTML, CSS, Javascript and PHP. The system also be able to diagnose the disease by answering the question that been given. Figure 5.0 shows the answer or result of web-based expert system for diagnose the disease by answering the symptom question and the result of the question.

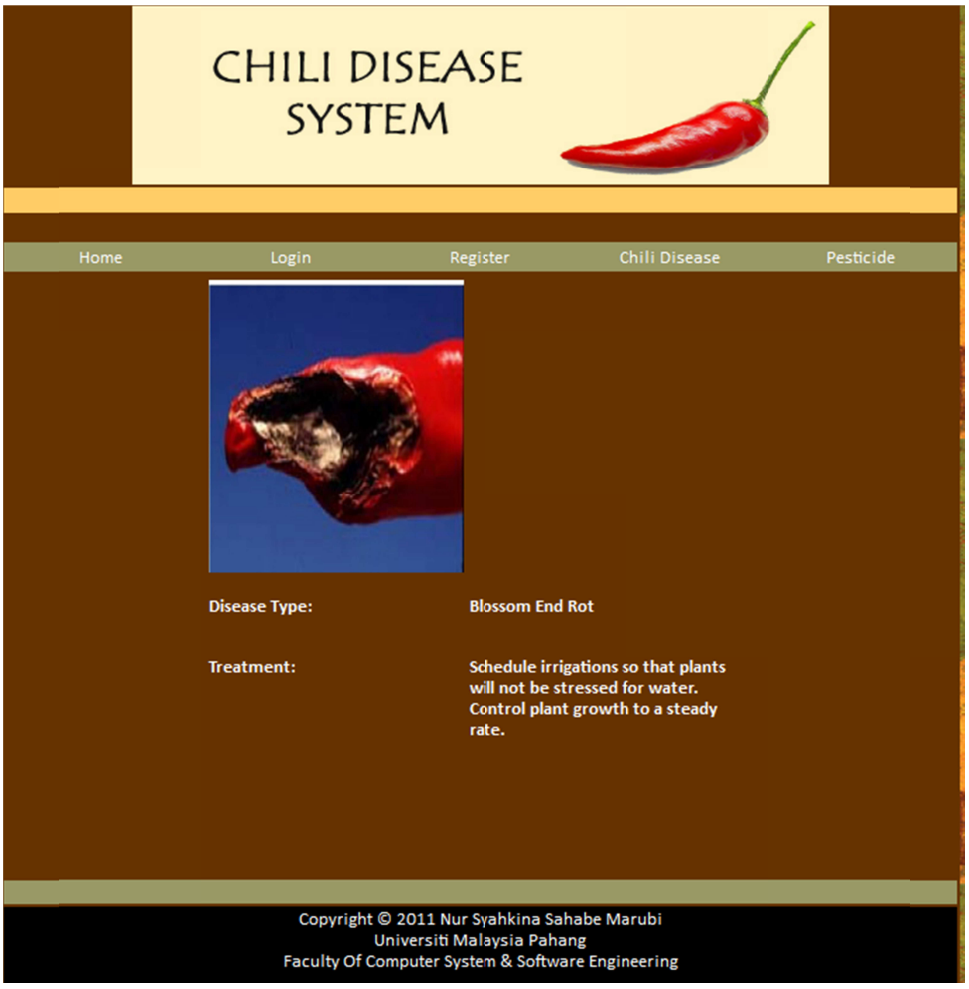


Figure 5.0: Diagnose of Chili Disease Result

Figure 5.1 is the coding to get the result of the diagnose. The system was checked the user input with the rule that has been designed at the database.

```
function
getChiliDiseaseType($a1,$a2,$a3,$a4,$a5,$a6,$a7,$a8,$a9,$a10,$a11,$a12,$a13,$a14,$a15)
{
include_once("ruleConfig.php");
$select1 = "SELECT * from fruitanswer WHERE symptomIndex1=$a1 AND
symptomIndex2=$a2 AND symptomIndex3=$a3 AND symptomIndex4=$a4 AND
symptomIndex5=$a5 AND symptomIndex6=$a6 AND symptomIndex7=$a7 AND
symptomIndex8=$a8 AND symptomIndex9=$a9 AND symptomIndex10=$a10 AND
symptomIndex11=$a11 AND symptomIndex12=$a12 AND symptomIndex13=$a13 AND
symptomIndex14=$a14 AND symptomIndex15=$a15 ";
$result1 = mysql_query($select1);
if($result1)
{
if(mysql_num_rows($result1) == 1)
{
$d = mysql_fetch_assoc($result1);
$dType=$d["dIndex"];
}else
{
echo "If rule failed";
}
}else{
echo "Query getChiliDiseaseType() failed";
}
return array ($dType);
}
```

Figure 5.1: Result coding

The second objective has been met by developed system as this system applies the component of rule based system technique which the component are knowledge base(rule), inference engine and working memory(database). The working memory will holds all the user input(symptom) that are needed to diagnose disease by match it with the fact in database by applied forward chaining technique. The knowledge base is shown in Figure 5.2.

dIndex	symptomIndex1	symptomIndex2	symptomIndex3	symptomIndex4	symptomIndex5	symptomIndex6
1	1	2	3	0	0	0
2	1	0	3	4	5	0
3	1	8	8	8	8	6
1	1	2	3	4	8	8
2	1	8	3	4	5	6
3	1	8	8	8	5	6
1	1	2	3	8	5	8
2	8	8	3	4	5	8
3	8	8	8	4	8	6
1	1	2	3	8	8	6
2	8	8	3	4	5	8
3	8	8	3	8	8	6
1	2	3	8	8	8	8
2	8	2	3	4	5	8
3	8	2	8	8	8	6
1	1	2	3	8	8	8
2	1	8	3	4	5	8
3	1	8	8	8	8	6

Figure 5.2: Knowledge base (rule)

5.2 Project Constraint

This project could be say as one of the challenges, because we had to manage our time wisely, in order to deliver it on time. Programming skills is the most important thing in developing a good system. We need to explore by own on how to use the related

tools so that the task given could be well done. It takes time to sharpen skills because the lecture given does not same with the practical.

In other to build the system better, the data need to be collected from the entire expert. But as know, the staff does not have enough data that needed in this system. There is no accurate data of result in CDS, therefore the difficulties of achieving and delivering formation about research.

5.3 System Constraint

CDS is a web based prototype to diagnose the disease of oil palm. It is will be useful system if this system online that can be access via internet, but not all user can access this system due to the problem they have. Not all users afford to get internet connection due to the location.

CDS only enables user to answer the question and give suggestion of the disease that they have might face. While the admin can only view, add, and update the disease and pesticide information.

5.4 Suggestion and Improvement

There are several suggestion and improvement of CDS:

i. Add more type of disease of chili to be diagnosed in CDS. If there is more disease, more knowledge and information also provide for user.

ii. Implement CDS in mobile technology such as PDA and mobile phone. User can be able to access system from anywhere and everywhere through PDA and their mobile phone by downloading the system into their devices.

CHAPTER 6

CONCLUSION

This chapter is devoted to conclude and summarize the project overall performance that brings to a close synopsis about all of the relevant information in this system concisely.

6.1 Summary

As a conclusion, the development of CDS had fulfilled all the requirements and also meets the objectives, problem statement and scopes of the system. The CDS is an expert system based on web application to detect and diagnose the disease of chili.

The tools that used in develop this system is Adobe Dreamweaver CS5 v11 for design the interface and PHP coding also MySQL as the database.

The techniques used in developed CDS are rule based expert system. Forward chaining is a means of utilizing a set of condition-action rules. In this mode of operation, a rule based system is data-driven. The data used to decide which rules are match and can fire, then it fire one of those rules, which may add to the data in working memory and then repeat the process until its establish a conclusion of result.

The IF-THEN structure used in forward chaining as inference engine process to get result of chili disease.

The Rapid Application Development method is used in developing CDS system. This methodology applied the concept of expert system on each of its stages to ease the system development process and fulfill the requirement.

References

[1] *The Chillies*. SoilMates. May. 8th 2011

[2] K. Darlington, "The Essence of Expert Systems", 2000

[3] E.Turban, J.E. Aronson, and T.P. Liang, "Decision Support Systems and Intelligent Systems", 7th Edition, Prentice Hall, 2005.

[4] S. Russel and P. Norvig, "Artificial Intelligence: A Modern Approach", 3rd Edition, 2010

[5] R.S. Engelmore, and E. Feigenbaum, May 1993

[6] Patridge, 2001

[7] A. Morsy, 2000

[8] J. Ireson, Paine, 1996

[9] James Freeman, Hargis, 2003

[10] Negnevitsky, 2005

[11] W.P. Wagner, 2003

[12] Hargis, 2003

[13] Irny, S.I. and Rose, A.A, "Designing a Strategic Information Systems Planning Methodology for Malaysian Institutes of Higher Learning (isp- ipta), Issues in Information System, Volume VI, No. 1, 2005

[14] *Cara Pengawalan Penyakit*, http://gemstani.blogspot.com

[15] M.A. Jackson, A.W. Roscoe (Eds.); *Chapter 13 of A Classical Mind: Essays in Honour of C.A.R. Hoare*, Prentice-Hal. 1994

[16] Whitten, 2004

[17] Totem, *CASEMaker.* 2004

[18] T. Mochal, July. 20th 2001: