Chapter 5

Analysis and Design

5.1 Introduction

Previous chapter described the approach used to develop the Agricultural Information system. It also included the information such as the type of inputs and outputs of the system, who are the users of the system, main processors, features and technological background of the implementation.

In this chapter, we introduce the architectural and designing prospective of the system. It introduces the modules of the system, what each module does, how each module integrated with other related modules and information related to the components of each module.

5.2 Main modules of the system

The system was implemented with web and SMS front ends and it enabled the easy accessibility for all users. This system was designed as three separate modules. Namely they are, communication module, agent module, and system access module. These modules consist of several components depending on the functionality of the each module.

Figure 5.1 shows the high level system architecture diagram which illustrates components of each main module and how all components are integrated with each other.

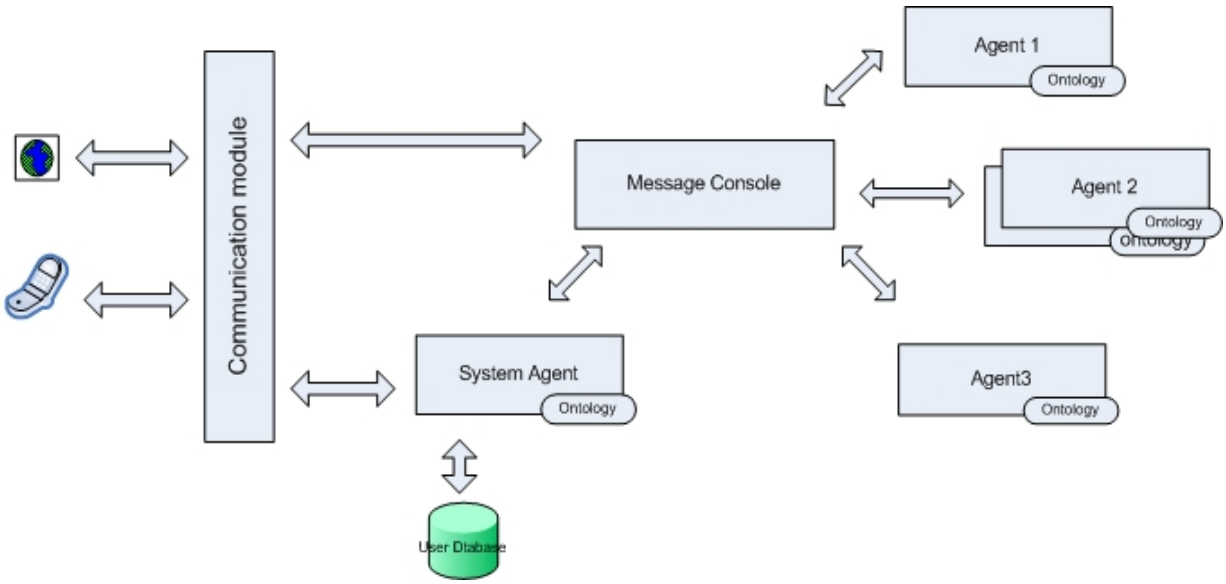

Figure 5.1: High level system architecture

The following sections will describe each module in detail describing the components of each module with their features and functionalities.

5.2.1 Communication module

Web front end and SMS module of the AIS were linked with the back end agent module using communication module. When user place a request through the web or SMS channels, communication module manages the requests based on the link and direct it to the agent module for processing. This is important when it comes to addressing the limitation of mobile channel. Furthermore communication module is also responsible for initialisation and start-up of the agent module. This is done by starting the message agent and consequently message agent will initiate the rest of the agents in the system.

5.2.2 Agent module

Agent module can be considered as the most important part of the implementation. It could be considered as the brain of the system. It consists of several agents, ontologies of these agents, knowledge bases and message console. Number of agents could vary depending on the implementation. For example when it comes to certain queries there may be several market agents participating depending on the problem and the

situation. This module has been implemented in such a way that any number of agents can plug in to the system depending on future needs. In order to cater this it is required to adhere to a common message interface which enables the communication between agents in a common language.

5.3 Type of Agents

It is possible to identify eight agent types in general. They are namely Message Agent, System Agent, Fertilizer Agent, Environment Agent, Pest Agent, Soil Agent, Market Agent and Crop Agent. However System may contain any number of instances from these agent types. These agents have their own knowledge base and their own ontologies based on the sub domain which they are operating. In addition it is also possible, that during the initialization some of the agents may have multiple appearances with different knowledge bases. For instance several agro sellers can be initialized as several market agents with different knowledge bases with the help of common or different market ontologies. Following subsections will describe the basic functionalities of these agents.

5.3.1 Message Agent

Message agent is responsible for system initialisation and communication with external systems such as web and SMS channels. It communicates with front end module via the communication interface and initialises the agent system as well as the message console. Message agent does not have any ontological knowledge representation or knowledge base as it does not participate in the process of problem solving. However it consists of a simple rule base to manage its functionality itself.

5.3.2 System Agent

System agent is mainly responsible for providing and managing user information and their cultivations. For example it provides information such as farm location, what are the cultivated vegetables, cultivated dates, medium, type of cultivation (Organic, Mixed). This information is used by other agent to resolve queries. On the other had these information could be considered as more structured and fixed. Therefore it is possible to represent this using a database technology.

5.3.3 Fertilizer Agent

Fertilizer agent consists of an ontology and a knowledge base. It will help to solve the queries from the fertilizer point of view. Its domain ontology and knowledge base contain information such as Organic and Inorganic fertilizers types, their usage, deficiency symptoms, excessive usage symptoms, different verities of fertilizers, and general usage. Depending on this knowledge base, fertilizer agent can help with the solutions from the fertilizer point of view to a given inquiry.

5.3.4 Environment Agent

Environment agent is responsible for providing feedback to a query related to an environmental condition. It consisting the information such as weather and climate information based on the location and duration, environmental related information such as CO2 concentration, Humility, Temperature, wind pattern and how these parameters effect on the growth of the crops and the harvest in the given climatic zone.

5.3.5 Pest Agent

Pest Agent contains an ontology and knowledge base related to pest and weed control. It is responsible for providing feedback to a query related to the control of pests and weeds. The knowledge base contains information such as identification and diagnosis of pests and weeds, methods of control such as organic and inorganic techniques, and greener techniques such as minimising the usage of inorganic pesticides.

5.3.6 Soil Agent

Soil agent is responsible for providing feedbacks related to soil conditions in agricultural lands. It closely work with system agent to retrieve the soil condition of a given farm and analyse its knowledge base consisting the information such as various soil types, soil structures, PH values of soil, how PH value effect on growth, different kind of textures, information related to variation of growth with the temperature of the soil and colour related information.

5.3.7 Crop Agent

Crop agent is responsible for providing feedbacks related to crops and seeds and optimum hybrids. Its ontology contain information related to crop plantations, type of seeds, information related to fertilisation, seed controlling, time required to cultivate them in farm and farming requirements.

5.3.8 Market Agent

Market agents always concentrate on marketing aspects of agriculture products and it handle both buying and selling of those. It also contains a knowledge base and market ontology. It contains marketing related information like price, demand, supply, quality, quantity and available locations. This is a common ontology for vegetable marketers as well as agro-chemical and agro-accessories market. For instance several agents may share the same basic ontology with some addition of their own knowledge.

5.4 Ontologies

Each agent has its own ontological representation of domain knowledge. AIS system uses several representation techniques to represent ontological aspects. There have been lot of methodologies available for the ontological knowledge representation. For instance more structural knowledge such as user information and cultivation related information could represent as database structures. On the other hand it is possible to store both ontology and knowledge base in the same structure. For example the fertilizer ontology and knowledge base represented in XML. At the same time some of the ontological representations were done with the help of an ontology definition tool.

5.5 Message Console

Message console is the main component of the agent communication. It enables the communication between all agents. During the communication agents can write their requests and responses on the message console. It is visible to all agents and once an agent place a request on the message console all others can response to that by writing

their responses on the message console. There are different types of message formats used to enable the communication.

5.6 Agent communication

Communication can be considered as the problem solving approach when it comes to agent technology. It mimics the way that humans handle ad solve the complexities in real life. There are various standards and communication protocols available for multi agent communication. For example KQML is considered as the first agent communication language developed in early 1990 as the part of US government's knowledge sharing project. In addition to that FIPA ACL also available and it is considered as the most used and studied agent communication language.

Nevertheless it is also possible to define an own communication protocol for a particular multi agent implementation. When it comes to AIS, we have defined our own communication protocol instead of standard protocols. It enables the simplicity of the communication among different agent entities and ultimately enhances the performance of the agent module.

This is very crucial as message processing adds an extra overhead over the multi agent module of the AIS. Table 5.1 represent the message formats that are used in AIS implementation.

Message	format	Description	Example
REQ	REQ: <request>	Place a request on the message console.	REQ: Yellow leaf
RES	RES:<response>	Place a response on the message console	RES: This is a deficiency symptom.
FREQ	FREQ: <request>	If an agent wants further information about a request, it	FREQ: Nitrogen deficiency.

		can use this message.	
FRES	FRES: <response>	Place a response to a further request.	FRES: Nitrogen is an Inorganic fertilizer.
NEGO	NEGO: <item>	This message uses with the state of negotiation between several agent.	NEGO: Organic Nitrogen
INFO	INFO :<from Agent |to Agent (request list)>	Request some information from a particular agent. For example Environment agent request information about the farm location from the system agent.	INFO: ENV|SYS(location)
IRES	IRES:<from Agent | to Agent(response list)>	Response to an information request. For example this shows a response posted by system agent to environment agent.	IRES:SYS|ENV(Bandarawela)
MPRA	MPRA: <Parameter>	This message indicates a mandatory parameter. For example sample message inform Organic Fertilizer as a mandatory parameter.	MPRA:Organic-Fertilizer
REJ	REJ:<condition>	This message uses to reject any condition. For example depending on their knowledge agents could reject some responses of other agents. For example suppose Soil agent respond “lack of water “ as the reason for “Yellow leaf”, but according to Environment agent this could be raining season.	REJ:lack-of-water

TER	TER:ENV	This indicates of a termination of an agent. This is important that other agents can see the termination of given agent.	TER: ENV
END	END	Indicate if the agent modules idle for some times. This is the indication that agent cannot help any more referring the query.	END

Table 5.1: List of message formats used in the AIS

5.7 Knowledge acquisition and validation

Knowledge acquisition is one of the primary requirements with Multi Agent Systems (MAS). When it comes to solving real world problems with MAS, knowledge as well as steadily defined domain ontology is essential. Each agent will have its own module to alter the knowledge base which enables the learning of new facts and knowledge. It includes mechanisms to validate existing rules, existing facts and learn new things. Figure 5.2 shows the system architecture of the learning module.

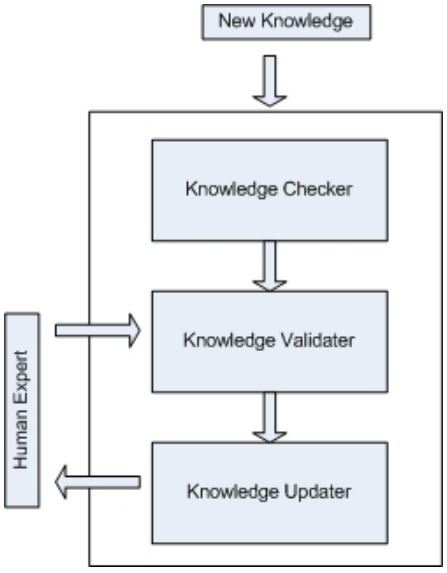

Figure 5.2: Knowledge acquisition module architecture

Learning module consists of three main parts, namely knowledge checker, knowledge validation module and knowledge update module. When a new fact or information arrived to the system, knowledge checker will match it with the existing knowledge base. If it is considered as new fact, knowledge validation module will try to validate the fats with the existing knowledge. At the same time, knowledge validation module can ask help or confirmation for the facts from a human expert. After the completion of knowledge validation, knowledge update module will update the knowledge base.

5.8 Rule processing

Rule processing is one of the mandatory requirements when it comes to a highly dynamic system. For example agents such as marketing, required to alter the rule and actions based on the business logic that are operating and business owner's requirements. These changes cannot address and predict or plan during designing phase.

However rules enable business analysts and developers to build decision logic based on the requirement during designing. It is possible to view a rule engine as a sophisticated interpreter of if-then statements and usage of the rule engine enables the dynamic changes that is required by certain inter connected domains.

5.9 User Information

Users of the system should register themselves in order to enable the full accessibility. At the registration point users should provide basic information such as user name, password, contact information and location information. After the registration users can log in to the system and they can maintain their farm related information within the system. This information will be used by the system agent to support the decision making process.

5.10 System access module

Accessibility is one of the primary requirements for any information system and it is directly related to the usability of the system. When compared with stand-alone

applications, web based solutions could be considered as the most promising approach which enable the usability and accessibility. Nowadays in Sri Lanka, lot of initiatives has been taken to provide Internet facilities to rural areas. Furthermore various mobile networks are able to provide Island wide coverage. All these factors added up to the fact that rural empowerment of ICT will provide a future to AIS. System access module of AIS consists of two main parts, WEB module and SMS module.

5.10.1 Web front end

Web based system accessing module facilitate the functionalities such as user registration and maintain user profiles. Furthermore it facilitates the registration of farms where farmer can register their farms within the system. Registration process requires the basic information to provide access to the system and the information about the crops cultivated. For example it include the information such as vegetable planted, seeds used, location of the plantation, what type of fertilizer used, types of pest controllers used. Registered information will be used by the agent module during the problem solving process.

5.10.2 SMS enabled access

SMS module facilitates only the limited functionality. For example users can get the general information from the system such as market movements and fertilizer prices through SMS. When the request is received, communication module directs the request to the common message console and mean time all the available agents are created. Depending on the request some of the agents will be terminated and some of them will remain alive. This process will happen depending on the agent's ontology and their knowledge.

Depending on the ontology and knowledge, agents will propose and communicate with each other to solve the request. For instance there may be more than one solution for a given problem. This is very normal with real life agriculture related solutions also. Finally the solution will be sent to the user with the reasons and explanations. This includes alternative solutions, suggestions, and explanations as well.

5.11 Summary

This chapter contain all the information related to the design aspects of the Agricultural Information System. It started with the introduction of the main modules of the system. Then it describe the functionality and components of each module and there functionalities.

The next chapter will present the information about how above modules and components were implemented and the technologies and tools used for the implementation.