ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES
FACULTY OF INFORMATICS

A Multi-Agent Decision Support Model for Medical Referral Indication

BY
TSEGAYE AYELE

OCTOBER 2008

A Thesis submitted to the School of Graduate Studies of Addis Ababa University in partial fulfillment of the requirements for the degree of Master of Science in Computer Science

A Multi-Agent Decision Support Model for Medical Referral Indication

BY

TSEGAYE AYELE

Name and Signature of Members of the Examining Board

No.	Name	Role	Signature
1	Dr. Mulugeta Libsie	Advisor	
2			
3			
4			

TABLE OF CONTENTS

LIST OF FIGURES

ACRONYMS AND ABBREVIATIONS

AMS:	Agent Management System
CKB:	Clinical Knowledgebase
DF:	Directory Facilitator
DSS:	Decision Support System
FIPA:	Foundation of Intelligent Physical Agents
GP:	General Practitioner
GUI:	Graphic User Interface
HAS:	Healthcare System Analyst Agent
IDE:	Integrated Development Environment
JADE:	Java Agent Development
JDK:	Java Development Kit
JRE:	Java Runtime Environment
KBWA:	Knowledge-base Wrapper Agent
LC:	Local Consultation
MADM:	Multi-Attribute Decision Making
MADMA:	Multi-Attribute Decision Making Agent
MADS:	Multi-Agent Decision Support
MADSS:	Multi-Agent Decision Support System
MARDS:	Multi-Agent Referral Decision Support
MAS:	Multi-Agent System
MASE:	Multi-Agent Software Engineering
MCA:	Medical Consultant Agent
MCDM:	Multi-Criteria Decision Making
MCDM:	Multi-Criteria Decision Making
PAA:	Patient Analyst Agent
PCP:	Primary Care Provider/Physician
PHA:	Physician Analyst Agent
PHA:	Physician Analyst Agent
RCA:	Referral Coordinator Agent
RGK:	Referral Guideline Knowledgebase

RI:	Referral Indication
SAA:	Service Analyst Agent
SE:	Standard Edition
SI:	Service Identification
UIA:	User Interface Agent

ACKNOWLEDGEMENTS

First of all, I praise God and I am very much grateful to Him for providing me the grace, resources, mercy and wisdom towards my success in every aspect of my life.

Next, I would like to thank all those who have made this thesis possible. Obviously, my special thanks goes to my advisor Dr. Mulugeta Libsie for taking consistent effort in reading this thesis work and providing me with invaluable comments throughout this thesis work. I have a great deal of honor and respect to Dr. Mulugeta for his professionalism, seriousness, and completeness while advising me on this work. I also extend my appreciation and thanks to members of medical staff of St. Paul Generalized Specialized Hospital who have dedicated their time for the interview that has been conducted to capture the domain experts' input in the characterization of the problem domain. I can't thank enough Zewdu Gebeyehu for his in-depth involvement, unreserved support and shaping of this thesis with useful insights and advise from the start to the end of this study.

I am also grateful to ActionAid for allowing me to manage my time with flexible working style while pursuing my studies on part time basis. I wish to extend my appreciation and thanks to my friend Semeneh Ayalew who has provided me editorial comments by reading my thesis. Thanks to my brother Dr. Frehun Ayele for providing professional input while considering clinical knowledge examples in the prototype development.

Lastly, but not least, I would like to thank my wife Hiwot Hailu, my son Nathan and my daughter Nita, without whom, this research would have not been possible. Thank you for your love and unreserved support to complete my study.

ABSTRACT

Medical Referral Decisions are clinical decisions by which clinicians determine referral indication, type of required services and selection of appropriate providers in medical referral systems. A referral indication is a decision made by clinicians to determine whether referral is needed or not for a patient case under consideration. These decisions are made in a clinical environment through referral systems that aim at providing of efficient healthcare services by improving patient outcomes and decreasing cost incurred and time spent for such services. The quality of referral decisions is highly dependent on the efficiency and soundness of the decision making process. This inherently complicated referral decision process depends on a complex mix of both clinical and non-clinical factors such as patient, clinicians and healthcare system determinants. Recently, a Multi-Agent Referral Decision Support (MARDS) framework [6] has been proposed with the aim of improving the quality of referral decisions. However, it doesn't fully address the referral indication aspect that is a key component of the medical referral process, which may cause under-referral and over-referral problems.

This thesis proposes a Multi-Agent Decision Support (MADS) model for Referral Indication, Service Identification and Local Consultation aspects of medical referral aimed at providing improved decision support to clinicians. The proposed decision support model undertakes the analysis of determinants related to the referral indication, service identification and local medical consultation. This aid is provided through the social agents that interact and cooperate in the clinical environment, which are designed to interact with the existing CIS (Clinical Information System) and the CKB (Clinical Knowledgebase) to fetch critical information which supports the analysis of decision making.

It is believed that this model extends the MARDS framework by addressing the referral indication and service identification aspect for its realization. Moreover, the local medical consultation service is believed to address the communication and organizational challenges of the medical consultation process and in turn contributes for the minimization of over-referrals and helps to overcome clinical uncertainties.

Keywords: Medical Referral Decisions, Referral Indication, Service Identification, Local Consultation, Multi-Agent Systems, Decision Support Systems.

CHAPTER 1 : INTRODUCTION

1.1. Background

In recent years, agent-based systems have emerged in IT as one of the most exciting and significant areas of research and development [1]. These systems have been used as approaches for designing and implementing autonomous, intelligent and social software assistants capable of supporting human decision making [2]. Studies [3, 4, 5] show that multi-agent systems have a set of characteristics that make them appropriate to be used to improve the provision of healthcare to citizens. Moreover, features of an application area that indicate an agent-oriented approach are identified by these studies. These features include: complexity of software solution, distributed nature of data, need of maintaining independence between healthcare providers, need of coordination among healthcare providers, lack of centralized solution, and others. Accordingly, the medical referral system is a potential application area characterized by the above-mentioned features.

Medical Referral Systems aim at achieving high standards of care by improving patient outcomes and decreasing costs through optimal use of medical services. The success of these systems is highly dependent on the quality of referral decision by which physicians determine referral indication, type of required services and selection of appropriate providers. In a referral process, primary care physicians determine referral indication (whether referral is needed or not) based on different determinant factors. However, this is not usually an easy task due to physician's clinical uncertainty and limited scope of practice [6].

Nowadays, many research works have been done in the areas of e-Health where a number of such projects are given high attention and are funded by countries of the developed world. In the developing world, there is a significant shortage of medical resources compared to the population that demand health care services. It is needless to mention that Ethiopia is not an exception to this situation.

In the past few years, various research works have been conducted to propose an electronic solution to the problems of referral management with an attempt of improving the efficiency of the overall process. Most recently a Multi-Agent Referral Decision Support (MARDS) framework has been proposed as a basis for developing models that improve the quality of referral decisions [6]. Specifically, a model for provider selection has been developed to be employed by the framework that helps a physician in making a better provider selection. However, it has not fully addressed the referral indication and service identification decision aspects of medical referral. This in turn limits the realization of the proposed framework.

Referral indication is one of the basic aspects of a referral process where physicians need consultation and decision support to make appropriate referral as well as overcome clinical uncertainty. If referral is indicated, then there is a need to identify appropriate medical services before transferring the referral to the service provider. Otherwise, if it is out of their scope, physicians should be provided with appropriate medical consultation in order to deal with the patient's case locally. The complexity of the process often causes inefficient referral indication decisions, which in turn affect the quality and cost of care [7]. Therefore, the aim of this thesis is to study and develop a multi-agent referral indication decision support model that could improve the medical referral process by addressing the above-mentioned problems.

1.2. Problem Statement

Several studies have been conducted in the past few years on e-Referral Systems to improve the efficiency of the referral process. However, most of these studies have focused on the communication aspect of referral process and less attention is given to the development of intelligent decision support for physicians. On the other hand, research work in [6] proposed a Multi-Agent Referral Decision Support framework that improves the overall referral process by extending the capabilities of current e-Referral systems with referral decision support. However, only the provider selection aspect of the referral process has been given emphasis. Besides, there are no adequate mechanisms for the local medical consultation to overcome clinical uncertainties and over-referrals.

Hence there is a need for more research on the remaining aspects of the referral process, specifically the referral indication, service identification and local medical consultation in order to improve the overall referral process aimed at quality and cost-effective patient care. Thus, the problem this thesis tries to address is how to enhance the MARDS framework by developing a multi-agent referral indication, service identification and local medical consultation decision support model. This is believed to contribute for the realization of the framework and also to enhance the overall outcome of patient care and cost by improving the quality of referral decisions.

1.3. Objectives

The general objective of this research is to design a multi-agent decision support model for referral indication, service identification and local medical consultation that enhances the existing MARDS framework.

The followings are the specific objectives:

- Analyzing components and related services of the referral indication decision and studying the associated determinants that affect physicians' decisions.
- Obtaining domain experts' (medical staff) input on referral decision making process focusing on the referral indication process.
- To study determinant factors that affect referral indication, service identification and local medical consultation decisions.
- To design a multi-agent model for referral indication, local medical consultation aspect to be employed by the referral decision support framework.

1.4. Research Activities

The initial step of this research has been review of relevant literature to study multi-agent systems and their applications to the medical referral system. Then related research works that motivated this study have been reviewed. At this stage, MARDS framework has been abstracted and areas that this research has tried to address have been pointed out.

Then, the analysis of the referral indication decision making process and factors that affect the decision in relation to the overall referral decision making process have been studied. Here, the characterization of the domain focuses on referral indication, service identification and local medical consultation processes and decisions. The domain analysis was supported by experts' input on referral decision making process. The medical staff members including physicians and a triage nurse of St. Paul Generalized Specialized Hospital have been interviewed with questions specifically designed for this purpose. St. Paul Generalized Specialized Hospital has been preferred due to their relevance for the study and previous support and contact for similar study. The interview questions design is annexed in Appendix I. During this time, we had a general discussion with the Medical Director of the hospital on our information need and we have reached on an agreement on the key respondents based on their role and experience in the area of the study. Accordingly, three specialists (ANT, triage and surgical) and a Head Nurse have been assigned and we were able to interview two of the specialists and the triage nurse, who was a representative of the head nurse. The key feedbacks that are believed to contribute for this study are captured and described in the analysis and characterization of the referral indication decision.

After the background study and domain analysis, the design and development of the referral indication decision support model has been conducted with the help of some identified multi-agent systems design methodologies. Then, a prototype is developed for the model based on agent implementation tools.

1.5. Thesis Organization

The rest of the thesis is organized as follows.

Chapter 2 presents the literature review of concepts that form the basis for this research. These include the review of software agents, agent technology and multi-agent systems along with their application to decision support systems. It also discusses the MARDS framework in relation to the application of multi-agent systems for medical referral decision support.

Chapter 3 discusses some related works that were done with regards to the application of multi-agent technology to the problems of medical referral.

In Chapters 4, the concepts and features of referral indication decision making process and factors that affect the decision are presented. It discusses the relation and impact of regular clinical process on referral indication, service identification and medical follow-up processes and decisions.

Chapter 5 presents a multi-agent decision support model developed for medical referral indication and related services. This chapter discusses the proposed Multi-Agent Decision Support (MADS) model for Referral Indication that extends the MARDS framework to support referral indication, service identification and local medical consultation aspects. The model is described in terms of the design goals and its detailed architecture that includes a decision aid strategy that agents employ.

In Chapter 6, a prototype implementation of Multi-Agent Decision Support (MADS) model for Referral Indication, Service Identification and Local Medical Follow-up decisions is discussed.

Finally, Chapter 7 presents a general conclusion, thesis contributions, challenges and possible future works.

CHAPTER 2 : LITERATURE REVIEW

This chapter presents the review of literatures on the concepts that are a basis for the research. Software agents, agent technology and multi-agent systems are discussed along with their application to decision support systems.

2.1. Software Agents and Agent Technology

2.1.1. Software Agents

Wooldridge in [8] defined an **agent** as a computer system that is situated in some environment and that is capable of autonomous action in this environment in order to meet its design objectives. An agent has a capability to function in dynamic, unpredictable, independent, and social environment that may contain other agents and processes [1].

The behaviors that best characterize agents and the most commonly accepted ones are autonomy, reactivity, pro-activity and sociability [3, 4, 5].

Autonomy: is a common behavior exhibited by all agents. This behavior refers to the ability of an agent to take its own decisions, based on its internal state and the information that it receives from the environment. As a result, agents provide an ideal pattern to implement systems in which each component models the behavior of a separate entity that wants to keep its autonomy and independence from the rest of the system.

Agents could be considered as **intelligent** entities when they exhibit capabilities of reactivity, pro-activity and sociability [3, 8].

Reactivity: refers to intelligent agents' ability to perceive their environment and respond in a timely fashion to changes that occur in their environment. This behaviour enables agents to satisfy their design objectives and to process external events.

Pro-activity: Intelligent agents are able to exhibit goal-directed behavior by taking the initiative in order to satisfy their design objectives. Using this behavior, agents could perform tasks that may be beneficial for the user, even if he/she has not explicitly demanded those tasks to be executed. For instance, an agent may find relevant information and shows it to the user before he/she requests.

Sociability: Intelligent agents are capable of interacting with other agents (and possibly humans) in order to satisfy their design objectives. This behavior shows the capability of intelligent agents to communicate among themselves using some kind of agent communication language, in order to exchange information. This way they can engage in complex dialogues, in which they can negotiate, coordinate their actions and collaborate in seeking solution to a problem. Using this behavior, agents are engaged in complex decision making processes in environments which may be dynamic, unpredictable, and uncertain.

2.1.2. Agent Technology

Agent technology has been described as a new metaphor of computation and its enabling technology in its recent roadmap. With the growth of the Internet and the World Wide Web over the last one and half decade, it has been a stance where a new computational metaphor is required. This metaphor is referred to as **computation as interaction**. In this metaphor, computing is something which happens by and through communication between computational entities. It is an activity that is inherently social, rather than solitary, leading to new ways of conceiving, designing, developing and managing computational systems. Agent technology is a solution that is introduced to exploit this new metaphor of computing as social activity, as interaction between independent and sometimes intelligent entities, adapting and co-evolving with one another [1].

In the same roadmap, agent technology has been considered from three perspectives, each outlined as follows:

Agents as Design Metaphor: Agents provide software designers and developers with a way of structuring an application around autonomous, communicative components, and lead to the construction of software tools and infrastructure to support the design metaphor. In this sense, they offer a new and often more appropriate route to the development of complex computational systems, especially in open and dynamic environments.

Agents as Sources of Technologies: Agent technology is believed to support a range of specific techniques and algorithms for dealing with interactions in dynamic, autonomous, and open environments. These address issues such as balancing reaction and deliberation in individual agent architectures, learning from and about other agents in the environment, eliciting and acting upon user preferences, finding ways to negotiate and cooperate with other agents, and developing appropriate means of forming and managing organizations.

Agents as Simulation: Multi-agent systems offer strong models for representing complex and dynamic real-world environments. For example, simulation of economies, societies, healthcare and biological environments are typical application areas.

In this study, agent technology will be considered as a design metaphor to come up with medical referral models through structuring of autonomous and communicative agents in healthcare domain.

Problems of the complex, realistic, and large-scale nature are beyond the capabilities of an individual agent. This is because the capacity of an intelligent agent is limited by its knowledge, its computing resources, and its perspective. This is one of the underlying reasons for creating problem-solving organizations and why multi-agent systems have been found to be ideal for such an approach [9].

2.1.2. MULTI-AGENT SYSTEMS

A **multi-agent system** (MAS) can be defined as a collection of autonomous agents that interact with each other, typically by exchanging messages through some computer network infrastructure, to coordinate their activities in order to be able to solve collectively a problem that could not be tackled by any agent individually [8].

In general, the agents in a multi-agent system will represent or act on behalf of users or owners with very different goals and motivations. In order to successfully interact, these agents thus require the ability to cooperate, coordinate, and negotiate with each other, in much similar way that people cooperate, coordinate, and negotiate with each other in everyday life [8]. In recent years, it has been argued that multi-agent systems may be considered as the latest software engineering paradigm [9].

The main characteristics of MASs, as stated in [10], are: (1) each agent has incomplete information or capabilities to solve a certain problem and, thus, has a limited viewpoint; (2) there is no global control to the system; (3) data are decentralized; and (4) computation is asynchronous.

The potential abilities of MASs which attracted increasing interest in MAS researches are:

- Their ability to solve problems that are too large for a centralized agent because of resource limitations or the risk of having one centralized system that could be a performance bottleneck which could fail at critical times.

- Their ability to allow the interconnection and interoperation of multiple existing legacy systems through incorporation of such systems into a wider cooperating agent community in which they can be exploited by other pieces of software. Incorporating legacy systems into an agent society can be done, for example, by building an agent wrapper around the software to enable it to interoperate with other systems.
- Their ability to provide solutions to problems that can naturally be regarded as a society of autonomous interacting components-agents.
- Their ability to provide solutions that efficiently use information sources that are spatially distributed.
- Their ability to provide solutions in situations where expertise is distributed. Examples of such problems include concurrent engineering, health care, and manufacturing.

In addition, the inherent nature of MASs helps them to enhance performance in terms of the following computational dimensions and features:

- *efficiency*, because concurrency of computation is exploited;
- *reliability*, that is, graceful recovery of component failures, because agents with redundant capabilities or appropriate inter-agent coordination are found dynamically (for example, taking up responsibilities of agents that fail);
- *extensibility* because the number and the capabilities of agents working on a problem can be altered;
- *robustness*, the system's ability to tolerate uncertainty, because suitable information is exchanged among agents;
- *maintainability*, because a system composed of multiple components (agents) is easier to maintain because of its modularity;
- *responsiveness*, because modularity can handle anomalies locally, not propagate them to the whole system;
- *flexibility*, because agents with different abilities can adaptively organize to solve the current problem;

- *reuse*, because functionally specific agents can be reused in different agent teams to solve different problems.

MASs can be applied to many domains; particularly, they are believed to be effectively employed in domains with the following features [3]:

- The knowledge required to solve a problem is spatially distributed in different locations.
- Several entities, while keeping their autonomous behavior, have to join their problem solving abilities to be able to solve a complex problem.
- Problem in a domain may be decomposed into different sub-problems that have some kind of inter-dependencies.

2.1.4. MULTI-AGENT DECISION SUPPORT SYSTEMS (MADSS)

Decision making is a process of making an informed choice among the alternative actions that are possible and actions that determine the direction of a subject or its outcome. Making fairly good decisions is an important issue in all domains. Decision makers, who have this responsibility, are very keen on the help of decision support systems to make informed decisions [11].

Decision Support Systems (DSSs) are software solutions designed to support and improve decision making through model-based set of procedures for processing data and judgments [12, 13]. Such systems could be designed and modeled in agents-based system that provides capabilities for specific problem processing and manipulation [14].

DSSs can be applied in various domains where the quality of decision making has major impact on the outcome and efficiency of the overall process. The features of domains and work processes that require decision support include:

- existence of uncertainty in decision making;
- need for analyzed recommendations;
- involvement of a number of individuals in decision making process in distributed environment;
- complexity of the decision making process that requires the assessment of the impact of the proposed solution;

In healthcare, there are various decision making processes, for example, diagnosis, treatment, prescription and referral decisions, that exhibit the above characteristics. Multi-agent techniques have been used to employ important decision support functions such as clinical diagnostics, treatments and monitoring in clinical management systems. Such clinical DSSs are aimed at helping healthcare professionals in making quality clinical decisions [15].

Multi-Agent Decision Support System (MADSS) is a system that employs both MASs and DSSs in domains characterized by the distributed decision making. As indicated in studies [3, 4, 5, 6, 15], various problems in the healthcare domain are characterized by the features mentioned under MAS & DSS. Consequently, there is a growing interest in adopting multi-agent decision techniques for the various aspects of the domain. For instance, a MADSS is proposed to assist organ transplants coordination of a hospital in the determination of the most suitable receivers for a given organ at a national level [11].

Therefore, medical referral in general and referral indication (that this thesis tries to work on) are believed to be addressed by the development of a multi-agent based decision support model.

2.2. Multi-Agent Referral Decision Support (MARDS)

This section presents background of medical referral in relation to the design of multi-agent decision support framework that motivates this study.

2.2.1. Medical Referral

Medical referral is a critical component of quality clinical care in any healthcare system. It is a determinant factor for the cost-effectiveness, efficiency and quality of patient care. Such valuable component contributes to high standards of care by limiting over-medication, permitting an efficient division of tasks between GPs (General Practitioners) and specialists, giving more free time for specialists to develop their special knowledge, and by decreasing the cost of medical care [6, 16].

In developing courtiers, according to [35], medical referral may have slightly different structure and definition due to the key problems faced in the real environment in which referral works. Accordingly, referral can be defined as any process in which health care providers at lower levels of the health system, who lack the skills, the facilities, or both to manage a given clinical condition, seek the assistance of providers who are better equipped or specially trained to guide them in managing or taking over responsibility for a particular episode of a clinical condition in a patient. Furthermore, higher-level hospitals in developing countries do not treat only referred patients; tertiary hospitals are frequently the first point of contact with health services for many patients.

Medical referral, as indicated in [6], is described in terms of three key characteristics: referral structure, referral process, and referral policy. A referral structure refers to the organization of healthcare providers as primary, secondary and tertiary levels. A referral process refers to flow of patient referrals which entail procedural steps in the management of a clinical referral for a patient through determination and actions taken at each level of the process. A referral policy refers to a set of guidelines and procedures designed to govern the way providers should process patient referrals.

This complex process operates in an environment of distributed and dynamic network of healthcare providers and it is characterized by distribution of knowledge, need for communication and coordination, and complexity of the problem. Thus, there is no single and simple solution for making an appropriate referral indication decision.

In studies [3, 4, 5, 6], it has been argued and justified that multi-agent systems are appropriate and adequate for healthcare applications in general, and to referral decision support in particular. Therefore, this study extends these approaches that allow structuring the referral indication decision support system around several intelligent, autonomous, communicative and cooperating agents that would help to improve the conceptualization, design and implementation of the decision support system. The aim here is to improve the workflow and the decision making process of the referral indication and the services that go with it: local consultation and service identification.

2.2.2. REFERRAL DECISIONS

In the healthcare domain, there are many situations where professionals have to make critical decisions which have major impact on the outcome. A referral decision is a clinical decision-making process by which physicians determine referral indication, type of required services and selection of appropriate providers [7]. Referral indication decision is a situation where a physician determines whether a patient needs a referral or not. It is an initial decision in the referral process and hence it is a vital step in determining the clinical outcomes and costs of patient referrals. If referral is indicated, the physician should determine the required service. Once a physician has made a decision on all required services, the next step is to select an appropriate provider.

Figure 2.1 summarizes the overall referral decision-making process [6].

Figure 2.1 Referral Decision-Making Process

As a referral indication and service identification aspects of referral decision are the focus of this study, these will be further discussed in Chapter 4.

2.2.3. MARDS FRAMEWORK

The study in [6] proposed a Multi-Agent Referral Decision Support (MARDS) framework that extends the capability of existing e-Referral Systems with an intelligent referral decision support. The framework is constituted from three layers as shown in Figure 2.2 [6]: **information system** layer (existing e-Referral systems designed to improve quality of referral communication); a **multi-agent system** layer (a referral decision support service); and a **communication** layer (existing distributed messaging middleware that can be employed for inter-agent communication). The multi-agent architecture is designed to support the overall referral decision-making process (referral indication decision, required service identification, and appropriate provider selection).

MARDS framework's design is based on hierarchical Ethiopian referral system and usability of existing e-referral system, network infrastructure and communication protocols. The design of the framework employed a pattern-oriented approach that integrates a goal-driven methodology and the application of agent patterns. The architectural design employed some aspects of GAIA design process besides a goal-oriented approach (to capture system requirements) and the identification of agents.

The purpose of the multi-agent system layer is to provide a referral decision support to the information system layer through collaborating agents with help of communication layer.

Figure 2.2 MARDS System Architecture

In the proposed MARDS system architecture, five types of agents are identified as:

- **User Interface Agent (UIA)**: interacts with the user via the information system layer, receiving user requests and delivering results of the MARDS system.
- **Referral Coordinator Agent (RCA)**: coordinates the process of referral decision support between referring and referred-to providers
- **Referral Specialist Agent (RSA)**: is an agent cluster (of Referral Analyst Agent (RAA) and Medical Consultant Agent (MCA)) designed to provide a decision support service for a referring provider regarding the need of referral and required

services. The RAA is designed to help providers in determining whether a referral is needed or not. On the other hand, the MCA is intended to provide local medical consultation for Primary Care Physicians (PCPs) when referral is not indicated. To provide this decision support, the RSA employs a Referral Guideline Knowledgebase (RGK) that serves as a checklist for the PCP to ensure that the most appropriate treatment is performed prior to a referral.

- **Provider Selection Agent (PSA)**: is an agent cluster designed to support a referring provider in selecting an appropriate provider by analyzing patient characteristics, referral policy and provider information. It is composed of two key agents: a Provider Filtering Agent (PFA) for filtering potential providers by consulting the Referral Policy Knowledgebase (RPK) and a Multi-Attribute Decision Making Agent (MADMA) for selecting and ranking final list of providers by employing a multi-attribute decision aid technique.
- **Medical Broker Agent (MBA)**: class of agents which maintains knowledge about providers of the system at different levels that helps to find providers information for the selection decision by the PSA.

This framework is believed to lay down a basis for the application of agent technology to referral decision support in healthcare. However, a decision support technique has not been fully addressed for the referral indication and service identification decision aspects of medical referral. For example, there are only high level agents identified for referral indication that may not be adequate to analyze all processes, information, and factors from the underlying information system. Therefore, the objective of this research is to study and propose a decision support technique for referral indication and service identification.

The next chapter discusses some related works that were done with regards to the application of multi-agent technology to the problems of medical referral.

CHAPTER 3 : RELATED WORKS

Healthcare deals with the prevention, treatment, and management of illness and the preservation of mental and physical wellbeing through the services offered by the medical, nursing, and other allied health professions [17]. It is a diverse and complex domain which aims at the delivery of cost-effective and quality health-care services to citizens. In addition, healthcare is an extensive, open and dynamic domain characterized by distributed decision making and management of care that requires communication and coordination of complex and diverse forms of information distributed among providers [5].

Many researches have been conducted to provide software and IT solutions to the diverse problems of healthcare. There have been various studies that are conducted to apply agent technology on healthcare systems, clinical decision support systems and referral systems. Recently, a study has been conducted to develop a multi-agent referral decision support framework that can assist physicians in making better referral decisions [6]. This framework that contributed to the motivation of this study and other related works are discussed in the following two sections.

3.1. Agent-based Systems for Healthcare

Moreno and Nealon in their various research works [3, 4, 5] showed an increasing significance of multi-agent systems (MAS) in the healthcare domain by enhancing the ability to model, design and build complex, distributed healthcare software systems. They have identified a set of characteristics of MAS that make them appropriate to be used to tackle various healthcare problems. These characteristics include: autonomy, sociability, pro-activity, and distribution of agents in MAS. In addition, features of an application area that makes an agent-oriented approach appropriate are indicated. These features of healthcare include: the distributed nature of the knowledge; complexity of the software solution; lack of centralized solution; the need to maintain autonomy among healthcare entities; the need to coordinate in order to provide specific services to individuals; and the need to receive information and advice proactively. Moreover, based on this characterization, some potential application areas are identified and exemplary agent-based applications like management of organ transplants, access to medical information, and care of palliative patients are discussed.

The research work in [18] proposed a distributed MAS system that combines medical guidelines with agent technology. The goal of the system is to improve utilization and management of the various resources in a real medical environment. The proposed agent-based architecture allows representing different entities in a medical centre and the relationships between them. This system enables physicians to follow a given set of guidelines while performing medical routines. It also provides decision support in some key clinical decision makings on the diagnosis or treatment. In this system, several agents, which have their own knowledge and some roles to accomplish, interact with each other, humans and resources (like physical equipment) in the MAS architecture in order to fulfill their design goals.

The work in [19] proposes a multi-agent system architecture that provides a decision support for heart failure management in a generic home care system. The main goal of the system is the characterization of the patients' health status and accordingly the notification of the corresponding medical personnel to take medical actions. The system integrates various processing modules based on statistics, rules and models that are applied to the medical data of patients. These modules can also be used as learning capabilities based on the medical interventions that take place, and negotiation schemes among agents.

The above mentioned studies discuss multi-agent system application on healthcare domain in particular to referral decisions. The works focus only on the referral communication. It does not address the problem of referral decision-making process, which is an important component of a referral process that can affect the overall outcome of a referral system.

3.2. Multi-Agent Referral Decision Support Framework

The study in [6] presented a study conducted on the possibility of adopting multi-agent technology to the problems of referral decision making. It proposed a comprehensive Multi-Agent Referral Decision Support framework, which incorporates five types of cooperating agents and two knowledge-bases. Though the framework has been designed to support all the three aspects of referral decisions, it is focused on the provider selection aspect. For provider selection aspect, a three phased selection model has been proposed. It is believed that this work has laid down a basis for a number of research works and development works towards the realization of an intelligent referral decision support.

However, the basic limitation of this work is that it doesn’t fully address the referral indication aspect that is a key component of the medical referral process, which may cause under-referral and over-referral problems. This in turn limits the realization of an intelligent referral decision support that could improve the quality of referral decisions.

The next chapter discusses the concepts and features of referral indication decision making process and factors that affect the decision.

CHAPTER 4 : REFERRAL INDICATION DECISION

This chapter discusses the analysis of the referral indication decision making process and factors that affect the decision in relation to the overall referral decision making process.

4.1. Referral Indication

A referral indication process is a set of steps taken in the course of determination about whether a referral is needed or not. It is a key aspect of medical referral that affects the quality of clinical outcomes and costs of patient referral. In this section, the process and alternatives services after referral indication decision will be discussed.

4.1.1. Referral Indication Process

A referral indication decision making could be initiated as a result of various referral consideration factors. Usually, primary care providers consider referral to the next level for one or more of the following reasons: clinical uncertainty, emergency cases, and shortage of medical facilities and or absence of specialist service. However, the referral decision is mainly influenced by three key factors (discussed in 4.2): patient, physician and healthcare system determinants.

In healthcare provision, clinical staff are frequently engaged in decision making tasks like diagnosis, therapy and test selection, and drug prescribing under uncertainty conditions [20]. A referral indication process can start at any stage of regular clinical process while medical staff are dealing with patient cases. The most common clinical decision making processes that lead to referral considerations include **examination** (monitoring & assessments to gather facts and data about patient situation), **diagnosis** (identification of a disease from its symptoms), and **treatment** (decision about the service and medication to be provided as a result of diagnosis, e.g. prescription) [15].

Figure 4.1 shows that a referral indication decision can be made at any of the three clinical processes (examination, diagnosis and treatment). These processes are interdependent and information can flow in both directions from and to each other. It is also possible to conduct examination, diagnosis and treatment during or after referral indication decision.

Figure 4.1 Flows and process in referral indication decision

Examination: refers to a process of thorough monitoring, investigation, assessment and testing of a patient with the help of facts and data gathered through different mechanisms. Patient examination involves various mechanisms including interviewing the patient, reviewing patient history, laboratory tests and radiology results, and etc. It is possible to consider referral indication at this stage looking at patient's factors and situation. After evaluating the collected data, it is possible to decide the next required intervention as either referral/consultation or diagnosis and treatment [21].

Diagnosis: refers to a decision making of a physician in identification of a disease. Such decisions are mainly made based on the symptoms observed compared to the known characteristics of a certain disease. A disease could be identified through consultation, review of similar cases, and analysis of diagnostic factors. The result of a diagnosis is determination of the disease as one or combination of these. For example, diagnosis as cancer, pneumonia, typhoid, diabetes, psychiatric, depression, chronic fatigue, malaria, etc. The identification of the disease is a key input and indicator for possible treatment the patient needs to be provided. At this stage, if there is any uncertainty on the identification of a disease, then it is more likely that referral would be considered.

Treatment: is a decision making process about the services, interventions and medication (prescription/ chemotherapy) to be provided to a patient based on the result of the diagnosis. Prescription is a decision about the type, dosage and alternative medications to be given for a patient. In addition to prescription, the physician may also make a decision on the need for services like admission, need for intensive care, further therapy, and/or surgery. Definitely such services require the availability of appropriate medical as well as other facilities. For example, admission to a specific treatment requires availability of beds and the required medical equipment, and surgery demands the availability of a surgeon. Due to these dependencies, a physician may frequently consider the need for referral as part of treatment. For instance, if certain medical equipment is not available at disposal of the physician, there is a good possibility for a referral indication to be considered.

The complexity of the referral indication process often causes inefficient decisions, which in turn affect the quality and cost of healthcare. As a result, the referral indication decision could be an appropriate or inappropriate referral. An appropriate referral is an efficient decision made based on the right information and analysis that leads to improved clinical outcomes and reasonable costs. On the other hand, an inappropriate referral decision is one that is impacted by a physician's clinical uncertainty and limited scope of practice (lack of adequate experience, skill, information and analysis of patient case). This in turn leads to under-referral or over-referral problems. Under-referral could be fatal which could lead to death of the patient due to lack of treatment by a specialist. This affects the quality of healthcare. On the other hand, over-referral could lead to inefficient use of scarce resources and unnecessary transportation cost. For instance, the time of the specialist may be consumed by over-referred cases while the person, who needs his/her service most, is left to wait or suffering. Moreover, the patient that could have received adequate treatment in a lower level medical center may spend unnecessary cost of transportation and accommodation to see a specialist in a referred center.

Moreover, studies in [22] indicate that high over-referrals can cause dissatisfaction to the patients as well as to those personnel to whom the referrals are made. Consequently, it reduces the effectiveness of the overall referral process. On the other hand, high under-referral is also problematic and often results in delayed diagnosis and related problems.

In referral indication process, physicians need consultation and decision support to make appropriate referral decisions and to overcome clinical uncertainty. If referral is indicated, then there is a need to identify the appropriate medical services before transferring the referral to the service provider. Otherwise, if physicians found it beyond their capacity to identify the magnitude of the problem, they should be provided with necessary medical consultation in order to deal with the patient.

In the Ethiopian case, according to a specialist in St. Paul Hospital, more over-referral has been observed mainly due to responsibility shift and lack of required assessment by referring provider. As a result, referral hospitals are overwhelmed by over-referrals and hence this affects the quality of the service. For instance, a triage nurse in St. Paul Hospital indicated that the hospital is overloaded by inappropriate referrals which they couldn't reject the referrals as the patients do not have an option. This needs some solution to deal the issue at the source, while making referral indication decision.

Figure 4.2 shows the possible flows of a referral process after the referral indication decision have been made.

Figure 4.2 Required Services after Referral Indication Process

These two services' flows are briefly discussed in the following two sub-sections.

4.1.2 Required Service Identification

Once the physician has decided to refer a patient, the next step is to determine the type of required services that the patient gets that depends mainly on the current clinical condition of the patient. For instance, if a patient is in life threatening situation, the physician may refer the patient for emergency surgery. Therefore, in the situation like this if the right service is not identified properly before transferring the patient to the referred-to provider, the result could be adverse.

A service identification decision can be made either by the physician or administrative staff. There are various factors that affect the decision of identification by the referring physician. These factors are type of required service, availability of the service with the referred-to provider, policy condition of the service, and affordability of the service.

Type of required service: A service identification decision is basically dependent on what service was required during the referral indication decision. The type of required service is determined by looking up all available services against the case of the patient. To identify required services the diagnosis result, frequency and duration of the required service are very important.

Availability of the service: Secondly, the referring physician should have information about what services are available with other service providers. In most cases, it is possible to access the list of services available at each provider. However, it is very difficult to access this list during the service identification stage. Health bureaus or providers' coordination offices may have the list of all available services and specialties under their authority. For example, the Addis Ababa health department distributes service and specialty list of all hospitals in the city to its providers who may be referring cases to other providers. This list can also be used as referral guideline for service identification by providers at a specific domain level.

Conditions of the service: It is important to take in to account the referral policy while making service identification decision. Even if providers know the availability of required services with a provider, the referral policy or guidelines should be referred before deciding a required service. According to the domain experts (of St. Paul Generalized Specialized Hospital), the referral guideline mentioned above has defined conditions like when to accept or reject what type of service and from which referring provider.

Affordability of the service: refers to the cost of the required service the patient is going to be referred to. It is generally assumed that the referring physician has an idea about the cost of the services before determining the type of the service. This can affect the decision of the physician as it can equally affect the practicality of the referral. If a patient can't afford for a service she/he is referred to, it is as good as not being referred.

In the local context, Abroad Referral is an important factor to consider when determining a required service. It is a referral indication decision made to refer a critical patient's case to abroad provider for the service that is unavailable locally. In this process, an initial referral recommendation is made by a specialist in referral hospital and the final required service and referral is approved by a National Medical Board. Hence, there is a need to ascertain whether the required service is available locally or not before making expensive abroad referrals.

4.1.3 LOCAL MEDICAL FOLLOW-UP

Local medical follow-up refers to any further clinical intervention and follow-up to a patient, if referral is not indicated after analyzing the patient, physician and system determinant factors. These could be consulting other clinicians, sending the patient to further diagnostic tests (laboratory, radiology, etc) or some other examination. In this case it is evident that the clinician is faced with some uncertainty about the case of the patient. This leaves clinicians (may be doctors, nurses, therapists or other medical professionals) seek advice from another clinician or specialist on diagnosis, evaluation, and or plan of treatment. Thus, the clinician must seek medical advice and opinion from someone assumed to have better knowledge about the case under consideration.

In the Ethiopian context, local consultation is a very important aspect according to feedbacks from the domain experts. Such local medical consultation services to medical staff are essential to overcome several problems in the health sector. Inter-consultation forms/letters are used to seek advice and second opinion from a specialist. One of the main challenges that hinders such local consultation is lack of effective communication means between medical service providers and professionals. The other obvious one is shortage of medical professionals (particularly shortage of specialists) in the country which is further complicated by the organizational problems in which healthcare providers operate. These three challenges are discussed below.

Communication problem between providers/specialists: Though there have been efforts made in enhancing the means of communication and technology among healthcare providers/specialists, it is still evident that such communication has never been robust and efficient and has not met the requirements of the complex healthcare information exchange. In effect, e-consultation and tele-consultation haven't gone that far. For instance, a specialist in St. Paul Hospital indicated that the paper based inter-consultation forms take long time to communicate, say for advice on urgent cases, and hence less efficient. Hence there is a clear need to research a way out for such problems.

Shortage of experienced medical staff/specialists: There is significant and noticeable shortage of experienced medical staff in general and specialists in particular in the country. This generally causes the medical referral and consultation process to be inefficient and affect the overall output of the healthcare sector. Due to this, a physician who needs to consult or seek second opinion from a specialist may not get such consultation at the right time and place. Hence, it is important to layout a method that enables a practitioner to access knowledge and learns from experiences of specialists locally.

Organizational challenges: In many ways the major bottleneck that hinders efficient medical service provision in the healthcare sector is how the healthcare providers are organized, cooperate and collaborate with each other. A patient's case that is referred to a certain service provider could be poorly dealt with due to lack of efficient administrative procedures and commitments binding the referring and referred-to providers. Long bureaucratic procedures usually contribute to such problems. Moreover, there are also some organizational challenges in the efficient utilization of e-consultation and telemedicine. If various parties that are involved in e-consultation and telemedicine services can't cooperate well, due to their organizational setup, the e-consultation/telemedicine in place could easily become underutilized. For example, if a cardiac surgeon is available in the country and if he/she couldn't participate in the e-consultation due to some organizational problem (say lack of cooperation with other providers) the output of the e-consultation will definitely reduce. Hence, it is important to foster a strong local consultation that is independent of organizational hurdles.

It is believed that if the above mentioned gaps could be filled, then rate of over-referrals are minimized. In effect, the overall efficiency of the medical sector could increase, which in turn improves healthcare outcomes.

In the following section, factors that influence a referral indication decision are presented.

4.2 Decision Factors

Referral indication decision is mainly influenced and affected by three factors: patient, physician and healthcare system. Figure 4.3 shows characteristics, conditions and behaviors related to these factors in the referral indication decision process. The description of each determinant factor is discussed below.

Figure 4.3 Factors and characteristics that influence referral indication decision

4.2.1 PATIENT DETERMINANTS

Patient determining factors are the patient's status and condition at the time of referral. These factors are identified as clinical characteristics through various clinical processes like physician consultation, medical diagnosis, patient history review and other tests. Physicians use this information to see referral needs and base their decision on these determinants.

In [23], it is shown that there are various types of determinants that contribute to the complexity of healthcare. One key determinant factor is patient characteristics such as compliance, predisposition to disease, age, propensity to use resources, high-risk behaviors (e.g., smoking), lifestyle, emergency status, functional status, and disease severity measures.

These patient determinant factors could be categorized as clinical characteristics, demographic characteristics, behavioral characteristics and socioeconomic status.

Clinical characteristics

Clinical characteristics refer to factors related to the result of clinical diagnosis and treatment of a patient. These characteristics can be identified through clinical examination, diagnosis and tests. Clinical characteristics of a patient include biological and physiological variables such as symptom status and functional status. A referral indication decision mainly depends on these characteristics. For example, if the severity of a certain disease is beyond the capacity of the care provider, the physician may decide to refer the patient to other provider or specialist.

The key clinical characteristics which affect the referral indication decision are:

- ***Diagnostic conditions*** – refers to the status of a patient based on diagnostic results. These can be determined from signs, symptoms, laboratory tests, radiological results and physical findings. The diagnostic results and conditions are the main input and a basis to make any type of clinical decision such as prescription, admission, referral, etc. Hence, these are important factors for referral indication.

- ***Disease severity*** – refers to the degree of injury, damage, or other mission-impairing factors (loss of body part, strength and so on) that could occur as a result of the disease. Care providers may have standard definitions for various disease severity levels. Commonly, severity can be defined as fatal/killer/life threatening/high, emergency, serious, normal/medium, or slight/low. In most cases, the higher the severity is the greater the referral needed.

- ***Comorbidity conditions*** – is a medical condition that exists in addition to and is caused or worsened by obesity or any other primary disease being studied or treated. It is a complication caused by two or more diseases indentified in a patient at the same time. This greatly increases the rate and need of referral as it causes more medical uncertainty and dilemma even for the specialist of one of the diseases.

Demographic characteristics

This refers to factual aspects of a patient that include race, age, sex, income, mobility (in terms of travel time to work or number of vehicles available), educational attainment, home ownership, employment status, and even location [21]. Particularly, age and sex of a patient under referral consideration can affect the referral indication decision more importantly. For example, there are cases where the same type of symptoms and diseases could be treated differently for adults and infants. In the local context, location is important as referral is coordinated hierarchically based on the regions and zones.

Behavioral characteristics

These are health habits, beliefs, attitudes, general health perceptions and preferences of the patient that affect the care process and the outcome of clinical characteristics.

Discussion in [24] indicates that most care providers depend on symptoms in the process of deciding to consult or to refer cases to specialists. Severe symptoms leave the afflicted person little alternatives but to only be able to recognize that he or she is ill, and in this case some kind of formal help is required. In this regard, the person's estimate of the present and future probabilities of danger will be taken into account. Naturally, the more symptoms are perceived as representing a serious illness, the more probable it is that a person will seek professional advice and agree to be referred to when the need arises. Oftentimes, it is the social disruption that the symptoms cause, and not the symptom per se, which necessitate medical help.

Socio-economic status

Socio-economic status refers to non-clinical factors like ability to cover costs of medical and nonmedical services. For instance, if a physician suggests a referral for a patient who can’t afford the service or transport cost of the targeted provider, it may be better to reverse the referral indication decision to find alternative treatments at a better reach.

Though all of the patient characteristics that are mentioned above influence the referral indication decision, clinical characteristics are the most crucial ones. Physicians also consider these determinants primarily as the basis of their decisions. However, there are also factors relevant to physicians themselves that affect referral indication decision.

4.2.2 Physician Determinants

Physician determinant factor refers to the scope of practice of the physician. It is mainly related with the tolerance level of clinical uncertainty as well as perception of disease severity and its potential impact on future health of a patient. Studies in [25] identified potential factors influencing physician's clinical decisions as previous clinical experience of doctors, their perception of absolute versus relative risks, physician's demographic factors and their habits. Among these, physicians' perception of severity and potential impact of the disease on the patient, and the scope of practice of the physician are very important.

Physician's perception of severity and impact: this refers to how a physician senses the level of the severity and impact of the patient's health problem during referral consideration. The physician's view of the disease's potential impact on the future health of the patient also matters. For example, if the patient faints during medical examination, different physicians may perceive and react to the situation differently due to their varying perception of the severity level. Hence, a physician that has a tendency of putting the severity level of a patient's case to a higher level tends to refer the patient to the specialist service. Also, if a physician thinks the impact of the disease under examination would be adverse on the patient, there would be a tendency to refer the case to a specialist.

Physician's scope of practice: this refers to the level of previous experience and training of a physician. In most cases, limited experience and training leads to medical uncertainty. The tolerance level to clinical uncertainty of physicians influences their referral indication decision. The lesser the tolerance level the higher the degree of referral indication.

According to a response from domain experts, in the Ethiopian context, physicians' lack of knowledge caused by limited experience and training is the main factor for over-referrals.

4.2.3 Healthcare System Determinants

Healthcare systems determinants are factors that affect the referral indication decision, which are mainly related to the availability of specialists, cost of specialized tests and referral policy and procedures. The structural characteristics of a healthcare system, its organizational aspect and financial issues also affect the referral indication decision. In the Ethiopian context, as most of the referral hospitals are government owned and their cost is lesser, the rate of referrals to these hospitals is high.

Availability of specialists: this refers to the number of specialists available at the primary care centers. As referral needs are guided by the availability of the specialist service at the referring care provider, the supply of specialist affects the rate and number of referrals needs. Lesser availability of specialists at primary care providers indicates increased number of referrals.

Cost of services: there are different costs that different healthcare providers require depending on various factors that are pertinent to the provider. Sometimes, referral indication decisions could be guided by such cost factors like service affordability, cost efficient alternatives and policies related to cost coverage. Accordingly, the high cost of specialized services of referred-to providers limits the number of referrals as some can't afford it.

Policy and procedures: a referral policy could define different criteria for referral decisions in general and for referral indication in particular. Hence, referral would be indicated or not in compliance to such polices and related procedures.

In the Ethiopian case, shortage of any of the key resources or medical facilities like specialist, certain equipment or bed could be the main driving factors for rate of referrals. For example, if there is a patient who needs urgent Appendix surgery, and if there is a specialist and equipment but no bed, then the case is referred to another hospital. This could adversely affect the overall outcome of the referral process.

In summary, Figure 4.4 shows the overall referral indication processes and their relation to the decision factors. This figure is the integration of processes and determinants that are depicted in figures 4.1, 4.2 and 4.3. It is indicated that referral indication decision making is a complex process that involves many other clinical decisions and processes like examination, diagnosis and treatment. The decision is also affected by factors of complex mix of patient, physician and health system determinants. The service identification and medical follow-up are also presented as alternative flows after referral indication process.

The next chapter discusses the design of the multi-agent decision support model for medical referral indication.

Figure 4.4 Clinical processes, factors and characteristics in referral indication decision

CHAPTER 5 : A MULTI-AGENT DECISION SUPPORT (MADS) MODEL FOR REFERRAL INDICATION

5.1. Overview

This chapter discusses the proposed Multi-Agent Decision Support (MADS) model for Referral Indication that extends the MARDS framework to support referral indication and service identification aspects. This model is described in terms of the design goals and its detailed architecture.

The architecture is composed of four major components: Clinical Environment, Referral Indication (RI), Service Identification (SI) and Local Consultation (LC). The Clinical Environment identifies and presents key resources such as Clinical Information System (CIS) and its clinical processes (Examination, Diagnosis, and Treatment). The RI, SI and LC contain intelligent agents that support referral analysis and decisions with the help of information from a Clinical Knowledgebase (CKB).

The reminder of this chapter is organized as follows. Section two presents design approach followed to design the model; section three discusses system design goals; section four presents an architectural design of the Multi-Agent Decision Support model for Referral Indication.

5.2. Design Approach

5.2.1. DESIGN ASSUMPTIONS

The following is a list of key design assumptions considered while designing the multi-agent referral analysis model.

- **MARDS framework:** this design is based on the multi-agent decision support framework for medical referral, which is proposed in [6]. Therefore, the assumptions taken in this framework also apply for our design. Specifically, the information layer of the framework is assumed to represent an existing e-Referral system that handles e-Transfer[1] of patient referrals among the distributed healthcare providers. Moreover, the CIS identified in this model is assumed to be part of the information system layer. Besides, the system is supposed to use the existing messaging mechanisms for its inter-agent communication.

[1] This refers to transfer of patients referral details from one healthcare provider to another electronically

- **System Design:** the design of the system focuses on the multi-agent system layer. It assumes that each healthcare provider has an e-Referral system connected to a Hospital Information System (HIS). Besides, the multi-agent system design is limited to an architectural design level. It does not deal with the detail design. However, a detail design and implementation of the core part of the multi-agent system will be dealt with during prototype development.
- **MAS Design:** the multi-agent system is the main aspect that this work focuses on. It assumes that the environment in which the MAS model works has a Clinical Information System (CIS). In this design, CIS and HIS refer to the samething with CIS focusing on the clinical processes in relation to medical referrals.
- **Architectural level design:** the multi-agent system design is limited to an architectural design level. It does not deal with the detail design.

5.2.2. METHODOLOGY OF FORMULATING THE MODEL

Several agent and multi-agent systems development methodologies have been reviewed with the aim of selecting the most appropriate methodology for this research that will be employed for the analysis and design of the envisioned MAS decision support model. These include: Gaia[2] [26], MaSE [27], PASSI[3] [28] and SODA[4] [29].

Finally Gaia has been chosen as analysis and design methodology for the referral MAS for the following reasons:

- **Design metaphor** – the ‘organizational’ design metaphor is pertinent to the healthcare domain. It is found to be convenient to our case since it reduces the conceptual distance between the target domain and the envisaged MAS system model.
- **Abstraction level** – the design models provided by the methodology are found to be at an acceptable level of abstraction.
- **User friendliness** – the Gaia methodology is found to be easier to learn and use.

However, Gaia methodology has its own limitations. It lacks robust requirement capturing capability; it does not deal with the activities of requirement capturing and modeling, which

[2] A Multi-Agent Systems Development Methodology
[3] A Process for Specifying and Implementing Multi-Agent Systems Using UML
[4] Societies and Infrastructures in the Analysis and Design of Agent-based Systems

is an important input for the multi-agent system analysis phase. Hence, Goal Oriented Requirement Engineering method [30] has been employed along with a pattern-oriented approach [31] for requirements specification purpose since their abstractions closely match those of agent-oriented computing. Goal-oriented requirements engineering is concerned with the use of goals for eliciting, elaborating, structuring, specifying, analyzing, negotiating, documenting, and modifying requirements.

The pattern-oriented approach is employed for its ability to complement a goal-oriented methodology with the application of agent patterns. This is believed to facilitate a goal-oriented design process by employing both top-down (by using goal-driven methodology) and bottom-up (by using agent patterns at different design stages) directions simultaneously.

The Gaia design process is constituted from three key phases: **analysis, architectural design** and **detailed design** [30]. The analysis phase lay a basis for a design by collecting and organizing the specification which at the end generates an environmental model, a preliminary roles model, a preliminary interactions model and a set of organizational rules. The second phase is the architectural design phase that identifies and specifies the overall architecture of the system. At the end of this phase, a MAS architecture together with its completed roles and interactions models is produced. The last phase is the detailed design phase, which targets in defining the agent and service models by making use of the architectural design models.

In this study, a customized design approach described in [6] has been followed. Figure 5.1 shows this approach used to design the architecture of the system.

Figure 5.1 Phases of the system design process

In the first phase of this approach, high-level system requirements that are specifically important for the architectural design of the system are captured and specified. In the second phase, the multi-agent system design phase, the system goals are analyzed with the aim of organizing the system specification for our MAS. As mentioned earlier, the analysis phase of Gaia methodology is employed here. The main analysis steps performed are: a) subdivision of the system into sub organizations; b) identification of environmental entities and resources; c) identification of preliminary roles and interactions; and d) identification of preliminary agents from the roles and interactions.

In the final phase of the Multi-Agent System Architectural Design, the overall system architecture is designed using the analysis models from the previous phase. In the process of designing the architecture, the following major design steps are carried out.

- Identification of organizational structure and organizational patterns
- Completion of preliminary roles and interactions
- Completion of preliminary agents

It is important to note that the existing design patterns (organizational patterns, social patterns and other agent patterns) are used in identifying agents, interactions and organizational structure.

5.3. System Goals

As defined in [27], a goal is a system-level objective that the system under consideration should achieve. Once the goals have been captured and explicitly stated, they are less likely to change than the detailed steps and activities involved in accomplishing them. There are two parts in capturing system goals: identification of goals from initial system specification and structuring of goals using goal hierarchy diagram.

A referral indication is a critical aspect of medical referral that is an important component of quality clinical care in any healthcare system. The cost-effectiveness, efficiency and quality of patient care are affected by the medical referral effectiveness and its various workflows and processes involved. Such valuable components contribute to high standards of care by limiting over-referrals and under referrals, and permitting only appropriate referrals. Hence, the overall goal of the medical referral Decision Support System is improving the referral process through minimization of over-referrals and under-referrals by supporting appropriate decisions.

As suggested in [6], to achieve the general goal, the system's goals are categorized as hard goals (whose satisfaction can be established through verification techniques) and soft goals (whose satisfaction cannot be established in a clear-cut sense) in the following sub-sections, and they are identified from the initial system specification.

5.3.1. HARD GOALS

As the model we are going to design is based on the MARDS framework in [6], the hard goals of the system are found to be the same and they are reused in this work. These goals are specified by a goal hierarchy diagram shown in Figure 5.2 [6].

Figure 5.2 Goal Hierarchy Diagram of the Medical Referral Decision Support System

However, the focus of this study is on the referral decision support (1.1 of Fig. 5.2), specifically on the referral analysis (referral indication and required service identification - 1.1.1 and 1.1.2 of Fig. 5.2). As part of the goals refinement, we have found that referral indication and service identification aspects are interrelated and the common goals is to analyze and support the referral decision through determination of the need for referral and the required service. Therefore, Goals 1.1.1 and 1.1.2 are further refined and merged as single broader Goal 1.1.12 and sub goals have been identified further as shown in Figure 5.3.

In order to support the referring provider in indicating the need for referral, the system should collect and analyze clinical data; and it should also need to analyze decision determinants (patient, physician, and healthcare system). Secondly, if referral is indicated, then to provide a service identification decision support, the services' information should be collected and decision factors should be analyzed. Thirdly, if referral is not indicated, then local medical consultation should be provided through clinical specialist consultation and patient's data collection and analysis.

Figure 5.3 Goal Hierarchy Diagram of Sub Goals

5.3.1. Soft Goals

As indicated in [30, 32], soft goals are those whose satisfaction cannot be established in a clear-cut sense and these are useful in refining the hard goals. These can also be referred as non-functional goals as stated in [30]. These goals also cover nonfunctional issues associated with quality of service – such as safety, security, accuracy, performance, and so forth.

Hence as part of the refinement and in order to achieve the general goals, the following soft goals are identified.

- **Under-referral minimization** – the system should provide a support to ensure good quality decision aimed at minimizing under-referrals in order to protect inappropriate, cost-ineffective, or dangerous treatment.

- **Over-referral minimization** – the system should provide a support to ensure good quality decision aimed at minimizing inappropriate or unnecessary referrals in order to free consultant resources for treating cases that are more complex.
- **Required service identification** – the system should identify the right services for referral efficiently.
- **Local consultation support** – the system should provide at most local specialist consultation in order to minimize communication and transportation overheads of distant consultation and referrals.

5.4. A Multi-Agent Referral Indication Architecture

5.4.1. Overview of the Architecture

A multi-agent system architecture is designed from the analysis that has been done in the previous section. Figure 5.4 shows the proposed architectural design for the referral decision analysis system that encompasses: referral indication, service identification and local consultation.

The architecture depicts the environment in which the MAS works and its entities, associated resources, the identified agents and their interaction. It has four major components:

Clinical Environment: a component that encompasses the Clinical Information System (CIS), the clinical processes, and other entities and resources in it.

Referral Indication (RI): it contains three agents responsible for referral indication decision support.

Service Identification (SI): contains an agent responsible for identification of the required service and interacts with Clinical Knowledgebase (CKB).

Local Consultation (LC): contains two agents responsible for local medical consultation and interacts with Clinical Knowledgebase (CKB).

Clinical Knowledgebase (CKB): this is a clinical knowledge repository that contains essential information to be retrieved during RI, SI and LC analysis and decision support processes.

This architecture is the extension of MARDS framework shown in Figure 2.2, where the CIS maps to the information system layer and the MAS layer is the extension of part of the same layer of the framework. Hence, it is assumed that it interacts with its users through the information system layer of the framework. Such interaction is achieved through User Interface Agent that accepts users request from the information system layer and passes on to the Referral Coordinator Agent. The RCA in turn interacts with all other agents and components of the MAS.

The agents in RI cooperate with CIS in the clinical environment to determine if referral is needed or not. If referral is indicated, then the SI agent determines the required service with the help of the service directory in the clinical knowledge base. Otherwise, the agents in LC cooperate to provide the user with local medical consultation with the help of local clinical knowledgebase.

The two agents responsible for user's interaction and overall coordination, and the CKB are described as follows:

- **User Interface Agent (UIA):** This is an agent that interacts with the user of the MAS by receiving user requests and delivering results of the system. The UIA passes the user's requests to and gets returns from the referral coordinator agent.
- **Referral Coordinator Agent (RCA):** This agent coordinates the referral indication and service identification decision support process. It acts as a coordinator between the three services systems: referral indication, service identification, and local consultation.
- **Clinical Knowledgebase (CKB):** it is a source and repository of various clinical guidelines, service directories, polices and standard procedures, and best practices. It is a clinical resource which captures and stores information and knowledge base that assist users in searching, retrieving, and applying relevant knowledge and best practices. It can be customized to be integrated into the underlying medical information system. It serves as a source of most clinical information for agents that participate in RI and LC services. This knowledgebase also provides list of services for the agents of the RI and SI in the form of service directory where all available services are listed with additional useful information.

5.4.2. THE CLINICAL ENVIRONMENT

The clinical environment is a basis for all clinical and referral decisions through its processes, information system and knowledge resources (medical professionals). The three key clinical processes interact with each other and with the CIS to store and retrieve key information about patients, physicians, diseases, and the workflows and results of the processes (examination, diagnosis and treatment).

Clinical Information System (CIS): This is an information system that integrates the flow of clinical processes (examination, diagnosis and treatment) of a patient, clinicians (nurse, doctor or technician) and knowledge resources (guidelines, policies and procedures). The agents in RI, SI and LC often interact with CIS to seek some key information about patients, clinicians and the healthcare system.

The clinical processes**:** ***Examination, Diagnosis and Treatment*** are standard clinical service processes and workflows that are followed by a clinician and a patient. These processes interface with many clinical decision making processes and with each other. They follow some clinical decision making process at their different stages of flow. For example, in the diagnosis process, the clinician has to make a decision about what is determined from the investigation, tests and symptoms. These processes depend on the underlying information system (CIS) to capture patient's records, to fetch required history and to acquire physicians' data. The existing clinical/medical information systems for examination, diagnosis or treatment can be customized to address the requirements for CIS to fit in this referral model.

5.4.3. The Referral Indication (RI)

The RI is the first stage in the referral decision process where physicians would make decisions as to whether referral is needed or not. As this is a decision in a clinical environment, any of the medical staff and even administrative staff could be involved in the process. Patients' information that could be obtained through clinical examination, personal record review and clinical diagnostic results are key input for RI decision. The inputs from the existing medical information systems are also crucial for RI process.

The physician or the clinician initiates the RI decision support request while undertaking the clinical process like examination, diagnosis or treatment. The clinician may also need to interact with CIS to retrieve various information systems as required. During the RI process, determinant factors of patients, physicians and healthcare systems are iteratively compared and analyzed based on various aspects.

The agents in the referral indication component are:

- **Patient Analyst Agent (PAA)**: This is an agent responsible for analyzing the patient determinants during referral indication consideration. It fetches patient information (patient's clinical, demographic, behavioral and socio-economic characteristics) from patient's record which is captured and stored in the Clinical Information System (CIS). Upon receiving a referral analysis request for a specific patient case from the RCA, along with patient determinants, this agent **searches** the CKB for the matching case to the requested case and **compares** the referral indication decision entries against each determinants captured in the request. Then it returns whether referral is needed or not based on the patient determinants.

- **Physician Analyst Agent (PHA)**: This agent analyses the physician determinants for referral indication from physician's record who is currently dealing with the patient case. It fetches the physician's perception of severity and impact as well as the scope of practice from the CIS. This information will be passed to this agent along with the patient case that needs referral indication decision support. PHA processes the referral indication need determination in a similar way as PAA, but it deals with the physician determinant factor. Some standard and best practice information about physicians against various patient cases that helps the analysis can be searched from CKB and compared with the physician's status obtained from the CIS along with a patient case.
- **Healthcare System Analyst Agent (HSA)**: This agent is responsible for analyzing the Healthcare System determinants like the supply of specialists, cost, policy and procedures based on the CKB. It searches the CKB for all healthcare system factors that are related to the patient case and decides whether these factors indicate referral or not. For example, if a specialist couldn't be found within that healthcare provider, then it indicates a referral need.

5.4.4 SERVICE IDENTIFICATION (SI)

In the RI, if referral is indicated, the system flow takes the SI path, a stage where a required service is identified before transferring the referral to the referred-to provider. At this stage a physician or administrative staff of the referring healthcare provider can request for SI decision support system that considers and analyses all the factors that affect the required service determination and finally suggests or recommends what service the patient is referred for. In such analysis, a service directory, which could be implemented in the CKB, contains the list of all available services, their types and conditions against various patient cases.

In order to identify the required service, the referring clinician first requests the SI decision support, which in turn consults the service guidelines available in the CKB and makes a decision based on the factors and conditions available. This SI decision support which consults the CKB for authorized service directory can also be used by referring physicians before Abroad Referral decision to check the availability of the service in the country. The identified service may be provided by multiple providers and in such a case the provider selection would take care of the selection of the best suitable provider based on its multi-criteria decision support techniques [6].

A **Service Analyst Agent (SAA)** is responsible to analyze the service directory from the CKB to identify the required service and to provide the best service for the referral. Whenever SAA receives a required service identification request for a patient case that needs to be referred, it **searches** the CKB for all matching services to the requested patient case and returns these as identified services to RCA. However, SI is less complex than RI and LC as the service to be identified is already known by the physician in most cases at RI stage.

5.4.5. THE LOCAL CONSULTATION (LC)

After referral consideration, if referral is not indicated, the patient is provided with local follow-up procedure. However, it is evident that if referral is not indicated then it means that there is some medical uncertainty with the physician considering the referral. It has also been mentioned that physicians need to consult specialists in order to overcome such uncertainty [14, 20]. Here, a clinical knowledgebase is proposed to be employed to provide the clinicians with the required information which can help them to make informed medical decision in general, and for local medical follow-up and medical consultation.

Local medical consultation is aimed at minimizing the rate of over-referrals by availing efficient local specialist consultation. LC takes place in a clinical environment as part of the normal clinical process: patient examination, diagnosis and treatment. As uncertainty and referral consideration could arise in any of these processes, the physician can request for follow-up treatment plan from the LC, which in turn consult the CKB that is supposed to capture the best practices of most of the clinical interventions and patient cases.

In LC, a physician initiates a consultation request, on any uncertainty in dealing a patient case, to the system and the system in turn consults the CKB on the cases that have been decided not to be referred. The locally deployed knowledgebase handles all consultation requests through the Knowledgebase Wrapper Agent (KBWA) and it provides best matched information for the requester.

The following agents are part of the LC:

- **Medical Consultant Agent (MCA)**: This agent provides medical consultation for the healthcare professionals. When there are clinical uncertainties, this agent plays the role of specialist consultant in supporting all clinicians in relation to the case under consideration. Moreover, this agent plays a key role in identification of best and appropriate service if referral is not needed after referral consideration. Upon receipt of medical consultation request for a specific patient case from the RCA, this agent passes a consultation **search** request to the KBWA. The consultation request can be alternative examinations and diagnosis approaches for a specific patient case and or preferred follow-up treatment plans that have been proved to be best practices and stored in the knowledgebase. If more alternatives are returned from the KBWA, then it ranks the list based on the information it has received from the user. Then it returns the alternative clinical process plans to the requester.

- **Knowledgebase Wrapper Agent (KBWA)**: this agent provides a knowledgebase wrapper service that searches and fetches required information from the CKB based on the request it receives from MCA and passes it back with preferred and understandable format for the requester. This agent receives a consultation requests from MCA and it undertakes a **wrapper search** process from the CKB. Then it returns the list of alternative processes and treatment plans to the MCA.

5.4.6 DECISION AID STRATEGY

In this section, decision aid technique that outlines how agents achieve their design goals is presented. Several agents which play various roles in the multi-agent decision aid model for referral analysis are identified. Figure 5.5 depicts the decision aid technique that explains how each agent of the model processes and cooperates with each other.

Figure 5.5: Agents flow and interaction algorithm

This algorithm identifies and describes key processes and decision aid flows each agent follows as well as their interaction.

As a starting point, the UIA agent wraps users request from the CIS and passes the request to the RCA. RCA in turn determines the type of request as RI, SI or LC request and trigger the appropriate agent that handles the request. If it is an RI request, it passes the request to PAA, which searches the CKB for matching the patient's case in the request. The search result values of referral indication are compared with the patient determinant values that have been passed along with the patient's case. The result from the comparison determines whether referral is needed based on the patient determinants. If referral is needed, the request will be passed to the PHA for further analysis based on the physician determinants, otherwise the feedback is sent back to the RCA for LC consideration. The PHA performs similar tasks like PAA, but based on physician determinants and passes the request to SAA in a case where referral is needed. Then the HSA searches the CKB for healthcare determinants related to the patient's case under consideration and evaluates referral needs. If HSA determines the need for referral, which means referral is indicated based on the three determinants, then the request is passed to SAA as SI request to identify a required service. The SAA searches the CKB for services that match the patient's case and returns the result to the RCA. In all cases where referral is not needed, the RCA triggers an LC consultation request to MCA, which in turn consults the CKB through KWBA for alternative medical consultation and treatment plan. After referral is indicated and the required service is identified the algorithm continues to the provider selection stage.

5.4.7 SUMMARY

In this section, we have proposed a multi-agent decision support model for referral analysis which undertakes the referral indication and service identification decisions as well as local medical consultation. Several agents have been identified that interact and cooperate to achieve their design goals. Important clinical environment components are also identified and integrated in the model. Then a decision aid technique that details the roles, processes and interactions each agent undertakes in the MAS have been presented.

It is believed that this model can be integrated with the MARDS framework to realize efficient and cost effective medical referral system. This in turn improves the overall healthcare outcomes.

In the next chapter, a prototype implementation of the MAS decision support model will be presented.

CHAPTER 6 : PROTOTYPE IMPLEMENTATION

6.1. Overview

This chapter discusses the prototype implementation of Multi-Agent Decision Support (MADS) model for Referral Indication and its decision pathways: Service Identification and Local Medical Follow-up aspects. The aim of this prototype is to demonstrate the feasibility of the proposed decision aid model.

In the first stage of the development of this prototype, the scope of the implementation, the tools and technology platform have been determined. Then detailed design has been done within the scope and the identified platform. Finally, a running example that tests the prototype is presented.

This implementation mainly focuses on the multi-agent design of the proposed model and four agents are developed. These include the design of RCA, PAA, SAA and MCA agents. The roles of each agent are discussed in the proposed MADS model in Chapter 5. Accordingly, the RCA plays the role of coordination between the clinical GUI (Graphical User Interface) and the other three agents. The UIA is simulated by GUI which acts as if it plays a wrapper role of fetching information from the CIS. The design of the CIS and the CKB are out of the scope of this prototype. However, some key aspects of the CKB, which will be used as an input for the agents design, has been simulated by some hardcoded Java Hashtable entries. The PAA agent implements the referral indication decision support service while the SAA handles the service identification requests. The MCA implements the provision of local medical consultation service to the user.

In this prototype, a user enters a patient case and its basic clinician data and requests the system for decision support through the GUI. Then the system provides the referral indication decision support by determining if referral is needed or not. If referral is indicated, it also helps to identify required services; otherwise it provides medical consultation through identification of follow-up treatment plan for a patient case under referral consideration.

A Java based JADE (Java Agent DEvelopment) platform has been employed for the agents development. The latest versions of JDK (Java Development Kit) and NetBeans IDE (Integrated Development Environment) have been used in the development of this prototype.

The architectural design of the prototype shows the main components and their interaction. These include the GUI, the RCA, PAA, SAA and MCA agents and the CKB Hashtables[5] that provide RI, SI and LC services. The detailed design of agents also depicts the role of each agent class and their interaction with each other and other components in the fulfillment of their design goals. The source code and the details of the design implementation of the prototype development is annexed in Appendix II as 'Java Documentation'

A running example has been set up to experiment the developed prototype and the results of selected patient cases are presented.

The reminder of this chapter is organized as follows: section two presents the development approach followed; section three discusses the system design, and finally section four presents the running example of the prototype.

6.2. Development Approach

6.2.1. Services of the prototype

The following is a list of key features of the prototype that provide referral decision support services to its users.

Clinical Information Maintenance: As indicated earlier, the implementation of Clinical Information System (CIS) is out of the scope of this prototype. Hence, this service simulates the CIS. It provides graphic user interfaces to allow users to enter key patient information which is important for referral indication decision.

Clinical Guideline Knowledgebase: Clinical guidelines capture best practices for providing medical care [33]. As the prototype doesn't implement the CKB, some key guidelines and knowledge items that are relevant to the experiment are hard coded and captured in Java Hashtable.

Referral Indication: This is a key service which implements the referral indication aspect of the proposed multi-agent decision support model. It enables users to enter a referral indication request along with patient case and its basic clinical information. Then it analyses and compares the patient decision factors based on the patient information from the CIS

[5] Hashtable is a Java class that implements a hashtable, which maps keys to values that could be any non-null object.

against the best practices and guidelines in the CKB and finally returns the recommendation for referral or otherwise.

Service Identification: If a referral is indicated, the system identifies the required service from the list of services captured in the Hashtable that simulates the clinical knowledgebase, which contains service directory. This service implements the service identification aspect of the proposed multi-agent decision support model. It allows users to enter the required service identification request and it analyses various factors and returns the service.

Local Consultation: If a referral is not indicated, then the system provides further medical consultation to the user about the next medical treatment plans required. This consultation is aimed at addressing uncertainties with users if there are any. This service is provided based on the analysis made on the clinical knowledgebase. Moreover, a user can consult this service at any time when alternative medical treatment plan or second opinion is needed from the clinical guidelines or knowledge.

6.2.2. DEVELOPMENT TOOLS

The development tools that have been deployed to implement the Multi-Agent Decision Support (MADS) Model for Referral Indication, Service Identification and Local Consultation are described as follows.

6.2.2.1 JADE Version 3.4. (Java Agent Development Framework)

JADE (Java Agent DEvelopment) is the main tool used in the development of the MADS system. It is a Java-based middleware that facilitates the development of multi-agent systems that conform to FIPA[6] (Foundation of Intelligent Physical Agents) specifications.

The JADE platform has three key features that have been employed in the development of this prototype [34]. First, it includes a **library of classes** that developers may use (directly or by customizing them) to develop their agents. This helps to define agent class services as cyclic, one shot and generic behaviours. Secondly, it has a **runtime environment** where JADE agents can run indefinitely. This environment must be active on a given host before one or more agents can be executed on that host. Thirdly, it has a set of **graphical tools** that

[6] FIPA specifications represent a collection of standards which are intended to promote the interoperation of heterogeneous agents and the services that they can represent (http://www.fipa.org/)

allows administrating and monitoring the activity of running agents that helps developers in debugging and deploying their system.

Figure 6.1 depicts the key components of JADE platform that are described as follows. **Container** is a running instance of the JADE runtime environment that can contain several agents. A set of active containers is called a **Platform**. A single special **Main container** must always be active in a platform and all other containers register with it as soon as they start.

As shown in Figure 6.1, there are two JADE platforms composed of 3 and 1 containers respectively. JADE agents are identified by a unique name and, provided they know each other's name, they can communicate transparently regardless of their actual location: same container (e.g. agents A2 and A3 in Figure 6-1), different containers in the same platform (e.g. A1 and A2) or different platforms (e.g. A4 and A5). In addition to the ability of accepting registrations from other containers, a main container differs from other containers as it holds two special agents: AMS (Agent Management System) and DF (Directory Facilitator). These agents are automatically started when the main container is launched.

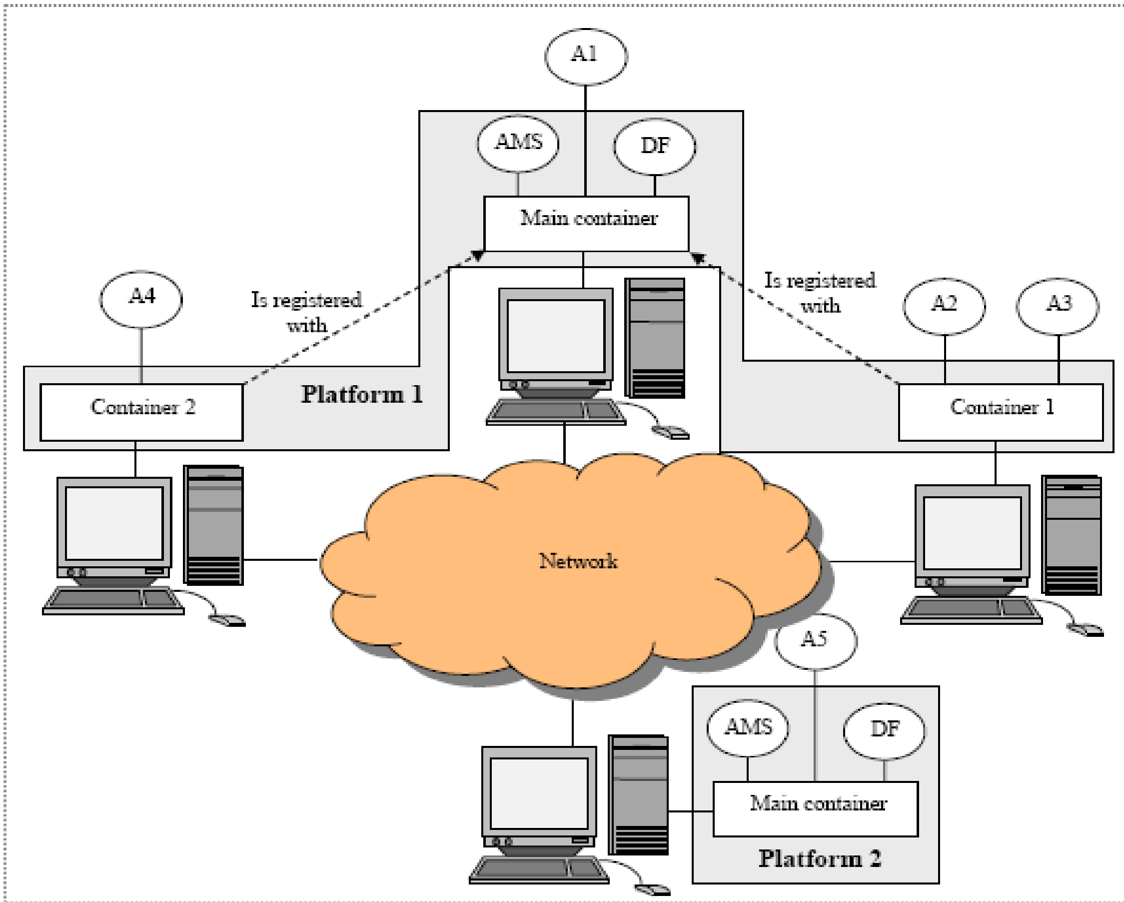

Figure 6.1 JADE Platform

The AMS provides a naming service and ensures that each agent in the platform has a unique name. It represents the authority of creating/killing agents on remote containers in the platform. The DF (Directory Facilitator) provides a yellow pages service by means of which an agent can find other agents providing the services it requires in order to achieve its goals.

6.2.2.2 JDK-6u6[7]

The JDK SE 6 is the current major release of the Sun's Java SE platform, which includes the Java Runtime Environment (JRE) and command-line development tools that are useful for developing applets and applications. This kit has been used to create the prototype agents using the Java programming language and JADE library classes.

[7] Java Development Kit - Standard Edition (SE) Platform– Version 6, Update 6

6.2.2.3 NetBeans IDE 6.1

NetBeans IDE is an open source Java development environment. This is the latest version of an integrated development environment (IDE) for writing, compiling, testing, and debugging applications for the Java platform. NetBeans IDE includes a full-featured text editor with syntax highlighting and error checking, visual design tools, and many other features. The agents of this prototype are created as Java class, packaged and compiled using this IDE before they are deployed in JADE platform.

6.3. System Design

The main design objective of the multi-agent decision support system (MADSS) is to experiment the feasibility of the proposed MADS model, which aims at helping clinicians in making better referral indication and service identification decisions as well as provision of local medical consultation. The system design is described in terms of its architectural, agent, and user interface designs.

6.3.1. System Architecture

The general architecture of the MADS system is depicted in Figure 6.2, It is organized and described in three layers: the clinical environment, MAS layer and JADE platform.

6.3.1.1 The Clinical Environment

This is the mapping of information layer of MARDS layer and it contains a CIS (Clinical Information System) that manages clinical processes and captures key information of patients, physicians and other entities in the environment.

A user of this system interacts to the GUI and will be able to enter a patient’s case along with associated clinical details. This GUI enables the user to submit requests like RI (referral indication), SI (service identification) and LC (local consultation). Then the GUI passes the user’s request to RCA, which in turn coordinates to pass the request to an appropriate agent in the multi-agent layer according to the type of a request.

UIA (User Interface Agent) GUI: This GUI simulates the CIS of a clinical environment through which a user can edit or maintain basic patient and disease related information (diagnostic conditions, disease severity, comorbidity, demographic characteristics, and behavioral characteristics). However, only clinical factors of a patient's case have been considered in this prototype and hence the user selects the values for these clinical characteristics from a drop-down list. After entering these characteristics, a user can submit a referral decision support or medical consultation request in this GUI.

6.3.1.2 The Decision Support/MAS layer

This is a multi-agent layer that provides the RI and SI decision support and LC service. This layer is implemented using four agents (RCA, PAA, SAA & MCA) and several Java Hashtables that simulate a Clinical Knowledge. First, the services of this layer are presented followed by the detailed description of each of the agents and their behaviour classes.

- **RI (Referral Indication) decision**: this is a decision made to determine whether referral is needed or not. It is a critical and basic decision that is taken in the whole referral decision making process. The referral indication decision is mainly affected by highly interrelated patients, physicians and healthcare system determinants. In this prototype, all the three determinants have been considered; however, in each category only limited factors have been taken into account. Patient determinants are captured from users through the GUI that simulates the UIA. In this design, only clinical characteristics of patients' factors have been considered. These factors are grouped into three categories: diagnostic conditions, disease severity and comorbidty. On the other hand, physician and healthcare system determinants are hard coded in the system using the Hashtables. For physician determinants, scope of GPs (General Practitioners) is considered. This is to determine if a patient's case under consideration could be handled by GPs or not. For known patient cases, the information in or out of the scope of practice of GPs is hard coded in a Hashtable. Finally, the availability of a specialist service for various patient cases within the provider is also hard coded for healthcare system determinants.
- **SI (Service Identification) decision:** this determines a required service from the list of available services. The system performs the identification with the help of a service directory in the Clinical Knowledgebase (CKB) and the analysis of specific case against relevant SI determinants. In this design, the SI decision support is achieved through a hashtable and the ***service identifier*** behaviour of SAA (Service Analyst Agent) agent.
- **LC (Local Consultation):** it is a medical consultation service provided by a Clinical Knowledgebase (CKB) for patients' cases to which a referral is not indicated after RI decision. The consultation happens through matching of the patient case under consideration with similar cases with best practice treatment and follow-up plan outlined in the CKB. The CKB is assumed to capture and implement all known

clinical cases so that the clinicians will have local medical consultation without communication and organizational hassles. The LC service is implemented by ***followupTreatmentPlans*** hashtable and ***Follow-up Treatment Plan Identifier*** behaviour of MCA agent.

The multi-agent decision support is achieved through RCA (Referral Coordinator Agent), PAA (Patient Analyst Agent), SAA (Service Analyst Agent) and MCA (Medical Consultant Agent) that interact both with GUI and the CKB HashTables. The four agents are shown in Figure 6.2 and described below.

RCA (Referral Coordinator Agent): This is a referral coordinator agent class, which coordinates the decision flow by handling requests from GUI and passing to the appropriate agent classes (PAA, SAA or MCA). The main objective of this agent class is to effect efficient interaction between the users and the decision making processes. To achieve this objective, it implements three agent behaviors (***RI request handler, SI request handler and LC request handler***) to handle all types of users' decision support requests.

- ***RI Request Handler:*** this is a generic JADE agent behaviour designed to implement the referral indication decision support request handling by waiting for a request from GUI and passes the request to PAA for referral indication analysis. It employs ACL (Agent Communication Language) message passing of JADE platform to communicate the PAA agent. Upon receipt of referral indication result, this agent passes the results to the GUI to be presented to the end user. In this prototype, the result is displayed in the JADE platform for a user.

- ***SI Request Handler:*** this is also a generic behaviour designed to implement the service identification decision support request of a user through the GUI and passes the request to SAA for service identification analysis. It also employs ACL message passing to communicate the SAA agent and upon receipt of service identification result, this agent passes the results to the requester.

- ***LC Request Handler:*** this is the third generic RCA behaviour designed to implement the local consultation requests of a user. Whenever it receives a LC request from a user through the GUI, it passes the request to MCA for identification follow-up treatment plan. It communicates to the MCA in the same way as above behaviors and upon receipt of medical follow-up treatment plan, this agent passes the results to the user.

PAA (Patient Analyst Agent): This is a patient case analyst agent class that analyzes the patient case under consideration with the aim of providing referral indication decision support to the user. To achieve this, it implements a cyclic agent behavior: referral need identifier, which is described below.

- ***Referral Need Identifier:*** this is a cyclic JADE agent behavior designed to compare the patient case under referral consideration (received from a user) with the referral indication determinants (patient, physician and healthcare systems). The specific clinical characteristics of a patient are received from the GUI. A referral need of each patient case under consideration is determined by fetching the referral indication value from patient's clinical factors, physicians and healthcare system determinants that are implemented in hashtables. These comparison results are captured as Boolean value of the referral need. Finally the results of the three determinants are computed to determine referral need as ***true*** or ***false*** values and this value will be passed as a referral indicated or not value.

SAA (Service Analyst Agent): This is a service analyst agent class that analyzes the patient case to providing service identification decision support to the user. This service is achieved through the implementation of an agent behavior referred as service identifier.

- ***Service Identifier:*** This is a cyclic behaviour designed to implement the identification of a required service from the service list. The aim of this service identifier behaviour of SAA agent class is to support service identification decision making processes for indicated referrals before provider selection is made. It analyzes the required service against the service identification determinants whose information can be retrieved from a hashtable. It retrieves a list of services from a services list for the patient's case under consideration. In this system, these services are fetched from the list of services that are implemented in a Hashtable class of Java platform.

MCA (Medical Consultant Agent): This is a medical consultant agent class that provides local consultation service to the user on patient cases that don't need referral. This is implemented by a follow-up treatment plan identifier with input from a clinical best practice and guidelines records in a hashtable.

- ***Follow-up Treatment Plan Identifier****:* this is a cyclic JADE agent behaviour designed to implement the local medical consultation. It provides an alternative follow-up treatment plan from the best practices and clinical patterns stored in CKB.

The main objective of this behaviour is to provide medical consultation service for patient cases which need the necessary treatment and follow-up plans. If referral is not indicated, then this service is consulted to identify and decide on the best follow-up treatment plan. This is achieved by comparing the patient case under consideration against the 'follow-up treatment plans' captured in the hashtable.

Clinical Knowledge Hashtables: This is a Java Hashtable class that simulates the CKB and contains various hashtables for different aspects of CKB simulation. These Hashtables has been mentioned above wherever they have been implemented. These Hashtables are implemented in each of the three agents: PAA, SAA and MCA.

6.3.1.3 JADE Agent Platform

This is a multi-agents development framework that has been employed for the implementation of agents design. Its features and services are described earlier. The agent classes of this prototype are deployed in JADE agent platform. In this system, all the inter-agent communications are implemented by using the ACL message based interaction of this platform.

6.3.2. DETAILED DESIGN

A detailed design of the MADSS prototype is presented in this section in terms of decision aid flow, agent model, interaction models and user interface designs.

6.3.2.1 Interaction Model

An interaction model defines the cooperation that should exist between various agents of a MAS to achieve the desired goal [38]. The interaction model of this design focuses on the decision aid flow of all key decision pathways in MADSS in its architectural design and inter-agent interactions. Figure 6.3 depicts this interaction model.

Figure 6.3: Interaction Model.

The system flow starts when it first takes a patient's clinical case as an input from a user through the GUI. Then the GUI passes the request to the RCA which in turn triggers one of the three agents (PAA, SAA or MCA) based on the type of request. The DF handles all agents' registration requests. These agnates handle the requests with the help of their behavior and the Hashtables.

6.3.2.2 Agent Model

As indicated in Gaia [38], an agent model identifies and specifies which agent classes are to be defined to play specific roles and how many instances of each class have to be instantiated in the actual system. This agent model can be defined using a simple diagram that specifies for each agent class which roles to play. In addition, the agent model can document the instances of a class that will appear in the MAS.

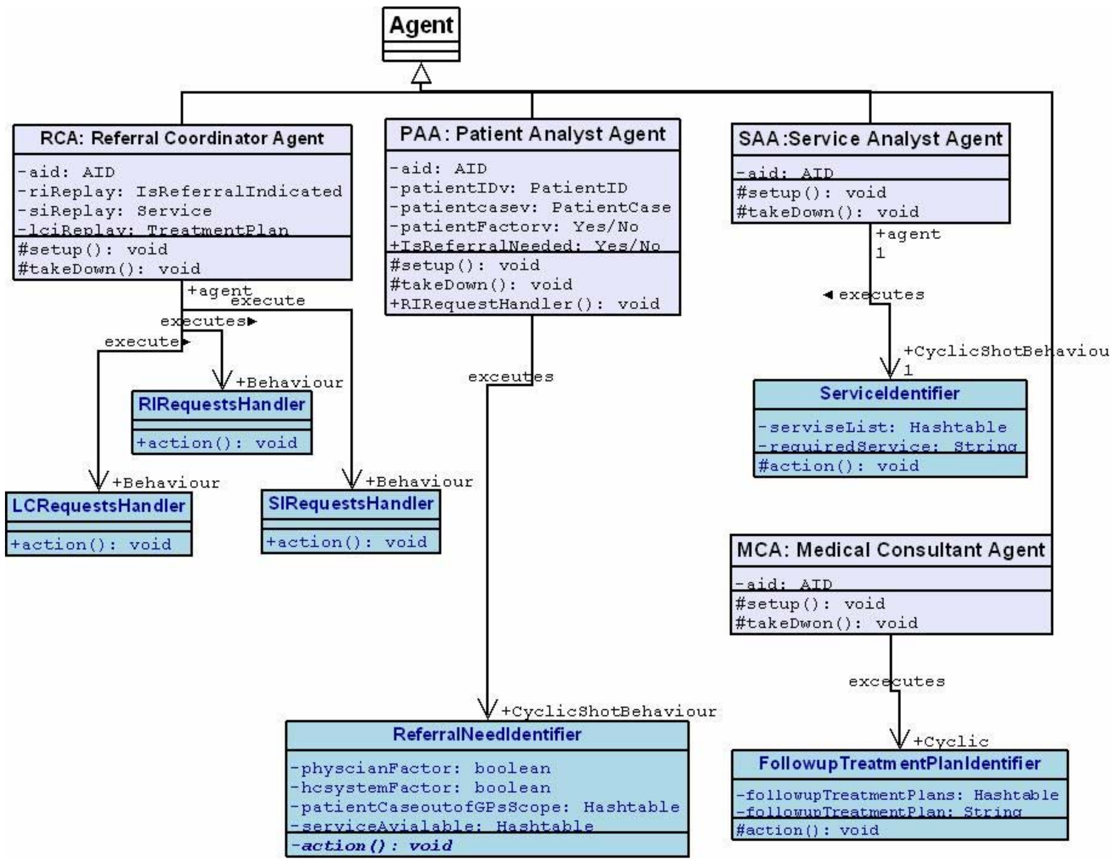

Figure 6.4: Agent model

As shown in Figure 6.4, the agent model of the MADS system specifies all the attributes, operations, roles and association between the agents and their components.

6.3.2.3 Interface Design

Figure 6.5 shows a Clinical GUI that simulates the CIS to capture and process patient information.

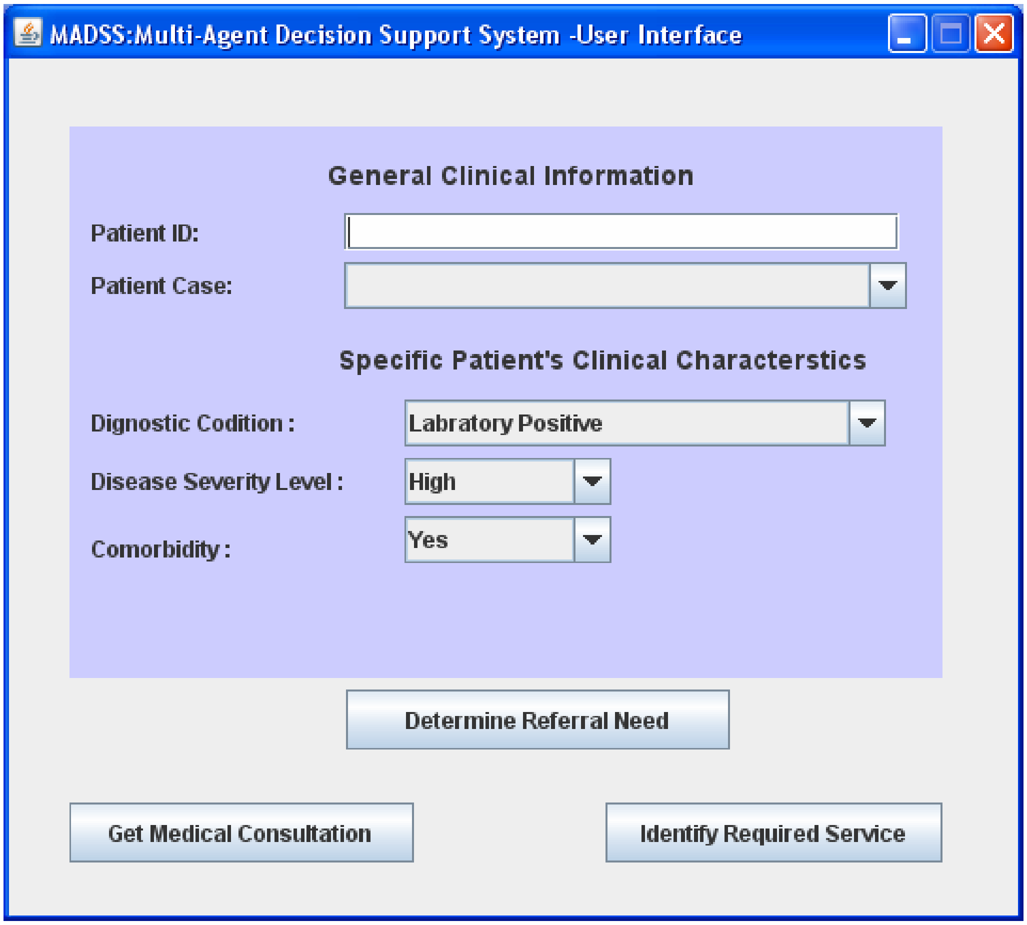

Figure 6.5: Screen shot for Clinical GUI of MADSS

Through this interface a user can enter a patient identification and related data. A Patient's Case is an important input that a user selects from the list of clinical known cases in the combo box. The specific patient clinical characteristics entries are meant to capture determinants that contribute for the referral indication decision. After making appropriate entries, a user can request referral indication decision support medical consultation or demand service by selecting the appropriate command.

6.4. Running Example

This section presents a running example aimed at showing how the system implements the referral decision analysis model. The scenario setup and the output of running the system will be presented in the following two sub-sections.

6.4.1. Scenario Setup

Two scenarios have been chosen to show how the system provides referral indication and service identification decisions as well as local consultation decision. These two cases are Cancer and Malaria cases as shown in Table 6.1.

Table 6.1: Testing patient cases and their alternative specific clinical characteristics

Clinical Information		**Specific Patient Case Characteristics**		
Patient ID	Patient Case	Diagnostic Condition	Severity	Comorbidity
PT101/2008	Cancer	Laboratory Positive	High	Yes
		Laboratory Positive	Low	No
PT102/2008	Malaria	Laboratory Positive	High	Yes
		Laboratory Negative	High	Yes

The RCA, PAA, SAA and MCA agents (named as **rca, paa, saa** and **mca** respectively) are initialized as shown on screenshot in Figure 6.6.

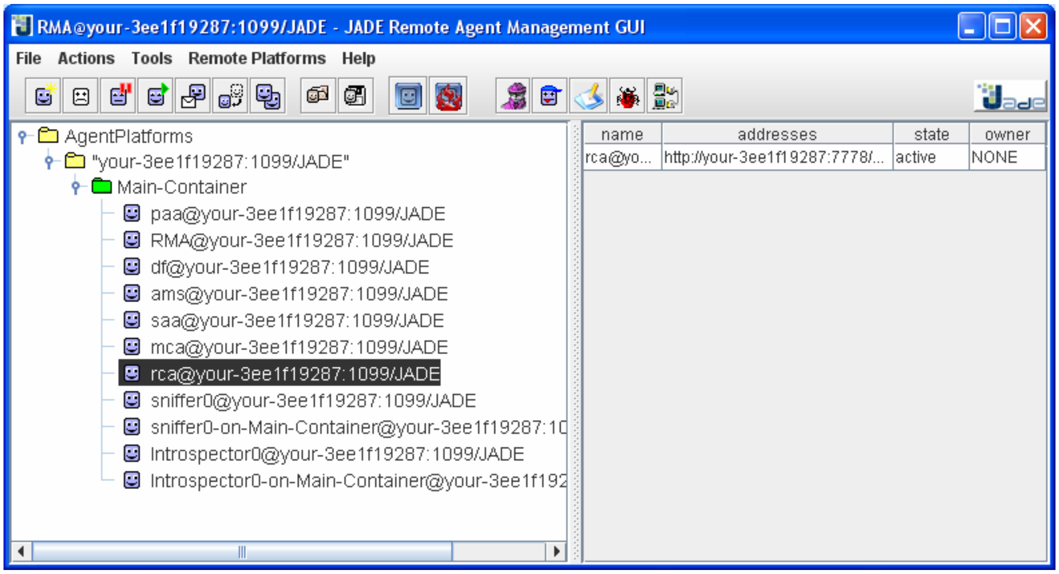

Figure 6.6: Screenshot of the MADSS and Built-in Agents Running in JADE Platform

6.4.2. THE OUTPUT /RESULT

In the above scenario, suppose that this system is running in a primary/district hospital where there are GPs and limited number of specialist services.

A user enters cases listed in Table 6.1 in that order. The GUI screenshots of the users request for the first case is shown in Figure 6.7. When a user selects ‘Determine Referral Need’ button on the GUI, the system handles the request and the result of this request is shown in Figure 6.8. This result is as expected, as most of the time referral is needed for cancer cases according to clinical practices.

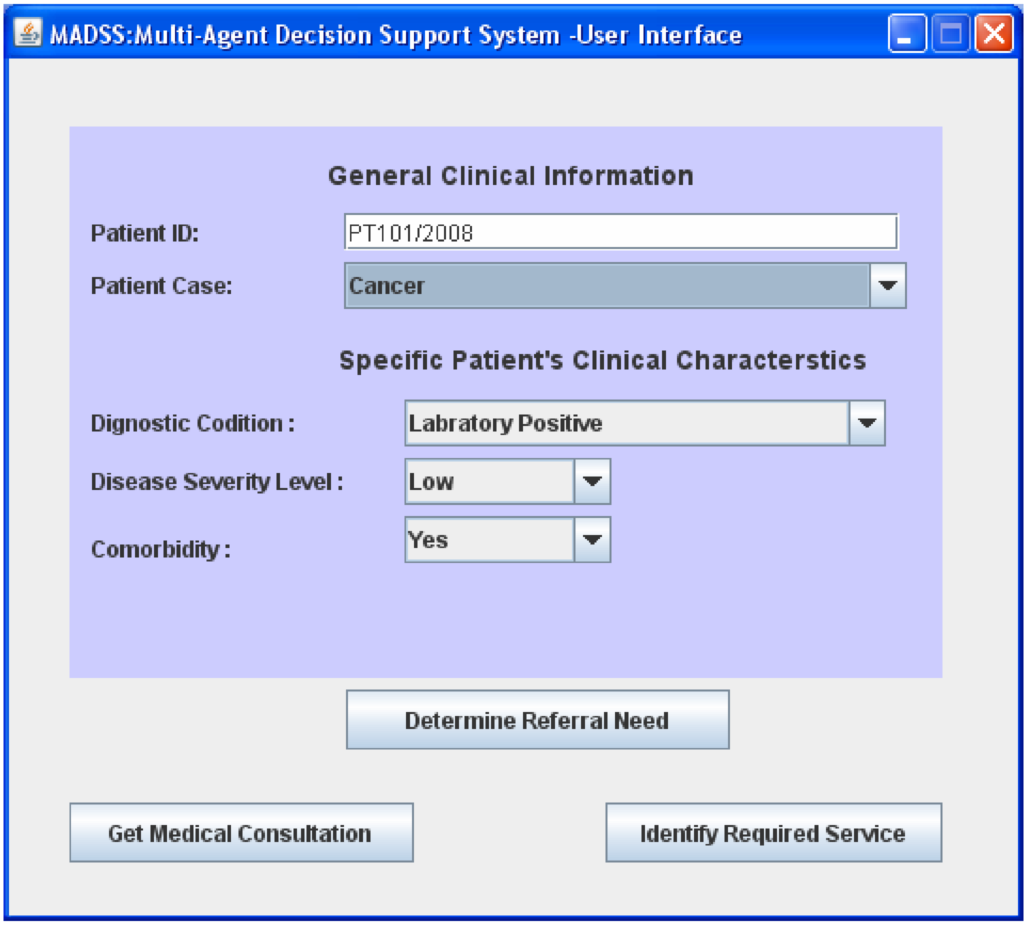

Figure 6.7: GUI for users' referral decision support request

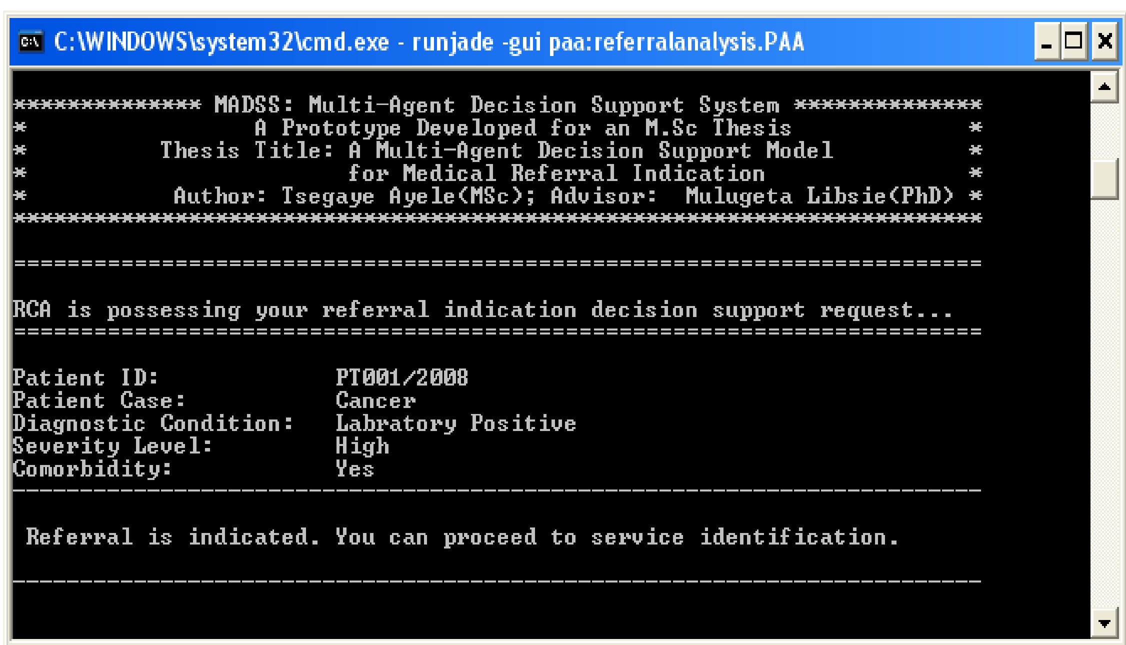

Figure 6.8: Screenshot of the output of referral indication request

As the result in Figure 6.8 suggests, the next step is to request for service identification. When a user selects 'Identify Required Service' from the GUI, the system returns the required service for Cancer case as shown in Figure 6.9.

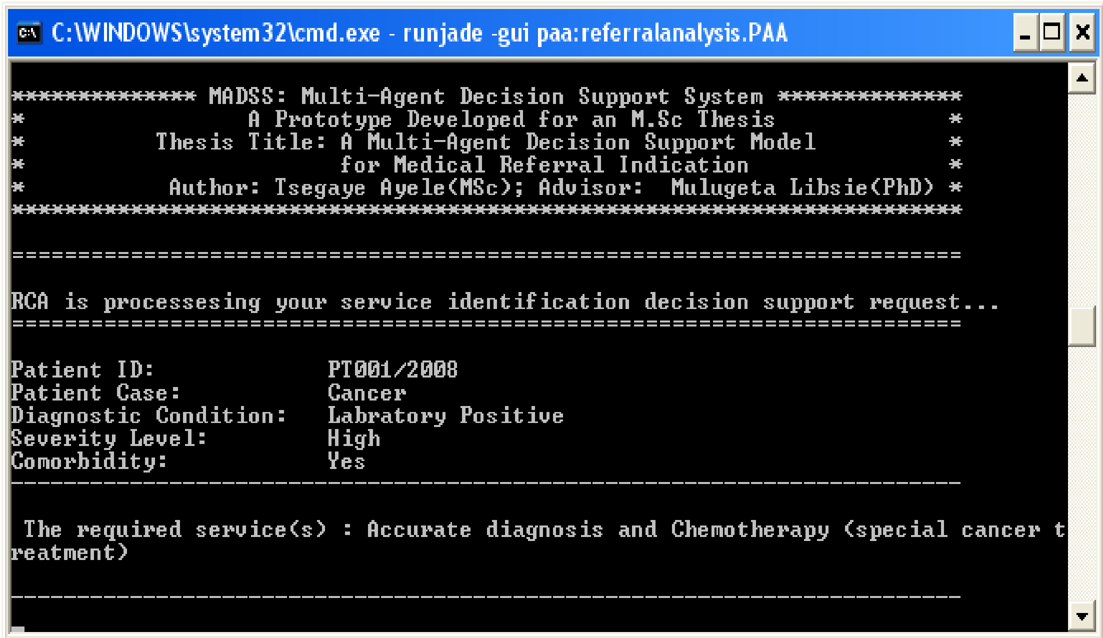

Figure 6.9: Screenshot of the output of service identification request

For the Cancer case, if a user selects request 'Determine Referral Need', but this time with the Severity of the disease set to 'Low', the system determines and displays that referral is not needed and suggest a user to proceed to local consultation as shown in Figure 6.10. Accordingly, when a user selects 'Get Medical Consultation', the system identifies and suggests a follow-up treatment plan for the user, as shown in Figure 6.11.

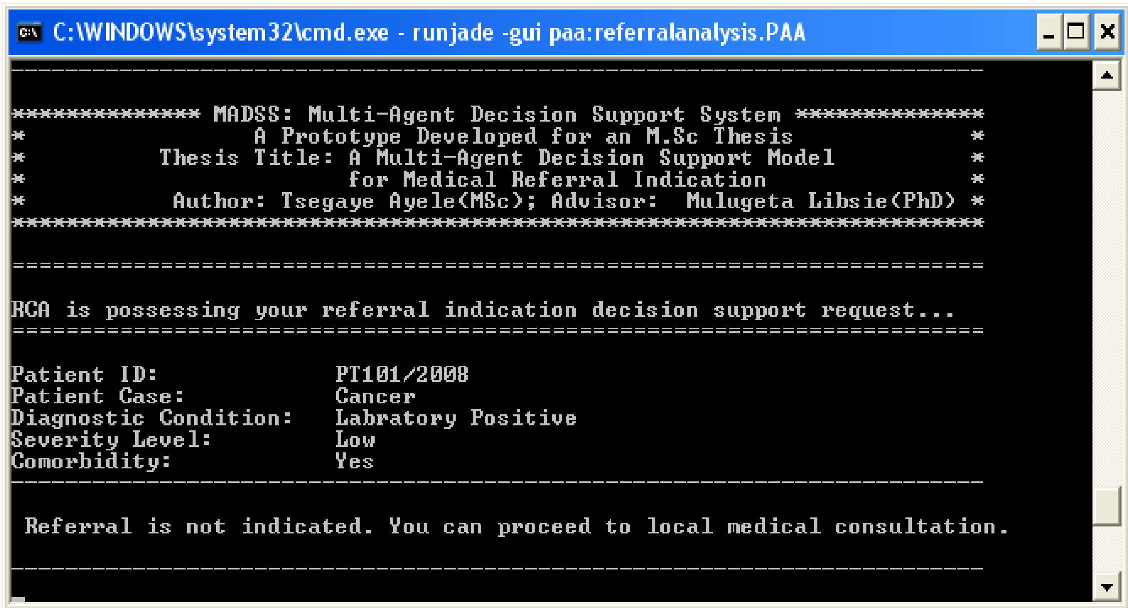

Figure 6.10: Screenshot of the output of referral indication request with diseases serverity set to 'Low'.

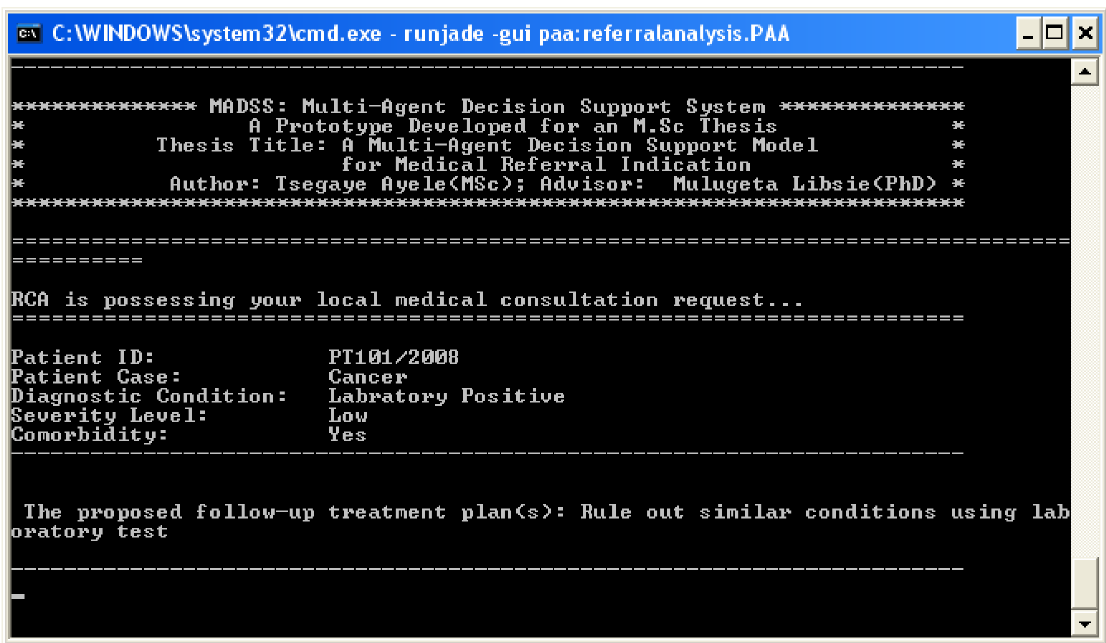

Figure 6.11: Screenshot of the output of local consultation request

In the second case of the scenario, Malaria has been selected as patient case and other values as set in Table 6.1. In this case, referral is not indicated and the user can get local medical consultation as shown on Figures 6.12 and 6.13.

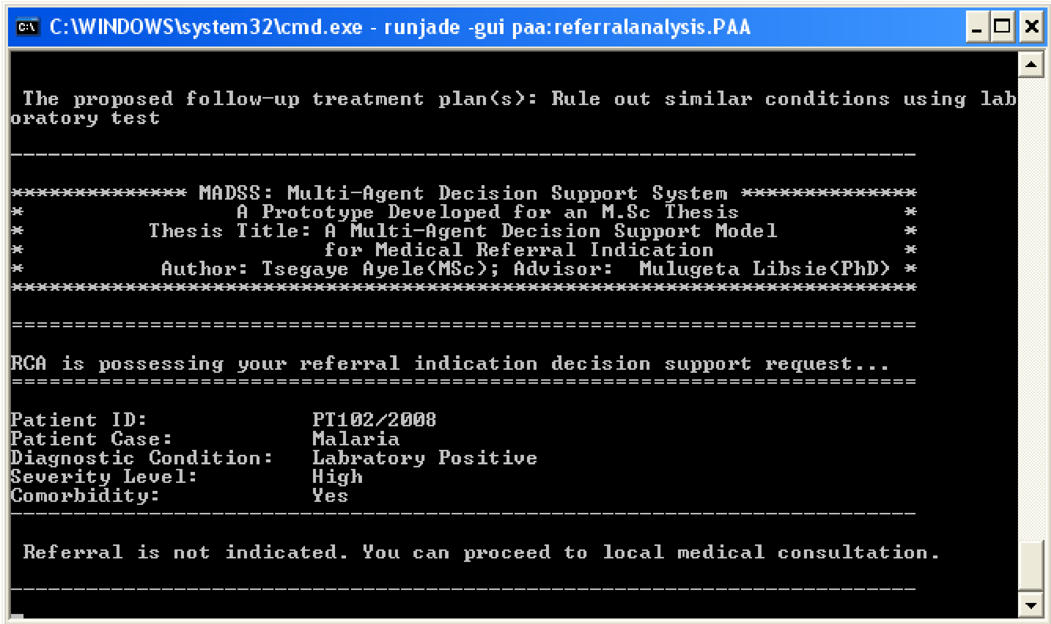

Figure 6.12: Screenshot of the output of referral indication request for Malaria case

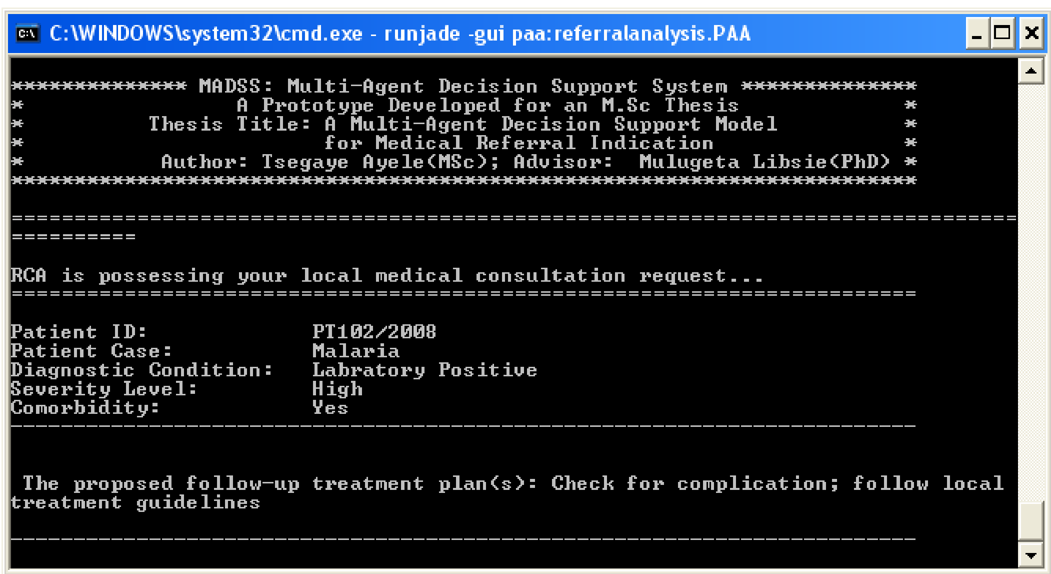

Figure 6.13: Screenshot of the output of local consultation request for Malaria case

It is important to note that referral is not indicated for both Malaria cases even if patient determinants indicate referral need since the physician factor, i.e, out of scope, couldn't indicate referral.

During the running of these cases, inter-agent interaction has been captured by the JADE platform Sniffer as shown on Figure 6.14. This indicates the desired interaction among agents where the platform takes care of the communication aspect of the agents.

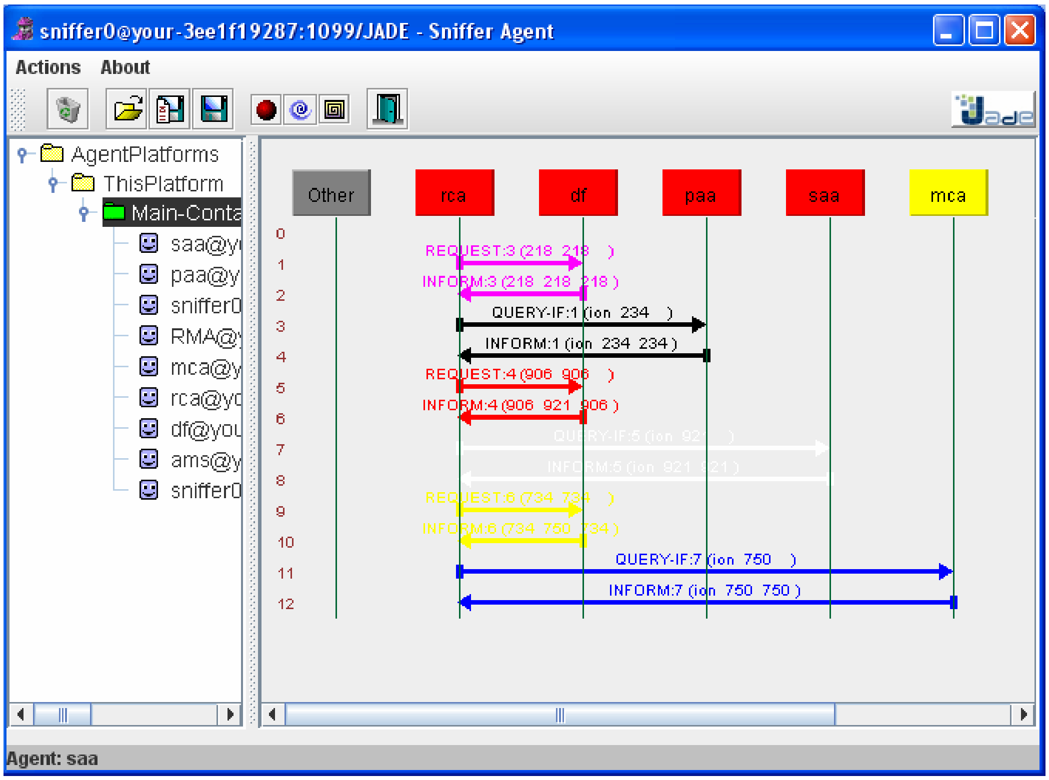

Figure 6.14: Screenshot of the Inter-agent Interaction

CHAPTER 7 : CONCLUSION AND FUTURE WORKS

7.1. Discussion

A Referral Indication is a crucial aspect of a Medical Referral System, which is an area of the healthcare domain characterized with diverse organizational and technical challenges. A referral indication is the first decision made by a clinician in the referral process. If referral is indicated, the clinician identifies a required service before provider selection; otherwise a further medical follow-up continues locally. These various aspects of the referral system contribute to the challenges of the referral system. Hence, it is not an easy task to provide an IT solution to the problems of such an area since it requires a coordinated effort of professionals from diverse fields.

In this thesis, a Multi-Agent Decision Support (MADS) model for Referral Indication, Service Identification and Local Consultation aspects of medical referral has been proposed. It is believed that this model extends the Multi-Agent Referral Decision Support (MARDS) framework. The proposed decision support model undertakes the analysis of determinants related to the referral indication, service identification decisions and local medical consultation and provides decision support to the user.

Various agents that interact and cooperate to achieve their design goals have been identified. Important clinical environment components are also identified and integrated in the model. The decision support process is supported by the information from the existing CIS (Clinical Information System) and the CKB (Clinical Knowledgebase). CIS provides basic clinical information while the CKB is used as reference for best practices and guidelines in undertaking decision support in various aspects of the referral process.

This model incorporates four cooperating agents with different roles that interact and exchange information with each other and with the CIS and CKB in order to achieve their design goals. The aim of decision support is helping physicians in making better referral decisions in terms of referral indications and service identification. Besides, the model has been designed to provide local medical consultation service in case where referral is not indicated.

The model has been implemented and tested by a prototype development of its core services. The Multi-Agent Decision Support System (MADSS) has shown the feasibility of the proposed decision support model.

This work is believed to enhance the MARDS framework by integrating the model that supports referral indication and service identification aspects of the referral system. Moreover, it addresses the communication and organizational challenges of medical consultation by providing local medical consultation service. In general, the result of this work is strongly believed to be one step towards the realization of the MARDS framework that enhances the referral systems and the provision of efficient healthcare service.

7.2 Contributions

The main contribution of this research work is the development of a Multi-Agent Decision Support (MADS) model for medical Referral Indication, Service Identification and Local Consultation while the specific contributions are described as follows.

- A MADS model for Medical Referral Indication is developed to aid clinicians in making an appropriate referral indication decision for patient cases. It analyses various determinant factors of patients, physicians and healthcare system to determine whether referral is needed or not. It does such analysis with the help of information passed from the CIS and the best clinical practices knowledge stored in the CKB.
- The Service Identification aspect of the medical referral process is also incorporated in the MADS model development. This model provides a decision support to clinicians in the identification of medical service after referral is indicated and before the provider selection stage in the patients' referral process. The model analyses the service's type, availability, condition and affordability determinants to suggest a service for which referral is required with the help of clinical knowledge in the CKB.
- The development of the Local Consultation component of the MADS model is a key contribution of this work. This service provides a medical consultation to clinicians who deal with a patient's case if referral is not indicated. The clinicians would be provided with a specialist consultation which is implemented in the CKB as best practices through LC service of this model. The LC component fetches and wraps medical follow-up, alternative diagnosis and treatment plans from the CKB so that clinicians address the patient case locally with minimized organizational and communication challenges. As a result, this consultation mechanism helps clinicians to overcome clinical uncertainties and contribute to minimize over-referrals.

7.3 Future Work

The MADS model proposed in this work will be used as a basis for the development of an intelligent medical referral decision support and contributes for the realization of MARDS framework. The recommended future works are:

- **Design of the Clinical Knowledgebase (CKB):** The MADS model for medical referral indication, service identification and local consultation has been developed based on the assumption of the availability of a knowledgebase that is capable of capturing, processing and storing key clinical best practices and guidelines. Hence, in order to realize this model and ensure the availability of the broader referral decision support, it is important to design such knowledgebase that provides required information and services for the multi-agent decision support models and systems.
- **Enhancement of the MADS Model:** The three core services of this model (referral indication, service identification and local consultation) are designed based on the domain experts' characterization of the service and on the assumption that this model would utilize the existing CIS and CKB. The decision aid strategy focuses on agent identification and design. Hence there is a need to study, improve and refine the decision aid techniques for these services with the application of formal methods and algorithms.
- **Integration of MRDS model:** Two types of integration work are recommended to utilize this model best:
    - The MADS model should be well integrated with target CIS and the associated clinical knowledgebase and guidelines developed as CKB. Such integration requires a deigning work of an interface with the CIS, or customization of the CIS to accommodate the model. The CKB integration can be done either during its design or as customization design for an existing CKB.
    - The other aspect is the integration of this model with the MARDS framework to come up with complete referral decision support system. This work includes the integration of this model and the provider selection model with in the formwork. This integration work can be extended to the development and

deployment of a multi-agent medical referral decision support system in a distributed environment of a healthcare domain.

REFERENCES

[1] M. Luck, P. MacBurney, O. Shehory and S.Willmott, “Agent Technoloy Roadmap”, AgentLinkII, September 2005.

[2] N.R. Jennings, “Building Complex Software Systems”, Communications of ACM, 44(4):35-41, June 2001.

[3] A. Moreno, “Medical Applications of Multi-Agent Systems”, Computer Science and Mathematics Department”, University of Rovira, Spain, 2003.

[4] J. Nealon and A. Moreno, “Agent-based applications in health-care”, In John L. Nealon and A. Moreno, editors, Applications of Software Agent Technology in the Health-Care Domain, Whitestein Series in Software Agent Technologies, 2003, pp 3–18, Switzerland.

[5] J. Nealon, A. Moreno, “The Application of Agent Technology to Health-Care”, AgentCities Working Group on Health-Care, 2000.

[6] Zewdu Gebeyehu, “A Framework for Multi-Agent Referral Decision Support”, Master´s Thesis, Unpublished, Addis Ababa University, 2007.

[7] Esther Morales, “Referral Management and Authorization”, Medical Management ‘Signature Series’ by Managed Care Resources, Inc., 1998.

[8] W. Michael, “An Introduction to Multi-Agent Systems”, John Wiley & Sons Ltd, 2002.

[9] Petrie C., “Agent-based Software Engineering”, in Agent Oriented Software Engineering Lecture Notes Artificial Intelligence, pp 58-76, 2000.

[10] S. Katia, ”Multi-Agent Systems”, AI Magazine, 1998, 19(2): 79-92.

[11] A. Valls, A. Moreno, D. S´anchez, “A Multi-Criteria Decision Aid Agent Applied to the Selection of the Best Receiver in a Transplant”, 4th. International Conference on Enterprise Information Systems, 2002.

[12] E. Turban, E. Aronson, “Decision Support Systems and Intelligent Systems”, Prentice Hall , 6th ed. 2001.

[13] L. John, Models and Managers, “The Concept of a Decision Calculus, Management Science”, 1970, 16(8)

[14] R. Bonczek, C. Holsapple, and A. Whinston, “The Evolving roles of Models in Decision Support Systems”. Decision Sciences, 1980. 11(2).

[15] D. Foster, C. McGregor, S. El-Masri , “A Survey of Agent-Based Intelligent Decision Support Systems to Support Clinical Management and Research”, in Autonomous Agents and Multi Agent Systems. 2005. Utrecht, Netherlands.

[16] B. Sweeney, “Referral System”, BMJ, November 1994, 309: 1180-1181.

[17] Wikipedia, “Health-Care”, http://en.wikipedia.org/wiki/Medcine, visited on January, 2008

[18] D. Isern and A. Moreno, “Distributed Guideline-Based Health Care System”, In: Intel. Systems Design and Applications, ISDA-2004, IEEE Press, 2004 145–150.

[19] V. Koutkias, I. Chouvardaa, N. Maglaveras, “Multi-Agent System Architecture for Heart Failure Management in a Home Care Environment”, IEEE, Computers in Cardiology, 2003, 30:383-386.

[20] J. Huang, N. R. Jennings, J. Fox, “An Agent-based Approach to Health Care Management”, 1995, Int. Journal of Applied Artificial Intelligence, 9 (4). pp. 401-420.

[21] W. Boissonnault, “Differential Diagnosis”, PT Magazine, November 2000, pp. 46-54.

[22] American Speech-Language-Hearing Association (1997), “Guidelines for Audiologic Screening”, 1997, IV - 74b.

[23] Nancy Burns & Susan K. Grove, Understanding Nursing Research: Building an Evidence-Based Practice, 4th Edition, ISBN: 9781416026402.

[24] Bernard W.K., "When And How Patients Seek Medical Help? An Exploration

In Hong Kong" ,Hong Kong Practitioner 18 (3), March 1996.

[25] D Klein, A MacDonald, N Drummond and A Cave, "A qualitative study to identify factors influencing COXIB prescribed by family physicians for musculoskeletal disorders", Oxford Journals Family Practice, 2006, 23(6):659-665.

[26] F. Zambonelli , N. Jennings , M. Wooldridge, “Developing multi-agent systems: The Gaia Methodology”, ACM Transactions on Software Engineering and Methodology (TOSEM), July 2003, 12(3):317-370

[27] F. Wood, A. DeLoach, “An Overview of the Multi-Agent Systems Engineering Methodology”, The First International Workshop on Agent-Oriented Software Engineering (AOSE2000) , 2000, 207-213

[28] M.Cossentino, C. Potts, "PASSI: a Process for Specifying and Implementing Multi-Agent Systems Using UML", URL: www.cc.gatech.edu/classes/AY2002/cs6300_fall/ICSE.pdf, last visited on April 28, 2008.

[29] A.Omicini, "SODA: Societies and Infrastructures in the Analysis and Design of Agent-based Systems", Proceedings of AOSE 2000, 2000.

[30] A. van Lamsweerde, "Goal-Oriented requirements Engineering: A Guided Tour", Proceedings of the 5th International Symposium on Requirements Engineering (RE'01), IEEE CS Press, 2001, 249-261

[31] Radziah Mohamad, Safaai Deris, Hany H. Ammar: "Pattern-Oriented Design for Multi-Agent System: A Conceptual Model", in Journal of Object Technology, vol. 6, no.4, May - June 2007, pp 55-75

[32] J. Mylopoulos, L. Chung, B. Nixon, "Representing and Using Nonfunctional equirements: A Process-Oriented Approach", IEEE Transaction on Software. Engineering, June 1992, 18(6):483-497.

[33] M. Humber, H. Butterworth, J. Fox, R. Thomson, "Medical decision support via the Internet: PROforma and Solo", Medinfo, 2001, 10(Pt 1):464-8.

[34] C. Giovanni, "JADE Tutorial: Jade Programming for Beginners", TILab S.p.A, December 2003, Advanced Computation Laboratory, Imperial Cancer Research Fund, London, UK.

[35] Martin Hensher, Max Price, and Sarah Adomakoh, "Disease Control Priorities in Developing Countries", 2005.

APPENDIX I: INTERVIEW QUESTIONS DESIGN

Interview questions for medical & admin staff for discussion on referral indication issues

(St. Paul Generalized Specialized Hospital)

A referral decision is a clinical decision-making process by which physicians determine referral indication, type of required services and selection of appropriate providers.

In referral indication, a physician decides whether a patient needs a referral or not. This decision is vital for it determines the clinical outcomes and costs of patient referral. If the decision is appropriate, it can lead to improved clinical outcomes and decreased costs. On the other hand, if it is inappropriate, it leads to under-referral (not referring patients who should have been referred) or over-referral (referring patients who should have not been referred) and all the associated problems

The aim of this interview is to obtain domain experts' input in designing of a model for Multi-Agent Decision Support for medical Referral Indication.

1. Briefly describe a medical referral process followed in your institution? Is there any documentation (referral guidelines and/or policies) that supports the referral management process?

2. What are the main challenges of medical referral in achieving quality and cost effective patient care? How best these challenges can be tackled?

3. How is medical referral knowledge shared among the professionals and the health care providers?

4. What processes and procedures that are involved in referral decision making process? Who is involved in this process? What are determinant factors that affect physicians' referral indication decisions? Are there any difficulties in these processes?

5. In referral process, what are the main sources medical/clinical uncertainties? How they are addressed?

6. What is medical consultation? How is specialist consultation done by GPs? What are the challenges?

7. What is second opinion? How is it done?

8. Role of GPs and Specialists- whose role is continuously following up the patient before and after referral process. If GPs need to follow-up the patient after referral consultation, how do they manage lack of enough feedbacks from specialist?

9. As specialist, what validation decisions process made for referral received from GPs or PCPs?

10. Any additional input ?

APPENDIX II: JAVA DOCUMENTATION

II.1 Package Description

II.1.1 Package Components

Class Summary	
RCA	Agent classes
MCA	
PAA	
SAA	
PatientCaseReply	Inter-agent Communication Objects
PatientCaseRequest	
ClinicalISGUI	GUI

II.1.2 Class Hierarchy

- java.lang.Object
  - jade.core.Agent (implements java.lang.Runnable, jade.util.leap.Serializable, jade.core.TimerListener)
    - referralanalysis.**MCA**
    - referralanalysis.**PAA**
    - referralanalysis.**RCA**
    - referralanalysis.**SAA**
  - jade.core.behaviours.Behaviour (implements jade.util.leap.Serializable)
    - referralanalysis.**RCA.LCRequestsHandler**
    - referralanalysis.**RCA.RIRequestsHandler**
    - referralanalysis.**RCA.SIRequestsHandler**
    - jade.core.behaviours.SimpleBehaviour
      - jade.core.behaviours.CyclicBehaviour
        - referralanalysis.**MCA.FollowupTreatmentPlanIdentifier**
        - referralanalysis.**PAA.ReferralNeedIdentifier**
        - referralanalysis.**SAA.ServiceIdentifier**
  - java.awt.Component (implements java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable)
    - java.awt.Container
      - java.awt.Window (implements javax.accessibility.Accessible)
        - java.awt.Frame (implements java.awt.MenuContainer)
          - javax.swing.JFrame (implements javax.accessibility.Accessible, javax.swing.RootPaneContainer, javax.swing.WindowConstants)

  - referralanalysis.**ClinicalISGUI**
- referralanalysis.**PatientCaseReply** (implements java.io.Serializable)
- referralanalysis.**PatientCaseRequest** (implements java.io.Serializable)

II.2 RCA Agent

II.2.1 Class RCA

```
java.lang.Object
  └jade.core.Agent
      └referralanalysis.RCA
```

All Implemented Interfaces:

jade.core.TimerListener, java.io.Serializable, java.lang.Runnable

---

```
public class RCA
extends jade.core.Agent
```

See Also:

Serialized Form

Nested Class Summary	
class	**RCA.LCRequestsHandler**
class	**RCA.RIRequestsHandler**
class	**RCA.SIRequestsHandler**

Nested classes/interfaces inherited from class jade.core.Agent
jade.core.Agent.Interrupted

Field Summary

Fields inherited from class jade.core.Agent
AP_ACTIVE, AP_DELETED, AP_IDLE, AP_INITIATED, AP_MAX, AP_MIN, AP_SUSPENDED, AP_WAITING, D_ACTIVE, D_MAX, D_MIN, D_RETIRED, D_SUSPENDED, D_UNKNOWN

Constructor Summary
RCA()

Method Summary	
protected void	**setup**()
protected void	**takeDown**()

Methods inherited from class jade.core.Agent
addBehaviour, afterClone, afterMove, beforeClone, beforeMove, blockingReceive, blockingReceive, blockingReceive, blockingReceive, changeStateTo, clean, doActivate, doClone, doDelete, doMove, doSuspend, doTimeOut, doWait, doWait, doWake, getAgentState, getAID, getAMS,

`getArguments, getContainerController, getContentManager, getCurQueueSize, getDefaultDF, getHap, getHelper, getLocalName, getName, getO2AObject, getProperty, getQueueSize, getState, here, isRestarting, join, notifyChangeBehaviourState, notifyRestarted, postMessage, putBack, putO2AObject, receive, receive, removeBehaviour, removeTimer, restartLater, restore, restoreBufferedState, run, send, setArguments, setEnabledO2ACommunication, setGenerateBehaviourEvents, setQueueSize, waitUntilStarted, write`

Methods inherited from class java.lang.Object
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

RCA

`public RCA()`

Method Detail

SETUP

`protected void setup()`

Overrides:
`setup` in class `jade.core.Agent`

TAKEDOWN

`protected void takeDown()`

Overrides:
`takeDown` in class `jade.core.Agent`

II.2.2 Class RCA.RIRequestsHandler

java.lang.Object
└ jade.core.behaviours.Behaviour
  └ **referralanalysis.RCA.RIRequestsHandler**

All Implemented Interfaces:
java.io.Serializable

Enclosing class:
RCA

---

public class **RCA.RIRequestsHandler**
extends jade.core.behaviours.Behaviour

See Also:
Serialized Form

Nested Class Summary
Nested classes/interfaces inherited from class jade.core.behaviours.Behaviour
`jade.core.behaviours.Behaviour.RunnableChangedEvent`

Field Summary

Fields inherited from class jade.core.behaviours.Behaviour
`myAgent, myEvent, NOTIFY_DOWN, NOTIFY_UP, parent, STATE_BLOCKED, STATE_READY, STATE_RUNNING`

Constructor Summary	
RCA.RIRequestsHandler`()`	

Method Summary	
`void`	**action**`()`
`boolean`	**done**`()`

Methods inherited from class jade.core.behaviours.Behaviour
`actionWrapper, block, block, getBehaviourName, getDataStore, getExecutionState, getParent, handle, isRunnable, onEnd, onStart, reset, restart, root, setAgent, setBehaviourName, setDataStore, setExecutionState`

Methods inherited from class java.lang.Object
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

RCA.RIREQUESTSHANDLER

public **RCA.RIRequestsHandler**()

Method Detail

ACTION

public void **action**()

Specified by:

`action` in class `jade.core.behaviours.Behaviour`

DONE

public boolean **done**()

Specified by:

`done` in class `jade.core.behaviours.Behaviour`

II.2.3 Class RCA.SIRequestsHandler

java.lang.Object
└ jade.core.behaviours.Behaviour
  └ **referralanalysis.RCA.SIRequestsHandler**

All Implemented Interfaces:

java.io.Serializable

Enclosing class:

RCA

---

public class **RCA.SIRequestsHandler**

extends jade.core.behaviours.Behaviour
See Also:
Serialized Form

Nested Class Summary
Nested classes/interfaces inherited from class jade.core.behaviours.Behaviour
`jade.core.behaviours.Behaviour.RunnableChangedEvent`

Field Summary
Fields inherited from class jade.core.behaviours.Behaviour
`myAgent, myEvent, NOTIFY_DOWN, NOTIFY_UP, parent, STATE_BLOCKED, STATE_READY, STATE_RUNNING`

Constructor Summary	
RCA.SIRequestsHandler`()`	

Method Summary	
`void`	**action**`()`
`boolean`	**done**`()`

Methods inherited from class jade.core.behaviours.Behaviour
`actionWrapper, block, block, getBehaviourName, getDataStore, getExecutionState, getParent, handle, isRunnable, onEnd, onStart, reset, restart, root, setAgent, setBehaviourName, setDataStore, setExecutionState`

Methods inherited from class java.lang.Object
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

RCA.SIREQUESTSHANDLER

public **RCA.SIRequestsHandler**()

Method Detail

ACTION

public void **action**()
Specified by:
`action` in class `jade.core.behaviours.Behaviour`

DONE

public boolean **done**()
Specified by:
`done` in class `jade.core.behaviours.Behaviour`

II.2.4 Class RCA.LCRequestsHandler

java.lang.Object
└ jade.core.behaviours.Behaviour
  └ **referralanalysis.RCA.LCRequestsHandler**

All Implemented Interfaces:
java.io.Serializable

Enclosing class:
RCA

---

public class **RCA.LCRequestsHandler**
extends jade.core.behaviours.Behaviour

See Also:
Serialized Form

Nested Class Summary
Nested classes/interfaces inherited from class jade.core.behaviours.Behaviour
`jade.core.behaviours.Behaviour.RunnableChangedEvent`

Field Summary
Fields inherited from class jade.core.behaviours.Behaviour
`myAgent, myEvent, NOTIFY_DOWN, NOTIFY_UP, parent, STATE_BLOCKED, STATE_READY, STATE_RUNNING`

Constructor Summary	
RCA.LCRequestsHandler`()`	

Method Summary	
`void`	**action**`()`
`boolean`	**done**`()`

Methods inherited from class jade.core.behaviours.Behaviour
`actionWrapper, block, block, getBehaviourName, getDataStore, getExecutionState, getParent, handle, isRunnable, onEnd, onStart, reset, restart, root, setAgent, setBehaviourName, setDataStore, setExecutionState`

Methods inherited from class java.lang.Object
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

RCA.LCREQUESTSHANDLER

public **RCA.LCRequestsHandler**()

Method Detail

ACTION

public void **action**()
Specified by:
`action` in class `jade.core.behaviours.Behaviour`

DONE

public boolean **done**()
Specified by:
`done` in class `jade.core.behaviours.Behaviour`

II.3 PAA Agent

II.3.1 Class PAA

java.lang.Object
└ jade.core.Agent
 └ **referralanalysis.PAA**

All Implemented Interfaces:
jade.core.TimerListener, java.io.Serializable, java.lang.Runnable

---

public class **PAA**
extends jade.core.Agent
See Also:
Serialized Form

Nested Class Summary	
`class`	**PAA.ReferralNeedIdentifier**

Nested classes/interfaces inherited from class jade.core.Agent
`jade.core.Agent.Interrupted`

Field Summary

Fields inherited from class jade.core.Agent
`AP_ACTIVE, AP_DELETED, AP_IDLE, AP_INITIATED, AP_MAX, AP_MIN, AP_SUSPENDED, AP_WAITING, D_ACTIVE, D_MAX, D_MIN, D_RETIRED, D_SUSPENDED, D_UNKNOWN`

Constructor Summary	
PAA`()`	

Method Summary	
`protected void`	**setup**`()`
`protected void`	**takeDown**`()`

Methods inherited from class jade.core.Agent
`addBehaviour, afterClone, afterMove, beforeClone, beforeMove, blockingReceive, blockingReceive, blockingReceive, blockingReceive, changeStateTo, clean, doActivate, doClone, doDelete, doMove, doSuspend, doTimeOut, doWait, doWait, doWake, getAgentState, getAID, getAMS, getArguments, getContainerController, getContentManager, getCurQueueSize, getDefaultDF, getHap, getHelper, getLocalName, getName, getO2AObject, getProperty, getQueueSize, getState, here, isRestarting, join, notifyChangeBehaviourState, notifyRestarted, postMessage, putBack, putO2AObject, receive, receive, removeBehaviour, removeTimer, restartLater, restore, restoreBufferedState, run, send, setArguments, setEnabledO2ACommunication, setGenerateBehaviourEvents, setQueueSize, waitUntilStarted, write`

Methods inherited from class java.lang.Object
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

PAA

public **PAA**()

Method Detail

SETUP

protected void **setup**()

Overrides:

`setup` in class `jade.core.Agent`

TAKEDOWN

protected void **takeDown**()

Overrides:

`takeDown` in class `jade.core.Agent`

II.3.2 Class PAA.ReferralNeedIdentifier

java.lang.Object
 └ jade.core.behaviours.Behaviour
  └ jade.core.behaviours.SimpleBehaviour
   └ jade.core.behaviours.CyclicBehaviour
    └ **referralanalysis.PAA.ReferralNeedIdentifier**

All Implemented Interfaces:

java.io.Serializable

Enclosing class:

PAA

---

public class **PAA.ReferralNeedIdentifier**
extends jade.core.behaviours.CyclicBehaviour

See Also:

Serialized Form

Nested Class Summary
Nested classes/interfaces inherited from class jade.core.behaviours.Behaviour
`jade.core.behaviours.Behaviour.RunnableChangedEvent`

Field Summary
Fields inherited from class jade.core.behaviours.Behaviour
`myAgent, myEvent, NOTIFY_DOWN, NOTIFY_UP, parent, STATE_BLOCKED, STATE_READY, STATE_RUNNING`

Constructor Summary
PAA.ReferralNeedIdentifier()

Method Summary	
void	action()

Methods inherited from class jade.core.behaviours.CyclicBehaviour
`done`

Methods inherited from class jade.core.behaviours.SimpleBehaviour
`reset`

Methods inherited from class jade.core.behaviours.Behaviour
`actionWrapper, block, block, getBehaviourName, getDataStore, getExecutionState, getParent, handle, isRunnable, onEnd, onStart, restart, root, setAgent, setBehaviourName, setDataStore, setExecutionState`

Methods inherited from class java.lang.Object
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

PAA.REFERRALNEEDIDENTIFIER

public **PAA.ReferralNeedIdentifier**()

Method Detail

ACTION

public void **action**()

Specified by:

`action` in class `jade.core.behaviours.Behaviour`

II.4 SAA Agent

II.4.1 Class SAA

java.lang.Object

└ jade.core.Agent
  └ **referralanalysis.SAA**

All Implemented Interfaces:

jade.core.TimerListener, java.io.Serializable, java.lang.Runnable

---

public class **SAA**
extends jade.core.Agent

See Also:

Serialized Form

Nested Class Summary	
Class	**SAA.ServiceIdentifier**

Nested classes/interfaces inherited from class jade.core.Agent
jade.core.Agent.Interrupted

Field Summary

Fields inherited from class jade.core.Agent
AP_ACTIVE, AP_DELETED, AP_IDLE, AP_INITIATED, AP_MAX, AP_MIN, AP_SUSPENDED, AP_WAITING, D_ACTIVE, D_MAX, D_MIN, D_RETIRED, D_SUSPENDED, D_UNKNOWN

Constructor Summary	
SAA()	

Method Summary	
protected void	**setup**()
protected void	**takeDown**()

Methods inherited from class jade.core.Agent
addBehaviour, afterClone, afterMove, beforeClone, beforeMove, blockingReceive, blockingReceive, blockingReceive, blockingReceive, changeStateTo, clean, doActivate, doClone, doDelete, doMove, doSuspend, doTimeOut, doWait, doWait, doWake, getAgentState, getAID, getAMS, getArguments, getContainerController, getContentManager, getCurQueueSize, getDefaultDF, getHap, getHelper, getLocalName, getName, getO2AObject, getProperty, getQueueSize, getState, here, isRestarting, join, notifyChangeBehaviourState, notifyRestarted, postMessage, putBack, putO2AObject, receive, receive, removeBehaviour, removeTimer, restartLater, restore, restoreBufferedState, run, send, setArguments, setEnabledO2ACommunication, setGenerateBehaviourEvents, setQueueSize, waitUntilStarted, write

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

SAA

public **SAA**()

Method Detail

SETUP

protected void **setup**()

Overrides:

`setup` in class `jade.core.Agent`

TAKEDOWN

protected void **takeDown**()

Overrides:

`takeDown` in class `jade.core.Agent`

II.4.2 Class SAA.ServiceIdentifier

java.lang.Object
└ jade.core.behaviours.Behaviour
  └ jade.core.behaviours.SimpleBehaviour
    └ jade.core.behaviours.CyclicBehaviour
      └ **referralanalysis.SAA.ServiceIdentifier**

All Implemented Interfaces:

java.io.Serializable

Enclosing class:

SAA

---

public class **SAA.ServiceIdentifier**
extends jade.core.behaviours.CyclicBehaviour

See Also:

Serialized Form

Nested Class Summary
Nested classes/interfaces inherited from class jade.core.behaviours.Behaviour
`jade.core.behaviours.Behaviour.RunnableChangedEvent`

Field Summary
Fields inherited from class jade.core.behaviours.Behaviour
`myAgent, myEvent, NOTIFY_DOWN, NOTIFY_UP, parent, STATE_BLOCKED, STATE_READY, STATE_RUNNING`

Constructor Summary	
SAA.ServiceIdentifier`()`	

Method Summary	
`void`	**action**`()`

Methods inherited from class jade.core.behaviours.CyclicBehaviour
`Done`

Methods inherited from class jade.core.behaviours.SimpleBehaviour
`Reset`

Methods inherited from class jade.core.behaviours.Behaviour
`actionWrapper, block, block, getBehaviourName, getDataStore, getExecutionState, getParent, handle, isRunnable, onEnd, onStart, restart, root, setAgent, setBehaviourName, setDataStore, setExecutionState`

Methods inherited from class java.lang.Object
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

SAA.SERVICEIDENTIFIER

public **SAA.ServiceIdentifier**()

Method Detail

ACTION

public void **action**()

Specified by:

`action` in class `jade.core.behaviours.Behaviour`

II.5 MCA Agent

II.5.1 Class MCA

java.lang.Object
└ jade.core.Agent
  └ **referralanalysis.MCA**

All Implemented Interfaces:

jade.core.TimerListener, java.io.Serializable, java.lang.Runnable

---

public class **MCA**
extends jade.core.Agent

See Also:

Serialized Form

Nested Class Summary	
`class`	**MCA.FollowupTreatmentPlanIdentifier**

Nested classes/interfaces inherited from class jade.core.Agent
`jade.core.Agent.Interrupted`

Field Summary

Fields inherited from class jade.core.Agent

AP_ACTIVE, AP_DELETED, AP_IDLE, AP_INITIATED, AP_MAX, AP_MIN, AP_SUSPENDED, AP_WAITING, D_ACTIVE, D_MAX, D_MIN, D_RETIRED, D_SUSPENDED, D_UNKNOWN

Constructor Summary	
MCA()	

Method Summary	
protected void	**setup**()
protected void	**takeDown**()

Methods inherited from class jade.core.Agent
addBehaviour, afterClone, afterMove, beforeClone, beforeMove, blockingReceive, blockingReceive, blockingReceive, blockingReceive, changeStateTo, clean, doActivate, doClone, doDelete, doMove, doSuspend, doTimeOut, doWait, doWait, doWake, getAgentState, getAID, getAMS, getArguments, getContainerController, getContentManager, getCurQueueSize, getDefaultDF, getHap, getHelper, getLocalName, getName, getO2AObject, getProperty, getQueueSize, getState, here, isRestarting, join, notifyChangeBehaviourState, notifyRestarted, postMessage, putBack, putO2AObject, receive, receive, removeBehaviour, removeTimer, restartLater, restore, restoreBufferedState, run, send, setArguments, setEnabledO2ACommunication, setGenerateBehaviourEvents, setQueueSize, waitUntilStarted, write

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

MCA

public **MCA**()

Method Detail

SETUP

protected void **setup**()

Overrides:

`setup` in class `jade.core.Agent`

TAKEDOWN

protected void **takeDown**()

Overrides:

`takeDown` in class `jade.core.Agent`

II.5.2 Class MCA.FollowupTreatmentPlanIdentifier

java.lang.Object
└ jade.core.behaviours.Behaviour
  └ jade.core.behaviours.SimpleBehaviour
    └ jade.core.behaviours.CyclicBehaviour
      └ **referralanalysis.MCA.FollowupTreatmentPlanIdentifier**

All Implemented Interfaces:
java.io.Serializable

Enclosing class:
MCA

---

public class **MCA.FollowupTreatmentPlanIdentifier**
extends jade.core.behaviours.CyclicBehaviour

See Also:
Serialized Form

Nested Class Summary
Nested classes/interfaces inherited from class jade.core.behaviours.Behaviour
`jade.core.behaviours.Behaviour.RunnableChangedEvent`

Field Summary
Fields inherited from class jade.core.behaviours.Behaviour
`myAgent, myEvent, NOTIFY_DOWN, NOTIFY_UP, parent, STATE_BLOCKED, STATE_READY, STATE_RUNNING`

Constructor Summary
MCA.FollowupTreatmentPlanIdentifier`()`

Method Summary	
`Void`	**action**`()`

Methods inherited from class jade.core.behaviours.CyclicBehaviour
`Done`

Methods inherited from class jade.core.behaviours.SimpleBehaviour
`Reset`

Methods inherited from class jade.core.behaviours.Behaviour
`actionWrapper, block, block, getBehaviourName, getDataStore, getExecutionState, getParent, handle, isRunnable, onEnd, onStart, restart, root, setAgent, setBehaviourName, setDataStore, setExecutionState`

Methods inherited from class java.lang.Object
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

MCA.FOLLOWUPTREATMENTPLANIDENTIFIER

public **MCA.FollowupTreatmentPlanIdentifier**()

Method Detail

ACTION

public void **action**()

Specified by:

`action` in class `jade.core.behaviours.Behaviour`

DECLARATION

The thesis is my original work and has not been presented for a degree in any other university, and all sources of material used for the thesis have been duly acknowledged.

Tsegaye Ayele