

An Introductory of a Content Provider Agent in Higher Learning Institutions

B.A.Jesica Malani¹ and Abu Bakar Md Sultan²

¹ Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
43400, UPM Serdang, Selangor Malaysia

² Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
43400, UPM Serdang, Selangor Malaysia

Abstract

Software agents, nowadays has become most favorable technology in developing computer systems. Software agent can be used to quickly and easily build integrated agent-based systems where agents can be applied in all domains of interest and one of them is in education. In the higher learning institution, lecturers are optimize with workloads such as lecturing, lecture preparation, student consultation and lecture content/module preparation etc. Looking at the current ways of materials preparation done by lecturers, the process might take hours, days and even months in order to prepare impressive teaching materials for students. In order to minimize the workload, therefore the author had proposed and developed a content provider agent (CPA) in order to assist lecturer in preparing teaching contents as requested. Evaluation is done by comparing the percentage of effective search process done by the CPA and the search done via search engines. The result is promising where the proposed agent satisfy the objectives of the CPA developed.

Keywords: *Software agents, CPA, Agent-based System, Agent technologies, Prometheus.*

1. Introduction

According to [3], computer systems in the twenty-first century are dramatically different from those that have been in the norm over the last 50 years. Unlike previous changes, which have been seen improvements in computing capacity and power, modern computing is defined by the interconnection of computers and all that it brings. The World Wide Web offers radically new intake on the information society and releasing information making it available to all. It also provided the basic infrastructure for the dynamic provision of online services.

The key difference now is that there is an environment made up of computers in the infrastructure in support systems and embedded in a vast array of devices as well as on the traditional desktop [10]. More importantly these computers are typically networked and can interact dynamically to form new configurations of systems to suit current needs. One of the latest approaches that can be used to tackle the emergent problems and to manage complexity that arises is the agent technology.

An agent or agent-based system is a software based computer system that has several properties such as: autonomous, social ability, reactivity, goal-directed, mobility and collaboration [9]. Agents are used to make a user's tasks easier. Every agent can perform its programmed tasks. Agents also can communicate with other users as well as other agents in a network. Agents are also helpful in exchanging data in cross-platform systems.

Agents, one of the most exciting new developments in computer software technology, can be used to quickly and easily build integrated enterprise systems. Famous domains where software agent can be applied such as in e-business, e-commerce, medical and mobile computing as well as any other domains.

In the above-mentioned domains, the natural next step is to use multiple software agents that communicate and cooperate with each other to solve complex problems and implement complex systems. Software agents provide a powerful new method for implementing these next-generation multi-agent systems. However for this study, the author narrowed the scope only to develop one agent, the Content Provider Agent (CPA) and the agent is constructed using Java.

1.1 Information Filtering and gathering

In the age of computers, information is readily available on the Internet, whether they are useful or useless. There is so much data available that we often claim to be "overloaded with information". Having too much data can cause as much trouble as having no data, as we must shift through so much information to get what we need [2]. We can categorize this information overload problem into two divisions:

- **Information filtering:** We must go through an enormous amount of information to find the small portion that is relevant to us.
- **Information gathering:** There is not enough information available to us and we have to search long and hard to find what we need.

1.2 Software Agents

An agent is 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 [1].

As defined above, agent should be autonomous, meaning that it not simply perform searching but agent can plan, arrange, compare, meaning that can carry out arrangements of all sorts that would normally be done by human user. There is a minimum set of common features that typify a software agent. The features are as mentioned:

- A software agent is **autonomous**; the agent is capable of operating as a standalone process and performing actions without user intervention.
- A software agent is **communicative**; it communicates with the user, other software agents, or other software processes.
- A software agent is **perceptive**; it is able to perceive and respond to changes in its environment.

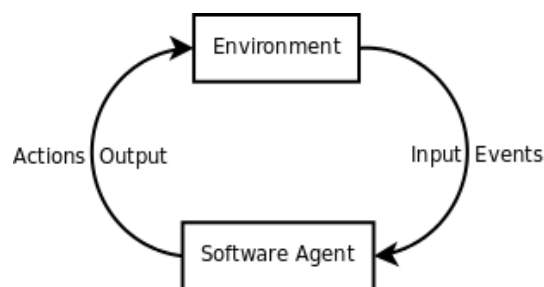


Fig. 1 Agent Environment

Figure 1 show an agent and its environment which consist of percepts or also known as events while actions known as output. The agent takes sensory input from the environment, and produces as output actions that affect it. The interaction is usually an ongoing non-terminating one [1].

Agents are meant to be autonomous meaning that it is automatically initiated without the intervention of human user. An agent also helps the user in tasks being programmed for them to be done [8]. Instead of user-initiated interaction via commands and/or direct manipulation, the user is engaged in a cooperative process in which human and computer agents both initiate communication, monitor events and perform tasks.

The rest of the paper is organized as follows. Section 2 relates to the study of the problem domain which covers why the study is proposed. Next, in section 3, it includes a brief presentation of the related work in the field of adopting software agents in legacy system in higher learning institutes and other domains too. Section 4 presents the design, architecture and its functionalities of the proposed agent. In section 5, example and promising result of the develop agent is discussed. Finally, section 6 summarizes the conclusion and suggests future plans for improvements and enhancements of the proposed agent.

2. Problem Statement

In a higher learning institution, a lecturer is required a minimal of Master's degree from a recognized university in his/her field of specialization. According to [7], the normal workload of lecturer shall include teaching; research, scholarly and creative activities; and service to the university in proportions of approximately 40%, 40% and 20% respectively of each lecturer's time.

From this statement, it is clearly shown that lecturer's time are fully utilize not just in educating but they also carries big responsibilities in servicing their higher institution as shown in figure 2.

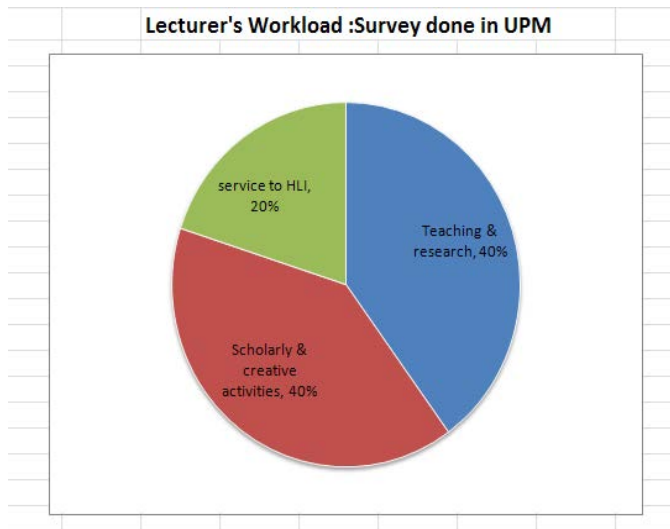


Fig. 2: Lecturer's portions of workload, as in [7].

The general definition of workload is the amount of tasks that a person needs to complete. In the context of this study, workload is also defined as all activities related to lecturer's professional responsibilities and interest.

2.1. Teaching Workload

A lecturer's responsibilities are mentioned in this subsection. From figure 2 above, the authors conclude that a lecturer's time is fully utilized for each working day. A large portion of the time is contributed for teaching. Teaching workload includes all related teaching activities such as material preparation and actual teaching. Both of these activities are influenced by factors such as number of students in the course, contact hours per week, level of the course and method of delivering the lecture. This strongly agrees that lecturers are overload with workloads and this can reduce their performance. Therefore the authors proved the necessity to propose and developed a content provider agent (CPA) in order to assist lecturer in preparing teaching contents as requested and at once minimize their workload.

3. Related Works

Utilization of agents is significant for providing intelligence in existing environment. Thus a lot of work has been done concerning these fields mainly focusing on adoption of agents to integrate legacy systems. According to [4], the need of users to have their search results effectively filtered is increasing important where the volume of information on the Internet growing at an

exponential rate. Therefore in the above mentioned paper, the authors examine how a tree threshold function can be used in an information filtering agent (IFA) to extend the original keyword search to cover other related words within the domain, creating a keyword weighted semantic tree. In order to recommend useful learning material to students, Andronico et al. [4] proposed a multi agent recommendation system that suggests educational resources to students into a mobile learning platform that supports mobile learning processes. The authors extended the Learning Management System (LMS) in order to incorporate mobile technologies, allowing users to interact with the systems using mobile devices.

According to [5], the intelligent agent community has been used in the system to help learners finding out the adapting courses and learning path. The features of the systems include analyzing the cause of learning inefficiency, promoting learners learning efficiency by personalized courses and learning paths through the information analyzed by agents, therefore learners will spend less time in making teaching materials for teachers.

4. The Content Provider Agent (CPA)

4.1 Defining CPA

This section introduces the Content Provider Agent (CPA) to assist lecturers (user) in preparing their teaching modules. The agent is programmed to filter information based on the request being entered.

The CPA should be able to monitor and analyze the specified requirement of the lecturer in the environment, by which using its knowledge gathering abilities could independently make suggestion to the preparation of teaching materials. The CPA is responsible to match requests made by lecturer. The request is set by the lecturer for example notes preparation on topic such as 'Introduction to computers'. The request is then sent for next step.

Lecturers are optimized with workloads such as lecturing, lecture preparation, student consultation and lecture content/module preparation etc. In order to minimize the workload, the Content Provider Agent is proposed.

The purpose of the paper is to propose a content provider agent for filtering information required by user. Besides that to help user in preparing contents of their interest, the CPA will provide content based on best search and the

result of the search should only resemble what the user searched for. To verify this claim, the author compares the performance with and without the CPA component.

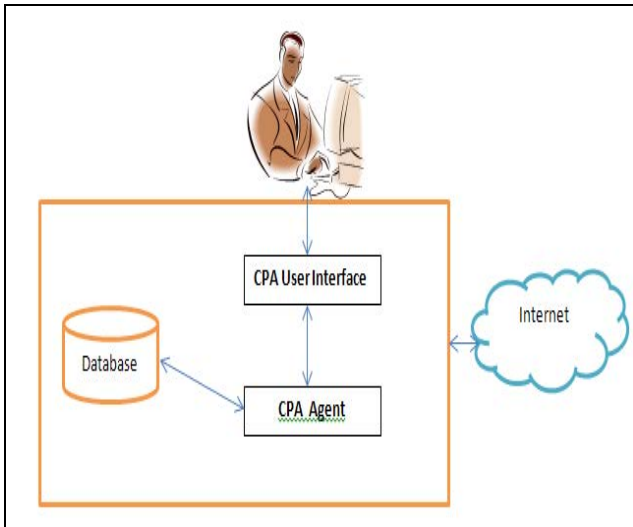


Fig. 3: Content Provider Agent structure

4.2 Research Methodology

4.2.1 Introduction to Prometheus

Referring back to the main purpose of this study, the functional requirements of the CPA should be identified. Therefore a detailed analysis is done in order to collect the functional requirements. The authors use the Prometheus Methodology to design and develop the CPA. Prometheus is best used for agent based system which means in creating a proactive system. This is due to its characteristic where it implies an iterative approach. Iterative allows the design and development of an agent as a whole without needing to integrate with the design of other agent in the system. Therefore for this study, focus only is on the CPA. Although this study is on the CPA, but it is important to follow the three main stages of Prometheus methodology which is the system specification stage, architectural design phase and the detailed design phase.

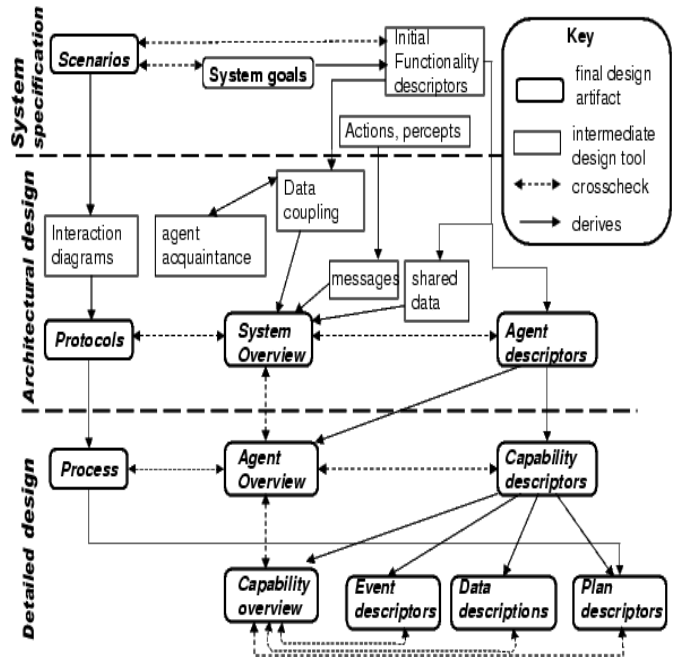


Fig. 4: Prometheus Methodology stages

4.2.2 System Specification Phase

Figure 4 shows three main stages of the Prometheus methodology. The first stage is the system specification which includes the scenarios, goals, functionalities, actions and percepts. Below, in figure 5 is the goal overview diagram of the study. The focus is on the CPA goal where the author will develop the CPA based on this design.

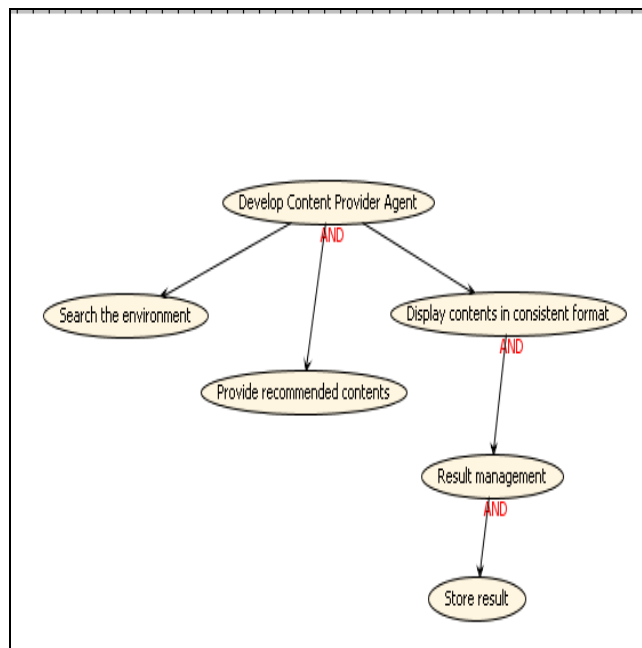


Fig. 5: Goal view diagram of CPA

Figure 5 shows the goals of the CPA. The main goal is to develop a Content Provider Agent. This CPA's main goal can be refined into several sub goals. Three sub goals of the CPA are :

- i. Search the environment
- ii. Provide recommended contents
- iii. Display contents in consistent format

While the 'display contents in consistent format' also can be refined into 'result management'. Lastly is to store the result.

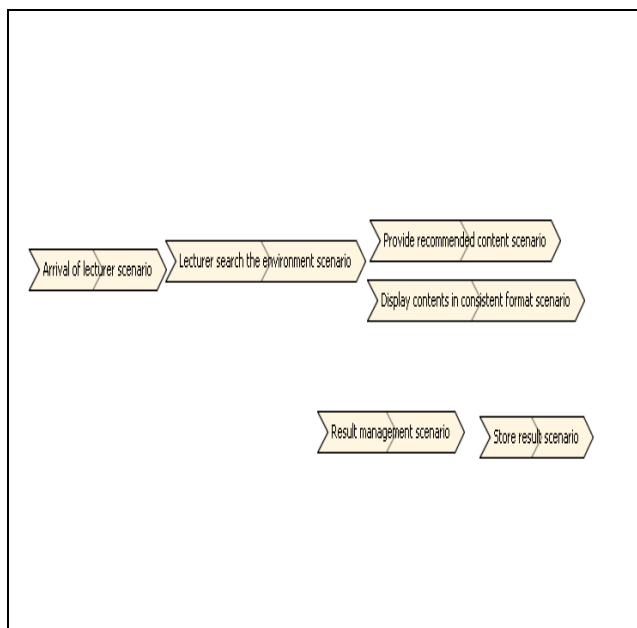


Fig 6: Scenarios Diagram

In figure 6 above, it is illustrated that the CPA consists of the following scenarios. There are five scenarios starting from the 'arrival of lecturer' scenario to the 'lecturer search the environment' scenario, where the lecturer will search for content of his/her teaching materials. The next scenario is the 'provide recommended content' scenario and the 'display content of search result in a consistent format' scenario. The content of the search result should be displayed to the lecturer. From this display, lecturer will be able to view the content and select only the content he/she requested for.

Table 1: Lecturer perform searching

Name	Search process by lecturer					
Description	Lecturer enter requested keyword to be search by CPA					
Priority	Not specified					
Actors	-					
Initiated by	system					
Trigger	Recommends content to lecturer					
Steps	#	Type	Name	Description	Data used	Data produced
	1	Goal	Search for content	The goal is to search the keyword entered by lecturer	-	-
	2	Percept	Content search	This is the input which is entered by lecturer i.e content needed.		
	3	Scenario	Provide recommended content	The recommended content will be sent to lecturer via GUI		Result of search process by CPA
	4	Scenario	Display content in consistent format	Content will be displayed to lecturer		Display the content for lecturer to choose.
	5	Goal	Store result	Result stored in DB		
Variation	-					

Table 1 shows the main scenario in the CPA which is the 'search environment' scenario. The table summarizes the whole description of the scenario on how it performs its function until it achieves its goal.

Figure 7 illustrates the system role diagram of the CPA. Once request has been made, the CPA will be invoked. Search process begins and recommended content will be collected and it will be displayed to lecturer. Result of the search process will be managed by the 'result management' scenario and will be stored.

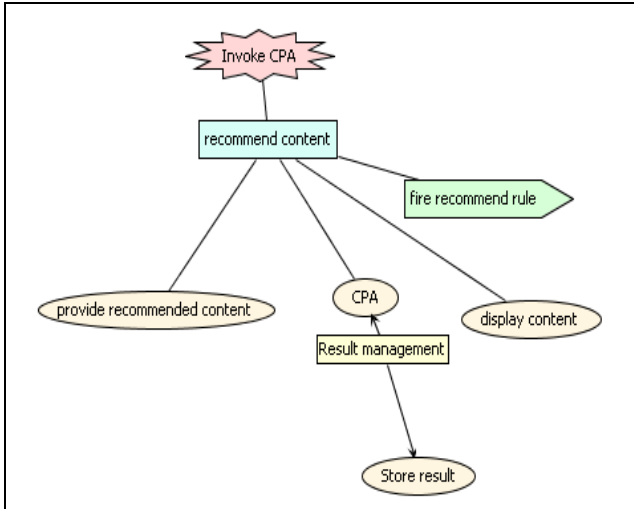


Fig 7: System role diagram

The second stage is the architectural design stage where the agent architecture will be identified. This stage includes the interaction diagrams, agent acquaintance, data coupling, messages, shared data, protocols and system overview and the agent descriptors. Figure 8, 9 and 10 shows the agent architecture and functionalities.

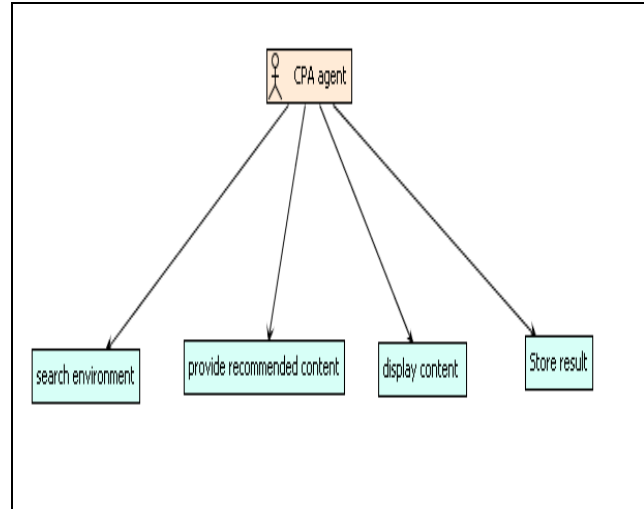


Fig 9: CPA Agent Role Diagram

The third stage is the detailed stage which includes process agent overview, agent descriptors, data descriptors and plan descriptors. Overall the advantage of using this methodology to design and develop the CPA is that the agent can perform autonomously without human intervention because during the design phase, the functional requirements of the CPA agent had been identified therefore it is easier to convert it into codes.

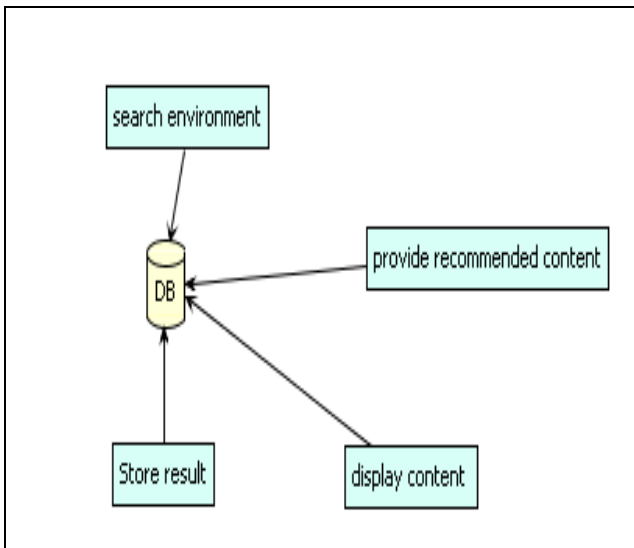


Fig 8: Data coupling diagram

In this phase, the authors need to identify what types of data need to be stored in the system as beliefs. By carefully checking the functionalities, it can be seen that there is a need to keep the information of all request sent. The detailed capabilities are being discussed in design phase.

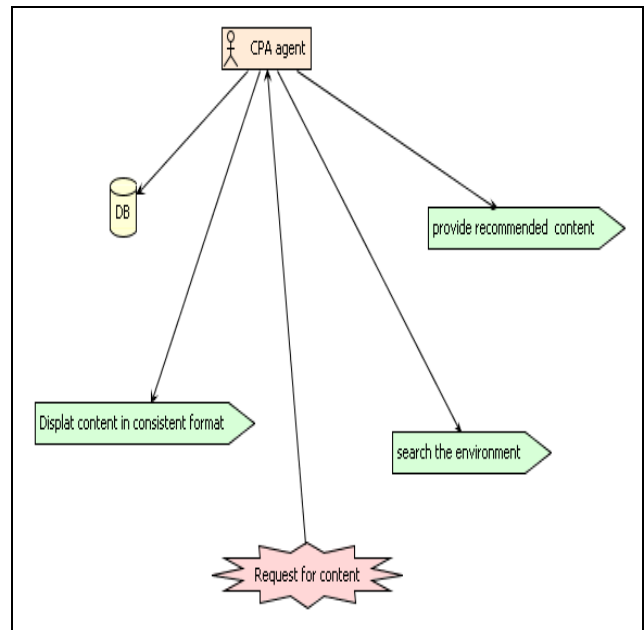


Fig 10: CPA overview diagram

In order to visualize the system using the Prometheus methodology, the Prometheus Design Tool is used. The Prometheus Design Tool (PDT) supports the structured design of intelligent agent systems. It supports the Prometheus methodology, but can also be used more generally.

5. Result and Discussion

Upon completing this study, promising result gained. In order to test the CPA, the authors have developed an independent module and performed some test. Two evaluation tests have been performed. The first test is based on a survey to get lecturer’s feedback on the implementation of the CPA agent and the second evaluation was to compare the percentage of content accuracy with and without the CPA agent. The result is summarized in table 2 below.

Table 2: Questionnaire given for lecturer’s survey

Tick your selected value for each question 1 to 6					
Question	NA	1	2	3	4
Q1. Do you think the CPA will help you in you daily task?					
Q2. Rate how the CPA reduces your workload while preparing contents for your lecture.					
Q3. Do you agree that the content provided by the CPA fulfills your needs?					
Q4. Is the CPA better than conventional search engines?					
Q5. Do you agree to integrate the CPA into the current environment?					

NA= “Not answered “, 1=”Poor”, 2=”Below average”, 3=”Good” and 4=”Excellent”

Table 3: Result of survey- Test 1

Q	Number of responses				
	Not Answered	Poor	Below average	Good	Excellent
Q1	2	1	1	18	8
Q2	0	0	1	19	10
Q3	0	2	2	15	11
Q4	0	0	0	12	18
Q5	0	0	4	19	7

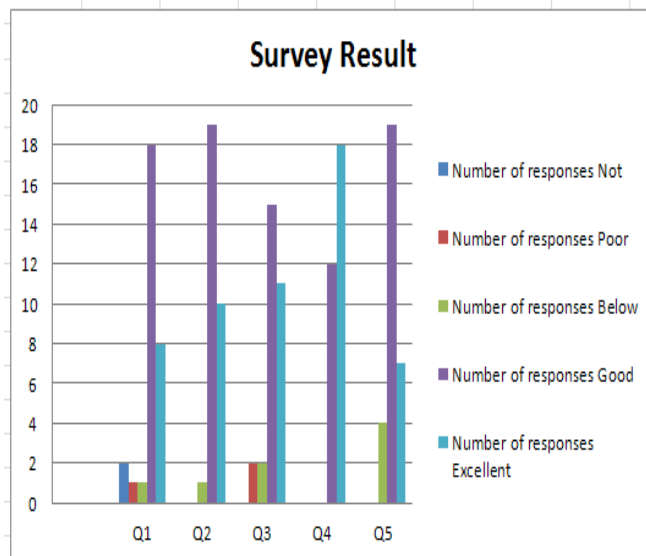


Fig 11: Graph of survey result

From the above survey result, the authors summarizes that 86.6% of the respondent agrees that the CPA will help them in their daily tasks and 96.6% respondents says that the CPA reduces their workload. Other than that, 86.6% respondents agreed that the content provided by CPA fulfills their requested needs. The result also shows that all respondents agreed that the CPA is better than the conventional search engines because contents are prepared based on their request. Finally, 86.6% of the respondents agreed to integrate the CPA into the current environment.

For the second form of evaluation, the developed CPA is being tested. Evaluation is done by comparing the percentage of effective search process done by the CPA and the search done via search engines. The result is shown in figure 12, 13 and figure 14.

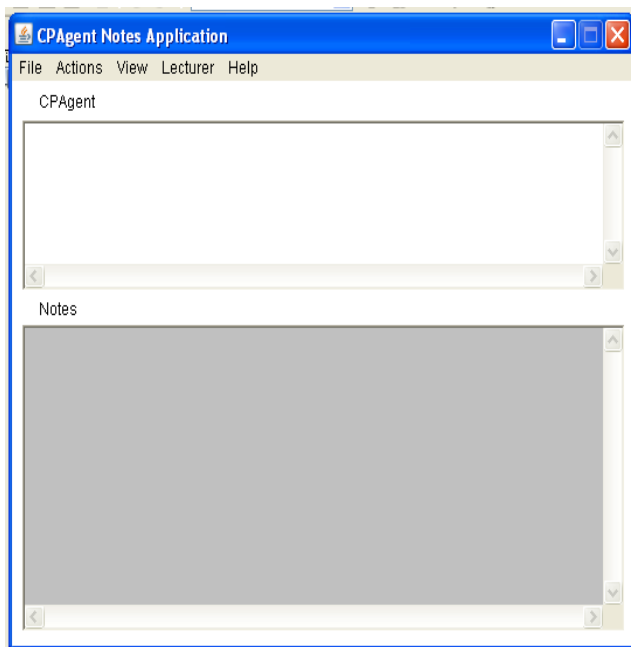


Fig 12: CPA interface

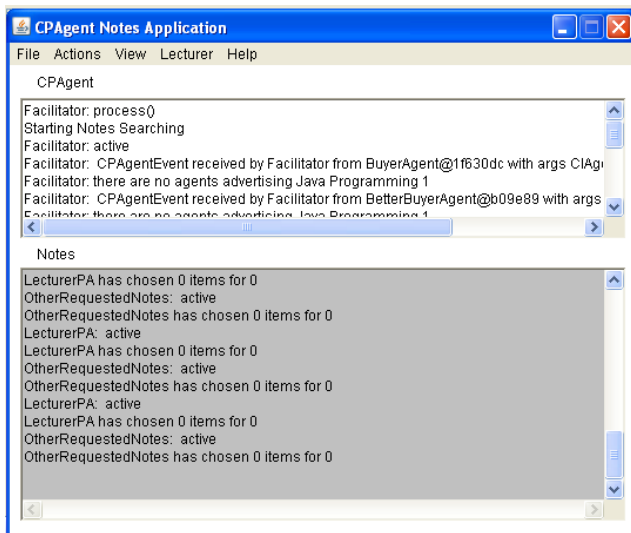


Fig 13: Running mode after request sent

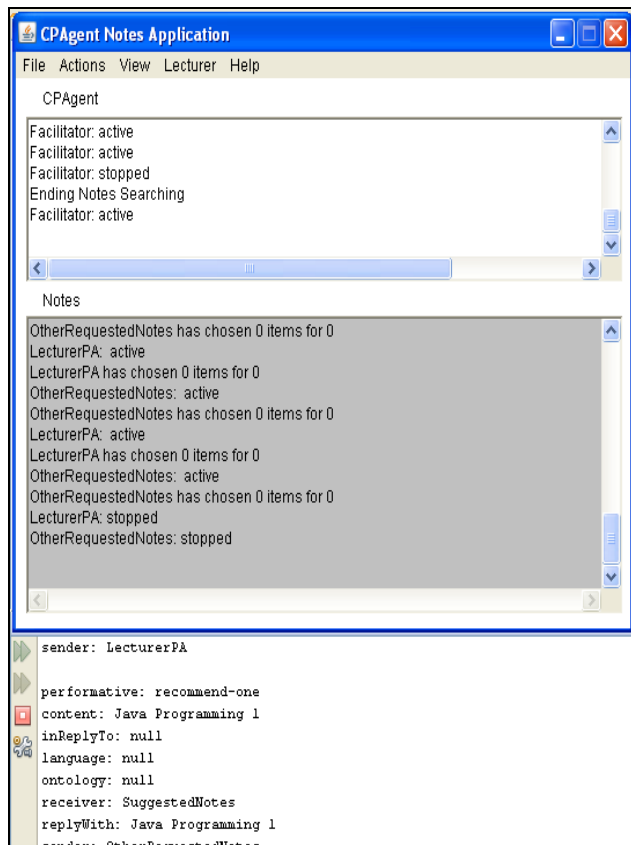


Fig 14: Stopped mode and summary

6. Conclusion

This paper presented a Content Provider Agent in the higher learning institution. The CPA agent which is constructed using Java Netbeans has been developed in order to meet the objectives of this study. Prometheus methodology and PDT tool has been used in the analysis and design of the CPA agent. However, future work has to be done in order to improve the current agent for meeting exactly its desired goals. Instead of using Java, it is suggested the development of the agent system is done via the JADE platform.

Acknowledgments

The author would like to thank Universiti Putra Malaysia for providing all needed resources in completing this study.

References

- [1] Micheal Wooldridge, Introduction to MultiAgent Systems, England: 2002.
- [2] Pattie Maes. "Agents that Reduce Work and Information Overload".
- [3] Michael Luck, Ronald Ahri, Mark D’Inervo, Agent-based Software Development.
- [4] A. Andronico, A. Carbonaro, G.Casadei, L. Colazzo, A. Molinari and M. Ronchetti, “ Integrating a multi-agent recommendation system into a Mobile Learning Managemnet System” in Proceedings of Artificial Intelligence in Mobile Syssem 2003(AIMS2003), October 12, Seattle, USA.
- [5] Suraya Masrom and Abdullah Sani, “An Adaptation of Agent Based Computer-Assisted Assessment into E-learning Environment” ,International Journal of Education and Information Technologies, Vol 3, Issue 3, 2009,pp163-170.
- [6]Hani Mahdi and Sally S. Attia. “Developing and Implementing a Multi-Agent System for Collaborative E-learning, E-learning”, Marina Buzzi (Ed.), 2010,ISBN: 978-953-7619-95-4, INTECH.
- [7] Ruhil Hayati Hashim, Jamaliah Abdul Hamid, Mohd Hassan Selamat, Hamidah Ibrahim, Rusli Abdullah, M ohd Ghazali Mohayidin, “Using Algorithmic Taxonomy to Evaluate Lecturer Workload”, Journal of Knowledge Management Practice, Vol 7, No.2, June 2006.
- [8]Zakrzewska D. “Building Group Recommendations in E-Learning Systems”, P.Jedrezejowicz et al. (Eds):KES-AMSTA 2010,2010, Part I, LNAI 6070, pp. 391-400. Springer, Heidelberg.
- [9] James Adridge. “Knowledge on Demand: An application of Software Agents & E-learning”,2002.
- [10]Agustin Casamayor, Analia Amandi, Marcelo Campo. “Intelligent assistance for teachers in collaborative e-learning environments.” Computers & Education,2009, 52(2009), 1147-1154.

B.A.Jesica Malani Sugathapala is currently a postgraduate student in Universiti Putra Malaysia. She has obtained her B.Sc. in Computer Science from Universiti Teknologi Malaysia. Currently she is in her final semester of Masters of Computer Science program specializing in Software Engineering.

Abu Bakar Md Sultan is currently a lecturer in the Faculty of Computer Science & Information Technology, Universiti Putra Malaysia(UPM). He obtained his PhD from UPM. His research interest includes Software Engineering and Artificial Intelligence.