Journal of Theoretical and Applied Information Technology
10th March 2015. Vol.73 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AN AUTONOMOUS SOFTWARE APPROACH TO ENHANCE INFORMATION SHARING IN UNIVERSITY COURSE TIMETABLE PLANNING

[1]NAJLAA ATEEQ MOHAMMED DRAIB, [2]ABU BAKAR MD SULTAN
[1,2] Faculty of Computer Science and Information Technology, Universiti Putra Malaysia
E-mail: [1]n.draib@gmail.com , [2]abakar@upm.edu.my

ABSTRACT

One of the major activities of university departments at the beginning of an academic semester is creating a course timetable. During course timetable creation, a department usually needs to book one or more courses from other departments. In order to book courses, a department needs to send request to another department that offers that course and exchange relevant information with it. This information is essential for the department in order to assign the academic provider resource to appropriate timeslot in its own courses timetable which has to satisfy specific conditions. Information sharing during timetable planning in academic departments still faces difficulties due to the low level of cross-department information sharing. These issues seriously restrict and delay the process of collaborative timetabling planning. In order to automate the information sharing between academic providers in timetabling planning we present a prototype of an autonomous and efficient information sharing tool. The aim of this tool is to reduce communication gaps among the departments. The proposed approach is applied on timetable planning for the department of Computer Science (CS) and Software Engineering (SE) at the Universiti of Putra Malaysia (UPM)

Keywords: *Information Sharing, University Course Timetabling, Sharing Academic Resources, Autonomous Software, Web-Based Decision Making Supporting Tool*

1. INTRODUCTION

University course timetable is one of the major administrative activities which consists of scheduling a set of given resources (lecturers, students, classrooms) to objects (courses) in a cyclic period of time satisfying a number of university constraints. University course timetabling is a challenging problem faced by educational institutions of many types. It is considered by researchers as a NP-Hard combinatorial optimization problem that does not have analytical solution methods.

The problem of generating course timetable has been tackled through various approaches. The suggested approaches have been proposed either to address course scheduling through techniques that autonomously generate course timetable with considering number of constraints [7,8,14,15] or by providing systems/tools to support schedulers in decision making during course timetabling [9,10,11,12,13]. However, almost no one of these studies has been implemented to enhance information sharing in course timetable planning or particularly stress on the importance of it. In this study, we improve this shortcoming by discussing the importance of information sharing during course timetabling and providing a tool to automate information sharing in course timetable planning.

Generally, timetabling problems have been classified into three main classes: course timetabling, examination timetabling, and school timetabling [1]. The focus of this paper is on sharing information during solving course timetabling problem. Timetabling of courses needs to be conducted with several slots and with various categories such as tutorials, lecturers and

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

lab/particular sessions, which fits within a week and repeats for the whole semester.

Universities in which students are divided into groups, courses for each group of students are predefined and assigned to academic semesters. In this case, choosing a course is not important and timetabling is almost repeated for every academic year with minor adjustments if needed. Scheduling timetable is more difficult in developed course choosing, where each student can choose a number of relevant courses from his department and outside departments and faculties as well. In such universities course scheduling is changed for each academic semester.

Typically, during course timetable planning, university's departments potentially need to request one or more courses from other departments at the same campus. In this case, there is a significant need of such information sharing system for exchanging information between these departments so that they use each other's data for making decisions and creating their own scheduling.

The Software Engineering department in the Faculty of Computer Science and Information Technology at Universiti Putra Malaysia still uses a manual process for scheduling courses. It is not able to exchange relevant information automatically with the other departments. At present, the information sharing takes place verbally among the heads of departments, schedulers, or persons that create the course timetable.

For the sake of improving information asymmetry in timetabling planning at universities and keeping information sharing efficiently communicated among university departments and faculties, current manual system of information sharing in the Faculty of Computer Science and Information Technology must be changed into automatic information sharing system that will make information symmetrical to each department.

This paper presented a prototype called (Timetabling Decision Making Supporting Tool (TDMST)) for sharing information among the involved departments in course timetable planning. This tool is web-based tool that enhances information sharing and cooperation between university departments by automating the information flow for the process of building course timetable. The proposed tool is smart enough to give suggestions, help in decision making, and automatically send feedback between the departments.

TDMST was applied in course timetable planning for the Faculty of Computer Science and Information Technology at Universiti Putra Malaysia. It was used at the department level to replace the existing manual system of exchanging information between involved departments in course scheduling. The implementation of the tool has generated a series of benefits due to the automation of sharing information process including a reduction of communication gaps among the departments during course timetable preparation and a reduction of time and efforts spent in course timetabling.

This paper is organized into eight sections. Introduction to course timetabling is in section 2. Section 3 is discussing the importance of information sharing. Related work is discussed in section 4. Section 5 sheds light on the development process of the proposed solution. Section 6, 7 and 8 discuss the result, limitations and the conclusion respectively.

2. INFORMATION SHARING

Sharing resources has a significant importance in public and private sectors due to its advantages to cut cost (time, money, and efforts) required to achieve goals. Information is one of the most important resources to be shared. For public government organizations, sharing information across organizational boundaries is an essential factor to accomplishing public benefits such as increased productivity, improved policymaking and integrated public services[2].

In academic environment, whenever resources are usable and sharable, there is a desire by the different departments to benefit from sharing or /and exchanging these resources either for their selfish interest or to be helpful with the other departments in the same campus. Typically, the process of sharing or exchanging resources includes sharing information about these resources.

Timetabling potentially involves scheduling courses shared from other departments. Indeed, requesting or providing these courses cannot be achieved without sharing/exchanging related information.

Information sharing has been defined as an act of exchanging information among community members who are in need of those information [3]. Therefore, information sharing may involves multiple parties who share information of common interest.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Those information are in the form of suggestions, opinions and answers to questions [4]. Basically, automated information sharing decreases paperwork burden, work processes and enhances "the formulation, implementation, and evaluation of policy" [5].

Management staff deliberately share information related to work and key developments and activities [6].

Although the issue of information sharing and information sharing systems are discussed a lot in literature, there are very few scholars who have particularly discussed timetabling information sharing system. Therefore, it is hard to find such papers in this research field.

In this paper we discuss the importance of information sharing in course timetabling and we propose a prototype to enhance exchanging information between university departments.

3. THE PROBLEM OF TIMETABLING PLANNING OF FSKTM

3.1. FSKTM (Faculty of Computer Science and Information Technology)

The Faculty of Computer Science and Information Technology at Universiti of Putra Malaysia (UPM) consists of four departments, Department of Computer Science, Department of Multimedia, Department of Computer Network, and Department of Software Engineering and Information Systems. These departments offer a four years program for PhD degree, a two years program for a master degree, and a four years program for bachelor. Timetabling problem is to be solved every academic semester based on the previous year's timetable; in another word the historical data. The historical data contains the offered courses associated with the experts group (lecturers who taught the course) for each course. In case the course is new, the selection of the lecturer will be based on either the collected information about lecturers' wishes, or based on the specializations of the lecturers. The curriculum structure in FSKTM consists of three classes of courses, university courses, core courses, and elective courses. University courses component must contain courses offered to university students to complete them with communication skills and business management. Core courses is a combination of various courses in the department specialization. Core courses usually contain a number of courses offered from outside the department (other departments). Elective courses can be selected by the student and offered to support the core courses. In this work we focus more on core courses, as they contain some courses offered by outside departments.

3.2. The Process of Timetabling

In FSKTM's departments, the process of course timetabling can be divided into two stages, schedule courses offered by the department itself and scheduling the courses that must be requested from the other departments. The order of performing these stages is not important. In addition to scheduling of certain number of its own courses and satisfying specific conditions, one university department usually needs to schedule one or more resources (lecturer(s)/room(s)) from other university departments. To be able to schedule these courses, an academic department needs first to send request to another department that offers that resource and exchange relevant information with it. This information is essential for the academic department in order to assign the requested courses to appropriate timeslot in its own courses timetable according to its own constraints.

The information sharing during building timetables in UPM's academic departments still face difficulties in timetable planning due to the lack of information sharing and low level of cross-department information sharing. These difficulties restrict and delay the process of timetabling. Figure 1, illustrates the manual procedure of sharing lecturer between two departments. In this paper, academic consumer represents the department that sends request to share/request a lecturer, whereas academic provider represents the department that offer the requested course.

Journal of Theoretical and Applied Information Technology
10th March 2015. Vol.73 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

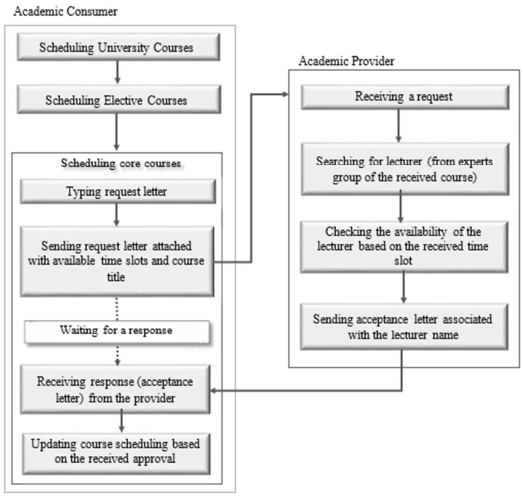

Figure 1: Manual Process of Booking Lecturer

In this work we propose a convenient and efficient information sharing tool to automate the information sharing between UPM departments in timetabling, this tool aims to reduce time and efforts spent on timetable generation. The proposed tool will be applied in timetable planning for the Faculty of Computer Science and Information Technology at the Universiti of Putra Malaysia.

3.3 Constraints

In general, timetabling constraints can be due to academic reasons, lecturers' personal preferences, or physical constraints:

- The lecturer cannot be at two rooms at the same time. Therefore, lecturer courses clashing should be avoided.
- Some lecturers prefer to have their courses at certain days or times.
- Heads of departments are lecturers as well, but they have extra work to be done. So they can held only one course.
- Avoid scheduling lecturers' courses one after the other at the same day.
- Based on UPM timeslot combination, the credits course session is divided into two sessions (two hours session and one hour session). These sessions are scheduled in two different days. So take the time slot combination in your account when scheduling the lecturer courses to avoid clashing.

4. RELATED WORK

A wide variety of approaches and models have been proposed in tackling the timetabling problem. These studies can be classified into three categories based on the underlying technologies adopted: operational research (OR), human-machine interaction, and artificial intelligence (AI) [7].

From software engineering perspective, many software engineering approaches have been introduced to either solve the timetabling problem by introducing techniques that can generate timetable with satisfaction of hard and soft constraints, or by providing tools to support timetable's builders in decision making during timetable planning.

An automatic software engineering approach is proposed by Lee et al to address the challenges in the timetabling problem. Task-based conceptual graphs provide the automation of software development processes including specification, verification, and automatic programming. This approach offers several benefits that are useful for addressing the challenges in the timetabling problem such as directly performing modification on the specifications rather than on the source code [7].

Another software engineering approach is to use Multi-Agent as a software solutions to course timetabling problem. Di Gaspero et al proposed a framework where each department has three cooperating agents: Solver to search for a local solution to scheduling issues, Manager to maintain the price quotations for academic resources, and Negotiator which is responsible for deliberating the list of sell/buy bids. This multi agent architecture is based on a marketplace and an artificial currency where each department exchanges its own resources for its own selfish interest [8]. In 2003, De Causmaecker et al suggested that the decision support within timetabling system could best be built on the multi agent paradigm. They claim that between real world operators, better decisions are obtained through a negotiation process in which all partners actively research better solutions and ways to alleviate other partner's problems [9].

Piechowiak et al designed an interactive support decision tool that helps user when faced with timetabling problem [10].

In 2012, Jaime et al implemented a web based scheduling system, known as *udpSkeduler* [11]. This system presented to support decision

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

making by creating a communication channel between the schedulers and the instructors in order to gather time availability information for teach courses.

Although the above papers discussed solutions to the problem of timetabling using different frameworks and approaches, they did not stress on the importance of information sharing in course timetable planning.

In this study we improve this shortcoming by providing a prototype system to enhance the information sharing in preparing course timetable.

TDMST has two essential elements that can distinguish it from the systems discussed in the literature above. The first element is the automation process of sharing information between the involved departments in course timetabling. TDMST focuses on automat sharing information during scheduling resources that need to be requested from outside the department. The second element has to do with automating the process of sending requests to other departments and providing techniques for supporting decision making when responding to these requests.

5. PROPOSED SYSTEM SOLUTION

We proposed autonomous and intelligent supporting tool that focusing on information sharing among university departments during the creation of timetabling stage. This tool is concerned with automating the flow of course information, while consisting of timeslot and lecturer among departments.

5.1 Requirements Specification

Requirements include descriptions of system properties and specifications for how the system should work. Generally, requirements are statements of what a system should do rather than how it should do it. The answers to how questions fall into the realm of design. For requirement elicitation from the user point of views, the most important factor is to determine the group of user that will directly and possibly use the system intensively. Then, we conduct interviews and analyze the potential user by using personas to get the rough idea about the requirements. In FSKTM, timetable creation is performed by the head of the departments, so they are the major potential users of the proposed tool. Therefore, head of CS and SE departments were interviewed to identify the requirements.

5.1.1 Functional requirements

The functional requirement for this system are stated below:

- The system should enable users to manage the academic resources data.
- The system should be able to build an automatic course timetable for all kind of courses (core and elective) based on the study scheme.
- Along with autonomously creating timetable, the system is required to automatically send request to book external courses from other departments, the selection of this courses is based on the study scheme and the semester number.
- Additionally, the system should be able to send a notification email to the head of department in order to urge him to respond to the sent request.
- The system should be able to provide its users with a technique to respond to the requisites sent from other departments along with guiding them to select appropriate lecturer and time slot in the case that the request details were not acceptable.
- Finally, system should provide functions to manage its user's accounts.

5.1.2 Nonfunctional requirements

This kind of requirements specify how a system works. The nonfunctional requirements are:

- The system shall not crash or fail in the middle of processing data with an exception of operating system error or external factors.
- The system should provide a consistent and friendly users interface, menus and commands across all its parts.
- The results or the output to the user should be correct and accurate based on what have been selected.

5.2 Overview and System Description

This section presents a summary overview of Timetabling Decision Making Supporting Tool (TDMST). This tool is mainly used to enhance information sharing and flow during course

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

timetable planning. For administration purposes, we added some extra functions.

TDMST provides a guidance and supports decision making in course timetabling. This function could be established by enhancing information sharing between academic consumer and academic provider. Additionally, several functions are available for different types of users. Figure 5.1 presents a general overview of the prototyping.

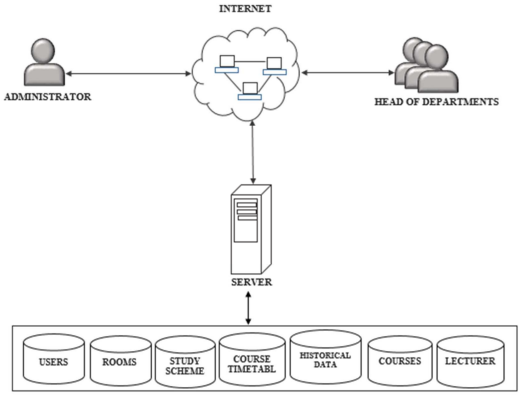

Figure 5.1: General Overview of TDMST.

TDMS tool users can be categorized into two categories: administrator and heads of departments (or any user who is responsible for creating course timetable).

Administrator can manage other users' data (add, remove and update) and rooms' data. In addition to updating study scheme and resources (lecturers, courses) data, head of department can build course timetable and response to the other department requests. Building course timetable procedure involves sending a request automatically to other departments with a list of needed lecturers based on study scheme. Simultaneously, a notification email is sent automatically and lecturer historical data is updated.

The area of this project is based on CS department resources dataset (lecturer, courses, course timetable (if it is already generated), study scheme, and historical data) and SE department dataset (lecturer, courses, course timetable (if it is already generated), study scheme, and historical data), in addition to one table stores users data. This data serve the aim of developing TDMS system. All of the information that is necessary to creating course timetable is stored in relational database.

5.3 System Context Diagram

TDMST context diagram is shown in figure 5.2 this context diagram is a high level of the tool defines the boundary between the system and its environment. TDMST environment is represented in three entities; administrator, academic provider, and academic consumer.

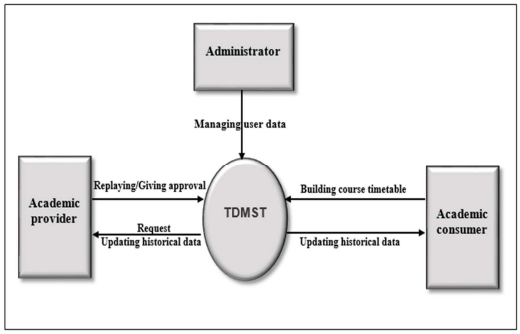

Figure 5.2: System Context Diagram

5.4 System Platform

TDMST is implemented to be used as internet application by more than one user. Currently, this system uses database stored on local server. NetBeans IDE 8.0 is used to develop the web application. NetBeans IDE lets us develop java web applications quickly and easily, as well as HTML5 applications with HTML, JavaScript, and CSS. In addition, NetBeanse can be used under any platform containing Java Development Kit (JDK) 5.0 or newer. To create the database, MySQL database engine is used. MySQL Data Management System was chosen for the following reasons:

- MySQL is Open Source System available any time.
- MySQL supports various development interfaces such as JDBC, ODBC, PHP, C++, and many others.

5.5 System Implementation

This section gives a detailed look at the system implementation. TDMST provides its user with all the information they need during the process of course timetable planning. In addition, it extracts the related information that is necessary to scheduling courses offered outside the department. One department can represent academic provider and academic consumer at the same time.

The generation of course timetable for a given semester is decomposed into three stages that involves the interaction of the departments. First

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

stage begins with the identification of the courses that will be offered in particular semester. Then, TDMST builds course timetable for local courses that are offered by the department itself. The necessary information is constructed from the database. This information includes the group of experts for each course and the available timeslots for each lecturer in the group. Based on this information, the system allocates lecturer and timeslot to the corresponding courses. Selection of these courses is done based on the information extracted from the study scheme.

Once the scheduling of courses offered inside the department is done, the procedure enters the second phase, here, a number of schedules should be compiled to build a temporary course timetable for those courses offered by other departments (academic providers). The temporary table is built based on information shared between the departments (academic consumer and academic provider). Thus, the system can assign appropriate lecturer and timeslot to a particular course. Once the temporary course timetable is ready, it will be sent autonomously to the academic provider in order to get an approval. Sending the temporary course timetable means that it will be displayed in the academic provider interface whenever the academic provider login into the system. To prompt the academic providers to login to their accounts, TDMST sends an email to the academic provider simultaneously with displaying the request in the academic provider main interface.

Third stage is to assist the academic provider to make decision. This guidance is provided in the case that the allocation of timeslot or lecturer to the course is not acceptable. In this case, TDMST supports the user by providing him by all the information needed to reallocate the timeslot or lecturer to the requested course. After the submission of the selected lecturer and/or timeslot, TDMST inserts the temporary table records into the courses timetable in the consumer department dataset and then resets the temporary course timetable. In this paper, we supposed that university courses are already scheduled. New allocation of lecturer to course will be added automatically to the historical data. The academic provider can be academic consumer as well, we have used these names to differentiate between the departments for description purpose. Figure 5.3 shows all three stages in course timetable planning.

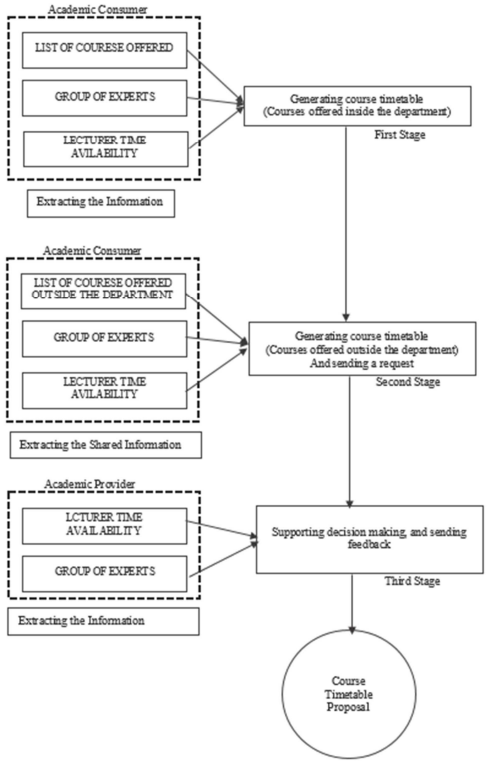

Figure 5.3: The Three Stages of Course Timetabling Process

5.6 System Interfaces

The authorized users to access to the system's interfaces are the administrator and the head of departments (the schedulers). Typically, login interface is the main interface that any TDMST user sees at the beginning. If the user is the administrator the system directs him to the admin interface which provides him by a set of functions to manage the users and rooms' data.

Interface shown in figure A.1 will appear after the successful login of one of the head of departments, for example head of CS department. If there is any requests that have been sent from other departments, it will be shown on this interface with the related functions (Accept or Change functions). This interface provides the user by various links to another interfaces. Every interface provides the user by set of functions that allow him to add, remove, and update the related database (lecturer, courses, and study scheme). Any updating of any field will be automatically associated with updating the related data in the other tables. For example, when the user updates the details of one course, the related information in study scheme

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

table and historical data table will be updated as well.

Once the department has received request from another department, the main interface displays a list of requested courses associated with proposed timeslot and lecturer name. The functions to accept or reallocate the timeslot and/or lecturer name are provided as well. In the case that there is a need to change the request details, system will direct the user to another interfaces that will help him to select appropriate lecturer and timeslot. Figure A.2 screenshots of SE head of department main interface when the SE department has received a request from CS department. Once the academic provider has responded to the academic consumer request, the system will combine the three parts of course timetable (university course timetable, local course timetable, and outside course timetable), and then enables the user to see the complete course timetable as shown in figure A.3.

6. SIMULATION PROCESS AND RESULTS

In the software development process, system validation and verification are important steps. During and after development process, any piece of the system should be tested in order to ensure it has met the requirement stated.

In the literature, a number of studies have discussed the problem of course timetabling for the sake of achieving different objectives. Each study addressed the problem of timetabling from different angles and based on problems encountered by researchers. The variety of objectives and problems researched makes it difficult to make a meaningful comparison [16]. Based on the literature, the approach adopted in the study at hand has not been discussed particularly before.

In our case, the final prototype is considered as the final project. Compared to the existing manual system for sharing information during course time table preparation, TDMST was able to produce better scheduling in terms of reduction of time and efforts required to generate course timetable. By using the manual system, requesting courses from outside FSKTM' department and exchanging related information may take many hours or sometimes many days to be done. However, that can be done immediately by using TDMST.

Besides, formative evaluation has been performed to the final project to ensure that it fully meets requirements stated and usability aspects. Based on the positive feedback received from the potential users of the system (head of departments) the final project satisfies their needs for supporting decision making in timetable planning and automating information sharing throughout sharing/exchanging academic resources.

Usability responds to the question "can the user use the system and can he or she do so effectively?" The user's ability and competency to complete a task successfully, promptly, and with little difficulty depend on how easy the user interface is to use and learn, and on whether errors or other problems exist in its design.

The main goal of usability testing was to identify, whether the prototype meets our usability goals or not. For this purpose, Usability testing was conducted. Two participants were given three functions to perform on DMST tool (prototype). Examples include update course data (add, delete, and update), create timetable, and respond to other department request (either by directly accepting the request or changing the request details). They were instructed to talk about positive and negative experience as they performed the tasks. The participants performed the task easily and understand the flow of the process in the application. They clearly know where to go in to complete the task. However, we identified some issues requiring attention. The main issue was adding functions to allow updating the proposal timetable.

7. LIMITATIONS

The main aim of this study was to propose an information sharing model for university course scheduling. Two resources (lecturer and course) only have been involved in the current study while the rest of the resources that are typically involved in course timetable generation such as students and rooms were ignored. The present authors intend to extend the research area to involve more resources.

8. CONCLUSION

This study aimed at designing an autonomous information sharing tool to support decision making when planning course timetable. It focuses on scheduling external courses that are requested/shared from another departments. The proposed tool has been evaluated, Timetabling Decision Making Supporting Tool was accepted from the users with number of suggestions to

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

improve the design and the interfaces for future work.

The results of TDMST evaluation demonstrate that information sharing during course timetable planning has led to several benefits. First of all, the time required for generating course timetable has reduced significantly. Second, the work load for schedulers has reduced which makes them free to do other tasks. Third, information sharing in certain stages of course scheduling has successfully automated. Fourth, a new channel of communication between the departments has created which facilitates and accelerates exchanging information between them. Finally, human errors has eliminated which reduces timetable conflicts and improves its quality.

This work is an ingoing work, and more work has to be done to improvise the presented approach. The aim will be to use software agents approach to improvise the proposed solution. In addition, we plan to extend the solution to involve more than two departments and more academic resources.

REFRENCES:

[1] Schaerf A. A survey of automated timetabling. Artif Intell Rev 1999;13~2!:87–127.

[2] Yan, Zhijun, Baowen Sun, and Tianmei Wang. "A study on E-government information sharing." *Advances in Grey Systems Research*. Springer Berlin Heidelberg, 2010. 571-580.

[3] Gardoni, M., M. Spadoni, and F. Vernadat. "Harnessing non-structured information and knowledge and Know-How capitalisation in integrated engineering: Case study at Aerospatiale Matra." *Concurrent Engineering* 8.4 (2000): 281-296.

[4] Rafaeli, Sheizaf, and Daphne R. Raban. "Information sharing online: a research challenge." *International Journal of Knowledge and Learning* 1.1 (2005): 62-79.

[5] Landsbergen Jr, David, and George Wolken Jr. "Realizing the promise: Government information systems and the fourth generation of information technology." *Public Administration Review* 61.2 (2001): 206-220.

[6] Bunderson, J. Stuart, and Kathleen M. Sutcliffe. "Comparing alternative conceptualizations of functional diversity in management teams: Process and performance effects." *Academy of management journal* 45.5 (2002): 875-893.

[7] Lee, Jonathan, et al. "University timetabling through conceptual modeling."*International journal of intelligent systems* 20.11 (2005): 1137-1160.

[8] Di Gaspero, Luca, Stefano Mizzaro, and Andrea Schaerf. "A multiagent architecture for distributed course timetabling." *Proceedings of the 5th International Conference on the Practice and Theory of Automated Timetabling (PATAT-2004)*. 2004.

[9] De Causmaecker, Patrick, Djamila Ouelhadj, and G. Vanden Berghe. "Agents in timetabling problems." *Proc. of the 1st Multidisciplinary International Conference on Scheduling Theory and Applications* (2003): 67-71.

[10] Piechowiak, Sylvain, and Christophe Kolski. "Towards a generic object oriented decision support system for university timetabling: an interactive approach."*International Journal of Information Technology & Decision Making* 3.01 (2004): 179-208.

[11] Miranda, Jaime, Pablo A. Rey, and José M. Robles. "udpSkeduler: A Web architecture based decision support system for course and classroom scheduling." *Decision Support Systems* 52.2 (2012): 505-513.

[12] Foulds, Les R., and David G. Johnson. "SlotManager: a microcomputer-based decision support system for university timetabling." *Decision Support Systems*27.4 (2000): 367-381.

[13] Piechowiak, Sylvain, Jingxua Ma, and René Mandiau. "An open interactive timetabling tool." *Practice and Theory of Automated Timetabling V*. Springer Berlin Heidelberg, 2005. 34-50.

[14] Moschopoulos, Charalampos N., et al. "A user-friendly evolutionary tool for high-school timetabling." *Tools and Applications with Artificial Intelligence*. Springer Berlin Heidelberg, 2009. 149-162.

[15] Müller, Tomáš, and Roman Barták. "Interactive timetabling: Concepts, techniques, and practical results." *PATAT*. 2002.

[16] Lewis, Rhydian. "A survey of metaheuristic-based techniques for university timetabling problems." *OR spectrum* 30.1 (2008): 167-190.

Journal of Theoretical and Applied Information Technology
10th March 2015. Vol.73 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Appendix A:

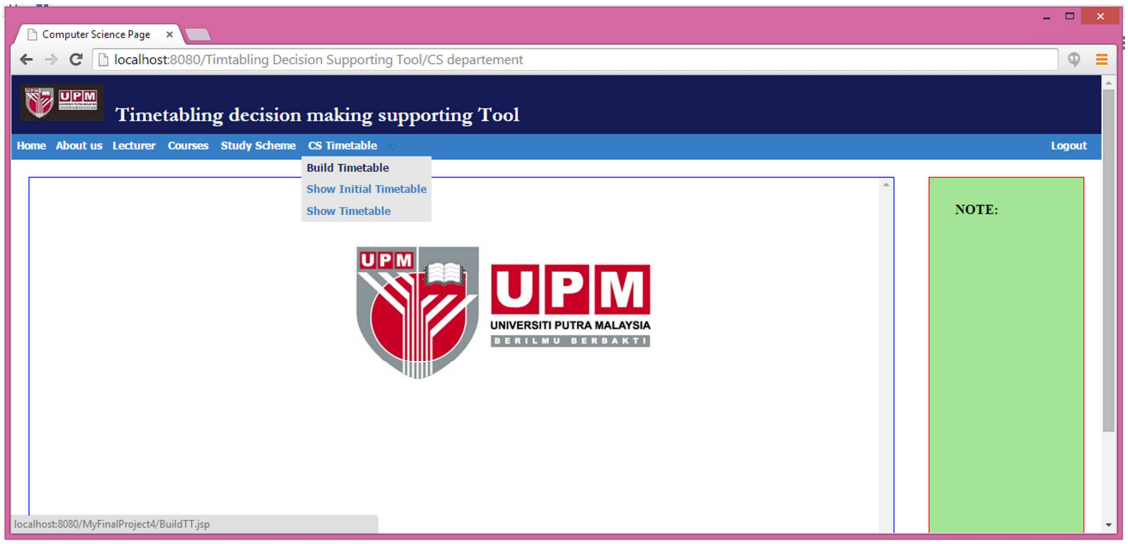

Figure A.1: The Main Interface of the Head of Department of CS

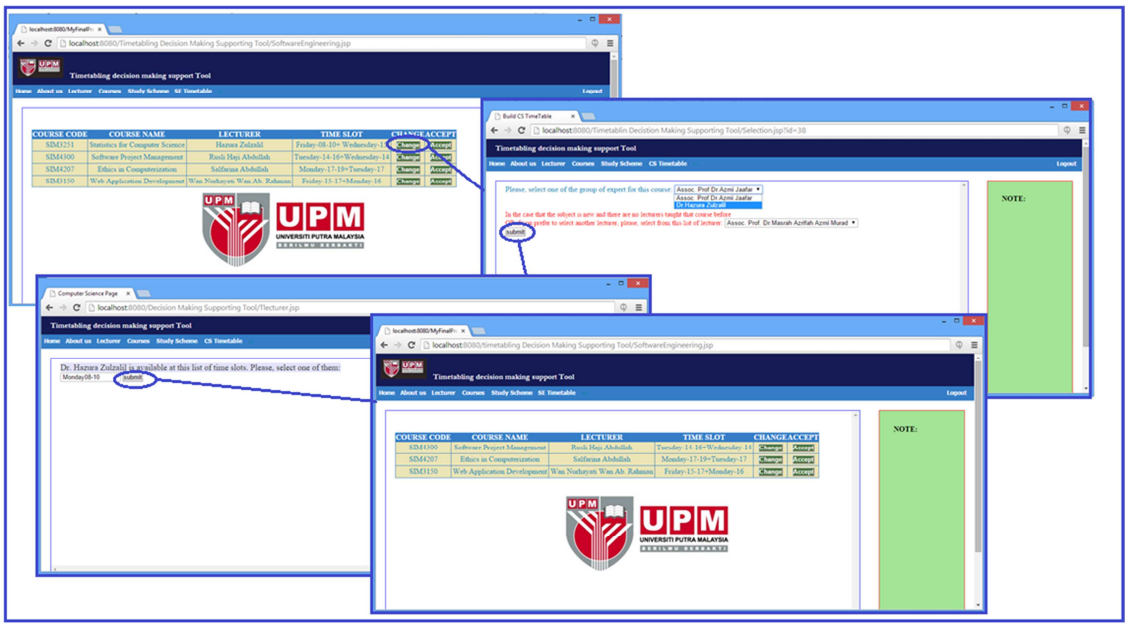

Figure A.2: Screenshots of the Interfaces Flow to Support Users in Making Decision

Journal of Theoretical and Applied Information Technology
10th March 2015. Vol.73 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure A.3: The Course Timetable Proposal