Journal of Engineering and Applied Sciences 12 (23): 7299-7305, 2017
ISSN: 1816-949X

Designing and Implementation a Web-Based Architecture for an Examination Timetabling System

[1]Syariza Abdul-Rahman, [1]Aida Mauziah Benjamin, [1]Mohd Faizal Omar, [1]Razamin Ramli,
[2]Ku-Ruhana Ku-Mahamud and [1]Waleed Khalid Abdul Jabbar
[1]Department of Decision Science, School of Quantitative Sciences,
[2]Department of Computer Science, School of Computing, Universiti Utara Malaysia,
06010 Sintok, Kedah, Malaysia

Abstract: In this study, we describe the design and implementation of a multi-objective examination timetabling system for the Universiti Utara Malaysia (UUM). As practice by most of the academic institutions, an examination timetable must be generated before the students sit for an examination. During the examination timetable generation, the assignment of the enrolled students of the offered courses to different timeslots and rooms should be considered in order to avoid clashes. At the same time, clashes of lecturers are also considered in this study since lecturer's attendance to the examination hall for the UUM case is compulsory for observing the progression of the examination. These requirements reflect the quality of the generated timetable. The aim of the developed system is to produce a conflict-free timetable while maximizing the satisfaction of preferences by students and lecturers. In addition of the automatic generation of the examination timetable, this study also emphasizes on the Graphical User Interface (GUI) of the system that enables the user to participate in the input, construction and modification of a timetable. In the construction and modification phase, the GUI enables the user to directly assign or reassign courses to timeslots which is guided by heuristics. The results of the examination timetabling system were compared with the current timetable that was manually constructed and shows improvement in terms of its duration, quality and flexibility. The proposed system has simplified the activity of generating examination timetables since manual approach was considered during timetable generation.

Key words: Examination timetabling, timetabling system, system architecture, graphical user interface, approach, during

INTRODUCTION

The academic institutions always seek to excellence through utilizing new technologies in the development of their infrastructure. The technology has facilitated the life of people by simplifying their traditional and complex daily activities (Gonen and Basaran, 2012; Tosunta *et al*., 2015). One of the essential technologies required by an academic institution is related with educational timetabling since the timetabling routine is a very tedious and difficult task. Moreover, it is a time-consuming task which occurs periodically depending on the institutions. The existence of technologies in developing timetabling system could help the academic institution to be more productive because the job can be done in a very quick time while satisfying all the related preferences. Examples of educational timetabling are examination timetabling, course timetabling and teacher scheduling (Qu *et al*., 2009).

This study focuses on the design and implementation of the web-based architecture for an examination timetabling system at the Universiti Utara Malaysia (UUM). As most of the academic institutions, UUM faces a problem of scheduling examinations in every semester. The timetabling procedure at UUM varies from manual timetabling to semi-automated timetabling. The current used manual timetabling process has some troubles such as time consuming, resources consuming and further it is not accurate (Abdul-Rahman *et al*., 2014). Due to large number of students, courses, exams, rooms and constraints, the difficulty of the problem increases and the need to have

Corresponding Author: Waleed Khalid Abdul Jabbar, Department of Decision Science, School of Quantitative Sciences, Universiti Utara Malaysia, 06010 Sintok, Kedah, Malaysia

an automated examination timetabling system that can produce feasible and high-quality timetables is becoming urgent.

In this study, an examination timetabling system is developed based on the list of hard and soft constraints of the identified problem as the preferences to reflect the real situation of the problem. By identifying the preferences as hard and soft constraints, it could simplify timetables generation process.

Literature review: Examination timetabling problem is defined as a process of assigning a set of examinations into a limited number of timeslots (time periods) and rooms (of certain capacity), subject to a set of constraints (Qu *et al*., 2009). Usually, there are two types of constraints involved in examination timetabling problem which are hard constraint and soft constraint. Hard constraint must be fulfilled and cannot be violated. For example, no student should undergo different exams at the same time. Meanwhile, soft constraint is defined as a condition that is desirable but is not compulsory to be achieved. Locating conflicting exams to a number of timeslot apart is one example of soft constraint. Feasible timetabling solution is one which satisfies all the hard constraint. Practically, it is impossible to satisfy all the soft constraints, thus, the quality of timetables is measured by checking to what degree is the soft constraints are achievable.

Research on automated system for examination timetabling problem has started to give impact to timetabling community since its applicability within academic institutions. Carter and Laporte (1997) discovered that most of the systems were implemented for particular institutions. Since, that more and more works have been done to study the centralization of examination timetabling systems. Throughout the years, there were various timetabling tools and languages employed by researchers. ECLIPse was used for constraint logic programming by Gomes (2004). A language called UniLang was proposed by Reis and Oliveira (2000) which represents data and solutions for university timetabling problems. Besides, to implement the calculation on local search algorithm, EASYLOCAL++ was introduced by Di-Gaspero and Schaerf (2003) where hill climbing, simulated annealing and Tabu search were used to solve a timetabling problem. The study found that the characteristic of the language was the main tool for generality and reusability when solving the problem. Another type of language used in timetabling system called XML was employed by Sheibani (2002). It is found that the process of constraint and resources identification is easier with the utilization of XML.

There are many approaches and techniques involved in solving examination timetabling problem. One popular approach in solving examination timetabling problem is heuristic technique based on graph colouring problem (Abdul-Rahman *et al*., 2014; Welsh and Powell, 1967). The vertices in the graph represent an examination while the edge between the vertices represents the conflicting between examinations. The aim is to ensure that no adjacent vertices have the same colour or in other words, the conflicting examinations should not be assigned into the same timeslot. Apart from that, fuzzy logic was applied along with graph coloring techniques (Asmuni *et al*., 2005). There are three main phases in fuzzy logic system which are fuzzification inference engine and defuzzification. The study found that hybridization between fuzzy logic and graph sequential techniques produced an appropriate evaluation thus yielding a better result. Other famous technique used in this research area is local search based techniques (e.g., simulated annealing, tabu search and Genetic algorithm). The algorithm of this technique started with selecting its candidate solution and then iteratively moves between its neighborhoods. OTTABU, a four-stage Tabu serach was developed by White and Xie (2001) for an examination timetabling problem at the University of Ottawa. Thompson and Dowsland (1988) applied simulated annealing process to enhance the soft constraint satisfaction. The developed system was successfully used in Swansea University, since, 1993 as its increases the flexibility in allocating the large and complicated exams of the timetable.

AlDeeb *et al*. (2015) employed Hybrid Intelligent Water Drops Algorithm (H-WID) in solving examination timetabling problem at the Universiti Sains Islamic Malaysia. The system was designed to figure out the problem of manual assignment during timetable construction. Two phases methodology involved in the development of the proposed system. The first phase is to generate the feasible solution which satisfied all the hard and soft constraints as much as possible using the H-WID. The second phase involved with the visualization of the result. Results from the system demonstrated that the proposed algorithm can deliver a high-quality examination timetable in a shorter time. A study by Mutingi and Mbohwa (2017) discussed the application of Fuzzy Grouping Genetic algorithm in solving the examination timetabling problem where the multifactor evaluation method was utilized to represent the constraints as weighted normalized cost functions. It is found that the fuzzy parameters of the examination timetabling problem can be used as the references for the decision maker and concluded that that the proposed approach is efficient compared with past approaches.

Table 1: Constraints for the UUM examination timetable problem

Hard constraints	Soft constraints
H_1: No student should be required to sit more than one examination simultaneously	S_1: Largest exam must be scheduled as early as possible
H_2: The total number of students assigned to a particular room (s) must be less than the total room capacity	S_2: Splitting exam into different rooms in predetermined neighborhood area
H_3: Every exam must be scheduled in exactly one timeslot	S_3: More spacing between exams should be allocated to maximize student's exams preparation (student's spreading)
H_4: All examinations in the same room should end at the same time (no mix durations)	S_4: lecturer's spreading
H_5: Every room must be assigned to at least one invigilators/lecturer	S_5: The 3 h exams are given priority to be scheduled in the morning slot.
H_6: An invigilator/lecturer must not be scheduled to invigilate more than one exam in the same timeslot	S_6: Certain exams are given priority to be assigned to certain rooms
H_7: Exams must not be scheduled during weekend and public holiday	S_7: Maximizing room utilization
H_8: Pre-assigned slot of examinations	S_8: Minimizing number of room used

Table 2: Student's information

IDs	Student's name	Course codes	Course names	Groups	Program's names
225371	Azizan Bin Muaz	BEEA1013	Economy	A	B.Sc. AgriBus. Mgmt. (Hons.)
225685	Zulkifli Bin Nur Afifah	BEGA1013	Biology	A	B.Sc. AgriBus. Mgmt. (Hons.)

Current research done in this field of study includes work by Oluwaniyi *et al.* (2016). The objective of the study is to build up an easy to use and efficient timetabling system in allocating invigilators and scheduling of examination. The system was developed using Macromedia Dreamweaver while the programming languages include Hypertext Preprocessor (PHP), Hyper Text Markup Language (HTML), Structured Query Language (SQL) and Cascading Style Sheets. The proposed system solved the problems found in manual assignment of examination timetable. Lach *et al.* (2016) discussed on the satisfaction of wide range of hard and soft constraint with the application of IT system Moses. In the proposed study, 2 primary users which were lecturers and supervisors were involved in managing the timetabling process. The system architecture involved the utilization of Java Server Faces as the interface. Not only that, the modern responsive design of the system that was based on Twitter Bootstrap framework was proposed to allow users to access the system from their own device. Moreira developed a system for automatic construction of examination timetable using Genetic algorithms and the algorithms were coded in visual basic language with prior definition of some initial conditions. Lim proposed a timetabling system namely as UTTS Exam for the National University of Singapore. It was successfully used and significantly produced a timetable with minimum examination period and save the time.

In summary, it can be concluded that research on examination timetabling problem system is gaining more popularity day by day. The requirement and demand to produce an efficient timetable with minimum time spend could drive more studies to be conducted in this field of research.

Problem description: An examination timetabling problem at the Universiti Utara Malaysia (UUM) is investigated. As the university grows with the increase of the students and new teaching programs, this has made the examination timetabling procedure at UUM becomes more complex. UUM is using a manual examination timetabling procedure where the current system may take about 4 weeks to produce a complete timetable. There are many weaknesses that have been encountered, many complaints and requests for changes might be received from students and lecturers after the first timetable is announced. Accordingly, the timetable must be re-timetabling manually many times before the final draft is obtained. Some of the complaints and requests are the occurrences of back-to-back examination, students have not enough time to do the revision, clashes of student or lecturer's examination and lecturers do not have enough time to do the marking. The existence of these unique preferences makes this problem challenging every semester due to the need to cater a set of constraints that have never been tackled before in the literature. Table 1 shows the hard and soft constraints for the UUM examination timetable problem a s summarized by Abdul-Rahman *et al.* (2014).

An examination timetabling problem at UUM for first semester of the 2013/2014 academic year in UUM is considered which involved 13,359 students taking one or more 639 examinations. The problem involved 718 lecturers and 40 rooms with conflict density of 4%. The problem considers two slots for assignment in each day which involved morning and afternoon session and no examinations during the weekend. The problem was solved using graph colouring heuristics as presented in Abdul-Rahman *et al.* (2014). The data format of the problem is as presented in Table 2-5 where it is provided in Excel file and uploaded to the system.

Data format of examination timetabling system of UUM: Table 2-5 is discussed in this study.

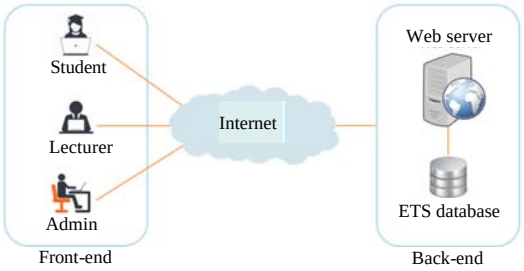

Fig. 1: Examination timetabling system architecture

Table 3: Lecturer's information

Courses	Groups	Staff IDs	Names
BEEB1013	G	1000	Patrick Naina
BEEB1013	H	1001	Rudy Mohamed

Table 4: Room's information

Rooms	Room names	Capacity
TE	Teater Eksperimen	212
DMS	Dewan Muadzam Shah	790

Table 5: Course's information

Terms	Courses	Groups	No. of students
A131	BECA1013	A	23
A131	BECA1023	A	21

System architecture: The main objective of this study is to describe the architectural design of the examination timetabling system of UUM. The practical approach for implementing the examination timetabling activity in UUM is based on an architecture which provides the required framework of the system components (Mahmood and Saeed, 2013). The system architecture defines the essential design features such as interfaces, users and required infrastructure (Bass *et al*., 2013). The architecture of the proposed system can be divided into two main components: the front-end and the back-end. Both relate to each other through the system. Front end is what the users (students, lecturers and admin) can access whereas the back end is a web server and database. Front end has the user's computer and the application required while the back end has the cloud computing services like web server and data storage. Figure 1 shows the components of the architecture of the proposed system.

As shown in Fig. 1, two-tier architectures have been proposed for capturing user's inputs in the form of timetables copies for authorized students and lecturers and generating the examination timetables managed by authorized admin. In the following section, the two tires are explained in detail.

Back end: The back end includes a web server and database. The web server is a standard web server that accepts secure https requests and handles them.

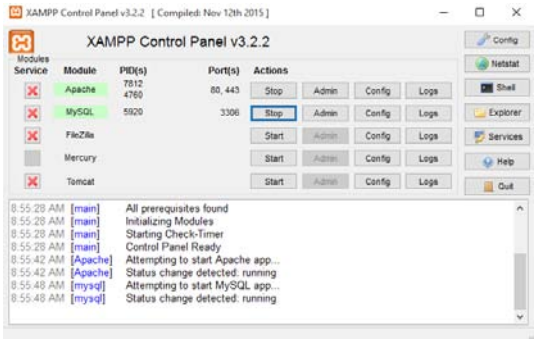

Fig. 2: Web server control panel

Figure 2 depicts a snapshot of the apache server which was used as a web server in this project and web server control panel screenshot is only accessible by server administrator.

Front end: The web-based examination timetabling system is developed mainly to serve the Department of Academic Affairs (HEA) at UUM in constructing the examination timetables automatically for the final examination at each semester. The users in the front-end are three main entities including the students and lecturers who can view their examination timetables and the admin who can manage the examination timetabling activity. To develop the system, the set of functional and non-functional requirements was collected from HEA. Then, it was represented and visualized using Unified Modelling Language (UML) to make it more understandable, readable and organized. Accordingly, the system was developed using PHP language for web pages and MySQL for database. The front-end interfaces were designed using HTML forms because of its simplicity and popularity. This page contained three types of activities as presented in Table 6.

The examination timetabling system is a full featured and flexible system. It supports the entire life cycle of generating timetables. The system shall be implemented to help HEA to manage the process of examination timetabling easily as its flexibility and to save the time of them through streamlining the process.

Graphical user interface: The Graphical User Interface (GUI) plays a key role in our examination timetabling system. Not only does it serve to organize the data provided for a timetabling problem but it also handles the complexities of modifying the underlying model while providing useful information before and after. Once the timetable has entered all the required information, a sequence of timetables is constructed and an evaluation

Table 6: Three types of activities involved for the UUM examination timetable system

Activities	Descriptions
Students	With respect to the student's interfaces, the students can perform the tasks of browsing their timetable and exporting the timetable into pdf file
Lecturers	With respect to the lecturer's interfaces, the lecturers can perform the tasks to the student's activities
Admin	The admin side consists of 6 sub-modules namely login, dashboard, settings, timetable, clashes, statistics, tables and logout. Sub-module login allows the admin to access the system. Sub-module dashboard provides the admin with information about the system. Sub-module settings allows the admin to manipulate the data inputs of the systems. Sub-module timetable allows the admin to generate and manipulate several copies of the timetables automatically. Sub-module clashes allows the admin to check the clashes. Sub-module statistics provide the admin with some statistical information about the generated timetables. Sub-module tables allows the admin to print student's, lecturer's androom's timetables. Sub-module logout is to end the session by the admin

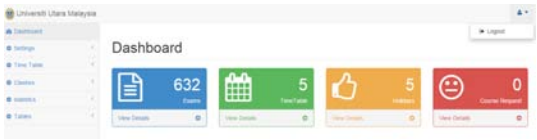

Fig. 3: Sub-module dashboard

function is applied to each. This evaluation is based on several characteristics including total conflict penalty, total proximity penalty and the number of classes without rooms.

In many cases, oversights regarding initial requirements and sudden changes may often warrant changes to the constructed timetable. Additionally, not all individuals involved will receive the schedules they desired. Hence, the GUI is designed to assist the user in evaluating the overall timetable investigating the effects of changes that could possibly be made to the timetable and confirming those changes as needed. To help the user start their evaluation, the GUI provides a few different views from which to evaluate the overall timetable whether by timeslots, classes instructors, conflicts between classes or grid. The proposed system consists of 6 sub-modules namely login, dashboard, settings, timetable, clashes, statistics, tables and logout. Sub-module login allows the admin to access the system. Sub-module dashboard provides the admin with information about the system. Sub-module settings allows the admin to manipulate the data inputs of the systems. Sub-module timetable allows the admin to generate and manipulate several copies of the timetables automatically. Sub-module clashes allows the admin to check the clashes. Sub-module statistics provide the admin with some statistical information about the generated timetables. Sub-module tables allows the admin to print student, lecturers and rooms timetables. Sub-module logout to end the session by the admin.

Figure 3 shows the sub-module dashboard where it provides information to the admins regarding students, lecturers and information regarding the timetable to be generated such as date and timetable length. Figure 4 shows the sub-module of timetable where when using the "timeslot view", the user is presented with the list of

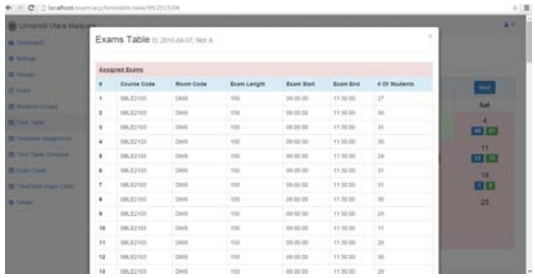

Fig. 4: Sub-module timetable

all timeslots. When a timeslot is selected, the classes scheduled in that timeslot are listed. Then, by selecting any of the given classes, further information is displayed, such as the assigned room and instructors as well as the classes with which the selected class conflicts.

RESULTS AND DISCUSSION

The tested dataset contains information related with examination timetabling which obtained from HEA, UUM through data collections and interviews with representatives from a number of timetable officers at UUM in order to know the preferences and restrictions in generating examination timetable. By using the graph colouring heuristics (Abdul-Rahman *et al*., 2014) embedded in the proposed system for solving the problem, the proposed system has successfully generated better timetable as presented in Table 7. Comparison between the manually and generated constructed examination timetables with graph colouring heuristics using the proposed system shows that in most cases the generated one is better than the manual constructed timetable proposed by UUM. The generated timetable using the proposed system shows it is better in terms of number of days and timeslot. Moreover, the proposed system successfully generated the timetable by minimizing the spreading student and lecturer where these two criteria reflect the quality of the generated timetable. Both manual and system generated can produce timetable without clashing, however the one generated using system is much faster than the manual one which is

Table 7: Comparison between the manually and generated constructed examination timetables with graph colouring heuristics

	Generated	
Variables	Manual	System
No. of days	19	11
No. of timeslots	38	22
Can handle holidays and off days?	Yes, sometimes	Yes, always
Mix duration examinations in a room	Yes	No
Student spread between examinations	No	Maximizes
Lecture spread between examinations	No	Maximizes
Time taken for timetable generation	4 weeks	1 min
The 3 h exams are given priority to be assigned in the morning slot	Yes but only to some exams	Yes
Certain exams are given priority to be assigned to certain exams	Yes	Yes
Multiple timetable generations	No	Yes
Accessible of timetable quality	No	Yes

required only about 1 min. of timetable generation. Despite, the system generated able to fulfil other criteria such as no examinations during weekend and no mix duration examinations in a room.

The quality of the generated timetable can be access though the calculation of the violation of the soft constraints. Furthermore, the proposed system can generate multiple timetable and can be compared based on the timetable quality. Other than that, the generated timetable can be improved for having a better timetable. This is done by having the function of timetable manipulation to improve the generated timetable by modifying the constructed timetable for timetable manipulation whenever the number of timeslot exceeds certain evaluation threshold, we attempt to improve the timetable by moving some unclashed examinations into new timeslots. By using this timetable manipulation strategy, the generated timetable can be improved and thus allowing flexibility in the timetable generation.

CONCLUSION

A web-based examination timetabling system was developed for the HEA at universiti Utara Malaysia. The system has generated satisfactory examination timetables on the university site and can be used for generating examination timetables based on a set of predefined constraints. The schedules and assignments have fulfilled policy goals such as the no mix examinations duration in a room and no examination slot during weekend. The results of the examination timetabling system implementation demonstrate that integration of the system into the course scheduling process may lead to multiple benefits. First, the time required for certain stages of the process was significantly reduced. Second, certain stages of the process were successfully automated. Third, course timetable conflicts and other human errors generated by the previous manual process were eliminated. The intention now is to evaluate the user's (HEA) acceptance of the new system in the real environment of UUM.

RECOMMENDATIONS

In addition, future research might examine more procedures related to the examination timetabling process and optimize the proposed system to cater more constraints, especially, whenever course scheduling is introduced. Finally, the rapidity of the scheduling process using the examination timetabling system allowed planners to devote more time to revising and fine-tuning definitive schedules and other tasks.

ACKNOWLEDGEMENTS

The researchers wish to thank the Ministry of Higher Education Malaysia for funding this study under the Exploratory Research Grant Scheme, S/O code 12826 and RIMC, Universiti Utara Malaysia, Kedah, for the administration of this study.

REFERENCES

Abdul-Rahman, S., N.S. Sobri, M.F. Omar, A.M. Benjamin and R. Ramli, 2014. Graph coloring heuristics for solving examination timetabling problem at Universiti Utara Malaysia. Proceedings of the Conference on AIP Vol. 1635, December 4-5, 2014 american Institute of Physics, College Park, Maryland, USA., pp: 491-496.

AlDeeb, B.A., N.M. Norwawi, M. Al-Betar and M. Jali, 2015. Intelligent examination timetabling system using hybrid intelligent water drops algorithm. Proceedings of the 5th International Conference on Computing and Informatics (ICOCI'15), August 11-13, 2015, Universiti Utara Malaysia istanbul, Turkey isBN:978-967-0910-02-4, pp: 318-325.

Asmuni, H., E.K. Burke and J.M. Garibaldi, 2005. Fuzzy multiple heuristic ordering for course timetabling. Proceedings of the 5th United Kingdom Workshop on Computational Intelligence (UKCI'05), September 5-7, 2005, University of London, London, UK., pp: 302-309.

Bass, L., P. Clements and R. Kazman, 2013. Architectural Tactics and Patterns. Software Architecture in Practice, Bass, L., P. Clements and R. Kazman (Eds.). Addison-Wesley, Boston, Massachusetts, USA. isBN-13:978-0-321-81573-6, pp: 203-250.

Carter, M. and G. Laporte, 1997. Recent developments in practical course timetabling. Proceedings of the 2nd International Conference on Practice and Theory of Automated Timetabling II, August 20-22, 1997, Toronto, Canada, pp: 3-19.

Di-Gaspero, L. and A. Schaerf, 2003. Multi-neighbourhood Local Search with Application to Course Timetabling. In: Practice and Theory of Automated Timetabling, Burke, E. and P. De-Causmaecker (Eds.). Springer, Berlin, Germany is BN:978-3-540-40699-0, pp: 262-275.

Gomes, C.P., 2004. Randomized Backtrack Search. In: Constraint and Integer Programming, Milano, M. (Ed.). Springer, Boston, Massachusetts is BN: 978-1-4613-4719-4, pp: 233-291.

Gonen, S. and B. Basaran, 2012. Learning Objects and their Applications. In: E-Learning Theories, Design, Software and Applications, Ghislandi, P. (Ed.). InTech, Rijeka, Croatia is BN: 978-953-51-0475-9, pp: 109-128.

Lach, G., M. Lach, J. Steenken and E. Zorn, 2016. Examination timetabling with Moses: System demonstration. Proceedings of the 11th International Confenference on Practice and Theory of Automated Timetabling (PATAT'16), August 23-26, 2016, PATAT Organization, Udine, Italy, pp: 511-514.

Mahmood, Z. and S. Saeed, 2013. Software Engineering Frameworks for the Cloud Computing Paradigm. Springer-Verlag, London, UK. is BN-13: 9781447150312, Pages: 365.

Mutingi, M. and C. Mbohwa, 2017. Fuzzy Grouping Genetic Algorithms: Advances for Real-World Grouping Problems. In: Grouping Genetic Algorithms, Mutingi, M. and C. Mbohwa (Eds.). Springer, Switzerland isBN:978-3-319-44393-5, pp: 67-86.

Oluwaniyi, N.O., B.O. Afeni and B.A. Onyekwelu, 2016. Development of an automated university examination timetabling system. Intl. J. Sci. Eng. Res., 7: 317-333.

Qu, R., E.K. Burke, B. McCollum, L.T. Merlot and S.Y. Lee, 2009. A survey of search methodologies and automated system development for examination timetabling. J. Schedul., 12: 55-89.

Reis, L.P. and E. Oliveira, 2000. A language for specifying complete timetabling problems. Proceedings of the International Conference on the Practice and Theory of Automated Timetabling, August 10-13, 2000, Springer, Berlin, Germany, pp: 322-341.

Sheibani, K., 2002. An evolutionary approach for the examination timetabling problems. Proceedings of the 4th International Conference on Practice and Theory of Automated Timetabling (PATAT'02), August 21-23, 2002, PATAT Organization, Gent, Belgium, pp: 387-396.

Thompson, J.M. and K.A. Dowsland, 1988. A robust simulated annealing based examination timetabling system. Comput. Oper. Res., 25: 637-648.

Tosunta, B., E. Karada and S. Orhan, 2015. The factors affecting acceptance and use of interactive whiteboard within the scope of FATIH project: A structural equation model based on the unified theory of acceptance and use of technology. Comput. Educ., 81: 169-178.

Welsh, D.J.A. and M.B. Powell, 1967. An bound for the chromatic number of a graph and its application to timetabling problems. Comput. J., 10: 85-86.

White, G.M. and B.S. Xie, 2001. Examination Timetables and Tabu Search with Longer-Term Memory. In: Practice and Theory of Automated Timetabling III, Burke, E. a nd W. Erben (Eds.). Springer, Berlin, Germany is BN: 978-3-540-42421, pp: 85-103.