ISSN No. 0976-5697

DOI: http://dx.doi.org/10.26483/ijarcs.v8i9.5079

Volume 8, No. 9,November-December 2017

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

IMPLEMENTATION OF A CONFLICT-FREEEXAMINATION SCHEDULING SYSTEM(ESS) IN ACADEMIC INSTITUTIONS

Tabrej Khan
Department of Information Sciences, Faculty of Computing
and Information Technology Rabigh (FCITR)
King Abdulaziz University,
Jeddah, Saudi Arabia

Syed Asif Hassan
Department of Computer Science, Faculty of Computing
and Information Technology Rabigh (FCITR)
King Abdulaziz University,
Jeddah, Saudi Arabia

***Abstract*:** University examination timetabling problem is a type of scheduling problem. Schedule the course with minimum conflict means that none of the students should have more than one exam on the same day. In this regard, we developed an Examination Scheduling System (ESS) using the available resources (Halls, Invigilators, and instructors)with no conflict and thus achievingapproval of the system among students and faculties. ESS application was made using ASP.net, C#, crystal report and SQL Server for building the relational database of ESS application.

Keywords: Examination Scheduling System (ESS), conflict, C#, Crystal Report, Asp.net, SQL Server

I. INTRODUCTION

The problem associated with Examination scheduling (ES) is encountered every semester in almost all universities. A new timetable is constructed each semester since the requirement and constraint of ES is directly related to lecturers, students, examination, and classrooms for conducting exams.The process of making exams schedule for any faculty is an exhausting process. Moreover, the way this process of timetabling is performed using primitive measure, namely using human efforts and paperwork, results in wastage of effort and precious time. The constantly occurring of conflicts in ES make the student lose their trust in the academic ES process and hampers their preparation as well as reduces their enthusiasm for exams. Nonetheless, manual ways of generating exam schedule are practiced until today. The registrar of the university prepares the exam schedule manually. The scheduling of exam using manual method by the registrar have the following disadvantages: 1) The manual process of exam scheduling takes a long time to complete; 2)random allocation of courses to faculty's without following any reasonable order results in conflicts and require registrar reschedule the examination time table till optimal schedule is made [1-6].

However, in some case after all this strenuous manual effort unexpected conflicts of exam timing happens for some unlucky students leading to frustration and underperformance. Therefore, in this regard, the development of electronic exams scheduling and management system is the order of the day to avoid conflicts and generate fair exam schedule for all students of any university.In the current research work, we have developed an intelligent examination scheduling algorithm for timetabling of exams in any university [7-12].

II. METHODOLOGY

To develop the ESSapplication we have used Waterfall methods[13].As shown in Figure 1, each phase in a waterfall model must be completed before the initiation of the next phase.The first phase of method deals with defining of the problem statement and performing functional and structural requirement analysis for building the ESS application. The requirement analysis is followed by designing of a prototype of ESS for implementation. The design phase involvesthe use of Unified Modeling Language (UML) [14] to picturize the framework of the ESS application. The internal database structure and functionality of ESS applicationwere defined using a SQL server, which is an open source relational database management system to store and manage the examination scheduling data as well as an improved algorithm for exam scheduling, respectively. Once the functionality and framework of ESSareestablished, the next big thing was to implement the system on the web by creating user-friendly dynamic web application of ESS. The dynamic web interface of ESS was created using the Model-View-Controller(MVC) architecture [15]. The model of ESS was generated using SQL server using an open source RDMS. ESS application query and the information retrieval web interfaces were developed using asp.net. While the controller mode of MVC architecture was used to receive inputs from the user and interact with the model and view using C# an object-oriented programming language. The internal structure of the model, as well as functionality of the application, were tested using white-box and black-box, respectively. Finally, the application was made operational in FCITR for scheduling the exam for each semester.

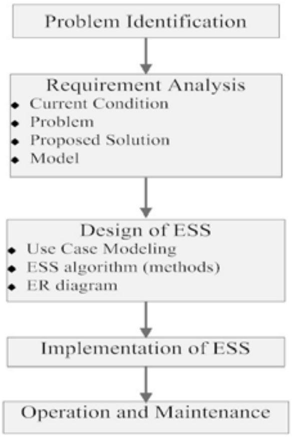


Fig. 1. Pictorial representation of the linear sequential life cycle of Waterfall model

III. RESULTS AND DISCUSSION

A. System requirement analysis and design

1) Use case diagram for ESS

The functional requirement of ESSwas based on scheduling the examination timetable without conflict. In this regard, the use case diagramwas used to model the interaction of the actor with ESS system as depicted in Figure 2.

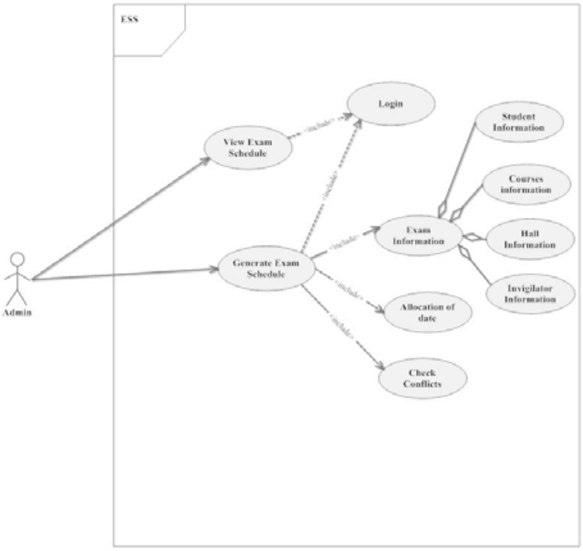


Fig. 2. A diagrammatic representation of ESS application Use Case

The use case diagram of ESShas one actor i.e., administrator and 10 use cases. To create examination scheduling the administrator have to login to thesystem with a valid and specific username and password.To generate the exam schedule the administrator needs to follow the following steps:1)Firstly,the administrator will add the examination information like student information, course information, hall information, and invigilator information;2) Secondly, the administrator will generate the conflicts matrix for each course; 3) Thirdly, the administrator will assign exam starting date, exam end date; 4) finally, in the fourth step the administrator will generate the examination schedule for each exam using crystal report.

2) Exam Scheduling Algorithm

The ESS application is built in order to produce an error-free solution for timetabling of theexam and the algorithm implemented to build the same is represented [12] in Fig 3.

The algorithm for ESSapplication can be explained as follow:

Step1: ESS first create conflict matrix i.e., acommon student between the courses.

Let's us consider we have course$_{(i)}$ where i=(1 to n)

Compare course$_{(i)}$ with course$_{(i+1)}$ to see how many students are common in both courses

Likewise

Compare course$_{(i+1)}$with course$_{(i+2)}$ to see how many students are common in both courses

And so on compare all the courses of the semester with each other.

Step 2: We pass the values of parameters

D=total number of days for any exam

FD=starting date of exam

Step 3: Schedule the courses, which have aconflict (common student) at first time period, one exam per day.

Step 4: Calculate

M_c=number of conflict courses

P= number of periods

Step 5: Set counters

Consider, D_i=0 Number of days

P_i=0 Number of Periods

C_i=0 Number of Courses

Step 6: System will check courses one by one with different constraint

$C_i= C_{i+}1$

Step 7: Number of courses is less than equal to number of conflict courses

$C_i<= M_c$

If Yes

Again check this course is scheduled or not

If Yes

Repeat Step 6 (check another course)

If no

The system will go for next time periods and check

$P_{i=} P_{i+}1$

Currently selected period is less than the equal to total number of time periods

$P_i<= P$

If yes

Increment the day and check increment day is less than equal to total number of day

$D_i<= D$

If yes

Schedule the course on that day and go to step 6.

If no

Set day is equal to zero

D_i=0 and increment the next period $P_{i=} P_{i+}1$

If no

Then scheduled the course.

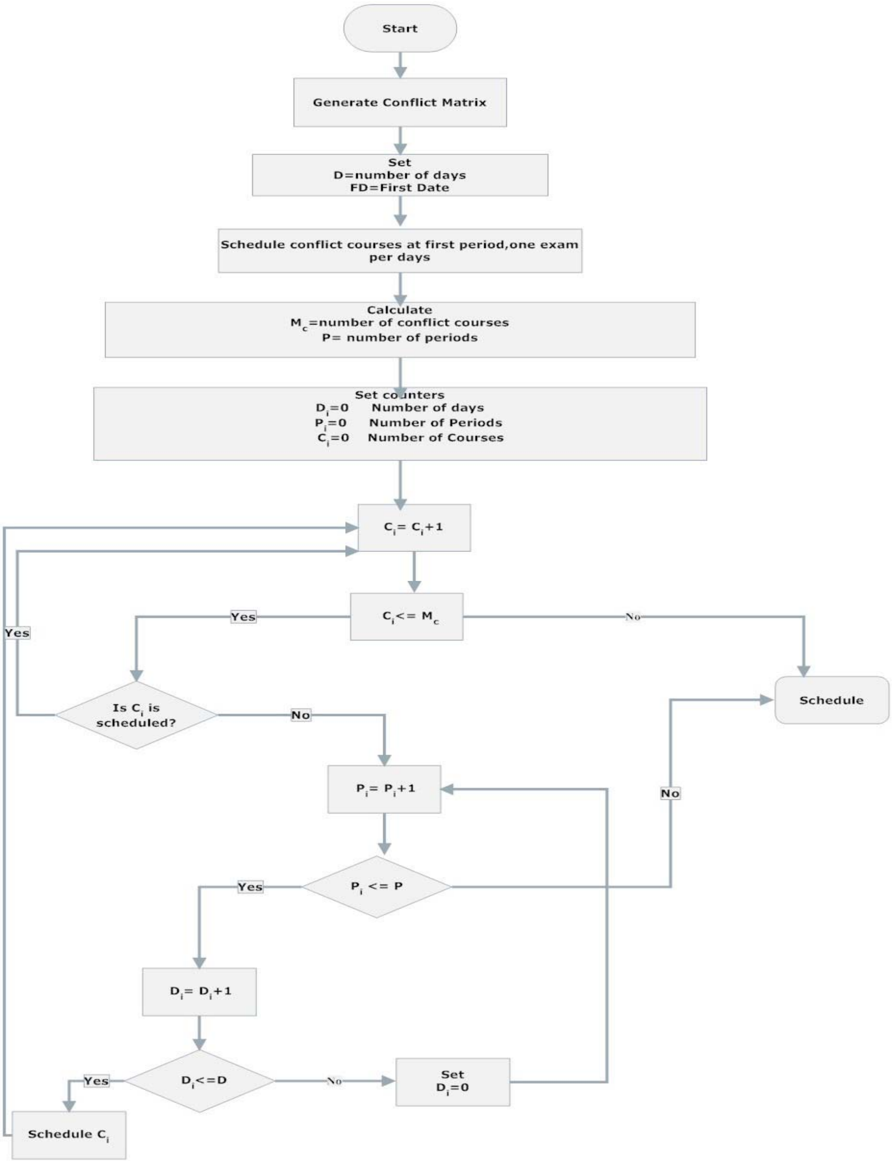


Fig. 3. Algorithmflowchart diagram of ESS

3) Entity Relation diagram for ESS

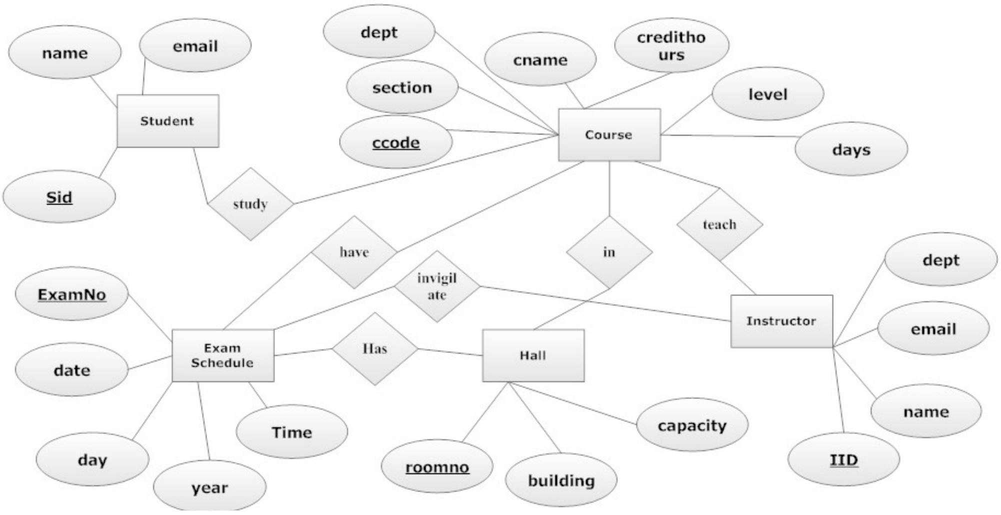


Fig. 4 E-R diagram of ESS

In E-R diagram have 5 entity namely 1) Courses 2) Student 3) Instructor 4) Hall and 5) Exam Schedule. Each entity has attributes and relationship as shown in theFig. 4. As represented in Fig 4 that one instructor can teach many courses and each course have many students. On the other hand, each course hasan exam schedule and many examsare invigilated by many instructorsand many examsarescheduled in manyhalls.

B. System Implementation

1) User Interface

a) Admin Login Page

Admin can login with secure username and password after that admin can access the application. Here we provide single admin who is in charge of designing the examination schedule

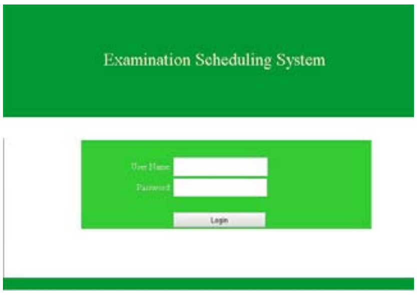


Fig. 5 Admin Page of ESS

b) Add Courses/ Hall/ Invigilators/Student Page

After login, admin can view the different option in theleft menu. Through this menu, admin can add course details, hall details, invigilators details and student details by simply uploading excel file. On the other hand, admin can upload the class roster.

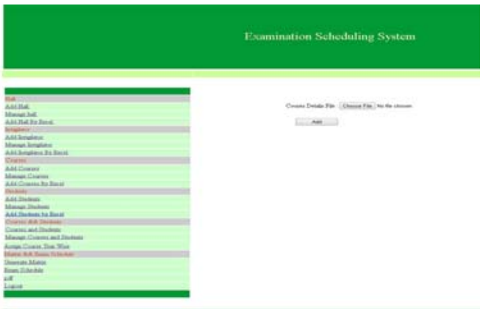


Fig. 6 Courses Page of ESS

c) Manage Courses/ Hall/ Invigilators/Student Page

Admin can manage anytime the data of any courses/ hall/ invigilators/student details by using the click edit and delete button to modify any wrong entry.

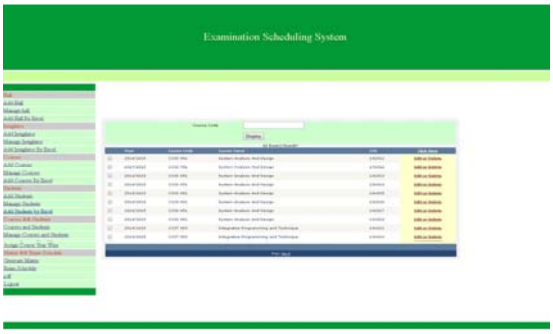


Fig. 7 Manage Page of ESS

d) Edit and Delete Course Page

This is aninterface to update and delete details of exam scheduling system if it has any wrong entry.

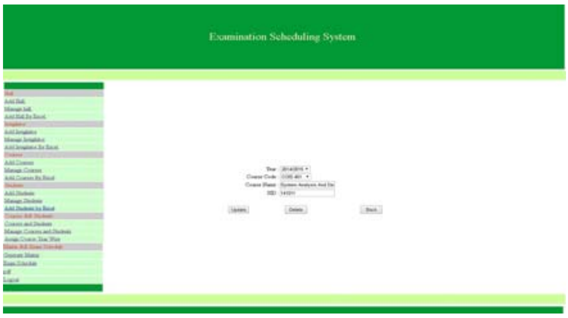


Fig. 8 E-R diagram of ESS

e) Generate Matrix Page

Admin can generate the conflict matrix by selecting one by one course.Moreover, through this approach admin will know about courses, which have acommon student.

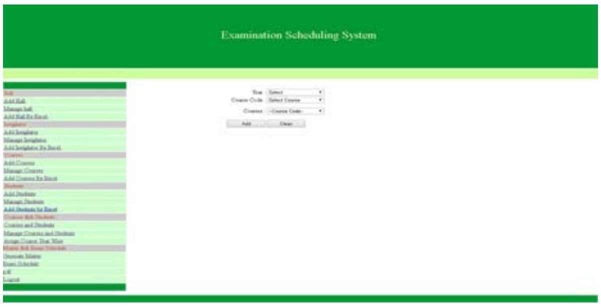


Fig. 9Generate Matrix page of ESS

f) Exam Schedule Setting Page

This is an exam scheduling setting page where admin can set the following parameters, 1) year, 2) exam type,3) exam start date and 4) a total number of days for theexam.

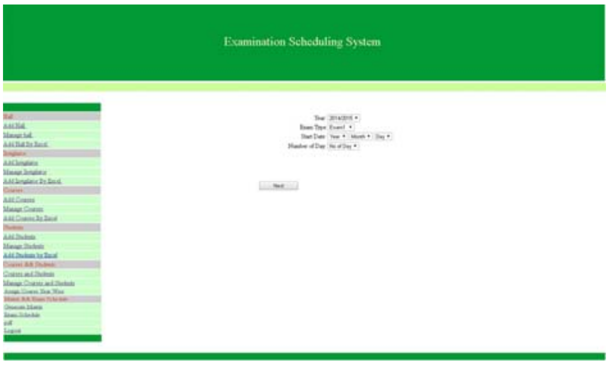
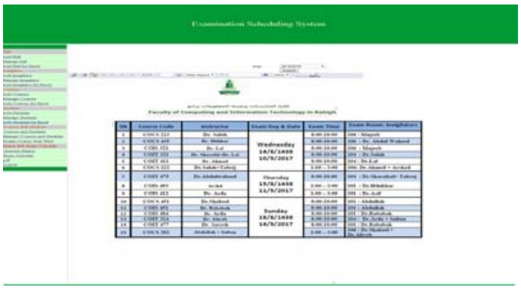


Fig. 10Exam Schedule Setting Page of ESS

g) View the Exam Schedule Page

Finally, we will get our exam schedule in crystal report.Afterwards, the admin can download the report in different format like pdf or word format.



Faculty of Computing and Information Technology in Rabigh

SN	Course Code	Instructor	Exam Day & Date	Exam Time	Exam Room: Invigilators
1	[illegible]	[illegible]	Wednesday 14/8/1438 10/5/2017	[illegible]	[illegible]
2	[illegible]	[illegible]		[illegible]	[illegible]
3	[illegible]	[illegible]		[illegible]	[illegible]
4	[illegible]	[illegible]		[illegible]	[illegible]
5	[illegible]	[illegible]		[illegible]	[illegible]
6	[illegible]	[illegible]		[illegible]	[illegible]
7	[illegible]	[illegible]	Thursday 15/8/1438 11/5/2017	[illegible]	[illegible]
8	[illegible]	[illegible]		[illegible]	[illegible]
9	[illegible]	[illegible]		[illegible]	[illegible]
10	[illegible]	[illegible]	Sunday 18/8/1438 14/5/2017	[illegible]	[illegible]
11	[illegible]	[illegible]		[illegible]	[illegible]
12	[illegible]	[illegible]		[illegible]	[illegible]
13	[illegible]	[illegible]		[illegible]	[illegible]
14	[illegible]	[illegible]		[illegible]	[illegible]
15	[illegible]	[illegible]		[illegible]	[illegible]

Fig. 11Exam Schedule Page of ESS

IV. CONCLUSION

ESSapplication is good for an academic institution like for example FCITR since before the exam timetabling was a tedious time-consumingtask and with lots of conflict between the exams. Due to this, the student uses to suffer a lot since due to conflict some student gets two or three exam on thesame day. This was totally unfair to some student. Therefore by applying our present application i.e., ESS for FCITR have made exam timetabling pretty easy since it is easy in this system to identify aconflict between courses i.e., acommon student between the courses. These conflict courses we can schedule easily through our application.

V. ACKNOWLEDGMENT

The authors thank the administration and especially the dean of FCITR for providing the facilities to carry out this work.

VI. REFERENCES

[1] A. Akbulut and G. Yılmaz, “University Exam Scheduling System Using Graph Coloring Algorithm and RFID Technology,” *Int. J. Innov. Manag. Technol.*, vol. 4, no. 1, pp. 66–72, 2013.

[2] E. BURKE, Y. BYKOV, J. NEWALL, and S. PETROVIC, “A time-predefined local search approach to exam timetabling problems,” *IIE Trans.*, vol. 36, no. 6, pp. 509–528, Jun. 2004.

[3] M. W. Carter and G. Laporte, “Recent developments in practical examination timetabling BT - Practice and Theory of Automated Timetabling: First International Conference Edinburgh, U.K., August 29–September 1, 1995, Selected Papers,” E. Burke and P. Ross, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1996, pp. 1–21.

[4] M. W. Carter, G. Laporte, and J. W. Chinneck, “A General Examination Scheduling System,” *Interfaces (Providence).*, vol. 24, no. 3, pp. 109–120, Jun. 1994.

[5] F. Cavdur and M. Kose, “A Fuzzy Logic and Binary-Goal Programming-Based Approach for Solving the Exam Timetabling Problem to Create a Balanced-Exam Schedule,” *Int. J. Fuzzy Syst.*, vol. 18, no. 1, pp. 119–129, 2016.

[6] M. Dimopoulou and P. Miliotis, “Implementation of a university course and examination timetabling system,” *Eur. J. Oper. Res.*, vol. 130, no. 1, pp. 202–213, 2001.

[7] A. Lim, J. Ang, W. Ho, and W. Oon, “UTTSExam□: A University Examination Timetable Scheduler,” *IEEE Intell. Syst.*, pp. 1004–1005, 2001.

[8] M. Malkawi, M. Al-Haj Hassan, and O. Al-Haj Hassan, “A New Exam Scheduling Algorithm Using Graph Coloring,” *Int. Arab J. Inf. Technol.*, vol. 5, no. 1, pp. 80–87, 2008.

[9] D. Sienko, “CASE STUDY An undergraduate uses O . R . to improve final exam schedules at her university 2. Approaches in the Literature to Final Examination Scheduling,” vol. 15, no. 1, pp. 33–38, 2015.

[10] A. M. Wahaishi and R. O. Aburukba, “An Agent-Based Personal Assistant for exam scheduling,” in *2013 World Congress on Computer and Information Technology (WCCIT)*, 2013, pp. 1–6.

[11] B. McCollum, P. McMullan, A. J. Parkes, E. K. Burke, and R. Qu, “A new model for automated examination timetabling,” *Ann. Oper. Res.*, vol. 194, no. 1, pp. 291–315, 2012.

[12] M. W. Carter, “A Survey of Practical Applications of Examination Timetabling Algorithms,” *Oper. Res.*, vol. 34, no. 2, pp. 193–202, Mar. 1986.

[13] T. Khan, A. H. Syed, and A. Hashmi, “An E-Commerce based Web Portal to Support Customize Clothes for Obese People,” *VFAST Trans. Softw. Eng.*, vol. 10, no. 2, 2016.

[14] I. Sommerville, *Software Engineering*. 2010.

[15] T. Khan. Syed, Asif Hassan, “SHPIS□: A Database of Medicinal Plants from Saudi Arabia,” vol. 8, no. 5, pp. 49–53, 2017.