

# E-Appointment Scheduling Using Constraints Logic Programming

N. Ahmad<sup>1</sup>, R. Abdul Hamid<sup>2</sup>, S.N. Abdul Samad<sup>3</sup>, M.N. Mohamad Kahar<sup>4</sup>, A. Mat Safri<sup>5</sup>  
<sup>1-5</sup>Faculty of Computer Systems & Software Engineering,  
University Malaysia Pahang,  
Locked Bag 12, 25000 Kuantan, Pahang, Malaysia.

*Abstract:* E-Appointment Scheduling is a system for students to make an appointment with lecturers in Faculty of Computer Systems & Software Engineering (FCSSE); and Student Medical Center to do the medical check up. The schedules are based on the timetable and university activities. Constraints Logic Programming (CLP) has been implemented to solve the scheduling problems by giving recommendation to the users in part of determining any available slots from the lecturers and doctors' timetable. By using this system, we can avoid wasting time and cost because this application will set an appointment by auto-generated. In addition, this system can be an alternative to the lecturers and doctors to make decisions whether to approve or reject the appointments.

**Keywords:** E-Appointment Scheduling, CLP, PHP, Apache Web Server, software.

## INTRODUCTION

E-Appointment Scheduling is a system for students to make an appointment with lecturers in Faculty of Computer Systems & Software Engineering (FCSSE) and Student Medical Center to do medical check up. It is an online application for FCSSE's student whenever applying to make appointment with lecturers or doctor. All applications have to be sent to the lecturers or doctor for approval. This system will give more interactive for student to make an appointment through an online system. By using this system it can avoid wasting time and cost because this application will set an appointment by auto-generated.

In realistic world, the need for the system arises as a result of problem faced by users in the current manual and online way to make an appointment. Therefore, this system is hopefully to solve problem for scheduling. E-Appointment Scheduling performs the related task associated with students or lecturers and doctor such as:

- a- Student needs to see lecturers or doctor first before make the appointment.
- b- Student needs to fill the form before meet with the doctor.
- c- Consume more time to key in he data into the current system.
- d- Increase error while entering data into current system.

e- The scheduling not effective for doctor and lecturers. The current schedule is in manual and lecturer must keep the schedule in file so it may become lost.

The objectives of this system: (i) to develop a prototype of the e-Appointment Scheduling in order to facilitate making an appointment between students and lecturers at Faculty of Computer Systems & Software Engineering and the doctor at Student Medical Center; (ii) to generate report for all appointments for the lecturers and doctor.

The system is implemented on Microsoft Windows operating systems and develop using PHP tool. PHP: Hypertext Preprocessor [1] is a tool that enabled us to create dynamic web pages. There are lot of advantages of using PHP includes easy learning curve, database integration, extensibility, object oriented programming, scalability and rich features [2,3,4]. It has been compiled with functions to interact with MySQL database system. The system is debugged in the local host server using Apache Server as the platform.

## RELATED WORK

### *UMP Appointment*

The current appointment in UMP is used by students, lecturers and doctor to carry out appointment. In this report explain the important of appointment at UMP meeting and how the system organizes the appointment. This system developed for e-meeting with C# and use for meeting management [5].

### *E-Appointment for National Health Services (NHS)*

Information technology impacts economy. It additionally has started changing the modern way of life, e. g., look at work on the so-called semantic web [6]. An online appointments system is helping to reduce National Health Services (NHS) waiting times. A hospital referral from a Government Policy (GP) is usually followed by a long wait, but under new government regulations this could become a thing of the past as part of the NHS Plan, all NHS trusts must have an online booking system in place by 2005 [7]. Although automated messages have been shown to improve appointment attendance, these studies have not focused on the messages themselves.

Improving the design of appointment messages may further increase the potential of automated systems for improving appointment adherence [8].

### Scheduling

Scheduling is a problem that is grounded in many different levels of computer science and computer hardware engineering. Various scheduling and sequencing problems have been addressed since the 1950's by researchers in computer science, operations research and discrete mathematics [9]. In a business-computing context, scheduling implies the automatic execution of background tasks (batch jobs) at pre-set points in time (e.g. every day at 8pm, midday on Wednesday). 3 types of scheduling are distinguished: native, basic and advanced scheduling [10].

### Constraint Logic Programming

Constraint Logic Programming (CLP) is a merger of two declarative paradigms: constraint solving and logic programming. Although a relatively new field, CLP has progressed in several quite different directions. In particular, the early fundamental concepts have been adapted to better serve in different areas of applications. In this survey of CLP, a primary goal is to give a systematic description of the major trends in terms of common fundamental concepts. The three main parts cover the theory [8].

#### TECHNIQUES USES IN E-APPOINTMENT SCHEDULING

Technique that use in e-Appointment Scheduling is Constraint Logic Programming. In this technique, using logic programming can write as logic programs that suitable with scheduling has been developed in this system. It solves some problems that are more natural or simple to write as logic programs, while some are more natural to write as constraint programs [8].

The constraint programming approach is to search for a state of the schedule in which a large number of constraints are satisfied at the same time. A problem is typically stated as a state of the schedule containing a number of unknown variables. The constraint program searches for values for all the variables. Constraints for lecturers and doctor for are defining as below:

- a) 5 constraints for lecturers based on higher priority:
  - i. Emergency leave
  - ii. Meeting
  - iii. Fix time table
  - iv. Appointment with student
  - v. Fix time office hour
- b) 5 constraints for doctor based on higher priority :
  - i. Emergency call
  - ii. Emergency leave
  - iii. Outside work
  - iv. Appointment with student
  - v. Fix time office hour

Algorithm that use in e-Appointment Scheduling is using Indigo algorithm as a guide line that can be used for scheduling in e-Appointment Scheduling. This algorithm needs lecturer and doctor list all the activities as constraints for system setup of the schedule. The algorithm is based on Fig. 1.

```

procedure Indigo(H: constraint hierarchy)
all_constraints <- list of constraints in H,
strongest first;
all_variables <- empty;
active_constraints <- empty;
for each v in all_variables do
  initialize v.bounds to unbounded;
endfor
/*This coding represents to list all the constraints for schedule
and rate the constraints from high to low constraints.*/

for current_constraint in all_constraints
do
  tight_variables <- empty;
  queue <- empty;
  queue <- queue + current_constraint;
  while queue not empty do
    cn <- queue.front;
    /*This coding represents for swap the current constraints
with new inputs that entered and also for specify the priority
of the constraints.*/

    tighten_bounds(cn,queue,tight_variables,
active_constraints);

    check_constraint(cn,active_constraints);
    queue.dequeue;
  endwhile
endfor
end Indigo
/*This coding represents for validate the constraints and update
the list of constraints priority.*/

```

Fig. 1. Algorithm in Constraints Logic programming

#### METHODOLOGY

Methodology in this system development is a standard process followed in an organization to conduct all the steps necessary analyze, design, implement, and maintain information system [11]. The development of the project usually follows a life cycle. To develop an E-Appointment Scheduling, System Development Life Cycle (SDLC) has been selected as the guide to develop the project. SDLC is an overall process of developing information systems through a multi step process from investigation of initial requirement through analysis, design, implementation and maintenance. In SDLC it is possible to complete some in one phase in parallel with some different activities from another phase and sometimes the life cycle is iterative so that the SDLC was choose because of the systematic flow and the circular process in the SDLC which is in the end of every phase will use as a reference for next phase. In this project, the six phases in the SDLC will be use for solving the entire problem. The six major phase that use are project identification, planning, analysis, design, development and implementation.

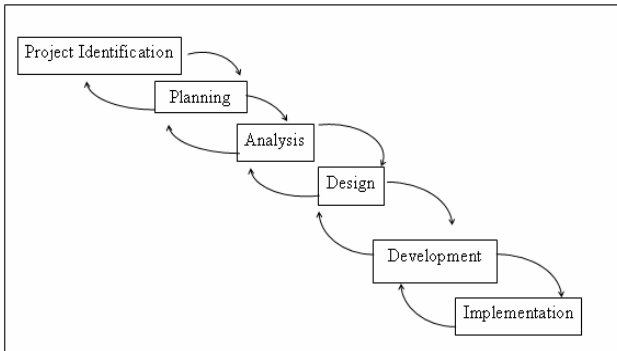


Fig. 2. System Development Life Cycle (SDLC).

*System Design*

The system design considers from input and output screen to reports, databases and computer processes. We also must provide the physical specifics of the system that have designed, either as a model or as detail documentation, to guide those who will build the new system. Before the development of the e-Appointment Scheduling even started, a design of the system is done to provide better and clear understanding about the process flow of the system. Unified Modeling Language (UML) is a tools developed to helps understanding the system.

*Use Case:* Use case is one of the steps to show what the user can do it in this system. For this system have two use cases that are create for every interaction between users. The use cases below are design to applicant, lecturer and doctor. For use cases between applicant and lecturer, it appears what applicant and lecturer can do it. The applicant can login, fill application form and receive approval application, view applicant info and check schedule. Lecturer can receive the application form, make approval application, and view applicant info, check schedule, edit schedule info and setup schedule. Fig. 3 shows uses case for student and lecturer. The uses case between applicant and doctor also same with use case between student and lecturer as shown in Fig. 4.

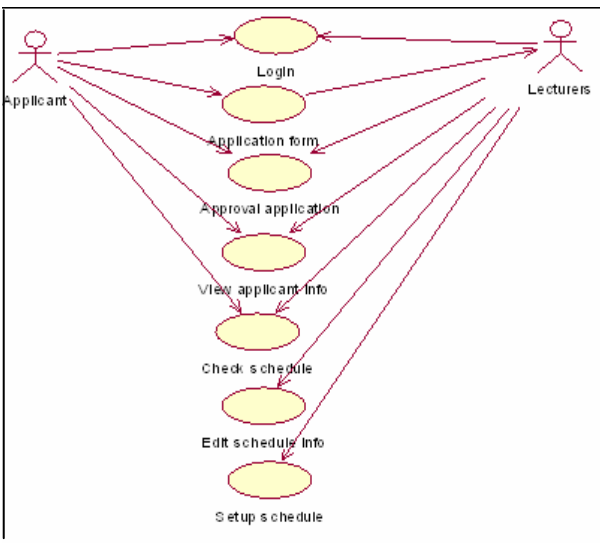


Fig. 3. Use case Specific e-Appointment (Lecturer).

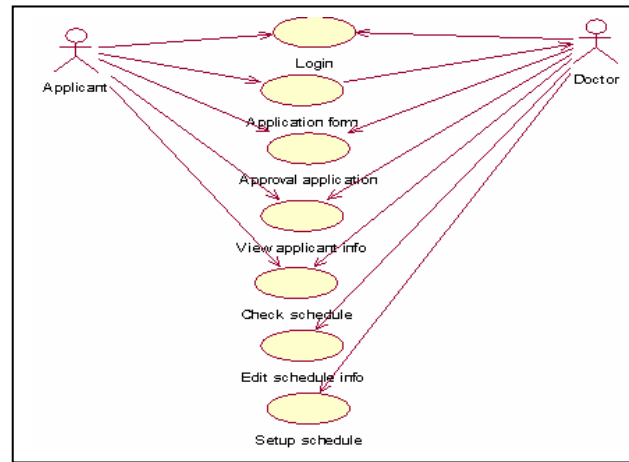


Fig. 4. Use case e-Appointment Scheduling (Doctor).

*General Sequence Diagram:* Applicants login the system and make appointment either with lecturer or doctor and submit the application form. System generates application for approval from lecturers or doctor. Fig.5 shows the general flow of the e-Appointment scheduling operated.

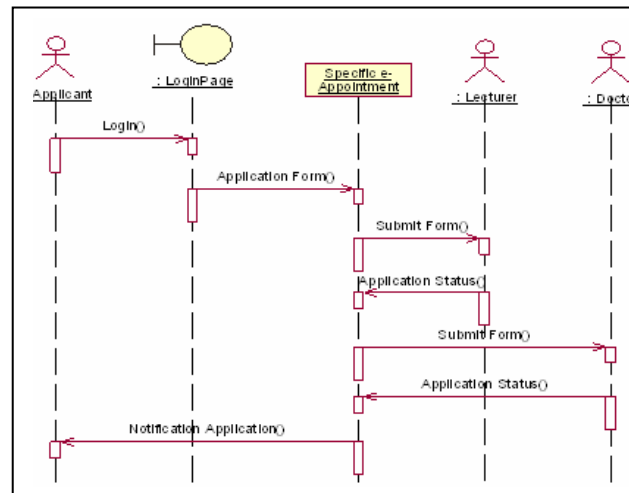


Fig. 5. General Sequence Diagram.

*Data Dictionary*

In e-Appointment Scheduling used four tables for database it is are:

Table 1: Lecturer's table

Variable	Declaration	Key
STUDENT_ID	VARCHAR(10)	PK
NAME	VARCHAR(30)	
LOCATION	VARCHAR(30)	
AGENDA	VARCHAR(50)	
DATEE	DATETIME	
TIMES	TIMESTAMP	
TIMET	TIMESTAMP	

Table 2: Search lecturer's table

Variable	Declaration	Key
LECT_ID	VARCHAR(10)	PK
NAMEL	VARCHAR(30)	

Table 3: Doctor's table

Variable	Declaration	Key
STUDENT_ID	VARCHAR(10)	PK
NAME	VARCHAR(30)	
LOCATION	VARCHAR(30)	
PROBLEM	VARCHAR(50)	
DATEE	DATETIME	
TIMES	TIMESTAMP	
TIMET	TIMESTAMP	

Table 4: Schedule table

Variable	Declaration	Key
NO	INT	auto_increment (PK)
DATEE	DATETIME	
TIMES	TIMESTAMP	
TIMET	TIMESTAMP	

RESULTS AND DISCUSSION

Results for Student Module

This module, follow the objective and can run on browser. Student can view availability of lecturers and doctor before make an appointment. If student choose to make an appointment with lecturer, student must search that lecturer by lecturer name, date and time. System will view the available slot that student want, if not system will suggest other slot to make an appointment. Then make the appointment through memo and the request has been triggered to that lecturer. It also same with the doctor module but the appointment only triggered to doctor. At student page, for the first time login into the system all student allow to use their student id as and username and password.

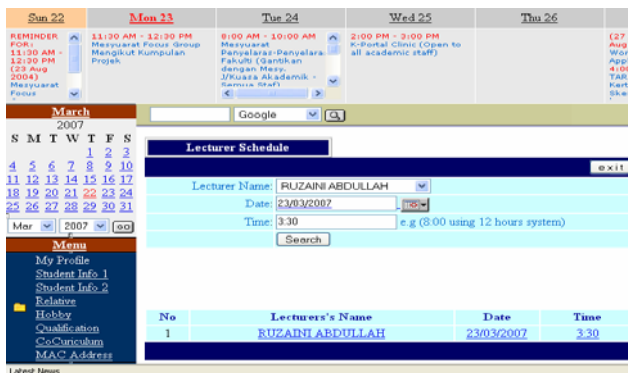


Fig. 6. Search Lecturer Schedule

Then, the system will show for search lecturer's schedule or doctor's schedule with insert the constraint by student to do the appointment. System will search the available slot that student need or give other suggestion available slot if the constraints not match. Students just click the result to do the appointment.

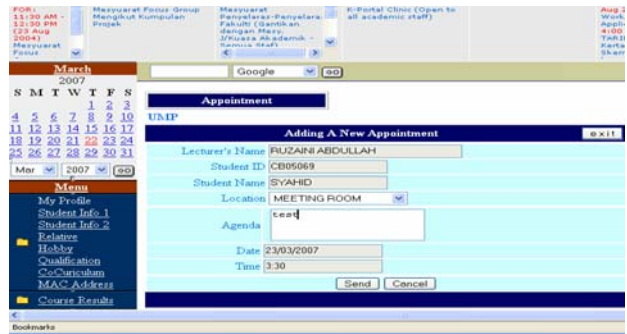


Fig. 7. Make an Appointment

Interface above will appear to make an appointment with lecturer or doctor, which student want date and time that generate from the previous interface. Student must insert location and agenda for the appointment; and click button send or cancel or exit to cancel the appointment.

Results for Lecturer Module

After students make an appointment, database is updated. The lecturer can view the request. Every appointment that received by lecturers is based on available slots. Lecturer can reject or edit the time and date to make other appointment when have emergency. Edit schedule can make up of lecturer to update the available schedule. If lecturer wants to check the record can click button report and can view the record.



Fig. 8. List of the Approved Appointment

This interface appears to show the approved appointment that make by system automatically. Lecturer can view the information about the application at this phase and also can view the detail of the applicant by click image view at column detail.

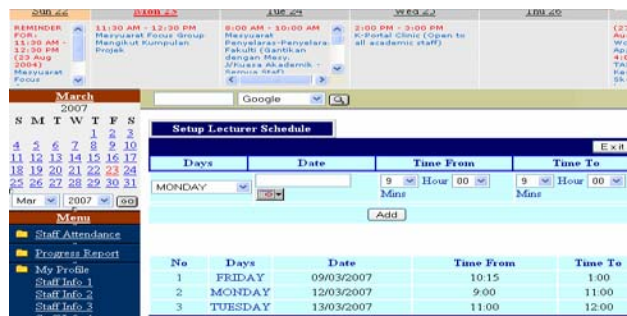


Fig. 9. Lecturer Setup Schedule

Lecturer can setup the schedule at the available time for the appointment. Lecturer can change the date and time for the appointment that student booked as shown in Fig. 10. System automatically update the appointment and sends it to the student through the memo.

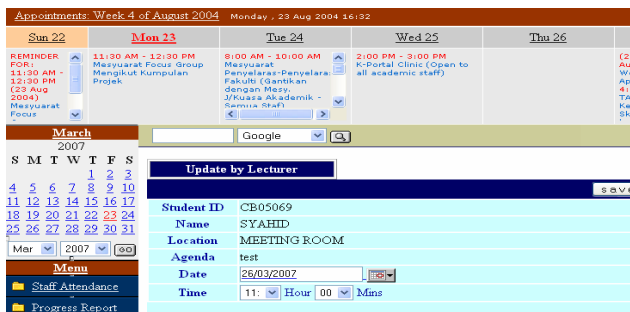


Fig. 10. Lecturer Change Date and Time

### Results for Doctor Module

After students make an appointment, database is updated and doctor can view the request. Every appointment that received by doctor are approved because the system only approve the appointment at available slot but doctor can reject or edit the time and date to make other an appointment when have emergency. Edit schedule can make up of doctor to update the available schedule. If doctor wants to check the record can click button report and can view the record.

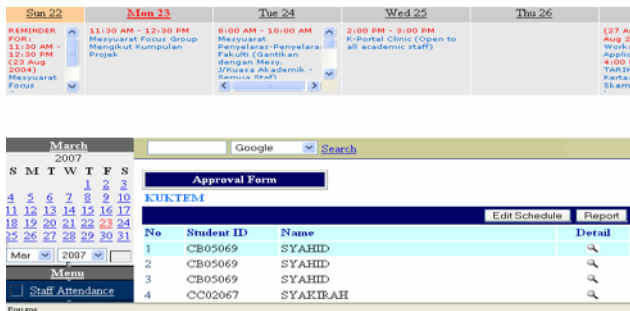


Fig. 11. List of the Approved Appointment

Fig 12 shows the approved appointment that make by system automatically. Doctor can view the information of application and the detail of the applicant by click image view at column details.

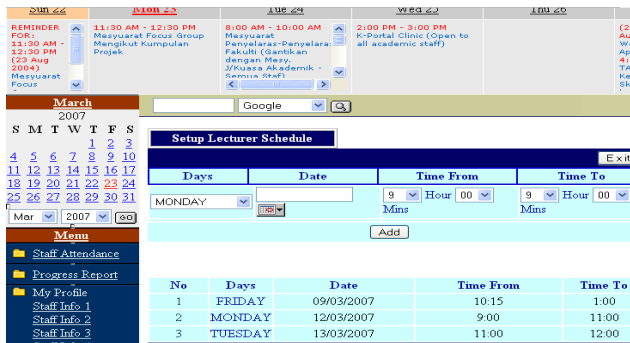


Fig. 12. Doctor Setup Schedule

Doctor can setup the schedule at the available time for the appointment. Doctor can change the date and time for the appointment that student booked as shown in Fig.13. Hence, system automatically update the appointment and sends it to the student through the memo.



Fig. 13. Doctor Change Date and Time

### CONCLUSIONS

In this system, Constraint Logic Programming (Indigo algorithm) has been used as a guide line for the scheduling in e-appointment. The algorithm automatically list all need the activities of lecturers and doctors as the constraints for system setup in the schedule. By using this system, student can make an appointment with lecturers and doctor in Student Medical Center easily. Student can get suggestion for available slot to make an appointment. Lecturer and doctor department can update their schedule also can give feedback to reject the appointment if have emergency. This system can be an alternative to the lecturers and doctors to make decisions whether to approve or reject the appointments.

### REFERENCES

- [1] F. Benhamou and A. Colmerauer, *What is Constraint (Logic) Programming*, Sept 1993.
- [2] <http://vsbabu.csoft.net> - 2001
- [3] <http://www.newbienetwork.net> - 2002
- [4] <http://zend.com/zend/tut/calendr-ver7f.php> - 2001
- [5] N.M. Nik Ismail, *E-Meeting With C#*, PSM from BCN for guideline, Version: 2004.
- [6] B. Lee, *Story Boarding for Web-Based Information System 1*. Reed Elsevier Business Publishing Ltd. National Health Service online appointment system, Karl Cushing, 2001.
- [7] C.Dudman, *Government Activity*, COPYRIGHT2001 Reed Elsevier Business Publishing Ltd., 2001.
- [8] Leirer, Morrow, Parinute & Doksum, *Appointment system*, 1980.
- [9] P.F.Dutot, G Mouni'e, D.Trystram, *Operating Systems Internals and Design Principles (fifth international edition)*. Prentice Hall, 2004.
- [10] N. Kaushik and S. Figueira, 2005. *Scheduling in e-business*. Dollar Universe United State of America.
- [11] J.A Hoffer, J.F. George, J.S Valancich, *Modern System Analysis & Design*. Third Edition. United State of America: Prentice Hall, 2002.