Kongres Pengajaran dan Pembelajaran UKM, 2010

Final Year Supervision Management System as a Tool for Monitoring Computer Science Projects

Marini Abu Bakar* Norleyza Jailani, Zarina Shukur, Noor Faezah Mohd Yatim

Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Malaysia

Abstract

In Computer Science Program run by Universiti Kebangsaan Malaysia, students are required to develop a software prototype and write a dissertation for their final year project. Preliminary observation has shown that for the last five years, an increasing number of students failed to complete prototype development within the allocated time. Current practice uses a log book to document meetings and discussions between supervisors and students. Although a detailed schedule has been proposed, there is no monitoring process to trigger specific actions to make sure all deadlines are met. To improve project monitoring and supervision, a prototype of a web-based supervision management system was designed and developed. The initial prototype consists of three modules, namely user profile, project monitoring (of software development and report writing) and appointment setting. Once the prototype is completed, we plan to conduct a user acceptance test.
Selection and/or peer-review under responsibility of Kongres Pengajaran & Pembelajaran UKM, 2010
Keywords: Computer Science final year project; Project supervision; Project monitoring system; Project management system; Software development;

1. Introduction

   Universiti Kebangsaan Malaysia (UKM) is one of the four universities in Malaysia that has been awarded research university status. To inculcate research skills for the students, research elements need to be incorporated into teaching and learning at all levels including undergraduate studies (Jailani et. al. 2010). To fulfill this objective, final year project is an appropriate avenue to introduce research to the undergraduates, since students are required to undertake research in a specific domain and develop software prototype as partial requirement for their final year project. Students are also required to write a dissertation as part of the course requirements. Preliminary observation has shown that for the last five years, an increasing number of students failed to complete their final year project within the allocated time. Although the role of the final year project in Computer Science curriculum is significant, some students (and also supervisors) fail to see its importance. The quality of the prototypes developed has also decreased. This phenomenon is not an isolated case for UKM alone but has been perceived in other universities in the United Kingdom (Bouki 2007).

---

* Corresponding author. Tel.: +0-603-8921-6183; fax: +0-603-8925-6732.
E-mail address: marini@ftsm.ukm.my.

A survey has been conducted to gather information on supervisors' current practices and problems occurring during the supervision through a set of questionnaire (Marini et al. 2010). Findings from this survey was analysed, and a detailed schedule was suggested. Based on this survey, a conceptual framework for the Web-based Supervision Management System was produced in accordance with the process currently executed in the Computer Science Department, Universiti Kebangsaan Malaysia. This paper provides a brief analysis on the current process practiced by the department, details the specification and system design, as well as presents the initial prototype system that has been developed.

2. Analysis of the Current System

Project management plays an important role in coordinating the planning, design and development processes of a software project. Detailed planning and monitoring from the early stage throughout the end would ensure that a project could be completed within the allocated time, while at the same time achieved its objectives. Existing methods used in scheduling projects take up a lot of time. This is especially difficult when a lecturer is supervising more than 10 students at the same time.

The process of managing final year projects involves 3 parties: students, supervisors and the head of department office. The head of department assigns a lecturer to supervise each student. It is then the student's and supervisor's responsibility to monitor the progress of system development and report writing. Methods or procedures involved in the monitoring process may vary between each supervisor. Supervisors also act as the mediators between the students and the head of department office.

Currently, log books are used to monitor the progress of students' projects and record all activities during supervision. Students are required to meet their supervisors at least 5 times during the project, and each meeting has to be recorded in the log books and signed by the supervisors. The disadvantage of the log book is that it only focuses on the progress of report writing. There is no indicator for students to fill up information on the progress of their software development, hence neglecting the process of monitoring software development progress. Students end up with a semi-finished dissertation with very little information pertaining to implementation and testing of the system, software or algorithm since they have not actually finished the software development.

The head of department office is the center of project management which is responsible for assigning a supervisor to each student, preparing and providing project schedules, collecting project proposal documents, scheduling and arranging project presentations, collecting the final project dissertation, and recording the students' grades into a student management system called *Sistem Maklumat Pelajar (SMP)*.

Based on the findings from the survey conducted, there is a need for a system to help manage the supervision of final year projects. An effective project monitoring system requires web-based tools to ease communications among all parties involved.

The initial system consists of 3 main modules: appointment, students' and lecturer's profiles, and schedule monitoring. The appointment module is used to arrange the time for meetings between a student and his/her supervisor. In this module, a student could suggest the time for a meeting which is then confirmed by the supervisor if agreed by both parties. A supervisor can also suggest particular time slots according to his/her own timetable.

The student's and lecturer's profiles module display the latest information on the students and lecturers. Students can update their personal information, academic, and other related information based on their supervisors' requirements. Similarly, lecturers can update their personal, academic and other information to be referred by their students.

The schedule monitoring module is the main module in this system. This module is used to support the monitoring process and detect whether a project is delayed or on time. Schedules are provided as Gantt charts and can be accessed according to the category of the tasks. Each project's schedule consists of 2 parts: system development and report writing. This module also provides mechanisms to trigger important deadlines determined for every stage in the system development process.

3. System Specification

Collofello (1999) suggested that in order to have an effective communication during software development project monitoring, web-based support tools should be provided. Many legacy information and database systems have been transferred to a web based environment so that they can be accessed through the internet (Murugesan et al. 2001). New technologies such as Google map, Adobe Flash Player, Java applet, and others are hooked into web application to facilitate the information sharing activities. Consequently, the web-based application design, development, operation and maintenance become more complex and challenging. Therefore, in order to support the systematic process and approach in web application development, a new discipline which is known as web engineering has been introduced. The features of current web-based applications itself has turned web engineering to become a multidiscipline body of knowledge which covers human computer interaction, software engineering, multimedia, user interface, system design and analysis, requirement engineering, information engineering, testing, modeling and simulation, as well as project management (Murugesan et al. 2001). Most web applications are dynamics where new changes in content, functions, structure, presentation or deployment are needed when the systems start to be used. The changes in web applications are more frequent compared to the traditional software applications. Therefore, it is impossible to determine the complete requirement at the beginning of the development. By considering these respective factors, this study have taken the approach of software prototyping to allow the main stakeholder for the system to test drive designs, by providing a similation of some behaviour and functionality (reynardthomson.com 2011). Specifically, we have chosen interactive prototype instead of using wireframes or visual prototypes as it is far more useful, though require an increased investment in time to create (Perine 2002). Interactive prototypes aim to model a system design more faithfully, and represent actual paths through that system. They generally will combine the visual aspects of a static prototype with a certain degree of interactive functionality. This might mean navigation, or the use of real web controls, or even mock data processing. As a platform for demonstrating a system, these are the richest, most useful types of prototype, although the slowest to create (reynardthomson.com 2011). Prototyping also allows us to capture new user requirements after test drive by system stakeholders. Hence, new functions can be developed and integrated as the system prototype is revised.

The conceptual design of the Final Year Supervision Management System proposed in this paper is constructed based on two studies. First, we examined the main component and functional of project management software project that are available in the market (TaskFreak 2010, HiTask 2010). Second, we revised the prototype of the Communication System between Student and Supervisor (Leong 2006) and Web-based Project Management System (Goh 2009) that were developed as final year projects by our previous students. To enhance the previous analysis as well as an ineffective execution of manual schedule monitoring, a few important modules have been identified. Users for the system are students, supervisors and head of department or administrators. The modules are:

1. Appointment module
    a. Appointment setting.
    b. Supervisor can set weekly timeslot for the whole semester and also unplanned time slot.
    c. Student can request for appointment timeslot and the supervisor can verify the time.

2. Students and lecturers profile module
    a. Records the most updated student's information such as the phone number and email address.
    b. Student can update the information.
    c. Student can input academic information such as grade for certain subjects, list of enrolled subjects in that semester and description of projects completed during their industrial training.

3. Schedule monitoring module
    a. Administrative officer can set a standard schedule for system development and report writing activities. However, supervisors can modify the schedule based on the project requirements or personal preferences.
    b. Supervisor can monitor student's progress whether it is on time or delayed. Different colour codes will be used to represent the early, on time or late process completion.

    c. Status of all students under a lecturer's supervision is displayed in a single screen so that the overall progress can be viewed.

    d. Notification messages are sent through email to remind students and lecturers on important dates. Students will be alarmed when the deadline is fast approaching, turned up or passed.

    e. Generates report that lists the overall student's status for supervisor reviewing and administrative purposes.

4. Log book module

    a. Logs the discussion during meetings.

    b. Provides separate areas or layout for entries regarding system development process and writing.

    c. Is able to upload reference documents.

    d. Is able to upload report documents for evaluation and reference.

5. Administrator module

    a. Managed by department officer.

    b. Is able to assign students to supervisors.

    c. Sets the system development schedule and chapters writing.

    d. Records the submissions of document to the department office.

By clustering the system functions into modules, we hope that designing of processes and system implementation will be easier. Any requirements for new functionalities to be integrated into the existing modules will be done after the system is being tested and evaluated by the stakeholder.

3. System Design

Figure 1 shows the context diagram of Final Year Supervision Management System. Context diagram is a level 0 of data flow diagram which illustrates the overall system. It also defines the scope and boundary between the system and an external entity. This diagram also shows the input and output flows of data to and from the system and external entities.
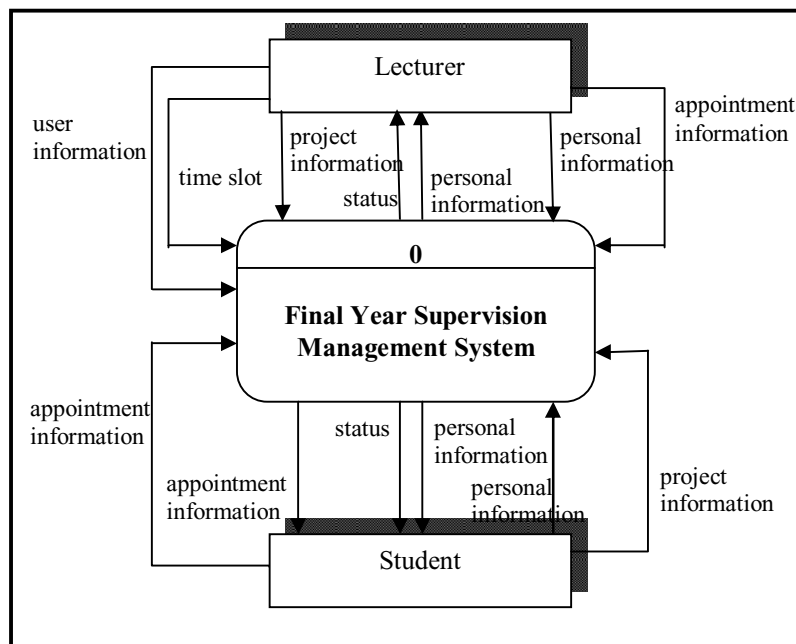
Figure 1 Context Diagram of Final Year Supervision Management System

In the design for the early prototype system as illustrated by Figure 1, only two categories of entities or users are included. The first entity is Lecturer who plays a role of a supervisor and the second is Student. Another entity namely, the Administrator which represents the department office will be included in future work. Hence, dividing system functionalities according to modules is an appropriate way of handling design and implementation tasks so that integrating future module for the Administrator into the system can be done in the next prototype. The following section will display the prototype system that has been developed so far.

4. System Prototype

The development of the system itself is carried out as a final year project (Izzul Sayyidi 2010). The system was written in PHP (hypertext preprocessor), the user interface was designed using Macromedia Dreamweaver CS3, and MySQL was used as the database system. phpMyAdmin was selected as an administration tool as it provides simple management of PHP applications and MySQL administration tools. The developer tried to adopt Human-Computer Interaction theories in the process of designing the user interface. Graphical user interface components are used to provide good navigation and easy access to frequently used data. User guide is also provided as references to system users.

At present only two user categories of the system has been implemented, namely for lecturers and students. The users have different menu in their interfaces, however the graphical user interfaces for the same functions are almost similar. Figure 2 shows the interface for managing project in the lecturer module. It lists down all students under her supervision. The lecturer can also create a new project if she is assigned a new student.

Figure 2 Project management interface for lecturer/supervisor

Figure 3 shows the interface for the project progress of a particular student. The supervisor can monitor a student's progress based on the tasks listed which can be divided into software development activities and dissertation writing. Status for each task is displayed in colour coding based on its progress. The colour codes used are white which indicates that the task has not began, yellow indicates the task is in progress, green indicates that the task is completed within the time limit allocated, and red indicates that there is a delay in the task completion. A lecturer is able to inspect the documents uploaded by his/her student and write remarks and comments on every listed task related to the project. The advantage of this feature is that via a single interface the lecturer is able to determine whether a student is behind schedule or otherwise. Nevertheless this feature can be improved by designing more intuitive indicators.

Pembangunan Sistem	Status Skedul	Tarikh Tamat	Sebelum/Selepas Dari Tarikh Tamat (Hari)	Tarikh Siap	Awal/Lewat Dari Tarikh Siap (Hari)	Serahan Dokumentasi	Muat Naik/Turun Dokumentasi	Komen	Hapus Tugas
Sorotan Kesusasteraan/ Kajian Sistem Semasa	Tamat	10/06/2010	-15	09/06/2010	1	* Borang usulan projek.	Muat Naik/Turun Dokumen	Comments[0]	Remove
Spesifikasi Sistem	Sedang Berjalan	28/06/2010	3	00/00/0000			Muat Naik/Turun Dokumen [1]	Comments[0]	Remove
Rekabentuk Antaramuka/ Skrin	Belum Bermula	01/07/2010	6	00/00/0000		* Bab 2	Muat Naik/Turun Dokumen	Comments[2]	Remove
Rekabentuk Sistem (ERD, DFD dll)	Lewat	07/06/2010	-18	00/00/0000			Muat Naik/Turun Dokumen	Comments[0]	Remove
Memahirkan Diri Dengan Persekitaran Pembangunan	Belum Bermula	14/06/2010	-11	00/00/0000		* Bab 3	Muat Naik/Turun Dokumen	Comments[0]	Remove
Pembangunan Sistem Utama (Versi Alfa)	Belum Bermula	21/06/2010	-4	00/00/0000			Muat Naik/Turun Dokumen	Comments[0]	Remove
Pengujian Sistem Dan Penambahbaikan (Versi Beta)	Belum Bermula	28/06/2010	3	00/00/0000		* Bab 4 dan 5. Draf Lengkap	Muat Naik/Turun Dokumen	Comments[0]	Remove
Penulisan Laporan	Status Skedul	Tarikh Tamat	Sebelum/Selepas Dari Tarikh Tamat (Hari)	Tarikh Siap	Awal/Lewat Dari Tarikh Siap (Hari)	Serahan Dokumentasi	Muat Naik/Turun Dokumentasi	Komen	Hapus Tugas
Bab 2 - Sorotan Kesusasteraan	Belum Bermula	05/07/2010	10	00/00/0000		Bab 2	Muat Naik/Turun Dokumen	Comments[0]	Remove
Bab 3 - Spesifikasi Sistem	Belum Bermula	12/07/2010	17	00/00/0000		Bab 3	Muat Naik/Turun Dokumen	Comments[0]	Remove
Bab 4 - Rekabentuk Sistem	Belum Bermula	19/07/2010	24	00/00/0000		Bab 4	Muat Naik/Turun Dokumen	Comments[0]	Remove
Bab 5 - Implementasi Dan	Belum	26/07/2010	31	00/00/0000		Bab 5	Muat Naik/Turun		

Figure 3 Project status interface

Another important function to facilitate supervision of final year project is the ability to set appointment. Lecturers are given the privilege to allocate specific time for supervising final year students. Figure 4 shows the interface for a lecturer to allocate appointment time slots. In normal circumstances, the students can only set appointments based on the allocated time slots. However in isolated cases students are allowed to request appointments from their lecturers at other times depending on the availability of the lecturers. Figure 5 illustrates the interface that lists down all appointments made by students with a specific lecturer. It is up to the lecturer to approve or reject a particular appointment.
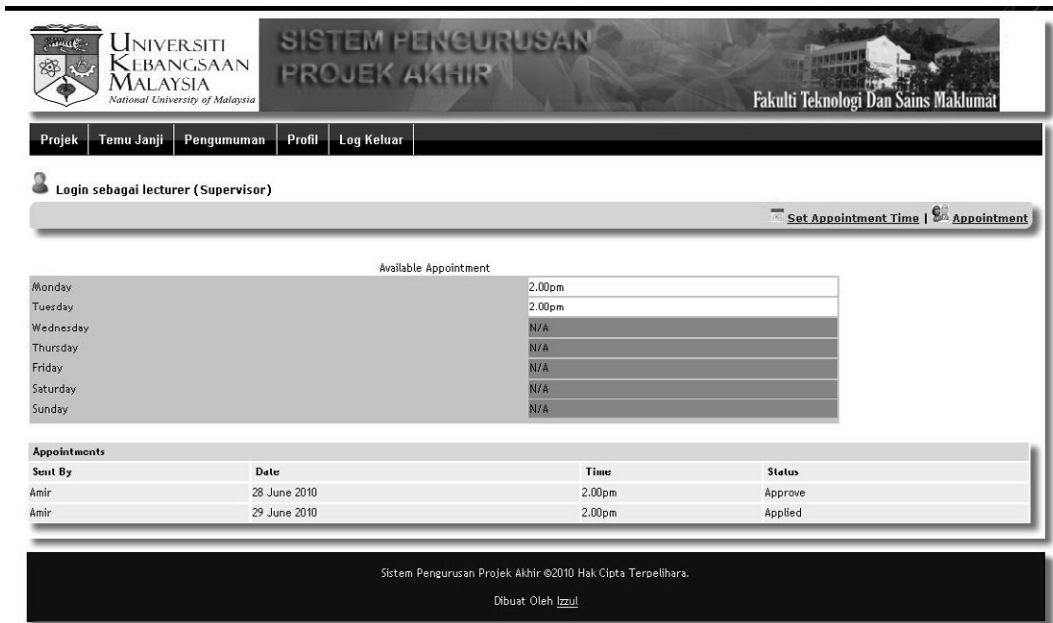
Figure 4 Appointment time allocation interface
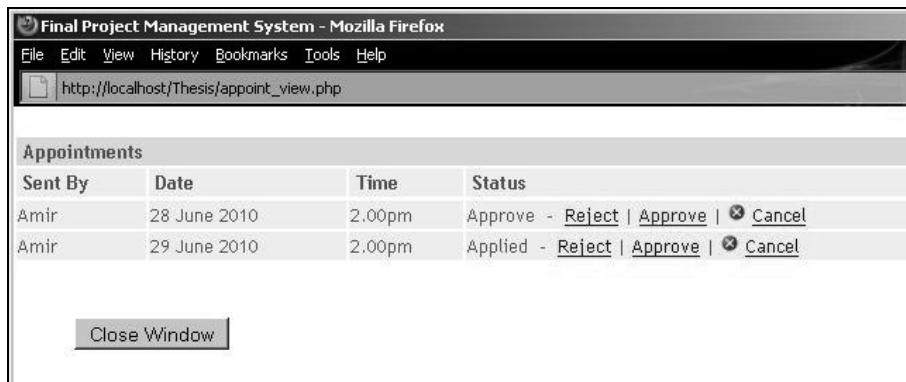
Figure 5 Interface to approve or reject appointment

From the student's perspective, he/she is able to view the project tasks list and also the progress of each task. The interface is similar to the one that can be viewed by their supervisor. The tasks are divided into two sections which are the prototype development tasks and the report writing tasks. Color coding is also used as indicator of the task progress. Students can also upload documents related to their project to be reviewed by supervisor. Figure 6 illustrates the interface for student's project status.

Figure 6 Project status interface for student

Figure 7 shows an interface for students to read comments or remarks posted by the lecturer. Students can also respond to the comments. The comments are associated to each task and are listed in the project schedule.

Figure 7 Interface for Student-Lecturer communication

5. Conclusion

The prototype of the students and lecturers profile module, schedule monitoring module and appointments module have successfully been developed. Other modules such as the log book module are currently under development while the management module will be intergrated into the system in the next prototype version. Testing and evaluation will be carried out once these modules are integrated. We plan to test drive the prototype by inviting a few lecturers and students to have a feel of using the system. Should we discover any new requirements suggested by these stakeholders, the prototype will be revised for further improvement.

6. Acknowledgement

References

Bouki, V. (2007). Undergraduate Computer Science Projects in UK: What is the point?, *Proceedings of the Informatics Education Europe II Conference*, 29-30 November 2007, Thessaloniki, 176-184.

Collofello, J.S. (1999). Monitoring Team Progress in a Software Engineering Project Class. *29th ASEE/IEEE Frontiers in Education Conference*. 10-13 Nov 1999. Puerto Rico.

Goh C. C.. (2009). Sistem Pengurusan Projek Akhir Berasaskan Web. Projek Prasiswazah. Universiti Kebangsaan Malaysia.

HiTask (2010, December). Retrieved from http://hitask.com.

Izzul Sayyidi, H. (2010). Sistem Pengurusan Penyeliaan Projek Akhir Jabatan Sains Komputer. Projek Prasiswazah. Universiti Kebangsaan Malaysia.

Jailani, N., Yatim, N. F. M. & Marini A. B. (2010). Pengurusan penyeliaan projek akhir bersifat penyelidikan: pengalaman dan pencapaian. Bab dalam buku *Penilaian Prestasi Pendidikan di Institusi Pengajian Tinggi*. (pp 19-38). Penerbit Universiti Kebangsaan Malaysia. Bangi.

Leong S. M. (2006). Sistem Komunikasi Antara Pelajar Dengan Penyelia. Projek Prasiswazah. Universiti Kebangsaan Malaysia.

Marini A. B., Norleyza J., Zarina S., & Noor Faezah M. Y.. (2010). Spesifikasi dan Rekabentuk Sistem Pengurusan Penyeliaan Projek Akhir Sains Komputer, Bab dalam buku Mercu Kesarjanaan Pengajaran dan Pembelajaran: Penyelidikan Tindakan di Institusi Pengajian Tinggi. (pp 16:225-237). Penerbit Universiti Kebangsaan Malaysia. Bangi

Murugesan S., Deshpande Y., Hansen S., Ginige A. (2001). Web Engineering: A New Discipline for Development of Web-Based Systems, *WebEngineering 2000*, Lecture Notes in Computer Science, 2016 3-13, Heidelberg: Springer-Verlag.

Perine, C., (2002). Interaction design prototyping of coomuncator devices: towards meeting the hardware-software challenge, *Interactions Magazine.* vol 9 issue 6, pp 36-46. ACM, New York.

TaskFreak!. (2010, December). Retrieved from http://www.taskfreak.com.

Reynard Thomson, What Is Software Prototyping? (2011, January). Retrieved from http://www.reynardthomson.com/what-is-prototyping.html