

SoftProMT: A Prototype Software Management Tool

Hanan Ettaher Dagez

Software Engineering Laboratory

Faculty of Computer Science and Information Technology

University of Malaya

50603 Kuala Lumpur

Khairuddin Hashim

Software Engineering Laboratory

Faculty of Information Technology

Universiti Tun Abdul Razak – The Virtual University

Pekeliling Business Centre

53000 Kuala Lumpur

E-mail: hanan@fsktm.um.edu.my, khairuddin@unitar.edu.my

ABSTRACT

Developers in software engineering project management are still searching for new features of tools to help improve the management of the development process, minimize risk and problems using new techniques. The tools support project managers in their responsibilities to deliver products on time, within budget and schedule. This paper presents new concepts and adds new features for software project management to help project managers manage, monitor and control projects to reach the target of low cost and high quality products.

KEYWORDS

Software, project management, analysis, availability, reusability, methodology, technology, and tracking.

INTRODUCTION.

Project management is divided into two major phases: planning a project and managing and controlling it [1]. Poor management can increase software costs and reduce the quality of the projects [2]. It is evidently clear that there is need for effective software project management and tools to support the managers to perform their responsibilities. Project management tools allow the managers throughout the organization to see what is happening and understand the impact of not maintaining their schedule providing better data to make better decisions. Furthermore, it save a lot of time spends on scheduling, calculating duration, evaluating cost, and producing reports in standard format. There are many project management tools; Most of them focus on: Work Break down Structure (WBS), Gantt chart, resources driven scheduling, calculates the activity duration (based on the resources assignment and the effort factor), display information and generates progress reports. However, the available software project management tools are inadequate to cover all of the software project's needs and problems. There are many statistics related the failure in software project to weakness in project management. Therefore, this paper attempts to highlight other software project management aspects which uncovered well in the available tools by introducing the main features of a prototype tool (SoftProMT) for software project management to reduce problems and to ensure project quality.

TOOL FEATUERS.

1. Analysis of old projects.

Japanese management is considered one of the most successful management in the world due to their concern in investigating the reasons for poor quality and their focus on improving it [3]. In the planning process, consulting the historical information for activities cost estimation, effort productivity, and related problems are very important. Pressman reported that those who cannot remember the past are condemned to repeat it [4]. For improved software cost estimation, the organization can get the management's support by making the manager aware of typical estimation problems and methods available to solve or avoid these problems [5]. One of the most important factor is to avoid areas where problems arise. Statistical quality assurance based on past projects can help enormously in identifying problem areas and resources [4]. Therefore this tool will implement this concept as in the following:

- (a) Displaying the problems in both current and past projects.
- (b) Displaying the problems faced due to changes in requirements.
- (c) Present the most critical problem due to changes in requirements, and in addition it will also show how frequents it occurs.
- (d) Determine the most critical phase of software development process in both past and current projects. It will also display the most critical problem and all information related to it such as in which project it occurred, who is/are the assigned people, and the authorized project managers.
- (e) Analyze and display all problems occurring in different specific phases such as, planning, requirement, specification, design, testing, coding and others. For each

selected phase, the problems will be analyzed and sorted based on the most to the least critical problem.

The tool will present all the information about each problem such as the factors responsible for triggering the problem and which projects have been affected by this problem. The user will get this information each time he/she accesses the option to see the problems.

2. Analyze and compare selected projects.

In this issue, the tool will help the project manager to analyze some projects, which might be related to each other or in the same field it is done by displaying the following:

- a- All problems and the responsible person(s) affecting the selected projects and highlighting the problems for each selected project.
- b- Similar problems and similar person(s) responsible for causing the problems as seen in Figure 1.
- c- Problems and person(s) responsible for changes in requirements.
- d- Problems and the responsible person(s) for the selected projects in specific phase.
- e- Similar problems in specific area.

3. Assign personnel.

It is obvious that any software system is dependent on the competency of the people who build it. To develop software project successfully, we need a number of different technical skills such as analysis ability, system design, program design, coding, integration and testing. Therefore the project manager's responsibility is to ensure that the right person is assigned for every task. For this reason, the following features will be provided in the tool to help the project manager assign the right person to the right position:

- (a) **Experience;** Assign people based on their years of experience such as non-experience, 1-5, 5-10, 10-15, 15-20, and over 25 years.
- (b) **Qualification;** Assign people based on their qualification such as undergraduate, graduate, postgraduate, and others.
- (c) **Reusability;** Reusability is to achieve rapid development of robust system to satisfy new requirements by using previously developed and tested components [6]. This aspect is very important in software engineering because it saves times with low cost. Indeed, the use of existing software component will be during the software development phases. However, the project manager needs to identify these components and assign qualified people early in the planning phase. Therefore, the tool will help in this aspect by keeping information about the ability level of using this technique for every person from the past projects such as 10%-20%, 30%-40%, 50%-60%, 70%-80%, and 90%. Additionally, it will help the project manager to select the people who need training and assign them together with the senior personnel to gain more experience. For example, reuse existing specifications, design, code, or test data developed for past projects that are related to the software to be built for the current project require substantial modification and indeed require qualified people. Therefore, the tool will keep from past projects the percentage of the component that has been reused by each person in each project. So when the manager wants to assign a staff to this activity, the tool will help him by searching for the person possessing the

biggest percentage of using the technique to ensure that the work will be perform on time and with zero or less defects. Also the tool will help the manager to identify the staff who has low percentages and train them.

- (d) **Detection;** One of the objectives of a quality product is to have zero defect and if possible, no defect at all. Defect removal is the most expensive activity and the information volume associated with the software bugs is the largest of any software artifact [7]. Norm Brown stressed that it is more cost effective to identify and fix defects in construction drawing than to disassemble the form. Furthermore, the contracts typically pay contractors by the staff hour and this provides little encouragement to reduce expensive rework [8]. This is the concern of most software developers these days. To ensure that there are no defects during software development, the tool will help the project manager to assign high-level error detection, especially for the critical activities. The tool will also help the project manager to select the staff who need training in this aspect.
- (e) **Programming language;** Assign people based on experience or knowledge in a specific programming language. Ferdinand stated that in his investigation, the highest level of defect avoidance is the choice of an implementation language [9]. In the implementation phase, choosing the programming language is a critical issue for the management due to unsuitable language that the customer wishes the product to be written in or if the product has to be implemented on a specific computer [10]. It is one of the critical factors affecting the success of the project due to lack of experience in programming language [11]. Therefore, assigning qualified and highly experience person in a chosen programming language is one of the factors that will ensure the completeness in zero defect level. For example, in the implementation phase, the manager selects the programming language and assigns the programmer(s) to develop the system. Here, the manager needs to know who is/are the best programmer(s) in the selected programming language (Figure 2). The tool will list the names of the experts in the selected programming language and will display information for each person such as in which projects, activities, and the period that the selected language was used.
- (f) **Specific activity;** Assign people who have experience with specific activity or have worked with similar activity. It is enormously helpful when doing duration and cost estimation. The tool will help the project manager to assign people based on their work experience such as analyst, member in SQA team, designer, interviewer, spokesman, testing or any other tasks in a specific activity. Moreover, the tool will keep other information such as the reusability level, and the detection level of the person. It will keep information about the task or the activity he/she has worked in and when he/she started and completed his/her work. Thereby, making it easier for the project manager to make decision for assigning the right people especially if any of the staff leaves.
- (g) **Specific requirement;** The tool will help assign qualified people based on a specific project management phases or software development phases such as planning, specification requirement, designing, testing, coding, staffing, directing, and controlling. Besides that, it will

aim to help the project manager to determine the lack of skilled personnel in specific areas for training.

- (h) **Specific status;** The project manager will also assign people based on their work i.e. full time, part time, or working under specific arrangement.
- (i) **Others;** The tool should keep other information about the person for the project manager to assign the person based on one of them such as knowledge in specific models (e.g. Booch method, SSADM, HCI, object-oriented methods, animation or multimedia.)

The tool will provide information about any staff that needs to be chosen for a specific job such as qualification, years of experience, training taken or needed, comment or notes from past projects, area he/she is qualified or requires training, reusability level, error detection level, and any additional information such as the method or the technique he/she is familiar with.

4. Availability

The ratio of the total time required for workers to perform activities within a given period is termed as the availability of a personnel [12]. The tool will enable the project manager to track availability in two ways.

- **Track availability for a specific person:** In planning and staffing the project the manager needs to know the staff's availability before assigning them to the tasks. The personnel availability becomes very important when any of the staff, who is working on a critical task suddenly turn over. Searching for the personnel availability manually is a waste of time and cause project delays. Therefore, the tool will help the manager to find the personnel available automatically by comparing the person assigned dates with the system's date and as shown in Figure 3 it is display the following information: (1) Project names that the personnel is working on or will work on. (2) The description of the activities that the personnel is working on or assigned to work on. (3) Activity start date. (4) Activity earlier finishing date. (5) Activity due date. (6) Overlaps between activities.
- **Track the availability within a period of time:** The manager in planning estimating or staffing the project, needs to know who are available within a period of time. Therefore, the tool will help the manager in this aspect by listing and displaying the staff who is not assigned to any project within the given dates and this will save a lot of time.

5. Change in requirement.

This is one of the most critical problem which causes delays and incomplete projects. This problem has affected software projects for a long time [13]. Therefore, monitoring it is very important and it should be one of the project management tools features. The tool helps the project manager to monitor the changes in requirements for each project by: (1) How many time the requirement has been changed. (2) What is/are the problem(s) causing the change. (3) Who is/are responsible for these changes. (4) What is the risk level for each change. (5) Report on all the changes in requirement(s). and (6) Keeps information about the last change and the previous status of the project. See Figure 4.

6. System integration.

Integration is to coordinate the many parts of a project into an effective final result. According to the survey result done in Malaysia, development tools should be integrated together with project management tools [14]. Proper tracking of details such as change in requirement, risk monitoring and controlling would be useful. The tool attempts to implement this integration, and reports on the specification of the project requirement and tracking the change in project requirements.

7. Methodology.

According to the survey result done in Malaysia it is useful to indicate the methodology used for developing software project in software project management tools to reduce the complexity and to guide the user on the software development process life cycle such as life cycle model, spiral model, incremental model, and prototype model [14]. Sommerville also stated that it is useful for the management to consider the phases of software life cycle to be distinct [15]. Therefore, the tool implements this methodology link. Once the user creates the project and chooses the process model for developing the project, the steps of the chosen process model will guide the user while the project is active.

8. Display information and generate reports.

The tool displays new information and generates reports. This will help the project manager enormously in his responsibilities. Information such as:

- a- The user requirement.
- b- Personnel working in a project full time.
- c- Personnel working in a project part time.
- d- Personnel assigned to external projects.
- e- Personnel working on short contract.
- f- Personnel based on their experience.
- g- Personnel based on their qualifications.
- h- Qualified people in specific area.
- i- Lack of skilled personnel in specific area.
- j- Personnel experienced in a specific activity or task.
- k- Personnel experienced in a specific programming language.
- l- Personnel with high, medium, or low level in using reusable software.
- m- Personnel with high, medium, or low level in error detection.
- n- Personnel knowledgeable in specific methodology or method.

Reports such as:

- a- The problems each project has faced.
- b- Problems in a specific project.
- c- Similar problems in specific projects.
- d- Problems due to change(s) of requirements.
- e- The critical problems in specific phase.
- f- Experts in specific area.
- g- Unskilled personnel in a specific area.
- h- Expert in a specific programming language.
- i- Expert in a specific methodology or model.
- j- Personnel highest, medium, or low level error detection.
- k- Personnel high, medium, or low level using the reusability technique.

9. Tracking and monitoring the project.

The tool enables project managers to track the progress and monitor the changes in the project by representing project information graphically. This information will be as the follows: (1) Project name, start and finish date. (2) All project activities and tasks represented by bar chart. (3) The description of each activity and task. (4) Start date, earlier finish date, and late finish date for each activity and task. (5) Indicator to show how many times the project has been changed. and (6) Display the risk level affecting the project from the last change in project requirement.

10. Technology issue.

Software, hardware tools and techniques are being rapidly developed and improved. One of the reasons for misunderstanding the software requirements is "Pushing the Technology Envelope" in other words when the system is expanding into areas of new technology [16]. There are many software companies competing to add new features, improve old features, and develop new techniques. To determine whether old, proven, or new unproven technology is to be used is one of the critical factor affecting the success of the project [17]. A dedicated group of people or department for tracking, searching, and testing the latest related hardware devices, software development tools and the new techniques can help and saves time in the implementation phase. Therefore, the tool will help in this aspect by writing the report for technology evaluation for future use and allow the user to choose the equipment such as software or hardware based on the report published by this group for each unit. In addition, it will also help to determine the needed equipment to complete the project.

CONCLUSION

This paper presents a prototyping tool (SoftProMT) to reduce problems, attempts to ensure project quality, and to allow a better software project management. SoftProMT focuses on several topics such as managing the human resources as a major element of project success, analysing and comparing projects, tracking the change in project requirements, tracking and monitoring a project's progress, providing an example on integrated system, reducing some complexity, and reducing the acquisition time for suitable equipment. The tool also generates reports to help managers make a better decision.

REFERENCE

- [1] CHM & Co. Project Management Consulting and Services-Western Australia, (1997) http://chm.merriweb.com.au/what_pm.html.
- [2] Derek Merrill and James S. Collofello. (1997) Improving Software Project Management Skills Using Software Project Simulator. By Frontiers in Education Conference.
- [3] Avraham Shtub, Jonathan F. Bard, and Shlomo Globerson. Project Management Engineering, Technology, and Implementation. (1994) Prentice-Hall, Inc. p. 278 – 280.
- [4] Roger S. Pressman, Software Engineering A Practitioner's Approach. (1997). Fourth edition by The McGraw-Hill.
- [5] Richard D. Stutzke. (1996). Software Estimating Technology: A Survey. Science Applications International Corp.
- [6] Emad H. Khan and Moheb. Girgis. (1996). The Effect of OO Life Cycle on Software project management. IEEE.
- [7] Capers Jones. (1996). Software Change Management. IEEE Volume: 29 Issue: 2 p.80 – 82.
- [8] Norm Brown. (1996). Industrial-Strength Management Strategies, Software Program Managers Network, <http://www.stsc.hill.af.mil/CrossTalk/1996/aug/industri.html>.
- [9] E. Ferdinand. (1993). Systems, Software, and Quality Engineering. Applying Defect Behavior Theory to programming. By Van Nostrand Reinhold.
- [10] Stephen R. Schach. (1997). Software Engineering with JAVA. By The McGraw-Hill.
- [11] D.G. Dologite. (1991). Using Knowledge Based Systems for Software Project Planning. IEEE.
- [12] Atsuo Hazeyama and Seiichi Komiya. (1992). A Process Model for Software Process Management .IEEE.
- [13] Raymond T.Yeh, Pamela Zave, and Alex Paul Conn, George E. Cole. (1984). Software Requirements: New Directions and Perspectives. Handbook of Software Engineering, 1984, p 519-543.
- [14] Khairuddin Hashim, Hanan Ettaher Dagez. (1998). Survey of Software Engineering Project Management Problems in Malaysia. Proceedings of the 16th IASTED Applied Informatics'98 , p123-126.
- [15] Ian Sommerville. (1995). Software Engineering. Fifth edition Addison-Wesley publishers Ltd.
- [16] Michael M. Lee. (1996). Object Oriented Analysis in the Real World. The global newsletter for information system development professional, VOL 4. No 4.
- [17] Derek Merrill and James S. Collofello. (1997). Improving software project management skills using a software simulator. IEEE.

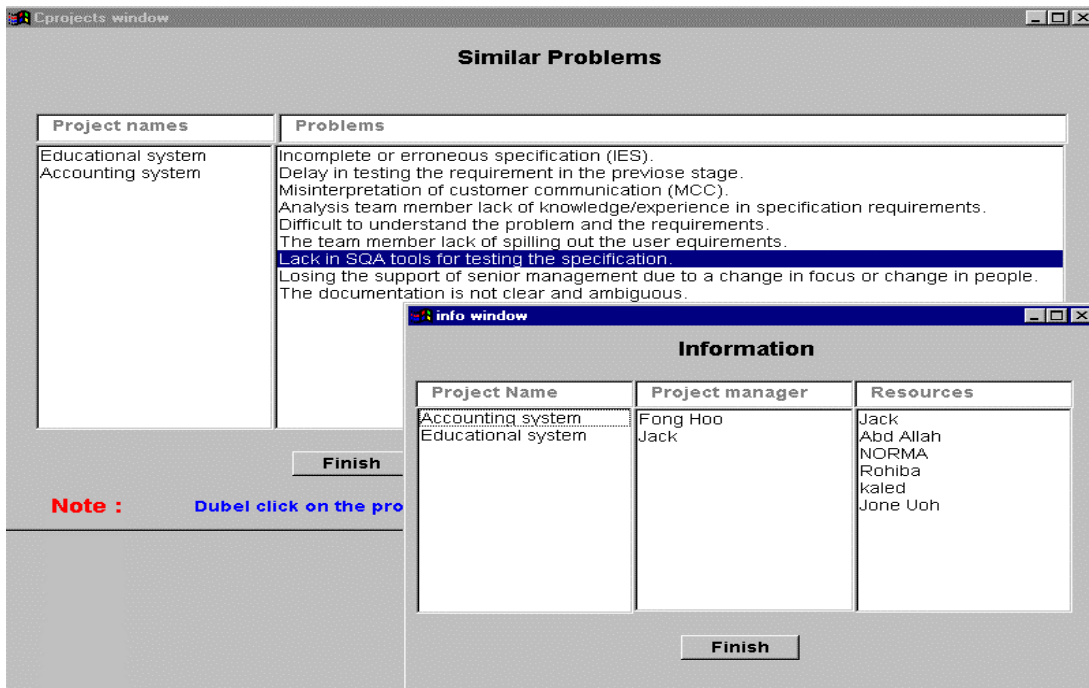


Figure 1: The similar problems in selected projects.

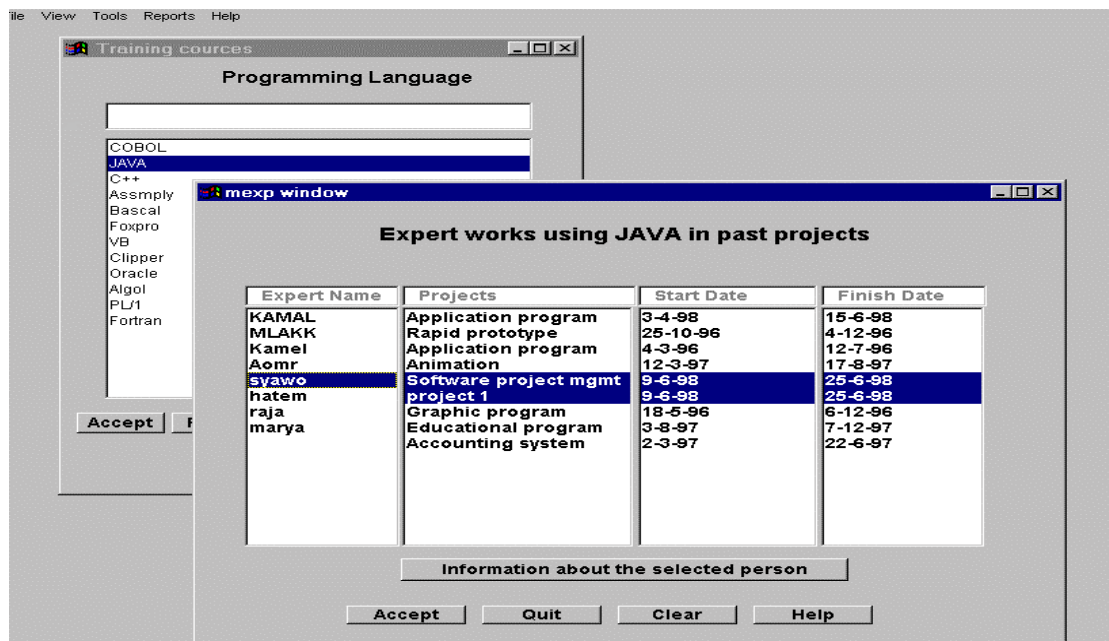


Figure 2: Assigning people based on experience in programming language

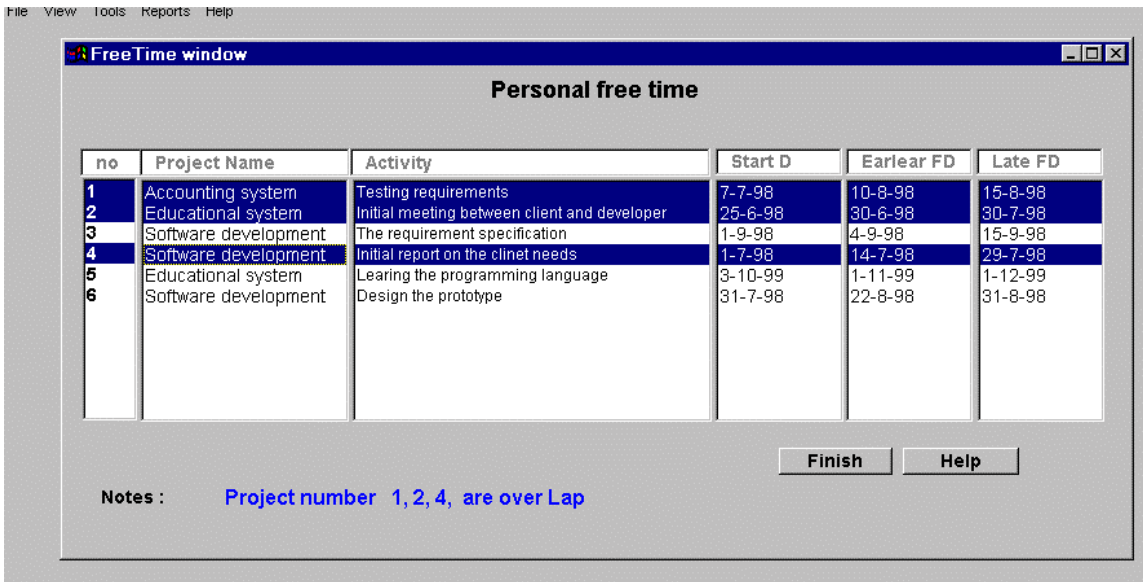


Figure 3: Person's availability

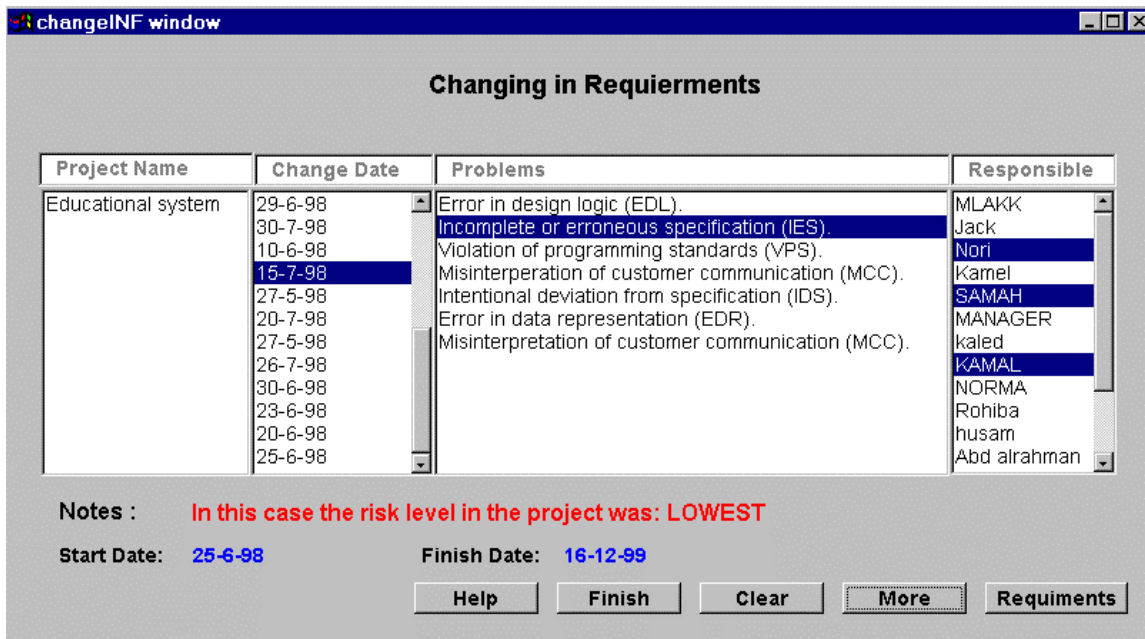


Figure 4: Change in requirements