

An Approach for Monitoring Software Development Using Timesheet and Project Plan

Sunart Wanapaisan, Taratip Suwannasart, and Apinporn Methawachananont

Abstract—The monitoring progress of software projects has an effect on the success of software development. To control the operation of the projects as planned we have to know the progress of the projects during each time period. Staff's timesheet is one choice to apply for monitoring progress of the projects. This paper proposes an approach for monitoring software development using timesheet and project plan and the framework for supporting tool development. This approach uses the staff's timesheet to compare with the staff's work products to monitor and alert the project managers for improving the project plan when the project's performance deviated from the project plan.

Index Terms—Project Monitoring and Control, Timesheet, Project Plan

I. INTRODUCTION

One of important processes to control the operation of the project according to the project plan is project monitoring and control. This process is one of the CMMI® for development [1] process areas. It was compared between the actual work products and the project plan and provides appropriate corrective action when the project's performance deviated from the project plan for controlling the software development project to be success on time.

To effectively monitor and control the project we need to know the progress of the project in each period of time. One method applies for monitoring progress of the projects is using the completed tasks report. The staff should send the completed tasks report to the manager [2]. The manager can use it to analyze the project's performance deviated from the project plan but this method can't monitor the progress of the working task. Renuka Sindhgatta groups proposed monitoring progress by using the task code to calculate task estimation time [3]. However this method can monitor progress only the implementation process but can't monitor progress the other processes.

Controlling the software development is important process like the monitoring process. The improving project plan is necessary when the project's performance deviated from the project plan. Increasing effort is one choice to apply for solving this problem. Many organizations

developed multi-project at the same time [4]. It is important to allocate resources to the other projects to be able to use resources more efficiently.

This paper proposes an approach for monitoring software development using timesheets and project plans and the framework for developing a supporting tool. The approach monitors progress of projects by comparing between the staff's timesheet and the staff's actual work products and alert to the project managers to improve the project plan when the project's performance deviated from the project plan on each period of time. The comparison between the staff's actual work products and the staff's timesheet can use to monitor the progress of all activities in the project plan. We also propose a supporting tool can verify which staff is available in that period to the project manager to use when improves the project plan. This tool able to generate the staff performance and the workload report for the director.

The remainder of this paper is organized as follows. Section 2 describes a background of the CMMI. Section 3 presents the related work. Section 4 describes process definition and development tool. Finally, conclusion and future work are shown in section 5.

II. BACKGROUND

A. CMMI® for development [1]

CMMI® (Capability Maturity Model® Integration) is a collection of best practices that helps organizations to improve their processes. It was developed by Software Engineering Institute (SEI) at the Carnegie Mellon University. It is divided into twenty two process areas. Each process area includes general practice and specific practice which are different each others.

This paper is interested in the Project Monitoring and Control process area. This process area provides an understanding of the project's progress so that appropriate corrective actions can be taken when the project's performance deviated from the project plan. This paper focuses on specific practice 1.1: Monitor Project Planning Parameters, 2.1: Analyze Issues, 2.2: Take Corrective Action, and 2.3: Manage Corrective Actions.

III. RELATED WORK

A. Task Organizer Assistant System (TOAS) [2]

This research was developed by F.H. Boukhadour groups. They presented a tool for managing the working process by assigning jobs to the appropriate staff and managing jobs in an organization. This research was conducted by surveying of staff. The conclusion of this survey showed the staff

Manuscript received January 06, 2013. revised January 26, 2013.

Sunart Wanapaisan is with Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok Thailand (email: Sunart.W@student.chula.ac.th).

Taratip Suwannasart with Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok Thailand (email: Taratip.S@chula.ac.th).

Apinporn Methawachananont is with National Electronics and Computer Technology Center, Bangkok Thailand (e-mail: Apinporn.methawachananont@nectec.or.th).

could work more effectively when jobs were assigned to the appropriate staff. This research is divided into three parts:

1) Assign job to the appropriate staff: This part is used to assign jobs to the most appropriate staff. Tool analyzes from the staff's experiences, the efficient points, and the amount of work.

2) Manage by project manager: This part is used to specify the efficient points and view the historical staff performance comparing with others.

3) Monitoring work: This part is used to monitor staff's working. The project manager can assign jobs to the staff, wait finished job report from the staffs, and verify result of the job.

B. Timesheet Assistant: Mining and Reporting Developer Effort [3]

This research was developed by Renuka Sindhgatta groups. They proposed way to use the staff's report to monitor time spent of the staff. Each entry in the timesheet by staff causes error easily. It has an effect on analyzing effort and hard to verify the staff's working data. Therefore,

they presented a tool for staff effort estimation. This tool could monitoring the staff's coding by retrieving the staff's code in each period of time to calculate and estimate effort of each job.

C. Software Multi-project Resource Scheduling: A Comparative Analysis [4]

This research was proposed by Fei Dong groups. They proposed the comparative analysis of software multi-project resource scheduling. Many organizations developed multi-project at the same time. The common project management project was not efficient enough. This research compared resource management method in different views.

IV. PROCESS DEFINITION AND THE APPROACH TO DEVELOPMENT

The process definition and the tool development for monitoring software development using timesheet and project plan is described as shown in Fig 1.

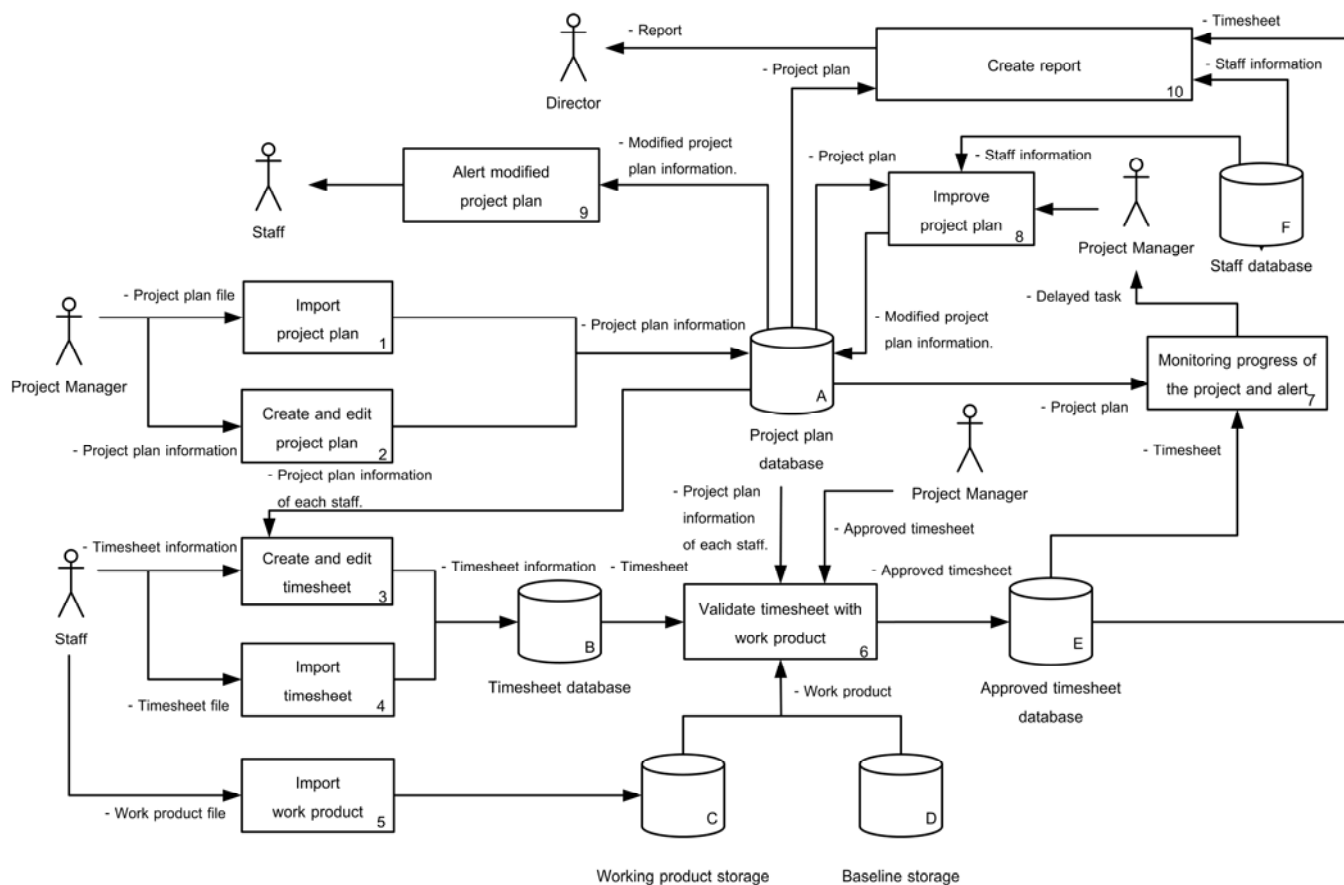


Fig 1 Overview of our Approach

An example of project plan information is shown in Table I. The project plan information is stored in the project plan database.

Table I An example of the project plan information

Project information	
Project code	PRJ_RES_01
Project name	Develop supporting tools project
Responsible project manager	PM_01 Mr. Alan Oscar
Task information	
Task code	JOB001
Task name	Add data A
Start date	15/2/2013
End date	20/2/2013
Responsible staff	OC_001 Mr. John William
Required position	Programmer
Required skill	Java
Work products	Job1_01.java, Job1_02.java, Job1_03.java, Job1_04.java
Milestone information	
Milestone code	SEND_01
Deadline	21/2/2013
Required work products	Job1_01.java, Job1_02.java, Job1_03.java, Job1_04.java

A. Import project plan

The project manager imports the project plan information to the system. This information is divided into three parts: Project information, Task information, and Milestone information.

B. Create and edit project plan

The project manager can create and edit the project plan interface. The tool retrieves the project plan from the project plan database to use as a template for the new project plan.

C. Create and edit timesheet

The staff can create and edit the timesheet. The tool retrieves staff's task from the project plan database. So the staff can enter the time spent on each task in each day, and the progress percentage into the timesheet that can be used to monitor and analyze the progress of the project.

Effort shall be measured in staff-hour units following IEEE standard [5, 6].

D. Import timesheet

The staff imports the timesheet information to the system. This information includes staff code, project code, task code, working date, time spent, and progress percentage. An example of the timesheet information is shown in Table II.

Table II An example of the timesheet information

Timesheet information						
Staff code	OC_001					
Project code	PRJ_RES_01					
Working information						
Task code	15/2 /13	16/2 /13	17/2 /13	18/2 /13	19/2 /13	%
JOB001	8	8	8	8	8	75
JOB002	0	0	0	0	0	0

E. Import work products

The staff stores the staff's work products to the working product storage, and the management configuration agent will move the finished work products to store in the baseline storage according to the folder of each project. These work products will be used to validate the staff's timesheet.

F. Validate timesheet with work products

The tool can validate the staff's timesheet by comparing the staff's timesheet from the timesheet database with the project plan. For example, the task in the staff's timesheet must be his responsibility. The tool will monitor the progress of the project by comparing between the staff's timesheet and the actual work products which should be gotten from the work product storage of this work. The project manager is responsible for approving results from the tool. The approved staff's timesheets are stored in the approved timesheet database.

G. Staff database

The staff database stores the staff information in organizations. This information is used for staff allocation to the right jobs. An example of staff information is shown in Table III.

Table III An example of staff information

Staff information	
Staff code	OC_001
Project code	Mr. John William
Team	PM_01
Position	Programmer
Skill	Java (Master), Oracle (Skilled)
Staff code	OC_002
Project code	Mr. Henry Alba
Team	PM_01
Position	Programmer
Skill	Java (Skilled)

H. Monitoring progress of the project and alert

The project manager can specify milestones of the projects. Then, the tool will compare the information of the project plan database with the approved timesheet database. The tool will return tasks which have risk on schedule and the amount of increasing resources that should be used to complete the schedule. This process is classified into two cases.

Case 1: Progress percentage is equal to zero.

That means this task has not been started. So, the amount of the resources that should be used is equal to the amount of the resources from the project plan.

Case 2: Progress percentage is not equal to zero.

The estimated effort can be calculated using the following formula,

$$Estimated\ Effort_t = (Actual_t \times \frac{100}{\%Completed_t}) - Actual_t \quad (1)$$

Where $Estimated\ Effort_t$ is the amount of resources that should be used to complete task t (Staff-Hour), $Actual_t$ is the amount of actual resources for task t that is used from start date, $\%Completed_t$ is the progress percentage of task t , t is task t

The amount of increasing resources that should be used to complete the schedule can be calculated by the following formula,

$$Added\ Effort_t = Estimated\ Effort_t - Remain\ in\ Plan_t \quad (2)$$

Where $Added\ Effort_t$ is the amount of increasing resources that should be used to complete the schedule, $Remain\ in\ Plan_t$ is the amount of remain resources in the project plan

For example, the tool calculates resources for the first milestone from the information in Table I that have four work products to be completed on 21 February 2013. (Job1_01.java, Job1_02.java, Job1_03.java, Job1_04.java)

The OC_01's timesheet is approved and found three work products in the work product storage. (Job1_01.java, Job1_02.java, Job1_03.java) So,

$$Estimated\ Effort_{SEND_01} = (40 \times \frac{100}{75}) - 40$$

The amount of resources that should be used to complete is $13.3 \approx 14$ Staff-Hour.

The monitoring progress date of the first milestone is on 19 February 2013.

The deadline date of JOB_01 from project plan is on 20 February 2013.

OC_01 has Staff-Hour to do this task = $1 * 8 = 8$ Staff-Hour. So,

$$Added\ Effort_{SEND_01} = 14 - 8$$

The amount of increasing resources that should be used to complete the schedule is 6 Staff-Hour.

I. Improve project plan

The project manager can modify the project plan to complete the schedule by using the amount of increasing resources from calculating increased resources to the tasks which are risk on schedule. The tool can show the list of staff who have free time sorted by the required skills of the staff. The modified project plan is stored in the project plan database.

J. Alert modified project plan

The tool will alert the staff who has an effect on the modified project plan, when the project plan is modified. So, the staff will know when they are assigned.

K. Create report

A report will be displayed as dashboard report for the director. This report will display the information based on viewer authorization. The project manager can view only his projects but the director can view all projects.

1) Staff performance report

This report displays the staff performance that is calculated by using the software productivity. The formula is as follow:

$$Staff\ Productivity_t = \frac{Size_t}{Effort_t} \quad (3)$$

Where $Staff\ Productivity_t$ is software productivity of task t , $Size_t$ is size of task t that staff is assigned, $Effort_t$ is the amount of resources that is used for task t (Staff-Hour).

$Size_t$ gets from the work product database and $Effort_t$ gets from the approved timesheet database. This report is used for evaluating the ability of the staff and performance of their operations.

2) Workload report

This report displays the amount of work (Staff-Hour) of each staff in each period of time. This information can be used to evaluate increase or decrease the number of staff in the organization.

The amount of works gets from the staff's assigned tasks of each project comparing with the common effort per day of each organization.

V. CONCLUSION AND FUTURE WORK

In this paper, we proposed an approach for monitoring software development using timesheet and project plan and the framework for developing a supporting tool. This proposed approach can monitor progress of the projects by comparing between the staff's timesheet and the staff's actual work products, and then alert to the project managers to improve the project plan when the project's performance deviated from the project plan on each period of time. The comparison between the staff's actual work products and the staff's timesheet can use to monitor the progress of all activity in the project plan.

In the future, we plan to comparison analyze among the staff's timesheets and the various work products for each staff's position and develop a supporting tool.

REFERENCES

- [1] CMU/SEI "Capability Maturity Model Integration, CMMI for Development, Version 1.3", CMU/SEI-2010 RT-033, Pittsburgh, Software Engineering Institute Carnegie Mellon University, 2010.
- [2] F. H. Boukhadour and A. Jalal-Karim, "Task Organizer Assistant System (TOAS)", Second International

Conference on Developments in eSystems Engineering,
2009, pp. 460-465.

- [3] R. Sindhgatta, N. C. Narendra, B. Sengupta, K. Visweswariah, and A. G. Ryman, "Timesheet Assistant: Mining and Reporting Developer Effort", ASE'10 Proceedings of the IEEE/ACM international conference on Automated software engineering, 2010, pp. 265-274.
- [4] D. Fei, L. Mingshu, Z. Yuzhu, L. Juan, and Y. Ye, "Software Multi-project Resource Scheduling: A Comparative Analysis", in *Lecture Notes in Computer Science 2008*, Volume 5007/2008, 63-75.
- [5] W. Goethert, E. Bailey, and M. Busby, "Software Effort and Schedule Measurement: A Framework for Counting Staff-Hours and Reporting Schedule Information", Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, Technical Report CMU/SEI-92-TR-021, 1992.
- [6] *IEEE Standard for Software Productivity Metrics*, IEEE Standard 1045, 1992.