



HAL
open science

The Agile Management of Development Projects of Software Combining Scrum, Kanban and Expert Consultation

Michel Evaristo Febles Parker, Yusleydi Del Monte

► **To cite this version:**

Michel Evaristo Febles Parker, Yusleydi Del Monte. The Agile Management of Development Projects of Software Combining Scrum, Kanban and Expert Consultation. 10th IFIP International Conference on Open Source Systems (OSS), May 2014, San José, Costa Rica. pp.176-180, 10.1007/978-3-642-55128-4_25 . hal-01373094

HAL Id: hal-01373094

<https://inria.hal.science/hal-01373094v1>

Submitted on 28 Sep 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

The agile management of development projects of software combining Scrum, Kanban and Expert consultation

Michel Evaristo Febles Parker^{1*}, Yusleydi Fernández del Monte¹

¹ University of Informatics Sciences (UCI). Havana. Cuba, 537-8372531.
{mfparker, ydelmonte}@uci.cu

Abstract. At the University of Informatics Sciences (UCI), Havana, Cuba, it is found The Center of Free Solutions of Software (CESOL) who has an informatic project named "Auditing of Source Code" (ACF). This project has as objective to develop an open source software solution to auditing the source code of several software solutions with an agile projects management. In the present investigation have been showed the experiences obtained in the mixed application of two methods of agile projects management; Kanban and Scrum, together with the method Judgment of Expert, during the stage of construction of the lifecycle of ACF, when it is was performed a quality auditing by specialists of the CALISOFT company. In the auditing were detected several errors and to resolve them was necessary to estimate efforts, time and to revalue the lifecycle of the project. Moreover, the investigation show how this method can be used as a guide for young project managers for a correct planification and how can be used as a personal organizational method.

Keywords: Scrum, Kanban, Agile management of projects,

1 Introduction

The Methods of agile software project management are guides for planning and control thereof. Currently several software development methodologies are focused on this style. The free software applications by the need to respond quickly to the constant changes in its requirements, technology and its short development period, are who most use them. In the Department of Operating System of the center CESOL, are developed open source software solutions using free tools and agile methodologies, organized in various development projects. Besides, the UCI is working with a view to improving the quality of his process of development up to Level 2 of CMMI. To check the correct execution of the model, audits and reviews are performed to the projects by the company specialists CALISOFT, the institution responsible for validating the quality of the process. ACF is a project that belong to the center CESOL, where a system to auditing the source code of software systems is made using an agile management and free software tools to develop. The present investigation shows the experiences gained during the combined application of the methods of agile project management, Kankan and Scrum, along with expert judgment method during the construction phase of the life cycle of the ACF project which was audited quality. Therefore the objective of this investigation is to show the experiences gained during the combined application of the agile methods project management, Scrum and Kankan, along with expert judgment, to achieve a pleasant management of a computer project.

2 Discussion

The agile management of projects is a management able to adapt and respond to new requirements and changes dictated by the environment [1]. Inside of this model are found the agile methodologies of development of software, that seek the early delivery of incremental software [2], among which are Extreme Programming (XP), Adaptive Software Development (ASD), Open Unified Process (OpenUp), Kanban and Scrum. Scrum is a process that apply regularly a set of best practices for

working collaboratively. [3] Divide the team into small specialized groups managed by themselves, dividing the work into a list of small tasks or requirements for deliverables. Sort the list by priority and estimated the relative effort of each element [4]. Kanban is used to monitor the progress of work in the context of a production line and is currently used for agile project management, often with Scrum (known as Scrumban) [5].

CMMI is a model that is not focused on the principles of agile development. It is an adaptable guide to raise the quality level of the software development process of an entity. Particularly in the UCI there is a project called Programa de mejora, currently at version 3.4, in order to adapt CMMI to the different development centers that exist in it; lightening the documentation as possible, in order to fulfill model without making conflict with the environments of agile development. Particularly the case study is an example of a stage of the life cycle of the ACF project where was used the estimation method in view of solving the macro task "Fix no conformities identified during the audit quality", conformed by small subtasks.

2.1 How to combine both methods?

It starts with a set of tasks or requirements (the term is to taste) to perform. Then it proceeds to prioritize tasks using various criteria defined by the person responsible for managing the process. These may be important for the customer, the level of complexity, the amount of resources required for the implementation and the dependency among the requirements. The criteria should not be less than three. Each task is evaluated using these criteria according to a metric that can also be defined by the person who manage the application of the method, preferably [1-5], [1-10] and [5-10]. After evaluating each task, the values obtained for each criterion are added together and this is the value to use as a criterion for prioritizing tasks, sorting them in descending leaving those with highest numerical value as the first to be executed. In the event that the comparison test match, you can optionally choose the order that those tasks will have between them. Later proceeds to define the time duration of the tasks. To make this process intervals are defined from the values obtained as a result of the comparison test; intervals can also be defined optionally. These intervals are associated durations for tasks. In the interval where the criterion of value is within, the time corresponding to the interval is associated to the task. The times are defined using analogies of old tasks, expert consultation, experience and personal judgment.

2.2 Experiences in the combined use of both methods:

In June 2013 the ACF project was audited by specialists CALISOFT where a set of non-conformances that must be resolved in the shortest time as possible for the project were consistent with the quality model and resume its planning in the shortest possible time too. The initial group of tasks to be performed was as follows:

1-Perform document "Technical project", 2- Perform document "Project plan", 3-Perform document "Glossary of terms", 4-Perform document "Validation of the requirements", 5-Perform document "Plan of iteration", 6-Perform document "Work item list", 7-Perform document "Use case specification", 8-Perform document "Use case model, 9-Perform document "Specification of the requirements", 10-Perform document "Vision", 11-Perform document of architecture, 12-Perform document "Art state of the product to develop", 13-Perform document "List of risks" and 14-Perform document "Requirements of support".

The criteria defined for determining priority were complexity (task difficulty), size (effort needed to accomplish the task), importance (importance to the project) and the interest (interest of the project

team of to execute the task), being the metric used 1-5. The result of the prioritization was the following list of tasks: Tasks 1, 2, 3, 4, 5 and 6 with priority 20; task 7 with priority 18; tasks 8, 9,10,11 and 12 with priority 17; task 13 priority 13 and task 14 with priority 12. The estimation of time intervals defined initially were: the tasks with priority [20] will last 3 days, with priority [19-18] will last 2 days and with priority [1-17] will last one day, which estimate a total of 29 days. Splitting the time between the number of workers on the project who is three, is obtained as a result approximately 9.7 days. To restrict the number of tasks was taken into account that two person on the team had the ability to perform two tasks simultaneously, for that reason was defined a working limit of 5 tasks for the columns Assigned, Developing and Reviewing.

Table 1. View of the Kanban board at the end of the first day of work, it can see that the task 3 was completed on the first day.

Task list	Assigned	In development	In review	Finished	% of execution
8,9,10,11,12,13,14	2,5,6,7	1,4		3	7%

The second day, during a review of the remaining tasks was determined reassess the priority of task 9, leaving with a score of 20 therefore ascends to be the first task to execute.

Table 2. View of the Kanban board at the end of the second day of work.

Task list	Assigned	In development	In review	Finished	% of execution
10,11,12,13,14	2,9,8	1,5,6,7		3,4	14%

At the end of the third day of work, the tasks 2 and 10 were assigned and the tasks 9 and 8 upgraded to review. An active risk1 in the ACF project is the lack material resources and eventually in the stage in question was materialized, affecting the implementation of task 11. Therefore, as corrective action was determined to eliminate the task 11 of the board because at that time do not had the necessary resources to execute it.

Table 3. View of the Kanban board at the end of the fourth day of work.

Task list	Assigned	In development	In review	Finished	% of execution
11	2,12,13,14	1,5,6,7		3,4,9,8,10	35.7%

Table 4. View of the Kanban board at the end of the fifth day of work

Task list	Assigned	In development	In review	Finished	% of execution
	2	1,5,6		3,4,9,8,10,12,13,14,11,7	71.4%

At the end of the sixth day of work the task 2 entered into development and tasks 1, 5, 6 and 2 into review when the seventh day finished, on both days the percentage of implementation was 71.4%. On the eighth day all tasks were completed for a 100% of execution.

3 Lessons Learned

It is possible reassess the task duration by increasing or decreasing their priority based on the new that arises in the development of software. The feedback determine that for this case, the future tasks that are similar to the first 6 and number 9, the duration should be 8 days, which represents over five days than estimated. For the tasks of 2 days, must add them one day and the tasks of 1 day behaved as estimated, allowing it to update the initial estimates.

It can be estimated and reassign tasks at the moment, allowing time to mitigate the risks that may exist. During process execution task 9 was prioritized again, flexibly changing the allocation of the task by a need of the development team. Following the realization of a risk associated with the project, task 11 was removed as a corrective measure.

In the prioritizing the tasks influence the characteristics of the team in terms of their skills and behaviors. The meetings of checking allow analysis of the performance of tasks and receive feedback from the experiences of the entire team to make adjustments to improve the planning, estimation and execution.

Allows to analyze in short term, the trends of the development team and take steps to improve. Allows to know the speed of the work team and for inexperienced project leaders to estimate and to update that estimate at the same time that the project it is running.

It can work objectively by prioritizing tasks. The prioritization of the requirements allows to obtain a list of work focused on the key elements to achieve the project objectives; because each time a task is completed the final product evolves.

4 Conclusions

The investigation arrived at four conclusions. The first is that was showed the experiences gained during the joint implementation of Kanban and Scrum are detailed, along with the expert consultation to achieve a pleasant management in the project ACF. As a second conclusion, is that the combination of these methods allows a more precise estimate of the work, mainly for inexperienced project leaders and ordinary people. The third conclusion is that with his application the project can quickly reach his objectives that the product evolves in each review, that the planning be flexible and analyzes the existence of risks, his mitigation and that the entire work team participate in the management of the project. The fourth conclusion is that Scrum and Kanban complement themselves, by the characteristics of an agile environment there will always be changes in the requirements or the tasks during the development process, being necessary to insert them in that process. The board of Kanban by him selves, describes the workflow very well, join it with the prioritized list of SCRUM, the possibility of Kanban to modify the tasks without having to wait for the next iteration and a workflow guided by goals, demonstrate why is best to use them together.

5 References

1. Palacios, J (2006). Agile project management: basics concepts: conceptos básicos. [Online]. Available at: http://www.navegapolis.net/files/s/NST-003_01.pdf.
2. Pressman, Roger. *Software engineering. Ed.McGraw-Hill, 6th Edition*. Chapter 4 Agile Development. Page 77.
3. Schwaber, K. Sutherland, J. “The Scrum Guide”, (2013) <https://www.scrum.org/Portals/0/Documents/Scrum%20Guides/2013/Scrum-Guide-ES.pdf#zoom=100>.
4. Kniberg, Henrik. Skarin, Mattias. (2010). *Kanban and Scrum – getting the best of both*. [Online]..
5. Garzás, Javier. “What is the Kanban method for project management?”. November 2011; <https://eventioz.com.ar/e/el-metodo-kanban-creando-un-cambio-evolutivo-exito>.