

A Study on the Influence of Use of Web 2.0 Collaboration Tool Reflecting Agile Practice on the Evaluation of OS Project

Hyo-jung Sohn¹, Min-gyu Lee², Baek-min Seong³ and Jong-Bae Kim^{4*}

^{1,2,3,4*} Graduate School of Software, Soongsil University, Seoul, 156-743, Korea
¹hyojung.sohn@gmail.com, ²marse101@naver.com, ³feeling127@naver.com,
^{4*}kjb123@ssu.ac.kr

Abstract

For the agile development methodology that repeats the frequent releases and the short development periods, there are some value practices that we have to keep. Among them, three typical practices were selected: use of Web 2.0 collaboration tool, adoption of the test driven development (TDD) and refactoring. The agile practice that forces the communication between team members was defined with the use of Web 2.0 collaboration tool. When this is applied to the development method of open-source projects which are dispersed geographically, temporally and culturally, it is expected to raise the satisfaction of OS project users and to contribute the maturity of community. It was applied to the study models that were drawn out from the precedent studies.

Today, OS project users are adopting diverse tools for the online communication. But the degree of communication that is emphasized in the agile methodology can be satisfied only with the use of two-way, open and communicable tools reflecting the characteristics of Web 2.0 rather than with the conventional one-way information transmitting tool. In this situation, it was examined how the conventional and the improved tools have influence on the users' satisfaction. This paper is focusing on the communication factor among 3 factors.

Keywords: Communication Tool, Collaboration Tool, Web 2.0, Agile practice, Methodology, Open source community

1. Introduction

As the environment of software development got uncertain more and more and the period of its revision got shorter and shorter, the traditional development method has repeated failures. To solve this problem, the so-called agile development methodology has become the center of attention that rapidly accepts the changes of developmental environment and actively responses against them. The agile development method that continuously repeats the process of demand analysis, design, coding and testing with relatively short periods while gradually increasing the size of software can be said to be very similar to the open source software development project (OSSDP).

The most typical agile methodologies are XP (eXtreme Programming)[1] and Scrum[2]. The Text Driven Development and Refactoring which are the development techniques emphasized in the XP methodology can pursue the high productivity through the re-use of software and easy maintenance. Furthermore, with the intensive communication emphasized in Scrum and its Time Boxing feature, it is asserted to facilitate the highly productive project team management. In this study, it is examined how the tools to facilitate the intensive and active communication emphasized in the agile methodology have influence on the satisfaction of OS project users while analyzing their correlations.

^{4*} Corresponding author. Tel. : +82-10-9027-3148.
Email address: kjb123@ssu.ac.kr(Jong-Bae Kim).

This study is focusing on the communication in the hypothesis suggested by precedent studies that 3 features (TDD, refactoring and Web 2.0 collaboration tool) of agile methodology would have influence on the satisfaction of OS project users. In the future, the research and analysis studies will also be carried out for the rest two factors to verify the validity of the hypothesis and to suggest a more developed OSSDP.

2. Related Works

2.1. Agile Value and Practice [3]

Value, principle and practice have a close relationship with each other and firmly support each other. Practice is the collection of repetitive actions carried out by people, which are made in the context of some principles. For example, the company that values the satisfaction of customers' desires can achieve the purpose with a practice such as TDD. Here, the practice of TDD is supported by the principle that gives the priority to the design.

The value pursued by the agile methodology is focusing on 4 items mentioned in the manifesto of Agile Alliance: communication, simplicity, feedback and courage. In the practice of the most typical XP development methodology in the agile methodologies, there are 12 practices including simple design, TDD(Test-Driven Development), refactoring, coding standard, pair programming, collective code ownership, continuous integration, planning game, small release, system metaphor, 40 hour work a week and on-site customers. The practices are the essential methods that make the agile methodology possible. Among them, three can be selected which are applicable to the open source project development: TDD, refactoring and intensive communication. By substituting the pair programming of XP and the various meetings (daily scrum, sprint review *etc.*) of Scrum with the similar Web 2.0 collaboration tool, it will be checked how each practice has influence on the open source project development.

2.2. Agile Methods

For the Agile development framework, diverse methodologies such as eXtreme Programming, Scrum, crystal and FDD have been used and developed. Among them, the most widely used ones are XP and Scrum, which are mainly examined in this paper. XP is a typical Agile methodology, which was established by applying and revising the most effective development techniques. In general, it is a light development methodology appropriate for small and medium development team units. XP emphasizes the customers' satisfaction and the communication in the team, focusing on the 5 values of communication, simplicity, feedback, courage and respect. In addition, it suggests 12 practices mentioned above in the development realm: simple design, TDD, refactoring, coding standard, pair programming, collective code ownership, continuous integration, planning game in the management realm, small release, metaphor, 40 hour work a week in the environment realm and on-site customers.

Scrum is a process framework focusing on the improvement and management of the team rather than the technique. The most important feature of Scrum is that, after collecting all requirements, it operates the sprint of very short period (usually 2~4 weeks) to realize the most important function. With this result, it has the feedback meeting with users to evaluate the development contents and to increase them. What make up the processes of Scrum are sprint, 3 meetings (daily scrum, sprint plan and sprint review) and 3 products(product backlog, sprint backlog, and burn down chart). Especially, the management method is to activate the communication. With this, we can see how the intensive communication is emphasized in Scrum. [4]

2.3. Collaboration Tool of OS Project [5]

In the software development project, the communication is one of the most important strategies. The participants of open source project are usually dispersed in geography and time zone. To solve the problem, a lot of tools have been used.

The collaboration tool that is used most diversely among the communication tools is the Mailing List. As a non-synchronous communication tool, the mailing list is a distribution list that, when a member sends the message to the project server, it is sent to all the members registered on the list. It has been used for a long time and can be used in diverse ways. But it is a one-way collaboration tool and every member has to register on the mailing list. A communication tool which the users with low interest can participate in and all the people can have access to, is the forum type communication. In it, the discussion for each topic can be made, and the members can receive the notices by subscribing the RSS (Really Simple Syndication) feed. But, to participate in it, each member has to bear the inconvenience to visit the website. Nowadays, it is evolving to the blog type.

The traditional synchronous tools that support the virtual meeting to overcome the limits of non-synchronous communication are IRC (Internet Relay Chat) and video conference system. These are the tools for discussion of goal setup, evaluation of progress situations and international issues, but they have technical and practical difficulties in the participation of all members.

Finally, the communication through documents can be made. To-Do List or Milestone is an indispensable tool for collaboration. To-do List is the work list in which the team members' names are included. It is especially useful in allotting Bug Fix. Milestone is the event calendar which contains important stages of the project along with its closing date. These are also the one-way communication tool that the participation in it is restricted by the existence of authority. Furthermore, it has a demerit that the record management is difficult. These days, the document collaboration tends to be made with Wiki.

These phenomena can be classified into the traditional communication tools of Web 1.0 such as Mailing List, To-do List and Milestone and the new communication tools of Web 2.0 such as Blog, Wiki and SNS (Social Network Service)[7] whose features are openness, interaction and participation.

Table 1. Classification of OS Project Collaboration Tools

	Web 1.0	Web 2.0
Features	<ul style="list-style-type: none"> • One-way Service • Unmodifiable by users • Portal-type Web 	<ul style="list-style-type: none"> • Two-way Service • Modifiable by users • Web as a platform
Collaboration Tools	Mailing List, To-do List, Milestone	Blog, Wiki, SNS

3. Research Model and Hypotheses

3.1. Study Hypotheses

The relation between an agile practice, which is the final goal of this study, and the 'community maturity' can be derived into a conceptual research model [6] as follows:

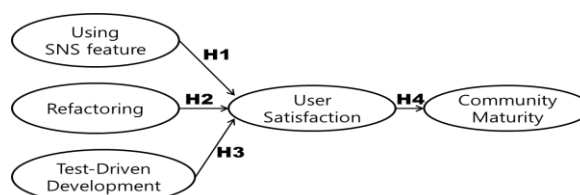


Figure 1. Research Model

With the above research model, the following hypotheses are suggested to show that the participants' satisfaction would increase in the OS project that uses the collaboration tools based on Web 2.0. As the Web 2.0 is defined to have features such as users' participation, common ownership and openness, Blog, Wiki and Social Network Services (SNS) were selected as the open source communication tools with the Web 2.0 features. On the other hand, Mailing List was selected as the one-way communication tool with the Web 1.0 features.

H1: The project using Blog will have a higher evaluation grade.

H2: The project using Blog will have a shorter update time.

H3: The project using Blog will have more downloads.

H4: The project using Wiki will have a higher evaluation grade.

H5: The project using Wiki will have a shorter update time.

H6: The project using Wiki will have more downloads.

H7: The project using Social Network Service will have a higher evaluation grade.

H8: The project using Social Network Service will have a shorter update time.

H9: The project using Social Network Service will have more downloads.

H10: The project using Mailing list will have a higher evaluation grade.

H11: The project using Mailing list will have a shorter update time.

H12: The project using Mailing list will have more downloads.

3.2. Data Collection

For the analysis, from 3 open source hosting sites with largest users – sourceforge.net[8], github.com[9] and code.google.com[10] – 30, 30 and 40 projects were selected respectively by categories. From above 100 project samples in total, the following items of data were drawn out to evaluate communication tools and users' satisfaction: project grade, project downloads, release cycle by versions, use of Blog, Wiki, SNS and Mailing List.

As the management types of the sites were different from each other, it was necessary to adjust the items for the coherence of data. In case of downloads, sourceforge.net showed the weekly downloads number, but code.google.com just showed the cumulative downloads number and therefore the number of downloads just for newly released versions. As github showed no downloads data, it was substituted with the number of forks. The average update time was classified into the periods of 1 month or shorter, 2 months, 3 months, 6 months or longer, 1 year or longer and none.

3.3. Data Analysis Method

The purpose of this analysis was to check which collaboration tools have influence on the satisfaction of OS project users. For this, 3 indices were selected as the evaluation items of users' satisfaction. If the satisfaction of OS project participants gets greater, they will participate in it more actively, causing the faster development speed and more frequent update of versions. Furthermore, the number of downloads for the relevant program will increase as well as its grade. And the use of collaboration tools such as Blogs, Wiki, SNS and Mailing List was also examined.

In this study, Chi-square test was carried out with an open source analyzer R v.3.2.0 for the collected data to verify the relation of each tool with the grade, the number of downloads and the update time.

Table 2. Project Data Collected from Github.com (Partially)

Project Name	Grade	Average Update time(cases)	Downloads (fork)	Collaboration Tools			
				Blog	Wiki	SNS	Mailing List
node	5	1 mon or shorter	5,000~10,000	Y	Y	Y	Y
mbostock/d3	5	1 mon or shorter	5,000~10,000	N	Y	Y	N
angular.js	5	1 mon or shorter	10,000~50,000	Y	Y	Y	Y
httppei	4	3 months	100~1,000	N	N	N	N
django	3	1 mon or shorter	5,000~10,000	Y	Y	Y	Y
flask	3	6 months	1,000~5,000	N	Y	N	Y
jekyll	4	1 mon or shorter	1,000~5,000	Y	Y	N	N
homebrew	5	None	10,000~50,000	N	Y	Y	N
rails	5	1 mon or shorter	10,000~50,000	Y	N	Y	Y
Elastic search	2	1 mon or shorter	1,000~5,000	Y	N	Y	N
...

4. Analysis Result

4.1. Result of Chi-square Test

For each null hypothesis, the Chi-square test was carried out and the result is shown in Figure 5. When the significant level of p-value was set as 5% in the Chi-square test, among above 12 hypotheses, Blog had the significant relation with the update time and the number of downloads; Wiki with the number of downloads; SNS with update time; Mailing List with the grade.

Table 3. Blog vs Update Time Pearson's Chi-Squared Test

Item	Value
Chi ²	13.33091
d.f.	5
p-value	0.02046747

The Chi-square (Chi²) was 13.33 and the degree of freedom (d.f.) was 5. As the p-value 0.02 is located within the significant level 5%, we can say that two variables are related with each other.

4.2. Result of Hypotheses Adoption

The participants who use the typical Web 2.0 tools of Blog, Wiki and SNS showed the relation with the update time and the number of downloads, but no relation with the grade. The participants who use the Mailing List showed the relation with the grade. It means that, when participating in the relatively compulsory and closed mailing list, the OS project users show more passive activities but those using more liberal collaboration tools such as Blog, Wiki and SNS participate in the activities of update and download more actively to make the project.

Table 4. Results of Alternative Hypothesis Evaluation

	Alternative Hypothesis	P-Value	Adoption
H1	The project using Blog will have a higher evaluation grade.	0.22	No
H2	The project using Blog will have a shorter update time.	0.02	Yes
H3	The project using Blog will have more downloads.	0.44	No
H4	The project using Wiki will have a higher evaluation grade.	0.17	No
H5	The project using Wiki will have a higher evaluation grade.	0.97	No
H6	The project using Wiki will have more downloads.	0.05	Yes
H7	The project using SNS will have a higher evaluation grade.	0.97	No
H8	The project using SNS will have a shorter update time.	0.03	Yes
H9	The project using SNS will have more downloads.	0.65	No
H10	The project using Mailing list will have a higher evaluation grade.	0.01	Yes
H11	The project using Mailing list will have a shorter update time.	0.09	No
H12	The project using Mailing list will have more downloads.	0.52	No

5. Future Research

In this study, the relation of the communication factor among the agile methodologies with the users' satisfaction was examined. In the future research, we will expand the collected project data to examine the relation of rest agile features (TDD and Refactoring) with the users' satisfaction. Through the users' satisfaction, the relation with the community maturity will finally be analyzed to suggest a more developed open source development methodology reflecting the agile features. The high maturity of the community can mean the successful development of OS project. Therefore, the final goal of this study is to contribute to the reuse of software and the elevation of productivity through the success of more diverse open source software.

6. Conclusion

There are many similarities between the agile development method and the open source software development process. But the features of agile development methodology, which is fast evolving for the success of projects in more diverse environments and conditions, are worth referring in OSSDP.

In this study, with expectation that the practices reflecting the value of agile development methodology derived from precedent studies would have positive influence on OSSDP, the relation between communication tool and users' satisfaction was analyzed first in the research model. For the communication tools, a hypothesis was set that Web 2.0 characteristic tools of Blog, Wiki and SNS would have influence on the users' satisfaction evaluation items such as OS project grade, update time and number of downloads while Web 1.0 tool of Mailing List would have little influence on the users' satisfaction. But, as a result of analysis, it turned out that Web 2.0 tools were related with the update time and the number of downloads and that Mailing List was related with the evaluation grade.

It is the result reflecting the participation positivity of the participants. It was analyzed that, as the chances for active communication participation of communication, participation and openness get greater, they will have influence on the direct activities such as update time and number of downloads among the users' satisfaction items. But this study had the limits that, as the grade calculation type and

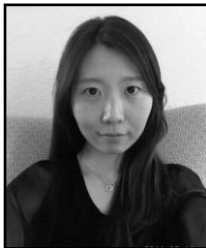
the download calculation period were different in each open source hosting sites, it was hard to level the data and that the data adjusting was carried out not by using the weighted value for each site but through the mechanical adjustment process by simply dividing the data into sections.

In the future studies, the rest factors of TDD and refactoring in the research model will be analyzed to evaluate the users' satisfaction. With this, it is expected to raise the users' satisfaction in the open source project of diverse conditions and consequently contribute to enhancing the maturity of open source community.

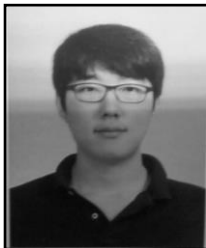
References

- [1] K. Beck, "Extreme programming explained: embrace change", Addison-Wesley Professional, (2000).
- [2] K. Schwaber, "What is Scrum? Scrum: It's about common sense", <http://www.controlchaos.com/about/>, (2007).
- [3] B. Russo, "Agile technologies in open source development", IGI Global., (2009).
- [4] S.W. Shin and H.K. Kim, "Agile Framework for SOA-based Application Development", The KIPS Transactions: PartD, vol. 16, no. 1, (2009), pp. 55-64.
- [5] A. Tucker, R. Morelli and C. D. Silva Editors, "Software Development: An Open Source Approach", CRC Press, Inc., (2011).
- [6] H. J. Sohn, M.G. Lee, B.M. Seong and J.B. Kim, "A Study of Influence of the Value-Practice of Agile Development Methodology on the Open Source Project Community", In proceedings of International Conference Future information and Communication Engineering, ICFICE2015, (2015), pp. 399-402.
- [7] M.G. Lee, "A Study of Contents Convergence Technology and Service", Korea Contents Association, vol. 12, no. 2, (2014), pp.78-84.
- [8] Sourceforge, <https://sourceforge.net>
- [9] github, <http://github.com>
- [10] google code, <https://code.google.com/>

Authors



Hyo-Jung Sohn, she received her bachelor's degree of Business Administration in Soongsil University, Seoul(2006). And she is studying her master's degree of software engineering in the Graduate School of Software, Soongsil University, Seoul. Her current research interests include pen source development and management information system.



Min-Gyu Lee, he received his bachelor's degree of Information and Telecommunication in Dongguk University (2014). And he is studying his master's degree of software engineering in the Graduate School of Software, Soongsil University, Seoul. His current research interests include Open source software and Security.



Baek-Min Seong, he received his bachelor's degree of Business Administration in Soongsil University, Seoul(2014). And he is studying his master's degree of software engineering in the Graduate School of Software, Soongsil University, Seoul. His current research interests include database.



Jong-Bae Kim, he received his bachelor's degree of Business Administration in University of Seoul, Seoul(1995) and master's degree(2002), doctor's degree of Computer Science in Soongsil University, Seoul(2006). Now he is a professor in the Graduate School of Software, Soongsil University, Seoul, Korea. His research interests focus on Software Engineering, and Open Source Software.