ANALYSIS OF DEVELOPMENT COOPERATION WITH SHARED AUTHORING ENVIRONMENT IN ACADEMIC SETTING

Lei Wu[1], Sharon White[2], Yi Feng[3]

[1,2]Software Engineering, University of Houston-Clear Lake, Houston, U.S.A
wul,whites@uhcl.edu
[3]Computer Science, Algoma University, Sault Ste. Marie, Canada
feng@algomau.ca

ABSTRACT

Team work is an important training element of future software engineers. However, the evaluation of the performance of collaboration among individuals is very subjective. Meanwhile, how to effectively promote the collaboration in an academic setting is an even more challenging task. The lack of a common standard or method for the assessment is a practical issue in software engineering projects. With the rapid development of shared authoring environments, such as Wiki, more and more educational institutions are studying the adaptability of such kind of collaborative platforms. In order to study the applicability of adopting wiki-based shared authoring environments in software engineering education, we have proposed three major research questions. By solving these problems, we try to answer some of the most important questions in adopting shared authoring platforms in academic settings.

.

KEYWORDS

Shared authoring platform, collaboration evaluation, task tracing, quality control

1. INTRODUCTION

Adopting Wiki-based shared authoring platforms in software engineering program is a newly emerged practice among many educational institutions [1][7][9]. The effectiveness and adaptability varies largely based on different education objectives [4][5][11]. In order to study the applicability of adopting wiki-based shared authoring environments in software engineering education, we have proposed three major research questions. By solving these problems, we try to answer some of the most important questions in adopting wiki in academic setting environment.

Based on these research questions, we closely monitored five senior software engineering projects. We have studied the effectiveness of using wikis-based environments to support software engineering development process in an educational setting. We further discuss the experiences and the discoveries learned from the development activities, and we also discuss the limitations of applying shared authoring environments in software engineering education.

2. RESEARCH QUESTIONS

The sponsored senior software engineering projects and the use of real stakeholders with close to industry-level requirements are critical to our research questions. These construct a suitable environment for us to monitor and evaluate the effects of certain types of software engineering practices and corresponding support environments. In our study, we focus on wikis-supported shared authoring environments. We have proposed following three major research questions:

1. Would the use of web-enabled groupware cause the students to have a better understanding of the stakeholders, their needs, and the development process itself?
2. What types of such groupware will gain more advantages in terms of collaboration support for each side of stakeholders?
3. What are the limitations of applying such systems in software engineering educational settings?

3. SOFTWARE ENGINEERING DEVELOPMENT PROJECTS

We have selected five software engineering projects in academic setting as individual study cases to analyze the effectiveness of using wiki-based development supporting systems in a typical software engineering project. The author has worked as faculty coach for these five projects. Detail project process data has been collected. The project team normally consist of four to five senior software engineering students. The development length is six months. The projects are proposed by a variety of commercial companies or not-for-profit organizations across the country in the USA. By nature, each project is an example of collaborative development involving at least two development stakeholders who are geographically separated. This presents a great challenge of distributive software development. How to support the collaborations becomes an important issue for such type of development activities. Following is a detail description of each project.

I. SMN Project (Shared Multimedia Network for Picture Frames)

Figure 1. Shared Multimedia Network for Picture Frames

The primary task of this project is to create a platform that will allow Kodak Research Lab to implement systems for transferring multimedia content (picture, music, and video files) between different personal devices in a household environment. Remote devices can also be accessed through an Internet connection. The connection between devices should be seamless and transparent (ad-hoc) to both experienced and inexperienced users. The secondary task is to demonstrate a few possible use cases by creating applications built upon the platform.

II. ITW Project (Intellisync Troubleshooting Wizard for NOKIA Mobile E-mail System)

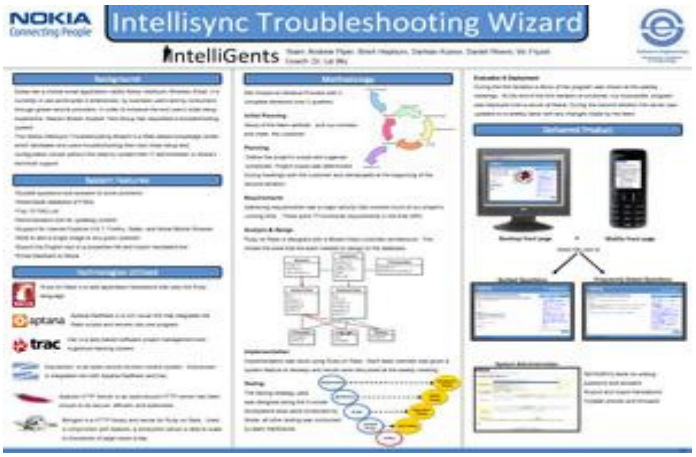

Figure 2. Intellisync Troubleshooting Wizard for NOKIA Mobile E-mail System

The Intellisync Troubleshooting Wizard is a self-help system designed to allow cell-phone users to troubleshoot their own mobile e-mail configuration problems regarded to the Intellisync system. The system displays a series of questions and predetermined responses that will guide the user toward an optimal solution to their Intellisync mobile e-mail service difficulties. The resulting product is a web-based solution with future extension to host on mobile device.

III. CASG Project (Computer Aided Scenario Generator for Environmental Decision Making)

Figure 3. Computer Aided Scenario Generator for Environmental Decision Making

This project is a part of a multi-disciplinary project which will assist policy analysts in understanding the impact of greenhouse gas emissions reduction policies on the automobile sector. It seeks to gather the interface requirements for policy analysts using multiple modeling systems to generate and execute policy scenarios. It also seeks to define the requirements and architecture for the underlying model interactions and how the input and output of these models

will be visually presented to and manipulated by the policy analysts. The key facets of the interface explorations is the abstraction and introspection of complex model input and output, the aggregation of input variables, and a documentation support system for the models being used.

IV. FADPR Project (Friedreich's Ataxia Disease Patient Registry)

Figure 4. Friedreich's Ataxia Disease Patient Registry

The patient registry captures demographic and basic clinical information from individuals with Friedreich's Ataxia (FA) Disease worldwide through an interactive website. The registry stores the information collected and FARA (FA Research Alliance) will be able to query the database for researchers and pharmaceutical companies who need candidates for clinical trials. A patients' basic contact information and personal health information (PHI) will be housed by the registry. Access to patients interested in and eligible for clinical trials will substantially aide in the reduction of clinical research resources, specifically time and cost. Implementation of this solution brings researchers and individuals with FA one step closer to a treatment.

V. SAHS Project (Secure Applications with Hardware Security Modules)

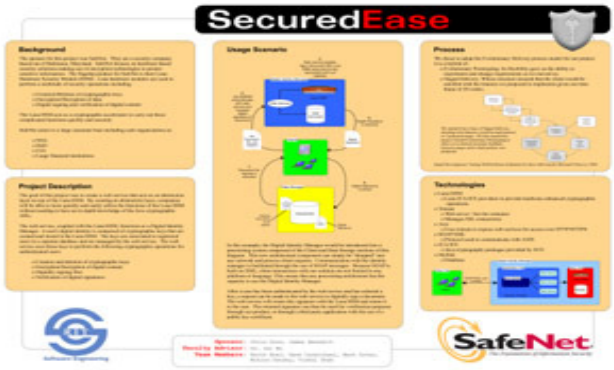

Figure 5. Secure Applications with Hardware Security Modules

The main task of this project is to evaluate the feasibility of developing a web service to interact with the Luna hardware security module (HSM) provided by the sponsor company-SafeNet Inc. The web service will act as a secure generic point of communication to access secure information guarded by the HSM. The web service will manage secure keys held by the HSM to encrypt/decrypt sensitive data held in a database for authenticated users.

4. PROJECT EXPERIMENTS AND WIKI-SUPPORTS ANALYSIS

Each project is proposed by external commercial sponsor or not-for-profit organization. The sponsor is responsible for providing all necessary development resources, as well as financial supports. Student development team will receive hardware and software equipment. In most cases, domain-specific training from sponsors is required for the development project. However, it is mainly the student team's decision on the development environment settings, process selections, and quality control mechanisms, even though sponsors may provide guidance or suggestions.

Therefore, we will see a wide variety of configurations of development environment settings. This gives us an opportunity to monitor the usage of collaborative groupware supporting systems in software project development activities, and evaluate the effectiveness of adopting such kind of systems. It enables us to further answer the research questions presented in section two.

Table_1 provides a brief description of the detail configurations of each project that the we have closely monitored. A comparative study has been conducted to analyze the effectiveness of adopting wiki-based supporting environment in software development projects.

Table 1. Project construction components

Project Name	Sponsor Stakeholder	Collaborative Development Nature & Shared Authoring Groupware Support
FADPR	FARA Friedreich's Ataxia (FA) Disease Research Alliance	• The team collaborated with EDS inc. and FARA both locally and remotely (cross states)
		• No shared authoring groupware support
SAHS	SafeNet Inc.	• SafeNet Inc. is located in Ottawa, Canada. • Remotely collaborated with SafeNet Inc. development team (cross countries)
		• With shared authoring • Not wiki
CASG	Department of Public Policy at Rochester Institute of Technology	• Michigan State University • California State University • Remotely collaborated with researchers from above universities (cross states)
		• With shared authoring groupware • Wiki-based system
ITW	Nokia Boston Lab	• Remotely collaborated with Nokia Boston Lab (cross states)
		• With shared authoring groupware • Wiki-based system with development support
SMN	Kodak Research Lab	• Both locally and remotely collaborated with Kodak Research Lab (within state)
		• With shared authoring groupware • Wiki-based system with development support

5. ASSESSMENT & LESSONS LEARNED

Team work is an important training element of future software engineers. However, the evaluation of the performance of collaboration among individuals is very subjective. The lack of a common standard or method for the evaluation is a practical problem in an academic setting. To alleviate the difficulty, we have adopted a three-point evaluation mechanism to leverage the limitations. This helps us to better monitor the team's performance.

One point of the evaluation process is the pure observation of team performance by the faculty coach. The coach will participate in the team's work twice a week in a fixed team room setting. During these two interactions, faculty coach will listen to team report, give advices, discuss with the students, observe team's interactions with sponsors, and monitor the internal team work involving all members, etc. The other assessment point is the sponsor's evaluation. Sponsors will be invited to participate in team's work at least once a week. Since it is not always possible for sponsors being locally available, a video/voice conference call is normally adopted to service the collaboration purpose. The third assessment point is the peer evaluation from each team member. We provide three different evaluation templates for each part of the stakeholders to evaluate the teamwork as a whole, and each individual student as well. The combined result will more likely to be subjective and fair.

Table 2 shows the assessment result for the five projects. An analysis study has been conducted to further investigate the cause.

Table 2. Assessment and Analysis

Project	Evaluation	Analysis
FADPR	• Major issue arisen for lack of communication • Sponsor pointed out one student works superior than others • Same student was pointed out as "Problem Creator" by all other students	• The team collaborated with EDS inc. (Sponsor 1) locally – relatively smooth • With FARA (Sponsor 2) remotely – mostly difficult
		• There's lack of communication about what each individual is currently working on and the progress • Overlaps of same work task among different students • Without shared authoring platform support, remote collaboration with sponsor 2 is most painful
SAHS	• Minimal collaboration issues during the middle of the process • Solved quickly • The sponsor company is very satisfied with the progress and final result	• Communication among two teams from two countries is a challenge • Lack of face to face discussions
		• Internet-based shared authoring groupware helped the team to smoothly convey development information to sponsor • When issues do arise, the web discussion from shared authoring platform ease the way of collaboration
CASG		• Multiple stakeholders provided multi-dimensional

	• Different viewpoints from different research groups creates big problems for students • Lack of standard terminology of requirement	requirements • Confusion among these requirements
		• Shared authoring platform Doku-Wiki provides a debate & agreement-making platform for development team to communicate with multiple stakeholders • Problems were solved quickly
ITW	• Consistent requirement change • Smooth interaction with Nokia Boston Lab	• Sponsor has made several major requirement changes during the whole development process
		• Shared authoring platform Trac-wiki has been proven as a successful supporting platform • Acted as both information portal and development center
SMN	• Two students significantly lack of contribution • Collaborative development is boosted by Trac-Wiki environment	• Kodak Research Lab noticed the inconsistency
		• Shared authoring platform Trac-Wiki successfully captured the lack of development contribution from specific students • The development bug ticketing feature from Trac-Wiki promoted the collaborative bug shooting, and increased the efficiency

From table 2, we can find out some interesting results. Project FADPR hasn't adopted any collaboration enforcement mechanism. It has witnessed severe issues that related with collaborative team-work. Project SAHS and CASG have used web-based groupware and Doku-Wiki [2] respectively to support the collaborations among development team members and stakeholders. Both have successfully solved the challenges that they have faced during the process of development. Project ITW and SMN have adopted Trac-wiki, a development-supported wiki platform, to foster the collaborative development activities among different teams. The experimental result shows that, Shared authoring platforms, the Trac-wiki and Doku-Wiki supporting environments are crucial for the success of these two projects. Meanwhile, Trac-wiki also captures the weakness among individual team members during different stages of the project. This provides a more substantial tool for project managers to monitor the progress of the project and the performance of individual member.

6. DISCUSSION AND CONCLUSION

Reviewing the three major research questions we have presented in section two, we would like to answer these questions based on the lessons we learned from our experiments.

Question 1: Would the use of groupware cause the students to have a better understanding of the stakeholders, their needs, and the development process itself?

Answer: Our response is yes. With the experience of applying shared authoring platform, we have noticed that, the understanding of the stakeholders' background and needs have been enhanced. This is achieved though the multiple versions of documenting customers' requirements, clarify the uncertainties, answering development team and customers' questions. On the other hand, the understanding of the current stages of development process is further

enhanced through the publication of project progress. All these are accommodated by the adopted shared authoring groupware platforms.

Question 2: What types of such groupware will gain more advantages in terms of collaboration support for each side of stakeholders?

Answer: wiki-based shared authoring platform environments have shown great advantages in this domain. There're two major factors that make wiki more capable than the majority of other general purpose groupware. First, wiki-based systems present a robust knowledge sharing center[8][11], which have simple and similar user interface for developers to use. This is very important for the success of software development project[9][12]. Secondly, wiki-based systems provide a variety of different supporting environments ranging from collaborative development process to testing, integration, transition, and maintenance, etc. These cover almost every aspect of software development process.

Question 3: What are the limitations of applying such systems in software engineering educational settings?

Answer: the major limitations of applying shared authoring systems in software engineering educational setting lie on two aspects. One is the learning curves of many wiki-based shared authoring platforms[18][20]. For example, in order to use Doku-Wiki and Trac-Wiki, user has to learn the syntax. This will impose a certain kind of learning activities to master them during the usage[19][16], even though this may not be a significant obstacle. The second limitation lies on the lack of proper usages of different types of shared authoring systems among different participants. Indeed, this is crucial to the success of applying such systems in development project[15][[6]. Shared authoring platform only provides the possibility of fostering collaboration[13][14]. However, it cannot guarantee the quality of collaboration [7][10]. There's a lack of a general mechanism to balance the enthusiasm and aloofness among different participants. Meanwhile, it also lacks of a mechanism to establish the authority level among a large number of co-authoring results.

With the developing and testing an explosion of ideas and practices for shared authoring platform community [2][3][17][21], we will greatly benefit from its development results. We strongly believe that the future wiki-based shared authoring platform will be designed not only suitable for software engineering activities, but also suitable for the vast majority of the "online generation", either it is project-oriented or organization-oriented.

REFERENCES

[1] Gotel, O. Scharff, C. and Seng, S. (2006), "Preparing Computer Science Students for Global Software Development", Proceedings 36th IEEE Annual Frontiers in Education Conference

[2] Dokuwiki, (2012),. http://wiki.splitbrain.org/wiki:dokuwiki.

[3] Trac-Wiki, (2012), http://trac.edgewall.org/wiki/TracWiki

[4] Decker, B.; Ras, E.; Rech, J.; Jaubert, P.; Rieth, M. (2007), "Wiki-based stakeholder participation in requirements engineering". IEEE Software March/April.

[5] W. Xiao, C. Chi, and M. Yang.(2007), "On-line collaborative software development via wiki". In Proceedings of the 2007 International Symposium on Wikis, pages 177–183, ACM.

[6] G. Oster, H. Skaf-Molli, P. Molli, and H. Naja-Jazzar. (2007), Supporting Collaborative Writing of XML Documents. In ICEIS (4), pages 335–341

[7] Sclater, M., & Bolander, K. (2004). Factors influencing students' orientations to collaboration in networked learning. In P. Goodyear, S. Banks, H. V & D. McConnell (Eds.), Advances in research on networked learning (pp. 175-203). Boston: Kluwer Academic Publishers.

[8] Lonchamp, J. (2006). Supporting synchronous collaborative learning: A generic, multi-dimensional model.International Journal of Computer Supported Collaborative Learning, 1(2), 247-276.

[9] Kasemvilas, S., & Olfman, L. (2009). Design alternatives for a MediaWiki to support collaborative writing in higher education classes. Issues in Informing Science and Information Technology, 6, 45-64.

[10] Helic, D., Maurer, H., & Scerbakov, N. (2002). Aspects of collaborative authoring in WBT systems. Proceedings of the International Conference in Infrastructure for Electronic Business, Education, Science,and Medicine on the Internet

[11] Lafifi, Y., Halimi, K., Ghodbani, A., & Salhi, N. (2009). Learners monitoring based on traces in CSCL system. Journal of Computer Science, 8(2), 61-72.

[12] Lonchamp, J. (2006). Supporting synchronous collaborative learning: A generic, multi-dimensional model.International Journal of Computer Supported Collaborative Learning, 1(2), 247-276.

[13] Serce, F-C., & Yildrim, S. (2006). A web-based synchronous collaborative review tool: A case study of an on-line graduate course. Pedagogical Technology & Society, 9(2).

[14] Soh, L-K., Khandaker, N., Liu, X., & Jiang, H. (2006). A computer-supported cooperative learning system with multi-agent intelligence. AAMAS'06,

[15] Steimle, J., et. al. (2009), CoScribe: Integrating Paper and Digital Documents for Collaborative Knowledge Work. IEEE Trans. on Learning Tech., 2(3):174-188

[16] Piper, A.M., et. al. (2012), TAP & PLAY: An end-user toolkit for authoring interactive pen and paper language activities. In Proc. of CHI

[17] Norman Heino, Sebastian Dietzold, Michael Martin, and Soren Auer. (2009), Developing semantic web applications with the ontowiki framework. In Networked Knowledge- Networked Media, SIC, pages 61-77. Springer

[18] Markus Krotzsch, Denny Vrande_ci_c, Max Volkel, Heiko Haller, and Rudi Studer. (2007), Semantic Wikipedia. Journal of Web Semantics

[19] Florian Matthes, Christian Neubert, and Alexander Steinho (2007), Hybrid wikis: Empowering users to collaboratively structure information. In ICSOFT (1), SciTePress

[20] Sebastian Scha_ert, Fran_cois Bry, Joachim Baumeister, and Malte Kiesel. (2008), Semantic wikis. IEEE Software

[21] D. T., Awang Hamid, M.R., Islam, N., Alias, and A., Hafidz H. Omar. (2010), An Efficient Authoring Activities Infrastructure Design through Grid Portal Technology. In Proc. of The 7th WSEAS International onference on Engineering Education (EDUCATION'10), pp146-151.

Authors

Lei Wu, Assistant Professor of software engineering at University of Houston-Clear Lake, Houston, U.S.A. His major research interests include software engineering with artificial intelligence, secure service-oriented architectures, software for robotics and embedded system intelligence, game software development, and pervasive computing. He can be reached atwul@uhcl.edu

Sharon White, Associate Professor of software engineering at University of Houston-Clear Lake, Houston, U.S.A. Her research interests include domain specification languages, architecture design languages, and software architecture. She can be reached at whites@uhcl.edu

Yi Feng, Associate Professor of computer science, Algoma University, Sault Ste. Marie, Canada. Her major research interests include formal verification, software engineering, signal processing, and system description languages. She can be reached at feng@algomau.ca