Finding Success in Rapid Collaborative Requirements Negotiation Using Wiki and Shaper

Abstract
Defining requirements without satisfying success-critical stakeholders often leads to expensive project failures. Enabling interdisciplinary stakeholders to rapidly and effectively collaborate in development of globally usable software-intensive systems remains a major challenge. At USC, 32 real-client, graduate-level team projects experimented with using the wiki-based requirements negotiation support tool WikiWinWin over a two-year period. Data collected from these projects indicated project outcome is correlated with several usage aspects, including early use, amount of use, frequency of use, shaper use, and evolving of negotiation artifacts. Several changes made based on our first-year’s experience also showed improvements in cost-effectiveness. User feedback generally confirmed that using a wiki-based negotiation tool was beneficial, and that improving on wiki-tool ease of use would yield further client satisfaction in the future.

1. Introduction
Defining requirements without satisfying success-critical stakeholders often leads to negative outcomes, such as expensive rework, loss in market share, customer dissatisfaction, etc. [1, 2, 3]. Frequently software projects fail because some key stakeholders are left in a lose position and decide not to participate, which eventually turns other “winners” into losers. To avoid win-lose situations, the Win-Win approach [4] involves having success-critical stakeholders participate in a collaborative negotiation so they can reach a mutually satisfactory set of requirements.

However, great collaborative challenges remain in having multi-culture, multi-disciplinary stakeholders rapidly form design coalitions to “compete on Internet time” [5]. Such challenges are also evidenced by books such as Friedman’s The World Is Flat [6], which characterizes the successful enterprises of the future as excelling in time-critical, globally distributed, information technology-intensive, interdisciplinary collaborations; and by series such as the Gartner Reports [7] showing the extreme swings in global collaborative project success (up to 50% failure rates) when intercultural and interdisciplinary collaboration is done well or poorly.

To help interdisciplinary stakeholders more easily and rapidly understand each other in designing software-intensive systems, we developed a wiki-based requirements negotiation support tool – WikiWinWin. Its underlying theories are the Value-Based System and Software Engineering (VBSSE) framework [8, 9], and the win-win equilibrium theory [36]. The Wiki technology [10] provides a more flexible and general-use Wikipedia-like platform for collaboration. We also adopted a successful practice of wiki use, the shaper role [11], from relevant research.

In fall 2007 (Year 1), we experimented with using WikiWinWin at University of Southern California (USC) graduate software engineering course. 20 real-client, graduate-level, e-services team projects used the WikiWinWin tool to negotiate their requirements. In an earlier paper [26], we reported our initial experience and results. A major finding was project teams tended to achieve better outcomes as the use of the tool increased. As stated in that paper, we made several suggested improvements to reduce the cognitive cost and increase the benefit to requirements negotiation, and experimented again in fall 2008 (Year 2) with another 12 projects. Based on data collected on the 32 project teams over the two-year period, we analyzed critical factors of use that tend to lead to successful outcomes. A number of hypotheses were formulated and tested. In addition, we summarized user feedback on using the wiki-based negotiation tool. This paper presents the latest results.

Though there are many studies reported the success use of Wiki in industry [23, 30], there still lacks empirical study on how the wiki and shaper can contribute to the success of collaboration in software requirements negotiation. Our study provides an example of using wiki-based groupware in the domain of software requirements negotiation. By analyzing the success factors and drawbacks of the WikiWinWin system, we may also contribute to the empirical research on successful transition of groupware systems.

The rest of the paper is organized as follows: Section 2 introduces related work and elaborates on the nature of the e-services projects and major WikiWinWin changes. Section 3 describes our analysis approach. Section 4 presents the results. Section 5 discusses major findings and threats to validity.
Section 6 provides conclusions and directions for future research.

2. Background

2.1 WinWin Equilibrium Theory

The meta-model underlying the WikiWinWin tool is the win-win negotiation model [36]. This model consists of four types of artifacts – Win Condition, Issue, Option, and Agreement.

- **Win Condition** – captures individual stakeholders’ desired objectives.
- **Issue** – captures conflicts between win conditions and their associated risks and uncertainties.
- **Option** – identifies candidate solutions to resolve an issue.
- **Agreement** – captures shared commitment of stakeholders with regard to accepted win conditions or adopted options.

![Figure 1: win-win negotiation model](image)

The win-win equilibrium theory [36] links win conditions, issues, options and agreements. It establishes a win-win equilibrium state where all win conditions are covered by agreements and all issues are resolved by options covered by agreements, and shows how this state is reached from other states.

2.2. Related Work

Stakeholder collaboration is the key success factor for requirements negotiation. During the negotiation, stakeholders have to collaboratively and incrementally find out what has to be built. And the requirements have to accommodate to all stakeholders goals and expectations.

The typical collaboration activities in requirements negotiation are sharing knowledge, understanding each other, resolving the conflict of interests, co-authoring software requirements, etc. Since 1994, five generations of collaboration support systems has been developed at USC to cope with the software requirements negotiation issues [12]. The collaboration technologies, negative results and success factors from previous tool generations have driven evolution of these requirements negotiation support systems.

The previous EasyWinWin system [13] was based on a Group Support System (GSS). EasyWinWin had been very good in capturing initial requirements. However, it has been less easy to adapt to the evolving nature of requirements. This led us to adopt a more flexible and general-use Wiki platform for developing the current WikiWinWin support tool.

Wiki is a collaboration technology becoming popular in recent years. It is a program makes it exceptionally easy for a group of collaborating users to manage, manipulate, and publish knowledge in web pages [14, 15]. These strengths of Wiki are very helpful for the sharing and co-development of knowledge during requirements negotiation. There are two types of contributions in using wiki: 1) contributing knowledge to the wiki; 2) contributing primarily by refactoring the works of others, also known as shaping [11]. Participants who perform the role of shaping are called shapers.

The studies of collaboration theories and groupware systems revealed a number of success factors. For example, Applegate’s model [16] proposed several stimulators and inhibitors of technology transfer, such as end users’ frustration with current processes, and using reward structure to support the use of innovation. Davis’s Technology Acceptance Model (TAM) [17] suggests that when users are presented with a new technology, a number of factors influence their decision about how and when they will use it, notably: perceived usefulness, and perceived ease of use. Field studies of GSS found that user friendliness [18], positive user feedback [19], and cognitive cost [20] are all determinants of sustainable use of a groupware. Briggs [21] proposed several guidelines for creating effective group support systems: 1) focusing on creating solutions for mission-critical collaborative tasks; 2) designing effective collaborative processes; 3) packaging facilitation skills and reduce GSS-usage conceptual load. EasyWinWin incorporated some useful collaboration techniques [22] to facilitate effective group interaction. These are used in WikiWinWin as well.

Additionally, we adopted a successful practice of wiki use - the shaper role from Majchrazk’s studies on corporate wiki usages [23, 24]. She found that one of the success factors in many cases is the shaper role. She argued that although the shaper may not have the same level of domain expertise as the knowledge contributors do, but the shaper is more effective in using wiki to identify business opportunities. Thus, they are able to help groups collaborate more effectively.

In summary, WikiWinWin is an integrated approach that leverages the Wiki collaboration technology and the successful practice of the shaper role, combines with the Value-Based System and Software Engineering (VBSSE) framework and
2.3. Projects

The study deals with 32 project teams at the USC software engineering course [27, 28]. All these teams were formed by first year graduate students and project clients. Students form their own teams. Typically each team consists of 6 on-campus students, 2 off-campus students, and 1 client representative. Off-campus students usually work full-time and have a lot of experience in the software engineering field. The majority of on-campus students come directly from undergraduate programs with little work experience. Although team composition generally stays the same, the quality of students may be different between years. The project clients come from USC-neighborhood small business, non-profit, and campus organizations needing improved e-services applications. We assign projects to teams based on their preferences. Table 1 lists a summary of these projects.

These projects were scoped with implementable and deployable within 24 weeks. In year2, we had several COTS/ web services intensive projects, as highlighted in gray color in Table 1.

Before project teams formally started negotiations, they went through some front-end mutual learning activities and training. During the negotiation, we used group-based access control to manage multiple teams concurrently and independently using the tool. All participants, including clients, were knowledge contributors in the negotiation. In addition, there were two shapers on each team, one was assigned to an off-campus student, and the other was assigned to an on-campus student.

The negotiations were carried out in both meetings and off-line situations. During the meetings, participants brainstormed win conditions, discussed issues and options. Aside from meetings, participants were free to use the tool at their convenience, as often as needed, for reviewing, commenting, adding, and prioritization.

The negotiation results provided basis for developing other project artifacts. For example, the prioritized agreements transformed into formal requirements; the unresolved issues represented risks that should be managed by the risk mitigation strategy, etc. The project package included the following artifacts: Operational Concept Description (OCD), System and Software Requirements Description (SSRD), System and Software Architecture Description (SSAD), Life Cycle Plan (LCP), and Feasibility Evidence Description (FED). Artifacts were graded at milestone anchor points for quality and completeness.

At the end of the semester, students submitted critiques on the various aspects of the course improvements, including the WikiWinWin tool.

<table>
<thead>
<tr>
<th>Table 1: Project Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>S#</td>
</tr>
<tr>
<td>----</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>11</td>
</tr>
<tr>
<td>12</td>
</tr>
<tr>
<td>13</td>
</tr>
<tr>
<td>14</td>
</tr>
<tr>
<td>15</td>
</tr>
<tr>
<td>16</td>
</tr>
<tr>
<td>17</td>
</tr>
<tr>
<td>18</td>
</tr>
<tr>
<td>19</td>
</tr>
<tr>
<td>20</td>
</tr>
<tr>
<td>21</td>
</tr>
<tr>
<td>22</td>
</tr>
<tr>
<td>23</td>
</tr>
<tr>
<td>24</td>
</tr>
<tr>
<td>25</td>
</tr>
<tr>
<td>26</td>
</tr>
<tr>
<td>27</td>
</tr>
<tr>
<td>28</td>
</tr>
<tr>
<td>29</td>
</tr>
<tr>
<td>30</td>
</tr>
<tr>
<td>31</td>
</tr>
<tr>
<td>32</td>
</tr>
</tbody>
</table>

*gray color indicates COTS/ web services intensive

2.4. Major Changes

2.4.1. WikiWinWin

A few changes were made to the WikiWinWin tool and process based on the first-year’s experience. These were summarized in Table 2. In addition, we also made
Pre-negotiation team building. Finding the right set of people for the project is critical for success. Usually on-campus students had chances to mingle with others. Off-campus students, on the other hand, didn’t have much interaction with the rest of the class. Based on feedback from year1, we organized a classroom social as well as used the course discussion board to facilitate team building in year2.

Site visits. Another lesson we learned in year1 is that students need to have a better understanding of the operational concept of their clients’ organization, for which the new system is designed. In year2, we added a site visit assignment as another front-end mutual learning activity. Students are asked to visit their client’s organization and produce a report of learning in terms of organizational objectives, shortfalls in the current operation, and major needs for improvement.

Training. In year2, we revised tutorials by adding negotiation process diagrams and step by step examples. We added hands-on practice to the homework assignment.

Shaper assignment. We found in Year1 that having off-campus shaper-facilitated meetings was not fully effective. In some teams, two shapers overwrote each other’s work in the wiki. In Year2, we let the off-campus shaper to be the lead shaper on shaping wiki content. The on-campus shaper was primarily the facilitator of negotiation meetings, but also helped the lead shaper in shaping overlaps and issues among stakeholder inputs.

2.4.2. Other Changes

In addition to the changes made to the WikiWinWin tool and instructions, there were some significant differences in other aspects of the course. All these changes confounded the project outcomes. As a result, we cannot fully compare class-project performance between year1 and year2.

Process model. In year1, project teams followed our previous LeanMBASE Model [8]. In year2, we introduced several process and product improvements via our new Incremental Commitment Model [9].

Process guideline. In year2, we also experimented with using electronic process guidelines [29] developed using the IBM Rational Method Composer toolset.

Project applications. There were a higher percentage of projects in year2 using web services compared to year1.

3. Analysis Approach

3.1. Data Collection and Metrics

As stated in Section1, an objective of the analysis was to identify critical factors of use that tend to lead to successful outcomes. This section describes the data collection and the corresponding metrics.

Tool use is measured by numbers of transactions, and number of days the tool is used. Wiki logs time-stamped transactions with user identity when users clicked on ‘save’. We obtained these usage data directly from the wiki log files using the Linux ‘grep’ command. The usage data was collected at two places: at the end of the 12-day negotiation period, and at the end of the inception phase about three weeks later. Because there was a difference in the number of calendar days between year1 and year2, using a fixed time frame - the 12-day negotiation period was preferred for keeping consistency. In addition, the use measured at the end of the inception phase was likely to be affected by the assignment deadline. This will be discussed in Section 4.

Project outcome or success defined as having a perfect grade on the project package, which includes
prototypes and documents for the operational concept, requirements, architecture and plans. Tool success is measured by numbers of positive and negative comments in user critiques. The outcomes were taken from graded project artifacts as well as student critiques. The graded project artifacts were measured by the project package grade at the end of the inception phase in the Incremental Commit Model process [29]. Two teams in year2 were excluded from correlation testing because their project packages were missing operational concept artifact and requirements artifact. The critiques were collected at the end of the semester.

Usage Metrics:

- Number of usage contributions by team. This is the sum of contributions by everybody in the project.
- Number of usage contributions by shaper(s). This is the sum of contributions made only by shaper(s) in the project.
- Number of team usage days. This is the total number of calendar days that the tool is used at least once by a project team.
- Number of shaper usage days. This is the total number of calendar days that shaper(s) in the project used the tool at least once.
- Number of days creating artifacts. This is the total number of calendar days that at least one win condition, issue, option, or agreement was created.

Outcome Metrics:

- Number of points lost in project package grade. The loss of grade indicates shortfalls in thoroughness, clarity, and degree of issue closure.
- Number of positive comments and negative comments of WikiWinWin tool from critiques.

3.2. Hypotheses

To examine the relationship between usage and outcome, a general null hypothesis was the lack of correlation between various usage aspects and outcome aspects. Five individual hypotheses that have a possible cause-and-effect relationship with a project’s outcome were:

H1: Higher or lower numbers of usage contributions by team has no correlation with loss in project package grade.

H2: Higher or lower number of usage contributions by shaper has no correlation with loss in project package grade.

H3: Higher or lower number of team usage days has no correlation with loss in project package grade.

H4: Higher or lower number of shaper usage days has no correlation with loss in project package grade.

H5: Higher or lower number of days creating artifacts has no correlation with loss in project package grade.

We attempted to disprove the above null hypotheses by using regression. Since we use loss in grade, the correlation coefficient should be negative. The log() was used because the data reflected an exponential distribution. A significance level below 0.05 indicates strong support for rejecting a hypothesis.

4. Results

4.1. Early Usage vs. Milestone Usage

As mentioned in Section 3.1, we collected tool usage at the end of the 12-day negotiation period (early usage) as well as at the end of the inception phase (milestone usage). To illustrate whether early usage or milestone usage is more appropriate for testing the hypotheses, we compared the Pearson correlation coefficients among team early usage, team milestone usage, and loss in project grade, as shown in Table 3 and Table 4. Cohen and Holliday [31] suggest the following rule of thumb to interpret the Pearson’s coefficient: 0.19 and below is very low correlation; 0.20 to 0.39 is low correlation; 0.40 to 0.69 is modest correlation; 0.70 to 0.89 is high correlation; and 0.90 to 1 is very high correlation.

<table>
<thead>
<tr>
<th>Year1 Team</th>
<th>Milestone usage</th>
<th>Early usage</th>
<th>Log(loss)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Milestone usage</td>
<td>1</td>
<td>0.67</td>
<td>-0.49</td>
</tr>
<tr>
<td>Early usage</td>
<td>1</td>
<td>-0.61</td>
<td></td>
</tr>
<tr>
<td>Log(loss)</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Year1 early usage vs. milestone usage

<table>
<thead>
<tr>
<th>Year2 Team</th>
<th>Milestone usage</th>
<th>Early usage</th>
<th>Log(loss)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Milestone usage</td>
<td>1</td>
<td>0.69</td>
<td>-0.39</td>
</tr>
<tr>
<td>Early usage</td>
<td>1</td>
<td>-0.60</td>
<td></td>
</tr>
<tr>
<td>Log (loss)</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Year2 early usage vs. milestone usage

In both years, team early usage and team milestone usage are correlated, however, early usage has a stronger correlation to the loss in project grade than the corresponding milestone usage: (-0.61) vs. (-0.49) in year1, (-0.60) vs. (-0.39) in year2. A possible reason is that some teams rushed with last-minute, low quality use when the assignment deadline was approaching. A breakdown of year2 usage by calendar week confirmed this ‘deadline effect’, as shown in Figure 1. The horizontal axis in Figure 1 indicates project team # (same as Table 1); the vertical axis is the number of total usage. The WinWin report was due in Week 3, which significantly influenced a few teams. This comparison supported our preference of choosing...
usage data measured at the end of 12-day early period for the subsequent hypothesis testing. It also provided evidence suggesting relationships between quality early usage and project outcome.

**Figure 2: Year2 weekly usage**

### 4.2. Loss in outcome vs. number of usages

H1: Number of usage contributions by team has no correlation with loss in project package grade.

H2: Number of usage contributions by shaper(s) has no correlation with loss in project package grade.

Figure 2 illustrates a linear relationship between log (loss in grade) and number of tool usage contributions by team, fitted with year1 data. The coefficient of determination ($R^2$) has value of 0.3672. A similar relationship is also confirmed with year2 data, as shown in Figure 3.

**Figure 3: Year 1 loss in grade vs. usage by team**

There are two points in Figure 3 that are far from the fitted regression line: (63, 0.641) and (254, 1.307). The team who had a smaller amount of usage (63 contributions) had used the tool more frequently (7 out of 12 days), compared to the other team (254 contributions) but only used the tool 4 days in the same period. This illustrates another interesting phenomenon: the frequency of use may be a useful factor that affects the project grade.

The relationship between log (loss in grade) and number of tool usage contributions by shaper(s) is confirmed with both year1 and year2 data, shown in Figure 4 and Figure 5 respectively. Again, both years have similar ($R^2$) value, but year1 has stronger significance value than year2.

**Figure 4: Year 2 loss in grade vs. usage by team**

**Figure 5: Year 1 loss in grade vs. usage by shaper**
4.3. Loss in outcome vs. number of usage days

H3: Number of team usage days has no correlation with loss in project package grade.
H4: Number of shaper usage days has no correlation with loss in project package grade.

Following the observation in Figure 3, we further plotted the relationship between log (loss in grade) and number of team usage days, fitted on year2 data. As shown in Figure 6, there is a clear linear correlation with the coefficient of determination (R^2) value of 0.725. The relationship between log (loss in grade) and number of shaper usage days is also confirmed with year2 data, as shown in Figure 7. After excluding one outlier, the (R^2) value is improved from 0.28 to 0.48.

4.4. Loss in outcome vs. number of days creating artifacts

H5: Number of days creating artifacts has no correlation with loss in project package grade.

Figure 8 illustrates a clear linear relationship between log (loss in grade) and number of days spent in creating negotiation artifacts, fitted on year2 data. It has a coefficient of determination (R^2) value of 0.596. This indicates that teams which spent only one or two days on creating negotiation artifacts had more shortfalls in their project packages than the teams who evolved their artifacts over the negotiation period.
4.5. User Feedback

Table 5 summarizes the positive and negative comments on the WikiWinWin tool from student critiques. There is a high consensus that WikiWinWin helps project teams get good results in their requirements negotiations. The most positive aspects of the usefulness include:
- Involve all stakeholders to collaborate
- Gather needs
- Better understanding of needs
- Communicate needs
- Prioritize on needs
- Eliminate paper work
- Allow individual use at convenience
- Keep history for tracking and future reference
- Capture important terms
- Generate border and deeper requirements

Table 5: Student Critiques Summary

<table>
<thead>
<tr>
<th>Year 1</th>
<th>Positive counts</th>
<th>Negative counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Useful for getting good results</td>
<td>83</td>
<td>103</td>
</tr>
<tr>
<td>Easy to use</td>
<td>37</td>
<td></td>
</tr>
<tr>
<td>UI</td>
<td>32</td>
<td></td>
</tr>
<tr>
<td>Easy to learn</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>Efficiency</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>Robustness</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td><strong>83</strong></td>
<td><strong>103</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Year 2</th>
<th>Positive counts</th>
<th>Negative counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Useful for getting good results</td>
<td>57</td>
<td></td>
</tr>
<tr>
<td>Easy to use</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>UI</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Easy to learn</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Efficiency</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Robustness</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td><strong>57</strong></td>
<td><strong>54</strong></td>
</tr>
</tbody>
</table>

Another observation in Table 5 is that in Year1, the number of negative counts was much higher than the number of positive counts. In Year2, we got more positive comments than negative comments. This indicates that we started seeing some positive effects from changes in Table 2, such as improving tool robustness, reducing learning curve and complexity, automating manual tasks, and simplifying negotiation process.

However, usability and making it easy to use remain major needed improvements. Key shortfalls include: some clients could not use it independently; difficult to modify artifacts; not straightforward for adding issues and options; and need simplified editing and navigation. Contrary to our assumption, users didn’t think the default wiki markup edit was easy and convenient. They preferred a more intuitive WYSIWYG (What You See Is What You Get) editor.

5. Discussion

5.1. Success Usage Factors

Among the various results presented in Section 4, we discuss the most significant factors that tend to influence the project outcomes and their implications.
- In our data, early use (during the 12 day negotiation period) had a higher correlation to project grade. This implies that the requirements negotiation process and support tool need to facilitate early stakeholder involvement in the negotiation.
- Higher tool usage tended to correlate with better grades. This implies better outcomes can be achieved when all the project stakeholders are involved and contribute to the negotiation.
- Frequent tool usage by project teams tended to correlate with better grades. This indicates using requirements negotiation tool is not a one-shot activity. When stakeholders are involved throughout the negotiation, better outcomes tend to be achieved.
- Higher and frequent shaper usage tended to correlate with better grades. This implies that the shaper is a success factor. It is important to make sure teams have capable and responsible shapers.
- Duration of creating artifacts tended to correlate with project grade. This suggests the need of continue evolving negotiation artifacts throughout the negotiation period.

5.2. Complementary Practices

It is necessary to point out that our win-win negotiation was instrumented together with a number of collaborative practices and techniques. These include: pre-negotiation team building, site visits and mutual learning, benefit-result chain analysis [32], model clash analysis [33], simplifiers and complicators [34], and concurrent prototyping [35].

A key aspect of our work is to continue identifying such complementary practices. Based on our year1 experience, we added site visits as another front-end activity. In year2 we had more projects using COTS and web services. We learned that we need to do more prototyping on evaluating web service products.
5.3. Threats to Validity

Major threats to internal validity include:

**Non-uniformity of projects.** There were sources of variability among projects. Each project was different from one another in terms of application, technical characteristics, client needs, and communication skills among team members. For example, one team was not very good at prototyping new features. In this case, even though the team was able to quickly reach an agreement with the client’s win condition, their project package grade was not very good because of the prototyping result. Also mentioned in earlier section, we had more projects involve using COTS and web services in year2, which confounded results. In some cases, even though the initial negotiation led to the right ball park with some candidate service products, if developers didn’t know how to evaluate a candidate vs. alternatives, it often limited them to find the most suitable solution. Such phenomena partially explained the lower significance and some outliers had to be excluded from the results due to COTS evaluation timing difficulties.

**Non-equality of team experiences.** Because students self-formed their teams, we didn’t control for equal team experience levels. This led to possible influence on the project outcomes in that more experienced teams may possess the skills to achieve higher grades, and they may also be more conscious of adopting successful practices introduced in the course.

**Other changes in the course.** As mentioned in section 2, changes in development process and guidelines also added confounding effects to year-to-year project outcomes.

The major threat to external validity is the generalization of our results. Our projects, to some extent, are representative of small development teams in industry given that these projects stemmed from concrete real world problems, involved real client representatives coming from variety of sources, developed by teams mixed with graduate-level students (bachelor degree, some problems with English communication) and full-time working professionals who were generally familiar with industry practices. Considering that these teams also involved co-located and remote team members (some located in mid west and east coast), our projects are also somewhat representative of distributed development teams over different times and locations.

6. Conclusions and Future Work

Poor requirements often lead to expensive project failures. Having multi-disciplinary stakeholders collaboratively negotiate mutually satisfactory requirements contributes to success, but remains a major challenge. Wiki is becoming more and more effective in supporting collaborative activities, e.g. the case of Wikipedia, which suggests that wiki can also be used to facilitate the collaborations during requirements negotiation.

We studied 32 software engineering projects at USC who experimented with negotiating requirements using the WikiWinWin tool over a two-year period. The study concluded the following: 1) Better project outcomes were correlated with several critical factors of use, including early use, amount of use, frequency of use, and evolving of negotiation artifacts; 2) User feedback generally confirmed that using a wiki-based negotiation tool was beneficial; 3) Several changes made based on our first-year’s experience also showed improvements in cost-effectiveness; 4) A number of complementary practices, such as front-end team-building, site-visits, mutual learning, concurrent prototyping and COTS/services evaluation, helped project teams achieve better outcomes; 5) Usability and making tools easy to use by non-expert clients remain major challenges.

We adopted the shaper role from successful wiki uses in industry. The significant correlation between shaper usage, team usage, and project outcome indicated that having capable and responsible shaper is another success factor for collaborative negotiation.

Our study provides an example of using wiki-based groupware in the domain of software requirements negotiation. User experience revealed some useful factors to consider when developing similar tool support for negotiation. These include: 1) Provide effective and robust support in gathering, understanding, and communicating needs; 2) Design flexible tool to accommodate individual use; 3) Facilitate knowledge sharing by allowing easy content update and preserving history; 4) Reduce manual effort to improve efficiency.

Future work will continue extending the current approach to address the following issues:

- Improve the ease of use aspect of the WikiWinWin tool.
- Make the negotiation process more flexible to cope with the increasing service-oriented projects.
- Improve feedback collection with a more structured questionnaire for qualitative responses.
- Evaluate improvements using future years’ software engineering team projects.
- Continue with similar analyses to investigate other success factors and complementary practices.

The major threats to external validity are:

- Non-uniformity of projects.
- Non-equality of team experiences.
- Other changes in the course.
7. References