Concept Mapping in Agile Usability: A Case Study

Jeremy Barksdale
Department of Computer Science
Virginia Tech
Blacksburg, VA 24060 USA
barksdale@vt.edu

D. Scott McCrickard
Department of Computer Science
Virginia Tech
Blacksburg, VA 24060 USA
mccricks@vt.edu

CHI 2010, April 10–15, 2010, Atlanta, Georgia, USA.
ACM 978-1-60558-930-5/10/04.

Abstract

In this paper we report on the experience of using our concept mapping approach on an agile software project to assess its fitness. Participants used our novel concept mapping approach over a four week period during the development of a software tool for a local nonprofit agency. Results indicate that our concept mapping approach has value as a visual tool in agile usability environments.

Keywords

Concept Mapping, Agile Usability, Distributed Cognition, DCog, Team Interaction, Collaboration

ACM Classification Keywords

H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.
D.2.2 Design Tools and Techniques.

General Terms

Experimentation, Management

Introduction

Software project teams are diversifying with specialists bringing their own tools and methods. A focus on improving cross-functional relationships in the agile usability community has lead to an increased study of

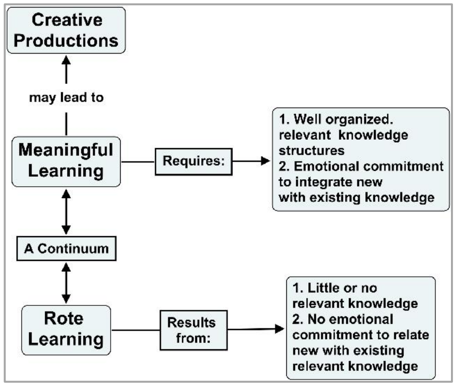

Creative Productions
may lead to
Meaningful Learning
Requires:
1. Well organized. relevant knowledge structures
2. Emotional commitment to integrate new with existing knowledge
A Continuum
Rote Learning
Results from:
1. Little or no relevant knowledge
2. No emotional commitment to relate new with existing relevant knowledge

Figure 1: Sample Concept Map

A concept map representation of Novak's interpretation of creativity as high-level of meaningful learning.

methods that facilitate the integration of activities across these roles [2]. However, more methods that facilitate effective collaboration and communication within these teams are needed. The contribution this study offers to Human Computer Interaction (HCI) is an affordance for greater application of HCI methods and more usable software through the removal of project team barriers. We draw upon existing models and theories as a basis to our novel collaborative approach to meet this need.

Background

Significant research has been conducted on how to deliver more usable software through incorporating the usability role and process on agile software projects. We were interested in focusing considering how knowledge, tacit or explicit, can be distributed between usability and agile software engineers through collaborative concept mapping.

Concept maps afford knowledge representation and organization [3]. Knowledge is captured through a collection of nodes and edges (see Figure 1). Each node is a concept and each edge shows the relationship between concepts. The edges also describe the relationship between concepts through the use of textual annotation. One example of its use is to capture notes in meetings. Topics discussed and persons involved might serve as concepts, and the description about those relationships is provided on the respective edges.

We modified the concept mapping approach prescribed by Novak to fit within the agile community, and to serve as a collaborative artifact that captures scenarios, feature relationships and related design rationale (see Figure 2 for a complete description of the process) [1]. It also serves as a knowledge structure that distributes cognition throughout the team. In this way, it functions as a negotiation tool, facilitating collaboration while allowing team members' autonomy to operate flexibly in their area of specialty without imposing major constraints on the other.

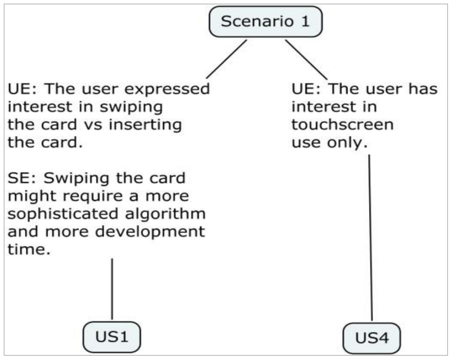
Case Study

The purpose of this project was to study the use of our concept mapping approach to validate its effectiveness. The product was developed over four one week iterations using Java as the programming language, MySQL as the database, Eclipse as the development environment, and CMapTools to manage the concept mapping activities. The product was specified through interaction with the client/user and was not predetermined.

Method

Participants included four upper level international Virginia Tech students with significant programming experience and agile usability training. Collectively, they served as the project team in the following roles: one Usability Engineer, one Project Lead/Software Engineer, one Software Engineer, and one Test Engineer. A local nonprofit mediation agency served as the client and end user. Participants were provided weekly surveys to capture each participant's perspective on the project and our concept mapping approach.

Participants were given an overview of the research study and began the two week training. The training encompassed the extreme scenario based design (XSBD) process [2], and a one week iteration of the


Scenario 1
UE: The user expressed interest in swiping the card vs inserting the card.
SE: Swiping the card might require a more sophisticated algorithm and more development time.
UE: The user has interest in touchscreen use only.
US1
US4

Scenario 1: Customer is withdrawing travel money from an ATM before going out of the country.

US1: Customers identify themselves using a bankcard.
US4: Customers can choose whether a receipt is printed.

Figure 2: Agile Usability Concept Map

The map's parent node labeled "Scenario 1" has child nodes based on the user stories, "US1", "US2","US3", and "US4." Connecting each of the child nodes to a parent node is an edge with text box in which the designers can describe the relationship to capture design rationale for knowledge reuse.

Steps for creating this concept map include: 1) Identification of a design scenario by the usability engineers; 2) Publication of the scenarios by the usability engineers; 3) Creation of user stories and linkage to published scenarios by the software engineers; 4) Collaborative annotation of the links between scenarios and stories; and 5) Agreement on the concept map by the usability and software engineers.

activities required for the actual project. Participants were then introduced to their actual client/user and were encouraged to use our concept mapping approach throughout the process.

Results and Discussion

Overview

The participants used our concept mapping approach during each iteration, but more so in the later iterations. Generally, the usability engineer and a development team member met with the client. The following week, the usability engineer entered scenarios and any features specifically requested by the client into the concept map format. The development team would then enter the remaining features. Following is a summary of the feedback received from the project team about their experiences.

Approach Value

The intention was for our concept mapping approach to improve the communication and collaboration among the project team by visually capturing the relationship between scenarios and features. The participants found the visual nature of the concept mapping approach (see Figure 3) helpful in facilitating collaboration, especially when the team members were in different locations. Participants commented that it served as a map, guide and path. One specifically noted that it helped to "know the weakness of your idea via your peer's evaluation." We believe this benefit is due to the inherent visual nature of the concept map and the ease of looking at a picture versus text.

Team Interaction

The aim was to improve relations among team members through the use of our concept mapping approach. Results showed that the interaction between the development team and the usability engineer fluctuated. Participants reported that, at times, the usability engineer would guess the client's wants instead of validating the needs with the client. One participant characterized it as a battle between the two. The usability engineer reported that they, at times, felt it appeared everyone was not on the same page and that the development team did not understand the scenarios the way they did. One software engineer did note, however, the respect, the ability to criticize the idea and not the person, and the common ground within the team. We believe that concept mapping made these more commonly hidden aspects of interaction more apparent for earlier resolution.

Approach Challenges and Recommendations

Challenges the team faced while using our concept mapping approach primarily included the task of creating a list of features from the scenarios. Participants mentioned experiencing challenges determining which features fit most appropriately with each scenario when creating relationships. One participant expressed difficulty with putting logic to the tasks (i.e. generating the knowledge to fit within the map structure). Another stated they got lost in establishing the relationships. In light of some of these challenges, the participants made ad-hoc modifications to add immediate value. Instead of creating the concept map and then proceeding through the process, they created the prototype in parallel with creating the features to the stories. They also began creating features after at least one or two scenarios were well defined. Complementary to these ad-hoc changes, the participants recommended providing a more involved training session, encouraging team members to discuss

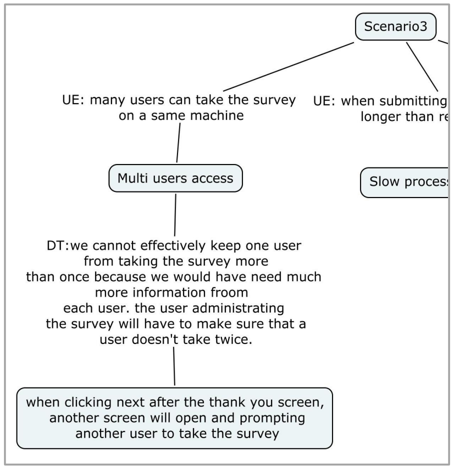

Scenario3
UE: many users can take the survey on a same machine
UE: when submitting
longer than re
Multi users access
Slow proces
DT:we cannot effectively keep one user from taking the survey more than once because we would have need much more information froom each user. the user administrating the survey will have to make sure that a user doesn't take twice.
when clicking next after the thank you screen, another screen will open and prompting another user to take the survey

Figure 3: Project Concept Map Detailed

This concept map detail for one scenario, produced by the project team during development, highlights the collaborative activity among the team members. Note the ad-hoc modification of adding a lower level that specifies information on how the feature will be implemented.

the ideas put on the map, providing prioritization to the map, and providing more details in the map beyond unique identifiers.

Lessons Learned

Familiarity with ones role is equally important during collaboration because of team expectations and role visibility. Collaborative artifacts won't fix problems that stem from lack of skill or performance; it will more so expose it. When the usability engineer did not know what the user wanted, this became more apparent and created frustration in the team during concept mapping. The benefit is that these occurrences are observed earlier and can be addressed before the project progresses too far.

Improved collaboration doesn't imply improved communication, and conversely, improved communication doesn't imply improved collaboration. Collaboration using our concept mapping approach might help with sharing the responsibility for a task, but could decrease the ability to communicate since each collaborator is more invested in the task(s). This could also happen in the other direction if there's collaboration, but unhealthy communication during the collaboration. An example of this from the study is when the participants would collaborate, but felt they were not on the same page. Although they were collaborating, they weren't communicating effectively on the collaboration.

Steps or heuristics for shared knowledge creation are important in addition to steps for artifact use. Steps for how to use and structure the artifact are different than instructions on how to create the content that is managed in the artifact. Our concept mapping approach provides instructions on how to structure the concept map for an agile usability project, but assumes the collaborators are familiar with the methods to create the content. During the study, participants consistently noted the challenges with creating features from scenarios.

Conclusion and Future Work

The contribution of this case study is not only the presentation of our concept mapping approach, but also the lessons learned for use by others in the community when designing methods to facilitate collaboration within agile usability teams. Additional study will seek insight about the scalability of our approach, yield more examples of our approach in practice, and offer more empirical quantitative data for analysis. As next steps, we plan to study the approach in a broader context and refine the approach based on findings.

Acknowledgements

We thank the participants, local nonprofit agency, and colleagues for their support of this work.

References

[1] Barksdale, Jeremy T., Ragan, Eric D., McCrickard, D. Scott. "Easing Team Politics in Agile Usability: A Concept Mapping Approach." Proc. of the 2009 Conference on Agile Software Development (Agile 2009), 2009.

[2] Lee, J. and McCrickard., Towards Extreme(ly) Usable Software: Exploring Tensions Between Usability and Agile Software Development. 2007: IEEE Computer Society

[3] Novak, J. and A. Cañas, The Theory Underlying Concept Maps and How to Construct Them. Florida Institute for Human and Machine Cognition, 2006.